Programmatūras projektējuma apraksts

NVD.VVIS.EVK.PPA.CR

Versija 4.4.6

Izmaiņu vēsture

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Nr. | Izmaiņu raksturojums | Datums | Autors | Versija | Pamatojums |
| 1. | Dokumenta izveidošana | 07.07.2011 | G.Lazdāne | 0.1 | Dokumenta sākotnējā versija |
| 2. | Versija sagatavota nodevumam | 28.10.2011 | A.Dubrovskis  E.Salna  J.Drobiševskis  G.Lazdāne | 1.0 |  |
| 3. | Versija drukāšanai | 02.12.2011 | G.Lazdāne | 1.2 |  |
| 4. | Viss dokuments papildināts | 11.04.2012 | G.Lazdāne  A.Dubrovskis  E.Salna | 2.0 | Izstrāde,  Vispārīgais izvērtējums |
| 5. | Dokuments papildināts ar otrās kārtas projektējumu | 29.10.2012 | R.Komeļevs | 2.05 | Apvienotā PPA versija |
| 6. | Dokuments papildināts ar trešās kārtas projektējumu | 15.02.2013 | R.Komeļevs | 2.1 | Iesniedzāma versija ar 3. Kārtas projektējumu |
| 7. | Dokumenta aktualizācija | 01.08.2013 | A.Dubrovskis | 2.2 | Iesniedzamā aktualizētā versija |
| 8. | Izmaiņas pacienta kartes iegūšanas uzvedībā | 05.08.2014 | A.Dubrovskis | 2.3 | Izmaiņas sistēmas uzvedībā |
| 9. | Izmaiņas saistītas ar monitoringa ieviešanu | 22.10.2014 | A.Dubrovskis | 2.4 | Izmaiņas sistēmas uzvedībā |
| 10. | Jaunā datu paka monitoringa risinājumā | 28.11.2014 | A.Dubrovskis | 2.5 | Izmaiņas sistēmas uzvedībā saistībā ar EVK integrācijas DU 5 |
| 11. | Ārzemnieku un nedrošu personas kodu pacienta kāršu veidošanas funkcionalitāte (6.2.5 6.2.9 6.2.11 6.3.2) | 21.05.2015 | A.Dubrovskis | 4.0 | Izmaiņas sistēmas uzvedība saistībā ar VVIS ieviešanu. |
| 12. | 3 komentāri, caurskatītas 4.0 versijā labotās sadaļas, pārskatīts dokumenta formatējums | 18.06.2015 | V. Rubene | 4.0 | Kvalitātes kontrole |
| 13. | Papildināta pacienta faktiskās adreses funkcionalitāte (6.1.2) | 05.08.2015 | A.Dubrovskis | 4.1 | Izmaiņas pēc pasūtītāja komentāriem. |
| 14. | Labojumu caurskate.  Jāieraksta 4.1 versijā veikto izmaiņu apraksts | 06.08.2015 | V. Rubene | 4.1 | Kvalitātes kontrole |
| 15. | Dokumentā papildinātas 2.3., 5.2.8., 6.1.20-6.1.26,6.2.24-6.2.28.,6.8., 6.12.6 – 6.12.7, 6.13.8-6.13.10 nodaļas. | 08.10.2015 | A.Dubrovskis | 4.2 | Izmaiņas saistībā ar DU nr. 9 (VMNVD 2014/3 ERAF-9). |
| 16. | Dokumentā papildinātas 2.3., 5.2.8., 6.1.20-6.1.26,6.2.24-6.2.28.,6.8., 6.12.6 – 6.12.7, 6.13.8-6.13.10 nodaļas. | 11.12.2015 | A.Dubrovskis | 4.3 | Izmaiņas saistībā ar DU nr. 9 (VMNVD 2014/3 ERAF-9)., NVD komentāri |
| 17. | Dokumentā papildinātas nodaļas: 6.2.6, 6.2.7, 6.1.8, 6.2.4 | 25.07.2016 | A.Dubrovskis | 4.4 | Izmaiņas saistībā ar IP.008 |
| 18. | Dokumentā papildināta nodaļa 6.2.13 | 09.11.2016 | A.Dubrovskis | 4.4.1 | Izmaiņās saistībā ar IP.017 |
| 19. | Dokumentā papildinātas nodaļas 6.12.6, 6.1.6.2 | 14.11.2016 | A.Dubrovskis | 4.4.2 | Izmaiņas saistībā ar IP.021 |
| 20. | Dokumentā papildinātas nodaļas  6.6.4, 6.6.2, 6.15.1, 6.13.12 | 01.03.2017 | A.Dubrovskis | 4.4.3 | Izmaiņas saistībā ar IP.052 |
| 21 | Dokumentā papildinātas nodaļas 6.2.29, 6.1.27 | 15.03.2017 | A.Dubrovskis | 4.4.4 | Izmaiņas saistībā ar IP.053 |
| 22 | Dokumentā papildinātas nodaļas  6.2.30, 6.1.28, 6.1.29 | 23.03.2017 | A.Dubrovskis | 4.4.5 | Izmaiņas saistībā ar IP.065 |
| 23. | Dokumenta apskate | 07.04.2017 | A. Spāģe | 4.4.5 | Kvalitātes kontrole |
| 24. | Papildinātas nodaļas 6.2.13.1, 6.2.7, 6.2.6, 6.120, 6.2.8 | 01.05.2017 | A.Dubrovskis | 4.4.6 | Izmaiņas saistībā ar IP.022 |
| 25. | Dokumenta apskate | 09.05.2017 | A. Spāģe | 4.4.6 | Kvalitātes kontrole |

Dokumenta sadaļu savietojamības tabula

| Sadaļas nosaukums VVIS 2. kārtas dokumenta veidnē NVD.VVIS.PPA.PARAUGS.1.0 | J - Jauns saturs;  M - Mainīts saturs;  N - Nemainīts saturs | Sadaļas nosaukums NVD.VVIS.EVK.PPA.CR4.01 dokumentā |
| --- | --- | --- |
| 1. Definīcijas, apzīmējumi un saīsinājumi | N | 1. Definīcijas, apzīmējumi un saīsinājumi |
| 2. Ievads | N | 2. Ievads |
| 2.1. Nolūks | N | 2.1. Nolūks |
| 2.1.1 Dokumenta mērķis | N | 2.1.1 Dokumenta mērķis |
| 2.1.2 Dokumenta auditorija | N | 2.1.2Dokumenta auditorija |
| 2.2. Darbības sfēra | N | 2.2. Darbības sfēra |
| 2.3. Atsauces | N | 2.3. Atsauces |
| 2.4. Dokumenta pārskats | N | 2.4. Dokumenta pārskats |
| 3. Dekompozīcijas apraksts | N | 3. Dekompozīcijas apraksts |
| 3.1. Moduļu dekompozīcija | N | 3.1. Moduļu dekompozīcija |
| 3.2. Starpmoduļu atkarības | N | 4.1. Starpmoduļu atkarības |
| 4. Detalizētais projektējums |  |  |
| 4.1. Datu modelis | N | 4.3. Datu atkarības  6.15. Datu detalizēts projektējums |
| 4.2. Funkciju detalizētais projektējums | N | 6.2. Pamatdatu un pierakstu pārvaldības moduļa projektējums  6.3. Validācijas moduļa projektējums  6.4. Dokumentu metasistēmas moduļa projektējums  6.5. Autorizācijas moduļa projektējums  6.6. NVD VIS datu aktualizācijas moduļa projektējums  6.7. Personas datu aktualizācijas moduļa projektējums  6.8. VDEĀVK IS datu aktualizācijas moduļa projektējums  6.9. Jaundzimušo reģistra datu aktualizācijas moduļa projektējums  6.11. eDoc Service projektējums  6.12. Administrēšanas modulis  6.13. Fona uzdevumu projektējums |
| 4.3. Datu struktūru detalizētais projektējums | N | 6.1. Datu struktūras |
| 5. Saskarnes apraksts |  |  |
| 5.1. Ārējās saskarnes | N | 5. Ārējo saskarņu projektējums |
| 5.2. Lietotāju saskarnes | N | 6.12. Administrēšanas modulis |
| 5.3. Procesu saskarnes | N | 6.5. Autorizācijas moduļa projektējums  6.6. NVD VIS datu aktualizācijas moduļa projektējums  6.7. Personas datu aktualizācijas moduļa projektējums  6.8. VDEĀVK IS datu aktualizācijas moduļa projektējums  6.9. Jaundzimušo reģistra datu aktualizācijas moduļa projektējums |
| 6. Kļūdu apstrāde | N | 6.10. IP audita un sistēmas žurnāla moduļa projektējums |
| 6.1. Kļūdu kodi | N | 6.14. Kļūdas kodu tabula |
| 7. Prasību trasējamība | J | 9. Prasību trasējamība |
| 8. Pielikumi | N | 10. Pielikumi |

Saturs

1. Definīcijas, apzīmējumi un saīsinājumi 17

2. Ievads 18

2.1. Nolūks 18

2.1.1. Dokumenta mērķis 18

2.1.2. Dokumenta auditorija 18

2.2. Darbības sfēra 19

2.3. Atsauces 19

2.4. Dokumenta pārskats 19

3. Dekompozīcijas apraksts 19

3.1. Moduļu dekompozīcija 20

3.1.1. Autorizācijas modulis 21

3.1.2. Pamatdatu un pierakstu pārvaldība 21

3.1.3. Metadatu pārvaldība 22

3.1.4. IP audita un sistēmas žurnāla bibliotēkas 22

3.1.5. eDoc pārbaudes serviss 22

3.1.6. Fona uzdevumi 22

3.1.7. Dokumentu validācijas modulis 22

3.1.8. Jaundzimušo datu aktualizācija 23

3.1.9. Ģimenes ārstu un EVAK datu aktualizācija 23

3.1.10. Personas datu aktualizācija 24

3.1.11. Administrēšanas modulis 24

3.2. Datu dekompozīcija 24

3.2.1. Personificēto datu bāze 24

3.2.2. Nepersonificēto datu un dokumentu datu bāze 25

3.2.3. Izmaiņu ēnošana 26

4. Atkarību apraksts 27

4.1. Starpmoduļu atkarības 27

4.2. Starpprocesu atkarības 29

4.2.1. Pacienta karte 29

4.2.2. Dokumentu pievienošana 34

4.3. Datu atkarības 38

4.3.1. Personificēto datu bāze 38

4.3.2. Nepersonificēto datu un dokumentu datu bāze 39

5. Ārējo saskarņu projektējums 40

5.1. Integrācijas platforma 40

5.2. Moduļu ārējās saskarnes 42

5.2.1. Pamatdatu un pierakstu pārvaldība 42

5.2.2. Metadatu pārvaldība 47

5.2.3. Autorizācijas modulis 48

5.2.4. Administrēšanas modulis 48

5.2.5. Jaundzimušo datu aktualizācija 49

5.2.6. Ģimenes ārstu un EVAK datu aktualizācija 50

5.2.7. Personas datu aktualizācija no PMLP IR 54

5.2.8. Personas datu aktualizācija no VDEĀVK IS 59

6. Detalizēts projektējums 61

6.1. Datu struktūras 61

6.1.1. Atribūtu grupa UpdateMode 61

6.1.2. Person (Pacienta kartes pamatdati) 62

6.1.3. PersonCardOperations (Personas kartes izveidošana; Sapludināšana) 86

6.1.4. PersonQueryByParameter (Personas datu pieprasījums) 87

6.1.5. CDA dokumenta identifikācija 90

6.1.6. DocumentStructure (CDA dokumenta aploksnes struktūra) 90

6.1.7. DocumentQueryByParameter (Dokumentu pieprasījums) 106

6.1.8. TemplateDocument (Veidnes struktūra) 116

6.1.9. TemplateQueryByParameter (Veidņu pieprasījums) 120

6.1.10. Ziņojumapmaiņas sistēmas koplietošanas datu struktūras 120

6.1.11. History 122

6.1.12. DetailedRole 122

6.1.13. ProfileSettings 123

6.1.14. NotificationConfiguration (Automātisko paziņojumu uzstādījumi) 124

6.1.15. DocNumber (Dokumenta numura iegūšanas sekvence) 126

6.1.16. EVKEpacParameters (EVK e-pakalpojumu parametri) 127

6.1.17. GetClassifierValues (EVK saglabāto klasifikatoru kopiju vērtības) 128

6.1.18. Practitioner (Ārstniecības plāna ārstniecības personu saraksts) 129

6.1.19. ScreeningConfigurationRequest (Skrīninga konfigurāciju saraksta iegūšana) 130

6.1.20. ScreeningConfiguration (Skrīninga konfigurācija) 131

6.1.21. ConsultationRecord (Konsultācijas ieraksts) 137

6.1.22. ConsultationsListQuery (Konsultāciju saraksta pieprasījums) 141

6.1.23. ConsultationsSuggestionsQuery (Konsultācijas meklēšanas pieprasījuma variantu parametri) 143

6.1.24. ConsultationsSuggestionsResponse (Konsultācijas meklēšanas pieprasījuma variantu rezultāts) 143

6.1.25. ConsultationsPractitionersQuery (Konsultācijas ārstu kontaktsaraksta pieprasījums) 144

6.1.26. ConsultationsPractitionersResponse (Konsultācijas ārstu kontaktsaraksta pieprasījuma rezultāts) 144

6.1.27. ClinicalDocumentUnavailableRequest (Dokumenta nepieejamības uzstādīšanas pieprasījums) 145

6.1.28. ClinicalDocumentStatusRequest (Dokumenta statusa pieprasījums) 146

6.1.29. ClinicalDocumentStatusResponse (Dokumenta statusa pieprasījuma atbilde) 147

6.2. Pamatdatu un pierakstu pārvaldības moduļa projektējums 148

6.2.1. addDocument (Pievienot dokumentu – sinhronā daļa) 148

6.2.2. addComments (Pievienot piezīmes) 153

6.2.3. setDocumentStatus (Mainīt dokumenta statusu) 154

6.2.4. setAccessRights (Mainīt aizliegumus) 155

6.2.5. createPatientCard (Izveidot pacienta kartes ierakstu) 156

6.2.6. getDocument (Iegūt dokumentu) 157

6.2.7. getDocumentList (Iegūt dokumentu sarakstu) 159

6.2.8. generateCCD (Izveidot veselības pamatdatu dokumentu) 161

6.2.9. getPatientCard (Iegūt pacienta karti) 165

6.2.10. mergePatientCards (Sapludināt pacienta kartes) 167

6.2.11. setPatientCard (Pievienot un papildināt pacienta kartes datus) 168

6.2.12. processDocument (Pievienot dokumentu –asinhronā daļa) 169

6.2.13. internalTaskProcessing (pamatdatu uzdevumu apstrāde) 178

6.2.14. externalTaskProcessing (ārējo uzdevumu apstrāde) 195

6.2.15. generatePDF (Ģenerēt PDF) 195

6.2.16. setPersonMajority (Uzstādīt personas pilngadības pazīmi) 196

6.2.17. getNotificationConfiguration (Iegūt paziņojumu konfigurāciju) 196

6.2.18. setNotificationConfiguration (Uzstādīt paziņojumu konfigurāciju) 197

6.2.19. getNextDocumentNumber (Iegūt jauna dokumenta numuru) 197

6.2.20. getEVKEpacParameters (Iegūt EVK e-pakalpojumu parametrus) 199

6.2.21. getClassifierValues (Iegūt EVK saglabāto klasifikatoru kopiju vērtības) 199

6.2.22. getProfileSettings (Iegūt personas profila uzstādījumus.) 200

6.2.23. setProfileSettings (Pievienot un mainīt personas profila uzstādījumus.) 201

6.2.24. GetConsultationsList (Iegūt konsultāciju ziņojumu sarakstu) 202

6.2.25. SetConsultation (Izveidot konsultācijas ziņojumu) 203

6.2.26. GetConsultationSearchSuggestions (Iegūt konsultāciju meklēšanai ieteiktās Ārstniecības personas) 204

6.2.27. GetConsultationPractitioners (Atlasīt pacienta ārstniecības personu kontaktu sarakstu) 205

6.2.28. SetConsultationPractitioners (Izveidot pacienta ārstniecības personu kontaktu sarakstu) 206

6.2.29. SetDocumentUnavailable (Padarīt dokumentu nepieejamu) 207

6.2.30. GetDocumentStatus (Iegūt dokumenta statusu) 208

6.3. Validācijas moduļa projektējums 209

6.3.1. validateDocumentModel (Validēt dokumentu pret modeli) 209

6.3.2. validateDocument (Validēt dokumentu) 209

6.4. Dokumentu metasistēmas moduļa projektējums 210

6.4.1. getDocumentTemplateList (Iegūt dokumentu veidņu sarakstu) 210

6.4.2. getDocTemplate (Iegūt dokumenta veidni) 211

6.4.3. setDocTemplate (Pievienot un papildināt dokumenta veidni) 212

6.5. Autorizācijas moduļa projektējums 213

6.5.1. checkRightsSetObjectAccessRights (Pārbauda tiesības uzstādīt aizliegumu) 213

6.5.2. checkDetailedRole (Noteikt detalizēto lomu) 214

6.5.3. checkRigthsAccessPatientCard (Pārbaudīt tiesības piekļūt pacienta kartei) 215

6.5.4. checkUserRightsAccessObject (Pārbaudīt personas tiesību līmeni uz objektu) 217

6.5.5. checkUserRight (Pārbaudīt personas tiesību) 218

6.6. NVD VIS datu aktualizācijas moduļa projektējums 220

6.6.1. getFamilyDoctor (Iegūt ģimenes ārsta datus) 220

6.6.2. setPatientFamilyDoctor (Uzstādīt ģimenes ārstu) 221

6.6.3. getEhicData (Iegūt EVAK kartes datus) 222

6.6.4. setPatientEhicData (Uzstādīt EVAK datus) 223

6.7. Personas datu aktualizācijas moduļa projektējums 224

6.7.1. getPersonData (Iegūt personas datus) 224

6.7.2. getChangesPersonData (Iegūt personas datu izmaiņas) 226

6.7.3. checkPersonDataUpdate (Personas datu aktualizācija) 229

6.8. VDEĀVK IS datu aktualizācijas moduļa projektējums 230

6.8.1. getDisabilityData (Iegūt personas invaliditātes datus) 230

6.8.2. setPatientDisability (Personas invaliditātes datu aktualizācija) 230

6.9. Jaundzimušo reģistra datu aktualizācijas moduļa projektējums 233

6.9.1. setNewBornPersonCodes (Iegūt jaundzimušo personas kodus) 233

6.10. IP audita un sistēmas žurnāla moduļa projektējums 234

6.10.1. Auditācijas pieraksti 234

6.10.2. writeJournal (Žurnalificēšanas pieraksta veidošana) 234

6.11. eDoc Service projektējums 235

6.11.1. CDA dokumenta izgūšana no eDoc faila 235

6.12. Administrēšanas modulis 236

6.12.1. Pievienot/rediģēt dokumenta veidni 236

6.12.2. Iegūt dokumenta veidņu sarakstu 237

6.12.3. Iegūt atļauju sarakstu 238

6.12.4. Pievienot/rediģēt atļaujas 238

6.12.5. Uzstādīt personas pilngadības statusu 240

6.12.6. Uzstādīt personas aizliegumu 240

6.12.7. GetScreeningConfigurationsList (Iegūt skrīninga konfigurāciju sarakstu) 241

6.12.8. SetScreeningConfiguration (Saglabāt skrīninga konfigurāciju) 242

6.13. Fona uzdevumu projektējums 243

6.13.1. Dokumenta apstrādes process 243

6.13.2. Dokumentu ierakstu apstrādes process 243

6.13.3. Pacienta kartiņas aizpilde 244

6.13.4. Pacienta kartiņu sapludināšana 245

6.13.5. Personas datu aktualizācija 246

6.13.6. Jaundzimušo personas kodu aktualizācija 247

6.13.7. Personas pilngadības aktualizācija 247

6.13.8. sb\_BackgroundScreeningRunStart (Skrīninga fona process) 247

6.13.9. ProcessNotificationsSendMessage (Notifikāciju sūtīšana) 248

6.13.10. \_ (EVK notifikācijas) 248

6.13.11. GetMedPersonProfiles (Ārstniecības personu profilu datu atlase) 249

6.13.12. BackgroundLogExport (Eksportēt kļūdu informāciju) 250

6.14. Kļūdas kodu tabula 251

6.15. Datu detalizēts projektējums 251

6.15.1. Datubāze: EVKPers 251

6.15.2. Datubāze: EVK – Nepersonificētā 260

7. Konfigurācijas projektējums 287

7.1. Klasifikatori 287

7.2. Sistēmas elementāro tiesību komplekts 289

7.3. Sistēmas audita ziņojumu nodošana Syslog serverim 292

8. Sākotnējās datu ielādes projektējums 292

8.1. Sākotnējās datu ielādes komponente 292

8.1.1. Datu kopas 292

8.1.2. Konfigurācija 293

8.2. Pacientu reģistri (PREDA) 293

8.2.1. Ar noteiktām slimībām slimojošu pacientu reģistrs par pacientiem, kuriem diagnosticēts cukura diabēts 294

8.2.2. Ar noteiktām slimībām slimojošu pacientu reģistrs par pacientiem, kuriem diagnosticēta onkoloģiska slimība 297

8.2.3. Ar noteiktām slimībām slimojošu pacientu reģistrs par pacientiem, kuriem diagnosticēta multiplā skleroze 299

8.2.4. Ar noteiktām slimībām slimojošu pacientu reģistrs par pacientiem, kuriem diagnosticēta tuberkuloze 301

8.2.5. Ar noteiktām slimībām slimojošu pacientu reģistrs par pacientiem, kuriem diagnosticēta iedzimta anomālija 303

8.2.6. Ar noteiktām slimībām slimojošu pacientu reģistrs par pacientiem, kuriem diagnosticēti psihiski un uzvedības traucējumi 305

8.2.7. Ar noteiktām slimībām slimojošu pacientu reģistrs par narkoloģiskajiem pacientiem 306

8.3. NVD VIS 308

8.4. VDEĀVK IS 310

9. Prasību trasējamība 310

9.1. EVK IS 1.kārtas prasību trasējamība 310

9.2. EVK IS 2.kārtas prasību trasējamība 310

10. Pielikumi 311

10.1. Lietotāju lomas un tiesības 311

10.1.1. Ārstniecības persona 311

10.1.2. Izmeklētājs 311

10.1.3. Pacients – P1 311

10.1.4. Pacients – P2 312

10.1.5. Pacients – P3, P4, P5, P7 312

10.1.6. Administrators 312

Attēlu saraksts

[1. attēls. Sistēmas moduļu diagramma 21](#_Toc482104893)

[2. attēls. Ģimenes ārstu datu apstrāde 23](#_Toc482104894)

[3. attēls. EVAK datu apstrāde 23](#_Toc482104895)

[4. attēls. Personificēto datu bāze 25](#_Toc482104896)

[5. attēls. Nepersonificēto datu bāze 26](#_Toc482104897)

[6. attēls. Starpmoduļu atkarības 28](#_Toc482104898)

[7. attēls. Pacienta kartes izveides process (personas kods) 29](#_Toc482104899)

[8. attēls. Pacienta kartes izveides process (alternatīvā identifikācija) 30](#_Toc482104900)

[9. attēls. Pacienta kartes stāvokļu pārejas diagramma 31](#_Toc482104901)

[10. attēls. Personas datu sinhronizācija ar IP (PMLP IR) 31](#_Toc482104902)

[11. attēls. Personas datu labošana no administratīvās saskarnes 32](#_Toc482104903)

[12. attēls. Personas datu aktualizācija (alternatīvā identifikācija) 33](#_Toc482104904)

[13. attēls. Pacienta karšu sapludināšana 34](#_Toc482104905)

[14. attēls. Dokumenta pievienošanas sinhronā apstrādes daļa 35](#_Toc482104906)

[15. attēls. Dokumenta pievienošanas asinhronā apstrādes daļa 36](#_Toc482104907)

[16. attēls. Dokumenta stāvokļu pārejas diagramma 37](#_Toc482104908)

[17. attēls. Personificēto datu atkarības 38](#_Toc482104909)

[18. attēls. Nepersonificēto datu atkarības 39](#_Toc482104910)

[19. attēls. PRPA\_MT201303UV02\_LV01.Person datu struktūra 65](#_Toc482104911)

[20. attēls. EN datu struktūra 70](#_Toc482104912)

[21. attēls. TEL\_LV01 datu struktūra 71](#_Toc482104913)

[22. attēls. AD\_LV01 datu struktūra 73](#_Toc482104914)

[23. attēls. PRPA\_MT201303UV02\_LV01.ContactParty datu struktūra 76](#_Toc482104915)

[24. attēls. PRPA\_MT201303UV02\_LV01.Delegate datu struktūra 78](#_Toc482104916)

[25. attēls. PRPA\_MT201303UV02\_LV01.CareGiver 82](#_Toc482104917)

[26. attēls. PRPA\_MT201303UV02\_LV01.EHIC 84](#_Toc482104918)

[27. attēls. PRPA\_MT201303UV02\_LV01.Section 85](#_Toc482104919)

[28. attēls. PRPA\_MT201390UV02\_LV01.PersonCardOperations datu struktūra 86](#_Toc482104920)

[29. attēls. PRPA\_MT201307UV02.QueryByParameter datu struktūra 88](#_Toc482104921)

[30. attēls. PRPA\_MT201307UV02.ParameterList datu struktūra 89](#_Toc482104922)

[31. attēls. RCMR\_MT000002UV02\_LV01.ClinicalDocument datu struktūra 92](#_Toc482104923)

[32. attēls. RCMR\_MT000002UV02\_LV01.RecordTarget datu struktūra 99](#_Toc482104924)

[33. attēls. RCMR\_MT000002UV02\_LV01.ConfidentialityRecord datu struktūra 100](#_Toc482104925)

[34. attēls. RCMR\_MT000002UV02\_LV01.author datu struktūra 103](#_Toc482104926)

[35. attēls. RCMR\_MT000002UV02\_LV01.Comment datu struktūra 104](#_Toc482104927)

[36. attēls. RCMR\_MT000003UV01\_LV01.QueryByParameter datu struktūra 107](#_Toc482104928)

[37. attēls. RCMR\_MT000003UV01\_LV01.ClinicalDocument.code datu struktūra 115](#_Toc482104929)

[38. attēls. RCMR\_MT000003UV01\_LV01.ClinicalDocument.type datu struktūra 115](#_Toc482104930)

[39. attēls. RCMR\_MT000003UV01\_LV01.ClinicalDocument.effectiveTime datu struktūra 115](#_Toc482104931)

[40. attēls. RCMR\_MT000003UV01\_LV01.ClinicalDocument.id datu struktūra 116](#_Toc482104932)

[41. attēls. RCMR\_MT000003UV01\_LV01.Patient.id datu struktūra 116](#_Toc482104933)

[42. attēls. RCMR\_MT000003UV01\_LV01.Performer.id datu struktūra 116](#_Toc482104934)

[43. attēls. RCMR\_MT000103UV01\_LV01.TemplateDocument datu struktūra 117](#_Toc482104935)

[44. attēls. MCCI\_MT000200UV01.Acknowledgement datu struktūra 121](#_Toc482104936)

[45. attēls. History datu struktūra 122](#_Toc482104937)

[46. attēls. LVEX\_MT000002UV01.ParameterList datu struktūra 123](#_Toc482104938)

[47. attēls. LVEX\_MT000001UV01.ProfileRecord datu struktūra 124](#_Toc482104939)

[48. attēls. LVEX\_MT000004UV01.NotificationConfiguration datu struktūra 125](#_Toc482104940)

[49. attēls. LVEX\_MT000005UV01.DocNumber datu struktūra 127](#_Toc482104941)

[50. attēls. LVEX\_MT000006UV01.EpacParameterGroup datu struktūra 127](#_Toc482104942)

[51. attēls. LVEX\_MT000008UV01.EVKClassifierList datu struktūra 128](#_Toc482104943)

[52. attēls. PRPA\_MT000400UV01\_LV01.Practitioner datu struktūra 129](#_Toc482104944)

[53. attēls. LVEX\_MT000100UV01.ScreeningConfigurationRequest datu struktūra 130](#_Toc482104945)

[54. attēls. LVEX\_MT000100UV01.ScreeningConfiguration datu struktūra 132](#_Toc482104946)

[55. attēls. LVEX\_MT000100UV01.ScreeningParameter datu struktūra 135](#_Toc482104947)

[56. attēls. PRPA\_MT000500UV01\_LV01.ConsultationRecord datu struktūra 137](#_Toc482104948)

[57. attēls. PRPA\_MT000500UV01\_LV01.ConsultationMessage datu struktūra 139](#_Toc482104949)

[58. attēls. PRPA\_MT000500UV01\_LV01.ConsultationsListQuery datu struktūra 141](#_Toc482104950)

[59. attēls. PRPA\_MT000500UV01\_LV01.ConsultationsSuggestionsQuery datu struktūra 143](#_Toc482104951)

[60. attēls. PRPA\_MT000500UV01\_LV01.ConsultationsSuggestionsResponse datu struktūra 143](#_Toc482104952)

[61. attēls. PRPA\_MT000500UV01\_LV01.ConsultationsPractitionersQuery datu struktūra 144](#_Toc482104953)

[62. attēls. PRPA\_MT000500UV01\_LV01.ConsultationsPractitionersResponse datu struktūra 145](#_Toc482104954)

[63. attēls. RCMR\_MT000002UV02\_LV01.ClinicalDocumentUnavailableRequest datu struktūra 146](#_Toc482104955)

[64. attēls. RCMR\_MT000002UV02\_LV01.ClinicalDocumentStatusRequest datu struktūra 146](#_Toc482104956)

[65. attēls. RCMR\_MT000002UV02\_LV01. ClinicalDocumentStatusResponse datu struktūra 147](#_Toc482104957)

[66. attēls. internalTaskProcessing 188](#_Toc482104958)

[67. attēls. Diagnozes noteikšanas algoritms 189](#_Toc482104959)

[68. attēls. frmSetTemplate formas skice 236](#_Toc482104960)

[69. attēls. frmGetTemplates formas skice 237](#_Toc482104961)

[70. attēls. frmGetDelegations formas skice 238](#_Toc482104962)

[71. attēls. frmSetDelegations formas skice 239](#_Toc482104963)

[72. attēls. frmSetMajority formas skice 240](#_Toc482104964)

[73. attēls. Personas aizliegumu uzstādīšanas formas skice 241](#_Toc482104965)

Tabulu saraksts

[2.1‑1. tabula. Definīcijas, apzīmējumi un saīsinājumi 17](#_Toc482104966)

[2.3‑1. tabula. Saistītie dokumenti 19](#_Toc482104967)

[3.1‑1. tabula. Sistēmas moduļi 20](#_Toc482104968)

[3.2‑1. tabula. Ēnu tabulas 26](#_Toc482104969)

[5.2‑1. tabula. Ieejas datu kopa 49](#_Toc482104970)

[5.2‑2. tabula. Saņemamā datu kopa 50](#_Toc482104971)

[5.2‑3. tabula. Ieejas datu kopa 51](#_Toc482104972)

[5.2‑4. tabula. Izejas datu kopa 51](#_Toc482104973)

[5.2‑5. tabula. Ieejas datu kopa 51](#_Toc482104974)

[5.2‑6. tabula. Izejas datu kopa 52](#_Toc482104975)

[5.2‑7. tabula. Ieejas datu kopa 53](#_Toc482104976)

[5.2‑8. tabula. Ieejas datu kopa 54](#_Toc482104977)

[5.2‑9. tabula. Ieejas datu kopa 55](#_Toc482104978)

[5.2‑10. tabula. Izejas datu kopa 55](#_Toc482104979)

[5.2‑11. tabula. Ieejas datu kopa 55](#_Toc482104980)

[5.2‑12. tabula. Izejas datu kopa 55](#_Toc482104981)

[5.2‑13. tabula. Ieejas datu kopa 56](#_Toc482104982)

[5.2‑14. tabula. Izejas datu kopa 56](#_Toc482104983)

[5.2‑15. tabula. Ieejas datu kopa 56](#_Toc482104984)

[5.2‑16. tabula. Izejas datu kopa 56](#_Toc482104985)

[5.2‑17. tabula. Ieejas datu kopa 58](#_Toc482104986)

[5.2‑18. tabula. Izejas datu kopa 58](#_Toc482104987)

[5.2‑19. tabula. Ieejas datu kopa 58](#_Toc482104988)

[5.2‑20. tabula. Izejas datu kopa 58](#_Toc482104989)

[5.2‑21. tabula. Ieejas datu kopa 59](#_Toc482104990)

[5.2‑22. tabula. Izejas datu kopa 59](#_Toc482104991)

[5.2‑23. tabula. Ieejas datu kopa 59](#_Toc482104992)

[5.2‑24. tabula. Izejas datu kopa 60](#_Toc482104993)

[5.2‑25. tabula. Invaliditātes datu kopa no VDEĀVK IS reģistra 61](#_Toc482104994)

[6.1‑1. tabula. UpdateMode izmantošana 61](#_Toc482104995)

[6.1‑2. tabula. Personas datu struktūra 65](#_Toc482104996)

[6.1‑3. tabula. EN datu struktūra 70](#_Toc482104997)

[6.1‑4. tabula. TEL datu struktūra 71](#_Toc482104998)

[6.1‑5. tabula. ADLV\_01 datu struktūra 73](#_Toc482104999)

[6.1‑6. tabula. ContactParty datu struktūra 76](#_Toc482105000)

[6.1‑7. tabula. Delegate datu struktūra 79](#_Toc482105001)

[6.1‑8. tabula. CareGiver datu struktūra 82](#_Toc482105002)

[6.1‑9. tabula. EHIC datu struktūra 84](#_Toc482105003)

[6.1‑10. tabula. EHIC datu struktūra 85](#_Toc482105004)

[6.1‑11. tabula. PersonCardOperations datu struktūra 87](#_Toc482105005)

[6.1‑12. tabula. QueryByParameter datu struktūra 88](#_Toc482105006)

[6.1‑13. tabula. ParameterList datu struktūra 89](#_Toc482105007)

[6.1‑14. tabula. ClinicalDocuments datu struktūra 93](#_Toc482105008)

[6.1‑15. tabula. Patient datu struktūra 99](#_Toc482105009)

[6.1‑16. tabula. ConfidentialityRecord datu struktūra 100](#_Toc482105010)

[6.1‑17. tabula. Author1 datu struktūra 103](#_Toc482105011)

[6.1‑18. tabula. Comment datu struktūra 104](#_Toc482105012)

[6.1‑19. tabula. QueryByParameter datu struktūra 108](#_Toc482105013)

[6.1‑20. tabula. TemplateDocument datu struktūra 118](#_Toc482105014)

[6.1‑21. tabula. QueryByParameter datu struktūra 120](#_Toc482105015)

[6.1‑22. tabula. Acknowledgment datu struktūra 121](#_Toc482105016)

[6.1‑23. tabula. AcknowledgmentDetail datu struktūra 122](#_Toc482105017)

[6.1‑24. tabula. Elementu apraksts 123](#_Toc482105018)

[6.1‑25. tabula. Elementu apraksts 124](#_Toc482105019)

[6.1‑26. tabula. NotificationConfiguration datu struktūra 126](#_Toc482105020)

[6.1‑27. tabula. LVEX\_MT000005UV01.DocNumber datu struktūra 127](#_Toc482105021)

[6.1‑28. tabula. LVEX\_MT000006UV01.EpacParameterGroup datu struktūra 128](#_Toc482105022)

[6.1‑29. tabula. LVEX\_MT000008UV01.EVKClassifierList datu struktūra 129](#_Toc482105023)

[6.1‑30. tabula. PRPA\_MT000400UV01\_LV01.Practitioner datu struktūra 130](#_Toc482105024)

[6.1‑31. tabula. LVEX\_MT000100UV01.ScreeningConfigurationRequest datu struktūra 131](#_Toc482105025)

[6.1‑32. tabula. LVEX\_MT000100UV01.ScreeningConfiguration datu struktūra 133](#_Toc482105026)

[6.1‑33. tabula. LVEX\_MT000100UV01.ScreeningParameter datu struktūra 135](#_Toc482105027)

[6.1‑34. tabula. PRPA\_MT000500UV01\_LV01.ConsultationRecord datu struktūra 137](#_Toc482105028)

[6.1‑35. tabula. PRPA\_MT000500UV01\_LV01.ConsultationMessage datu struktūra 139](#_Toc482105029)

[6.1‑36. tabula. PRPA\_MT000500UV01\_LV01.ConsultationsListQuery datu struktūra 141](#_Toc482105030)

[6.1‑37. tabula. PRPA\_MT000500UV01\_LV01.ConsultationsSuggestionsQuery datu struktūra 143](#_Toc482105031)

[6.1‑38. tabula. PRPA\_MT000500UV01\_LV01.ConsultationsSuggestionsResponse datu struktūra 144](#_Toc482105032)

[6.1‑39. tabula. PRPA\_MT000500UV01\_LV01. ConsultationsPractitionersQuery datu struktūra 144](#_Toc482105033)

[6.1‑40. tabula. PRPA\_MT000500UV01\_LV01.ConsultationsPractitionersResponse datu struktūra 145](#_Toc482105034)

[6.1‑41. tabula. RCMR\_MT000002UV02\_LV01.ClinicalDocumentUnavailableRequest datu struktūra 146](#_Toc482105035)

[6.1‑42. tabula. RCMR\_MT000002UV02\_LV01.ClinicalDocumentStatusRequest datu struktūra 147](#_Toc482105036)

[6.1‑43. tabula. RCMR\_MT000002UV02\_LV01. ClinicalDocumentStatusResponse datu struktūra 148](#_Toc482105037)

[6.2‑1. tabula. Veselības pamatdatu objektu atbilstības tabula 178](#_Toc482105038)

[6.14‑1. tabula. Kļūdas kodu tabula 251](#_Toc482105039)

[7.1‑1. tabula. Klasifikatori 287](#_Toc482105040)

[7.2‑1. tabula. Tiesību komplekts 289](#_Toc482105041)

[8.2‑1. tabula. Datu atlases XSD shēmas lauku atbilstība 294](#_Toc482105042)

[8.2‑2. tabula. Diagnozes datu kopa no cukura diabēta reģistra 295](#_Toc482105043)

[8.2‑3. tabula. Datu atlases XSD shēmas lauku atbilstība 296](#_Toc482105044)

[8.2‑4. tabula. Medikamenta datu kopa no cukura diabēta reģistra 297](#_Toc482105045)

[8.2‑5. tabula. Datu atlases XSD shēmas lauku atbilstība 298](#_Toc482105046)

[8.2‑6. tabula. Diagnozes datu kopa no onkoloģisko slimību reģistra 299](#_Toc482105047)

[8.2‑7. tabula. Datu atlases XSD shēmas lauku atbilstība 299](#_Toc482105048)

[8.2‑8. tabula. Diagnozes datu kopa no onkoloģisko slimību reģistra 301](#_Toc482105049)

[8.2‑9. tabula. Datu atlases XSD shēmas lauku atbilstība 301](#_Toc482105050)

[8.2‑10. tabula. Diagnozes datu kopa no tuberkulozes slimību reģistra 303](#_Toc482105051)

[8.2‑11. tabula. Datu atlases XSD shēmas lauku atbilstība 303](#_Toc482105052)

[8.2‑12. tabula. Diagnozes datu kopa no iedzimto anomāliju reģistra 304](#_Toc482105053)

[8.2‑13. tabula. Datu atlases XSD shēmas lauku atbilstība 305](#_Toc482105054)

[8.2‑14. tabula. Diagnozes datu kopa no psihisko slimību reģistra 306](#_Toc482105055)

[8.2‑15. tabula. Datu atlases XSD shēmas lauku atbilstība 307](#_Toc482105056)

[8.2‑16. tabula. Diagnozes datu kopa no narkoloģijas reģistra 308](#_Toc482105057)

[9.2‑1. tabula. EVK IS 2.kārtas prasību trasējamības tabula 310](#_Toc482105058)

# Definīcijas, apzīmējumi un saīsinājumi

2.1‑1. tabula. Definīcijas, apzīmējumi un saīsinājumi

| Termins | Skaidrojums |
| --- | --- |
| Abstrakcija | Būtisko aspektu izdalīšana un mazsvarīgu aspektu ignorēšana. |
| Atribūts | Klases objekta informatīvs elements, kas apzīmē kādu īpašību. |
| ATVK | Administratīvi teritoriālās vienības kods |
| BPMN | No angļu valodas [Business Process Modeling Notation] – biznesa procesu modelēšanas notācija |
| CCD | No angļu valodas [Continuity of Care Document] – dokuments aprūpes nepārtrauktībai |
| CDA | No angļu valodas [Clinical Document Architecture] – klīnisko dokumentu arhitektūra |
| EVAK | Eiropas veselības apdrošināšanas karte |
| GMDN | No angļu valodas [Global Medical Device Nomenclature] – globāla medicīnas ierīču nomenklatūra |
| HL7 | No angļu valodas [Health Level 7] – klīnisko dokumentu elektroniskā veida standarts |
| IIS | No angļu valodas [Internet Information Services] - interneta informācijas serviss |
| Klase | Objektu kopa, kam ir kopīgi atribūti un pārvaldības darbības. |
| MDHT | No angļu valodas [Model-Driven Health Tools] Modeļu bāzēti veselības rīki |
| Medicīniska dokumenta sekcija | Medicīniskā dokumenta struktūras daļa, ar noteiktu struktūru un datu kopu, kas satur gan klasificētas vērtības, gan tekstuālu aprakstu. |
| Medicīnisks dokuments | EVK kontekstā medicīniskais dokuments ir elektronisks medicīnisks ieraksts, kas pievienots pacienta kartei. |
| NMPD | Neatliekamās medicīniskās palīdzības dienests |
| NVD VIS | Veselības aprūpes pakalpojumu apmaksas norēķinu sistēma "Vadības informācijas sistēma" |
| Objekts | Vienība, kas satur informāciju un pārvaldības darbības. |
| OID | No angļu valodas [Object Identifier Definition] – objekta identifikatora definīcija |
| OMD | Operatīvais medicīniskais dienests |
| Pacienta karte | EVK kontekstā pacienta karte ir elektroniska datu kopa, kas satur pacienta personas datus, veselības pamatdatus un medicīniskos dokumentus. |
| Pārvaldības darbība | Darbība vai transformācija, ko piemēro klases objektam vai ko izpilda pats objekts. |
| PREDA | Ar noteiktām slimībām slimojošu pacientu reģistrs |
| Projekts | Elektroniskās veselības kartes informācijas sistēmas izstrādes projekts saskaņā ar līgumu VEC\_2010/2/ERAF. |
| RIM | No angļu valodas [Reference Information Model] - atsauces informācijas modelis |
| SBS | SQL Service Broker Services |
| Sistēma | Elektroniskās veselības kartes informācijas sistēma, EVK IS. |
| SSK | Starptautisks slimību katalogs |
| Tīmekļa pakalpes | Vienots veids, kā sistēmām, kas strādā dažādās platformās, rakstītas dažādās valodās u.t.t., savstarpēji sazināties. Tīmekļa pakalpes nodrošina informācijas apmaiņas iespēju, nepārzinot otras puses sistēmas un programmatūru. Tiek izdalīti pakalpes lietotāji un pakalpes sniedzējs. |
| TS | Tehniskā specifikācija |
| VDEĀVK | Veselības un darbspēju ekspertīzes ārstu valsts komisija |
| VNC | Veselības Norēķinu Centrs  Saskaņā ar MK rīkojumu Nr. 436 „Par Veselības norēķinu centra un Veselības ekonomikas centra reorganizāciju”, kopš 01.11.2011. Nacionālais veselības dienests, ir VNC funkciju, pārvaldes uzdevumu, tiesību, saistību, finanšu līdzekļu, mantas un lietvedības pārņēmējs. |
| XML | No angļu valoda [eXtensible Markup Language] – paplašināmā iezīmēšanas valoda |
| XSL | No angļu valodas [eXtensible Stylesheet Language] – paplašināmā stila lapu valoda |
| XSLT | No angļu valodas [XSL Transformations]  XSL valodas transformācija stila lapas formā, ar labi noformētu XML[XML 1.0] sintaksi, kas atbilst XML nosaukumvietas rekomendācijām [Namespaces in XML Recomendations]. |
| Ziņojums | Objektu saziņas līdzeklis. |

# Ievads

## Nolūks

### Dokumenta mērķis

Elektroniskās Veselības Kartes informācijas sistēmas (EVK IS) programmatūras projektējuma apraksta mērķis ir detalizēti atspoguļot izstrādājamās sistēmas struktūru, komponentes, saskarnes un datu modeli. Projektējums nosedz 1. [1], 2. [4] un 3. [5] izstrādes kārtas programmatūras prasību specifikācijās noteiktās sistēmas prasības. Projektējums ir detalizēts pamats projekta izstrādes darbu veikšanai.

### Dokumenta auditorija

Dokuments ir saistošs:

* no Pasūtītāja puses projektā iesaistītajām personām,
* EVK IS izstrādātājiem,
* EVK IS testētājiem,
* kā arī informatīvs materiāls citiem projektā iesaistītajiem dalībniekiem un projekta vadītājiem.

Dokuments izstrādāts, izmantojot standartu IEEE 1016-2009.

## Darbības sfēra

Dokumentā detalizēti aprakstītas sistēmas prasības un lietojumi, kurus jāatbalsta Elektroniskās veselības kartes informācijas sistēmai, un kuras izstrādājamas saskaņā ar projekta izstrādes plānu.

Šis dokuments tiks izmantots kā pamats turpmākajiem projekta darbiem: EVK IS sistēmas projektēšanai, izstrādei un testēšanai.

## Atsauces

2.3‑1. tabula. Saistītie dokumenti

|  |  |  |
| --- | --- | --- |
| Nr. | Identifikators, versija | Dokumenta nosaukums |
|  | VEC.EVK.PPS.CR1.1.0 | Programmatūras prasību specifikācija. Elektroniskās veselības kartes informācijas sistēma. Rīga, 2011 |
|  | VEC. VEC-132-E-VES\_IP-RAVD | Projekta Integrācijas Platformas Informācijas sistēmas izstrāde arhitektūras risinājuma vīzijas dokuments. Rīga, 2011 |
|  | VEC.EVK.PPA.KM1.01 | Klasifikatoru modulis. Programmatūras projektējuma apraksts. Elektroniskās Veselības Kartes Informācijas sistēma. Rīga, 2011 |
|  | NVD.EVK.PPS.CR2.1.0 | Programmatūras prasību specifikācija. 2.izstrādes kārta. Elektroniskās veselības kartes informācijas sistēma. Rīga, 2012 |
|  | NVD.EVK.PPS..CR3.0.1 | Programmatūras prasību specifikācija. 3. izstādes kārta. Elektroniskās veselības kartes informācijas sistēma. Rīga, 2012 |
|  | PMLP\_MONIT\_LRG | PMLP IR monitoringa datu apstrādes procesora lietošanas pamācība. Rīga, 2012 |
|  | NVD.VVIS.VAC.PAK.PPS | Programmatūras prasību specifikācija. Vakcināciju modulis.Rīga, 2015 |

Izmantotie dokumenti pieejami projekta dokumentu bibliotēkā nvdshare.online.lv.

## Dokumenta pārskats

Dokumentā iekļautas šādas nodaļas:

**1. nodaļā** aprakstīti dokumentā lietotie apzīmējumi un saīsinājumi.

**2. nodaļā** aprakstīts dokumenta nolūks, kā arī norādīta saistība ar citiem dokumentiem un materiāliem.

**3. nodaļa** satur moduļu dekompozīcijas un datu dekompozīcijas aprakstus.

**4. nodaļā** aprakstītas starpmoduļu atkarības, starpprocesu atkarības, kā arī datu atkarības.

**5. nodaļā** aprakstīts ārējo saskarņu projektējums.

**6. nodaļā** iekļauts sistēmas detalizētais projektējums.

**7. nodaļā** aprakstīts konfigurācijas projektējums.

**8. nodaļā** aprakstīts sākotnējās datu ielādes projektējums.

**9. nodaļā** iekļauti pielikumi.

# Dekompozīcijas apraksts

Nodaļa satur EVK IS sistēmas dekompozīcijas aprakstu. Dekompozīcijas mērķis ir sadalīt sistēmu projektējamās vienībās.

EVK tiek izdalītas šādas projektējamās vienības:

* Sistēmas moduļi, kas apvieno noteiktu funkciju kopu;
* Datu elementi, kas veido datu bāzes struktūru;
* Ārējās saskarnes;
* Sākotnējo datu ielāde.

Projektējamās vienības tiek aprakstītas vienkāršā valodā, izmantojot komponenšu, dekompozīcijas un datu modelēšanas diagrammas.

Katrai projektējamajai vienībai tiek piešķirts identifikators, kas tiek izmantots tālākai detalizācijai un trasējamības nodrošināšanai gan detalizētajā projektējuma daļā gan programmatūras kodā.

Katrai projektējamajai vienībai tiek identificēti un aprakstīti atribūti:

* Veids (modulis vai datu elements);
* Mērķis;
* Funkcijas;
* Sadalījums.

Atribūti tiek detalizēti pēc nepieciešamības.

## Moduļu dekompozīcija

EVK IS dalīta moduļos pēc funkcionalitātes principa. Tas nozīmē, ka katram modulim ir noteikta funkciju kopa, ko tas nodrošina. Moduļiem ir kopīgi datu objekti, ko tie izmanto, tāpēc būtiski definēt datu atkarības.

Datu objekti

Moduļiem ir gan iekšējās, gan ārējās saskarnes. Iekšējās saskarnes tiek izmantotas starpmoduļu saskarnei. Ārējās saskarnes pieejamas citām sistēmām WebServices metožu veidā. Ārējās saskarnes aprakstītas 5. nodaļā „Ārējo saskarņu projektējums”.

Koda līmenis

Moduļi tiek atdalīti koda līmenī, katram veidojot savu bibliotēku.

Sistēmas moduļi pārskaitīti 3.1‑1 tabulā. To detalizētāks apraksts pieejams apakšnodaļās par katru moduli atsevišķi.

3.1‑1. tabula. Sistēmas moduļi

| # | Kods | Nosaukums | Projektējumā izmantotais nosaukums angļu valodā |
| --- | --- | --- | --- |
| 1 | AUTH | Autorizācijas modulis | Authorization Module |
| 2 | BASE | Pamatdatu un pierakstu pārvaldība | Basic Data and Record Management Module |
| 3 | META | Metadatu pārvaldība | MetaData Management Module |
| 4 | AUDT | IP audita un sistēmas žurnāla bibliotēkas | IP Audit and System Log library Module |
| 5 | SIGN | eDoc pārbaudes serviss | eDoc Inspection Service Module |
| 6 | TASK | Fona uzdevumi | Background Tasks Module |
| 7 | DOCV | Validatori | CDA Document Validation Module |
| 8 | NEWB | Jaundzimušo datu aktualizācija | Newborn Data actualization Module |
| 9 | VNCD | Ģimenes ārstu un EVAK datu aktualizācija | NVD VIS Data actualization Module |
| 10 | PERS | Personas datu aktualizācija | Person Data actualization Module |
| 11 | ADMN | Administrēšanas modulis | Administration Module |
| 12 | CLAS | Klasifikatoru uzturēšanas modulis | Classification Module |
| 13 | VACC | Vakcināciju modulis | Vaccination Module |

Moduļi iedalīti 3 grupās (sk. 1. attēlu):

* Pārvaldības moduļi (ADMN, META)

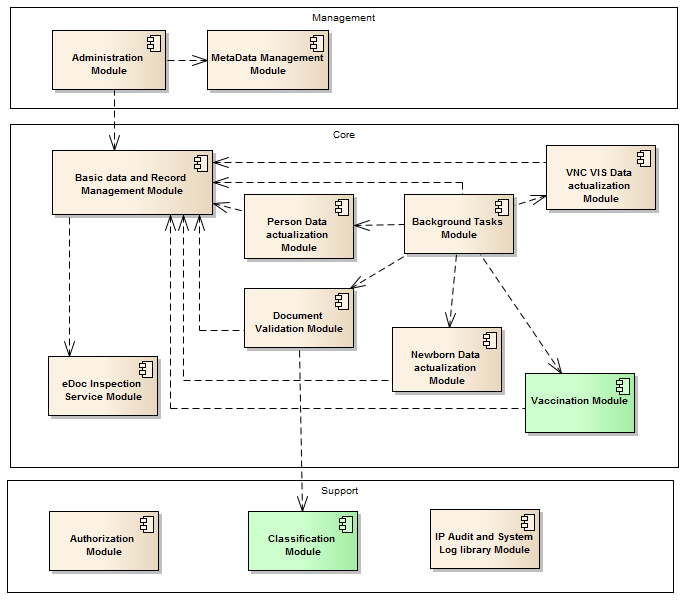
Satur administratīvās funkcijas sistēmas darbības nodrošināšanai.

* Pamat funkcionalitātes moduļi (BASE, SIGN, TASK, DOCV, NEWB, VNCD, PERS, VACC)

Satur sistēmas pamatfunkcionalitāti.

* Atbalsta moduļi (AUTH, CLAS, AUDT)

Satur pārvaldības un pamata funkcionalitātes atbalsta funkcijas.



1. attēls. Sistēmas moduļu diagramma

Attēlā atzīmētas atkarības starp moduļiem.

Atbalsta *[Support]* moduļi tiek izmantoti praktiski visos sistēmas moduļos, tāpēc, lai atvieglotu attēlu, šīs atkarības nav iezīmētas.

Moduļi, kas iekrāsoti zaļā krāsā (Vakcinācija un Klasifikatori) aprakstīti atsevišķos projektējumu dokumentos. Šajā dokumentā minētas tikai saskarnes, kas saistībā ar šiem moduļiem tiek izmantotas iekšēji sistēmā.

### Autorizācijas modulis

Modulis nodrošina EVK IS autorizācijas funkcionalitāti. Iedalāms:

* Funkcionalitāte, kas nodrošina aizliegumu, pilnvaru un atļauju apstrādi,
* Funkcionalitāte, kas nodrošina tiesību pārbaudi.

### Pamatdatu un pierakstu pārvaldība

Modulis nodrošina EVK IS pamata funkcionalitāti, kas saistīta ar pacientu kartēm un medicīniskajiem dokumentiem.

Modulis dalīts divās daļās:

* Pacienta kartes funkcijas, kas nodrošina jauna ieraksta veidošanu, labošanu un anulēšanu tādām datu kopām kā pacienta karte, personas kontaktinformācija, kontaktpersonas.
* Medicīnisko dokumentu apstrādes funkcijas, kas nodrošina dokumentu pievienošanu, validāciju, kā arī datu izgūšanu no sekcijām un anulēšanu.

### Metadatu pārvaldība

Modulis nodrošina administratīvu funkcionalitāti, kas saistīta ar dokumentu metasistēmu – dokumentu tipu, veidņu, modeļu pārvaldību.

### IP audita un sistēmas žurnāla bibliotēkas

Modulis nodrošina audita un sistēmas žurnāla ierakstu veidošanu. Pats audita un sistēmas žurnāls ir IP risinājuma komponente, kuru izmantojot paredzēta piekļūšana šiem datiem.

### eDoc pārbaudes serviss

Modulis nodrošina eDoc elektroniskā paraksta pārbaudi, izmantojot latvija.lv e-pakalpojumu.

### Fona uzdevumi

Fona uzdevumu modulis nodrošina asinhroni izpildāmos darbus. Tie tiek iedalīti:

* Regulāra datu sinhronizācija, kas tiek izpildīta ar noteiktu periodiskumu. Izdalīti šādi uzdevumi:
* personas dati;
* ģimenes ārsta dati;
* jaundzimušo personu kodi;
* EVAK dati;
* pilngadības pazīme;
* invaliditātes dati;
* skrīnings.
* Rindu uzdevumi, kas tiek izpildīti nepārtraukti. Uzdevumus rindās ievieto citi procesi. Rindas tiek realizētas ar SQL ServiceBroker iebūvētu funkcionalitāti. Izdalītās rindas:
* Dokumentu asinhronā apstrāde;
* Personas datu pielasīšana;
* Pacienta karšu sapludināšana.

### Dokumentu validācijas modulis

Modulis nodrošina CDA dokumentu validāciju pret modeli, izmantojot CDA validatora komponenti.

CDA dokumenta modelis tiek veidots izmantojot MDHT (Model-Driven Health Tools). Dokumenta izstrādes rezultāts ir dokumenta XSD shēma, kas nosaka kādi dokumenta elementi un atribūti ir obligāti, kādi var atkārtoties vairākas reizes. Ārpus XSD shēmas tiek norādīts arī kas tiek glabāts CDA dokumenta elementos, piemēram, kādu klasifikatoru drīkst izmantot konkrētais CDA dokumenta elementa aizpildei.

Pievienojot jaunu dokumentu, sinhronajā daļā tiek pārbaudīta tā atbilstība XSD shēmai, tomēr netiek pārbaudīta attiecīgā dokumenta saturiskā atbilstība CDA modelim. MDHT nodrošina java klašu ģenerāciju, kas papildus XSD shēmu validācijai veic arī dokumenta saturisko atbilstību modelim. Dokumenta validācija notiek, veidojot attiecīgās klases instanci un pārbaudot tās atbilstību modelim. Ja tiek konstatēta neatbilstība, tiek izvadīts kļūdu saraksts.

Dokumentu validācijas modulis tiek realizēts kā GlassFish Server Web Serviss, kuram ir dokumenta validācijai paredzēta funkcionalitāte. Web servisam tiek padots CDA XML dokuments. Serviss no saņemtā XML dokumenta veido klases instanci un pārbauda dokumentu atbilstību MDHT rīkā izveidotajam CDA dokumenta modelim, tātad, ieskaitot dokumenta saturisko pārbaudi. Ja dokuments neatbilst modelim, tiek izvadīts kļūdu saraksts. Šis kļūdu saraksts tiek atgriezts izsaucošajai metodei, un tas to tālāk saglabā pie dokumenta kļūdu sarakstā, vienlaicīgi nomainot dokumenta statusu.

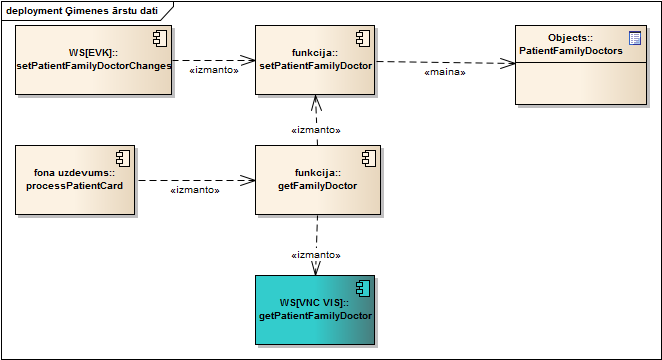
### Jaundzimušo datu aktualizācija

Modulis nodrošina datu apmaiņu ar Jaundzimušo reģistru.

### Ģimenes ārstu un EVAK datu aktualizācija

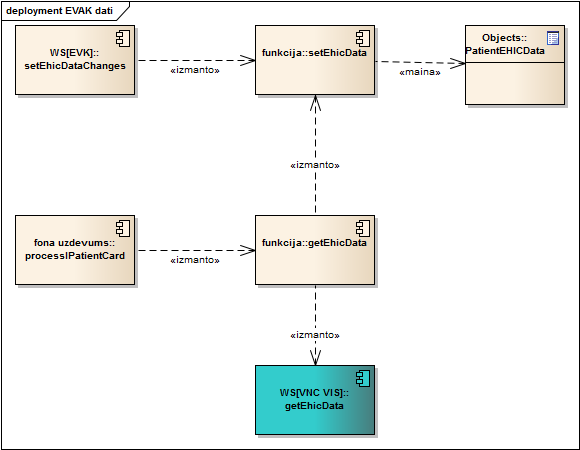
Modulis nodrošina datu apmaiņu ar NVD VIS ģimenes ārstu un EVAK karšu reģistriem.

Ģimenes ārstu datu apstrādes plūsmas redzamas 2. attēlā.



2. attēls. Ģimenes ārstu datu apstrāde

EVAK datu apstrādes plūsmas redzamas 3. attēlā.



3. attēls. EVAK datu apstrāde

### Personas datu aktualizācija

Modulis nodrošina datu apmaiņu ar PMLP IR reģistru.

### Administrēšanas modulis

Modulis nodrošina administratora funkciju lietotāja saskarni administratora darba vietā, kas izvietota uz IP platformas.

## Datu dekompozīcija

Datu dekompozīcijas mērķis ir identificēt datu tabulas, saites un atkarības starp tiem. Datu dekompozīcija veido augsta līmeņa izpratni par EVK IS datu bāzes uzbūvi.

Datu bāzes

EVK IS izdalīti divi datu apgabali, lai datu līmenī atdalītu personificētos datus:

* Personificēto datu bāze;
* Nepersonificēto datu, medicīnisko dokumentu un pārvaldības datu bāze.

Sasaiste starp datu bāzēm nodrošināta ar šifrētu saiti starp personas personificēto un nepersonificēto ierakstu. Saites šifrēšana datu bāzes līmenī tiek realizēta, izmantojot simetriskās atslēgas šifrēšanas algoritmu AES\_256. Atslēgas, kas nepieciešamas, lai iedarbinātu šifrētās saites atšifrēšanas mehānismu, tiek glabātas atdalīti no datu bāzes. Risinājums nodrošina, ka datu bāze pati par sevi nav atšifrējama, jo atslēga ir šifrēta, izmantojot SQL Server instances atslēgu. Šifrēšanas uzstādījumi ir fiksēti: EVK datubāzē, administratīvajā datu bāzē (*master*), SQL servera instancē.

CDA dokumenti un to html tiek šifrēti līdzīgi kā šifrētā saite, izmantojot simetriskās atslēgas šifrēšanas algoritmu AES\_256.

Papildus tam SQL Server lietotājam(iem) ir tiesības piekļūt tikai nepersonificētajai datu bāzei. Piekļuve personificētajai datu bāzei notiek caur ar sertifikātu parakstītām procedūrām, tādējādi tiek nodrošināts, ka lietotājam(iem) nav tiešas piekļuves datu bāzes tabulām. Procedūru izmaiņu gadījumā tā ir atkārtoti jāparaksta.

Tiek šifrēti arī pilnīgi visi dati izmantojot TDE (*Transparent Data Encryption*). TDE šifrēšana nodrošina datu fizisko šifrēšanu – pilnīgi visi dati tiek šifrēti, saglabājot tos uz diska, dati tiek atšifrēti brīdī, kad tos ielasa operatīvajā atmiņā no diska. Tiek šifrēti gan datu bāzes faili, gan log fails, gan arī tempdb dati. Izmantojot TDE tiek nodrošināts, ka datu bāzes visi dati nav lietojami bez atslēgas.

### Personificēto datu bāze

***Tabula [Patients]***  satur personas personificēto datu kopu.

***Tabula [PatientIdentifications]*** satur personas identifikācijas datus.

***Tabula [PatientContacts]*** satur personas kontaktinformāciju.

***Tabula [PatientContactPersons]*** satur personas kontaktpersonu informāciju.

***Tabula [PatientDelegations]*** satur personas pilnvaroto personu informāciju.

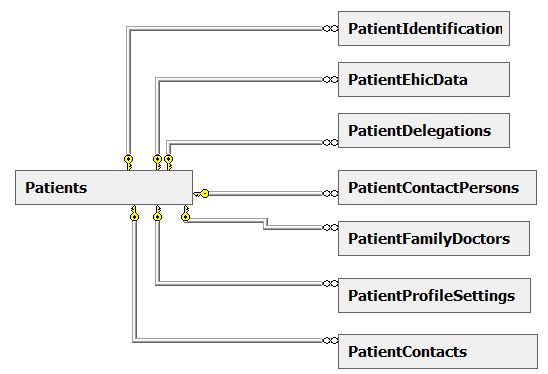
***Tabula [PatientEhicData]*** satur personas EVAK kartes datus.

***Tabula [PatientFamilyDoctors]*** satur personas ģimenes ārsta informāciju.

***Tabula [PatientProfileSettings]*** satur personas profila uzstādījumus.

***Tabula [PatientNotifications]***  satur personas automatizēto paziņojumu uzstādījumus.

***Tabula [UpdateList]*** satur informāciju pacientu datu aktualizācijai.



4. attēls. Personificēto datu bāze

### Nepersonificēto datu un dokumentu datu bāze

***Tabula [NPatients]*** satur personas nepersonificēto datu kopu.

***Tabula [NPatientAllergies]*** satur personas veselības pamatdatus – alerģijas.

***Tabula [NPatientAllergiesLinkToDocuments]*** satur saites starp alerģijām un dokumentiem.

***Tabula [NPatientDiagnosis]*** satur personas veselības pamatdatus – diagnozes.

***Tabula [NPatientDiagnosisLinkToDocuments]*** satur saites starp diagnozēm un dokumentiem.

***Tabula [NPatientDisability]*** satur personas veselības pamatdatus – invaliditāti.

***Tabula [NPatientDisabilityLinkToDocuments]*** satur saites starp invaliditāti un dokumentiem.

***Tabula [NPatientMedicalDevices]*** satur personas veselības pamatdatus – medicīniskās iekārtas.

***Tabula [NPatientMedicalDevicesLinkToDocuments]*** satur saites starp medicīniskajām iekārtām un dokumentiem.

***Tabula [NPatientMedications]*** satur personas veselības pamatdatus – medikamentus.

***Tabula [NPatientMedicationsLinkToDocuments]*** satur saites starp medikamentiem un dokumentiem.

***Tabula [NPatientSurgeries]*** satur veselības pamatdatu informāciju par veiktajām ķirurģiskajām iejaukšanām.

***Tabula [NPatientSurgeriesLinkToDocument]*** satur veselības pamatdatu ķirurģisko iejaukšanās faktu sasaisti ar datu avota dokumentiem.

***Tabula [NPatientTransfusions]*** satur veselības pamatdatu informāciju par veiktajām asins pārliešanām.

***Tabula [NPatientTransfusionsLinkToDocument]*** satur veselības pamatdatu asins pārliešanas faktu sasaisti ar datu avota dokumentiem.

***Tabula [NPatientWarnings]*** satur personas veselības pamatdatus – brīdinājumus.

***Tabula [NPatientWarningsLinkToDocuments]*** satur saites starp brīdinājumiem un dokumentiem.

***Tabula [Documents]*** satur medicīniskos dokumentus – gan no dokumenta izgūtos indeksētos laukus, gan arī pašu dokumenta XML kodu. Dokumenta XML kods datu bāzē tiek glabāts šifrētā veidā.

***Tabula [DocumentErrors]*** satur medicīniskā dokumenta validācijas kļūdas.

***Tabula [DocumentComments]*** satur medicīniskā dokumenta piezīmes.

***Tabula [DocumentSections]*** satur medicīniskā dokumenta sekcijas atdalītā veidā nākotnē paredzētās datu analīzes nodrošināšanai. Sekciju dati nomināli nesatur personificētus datus, tikai medicīniskus.

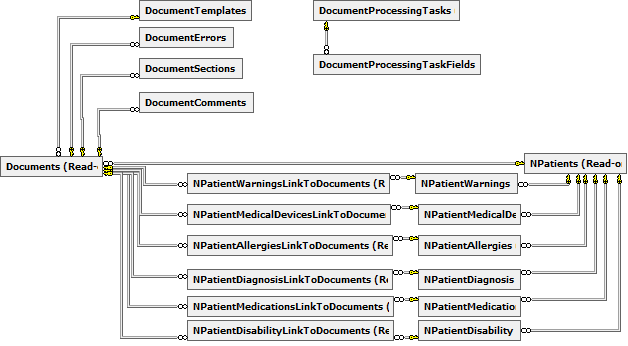
***Tabula [DocumentTemplates]*** satur dokumentu tipu veidnes.

***Tabula [DocumentProcessingTasks]*** satur apstrādes uzdevumu definīcijas.

***Tabula [DocumentProcessingTaskFields]*** satur apstrādes uzdevuma lauku definīcijas.

***Tabula [DocumentNumberSequence]*** satur dokumentu numerācijas datus pa ārstniecības iestādēm.

***Tabula [Parameters]*** satur EVK sistēmas parametrus, tai skaitā e-pakalpojumos izmantojamus parametrus.



5. attēls. Nepersonificēto datu bāze

### Izmaiņu ēnošana

Datu bāzes tabulām, kuras pakļautas *update* vai *delete* darbībām, tiek veidotas ēnu tabulas, kurās tiek fiksēti mainītie ieraksti un to vecās vērtības.

Ēnu tabulu nosaukumi tiek veidoti pēc principa: Pamattabulas nosaukums + *Modif*.

3.2‑1. tabula. Ēnu tabulas

| # | Tabula | Ēnu tabula |
| --- | --- | --- |
| 1 | Patients | PatientsModif |
| 2 | PatientContacts | PatientContactsModif |
| 3 | PatientContactPersons | PatientContactPersonsModif |
| 4 | PatientDelegations | PatientDelegationsModif |
| 5 | PatientEhicData | PatientEhicDataModif |
| 6 | PatientFamilyDoctors | PatientFamilyDoctorsModif |
| 7 | PatientIdentifications | PatientIdentificationsModif |
| 8 | PatientNotifications | PatientNotificationsModif |
| 9 | PatientProfileSettings | PatientProfileSettings |
| 10 | NPatients | NPatientsModif |
| 11 | NPatientAllergies | NPatientAllergiesModif |
| 12 | NPatientAllergiesLinkToDocument | NPatientAllergiesLinkToDocumentModif |
| 13 | NPatientDiagnosis | NPatientDiagnosisModif |
| 14 | NPatientDiagnosisLinkToDocument | NPatientDiagnosisLinkToDocumentModif |
| 15 | NPatientDisability | NPatientDisabilityModif |
| 16 | NPatientDisabilityLinkToDocuments | NPatientDisabilityLinkToDocumentsModif |
| 17 | NPatientMedicalDevices | NPatientMedicalDevicesModif |
| 18 | NPatientMedicalDevicesLinkToDocument | NPatientMedicalDevicesLinkToDocumentModif |
| 19 | NPatientMedications | NPatientMedicationsModif |
| 20 | NPatientMedicationsLinkToDocument | NPatientMedicationsLinkToDocumentModif |
| 21 | NPatientSurgeries | NPatientSurgeriesModif |
| 22 | NPatientSurgeriesLinkToDocument | NPatientSurgeriesLinkToDocumentModif |
| 23 | NPatientTransfusions | NPatientTransfusionsModif |
| 24 | NPatientTransfusionsLinkToDocument | NPatientTransfusionsLinkToDocumentModif |
| 25 | NPatientWarnings | NPatientWarningsModif |
| 26 | NPatientWarningsLinkToDocument | NPatientWarningsLinkToDocumentModif |
| 27 | Documents | DocumentsModif |
| 28 | DocumentErrors | - |
| 29 | DocumentComments | - |
| 30 | DocumentSections | - |
| 31 | DocumentTemplates | - |

# Atkarību apraksts

Nodaļa satur projektējamo vienību atkarību aprakstu.

## Starpmoduļu atkarības

Starpmoduļu atkarību definē ar funkcijām, kuras tiek izmantotas moduļos.

Attēlā parādītas moduļu atkarības no funkciju skatu punkta.



6. attēls. Starpmoduļu atkarības

## Starpprocesu atkarības

Nodaļā apskatīti pacienta kartes izveides un dokumentu pievienošanas procesi.

### Pacienta karte

Pacienta karte tiek izveidota šādos gadījumos:

* karte izveidota līdz ar sākotnējo datu ielādi,
* pēc speciāla pieprasījuma (WS:*createPatientCard*),
* saņemts jauns dokuments, kas noformēts personai, kurai nav izveidota karte, un kuru identificē personas kods,
* saņemts jauns dokuments, kas noformēts personai, kurai nav izveidota karte, un kuru identificē alternatīva identifikācija (jaundzimušo/ārzemnieku/nedrošas identifikācijas gadījums).

Tas rezultējas divos sistēmas scenārijos:

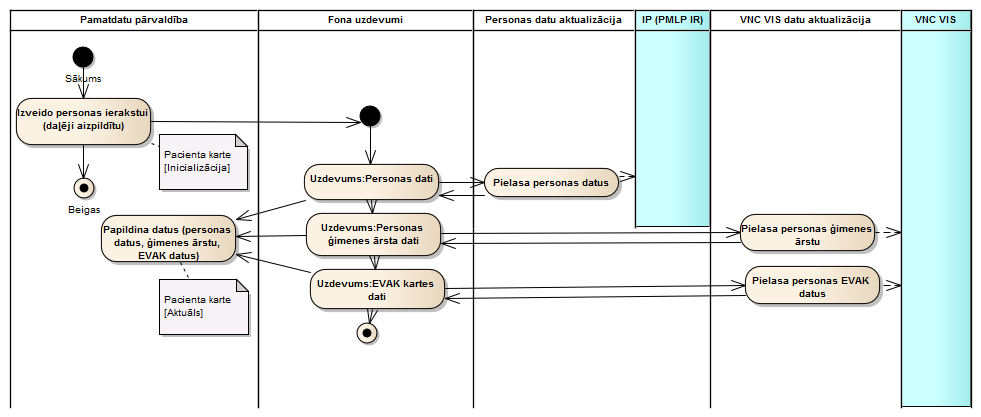
* pacienta kartes izveide ar zināmu personas kodu,
* pacienta kartes izveide ar alternatīvo identifikāciju.

#### Pacienta kartes izveide (personas koda identifikācija)

Sākumstāvoklis: Zināms personas kods (pārbaudīts, eksistējošs)

Pacienta kartes izveidi iniciē:

* Sākotnējo datu ielādes process,
* Pieprasījums izveidot pacienta karti,
* Saņemts jauns medicīniskais dokuments ar personas koda identifikāciju.



7. attēls. Pacienta kartes izveides process (personas kods)

Pacienta kartes izveide sadalīta 2 daļās: sinhronajā un asinhronajā. Sinhronā daļa nodrošina:

* personas ieraksta izveidi ar aizpildītu personas kodu (personas ieraksts ir pacienta kartes bāzes ieraksts),
* izveidotajai pacienta kartei uzstādīts statuss [Inicializācija],
* nosūtīts signāls modulim „Fona uzdevumi” par pacienta kartes aizpildi.

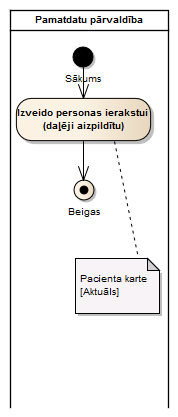
Asinhronā daļa nodrošina datu savākšanu no ārējām sistēmām un to aizpildi pacienta kartē. Pēc asinhronās daļas izpildes kartei tiek uzstādīts statuss [Aktuāls].

#### Pacienta kartes izveide (alternatīvā identifikācija)

Sākumstāvoklis: Zināma personas alternatīvā identifikācija

Pacienta kartes izveidi iniciē:

* Saņemts jauns medicīniskais dokuments bez personas koda identifikācijas (piemēram, par jaundzimušo).



8. attēls. Pacienta kartes izveides process (alternatīvā identifikācija)

Sinhronajā procesa daļā tiek izveidots personas ieraksts ar alternatīvo identifikāciju. Asinhronās daļas nav.

Alternatīvās identifikācijas personas ieraksts sistēmā tiek uzglabāts līdz brīdim, kad tiek veikta personas identificēšana ar personas kodu un/vai notiek ierakstu sapludināšana.

#### Pacientu kartes stāvokļi

Pacienta kartei iespējami šādi stāvokļi:

**Inicializācija** – pirmais stāvoklis pacienta kartei ar personas koda identifikāciju, kuras laikā notiek personas, ģimenes ārsta un EVAK datu aizpildīšana. Šai statusā pacienta karte ir daļēji aizpildīta, tomēr jau iespējams veikt darbības un pievienot medicīnisko dokumentu.

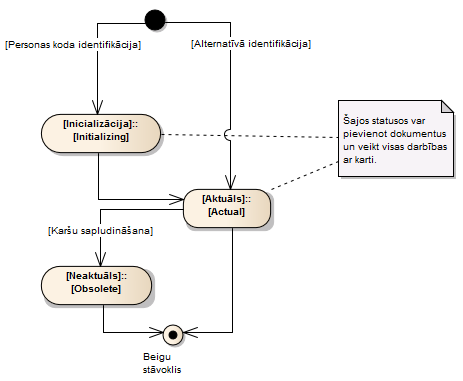
**Aktuāls** – pacienta kartes pamata stāvoklis. Pirmais stāvoklis pacienta kartei ar alternatīvo identifikāciju. Aktuāls statuss saglabājas arī personai pēc nāves, miršanas faktu raksturo personas statuss.

**Neaktuāls** – statuss rodas:

* sapludināšanas rezultātā kartei, kura tiek likvidēta.

Neaktuālās kartes nepiedalās sistēmas procesos, tām nevar pievienot dokumentus un veikt citas darbības.

Stāvokļa pārejas diagramma redzama 9. attēlā.



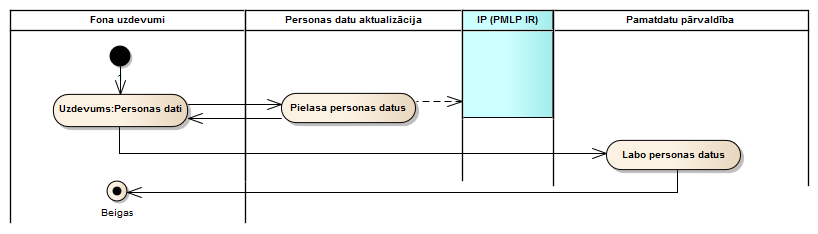
9. attēls. Pacienta kartes stāvokļu pārejas diagramma

#### Personas datu aktualizācija (personas koda identifikācija)

Personas dati tiek aktualizēti/modificēti šādos gadījumos:

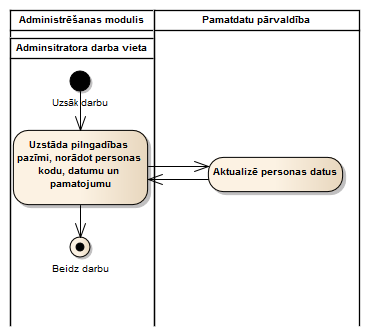
* Periodiski sinhronizējot ar IP (PMLP IR),
* Administratīvais lietotājs papildina/modificē personas pilngadības faktu.

Personas datu periodisku sinhronizāciju nodrošina fona uzdevums, un tajā iesaistīta ārējā saskarne ar IP (PMLP IR).



10. attēls. Personas datu sinhronizācija ar IP (PMLP IR)

Otrs gadījums, kad administratīvais lietotājs uzstāda personas pilngadības faktu pirms 18 gadu sasniegšanas. Tas tiek nodrošināts izmantojot administratīvo darba vietu un pamatdatu pārvaldības moduļa funkcionalitāti.

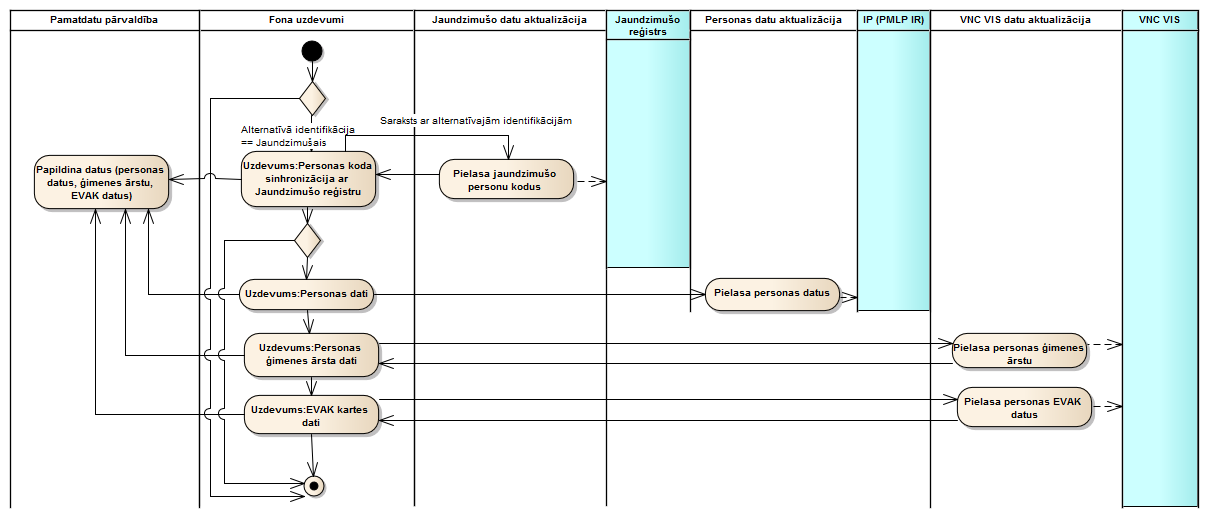


11. attēls. Personas datu labošana no administratīvās saskarnes

#### Personas datu aktualizācija (alternatīvā identifikācija)

Alternatīvās identifikācijas personas dati tiek aktualizēti gadījumā, ja tā ir jaundzimušā identifikācija. Šādi personu dati periodiski tiek sinhronizēti ar Jaundzimušo reģistru ar mērķi identificēt personas kodu, jo personas kodu piesaiste jaundzimušajiem primāri tiek veikta šajā reģistrā.

Kad personas kods identificēts, tiek veikta personas datu aizpilde no IP (PMLP IR), ģimenes ārsta un EVAK datu aizpilde no NVD VIS.

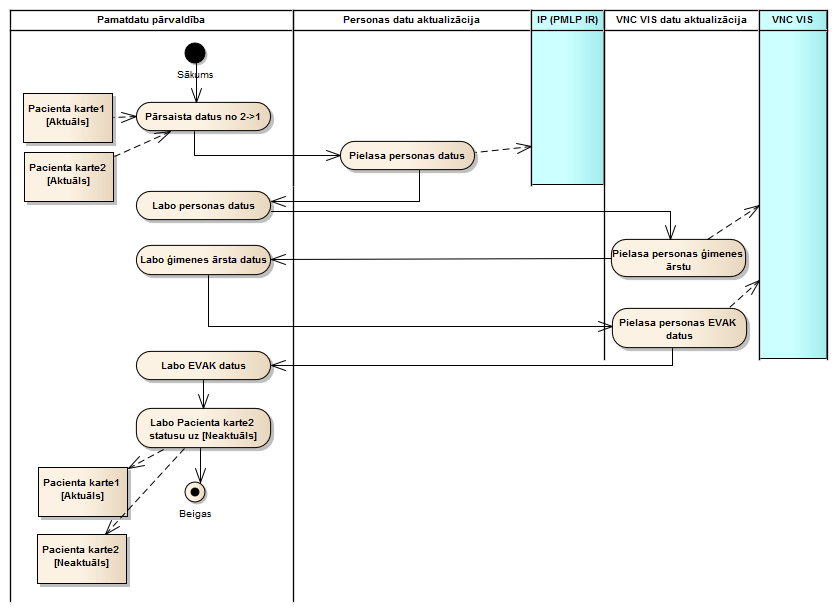


12. attēls. Personas datu aktualizācija (alternatīvā identifikācija)

#### Pacienta karšu sapludināšana

Sākumstāvoklis: Divas aktuālas pacientu kartes

Pacientu karšu sapludināšana ir ārstu iniciēta funkcionalitāte. Procesa laikā notiek ierakstu pārsaiste no deaktualizējamās kartes uz aktuālo, notiek datu aktualizācija un process beidzas ar to, ka vienai pacienta kartei mainīts statuss uz [Neaktuāls].



13. attēls. Pacienta karšu sapludināšana

### Dokumentu pievienošana

Dokumenta pievienošana dalīta sinhronajā un asinhronajā daļā.

Sinhronajā daļā tiek nodrošināts apstrādes minimums, kas nepieciešams, lai dokumentu piesaistītu pacienta kartei. Asinhronajā daļā jau notiek dokumenta detalizēta validācijas pārbaude un datu apstrāde.

Dokumenta pievienošanas laikā tiek izveidota pacienta karte, ja pacientam tā līdz tam nebija izveidota.

#### Sinhronā daļa

Sinhronās apstrādes daļas galvenais uzdevums ir nodrošināt, ka:

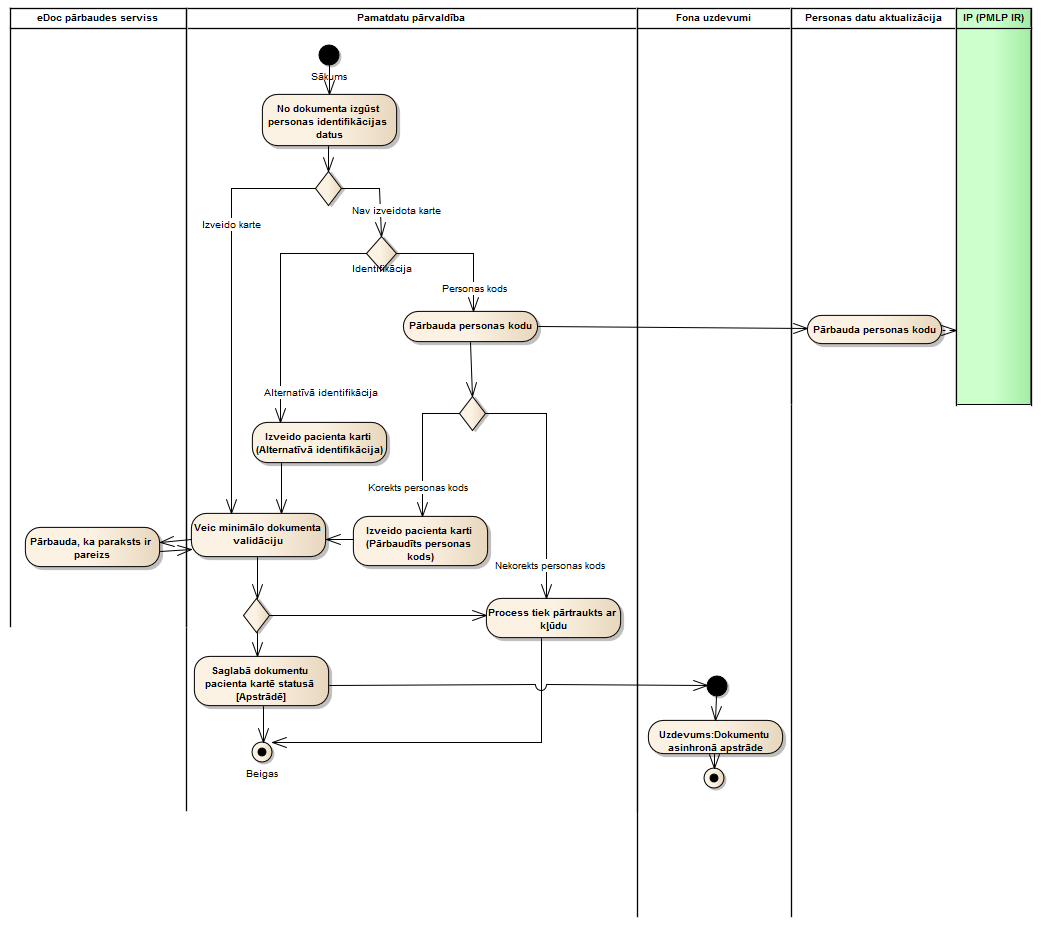
* dokumentā norādītais pacients ir identificējams (ar personas kodu vai alternatīvo identifikāciju),
* dokuments atbilst CDA kopējai shēmai (augsta līmeņa validācija),
* veikta elektroniskā paraksta pārbaude.

Sinhronās apstrādes laikā var tikt izveidota pacienta karte, gan ar personas koda gan alternatīvo identifikāciju, ja tā iepriekš sistēmā neeksistēja.

Veiksmīga sinhronā apstrāde beidzas ar to, ka:

* dokuments pievienots pacienta kartei,
* dokumenta statuss ir [Apstrādē],
* dots uzdevums dokumenta asinhronajai apstrādei.

Neveiksmīga sinhronā apstrāde beidzas ar to, ka dokuments sistēmā netiek saglabāts un tiek atgriezta kļūda.



14. attēls. Dokumenta pievienošanas sinhronā apstrādes daļa

#### Asinhronā daļa

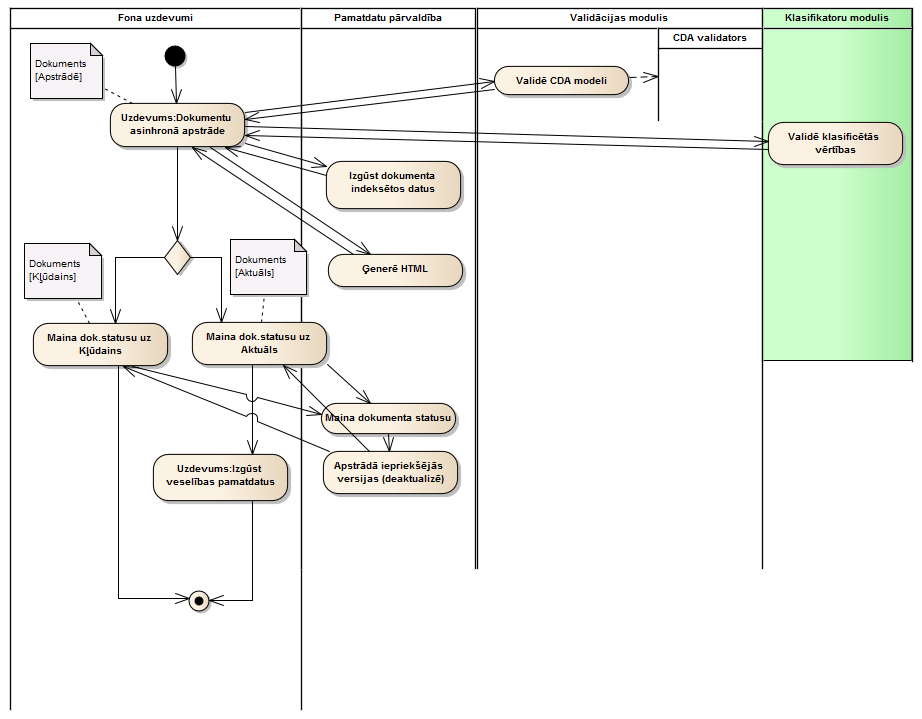
Asinhronās daļas galvenie uzdevumi ir:

* validēt dokumentu pret CDA modeli,
* validēt dokumentā iekļautās klasificētās vērtības,
* izgūt indeksējamos datus,
* ģenerēt HTML failu,
* piemērot sekciju apstrādes uzdevumus, izgūstot no dokumenta sekcijām veselības pamatdatus un vakcinācijas datus.

Veiksmīga asinhronā apstrāde beidzas ar to, ka:

* dokumenta apstrāde pilnībā ir pabeigta un
* dokumenta statuss ir [Aktuāls].

Neveiksmīga asinhronā apstrāde beidzas ar to, ka dokumenta apstrāde ir veikta daļēji un tam mainīts statuss uz [Kļūdains].



15. attēls. Dokumenta pievienošanas asinhronā apstrādes daļa

#### Dokumenta versijas

Jaunas dokumenta versijas tiek pievienotas kā jebkurš dokuments (*addDocument*). Tas, ka iesniegta jauna dokumenta versija, tiek noteikts pēc principa: jaunajam dokumentam ir jau eksistējoša dokumenta *setId* vērtība un lauka *versionNumber* vērtība ir lielāka par esošā dokumenta vērtību.

Iepriekšējo versiju apstrāde notiek asinhronā procesa daļā, veicot statusa maiņu uz Aktuāls vai Kļūdains. Apstrāde nodrošina, ka:

* dokumenta iepriekšējās aktuālās versijas statuss tiek mainīts uz Anulēts;
* veselības pamatdatu ierakstiem, kas veidoti no iepriekšējās aktuālās versijas, statuss tiek mainīts uz Anulēts.

#### Dokumenta stāvokļi

Dokumentam iespējami šādi stāvokļi:

**Apstrādē** – jebkura dokumenta pirmais stāvoklis, kas nozīmē, ka dokuments izgājis sinhronās daļas apstrādi. Šai statusā dokumenta ieraksts datu bāzē ir daļēji aizpildīts.

**Aktuāls** – statuss rodas veiksmīgas asinhronās apstrādes rezultātā. Beigu stāvoklis, ja dokuments netiek anulēts. Pilnvērtīgs dokuments.

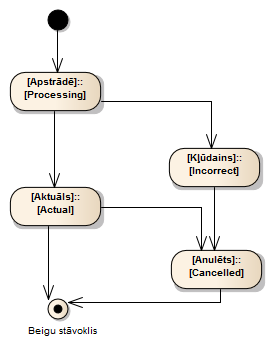
**Kļūdains** – statuss rodas neveiksmīgas asinhronās apstrādes rezultātā. Šāda statusa dokumenti ir pieejami ārstniecības personām, lai sagatavotu jaunu versiju vai tos anulētu. Pacienti tos neredz. Kļūdainam dokumentam pieejams kļūdu saraksts, kas konstatētas validācijas laikā.

**Anulēts** – statuss rodas:

* dokuments tiek anulēts pēc pieprasījuma (*setDocumentStatus*);
* dokumentam tiek pievienota jauna versija (*addDocument*).

Anulēts statuss ir beigu statuss.

Stāvokļa pārejas diagramma redzama 16. attēlā.



16. attēls. Dokumenta stāvokļu pārejas diagramma

## Datu atkarības

### Personificēto datu bāze



17. attēls. Personificēto datu atkarības

### Nepersonificēto datu un dokumentu datu bāze



18. attēls. Nepersonificēto datu atkarības

# Ārējo saskarņu projektējums

Nodaļa satur ārējo koplietošanas saskarņu projektējumu.

## Integrācijas platforma

Sadarbības principi, transporta aploksne, WS kopējās vadlīnijas.

Pārraides iekļaujošā apvalka elementu apraksts

| Elementa nosaukums | Apraksts | Obl | XML piemērs |
| --- | --- | --- | --- |
| templateId | Unikālais ziņojuma šablona identifikators, kur „1.3.6.1.4.1.38760.1.2” ir medicīnisko dokumentu standarti, bet „1.3.6.1.4.1.38760.1.2.1” MDHT pilota medicīniskais ziņojums. | Jā | <templateId root=" 1.3.6.1.4.1.38760.1.2" extension=" 1.3.6.1.4.1.38760.1.2.1"/> |
| Id | Ziņojuma unikālais identifikators sistēmas ietvaros. OID (sakne) un objekta kodam (paplašinājumam) jāizveido unikāla kombinācija. Piemēram, numuri kas veido unikālo ziņojuma identifikatoru:   * 1.3.6.1.4.1.38760.3.4.1 (GUID) un numurs „f190bee0-16d2-4471-b31c-b2d561e5ddc9” * 1.3.6.1.4.1.38760.3.4.2 (IVIS URN) un numurs „URN:IVIS:100001:EP-EP00-v1-6-TR-41159”   A unique identifier for this message. It will never be reused in time, nor will it be used by another sending application. As such, it can serve to reliably allow an acknowledgment message to refer to the message that it acknowledges, and can be used as a bookkeeping index to keep track of messages that have been sent and handled. Note that, even though an ever-increasing numbering scheme is often being used by the sending application, the id of a message can't be used to determine the sequence of messages. | Jā | <Id root="1.3.6.1.4.1.38760.3.4.1" extension= "f190bee0-16d2-4471-b31c-b2d561e5ddc9" />  <Id root="1.3.6.1.4.1.38760.3.4.2" extension= "URN:IVIS:100001:EP-EP00-v1-6-TR-41159" /> |
| securityText | Tiks definēts vēlāk. |  |  |
| creationTime | Laiks, kad ziņojums bija izveidots pretēji laikam, kad notikums bija aprakstīts | Jā | <creationTime value=" 20060421140556.123456780"/> |
| versionCode | Nosaka ziņojuma versiju HL7. Pašlaik tiek izmantota šāda ziņojumu versija:   * V3-2006-NORMATIVE(V3-NE-2006) | Jā | <versionCode code="NE-2006"/> |
| interactionId | Konkrēta informācijas apmaiņas vienuma identifikators. The attribute values are derived from the HL7 MDF interaction names, examples include "POLB\_IN100100" and "COMT\_IN300652". The receiver responsibilities, including application/functional responses, will be determined by the interaction that this identifier represents. Satur Root, kas parasti ir vienāda ar „2.16.840.1.113883” (Health Level Seven, Inc. (HL7)) un paplašinājumu, kas parasti atbilst saknes elementa nosaukumam, piemēram RCMR\_IN000005UV01. | Jā | <interactionId root="2.16.840.1.113883" extension="RCMR\_IN000005UV01"/> |
| processingCode | Nosaka ziņojuma sūtīšanas režīmu. Iespējamie nosūtīšanas ceļi:   * P (produkcijas) – ražošanas vides, ko īsteno dzīvajā (LIVE); * T (apmācība) – testa vidē,demo/ testa pielietojums; * D (atkļūdošana) – izstrādes, atkļūdošanas vidē. | Jā | <processingCode code="P"/> |
| processingModeCode | Nosaka, kādā veidā dati tiek nodoti. Ir trīs veidi – arhīva, oriģinālie un operatīvie dati.  Iespējamie nosūtīšanas veidi ir:   * A (arhīva) – arhīva dati; * I (sākumielāde) – sākotnējā informācija par lasīšanu; * R (atjaunošana no arhīva) – arhīva atjaunošana. * T (normāla apstrāde) – operatīvie dati. | Jā | <processingModeCode code="T"/> |
| acceptAckCode | Nosaka, vai un kādos apstākļos nepieciešams nosūtīt atbildes ziņojumu, par to, ka ziņojums saņemts. Iespējamās vērtības:   * AL (vienmēr) – atbilde vienmēr tiek nosūtīta; * NE (nekad) – atbilde netiek sūtīta; * ER (kļūda, tikai atteikums) – atbildes ziņojums tiek nosūtīts tikai atteikuma gadījumā. | Jā | <acceptAckCode code="AL"/> |
|  | Vienmēr tiek izmantots AL vai ER. |  |  |
| sequenceNumber | Papildus ziņojuma izveidošanas laikam, tiek norādīta arī ziņojumu secība. Numurs ir vesels pozitīvs skaitlis, kas palielinās par vienu, tādējādi tiek sekots ziņojumu izsūtīšanas secībai, kas garantē pārraides integritāti. | Nē | <sequenceNumber value="123456"/> |
| receiver | Norāda informācijas sistēmu saņēmēju, kuram adresēts ziņojums. OID (root) vienmēr ir jābūt „1.3.6.1.4.1.38760.2.3” un paplašinājumam jāizveido unikāla kombinācija, kas atbilst ziņojuma saņēmējam – Informācijas sistēmu klasifikatora vērtībai.  Varētu būt norādīti vairāki saņēmēji.  Apstiprināšanas ziņojumu gadījumā (ACK) par saņēmēju tiek norādīta informācijas sistēma pieprasījuma sūtītājs. | Jā | <receiver typeCode="RCV">  <device classCode="DEV" determinerCode="INSTANCE">  <id root="1.3.6.1.4.1.38760.2.3" extension="12134"/>  </device>  </receiver> |
| sender | Pēc analoģijas ar ziņojumu saņēmēju, tiek norādīts arī ziņojuma informācijas sistēma – nosūtītājs. OID (root) vienmēr ir jābūt „1.3.6.1.4.1.38760.2.3” un paplašinājumam jāizveido unikāla kombinācija, kas atbilst ziņojuma saņēmējam – Informācijas sistēmu klasifikatora vērtībai.  Apstiprināšanas ziņojumu gadījumā (ACK) par sūtītāju tiek norādīta informācijas sistēma – pieprasījuma saņēmējs. | Jā | <sender typeCode="SND">  <device classCode="DEV" determinerCode="INSTANCE">  <id root="1.3.6.1.4.1.38760.2.3" extension="34344"/>  </device>  </receiver> |
| attentionLine | Sadaļa ļauj nosūtīt informāciju, kas nepieciešama, lai pilnībā saprastu ziņojumu. Esošā realizācijā tiek izmantoti šādi papildu atribūti:   * sequenceId – definē E-veselības centrālā sistēmas žurnāla sasaistes identifikators, kur tas „1.3.6.1.4.1.38760.3.4.3”. | Nē | <attentionLine> <keyWordText>sequenceId</keyWordText>  <value xsi:type="II" root="1.3.6.1.4.1.38760.3.4.3" extension= "41819f90-8ed6-4e13-b657-931f29653039"/>  </attentionLine> |

## Moduļu ārējās saskarnes

Moduļu ārējās saskarnes ir tās moduļu funkcijas, kas publiski pieejamas tīmekļa pakalpju veidā, izmantojot IP.

Atsevišķi izdalītas privātās funkcijas, kas pieejamas EVK starpmoduļu sadarbībai un privātās moduļu iekšējās funkcijas, kas nodrošina noteiktu algoritmu apstrādi.

Moduļu ārējās saskarnes iedalāmas tipos:

* EVK eksponēta metode;
* nepieciešama metode no citas informāciju sistēmas.

### Pamatdatu un pierakstu pārvaldība

#### Iegūt pacienta karti

**Tips:** EVK eksponēta metode

**Nosaukums**: ***WS:getPatientCard***

**HL7 ziņojuma tips**: ***PRPA\_MT201307UV01\_LV01***

**Mērķis**: Iegūt pacienta karti ar pamatdatiem. Funkcija atbilstoši lietotāja tiesībām atgriež:

* Personas datus
* Pilnā aizlieguma datus
* Kontaktinformāciju
* Kontaktpersonu datus
* Ģimenes ārstu
* EVAK kartes datus
* Pilnvarojumu datus

Lietotāju grupa: Pacients, Ārstniecības personāls, Izmeklētājs

Funkcijas apraksts: 6.2.9 getPatientCard (Iegūt pacienta karti).

Ieejas datu shēma un apraksts: 6.1.4 PersonQueryByParameter (Personas datu pieprasījums)

Izejas datu shēma un apraksts: 6.1.2 Person (Pacienta kartes pamatdati)

Funkcija atgriež pacienta kartes pamatdatus, ja karte eksistē.

Primārās tiesības: EvkRghtPatientCardGet – Pacienta karte

Saistītās tiesības:

EvkRghtPatientPersonDataGet - Personas dati

EvkRghtPatientContactsGet – Kontaktinformācija

EvkRghtPatientContactPersonsGet – Kontaktpersonu dati

EvkRghtPatientFamilyDoctorGet – Ģimenes ārsta dati

EvkRghtPatientEhicGet – EVAK kartes dati

EvkRghtPatientDelegationsGet – Pilnvarojumu dati (atļauju tips =”Pilnvarojums”)

EvkRghtPatientPermissionsGet – Visu atļauju (t.sk. pilnvarojumi, vecāki, izmeklētāji, u.c.) dati

#### Izveidot pacienta karti

Tips: EVK eksponēta metode

Nosaukums: ***WS:createPatientCard***

HL7 ziņojuma tips: ***PRPA\_MT201307UV01\_LV01***

Mērķis:

1. Izveido pacienta karti personai, kurai līdz šim tā nebija izveidota.

Lietotāju grupa: Pacients, Ārstniecības personāls

Funkcijas apraksts: 6.2.5 createPatientCard (Izveidot pacienta kartes ierakstu)

Tiesības:

EvkRghtPatientCardCreate

Ieejas datu shēma un apraksts: 6.1.3 PersonCardOperations (Personas kartes izveidošana; Sapludināšana).

Izejas datu shēma un apraksts: 6.1.10.1 ack (Tehniskā atbildes datu struktūra).

#### Sapludināt pacienta kartes

Tips: EVK eksponēta metode

Nosaukums: ***WS:mergePatientCards***

HL7 ziņojuma tips: ***PRPA\_MT201307UV01\_LV01***

Mērķis:

1. Sapludina vairākas vienas personas pacienta kartes.

Lietotāju grupa: Ārstniecības personāls

Funkcijas apraksts: 6.2.10 mergePatientCards (Sapludināt pacienta kartes)

Tiesības:

EvkRghtPatientCardsMerge

Ieejas datu shēma un apraksts: 6.1.3 PersonCardOperations (Personas kartes izveidošana; Sapludināšana).

Izejas datu shēma un apraksts: 6.1.10.1 ack (Tehniskā atbildes datu struktūra).

#### Pievienot un papildināt pacienta kartes datus

Tips: EVK eksponēta metode

Nosaukums: ***WS:setPatientCard***

Mērķis: Pacienta kartē pievieno un papildina:

* Kontaktinformāciju
* Kontaktpersonu datus
* Pilnvaras un atļaujas
* Pilngadības pazīmi

Lietotāju grupa: Pacients, Ārstniecības personāls, Administrators

Funkcijas apraksts: 6.2.11 setPatientCard (Pievienot un papildināt pacienta kartes datus)

Ieejas datu shēma un apraksts: 6.1.2 Person (Pacienta kartes pamatdati)

Izejas datu shēma un apraksts: 6.1.2 Person (Pacienta kartes pamatdati)

Funkcija atgriež izmainītu un papildinātu pacienta kartes datu kopu.

Tiesības:

Primārās tiesības: EvkRghtPatientCardSet – Pacienta karte

Saistītās tiesības:

EvkRghtPatientPersonDataSet – Personas dati

EvkRghtPatientMajoritySet – Pilngadības pazīme

EvkRghtPatientContactsSet – Kontaktinformācija

EvkRghtPatientContactPersonSet – Kontaktpersonu dati

EvkRghtPatientDelegationSet – Pilnvarojumu dati (atļauju tips =”Pilnvarojums”)

EvkRghtPatientPermissionSet – Visu atļauju (t.sk. pilnvarojumi, vecāki, izmeklētāji, u.c.) dati

#### Pievienot dokumentu

Nosaukums: ***WS:addDocument***

Tips: EVK eksponēta metode

Mērķis: Pacienta kartē pievieno jaunu medicīnisko dokumentu vai tā versiju, tai skaitā veselības pamatdatus.

Lietotāju grupa: Ārstniecības personāls

Funkcijas apraksts: 6.2.1 addDocument (Pievienot dokumentu – sinhronā daļa)

Ieejas datu shēma un apraksts: 6.1.6 DocumentStructure (CDA dokumenta aploksnes struktūra)

Izejas datu shēma un apraksts: 6.1.10.1 ack (Tehniskā atbildes datu struktūra)

Primārās tiesības: EvkRghtDocumentAdd

Saistītās tiesības:

EvkRghtDocumentCdaAdd – CDA dokuments

EvkRghtDocumentCcdAdd – Veselības pamatdatu dokuments

#### Pievienot piezīmes

Nosaukums: ***WS:addComments***

Tips: EVK eksponēta metode

Mērķis: Medicīniskam dokumentam vai veselības pamatdatu ierakstam (entry) pievieno piezīmi.

Lietotāju grupa: Pacients, Ārstniecības personāls

Funkcijas apraksts: 6.2.2 addComments (Pievienot piezīmes)

Ieejas datu shēma un apraksts: 6.1.6.4 Comment (Komentāra struktūra)

Izejas datu shēma un apraksts: 6.1.10.1 ack (Tehniskā atbildes datu struktūra)

Tiesības:

EvkRghtDocumentCommentAdd – Piezīmes

#### Labot dokumenta statusu

Tips: EVK eksponēta metode

Nosaukums: ***WS:setDocumentStatus***

Mērķis: Maina dokumenta statusu.

Lietotāju grupa: Ārstniecības personāls

Funkcijas apraksts: 6.2.3 setDocumentStatus (Mainīt dokumenta statusu)

Ieejas datu shēma un apraksts: 6.1.6 DocumentStructure (CDA dokumenta aploksnes struktūra)

Izejas datu shēma un apraksts: 6.1.10.1 ack (Tehniskā atbildes datu struktūra)

Tiesības:

EvkRghtDocumentStatusSet –Dokumenta statusa uzstādīšana

#### Uzstādīt un noņemt aizliegumus

Tips: EVK eksponēta metode

Nosaukums: ***WS:setAccessRights***

Mērķis: Maina pacienta kartes, dokumenta (CDA), sekcijas (section) vai ieraksta (entry) piekļuves deskriptoru (aizliegumu) AccessRights.

Lietotāju grupa: Pacients, Ārstniecības personāls

Funkcijas apraksts: Uzstāda un noņem:

* Pacienta kartes pilno aizliegumu
* Aizliegumu dokumentam
* Aizliegumu veselības pamatdatu ierakstam

Skatīt: 6.2.4 setAccessRights (Mainīt aizliegumus)

Ieejas datu shēma un apraksts: 6.1.6.2 ConfidentialityRecord (Aizliegumu struktūra)

Izejas datu shēma un apraksts: 6.1.10.1 ack (Tehniskā atbildes datu struktūra)

Primārās tiesības: EvkRghtAccessRightsSet – Labot aizlieguma datus

Saistītās tiesības:

EvkRghtPatientCardAccessRightsSet – pilnā aizlieguma dati

EvkRghtDocumentAccessRightsSet – dokumentu aizlieguma dati

EvkRghtHealthBasicDataAccessRightsSet – veselības pamatdatu aizlieguma dati

#### Iegūt dokumentu sarakstu

Tips: EVK eksponēta metode

Nosaukums: ***WS:getDocumentsList***

Mērķis: Iegūt dokumentu sarakstu.

Lietotāju grupa: Pacients, Ārstniecības personāls, Izmeklētājs

Funkcijas apraksts: 6.2.7 getDocumentList (Iegūt dokumentu sarakstu)

Ieejas datu shēma un apraksts: 6.1.7 DocumentQueryByParameter (Dokumentu pieprasījums)

Izejas datu shēma un apraksts: 6.1.6 DocumentStructure (CDA dokumenta aploksnes struktūra)

Primārās tiesības: EvkRghtDocumentsGet

Saistītās tiesības:

EvkRghtDocumentsActualGet – Dokumenti statusā „Aktuāls”

EvkRghtDocumentsAllGet – Dokumenti jebkurā statusā

#### Iegūt dokumentu

Tips: EVK eksponēta metode

Nosaukums: ***WS:getDocument***

Mērķis: Iegūt CDA dokumentu vai veselības pamatdatu dokumentu CCD (ContinuesCareDocument).

Lietotāju grupa: Pacients, Ārstniecības personāls, Izmeklētājs

Funkcijas apraksts:

Pēc ieejas datos norādītā parametra *templateId* identificē, vai pieprasīti veselības pamatdati.

* CDA dokuments: 6.2.6 getDocument (Iegūt dokumentu)
* Veselības pamatdati: 6.2.8 generateCCD (Izveidot veselības pamatdatu dokumentu)

Ieejas datu shēma un apraksts: 6.1.7 DocumentQueryByParameter (Dokumentu pieprasījums)

Izejas datu shēma un apraksts: 6.1.6 DocumentStructure (CDA dokumenta aploksnes struktūra)

Primārās tiesības: EvkRghtDocumentsGet

Saistītās tiesības:

EvkRghtDocumentsActualGet – Dokumenti statusā „Aktuāls”

EvkRghtDocumentsAllGet – Dokumenti jebkurā statusā

EvkRghtDocumentCcdGet –Veselības pamatdatu dokuments

EvkRghtDocumentCommentsGet – Piezīmes

#### Iegūt uzstādījumus

**Tips:** EVK eksponēta metode

**Nosaukums:** ***WS:getProfileSettings***

**Mērķis:** Iegūt personas profila uzstādījumus.

**Lietotāju grupa:** Pacients

**Funkcijas apraksts:**

* Pēc ievaddatiem atrod pacienta karti.

Ja karte nav atrasta, tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte).

* Atlasa profila uzstādījumus un atgriež to izejas struktūrā.

**Ieejas datu shēma un apraksts:** 6.1.13.1 LVEX\_MT000002UV01.ParameterList

**Izejas datu shēma un apraksts:** 6.1.13.2 LVEX\_MT000001UV01.ProfileRecord

**Tiesības:** EvkRghtProfileSettings - Profilu uzstādījumu atlase

#### Pievienot un mainīt uzstādījumus

**Tips**: EVK eksponēta metode

**Nosaukums**: ***WS:setProfileSettings***

**Mērķis**: Pievienot un mainīt personas profila uzstādījumus.

**Lietotāju grupa**: Pacients

**Funkcijas apraksts**:

* Pēc ievaddatiem atrod pacienta karti.

Ja karte nav atrasta, tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte).

* Cikliski apstrādā norādītos uzstādījumus:
  + Ja UpdateMode = A [Jauns], uzstādījums tiek pievienots;
  + Ja UpdateMode = AR [Labot], uzstādījums tiek labots;
  + Ja UpdateMode = D [Dzēst], uzstādījums tiek dzēsts.

**Ieejas datu shēma un apraksts**: 6.1.13.2 LVEX\_MT000001UV01.ProfileRecord

**Izejas datu shēma un apraksts**: 6.1.10.1 ack (Tehniskā atbildes datu struktūra)

**Tiesības**: EvkRghtProfileSettings - Profilu uzstādījumu uzstādīšana

#### Iegūt automātisko paziņojumu konfigurāciju

**Tips**: EVK eksponēta metode

**Nosaukums**: ***WS:*getNotificationConfiguration**

**Mijiedarbības tips**:LVEX\_IN000004UV01

**Mērķis**: Iegūt pacienta automātisko paziņojumu konfigurācijas datus.

**Lietotāju grupa**: Ārstniecības personāls, Pacients, Izmeklētājs

**Funkcijas apraksts**: 6.2.17 getNotificationConfiguration (Iegūt paziņojumu konfigurāciju).

**Ieejas datu shēma un apraksts**: 6.1.14 NotificationConfiguration (Automātisko paziņojumu uzstādījumi)

**Izejas datu shēma un apraksts**: 6.1.14 NotificationConfiguration (Automātisko paziņojumu uzstādījumi)

**Primārās tiesības**: EvkRghtNotificationConfigurationGet

#### Uzstādīt automātisko paziņojumu konfigurāciju

**Tips**: EVK eksponēta metode

**Nosaukums**: **WS:setNotificationConfiguration**

**Mijiedarbības tips**:LVEX\_IN000004UV01

**Mērķis**: Uzstādīt pacienta paziņojumu konfigurāciju EVK datu bāzē.

**Lietotāju grupa**: Ārstniecības personāls, Pacients

**Funkcijas apraksts**:. 6.2.18 setNotificationConfiguration

**Ieejas datu shēma un apraksts**: 6.1.14 NotificationConfiguration (Automātisko paziņojumu uzstādījumi)

**Izejas datu shēma un apraksts**: 6.1.14 NotificationConfiguration (Automātisko paziņojumu uzstādījumi)

**Primārās tiesības**: EvkRghtNotificationConfigurationSet

#### Iegūt jauna dokumenta numuru

**Tips**: EVK eksponēta metode

**Nosaukums**: **WS:getNextDocumentNumber**

**Mijiedarbības tips**:LVEX\_IN000005UV01

**Mērķis**: Iegūt jauna dokumenta identifikācijas numuru. Servisu paredzēts izmantot tikai e-pakalpojumu lietojuma gadījumā, reģistrējot jaunu medicīnisku dokumentu. Pārējos jaunu dokumentu veidošanas gadījumos identifikatoru piešķir ārstniecības iestāde savā informācijas sistēmā.

**Lietotāju grupa**: Ārstniecības personāls

**Funkcijas apraksts**: 6.2.19 getNextDocumentNumber

**Ieejas datu shēma un apraksts**: 6.1.15 DocNumber (Dokumenta numura iegūšanas sekvence)

**Izejas datu shēma un apraksts**: 6.1.15 DocNumber (Dokumenta numura iegūšanas sekvence)

**Primārās tiesības**: EvkRghtDocumentNumberGet

#### Validēt dokumentu

**Tips**: EVK eksponēta metode

**Nosaukums**: **WS:validateDocument**

**Mijiedarbības tips**:RCMR\_IN000002UV01\_LV01

**Mērķis**: Pārbaudīt noformēta CDA dokumenta XML koda pareizību pret klasifikatoriem un dokumenta modeli.

**Lietotāju grupa**: Ārstniecības personāls

**Funkcijas apraksts**: 6.3.2 validateDocument (Validēt dokumentu)

**Ieejas datu shēma un apraksts**: RCMR\_MT000002UV02\_LV01.ClinicalDocument

**Izejas datu shēma un apraksts**: 6.1.10.1 ack (Tehniskā atbildes datu struktūra)

**Primārās tiesības**: EvkRghtDocumentValidate

#### Iegūt EVK e-pakalpojumu parametrus

**Tips**: EVK eksponēta metode

**Nosaukums**: **WS:getEVKEpacParameters**

**Mijiedarbības tips**:LVEX\_IN000006UV01

**Mērķis**: Iegūt EVK e-pakalpojumu parametrus.

**Lietotāju grupa**: Iedzīvotāji

**Funkcijas apraksts**: 6.2.20 getEVKEpacParameters (Iegūt EVK e-pakalpojumu parametrus)

**Ieejas datu shēma un apraksts**: 6.1.16 EVKEpacParameters

**Izejas datu shēma un apraksts**: 6.1.16 EVKEpacParameters

**Primārās tiesības**: EvkRghtEpacParametersGet

### Metadatu pārvaldība

#### Pievienot un papildināt dokumenta veidni

Tips: EVK eksponēta metode

Nosaukums: ***WS:setDocumentTemplate***

Mērķis: Pievienot dokumenta veidni veidņu reģistram un papildināt vai mainīt veidnes informāciju.

Lietotāju grupa: Administrators

Funkcijas apraksts: 6.4.3 setDocTemplate (Pievienot un papildināt dokumenta veidni)

Ieejas datu shēma un apraksts: 6.1.8 TemplateDocument (Veidnes struktūra)

Izejas datu shēma un apraksts: 6.1.10.1 ack (Tehniskā atbildes datu struktūra)

Tiesības:

EvkRghtDocumentTemplateSet – Dokumenta veidne

#### Iegūt dokumenta veidnes datus

Tips: EVK eksponēta metode

Nosaukums: ***WS:getDocumentTemplate***

Mērķis: Iegūt dokumenta veidnes datus.

Lietotāju grupa: Administrators

Funkcijas apraksts: 6.4.2 getDocTemplate (Iegūt dokumenta veidni)

Ieejas datu shēma un apraksts: 6.1.9 TemplateQueryByParameter (Veidņu pieprasījums)

Izejas datu shēma un apraksts: 6.1.8 TemplateDocument (Veidnes struktūra)

Tiesības:

EvkRghtDocumentTemplateGet – Dokumenta veidne

#### Iegūt dokumentu veidņu sarakstu

Tips: EVK eksponēta metode

Nosaukums: ***WS:getDocumentTemplateList***

Mērķis: Iegūt dokumentu veidņu sarakstu.

Lietotāju grupa: Administrators

Funkcijas apraksts: 6.4.1 getDocumentTemplateList (Iegūt dokumentu veidņu sarakstu)

Ieejas datu shēma un apraksts: 6.1.9 TemplateQueryByParameter (Veidņu pieprasījums)

Izejas datu shēma un apraksts: 6.1.8 TemplateDocument (Veidnes struktūra)

Tiesības:

EvkRghtDocumentTemplateGet –Dokumenta veidne

#### Iegūt EVK saglabāto klasifikatoru vērtības

**Tips**: EVK eksponēta metode

**Nosaukums**: **WS:getClassifierValues**

**Mijiedarbības tips**:LVEX\_IN000008UV01

**Mērķis**: Iegūt EVK saglabāto klasifikatoru kopiju vērtības.

**Lietotāju grupa**: Iedzīvotāji

**Funkcijas apraksts**: 6.2.21 getClassifierValues (Iegūt EVK saglabāto klasifikatoru kopiju vērtības)

**Ieejas datu shēma un apraksts**: 6.1.17 GetClassifierValues (EVK saglabāto klasifikatoru kopiju vērtības)

**Izejas datu shēma un apraksts**: 6.1.17 GetClassifierValues (EVK saglabāto klasifikatoru kopiju vērtības)

**Primārās tiesības**: EvkRghtClassifiers

### Autorizācijas modulis

#### Noteikt detalizēto lomu

Tips: EVK eksponēta metode

Nosaukums: ***WS:checkDetailedRole***

Mērķis: Iegūt lietotāja detalizēto lomu attiecība uz konkrētu pacientu.

Lietotāju grupa: Pacients

Funkcijas apraksts: 6.5.2 checkDetailedRole (Noteikt detalizēto lomu)

Ieejas datu shēma un apraksts: 6.1.12 DetailedRole

Izejas datu shēma un apraksts: 6.1.12 DetailedRole

Tiesības: EvkRghtCheckDetailedRole

### Administrēšanas modulis

#### Pievienot un labot dokumenta veidni

Tips: Nepieciešama metode (EVK IS Metadatu pārvaldības modulis)

Nosaukums: ***WS:setDocumentTemplate***

Funkcijas apraksts: 6.12.1 Pievienot/rediģēt dokumenta veidni

Ieejas datu shēma un apraksts: 6.1.8 TemplateDocument (Veidnes struktūra)

Izejas datu shēma un apraksts: 6.1.10.1 ack (Tehniskā atbildes datu struktūra)

#### Iegūt dokumenta veidni

Tips: Nepieciešama metode (EVK IS Metadatu pārvaldības modulis)

Nosaukums: ***WS:getDocumentTemplate***

Funkcijas apraksts: 6.12.1 Pievienot/rediģēt dokumenta veidni

Ieejas datu shēma un apraksts: 6.1.9 TemplateQueryByParameter (Veidņu pieprasījums)

Izejas datu shēma un apraksts: 6.1.8 TemplateDocument (Veidnes struktūra)

#### Iegūt dokumenta veidņu datus

Tips: Nepieciešama metode (EVK IS Metadatu pārvaldības modulis)

Nosaukums: ***WS:getDocumentTemplateList***

Funkcijas apraksts: 6.12.2 Iegūt dokumenta veidņu sarakstu

Ieejas datu shēma un apraksts: 6.1.9 TemplateQueryByParameter (Veidņu pieprasījums)

Izejas datu shēma un apraksts: 6.1.8 TemplateDocument (Veidnes struktūra)

#### Pievienot un labot atļauju

Tips: Nepieciešama metode (EVK IS Autorizācijas modulis)

Nosaukums: ***WS:setPatientCard***

Funkcijas apraksts: 6.12.4 Pievienot/rediģēt atļaujas

Ieejas datu shēma un apraksts: 6.1.2 Person (Pacienta kartes pamatdati)

Izejas datu shēma un apraksts: 6.1.2 Person (Pacienta kartes pamatdati)

#### Iegūt atļauju datus

Tips: Nepieciešama metode (EVK IS Autorizācijas modulis)

Nosaukums: ***WS:getPatientCard***

Funkcijas apraksts: 6.12.3 Iegūt atļauju sarakstu

Ieejas datu shēma un apraksts: 6.1.4 PersonQueryByParameter (Personas datu pieprasījums)

Izejas datu shēma un apraksts: 6.1.2 Person (Pacienta kartes pamatdati)

#### Labot personas datus

Tips: EVK eksponēta metode

Nosaukums: ***WS:setPatientCard***

Funkcija: 6.12.5 Uzstādīt personas pilngadības statusu

Ieejas datu shēma un apraksts: 6.1.2 Person (Pacienta kartes pamatdati)

Izejas datu shēma un apraksts: 6.1.2 Person (Pacienta kartes pamatdati)

### Jaundzimušo datu aktualizācija

Tips: Nepieciešama metode (IP, Jaundzimušo reģistrs)

Nosaukums: ***WS:getNewbornPersonCodes***

Apraksts

Metode tiks pievienota esošajam Jaundzimušo reģistra IS WCF (Windows Communication Foundation), ko jāreģistrē IVIS platformā.

Pakalpojums nodrošina bērna personas koda izgūšanu pēc jaundzimušā alternatīvās identifikācijas. Pakalpojums tiks izsaukts periodiski, reizi dienā (izsaukšanas biežums ir konfigurējams Fona uzdevumu modulī), veicot datu pieprasījumu par visiem pacientiem, kas reģistrēti EVK, un kuriem eksistē jaundzimušā alternatīvais identifikators un neeksistē personas koda identifikators.

Ieejas datu shēma un apraksts

5.2‑1. tabula. Ieejas datu kopa

| Nr. | Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- | --- |
| 1 | Newborns | Saraksts | 1..1 | Jaundzimušo saraksts |
| 2 | Newborn | Salikts elements | 1..n | Jaundzimušais |
| 3 | MotherPersonCode | char(11) | 1..1 | Mātes personas kods |
| 4 | BirthDateTime | Datetime | 1..1 | Dzimšanas datums un laiks |

XML piemērs:

<Newborns>

<Newborn>

<MotherPersonCode>01018012345</MotherPersonCode>

<BirthDateTime>2011-10-01-01:10</BirthDateTime>

</Newborn>

<Newborn>

<MotherPersonCode>15027812345</MotherPersonCode>

<BirthDateTime>2011-10-03-14:35</BirthDateTime>

</Newborn>

</Newborns>

Izejas datu shēma un apraksts

5.2‑2. tabula. Saņemamā datu kopa

| Nr. | Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- | --- |
| 1 | Newborns | Saraksts | 1..1 | Jaundzimušo saraksts |
| 2 | Newborn | Salikts elements | 1..n | Jaundzimušais |
| 3 | MotherPersonCode | char(11) | 1..1 | Mātes personas kods (vērtība no ieejas datiem) |
| 4 | BirthDateTime | Datetime | 1..1 | Dzimšanas datums un laiks (vērtība no ieejas datiem) |
| 5 | PersonCode | char(11) | 0..1 | Personas kods |
| 6 | DeathDate | Datetime | 0..1 | Miršanas datums |

XML piemērs:

<Newborns>

<Newborn>

<MotherPersonCode>01018012345</MotherPersonCode>

<BirthDateTime>2011-10-01-01:10</BirthDateTime>

<PersonCode>01101121234</PersonCode>

<DeathDate/>

</Newborn>

<Newborn>

<MotherPersonCode>15027812345</MotherPersonCode>

<BirthDateTime>2011-10-03-14:35</BirthDateTime>

<PersonCode/> <DeathDate/>

</Newborn>

</Newborns>

Apstrādes algoritms Jaundzimušo reģistra pusē

Tiek izpildīta šāda darbu secība:

Tiek atlasīta jaundzimušā karte pēc mātes personas koda (parametrs MotherPersonCode) un bērna dzimšanas datuma (parametram BirthDateTime tiek piemērota apstrādes funkcija, kas atstāj tikai datumu). Ja atlasītas vairāk par vienu karti, tad ir bijušas vairāku augļu dzemdības un precīzā karte jāpielasa pēc dzimšanas laika (parametrs BirthDateTime).

Jaundzimušā kartē pielasa bērna personas koda vērtību un aizpilda atgriežamo PersonCode parametru.

Ja bērna personas koda vērtība ir tukša, tad jaundzimušā kartē pielasa miršanas datumu un aizpilda atgriežamo DeathDate parametru.

### Ģimenes ārstu un EVAK datu aktualizācija

#### Iegūt pacienta ģimenes ārstu no NVD VIS

Tips: Nepieciešams serviss (IP, NVD VIS)

Nosaukums: ***WS:VNC-getPatientFamilyDoctorFromVNC***

Servisa apraksts: Serviss veic viena pacienta ģimenes ārsta datu atlasi uz norādīto datumu no NVD VIS. Serviss tiks izsaukts pie jaunas EVK kartes izveides procesa no funkcijas: 6.6.1 getFamilyDoctor (Iegūt ģimenes ārsta datus).

Ieejas datu shēma un apraksts:

5.2‑3. tabula. Ieejas datu kopa

| Nr. | Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- | --- |
| 1 | PatientPersonCode | char(11) | 1..1 | Personas kods, par kuru tiek pieprasīti dati. |
| 2 | Date | date | 1..1 | Datums, uz kuru dati tiek pieprasīti. |

Lietotāja identifikators, kas veic pieprasījumu, un transakcijas identifikators tiek ievietots integrācijas platformas transporta aploksnē.

Izejas datu shēma un apraksts:

5.2‑4. tabula. Izejas datu kopa

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | PatientFamilyDoctorList | Saraksts | 1..1 | Saraksts ar pacienta ģimenes ārstu attiecību datiem. | |
| 2 | PatientFamilyDoctor | Salikts elements | 1..n | Pacienta un ģimenes ārsta attiecību saliktais elements. Metodes kontekstā, izejas datos ir viens šāds saliktais elements. | |
| 3 | PatientPersonCode | char(11) | 1..1 | Personas kods, par kuru tiek pieprasīti dati. |
| 4 | PhysicianId |  | 1..1 | Ārsta identifikators no ĀP reģistra.  Reģistra OID: 1.3.6.1.4.1.38760.2.1 | |
| 5 | InstitutionId |  | 1..1 | Ārstniecības iestādes identifikators no ĀI reģistra.  Reģistra OID: 1.3.6.1.4.1.38760.2.23 | |
| 6 | SpecialityId |  | 1..1 | Ārsta specialitātes identifikators no klasifikatora.  Klasifikatora OID: 1.3.6.1.4.1.38760.2.38 | |
| 7 | PatientContractDateFrom | Datetime | 1..1 | Datums, no kura ir nodibinātas ģimenes ārsta attiecības ar aprūpējamo. | |
| 8 | PatientContractDateTill | Datetime | 0..1 | Datums, līdz kuram ģimenes ārsta attiecības ar aprūpējamo ir spēkā. | |

#### Iegūt pacienta EVAK datus no NVD VIS

Tips: Nepieciešams serviss (IP, NVD VIS)

Nosaukums: ***WS:VNC-getPatientEhicDataFromVNC***

Servisa apraksts: Funkcija veic viena pacienta EVAK datu atlasi no NVD VIS. Funkcija tiks izsaukta pie jaunas EVK kartes izveides procesa.

Apstrādes funkcija: 6.6.3 getEhicData (Iegūt EVAK kartes datus)

Ieejas datu shēma un apraksts:

5.2‑5. tabula. Ieejas datu kopa

| Nr. | Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- | --- |
| 1 | PatientPersonCode | char(11) | 1..1 | Personas kods, par kuru tiek pieprasīti dati. | |
| 2 | Date | date | 1..1 | Datums, uz kuru dati tiek pieprasīti. |

Lietotāja identifikators, kas veic pieprasījumu, un transakcijas identifikators tiek ievietots integrācijas platformas transporta aploksnē.

Izejas datu shēma un apraksts:

5.2‑6. tabula. Izejas datu kopa

| Nr. | Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- | --- |
| 1 | PatientEVAKList | Saraksts | 1..1 | Saraksts ar pacienta EVAK datiem. |
| 2 | PatientEVAK | Salikts elements | 1..n | Pacienta EVAK datu saliktais elements. Metodes kontekstā, izejas datos ir viens šāds saliktais elements. |
| 3 | PatientPersonCode | char(11) | 1..1 | Personas kods, par kuru tiek pieprasīti dati. | |
| 4 | EHICIdentificationNumber | Varchar(50) | 1..1 | EVAK identifikācijas numurs |
| 5 | EHICIssueDate | datetime | 0..1 | EVAK izsniegšanas datums |
| 6 | EHICDateFrom | datetime | 1..1 | EVAK derīgs no |
| 7 | EHICDateTill | datetime | 1..1 | EVAK derīgs līdz |
| 8 | EHICStatus | varchar | 1..1 | EVAK statuss | |
| 9 | EHICIssuingIdentificationNumber | Varchar(50) | 1..1 | EVAK izdevējiestādes identifikācijas numurs un nosaukums. |
| 10 | EhicType | varchar | 0..1 | Vērtība netiek apstrādāta un netiek saglabāta EVK sistēmā.  Lauks pievienots nākotnes izmaiņu pieprasījuma atvieglošanai. |

#### Uzstādīt ģimenes ārstu izmaiņas

Tips: EVK eksponēta metode

Nosaukums: ***WS:setFamilyDoctor***

Mērķis: Aktualizēt pacienta ģimenes ārsta informāciju EVK pacienta kartē, ja notikušas izmaiņas NVD VIS sistēmā.

Lietotāju grupa: NVD VIS informācijas sistēma

Apstrādes funkcija: 6.6.2 setPatientFamilyDoctor (Uzstādīt ģimenes ārstu).

Saskarnes lietojuma apraksts: Serviss aktualizē pacientu ģimenes ārsta datus, balstoties uz saņemto informāciju. Paredzams, ka serviss tiks izsaukts pēc izmaiņu veikšanas personas ģimenes ārstu datos NVD VIS informācijas sistēmā. Trigera notikumi NVD VIS sistēmā:

* Reģistrācija pie ģimenes ārsta;
* Atreģistrācija no ģimenes ārsta.

Informācija no Valdas Brantas:

Pacienta un reģistrētā ģimenes ārsta attiecības VIS sistēmā regulē 2 notikumi - reģistrācija un atreģistrācija, no kuriem tiek būvētas notikumu virknes, kas apraksta katru konkrētu situāciju – sākotnējo reģistrāciju, pārreģistrāciju, utt.

Jebkuru pacienta un reģistrētā ģimenes ārsta attiecību izmaiņu gadījumā uz EVK tiks sūtīta visa pacienta un ģimenes ārsta (-u) attiecības aprakstošā reģistrācijas un atreģistrācijas notikumu virkne / saraksts.

Piem., ja pacients 15.11.2011 pārreģistrējas vai tiek pārreģistrēts pie cita ģimenes ārsta, tad attiecībā uz „veco” ārstu notiek atreģistrācijas notikums un attiecībā uz „jauno” ārstu notiek reģistrācijas notikums – šādā gadījumā VIS sūtīs uz EVK sarakstu no 2 notikumiem :

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | PatientPersonCode | PhysicianId | InstitutionId | SpecialityId | PatientContractDateFrom | PatientContractDateTill |
| 1 | 051160-12345 | *Vecā ārsta ārstniecības personas ID* | *Vecā ārsta iestādes ID* | *Vecā ārsta specialitātes ID* | 01.01.2005 | 14.11.2011 |
| 2 | 051160-12345 | *Jaunā ārsta ārstniecības personas ID* | *Jaunā ārsta iestādes ID* | *Jaunā ārsta specialitātes ID* | 15.11.2011 | Null |

Ja izmaiņas saņemtas par pacientu, kuram EVK sistēmā nav izveidota pacienta karte, tad tiek atgriezta pozitīva atbilde un izmaiņas EVK datos netiek veiktas.

Ieejas datu shēma un apraksts:

5.2‑7. tabula. Ieejas datu kopa

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | PatientFamilyDoctorList | Saraksts | 1..1 | Saraksts ar pacienta ģimenes ārstu attiecību datiem. | |
| 2 | PatientFamilyDoctor | Salikts elements | 1..n | Pacienta un ģimenes ārsta attiecību saliktais elements. | |
| 3 | PatientPersonCode | char(11) | 1..1 | Personas kods, par kuru tiek pieprasīti dati. |
| 4 | PhysicianId |  | 1..1 | Ārsta identifikators no ĀP reģistra.  Reģistra OID: 1.3.6.1.4.1.38760.2.1 | |
| 5 | InstitutionId |  | 1..1 | Ārstniecības iestādes identifikators no ĀI reģistra.  Reģistra OID: 1.3.6.1.4.1.38760.2.23 | |
| 6 | SpecialityId |  | 1..1 | Ārsta specialitātes identifikators no klasifikatora.  Klasifikatora OID: 1.3.6.1.4.1.38760.2.38 | |
| 7 | PatientContractDateFrom | datetime | 1..1 | Datums, no kura ir nodibinātas ģimenes ārsta attiecības ar aprūpējamo. | |
| 8 | PatientContractDateTill | datetime | 0..1 | Datums, līdz kuram ģimenes ārsta attiecības ar aprūpējamo ir spēkā. | |

Lietotāja identifikators, kas veic pieprasījumu, un transakcijas identifikators tiek ievietots transporta aploksnē.

Izejas datu shēma un apraksts: 6.1.10.1 ack (Tehniskā atbildes datu struktūra)

Tiesības: EvkRghtPatientFamilyDoctorSet

#### Uzstādīt EVAK karšu izmaiņas

Tips: EVK eksponēta metode

Nosaukums: ***WS:setEhicData***

Mērķis: Aktualizēt EVAK datus EVK pacienta kartē, ja notikušas izmaiņas NVD VIS sistēmā.

Lietotāju grupa: NVD VIS informācijas sistēma

Apstrādes funkcija: 6.6.4 setPatientEhicData (Uzstādīt EVAK datus)

Saskarnes lietojuma apraksts: Serviss aktualizē pacientu EVAK datus, balstoties uz saņemto informāciju. Paredzams, ka serviss tiks izsaukts pēc izmaiņu reģistrēšanas EVAK datos NVD VIS informācijas sistēmā. Trigera notikumi NVD VIS sistēmā:

* EVAK kartes izsniegšana

Izsniegšana = notikums, kurš VIS identificē kartes izgatavošanu un izsniegšanu pacientam.

* EVAK kartes ieraksta labojumi

Ieraksta labojumi pēc kartes izsniegšanas.

* Kartes anulēšana

Kartes anulēšana = VNC darbinieks pārtrauc kartes darbību pirms darbības termiņa beigām, EVAK statuss mainās no „spēkā” uz „anulēts”.

Ja izmaiņas saņemtas par pacientu, kuram EVK sistēmā nav izveidota pacienta karte, tad tiek atgriezta pozitīva atbilde un izmaiņas EVK datos netiek veiktas.

Ieejas datu shēma un apraksts:

5.2‑8. tabula. Ieejas datu kopa

| Nr. | Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- | --- |
| 1 | PatientEVAKList | Saraksts | 1..1 | Saraksts ar pacienta EVAK datiem. |
| 2 | PatientEVAK | Salikts elements | 1..n | Pacienta EVAK datu saliktais elements. Metodes kontekstā, izejas datos ir viens šāds saliktais elements. |
| 3 | PatientPersonCode | char(11) | 1..1 | Personas kods, par kuru tiek pieprasīti dati. | |
| 4 | EhicIdentificationNumber | varchar(30) | 1..1 | EVAK identifikācijas numurs |
| 5 | EhicIssueDate | datetime | 0..1 | EVAK izsniegšanas datums |
| 6 | EhicDateFrom | datetime | 1..1 | EVAK derīgs no |
| 7 | EhicDateTill | datetime | 1..1 | EVAK derīgs līdz |
| 8 | EhicStatus | varchar | 1..1 | EVAK status |
| 9 | EHICIssuingIdentificationNumber | varchar(50) | 1..1 | EVAK izdevējiestādes identifikācijas numurs un nosaukums |
| 10 | ChangeDate | datetime | 1..1 | Izmaiņu datums |
| 11 | EhicType | varchar | 1..1 | Vērtība netiek apstrādāta un netiek saglabāta EVK sistēmā.  Lauks pievienots nākotnes izmaiņu pieprasījuma atvieglošanai. |

Lietotāja identifikators, kas veic pieprasījumu, un transakcijas identifikators tiek ievietots integrācijas platformas transporta aploksnē.

Izejas datu shēma un apraksts: 6.1.10.1 ack (Tehniskā atbildes datu struktūra)

Tiesības: EvkRghtPatientEhicSet

### Personas datu aktualizācija no PMLP IR

Modulis „Personas datu aktualizācija” attiecas uz personas datu izgūšanu no iedzīvotāju reģistra (PMLP IR).

#### Pārbaudīt personas kodu

Tips: Nepieciešama metode vai metožu kopa (IP, PMLP IR)

Nosaukums: ***WS:PMLP-GetActualPersonCode***

Funkcijas apraksts: Funkcija pārbauda norādīto personas kodu, vai tas Iedzīvotāju reģistrā ir aktuālais personas kods. Serviss atgriež pārbaudes rezultātu un aktuālo personas kodu.

Funkcija netiek izmantota EVK darbībā

Ieejas datu shēma un apraksts:

5.2‑9. tabula. Ieejas datu kopa

| Nr. | Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- | --- |
| 1 | SessionId | string | 1..1 | Sesijas identifikators |
| 2 | PersonCode | string | 1..1 | Personas kods. |

Lietotāja identifikators, kas veic pieprasījumu, un transakcijas identifikators tiek ievietots integrācijas platformas transporta aploksnē.

Izejas datu shēma un apraksts:

5.2‑10. tabula. Izejas datu kopa

| Nr. | Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- | --- |
| 1 | SessionID | string | 1..1 | Sesijas identifikators |
| 2 | Error: Code, Descrioption, Severity | Salikts elements | 1..1 | Ja metodes izpildes laikā rodas kļūda, tad kļūdas informācija tiek aizpildīta kļūdas struktūrā. |

#### Iegūt personas datus

Tips: Nepieciešama metode vai metožu kopa (IP, PMLP IR)

Nosaukums: ***WS:PMLP-GetPersonName***

Funkcijas apraksts: Funkcija iegūst personas vārdu un uzvārdu.

Ieejas datu shēma un apraksts:

5.2‑11. tabula. Ieejas datu kopa

| Nr. | Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- | --- |
| 1 | SessionId | string | 1..1 | Sesijas identifikators |
| 2 | PersonCode | string | 1..1 | Personas kods. |

Lietotāja identifikators, kas veic pieprasījumu, un transakcijas identifikators tiek ievietots integrācijas platformas transporta aploksnē.

Izejas datu shēma un apraksts:

5.2‑12. tabula. Izejas datu kopa

| Nr. | Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- | --- |
| 1 | SessionId | string | 1..1 | Sesijas identifikators |
| 2 | Error: Code, Descrioption, Severity | Salikts elements | 1..1 | Ja metodes izpildes laikā rodas kļūda, tad kļūdas informācija tiek aizpildīta kļūdas struktūrā. |
| 3 | PersonFullName:  LastName  FirstName  OtherNames | Salikts elements  String  String  string | 1..1 | Personas uzvārds, vārds un citi vārdi |

Nosaukums: ***WS:PMLP-GetBirthDate***

Funkcijas apraksts: Funkcija iegūst personas dzimšanas datumu.

Ieejas datu shēma un apraksts:

5.2‑13. tabula. Ieejas datu kopa

| Nr. | Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- | --- |
| 1 | SessionId | string | 1..1 | Sesijas identifikators |
| 2 | PersonCode | string | 1..1 | Personas kods. |

Lietotāja identifikators, kas veic pieprasījumu, un transakcijas identifikators tiek ievietots integrācijas platformas transporta aploksnē.

Izejas datu shēma un apraksts:

5.2‑14. tabula. Izejas datu kopa

| Nr. | Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- | --- |
| 1 | SessionID | string | 1..1 | Sesijas identifikators |
| 2 | Error: Code, Descrioption, Severity | Salikts elements | 1..1 | Ja metodes izpildes laikā rodas kļūda, tad kļūdas informācija tiek aizpildīta kļūdas struktūrā. |
| 3 | BirthDateAndTime:  PersonBirthDate | Salikts elements DateTime | 1..1 | Personas dzimšanas datums. |

Nosaukums: ***WS:PMLP-GetPersonLivingPlace***

Funkcijas apraksts: Funkcija pēc personas koda iegūst tās deklarēto dzīvesvietas adresi.

Ieejas datu shēma un apraksts:

5.2‑15. tabula. Ieejas datu kopa

| Nr. | Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- | --- |
| 1 | SessionId | string | 1..1 | Sesijas identifikators |
| 2 | PersonCode | string | 1..1 | Personas kods. |

Lietotāja identifikators, kas veic pieprasījumu, un transakcijas identifikators tiek ievietots integrācijas platformas transporta aploksnē.

Izejas datu shēma un apraksts:

5.2‑16. tabula. Izejas datu kopa

| Nr. | Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- | --- |
| 1 | SessionId | string | 1..1 | Sesijas identifikators |
| 2 | Error: Code, Descrioption, Severity | Salikts elements | 1..1 | Ja metodes izpildes laikā rodas kļūda, tad kļūdas informācija tiek aizpildīta kļūdas struktūrā. |
| 3 | PostIndex | Varchar(6) | 1..1 | Latvijas adreses struktūra: pasta kods (mask: LV\d\d\d\d) |
| 4 | ARCode | Varchar(9) | 1..1 | Latvijas adreses struktūra: Adrešu reģistra objekta kods (deviņi cipari) |
| 5 | ATUCodeLevel1 | Varchar(7) | 1..1 | Latvijas adreses struktūra: Pirmā līmeņa administratīvi teritoriālas vienības kods (apriņķis) |
| 6 | ATUNameLevel1 | Nvarchar(128) | 1..1 | Latvijas adreses struktūra: Pirmā līmeņa administratīvi teritoriālas vienības nosaukums (apriņķis) |
| 7 | ATUCodeLevel2 | Varchar(7) | 1..1 | Latvijas adreses struktūra: Otrā līmeņa administratīvi teritoriālas vienības kods (rajons / republikas nozīmes pilsēta) |
| 8 | ATUNameLevel2 | Nvarchar(128) | 1..1 | Latvijas adreses struktūra: Otrā līmeņa administratīvi teritoriālas vienības nosaukums (rajons / republikas nozīmes pilsēta) |
| 9 | ATUCodeLevel3 | Varchar(7) | 1..1 | Latvijas adreses struktūra: Trešā līmeņa administratīvi teritoriālas vienības kods (novads / pilsēta ar lauku teritoriju / pilsēta / pagasts) |
| 10 | ATUNameLevel3 | Nvarchar(128) | 1..1 | Latvijas adreses struktūra: Trešā līmeņa administratīvi teritoriālas vienības nosaukums (novads / pilsēta ar lauku teritoriju / pilsēta / pagasts) |
| 11 | ATUCodeLevel4 | Varchar(7) | 1..1 | Latvijas adreses struktūra: Ceturtā līmeņa administratīvi teritoriālas vienības kods (pilsēta / lauku teritorija / pagasts) |
| 12 | ATUNameLevel4 | Nvarchar(128) | 1..1 | Latvijas adreses struktūra: Ceturtā līmeņa administratīvi teritoriālas vienības nosaukums (pilsēta / lauku teritorija / pagasts) |
| 13 | VillageCode | Varchar(8) | 1..1 | Latvijas adreses struktūra: Ciema kods |
| 14 | VillageName | Nvarchar(128) | 1..1 | Latvijas adreses struktūra: Ciema nosaukums |
| 15 | StreetCode | Varchar(13) | 1..1 | Latvijas adreses struktūra: Ielas kods |
| 16 | StreetName | Nvarchar(128) | 1..1 | Latvijas adreses struktūra: Ielas nosaukums |
| 17 | HomeTitle | Nvarchar(150) | 1..1 | Latvijas adreses struktūra: Mājas nosaukums |
| 18 | HouseNo | int | 1..1 | Latvijas adreses struktūra: Ēkas numurs |
| 19 | HouseNoText | Nvarchar(16) | 1..1 | Latvijas adreses struktūra: Ēkas numura teksta daļa |
| 20 | BlockNo | Nvarchar(8) | 1..1 | Latvijas adreses struktūra: Telpu grupas numurs |
| 21 | ApartmentNo | int | 1..1 | Latvijas adreses struktūra: Dzīvokļa numurs |
| 22 | ApartmentNoText | Nvarchar(16) | 1..1 | Latvijas adreses struktūra: Dzīvokļa numura teksta daļa |
| 23 | Country: Code | Varchar(3) | 1..1 | Ārzemju adreses struktūra: Valsts kods |
| 24 | Country: Name | Nvarchar | 1..1 | Ārzemju adreses struktūra: Valsts nosaukums |
| 25 | AddressAbroad | Nvarchar | 1..1 | Ārzemju adreses struktūra: Adrese ārzemēs |

Nosaukums: ***WS:PMLP-GetPersonSex***

Funkcijas apraksts: Funkcija pēc personas koda iegūst tās dzimumu.

Ieejas datu shēma un apraksts:

5.2‑17. tabula. Ieejas datu kopa

| Nr. | Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- | --- |
| 1 | SessionId | string | 1..1 | Sesijas identifikators |
| 2 | PersonCode | string | 1..1 | Personas kods |

Lietotāja identifikators, kas veic pieprasījumu, un transakcijas identifikators tiek ievietots integrācijas platformas transporta aploksnē.

Izejas datu shēma un apraksts:

5.2‑18. tabula. Izejas datu kopa

| Nr. | Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- | --- |
| 1 | SessionID | string | 1..1 | Sesijas identifikators |
| 2 | Error: Code, Descrioption, Severity | Salikts elements | 1..1 | Ja metodes izpildes laikā rodas kļūda, tad kļūdas informācija tiek aizpildīta kļūdas struktūrā. |
| 3 | PersonSex:  PersonSexCode  PersonSex | Salikts elements  Enumerator  Enumerator | 1..1 | Personas kods, par kuru tiek pieprasīti dati.  Atbilstošās vērtības: N nezināma, V vīrietis, S sieviete, Z neizvēlēta | |

#### Iegūt personas datu izmaiņas

Tips: Nepieciešama metode (IP, PMLP IR)

Nosaukums: ***WS:getPersonDataChangeSet***

Funkcijas apraksts: Funkcija atgriež izmaiņu datu kopu.

Plānots izmantot IP DIT - Datu Izplatīšanas tīklu.

#### Pārbaudīt likumīgo pārstāvi

Tips: Nepieciešama metode (IP, PMLP IR)

Nosaukums: ***WS:checkRepresentation***

Funkcijas apraksts: Pārbauda, vai pakalpojuma pieprasītājs ir likumīgais pārstāvis personai, kuras personas kodu ir norādījis pakalpojuma izpildē. Izsaucot IS servisu, jānorāda pakalpojuma prasītāja personas kods, pārstāvētās personas kods un pārstāvības veida kods (‘B’ – „aizbildnis”, ‘G’ – „aizgādnis”, ‘V’ – „vecāks nepilngadīgam bērnam”). Serviss atgriež ievadē saņemtos datus, ja visas pārbaudes ir veiksmīgi izturētas, vai pretējā gadījumā kļūdu.

Ieejas datu shēma un apraksts:

5.2‑19. tabula. Ieejas datu kopa

| Nr. | Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- | --- |
| 1 | RepresentativePersonCode | string | 1..1 | Pārstāvja personas kods |
| 2 | RepresentedPersonCode | string | 1..1 | Pārstāvētā personas kods |
| 3 | RepresentationKind | string | 1..1 | Pārstāvības veids |

Lietotāja identifikators, kas veic pieprasījumu, un transakcijas identifikators tiek ievietots integrācijas platformas transporta aploksnē.

Izejas datu shēma un apraksts:

5.2‑20. tabula. Izejas datu kopa

| Nr. | Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- | --- |
| 1 | RepresentativePersonCode | string | 1..1 | Pārstāvja personas kods |
| 2 | RepresentedPersonCode | string | 1..1 | Pārstāvētā personas kods |
| 3 | RepresentationKind | string | 1..1 | Pārstāvības veids |

#### Pārbaudīt personas rīcībspēju

Tips: Nepieciešama metode (IP, PMLP IR)

Nosaukums: ***WS:getIsPersonLegalyCapable***

Funkcijas apraksts: Funkcija pārbauda vai personai ar doto personas kodu nav ierobežota rīcībspēja.

Ieejas datu shēma un apraksts:

5.2‑21. tabula. Ieejas datu kopa

| Nr. | Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- | --- |
| 1 | SessionId | string | 1..1 | Sesijas identifikators |
| 2 | PersonCode | string | 1..1 | Personas kods. |

Lietotāja identifikators, kas veic pieprasījumu, un transakcijas identifikators tiek ievietots integrācijas platformas transporta aploksnē.

Izejas datu shēma un apraksts:

5.2‑22. tabula. Izejas datu kopa

| Nr. | Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- | --- |
| 1 | SessionID | string | 1..1 | Sesijas identifikators |
| 2 | Error: Code, Descrioption, Severity | Salikts elements | 1..1 | Ja metodes izpildes laikā rodas kļūda, tad kļūdas informācija tiek aizpildīta kļūdas struktūrā | |
| 3 | LegalCapability | string | 1..1 | Rīcībspēja ar iespējamām vērtībām: capable, incapable, partiallyCapable |

### Personas datu aktualizācija no VDEĀVK IS

#### Iegūt personas invaliditāti

Tips: Nepieciešama metode vai metožu kopa (IP, IVIS)

Nosaukums: ***WS:VDEAVK-GetPersonDisability***

Funkcijas apraksts: Funkcija pārbauda norādīto personas kodu, vai tam Veselības un darbspēju ekspertīzes ārstu valsts komisija reģistrā ir piešķirta invaliditātes grupa. Serviss atgriež pārbaudes rezultātu un personai piešķirtās invaliditātes grupas informāciju.

Ieejas datu shēma un apraksts:

5.2‑23. tabula. Ieejas datu kopa

| Nr. | Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- | --- |
| 1 | IdentityNumber | string  pattern: [0-9]{11} | 1..1 | Pieprasītāja personas kods |
| 2 | InstitutionCode | string | 1..1 | Pieprasītāja iestādes kods |
| 3 | AboutIdentityNumber | string  pattern: [0-9]{11} | 1..1 | Pieprasītais personas kods |
| 4 | EffectiveDate | date | 1..1 | Informācijas aktualitātes datums (pieprasījuma veikšanas datums) |
| **5** | SessionId | string | 1..1 | Sesijas identifikators (tas tiek iegūts izmantojot autentifikācijas WS) |

XML piemērs:

<UniversalServiceDataRequest>

<IdentityNumber>010191111222</IdentityNumber>

<InstitutionCode>str1234</InstitutionCode>

<AboutIdentityNumber>12121211122</AboutIdentityNumber>

<EffectiveDate>2012-12-13</EffectiveDate>

<SessionId>str1234</SessionId>

</UniversalServiceDataRequest>

Izejas datu shēma un apraksts:

5.2‑24. tabula. Izejas datu kopa

| Nr. | Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- | --- |
| 1 | CertExists | Boolean | 1..1 | Pazīme, vai eksistē pieprasītais apliecības numurs |
| 2 | PersonExists | Boolean | 1..1 | Pazīme, vai eksistē pieprasītā persona |
| 3 | ResultData | Salikts elements | 1..1 | Rezultāts |
| 4 | Rows | Salikts elements | 0..\* | Rezultāta rinda |
| 5 | ColumnValues | string | 0..1 | Ieraksta lauku vērtības |
| 6 | TableDefinition | Salikts elements | 0..1 | Universālā pakalpojuma datu tabulas definīcija |
| 7 | Columns | Salikts elements | 0..\* | Universālā pakalpojuma datu tabulas kolonnas |
| 8 | NameExternal | string | 0..1 | Kolonnas ārējais (lietotāja saskarnē attēlojamais) nosaukums |
| 9 | NameInternal | string | 0..1 | Kolonnas iekšējais nosaukums |

XML piemērs:

<UniversalServiceDataResponse>

<CertExists>true</CertExists>

<PersonExists>true</PersonExists>

<ResultData>

<Rows>

<ColumnValues>1</ColumnValues>

<ColumnValues>Pirmā grupa</ColumnValues>

<ColumnValues>13.12.2010</ColumnValues>

<ColumnValues>13.12.2022</ColumnValues>

</Rows>

<Rows>

<ColumnValues>2</ColumnValues>

<ColumnValues>Otrā grupa</ColumnValues>

<ColumnValues>20.10.2009</ColumnValues>

<ColumnValues>20.10.2029</ColumnValues>

</Rows>

<TableDefinition>

<Columns>

<NameExternal>Invaliditātes grupa </NameExternal>

<NameInternal>INV\_GROUP</NameInternal>

</Columns>

<Columns>

<NameExternal> Invalid. grupas nosaukums </NameExternal>

<NameInternal> INV\_GROUP\_NAME </NameInternal>

</Columns>

<Columns>

<NameExternal> Invaliditātes lēmuma datums </NameExternal>

<NameInternal> INV\_FROM </NameInternal>

</Columns>

<Columns>

<NameExternal> Invaliditāte noteikta līdz </NameExternal>

<NameInternal> INV\_TO </NameInternal>

</Columns>

</TableDefinition>

</ResultData>

</UniversalServiceDataResponse>

5.2‑25. tabula. Invaliditātes datu kopa no VDEĀVK IS reģistra

| Nr. | Lauks | Atbilstošais lauks reģistrā |
| --- | --- | --- |
| 01. | Invaliditātes grupa | Nosaka no reģistra uzskaites kartes lauka UPAK.INV\_GROUP (Invaliditātes grupa) |
| 02. | Invaliditātes grupas nosaukums | Nosaka no reģistra uzskaites kartes lauka UPAK.INV\_GROUP\_NAME (Invalid. grupas nosaukums) |
| 03. | Invaliditātes lēmuma datums | Nosaka no reģistra uzskaites kartes lauka UPAK.INV\_FROM (Invaliditātes lēmuma datums) |
| 04. | Invaliditāte noteikta līdz | Nosaka no reģistra uzskaites kartes lauka UPAK.INV\_TO (Invaliditāte noteikta līdz) |
| 05. | Invaliditātes lēmuma numurs | Nosaka no reģistra uzskaites kartes lauka INV\_NR |
| 06. | Invaliditātes amatpersona | Nosaka no reģistra uzskaites kartes lauka INV\_OFFICIAL\_NAME un INV\_OFFICIAL\_SURNAME |
| 07. | Invaliditātes iemesls (diagnoze) | Nosaka no reģistra uzskaites kartes lauka INV\_DIAGNOSIS\_CODE |

# Detalizēts projektējums

Nodaļa satur EVK IS moduļu detalizētu projektējumu.

Klasifikatoru moduļa detalizēts projektējums aprakstīts atsevišķi dokumentā [3].

Vakcinācijas reģistra detalizēts projektējums aprakstīts dokumentā [7].

## Datu struktūras

### Atribūtu grupa UpdateMode

UpdateMode ir šādas atribūta vērtības:

* A – pievienot jauno datu elementu;
* AR – pievienot jauno datu elementu vai aizvietot iepriekšējo elementu ar jauno versiju;
* N – nav izmaiņu (izmaiņas tiek ignorētas) – pēc noklusējuma;
* D – atbilstošais datu elements tiek izdzēsts (loģiski, uzstādot dzēšanas pazīmi).

Ja UpdateMode netiek norādīts, tad pēc noklusējuma tiek lietota vērtība N [ignorēt].

UpdateMode tiek izmantots datu struktūrām:

6.1‑1. tabula. UpdateMode izmantošana

| Nr. | Datu struktūra | UpdateMode atļautās vērtības | Piemērs | Piezīmes | |
| --- | --- | --- | --- | --- | --- |
| 1.1 | ContactParty | A [jauns] | <contactParty updateMode="A">  <code code="TEVS" codeSystem="1.3.6.1.4.1.38760.2.68"/>  <telecom value="E-pasts:pasha@pavel.com"/>  <contactPerson>  <name>  <family> Vorobjovs</family>  <given>Pāvels</given>  </name>  </contactPerson>  </contactParty> | Pievienojot jaunu ierakstu, EVK netiek analizēts, vai ieraksts ar analoģiskām vērtībām jau eksistē | |
| 1.2 | ContactParty | AR [labot]  N [ignorēt]  D [dzēst] | <contactParty updateMode="AR">  <id root =”1.3.6.1.4.1.38760.3.4.5.27” extension="XXX"/>  <code code="CODEfromOID" codeSystem="1.3.6.1.4.1.38760.2.68" displayName="CodeName"/>  <telecom value="tel:113"/>  <contactPerson>  <name>  <family> 0</family>  <given>1</given>  </name>  </contactPerson>  </contactParty> | XXX – ieraksta identifikators | |
| 2.1 | Telecom | A [jauns] | <telecom updateMode ="A" value="tel:113"/> |  | |
| 2.2 | Telecom | AR [labot]  N [ignorēt]  D [dzēst] | <telecom updateMode ="AR" value="tel:113" contactID="XXX"/> | XXX – ieraksta identifikators | |
| 3.1 | Delegation | A [jauns] | <hl7:delegation updateMode="AR"/>  <hl7:id extension=**"15307"** root=**"1.3.6.1.4.1.38760.3.4.5.30"**/>  ....  </hl7:delegation> |  | |
| 3.2 | Delegation | AR [labot]  N [ignorēt]  D [dzēst] |  | |
| 4 | Majority |  | <hl7:majority updateMode="AR">  <hl7:isMajor value=**"true"**/>  **....**  </hl7:majority> |  | |
| 5 | ConfidentialityRecord |  | <confidentialityRecord updateMode=**"AR"** >  .... |  | |
| 6 | ProfileSettings | A [jauns]  AR [labot]  N [ignorēt]  D [dzēst] | <hl7:LVEX\_MT000001UV01.ProfileRecord updateMode=**"AR"** classCode=**"DOCCLIN"** moodCode=**"EVN"**> |  |

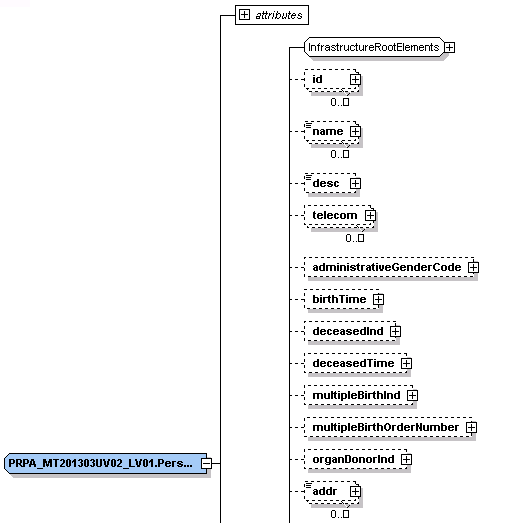
### Person (Pacienta kartes pamatdati)

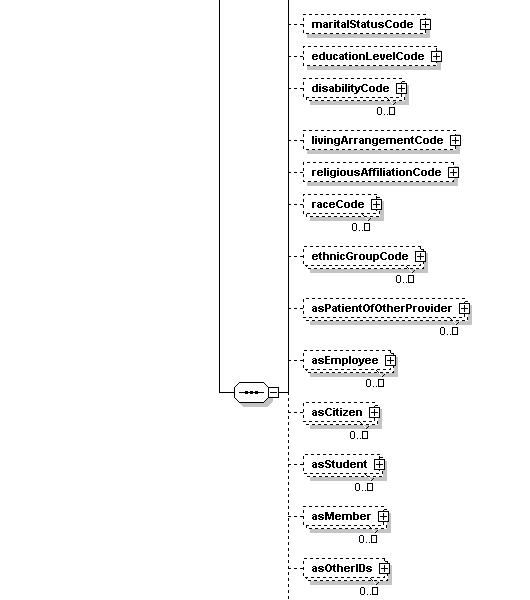
Identifikācija: PRPA\_MT201303UV02\_LV01.Person.

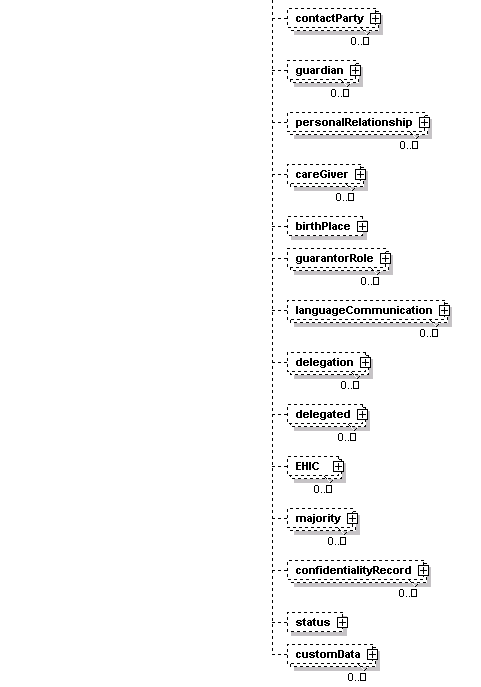
Datu struktūra balstīta uz HL7 PRPA\_MT201303UV02.Person.

Struktūra tiek izmantota:

1. Pacienta kartes pamatdatu atgriešanai (funkcija 6.2.9 getPatientCard (Iegūt pacienta karti))
2. Pacienta kartes pamatdatu papildināšanai un labošanai (funkcija 6.2.11 setPatientCard (Pievienot un papildināt pacienta kartes datus))







19. attēls. PRPA\_MT201303UV02\_LV01.Person datu struktūra

6.1‑2. tabula. Personas datu struktūra

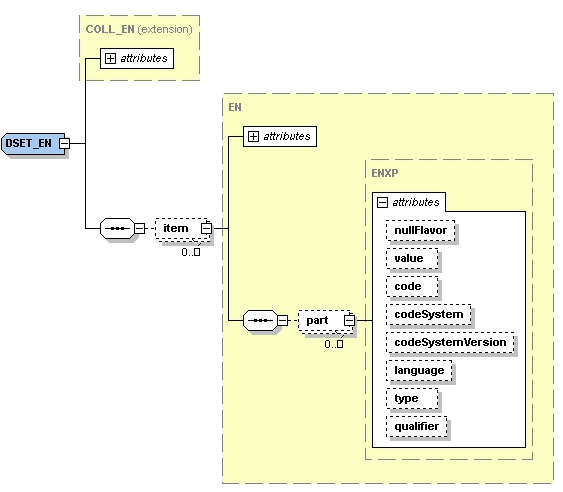
| Nr. | | | Parametrs | | Tips | | Skaits | | Apraksts | | Izgūt no DB (get) | | Saglabāt DB (set) |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | id | | II | | 0..1 | | Pacienta OID identifikators, atbilstoši:  Personas kods: 1.3.6.1.4.1.38760.3.1.1  Jaundzimušā identifikācija: 1.3.6.1.4.1.38760.3.1.3  Nedroša personas koda identifikācija: 1.3.6.1.4.1.38760.3.1.7  Ārzemnieka identifikācija:  1.3.6.1.4.1.38760.3.1.8.x | | Patients. IdentificationType  Patients. IdentificationCode | | Netiek apstrādāts | |
| 2 | name | | EN\_LV01 | | 1..1 | | Pacienta vārds/uzvārds. Kolekcija, kas satur vārda un uzvārda elementus | | Skat 6.1.2.1 PersonNames (Personas vārds) datu struktūru | | Var tikt apstrādāts ne personas koda autentificētām personām, skat. Skat 6.1.2.1 PersonNames (Personas vārds) datu struktūru. | |
| 3 | desc | | ED | | 0..1 | | Apraksts | | Netiek aizpildīts | | Netiek apstrādāts | |
| 4 | telecom | | TEL\_LV01 | | 0..\* | | Pacienta kontakti (telefons, epasts, citi). | | Skat. 6.1.2.2 PersonTelecom (Personas kontaktinformācija) datu struktūru | | Skat. 6.1.2.2 PersonTelecom (Personas kontaktinformācija) datu struktūru.  Dati datu bāzē tiek laboti un papildināti atbilstoši aizpildīšanas procedūras principiem | |
| 5 | administrativeGenderCode | | CE | | 0..1 | | Personas dzimums | | NPatients. GenderValue | | Var tikt apstrādāts ne personas koda autentificētām personām, NPatients. GenderValue  Lauks obligāts, ja tiek aizpildīti šī veida dati ārzemnieka kartei (identifikācijas OID  1.3.6.1.4.1.38760.3.1.8) | |
| 6 | birthTime | | TS | | 0..1 | | Pacienta dzimšanas datums | | NPatients. BirthTime | | Var tikt apstrādāts ne personas koda autentificētām personām, NPatients. BirthTime  Lauks obligāts, ja tiek aizpildīti šī veida dati ārzemnieka kartei (identifikācijas OID  1.3.6.1.4.1.38760.3.1.8). | |
| 7 | deceasedInd | | BL | | 0..1 | | Miršanas pazīme | | Aizpilda, ja  NPatients. PersonStatus = „MIRIS” | | Var tikt apstrādāts ne personas koda autentificētām personām NPatients. PersonStatus tiek uzlikts „MIRIS” | |
| 8 | deceasedTime | | TS | | 0..1 | | Pacienta miršanas laiks. | | Patients.DateOfDeath | | Var tikt apstrādāts ne personas koda autentificētām personām Patients.DateOfDeath  Ja tiek saņemts datums šājā laukā jāuzstāda arī NPatients. PersonStatus vērtība „MIRIS” | |
| 9 | multipleBirthInd | | BL | | 0..1 | | Pazīme, ka persona dzimusi vairāku augļu grūtniecībā | | Netiek aizpildīts | | Netiek apstrādāts | |
| 10 | multipleBirthOrderNumber | | INT | | 0..1 | | Dzimšanas kārtas numurs | | Netiek aizpildīts | | Netiek apstrādāts | |
| 11 | organDonorInd | | BL | | 0..1 | | Pazīme par orgānu donāciju. | | Netiek aizpildīts | | Netiek apstrādāts | |
| 12 | addr | | AD\_LV01 | | 0..\* | | Pacienta adreses (deklarēta adrese un faktiskā adrese). Deklarētai adresei addr struktūra aizpildīta pilnīgi, faktiskai adresei aizpildīta tikai struktūras StretAddressLine daļa. Faktiskā adrese tiek identificēta pēc addr tipa elementa ‘usage’ ar vērtību „FACT” | | Skat. 6.1.2.3 PersonAddress (Personas adrese) datu struktūru | | Var tikt apstrādāts ne personas koda autentificētām personām Skat. 6.1.2.3 PersonAddress (Personas adrese) datu struktūru | |
| 13 | maritalStatusCode | | CE | | 0..1 | | Netiek aizpildīts | | Netiek aizpildīts | | Netiek apstrādāts | |
| 14 | educationLevelCode | | CE | | 0..1 | | Netiek aizpildīts | | Netiek aizpildīts | | Netiek apstrādāts | |
| 15 | disabilityCode | | CE | | 0..1 | | Invaliditāte | | Netiek aizpildīts | | Netiek apstrādāts | |
| 16 | livingArrangementCode | | CE | | 0..1 | | Netiek aizpildīts | | Netiek aizpildīts | | Netiek apstrādāts | |
| 17 | religiousAffiliationCode | | CE | | 0..1 | | Netiek aizpildīts | | Netiek aizpildīts | | Netiek apstrādāts | |
| 18 | raceCode | | CE | | 0..1 | | Netiek aizpildīts | | Netiek aizpildīts | | Netiek apstrādāts | |
| 19 | ethnicGroupCode | | CE | | 0..1 | | Pacienta etniskā piederība (tautība). Vērtība no klasifikatora OID.226. | | Netiek aizpildīts | | Netiek apstrādāts | |
| 20 | asPatientOfOtherProvider | | PRPA\_HD201303UV02.PatientOfOtherProvider | | 0..\* | | Netiek aizpildīts | | Netiek aizpildīts | | Netiek apstrādāts | |
| 21 | asEmployee | | PRPA\_HD201303UV02.Employee | | 0..\* | | Netiek aizpildīts | | Netiek aizpildīts | | Netiek apstrādāts | |
| 22 | asCitizen | | PRPA\_HD201303UV02.Citizen | | 0..\* | | Netiek aizpildīts | | Netiek aizpildīts | | Netiek apstrādāts | |
| 23 | asStudent | | PRPA\_HD201303UV02.Student | | 0..\* | | Netiek aizpildīts | | Netiek aizpildīts | | Netiek apstrādāts | |
| 24 | asMember | | PRPA\_HD201303UV02.Member | | 0..\* | | Netiek aizpildīts | | Netiek aizpildīts | | Netiek apstrādāts | |
| 25 | asOtherIDs | | PRPA\_HD201303UV02.OtherIDs | | 0..\* | | Netiek aizpildīts | | Netiek aizpildīts | | Netiek apstrādāts | |
| 26 | contactParty | | PRPA\_HD201303UV02.ContactParty | | 0..\* | | Pacienta kontaktpersonas.  Aizpildīts ar PatientContactPersons Tabulas saturu | | Skat. (0 PersonContactPersons (Personas kontakt personas) datu struktūru | | Skat. (0 PersonContactPersons (Personas kontakt personas) datu struktūru.  Dati datu bāzē tiek laboti un papildināti atbilstoši aizpildīšanas procedūras principiem | |
| 27 | guardian | | PRPA\_HD201303UV02.Guardian | | 0..\* | | Netiek aizpildīts | | Netiek aizpildīts | | Netiek apstrādāts | |
| 28 | personalRelationship | | PRPA\_HD201303UV02.PersonalRelationship | | 0..\* | | Netiek aizpildīts | | Netiek aizpildīts | | Netiek apstrādāts | |
| 29 | careGiver | | PRPA\_HD201303UV02.CareGiver | | 0..\* | | Pacienta ģimenes ārsts | | Skat. (6.1.2.5PersonCareGiver (Personas ģimenes ārsts)) datu struktūru | | Netiek apstrādāts | |
| 30 | birthPlace | | PRPA\_HD201303UV02.BirthPlace | | 0..1 | | Dzimšanas vieta | | Netiek aizpildīts | | Netiek apstrādāts | |
| 31 | guarantorRole | | COCT\_MT670000UV04.GuarantorRole | | 0..\* | | Netiek aizpildīts | | Netiek aizpildīts | | Netiek apstrādāts | |
| 32 | languageCommunication | | PRPA\_HD201303UV02.LanguageCommunication | | 0..\* | | Netiek aizpildīts | | Netiek aizpildīt. | | Netiek apstrādāts | |
| 33 | delegation | | PRPA\_MT201303UV02\_LV01.Delegate | | 0..\* | | Atļaujas piekļūt pacienta kartei - Pilnvarotās personas un izsniegtās pilnvaras. | | Skat. (6.1.2.4 PersonDelegations (Pacienta kartes atļaujas)) datu struktūru, ar parametru Direction=”IN”. | | Skat. (6.1.2.4 PersonDelegations (Pacienta kartes atļaujas)) datu struktūru, ar parametru Direction=”IN”.  Tiek pārbaudīts, vai lietotājam ir tiesības noņemt vai pievienot atļauju. Labot un dzēst atļauju var tikai, ja parametrā Source ir paša pacienta identifikācijas kods (atbilstoši, tas, ko pats pacients ir izveidojis).  Dati datu bāzē tiek laboti un papildināti atbilstoši aizpildīšanas procedūras principiem | |
| 34 | delegated | | PRPA\_MT201303UV02\_LV01.Delegate | | 0..\* | | Tiesības pacientam piekļūt citu pacientu kartēm, piemēram, bērnu | | Skat. (6.1.2.4 PersonDelegations (Pacienta kartes atļaujas)) datu struktūru, ar parametru Direction=”OUT” | | Netiek apstrādāts | |
| 35 | EHIC | | PRPA\_MT201303UV02\_LV01.EHIC | | 0..\* | | Pacienta EVAK karšu dati | | (skat. 6.1.2.6 EHIC (EVAK kartes dati) | | Netiek apstrādāts | |
| 36 | majority | | PRPA\_MT201303UV02\_LV01.Majority | | 0..1 | | Pacienta pilngadības pazīme | | Aizpildīšana:  isMajor = NPatients.Majority  majorityDate = Patients.MajorityDate  majorityReason=Patients.MajorityReason | | Saņemot personas ieraksta izmaiņas, datu bāzes lauki tiek aizpildīti pēc izgūšanas principa. Lietotājam, kurš veic pilngadības pazīmes maiņu, nepieciešama tiesība  EvkRghtPatientMajoritySet | |
| 37 | confidentialityRecord | | RCMR\_MT000002UV02\_LV01.ConfidentialityRecord | | 0..\* | | Pacienta kartes pieejas ierobežojumu ieraksts.Skat. ConfidentialityRecord (Aizliegumu struktūra) | | Skat. ConfidentialityRecord (Aizliegumu struktūra) | | Skat. ConfidentialityRecord (Aizliegumu struktūra) | |
| 38 | status | | CS | | 0..1 | | Pacienta kartes statuss, atbilstoši klasifikatoram 1.3.6.1.4.1.38760.2.63 | | Tiek iegūta no NPatients.Status | | Netiek saglabāts no ieejas parametriem. | |
| 39 | customData | | PRPA\_MT201303UV02\_LV01.Section | | 0..\* | | Papildu dati, kas pieprasīti pie pacienta kartes. Papildu datu sekcijas aprakstītas pie atbilstošas datu struktūras | | Atkarīgs no datu grupas | | Atkarīgs no datu grupas | |

#### PersonNames (Personas vārds)

Identifikācija: PersonNames

*PersonNames* ir *Person* sastāvdaļa.

Datu struktūra balstīta uz HL7 EN salikto datu tipu.



20. attēls. EN datu struktūra

6.1‑3. tabula. EN datu struktūra

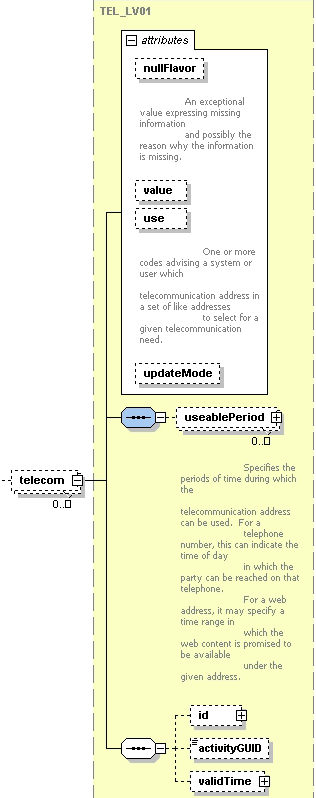
| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) | Saglabāt DB (set) |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | item | EN | 1..\* | Apraksta vārdu | - | - |
| 2 | part | ENXP | 1..\* | Atkarībā no atribūta Type vārds vai uzvārds. | Tiek aizpildīts no tabulas Patients.  Vārds:  Type =”GIV”,  Value = Patients.GivenName.  Uzvārds:  Type = „FAM”, Value = Patients.FamilyName. | Var tikt apstrādāts ne personas koda autentificētām personām  Vārds:  Type =”GIV”,  Value = Patients.GivenName.  Uzvārds:  Type = „FAM”, Value = Patients.FamilyName. |

#### PersonTelecom (Personas kontaktinformācija)

Identifikācija: PersonTelecom

*PersonTelecom* ir *Person* sastāvdaļa. Personai var būt neviens, viens vai vairāki telecom elements, kas satur kontaktinformāciju.

Datu struktūra balstīta uz HL7 TEL salikto datu tipu.



21. attēls. TEL\_LV01 datu struktūra

6.1‑4. tabula. TEL datu struktūra

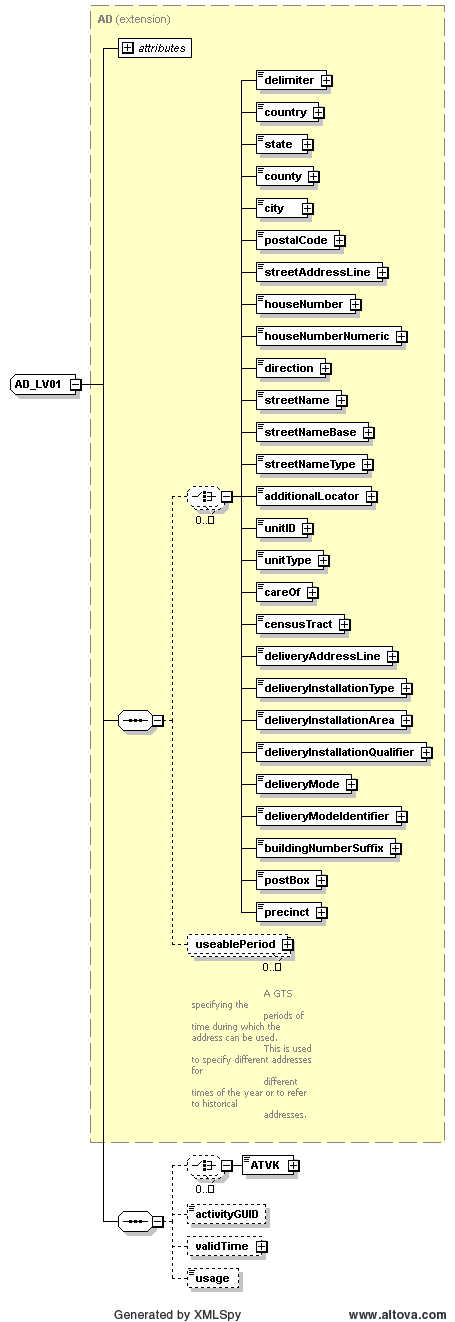
| Nr. | | | Parametrs | | Tips | | Skaits | | Apraksts | | Izgūt no DB (get) | | Saglabāt DB (set) |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1. | telecom | | TEL\_LV01 | | 0..\* | | Pacienta kontaktinformācija | | Atribūts ‘value’ tiek aizpildīts no tabulas PatientContacts.  ContactTypeCode Un Value.  Piemērs:  „tel: +371 123 456”,  kur:  tel –ContactTypeCode  +371... – Value  Atributs „UpdateMode” tiek izmantots atbilstoši standarta UpdateMode lietošanas veidam | | Value vērtība tiek saglabāta:  PatientContacts.ContactTypeCode, PatientContacts.Value. | |
| 2. | useablePeriod | | SXCM\_TS | | 0..\* | | Netiek izmantots | |  | |  | |
| 3. | id | | II | | 0..1 | | TEL\_LV01 ieraksta identifikators. Tiek izmantots Telekom ieraksta atrašanai papildināšanas vai dzēšanas gadījumā | | PatientContacts.PatientContactId | | PatientContacts.PatientContactId | |
| 4. | activityGUID | | String | | 0..1 | | TEL\_LV01 pēdējās izmaiņas ziņojuma Id | | PatientContacts.ActivityGUID | | PatientContacts.ActivityGUID | |
| 5. | validTime | | IVL\_TS | | 0..1 | | Pēdējās izmaiņas datums | | PatientContacts.ModifDate | | PatientContacts.ModifDate | |

#### PersonAddress (Personas adrese)

Identifikācija: PersonAddress

*PersonAddress* ir *Person* sastāvdaļa. Personai ir neviens vai viens adress elements.

Datu struktūra balstīta uz HL7 AD salikto datu tipu.



22. attēls. AD\_LV01 datu struktūra

6.1‑5. tabula. ADLV\_01 datu struktūra

|  |  |  |
| --- | --- | --- |
| **Elements** | **partType** | **Taga teksts** |
| **1 līmeņa ATV nosaukums (apriņķis)** | LatvianAddress1Level | NPatients.ATUName1 |
| **2 līmeņa ATV nosaukums (rajons / republikas nozīmes pilsēta)** | LatvianAddress2Level | NPatients.ATUName2 |
| **Trešā līmeņa ATV nosaukums (novads / pilsēta ar lauku teritoriju / pilsēta / pagasts)** | LatvianAddress3Level | NPatients.ATUName3 |
| **Ceturtā līmeņa ATV nosaukums (pilsēta / lauku teritorija / pagasts)** | LatvianAddress4Level | NPatients.ATUName4 |
| **1 līmeņa ATV nosaukums (apriņķis)** | LatvianAddress1LevelCode | NPatients.ATUCode1 |
| **2 līmeņa ATV nosaukums (rajons / republikas nozīmes pilsēta)** | LatvianAddress2LevelCode | NPatients.ATUCode2 |
| **Trešā līmeņa ATV nosaukums (novads / pilsēta ar lauku teritoriju / pilsēta / pagasts)** | LatvianAddress3LevelCode | NPatients.ATUCode3 |
| **Ceturtā līmeņa ATV nosaukums (pilsēta / lauku teritorija / pagasts)** | LatvianAddress4LevelCode | NPatients.ATUCode4 |
| **Latvijas adreses struktūra: Ielas kods** | LatvianAddressStreetCode | Patients.StreetCode |
| **Latvijas adreses struktūra: Ciema kods** | LatvianAddressVillageCode | Patients.VillageCode |

Šajā struktūrā datu saglabāšana DB ir pieļaujamā gadījumā, ja tiek izpildīta pacienta kartes datu atjaunošana pacienta kartei bez PMLP verificēta personas koda tipa validācijas.

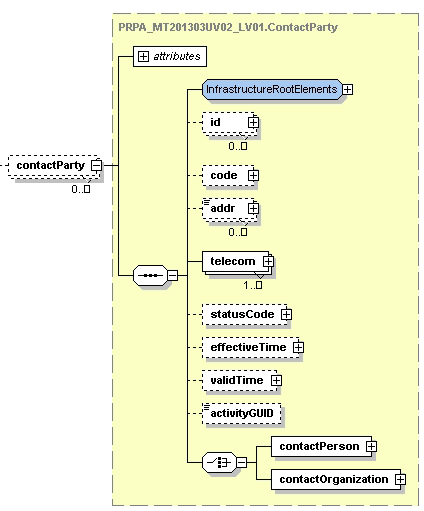
| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) | Saglabāt DB (set) |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | addr | AD | 0..1 | Pacienta adrešu komplekts (pacienta pamatadrese). | - | - |
| 2 | item | AD | 0..\* | Pacienta adrese.  Elementa atribūti netiek aizpildīti. | - | - |
| 3 | country | ADXP | 0..1 | Valsts kods | NPatients.ICAOCode | NPatients.ICAOCode |
| 4 | county | ADXP | 0..1 | Ciems | NPatients.Village | NPatients.Village |
| 5 | additionalLocator | ADXP | 0..1 | Adrešu reģistra kods | Patients.ARCode | Patients.ARCode |
| 6 | streetName | ADXP | 0..1 | Iela | Patients.Street | Patients.Street |
| 7 | houseNumber | ADXP | 0..1 | Mājas nosaukums | Patients.HouseName | Patients.HouseName |
| 8 | houseNumberNumeric | ADXP | 0..1 | Mājas numurs | Patients.HouseNumber | Patients.HouseNumber |
| 9 | buildingNumberSuffix | ADXP | 0..1 | Korpuss | Patients.HouseCorpsNumber | Patients.HouseCorpsNumber |
| 10 | unitId | ADXP | 0..1 | Dzīvokļa numurs | Patients.FlatNumber | Patients.FlatNumber |
| 11 | postalCode | ADXP | 0..1 | Pasta indekss | Patients.PostalCode | Patients.PostalCode |
| 12 | streetAdrressLine | ADXP | 0..1 | Adrese ārvalstī | Patients.AddressAbroad | Patients.AddressAbroad |
| 13 | usage | ADXP.usage | 0..1 | Adreses veids | Ja lauks Patients.FactAddress nav tukšs, tad tiek izveidota addr struktūra norādot vērtību FACT, citādi – tikai DECLARED | Gadījumā, ja „usage” elementa vērtība ir FACT, no adrešu datu struktūras tiks atlasīts lauks „streetAddressLine” un saglabāts Tabulā Patients kolonnā FactAddress |

*PersonContactPersons (Personas kontakt personas)*

*Identifikācija: PersonContactPersons*

*PersonContactPersons* ir *Person* sastāvdaļa. Personai ir neviens, viens vai vairāki contactPerson elementi.

Datu struktūra balstīta uz HL7 PRPA\_MT201303UV02.ContactParty salikto datu tipu.



23. attēls. PRPA\_MT201303UV02\_LV01.ContactParty datu struktūra

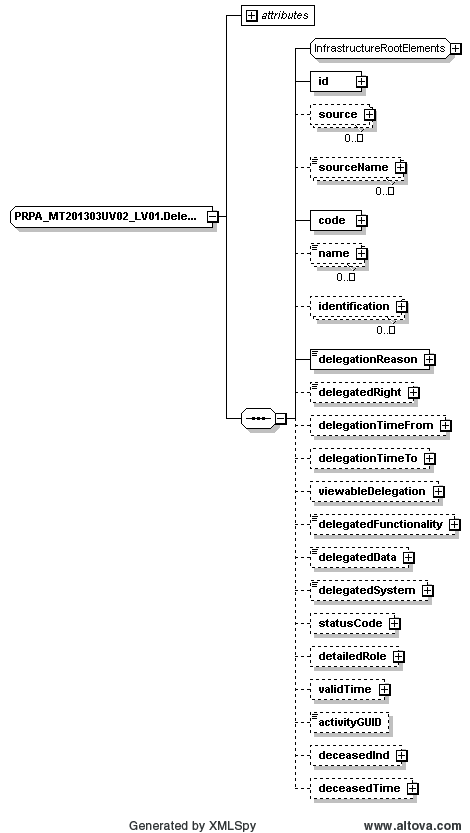
6.1‑6. tabula. ContactParty datu struktūra

| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) | Saglabāt DB (set) |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | id | II | 0..1 | Pacienta kontakta identifikatoru komplekts | Tiek aizpildīta viena identifikatora vērtība: id=PatientContactPersons.ContactPersonID | Ja veido jaunu kontaktpersonu, tad neaizpilda. Ja labo esošu ierakstu, tad aizpilda ar labojamo ierakstu |
| 2 | code | CD | 1..1 | Kontaktpersonas tipa kods | Aizpildīts no lauka: PatientContactPersons.ContactPersonTypeCode | Veidojot jaunu vai labojot, norāda:  kontaktpersonas tipa klasifikators  OID:  1.3.6.1.4.1.38760.2.68,  Code – tipa kods |
| 3 | addr | AD | 0..1 | Netiek aizpildīts | Netiek aizpildīts | Netiek apstrādāts |
| 4 | telecom | TEL | 1..1 | Kontaktinformācija  Atribūts use – netiek izmantots,  Atribūts capabilities- netiek izmantots. | Tiek aizpildīts no tabulas PatientContactPersons;  Atribūts value= PatientContactPersons.Value | PatientContactPersons. Value |
| 5 | statusCode | CS | 0..1 | Kontakta informācijas statusa kods | Aizpildāms no PatientContactPersons.Status | - |
| 6 | effectiveTime | IVL\_TS | 0..1 | Netiek izmantots | Netiek aizpildīts | Netiek apstrādāts |
| 7 | validTime | IVL\_TS | 0..1 | Pēdējas izmaiņas datums | PatientContactPersons.ModifDate | Aizpildīts PatientContactPersons.ModifDate automātiski |
| 8 | activityGUID | string | 0..1 | Izmaiņas ziņojuma identifikators | PatientContactPersons.ActivityGUID | Aizpildīts PatientContactPersons.ActivityGUID automātiski |
| 9 | contactPerson | COCT\_MT030207UV07.Person | 1..1 | Tiek izmantots tikai elements „name”. Tiek aizpildīts tikai viens „name” eksemplārs.  Elements tiek aizpildīts atbilstoši datu struktūrai EN | Type=”GIV”  Value = PatientContactPersons.GivenName  Type= ”FAM”  Value = PatientContactPersons.FamilyName | Type=”GIV”  Value = PatientContactPersons.GivenName  Type= ”FAM”  Value = PatientContactPersons.FamilyName |
| 10 | contactOrganization | COCT\_MT150007UV.Organization | 1..1 | Netiek izmantots | Netiek aizpildīts | Netiek apstrādāts |

#### PersonDelegations (Pacienta kartes atļaujas)

Identifikācija: PersonDelegations

*PersonDelegations* ir *Person* sastāvdaļa. Personai ir neviens, viens vai vairāki PersonDelegations elementi.



24. attēls. PRPA\_MT201303UV02\_LV01.Delegate datu struktūra

6.1‑7. tabula. Delegate datu struktūra

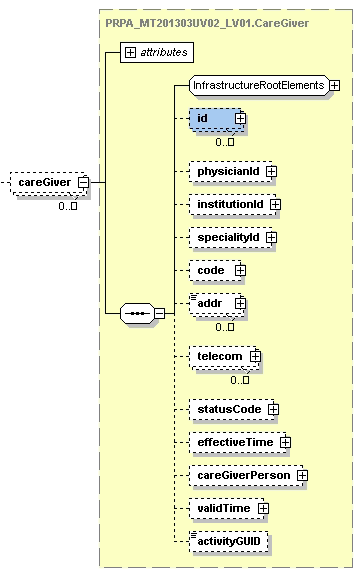
| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) | Saglabāt DB (set) |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | id | II | 0..1 | Atļaujas identifikators | Tiek aizpildīts ar PatientDelegations.DelegationID | Ja veido jaunu kontaktpersonu, tad neaizpilda. Ja labo esošu ierakstu, tad aizpilda ar labojamo ierakstu |
| 2 | source | II | 0..\* | Pacienta identifikators, atbilstoši:  Personas kods: 1.3.6.1.4.1.38760.3.1.1  Jaundzimušā identifikācija: 1.3.6.1.4.1.38760.3.1.3  Ārzemnieka Id:  1.3.6.1.4.1.38760.3.1.8.xx  Nedroša Latvijas iedzīvotāja id  1.3.6.1.4.1.38760.3.1.7 | Izmanto, lai pēc pacienta identifikatora iegūtu visus PatientDelegations. | Netiek apstrādāts |
|  | sourceName | EN | 0..\* | Pacienta vārds/uzvārds | No tabulas Patients.GivenName un Patients.FamilyName | Netiek apstrādāts |
| 3 | code | CS | 1..1 | Atļaujas tips | PatientDelegations.DelegationType | PatientDelegations.DelegationType |
| 4 | name | EN | 0..\* | Apraksta vārda struktūru | Tiek izveidots viens ieraksts (item), aizpildot tam vārda daļas. Tiek aizpildīts ar  PatientDelegations tabulas laukiem:  PatientDelegations.GivenName type=”GIV”  PatientDelegations.FamilyName type=”FAM” | PatientDelegations.GivenName type=”GIV”  PatientDelegations.FamilyName type=”FAM” |
| 6 | delegationReason | ST | 1..1 | Atļaujas piešķiršanas pamatojums. Brīva teksta atļaujas paskaidrojums | PatientDelegations.DelegationReason | PatientDelegations.DelegationReason |
| 7 | delegatedRight | ST | 0..1 | Atļauta tiesība | PatientDelegations.DelegatedRight | PatientDelegations.DelegatedRight |
| 8 | delegationTimeFrom | TS | 0..1 | Atļaujas derīguma termiņa sākums | PatientDelegations.DelegatedTimeFrom | PatientDelegations.DelegatedTimeFrom |
| 9 | delegationTimeTo | TS | 0..1 | Atļaujas derīguma termiņa beigas | PatientDelegations.DelegatedTimeTo | PatientDelegations.DelegatedTimeTo |
| 11 | delegatedFunctionality | ST | 0..1 | Sistēmai neinterpretējama funkcionalitātes apraksta informācija (drošības talona daļa) | PatientDelegations.DelegatedFunctionality | PatientDelegations.DelegatedFunctionality |
| 12 | delegatedData | ST | 0..1 | Sistēmai neinterpretējama datu apraksta informācija (laika periods, vai autors, vai ārstniecības iestāde) | PatientDelegations.DelegatedData | PatientDelegations.DelegatedData |
| 13 | delegatedSystem | ST | 0..1 | Atļaujas sistēma | PatientDelegations.DelegatedSystem | PatientDelegations.DelegatedSystem |
| 14 | statusCode | CS | 0..1 | Atļaujas statuss no klasifikatora Atļauju statusi | PatientDelegations.Statuss | PatientDelegations.Statuss |
|  | detailedRole | CS | 0..1 | Atļaujai piesaistīta detalizēta loma. | Loma tiek izrēķināta pēc checkDetailedRole aprēķina algoritma, kā izejas datus izmantojot pacienta un atļaujas identifikācijas | Netiek apstrādāts |
| 15 | viewableDelegation | BL | 0..1 | Atļauja un atļaujas audita informācija ir redzama pacientam | PatientDelegations.viewableDelegation; Ieraksti, kuriem šī lauka vērtība ir „false” tiek atgriezti tikai ja lietotājam ir loma INVESTIGATOR | PatientDelegations.viewableDelegation |
|  | validTime | IVL\_TS | 0..1 | Ieraksta modificēšanas datums/laiks | Personalizētas datubāzes tabulas PatientDelegations.ModifDate | Netiek apstrādāts |
|  | activityGUID | string | 0..1 | Ieraksta modificēšanas ziņojuma identifikators | Personalizētas datubāzes tabulas PatientDelegations.ActivityGUID | Netiek apstrādāts |
|  | deceasedInd | BL | 0..1 | Miršanas pazīme | Aizpilda, ja  NPatients. PersonStatus = „MIRIS” | Netiek apstrādāts |
|  | deceasedTime | TS | 0..1 | Deleģētās personas miršanas laiks | Patients.DateOfDeath | Netiek apstrādāts |

#### PersonCareGiver (Personas ģimenes ārsts)

Identifikācija: PersonCareGiver

*PersonCareGiver* ir *Person* sastāvdaļa.

Datu struktūra balstīta uz HL7 PRPA\_MT201303UV02.CareGiver salikto datu tipu.



25. attēls. PRPA\_MT201303UV02\_LV01.CareGiver

6.1‑8. tabula. CareGiver datu struktūra

| Nr. | | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) | Saglabāt DB (set) | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | id | II | 0..\* | Ārsta identifikatoru komplekts | Tiek aizpildīts ar tabulas PatientFamilyDoctors, lauks  Root = 1.3.6.1.4.1.38760.3.3.5.1  Extension = PatientFamilyDoctors .PhysicianContractCode | Netiek apstrādāts |
| 2 | code | CD | 0..1 | Netiek izmantots |  | Netiek apstrādāts |
| 3 | addr | COLL\_AD | 0..1 | Netiek izmantots |  | Netiek apstrādāts |
| 4 | telecom | COLL\_TEL | 0..1 | Netiek izmantots |  | Netiek apstrādāts |
| 5 | statusCode | CS | 0..1 | Ārsta ieraksta statuss kods.  Tiek aizpildīts ar tabulas PatientFamilyDoctors, lauku RecordStatus |  | Netiek apstrādāts |
| 6 | effectiveTime | IVL\_TS | 0..1 | Ģimenes ārsta piesaistes laiks | Tiek aizpildīta IVL\_TS datu struktūra;  Low = DoctorTimeFrom  High = DoctorTimeTo | Netiek apstrādāts |
| 7 | careGiverPerson | COCT\_MT030207UV07.Person | 1..1 | Ārsts | Tiek aizpildīts tikai elements „name”. Elements tiek aizpildīts atbilstoši datu struktūrai EN.  Struktūra tiek aizpildīta ar ārsta informāciju no klasifikatora 1.3.6.1.4.1.38760.2.1 Ārstniecības personas.  Vārds type=”GIV”  Uzvārds type=”FAM” | Netiek apstrādāts |
| 8 | validTime | IVL\_TS | 0..1 | Ieraksta izmaiņas datums/laiks | PatientFamilyDoctors.ModifTime | Netiek apstrādāts |
| 9 | activityGUIF | string | 0..1 | Ieraksta izmaiņas ziņojuma identifikators | PatientFamilyDoctors.ActivityGUID | Netiek apstrādāts |

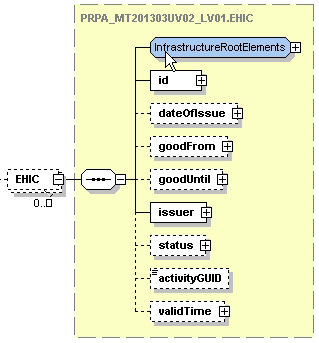
Alternatīvie scenāriji

Nav atrasta pacienta karte. Tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte) un izveidots audita ieraksts.

#### EHIC (EVAK kartes dati)

Identifikācija: EHICdata

*EHIC* ir *Person* sastāvdaļa.



26. attēls. PRPA\_MT201303UV02\_LV01.EHIC

6.1‑9. tabula. EHIC datu struktūra

| Nr. | | Parametrs | | Tips | Skaits | | Apraksts | Izgūt no DB (get) | Saglabāt DB (set) | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | Id | II | | 1..1 | EVAK kartes numurs | | root = 1.3.6.1.4.1.38760.3.3.1.3  Extension = PatientEhicData.EhicNumber | Netiek apstrādāts |
| 2 | dateOfIssue | TS | | 0..1 | EVAK izsniegšanas datums | | PatientEhicDataюDateOfIssue | Netiek apstrādāts |
| 3 | goodFrom | TS | | 0..1 | EVAK derīgs no | | PatientEhicData.ValidFrom | Netiek apstrādāts |
| 4 | goodUntil | TS | | 0..1 | EVAK derīgs līdz | | PatientEhicData.ValidTill | Netiek apstrādāts |
| 5 | Issuer | CD | | 1..1 | Izsniegšanas organizācijas kods un nosaukums | | PatientEhicData.IssuingInstitution | Netiek apstrādāts |
| 6 | Status | CD | | 0..1 | EVAK status kods un nosaukums | | PatientEhicData.status | Netiek apstrādāts |
| 7 | activityGUID | II | | 0..1 | EVAK izmaiņas operācijas identifikators no IP | | PatientEhicData.activityGUID | Netiek apstrādāts |
| 8 | validTime | IVL\_TS | | 0..1 | EVAK izmaiņas operācijas datums/laiks | | PatientEhicData.ModifDate | Netiek apstrādāts |

Alternatīvie scenāriji

Nav atrasta pacienta karte. Tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte) un izveidots audita ieraksts.

#### Section (Pacienta kartes datu grupas)

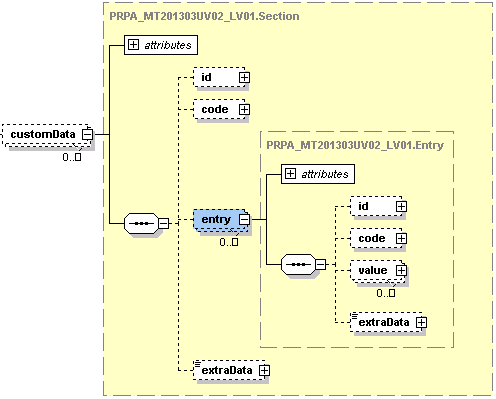
Identifikācija: Section

*Section* ir *Person* sastāvdaļa.

Iespējāmas sekcijas un elementu identifikatori:

1.3.6.1.4.1.38760.3.4.5.46.1 PREDA demogrāfijas dati

1.3.6.1.4.1.38760.3.4.5.46.1.1 Nāves celonis



27. attēls. PRPA\_MT201303UV02\_LV01.Section

6.1‑10. tabula. EHIC datu struktūra

| Nr. | | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) | Saglabāt DB (set) | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | Id | II | 0..1 | Sekcijas identifikators |  |  |
| 2 | code | CD | 0..1 | Sekcijas kods un nosaukums (vērtrība no OID 2.21 klasifikatora) |  |  |
| 3 | entry | PRPA\_MT201303UV02\_LV01.Entry | 0..\* | Saraksts ar paplašinājuma datiem |  |  |
|  | id | II | 0..1 | Elementa identifikators |  |  |
|  | code | CD | 0..1 | Elementa kods un nosaukums (vērtība no OID 2.21 klasifikatora) |  |  |
|  | value | ANY | 0..\* | Datu elements (atkarība no elementa veida) |  |  |
|  | extraData | ED | 0..1 | Nestrukturēta datu informācija. |  |  |
| 4 | extraData | ED | 0..1 | Nestrukturēta datu informācija |  |  |

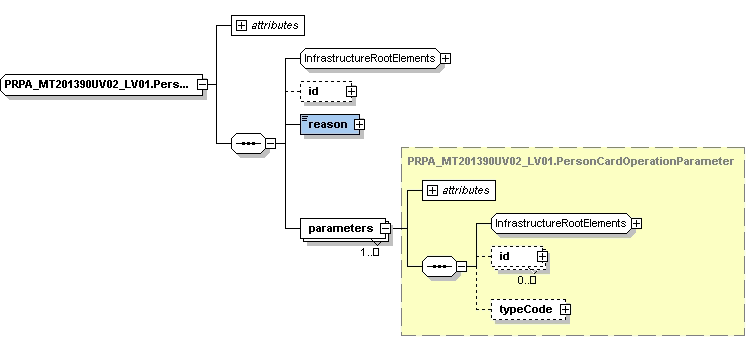
### PersonCardOperations (Personas kartes izveidošana; Sapludināšana)

Identifikācija: PRPA\_MT201390UV02\_LV01.PersonCardOperations

Datu struktūra balstīta uz HL7 PRPA\_MT201390UV02.PersonCardOperations datu tipu.

Struktūra tiek izmantota:

1. Pacienta kartes izveidei (funkcija 6.2.5 createPatientCard (Izveidot pacienta kartes ierakstu))
2. Pacienta karšu sapludināšanai (funkcija 6.2.10 mergePatientCards (Sapludināt pacienta kartes))



28. attēls. PRPA\_MT201390UV02\_LV01.PersonCardOperations datu struktūra

6.1‑11. tabula. PersonCardOperations datu struktūra

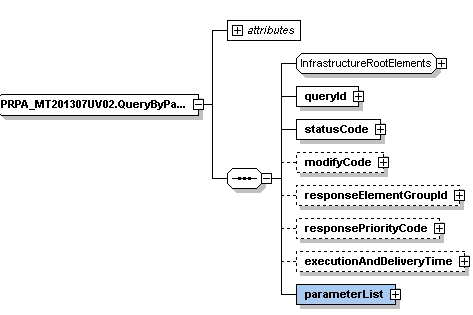
| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) | Saglabāt DB (set) |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | id | II | 0..1 | Operācijas identifikators | Netiek izmantots (netiek atgriezts no EVK) | Tiek izmantots audita mērķos. |
| 3 | reason | ST | 1..1 | Pamatojums brīvā tekstā | Netiek izmantots (netiek atgriezts no EVK) | Tiek izmantots audita mērķos. |
| 4 | parameters | PRPA\_MT201390UV02\_LV01.PersonCardOperationParameter | 1..\* | Pacienta kartes identifikācija, kā arī loma, kuru tā spēlē konkrētā operācijā | Netiek izmantots (netiek atgriezts no EVK) | Skat. tālāk: |
| 5 | Id | II | 0..\* | Pacienta kartes identifikācija | Netiek izmantots (netiek atgriezts no EVK) | Tiek izmantots pacienta kartes identifikācijai, lai izpildītu pieprasīto operāciju. |
| 6 | typeCode | CS | 0..1 | Operācijas parametra tips | Netiek izmantots (netiek atgriezts no EVK) | Tiek izmantots operācijas parametra tipa noteikšanai. Vērtības no klasifikatora Personas karšu operācijas parametri |
| 1 | id | II | 0..1 | Operācijas identifikators | Netiek izmantots (netiek atgriezts no EVK) | Tiek izmantots audita mērķos. |

### PersonQueryByParameter (Personas datu pieprasījums)

Identifikācija: PRPA\_MT201307UV02\_LV01.PersonQueryByParameter

Datu struktūra balstīta uz HL7 PRPA\_MT201307UV02.QueryByParameter datu tipu.

Struktūra tiek izmantota pacienta kartes pamatdatu pieprasījumā (funkcija 6.2.9 getPatientCard (Iegūt pacienta karti)).



29. attēls. PRPA\_MT201307UV02.QueryByParameter datu struktūra

6.1‑12. tabula. QueryByParameter datu struktūra

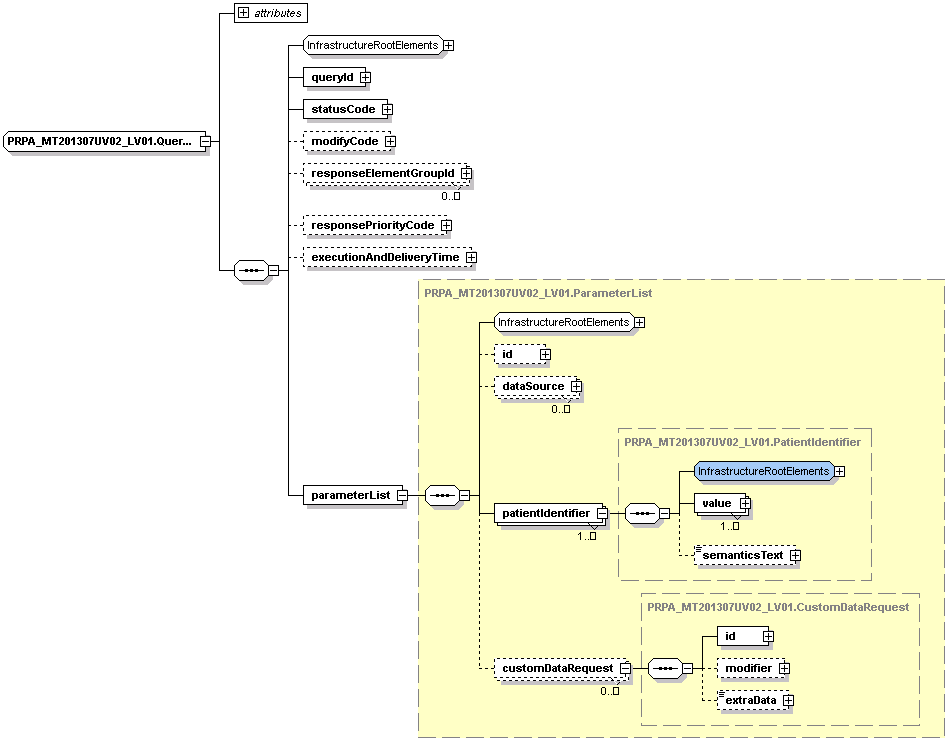
| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | | queryId | II | 1..1 | Pieprasījuma identifikators.  Root = Pieprasījuma tipa OID (1.3.6.1.4.1.38760.3.4.5.18) |
| 2 | | statusCode | CS | 1..1 | Vai jāmeklē tikai aktuālus datus, vai jāizmanto arī vēsturi. No klasifikatora Meklēšanas veidi |
| 3 | | modifyCode | CS | 0..1 | Netiek izmantots |
| 4 | | responseElementGroupId | II | 0..1 | Netiek izmantots |
| 5 | | responsePriorityCode | CS | 0..1 | Netiek izmantots |
| 6 | | executionAndDeliveryTime | TS | 0..1 | Netiek izmantots |
| 7 | | parameterList | PRPA\_MT201307UV02.ParameterList | 1..1 | Sk. 6.1.4.1 ParameterList. |

#### ParameterList

Identifikācija: PRPA\_MT201307UV02\_LV01. ParameterList

*ParameterList* ir *QueryByParameter* sastāvdaļa. Var būt viens ParameterList elements.

Datu struktūra balstīta uz HL7 PRPA\_MT201307UV02.ParameterList salikto datu tipu.



30. attēls. PRPA\_MT201307UV02.ParameterList datu struktūra

PatientIdentifier datu struktūras apraksts:

6.1‑13. tabula. ParameterList datu struktūra

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | value | II | 1..\* | Saraksts ar personu identifikatoriem.  Satur neierobežoto daudzumu item elementu, ar II datu tipa ierakstu.  II ieraksts aizpildīts ar:  Root – pacienta identifikācijas tips (OID)  Extension – pacienta identifikācijas kods |
| 2 | semanticsText | ST | 1..1 | Vērtība tiek ignorēta |
| 3 | id | II | 0..1 | Netiek izmantots |
| 4 | dataSource | PRPA\_MT201307UV02\_LV01.DataSource | 0..\* | Netiek izmantots |
| 5 | customDataRequest | PRPA\_MT201307UV02\_LV01.CustomDataRequest | 0..\* | Saraksts ar papildu pieprasījuma parametriem, kas definē pacienta kartes rezultāta atgriešanas datus |
|  | Id | II | 1..1 | Atgriežāmā datu komplekta root saraksts ar iespējamiem datu komplektiem ir definēts 6.1.2.7 |
|  | Modifier | CS | 0..1 | Vai dati ar Id lauka norādīto sekciju jāiekļauj rezultātā INCLUDE, vai jāizņem no rezultāta EXCLUDE |
|  | extraData | ED | 0..1 | Definējams atbilstoši konkrētai datu grupai. |

### CDA dokumenta identifikācija

Dokumenta identifikāciju aizpilda dokumenta izveidotājs.

Dokumenta identifikācijas datu komplektu veido:

* Id – dokumenta versijas identifikators
* SetId – dokumenta identifikators, kas vienāds visām viena dokumenta versijām
* VersionNumber – dokumenta versijas kārtas numurs

#### Id

Dokumenta versijas identifikators.

Formāts: ĀI.NR.V, kur

* ĀI – ārstniecības iestādes kods (no NVD VIS sistēmas),
* NR – dokumenta numurs, kam jābūt unikālam vienas ārstniecības iestādes ietvaros,
* V – dokumenta versijas numurs, kam jābūt unikālam viena dokumenta ietvaros.

Piemērs: 25489.653.1

Kur:

* 25489 – ārstniecības iestāde
* 653 – dokumenta numurs konkrētajā ārstniecības iestādē
* 1 – dokumenta versijas numurs

#### SetId

Dokumenta identifikators.

Formāts: ĀI.NR, kur

* ĀI – ārstniecības iestādes kods (no NVD VIS sistēmas),
* NR – dokumenta numurs, kam jābūt unikālam vienas ārstniecības iestādes ietvaros.

Piemērs: 25489.653

Kur:

* 25489 – ārstniecības iestāde
* 653 – dokumenta numurs konkrētajā ārstniecības iestādē

#### VersionNumber

Dokumenta versijas numurs.

Dokumenta versijas numuram jābūt unikālam viena dokumenta (SetId) ietvaros.

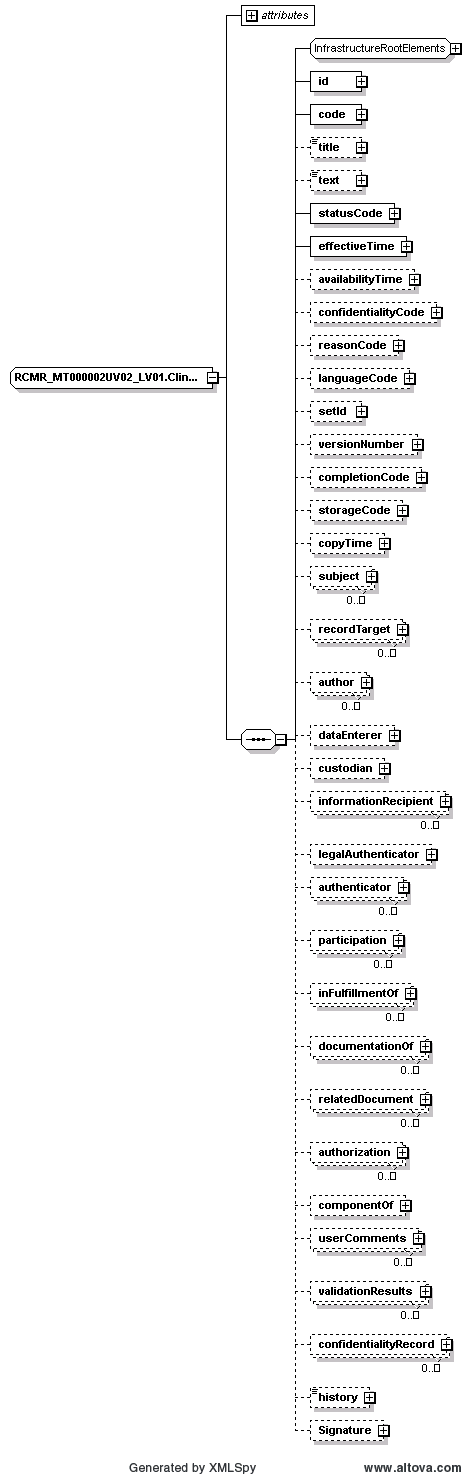
### DocumentStructure (CDA dokumenta aploksnes struktūra)

Identifikācija: DocumentStructure

Datu struktūra balstīta uz HL7 RCMR\_MT000002UV02\_LV01.ClinicalDocument salikto datu tipu.

Struktūra tiek izmantota:

1. Dokumentu pievienošanai (funkcija 6.2.1 addDocument (Pievienot dokumentu – sinhronā daļa))
2. Dokumentu statusu maiņai (funkcija 6.2.3 setDocumentStatus (Mainīt dokumenta statusu))
3. Dokumenta atgriešanai (funkcija 6.2.6 getDocument (Iegūt dokumentu))
4. Dokumentu saraksta atgriešanai (funkcija 6.2.7 getDocumentList (Iegūt dokumentu sarakstu))



31. attēls. RCMR\_MT000002UV02\_LV01.ClinicalDocument datu struktūra

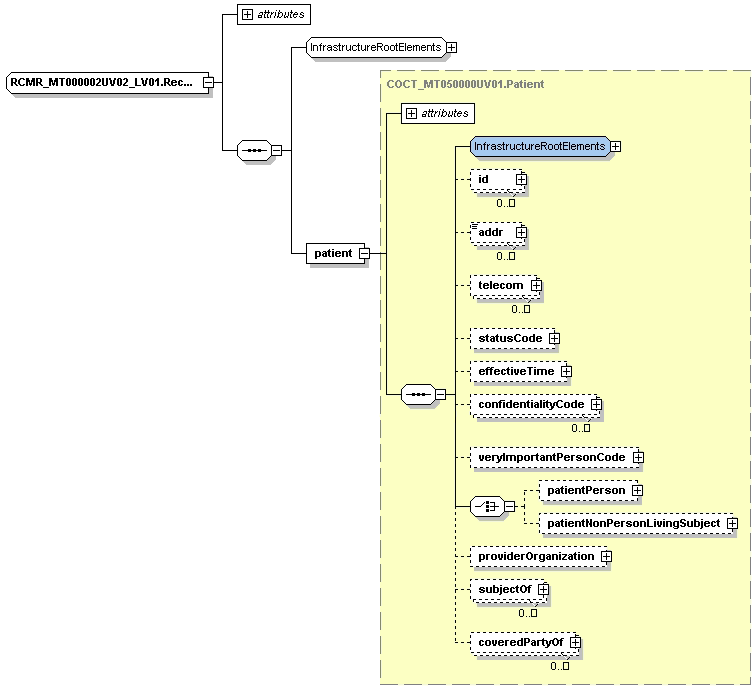
6.1‑14. tabula. ClinicalDocuments datu struktūra

| Nr. | | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) | Saglabāt DB (set) |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | id | II | 1..1 | Dokumenta versijas identifikators | Root = 1.3.6.1.4.1.38760.3.3.4.1  Extension = Documents.DocumentID | Ja Root = 1.3.6.1.4.1.38760.3.3.4.1  Documents.DocumentID = Extension |
|  | code | CE | 1..1 | Dokumenta tipa kods | Code = DocumentTemplates.DocumentTypeOID: 1.3.6.1.4.1.38760.2.21 | DocumentTemplates.DocumentTypeOID = Code  , CodeSystem = 1.3.6.1.4.1.38760.2.21 |
|  | text | ED | 0..1 | Dokumenta saturs | Documents.CDA (ja documentFormat = XML), vai Documents.HTML (ja documentFormat = HTML) vai dokumenta PDF (ja documentFormat = PDF). Dokumenta saturs tiek atgriezts kā base64 teksts | Saglabājot dokumentu datu bāzē, tiek aizpildīts lauks  Documents.CDA |
|  | statusCode | CS | 1..1 | Dokumenta statuss | Code = Documents.Statuss | Documents.Statuss = Code |
|  | effectiveTime | TS | 1..1 | Dokumenta datums/laiks | Documents.EffectiveTime | Documents.EffectiveTime |
|  | availabilityTime | TS | 0..1 | Netiek izmantots | Netiek aizpildīts | Netiek apstrādāts |
|  | confidentialityCode | CE | 0..1 | Dokumenta pieejas tiesību deskriptors | Code = Documents.AccessRights  No klasifikatora Objektu pieejas struktūra | Netiek apstrādāts. Dokumenta pieejas tiesības tiek uzstādītas, izmantojot confidentialityRecord struktūru |
|  | reasonCode | CE | 0..1 | Netiek izmantots | Netiek aizpildīts | Netiek apstrādāts |
|  | languageCode | CS | 0..1 | Netiek izmantots | Netiek aizpildīts | Netiek apstrādāts |
|  | setId | II | 0..1 | Dokumenta identifikators | Root = 1.3.6.1.4.1.38760.3.3.4.2  Extension = Documents.SetID | Ja Root = 1.3.6.1.4.1.38760.3.3.4.2  , tad Documents.SetID = Extension |
|  | versionNumber | INT | 0..1 | Dokumenta versijas numurs | Documents.VersionNumber | Documents.VersionNumber |
|  | completionCode | CE | 0..1 | Netiek izmantots | Netiek aizpildīts | Netiek apstrādāts |
|  | storageCode | CE | 0..1 | Netiek izmantots | Netiek aizpildīts | Netiek apstrādāts |
|  | copyTime | TS | 0..1 | Netiek izmantots | Netiek aizpildīts | Netiek apstrādāts |
|  | subject | RCMR\_MT000002UV02\_LV01.Subject | 0..\* | Netiek izmantots | Netiek aizpildīts | Netiek apstrādāts |
|  | recordTarget | RCMR\_MT000002UV02\_LV01.RecordTarget | 1..\* | Pacienta ieraksts | Atbilstoši pacienta ieraksta datu struktūrai (skat. 6.1.6.1 DocumentRecordTarget (CDA dokumenta pacienta ieraksts)) | Pacients, kam tiek pievienots dokuments, tiek atrasts atbilstoši ( 6.1.6.1 DocumentRecordTarget (CDA dokumenta pacienta ieraksts)) aprakstītiem principiem |
|  | responsibleParty | RCMR\_MT000002UV02\_LV01.ResponsibleParty | 0..1 | Netiek izmantots | Netiek aizpildīts | Netiek apstrādāts |
|  | author | RCMR\_MT000002UV02\_LV01.Author1 | 1..\* | Dokumenta autors | Tiek aizpildīts no lauka Documents.Author (tabulā tiek glabāts xml atbilstoši dokumenta autora shēmai) | Tiek papildināts lauks Documents.Author (tabulā tiek glabāts xml atbilstoši dokumenta autora shēmai) |
|  | dataEnterer | RCMR\_MT000002UV02\_LV01.DataEnterer | 0..1 | Netiek izmantots | Netiek aizpildīts | Netiek apstrādāts |
|  | custodian | RCMR\_MT000002UV02\_LV01.Custodian | 1..1 | Medicīnas organizācija | Struktūrā tiek aizpildīts tikai organizācijas identifikators pēc šāda principa:  <custodian>  <assignedCustodian>  <representedOrganization>  <id root = Documents.MedicalInstitutionRoot extension = Document.MedicalInstitutionExtension | Datu bāzē tiek saglabāts tikai organizācijas identifikators pēc šāda principa:  <custodian>  <assignedCustodian>  <representedOrganization>  <id root = Documents.MedicalInstitutionRoot extension = Document.MedicalInstitutionExtension |
|  | informationRecipient | RCMR\_MT000002UV02\_LV01.InformationRecipient | 0..\* | Netiek izmantots | Netiek aizpildīts | Netiek apstrādāts |
|  | legalAuthenticator | RCMR\_MT000002UV02\_LV01.LegalAuthenticator | 0..1 | Netiek izmantots | Netiek aizpildīts | Netiek apstrādāts |
|  | authenticator | RCMR\_MT000002UV02\_LV01.Authenticator | 0..\* | Netiek izmantots | Netiek aizpildīts | Netiek apstrādāts |
|  | participant | RCMR\_MT000002UV02\_LV01.Participant | 0..\* | Netiek izmantots | Netiek aizpildīts | Netiek apstrādāts |
|  | inFulfillmentOf | RCMR\_MT000002UV02\_LV01.InFulfillmentOf | 0..\* | Netiek izmantots | Netiek aizpildīts | Netiek apstrādāts |
|  | documentationOf | RCMR\_MT000002UV02\_LV01.DocumentationOf | 0..\* | Netiek izmantots | Netiek aizpildīts | Netiek apstrādāts |
|  | relatedDocument | RCMR\_MT000002UV02\_LV01.RelatedDocument2 | 0..\* | Saistītie dokumenti | Struktūra aizpildīta šādi (visi lauki no tabulas DocumentLinkToDocument):  templateId= ParentDocumentTemplateId  id= parentDocumentId  code=CD vērtības, kas atbilst ParentDocumentTemplateId  setId=ParentDocumentSetId | Netiek apstrādāts |
|  | componentOf | RCMR\_MT000002UV02\_LV01.Component1 | 0..1 | Netiek izmantots | Netiek aizpildīts | Netiek apstrādāts |
|  | userComments | RCMR\_MT000002UV02\_LV01.Comment | 0..\* | Lietotāju komentāri par dokumentu. | Struktūra netiek aizpildīta getDocumentList gadījumā.  Struktūra tiek aizpildīta ar tabulas DocumentComments saturu par konkrēto dokumentu (pēc SetID).  Tiek aizpildīti šādi elementi:  Id (root = 1.3.6.1.4.1.38760.3.4.5.5 extension=DocumentCommentID)  commentIdx – DocumentCommentID  commentText = DocumentComments.Comments  commentAuthor = 6.1.6.3 Author (Autora struktūra) (komentāra autora struktūra, atbilstoši aizpildīšanas nosacījumiem) | Lietotāja komentāri tiek saglabāti tabulā DocumentComments.  Tiek saglabāti šādi elementi:  DocumentCommentID = Id.extension  DocumentCommentID = commentIdx  DocumentComments.Comments = commentText  DocumentComments.Author = commentAuthor |
| 30 | validationResults | RCMR\_MT000002UV02\_LV01.ValidationResult | 0..\* | Dokumenta automātiskas validācijas rezultāts (ja ir). | Tiek aizpildīts no tabulas DocumentErrors, kur ieraksti atbilst dokumenta identifikatoram.  Id – root = 1.3.6.1.4.1.38760.3.4.5.2  Extension = DocumentErrors.ErrorID  Code – kļūdas kods DocumentErrors.ErrorCode  errorData – kļūdas informācija DocumentErrors.ErrorData | Netiek apstrādāts |
| 31 | confidentialityRecord | RCMR\_MT000002UV02\_LV01.ConfidentialityRecord | 0..\* | Dokumenta, dokumenta sadaļas vai pacienta kartes aizliegums. | Skat. 6.1.6.2 ConfidentialityRecord (Aizliegumu struktūra) | Skat. 6.1.6.2 ConfidentialityRecord (Aizliegumu struktūra) |
| 32 | history | ED | 0..1 | Dokumenta izmaiņu vēstures informācija. Satur dokumenta izmaiņas datumu/laiku, izmaiņu ziņojuma identifikatoru un izmaiņas būtību | No tabulas Documents lauka history | Tiek aizpildīts automātiski. Saglabājot datus, šīs lauks datu struktūrā tiek ignorēts. |
| 33 | Signature | xmldsig | 0..1 | Dokumenta digitālais paraksts | No tabulas DocumentsLOB.Signature | Uz tabulu DocumentsLOB.Signature |

#### DocumentRecordTarget (CDA dokumenta pacienta ieraksts)

Identifikācija: DocumentRecordTarget

Datu struktūra balstīta uz HL7 RCMR\_MT000002UV02\_LV01.RecordTarget salikto datu tipu, kas satur Patient datu struktūru no COCT\_MT050000UV01.Patient.



32. attēls. RCMR\_MT000002UV02\_LV01.RecordTarget datu struktūra

Tiek aizpildīta datu struktūra Patient, ar tipu COCT\_MT050000UV01.Patient.

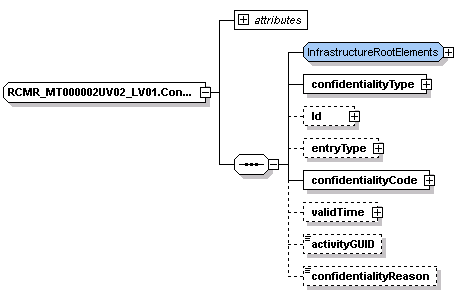
6.1‑15. tabula. Patient datu struktūra

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
|  | id | II | 1..\* | Pacienta identifikatori. Tiek atgriezti visi pacienta identifikatori no tabulas PatientIdentifications, kur lauks PatientGUID atbilst konkrēta pacienta kartes GUID laukam;  Root = IdentificationType  Extension = IdentificationCode |
|  | addr | AD | 0..\* | Netiek aizpildīts |
|  | telecom | TEL | 0..\* | Netiek aizpildīts |
|  | statusCode | CS | 1..1 | Tabulas NPatients lauks PersonStatus |
|  | effectiveTime | IVL\_TS | 0..1 | Netiek aizpildīts |
|  | confidentialityCode | CE | 0..1 | Netiek aizpildīts |
|  | veryImportantPersonCode | CE | 0..1 | Netiek aizpildīts |
|  | patientPerson | COCT\_MT030000UV04.Person | 0..1 | Netiek aizpildīts |
|  | patientNonPersonLivingSubject | COCT\_MT030000UV04.NonPersonLivingSubject | 0..1 | Netiek aizpildīts |
|  | providerOrganization | COCT\_MT150000UV01.Organization | 0..1 | Netiek aizpildīts |
|  | subjectOf | COCT\_MT050000UV01.Subject4 | 0..\* | Netiek aizpildīts |
|  | coveredPartyOf | COCT\_MT050000UV01.CoveredParty | 0..\* | Netiek aizpildīts |

#### ConfidentialityRecord (Aizliegumu struktūra)

Identifikācija: ConfidentialityRecord

Datu struktūra tiek izmantota dokumentu, medicīnas pamatdatu un pacienta karšu aizliegumu uzstādīšanai (funkcija 6.2.4 setAccessRights (Mainīt aizliegumus)).



33. attēls. RCMR\_MT000002UV02\_LV01.ConfidentialityRecord datu struktūra

Tiek aizpildīts tikai gadījumā, ja dokumenta šablona identifikators ir 1.3.6.1.4.1.38760.1.2.1.1 (Dokuments ir medicīnas pamatdatu kopsavilkuma dokuments).

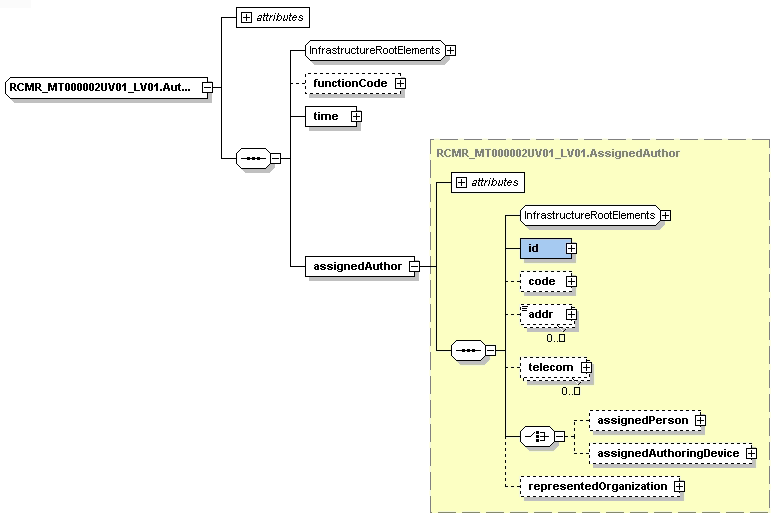
6.1‑16. tabula. ConfidentialityRecord datu struktūra

| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) | Saglabāt DB (set) | |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | confidentialityType | CS | 1..1 | Dokumenta, dokumenta sadaļas vai pacienta kartes aizliegums | Ja tiek aizpildīts no datu bāzes, tad tiek uzstādīts tikai ENTRYABS vērtība. | Ja lauka vērtība ir:  ENTRYABS: dokumenta sadaļa pēc datu bāzes atslēgas  ENTRYRL: dokumenta sadaļa pēc dokumenta iekšējas atslēgas  DOC: dokuments pēc identifikatora  CARD: pacienta karte pēc pacienta identifikatora |
|  | Id | II | 1..1 | Identifikators objektam, kuram tiek uzstādīts aizliegums | Ja tiek aizpildīts no datu bāzes, tad pamatdatu identifikators (no tabulām NPatientsAllergies; NPatientDiagnosis; NPatientMedicalDevices; NPatientMedications; NPatientWarnings; NPatientDisability). Visiem identifikatoriem ir kopējais root= 1.3.6.1.4.1.38760.3.4.5.3 | Identifikators, atbilstoši confidentialityType vērtībai |
|  | entryType | CS | 0..1 | Ja aizliegums tiek uzlikts dokumenta sadaļai (veselības pamatdatu gadījumā), tad šīs lauks norāda uz veselības pamatdatu veidu. | Veselības pamatdatu veids (entry) OID. | Tabulas  NPatientsAllergies = 1.3.6.1.4.1.38760.1.2.3.1; NPatientDiagnosis = 1.3.6.1.4.1.38760.1.2.3.2; NPatientMedicalDevices = 1.3.6.1.4.1.38760.1.2.3.3; NPatientMedications = 1.3.6.1.4.1.38760.1.2.3.4  NPatientWarnings = 1.3.6.1.4.1.38760.1.2.3.5  Izvēle atbilstoši klasifikatoram |
|  | confidentialityCode | CE | 1..1 | Aizlieguma vērtība | Tabulu NPatientsAllergies; NPatientDiagnosis; NPatientMedicalDevices; NPatientMedications; NPatientWarnings  NPatients  Documents  (atkarībā no konteksta) lauks AccessRight | Atbilstošas tabulas lauks AccessRights |
| 5. | validTime | IVL\_TS | 0..1 | Aizlieguma derīguma termiņš | Tabulu NPatientsAllergies; NPatientDiagnosis; NPatientMedicalDevices; NPatientMedications; NPatientWarnings  NPatients  Documents  (atkarībā no konteksta) lauks ModifDate (ja tiek mainīts aizliegums) | Atbilstošas tabulas lauks ModifDate |
| 6. | activityGUID | string | 0..1 | Aizlieguma uzstādīšanas ziņojuma ID | Tabulu NPatientsAllergies; NPatientDiagnosis; NPatientMedicalDevices; NPatientMedications; NPatientWarnings  NPatients  Documents  (atkarībā no konteksta) lauks ActivityGUID | Atbilstošas tabulas lauks ActivityGUID |
| 7. | confidentialityReason | string | 0..1 | Aizlieguma uzstādīšanas iemesls | Netiek aizpildīts | Atbilstošas tabulas lauks confidentialityReason |

#### Author (Autora struktūra)

Identifikācija: Author

Datu struktūra tiek izmantota dokumentu un lietotāja komentāru autora uzstādīšanai.



34. attēls. RCMR\_MT000002UV02\_LV01.author datu struktūra

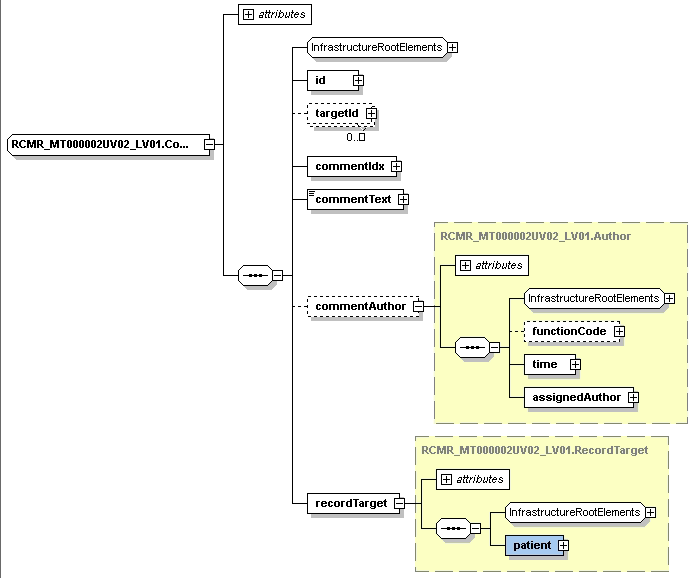
6.1‑17. tabula. Author1 datu struktūra

| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) | Saglabāt DB (set) | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1. | functionCode | CE | 0..1 | Netiek izmantots | Netiek aizpildīts | Netiek apstrādāts |
| 2. | time | TS | 1..1 | Autora dokumenta veidošanas laiks | No atbilstošas tabulas Author lauka | Tabulas Author lauks |
| 3. | assignedAuthor | RCMR\_MT000002UV02\_LV01.AssignedAuthor | 1..1 | Autora apakšstruktūra | No atbilstošas tabulas Author lauka  Tiek aizpildīts tikai autora identifikācijas lauks Id | Tabulas Author lauks |

#### Comment (Komentāra struktūra)

Identifikācija: Comment

Datu struktūra tiek izmantota dokumentu un pamatdatu komentāru glabāšanai un sūtīšanai.



35. attēls. RCMR\_MT000002UV02\_LV01.Comment datu struktūra

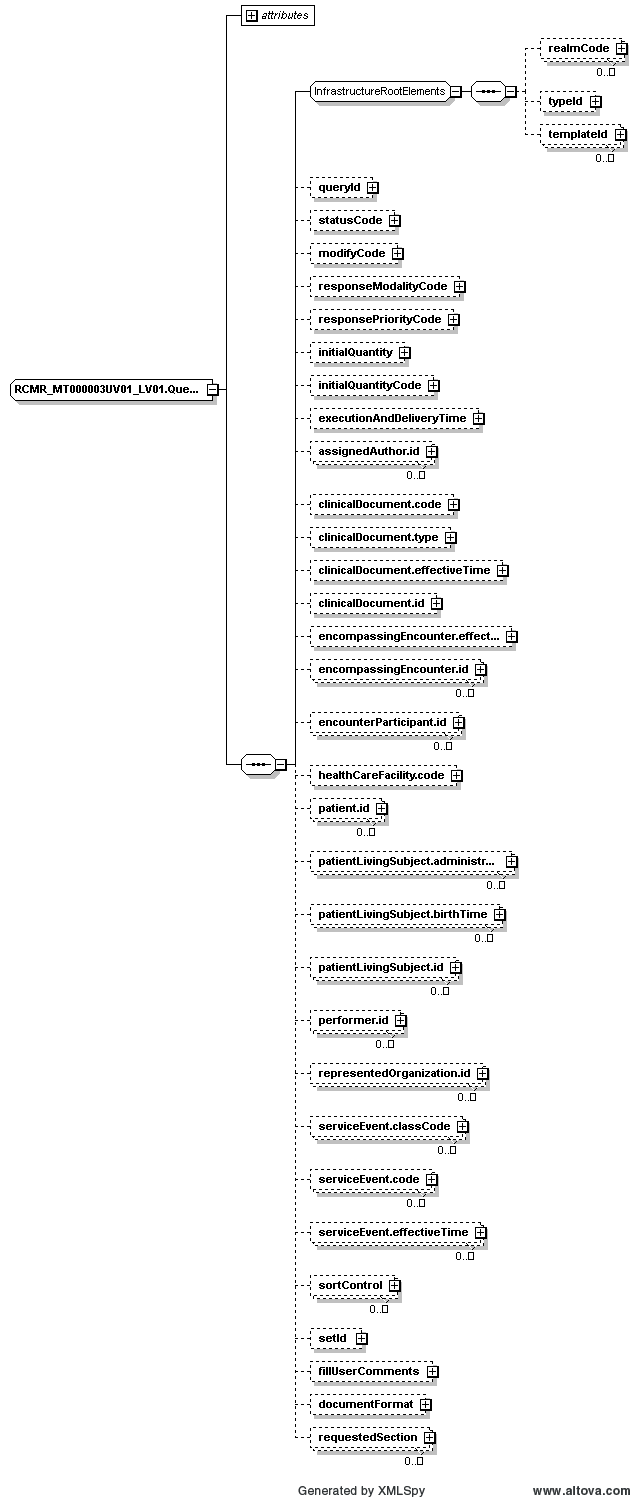
6.1‑18. tabula. Comment datu struktūra

| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) | Saglabāt DB (set) | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | id | II | 1..1 | Komentāra identifikators | Dokumenta komentāriem  Root = 1.3.6.1.4.1.38760.3.4.5.5  Pamatdatu komentāriem  Root =  1.3.6.1.4.1.38760.3.4.5.28  DocumentComments.DocumentCommentID | Dokumenta komentāriem  Root = 1.3.6.1.4.1.38760.3.4.5.5  Pamatdatu komentāriem  Root =  1.3.6.1.4.1.38760.3.4.5.28  Datu saglabāšanā netiek izmantots (jaunajiem komentāriem identifikators tiek ģenerēts no jauna) |
| 2 | targetId | II | 0..1 | Komentējamā objekta identifikators | Dokumentu komentāriem:  Root = 1.3.6.1.4.1.38760.3.4.5.1  Extension = Documents.DocumentID  Veselības pamatdatiem:  Root= viens no  1.3.6.1.4.1.38760.1.2.3.1  1.3.6.1.4.1.38760.1.2.3.1.1  1.3.6.1.4.1.38760.1.2.3.2  1.3.6.1.4.1.38760.1.2.3.2.1  1.3.6.1.4.1.38760.1.2.3.3  1.3.6.1.4.1.38760.1.2.3.3.1  1.3.6.1.4.1.38760.1.2.3.4  1.3.6.1.4.1.38760.1.2.3.4.1  1.3.6.1.4.1.38760.1.2.3.5 atkarībā no pamatdatu tipa | Ja komentārs tiek uzstādīts dokumentam,  DocumentComments.DocumentID, DocumentComments.SetId.  Ja komentārs tiek uzstādīts pamatdatiem, tad EntryTemplateOID un EntryID |
| 3 | commentIdx | INT | 1..1 | Komentāra indekss visu objekta komentāru sarakstā | Pirms atgriezt komentāru struktūru, komentāri tiek sakārtoti pēc modifDate laukā, un tiem piešķirti kārtas numuri atbilstoši šai secībai | Tiek ignorēts |
| 4 | commentText | ST | 1..1 | Komentāra teksts | DocumentComments.Comments | DocumentComments.Comments |
| 5 | commentAuthor | RCMR\_MT000002UV02\_LV01.Author | 0..1 | Komentāra autors (Skat. 6.1.6.3 Author (Autora struktūra) | DocumentComments.Author  Skat. struktūras apraksts | DocumentComments.Author  Skat. struktūras apraksts |
| 6 | recordTarget | RCMR\_MT000002UV02\_LV01.RecordTarget | 1..1 | 6.1.6.1 DocumentRecordTarget (CDA dokumenta pacienta ieraksts) | Skat. struktūras apraksts | Skat. struktūras apraksts |

### DocumentQueryByParameter (Dokumentu pieprasījums)

Identifikācija: DocumentQueryByParameter

Datu struktūra balstīta uz HL7 RCMR\_MT000003UV01\_LV01.QueryByParameter datu tipu.



36. attēls. RCMR\_MT000003UV01\_LV01.QueryByParameter datu struktūra

6.1‑19. tabula. QueryByParameter datu struktūra

| Nr. | Parametrs | Tips | Skaits | Apraksts:  getDocumentList | Apraksts:  getDocument-CDA | Apraksts:  getDocument-CCD |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | queryId | II | 0..1 | Pieprasījuma identifikators.  Root = 1.3.6.1.4.1.38760.3.4.5.33 (pieprasījuma tipa OID) | Pieprasījuma identifikators.  Root = 1.3.6.1.4.1.38760.3.4.5.6 (pieprasījuma tipa OID) | Pieprasījuma identifikators.  Root = 1.3.6.1.4.1.38760.3.4.5.6 (pieprasījuma tipa OID) |
| 2 | statusCode | CS | 0..1 | Iespējamās vērtības ACTUAL, ALL vai LATEST.  ACTUAL – tiek atgriezti tikai aktuālie ieraksti.  ALL – tiek atgriezti visi ieraksti, neatkarīgi no statusa  LATEST – katram dokumentam tiek atgriezta pēdēja dokumenta versija, neatkarīgi no tās statusa | Iespējāmas vērtības ACTUAL, ALL vai LATEST.  ACTUAL – tiek atgriezti tikai aktuālie ieraksti.  ALL – tiek atgriezti visi ieraksti, neatkarīgi no statusa  LATEST – katram dokumentam tiek atgriezta pēdēja dokumenta versija, neatkarīgi no tās statusa | Vai jāmeklē tikai aktuālie ieraksti, vai statuss nav jāņem vērā (atgriež visus). No klasifikatora Meklēšanas veidi. |
| 3 | modifyCode | CS | 0..1 | Netiek izmantots | Netiek izmantots | Netiek izmantots |
| 4 | responseModalityCode | CS | 0..1 | Netiek izmantots | Netiek izmantots | Netiek izmantots |
| 5 | responsePriorityCode | CS | 0..1 | Netiek izmantots | Netiek izmantots | Netiek izmantots |
| 6 | initialQuantity | INT | 0..1 | Netiek izmantots | Netiek izmantots | Netiek izmantots |
| 7 | initialQuantityCode | CE | 0..1 | Netiek izmantots | Netiek izmantots | Netiek izmantots |
| 8 | executionAndDeliveryTime | TS | 0..1 | Netiek izmantots | Netiek izmantots | Netiek izmantots |
| 9 | sampleSize | INT | 0..1 | Atgriežāma saraksta garums | Netiek izmantots | Netiek izmantots |
| 10 | offset | INT | 0..1 | Atgriežāma saraksta | Netiek izmantots | Netiek izmantots |
| 11 | assignedAuthor.id | RCMR\_MT000003UV01\_LV01.AssignedAuthor.id | 0..\* | Netiek izmantots | Netiek izmantots | Netiek izmantots |
| 12 | clinicalDocument.code | RCMR\_MT000003UV01\_LV01.ClinicalDocument.code | 0..1 | Medicīniskā dokumenta veidnes kods. Var saturēt kodu sarakstu.  Tiek aizpildīts elementu value saraksts (aizpildot CE vērtību ar  Code – templateOID  CodeSystem – OID:  1.3.6.1.4.1.38760.1.2.1 | Netiek izmantots | Aizpilda ar CCD veidnes tipu, norādot:  Code – DocumentTypeOID  CodeSystem – OID (OID klasifikatora OID). |
| 13 | clinicalDocument.type | RCMR\_MT000003UV01\_LV01.ClinicalDocument.type | 0..1 | Medicīniskā dokumenta veidnes tips. Var saturēt kodu sarakstu.  Tiek aizpildīts elementu value saraksts (aizpildot CE vērtību ar  Code – DocumentTypeOID  CodeSystem – OID:  1.3.6.1.4.1.38760.1.2.1 | Netiek izmantots | Netiek izmantots |
| 14 | clinicalDocument.effectiveTime | RCMR\_MT000003UV01\_LV01.ClinicalDocument.effectiveTime | 0..1 | Dokumenta meklēšanas laika intervāls. Tiek aizpildīts elementa value (IVL\_TS) apakšelements low un/vai high.  Low – meklēšanas laika intervāla sākums  High – meklēšanas laika intervāla beigas | Netiek izmantots | Veidojamā CCD dokumenta pamatdatu laika intervāls. Tiek atgriezti tikai tie pamatdatu ieraksti, kuri atbilst norādītā intervāla sākumam un beigām. |
| 15 | clinicalDocument.id | RCMR\_MT000003UV01\_LV01.ClinicalDocument.id | 0..1 | Meklējamā dokumenta identifikators. Var būt norādīts iespējamo identifikatoru saraksts. Tiek analizēts elements value, kas satur II (dokumenta identifikatoru) | Dokumenta identifikators. Var būt norādīts iespējamo identifikatoru saraksts. Tiek analizēts elements value, kas satur II (dokumenta identifikatoru). | Netiek izmantots |
| 16 | clinicalDocument.activity | uid | 0..1 | AddDocument izsaukuma activityId  Ja lauka vērtība ir norādīta, tā tiek izmantota dokumenta meklēšanā, un rezultāts tiek atgriezts, neskatoties uz pacienta uzstādītiem kartes piekļuves ierobežojumiem. | Netiek izmantots | Netiek izmantots |
| 17 | encompassingEncounter.effectiveTime | RCMR\_MT000003UV01\_LV01.EncompassingEncounter.effectiveTime | 0..1 | Netiek izmantots | Netiek izmantots | Netiek izmantots |
| 18 | encompassingEncounter.id | RCMR\_MT000003UV01\_LV01.EncompassingEncounter.id | 0..\* | Netiek izmantots | Netiek izmantots | Netiek izmantots |
| 19 | encounterParticipant.id | RCMR\_MT000003UV01\_LV01.EncounterParticipant.id | 0..\* | Netiek izmantots | Netiek izmantots | Netiek izmantots |
| 20 | healthCareFacility.code | RCMR\_MT000003UV01\_LV01.HealthCareFacility.code | 0..1 | Netiek izmantots | Netiek izmantots | Netiek izmantots |
| 21 | patient.id | RCMR\_MT000003UV01\_LV01.Patient.id | 0..\* | Pacients.  Lietotājam tiek pārbaudītas tiesības piekļūt pacienta datiem.  Pacientu identifikācija tiek atkodēta no patient.id struktūras lauka value. | Pacients. Pacienta identifikācijā tiek izmantots tikai pirmā patient.id saraksta vērtība. Pārējie ievadītie patient.id ir ignorēti. | Pacients.  Lietotājam tiek pārbaudītas tiesības piekļūt pacienta datiem.  Pacientu identifikācija tiek atkodēta no patient.id struktūras lauka value.  Pacienta identifikācijā tiek izmantots tikai pirmā patient.id saraksta vērtība. Pārējie ievadītie patient.id ir ignorēti. |
| 22 | patientLivingSubject.administrativeGenderCode | RCMR\_MT000003UV01\_LV01.PatientLivingSubject.administrativeGenderCode | 0..\* | Netiek izmantots | Netiek izmantots | Netiek izmantots |
| 23 | patientLivingSubject.birthTime | RCMR\_MT000003UV01\_LV01.PatientLivingSubject.birthTime | 0..\* | Netiek izmantots | Netiek izmantots | Netiek izmantots |
| 24 | patientLivingSubject.id | RCMR\_MT000003UV01\_LV01.PatientLivingSubject.id | 0..\* | Netiek izmantots | Netiek izmantots | Netiek izmantots |
| 25 | performer.id | RCMR\_MT000003UV01\_LV01.Performer.id | 0..\* | Ārsta identifikācija. Tiek aizpildīts datu struktūras lauks value, atribūti root un extension, kas atbilst meklējamā autora dokumentiem | Netiek izmantots | Netiek izmantots |
| 26 | representedOrganization.id | RCMR\_MT000003UV01\_LV01.RepresentedOrganization.id | 0..\* | Ārstniecības organizācija. Tiek aizpildīts lauks value ar organizācijas identifikatoru. | Netiek izmantots | Netiek izmantots |
| 27 | serviceEvent.classCode | RCMR\_MT000003UV01\_LV01.ServiceEvent.classCode | 0..\* | Netiek izmantots | Netiek izmantots | Netiek izmantots |
| 28 | serviceEvent.code | RCMR\_MT000003UV01\_LV01.ServiceEvent.code | 0..\* | Netiek izmantots | Netiek izmantots | Netiek izmantots |
| 29 | serviceEvent.effectiveTime | RCMR\_MT000003UV01\_LV01.ServiceEvent.effectiveTime | 0..\* | Netiek izmantots | Netiek izmantots | Netiek izmantots |
| 30 | sortControl | RCMR\_MT000003UV01\_LV01.SortControl | 0..\* | Netiek izmantots | Netiek izmantots | Netiek izmantots |
| 31 | setId | II | 0..1 | Dokumenta (visu versiju) numurs. Tiek aizpildīts ar vērtību, kas atbilst tabulas Documents laukam SetID | Netiek izmantots | Netiek izmantots |
| 32 | fillUserComments | BL | 0..1 | Netiek izmantots | Atgriežamā dokumenta ierakstam tiek pievienoti lietotāja dokumenta komentāri | Atgriežamā dokumenta ierakstam tiek pievienoti lietotāja dokumenta komentāri |
| 33 | documentFormat | CS | 0..1 | Netiek izmantots | Atgriežamā dokumenta formāts.  Vērtības:  XML vai HTML vai PDF. | Atgriežamā dokumenta formāts.  Vērtības:  XML vai HTML vai PDF. |
| 34 | requestedSection | II | 0..\* | Netiek izmantots | Netiek izmantots | CCD dokumenta satura kontrole. Var saturēt OIDus, kas atbilst CCD dokumenta sekcijām. Ja neaizpildīts, tiks veidots CCD, kas satur visas sekcijas, izņemot OID \*.58 |
| 35 | parentDocumentSetId | II | 0..1 | Saistīta dokumenta identifikators. Tiek atlasīti tikai dokumenti, kam saistīta dokumenta identifikators atbilst norādītam. | Saistīta dokumenta identifikators. Tiek atlasīti tikai dokumenti, kam saistīta dokumenta identifikators atbilst norādītam. | Netiek izmantots |

CCD dokumenta iespējamo sekciju saraksts:

1.3.6.1.4.1.38760.1.2.2.1 - Alerģijas

1.3.6.1.4.1.38760.1.2.2.2 - Diagnozes

1.3.6.1.4.1.38760.1.2.2.3 – Medicīnas iekārtas

1.3.6.1.4.1.38760.1.2.2.4 - Medikamenti

1.3.6.1.4.1.38760.1.2.2.5 - Brīdinājumi

1.3.6.1.4.1.38760.1.2.2.48 – Asins pārliešanas

1.3.6.1.4.1.38760.1.2.2.49 – Ķirurģiskās iejaukšanās

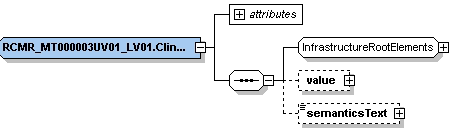
1.3.6.1.4.1.38760.1.2.2.58 – Visas diagnozes

1.3.6.1.4.1.38760.1.2.2.142 – Invaliditāte

Lauki, kuri tiek sūtīti uz select:

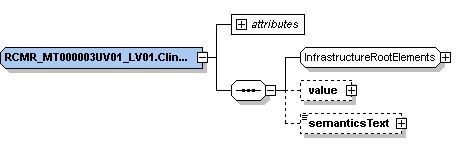
* query.clinicalDocumentcode
* query.clinicalDocumenteffectiveTime (low)
* query.clinicalDocumenteffectiveTime (high)
* Obligāts: query.statusCode.code
* query.clinicalDocumentid
* query.setId
* patientIds

#### ClinicalDocument.code



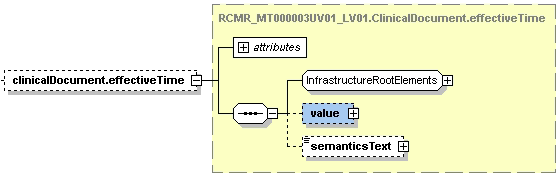
37. attēls. RCMR\_MT000003UV01\_LV01.ClinicalDocument.code datu struktūra

#### ClinicalDocument.type



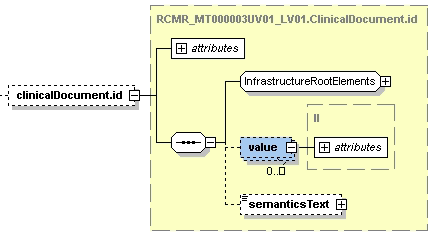
38. attēls. RCMR\_MT000003UV01\_LV01.ClinicalDocument.type datu struktūra

#### ClinicalDocument.effectiveTime



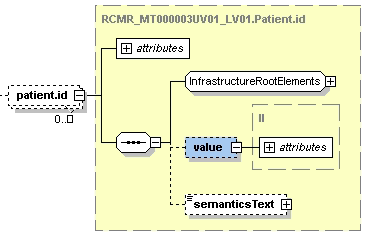
39. attēls. RCMR\_MT000003UV01\_LV01.ClinicalDocument.effectiveTime datu struktūra

#### ClinicalDocument.id



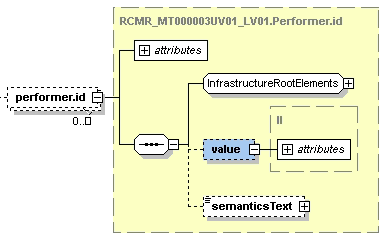
40. attēls. RCMR\_MT000003UV01\_LV01.ClinicalDocument.id datu struktūra

#### Patient.id



41. attēls. RCMR\_MT000003UV01\_LV01.Patient.id datu struktūra

#### Performer.id

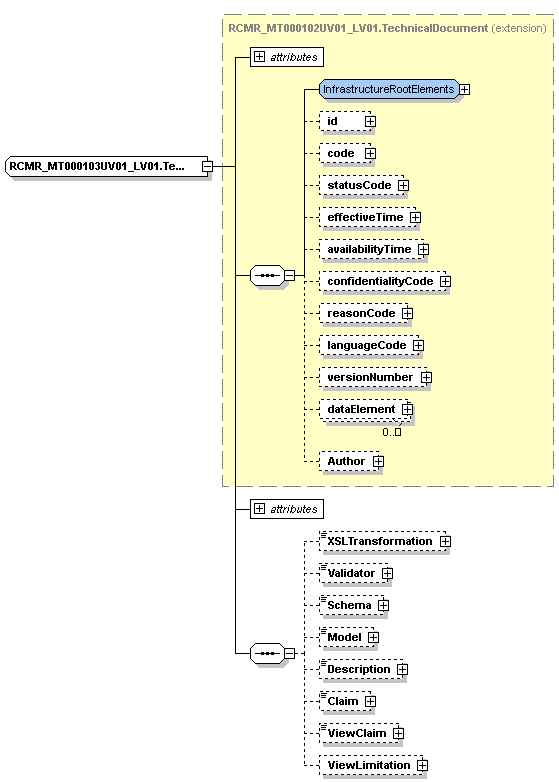


42. attēls. RCMR\_MT000003UV01\_LV01.Performer.id datu struktūra

### TemplateDocument (Veidnes struktūra)

Identifikācija: RCMR\_MT000103UV01\_LV01.TemplateDocument.

Datu struktūra ir RCMR\_MT000103UV01.TemplateDocument datu tips. Dokumenta šablona saglabāšana/pievienošana tiek kontrolēta ar updateMode atribūta palīdzību.



43. attēls. RCMR\_MT000103UV01\_LV01.TemplateDocument datu struktūra

6.1‑20. tabula. TemplateDocument datu struktūra

| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) | Saglabāt DB (set) | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | id | II | 1..1 | Veidnes identifikators | DocumentTemplates.DocumentTypeID (root)  DocumentTemplates.DocumentsTemplateOID (extension) | DocumentTemplates.DocumentsTypeID (root)  DocumentTemplates.DocumentsTemplateOID (extension) |
| 2 | code | CE | 1..1 | Dokumenta veidnes kods | Dokumenta kods, atbilstoši klasifikatoram 1.3.6.1.4.1.38760.2.21 Dokumentu tipi | Dokumenta kods, atbilstoši klasifikatoram 1.3.6.1.4.1.38760.2.21 Dokumentu tipi |
| 3 | statusCode | CS | 1..1 | Dokumenta veidnes statusa kods, atbilstoši klasifikatoram Veidņu statusi | Tiek aizpildīts ar izpildāmas aktivitātes kodu. |  |
| 4 | effectiveTime | TS | 1..1 | Dokumenta veidnes aktivitātes sākuma laiks. | DocumentTemplates .ValidFrom | DocumentTemplates .ValidFrom |
| 5 | availabilityTime | TS | 0..1 | Dokumenta veidnes aktivitātes beigu laiks (ja uzstādīts) | DocumentTemplates .ValidTo | DocumentTemplates .ValidTo |
| 6 | confidentialityCode | CE | 0..1 |  | Netiek aizpildīts | Netiek aizpildīts |
| 7 | reasonCode | CE | 0..1 |  | Netiek aizpildīts | Netiek aizpildīts |
| 8 | languageCode | CS | 0..\* |  | Netiek aizpildīts | Netiek aizpildīts |
| 9 | versionNumber | INT | 0..1 | Dokumenta veidnes versija | DocumentTemplates.Version | DocumentTemplates.Version |
| 10 | XSLTransformation | ED | 0..1 | XSL transformācija. | DocumentsTemplate.XSLTransformation.  Priekš getList neaizpilda. | DocumentsTemplate.XSLTransformation |
| 11 | [Validator](file://vilks/Projekti/VEC/EVK/03_Design/Schemas/Docs/RCMR_MT000103UV01_LV01.html#Link6B7) | ST | 0..1 | CDA validatora klases vārds. | DocumentTemplates.Validator.  Priekš getList neaizpilda. | DocumentTemplates.Validator |
| 12 | [Schema](file://vilks/Projekti/VEC/EVK/03_Design/Schemas/Docs/RCMR_MT000103UV01_LV01.html#Link6B8) | ED | 0..1 | CDA xsd shēma. | DocumentTemplates.Schema.  Priekš getList neaizpilda. | DocumentTemplates.Schema |
| 13 | [Model](file://vilks/Projekti/VEC/EVK/03_Design/Schemas/Docs/RCMR_MT000103UV01_LV01.html#Link6B9) | ED | 0..1 | CDA MDHT uml modelis. | DocumentTemplates.Model.  Priekš getList neaizpilda. | DocumentTemplates.Model |
| 14 | [Description](file://vilks/Projekti/VEC/EVK/03_Design/Schemas/Docs/RCMR_MT000103UV01_LV01.html#Link6BA) | ED | 0..1 | CDA veidnes apraksts. | DocumentTemplates.Description.  Priekš getList neaizpilda. | DocumentTemplates.Description |
| 15 | Author | RCMR\_MT000002UV02\_LV01.AssignedAuthor | 0..1 | Dokumenta veidnes autora struktūra | Aizpildās pēc vienotā principa skat. 6.1.6.3 Author (Autora struktūra) | Aizpildās pēc vienotā principa skat. 6.1.6.3 Author (Autora struktūra) |
| 16 | Claim | ED | 0..1 | Dokumenta pievienošanas Claim. Ja aizpildīts, dokumentu var pievienot tikai lietotājs ar norādīto Claim. | DocumentTemplates.Claim | DocumentTemplates.Claim |
| 17 | ViewClaim | ED | 0..1 | Dokumenta apskatīšanas Claim. Ja aizpildīts - dokumentu var apskatīt tikai lietotājs ar norādīto Claim. ViewClaim var saturēt sarakstu ar vairākām tiesībām, kas sadalīts ar semikolu “; ”  ViewClaim var saturēt pseidotiesību EvkRghtFamilyDoctor – kas atbilst dokumenta pieejāmībai konkrēta pacienta ģimenes ārstam | DocumentTemplates.ViewClaim | DocumentTemplates.ViewClaim |
| 18 | ViewLimitation | CS | 0..1 | Dokumenta redzamība aizlieguma gadījumā. Ja norādīts, ka dokuments ir redzams vienmēr, tad to neaiztiek pacienta pilna aizlieguma uzstādījuma gadījumā.  Iespējamās vērtības:  NULL - pēc noklusēšanas;  ALWAYS – redzams vienmēr  STD – standarta uzvedība | DocumentTemplates.ViewLimitation | DocumentTemplates.ViewLimitation |

### TemplateQueryByParameter (Veidņu pieprasījums)

Identifikācija: RCMR\_MT000003UV.QueryByParameter

Datu struktūra ir standarta HL7 RCMR\_MT000003UV.QueryByParameter saliktā datu tipa struktūra.

6.1‑21. tabula. QueryByParameter datu struktūra

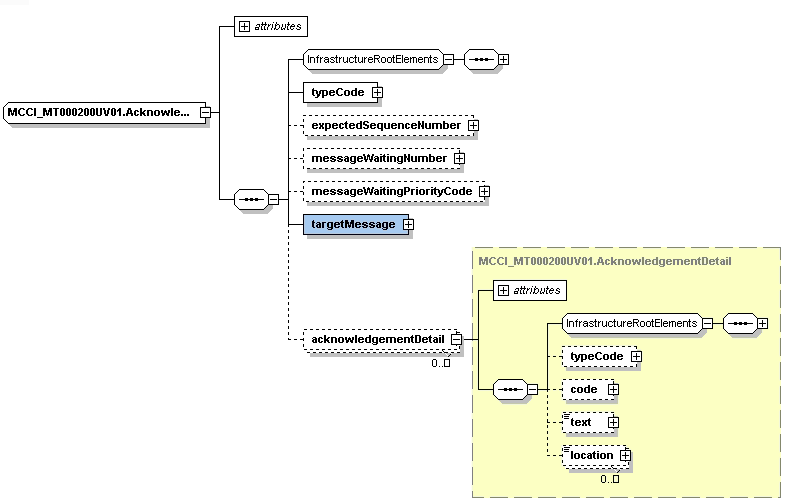
| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | | queryId | II | 0..1 | queryID.root – 1.3.6.1.4.1.38760.3.4.5.7 Pieprasījuma OID.  queryID. |
| 2 | | statusCode | CS | 0..1 | Vai jāmeklē tikai aktuālus datus, vai jāizmanto arī vēsturi. No klasifikatora Meklēšanas veidi.  All – jebkurš statuss  Actual – tikai statusā „Aktuāls” (pēc noklusējuma) |
| 3 | | clinicalDocument.code | RCMR\_MT000003UV.ClinicalDocument.code | 0..1 | Dokumenta veidnes tipa identifikators - TypeOID. |
| 4 | | clinicalDocument.effectiveTime | RCMR\_MT000003UV.ClinicalDocument.effectiveTime | 0..1 | Dokumenta veidnes derīguma termiņš |
| 5 | | clinicalDocument.id | RCMR\_MT000003UV.ClinicalDocument.id | 0..1 | Dokumenta veidnes identifikators - TemplateOID.  Obligāti aizpildāms getDocTemplate funkcijas gadījumā. |

### Ziņojumapmaiņas sistēmas koplietošanas datu struktūras

#### ack (Tehniskā atbildes datu struktūra)

Identifikācija: MCCI\_MT000200UV01.Acknowledgement

Struktūra tiek izmantota kā atbilde eksponējamās funkcijās, kurām netiek prasīta datu atgriešana (piem., dokumenta pievienošana, jaunas pacienta kartes veidošana)



44. attēls. MCCI\_MT000200UV01.Acknowledgement datu struktūra

Datu struktūrai nav definētas operācijas saglabāšanai datu bāzē, jo šī struktūra tiek tikai atgriezta no EVK citām sistēmām, un netiek pieņemta no ārējām sistēmām.

6.1‑22. tabula. Acknowledgment datu struktūra

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | typeCode | CS | 0..1 | Ziņojuma apstrādes rezultāts.  Tiek atgriezts viens no iespējamiem ziņojuma apstrādes rezultātiem:  AA Ziņojums pieņemts  AE Ziņojums ir kļūdains  AR Ziņojums nav pieņemts |
| 2 | expectedSequenceNumber | INT | 0..1 | Netiek aizpildīts |
| 3 | messageWaitingNumber | INT | 0..1 | Netiek aizpildīts |
| 4 | messageWaitingPriorityCode | CD | 0..1 | Netiek aizpildīts |
| 5 | targetMessage | MCCI\_MT000200UV01.TargetMessage | 1..1 | Akceptējama ziņojuma identifikācija.  Tiek aizpildīts lauks id (lauks tiek aizpildīts no atbildamā ziņojuma transporta aploksnes MCCI\_MT000100UV01.Message elements id). |
| 6 | acknowledgementDetail | MCCI\_MT000200UV01.AcknowledgementDetail | 0..\* | Rezultāta apraksts.  Tiek aizpildīts ar ziņojuma apstrādes rezultātu (skat. tabulu zemāk). |

acknowledgementDetail struktūras apraksts:

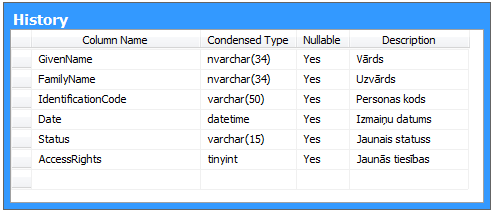
6.1‑23. tabula. AcknowledgmentDetail datu struktūra

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | TypeCode | CS | 0..1 | Netiek izmantots |
| 2 | Code | CD | 0..1 | Ziņojuma apstrādes rezultāts  Ziņojuma kods atbilstoši šī dokumenta definētas kļūdas kodu tabulas (skat. 6.13.8 ) |
| 3 | Text | ED | 0..1 | Saraksts ar kļūdām no ziņojuma apstrādes funkcijas.  Aizpildīšanas piemēri ir stack trace sistēmas kļūdas gadījumā, vai validācijas ziņojums no dokumenta modeļa |
| 4 | Location | ST | 0..\* | Netiek izmantots |

### History

Identifikācija: History

Struktūra tiek izmantota, lai pie objekta fiksētu tā izmaiņas, kas attiecas uz statusa maiņu un tiesību deskriptora maiņu. Struktūra aktuāla CDA dokumentu ierakstiem un veselības pamatdatu ierakstiem.



45. attēls. History datu struktūra

SQL Server pusē saņemot aizpildītu History struktūru, tiek paslēpti sensitīvie dati (Vārds, Uzvārds, Personas kods).

### DetailedRole

Identifikācija: DetailedRole

<?xml version="1.0" encoding="UTF-8" standalone="no"?>

<!-- edited with XMLSpy v2011 rel. 3 sp1 (http://www.altova.com) by Vilnis Sjomkans (AS Datorzinibu centrs) -->

<xs:schema xmlns:ex="urn:hl7-org/v3-example" xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns="urn:hl7-org:v3" targetNamespace="urn:hl7-org:v3" elementFormDefault="qualified">

<xs:include schemaLocation="../coreschemas/infrastructureRoot.xsd"/>

<xs:include schemaLocation="PRPA\_MT201303UV02\_LV01.xsd"/>

<xs:complexType name="LVAU\_MT000105UV01\_LV01.DetailedRoleResponse">

<xs:sequence>

<xs:group ref="InfrastructureRootElements"/>

<xs:element name="id" type="II" minOccurs="1" maxOccurs="unbounded"/>

<xs:element name="detailedRole" type="CS" minOccurs="1" maxOccurs="1"/>

<xs:element name="delegation" type="PRPA\_MT201303UV02\_LV01.Delegate" nillable="true" minOccurs="0" maxOccurs="unbounded"/>

<xs:element name="delegated" type="PRPA\_MT201303UV02\_LV01.Delegate" nillable="true" minOccurs="0" maxOccurs="unbounded"/>

<xs:element name="majority" type="PRPA\_MT201303UV02\_LV01.Majority" nillable="true" minOccurs="0" maxOccurs="1"/>

</xs:sequence>

<xs:attributeGroup ref="InfrastructureRootAttributes"/>

<xs:attribute name="nullFlavor" type="NullFlavor" use="optional"/>

<xs:attribute name="classCode" type="ActClassObservation" use="required"/>

<xs:attribute name="moodCode" type="ActMood" use="optional" fixed="EVN"/>

</xs:complexType>

<xs:complexType name="LVAU\_MT000105UV01\_LV01.DetailedRoleRequest">

<xs:sequence>

<xs:group ref="InfrastructureRootElements"/>

<xs:element name="userId" type="II" minOccurs="1" maxOccurs="1"/>

<xs:element name="targetCard" type="II" minOccurs="1" maxOccurs="1"/>

</xs:sequence>

<xs:attributeGroup ref="InfrastructureRootAttributes"/>

<xs:attribute name="nullFlavor" type="NullFlavor" use="optional"/>

<xs:attribute name="classCode" type="RoleClass" use="optional" fixed="BIRTHPL"/>

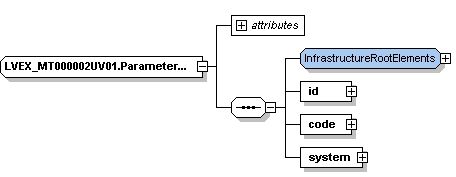
</xs:complexType>

</xs:schema>

### ProfileSettings

#### LVEX\_MT000002UV01.ParameterList

Identifikators: LVEX\_MT000002UV01



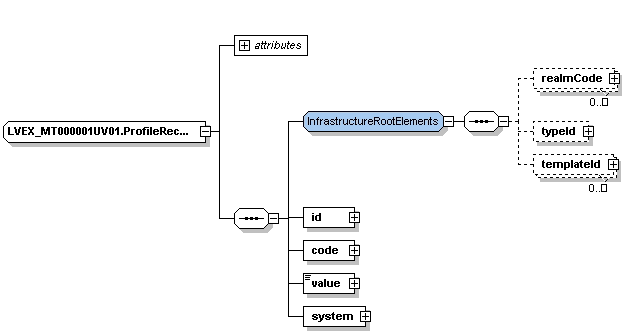
46. attēls. LVEX\_MT000002UV01.ParameterList datu struktūra

6.1‑24. tabula. Elementu apraksts

| Elements | Tips | Skaits | Apraksts |
| --- | --- | --- | --- |
| id | II | 1..1 | Pacienta identifikācija |
| code | CS | 0..1 | Pieprasāmā pacienta profila elementa kods |
| system | CS | 0..1 | Pieprasījuma pacienta profila sistēmas kods |

#### LVEX\_MT000001UV01.ProfileRecord

Identifikators: LVEX\_IN000001UV01



47. attēls. LVEX\_MT000001UV01.ProfileRecord datu struktūra

6.1‑25. tabula. Elementu apraksts

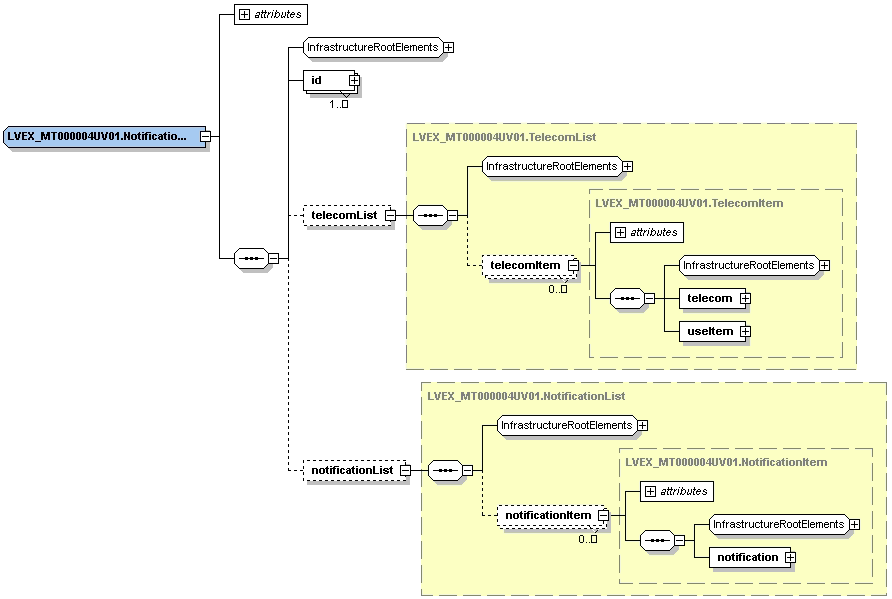
| Elements | Tips | Skaits | Apraksts |
| --- | --- | --- | --- |
| id | II | 1..1 | Pacienta identifikācija. |
| code | CS | 1..1 | Pacienta profila elementa kods. |
| value | ED | 1..1 | Pacienta profila elementa vērtība (Base64 kodēta virkne). EVK IS neinterpretē un neapstrādā šo informāciju, tā tikai tiek glabāta sistēmā. |
| system | CD | 1..1 | Informācijas sistēmas kods |

### NotificationConfiguration (Automātisko paziņojumu uzstādījumi)

Identifikācija: LVEX\_MT000004UV01.NotificationConfiguration.

Datu struktūra tiek izmantota:

1. Automātisko paziņojumu sūtīšanas uzstādījumu atgriešanai;
2. Automātisko paziņojumu sūtīšanas uzstādījumu maiņai (telekomunikācijas objektu komplekts un pacienta ziņojumu izvēle).



48. attēls. LVEX\_MT000004UV01.NotificationConfiguration datu struktūra

6.1‑26. tabula. NotificationConfiguration datu struktūra

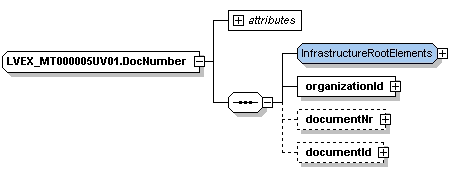
| Nr | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) | Saglabāt DB (set) |
| --- | --- | --- | --- | --- | --- | --- |
| 1. | id | II | 1..\* | Pacienta identifikācija. Telekomunikāciju saraksta elementi ir saistīti ar vienu pacientu | PatientIdentifications. IdentificationType un IdentificationCode | Netiek apstrādāts |
| 2. | telecomList |  | 1..1 | Pacienta komunikāciju veidi, kas tiek vai var tikt izmantoti automātisko ziņojumu sūtīšanai | 2. | telecomList |
| 3. | telecomItem |  | 0..\* |  |  |  |
|  | telecom | TEL\_LV01 | 1..1 | Komunikācijas veids un vērtība (e-pasts vai telefona numurs SMS sūtīšanai) | PatientContacts. ContactTypeCode un Value | PatientContacts.ContactTypeCode un Value |
|  | useItem | BL | 1..1 | Atzīme, vai konkrēta komunikācijas adrese tiek izmantota automātisko ziņojumu sūtīšanai | PatientContacts. useForNotifications | PatientContacts. useForNotifications |
| 3. | notificationList | 1..1 | Apziņošanas servisa ziņojumu grupas | | | |
| 4. | notificationItem | 0..\* | Atribūts updateMode norāda, vai ieraksts ir jauns, labots vai dzēšams. updateMode vērtības skatīt CR1.PPA dokumentā | | | |
|  | notification | CD | 1..1 | Paziņojuma veids (kods un nosaukums)  Pēc klasifikatora  1.3.6.1.4.1.38760.2.22 – (Apziņošanas servisa ziņojumu grupas) | PatientNotifications lauki  NotificationCode  NotificationCodeSystem  NotificationDisplayValue | PatientNotifications lauki  NotificationCode  NotificationCodeSystem  NotificationDisplayValue |

### DocNumber (Dokumenta numura iegūšanas sekvence)

Identifikācija: DocNumber.

Datu struktūra tiek izmantota:

1. Lai saņemtu jauna medicīniskā dokumenta identifikācijas numuru latvija.lv e-pakalpojumā.



49. attēls. LVEX\_MT000005UV01.DocNumber datu struktūra

6.1‑27. tabula. LVEX\_MT000005UV01.DocNumber datu struktūra

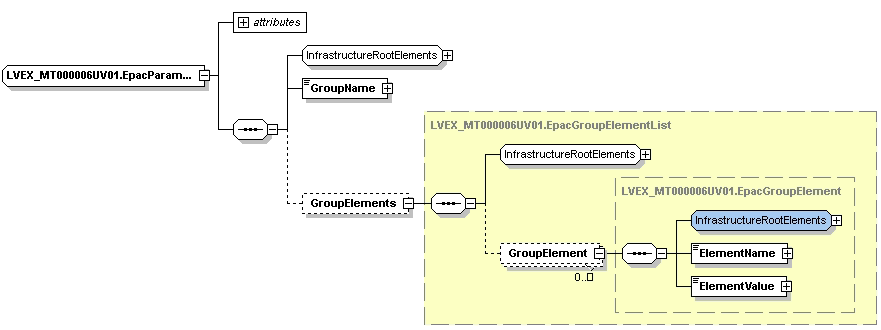
| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) |
| --- | --- | --- | --- | --- | --- |
| 1. | organizationId | II | 1..1 | Dokumenta numuru pieprasoša ārstniecības iestāde.  Atbilstoši ārstniecības iestādes klasifikatoram:  Root: 1.3.6.1.4.1.38760.2.23  Extension: Ārstniecības iestādes kods | Tabula DocumentNumberSequence.Custodian |
| 2. | documentNr | INT | 0..1 | Iegūts jauns dokumenta numurs | Aizpildīts ar jaunizveidoto dokumenta numuru |
| 3. | documentId | II | 0..1 | Iegūts jauns dokumenta identifikators (pēc EVK prasībām) | Aizpildīts ar jaunizveidoto dokumenta numuru, atbilstoši EVK prasībām (II formātā) |

### EVKEpacParameters (EVK e-pakalpojumu parametri)

Identifikācija: LVEX\_MT000006UV01.EpacParameterGroup

Datu struktūra tiek izmantota:

1. Lai atlasītu EVK e-pakalpojumos izmantojamos parametrus.



50. attēls. LVEX\_MT000006UV01.EpacParameterGroup datu struktūra

6.1‑28. tabula. LVEX\_MT000006UV01.EpacParameterGroup datu struktūra

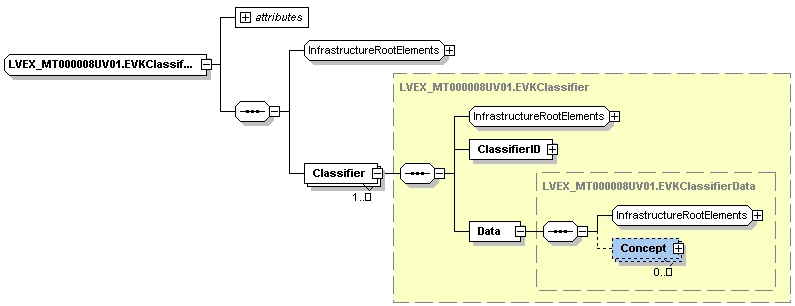
| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) |
| --- | --- | --- | --- | --- | --- |
| 1. | EpacParameterGroup |  |  | LVEX\_MT000006UV01.EpacParameterGroup. Apkopojošais objekts | - |
| 2. | GroupName | ST | 1..1 | Pieprasāmas vai atgriežamas e-pakalpojumu parametru grupas nosaukums | Parameters.Group |
| 3. | GroupElements |  | 0..1 | LVEX\_MT000006UV01.EpacGroupElementList. Apkopojošais objekts |  |
| 4. | GroupElement |  | 0..\* | LVEX\_MT000006UV01.EpacGroupElement. Individuālais e-pakalpojumu parametrs | Parameters |
| 5. | ElementName | ST | 1..1 | Parametra nosaukums. | Parameters.Name |
| 6. | ElementValue | ST | 1..1 | Parametra vērtība | Parameters.Value |

### GetClassifierValues (EVK saglabāto klasifikatoru kopiju vērtības)

Identifikācija: LVEX\_MT000008UV01.EVKClassifierList

Datu struktūra tiek izmantota:

1. Lai atlasītu EVK saglabāto klasifikatoru kopoju vērtības.



51. attēls. LVEX\_MT000008UV01.EVKClassifierList datu struktūra

6.1‑29. tabula. LVEX\_MT000008UV01.EVKClassifierList datu struktūra

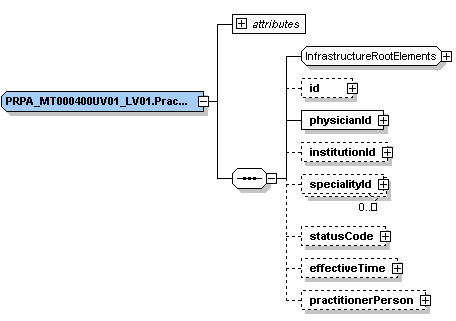
| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) |
| --- | --- | --- | --- | --- | --- |
| 1. | Classifier |  | 1..\* | Grupējošais elements (apvieno klasifikatora datus un klasifikatora OID) |  |
| 2. | ClassifierID | II | 1..1 | Klasitfikatora OID un klasifikatora nosaukums. |  |
| 3. | Data |  | 1..1 | Klasifikatora datu komplekts (katrā klasifikatora ierakstā jābūt 1 klasifikatoru datu komplektam) |  |
| 4. | Concept | CE | 0..\* | Klasifikatora kodi un vērtības (tiek izmantoti lauki:  Code: klasifikatora kods  codeSystem: klasifikatora OID  codeSystemVersion: klasifikatora versija  displayValue: lietotājam redzama vērtība |  |

### Practitioner (Ārstniecības plāna ārstniecības personu saraksts)

Identifikācija: PRPA\_MT000400UV01\_LV01.Practitioner

Datu struktūra tiek izmantota:

1. Sastāvdaļā ārstniecības plāna struktūrai.



52. attēls. PRPA\_MT000400UV01\_LV01.Practitioner datu struktūra

6.1‑30. tabula. PRPA\_MT000400UV01\_LV01.Practitioner datu struktūra

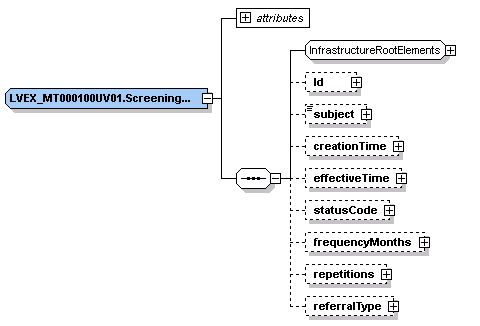
| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) |
| --- | --- | --- | --- | --- | --- |
| 1. | id | II | 0..1 | Ārstniecības plāna ārstniecības personas identifikators | hpPractitioner. PractitionerID |
| 2. | physicianId | CD | 1..1 | Ārstniecības personas identifikācijas kods (klasifikatora 1.3.6.1.4.1.38760.2.1 vērtība) | hpPractitioner. PhysicianID |
| 3. | institutionId | CD | 1..1 | Ārstniecības iestādes ID, atbilstoši ĀI (1.3.6.1.4.1.38760.2.23) klasifikatoram | hpPractitioner. InstitutionID |
| 4. | specialityId | CD | 0..\* | Ārstniecības personas specialitātes kodi, atbilstoši AP specialitāšu (1.3.6.1.4.1.38760.2.38) klasifikatoram | hpPractitioner. SpecialityID |
| 5. | statusCode | CS | 0..1 | Ārstniecības personas ārstniecības plāna kods | hpPractitioner. Status atļautas vērtības Active Cancelled |
| 6. | effectiveTime | IVL\_TS | 0..1 | Ārstniecības personas aktivitātes termiņš | hpPractitioner.No hpPractitioner.Lidz |

### ScreeningConfigurationRequest (Skrīninga konfigurāciju saraksta iegūšana)

Identifikācija: LVEX\_MT000100UV01.ScreeningConfigurationRequest

Datu struktūra tiek izmantota:

1. GetScreeningConfigurationsList.



53. attēls. LVEX\_MT000100UV01.ScreeningConfigurationRequest datu struktūra

6.1‑31. tabula. LVEX\_MT000100UV01.ScreeningConfigurationRequest datu struktūra

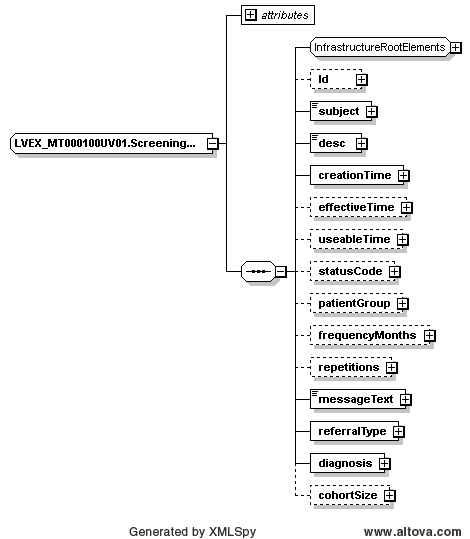
| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) |
| --- | --- | --- | --- | --- | --- |
| 1. | Id | II | 0..1 | Skrīninga konfigurācijas identifikators | ScreeningDefinitions.ScreeningDefinitionID |
| 2. | subject | ED | 0..1 | Skrīninga konfigurācijas nosaukums vai tā daļa | ScreeningDefinitions.Subject |
| 3. | creationTime | IVL\_TS | 0..1 | Skrīninga kongfigurācijas veidošanas datums (no-līdz) | ScreeningDefinitions.CreationTime |
| 4. | effectiveTime | IVL\_TS | 0..1 | Skrīninga aktivitātes datums (no-līdz) | ScreeningDefinitions.RunFrom ScreeningDefinitions.RunTill |
| 5. | statusCode | CS | 0..1 | Skrīninga konfigurācijas statusa kods | ScreeningDefinitions.statusCode |
| 6. | frequencyMonths | INT | 0..1 | Skrīninga izpildes biežums | ScreeningDefinitions.MonthsInterval |
| 7. | repetitions | INT | 0..1 | Skrīngina konfigurācijā ierakstītais atkārtojumu skaits | ScreeningDefinitions.RepetionCount |
| 8. | referralType | CD | 0..1 | Skrīningā iekļauts pakalpojums | ScreeningDefinitions.ReferralType |

### ScreeningConfiguration (Skrīninga konfigurācija)

Identifikācija: LVEX\_MT000100UV01.ScreeningConfiguration

Datu struktūra tiek izmantota:

1. SetScreeningConfiguration.



54. attēls. LVEX\_MT000100UV01.ScreeningConfiguration datu struktūra

6.1‑32. tabula. LVEX\_MT000100UV01.ScreeningConfiguration datu struktūra

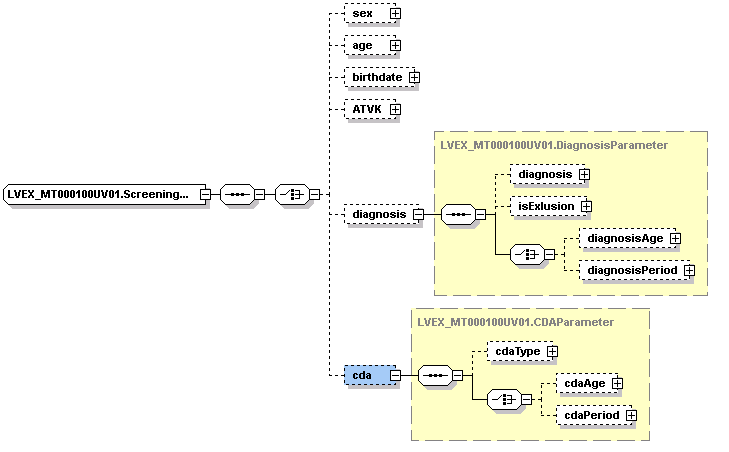
| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) |
| --- | --- | --- | --- | --- | --- |
| 1. | Id | II | 0..1 | Skrīninga konfigurācijas identifikators | ScreeningDefinitions.ScreeningDefinitionID |
| 2. | subject | ED | 1..1 | Skrīninga konfigurācijas nosaukums | ScreeningDefinitions.Subject |
| 3. | desc | ED | 1..1 | Skrīninga konfigurācijas apraksts | ScreeningDefinitions.Description |
| 4. | creationTime | TS | 1..1 | Skrīninga konfigurācijas veidošanas datums | ScreeningDefinitions.CreationTime  Netiek saglabāts datu bāzē no pieprasījuma datiem – tiek aizpildīts ar pašreizējo datumu. |
| 5. | effectiveTime | IVL\_TS | 0..1 | Skrīninga aktivitātes datums (no-līdz). Ja nav norādīts – nav augšējās robežas. | ScreeningDefinitions. useableTimeFrom ScreeningDefinitions. useableTimeTill |
| 6. | useableTime | INT | 0..1 | Skrīninga izveidoto nosūtījumu derīguma termiņš mēnešos | ScreeningDefinitions.useableTime |
| 7. | statusCode | CS | 0..1 | Skrīninga konfigurācijas statusa kods.  Iespējamās vērtības:  Active - veiksmīgi saglabāta skrīninga konfigurācija  Updating – tiek rēķināts pacientu aptuvenais daudzums  Cancelled – skrīninga konfigurācija ir dzēsta un vairs netiek izmantota | ScreeningDefinitions.statusCode |
| 8. | patientGroup | LVEX\_MT000100UV01.ScreeningParameters | 0..1 | Skrīninga pacientu grupu noteicošo meklēšanas parametru komplekts | Saraksts ar neierobežotu (0..\*) daudzumu vērtību ar tipu ScreeningParameter skat. 6.1.20.1 |
| 9. | frequencyMonths | INT | 0..1 | Skrīninga izpildes biežums.  Nevar būt negatīvs.  Ja nav norādīts – 1 mēnesis | ScreeningDefinitions.MonthsInterval |
| 10. | repetitions | INT | 0..1 | Skrīninga konfigurācijā ierakstītais atkārtojumu skaits.  Nevar būt negatīvs.  Ja nav norādīts – 1 reize. Ja norādīts 0 – tad nav norādīts atkārtojumu skaits | ScreeningDefinitions.RepetionCount |
| 11. | messageText | ED | 1..1 | Skrīninga rezultāta izveidotais paziņojuma teksts | ScreeningDefinitions.Message |
| 12. | referralType | CD | 1..1 | Skrīningā iekļautais pakalpojums (OID 105) | ScreeningDefinitions.ReferralType |
| 13. | diagnosis | CD | 1..1 | Nosūtījumā iekļautā diagnoze (OID 159) | ScreeningDefinitions.Diagnosis |
| 14. | cohortSize | INT | 0..1 | Skrīningā iekļauts pacientu daudzums (vērtība nav precīza, un tiek rēķināta fona režīmā) | ScreeningDefinitions.PatientCount |

#### ScreeningParameter (Skrīninga pacientu grupas atlases parametrs)

Identifikācija: LVEX\_MT000100UV01.ScreeningParameter

Datu struktūra tiek izmantota:

1. Sastāvdaļa skrīninga konfigurācijas struktūrai.



55. attēls. LVEX\_MT000100UV01.ScreeningParameter datu struktūra

6.1‑33. tabula. LVEX\_MT000100UV01.ScreeningParameter datu struktūra

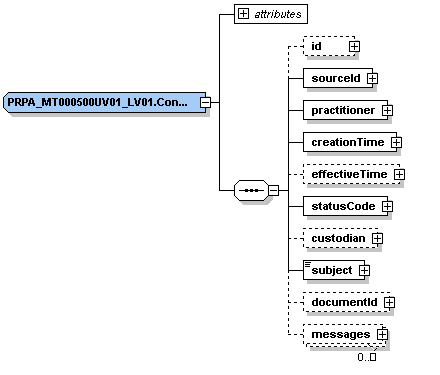
| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) |
| --- | --- | --- | --- | --- | --- |
| 1. | Id | II | 0..1 | Skrīninga parametra identifikators | ScreeningDefinitions.Parameters (visi skrīninga konfigurācijas parametri tiek glabāti kā viens xml kolonnā parameters) |
|  |  |  |  | **Izvēle no šādiem parametriem:** | (visi skrīninga konfigurācijas parametri tiek glabāti kā viens xml kolonnā parameters) |
| 2. | sex | CD | 0..1 | Skrīninga pacientu grupas dzimums (OID 112) | ScreeningDefinitions.Parameters |
| 3. | age | REAL | 0..1 | Skrīninga pacientu grupas vecums | ScreeningDefinitions.Parameters |
| 4. | birthdate | IVL\_TS | 0..1 | Skrīninga pacientu grupas dzimšanas datums (intervāls no-līdz) | ScreeningDefinitions.Parameters |
| 5. | ATVK | II | 0..1 | Skrīninga pacientu grupas ģeogrāfiskās lokalizācijas kods (ATVK - 1.3.6.1.4.1.38760.3.4.5.13). Pārbaude pēc (klasifikatora OID 2.8, 2.113) | ScreeningDefinitions.Parameters |
| 6. | diagnosis | LVEX\_MT000100UV01.DiagnosisParameter | 0..1 | Skrīninga pacientu grupas diagnozes struktūra. | ScreeningDefinitions.Parameters |
|  | diagnosis | CD | 1..1 | Skrīninga pacientu grupas diagnoze (OID 159) |  |
|  | isExclusion | BL | 0..1 | Vai pacients ir iekļauts skrīningā, ja viņam ir uzstādīta konkrēta diagnoze, vai, ja šī diagnoze ir, tad šis pacients netiek iekļauts skrīningā |  |
|  | diagnosisAge | INT | 0..1 | Laiks (dienās), kas pagāja no diagnozes uzstādīšanas (datumu intervāls relatīvs uz skrīninga procesa palaišanu). Ja ir aizpildīts šīs lauks, lauks diagnosisPeriod nevar būt aizpildīts. |  |
|  | diagnosisPeriod | IVL\_TS | 0..1 | Diagnozes uzstādīšanas periods (fiksēts datumu intervāls). Ja ir aizpildīts šīs lauks, lauks diagnosisAge nevar būt aizpildīts |  |
| 7. | cda | LVEX\_MT000100UV01.CDAParameter | 0..1 | Skrīninga pacientu grupas cda dokumentu struktūra | ScreeningDefinitions.Parameters |
|  | cdaType | CD | 1..1 | Skrīninga pacientu grupas cda dokumenta tips (OID 21) |  |
|  | cdaAge | INT | 0..1 | Laiks (dienās), kas pagāja no cda dokumenta veidošanas (datumu intervāls relatīvs uz skrininga procesa palaišanu). Ja ir aizpildīts šīs lauks, lauks cdaPeriod nevar būt aizpildīts |  |
|  | cdaPeriod | IVL\_TS | 0..1 | cda dokumenta izveides periods (fiksēts datumu intervāls). Ja ir aizpildīts šīs lauks, lauks cdaAge nevar būt aizpildīts |  |

### ConsultationRecord (Konsultācijas ieraksts)

Identifikācija: PRPA\_MT000500UV01\_LV01.ConsultationRecord

Datu struktūra tiek izmantota:

1. GetConsultationsList
2. SetConsultation



56. attēls. PRPA\_MT000500UV01\_LV01.ConsultationRecord datu struktūra

6.1‑34. tabula. PRPA\_MT000500UV01\_LV01.ConsultationRecord datu struktūra

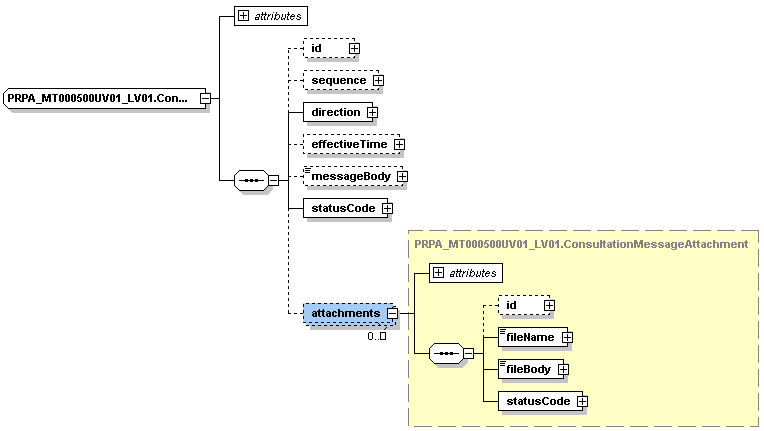
| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) |
| --- | --- | --- | --- | --- | --- |
| 1. | id | II | 0..1 | Konsultācijas ieraksta identifikators OID 1.3.6.1.4.1.38760.3.4.5.43 | ConsultationRecord.ConsultationRecordId |
| 2. | sourceId | II | 1..1 | Pacienta OID identifikators, atbilstoši:  Personas kods: 1.3.6.1.4.1.38760.3.1.1  Ārstniecības personas id, atbilstoši klasifikatoram OID 1.3.6.1.4.1.38760.2.1 | ConsultationRecord.SourceType (root), ConsultationRecord.SourceCode (extension) |
| 3. | practitioner | PRPA\_MT000400UV01\_LV01.Practitioner | 1..1 | Ārstniecības personas datu struktūra. Skat. 6.1.18 | ConsultationRecord.PractitionerCode |
| 4. | creationTime | TS | 1..1 | Ieraksta izveidošanas datums. Saglabājot ierakstu, tiek ierakstīts sistēmas datums un datu struktūrā ierakstītā vērtība tiek ignorēta | ConsultationRecord.CreationTime |
| 5. | effectiveTime | IVL\_TS | 0..1 | Konsultācijas termiņš (no sākuma līdz slēgšanai vai pabeigšanai). Saglabājot ierakstu, lauks tiek pārrēķināts no ziņojumiem, struktūrā ierakstītā vērtība tiek ignorēta | ConsultationRecord.EffectiveTimeFrom ConsultationRecord.EffectiveTimeTo |
| 6. | statusCode | CS | 1..1 | Konsultācijas statuss var būt viens no:  Active  Read  Answered  Complete | ConsultationRecord.StatusCode |
| 7. | custodian | CD | 0..1 | Organizācijas identifikators un nosaukums  Vērtība no klasifikatora OID 1.3.6.1.4.1.38760.2.23 Ārstniecības iestādes | ConsultationRecord.CustodianCode |
| 8. | subject | ED | 1..1 | Pacienta ievadītais konsultācijas temats | ConsultationRecord.Subject |
| 9. | documentId | II | 0..1 | Ārstniecības personas ievadītais dokumenta identifikators | ConsultationRecord.SetId |
| 10. | messages | PRPA\_MT000500UV01\_LV01.ConsultationMessage |  | Saraksts ar konsultācijas ziņojumiem | Skat. 6.1.21.1 |

#### ConsultationMessage (Konsultācijas ziņojums)

Identifikācija: PRPA\_MT000500UV01\_LV01.ConsultationMessage

Datu struktūra tiek izmantota:

1. Sastāvdaļa konsultācijas ierakstam



57. attēls. PRPA\_MT000500UV01\_LV01.ConsultationMessage datu struktūra

6.1‑35. tabula. PRPA\_MT000500UV01\_LV01.ConsultationMessage datu struktūra

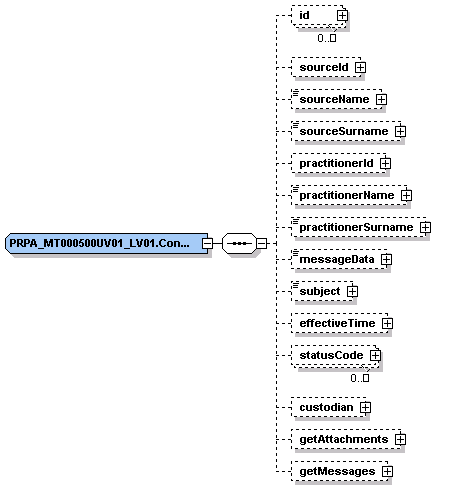
| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) |
| --- | --- | --- | --- | --- | --- |
| 1. | id | II | 0..1 | Konsultācijas ziņojuma identifikators. OID 1.3.6.1.4.1.38760.3.4.5.44 | ConsultationMessage.ConsultationMessageId |
| 2. | sequence | INT | 0..1 | Ziņojumu secība konsultācijas ierakstā | ConsultationMessage.Sequence |
| 3. | direction | CS | 0..1 | Ziņojuma virziens  IN – no iniciatora ārstam  OUT- no ārsta iniciatoram | ConsultationMessage.Direction |
| 4. | effectiveTime | TS | 0..1 | Ziņojuma veidošanas datums/laiks. Saglabājot ierakstu, tiek ierakstīts sistēmas datums un datu struktūrā ierakstītā vērtība tiek ignorēta | ConsultationMessage.EffectiveTime |
| 5. | messageBody | ED | 0..1 | Ziņojuma teksts. Sistēma nepārbauda tekstu – tas tiek saglabāts tādā veidā, kā saņemts | ConsultationMessage.MessageBody |
| 6. | statusCode | CS | 1..1 | Ziņojuma statusa kods. Iespējamās vērtības:  Active – ziņojums ir pieejams  Cancelled – ziņojums ir dzēsts | ConsultationMessage.StatusCode |
| 7. | attachments | PRPA\_MT000500UV01\_LV01.ConsultationMessageAttachment | 0..\* |  |  |
|  | id | II | 0..1 | Ziņojuma pielikuma identifikators. OID 1.3.6.1.4.1.38760.3.4.5.45 | ConsultationMessageAttachment .ConsultationMessageAttachmentId |
|  | fileName | ED | 1..1 | Ziņojuma pielikuma faila nosaukums. | ConsultationMessageAttachment .FileName |
|  | fileBody | ED | 1..1 | Ziņojuma pielikuma faila saturs. Vienmēr tiek atgriezts no sistēmas arhivētā veidā (izmantojot zip arhivēšanu). MediaType tiek uzlikts application/zip. Ja saņemts no ārējas sistēmas ar šādu MediaType, vērtība tiek saglabāta datu bāzē bez arhivēšanas, visos pārējos gadījumos vērtība tiek saglabāta EVK to arhivējot. | ConsultationMessageAttachment .FileBody |
|  | statusCode | CS | 1..1 | Ziņojuma pielikuma statusa kods. Iespējamās vērtības:  Active – pielikums ir pieejams  Cancelled – pielikums ir dzēsts | ConsultationMessageAttachment .StatusCode |

### ConsultationsListQuery (Konsultāciju saraksta pieprasījums)

Identifikācija: PRPA\_MT000500UV01\_LV01.ConsultationsListQuery

Datu struktūra tiek izmantota:

1. GetConsultationsList.



58. attēls. PRPA\_MT000500UV01\_LV01.ConsultationsListQuery datu struktūra

6.1‑36. tabula. PRPA\_MT000500UV01\_LV01.ConsultationsListQuery datu struktūra

| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) |
| --- | --- | --- | --- | --- | --- |
| 1. | id | II | 0..\* | Konsultācijas ieraksta identifikators OID 1.3.6.1.4.1.38760.3.4.5.43 | ConsultationRecord.ConsultationRecordId |
| 2. | sourceId | II | 0..1 | Pacienta OID identifikators, atbilstoši:  Personas kods: 1.3.6.1.4.1.38760.3.1.1  Ārstniecības personas id, atbilstoši klasifikatoram OID 1.3.6.1.4.1.38760.2.1 | ConsultationRecord.SourceType (root), ConsultationRecord.SourceCode (extension) |
| 3. | sourceName | ED | 0..1 | Pacienta vai ārstniecības personas, kura uzsākusi konsultāciju, vārds. | Patients. GivenName / MedPersons. DisplayName |
| 4. | sourceSurname | ED | 0..1 | Pacienta vai ārstniecības personas, kura uzsākusi konsultāciju, uzvārds. | Patients. FamilyName / MedPersons. DisplayName |
| 5. | practitionerId | II | 0..1 | Ārstniecības personas id, atbilstoši klasifikatoram OID 1.3.6.1.4.1.38760.2.1 Ārstniecības personu un ārstniecības atbalsta personu reģistrs | ConsultationRecord.PractitionerCode |
| 6. | practitionerName | ED | 0..1 | Konsultācijas adresāta vārds. | MedPersons. DisplayName |
| 7. | practitionerSurname | ED | 0..1 | Konsultācijas adresāta uzvārds | MedPersons. DisplayName |
| 8. | messageData | ED | 0..1 | Brīvs ziņojuma teksts | ConsultationMessage.MessageBody |
| 9. | subject | ED | 0..1 | Brīvs konsultācijas virsraksta teksts | ConsultationRecord.Subject |
| 10. | effectiveTime | IVL\_TS | 0..1 | Konsultācijas termiņš (no sākuma līdz slēgšanai vai pabeigšanai). Lauka vērtība tiek uzstādīta automātiski un vērtības, kas saņemtas kā ieejas parametrs, netiek apstrādātas | ConsultationRecord.EffectiveTimeFrom ConsultationRecord.EffectiveTimeTo |
| 11. | statusCode | CS | 0..\* | Konsultācijas statuss var būt kombinācija no:  Active  Read  Answered  Complete  Ja parametrs nav norādīts, tiek atgrieztas visas konsultācijas | ConsultationRecord.StatusCode |
| 12. | custodian | II | 0..1 | Organizācijas identifikators  Vērtība no klasifikatora OID 1.3.6.1.4.1.38760.2.23 Ārstniecības iestādes | ConsultationRecord.CustodianCode |
| 13. | getAttachments | BL | 0..1 | Vai atgriezt pielikumus kopā ar rezultātiem. | - |
| 14. | getMessages | BL | 0..1 | Vai atgriezt ziņojumu tekstu kopā ar rezultātiem. | - |

### ConsultationsSuggestionsQuery (Konsultācijas meklēšanas pieprasījuma variantu parametri)

Identifikācija: PRPA\_MT000500UV01\_LV01.ConsultationsSuggestionsQuery

Datu struktūra tiek izmantota:

1. GetConsultationSearchSuggestions.



59. attēls. PRPA\_MT000500UV01\_LV01.ConsultationsSuggestionsQuery datu struktūra

6.1‑37. tabula. PRPA\_MT000500UV01\_LV01.ConsultationsSuggestionsQuery datu struktūra

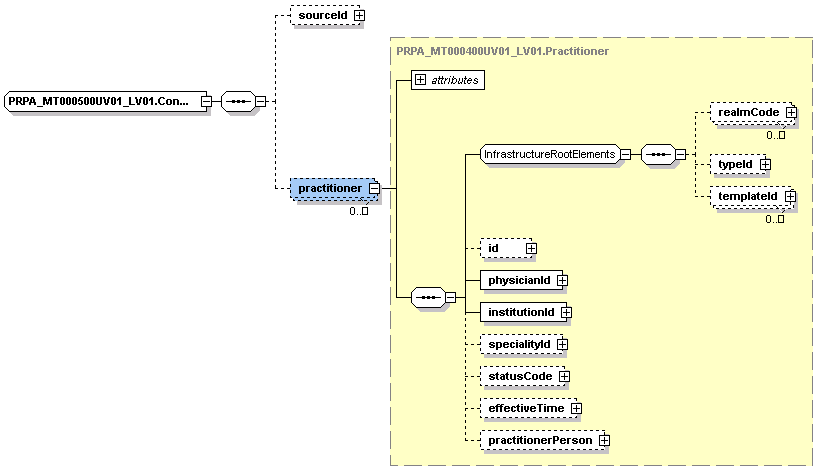
| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) |
| --- | --- | --- | --- | --- | --- |
| 1. | sourceId | II | 0..1 | Pacienta OID identifikators, atbilstoši:  Personas kods: 1.3.6.1.4.1.38760.3.1.1  Ārstniecības personas id, atbilstoši klasifikatoram OID 1.3.6.1.4.1.38760.2.1 | ConsultationRecord.SourceType (root), ConsultationRecord.SourceCode (extension) |

### ConsultationsSuggestionsResponse (Konsultācijas meklēšanas pieprasījuma variantu rezultāts)

Identifikācija: PRPA\_MT000500UV01\_LV01.ConsultationsSuggestionsResponse

Datu struktūra tiek izmantota:

1. GetConsultationSearchSuggestions.



60. attēls. PRPA\_MT000500UV01\_LV01.ConsultationsSuggestionsResponse datu struktūra

6.1‑38. tabula. PRPA\_MT000500UV01\_LV01.ConsultationsSuggestionsResponse datu struktūra

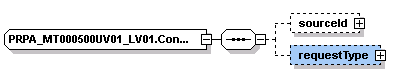
| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) |
| --- | --- | --- | --- | --- | --- |
| 1. | sourceId | II | 0..1 | Pacienta OID identifikators, atbilstoši:  Personas kods: 1.3.6.1.4.1.38760.3.1.1  Ārstniecības personas id, atbilstoši klasifikatoram OID 1.3.6.1.4.1.38760.2.1 | ConsultationRecord.SourceType (root), ConsultationRecord.SourceCode (extension) |
| 2. | practitioner | PRPA\_MT000400UV01\_LV01.Practitioner | 0..\* | Ārstniecības personas ierakstu saraksts | Skat. 6.1.18 |

### ConsultationsPractitionersQuery (Konsultācijas ārstu kontaktsaraksta pieprasījums)

Identifikācija: PRPA\_MT000500UV01\_LV01. ConsultationsPractitionersQuery

Datu struktūra tiek izmantota:

1. GetConsultationPractitioners.



61. attēls. PRPA\_MT000500UV01\_LV01.ConsultationsPractitionersQuery datu struktūra

6.1‑39. tabula. PRPA\_MT000500UV01\_LV01. ConsultationsPractitionersQuery datu struktūra

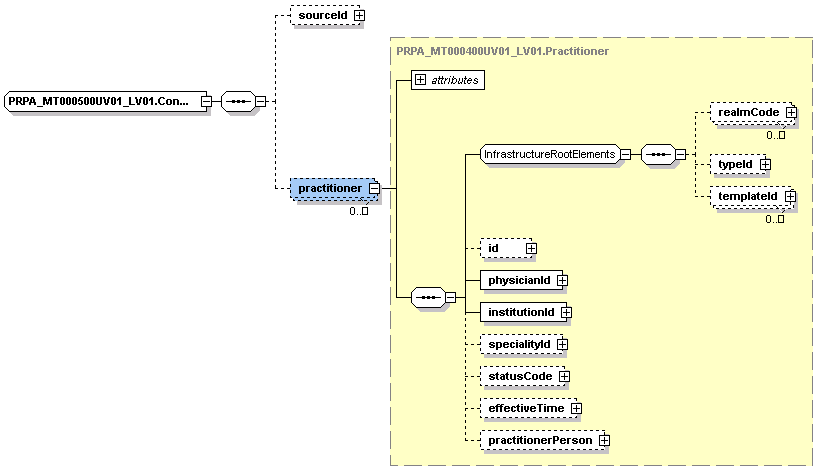
| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) |
| --- | --- | --- | --- | --- | --- |
| 1. | sourceId | II | 0..1 | Pacienta OID identifikators, atbilstoši:  Personas kods: 1.3.6.1.4.1.38760.3.1.1  Ārstniecības personas id, atbilstoši klasifikatoram OID 1.3.6.1.4.1.38760.2.1 | ConsultationPractitioners.Type, ConsultationPractitioners.Code  (Pacienta gadījumā lauks Code satur nevis personas identificējošo informāciju, bet gan NPatientID kodu) |
| 2. | requestType | CS | 0..1 | Pieprasījuma tips. Iespējamās divas vērtības – RELATED vai ALL. Ja ir RELATED, tiek atgriezti tikai ārsti, kas ir autori šī pacienta medicīnas dokumentiem. Ja ir ALL – tad visi ārsti, kas sniedz konsultācijas. |  |

### ConsultationsPractitionersResponse (Konsultācijas ārstu kontaktsaraksta pieprasījuma rezultāts)

Identifikācija: PRPA\_MT000500UV01\_LV01.ConsultationsPractitionersResponse

Datu struktūra tiek izmantota:

1. GetConsultationPractitioners.
2. SetConsultationPractitioners



62. attēls. PRPA\_MT000500UV01\_LV01.ConsultationsPractitionersResponse datu struktūra

6.1‑40. tabula. PRPA\_MT000500UV01\_LV01.ConsultationsPractitionersResponse datu struktūra

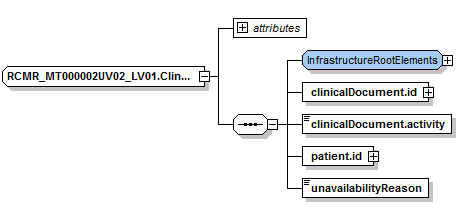
| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) |
| --- | --- | --- | --- | --- | --- |
| 1. | sourceId | II | 0..1 | Pacienta OID identifikators, atbilstoši:  Personas kods: 1.3.6.1.4.1.38760.3.1.1  Ārstniecības personas id, atbilstoši klasifikatoram OID 1.3.6.1.4.1.38760.2.1 | ConsultationPractitioners.Type, ConsultationPractitioners.Code  (Pacienta gadījumā lauks Code satur nevis personas identificējošo informāciju, bet gan NPatientID kodu). |
| 2. | practitioner | PRPA\_MT000400UV01\_LV01.Practitioner | 0..\* | Ārstniecības personas ierakstu saraksts. | Saraksts ar ārstniecības personām no lauka ConsultationPractitioners.Value  Skat. 6.1.18 |

### ClinicalDocumentUnavailableRequest (Dokumenta nepieejamības uzstādīšanas pieprasījums)

Identifikācija: RCMR\_MT000002UV02\_LV01.ClinicalDocumentUnavailableRequest

Datu struktūra tiek izmantota:

1. SetDocumentUnavailable.



63. attēls. RCMR\_MT000002UV02\_LV01.ClinicalDocumentUnavailableRequest datu struktūra

6.1‑41. tabula. RCMR\_MT000002UV02\_LV01.ClinicalDocumentUnavailableRequest datu struktūra

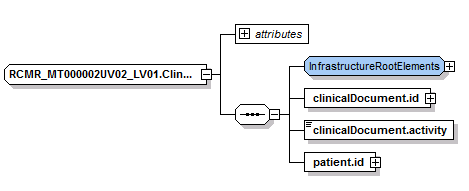
| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) |
| --- | --- | --- | --- | --- | --- |
| 1. | clinicalDocument.id | II | 1..1 | Dokumenta identifikators | Documents.DocumentID |
| 2. | clinicalDocument.activity | Uid | 1..1 | Dokumenta veidošanas activityId. | Documents.ActivityGUID |
| 3. | patient.id | II | 1..1 | Pacienta identifikators | PatientIdentifications.IdentificationType un IdentificationCode |
| 4. | unavailabilityReason | xs:string | 1..1 | Dokumenta slēpšanas iemesls | Documents.confidentialityReason |

### ClinicalDocumentStatusRequest (Dokumenta statusa pieprasījums)

Identifikācija: RCMR\_MT000002UV02\_LV01.ClinicalDocumentStatusRequest

Datu struktūra tiek izmantota:

1. GetDocumentStatus.



64. attēls. RCMR\_MT000002UV02\_LV01.ClinicalDocumentStatusRequest datu struktūra

6.1‑42. tabula. RCMR\_MT000002UV02\_LV01.ClinicalDocumentStatusRequest datu struktūra

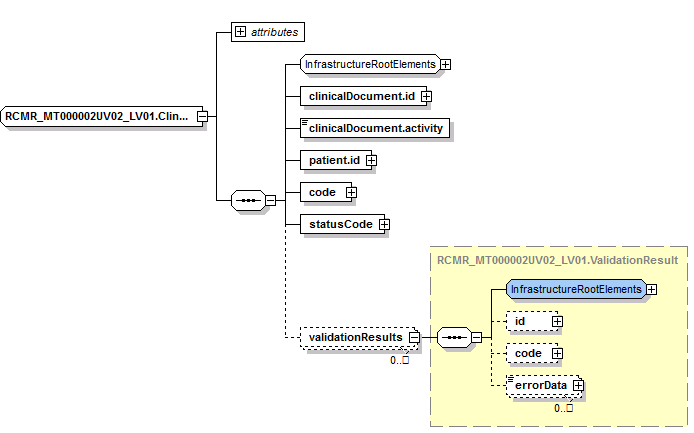
| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) |
| --- | --- | --- | --- | --- | --- |
| 1. | clinicalDocument.id | II | 1..1 | Dokumenta identifikators | Documents.DocumentID |
| 2. | clinicalDocument.activity | string | 1..1 | Dokumenta veidošanas activityId. | Documents.ActivityGUID |
| 3. | patient.id | II | 1..1 | Pacienta identifikators | PatientIdentifications.IdentificationType un IdentificationCode |

### ClinicalDocumentStatusResponse (Dokumenta statusa pieprasījuma atbilde)

Identifikācija: RCMR\_MT000002UV02\_LV01. ClinicalDocumentStatusResponse

Datu struktūra tiek izmantota:

1. GetDocumentStatus.



65. attēls. RCMR\_MT000002UV02\_LV01. ClinicalDocumentStatusResponse datu struktūra

6.1‑43. tabula. RCMR\_MT000002UV02\_LV01. ClinicalDocumentStatusResponse datu struktūra

| Nr. | Parametrs | Tips | Skaits | Apraksts | Izgūt no DB (get) |
| --- | --- | --- | --- | --- | --- |
| 1. | clinicalDocument.id | II | 1..1 | Dokumenta identifikators | Documents.DocumentID |
| 2. | clinicalDocument.activity | string | 1..1 | Dokumenta veidošanas activityId. | Documents.ActivityGUID |
| 3. | patient.id | II | 1..1 | Pacienta identifikators | PatientIdentifications.IdentificationType un IdentificationCode |
| 4. | code | CE | 1..1 | Dokumenta šablona OID | Document.DocumentTemplateOID |
| 5. | statusCode | CS | 1..1 | Dokumenta statusa lauks | Document.Status |
| 6. | validationResults | RCMR\_MT000002UV02\_LV01.ValidationResult | 0..\* | Dokumenta validācijas kļūdu saraksts  Tiek aizpildīts no tabulas DocumentErrors, kur ieraksti atbilst dokumenta identifikatoram.  Id – root = 1.3.6.1.4.1.38760.3.4.5.2  Extension = DocumentErrors.ErrorID  Code – kļūdas kods DocumentErrors.ErrorCode  errorData – kļūdas informācija DocumentErrors.ErrorData |  |

## Pamatdatu un pierakstu pārvaldības moduļa projektējums

### addDocument (Pievienot dokumentu – sinhronā daļa)

Identifikācija

Funkcija: addDocument

Lietotāju grupa : Ārstniecības personāls.

Tiesības:

EvkRghtDocumentAdd - primāra funkcijas tiesība

EvkRghtDocumentCdaAdd – tiesības CDA dokumenta pievienošanai

EvkRghtDocumentCcdAdd – tiesības CCD jeb veselības pamatdatu pievienošanai

Apraksts

Funkcija pacientu kartēm pievieno vienu vai vairākus dokumentus pēc pieprasījuma. Dokumentam jābūt CDA dokumentam (tai skaitā CCD).

Izmantojot šo funkciju, pievieno arī veselības pamatdatu ierakstus (alerģiju, brīdinājumu, diagnozi, medicīnisko ierīci, medikamentu). Arī šajā gadījumā tiek veidots CDA dokuments ar veselības pamatdatu sekciju/entry.

Dokuments var būt jauna versija iepriekš pievienotam dokumentam. Dokumentu sasaisti nodrošina elements SetId, kas viena dokumenta dažādām versijām sakrīt.

Ieejas Parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- | --- |
| 1. | RCMR\_MT000002UV02\_LV01. ClinicalDocument | RCMR\_MT000002UV02\_LV01. ClinicalDocument | 1..1 | Dokumenta pievienošana |

Apstrādes algoritms

1. Pārbauda, vai RCMR\_MT000002UV02\_LV01. ClinicalDocument sarakstā ir viens dokuments. Ja ir vairāki, atgriež kļūdu EVK\_0049 Parametri norādīti nekorekti
2. Pārbauda, vai lietotājam ir tiesības izsaukt šo funkciju (skat.6.5.5 checkUserRight (Pārbaudīt personas tiesību)):

* Ja dokumenta aploksnes ClinicalDocument.code = 1.3.6.1.4.1.38760.1.2.1.1,

tad pārbauda, vai lietotājam ir tiesības EvkRghtDocumentCcdAdd;

* Pretējā gadījumā pārbauda tiesības EvkRghtDocumentCdaAdd.

1. Deserializē CDA XML:

* Dekodē ClinicalDocument.text elementa saturu no B64 uz tekstu;
* Ja dokuments ir eDoc, tad pārbauda parakstu un izgūst CDA XML(6.11.1 CDA dokumenta izgūšana no eDoc faila). Kļūdaina vai nekorekta e-paraksta gadījumā tiek atriezts kļūdas paziņojums un process tiek pārtraukts;
* Dokumenta saturu deserializē uz CDA XML.

1. Pārbauda dokumenta digitālo parakstu:

* Ja dokumenta struktūrā ir aizpildīts lauks Signature, tā tiek pārbaudīta uz pareizību un tās sertifikāts uz to, ka sertifikācijas ķēde ir no uzticamām sertifikācijas organizacijām (piem. LVRTC);
* Ja iepriekšējā posma pārbaudes atgrieza kļūdu, dokuments netiek pieņemts EVK un tiek atgriezts kļūdas ziņojums (EVK\_0067 Dokumenta paraksts ir kļūdains).

1. Pārbauda dokumenta identifikāciju (skat. 6.1.5 CDA dokumenta identifikācija):

* Id jāsastāv no SetId un VersionNumber vērtībām.

Piemērs:

Ja SetId ir 25489.653 un VersionNumber = 1 tad Id jābūt 25489.653.1.

* ! Ārstniecības iestādes koda atbilstību NVD VIS nepārbauda.
* Ja identifikācija nav korekta, tiek atgriezts kļūdas paziņojums (EVK\_0053 Nekorekta dokumenta identifikācija), un process tiek pārtraukts.

1. Pārbauda obligātos laukus.

* Tiek pārbaudīti CDA XML struktūras obligātie elementi:
* Id – dokumenta identifikators,
* effectiveTime – dokumenta datums,
* templateId – veidnes versija,
* code – dokumenta tips,
* confidentialityCode,
* versionNumber – dokumenta versijas kārtas numurs,
* setId – dokumenta identifikators,
* author – dokumenta autors,
* custodian – dokumenta atbildīgais,
* recordTarget – pacienta identifikācija
  + - root – identifikācijas tips
    - extension – identifikācijas vērtība,
* title – dokumenta nosaukums
* Ja kāds no obligātajiem elementiem nav aizpildīts, tiek atgriezts kļūdas paziņojums (EVK\_0044 Nav aizpildīts obligāts CDA dokumenta elements), un process tiek pārtraukts.
* Tiek salīdzinātas lauku vērtības CDA XML un ClinicalDocument aploksnē:
* Id – dokumenta identifikators,
* effectiveTime – dokumenta datums,
* code – dokumenta tips,
* author – dokumenta autors,
* custodian – dokumenta atbildīgais,
* recordTarget – pacienta identifikācija:
  + - root – identifikācijas tips,
    - extension – identifikācijas vērtība.

Ja šo lauku vērtības atšķiras, tiek atgriezts kļūdas paziņojums (EVK\_0034 Lauku vērtības CDA XML un ClinicalDocument aploksnē nav vienādas), un process tiek pārtraukts.

! Visas pārbaudes un datu saglabāšana tiek veikta ar datiem no CDA XML, kas iegūts no ClinicalDocument.text .

Pārbauda dokumenta tipu un veidnes versiju.

* No elementa CDA.effectiveTime [value] tiek iegūts datums, pēc kura pārbauda veidnes versijas (templateId) spēkā esamību:

DocumentTemplates.validfrom <= CDA.effectiveTime [value] <= DocumentTemplates.validTill

* Ja veidne nav spēkā, tiek atgriezts kļūdas paziņojums (EVK\_0035 Nav atrasts aktuāls dokumenta šablons atbilstoši iesniedzamā dokumenta elementam templateid), un process tiek pārtraukts.
* Pārbauda, vai veidnes versija (templateId) atbilst norādītajam dokumenta tipam: CDA.code [code] = DocumentTemplates.DocumentTypeOID. Ja neatbilst, tiek atgriezts kļūdas paziņojums (EVK\_0037 Veidnes versija (templateId) neatbilst dokumenta tipam), un process tiek pārtraukts.

Pārbauda dokumenta versijas.

* Elementa CDA.setId [extension] vērtību meklē Documents tabulā laukā SetID.
* Ja ierakstu atrod, tas nozīmē, ka saņemtais dokuments ir eksistējoša dokumenta jauna versija. CDA.versionNumber vērtībai jābūt lielākai par maksimālo datu bāzē esošo VersionNumber vērtību. Pretējā gadījumā tiek atgriezts kļūdas paziņojums (EVK\_0038 Dokumenta versija nav norādīta korekti), un process tiek pārtraukts.

Pārbauda lietotāja tiesības uzstādīt norādīto aizliegumu RCMR\_MT000002UV02\_LV01.ClinicalDocument.confidentialityRecord. Tiek izsaukta funkcija 6.5.1 checkRightsSetObjectAccessRights (Pārbauda tiesības uzstādīt aizliegumu), kurai tiek nodoti parametri:

* RightDescriptor – konstanta vērtība „111”;
* NewRightDescriptor – elementā RCMR\_MT000002UV02\_LV01.ClinicalDocument.confidentialityRecord tiek atrasts ieraksts, kura confidentialityType vērtība ir „DOC” un funkcijai tiek padota atbilstoša ieraksta confidentialityCode [code] vērtība šī vērtība tiek piefiksēta „ConfidentialityCode”;
* UserRightDescriptor – lietotāja lomai atbilstošais tiesību deskriptors.

Ja pārbaudes rezultāts neveiksmīgs, tad tiek atgriezts kļūdas paziņojums (EVK\_0018 Nav tiesības uzstādīt aizliegumu (pacienta karte, dokuments, veselības pamatdati)), un process tiek pārtraukts.

Pacienta identifikācija.

* No elementa CDA.recordTarget tiek iegūti pacienta identifikācijas dati:
* IdentificationType = CDA.recordTarget.patientRole.id [root]
* IdentificationCode = CDA.recordTarget.patientRole.id [extension]
* Tiek pārbaudīts vai norādītais personas identifikators atbilst formātam:
* Ja identifikācijas tips ir LV personas kods (root vērtība ir 1.3.6.1.4.1.38760.3.1.1 OID shēma ), tad elementa extension vērtība ir personas kods. Ja formāts neatbilst, tad process tiek pārtraukts atgriežot kļūdas paziņojumu EVK\_0047 (Pacienta identifikatora formāts neatbilst identifikācijas tipam) un izveidots audita ieraksts.
* Ja identifikācijas tips ir jaundzimušais (root vērtība ir 1.3.6.1.4.1.38760.3.1.3 OID shēma), tad elementa extension vērtība ir mātes personas kods dzimšanas datums, laiks ar precizitāti līdz minūtei (150385-11229/18.10.2011 10:03). Ja formāts neatbilst, tad process tiek pārtraukts, atgriežot kļūdas paziņojumu EVK\_0047 (Pacienta identifikatora formāts neatbilst identifikācijas tipam) un izveidots audita ieraksts.
* Citiem identifikācijas tipiem formātus nepārbauda.
* Pārbauda, vai EVK DB eksistē pacienta karte pēc EVKPers..PatientIdentifications.IdentificationCode un EVKPers..PatientIdentifications.IdentificationType lauka datiem:
* Ja neeksistē, tad izsauc funkciju createPatientCard (6.2.5 createPatientCard (Izveidot pacienta kartes ierakstu). Ja pacienta karti neizdevās izveidot, tad tiek atgriezts kļūdas paziņojums (EVK\_0039 Personas kods nav korekts), un process tiek pārtraukts;
* Ja eksistē, tad tiek piefiksēts pacienta kartes lauks „NPatients.NPatientID”.

Dokumenta saglabāšana pacienta kartē.

* Izveido ierakstu Documents tabulā.

| Nr. | Datu lauks | Tips | Aizpilda | Piezīmes |
| --- | --- | --- | --- | --- |
| 1 | DocumentID | GUID | CDA.id [Extension] | Dokumenta instances identifikators, ko piešķir autors vai autora sistēma |
| 2 | SetID | GUID | CDA.setID [Extension] | Dokumenta GUID identifikators, kurš ir vienāds visiem vienas versijas dokumentiem |
| 3 | CDA | varbinary  (max) | Šifrēts CDA XML  Šifrēšanai piemēro AES256 algoritmu | Šifrēts CDA XML formāta dokuments |
| 4 | Signature | varbinary  (max) | Šifrēts eDoc.  Šifrēšanai piemēro AES256 algoritmu | Elektroniskais paraksts.  Aizpilda tikai eDoc gadījumā.  Citi elektroniskā paraksta gadījumi izstrādes pirmajā posmā nav identificēti, bet var tikt identificēti nākotnē |
| 5 | NPatientID | int | NPatients.NPatientID | Atsauce uz pacientu, kura kartē dokuments tiek saglabāts |
| 6 | EffectiveTime | datetime | CDA.effectiveTime [value] | Dokumenta datums un laiks |
| 7 | Status | varchar | “Processing” | Dokumenta statuss “Apstrādē” |
| 8 | VersionNumber | int | CDA.versionNumber [value] | Dokumenta versijas numurs |
| 9 | Title | nvarchar | CDA.title | Dokumenta nosaukums |
| 10 | AccessRights | tinyint | ClinicalDocument.confidentialityRecord.confidentialityCode [code] | Piekļuves tiesības pacientam pašam/delegātiem/ārstiem |
| 11 | DocumentTemplateOID |  | CDA.templateId [root] | Dokumenta veidne |

Saistītie dokumenti

* Pārbauda, vai dokumentā ir aizpildīta saistīto dokumentu sadaļa (parentDocument lauks xml).
* Gadījumā, ja datu struktūra ir aizpildīta ar datiem, tiek veidots ieraksts tabulā DocumentLinkToDocuments, aizpildot datus atbilstoši tabulai:

| Nr. | Datu lauks | Tips | Aizpilda | Piezīmes |
| --- | --- | --- | --- | --- |
| 1 | DocumentID | string | CDA.id [Extension] | Dokumenta instances identifikators, ko piešķir autors vai autora sistēma |
| 2 | SetID | String | CDA.setID [Extension] | Dokumenta identifikators, kurš ir vienāds visiem vienas versijas dokumentiem |
| 3 | EffectiveTime | Datetime | CDA.effectiveTime [value] | Dokumenta datums un laiks |
| 4 | ParentDocumentId | string | CDA.parentDocument.id [Extension] | Saistīta dokumenta instances identifikators, ko piešķir autors vai autora sistēma. Dokumenta eksistence netiek pārbaudīta, saglabājot datus |
| 5 | ParentDocumentSetId | string | CDA.parentDocument.setId [Extension] | Saistīta dokumenta identifikators, kas vienāds visām saistīta dokumenta versijām. Dokumenta eksistence netiek pārbaudīta saglabājot datus |
| 6. | ParentDocumentTemplateId | string | CD.parentDocument.templateId | Saistīta dokumenta tipa kods. Atbilst dokumenta šablona OID |

Dokumentu pievieno asinhronajai apstrādei:

* Asinhronā apstrāde tiek realizēta, izmantojot fona uzdevumu (skatīt 6.13.1. Dokumenta apstrādes process), kas izpilda dokumenta validāciju pret modeli, klasifikatoriem; HTML ģenerēšanu, indeksēto datu izgūšanu, veselības pamatdatu izgūšanu un vakcinācijas datu izgūšanu.
* Sūta ziņojumu [EVK/CDA/Document] SBS CDADocumentService.
* Ziņojuma saturā ir CDA XML

Alternatīvie scenāriji

Nav tiesības pievienot dokumentu. Tiek atgriezts kļūdas paziņojums (EVK\_0029 Nav tiesības veikt darbību) un izveidots audita ieraksts.

### addComments (Pievienot piezīmes)

Identifikācija

Funkcija: addComments

Tiesības: EvkRghtDocumentCommentAdd

Apraksts

Funkcija pievieno komentāru norādītajam dokumentam (vai dokumenta elementam).

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | | RCMR\_MT000002UV02\_LV01. Comment | RCMR\_MT000002UV02\_LV01. Comment | 1..1 | Dokumenta/pamatdatu komentāra struktūra  (6.1.6.4 Comment (Komentāra struktūra) ) |

Apstrādes algoritms

1. Pārbauda, vai lietotājam ir tiesības izsaukt šo funkciju (skat.6.5.5 checkUserRight (Pārbaudīt personas tiesību) - EvkRghtDocumentCommentAdd).
2. Tiek pārbaudīts, vai pieprasīto dokumentu pacientu kartei atļauts piekļūt, izsaucot funkciju 6.5.4 checkUserRightsAccessObject (Pārbaudīt personas tiesību līmeni uz objektu).

Tiek pārbaudīts, vai dokuments vai pamatdatu ieraksts (atkarībā no komentāru struktūras lauka targetId), kuram tiek pievienoti komentāri, eksistē. Ja dokuments neeksistē vai pamatdatu ieraksts neeksistē, tad tiek atgriezta kļūda un izveidots audita ieraksts.

Saņemtie komentāri (no struktūras userComments), tiek ierakstīti tabulā DocumentComments (atbilstoši atribūta „updateMode” un aizpildīšanas principiem, kas aprakstīti 6.1.6 DocumentStructure (CDA dokumenta aploksnes struktūra)).

Tiek izveidota izejas datu struktūra, kas apraksta komentāru pievienošanas rezultātu (skat. 6.1.10.1 ack (Tehniskā atbildes datu struktūra))

Alternatīvie scenāriji

Nav tiesības piekļūt pacienta kartei. Tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte) un izveidots audita ieraksts

Pacienta karte nav atrasta. Tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte) un izveidots audita ieraksts.

Nav tiesības komentēt dokumentu. Tiek atgriezts kļūdas paziņojums (EVK\_0029 Nav tiesības veikt darbību) un izveidots audita ieraksts.

### setDocumentStatus (Mainīt dokumenta statusu)

Identifikācija

Funkcija: setDocumentStatus

Tiesības: EvkRghtDocumentStatusSet

Apraksts

Funkcija anulē norādīto dokumentu.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | | RCMR\_MT000002UV02\_LV01. ClinicalDocument | RCMR\_MT000002UV02\_LV01. ClinicalDocument | 1..1 | Dokumenta struktūra  (6.1.6 DocumentStructure (CDA dokumenta aploksnes struktūra)). |

Apstrādes algoritms

1. Pārbauda, vai lietotājam ir tiesības izsaukt šo funkciju (skat.6.5.5 checkUserRight (Pārbaudīt personas tiesību)). Ja tiesību nav, funkcija tiek pārtraukta, tiek atgriezts kļūdas paziņojums (EVK\_0008 Dokuments nav atrasts vai nav pieejams), un izveidots audita ieraksts.
2. Tiek pārbaudīts, vai pacienta kartei nav uzlikts pilnais aizliegums, un vai dokumentam nav uzstādīts aizliegums pret lietotāju, izsaucot funkciju 6.5.4 checkUserRightsAccessObject (Pārbaudīt personas tiesību līmeni uz objektu).

Tiek pārbaudīta ieejas datu struktūra un tiek norādīts pacienta identifikators. Ieejas datu struktūrā jābūt aizpildītiem laukiem:

Id, code, statusCode, effectiveTime, recordTarget, author, custodian.

Lauku author, custodian, un effectiveTime vērtības netiek izmantotas funkcijas darbā.

Ja struktūra ir nekorekta, tiek atgriezts kļūdas paziņojums (EVK\_0049 Parametri norādīti nekorekti), tiek izveidots audita ieraksts un process tiek pārtraukts.

Tiek pārbaudīts, vai dokuments, kuram tiek uzstādīts aizliegums eksistē. Ja dokuments neeksistē, tad tiek atgriezts kļūdas paziņojums (EVK\_0008 Dokuments nav atrasts vai nav pieejams) un izveidots audita ieraksts.

Tiek pārbaudīts, vai dokumenta statuss ir ‘Aktuāls’. Citādi tiek atgriezta kļūda un izveidots audita ieraksts.

Tiek izsaukta funkcija 6.2.12.4 setDocumentStatusToCancelled (dokumenta statusa maiņa uz „Anulēts”), padodot funkcijai dokumenta Id parametru, un SAML token norādītā lietotāja informācija (personas kods un vārds).

Tiek izveidota izejas datu struktūra, kas apraksta dokumenta anulēšanas rezultātu (skat. 6.1.10.1 ack (Tehniskā atbildes datu struktūra))

Alternatīvie scenāriji

Nav tiesības anulēt dokumentu. Tiek atgriezts kļūdas paziņojums (EVK\_0008 Dokuments nav atrasts vai nav pieejams) un izveidots audita ieraksts.

Nav tiesības piekļūt pacienta kartei. Tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte) un izveidots audita ieraksts.

Dokuments nav atrasts. Tiek atgriezts kļūdas paziņojums (EVK\_0008 Dokuments nav atrasts vai nav pieejams). un izveidots audita ieraksts.

Dokumenta versiju nav iespējams anulēt, jo tā jau ir anulēta. Tiek atgriezts kļūdas paziņojums (EVK\_0042 Dokumenta versiju nav iespējams anulēt, jo tā jau ir anulēta) un izveidots audita ieraksts.

### setAccessRights (Mainīt aizliegumus)

Identifikācija

Funkcija: setAccessRights

Tiesības:

EvkRghtAccessRightsSet

EvkRghtDocumentAccessRightsSet

EvkRghtPatientCardAccessRightsSet

EvkRghtHealthBasicDataAccessRightsSet

Apraksts

Funkcija modificē dokumenta uzstādīto aizliegumu komplektu.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | | RCMR\_MT000002UV02\_LV01. ConfidentialityRecord | RCMR\_MT000002UV02\_LV01. ConfidentialityRecord | 1..1 | Aizliegumu struktūra  6.1.6.2 ConfidentialityRecord (Aizliegumu struktūra) |

Apstrādes algoritms

1. Pārbauda, vai lietotājam ir tiesības izsaukt šo funkciju (skat.6.5.5 checkUserRight (Pārbaudīt personas tiesību)- EvkRghtAccessRightsSet).
2. Tiek pārbaudīts, vai pieprasīto dokumentu pacientu kartei atļauts piekļūt, izsaucot funkciju 6.5.4 checkUserRightsAccessObject (Pārbaudīt personas tiesību līmeni uz objektu).
3. Katrai confidentialityRecord rindai tiek:

* Pārbaudīts, kāds aizliegums tiek uzstādīts konkrētā rindā, analizējot vērtību laukā confidentialityType: (var tikt uzstādīts aizliegums uz dokumentu (DOC) , pacienta karti (CARD) vai dokumenta sadaļu (ENTRYABS)).

1. Ja tiek uzstādīts aizliegums uz dokumentu (confidentialityType=DOC):

* Pārbauda, vai lietotājam ir tiesības izsaukt šo funkciju (skat.6.5.5 checkUserRight (Pārbaudīt personas tiesību) – EvkRghtDocumentAccessRightsSet).
* Tiek pārbaudīts, vai dokuments, kuram tiek uzstādīts aizliegums, eksistē. Ja dokuments neeksistē, tad tiek atgriezts kļūdas paziņojums (EVK\_0008 Dokuments nav atrasts vai nav pieejams) un izveidots audita ieraksts.
* Tiek pārbaudīts, vai dokumentam uzstādāms aizliegums atbilst lietotāja tiesībām uzstādīt aizliegumus (skat. 6.5.1 checkRightsSetObjectAccessRights (Pārbauda tiesības uzstādīt aizliegumu)).
* Tiek pārbaudīts, vai dokumenta šablona lauks ViewLimitations satur vērtības, kas nav „ALWAYS”. Ja vērtība ir „ALWAYS”, tad ārstniecības personu tiesības nevar būt ierobežotas. Ja confidentialityCode lauks satur ierobežojumu ārstniecības personām, tad tiek atgriezts kļūdas paziņojums (EVK\_0082 Aizliegts ierobežot pieeju dokumentam) un izveidots audita ieraksts
* Gadījumā, ja lietotājam ir tiesības uzstādīt pieprasīto aizliegumu, dokumenta aizliegums tiek uzstādīts, ierakstot confidentialityCode lauka saturu tabulas Documents laukā AccessRights.

1. Ja tiek uzstādīts aizliegums uz karti (confidentialityType=CARD):

* Pārbauda, vai lietotājam ir tiesības izsaukt šo funkciju (skat.6.5.5 checkUserRight (Pārbaudīt personas tiesību)- EvkRghtPatientCardAccessRightsSet).
* Tiek pārbaudīts, vai lietotājam ir tiesības uzstādīt šādu aizliegumu (skat. 6.5.1 checkRightsSetObjectAccessRights (Pārbauda tiesības uzstādīt aizliegumu)).
* Papildus tiek pārbaudīts, vai lietotājam ir tiesība uzstādīt pacienta kartes aizliegumu (tas var būt tikai pats pacients vai viņa aizbildņi un aizgādņi). Ja lietotājam nav tiesības, procedūra tiek pārtraukta un izveidots audita ieraksts, un atgriezts kļūdas paziņojums (EVK\_0018 Nav tiesības uzstādīt aizliegumu).
* Gadījumā, ja lietotājam ir tiesības uzstādīt pieprasīto aizliegumu, kartes aizliegums tiek uzstādīts, ierakstot confidentialityRecord.confidentialityCode lauka saturu tabulas NPatients laukā AccessRights.

1. Ja tiek uzstādīts aizliegums uz veselības pamatdatiem (confidentialityType=ENTRYABS):

* Pārbauda, vai lietotājam ir tiesības izsaukt šo funkciju (skat.6.5.5 checkUserRight (Pārbaudīt personas tiesību)- EvkRghtHealthBasicDataAccessRightsSet).
* Tiek identificēts, kādam pamatdatu veidam ir uzstādīts aizliegums (pēc lauka entryType, kuram jābūt obligāti aizpildītam, ja tiek mainīti aizliegumi uz dokumenta sadaļām).
* Tiek atlasīts veselības pamatdatu elements, kam atbilst saņemtais aizliegums (pēc lauka Id), no atbilstošas tabulas (NPatientAllergies, NPatientDiagnosis, NPatientMedicalDevices, NPatientMedications).
* Tiek pārbaudīts, vai lietotājam ir tiesības uzstādīt šim medicīnas pamatdatu elementam aizliegumus (skat. 6.5.4 checkUserRightsAccessObject (Pārbaudīt personas tiesību līmeni uz objektu). Tiek atgriezts kļūdas paziņojums (EVK\_0008 Dokuments nav atrasts vai nav pieejams) un izveidots audita ieraksts.
* Ja pacientam ir tiesības uzstādīt aizliegumus medicīnas pamatdatiem, tad pēc pārbaudes (skat 6.5.1 checkRightsSetObjectAccessRights (Pārbauda tiesības uzstādīt aizliegumu)) aizliegums (confidentialityRecord.confidentialityCode) tiek ierakstīts atbilstošas tabulas laukā AccessRights. Tiek atgriezts kļūdas paziņojums (EVK\_0018 Nav tiesības uzstādīt aizliegumu) un izveidots audita ieraksts.

1. Tiek izveidota izejas datu struktūra, kas apraksta dokumenta aizlieguma uzstādīšanas rezultātu (skat. 6.1.10.1 ack (Tehniskā atbildes datu struktūra))

Alternatīvie scenāriji

Nav tiesības piekļūt pacienta kartei. Tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte), un izveidots audita ieraksts.

Nav tiesības uzstādīt aizliegumu. Tiek atgriezts kļūdas paziņojums (EVK\_0018 Nav tiesības uzstādīt aizliegumu) un izveidots audita ieraksts.

Pacienta karte nav atrasta. Tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte) un izveidots audita ieraksts.

Nav tiesības modificēt dokumenta atļaujas. Tiek atgriezts kļūdas paziņojums (EVK\_0029 Nav tiesību veikt darbību) un izveidots audita ieraksts.

Dokumentam nedrīkst likt redzamības ierobežojumu. Tiek atgriezts kļūdas paziņojums (EVK\_0082 Aizliegts ierobežot pieeju dokumentam) un izveidots audita ieraksts.

### createPatientCard (Izveidot pacienta kartes ierakstu)

Identifikācija

Funkcija: createPatientCard

Tiesības:

EvkRghtPatientCardCreate

EvkRghtPatientCardCreateUnconfirmed

Apraksts

Funkcija izveido jaunu pacienta kartes ierakstu un atgriež tehnisko ziņojumapmaiņas aploksnes datu struktūru (skat. 6.1.10.1 ack (Tehniskā atbildes datu struktūra)).

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | | PRPA\_MT201390UV02\_LV01.PersonCardOperations | PRPA\_MT201390UV02\_LV01.PersonCardOperations | 1..1 | Datu struktūra, kas apvieno pacienta identifikatoru un datu operāciju, kuru jāizpilda ar minēto pacientu kartēm |

Apstrādes algoritms

1. Tiek pārbaudīts, vai lietotājam ir tiesības izsaukt šo funkciju (skat. (6.5.5 checkUserRight (Pārbaudīt personas tiesību)).

Tiek pārbaudīta ieejas datu struktūra (tiek pārbaudīts, vai ir pareizi uzstādīta parametra „operation” vērtība (jābūt CREATE) un tiek norādīts pacienta identifikators.

Neatkarīgi no pacienta kartes veidošanā izmantotā identifikācijas veida tiek pārbaudīts, vai karte ar pieprasīto identifikāciju jau eksistē. Ja tāda karte jau eksistē, tiek atgriezta kļūda un izveidots audita ieraksts.

1. Ja tiek norādīts Latvijas iedzīvotāja personas koda OID (1.3.6.1.4.1.38760.3.1.1) tad tiek pieprasīta pacienta personas pārbaude no PMLP (skat. 5.2.7.2 Iegūt personas datus funkcija GetPersonName).
2. Ja pacienta ieraksts nav atrasts PMLP sistēmā, tad tiek atgriezts kļūdas ziņojums, un izveidots audita ieraksts.
3. Gadījumā, ja pacienta karte tiek veidota, nenorādot Latvijas iedzīvotāja personas koda OID, ir iespējami šādi alternatīvie scenāriji:
4. Ja tiek norādīts jaundzimušā identifikācijas OID (1.3.6.1.4.1.38760.3.1.3 ) tad karte tiek izveidota, izmantojot nosūtīto identifikācijas ierakstu.
5. Ja tiek norādīts nedrošas LV personas koda identifikācijas OID (1.3.6.1.4.1.38760.3.1.7), tad tiek pārbaudīts, vai izsaucējam ir tiesība **EvkRghtPatientCardCreateUnconfirmed**, un, ja ir, tad karte tiek izveidota, izmantojot nosūtīto identifikācijas ierakstu.
6. Ja tiek norādīts Ārzemnieka identifikācijas OID (1.3.6.1.4.1.38760.3.1.8 ), vai viens no tā identifikācijas apakškokā reģistrētiem OID (klasifikatora 1.3.6.1.4.1.38760.2.21 vērtības, kas sakās ar kodu ‘1.3.6.1.4.1.38760.3.1.8’ ), tad tiek pārbaudīts, vai izsaucējam ir tiesība **EvkRghtPatientCardCreateUnconfirmed** un, ja ir, tad karte tiek izveidota, izmantojot nosūtīto identifikācijas ierakstu.
7. Ja pacienta identifikācijas veids ir PMLP, tad, pēc pamatdatu aizpildīšanas no PMLP datiem, izveidots pacienta ieraksts tiek pievienots fona procesam (skat. 6.13.3 Pacienta kartiņas aizpilde).

Tiek izveidota izejas datu struktūra, kas apraksta pacienta kartes veidošanas rezultātu (skat. 6.1.10.1 ack (Tehniskā atbildes datu struktūra))

Alternatīvie scenāriji

Nav tiesības veidot pacienta kartes. Tiek atgriezts kļūdas paziņojums (EVK\_0029 Nav tiesību veikt darbību) un izveidots audita ieraksts.

Karte ar pieprasīto identifikāciju jau eksistē. Tiek atgriezts kļūdas paziņojums (EVK\_0011 Karte ar pieprasīto identifikāciju jau eksistē) un izveidots audita ieraksts.

### getDocument (Iegūt dokumentu)

Identifikācija

Funkcija: getDocument

Tiesības:

* EvkRghtDocumentsGet
* EvkRghtDocumentsActualGet
* EvkRghtDocumentsAllGet
* EvkRghtDocumentCcdGet
* EvkRghtDocumentCommentsGet

Apraksts

Funkcija atgriež vienu dokumentu pēc pieprasījuma, un atgriež dokumenta aploksnes datu struktūru RCMR\_MT000002UV02\_LV01.ClinicalDocument (6.1.6 DocumentStructure (CDA dokumenta aploksnes struktūra)).

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | | RCMR\_MT000003UV01\_LV01.QueryByParameter | RCMR\_MT000003UV01\_LV01.QueryByParameter | 1..1 | Dokumenta iegūšanas pieprasījums  (aprakstu skat. 6.1.7 DocumentQueryByParameter (Dokumentu pieprasījums)) |

Apstrādes algoritms

1. Tiek pārbaudīts, vai lietotājam ir tiesības izsaukt šo funkciju (skat. 6.5.5 checkUserRight (Pārbaudīt personas tiesību) - EvkRghtDocumentsGet). Ja tiesību nav, tiek atgriezts kļūdas paziņojums (EVK\_0029 Nav tiesību veikt darbību) un izveidots audita ieraksts.
2. Tiek izpildīts lietotāja datu pieprasījums (skat. 6.1.7 DocumentQueryByParameter (Dokumentu pieprasījums)).
3. Ja lietotāja talons ir izdots ārstniecības personai, tad tiek izpildīta šāda darbību secība:

* Ārstniecības personai tiek izveidots visu pieprasījumā iekļauto pacientu saraksts, kuriem šī persona ir ģimenes ārsts.
* No lietotāju talona un dokumentu šablonu saraksta tiek uzbūvēts lietotājam pieejamo dokumentu tipu saraksts (tas satur visus dokumentu šablonus, kuriem nav norādīti speciālas redzamības Claim – lauks ViewClaim ir tukšs vai vismaz viens no norādītajiem ViewClaim ir pieejams lietotāja talonā). Ja ārstniecības persona ir ģimenes ārsts vienam no pacientiem, kuriem ir pieprasīti dati, tad dokumentu tipu sarakstam tiek pievienoti visi dokumentu tipi, kuriem ViewClaim satur vērtību EvkRghtFamilyDoctor (neatkarīgi no ārstniecības personas talona satura). Dokumentu tipi, kas iekļauti sarakstā tikai pēc EvkRghtFamilyDoctor tiesības analīzes, ir saglabāti atsevišķā sarakstā.

1. Ja lietotājs ir privātpersona vai izmeklētājs, tam pieejamo dokumentu saraksts satur visus dokumentu tipus.
2. Ja lietotājs ir privātpersona, tad tiek pārbaudīts, vai pieprasītā dokumenta pacienta kartei atļauts piekļūt, izsaucot funkciju 6.5.3 checkRigthsAccessPatientCard (Pārbaudīt tiesības piekļūt pacienta kartei).
3. Tiek atlasīts visu pieprasīto pacientu karšu pieejas ierobežojumu saraksts (sarakstā iekļautas vērtības no tabulas NPatients.AccessRights).
4. Ja tiek pieprasīts CDA dokuments (clinicalDocument.Code = CDA)

* Ja pieprasīti visu statusu dokumenti vai pēdējās versijas (statusCode = All vai Latest), tad pārbauda tiesības (skat. 6.5.5 checkUserRight (Pārbaudīt personas tiesību) – EvkRghtDocumentsAllGet). Ja tiesību nav, tiek atgriezts kļūdas paziņojums (EVK\_0029 Nav tiesību veikt darbību) un izveidots audita ieraksts.
* Ja pieprasīti tikai aktuālie dokumenti (statusCode – Actual), tad pārbauda tiesības (skat. 6.5.5 checkUserRight (Pārbaudīt personas tiesību) – EvkRghtDocumentsActualGet). Ja tiesību nav, tiek atgriezts kļūdas paziņojums (EVK\_0029 Nav tiesību veikt darbību) un izveidots audita ieraksts.
* Visiem atrastiem rezultātiem tiek izsaukta pārbaudes funkcija (skat. 6.5.4 checkUserRightsAccessObject (Pārbaudīt personas tiesību līmeni uz objektu)). Funkcija tiek izsaukta gan pārbaudot lietotāja pieejas tiesību pašam objektam, gan atbilstošai pacienta kartei.

Dokuments tiek izņemts no atgriežamiem rezultātiem, ja formula atgriež „false”:

(VL==”ALWAYS”) OR (UR(D) AND UR(C)) kur:

VL – lauka ViewLimitation vērtība dokumenta šablonā;

UR(D) – funkcijas checkUserRightsAccessObject rezultāts, izpildot to pret dokumenta pieejas tiesību aprakstu;

UR(C) – funkcijas checkUserRightsAccessObject rezultāts, izpildot to pret pacienta kartes pieejas tiesību aprakstu.

1. No datu bāzes atgrieztiem dokumentiem tiek veikta papildu pārbaude tam, vai dokuments ir pieejams ārstniecības personai. Tiek pārbaudīti dokumenti ar tipiem, kas ir saglabāti sarakstā (skat. punktu 3). Ja ārstniecības persona nav ģimenes ārsts konkrētam pacientam, tad dokuments tiek izņemts no atgriežamo dokumentu saraksta.
2. Dokumenti, kuriem lietotājam ir pieeja, tiek noformēti atbilstoši izejas struktūrai (skat. 6.1.6 DocumentStructure (CDA dokumenta aploksnes struktūra)). Pārbauda, vai lietotājam pieejami dokumenta komentāri (skat. 6.5.5 checkUserRight (Pārbaudīt personas tiesību) - EvkRghtDocumentCommentsGet). Ja komentāri nav pieejami, tad tos neiekļauj izejas datu struktūrā.

Ja tiek pieprasīts CCD dokuments (clinicalDocument.Code = CCD):

* Pārbauda tiesības (skat. 6.5.5 checkUserRight (Pārbaudīt personas tiesību) – EvkRghtDocumentCcdGet). Ja tiesību nav, tiek atgriezts kļūdas paziņojums (EVK\_0029 Nav tiesību veikt darbību), un izveidots audita ieraksts.
* Tiek ģenerēts CCD dokuments , izsaucot funkciju 6.2.8 generateCCD (Izveidot veselības pamatdatu dokumentu).

1. Sagatavo pieprasīto dokumenta formātu (documentFormat):

* XML;
* HTML – ja dokumentam nav pieejams HTML formāts, tad veic XML transformāciju uz HTML, izmantojot funkciju 6.2.12.2 generateHTML (HTML ģenerācija no CDA dokumenta);
* PDF – veic dokumenta HTML transformāciju uz PDF, izmantojot funkciju 6.2.15 generatePDF (Ģenerēt PDF).

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | | RCMR\_MT000002UV02\_LV01.ClinicalDocument | RCMR\_MT000002UV02\_LV01.ClinicalDocument | 1..\* | Ieejas parametros norādītajam pieprasījumam atbilstošo dokumentu kopa  (aprakstu skat. 6.1.6 DocumentStructure (CDA dokumenta aploksnes struktūra)) |

Alternatīvie scenāriji

Nav tiesību apskatīt pacienta kartes. Tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte) un izveidots audita ieraksts.

Dokumentiem uzstādīti aizliegumi. Tiek atgriezta tukša atbilde (kas nesatur nevienu dokumentu).

Visām pieprasītām kartēm aizliegta pieeja. Tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte), un izveidots audita ieraksts.

### getDocumentList (Iegūt dokumentu sarakstu)

Identifikācija

Funkcija: getDocumentList

Tiesības:

* EvkRghtDocumentsGet
* EvkRghtDocumentsActualGet
* EvkRghtDocumentsAllGet
* EvkRghtDocumentCcdGet
* EvkRghtDocumentCommentsGet

Apraksts

Funkcija atgriež vienu vai vairākus dokumentu metadatus (sarakstu), izmantojot dokumenta aploksnes datu struktūru RCMR\_MT000002UV02\_LV01.ClinicalDocument (6.1.6 DocumentStructure (CDA dokumenta aploksnes struktūra)).

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | | RCMR\_MT000003UV01\_LV01.QueryByParameter | RCMR\_MT000003UV01\_LV01.QueryByParameter | 1..1 | Dokumenta iegūšanas pieprasījums  (aprakstu skat. 6.1.7 DocumentQueryByParameter (Dokumentu pieprasījums)) |

Apstrādes algoritms

1. Tiek pārbaudīts, vai lietotājam ir tiesības izsaukt šo funkciju (skat. 6.5.5 checkUserRight (Pārbaudīt personas tiesību) – EvkRghtDocumentsGet). Ja tiesību nav, tiek atgriezts kļūdas paziņojums (EVK\_0029 Dokuments nav atrasts vai nav pieejams) un izveidots audita ieraksts.

Tiek izpildīts lietotāja datu pieprasījums (skat. 6.1.7 DocumentQueryByParameter (Dokumentu pieprasījums)).

Tiek pārbaudīts, vai meklēšanas kritērijos tiek uzstādīts lauks clinicalDocument.activityId. Gadījumā, ja šīs lauks ir uzstādīts, tad nākamā pārbaude netiek veikta

1. Ja lietotāja talons ir izdots ārstniecības personai, tad tiek izpildīta šāda darbību secība:

* Ārstniecības personai tiek izveidots visu pieprasījumā iekļauto pacientu saraksts, kuriem šī persona ir ģimenes ārsts.
* No lietotāju talona un dokumentu šablonu saraksta tiek uzbūvēts lietotājam pieejamo dokumentu tipu saraksts (tas satur visus dokumentu šablonus, kuriem nav norādīti speciālas redzamības Claim – lauks ViewClaim ir tukšs vai vismaz vienos no norādītaijiem ViewClaim ir pieejams lietotāja talonā). Ja ārstniecības persona ir ģimenes ārsts vienam no pacientiem, kuriem ir pieprasīti dati, tad dokumentu tipu sarakstam tiek pievienoti visi dokumentu tipi, kuriem ViewClaim satur vērtību EvkRghtFamilyDoctor (neatkarīgi no ārstniecības personas talona satura). Dokumentu tipi, kas iekļauti sarakstā tikai pēc EvkRghtFamilyDoctor tiesības analīzes, ir saglabāti atsevišķā sarakstā.

Ja lietotājs ir privātpersona vai izmeklētājs, tam pieejamo dokumentu saraksts satur visus dokumentu tipus.

Ja lietotājs ir privātpersona, tad tiek pārbaudīts, vai pieprasītā dokumenta pacienta kartei atļauts piekļūt, izsaucot funkciju 6.5.3 checkRigthsAccessPatientCard (Pārbaudīt tiesības piekļūt pacienta kartei).

Tiek atlasīts saraksts ar visiem pieprasīto pacientu karšu pieejas ierobežojumiem (sarakstā iekļautas vērtības no tabulas NPatients.AccessRights).

Ja pieprasīti visu statusu dokumenti vai pēdējās versijas (statusCode = All vai Latest), tad pārbauda tiesības (skat. 6.5.5 checkUserRight (Pārbaudīt personas tiesību) - EvkRghtDocumentsAllGet). Ja tiesību nav, tiek atgriezts kļūdas paziņojums (EVK\_0029 Dokuments nav atrasts vai nav pieejams) un izveidots audita ieraksts.

Ja pieprasīti tikai aktuālie dokumenti (statusCode – Actual), tad pārbauda tiesības (skat. 6.5.5 checkUserRight (Pārbaudīt personas tiesību) - EvkRghtDocumentsActualGet). Ja tiesību nav, tiek atgriezts kļūdas paziņojums (EVK\_0029 Dokuments nav atrasts vai nav pieejams), un izveidots audita ieraksts.

Tiek pārbaudīts, vai meklēšanas kritērijos tiek uzstādīts lauks clinicalDocument.activityId. Gadījumā, ja šīs lauks ir uzstādīts, tad nākamā pārbaude netiek veikta.

No datu bāzes atgrieztiem dokumentiem tiek veikta papildu pārbaude tam, vai dokuments ir pieejams ārstniecības personai. Tiek pārbaudīti dokumenti ar tipiem, kas ir saglabāti sarakstā (skat. punktu 4). Ja ārstniecības persona nav ģimenes ārsts konkrētam pacientam, tad dokuments tiek izņemts no atgriežamo dokumentu saraksta.

Visiem atrastiem rezultātiem tiek izsaukta pārbaudes funkcija (skat. 6.5.4 checkUserRightsAccessObject (Pārbaudīt personas tiesību līmeni uz objektu)). Funkcija tiek izsaukta gan pārbaudot lietotāja pieejas tiesības pašam objektam, gan atbilstošai pacienta kartei.

Dokuments tiek izņemts no atgriežamiem rezultātiem, ja formula atgriež „false”:

(VL==”ALWAYS”) **OR** (UR(D) **AND** UR(C)) kur:

VL – lauka ViewLimitation vērtība dokumenta šablonā;

UR(D) – funkcijas checkUserRightsAccessObject rezultāts, izpildot to pret dokumenta pieejas tiesību aprakstu;

UR(C) – funkcijas checkUserRightsAccessObject rezultāts, izpildot to pret pacienta kartes pieejas tiesību aprakstu.

Dokumenti, kuriem lietotājam ir pieeja, tiek noformēti atbilstoši izejas struktūrai (skat. 6.1.6 DocumentStructure (CDA dokumenta aploksnes struktūra)).

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | | RCMR\_MT000002UV02\_LV01.ClinicalDocument | RCMR\_MT000002UV02\_LV01.ClinicalDocument | 1..\* | Ieejas parametros norādītajam pieprasījumam atbilstošo dokumentu saraksts (sarakstā netiek iekļauts dokumentu teksts – tikai to metadati).  (aprakstu skat. 6.1.6 DocumentStructure (CDA dokumenta aploksnes struktūra) |

Alternatīvie scenāriji

Nav tiesību apskatīt pacienta kartes. Tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte) un izveidots audita ieraksts.

Visiem dokumentiem tiek uzstādīti aizliegumi. Tiek atgriezta tukša atbilde (kas nesatur nevienu dokumentu).

Visām pieprasītajām kartēm aizliegta pieeja. Tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte), un izveidots audita ieraksts.

### generateCCD (Izveidot veselības pamatdatu dokumentu)

Identifikācija

Funkcija: generateCCD

Tiesības: -

Apraksts

Funkcija tiek izsaukta no:

* WebServices::getDocument.

Funkcija ģenerē CDA dokumentu ContinuesCareDocument ar iekļautiem pacienta veselības pamatdatiem laika periodā, ar iespēju norādīt un iekļaut dokumentā tikai nepieciešamos veselības pamatdatu blokus.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | | NPatientID | GUID | 1..1 | Pacienta identifikators |
| 2 | | identificationType | OID | 1..1 | Pacienta identifikācijas tips |
| 3 | | identificationCode |  | 1..1 | Pacienta identifikācijas kods |
| 4 | | DateFrom | Date | 0..1 | Pamatdatu ierakstu sākuma datums |
| 5 | | DateTill | Date | 0..1 | Pamatdatu ierakstu beigu datums |
| 6 | | UserRights |  | 1..1 | Lietotāja tiesību deskriptors |
| 7 | | requestedSections | II | 0..\* | Dokumenta sekciju pieprasījums |
| 8 | | dataFormats | CS | 0..1 | Dokumenta formatēšanas izvēle (XML, HTML vai PDF) |
| 9 | | fillUserComments | BL | 0..1 | Vai pievienot lietotāja komentārus dokumentam |

Apstrādes algoritms

1. Tiek izveidots saraksts ar lietotāja pieprasītām sekcijām, atbilstoši pamatdatu elementiem piešķirtiem OID. Ja lietotāja norādīts sekcijas OID neatbilst CCD iespējamiem, tādi OID tiek ignorēti. Ja pieprasījumā nav norādīta neviena CCD dokumenta sekcija, tad dokumentā tiek iekļautas visas sekcijas izņemot „Visas diagnozes” (OID: 1.3.6.1.4.1.38760.1.2.2.58).
2. No datubāzes tabulām:

* Brīdinājumi - tabula [NPatientWarnings]
* Alerģijas - tabula [NPatientAllergies]
* Diagnozes - tabula [NPatientDiagnosis]
* Medikamenti - tabula [NPatientMedications]
* Medicīniskas iekārtas - tabula [NPatientMedicalDevices]
* Asins pārliešanas – tabula [NPatientTransfusions]
* Ķirurģiskās iejaukšanas – tabula [NPatientSurgeries]
* Visas diagnozes – tabula [NPatientDiagnosisLinkToDocuments]
* Invaliditāte - tabula [NPatientDisability]

1. tiek atlasīti pacienta veselības pamatdati, kuriem izpildās nosacījumi:

* NPatientID = [parametrs: NPatientID];
* Status = [Aktuāls];
* [parametrs:DateFrom] <= Date <=[parametrs:DateTill], ja datumi aizpildīti;
* Diagnožu tabulai (NPatientDiagnoses) tiek pievienots papildu nosacījums – [parametrs:DateTill] <= Date + DateActual <=[parametrs:DateTill] (tas ir, tiek pārbaudīts ne tikai diagnozes konstatācijas datums, bet arī diagnozes aktualitātes datums);
* Tabulas pamatdatu tipa OID ir starp pieprasīto OID komplektā.

Pārbauda lietotāja tiesības iegūt konkrēto veselības pamatdatu informāciju, izmantojot funkciju. 6.5.4 checkUserRightsAccessObject (Pārbaudīt personas tiesību līmeni uz objektu). Datus, kas nav pieejami, tālākai apstrādei neizmanto.

Papildus pārbauda lietotāja tiesības uz pieeju dokumentiem, kuri ir saistīti ar pamatdatiem. Pārbaude tiek veikta, izmantojot pamatdatu sasaistes lauku DocumentTypeOID un atbilstoša dokumenta tipa lauku ViewClaim. Tiek pārbaudīts, vai ViewClaim laukā norādītās tiesības ir lietotāja talonā (ja lietotājs ir āstniecības persona), un, gadījumā ja ViewClaim satur tiesību EvkRghtFamilyDoctor, vai ārstniecības persona ir ģimenes ārsts konkrētam pacientam. Gadījumā, ja neviens no nosacījumiem neizpildās, dokumenta saite tiek izņemta no rezultātu saraksta.

Ja pēc filtrācijas konkrētai diagnozei nepaliek nevienas saites uz dokumentiem, tad diagnoze tiek izņemta no saraksta.

Tiek ģenerēts dokuments pēc šāda algoritma:

Header daļu aizpilda atbilstoši tabulai:

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | templateId | [root] 1.3.6.1.4.1.38760.1.2.1.1  [extension] 1.3.6.1.4.1.38760.1.2.1.1.1 |
| 2 | TypeId | [root] 2.16.840.1.113883.1.3  [extension] POCD\_HD000040 |
| 3 | id | [root] 1.3.6.1.4.1.38760.3.4.5.1  [extension] sistēmas ģenerēts GUID |
| 4 | code | [code] 1.3.6.1.4.1.38760.1.2.1.1  [codeSystem] 1.3.6.1.4.1.38760.1.2.1  [displayName] Medicīnas pamatdatu kopsavilkuma dokuments |
| 5 | title | „Medicīnas pamatdatu kopsavilkuma dokuments” |
| 6 | effectiveTime | Ģenerēšanas datums un laiks |
| 7 | confidentialityCode | „111” |
| 8 | languageCode | „LV” |
| 9 | setId | [root] 1.3.6.1.4.1.38760.3.4.5.4 |
| 10 | versionNumber | Netiek aizpildīts |
| 11 | recordTarget.id | [root] identificationType  [extension] identificationCode |
| 12 | recordTarget name | recordTarget.patientRole.patient.name.given pacienta vārds  recordTarget.patientRole.patient.name.family pacienta uzvārds | |
| 13 | recordTarget addr | Elements: recordTarget.patientRole.addr  addr.country ICAOCode; ICAOName;  addr.censusTract LatvianAddress1Level; ATUName1;  addr.censusTract LatvianAddress2Level; ATUName2;  addr.censusTract LatvianAddress3Level; ATUName3;  addr.censusTract LatvianAddress4Level; ATUName4;  addr.censusTract LatvianAddressStreetCode; StreetCode;  addr.censusTract LatvianAddressStreetCode; VillageCode;  addr.country LatvianAddress1LevelCode; ATUCode1;  addr.country LatvianAddress2LevelCode; ATUCode2;  addr.country LatvianAddress3LevelCode; ATUCode3;  addr.country LatvianAddress4LevelCode; ATUCode4;  addr.county Village;  addr.streetName Street;  addr.houseNumber HouseName;  addr.houseNumberNumeric HouseNumber;  addr.buildingNumberSuffix HouseBlockNumber;  addr.unitID FlatNumber;  addr.postalCode PostalCode  addr.streetAddressLine AddressAbroad | |
| 14 | recordTarget administrativeGenderCode | recordTarget.patientRole.patient.  administrativeGenderCode.code pacienta dzimuma kods;  recordTarget.patientRole.patient.  administrativeGenderCode.displayName pacienta dzimuma koda attēlojamā vērtība | |
| 15 | author | author.assigendAuthor.id.root 1.3.6.1.4.1.38760.3.4.5.29  author.time Ģenerēšanas datums un laiks |
| 16 | Custodian | custodian.assignedCustodian.representedCustodianOrganization.  id.root 1.3.6.1.4.1.38760.3.4.5.29 |

Katram pamatdatu tipam veido sekciju, kuru aizpilda atbilstoši tabulai

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | Brīdinājumi | templateId.[root] 1.3.6.1.4.1.38760.1.2.2.5.1  code.[code] 1.3.6.1.4.1.38760.1.2.2.5  code.[displayName] Pacienta pamatdatu ieraksts brīdinājumam  code.[codeSystem] 1.3.6.1.4.1.38760.2.21  code.[codeSystemName] OID reģistrs  [title] „Brīdinājumi” |
| 2 | Alerģijas | templateId.[root] 1.3.6.1.4.1.38760.1.2.2.1.1  code.[code] 1.3.6.1.4.1.38760.1.2.2.1  code.[displayName] Pacienta pamatdatu alerģijas sekcija  code.[codeSystem] 1.3.6.1.4.1.38760.2.21  code.[codeSystemName] OID reģistrs  [title] „Alerģijas” |
| 3 | Diagnozes | [root] 1.3.6.1.4.1.38760.1.2.2.2.1  code.[code] 1.3.6.1.4.1.38760.1.2.2.2  code.[displayName] Pacienta pamatdatu diagnozes sekcija  code.[codeSystem] 1.3.6.1.4.1.38760.2.21  code.[codeSystemName] OID reģistrs  [title] „Diagnozes” |
| 4 | Medikamenti | [root] 1.3.6.1.4.1.38760.1.2.2.4.1  code.[code] 1.3.6.1.4.1.38760.1.2.2.4  code.[displayName] Pacienta pamatdatu medikamenta sekcija  code.[codeSystem] 1.3.6.1.4.1.38760.2.21  code.[codeSystemName] OID reģistrs  [title] „Medikamenti” |
| 5 | Medicīniskas iekārtas | [root] 1.3.6.1.4.1.38760.1.2.2.3.1  code.[code] 1.3.6.1.4.1.38760.1.2.2.3  code.[displayName] Pacienta pamatdatu medicīnisko iekārtu sekcija  code.[codeSystem] 1.3.6.1.4.1.38760.2.21  code.[codeSystemName] OID reģistrs  [title] „Medicīnas iekārtas” |
| 6 | Asins pārliešanas | [root] 1.3.6.1.4.1.38760.1.2.2.48.1  code.[code] 1.3.6.1.4.1.38760.1.2.2.48  code.[displayName] Pacienta asins pārliešanas sekcija  code.[codeSystem] 1.3.6.1.4.1.38760.2.21  code.[codeSystemName] OID reģistrs  [title] „Asins pārliešanas” |
| 7 | Ķirurģiskās iejaukšanas | [root] 1.3.6.1.4.1.38760.1.2.2.49.1  code.[code] 1.3.6.1.4.1.38760.1.2.2.49  code.[displayName] Pacienta ķirurģiskās iejaukšanās  code.[codeSystem] 1.3.6.1.4.1.38760.2.21  code.[codeSystemName] OID reģistrs  [title] „Ķirurģiskās iejaukšanas” |
| 8 | Visas diagnozes | [root] 1.3.6.1.4.1.38760.1.2.2.58.1  code.[code] 1.3.6.1.4.1.38760.1.2.2.58  code.[displayName] Visas diagnozes  code.[codeSystem] 1.3.6.1.4.1.38760.2.21  code.[codeSystemName] OID reģistrs  [title] „Visas diagnozes” |
| 9 | Invaliditāte | [root] 1.3.6.1.4.1.38760.1.2.2.142.1  code.[code] 1.3.6.1.4.1.38760.1.2.2.142  code.[displayName] Pacienta invaliditātes sekcija  code.[codeSystem] 1.3.6.1.4.1.38760.2.21  code.[codeSystemName] OID reģistrs  [title] „Invaliditāte” |

Sekcijas saturu aizpilda atbilstoši tabulai

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | entry | Entry bloku veido atbilstoši dokumenta „1.3.6.1.4.1.38760.1.2.1.1 - Medicīnas pamatdatu kopsavilkuma dokuments” modelim. |
| 2 | text | Netiek aizpildīts |

Entry elementa atribūtus aizpilda atbilstoši tabulai

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | Brīdinājumi | Template.[root] 1.3.6.1.4.1.38760.1.2.3.5.1 |
| 2 | Alerģijas | Template.[root] 1.3.6.1.4.1.38760.1.2.3.1.1 |
| 3 | Diagnozes | Template.[root] 1.3.6.1.4.1.38760.1.2.3.2.1 |
| 4 | Medikamenti | Template.[root] 1.3.6.1.4.1.38760.1.2.3.4.1 |
| 5 | Medicīniskās iekārtas | Template.[root] 1.3.6.1.4.1.38760.1.2.3.3.1 |
| 6 | Asins pārliešanas | Template.[root] 1.3.6.1.4.1.38760.1.2.3.243.1 |
| 7 | Ķirurģiskās iejaukšanas | Template.[root] 1.3.6.1.4.1.38760.1.2.3.246.1 |
| 8 | Visas diagnozes | Template.[root] 1.3.6.1.4.1.38760.1.2.3.294.1 |
| 9 | Invaliditāte | Template.[root] 1.3.6.1.4.1.38760.1.2.3.781.1 |

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | | CCD dokuments | XML | 1..1 | CDA dokumenta XML. |

Alternatīvie scenāriji

Pacienta karte neeksistē. Tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte) un izveidots audita ieraksts.

### getPatientCard (Iegūt pacienta karti)

Identifikācija

Funkcija: getPatientCard

Tiesības: EvkRghtPatientCardGet

Apraksts

Funkcija atgriež pacienta kartes pamatinformāciju (demogrāfijas informāciju). Netiek atgriezta medicīniskā rakstura informācija, un medicīnas dokumenti.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | | PRPA\_MT201307UV02 | PRPA\_MT201307UV02 | 1..1 | Pieprasījumu parametru saraksts.  6.1.4 PersonQueryByParameter (Personas datu pieprasījums) |

Apstrādes algoritms

1. Tiek pārbaudīts, vai lietotājam ir tiesības izsaukt šo funkciju (6.5.5 checkUserRight (Pārbaudīt personas tiesību)).

Ja lietotājs ir pacients, tiek pārbaudīts, vai lietotājam ir tiesības piekļūt pieprasītā pacienta kartei (skat. 6.5.3 checkRigthsAccessPatientCard (Pārbaudīt tiesības piekļūt pacienta kartei)).

1. Tiek pārbaudīts, vai pacienta karte lietotājam ir pieejama, pārbaudot uzstādīto aizliegumu un lietotāja tiesību deskriptoru (skat. 6.5.4 checkUserRightsAccessObject (Pārbaudīt personas tiesību līmeni uz objektu)). Ja tiesību nav, tiek atgriezts kļūdas ziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte). **Pacienta aizliegumu pārbaude netiek veikta ārstniecības personām** (ārstniecības personām ir pieejamas visu pacientu kārtes demogrāfiskā informācija, arī tajos gadījumos, kad pacients ir aizliedzis pieeju savai kartei).

Pieprasītajai pacienta kartei tiek aizpildīta izejas datu struktūra (skat. 6.1.2 Person (Pacienta kartes pamatdati)). Katram saistītajam datu blokam pārbauda piekļuves tiesības:

* EvkRghtPersonDataGet - Personas dati
* EvkRghtContactsGet – Kontaktinformācija
* EvkRghtContactPersonsGet – Kontaktpersonu dati
* EvkRghtFamilyDoctorGet – Ģimenes ārsta dati
* EvkRghtEhicGet – EVAK kartes dati
* EvkRghtDelegationsGet – Pilnvarojumu dati (atļauju tips =”Pilnvarojums”)
* EvkRghtPermissionsGet – Visu atļauju (t.sk. pilnvarojumi, vecāki, izmeklētāji, u.c.) dati
* PersonDelegation datu struktūru aizpilda atkarībā no parametra Direction:

1. Ja parametrs ir „IN”, tad tiek izpildīts datu bāzes pieprasījums, kas atlasa no PatientDelegations tabulas visus ierakstus, kam ieejas parametrs PatientID vienāds ar PatientDelegations.PatientID, ieejas parametrs ActiveDate starp DelegatedTimeFrom un DelegatedTimeTo (pieņemot, ka DelegatedTimeTo var būt nenorādīts) un DelegatedSystem ir „EVK” (no klasifikatora e-veselības sistēmas) un Statuss ir ACTIVE (no klasifikatora Atļauju statusi)

2. Ja parametrs ir „OUT”, tad no tabulas Patients tiek atlasīta ieejas parametram PatientID atbilstoša identifikācija (lauki IdentificationType un IdentifcationCode). Tad tiek izpildīts datu bāzes pieprasījums, kas atlasa no PatientDelegations tabulas visus ierakstus, kam lauki IdentificationType un IdentificationCode sakrīt ar atlasītiem, ieejas parametrs ActiveDate starp DelegatedTimeFrom un DelegatedTimeTo (pieņemot, ka DelegatedTimeTo var būt nenorādīts) un DelegatedSystem ir „EVK” (no klasifikatora e-veselības sistēmas) un Statuss ir ACTIVE (no klasifikatora Atļauju statusi).

Tiek pārbaudīts, vai pacienta invaliditātes stāvoklis ir aktuāls, un gadījumā, ja dati ir novecojuši (tiek pārbaudīts pēc sistēmas parametra), tiek izsaukta invaliditātes datu atjaunošana (skat. 6.8.1 getDisabilityData (Iegūt personas invaliditātes datus))

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | | PRPA\_MT201303UV02\_LV01 | PRPA\_MT201303UV02\_LV01 | 1..1 | Personu datu saraksts. Skat. ( 6.1.2 Person (Pacienta kartes pamatdati) ) |

Alternatīvie scenāriji

Nav tiesības apskatīt pacienta kartes. Tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte) un izveidots audita ieraksts.

Visām pieprasītām kartēm aizliegta pieeja. Tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte) un izveidots audita ieraksts.

### mergePatientCards (Sapludināt pacienta kartes)

Identifikācija

Funkcija: mergePatientCards

Tiesības: EvkRghtPatientCardsMerge

Apraksts

Funkcija apvieno pacienta kartes ierakstus un atgriež tehnisko ziņojumapmaiņas aploksnes datu struktūru (skat. 6.1.10.1 ack (Tehniskā atbildes datu struktūra)).

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | PRPA\_MT201390UV02\_LV01.PersonCardOperations | PRPA\_MT201390UV02\_LV01.PersonCardOperations | 1..1 | Datu struktūra, kas apvieno pacienta identifikatoru un datu operāciju, kuru jāizpilda ar minēto pacientu kartēm |

Apstrādes algoritms

1. Tiek pārbaudīts, vai lietotājam ir tiesības izsaukt šo funkciju (skat. 6.5.5 checkUserRight (Pārbaudīt personas tiesību)).

Tiek pārbaudīts:

* vai ir pareizi uzstādīts operācijas kods (jābūt MERGE);
* vai vienam un tikai vienam no parametriem norādīts typeCode ar vērtību TARGET;
* vai nav pārkāpts ierobežojums, ka gadījumā, ja kartiņai, kas atbilst TARGET parametram, ir ārzemnieka vai nedrošas autentifikācijas personas identifikācijas kods (OID atbilstoši 1.3.6.1.4.1.38760.3.1.8 vai 1.3.6.1.4.1.38760.3.1.7) un kādai no kartēm, kas atzīmēte ar typeCode SOURCE ir Latvijas personas koda identifikācija 1.3.6.1.4.1.38760.3.1.1;
* vai vismaz vienam no parametriem norādīts typeCode ar vērtību SOURCE;
* vai katram no operācijas parametriem norādīts pacienta identifikators (personas kods vai alternatīva identifikācija).

Ja pārbaude neveiksmīga, tad tiek atgriezts kļūdas ziņojums (EVK\_0049 Parametri norādīti nekorekti).

Tiek pārbaudīts, vai EVK datu bāzē eksistē sapludināmās pacientu kartes. Ja kāda no kartēm neeksistē, tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte) un izveidots audita ieraksts.

Tiek izveidots fona apstrādes uzdevums karšu sapludināšanai. Uzdevumam tiek nodota datu struktūra:

1. <MergePatientCard targetPatientGUID="C227ACE4-34D3-4F72-BAA9-00030C91D079" activityGUID="C227ACE4-34D3-4F72-BAA9-00030C91D079" reason="Pamatojums brīvā tekstā">
2. <PatientGUID>AAB78F70-F0DF-4EF5-B2EA-001868061F6B</PatientGUID>
3. <PatientGUID>94B41DD8-7A19-40BA-9815-00385F4AF3C5</PatientGUID>
4. </MergePatientCard>

Procesa apraksts skat. 6.13.4 Pacienta kartiņu sapludināšana.

Tiek izveidota izejas datu struktūra, kas apraksta pacienta kartes veidošanas rezultātu (skat. 6.1.10.1 ack (Tehniskā atbildes datu struktūra)).

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | MCCI\_MT000200UV01.Acknowledgement | MCCI\_MT000200UV01.Acknowledgement | 1..1 | Rezultāta datu struktūra. 6.1.10.1 ack (Tehniskā atbildes datu struktūra) |

Alternatīvie scenāriji

Nav tiesības apvienot pacienta kartes. Tiek atgriezts kļūdas paziņojums (EVK\_0029 Nav tiesību veikt darbību) un izveidots audita ieraksts.

Karte ar pieprasīto identifikāciju neeksistē. Tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte) un izveidots audita ieraksts.

### setPatientCard (Pievienot un papildināt pacienta kartes datus)

Identifikācija

Funkcija: setPatientCard

Tiesības:

* EvkRghtPatientCardSet
* EvkRghtPatientPersonDataSet – Personas dati
* EvkRghtPatientContactsSet – Kontaktinformācija
* EvkRghtPatientContactPersonSet – Kontaktpersonu dati
* EvkRghtPatientDelegationSet – Pilnvarojumu dati (atļauju tips =”Pilnvarojums”)
* EvkRghtPatientPermissionSet – Visu atļauju (t.sk. pilnvarojumi, vecāki, izmeklētāji, u.c.) dati
* EvkRghtPatientCoreSet – Personas pamatdati (vārds, uzvārds, dzimšanas datums utml).

Apraksts

Funkcija saglabā saņemtās izmaiņas pacienta kartē un atgriež izmainīto pacienta kartes pamatinformāciju (demogrāfijas informāciju). Netiek atgriezta medicīniska rakstura informācija un medicīnas dokumenti.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | PRPA\_MT201303UV02\_LV01.Person | PRPA\_MT201303UV02\_LV01.Person | 1..1 | Pacienta kartes datu izmaiņu pieprasījums (datu struktūru skat. 6.1.2 Person (Pacienta kartes pamatdati)) |

Apstrādes algoritms

1. Tiek pārbaudīts, vai lietotājam ir tiesības izsaukt šo funkciju (skat. (6.5.5 checkUserRight (Pārbaudīt personas tiesību)- EvkRghtPatientCardSet).

Tiek pārbaudīts, vai lietotājam ir tiesības piekļūt modificējamai kartei, izsaucot funkciju checkRigthsAccessPatientCard (skat 6.5.3 checkRigthsAccessPatientCard (Pārbaudīt tiesības piekļūt pacienta kartei)).

1. Tiek pārbaudīts, vai pacienta karte eksistē. Ja pacienta karte neeksistē, tad tiek atgriezts kļūdas ziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte).

Tiek pārbaudīts, vai pacienta karte lietotājam ir pieejama, pārbaudot uzstādīto aizliegumu un lietotāja tiesību deskriptoru (skat. 6.5.4 checkUserRightsAccessObject (Pārbaudīt personas tiesību līmeni uz objektu)). Ja tiesību nav, tiek atgriezts kļūdas ziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte).

Pieprasītajai pacienta kartei tiek pievienoti vai papildināti dati. Katram saistītajam datu blokam pārbauda labošanas tiesības:

* EvkRghtPatientPersonDataSet – Personas dati
* EvkRghtPatientContactsSet – Kontaktinformācija
* EvkRghtPatientContactPersonSet – Kontaktpersonu dati
* EvkRghtPatientDelegationSet – Pilnvarojumu dati (atļauju tips =”Pilnvarojums”)
* EvkRghtPatientPermissionSet – Visu atļauju (t.sk. pilnvarojumi, vecāki, izmeklētāji, u.c.) dati
* EvkRghtPatientCoreSet – Personas pamatdati (pielietojami tikai gadījumā, ja personai ir nedroša vai ārzemnieka identifikācija).

Tiek veikta pacienta kartes datu saglabāšana (skat. 6.1.2 Person (Pacienta kartes pamatdati)).

Izmaiņas tiek identificētas ar katra iekļautā elementa atribūtu UpdateMode. Tiek apstrādātas šādas atribūta vērtības:

* A – pievienot jauno datu elementu
* AR – pievienot jauno datu elementu vai aizvietot iepriekšējo elementu ar jauno versiju
* N – nav izmaiņu (izmaiņas tiek ignorētas)
* D – atbilstošais datu elements tiek izdzēsts (loģiski, uzstādot dzēšanas pazīmi).

Visām kartēm, kurām veiksmīgi izdevās saglabāt datu izmaiņas, tiek aizpildīta izejas datu struktūra skat. (6.1.2 Person (Pacienta kartes pamatdati))

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | PRPA\_MT201303UV02\_LV01.Person | PRPA\_MT201303UV02\_LV01.Person | 1..1 | Pacienta kartes pamatdati. (datu struktūru skat 6.1.2 Person (Pacienta kartes pamatdati)) |

Alternatīvie scenāriji

Pacienta karte neeksistē. Tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte), un izveidots audita ieraksts.

Nav tiesības saglabāt izmaiņas pacienta kartē. Tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte) un izveidots audita ieraksts.

### processDocument (Pievienot dokumentu –asinhronā daļa)

Funkcija tiek izsaukta no Windows. Funkcijai kā ieejas parametrs tiek iedots CDA dokuments. Funkcija nodrošina CDA dokumenta pilnu apstrādi. Rezultātā tiek iegūta pazīme par CDA dokumenta apstrādes procesa beigām. Kā arī iniciē veselības pamatdatu apstrādes procesus.

Identifikācija

Funkcija: processDocument

Tiesības: sistēmas funkcija (tiesības netiek pārbaudītas)

Procesa apraksts

1. Validē CDA dokumentu pret modeli - 6.3.1 validateDocumentModel (Validēt dokumentu pret modeli).

* Tiek atgriezts saraksts ar validācijas kļūdām. Validācijas kļūdas tiek reģistrētas procesa mainīgajā „DocumentErrors”.
* Ja validācijas kļūdas nav, tad saraksts ir tukšs, un tiek uzskatīts, ka CDA dokuments pilnībā atbilst modelim.
* Process turpinās ar nākamo soli.

Pārbauda CDA dokumenta klasificētās vērtības - 6.2.12.1 extractAndValidateClassifiedValues (Validēt dokumenta klasificētās vērtības), un veic ārsta un ārstniecības iestādes datu izgūšanu no CDA dokumenta un šo datu saglabāšanu datu bāzes tabulā „Documents”.

* Tiek atgriezts saraksts ar klasificēto vērtību kļūdām. Tās tiek reģistrētas procesa mainīgajā „DocumentErrors”.
* Ja klasificētajās vērtībās kļūdas nav, tad saraksts ir tukšs, un tiek uzskatīts, ka CDA dokumentā iekļautās klasificētās vērtības atbilst klasifikatoriem.
* Process turpinās ar nākamo soli.

Ģenerē CDA dokumenta HTML - 6.2.12.2 generateHTML (HTML ģenerācija no CDA dokumenta).

Procesu turpina ar nākamo soli. Ja „DocumentErrors” saraksts ir tukšs, tad apstrādā CDA dokumenta iepriekšējo aktuālo versiju, ja tāda ir.

* Atrod dokumenta iepriekšējo aktuālo versiju - 6.2.12.3 findDocumentPreviousActualVersion ( atrast dokumenta iepriekšējo aktuālo versiju).
* Ja dokumenta versija atrasta, tad to anulē - 6.2.12.4 setDocumentStatusToCancelled (dokumenta statusa maiņa uz „Anulēts”)
* Procesu turpina ar nākamo soli.

Ja „DocumentErrors” saraksts ir tukšs, tad CDA dokumentam maina statusu uz „Aktuāls” - 6.2.12.5 setDocumentStatusToActual (dokumenta statusa maiņa uz „Aktuāls”).

Procesu turpina ar nākamo soli.

Ja „DocumentErrors” sarakstā reģistrēts vismaz viens ieraksts, tad :

* „DocumentErrors” sarakstā reģistrētās kļūdas saglabā CDA dokumenta kļūdu tabulā datu bāzē.
* CDA dokumentam uzstāda statusu „Kļūdains” - 6.2.12.6 setDocumentStatusToIncorrect (dokumenta statusa maiņa uz „Kļūdains”).
* Dokumenta apstrāde tiek beigta.

Iniciē CDA dokumenta ierakstu apstrādes procesu - 6.2.12.7 documentEntryProcessing (dokumenta ierakstu apstrāde).

Tiek veikta pārbaude, vai dokumenta tips pieder PREDA dokumentiem (piederība tiek pārbaudīta, atlasot no sistēmas parametriem grupu PredaDocOids un pārbaudot, vai dokumenta šablona identifikators ir starp OIDiem, kas definētie šajā grupā), un, gadījumā, ja pieder – tiek veikta papildu dokumenta apstrāde:



PREDA dokumenta apstrāde sākās pēc veiksmīgas CDA dokumenta asinhronas pārbaudes. Dokumentam nav jābūt konstatētām shēmas vai klasifikatoru kļūdu.

Ja dokumentam ir konstatētas sinhronas vai asinhronas apstrādes kļūdas, tam tiek uzstādīts statuss „incorrect” un dokumenta apstrāde tiek pārtraukta.

PREDA dokumenta apstrāde notiek pēc šāda algoritma:

* No CDA dokumenta tipa identifikatora tiek noteikts PREDA reģistrs, uz kuru attiecas konkrētais dokuments.
* Tiek atlasīti PREDA dati par konkrētu pacientu, izmantojot atbilstoša reģistra funkciju (piemērām, Cukura diabēta reģistram tā ir CDLasit). Gadījumā, ja pacients vēl nav reģistrēts atbilstošā reģistrā, funkcija XXLasit atgriezīs tukšu datu kopu.
* Pēc pacienta datu ielādes no PREDA DB, tiek atrasts nosūtīts dokuments (pēc CDA lauka SetId). Gadījumā, ja dokuments nav atrasts PREDA ierakstā, datu kopa tiek papildināta ar jauno dokumentam atbilstošo ierakstu.
* Atrasta vai izveidota dokumenta lauki tiek aizpildīti ar CDA datiem. Atsevišķi PREDA tiek saglabāta CDA dokumenta lauka SetId vērtība. Pēc aizpildīšanas tiek izpildīta PREDA funkcija XXRakstit (XX šajā gadījumā attiecas uz to pašu reģistru, kas Lasit funkcijas izsaukumā, piemērām CDLasit atbilst CDRakstit un PSIHLasit atbilst PSIHRakstit). Gadījumā, ja CDA dokuments aizpildīts atbilstoši PREDA datu saglabāšanas noteikumiem, notiek datu saglabāšana PREDA datu bāzē, un dokumentam tiek uzstādīts statuss „actual” . Gadījumā, ja saglabāšanas rezultātā tiek konstatētas kļūdas, tās tiek pievienotas pie CDA dokumenta kļūdu saraksta, un dokumenta statuss tiek uzstādīts uz „incorrect”

Kļūdas gadījums

Ja kādā no soļiem tiek konstatēta neparedzēta kļūda, tad dokumenta apstrādes process tiek pārtraukts, un tiek uzsākts atrites (*rollback*) process datu bāzē veiktajām darbībām. CDA dokumentu, kuru neizdevās apstrādāt, atstāj rindā un apstrādā atkārtoti.

Atkārtotu apstrādes mēģinājumu skaits ir konfigurējams parametrs.

#### extractAndValidateClassifiedValues (Validēt dokumenta klasificētās vērtības)

Identifikācija

Funkcija: extractAndValidateClassifiedValues

Ieejas parametri

Funkcijai kā ieejas parametrs ir CDA dokuments.

Apraksts

No CDA dokumenta tiek iegūtas visas klasificējamās vērtības, kas tiek ievietotas sarakstā:

* OrderNr – saraksta iekšējais identifikators (autoincrement);
* Code – klasificējamās vērtības kods;
* DisplayName – attēlojamā vērtība;
* CodeSystem – klasifikatora identifikators „OID”;
* CodeSystemVersion – Klasifikatora versija – datu lauks tiek aizpildīts, ja klasificējamā vērtība CDA dokumentā pēc modeļa uzbūves to satur (elements ar tipu „CD”);
* Date – datuma lauks tiek aizpildīts, ja klasificējamā vērtība CDA dokumentā pēc modeļa uzbūves nesatur versiju (elements „id” ar bāzes tipu „II”), kur datums tiek paņemts no CDA dokumenta „header” sadaļas elementa „effectiveTime”;

Klasifikatoru validācijas saraksta aizpildīšana.

CDA dokumenta „header” daļas apstrāde:

* Dokumenta klasifikators
* XPath: /ClinicalDocument/code; Tiek iegūti atribūti:
* Code = (./@code)[1]
* DisplayName = (./@displayName)[1];
* CodeSystem = (./@codeSystem)[1];
* CodeSystemName = (./@codeSystemName)[1];
* CodeSystemVersion = (./@codeSystemVersion)[1].
* Konfidencialitātes klasifikators
* XPath: /ClinicalDocument/confidentialityCode; Tiek iegūti atribūti:
* Code = (./@code)[1]
* DisplayName = (./@displayName)[1];
* CodeSystem = (./@codeSystem)[1];
* CodeSystemName = (./@codeSystemName)[1];
* CodeSystemVersion = (./@codeSystemVersion)[1].
* Dzimuma klasifikators
* XPath: /ClinicalDocument/recordTarget/patientRole/patient/administrativeGenderCode; Tiek iegūti atribūti:
* Code = (./@code)[1]
* DisplayName = (./@displayName)[1];
* CodeSystem = (./@codeSystem)[1];
* CodeSystemName = (./@codeSystemName)[1];
* CodeSystemVersion = (./@codeSystemVersion)[1].
* Ārstniecības personas klasifikators
* XPath: /ClinicalDocument/author/assignedAuthor/id; Tiek iegūti atribūti:
* Code = (./@extension)[1]
* CodeSystem = (./@root)[1];
* Date = XPath: /ClinicalDocument/effectiveTime (./@value)[1].

Iegūtā vērtība Code tiek piefiksēta atmiņas mainīgajā „AuthorExtension”, vērtība tiks izmantota zemāk.

* Ārstniecības iestādes klasifikators
* XPath: /ClinicalDocument/custodian/assignedCustodian/representedCustodianOrganization; Tiek iegūti atribūti:
* Code = (./id/@extension)[1]
* DisplayName = (./name)[1]', 'nvarchar(1000);
* CodeSystem = (./id/@root)[1]', 'nvarchar(100);
* Date = XPath: /ClinicalDocument/effectiveTime (./@value)[1].

Iegūtā vērtība Code tiek piefiksēta atmiņas mainīgajā „CustodianExtension”, vērtība tiks izmantota zemāk.

CDA dokumenta „body” daļas apstrāde notiek, piemērojot vienu universālu XPath, iegūstot sarakstu ar klasificējamām vērtībām:

* XPath: /ClinicalDocument/component/structuredBody/component/section//\*[./@code]; Tiek iegūti atribūti:
* Code = (./@code)[1]
* DisplayName = (./@displayName)[1];
* CodeSystem = (./@codeSystem)[1];
* CodeSystemName = (./@codeSystemName)[1];
* CodeSystemVersion = (./@codeSystemVersion)[1].

Tiek izsaukta Klasifikatora moduļa saskarnes metode „ValidateConcepts”, kurai kā ieejas parametrs tiek nodots iegūtais vērtību saraksts. Metodes „ValidateConcepts” atbilde:

* Kļūdas ziņojums.

Nākamā darbība: Validācijas funkcijas darbība tiek pārtraukta un šī kļūda tiek nodota izsaucošajai funkcijai (Exception).

* Saraksts ar klasifikatora validācijas rezultātu. Saraksts satur šādus laukus:
* OrderNr – klasificējamās vērtības identifikators;
* Validācijas rezultāts – pazīme, pēc kuras var noteikt, vai klasificējamā vērtība ir izturējusi validāciju iespējamās vērtības {valid; invalidCodeSystem; invalidCodeSystemVersion; invalidCode; invalidDisplayName}.

No iegūtā saraksta tiek atlasīti tie ieraksti, kuriem validācijas rezultāts nav „valid”. Katram no šiem ierakstiem tiek veidots ieraksts metodes rezultāta sarakstā, norādot pilnu informāciju par elementu, kas neatbilst klasifikācijai: Code, DisplayName, CodeSystem, CodeSystemName, CodeSystemVersion, Validācijas rezultāts.

No CDA dokumenta tiek iegūtas papildus vērtības un saglabātas datu bāzes tabulā „Documents” pie attiecīgā datu bāzes ieraksta, kur tabulas lauks „DocumentID” sakrīt ar vērtību no CDA dokumenta header daļas elementa „id” atribūta „extension”.

Ārstniecības persona jau tika izgūta no dokumenta, veicot klasificēto vērtību validāciju. Vērtība no atmiņas mainīgā „AuthorExtension” tiek ierakstīta datu bāzes tabulas laukā „Documents.AuthorExtension”.

Autora pilnā XML struktūra tiek iegūta pēc XPath: „//\*/author” un ierakstīta datu bāzes tabulas laukā „Documents.Author”.

Ārstniecības iestāde jau tika izgūta no dokumenta, veicot klasificēto vērtību validāciju. Vērtība no atmiņas mainīgā „CustodianExtension” tiek ierakstīta datu bāzes tabulas laukā „Documents.CustodianExtension”.

Rezultāts

Funkcijas rezultāts ir saraksts ar neatbilstībām, kas tika atrastas klasificēto vērtību validācijā.

Ja funkcijas darbības rezultātā tiek iegūta neapstrādāta kļūda, tad validācijas funkcijas darbība tiek pārtraukta un šī kļūda tiek nodota izsaucošajai funkcijai (Exception).

#### generateHTML (HTML ģenerācija no CDA dokumenta)

Identifikācija

Funkcija: generateHTML

Ieejas parametri

Funkcijai kā ieejas parametrs ir CDA dokuments.

Apraksts

No tabulas „DocumentTemplates” tiek piemeklēta apstrādājamā dokumenta šablona ieraksts pēc saites „CDA:TemplateID = DocumentTemplates.DocumentTemplateOID”. No atrastā ieraksta tiek iegūta lauka „XSLTransformation” vērtība.

CDA dokumentam tiek piemērota XSL transformācija, kā rezultātā tiek iegūts HTML. Šis HTML tiek saglabāts pie attiecīgā dokumenta ieraksta tabulā „Documents” (Documents.DocumentID = ClinicalDocument/id:extension).

Rezultāts

Metodes rezultātā ir pazīme, par metodes veiksmīgu izpildi.

Ja funkcijas darbības rezultātā tiek iegūta neapstrādāta kļūda, tad validācijas funkcijas darbība tiek pārtraukta un šī kļūda tiek nodota izsaucošajai funkcijai (Exception).

#### findDocumentPreviousActualVersion ( atrast dokumenta iepriekšējo aktuālo versiju)

Identifikācija

Funkcija: findDocumentPreviousActualVersion

Ieejas parametri

Funkcijai kā ieejas parametrs ir CDA dokuments.

Apraksts

No CDA dokumenta tiek iegūta vērtība no elementa „SetID” atribūta „extension”. Pēc šīs vērtības tiek atlasīts ieraksts no tabulas „Documents”, filtrējot pēc lauka „Documents.SetID” un lauka „Documents.Status”, kura vērtība atbilst statusam „Aktuāls”.

Ja tiek atrasts ieraksts, tad CDA dokumenta iepriekšējās versijas identifikators ir atrastā ieraksta lauka „Documents.DocumentID” vērtība. Pretējā gadījumā CDA dokumenta iepriekšējās versijas identifikators ir „Nedefinēts” (NULL).

Rezultāts

CDA dokumenta iepriekšējās versijas identifikators. Ja iepriekšējo versiju atrast nebija izdevies tad vērtība „Nedefinēts” (NULL).

Ja funkcijas darbības rezultātā tiek iegūta neapstrādāta kļūda, tad validācijas funkcijas darbība tiek pārtraukta un šī kļūda tiek nodota izsaucošajai funkcijai (Exception).

#### setDocumentStatusToCancelled (dokumenta statusa maiņa uz „Anulēts”)

Identifikācija

Funkcija: setDocumentStatusToCancelled

Ieejas parametri

* DocumentID – CDA dokumenta identifikators
* History – lietotājs, kas veic darbību, laiks un jaunais statuss (6.1.11 History)

Apraksts

1. Tiek sameklēts dokumenta ieraksts datu bāzes tabulā „Documents” pēc parametra DocumentID.

* Dokumenta ierakstam laukā „History” tiek pievienots History XML elements.

Dokumenta ierakstam tiek uzstādīts statuss „Anulēts”, mainot lauka „Statuss” vērtību uz atbilstošu statusam „Anulēts”.

Atrastajam dokumenta ierakstam tiek apstrādāti veselības pamatdati secīgi pa grupām:

| Nr. | Veselības pamatdatu grupa | Tabula | Sasaistes tabula uz dokumentu | |
| --- | --- | --- | --- | --- |
| 1. | Alerģijas | NPatientAllergies  Ēnu tabula: NPatientAllergiesModif | NPatientAllergiesLinkToDocuments  Ēnu tabula:  NPatientAllergiesLinkToDocumentsModif |
| 2. | Diagnozes | NPatientDiagnosis  Ēnu tabula: NPatientDiagnosisModif | NPatientDiagnosisLinkToDocuments  Ēnu tabula:  NPatientDiagnosisLinkToDocumentsModif |
| 3. | Medicīniskās iekārtas | NPatientMedicalDevices  Ēnu tabula: NPatientMedicalDevicesModif | NPatientMedicalDevicesLinkToDocuments  Ēnu tabula:  NPatientMedicalDevicesLinkToDocumentsModif |
| 4. | Medikamenti | NPatientMedications  Ēnu tabula: NPatientMedicationsModif | NPatientMedicationsLinkToDocuments  Ēnu tabula:  NPatientMedicationsLinkToDocumentsModif |
| 5. | Brīdinājumi | NPatientWarnings  Ēnu tabula: NPatientWarningsModif | NPatientWarningsLinkToDocuments  Ēnu tabula:  NPatientWarningsLinkToDocumentsModif |
| 6. | Asins pārliešanas | NPatientTransfusions  Ēnu tabula: NPatientTransfusionsModif | NPatientTransfusionsLinkToDocuments  Ēnu tabula:  NPatientTransfusionsLinkToDocumentsModif |
| 7. | Ķirurģiskās iejaukšanas | NPatientSurgeries  Ēnu tabula: NPatientSurgeriesModif | NPatientSurgeriesLinkToDocuments  Ēnu tabula:  NPatientSurgeriesLinkToDocumentsModif |
| 8. | Invaliditāte | NPatientDisability  Ēnu tabula:  NPatientDisabilityModif | NPatientDisabilityLinkToDocuments  Ēnu tabula:  NPatientDisabilityLinkToDocumentsModif |

Veselības pamatdatu un sasaistes tabulā tiek dzēsta saite starp dokumentu un veselības pamatdatu ierakstu. Dzēstā saite tiek pārvietota uz ēnu tabulu. Tai tiek pievienots datums un iemesls, kas atbilst vērtībai „Dokumenta statusa maiņa”.

Pārbauda, vai veselības pamatdatu ierakstam ir sasaiste ar kādu dokumentu. Ja sasaistes nav, tad veselības pamatdatu ieraksts tiek dzēsts. Tiek izveidota dzēstā ieraksta kopija ēnu tabulā. Tai tiek pievienots datums un iemesls, kas atbilst vērtībai „Dzēstas visas sasaistes”.

#### setDocumentStatusToActual (dokumenta statusa maiņa uz „Aktuāls”)

Identifikācija

Funkcija: setDocumentStatusToActual

Ieejas parametri

* DocumentID – CDA dokumenta identifikators
* History – lietotājs, kas veic darbību, laiks un jaunais statuss (6.1.11 History)

Apraksts

1. Tiek sameklēts dokumenta ieraksts datu bāzes tabulā „Documents” pēc parametra DocumentID.
2. Dokumenta ierakstam laukā „History” tiek pievienots History XML elements.
3. Dokumenta ierakstam tiek uzstādīts statuss „Aktuāls”, mainot lauka „Statuss” vērtību uz atbilstošu statusam „Aktuāls”.

Rezultāts

Funkcijas rezultāts ir pazīme par veiksmīgu/neveiksmīgu izpildi.

Ja funkcijas darbības rezultātā tiek iegūta neapstrādāta kļūda, tad validācijas funkcijas darbība tiek pārtraukta, un kļūda tiek nodota izsaucošajai funkcijai (Exception).

#### setDocumentStatusToIncorrect (dokumenta statusa maiņa uz „Kļūdains”)

Identifikācija

Funkcija: setDocumentStatusToIncorrect

Ieejas parametri

* DocumentID – CDA dokumenta identifikators
* History – lietotājs, kas veic darbību, laiks un jaunais statuss (6.1.11 History)

Apraksts

1. Tiek sameklēts dokumenta ieraksts datu bāzes tabulā „Documents” pēc parametra DocumentID.
2. Dokumenta ierakstam laukā „History” tiek pievienots History XML elements.
3. Dokumenta ierakstam tiek uzstādīts statuss „Kļūdains”, mainot lauka „Statuss” vērtību uz atbilstošu statusam „Kļūdains”.

Rezultāts

Funkcijas rezultāts ir pazīme par veiksmīgu/neveiksmīgu izpildi.

Ja funkcijas darbības rezultātā tiek iegūta neapstrādāta kļūda, tad validācijas funkcijas darbība tiek pārtraukta, un kļūda tiek nodota izsaucošajai funkcijai (Exception).

#### documentEntryProcessing (dokumenta ierakstu apstrāde)

Identifikācija

Funkcija: documentEntryProcessing

Ieejas parametri

CDA dokuments

Apraksts

1. No CDA dokumenta tiek izgūta informācija par dokumentā iekļauto ierakstu „Entry” veidiem. Darbības rezultātā tiek aizpildīts saraksts ar laukiem (tālākā saraksta apstrādē saraksts tiek identificēts kā „Entry” saraksts):

* DocumentTemplateOID – tiek iegūts no CDA dokumenta pēc XPath „(/ClinicalDocument/templateId)[1]/@root”
* SectionTemplateOID – no dokumenta tiek iegūta katras sekcijas „templateId” atribūta „root” vērtība. XPath „/ClinicalDocument/component/structuredBody/component/section”, kur no katras sekcijas tiek iegūta elementa „templateId” atribūta „root” vērtība.
* EntryTemplateOID – no katras atrastās sekcijas tiek atlasīti visi sekcijā iekļautie pirmā dziļuma līmeņa „entry” elementi pēc ceļa „entry/\*/templateId”, kur „\*” var būt viena no vērtībām: observation, regionOfInterest, observationMedia, substanceAdministration, supply, procedure, encounter, organizer, act un iegūtas elementa „templateId” atribūta „root” vērtības.
* DocumentID – tiek iegūts no CDA dokumenta „header” daļas lauka „id” atribūta „extension”.
* NPatientID – tiek iegūts no tabulas „Documents” lauka „NPatientID”, filtrējot pēc lauka „DocumentID”.
* EntryXML – apstrādājot katru entry pie „EntryTemplateOID” iegūšanas, tiek piefiksēts arī šī entry pilns XML.

Katram „Entry” saraksta ierakstam tiek pielasītas atbilstošās rindas no tabulas „DocumentProcessingTasks” pēc filtriem:

* **Visi Entry, neatkarīgi no sekcijas tipa un neatkarīgi no dokumenta tipa:**

DocumentProcessingTasks.EntryTemplateOID = Saraksts.EntryTemplateOID UN DocumentProcessingTasks.SectionTemplateOID nav norādīts UN DocumentProcessingTasks. DocumentTemplateOID nav norādīts

* **Visi Entry pie noteikta sekcijas tipa un neatkarīgi no dokumenta tipa:**

DocumentProcessingTasks.EntryTemplateOID = Saraksts.EntryTemplateOID UN DocumentProcessingTasks.SectionTemplateOID = Saraksts.SectionTemplateOID UN DocumentProcessingTasks. DocumentTemplateOID nav norādīts

* **Visi Entry pie noteikta sekcijas tipa un pie noteikta dokumenta tipa:**

DocumentProcessingTasks.EntryTemplateOID = Saraksts.EntryTemplateOID UN DocumentProcessingTasks.SectionTemplateOID = Saraksts.SectionTemplateOID UN DocumentProcessingTasks. DocumentTemplateOID = Saraksts.DocumentTemplateOID

Izmantojot no CDA dokumenta izgūto „Entry” sarakstu un tam piesaistīto sarakstu no tabulas „DocumentProcessingTasks”, tiek veidots saraksts „ProcessableTasks” ar šādu struktūru:

| Field | Izgūst no | Ieguve |
| --- | --- | --- |
| Task | DocumentProcessing  Tasks | DocumentProcessingTasks.Task  (Allergies, Medications, MedicalDevices, Warnings, Diagnosis, Surgeries, MedicalDevices, Disability) |
| TaskType | DocumentProcessing  Tasks | DocumentProcessingTasks.TaskType  (Internal vai External) |
| DocumentID | Entry saraksts | no saraksta „Entry” lauka „DocumentID” |
| NPatientID | Entry saraksts | no saraksta „Entry” lauka „NPatientID” |
| DocumentTemplateID | Entry saraksts | no saraksta „Entry” lauka „DocumentTemplateID” |
| SectionTemplateID | Entry saraksts | no saraksta „Entry” lauka „SectionTemplateID” |
| EntryTemplateID | Entry saraksts | no saraksta „Entry” lauka „EntryTemplateID” |
| NegationInd | EntryXML | Piemēro XPath no „DocumentProcessingTasks.NegationIndXPath” |
| Fields |  | Medicīniskā pamatdatu ieraksta atribūti. Viens atribūts atbilst vienam ierakstam tabulā DocumentProcessingTaskFields. |
| FieldName | DocumentProcessingTaskFields | lauka „DocumentProcessingTaskFields.Field” vērtība |
| FieldValue | EntryXML | Piemēro XPath no „DocumentProcessingTaskFields.XPath” |

1. Saraksta „ProcessingTasks” ierakstus nodod fona uzdevumam asinhronai apstrādei - 6.13.2 Dokumentu ierakstu apstrādes process.

* Ja TaskType = Internal, tad saņēmēja serviss = InternalTaskService
* Ja TaskType = External, tad saņēmēja serviss = ExternalTaskService

Rezultāts

Funkcijas rezultāts ir pazīme par veiksmīgu/neveiksmīgu izpildi.

Ja funkcijas darbības rezultātā tiek iegūta neapstrādāta kļūda, tad darbība tiek pārtraukta, un kļūda tiek nodota izsaucošajai funkcijai (Exception).

### internalTaskProcessing (pamatdatu uzdevumu apstrāde)

Identifikācija

Funkcija: InternalTaskProcessing

Ieejas parametri

ProcessableTasks datu struktūra (sk. 6.2.12.7 documentEntryProcessing (dokumenta ierakstu apstrāde)).

Piemērs:

<Task>

<TaskName>NPatientAllergiesTask</TaskName>

<TaskID>1</TaskID>

<DocumentID>14897.2.1</DocumentID>

<NPatientID>1265</NPatientID>

<TaskData>

<AlergyCodeSystem>1.3.6.1.4.1.38760.1.2.3</AlergyCodeSystem>

<AlergyCode>1.3.6.1.4.1.38760.1.2.3.1</AlergyCode>

<AlergyDisplayValue>Pacienta pamatdatu ieraksts aler?ijai</AlergyDisplayValue>

<AlergyGroupCodeSystem>1.3.6.1.4.1.38760.1.2.3</AlergyGroupCodeSystem>

<AlergyGroupCode>1.3.6.1.4.1.38760.1.2.3.83</AlergyGroupCode>

<AlergyGroupDisplayValue>Pacienta aler?ijas grupa</AlergyGroupDisplayValue>

<Date>20111111</Date>

</TaskData>

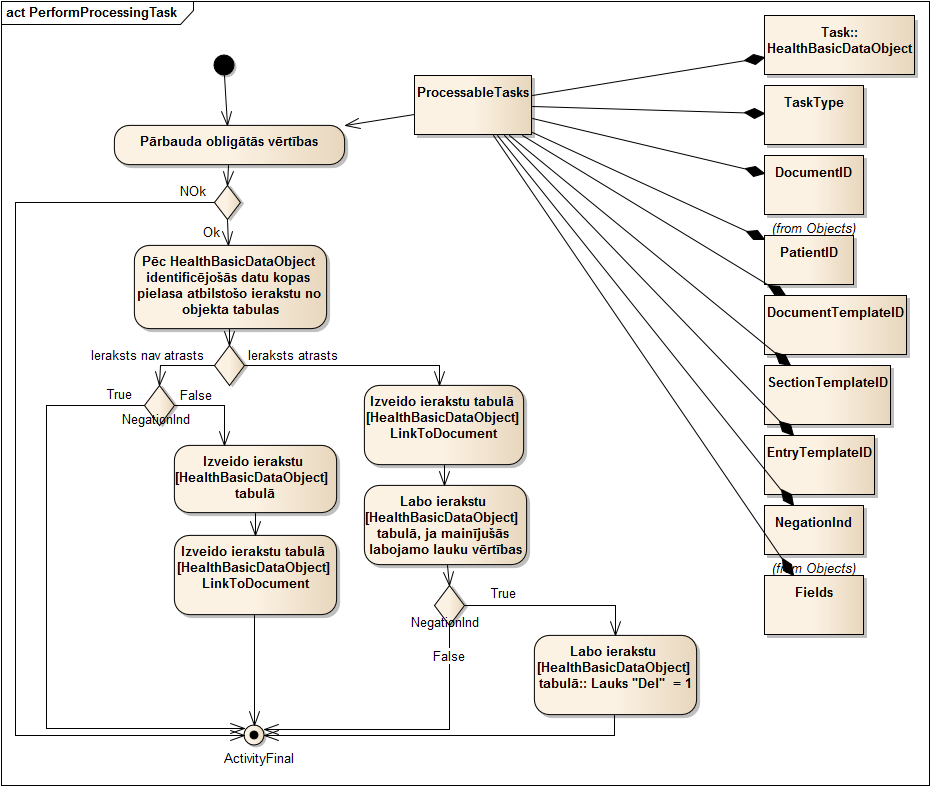
</Task>', 1

Apraksts

1. Tiek veikta katra saraksta „ProcessableTasks” elementa apstrāde. Elementa „Task” vērtība nosaka veselības pamatdatu objektu, uz kuru attiecas minētais saraksta ieraksts. Process veic diferencētu ieraksta apstrādi atkarībā no elementa „Task” vērtības.

6.2‑1. tabula. Veselības pamatdatu objektu atbilstības tabula

| DB tabulas lauks | Field vērtība | Obl | ID | Apraksts |
| --- | --- | --- | --- | --- |
| Allergies - Alerģijas | | | | |
| PatientAllergyID |  |  |  | Ieraksta EntryID  Ja vērtība norādīta, tad operācija ir „Labot”, ja nav norādīta, tad operācija ir „Jauns”. |
| NPatientID |  | Jā | x | Ieraksta NPatientID, turpmāk izmantots kā „@NPatientID” |
| AlergyCodeSystem | Alergy CodeSystem | Jā | x | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘AlergyCodeSystem”, un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| AlergyCode | AlergyCode | Jā | x | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘AlergyCode’”, un vērtība tiek paņemta no atbilstoša „FieldValue” lauka, turpmāk izmantots kā „@AlergyCode”. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| AlergyDisplayValue | Alergy DisplayValue | Jā |  | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘AlergyDisplayValue’”, un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| DateFirst | Date | Jā |  | Date = vērtība tiek sameklēta starp „Date” vērtībām, kur laukā „FieldName = ‘Date’”, un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| DateLast | Date | Jā |  |  |
| AccessRights |  | Jā |  | Tiek sameklēts tabulas „Documents” ieraksts, kas atbilst filtram „Documents.DocumentID = ieraksts.DocumentID”, no atrastā dokumenta ieraksta tiek izmantota lauka „AccessRights” vērtība. |
| AlergyGroup CodeSystem | AlergyGroup CodeSystem |  |  | Klasifikators alerģijas grupai |
| AlergyGroupCode | AlergyGroup Code |  |  | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘GroupID’”, un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| AlergyGroup DisplayValue | AlergyGroup DisplayValue | Jā |  |  |
| History | History |  |  |  |
| ActivityGUID |  |  |  |  |
| ModifDate |  |  |  |  |
| Del |  |  |  |  |
| Diagnosis - Diagnozes | | | | |
| PatientDiagnosisID |  |  |  | Ieraksta EntryID  Ja vērtība norādīta, tad operācija ir „Labot”, ja nav norādīta, tad operācija ir „Jauns”. |
| NPatientID |  |  | x | Ieraksta NPatientID, turpmāk izmantots kā „@NPatientID” |
| Diagnosis CodeSystem | Diagnosis CodeSystem | Jā | x | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘DiagnosisCodeSystem” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| DiagnosisCode | DiagnosisCode | Jā | x | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘DiagnosisCode” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka, turpmāk izmantots kā „@DiagnosisCode”. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| Diagnosis DisplayValue | Diagnosis DisplayValue | Jā |  | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘DiagnosisDisplayValue” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| DateFirst | Date | Jā |  | Date = vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘Date” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| DateLast | Date | Jā |  |  |
| AccessRights |  | Jā |  | Tiek sameklēts tabulas „Documents” ieraksts, kas atbilst filtram „Documents.DocumentID = ieraksts.DocumentID”, no atrastā dokumenta ieraksta tiek izmantota lauka „AccessRights” vērtība. |
| ActivityGUID |  |  |  |  |
| ModifDate |  |  |  |  |
| Del |  |  |  |  |
| MedicalDevices - Medicīniskās iekārtas | | | | |
| PatientMedical DeviceID |  |  |  | Ieraksta EntryID  Ja vērtība norādīta, tad operācija ir „Labot”, ja nav norādīta, tad operācija ir „Jauns”. |
| NPatientID |  |  | x | Ieraksta NPatientID, turpmāk izmantots kā „@NPatientID” |
| DeviceCodeSystem | Device CodeSystem |  | x | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘DeviceCodeSystem” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| DeviceCode | DeviceCode |  | x | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘DeviceCode” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka, turpmāk izmantots kā „@DeviceCode”. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| DeviceDisplayValue | Device DisplayValue |  |  | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘DeviceDisplayValue” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| Manufacturer | Manufacturer | Nē |  | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘Manufacturer” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| Series | Series | Nē | x | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘Series” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| MedicalInstitution CodeSystem | MedicalInstitution CodeSystem | Nē |  | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘MedicalInstitutionCodeSystem” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| MedicalInstitution Code | MedicalInstitution Code | Nē |  | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘MedicalInstitutionCode” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka, turpmāk izmantots kā „@DeviceCode”. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| MedicalInstitution DisplayValue | MedicalInstitution DisplayValue | Nē |  | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘MedicalInstitutionDisplayValue” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| DateFirst | Date |  |  | Date = vērtība tiek sameklēta starp „Date” vērtībām, kur laukā „FieldName = ‘Date” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| DateLast | Date |  |  |  |
| AccessRights |  |  |  | Tiek sameklēts tabulas „Documents” ieraksts, kas atbilst filtram „Documents.DocumentID = ieraksts.DocumentID”, no atrastā dokumenta ieraksta tiek izmantota lauka „AccessRights” vērtība. |
| ActivityGUID |  |  |  |  |
| ModifDate |  |  |  |  |
| Del |  |  |  |  |
| Medications - Medikamenti | | | | |
| Patient MedicationID |  |  |  | Ieraksta EntryID  Ja vērtība norādīta, tad operācija ir „Labot”, ja nav norādīta, tad operācija ir „Jauns”. |
| NPatientID |  |  | x | Ieraksta NPatientID, turpmāk izmantots kā „@NPatientID” |
| Medication CodeSystem | Medication CodeSystem |  | x | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘MedicationCodeSystem” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| MedicationCode | MedicationCode |  | x | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘MedicationCode” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka, turpmāk izmantots kā „@MedicationCode”. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| Medication DisplayValue | Medication DisplayValue |  |  | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘MedicationDisplayValue” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| DateFrom | DateFrom | Nē | x | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘DateFrom” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| DateTo | DateTo | Nē | x | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘DateTo” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| Notes | Notes |  |  | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘Notes” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| RelatedEntryXML | RelatedEntryXML | Nē |  | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘RelatedEntryXML” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| DateFirst | Date |  |  | Date = vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘Date” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| DateLast | Date |  |  |  |
| AccessRights |  |  |  | Tiek sameklēts tabulas „Documents” ieraksts, kas atbilst filtram „Documents.DocumentID = ieraksts.DocumentID”, no atrastā dokumenta ieraksta tiek izmantota lauka „AccessRights” vērtība. |
| ActivityGUID |  |  |  |  |
| ModifDate |  |  |  |  |
| Del |  |  |  |  |
| Warnings - Brīdinājumi | | | | |
| PatientWarningID |  |  | x | Ja vērtība norādīta, tad operācija ir „Labot”, ja nav norādīta, tad operācija ir „Jauns”. |
| NPatientID |  |  | x | Ieraksta NPatientID, turpmāk izmantots kā „@NPatientID” |
| Warnings CodeSystem | Warnings CodeSystem | Jā | x | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘WarningsCodeSystem” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| WarningsCode | WarningsCode | Jā | x | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘WarningsCode” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka, turpmāk izmantots kā „@WarningsCode”. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| Warnings DisplayValue | Warnings DisplayValue | Jā |  | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘WarningsDisplayValue” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| Description | Description | Jā | x | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘Description” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| DateFirst | Date |  |  | Date = vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘Date” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| DateLast | Date |  |  |  |
| EntryXML |  |  |  | Vērtība no ieraksta EntryXML |
| AccessRights |  |  |  | Tiek sameklēts tabulas „Documents” ieraksts, kas atbilst filtram „Documents.DocumentID = ieraksts.DocumentID”, no atrastā dokumenta ieraksta tiek izmantota lauka „AccessRights” vērtība. |
| ActivityGUID |  |  |  |  |
| ModifDate |  |  |  |  |
| Del |  |  |  |  |
| BloodTransfusion - Asins pārliešanas | | | | |
| NPatientTransfusions.  PatientTransfusionID |  |  |  | Ieraksta EntryID  Ja vērtība norādīta, tad operācija ir „Labot”, ja nav norādīta, tad operācija ir „Jauns”. |
| NPatientTransfusions.  NPatientID |  | Jā | X | Ieraksta NPatientID, turpmāk izmantots kā „@NPatientID” |
| NPatientTransfusions.  DateStart | DateStart | Jā | X | Pārliešanas sākums; DateStart = vērtība tiek sameklēta starp „Date” vērtībām, kur laukā „FieldName = ‘DateStart’”, un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| NPatientTransfusions.  DateFinish | DateFinish | Jā | X | Pārliešanas beigas; vērtība tiek sameklēta starp „Date” vērtībām, kur laukā „FieldName = ‘DateFinish’”, un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| NPatientTransfusions.  TransfusionVolume | TransfusionVolume | Jā | X | Pārliešanas apjoms, ml; vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘TransfusionVolume’”, un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| NPatientTransfusions.  BloodComponentCodeSystem | BloodComponentCodeSystem |  |  | Asins komponentes koda sistēma; Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘BloodComponentCodeSystem’”, un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| NPatientTransfusions.  BloodComponentCode | BloodComponentCode |  | X | Asins komponentes kods; Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘BloodComponentCode’”, un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| NPatientTransfusions.  BloodComponentDisplayName | BloodComponentDisplayName |  |  | Lietotājam draudzīgs asins komponentes nosaukums; Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘BloodComponentDisplayName’”, un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| NPatientTransfusions.  BloodGroupCodeSystem | BloodGroupCodeSystem |  |  | Asins grupas koda sistēma; Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘BloodGroupCodeSystem’”, un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| NPatientTransfusions.  BloodGroupCode | BloodGroupCode |  | X | Asins grupas kods; Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘BloodGroupCode’”, un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| NPatientTransfusions.  BloodGroupDisplayName | BloodGroupDisplayName |  |  | Lietotājam draudzīgs asins grupas nosaukums; Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘BloodGroupDisplayName’”, un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| NPatientTransfusions.  Complications | Complications |  |  | Asins pārliešanas laikā konstatētās blaknes; Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘Complications’”, un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| NPatientTransfusions.  AccessRights | AccessRights | Jā |  | Tiesības; tiek sameklēts tabulas „Documents” ieraksts, kas atbilst filtram „Documents.DocumentID = ieraksts.DocumentID”, no atrastā dokumenta ieraksta tiek izmantota lauka „AccessRights” vērtība. |
| NPatientTransfusions.  AcitivityGUID | ActivityGUID |  |  | Oriģinālā ziņojuma identifikators - message ID no IP |
| NPatientTransfusions.  ModifDate | ModifDate | Jā |  | Pēdējo izmaiņu datums |
| NPatientTransfusions.  Del | Del | Jā |  | Dzēšanas pazīme |
| Surgery - Ķirurģiskās iejaukšanās | | | | |
| NPatientSurgeries.  PatientSurgeryID |  |  |  | Ķirurģiskās iejaukšanās identifikators;  Ja vērtība norādīta, tad operācija ir „Labot”, ja nav norādīta, tad operācija ir „Jauns”. |
| NPatientSurgeries. NPatientID |  | Jā | X | Saite ar pacienta ierakstu; ieraksta NPatientID, turpmāk izmantots kā „@NPatientID” |
| NPatientSurgeries. DateFirst | Date | Jā |  | Pirmā konstatētā šāda veida ķirurģiskā iejaukšanās; Date = vērtība tiek sameklēta starp „Date” vērtībām kur laukā „FieldName = ‘Date” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| NPatientSurgeries. DateLast | Date | Jā |  | Pēdējā konstatētā šāda veida ķirurģiskā iejaukšanās. |
| NPatientSurgeries. SurgeryCodeSystem | SurgeryCodeSystem |  | X | Ķirurģiskās iejaukšanās kodu sistēma; Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘SurgeryCodeSystem’”, un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| NPatientSurgeries. SurgeryCode | SurgeryCode |  | X | Ķirurģiskās iejaukšanās kods; Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘SurgeryCode’”, un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| NPatientSurgeries. SurgeryDisplayName | SurgeryDisplayName |  |  | Ķirurģiskās iejaukšanās lietotājam draudzīgs nosaukums; Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘SurgeryDisplayName’”, un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| NPatientSurgeries. AccessRights | AccessRights | Jā |  | Tiesības; tiek sameklēts tabulas „Documents” ieraksts kas atbilst filtram „Documents.DocumentID = ieraksts.DocumentID”, no atrastā dokumenta ieraksta tiek izmantota lauka „AccessRights” vērtība. |
| NPatientSurgeries. ActivityGUID | ActivityGUID |  |  | Oriģinālā ziņojuma identifikators - message ID no IP |
| NPatientSurgeries. ModifDate | ModifDate | Jā |  | Pēdējo izmaiņu datums |
| NPatientSurgeries. Del | Del | Jā |  | Dzēšanas pazīme |
| AllDiagnosis – Visas diagnozes | | | | |
| PatientDiagnosisID |  |  |  | Diagnozes ieraksta identifikators.  Ja vērtība norādīta, tad operācija ir „Labot”, ja nav norādīta, tad operācija ir „Jauns”. |
| NPatientID |  |  | x | ieraksta NPatientID, turpmāk izmantots kā „@NPatientID” |
| Diagnosis CodeSystem | Diagnosis CodeSystem | Jā | x | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘DiagnosisCodeSystem” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| DiagnosisCode | DiagnosisCode | Jā | x | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘DiagnosisCode” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka, turpmāk izmantots kā „@DiagnosisCode”. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| Diagnosis DisplayValue | Diagnosis DisplayValue | Jā |  | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘DiagnosisDisplayValue” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| NPatientDiagnosisLinkToDocument.  DiagnosisType | DiagnosisType |  |  | Diagnozes veids.  Vērtība tiek sameklēta starp „Fields” vērtībām. Ja nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| Date | Date | Jā |  | Date = vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘Date” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| AccessRights |  | Jā |  | Tiek sameklēts tabulas „Documents” ieraksts, kas atbilst filtram „Documents.DocumentID = ieraksts.DocumentID”, no atrastā dokumenta ieraksta tiek izmantota lauka „AccessRights” vērtība. |
| ActivityGUID |  |  |  |  |
| ModifDate |  |  |  |  |
| Del |  |  |  |  |
| Disability – Invaliditāte | | | | |
| NPatientDisability.PatientDisabilityID |  | Jā | X | Ja vērtība norādīta, tad operācija ir „Labot”, ja nav norādīta, tad operācija ir „Jauns”. |
| NPatientDisability. InvGroupCode |  | Jā | X | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘InvGroupCode” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka, turpmāk izmantots kā „@InvGroupCode”. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| NPatientDisability.InvGroupDisplayValue |  |  | X | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘InvGroupDisplayValue” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| NPatientDisability.DateFrom | DateFrom | Jā |  | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘DateFrom” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| NPatientDisability.DateTo | DateTo |  |  | Vērtība tiek sameklēta starp „Fields” vērtībām, kur laukā „FieldName = ‘DateTo” un vērtība tiek paņemta no atbilstoša „FieldValue” lauka. Ja starp Fields vērtībām nebija izdevies atrast šo lauku, tad vērtība tiek atstāta ar nenoteiktu vērtību. |
| NPatientDisability. AccessRights | AccessRights | Jā |  | Tiesības: tiek sameklēts tabulas „Documents” ieraksts, kas atbilst filtram „Documents.DocumentID = ieraksts.DocumentID”, no atrastā dokumenta ieraksta tiek izmantota lauka „AccessRights” vērtība. |
| NPatientDisability.ActivityGUID | ActivityGUID |  |  | Oriģinālā ziņojuma identifikators - message ID no IP |
| NPatientDisability.ModifDate | ModifDate | Jā |  | Pēdējo izmaiņu datums |
| NPatientDisability.Del | Del | Jā |  | Dzēšanas pazīme |



66. attēls. internalTaskProcessing

1. Pārbauda obligātās vērtības saskaņā ar tabulu „Veselības pamatdatu objektu atbilstības tabula”. Ja uzdevumam nav pieejamas obligātās vērtības, process tiek pārtraukts kā neveiksmīgs.
2. Ja lauks TaskName <> NPatientDiagnosisTask, tad visām vērtībām, kam ir pievienots atribūts CodeSystem, tiek izpildīta šāda darbu secībā:
3. Tiek pārbaudīts, vai eksistē klasifikators ar OID kodu, kas atbilst atribūta CodeSystem saturam.
4. Ja šāds klasifikators atrasts, tiek pārbaudīts, vai tas satur atbilstošu elementa vērtību.
5. Ja vērtība ir atrasta klasifikatorā, tad importa process tiek turpināts, un ja vērtība nav atrasta, tad apstrādes process tiek pabeigts un vērtība netiek importēta sistēmā.
6. Ja lauks TaskName = NPatientDiagnosisTask, tad visām vērtībām, kam ir pievienots atribūts CodeSystem, tiek izpildīta šāda darbu secība:



67. attēls. Diagnozes noteikšanas algoritms

1. Tiek pārbaudīts, vai nepieciešams izmantot pacienta veselības pamatdatos automātiski iekļaujamo diagnožu klasifikatoru (OID = 1.3.6.1.4.1.38760.2.71), atlasot atbilstošo vērtību no sistēmas parametriem:
2. Ja šis klasifikators nav jāizmanto, tiek pārbaudīts, vai diagnoze satur paplašinājumu (DiagnosisCode vērtība satur punktu – ‘.’, piemēram, „J10.1”). Diagnoze ar paplašinājumu tiek saglabāta pacienta pamatdatos (iepriekš aprakstīta diagnozes paplašinājuma apstrādes funkcionalitāte tiek izņemta no sistēmas);
3. Ja automātiski iekļaujamo diagnožu klasifikators ir jāizmanto, tiek pārbaudīts, vai tas satur diagnozes vērtību:
   1. ja klasifikators satur diagnozi, no klasifikatora tiek atlasīts atribūts „Mēnešu skaits” (atribūta numurus: 362).
      1. ja klasifikators satur atribūtu „Mēnešu skaits”, tas tiek atlasīts un iegūtā vērtība tiek izmantota, veidojot ierakstu tabulā NPatientDiagnosis, aizpildot lauku DateActual, pievienojot dokumenta ieraksta laukam EffectiveTime norādīto mēnešu skaitu. Ja šādi iegūts skaitlis ir lielāks par iepriekšējo tabulas NPatientDiagnosis lauka DateActual vērtību, tad tā tiek ierakstīta datu bāzē, citādi tiek atstāta iepriekšēja lauka DateActual vērtība.
      2. ja klasifikators nesatur atribūtu „Mēnešu skaits”, tad tiek atlasīts sistēmas parametrs „Diagnožu iekļaušanas periods”. Iegūtā vērtība tiek interpretēta kā mēnešu skaits .
   2. ja klasifikators nesatur diagnozi, no sistēmas parametriem tiek atlasīts parametrs „Diagnožu iekļaušanas periods”. Iegūtā vērtībā tiek interpretēta kā mēnešu skaits. Diagnoze tiek saglabāta pacienta pamatdatos.
4. Pielasa atbilstošo ierakstu no datu bāzes, izmantojot objekta identificējošo datu kopu saskaņā ar tabulu „Veselības pamatdatu objektu atbilstības tabula”.
5. Ja ieraksts atrasts, tad:

* Izveido ierakstu saites tabulā LinkToDocument.
* Labo veselības pamatdatu ierakstu, ja mainījušās labojamo lauku vērtības. Labojamos laukus nosaka saskaņā ar tabulu „Veselības pamatdatu objektu atbilstības tabula”.
* Pārbauda NegationInd vērtību. Ja NegationInd = True (veselības pamatdatu ieraksts tiek anulēts), tad ierakstam uzstāda dzēšanas pazīmi Del=1.

1. Ja ieraksts nav atrasts, tad:

* Ja NegationInd vērtība ir True (veselības pamatdatu ieraksts anulēts), tad apstrādi pārtrauc, jo anulējamais ieraksts neeksistē.
* Ja NegationInd = False , tad:

1. Izveido ierakstu veselības pamatdatu tabulā.
2. Izveido ierakstu saites tabulā LinkToDocument.

Rezultāts

Funkcijas rezultāts ir pazīme par veiksmīgu/neveiksmīgu izpildi.

#### Pamatdatu atlases konfigurācija

Identifikācija

Sākotnēja pamatdatu atlases konfigurācija

Apraksts

Pamatdatu atlase sistēmas piegādes momentā ir sakonfigurētā šādā veidā: pēc noklusēšanas visām diagnozēm, alerģijām un brīdinājumiem kā datums tiek paņemts dokumenta lauks EffectiveTime. Dažiem dokumentu tipiem pamatdatu datums tiek iegūts no cita dokumenta lauka – tas tiek norādīts tabulā.

| Nr. | Dokuments | Sekcija | Entry |
| --- | --- | --- | --- |
| 1. | NULL | 1.3.6.1.4.1.38760.1.2.2.20.1 Klīniskā diagnoze | 1.3.6.1.4.1.38760.1.2.3.101.1 Laboratorijas izmeklējumu veicēja specialitāte |
|  | NULL | 1.3.6.1.4.1.38760.1.2.2.20.1 Klīniskā diagnoze | 1.3.6.1.4.1.38760.1.2.3.12.1 Diagnoze |
|  | NULL | 1.3.6.1.4.1.38760.1.2.2.16.1 Alerģiskās reakcijas | 1.3.6.1.4.1.38760.1.2.3.92.1 Alerģijas fakts |
|  | NULL | 1.3.6.1.4.1.38760.1.2.2.25.1 Izrakstītie medikamenti | 1.3.6.1.4.1.38760.1.2.3.118.1 Izrakstītā medikamenta informācija |
|  | NULL | 1.3.6.1.4.1.38760.1.2.2.8.1 Pacienta kustība | 1.3.6.1.4.1.38760.1.2.3.12.1 Diagnoze |
|  | NULL | 1.3.6.1.4.1.38760.1.2.2.8.1 Pacienta kustība | 1.3.6.1.4.1.38760.1.2.3.13.1 Patologanatomiskā diagnoze |
|  | NULL | 1.3.6.1.4.1.38760.1.2.2.49.1 Pacienta ķirurģiskās iejaukšanās | 1.3.6.1.4.1.38760.1.2.3.246.1 Pacienta ķirurģiskās iejaukšanās notikums |
|  | NULL | 1.3.6.1.4.1.38760.1.2.2.48.1 Pacienta asins pārliešanas sekcija | 1.3.6.1.4.1.38760.1.2.3.243.1 Pacienta asins pārliešanas notikums |
|  | 1.3.6.1.4.1.38760.1.2.1.1.1 Medicīnas pamatdatu kopsavilkuma dokuments | 1.3.6.1.4.1.38760.1.2.2.1.1 Pacienta pamatdatu alerģijas sekcija | 1.3.6.1.4.1.38760.1.2.3.1.1 Pacienta pamatdatu ieraksts alerģijai |
|  | 1.3.6.1.4.1.38760.1.2.1.1.1 Medicīnas pamatdatu kopsavilkuma dokuments | 1.3.6.1.4.1.38760.1.2.2.2.1 Pacienta pamatdatu diagnozes sekcija | 1.3.6.1.4.1.38760.1.2.3.2.1 Pacienta pamatdatu ieraksts diagnozei |
|  | 1.3.6.1.4.1.38760.1.2.1.1.1 Medicīnas pamatdatu kopsavilkuma dokuments | 1.3.6.1.4.1.38760.1.2.2.3.1 Pacienta pamatdatu medicīnas ierīces sekcija | 1.3.6.1.4.1.38760.1.2.3.3.1 Pacienta pamatdatu ieraksts medicīnas ierīcei |
|  | 1.3.6.1.4.1.38760.1.2.1.1.1 Medicīnas pamatdatu kopsavilkuma dokuments | 1.3.6.1.4.1.38760.1.2.2.4.1 Pacienta pamatdatu medikamenta sekcija | 1.3.6.1.4.1.38760.1.2.3.4.1 Pacienta pamatdatu ieraksts medikamentam |
|  | 1.3.6.1.4.1.38760.1.2.1.1.1 Medicīnas pamatdatu kopsavilkuma dokuments | 1.3.6.1.4.1.38760.1.2.2.5.1 Pacienta pamatdatu brīdinājuma sekcija | 1.3.6.1.4.1.38760.1.2.3.5.1 Pacienta pamatdatu ieraksts brīdinājumam |
|  | 1.3.6.1.4.1.38760.1.2.1.1.1 Medicīnas pamatdatu kopsavilkuma dokuments | 1.3.6.1.4.1.38760.1.2.2.1.1 Pacienta pamatdatu alerģijas sekcija | 1.3.6.1.4.1.38760.1.2.3.211.1 Pacienta pamatdatu neklasificētas alerģijas sekcija |
|  | 1.3.6.1.4.1.38760.1.2.1.1.1 Medicīnas pamatdatu kopsavilkuma dokuments | 1.3.6.1.4.1.38760.1.2.2.3.1 Pacienta pamatdatu medicīnas ierīces sekcija | 1.3.6.1.4.1.38760.1.2.3.212.1 Pacienta pamatdatu neklasificētas ieraksts medicīnas ierīcei |
|  | 1.3.6.1.4.1.38760.1.2.1.1.1 Medicīnas pamatdatu kopsavilkuma dokuments | 1.3.6.1.4.1.38760.1.2.2.65.1 Transfūzijas notikuma informācija | 1.3.6.1.4.1.38760.1.2.3.307.1 Transfūzijas notikums |
|  | 1.3.6.1.4.1.38760.1.2.1.1.1 Medicīnas pamatdatu kopsavilkuma dokuments | 1.3.6.1.4.1.38760.1.2.2.142.1 Pacienta invaliditātes sekcija | 1.3.6.1.4.1.38760.1.2.3.781.1 Pacienta invaliditātes grupa |
|  | NULL | 1.3.6.1.4.1.38760.1.2.2.1.1 Pacienta pamatdatu alerģijas sekcija | 1.3.6.1.4.1.38760.1.2.3.1.1 Pacienta pamatdatu ieraksts alerģijai |
|  | 1.3.6.1.4.1.38760.1.2.1.15.1 Ar noteiktām slimībām slimojošo pacientu uzskaites dati | 1.3.6.1.4.1.38760.1.2.2.2.1 Pacienta pamatdatu diagnozes sekcija | 1.3.6.1.4.1.38760.1.2.3.2.1 Pacienta pamatdatu ieraksts diagnozei |
|  | 1.3.6.1.4.1.38760.1.2.1.15.1 Ar noteiktām slimībām slimojošo pacientu uzskaites dati | 1.3.6.1.4.1.38760.1.2.2.4.1 Pacienta pamatdatu medikamenta sekcija | 1.3.6.1.4.1.38760.1.2.3.4.1 Pacienta pamatdatu ieraksts medikamentam |
|  | 1.3.6.1.4.1.38760.1.2.1.17.1 Izraksts no [stacionāra pacienta]/[amblatora pacienta] medicīniskās kartes | 1.3.6.1.4.1.38760.1.2.2.54.1 Slimības gaita, veselības stāvoklis | 1.3.6.1.4.1.38760.1.2.3.12.1 Diagnoze |
|  | 1.3.6.1.4.1.38760.1.2.1.2.1 No stacionāra izrakstīta (miruša) pacienta karte | 1.3.6.1.4.1.38760.1.2.2.8.1 Pacienta kustība | 1.3.6.1.4.1.38760.1.2.3.12.1 Diagnoze |
|  | 1.3.6.1.4.1.38760.1.2.1.2.1 No stacionāra izrakstīta (miruša) pacienta karte | 1.3.6.1.4.1.38760.1.2.2.1.1 Pacienta pamatdatu alerģijas sekcija | 1.3.6.1.4.1.38760.1.2.3.1.1 Pacienta pamatdatu ieraksts alerģijai |
|  | 1.3.6.1.4.1.38760.1.2.1.22.1 Transfūzijas protokols | 1.3.6.1.4.1.38760.1.2.2.65.1 Transfūzijas notikuma informācija | 1.3.6.1.4.1.38760.1.2.3.307.1 Transfūzijas notikums |
|  | 1.3.6.1.4.1.38760.1.2.1.24.1 E-recepte | 1.3.6.1.4.1.38760.1.2.2.88.1 Ārstniecības līdzeklis | 1.3.6.1.4.1.38760.1.2.3.535.1 Zāļu reģistrācijas numurs |
|  | 1.3.6.1.4.1.38760.1.2.1.31.1 Ginekologa profilaktiskā apskate organizētā vēža skrīningprogrammas ietvaros | 1.3.6.1.4.1.38760.1.2.2.95.1 Skrīningtestēšanas informācija | 1.3.6.1.4.1.38760.1.2.3.12.1 Diagnoze |
|  | 1.3.6.1.4.1.38760.1.2.1.33.1 Skrīningmamogrāfijas pakalpojuma rezultāts | 1.3.6.1.4.1.38760.1.2.2.99.1 Skrīningmamogrāfijas rezultāti | 1.3.6.1.4.1.38760.1.2.3.12.1 Diagnoze |
|  | 1.3.6.1.4.1.38760.1.2.1.35.1 Ambulatorā pakalpojuma rezultāts | 1.3.6.1.4.1.38760.1.2.2.103.1 Ambulatoro pakalpojumu dati | 1.3.6.1.4.1.38760.1.2.3.606.1 NCSP kods |
|  | 1.3.6.1.4.1.38760.1.2.1.35.1 Ambulatorā pakalpojuma rezultāts | 1.3.6.1.4.1.38760.1.2.2.53.1 Izraksta diagnozes | 1.3.6.1.4.1.38760.1.2.3.12.1 Diagnoze |
|  | 1.3.6.1.4.1.38760.1.2.1.45.1 Pacienta karte veselības aprūpei mājās | 1.3.6.1.4.1.38760.1.2.2.149.1 Pacienta veselības stāvokļa novērtējuma protokols | 1.3.6.1.4.1.38760.1.2.3.864.1 Uz ko ir alerģijas |
|  | 1.3.6.1.4.1.38760.1.2.1.52.1 Zobārstniecības pacienta ambulatorā karte - rezultāts | 1.3.6.1.4.1.38760.1.2.2.129.1 Ārstēšana | 1.3.6.1.4.1.38760.1.2.3.12.1 Diagnoze |
|  | 1.3.6.1.4.1.38760.1.2.1.53.1 Izraksts - epikrīze | 1.3.6.1.4.1.38760.1.2.2.167.1 Procedūras,operācijas,terapija | 1.3.6.1.4.1.38760.1.2.3.1070.1 Operācija |
| xxx | 1.3.6.1.4.1.38760.1.2.1.25.1 Onkoloģiskā pacienta reģistrācijas karte | 1.3.6.1.4.1.38760.1.2.2.72.1 Onkoloģiskās diagnozes fakts | 1.3.6.1.4.1.38760.1.2.3.393.1 Onkoloģiskā diagnoze |
| xxx | 1.3.6.1.4.1.38760.1.2.1.26.1 Cukura diabēta pacienta karte | 1.3.6.1.4.1.38760.1.2.2.74.1 Cukura diabēta diagnozes fakti | 1.3.6.1.4.1.38760.1.2.3.644.1 Diabēta diagnoze |
| xxx | 1.3.6.1.4.1.38760.1.2.1.27.1 Karte narkoloģiskajam pacientam un personai, kura lieto atkarību izraisošās vielas | 1.3.6.1.4.1.38760.1.2.2.23.1 Atkarības - ārstniecības detaļas | 1.3.6.1.4.1.38760.1.2.3.646.1 Narkoloģijas un atkarību diagnoze  1.3.6.1.4.1.38760.1.2.3.450.1 Diagnozes veids  Dokumenta datums iegūts no:  *1.3.6.1.4.1.38760.1.2.3.129.1 Ārstēšanas epizodes uzsākšanas datums (vērtība no lauka effectiveTime)* |
| xxx | 1.3.6.1.4.1.38760.1.2.1.28.1 Traumu, ievainojumu un saindēšanās gadījumu uzskaites karte | 1.3.6.1.4.1.38760.1.2.2.81.1 Traumas, ievainojumu un saindēšanās diagnozes dati | 1.3.6.1.4.1.38760.1.2.3.645.1 Traumas, ievainojumu un saindēšanās diagnoze  *\* IDB diagnoze neatrodas tieši sekcijā, bet gan papildus organizerā ar OID 1.3.6.1.4.1.38760.1.2.3.476.1 Ievainojuma diagnozes datu komplekts* |
| xxx | 1.3.6.1.4.1.38760.1.2.1.28.1 Traumu, ievainojumu un saindēšanās gadījumu uzskaites karte | 1.3.6.1.4.1.38760.1.2.2.81.1 Traumas, ievainojumu un saindēšanās diagnozes dati | 1.3.6.1.4.1.38760.1.2.3.521.1 Traumas, ievainojumu un saindēšanās blakusdiagnoze  Atlasītās diagnozes tiek reģistrētas kā blakusdiagnozes pamatdatu ierakstos. |
| xxx | 1.3.6.1.4.1.38760.1.2.1.55.1 Onkoloģiskā pacienta ārstēšanas reģistrācijas karte | 1.3.6.1.4.1.38760.1.2.2.175.1 Onkoloģiska ārstniecība | 1.3.6.1.4.1.38760.1.2.3.1360.1 Onkoloģiskā diagnoze |
| xxx | 1.3.6.1.4.1.38760.1.2.1.56.1 Pacienta karte pacientiem ar psihiskiem un uzvedības traucējumiem | 1.3.6.1.4.1.38760.1.2.2.178.1 Pacienta ar psihiskiem un uzvedības traucējumiem diagnozes | 1.3.6.1.4.1.38760.1.2.3.1129.1 Psihisko traucējumu pamatdiagnoze |
| xxx | 1.3.6.1.4.1.38760.1.2.1.56.1 Pacienta karte pacientiem ar psihiskiem un uzvedības traucējumiem | 1.3.6.1.4.1.38760.1.2.2.178.1 Pacienta ar psihiskiem un uzvedības traucējumiem diagnozes | 1.3.6.1.4.1.38760.1.2.3.1130.1 Psihisko traucējumu blakusdiagnozes  \* Diagnozes veids - Blakusdiagnoze |
| xxx | 1.3.6.1.4.1.38760.1.2.1.56.1 Pacienta karte pacientiem ar psihiskiem un uzvedības traucējumiem | 1.3.6.1.4.1.38760.1.2.2.178.1 Pacienta ar psihiskiem un uzvedības traucējumiem diagnozes | 1.3.6.1.4.1.38760.1.2.3.1131.1 Citas blakusdiagnozes  \* Diagnozes veids - Blakusdiagnoze |
| xxx | 1.3.6.1.4.1.38760.1.2.1.56.1 Pacienta karte pacientiem ar psihiskiem un uzvedības traucējumiem | 1.3.6.1.4.1.38760.1.2.2.178.1 Pacienta ar psihiskiem un uzvedības traucējumiem diagnozes | 1.3.6.1.4.1.38760.1.2.3.1132.1 Sociālās blakusdiagnozes  \* Diagnozes veids - Blakusdiagnoze |
| xxx | 1.3.6.1.4.1.38760.1.2.1.57.1 Tuberkulozes pacienta karte | 1.3.6.1.4.1.38760.1.2.2.181.1 Diagnoze tuberkulozes pacienta kartei | 1.3.6.1.4.1.38760.1.2.3.1152.1 Diagnozes kods 1  Dokumenta datums tiek iegūts no entry  *1.3.6.1.4.1.38760.1.2.3.1150.1 Datums, kad pacients iekļauts reģistrā (vērtība effectiveTime)* |
| xxx | 1.3.6.1.4.1.38760.1.2.1.57.1 Tuberkulozes pacienta karte | 1.3.6.1.4.1.38760.1.2.2.181.1 Diagnoze tuberkulozes pacienta kartei | 1.3.6.1.4.1.38760.1.2.3.1153.1 Diagnozes kods 2  \* Diagnozes veids - Blakusdiagnoze |
| xxx | 1.3.6.1.4.1.38760.1.2.1.57.1 Tuberkulozes pacienta karte | 1.3.6.1.4.1.38760.1.2.2.183.1 Atkārtota ārstēšana tuberkulozes pacienta kartei | 1.3.6.1.4.1.38760.1.2.3.1152.1 Diagnozes kods 1  \* Diagnozes veids - Blakusdiagnoze |
| xxx | 1.3.6.1.4.1.38760.1.2.1.58.1 Arodslimību pacienta reģistrācijas karte | 1.3.6.1.4.1.38760.1.2.2.185.1 Diagnoze arodslimības pacienta kartei | 1.3.6.1.4.1.38760.1.2.3.1187.1 Arodslimību diagnozes  Datums diagnozei tiek iegūts no:  *1.3.6.1.4.1.38760.1.2.3.1184.1 Datums, kad arodslimība noteikta pirmreizēji* |
| xxx | 1.3.6.1.4.1.38760.1.2.1.58.1 Arodslimību pacienta reģistrācijas karte | 1.3.6.1.4.1.38760.1.2.2.185.1 Diagnoze arodslimības pacienta kartei | 1.3.6.1.4.1.38760.1.2.3.1207.1 Nāves cēlonis |
| xxx | 1.3.6.1.4.1.38760.1.2.1.58.1 Arodslimību pacienta reģistrācijas karte | 1.3.6.1.4.1.38760.1.2.2.187.1 Arodslimību un to izraisītājfaktoru klasifikācija atbilstoši arodslimību izmeklēšanu un uzskaiti regulējošajiem normatīvajiem aktiem | 1.3.6.1.4.1.38760.1.2.3.1195.1 Arodslimība |
| xxx | 1.3.6.1.4.1.38760.1.2.1.58.1 Arodslimību pacienta reģistrācijas karte | 1.3.6.1.4.1.38760.1.2.2.187.1 Arodslimību un to izraisītājfaktoru klasifikācija atbilstoši arodslimību izmeklēšanu un uzskaiti regulējošajiem normatīvajiem aktiem | 1.3.6.1.4.1.38760.1.2.3.1196.1 Konkrētā arodslimības izraisītājfaktora noteicošās arodslimības diagnoze |
| xxx | 1.3.6.1.4.1.38760.1.2.1.59.1 Arodslimību pacienta talons | 1.3.6.1.4.1.38760.1.2.2.188.1 Diagnoze arodslimības pacienta talonam | 1.3.6.1.4.1.38760.1.2.3.1210.1 Pārskata gadā pārciestās slimības diagnoze  *\* Diagnoze netrodas tieši sekcijā, bet ir saistīta caur organizeru: 1.3.6.1.4.1.38760.1.2.3.1356.1 Pārciestās slimības atskaites gadā* |
| xxx | 1.3.6.1.4.1.38760.1.2.1.59.1 Arodslimību pacienta talons | 1.3.6.1.4.1.38760.1.2.2.188.1 Diagnoze arodslimības pacienta talonam | 1.3.6.1.4.1.38760.1.2.3.1216.1 Hronisko slimību diagnozes līdz kaitīgā darba sākumam |
| xxx | 1.3.6.1.4.1.38760.1.2.1.59.1 Arodslimību pacienta talons | 1.3.6.1.4.1.38760.1.2.2.189.1 Invaliditāte arodslimības pacienta talonā | 1.3.6.1.4.1.38760.1.2.3.1203.1 Arodslimību diagnozes |
| xxx | 1.3.6.1.4.1.38760.1.2.1.59.1 Arodslimību pacienta talons | 1.3.6.1.4.1.38760.1.2.2.189.1 Invaliditāte arodslimības pacienta talonā | 1.3.6.1.4.1.38760.1.2.3.1204.1 Vispārējās slimības diagnoze |
| xxx | 1.3.6.1.4.1.38760.1.2.1.59.1 Arodslimību pacienta talons | 1.3.6.1.4.1.38760.1.2.2.187.1 Arodslimību un to izraisītājfaktoru klasifikācija atbilstoši arodslimību izmeklēšanu un uzskaiti regulējošajiem normatīvajiem aktiem | 1.3.6.1.4.1.38760.1.2.3.1195.1 Arodslimība |
| xxx | 1.3.6.1.4.1.38760.1.2.1.59.1 Arodslimību pacienta talons | 1.3.6.1.4.1.38760.1.2.2.187.1 Arodslimību un to izraisītājfaktoru klasifikācija atbilstoši arodslimību izmeklēšanu un uzskaiti regulējošajiem normatīvajiem aktiem | 1.3.6.1.4.1.38760.1.2.3.1196.1 Konkrētā arodslimības izraisītājfaktora noteicošās arodslimības diagnoze |
| xxx | 1.3.6.1.4.1.38760.1.2.1.60.1 Multiplās sklerozes pacienta karte | 1.3.6.1.4.1.38760.1.2.2.193.1 Diagnoze pacienta kartē | 1.3.6.1.4.1.38760.1.2.3.1256.1 Diagnoze |
| xxx | 1.3.6.1.4.1.38760.1.2.1.61.1 Reģistrācijas karte pacientam ar iedzimtām anomālijām | 1.3.6.1.4.1.38760.1.2.2.200.1 Diagnoze bērnam | 1.3.6.1.4.1.38760.1.2.3.1286.1 Diagnožu saraksts  *\* Diagnoze netrodas tieši sekcijā, bet ir saistīta caur organizeru: 1.3.6.1.4.1.38760.1.2.3.1286.1 Diagnožu saraksts* |

### externalTaskProcessing (ārējo uzdevumu apstrāde)

Identifikācija

Funkcija: ExternalTaskProcessing

Ieejas parametri

ProcessableTasks ( 6.2.12.7 documentEntryProcessing (dokumenta ierakstu apstrāde)), kur TaskType = „External”

Apraksts

Tiek veikta katra saraksta „ProcessableTasks” ieraksta apstrāde. Ieraksta dati tiek sūtīti uz Vakcinācijas reģistra Web servisa metodi „ProcessTask”, kā ieejas parametru nododot apstrādājamo uzdevumu. Rezultātā tiek sagaidīta pazīme par veiksmīgu ieraksta apstrādi vai kļūdas ziņojums gadījumā, ja ieraksta apstrāde nav veiksmīga.

Rezultāts

Funkcijas rezultāts ir pazīme par veiksmīgu/neveiksmīgu izpildi.

### generatePDF (Ģenerēt PDF)

Identifikācija

Funkcija: generatePDF

Apraksts

Funkcija transformē HTML formāta dokumentu PDF formātā.

Ieejas Parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | HTML | nvarchar | 1..1 | Transformējamais dokuments. |

Apstrādes algoritms

1. No sistēmas konfigurācijas tiek iegūts transformācijas direktorijas nosaukums.

Transformācijas direktorijā tiek izveidots jauns html fails, kas satur transformējamo dokumentu.

Tiek izpildīta transformācija, izsaucot šādu skriptu:

* wkhtmltopdf\wkhtmltopdf.exe <avota fails>.html <rezultāta fails>.pdf
* Izveidots PDF fails tiek nolasīts no diska un atgriezts izsaucošai funkcijai.

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | PDF | varbinary | 1..1 | Transformētais dokuments. |

Alternatīvie scenāriji

Neizdevās uzģenerēt PDF; Tiek atgriezta kļūda ACK.

### setPersonMajority (Uzstādīt personas pilngadības pazīmi)

Identifikācija

Funkcija: setPersonMajority

Tiesības: sistēmas funkcija (tiesības netiek pārbaudītas)

Apraksts

Funkcija atlasa personas, kuru vecums pārsniedz 18 gadus, un kuru pilngadības pazīme = False un maina pazīmi uz True.

Ieejas parametri

Nav ieejas parametru.

Apstrādes algoritms

Tiek izpildīta šāda darbību secība:

1. Tiek atlasīts ierakstu saraksts no tabulām Patients un NPatients, kuriem izpildās nosacījumi:
2. [Šā brīža datums] - Patients.BirthTime >18 gadi
3. NPatients.Majority = 0
4. NPatients.PersonStatus = „Dzīvs”
5. NPatients.Status = [Aktuāls]
6. Atlasītajiem ierakstiem maina pilngadības pazīmi:
7. NPatients.Majority = 1
8. Patients.MajorityDate = [šī brīža datums]
9. Patients.MajoorityReason = neaizpilda
10. Tiek pārbaudīts, vai personai ir vecāku (ar tipu PARENT) delegāciju ieraksti; Ja tādi tiek atrasti, vecāku delegācijām tiek uzstādīts beigu termiņš vienāds ar pašreizējo datumu, tādējādi pārtraucot personas datu delegāciju.

### getNotificationConfiguration (Iegūt paziņojumu konfigurāciju)

Identifikācija

Funkcija: getNotificationConfiguration

Tiesības:

EvkRghtMessageConfigurationGet – tiesības saņemt automātisko paziņojumu konfigurāciju.

Apraksts

Tiek atlasīta un atgriezta pieprasītā pacienta automātisko paziņojumu konfigurācija.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | subject | LVEX\_MT000004UV01. NotificationConfiguration | 0..1 | Pieprasījums pacienta paziņojumu konfigurācijai. |

Apstrādes algoritms

1. Tiek pārbaudīts, vai lietotājam ir tiesības izpildīt darbību, izsaucot funkciju 6.5.5 checkUserRight (Pārbaudīt personas tiesību). Ja tiesību nav, tiek atgriezta kļūda, un izveidots audita ieraksts.
2. Tiek identificēta pacienta karte, analizējot informāciju ieejas struktūras laukā id.
3. Darbības ietvaros tiek pārbaudīts, vai pieprasītai pacienta kartei atļauts piekļūt, izsaucot funkciju 6.5.4 checkUserRightsAccessObject (Pārbaudīt personas tiesību līmeni uz objektu).
4. Ja pacienta karte nav atrasta, vai lietotājam tā nav pieejama, tiek atgriezta kļūda.
5. Tiek izpildīts pieprasījums, atlasot pacienta automātisko paziņojumu konfigurācijas informāciju un aizpildot atbildes datu struktūru (skat. 6.1.14 NotificationConfiguration (Automātisko paziņojumu uzstādījumi)) .

Alternatīvie scenāriji

Nav tiesību veikt darbību - tiek atgriezts kļūdas paziņojums EVK\_0029 (Nav tiesību veikt darbību) un izveidots audita ieraksts.

Pacienta karte nav atrasta vai nav pieejama - tiek atgriezts kļūdas paziņojums EVK\_0001 (Nav atrasta vai nav pieejama pacienta karte) un izveidots audita ieraksts.

### setNotificationConfiguration (Uzstādīt paziņojumu konfigurāciju)

Identifikācija

Funkcija: setNotificationConfiguration

Tiesības:

EvkRghtNotificationConfigurationSet – tiesības uzstādīt pacienta ziņojumu konfigurāciju.

Apraksts

Tiek saņemta no lietotāja un saglabāta EVK datu bāzē pacienta uzstādīta ziņojumu konfigurācija.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | subject | LVEX\_MT000004UV01.NotificationConfiguration | 0..1 | Pieprasījums lietotāja ziņojumu konfigurācijai. |

Apstrādes algoritms

1. Tiek pārbaudīts, vai pieprasītai pacienta kartei atļauts piekļūt, izsaucot funkciju 6.5.4 checkUserRightsAccessObject (Pārbaudīt personas tiesību līmeni uz objektu).
2. Tiek pārbaudīts, vai lietotājam ir tiesības izsaukt šo funkciju (skat. 6.5.5 checkUserRight (Pārbaudīt personas tiesību) – EvkRghtMessageConfigurationSet). Ja tiesību nav, tiek atgriezts kļūdas paziņojums EVK\_0029 (Nav tiesību veikt darbību), un izveidots audita ieraksts.
3. Tiek izpildīts lietotāja pieprasījums, saglabājot pacienta paziņojumu konfigurācijas informāciju (skat. 6.1.14 NotificationConfiguration (Automātisko paziņojumu uzstādījumi)), saskaņā ar uzstādītiem parametriem un UpdateMode atribūtiem (kas norāda, kura no operācijām tiek veikta ar ierakstu).
4. Lietotājam tiek atgriezta aktualizētā paziņojumu konfigurācijas informācija vai kļūdas ziņojums gadījumā, ja notika neparedzēta kļūda.

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | subject | LVEX\_MT000004UV01.NotificationConfiguration | 0..1 | Ziņojumu konfigurācija. |

Alternatīvie scenāriji

Nav tiesību veikt darbību – tiek atgriezts kļūdas paziņojums EVK\_0029 (Nav tiesību veikt darbību) un izveidots audita ieraksts.

Pacienta karte nav atrasta vai nav pieejama –tiek atgriezts kļūdas paziņojums EVK\_0001 (Nav atrasta vai nav pieejama pacienta karte) un izveidots audita ieraksts.

### getNextDocumentNumber (Iegūt jauna dokumenta numuru)

Identifikācija

Funkcija: getNextDocumentNumber

Tiesības: EvkRghtNextDocumentNumberGet

Apraksts

Funkcija veido jauno dokumenta numuru, pārbaudot, vai kešatmiņā atrodas dokumentu numuri dotai organizācijai, un ja ir, tad tiek atgriezts viens no tiem. Ja tādu numuru nav, tad tiek iegūts jauns dokumentu numuru komplekts, tie ievietoti kešatmiņā un tiek atgriezts pirmais no šiem numuriem.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | Subject | LVEX\_MT000005UV01.DocNumber | 1..1 | Dokumenta numura pieprasījuma/rezultāta datu struktūra. |

Apstrādes algoritms

1. Tiek pārbaudīts, vai lietotājam ir tiesības izpildīt darbību, izsaucot funkciju 6.5.5 checkUserRight (Pārbaudīt personas tiesību). Ja tiesību nav, tiek atgriezts kļūdas paziņojums un izveidots audita ieraksts.
2. Tiek pārbaudīts, vai lietotājam ir tiesības piekļūt pieprasītai organizācijas informācijai, pārbaudot lietotāja talona lauku „VIAuthorityCode” un salīdzinot to ar ieejas parametru struktūras lauku „organizationId”. Ja lietotāja talona saturs un pieprasītas organizācijas identifikators nesakrīt (lietotājs pieder citai organizācijai), tiek atgriezts kļūdas paziņojums EVK\_0029 (Nav tiesību veikt darbību) un tiek veidots audita ieraksts.
3. Ja lietotājam ir tiesības izsaukt funkciju, tad tiek pārbaudīts, vai kešatmiņā ir piereģistrēti dokumentu numuri norādītai organizācijai. Ja dokumenta numurs un dokumentu numuru apgabala beigas ir piereģistrētas, un dokumentu numuru apgabala beigas ir lielākas pār piereģistrēto numuru, tad dokumenta numurs ir palielināts ar 1, un tiek atgriezts lietotājam.
4. Ja kešatmiņā nav atrasti dokumentu numuri pieprasītai organizācijai, tad tiek iegūts dokumenta numurs no tabulas „DocumentNumberSequence” – tiek atlasīts nākamā dokumenta numurs un numuru veidošanas solis. Tabulas „DocumentNumberSequence” dokumenta numuru vērtība palielināta par numuru veidošanas soļa vienībām.
5. Ja dokumenta numurs netiek atrasts tabulā „DocumentNumberSequence”, tad tiek atlasīts dokumenta numurs pēc noklusēšanas no sistēmas parametra „DefaultDocumentNumber” un saglabāts tabulā „DocumentNumberSequence”.
6. Atlasītais dokumenta numurs (DocNr) un numuru intervāla beigās (vienāds ar DokNr + veidošanas solis) tiek saglabāts kešatmiņā.
7. Dokumenta numurs tiek formatēts atbilstoši numura formātam: AAAAA.NN, kur:

* AAAAA - Ārstniecības iestādes kods;
* NN – numurs.

1. Iegūtais dokumenta numurs tiek saglabāts LVEX\_MT000005UV01.DocNumber datu struktūrā

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | subject | LVEX\_MT000005UV01.DocNumber | 1..1 | Dokumenta numura pieprasījuma/rezultāta datu struktūra. |

Alternatīvie scenāriji

Nav tiesību veikt darbību – tiek atgriezts kļūdas paziņojums EVK\_0029 (Nav tiesību veikt darbību) un izveidots audita ieraksts.

Lietotāja ārstniecības iestāde atšķiras no pieprasītās ārstniecības iestādes – tiek atgriezts kļūdas paziņojums EVK\_0029 (Nav tiesību veikt darbību) un izveidots audita ieraksts.

### getEVKEpacParameters (Iegūt EVK e-pakalpojumu parametrus)

Identifikācija

Funkcija: getEVKEpacParameters

Tiesības: EvkRghtEpacParametersGet

Apraksts

Funkcija nodrošina e-pakalpojumiem vajadzīgo parametru iegūšanu. Latvija.lv lietotājs izsauc šo funkciju, norādot vēlamās parametru grupas nosaukumu, un saņem atpakaļ no tabulas Parameters iegūto informāciju.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | Subject | LVEX\_MT000006UV01.EpacParameterGroup | 0..1 | Parametru grupas pieprasījums (jābūt aizpildītam tikai grupas nosaukumam) |

Apstrādes algoritms

1. Tiek pārbaudīts, vai lietotājam ir tiesības izpildīt darbību, izsaucot funkciju 6.5.5 checkUserRight (Pārbaudīt personas tiesību). Ja tiesību nav, tiek atgriezts kļūdas paziņojums, un izveidots audita ieraksts.
2. Tiek pārbaudīts, vai ieejas parametra lauks GroupName aizpildīts ar vienu no atļautām grupām.
3. No tabulas parameters tiek atlasīta atbilstoša informācija – parametru grupa ar visiem parametriem, nosaukumiem un vērtībām, un tiek aizpildīta izejas parametru struktūra.
4. Izejas parametru struktūra tiek atgriezta izsaucējam.

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | Subject | LVEX\_MT000006UV01.EpacParameterGroup | 0..1 | Parametru grupas rezultāts |

Alternatīvie scenāriji

Nav tiesību veikt darbību – tiek atgriezts kļūdas paziņojums EVK\_0029 (Nav tiesību veikt darbību) un izveidots audita ieraksts.

### getClassifierValues (Iegūt EVK saglabāto klasifikatoru kopiju vērtības)

Identifikācija

Funkcija: getClassifierValues

Tiesības: EvkRghtClassifiers

Apraksts

Funkcija nodrošina e-pakalpojumiem vajadzīgo klasifikatoru vērtību iegūšanu. Latvija.lv lietotājs izsauc šo funkciju, norādot pieprasāmo klasifikatoru identifikatorus (OID), un saņem atpakaļ no atbilstošajām klasifikatoru tabulām iegūto informāciju.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | Subject | LVEX\_MT000008UV01.EVKClassifierList | 0..1 | Klasifikatoru vērtību pieprasījums |

Apstrādes algoritms

1. Tiek pārbaudīts, vai lietotājam ir tiesības izpildīt darbību, izsaucot funkciju 6.5.5 checkUserRight (Pārbaudīt personas tiesību). Ja tiesību nav, tiek atgriezts kļūdas paziņojums, un izveidots audita ieraksts.
2. Tiek pārbaudīts, vai ieejas parametra lauks ClassifierID aizpildīts ar eksistējošu klasifikatoru OID tabulā CodeSystems.
3. No tabulas CodeSystems tiek atlasīta pieprasīto klasifikatoru tabulas nosaukumi
4. Secīgi tiek atlasītas klasifikatoru vērtības (kods, attēlojamā vērtība, versija) no atbilstošajām klasifikatoru tabulām un tiek aizpildīta izejas parametru struktūra.
5. Izejas parametru struktūra tiek atgriezta izsaucējam.

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | Subject | LVEX\_MT000008UV01.EVKClassifierList | 0..1 | Klasifikatora vērtību rezultāts |

Alternatīvie scenāriji

Nav tiesību veikt darbību – tiek atgriezts kļūdas paziņojums EVK\_0029 (Nav tiesību veikt darbību) un izveidots audita ieraksts.

### getProfileSettings (Iegūt personas profila uzstādījumus.)

Identifikācija

Funkcija: getProfileSettings

Tiesības: EvkRghtProfileSettings – Profilu uzstādījumu atlase

Apraksts

Funkcija nodrošina pacienta profila atgriešanu izsaucošai sistēmai. Pacienta profila uzstādījumi netiek interpretēti EVK sistēmā (tiek uzskatīti par nemaināmu teksta informāciju un saglabāti bez apstrādes).

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | Subject | LVEX\_MT000002UV01.ParameterList | 0..1 | Profila pieprasījuma parametru komplekts |

Apstrādes algoritms

1. Tiek pārbaudīts, vai pieprasītai pacienta kartei atļauts piekļūt, izsaucot funkciju 6.5.4 checkUserRightsAccessObject (Pārbaudīt personas tiesību līmeni uz objektu).
2. Tiek pārbaudīts, vai lietotājam ir tiesības izsaukt šo funkciju (skat. 6.5.5 checkUserRight (Pārbaudīt personas tiesību) - EvkRghtProfileSettings). Ja tiesību nav, tiek atgriezts kļūdas paziņojums EVK\_0029 (Nav tiesību veikt darbību), un izveidots audita ieraksts.
3. Tiek izpildīts lietotāja pieprasījums, atlasot pacienta profila informāciju no tabulas PatientProfileSettings (filtrējot to pēc laukiem PatientGUID, Name, System).
4. Ja iepriekšējā solī tiek atlasītas profila vērtības, ar tiem tiek aizpildīta struktūra LVEX\_MT000001UV01.ProfileRecord un tā tiek atgriezta funkcijas izsaucējām.

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | Subject | LVEX\_MT000001UV01.ProfileRecord | 0..1 | Profila parametru datu struktūra |

Alternatīvie scenāriji

Pacienta karte nav atrasta vai nav pieejama – tiek atgriezts kļūdas paziņojums EVK\_0001 (Nav atrasta vai nav pieejama pacienta karte) un izveidots audita ieraksts.

Nav tiesību veikt darbību – tiek atgriezts kļūdas paziņojums EVK\_0029 (Nav tiesību veikt darbību) un izveidots audita ieraksts.

### setProfileSettings (Pievienot un mainīt personas profila uzstādījumus.)

Identifikācija

Funkcija: setProfileSettings

Tiesības: EvkRghtProfileSettings – Profilu uzstādījumu uzstādīšana

Apraksts

Funkcija nodrošina pacienta profila uzstādījumu saglabāšanu datu bāzē. Profila uzstādījumi netiek apstrādāti EVK un tiek saglabāti tieši tādā veidā, kā saņemti.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | Subject | LVEX\_MT000001UV01.ProfileRecord | 0..1 | Uzstādāma profila parametru datu struktūra |

Apstrādes algoritms

1. Tiek pārbaudīts, vai pieprasītai pacienta kartei atļauts piekļūt, izsaucot funkciju 6.5.4 checkUserRightsAccessObject (Pārbaudīt personas tiesību līmeni uz objektu).
2. Tiek pārbaudīts, vai lietotājam ir tiesības izsaukt šo funkciju (skat. 6.5.5 checkUserRight (Pārbaudīt personas tiesību) – EvkRghtProfileSettings). Ja tiesību nav, tiek atgriezts kļūdas paziņojums EVK\_0029 (Nav tiesību veikt darbību), un izveidots audita ieraksts.
3. Tiek izpildīts lietotāja pieprasījums, saglabājot pacienta profila informāciju tabulā PatientProfileSettings:
   1. Atbilstoši pieprasījumā norādītiem laukiem Id – tiek atrasts pacients un meklēšanai izmantots lauks PatientGUID
   2. Pieprasījumā lauka Code vērtība tiek izmantota saglabājot tabulas laukā Name
   3. Pieprasījuma lauks System tiek izmantots saglabājot to vērtību tabulas laukā system
4. Izsaucējām tiek atgriezta tehniskā atbildes struktūra atbilstoši funkcijas izpildes rezultātiem.

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | Subject | MCCI\_MT000200UV01.Acknowledgement | 0..1 | Tehniskā atbildes struktūra |

Alternatīvie scenāriji

Nav tiesību veikt darbību – tiek atgriezts kļūdas paziņojums EVK\_0029 (Nav tiesību veikt darbību) un izveidots audita ieraksts.

### GetConsultationsList (Iegūt konsultāciju ziņojumu sarakstu)

Identifikācija

Funkcija: GetConsultationsList

Tiesības: EvkRghtConsultationsListGet

Apraksts

Funkcija nodrošina pacienta vai ārstniecības personas konsultāciju saraksta atlasi un atgriešanu izsaucējam. Procedūra tiek izmantota gan pilna konsultāciju saraksta atlasei, gan individuālas konsultācijas iegūšanai. Ir iespējami procedūras izsaukuma varianti, ar vai bez konsultāciju pielikumu atlasīšanu.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | GetConsultationsList | PRPA\_MT000500UV01\_LV01.ConsultationsListQuery | 0..1 | Konsultācijas ziņojumu pieprasījuma struktūra. Skat. 6.1.22 |

Apstrādes algoritms

Izpildot konsultāciju saraksta iegūšanas pieprasījumu, tiek izpildīta šāda darbību secība:

1. Tiek pārbaudītas lietotāja tiesības izsaukt funkciju (skat. 6.5.5 checkUserRight (Pārbaudīt personas tiesību)). Ārstniecības personām tiek atlasītas tikai tās konsultācijas, kurās konkrētā ārstniecības persona ir kā ziņojuma iniciators vai saņēmējs.
2. No tabulām „ConsultationRecord”, „ConsultationMessage” un „MessageAttachment” tiek atlasīta informācija, kas atbilst meklēšanas parametriem. Gadījumā, ja izsaukuma parametros nav uzstādīta ziņojumu pielikumu iegūšana, tabulas „MessageAttachment” failu satura dati netiek atlasīti un netiek pievienoti rezultāta datu struktūrai. Gadījumā, ja izsaukuma parametros nav uzstādīta ziņojumu tekstu iegūšana, ziņojumi nav atgriezti.
3. Tiek sagatavota rezultāta datu struktūra, kas aizpildīta ar meklēšanas rezultātā atrasto konsultācijas ziņojumu datiem.
4. Pirms atgriezt rezultātu, tiek izveidots žurnāla ieraksts par veikto darbību.

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | ConsultationRecord | PRPA\_MT000500UV01\_LV01.ConsultationRecord | 0..\* | Konsultāciju ziņojumu saraksts. Skat. 6.1.21 |

Alternatīvie scenāriji

Nav tiesību veikt darbību - tiek atgriezts kļūdas paziņojums EVK\_0029 (Nav tiesību veikt darbību) un izveidots audita ieraksts.

### SetConsultation (Izveidot konsultācijas ziņojumu)

Identifikācija

Funkcija: SetConsultation

Tiesības: EvkRghtConsultationSet

Apraksts

Nodrošina iespēju izveidot vai saglabāt pēc rediģēšanas konsultācijas ziņojuma ierakstu.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | ConsultationRecord | PRPA\_MT000500UV01\_LV01.ConsultationRecord | 0..\* | Konsultāciju ziņojumu saraksts. Skat. 6.1.21 |

Apstrādes algoritms

Izpildot konsultācijas ziņojuma saglabāšanu, tiek izpildīta šāda darbību secība:

1. Tiek pārbaudītas lietotāja tiesības izsaukt funkciju (skat. 6.5.5 checkUserRight (Pārbaudīt personas tiesību)). Ārstniecības personām atļauts rediģēt tikai tās konsultācijas, kurās konkrētā ārstniecības persona ir kā ziņojuma iniciators vai saņēmējs.
2. Atkarībā no ConsultationRecord datu struktūrā iekļautā atribūta „UpdateMode”, tiek izpildīta jauna ieraksta pievienošana vai eksistējoša ieraksta rediģēšana.
3. Pirms saglabāt konsultācijas ierakstu datu bāzē, tiek veiktas šādas validācijas:

* Pacienta lietotājam jābūt pieejai pacienta kartei, kurai tiek saglabāta konsultācija;
* Ārstniecības personai ir pieejamas tikai tās konsultācijas, kurās viņš ir konsultācijas adresāts vai konsultācijas veidotājs (sourceId root = 1.3.6.1.4.1.38760.2.1 un extension = ārstniecības personas kodu);
* Ja ārstniecības persona ir konsultācijas adresāts, tad ārstniecības persona var veidot tikai atbildes ziņojumus un nevar rediģēt ienākošos ziņojumus;
* Ja ārstniecības persona ir konsultācijas veidotājs, tad ārstniecības persona var veidot tikai ienākošos ziņojumus un nevar rediģēt atbildes ziņojumus;
* Pacienta lietotājs var veidot tikai ienākošos ziņojumus un nevar rediģēt atbildes ziņojumus;
* Ja konsultācija ir statusā „Pabeigts” (Complete), tad tās labošana nav iespējama;
* Laukā documentId (root = „1.3.6.1.4.1.38760.3.4.5.1”) ierakstītajam dokumenta numuram ir jāatbilst eksistējošam dokumentam, un dokumenta autors ir konsultācijā iekļautā ārstniecības persona. Tikai ārstniecības persona var labot šī lauka saturu.

1. Tabulā ConsultationRecord tiek saglabāts jauns konsultācijas ieraksts, kas atbilst saņemtajai ConsultationRecord datu struktūrai.
2. Tiek sagatavota rezultāta datu struktūra, kas aizpildīta ar saglabātā ieraksta datiem.
3. Pirms atgriezt rezultātu, tiek izveidots žurnāla ieraksts par veikto darbību.

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | ConsultationRecord | PRPA\_MT000500UV01\_LV01.ConsultationRecord | 0..\* | Konsultāciju ziņojumu saraksts. Skat. 6.1.21 |

Alternatīvie scenāriji

Nav tiesību veikt darbību – tiek atgriezts kļūdas paziņojums EVK\_0029 (Nav tiesību veikt darbību) un izveidots audita ieraksts.

Kļūdas datu validācija – tiek atgriezts kļūdas paziņojums EVK\_0049 (Parametri norādīti nekorekti).

Nav atrasts atjaunojamais konsultāciju ieraksts – tiek atgriezts kļūdas paziņojums EVK\_0056 (Norādītais objekts neeksistē).

Nav atrasts pacients, kam tiek veidota konsultācija – tiek atgriezts kļūdas paziņojums EVK\_0001 Nav atrasta vai nav pieejama pacienta karte.

### GetConsultationSearchSuggestions (Iegūt konsultāciju meklēšanai ieteiktās Ārstniecības personas)

Identifikācija

Funkcija: GetConsultationSearchSuggestions

Tiesības: EvkRghtConsultationSearchSuggestionsGet

Apraksts

Atgriezt pacientam sarakstu ar iespējamo ārstniecības personu sarakstu, pie kuriem pacientam jau bija konsultācijas.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | GetConsultationSearchSuggestions | PRPA\_MT000500UV01\_LV01.ConsultationsSuggestionsQuery | 0..1 | Pacienta identifikators, skat. 6.1.23 |

Apstrādes algoritms

Izpildot konsultāciju ārstniecības personu meklēšanas pieprasījumu, tiek izpildīta šāda darbību secība:

1. Tiek pārbaudītas lietotāja tiesības izsaukt funkciju (skat. 6.5.5 checkUserRight (Pārbaudīt personas tiesību)).
2. No tabulām „ConsultationRecord” tiek atlasītas visas konkrētajam pacientam vai ārstniecības personai reģistrētās konsultācijas, atlasot visas unikālās (distinct) ārstniecības personas (PractitionerCode) un ārstniecības organizācijas (CustodianCode), kurās tās reģistrētas.
3. Tiek sagatavota rezultāta datu struktūra, kas aizpildīta ar iepriekšējā solī atlasītajiem datiem.
4. Pirms atgriezt rezultātu, tiek izveidots žurnāla ieraksts par veikto darbību.

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | ConsultationsSuggestions | PRPA\_MT000500UV01\_LV01.ConsultationsSuggestionsResponse | 0..1 | Konsultāciju veikušās ārstniecības personas, skat. 6.1.24 |

Alternatīvie scenāriji

Nav tiesību veikt darbību – tiek atgriezts kļūdas paziņojums EVK\_0029 (Nav tiesību veikt darbību) un izveidots audita ieraksts.

Kļūdas datu validācija – tiek atgriezts kļūdas paziņojums EVK\_0049 (Parametri norādīti nekorekti)

Nav atrasts vai nav pieejams pacienta ieraksts – tiek atgriezts kļūdas paziņojums EVK\_0001 Nav atrasta vai nav pieejama pacienta karte

### GetConsultationPractitioners (Atlasīt pacienta ārstniecības personu kontaktu sarakstu)

Identifikācija

Funkcija: GetConsultationPractitioners

Tiesības: EvkRghtConsultationPractitionersGet

Apraksts

Atlasa no pacienta profila konsultācijām izvēlētās ārstniecības personas.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | Subject | PRPA\_MT000500UV01\_LV01.ConsultationsPractitionersQuery | 0..1 | Pacienta konsultāciju kontaktsaraksta pieprasījums. Skat. 6.1.25 |

Apstrādes algoritms

1. Tiek pārbaudītas lietotāja tiesības izsaukt funkciju (skat. 6.5.5 checkUserRight (Pārbaudīt personas tiesību)).
2. Tiek pārbaudīts, vai pieprasītajai pacienta kartei atļauts piekļūt, izsaucot funkciju 6.5.4 checkUserRightsAccessObject (Pārbaudīt personas tiesību līmeni uz objektu).
3. Tiek izpildīts lietotāja pieprasījums, atlasot pacienta profila informāciju no tabulas ConsultationPractitioners (filtrējot to pēc lauka Code, atkarībā no lauka Type – Type satur pacienta vai ārstniecības personas identifikatora OID un code – ārstniecības personas identifikatoru vai pacienta NPatientID).
4. Ja iepriekšējā solī tiek atlasītas kontaktsaraksta vērtības, ar tām tiek aizpildīta struktūra PRPA\_MT000500UV01\_LV01.ConsultationsPractitionersResponse un tā tiek atgriezta funkcijas izsaucējam.

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | Subject | PRPA\_MT000500UV01\_LV01.ConsultationsPractitionersResponse | 0..1 | Kontaktu saraksta datu struktūra. Skat. 6.1.26 |

Alternatīvie scenāriji

Nav tiesību veikt darbību – tiek atgriezts kļūdas paziņojums EVK\_0029 (Nav tiesību veikt darbību) un izveidots audita ieraksts.

Kļūdas datu validācija – tiek atgriezts kļūdas paziņojums EVK\_0049 (Parametri norādīti nekorekti)

Nav atrasta ārstniecības persona – tiek atgriezts kļūdas paziņojums EVK\_0056 (Norādītais objekts neeksistē)

Nav atrasts vai nav pieejams pacienta ieraksts – tiek atgriezts kļūdas paziņojums EVK\_0001 Nav atrasta vai nav pieejama pacienta karte

### SetConsultationPractitioners (Izveidot pacienta ārstniecības personu kontaktu sarakstu)

Identifikācija

Funkcija: SetConsultationPractitioners

Tiesības: EvkRghtConsultationPractitionersSet

Apraksts

Aizvieto pacienta profila konsultācijām izvēlētās ārstniecības personas ar nosūtīto sarakstu.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | Subject | PRPA\_MT000500UV01\_LV01.ConsultationsPractitionersResponse | 0..1 | Kontaktu saraksta datu struktūra (skat. 6.1.26) |

Apstrādes algoritms

1. Tiek pārbaudītas lietotāja tiesības izsaukt funkciju (skat. 6.5.5 checkUserRight (Pārbaudīt personas tiesību)).
2. Tiek pārbaudīts, vai pieprasītajai pacienta kartei atļauts piekļūt, izsaucot funkciju 6.5.4 checkUserRightsAccessObject (Pārbaudīt personas tiesību līmeni uz objektu).
3. Tiek pārbaudīta saņemtā datu struktūra, veicot šādas validācijas:

* Atribūtam ar norādīto *id* vērtību jāeksistē datu bāzē, un jāpieder norādītajam pacientam vai ārstniecības personai (pacientam vai ĀP ar pieprasījumā norādīto identifikāciju);
* Tiek pārbaudīts, vai pacienta ģimenes ārsta kods ir iekļauts sarakstā. Ja nav iekļauts, tad tāds kods tiek pievienots.
* Visiem ārstiem, kas iekļauti sarakstā, tiek veikta pārbaude, vai tiem profilā atzīmēts, ka viņi sniedz konsultācijas. Tie ārsti, kas nesniedz konsultācijas, netiek iekļauti saglabājamajā sarakstā;
* Visiem ārstiem, kas iekļauti sarakstā, tiek pārbaudīts, vai viņi sniedz konsultācijas tikai saviem pacientiem, un ja jā, un pacientam ir cits ģimenes ārsts, ārstniecības persona tiks izņemta no saraksta

1. Tiek izpildīts lietotāja pieprasījums, saglabājot pacienta profila informāciju tabulā ConsultationPractitioners

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | Subject | PRPA\_MT000500UV01\_LV01.ConsultationsPractitionersResponse | 0..1 | Kontaktu saraksta datu struktūra (skat. 6.1.26) |

Alternatīvie scenāriji

Nav tiesību veikt darbību – tiek atgriezts kļūdas paziņojums EVK\_0029 (Nav tiesību veikt darbību) un izveidots audita ieraksts.

Kļūdas datu validācija – tiek atgriezts kļūdas paziņojums EVK\_0049 (Parametri norādīti nekorekti)

Nav pieejamas ārstniecības personas – tiek atgriezts kļūdu paziņojums EVK\_0056 (Norādītais objekts neeksistē)

Nav atrasts vai nav pieejams pacienta ieraksts – tiek atgriezts kļūdas paziņojums EVK\_0001 Nav atrasta vai nav pieejama pacienta karte

### SetDocumentUnavailable (Padarīt dokumentu nepieejamu)

Identifikācija

Funkcija: SetDocumentUnavailable

Tiesības: EvkRghtDocumentUnavailableSet

EvkRghtDocumentStatusSet

Apraksts

Funkcija nodrošina origināla dokumenta veidotājorganizācijai iespējas padarīt dokumentu nepieejamu, gadījumā, ja dokuments ir kļūdains.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | Subject | RCMR\_MT000002UV02\_LV01.ClinicalDocumentUnavailableRequest | 1..1 | Dokumenta nepieejamības pieprasījuma datu struktūra (skat. 6.1.27) |

Apstrādes algoritms

1. Tiek pārbaudītas lietotāja tiesības izsaukt funkciju (skat. 6.5.5 checkUserRight (Pārbaudīt personas tiesību)).
2. Tiek pārbaudīts, vai pacienta karte eksistē. Gadījumā, ja nē, tad tiek atgriezta kļūda EVK\_0001 Nav atrasta vai nav pieejama pacienta karte un procedūra tiks pārtraukta.
3. Tiek pārbaudīts, vai ieejas parametri atbilst tipiem, vai II elementu root OID atbilst pieļaujamiem. Gadījumā, ja parametri netika validēti, tiek atgriezta kļūda EVK\_0049 (Parametri norādīti nekorekti) un procedūras izpilde ir pārtraukta.
4. Pārbaudīt pieprasītā dokumenta numuru, salīdzinot pirmo numura daļu ar lietotāja organizācijas numuru (nameidentifier talona lauka organizācijas daļai jāsakrīt ar dokumenta numura pirmo daļu). Gadījumā, ja numurs nesakrīt ar nameidentifier daļu, tiek atgriezta kļūda (EVK\_0009 Nav tiesības modificēt dokumenta piekļuves tiesības) un procedūras izpilde ir pārtraukta.
5. Tiek pārbaudīts, vai dokuments ekistē norādītajam pacientam (dokuments tiek atlasīts, izmantojot dokumenta id, activityGUID un pacienta identifikācijas pieprasījuma objekta laukus). Gadījumā, ja dokuments nebūs atrasts, tad tiek atgriezta kļūda (EVK\_0008 Dokuments nav atrasts vai nav pieejams) un procedūras izpilde ir pārtraukta.
6. Ja dokuments eksistē, tad tiek pārbaudīts, vai dokumenta stāvoklis nav “Processing”. Gadījumā, ja dokuments ir stāvoklī “Processing”, tiek atgriezta kļūda (EVK\_0083 Nav iespējams turpināt darbību, dokuments vēl ir apstrādē) un procedūras izpilde ir pārtraukta.
7. Ja dokuments eksistē, tad tiek pārbaudīts, vai dokumenta stāvoklis ir “Cancelled” vai “Incorrect”. Gadījumā, ja dokuments ir stāvoklī “Actual”, tad dokumentam tiek uzstādīts stāvoklis “Cancelled”, izmantojot funkciju SetDocumentStatus (skat. 6.2.3).
8. Dokumenta vēstures ieraksts tiek papildināts ar ierakstu ModifAction=“Dokumenta nepieejāmības uzstāīšana” un laukā Details=unavailabilityReason
9. Dokumenta ieraksta lauks confidentialityReason tiek aizpildīts ar datiem no pieprasījuma struktūras lauka unavailabilityReason
10. Dokumentam tiek uzstādīta lauka AccessRights vertība 0

Izejas parametri

nav

Alternatīvie scenāriji

Nav tiesību veikt darbību – tiek atgriezts kļūdas paziņojums EVK\_0029 (Nav tiesību veikt darbību) un izveidots audita ieraksts.

Pacienta karte neeksistē – tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte).

Kļūdas datu validācija – tiek atgriezts kļūdas paziņojums EVK\_0049 (Parametri norādīti nekorekti).

Ja organizācijas numurs nesakrīt ar norādītā dokumenta identifikatora pirmo daļu, tiek atgriezts kļūdas paziņojums EVK\_0009 Nav tiesības modificēt dokumenta piekļuves tiesības.

Ja dokumentu nevar atrast, izmantojot norādīto id un activityGUID, tad tiek atgriezts kļūdas paziņojums EVK\_0008 Dokuments nav atrasts vai nav pieejams,

Ja dokuments ir statusā “Processing”, tad tiek atgriezts kļūdas paziņojums “EVK\_0083 Nav iespējams turpināt darbību, dokuments vēl ir apstrādē”.

### GetDocumentStatus (Iegūt dokumenta statusu)

Identifikācija

Funkcija: GetDocumentStatus

Tiesības: EvkRghtDocumentsGet

Apraksts

Funkcija nodrošina dokumenta statusa un kļūdu saraksta atgriešanu (kļūdu saraksts tiek atgriezts, ja dokuments ir statusā Incorrect).

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | Subject | RCMR\_MT000002UV02\_LV01.ClinicalDocumentStatusRequest | 1..1 | Dokumenta statusa pieprasījuma datu struktūra (skat. 6.1.28) |

Apstrādes algoritms

1. Tiek pārbaudītas lietotāja tiesības izsaukt funkciju (skat. 6.5.5 checkUserRight (Pārbaudīt personas tiesību)).
2. Tiek pārbaudīts, vai pacienta karte eksistē. Gadījumā, ja nē, tad tiek atgriezta kļūda EVK\_0001 Nav atrasta vai nav pieejama pacienta karte un procedūra tiks pārtraukta.
3. Tiek pārbaudīts, vai ieejas parametri atbilst tipiem, vai II elementu root OID atbilst pieļaujamiem. Gadījumā, ja parametri netika validēti, tiek atgriezta kļūda EVK\_0049 (Parametri norādīti nekorekti) un procedūras izpilde ir pārtraukta.
4. Tiek pārbaudīts, vai dokuments ekistē norādītajam pacientam (dokuments tiek atlasīts, izmantojot dokumenta id, activityGUID un pacienta identifikācijas pieprasījuma objekta laukus). Gadījumā, ja dokuments nebūs atrasts, tad tiek atgriezta kļūda (EVK\_0008 Dokuments nav atrasts vai nav pieejams) un procedūras izpilde ir pārtraukta.
5. Ja dokuments eksistē, tad tiek pārbaudīts, vai dokumenta stāvoklis nav “Incorrect”. Gadījumā, ja dokuments ir stāvoklī “Incorrect”, no tabulas DocumentErrors tiek atlasītas dokumenta validācijas kļūdas un papildināts rezultātu komplekts.
6. Dokumenta stāvokļa datu struktūra tiek atgriezta izsaucējam.

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | Subject | RCMR\_MT000002UV02\_LV01.ClinicalDocumentStatusResponse | 0..1 | Dokumenta stāvokļa datu struktūra (skat. 6.1.29) |

Alternatīvie scenāriji

Nav tiesību veikt darbību – tiek atgriezts kļūdas paziņojums EVK\_0029 (Nav tiesību veikt darbību) un izveidots audita ieraksts.

Pacienta karte neeksistē – tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte).

Kļūdas datu validācija – tiek atgriezts kļūdas paziņojums EVK\_0049 (Parametri norādīti nekorekti).

Ja dokumentu nevar atrast, izmantojot norādīto id un activityGUID, tad tiek atgriezts kļūdas paziņojums EVK\_0008 Dokuments nav atrasts vai nav pieejam.

## Validācijas moduļa projektējums

### validateDocumentModel (Validēt dokumentu pret modeli)

Identifikācija

Funkcija: validateDocumentModel

Ieejas parametri

CDA dokuments

Apraksts

Funkcija nodrošina CDA dokumenta validāciju pret modeli, izmantojot CDA validatora komponenti.

Funkcija no saņemtā XML dokumenta veido klases instanci un pārbauda dokumenta atbilstību MDHT rīkā izveidotajam CDA dokumenta modelim. Ja dokuments neatbilst modelim, tiek atgriezts kļūdu saraksts. Kļūdas atkarīgas no CDA modelī definētajiem nosacījumiem.

Izsaucošā funkcija to tālāk saglabā pie dokumenta kļūdu sarakstā, vienlaicīgi nomainot dokumenta statusu.

Rezultāts

Funkcijas rezultāts ir validācijas procesā iegūtais validācijas kļūdu saraksts.

Ja funkcijas darbības rezultātā tiek iegūta neapstrādāta kļūda, tad validācijas funkcijas darbība tiek pārtraukta, un šī kļūda tiek nodota izsaucošajai funkcijai (Exception).

### validateDocument (Validēt dokumentu)

Identifikācija

Funkcija: validateDocument

Tiesības : EvkRghtDocumentValidate

Apraksts

Funkcija izsauc dokumenta pārbaudes procesu un atgriež izsaucējam darbības rezultātu (dokumenta atbilstību modelim, dokumentā izmantoto klasificēto vērtību atbilstību klasifikatoriem ). Funkcija izpildās sinhronā režīmā.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | ClinicalDocument | RCMR\_MT000002UV02\_LV01. ClinicalDocument | 0..1 | Pārbaudāmais dokuments |

Apstrādes algoritms

1. Pārbauda, vai lietotājam ir tiesības izsaukt šo funkciju, izmantojot EVK funkciju checkUserRights.
2. Tiek pārbaudīta dokumenta struktūra un pamatelementi atbilstoši EVK funkcijai (addDocument (Pievienot dokumentu – sinhronā daļa) ). Pēc pārbaudes dokuments netiek saglabāts EVK datu bāzē.
3. Pēc pamata pārbaudēm, ja dokuments ir pareizs, tad tas tiek pārbaudīts pret modeli un klasifikatoriem atbilstoši šādam algoritmam:
4. Validē CDA dokumentu pret modeli – atbilstoši funkcijai (validateDocumentModel (Validēt dokumentu pret modeli)).

* Tiek atgriezts saraksts ar validācijas kļūdām. Validācijas kļūdas tiek reģistrētas procesa mainīgajā „DocumentErrors”.
* Ja validācijas kļūdas nav, tad saraksts ir tukšs, un tiek uzskatīts, ka CDA dokuments pilnībā atbilst modelim.
* Process turpinās ar nākamo soli.

1. Pārbauda CDA dokumenta klasificētās vērtības – atbilstoši funkcijai extractAndValidateClassifiedValues (Validēt dokumenta klasificētās vērtības).

* Tiek atgriezts saraksts ar klasificēto vērtību kļūdām. Tās tiek reģistrētas procesa mainīgajā „DocumentErrors”.
* Ja klasificētajās vērtībās kļūdas nav, tad saraksts ir tukšs, un tiek uzskatīts, ka CDA dokumentā iekļautās klasificētās vērtības atbilst klasifikatoriem.
* Process turpinās ar nākamo soli.

1. Ja „DocumentErrors” sarakstā nav reģistrētas kļūdas un dokuments ir viens no PREDA dokumentu tipiem, tad:

* Dokuments tiek pārbaudīts pret PREDA validācijas procedūrām.
* Gadījumā, ja PREDA lietotne atgrieza kļūdas, tās tiek pievienotas pie „DocumentErrors” saraksta.

1. Ja „DocumentErrors” sarakstā reģistrēts vismaz viens ieraksts, tad:

* „DocumentErrors” sarakstā reģistrētās kļūdas saglabā funkcijas izejas parametra kļūdas sadaļā.
* Izejas parametra statusa laukam tiek uzstādīta vērtība „AE” – Ziņojums ir kļūdains
* Dokumenta apstrāde tiek beigta.

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | ClinicalDocument | RCMR\_MT000002UV02\_LV01. ClinicalDocument | 0..1 | Pārbaudāmā dokumenta EVK pārbaužu rezultāts. |

Alternatīvie scenāriji

Notiek neparedzēta kļūda; Kļūdas ziņojums tiek atgriezts izsaucošajai funkcijai.

## Dokumentu metasistēmas moduļa projektējums

### getDocumentTemplateList (Iegūt dokumentu veidņu sarakstu)

Identifikācija

Tikla pakalpe: getDocTemplateList

Tiesības: EvkRghtDocumentTemplateGet

Apraksts

Funkcija atgriež dokumentu veidņu sarakstu pēc pieprasījuma parametriem.

Ieejas Parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | HL7 RCMR\_MT000003UV.QueryByParameter | HL7 RCMR\_MT000003UV.QueryByParameter | 1..1 | Dokumentu veidņu atlases parametri (6.1.9 TemplateQueryByParameter) |

Apstrādes algoritms

Izpildot dokumenta veidņu meklēšanas pieprasījumu, tiek izpildīta šāda darbību secība:

1. Tiek pārbaudītas lietotāja tiesības izsaukt funkciju (skat. 6.5.5 checkUserRight (Pārbaudīt personas tiesību)).
2. Tiek pārbaudīts, vai queryId parametra padotie OID un pieprasījuma nosaukums atbilst datu struktūras definīcijā minētajiem OID un pieprasījuma nosaukumam.
3. No tabulas „DocumentTemplates” tiek atlasīta informācija, kas atbilst meklēšanas parametriem
4. Tiek sagatavots rezultāta dokuments, kas tiek aizpildīts ar meklēšanas rezultātā atrasto dokumentu veidņu datiem.
5. Pirms atgriezt rezultātu tiek izveidots audita ieraksts par veikto darbību.

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | HL7 RCMR\_MT000103UV01\_LV01.TemplateDocument | HL7 RCMR\_MT000103UV01\_LV01.TemplateDocument | 1..1 | Dokumenta veidne (6.1.8 TemplateDocument) |

### getDocTemplate (Iegūt dokumenta veidni)

Identifikācija

Funkcija: getDocTemplate

Tiesības: EvkRghtDocumentTemplateGet

Apraksts

Funkcijas ieejas parametri tiek saņemti, izmantojot HL7 standarta datu tipu: RCMR\_MT000003UV01.QueryByParameter (Medical Record Parameter Query message).

Tīkla pakalpes izpildīšanai tiek izveidots datu tipa paplašinājums RCMR\_MT000003UV01\_LV01 (skat. 6.1.8 TemplateDocument).

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | HL7 RCMR\_MT000003UV.QueryByParameter | HL7 RCMR\_MT000003UV.QueryByParameter | 1..1 | Dokumentu veidņu atlases parametri (6.1.9 TemplateQueryByParameter)  Obligāti jāaizpilda:  queryId  clinicalDocument.id |

Apstrādes algoritms

Izpildot dokumenta šablona meklēšanas pieprasījumu, tiek izpildīta šāda darbību secība:

1. Tiek pārbaudītas lietotāja tiesības izsaukt funkciju (skat. 6.5.5 checkUserRight (Pārbaudīt personas tiesību)).
2. Tiek pārbaudīts, vai queryId parametra padotie OID un pieprasījuma nosaukums atbilst tīkla pakalpes definīcijā minētajiem OID un pieprasījuma nosaukumam.
3. Tiek pārbaudīts, vai ir aizpildīts pieprasījuma struktūras lauks clinicalDocument.id. Ja lauks nav aizpildīts, tad funkcijas izpilde beidzas ar kļūdu.
4. No tabulas „DocumentTemplates” tiek atlasīti dati: TypeId = id.root atribūts un TemplateOID = id.extension atribūts.

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | HL7 RCMR\_MT000103UV01\_LV01.TemplateDocument | HL7 RCMR\_MT000103UV01\_LV01.TemplateDocument | 1..1 | Dokumenta veidne (6.1.8 TemplateDocument) |

### setDocTemplate (Pievienot un papildināt dokumenta veidni)

Identifikācija

Funkcija: setDocTemplate

Tiesības: EvkRghtDocumentTemplateSet

Apraksts

Funkcijas ieejas parametri tiek saņemti, izmantojot datu tipu: RCMR\_MT000103UV01\_LV01.TemplateDocument (6.1.8 TemplateDocument).

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | RCMR\_MT000103UV01\_LV01.TemplateDocument | RCMR\_MT000103UV01\_LV01.TemplateDocument | 1..1 | Dokumenta veidnes dati (6.1.8 TemplateDocument) |

Apstrādes algoritms

1. Tiek pārbaudītas lietotāja tiesības izsaukt funkciju (skat. 6.5.5 checkUserRight (Pārbaudīt personas tiesību)). Ja tiesību nav, tad funkcija tiek pārtraukta, tiek atgriezta kļūda un izveidots audita ieraksts.
2. Tiek pārbaudīts, vai ieejas datu struktūras lauks statusCode ir aizpildīts ar statusu ACTUAL (atbilstoši klasifikatoram Dokumentu veidņu statusi).
3. Gadījumā, ja UpdateMode uzstādīts kods A, tad tiek pievienots jauns ieraksts tabulā DocumentTemplates.
4. Ja ir uzstādīts UpdateMode kods AR vai D, tabulā DocumentTemplates tiek atrasts dokumenta veidnes ieraksts ar identifikatoru, kas atbilst pieprasījumā laukam id. Dokumentu veidnei tiek veidota jauna versija, saglabājot identifikācijas numuru un palielinot lauka [version] vērtību par 1.
5. Tiek pārbaudīts, vai aizpildīts lauks ValidFrom un ValidTo un to vērtības nav pretrunīgas (ValidTo > ValidFrom, vai ValidTo nav aizpildīts).
6. Tiek pārbaudīts, vai tika aizpildīta autora datu struktūra.
7. Tiek pārbaudīts, vai aizpildīti lauki Validator, Schema, Model, Description. Ja viens no laukiem nav aizpildīts, funkcijas izpilde beidzas ar kļūdu.
8. Tiek pārbaudīts, vai sistēmā ir piereģistrēta dokumenta validatora klase. Ja validatora klase netiek atrasta, tiek atgriezta kļūda un izveidots audita ieraksts.
9. Tiek aizpildīti datu bāzes tabulas DocumentTemplates lauki, atbilstoši veidnes statusam.
10. Ja dokumenta veidne tika rediģēta, tad iepriekšējai veidnes versijai tiek uzstādīts statuss (DocumentTemplates.statuss) ‘Anulēts’ un derīguma termiņš (DocumentTemplates.ValidTo), kas vienāds ar rediģēšanas momentu (atbilstoši, vienlaicīgi nevar būt divas aktīvas viena dokumenta veidņu versijas).

Alternatīvie scenāriji

Nav atrasta dokumenta veidne un lauks UpdateMode ir AR. Tiek atgriezts kļūdas paziņojums (EVK\_0032 Dokumenta veidne nav atrasta).

Ir atrasts dokumenta šablons ar ieejas parametros norādīto Id, bet ieejas parametros lauks statusCode ir A. Tiek atgriezts kļūdas paziņojums (EVK\_0033 Dokumenta veidne jau izveidota).

## Autorizācijas moduļa projektējums

### checkRightsSetObjectAccessRights (Pārbauda tiesības uzstādīt aizliegumu)

Identifikācija

Funkcija: checkRightsSetObjectAccessRights

Tiesības: sistēmas funkcija (tiesības netiek pārbaudītas)

Apraksts

Funkcija pārbauda, vai lietotājam ir tiesības uzstādīt objektam jaunas pieejas tiesības, un ja ir, tad tiek akceptēts jauns tiesību komplekts, bet ja nē, tad tiek atgriezta kļūda.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | RightDescriptor | Varchar(3) | 1..1 | Oriģināls objekta (CDA dokuments, pamatdatu elements) tiesību apraksts, kas nosaka, kurām lietotāju kategorijām ir tiesības piekļūt objektam |
| 2 | NewRightDescriptor | Varchar(3) | 1..1 | Uzstādāms objekta (CDA dokuments, pamatdatu elements) tiesību apraksts, kas nosaka, kurām lietotāju kategorijām ir tiesības piekļūt objektam |
| 3 | UserRightDescriptor | Varchar(6) | 1..1 | Lietotāja tiesību apraksts, kas norāda, kādas operācijas lietotājam ir tiesības veikt ar objektiem |

Apstrādes algoritms

Tiek izpildīta šāda darbu secība:

1. Tiek pārbaudīta tiesību aprakstu atbilstība tiesību aprakstu formātiem (tiesību aprakstu formātus skat. 6.5.4 checkUserRightsAccessObject (Pārbaudīt personas tiesību līmeni uz objektu) ). Ja viens no tiesību aprakstiem neatbilst tiesību apraksta formātam, procedūra beidzās ar kļūdas paziņojumu (EVK\_0018 Nav tiesības uzstādīt aizliegumu), un tiek izveidots audita ieraksts (skat. 6.10.2 writeJournal (Žurnalificēšanas pieraksta veidošana)).

Tiek izrēķināts, kādi tiesību apraksta elementi tiek mainīti tiesību uzstādīšanas procesā rezultātā. Tas tiek noskaidrots, izrēķinot XOR operāciju ar sākotnējo (RightDescriptor) un jauno (NewRightDescriptor) objekta tiesību aprakstiem. Piemēram, sākotnējais objekta tiesību apraksts bija 110 un jauns ir 100; XOR operācijas rezultāts ir 010.

Tiek pārbaudīts, vai lietotājam ir tiesības veikt tādu tiesību maiņu, kura tiek pieprasīta. Lai to noskaidrotu, tiek salīdzināta 2. solī izrēķinātā tiesību atšķirība ar lietotāja objektu tiesību uzstādīšanas atļaujām.

Piemēram, lietotājam ir nodefinētas tiesību maiņas privilēģijas (UserRightDescriptor pēdējās trīs zīmes) 011. Tiek salīdzinātas lietotāja privilēģijas un pieprasīto tiesību maiņa (izpildot AND operāciju). Ja rezultāts sakrīt ar pieprasīto tiesību maiņu, tad procedūra atgriež „true”, citādi tiek izveidots audita ieraksts (skat. 6.10.2 writeJournal (Žurnalificēšanas pieraksta veidošana)), un procedūra atgriež „false”

Datu piemērs:

| Nr. | Right Descriptor | NewRight Descriptor | UserRightDescriptor | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | 111 | 110 | 100 011 | Pieprasīta maiņa: (111 XOR 110) = 001  Lietotāja tiesību pārbaude: (001 AND 011) = 001  Funkcija atgriež „true” |
| 2 | 101 | 001 | 100 011 | Pieprasīta maiņa: (101 XOR 001) = 100  Lietotāja tiesību pārbaude: (100 AND 011) = 000  Funkcija atgriež „false” |
| 3 | 110 | 100 | 100 000 | Pieprasīta maiņa: (110 XOR 100) = 010  Lietotāja tiesību pārbaude: (010 AND 000) = 000  Funkcija atgriež „false” |
| 4 | 100 | 111 | 001 110 | Pieprasīta maiņa: (100 XOR 111) = 011  Lietotāja tiesību pārbaude: (011 AND 110) = 010  Funkcija atgriež „false” |

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | CheckResult | boolean | 1..1 | Vai lietotājam ir atļauts mainīt objekta tiesības |

Alternatīvie scenāriji

Objekta tiesību apraksts neatbilst formātam. Tiek atgriezts kļūdas paziņojums (EVK\_0019 Objekta tiesību apraksts neatbilst formātam) un izveidots audita ieraksts.

### checkDetailedRole (Noteikt detalizēto lomu)

Funkcija: checkDetailedRole

Apraksts

Noteikt lietotāja detalizēto lomu attiecībā uz konkrētu pacientu. Gan lietotājs, gan pacients var nebūt reģistrēts EVK sistēmā, bet lietotājam jābūt identificējamam ar personas koda identifikāciju.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | | UserIdentificationCode | nvarchar(50) | 1..1 | Lietotāja identifikācijas kods (personas kods) |
| 2 | | PatientIdentificationType | nvarchar(50) | 1..1 | Pacienta identifikācijas tips (LV personas koda OID) |
| 3 | | PatientIdentificationCode | nvarchar(50) | 1..1 | Pacienta identifikācijas kods (personas kods) |

Apstrādes algoritms

Nosaka PatientRole

1. Pēc PatientIdentificationType un Code pielasa datus no EVK datu bāzes.

* Ja pacienta ieraksts atrasts, tad nolasa vērtības:
* Dzimšanas datums –Patients.BirthTime
* Pilngadības pazīme – NPatients.Majority
* Ja pacienta ieraksts netika atrasts EVK datu bāzē (nav izveidota pacienta karte), tad pēc PatientIdentificationCode pielasa datus no PMLP. Nosaka:
* Dzimšanas datums
* Pilngadības pazīme – aprēķina no Dzimšanas datuma vecumu. Ja vecums>= 18, tad pazīme – True.

Ja PMLP neatrod šādu PK, tad tiek atgriezts kļūdas paziņojums (EVK\_0039 Personas kods nav korekts).

1. Aprēķina pacienta vecumu pilnos gados.
2. Nosaka pacienta lomu PatientRole:

* Ja vecums ir < 14, tad PatientRole = P1
* Ja 14 <= vecums un pilngadības pazīme nav True, tad PatientRole = P2.
* Ja pilngadības pazīme = True, tad PatientRole = P3.

1. Ja PatientDelegation tabulā ir ieraksts ar DelegationType = [Aizgādnis], periods atbilst esošajam datumam un atļaujas statuss = [Aktuāls], tad pārbauda aizgādnības faktu PMLP IR.

* Ja aizgādnība ir spēkā esoša, tad PatientRole = P1 (rīcībnespējīgs)
* Ja aizgādnība nav spēkā esoša, tad PatientRole nemainās. Šajā gadījumā maina atļaujas statusu uz [Neaktuāls].

Nosaka UserRole

1. Ja pacienta identifikācija sakrīt ar lietotāja identifikāciju, tad nosaka, ka UserRole = PatientRole.
2. Ja pacienta identifikācija atšķiras no lietotāja identifikācijas, tad pārbauda lietotāja un pacienta attiecību tipu. Datus pieprasa no PMLP IR. Pēc tipa nosaka UserRole:

* „Viens no vecākiem” vai „Aizbildnis”:

1. Ja PatientRole = P1, tad UserRole = P4. Atjauno vai izveido ierakstu PatientDelegations tabulā ar DelegationType = [Viens no vecākiem] vai [Aizbildnis].
2. Ja PatientRole = P2, tad UserRole = P5. Atjauno vai izveido ierakstu PatientDelegations tabulā ar DelegationType = [Viens no vecākiem] vai [Aizbildnis].
3. Ja PatientRole = P3, tad UserRole = P6

* „Aizgādnis”:

1. Pārbauda, vai PatientDelegations tabulā pacientam ir atļaujas ieraksts ar DelegationType = [Aizgādnis], DelegationPerson = lietotājs, DelegatedTimeFrom un DelegatedTimeTo atbilst esošajam datumam un ieraksta statuss = [Aktuāls]. Ja ir, tad UserRole = P7.

Ja UserRole vēl nav noteikts, tad pielasa informāciju no tabulas PatientDelegations, kur DelegationType = [Pilnvarojums], DelegationPerson = lietotājs, DelegatedTimeFrom un DelegatedTimeTo atbilst esošajam datumam un ieraksta statuss = [Aktuāls]. Ja šāds ieraksts atrasts, tad UserRole = PatientDelegations.DelegatedRole.

1. Ja UserRole nav noteikta, tad lietotājam nav tiesību uz pacientu, tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte).

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | UserRole | CD | 1..1 | Detalizētā loma. |

### checkRigthsAccessPatientCard (Pārbaudīt tiesības piekļūt pacienta kartei)

Funkcija: checkRightsAccessPatientCard

Tiesības: sistēmas funkcija (tiesības netiek pārbaudītas)

Apraksts

Pārbaudīt tiesības lietotājam izpildīt pieprasīto funkciju vai piekļūt pacienta kartes datiem.

Ieejas Parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | | PatientID | int | 0..1 | Pacienta kartes sistēmas identifikators |
| 2 | | IdentificationType | nvarchar(50) | 0..1 | Pacienta kartes identifikācijas tips (personas kods vai alternatīva identifikācija). Jābūt aizpildītām, ja nav aizpildīts lauks PatientID |
| 3 | | IdentificationCode | Nvarchar(50) | 0..1 | Pacienta kartes identifikators (personas kods, vai alternatīva identifikācija). Lauks jābūt aizpildīts, ja nav aizpildīts PatientID |
| 4 | | RequesterIdentificationType | Char(1) | 1..1 | Pieprasītāja identifikācijas tips (personas kods vai alternatīva identifikācija). |
| 5 | | RequesterIdentificationCode | Nvarchar(50) | 1..1 | Pieprasītāja identifikācijas kods (personas kods vai alternatīva identifikācija). |

Apstrādes algoritms

Tiek izpildīta šāda darbu secība:

1. Ja funkcijai netika nodots pacienta kartes identifikators, tad tiek atlasīts identifikācijas tipam un kodam atbilstošs pacienta kartes identifikators no tabulas Patients. Ja nav atrasts pacients, kas atbilst identifikācijas kodam un tipam, tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte), un izveidots audita ieraksts.

Nosaka lietotāja grupu, izmantojot nodoto SAML tokenu: (Pacients, Ārstniecības persona, Izmeklētājs).

Ja lietotājs ir pacients, un pieprasītāja identifikācija vienāda ar pieprasītās kartes identifikāciju (RequesterIdentificationCode = IdentificationCode), tad nosaka pacienta detalizēto lomu (6.5.2 checkDetailedRole (Noteikt detalizēto lomu)), un no SAML talona tiek iegūts tiesību komplekts.

Ja lietotājs ir pacients, un pieprasītāja identifikācija nav vienāda ar pieprasītās kartes identifikāciju (RequesterIdentificationCode <> IdentificationCode), tad tiek pārbaudīts pacientu atļauju saraksta saturs:

* No tabulas PatientDelegations tiek atlasīti ieraksti, kas atbilst šādiem nosacījumiem:
  + - PatienID = PatientID (funkcijas parametrs);
    - IdentificationCode = RequesterIdentificationCode
    - IdentificationType = RequesterIdentificationType
    - Statuss = ACTIVE
    - DelegatedSystem = EVK
    - ViewableDelegation = 1

Ja atlasīto ierakstu daudzums ir lielāks par nulli, tad tiek iegūts pilns pieprasošās personas tiesību komplekts uz pieprasāmo pacienta karti (apvienots kopā visu ierakstu lauka DelegatedFunctionality saturs).

Ja lietotājs ir izmeklētājs, tad tiek pārbaudīts pacientu atļauju saraksta saturs:

* No tabulas PatientDelegations tiek atlasīti ieraksti, kas atbilst šādiem nosacījumiem:
  + - PatienID = PatientID (funkcijas parametrs);
    - IdentificationCode = RequesterIdentificationCode
    - IdentificationType = RequesterIdentificationType
    - Statuss = ACTIVE
    - DelegatedSystem = EVK
    - DelegationType = „Izmeklētājs”

Ja atlasīto ierakstu daudzums ir lielāks par nulli, tad tiek iegūts pilns pieprasošās personas tiesību komplekts uz pieprasāmo pacienta karti (apvienots kopā visu ierakstu lauka DelegatedFunctionality saturs).

Ja lietotājs ir ārstniecības persona, tad atļauj piekļuvi pacienta kartei, saglabājot no drošības talona saņemto tiesību komplektu.

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | UserCheckResult | boolean | 1..1 | Vai lietotājam ir atļautā pieeja pieprasāmai pacienta kartei |
| 2 | UserRightSet | xml | 1..1 | Lietotāja pilns elementāro tiesību komplekts, kas attiecas uz pieprasīto pacienta karti |

Alternatīvie scenāriji

Nav atrasta pacienta karte. Tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte) un izveidots audita ieraksts.

Pieprasītājam nav tiesības apskatīt pieprasīto karti. Tiek atgriezts kļūdas paziņojums (EVK\_0001 Nav atrasta vai nav pieejama pacienta karte) un izveidots audita ieraksts.

### checkUserRightsAccessObject (Pārbaudīt personas tiesību līmeni uz objektu)

Identifikācija

Funkcija: checkUserRightsAccessObject

Tiesības: sistēmas funkcija (tiesības netiek pārbaudītas)

Apraksts

Tiek pārbaudītas sistēmu izsaucošā lietotāja tiesības uz konkrēto objektu

Ieejas Parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | ObjectRightDescriptor | Varchar(3) | 1..1 | Objekta (CDA dokuments, pamatdatu elements) tiesību apraksts, kas nosaka, kādām lietotāju kategorijām ir tiesības piekļūt objektam |
| 2 | UserRightDescriptor | Varchar(6) | 1..1 | Lietotāja tiesību apraksts, kas nosaka, kādas operācijas lietotājam ir tiesības veikt ar objektiem |

Apstrādes algoritms

Objekta tiesību apraksts tiek glabāts datu bāzē pie katra objekta. Tiesību aprakstam ir formāts [DDD], kur D ir 1 vai 0. Atbilstoši tiesību apraksta piemēri ir [000] vai [101]. Tiesību apraksts nosaka, kādām lietotāju grupām ir tiesības redzēt objektu:

* pirmais cipars tiesību aprakstā norāda, vai objektu ir tiesības redzēt pašam pacientam;
* otrais – viņa delegātiem (vecākiem, aizbildņiem) un
* trešais – ārstiem.

0 tiesību aprakstā nozīmē, ka atbilstošas lietotāju grupas pārstāvjiem nav tiesības apskatīt objektu, vērtība 1 – nozīmē, ka tādas tiesības ir.

Lietotāja tiesību apraksts tiek glabāts klasifikatoru reģistrā pie sistēmas definēto lomu klasifikatora. Lietotāja tiesību aprakstam ir formāts [DDD]/[DDD], kur D ir 0 vai 1. Tiesību apraksta pirmā ciparu grupa nosaka, pie kuras lietotāju grupas pieder lietotājs – atbilstošā pozīcijā stāv 1. Otrā ciparu grupa nosaka, lietotāja tiesības mainīt dokumenta redzamību. Lietotāja tiesību apraksta pirmajai ciparu grupai vienmēr jāsatur vienu un tikai vienu vieninieku.

Piemēram, lietotāja tiesību apraksts 100/011 nozīmē, ka lietotājam ir tiesības skatīties pašam pacientam redzamos objektus, un lietotājam ir tiesības likt aizliegumus objektus skatīties delegātiem (vecākiem, aizbildņiem, aizgādņiem) kā arī ārstiem. Tiesību apraksts 100/000 nozīmē, ka lietotājam ir tiesības apskatīt pacientam redzamus objektus, bet nav tiesību uzstādīt papildu aizliegumus.

Nosakot konkrēta objekta redzamību, tiek izpildīta šāda darbību secība:

* Tiek pārbaudīts funkcijai nosūtīts UserRightDescriptor lauka saturs (lietotāju tiesību aprakstam jāatbilst formātam)
* Tiek pārbaudīts funkcijai nosūtīts RightDescriptor lauka saturs (objekta tiesību aprakstam ir jāatbilst formātam)
* Tiek atdalīta lietotāju tiesību apraksta pirmā daļa (trīs simboli)
* Tiek izpildīta loģiskā AND operācija starp objekta tiesību aprakstu un lietotāja tiesību aprakstu. Ja rezultātā saņemts 000, tad funkcija atgriež „false”. Ja rezultāts atšķiras no 000, tad funkcija atgriež „true”.

Funkcijas datu piemēri:

| Nr. | Objekta tiesību apraksts | Lietotāja tiesību apraksts | Rezultāts | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | 111 | 100 011 | True | Lietotājs apskata visiem redzamo objektu |
| 2 | 110 | 100 000 | True | Lietotājs (bērns) apskata tikai viņam redzamo objektu |
| 3 | 101 | 010 101 | False | Lietotāja delegāts (vecāks, aizbildnis) nevar redzēt no delegātiem noslēpto objektu |
| 4 | 011 | 100 000 | False | Lietotājs (bērns) nevar redzēt no lietotāja noslēpto objektu |
| 5 | 100 | 001 110 | False | Lietotājs (ārsts) nevar redzēt no ārstiem un delegātiem noslēpto objektu |

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | CheckResult | boolean | 1..1 | Vai lietotājam ir tiesības apskatīt konkrēto objektu. |

Alternatīvie scenāriji

Objekta tiesību apraksts neatbilst formātam; Tiek atgriezts kļūdas paziņojums (EVK\_0019 Objekta tiesību apraksts neatbilst formātam) un izveidots audita ieraksts.

### checkUserRight (Pārbaudīt personas tiesību)

Identifikācija

Funkcija: checkUserRight

Tiesības: sistēmas funkcija (tiesības netiek pārbaudītas)

Apraksts

Tiek pārbaudītas, vai lietotāja tiesību komplektā ir iekļauta specifiska tiesība.

Ieejas Parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | UserRight | String | 1..1 | Pārbaudāma lietotāja tiesība |
| 2 | Id | II | 1..1 | Pieprasījuma pacienta identifikācija |

Apstrādes algoritms

Tiek izpildīta šāda darbību secība:

1. Tiek iegūts pieprasījuma saml token.
2. No tokena tiek iegūta lietotāja tiesību struktūra „Attribute” (struktūras piemērs ir zemāk):

<AttributeStatement>

<Subject>

<NameIdentifier Format=**"urn:ivis:100001:id-eves"**>**1.3.6.1.4.1.38760.3.4.9.4:dc5da2967c2b48838117bdf21ca2fe32-1.3.6.1.4.1.38760.3.1.1:14028610503**</NameIdentifier>

<SubjectConfirmation>

<ConfirmationMethod>**urn:oasis:names:tc:SAML:1.0:cm:bearer**</ConfirmationMethod>

</SubjectConfirmation>

</Subject>

<Attribute AttributeName=**"action"** AttributeNamespace=**"http://docs.oasis-open.org/wsfed/authorization/200706/claims"**>

<AttributeValue>**EvkRghtAccessRightsSet**</AttributeValue>

<AttributeValue>**EvkRghtClassifiers**</AttributeValue>

**...**

<AttributeValue>**EvkRghtProfileSettings**</AttributeValue>

<AttributeValue>**EvkRghtRequestVaccinations**</AttributeValue>

</Attribute>

<Attribute AttributeName=**"PatientType"** AttributeNamespace=**"http://ip.vm.gov.lv/ws/2011/11/identity/claims"**>

<AttributeValue>**P3\_MAJOR**</AttributeValue>

</Attribute>

<Attribute AttributeName=**"name"** AttributeNamespace=**"http://schemas.xmlsoap.org/ws/2005/05/identity/claims"**>

<AttributeValue>**RONALDS**</AttributeValue>

</Attribute>

<Attribute AttributeName=**"surname"** AttributeNamespace=**"http://schemas.xmlsoap.org/ws/2005/05/identity/claims"**>

<AttributeValue>**ZARITS**</AttributeValue>

</Attribute>

<Attribute AttributeName=**"privatepersonalidentifier"** AttributeNamespace=**"http://schemas.xmlsoap.org/ws/2005/05/identity/claims"**>

<AttributeValue>**14028610503**</AttributeValue>

</Attribute>

<Attribute AttributeName=**"Delegations"** AttributeNamespace=**"http://ip.vm.gov.lv/ws/2011/11/identity/claims"**>

<AttributeValue>

<Actor>

<saml:Attribute xmlns:saml=**"urn:oasis:names:tc:SAML:1.0:assertion"** AttributeName=**"PatientType"** AttributeNamespace=**"http://ip.vm.gov.lv/ws/2011/11/identity/claims"**>

<saml:AttributeValue>**P8\_DELEGATE**</saml:AttributeValue>

</saml:Attribute>

<saml:Attribute xmlns:saml=**"urn:oasis:names:tc:SAML:1.0:assertion"** AttributeName=**"privatepersonalidentifier"** AttributeNamespace=**"http://schemas.xmlsoap.org/ws/2005/05/identity/claims"**>

<saml:AttributeValue>**01017011223**</saml:AttributeValue>

</saml:Attribute>

<saml:Attribute xmlns:saml=**"urn:oasis:names:tc:SAML:1.0:assertion"** AttributeName=**"surname"** AttributeNamespace=**"http://schemas.xmlsoap.org/ws/2005/05/identity/claims"**>

<saml:AttributeValue>**DUBROVSKIS**</saml:AttributeValue>

</saml:Attribute>

<saml:Attribute xmlns:saml=**"urn:oasis:names:tc:SAML:1.0:assertion"** AttributeName=**"action"** AttributeNamespace=**"http://docs.oasis-open.org/wsfed/authorization/200706/claims"**>

<saml:AttributeValue>**EvkRghtAccessRightsSet**</saml:AttributeValue>

<saml:AttributeValue>**EvkRghtClassifiers**</saml:AttributeValue>

<saml:AttributeValue>**EvkRghtContactPersonGet**</saml:AttributeValue>

**...**

<saml:AttributeValue>**EvkRghtPersonDataGet**</saml:AttributeValue>

<saml:AttributeValue>**EvkRghtProfileSettings**</saml:AttributeValue>

<saml:AttributeValue>**EvkRghtRequestVaccinations**</saml:AttributeValue>

</saml:Attribute>

</Actor>

</AttributeValue>

<AttributeValue>

<Actor>

<saml:Attribute xmlns:saml=**"urn:oasis:names:tc:SAML:1.0:assertion"** AttributeName=**"PatientType"** AttributeNamespace=**"http://ip.vm.gov.lv/ws/2011/11/identity/claims"**>

<saml:AttributeValue>**P4\_MINOR\_GUARDIAN**</saml:AttributeValue>

</saml:Attribute>

<saml:Attribute xmlns:saml=**"urn:oasis:names:tc:SAML:1.0:assertion"** AttributeName=**"privatepersonalidentifier"** AttributeNamespace=**"http://schemas.xmlsoap.org/ws/2005/05/identity/claims"**>

<saml:AttributeValue>**14028610503/201210251256**</saml:AttributeValue>

</saml:Attribute>

<saml:Attribute xmlns:saml=**"urn:oasis:names:tc:SAML:1.0:assertion"** AttributeName=**"action"** AttributeNamespace=**"http://docs.oasis-open.org/wsfed/authorization/200706/claims"**>

<saml:AttributeValue>**EvkRghtAccessRightsSet**</saml:AttributeValue>

<saml:AttributeValue>**EvkRghtClassifiers**</saml:AttributeValue>

**...**

<saml:AttributeValue>**EvkRghtProfileSettings**</saml:AttributeValue>

<saml:AttributeValue>**EvkRghtRequestVaccinations**</saml:AttributeValue>

</saml:Attribute>

</Actor>

</AttributeValue>

</Attribute>

1. </AttributeStatement>No struktūras tiek atlasīts saraksts ar Attribute elementiem, kam AttributeName = **"privatepersonalidentifier"**, un AttributeName = „PatientType” (šis ir gadījums, kad tiek apstrādāts pacienta tiesību talons)**.** Tiek pārbaudīts, vai kāds no šādiem atribūtiem sakrīt ar funkcijas parametra Id extension apakšelementu. Gadījumā, ja tāds atribūts atrasts, tad turpmākais darbs notiek ar talona daļu kurā tika atrasta sakritība .
   1. Lietotāja loma tiek noteikta, atlasot no talona šādu struktūru:

<saml:Attribute xmlns:saml=**"urn:oasis:names:tc:SAML:1.0:assertion"** AttributeName=**"PatientType"** AttributeNamespace=**"http://ip.vm.gov.lv/ws/2011/11/identity/claims"**>

<saml:AttributeValue>**P3\_MAJOR**</saml:AttributeValue>

</saml:Attribute>Lietotāja loma tiek iegūta no AttributeValue XML elementa vērtības (šajā gadījumā P3\_MAJOR).

* 1. Tiek atrasts elements ar nosaukumu „Name”, vienāds ar funkcijas parametru „UserRight”. Ja funkcijai iespējamas vairākas tiesības, kuras tiek kontrolētas funkcijas iekšienē, tad jāmeklē vismaz viens no šiem tiesību punktiem.
  2. Ja tāds elements ir atrasts, tad lietotājam ir nepieciešamās tiesības, pretējā gadījumā – nav.

1. Gadījumā, ja AttributeName = „PatientType” tipa elements nav atrasts (profesionāla lietotāja gadījums), talons nesaturēs Delegācijas apakšsekciju un PatientType atribūtu. Šājā gadījumā lietotāja tiesība tiek iegūta no talona galvenē esoša tiesību saraksta.

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | CheckResult | boolean | 1..1 | Vai lietotājam ir konkrēta tiesība |

Alternatīvie scenāriji

Nav

## NVD VIS datu aktualizācijas moduļa projektējums

Nodrošina saskarnes funkcijas ar IP(NVD VIS).

### getFamilyDoctor (Iegūt ģimenes ārsta datus)

Identifikācija

Procedūra: getFamilyDoctor

Tiesības: sistēmas funkcija (tiesības netiek pārbaudītas)

Apraksts

No NVD VIS tiek iegūti personas ģimenes ārsta dati, un tie tiek saglabāti pacienta kartē.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | PersCode | string | 1..1 | Personas kods, kam jāaizpilda ģimenes ārsta dati |

Apstrādes algoritms

1. Tiek izsaukts NVD VIS ģimenes ārstu datu iegūšanas serviss (5.2.6.1 Iegūt pacienta ģimenes ārstu no NVD VIS). Servisam tiek nodoti parametri:

* PatientPersonCode – PersCode
* Date – sistēmas datums

1. Serviss atgriež personas ģimenes ārsta datu komplektu.
2. Ja saņemtais datu komplekts nav tukšs (personai ir iegūta ģimenes ārsta informācija), tad tiek izsaukta funkcija setPatientFamilyDoctor (6.6.2 setPatientFamilyDoctor (Uzstādīt ģimenes ārstu)).

Rezultāts

Funkcijas rezultāts ir pazīme par veiksmīgu/neveiksmīgu izpildi.

Alternatīvie scenāriji

### setPatientFamilyDoctor (Uzstādīt ģimenes ārstu)

Identifikācija

Procedūra: setPatientFamilyDoctor

Tiesības: sistēmas funkcija (tiesības netiek pārbaudītas)

Apraksts

Funkcija tiek izsaukta no:

1. funkcijas getFamilyDoctor (6.6.1 getFamilyDoctor (Iegūt ģimenes ārsta datus)),
2. tīmekļa pakalpes WS:setPatientFamilyDoctorChanges (5.2.6.3 Uzstādīt ģimenes ārstu izmaiņas), kas no NVD VIS saņem datu komplektu ar pacientu ģimenes ārstu izmaiņām.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | PatientFamilyDoctorList | Salikts elements | 1..n | Personas un ģimenes ārsta attiecības |
| 2 | PersCode | string | 1..1 | Personas kods |
| 3 | PhysicianId |  | 1..1 | Ārsta identifikators |
| 4 | InstitutionId |  | 1..1 | Ārstniecības iestādes identifikators |
| 5 | SpecialityId |  | 1..1 | Ģimenes ārsta specialitātes identifikators |
| 6 | ContractDateFrom | date | 1..1 | Datums, no kura ir spēkā ģimenes ārsta attiecības ar aprūpējamo |
| 7 | ContractDateTill | date | 1..1 | Datums, līdz kuram ir spēkā ģimenes ārsta attiecības ar aprūpējamo |

Nosacījumi

Personas ģimenes ārstu identificē datu kopa:

* Personas kods
* Ārsts
* Ārstniecības iestāde
* Ārsta specialitāte
* Datums, no kura ir spēkā attiecības

Šie dati nevar mainīties bez jaunu attiecību izveides. Līdz ar to, vienīgais maināmais datu vienums ir datums, līdz kuram spēkā attiecības.

Apstrādes algoritms

1. Secīgi apstrādā katru saņemto personas un ģimenes ārsta attiecību faktu.
2. Pēc personas koda tiek atlasīta pacienta karte, kurai:

* IdentificationType = konstanta OID vērtība, kas atbilst LV personas koda identifikācijai: 1.3.6.1.4.1.38760.3.1.1
* IdentificationCode = saņemtā vērtība: PersCode

1. Ja pacienta karte nav atrasta, tad funkcija tiek turpināta ar jauno pacienta ģimenes ārsta ierakstu.
2. Ja pacienta karte atrasta, tad iegūst PatientGUID vērtību.
3. Pielasa informāciju no tabulas PatientFamilyDoctors pēc atlases parametriem:

* PatientGUID = iegūtais PatientGUID
* PhysicianId = saņemtais parametrs: PhysicianId
* InstitutionId = saņemtais parametrs: InstitutionId
* SpecialityId = saņemtais parametrs: SpecialityId
* DoctorTimeFrom = saņemtais parametrs: ContractDateFrom

1. Ja ieraksts vai ieraksti tabulā PatientFamilyDoctors eksistē, tad šiem ierakstiem labo lauka vērtību:

* DoctorTimeTo = saņemtais parametrs: ContractDateTill

1. Tiek pārbaudīti ģimenes ārsta, organizācijas un specialitātes dati uz atbilstību klasifikatoriem aizpildīšanas obligātumu. Gadījumā, ja datos ir kļūda (dati neatbilst klasifikatoriem, vai datu formāts ir nepareizs, tad kļūdains ieraksts tiek saglabāts tabulā EVKPers.ErrorLog. Kļūdas individuālājos ģimenes ārsta izmaiņu ierakstos neietekmē procedūras kopējo izpildi (visi iesūtītie ģimenes ārstu izmaiņu ieraksti tiks apstrādāti).
2. Gadījumā, ja datos nav kļūdas, tiek veidots jauns ieraksts tabulā PatientFamilyDoctors, aizpildot datu laukus:

* PatientFamilyDoctorID – tabulas primārā atslēga
* PatientGUID – parametrs:PatientGUID
* PhysicianID – saņemtā vērtība:PhysicianId
* InstitutionID – saņemtā vērtība:InstitutionId
* SpecialityID – saņemtā vērtība:SpecialityId
* RecordStatus – vērtība, kas atbilst statusam Aktuāls
* DoctorTimeFrom – saņemtā vērtība:PatientContractDateFrom
* DoctorTimeTo – saņemtā vērtība:PatientContractDateTill

Rezultāts

Funkcijas rezultāts ir pazīme par veiksmīgu/neveiksmīgu izpildi.

Ja funkcijas darbības rezultātā tiek iegūta neapstrādāta kļūda, tad darbība tiek pārtraukta – tiek veikta funkcijā veikto darbību atcelšana (transaction rollback), un kļūda tiek nodota izsaucošajai funkcijai (Exception).

### getEhicData (Iegūt EVAK kartes datus)

Identifikācija

Procedūra: getEhicData

Tiesības: sistēmas funkcija (tiesības netiek pārbaudītas)

Apraksts

No NVD VIS tiek iegūti pacientam reģistrētās aktuālās EVAK kartes dati, un tie tiek saglabāti pacienta kartē.

Ieejas Parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | PersCode | string | 1..1 | Personas kods, kam nepieciešams atlasīt EVAK datus no VNC IS |

Apstrādes algoritms

1. Tiek izsaukts NVD VIS EVAK karšu datu iegūšanas serviss (5.2.6.2 Iegūt pacienta EVAK datus no NVD VIS). Servisam tiek nodoti parametri:

* PatientPersonCode – PersCode
* Date – sistēmas datums

1. Serviss atgriež personas EVAK datu komplektu:
2. Ja saņemtais datu komplekts nav tukšs (personai ir EVAK kartes informācija), tad tiek izsaukta funkcija setPatientEhicData (6.6.4 setPatientEhicData (Uzstādīt EVAK datus).

Rezultāts

Funkcijas rezultāts ir pazīme par veiksmīgu/neveiksmīgu izpildi.

Alternatīvie scenāriji

### setPatientEhicData (Uzstādīt EVAK datus)

Identifikācija

Procedūra: setPatientEhicData

Tiesības: sistēmas funkcija (tiesības netiek pārbaudītas)

Apraksts

Funkcija tiek izsaukta no:

1. funkcijas getEhicData (6.6.3 getEhicData (Iegūt EVAK kartes datus)),
2. tīmekļa pakalpes WS:setEhicChanges (5.2.6.4 Uzstādīt EVAK karšu izmaiņas), kas no NVD VIS saņem datu komplektu ar pacienta EVAK karšu datu izmaiņām.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | PatientEhic | Salikts elements | 1..n | Personas EVAK kartes dati |
| 2 | PersCode | string | 1..1 | Personas kods |
| 3 | EhicNumber |  | 1..1 | Kartes identifikācijas numurs |
| 4 | EhicIssueDate |  | 0..1 | Kartes izsniegšanas datums |
| 5 | IssuingInstitution | nvarchar | 1..1 | Kartes izsniedzēja organizācija: kods un nosaukums |
| 6 | ValidFrom | date | 1..1 | Datums, no kura karte ir spēkā |
| 7 | ValidTill | date | 1..1 | Datums, līdz kuram karte ir spēkā |
| 8 | EhicStatus | nvarchar | 1..1 | Kartes statuss |

Nosacījumi

EVAK karti identificē datu kopa:

* Kartes identifikācijas numurs
* Izsniedzēja organizācija

Apstrādes algoritms

1. Secīgi apstrādā katru saņemto EVAK kartes datu kopu.
2. Pēc personas koda tiek atlasīta pacienta karte, kurai:

* IdentificationType = konstanta OID vērtība, kas atbilst LV personas koda identifikācijai: 1.3.6.1.4.1.38760.3.1.1
* IdentificationCode = saņemtā vērtība: PersCode

1. Ja pacienta karte nav atrasta, tad funkcijas izpilde tiek turpināta ar nākamo EVAK kartes ierakstu.
2. Ja pacienta karte atrasta, tad iegūst PatientGUID vērtību.
3. Pielasa informāciju no tabulas PatientEhicData pēc atlases parametriem:

* PatientGUID = iegūtais PatientGUID
* EhicNumber = saņemtais parametrs: EhicNumber
* IssuingInstitution = saņemtais parametrs: IssuingInstitution

1. Ja ieraksts vai ieraksti tabulā PatientEhicData eksistē, tad šiem ierakstiem labo lauku vērtību:

* ValidTill = saņemtais parametrs: ValidTill
* Status = saņemtais statuss: Status

1. Ja ieraksts tabulā PatientEhicData neeksistē, tad:
2. Pārbauda, vai saņemtie dati atbilst klasifikatoriem. Tiek pārbaudīti izdošanas organizācijas dati. Gadījumā, ja datos ir kļūda (dati neatbilst klasifikatoriem, vai datu formāts ir nepareizs), kļūdainais ieraksts tiek saglabāts tabulā EVKPers.ErrorLog. Kļūdas individuālājos EVAK karšu ierakstos neietekmē procedūras kopējo izpildi (visi iesūtītie pacientu EVAK karšu izmaiņu ieraksti tiks apstrādāti).
3. Gadījumā, ja datos nav kļūdas, tiek veidots jauns ieraksts, aizpildot datu laukus:

* PatientEhicID – tabulas primārā atslēga
* PatientGUID – parametrs:PatientGUID
* EhicNumber – saņemtā vērtība:EhicNumber
* DateOfIssue – saņemtā vērtība:EhicIssueDate
* IssuingInstitution – saņemtā vērtība:IssuingInstitution
* ValidFrom – saņemtā vērtība:ValidFrom
* ValidTill – saņemtā vērtība:ValidTill
* Status – saņemtā vērtība:EhicStatus

Rezultāts

Funkcijas rezultāts ir pazīme par veiksmīgu/neveiksmīgu izpildi.

Ja funkcijas darbības rezultātā tiek iegūta neapstrādāta kļūda, tad darbība tiek pārtraukta – tiek veikta funkcijā veikto darbību atcelšana (transaction rollback), un kļūda tiek nodota izsaucošajai funkcijai (Exception).

## Personas datu aktualizācijas moduļa projektējums

Nodrošina saskarnes funkcijas ar IP(PMLP IR).

### getPersonData (Iegūt personas datus)

Identifikācija

Procedūra: getPersonData

Tiesības: sistēmas funkcija (tiesības netiek pārbaudītas)

Apraksts

Tiek veikts secīgs PMLP funkciju izsaukums lai savāktu nepieciešamos personas datus, un saglabātu šos datu datubāzē.

Ieejas Parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | PersCode | string | 1..1 | Personas kods, kam nepieciešams atlasīt personas datus no PMLP VIS |
| 2 | PatientGUID | guid | 1..1 | Pacienta kartes identifikators |

Apstrādes algoritms

Tiek izpildīta šāda darbību secība:

1. Tiek izsaukts PMLP IR vārda un uzvārda iegūšanas serviss „PMLP-GetPersonName” (sk. 5.2.7.2 Iegūt personas datus), nododot tam kā parametru PersCode vērtību no ieejas parametriem.
2. Tiek sagaidīta atbilde un iegūtā struktūra saglabāta datu bāzes tabulā „Patients” pie ieraksta, kuram atbilst filtrs „Patients.PatientGUID = ieejas parametra PatientGUID” laukos:
   * GivenName = FirstName + OtherNames;
   * FamilyName = LastName.
3. Tiek izsaukts PMLP IR dzimšanas datuma iegūšanas serviss „PMLP-GetBirthDate” (sk. 5.2.7.2 Iegūt personas datus), nododot tam kā parametru PersCode vērtību no ieejas parametriem.
4. Tiek sagaidīta atbilde un iegūtā struktūra saglabāta datu bāzes tabulā „Patients” pie ieraksta, kuram atbilst filtrs „Patients.PatientGUID = ieejas parametra PatientGUID” laukos:
   * BirthTime= PersonBirthDate.
5. Iegūtā struktūra tiek saglabāta datu bāzes tabulā „NPatients” pie ieraksta, kuram atbilst filtrs „NPatients.PatientID = PatientID”, kur PatientID tiek sameklēts tabulā „NPatients”, veicot meklēšanu pēc ieejas parametros norādītā PatientGUID laukos:
   * YearOfBirth= year(PersonBirthDate).
6. Tiek izsaukts PMLP IR dzimuma iegūšanas serviss „PMLP-GetPersonSex” (sk. 5.2.7.2 Iegūt personas datus), nododot tam kā parametru PersCode vērtību no ieejas parametriem.
7. Tiek sagaidīta atbilde un iegūtā struktūra saglabāta datu bāzes tabulā „NPatients” pie ieraksta, kuram atbilst filtrs „NPatients.PatientID = PatientID”, kur PatientID tiek izmantots no iepriekšējā soļa:
   * GenderValue= PersonSexCode;
   * GenderName= PersonSex.
8. Tiek izsaukts PMLP IR adresi iegūšanas serviss „PMLP-GetPersonLivingPlace” (sk. 5.2.7.2 Iegūt personas datus), nododot tam kā parametru PersCode vērtību no ieejas parametriem.
9. Tiek sagaidīta atbilde un iegūtā struktūra saglabāta datu bāzes tabulā „Patients” pie ieraksta, kuram atbilst filtrs „Patients.PatientGUID = ieejas parametra PatientGUID” laukos:
   * Village= VillageName;
   * Street = StreetName;
   * HouseName = HomeTitle;
   * HouseNumber = HouseNo + HouseNoText;
   * HouseBlockNumber = BlockNo;
   * FlatNumber = ApartmentNo + ApartmentNoText;
   * PostalCode = PostIndex;
   * AddressAbroad = AddressAbroad.
10. Tiek sagaidīta atbilde un iegūtā struktūra tiek saglabāta datu bāzes tabulā „NPatients” pie ieraksta, kuram atbilst filtrs „NPatients.PatientID = PatientID”, kur PatientID tiek izmantots no iepriekšējā soļa:
    * ICAOCode = Country:Code;
    * ICAOName = Country:Name;
    * ARCode = ARCode;
    * ATUCode1 = ATUCode1;
    * ATUName1 = ATUName1;
    * ATUCode2 = ATUCode2;
    * ATUName2 = ATUName2;
    * ATUCode3 = ATUCode3;
    * ATUName3 = ATUName3;
    * ATUCode4 = ATUCode4;
    * ATUName4 = ATUName4;
    * StreetCode = StreetCode;
    * VillageCode = VillageCode;

Personai tiek izveidots pieprasījums vecāku datu iegūšanai no PMLP; tiek izveidots ieraksts tabulā PatientMonitoring, norādot:

* pieprasījuma tipu (atbilst vecāku iegūšanas tipam)
* personas kodu
* pacienta identificējošo GUID vērtību
* mēģinājumu skaitu (0)
* ieraksta statusu (tiek uzstādīta vērtība NEW)
* kļūdas aprakstu (paliek tukšs).

1. Ja apstrāde notikusi veiksmīgi, tad tiek atgriezts veiksmes paziņojums, citādi – atbilstošais kļūdas kods.

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | Error | Array[string] | 1..1 | Saraksts ar procesā notikušajiem kļūdām |

Alternatīvie scenāriji

Funkcijas izpildes laikā radusies kļūda: audita žurnālā tiek veikts ieraksts, par radušos kļūdu. Tiek pārtraukta funkcijas izpilde.

### getChangesPersonData (Iegūt personas datu izmaiņas)

Identifikācija

Procedūra: getChangesPersonData

Tiesības: sistēmas funkcija (tiesības netiek pārbaudītas)

Apraksts

Tiek nodrošināta piereģistrēto pacientu personas kodu reģistrācija PMLP monitoringa sistēmā, un izmaiņu datu iegūšana no monitoringa DIT kanāla. Iegūtie dati tiek ierakstīti EVK datu bāzē.

Ieejas Parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | PersonCode | string | 1..1 | Pacienta personas kods |

Apstrādes algoritms

Pacienta dati tiek pievienoti monitoringam, kad tiek veidota pacienta karte. Tiek izpildītas šādas darbības:



* Veidojot pacienta karti, tiek izveidots pacienta identifikācijas ieraksts tabulā PatientIdentifications. Lauki MonitoringOnDate un MonitoringOffDate netiek aizpildīti.
* Sistēmā tiek nokonfigurēts process, kurš nosūta personu datus monitoringam (procesa palaišanas laiks tiek kontrolēts ar standarta SQLServer līdzekļiem). Process sastāv no šādas darbību secības:
* Tiek atlasītas visas personu identifikācijas, kuriem lauks MonitoringOnDate nesatur vērtības (satur NULL vērtību) un identifikācijas tips ir personas kods (OID: 1.3.6.1.4.1.38760.3.1.1);
* Tiek izveidots fails, kas satur visas šādi atlasītas vērtības (faila struktūra atbilst IP monitoringa pieprasījuma faila struktūrai – skat. [6]);
* Visiem pacienta identifikācijas ierakstiem, kuri tika pievienoti failam tiek uzstādīta lauka MonitoringOnDate vērtība vienāda ar pašreizējo datumu/laiku;
* Tiek izveidots jauns GUID, kas unikāli identificē izveidoto personas kodu failu;
* Izveidotais fails tiek nosūtīts kā ziņojums servisam BackgroundService, nosūtīšanai uz PMLP monitoringa DIT kanālu;
* Saņemot atbildi no BackgroundService servisa, tiek palaista personas kodu faila pievienošanas veiksmīguma pārbaude, nosūtot servisam BackgroundMonitoringStart ziņojumu checkStart .

Izveidota personas kodu faila piemērs:

<MonitoringRequest xmlns="http://ivis.eps.gov.lv/XMLSchemas/100001/Monitoring/v1-0">

<Switch>ON</Switch>

<PersonCodeList>

<PersonCode xmlns="http://ivis.eps.gov.lv/XMLSchemas/100001/Monitoring/v1-0">personas\_kods</PersonCode>

.....

<PersonCode xmlns="http://ivis.eps.gov.lv/XMLSchemas/100001/Monitoring/v1-0">01011211123</PersonCode>

</PersonCodeList>

</MonitoringRequest>

Saņemto personas kodu sarakstu BackgroundService apstrādā pēc šāda algoritma:



* Saņemot ziņojumu ar personas kodu failu, serviss slēdzās pie DIT sistēmas, atlasot DIT kanāla adresi no sistēmas parametru tabulas (parametra nosaukums MonitoringWriteDITAddress);
* Tiek izveidots sftp pieslēgums pie DIT sistēmas, un pēc iepriekšēja solī iegūtās adreses saglabāts ziņojumā saņemtais personas kodu fails;
* Kad no ftp tiek saņemts veiksmīgas saglabāšanas apstiprinājums, sistēma raksta audita ziņojumu par veiksmīgo pacientu personas kodu komplekta pievienošanu monitoringam un palaiž kļūdu pārbaudes procedūru, nosūtot ziņojumu checkErrors servisam BackgroundService;
* Saņemot šo ziņojumu, BackgroundService atlasa DIT kļūdu kanāla adresi (sistēmas parametra nosaukums MonitoringWriteErrorsDITAddress);
* Tiek iegūts visu nepareizi saglabāto failu saraksts no DIT kanāla direktorijas ReadErrors. Ja pēc sistēmas parametros definēta laika izteces direktorijā neparādās ziņojums par neveiksmīgi ielasīto personas kodu komplektu, tas skaitās veiksmīgi saglabāts;
* BackgroundMonitoringStart servisam tiek atgriezts personas kodu monitoringa saglabāšanas rezultāts un tiek izveidots audita ieraksts (ierakstot audita loga saglabāšanas rezultātu – veiksmīgs vai kļūda).

Pacienta datu izmaiņas tiek atlasītas no cita PMLP monitoringa DIT kanāla („MonitoringaDati” ). Šī kanāla datu apstrādes algoritms ir aprakstīts zemāk:



* Pēc taimera tiek nosūtīts ziņojums uz BackgroundService servisu, saņemot kuru, notiek savienošanās ar DIT sistēmu (kanāla nosaukums tiek atlasīts no sistēmas parametru tabulas – parametra nosaukums MonitoringReadDITAddress);
* No DIT kanāla tiek saņemts saraksts ar monitoringa datu izmaiņu failiem un atgriezts servisam BackgroundMonitoringStart;
* Failu saraksts tiek apstrādāts, atlasot failu nosaukumus un identifikatorus un reģistrējot tos sistēmas tabulā SBBackgroundFiles;
* Katram šādi iegūtam failam tiek izsaukts BackgroundService un fails tiek atlasīts no DIT sistēmas;
* Fails tiek atgriezts un tā saturs saglabāts tabulā SBBackgroundFiles;
* Faila saturs tiek apstrādāts, sadalot to pēc personas kodiem, identificējot kādas personas datu izmaiņas notika katrai reģistrētai personai;
* Tiek sastādīts saraksts ar katras personas datu grupām, kurās notika izmaiņas;
* Saraksts tiek sadalīts pa personas kodiem un katram personas kodam tiek izveidots ieraksts tabulā PatientMonitoring, norādot:
* pieprasījuma tipu;
* personas kodu;
* pacienta identificējošo GUID vērtību;
* mēģinājumu skaitu (0);
* ieraksta statusu (tiek uzstādīta vērtība NEW);
* kļūdas aprakstu (paliek tukšs).

Izejas parametri

Funkcija neatgriež vērtības. Gadījumā, ja izpildot funkciju notiek kļūda, tad tiek veidots ieraksts sistēmas auditā.

Alternatīvie scenāriji

Nav.

### checkPersonDataUpdate (Personas datu aktualizācija)

Identifikācija

Funkcija: checkPersonDataUpdate

Tiesības : -

Apraksts

Funkcija nosaka, kad pacienta dati tika pārbaudīti PMLP IR sistēmā pēdējo reizi. Gadījumā, ja ir realizēta PMLP IR monitoringa funkcionalitāte, tad tiek pārbaudīti tikai dati par pacienta bērniem. Funkcija tiek izsaukta, kad pacients pieslēdzas EVK apskatīt savus datus (tas tiek pārbaudīts checkDetailedRole izsaukuma laikā).

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | Pacienta id | II | 1..1 | Lietotāja, kurš pieslēdzas EVK sistēmai, identifikators (funkcija tiek izsaukta tikai gadījumā, kad pacients pieslēdzas apskatīt savus datus). |

Apstrādes algoritms

1. Tiek pārbaudīts, vai personai ir izveidota pacienta karte. Ja pacienta karte nav atrasta, tad tiek atgriezta kļūda un funkcijas izpilde tiek pārtraukta.
2. No tabulas UpdateList tiek atlasīta informācija, kad pacienta personas dati tika atjaunoti pēdējo reizi.
3. Tiek atlasīts sistēmas parametrs PersonUpdateDays un lietotāja datu pēdējā atjaunošana tiek salīdzināta ar šo parametra vērtību.
4. Ja tiek konstatēts, ka lietotāja dati nav atjaunoti ilgāk par parametrā noradīto dienu skaitu, tad tiek izsaukta (asinhrona režīmā) pacienta datu atjaunošanas procedūra (skat. 6.13.3 Pacienta kartiņas aizpilde). Izsaucot pacienta kārtiņas aizpildi, atkarībā no sistēmas parametra PersonUpdateMode tiek pieprasīta vai nu visu pacienta personas datu atlasīšana (gadījumā, ja nav pieejams PMLP monitorings) vai tikai bērnu informācija (ja monitorings ir pieejams)
5. Tiek uzstādīts jauns pacienta datu pēdējās izmaiņas datums (norādot pašreizējo momentu kā šo datumu).

Izejas parametri

1. Funkcija neatgriež vērtības. Gadījumā, ja izpildot funkciju notiek kļūda, tad tiek atgriezts kļūdas ziņojums (Exception).

Alternatīvie scenāriji

Notiek neparedzēta kļūda. Kļūdas ziņojums tiek atgriezts izsaucošai funkcijai.

## VDEĀVK IS datu aktualizācijas moduļa projektējums

### getDisabilityData (Iegūt personas invaliditātes datus)

Identifikācija

Procedūra: getDisabilityData

Tiesības: sistēmas funkcija (tiesības netiek pārbaudītas)

Apraksts

No IVIS tiek iegūti personas invaliditātes dati, un tie tiek saglabāti pacienta kartē.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | PersCode | string | 1..1 | Personas kods, kam jāiegūst invaliditātes dati |

Apstrādes algoritms

1. Tiek izsaukts IVIS invaliditātes informācijas serviss (5.2.8.1 Iegūt personas invaliditāti). Servisam tiek nodoti parametri:

* AboutIdentityNumber – Personas kods;
* EffectiveDate – sistēmas datums;

1. Serviss atgriež personas invaliditātes datu komplektu;
2. Tiek izsaukta funkcija setPatientDisabilityData (6.8.2 setPatientDisability (Personas invaliditātes datu aktualizācija)), kā parametrus tai nododot saņemto datu komplektu.

Rezultāts

Funkcijas rezultāts ir pazīme par veiksmīgu/neveiksmīgu izpildi.

### setPatientDisability (Personas invaliditātes datu aktualizācija)

Identifikācija

Procedūra: setPatientDisability

Tiesības: sistēmas funkcija (tiesības netiek pārbaudītas).

Apraksts

Funkcija tiek izsaukta no funkcijas getDisabilityData (6.8.1 getDisabilityData (Iegūt personas invaliditātes datus)).

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1. | PersonInv | Salikts elements | 1..\* | Personas invaliditātes dati |
| 2. | PersCode | string | 1..1 | Personas kods |
| 3. | InvGroup | string | 1..1 | Invaliditātes grupa |
| 4. | InvGroupName | string | 1..1 | Invalid. grupas nosaukums |
| 5. | InvFrom | string | 1..1 | Invaliditātes lēmuma datums |
| 6. | InvTo | string | 1..1 | Invaliditāte noteikta līdz |
| 7. | InvDiagnosis | string | 1..1 | Invaliditātes diagnoze |
| 8. | InvNumber | string | 1..1 | Invaliditātes lēmuma numurs |
| 9. | InvOfficialName | string | 1..1 | Invaliditāti piešķirošās amatpersonas vārds |
| 10. | InvOfficialSurname | string | 1..1 | Invaliditāti piešķirošās amatpersonas uzvārds |

Apstrādes algoritms

1. Pielasa atbilstošās personas invaliditātes pamatdatus no datu bāzes, izmantojot objekta identificējošo datu kopu saskaņā ar tabulu „Veselības pamatdatu objektu atbilstības tabula”.
2. Secīgi apstrādā katru saņemto personas invaliditātes ierakstu un salīdzina to ar no pamatdatiem izgūto informāciju.
3. Tiek sagatavots CDA dokuments atbildoši pamatdatu dokumenta modelim „1.3.6.1.4.1.38760.1.2.1.1.1”. Gadījumā, ja no VDEĀVK saņemtajā datu kopā netiek identificēts ieraksts, kurš ir reģistrēts pamatdatos, pamatdatos reģistrētajai informācijai tiek veidota sekcija ar ierakstu „Pacienta invaliditātes grupa” pievienojot parametru [negationInd = 1].

Dokuments tiek ģenerēts pēc šāda algoritma:

1. Aizpilda galveni (header) pēc šādas tabulas informācijas :

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | templateId | [root] 1.3.6.1.4.1.38760.1.2.1.1  [extension] 1.3.6.1.4.1.38760.1.2.1.1.1 |
| 2 | TypeId | [root] 2.16.840.1.113883.1.3  [extension] POCD\_HD000040 |
| 3 | Id | [root] 1.3.6.1.4.1.38760.3.4.5.1  [extension] VDEAVK.[KeyWords].NewDocumentNo.versionNumber. |
| 4 | Code | [code] 1.3.6.1.4.1.38760.1.2.1.1  [codeSystem] 1.3.6.1.4.1.38760.1.2.1  [displayName] Medicīnas pamatdatu kopsavilkuma dokuments |
| 5 | Title | „Medicīnas pamatdatu kopsavilkuma dokuments invaliditātes informācijas sistēmas reģistrs” |
| 6 | effectiveTime | Ģenerēšanas datums un laiks |
| 7 | confidentialityCode | „N” |
| 8 | languageCode | „LV” |
| 9 | setId | [root] 1.3.6.1.4.1.38760.3.4.5.4  [extension] VDEAVK.[KeyWords].NewDocumentNo |
| 10 | versionNumber | 1 |
| 11 | recordTarget.patientRole.id | [root] 1.3.6.1.4.1.38760.3.1.1  [extension] PersonasKods |
| 12 | author | author.time  [value] = Ģenerēšanas datums un laiks  Author.assignedAuthor.id  [root]= 1.3.6.1.4.1.38760.3.4.5.29 |
| 13 | custodian | custodian.assignedCustodina.representedCustodianOrganization.Id  [root] = 1.3.6.1.4.1.38760.3.2.7  [extension] = „VDEAVK”  custodian.assignedCustodina.representedCustodianOrganization.name = „Veselības un darbspēju ekspertīzes ārstu valsts komisija” |

1. Pamatdatu tipam veido sekciju (section), kuru aizpilda pēc atbilstošas tabulas :

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | Section | [code.code] = „1.3.6.1.4.1.38760.1.2.2.142”  [code.codeSystem] = „1.3.6.1.4.1.38760.2.21”  [code.codeSystemName] = „OID reģistrs”  [code.displayName] = „Pacienta invaliditātes sekcija”  [title] = „Invaliditāte” |

1. Pamatdatu tipam veido ierakstus (entry), kurus aizpilda pēc atbilstošas tabulas :

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1. | Entry | [code.code] = „1.3.6.1.4.1.38760.1.2.3.781”  [code.codesystem] = „1.3.6.1.4.1.38760.2.21”  [code.codeSystemName] = „OID reģistrs”  [code.displayName] = „Pacienta invaliditātes grupa”  [value.code] = „I”  [value.displayName] = „I invaliditātes grupu — ļoti smaga invaliditāte”  [value.codeSystem] =  [negationInd] = 1 (gadījumā, ja invaliditātes informāciju nepieciešams atcelt) |
| 2. | Entry  entryRelationship[1.] | [code.code] = „1.3.6.1.4.1.38760.1.2.3.782”  [code.codesystem] = „1.3.6.1.4.1.38760.2.21”  [code.codeSystemName] = „OID reģistrs”  [code.displayName] = „Invaliditātes lēmuma datums”  [effectiveTime.value] = [InvFrom] |
| 3. | Entry  entryRelationship[1.] | [code.code] = „1.3.6.1.4.1.38760.1.2.3.783”  [code.codesystem] = „1.3.6.1.4.1.38760.2.21”  [code.codeSystemName] = „OID reģistrs”  [code.displayName] = „Invaliditāte noteikta līdz”  [effectiveTime.value] = [InvTo] |
| 4. | Entry  entryRelationship[1.] | [code.code] = „1.3.6.1.4.1.38760.1.2.3.1393”  [code.codesystem] = „1.3.6.1.4.1.38760.2.21”  [code.codeSystemName] = „OID reģistrs”  [code.displayName] = „Invaliditātes diagnoze”  [value.code] = „InvDiagnosis”  [value.displayName] = 1.3.6.1.4.1.38760.2.159 klasifikatora atbilstošais displayName  [value.codeSystem] = 1.3.6.1.4.1.38760.2.159 |
| 5. | Entry  entryRelationship[1.] | [code.code] = „1.3.6.1.4.1.38760.1.2.3.1394”  [code.codesystem] = „1.3.6.1.4.1.38760.2.21”  [code.codeSystemName] = „OID reģistrs”  [code.displayName] = „Lēmuma numurs”  [value] = „InvNumber” |
| 6. | Entry  entryRelationship[1.] | [code.code] = „1.3.6.1.4.1.38760.1.2.3.1395”  [code.codesystem] = „1.3.6.1.4.1.38760.2.21”  [code.codeSystemName] = „OID reģistrs”  [code.displayName] = „Lēmuma amatpersona”  [value.given] = InvOfficialName  [value.family] = InvOfficialSurname |

1. CDA dokuments tiek sūtīts uz pakalpi „6.2.1 addDocument ”, kas nodrošina CDA dokumentu reģistrāciju un apstrādi. Atbilstoši CDA dokumentu sekciju ierakstu apstrādes uzdevumam „6.2.12.7 documentEntryProcessing (dokumenta ierakstu apstrāde)”, CDA dokumentā iekļautie dati tiks izgūti no CDA dokumenta sekciju ierakstiem un ievietoti personas pamatdatu tabulās.

## Jaundzimušo reģistra datu aktualizācijas moduļa projektējums

### setNewBornPersonCodes (Iegūt jaundzimušo personas kodus)

Identifikācija

Funkcija: setNewbornPersonCodes

Tiesības: sistēmas funkcija (tiesības netiek pārbaudītas)

Apraksts

Tiek pārbaudīti sistēmā reģistrētie pacienti bez personas koda identifikācijas un tiek izdarīts mēģinājums iegūt personas kodu no Jaundzimušo reģistra sistēmas.

Ieejas parametri

Nav ieejas parametru

Apstrādes algoritms

Tiek izpildīta šāda darbību secība:

1. Tiek izveidots saraksts ar sistēmā reģistrētiem pacientiem, kam ir tikai alternatīva (jaundzimušā) identifikācija.

* Tiek atlasīti visi ieraksti no pacientu identifikāciju tabulas (PatientIdentifications), kur vienai PatientIdentificationId lauka vērtībai atbilst tikai viens identifikācijas veids, un identifikācijas tips (IdentificationType) ir 1.3.6.1.4.1.38760.3.1.3.

1. Tiek izveidots pieprasījums jaundzimušo reģistra sistēmai.

* Katrai rindai no iepriekšējā punkta tiek izveidots pieprasījuma elements Newborn; Tiek iegūta pacienta identifikācija (piemēram 15038511229/20111018101003). Alternatīvās identifikācijas kods tiek sadalīts divās daļās pēc simbola /.
* No pirmās identifikatora daļas tiek izveidots lauks MotherPersonCode (dzēšot no personas koda ieraksta „-„).
* No daļas pēc / tiek izveidots lauks BirthDateTime.

1. No visiem iepriekšējā punktā izveidotiem pieprasījuma elementiem tiek sastādīts saraksts un nosūtīts Jaundzimušo reģistra personas kodu atgriešanas servisam.
2. No Jaundzimušo reģistra tiek saņemts saraksts ar personas kodiem tiem ierakstiem, kuriem personas kods jau ir uzstādīts.
3. Katram saņemtam ierakstam tiek papildināta pacienta karte;

* Saņemtā ieraksta piemērs:

<Newborn>

<MotherPersonCode>01018012345</MotherPersonCode>

<BirthDateTime>2011-10-01 01:10</BirthDateTime>

<PersonCode>01101121234</PersonCode>

</Newborn>

* Tiek atrasta pacienta karte pēc saņemtajos datos esošā pacienta alternatīvās identifikācijas (mātes personas kods un dzimšanas datums/laiks).
* Ja ir atrasta viena karte, tad tabulā PatientIdentifications tiek pievienots ieraksts, ar atbilstoša pacienta identifikāciju (lauks PatientIdentificationID), Identification type tiek uzstādīts 1.3.6.1.4.1.38760.3.1.1, Identification code tiek aizpildīts ar lauka PersonCode vērtību.
* Tiek izpildīta meklēšana datu bāze pēc saņemtā personas koda.
* Ja ir atrasta karte ar tādu pašu personas kodu, kas tiek atgriezts no Jaundzimuša reģistra, tiek izpildīta pacientu karšu sapludināšanas procedūra (skat. 6.2.10 mergePatientCards (Sapludināt pacienta kartes)).
* Ja netika izpildīta pacientu karšu sapludināšana, tad pēc veiksmīgas pacienta kartes papildināšanas pacienta ieraksts tiek pievienots pacienta apstrādes fona procesam (skat. 6.13.3 Pacienta kartiņas aizpilde).

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | Error | Array[string] | 1..1 | Saraksts ar procesā notikušajām kļūdām |

Alternatīvie scenāriji

Nevar pieslēgties jaundzimušo reģistram. Tiek atgriezta kļūda un izveidots audita ieraksts.

Netiek saņemta atbilde no jaundzimušo reģistra. Tiek atgriezta kļūda un izveidots audita ieraksts.

## IP audita un sistēmas žurnāla moduļa projektējums

### Auditācijas pieraksti

Sistēmas audits tiek nodrošināts, izmantojot ziņojumu tipiem izstrādātas xslt transformācijas, kuras tiek piesaistītas katram ziņojumam, un nodrošina ziņojumu avota un veicamās operācijas auditāciju IP līmenī.

Prasības transformācijām tiek noteiktas no IP puses, atbilstoši audita apakšsistēmas projektējumam.

Sīkāk auditācijas pierakstu veidošana aprakstīta IP vīzijas dokumentā [2] sadaļā Auditācijas pieraksti.

### writeJournal (Žurnalificēšanas pieraksta veidošana)

Identifikācija

Funkcija: Diagnostic.dll žurnalēšanas procedūra LogUtility.Write

Tiesības: sistēmas funkcija (tiesības netiek pārbaudītas)

Apraksts

Dokumenta sadaļa apraksta žurnalifikācijas funkcijas izmantošanas principus.

Ieejas Parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | Message | String | 1..1 | Žurnalēšanas ziņojums (satur kļūdas/informācijas kodu un aprakstu) |
| 2 | Category | String | 1..1 | Ziņojuma kategorija |
| 3 | Priority | Int | 1..1 | Ziņojuma prioritāte:  public const int Lowest = 0;  public const int Low = 1;  public const int Normal = 2;  public const int High = 3;  public const int Highest = 4; |
| 4 | eventId | Int | 1..1 | Notikuma iekšējais identifikators |
| 5 | severity | TraceEventType | 1..1 | Notikuma svarīgums:  Fatal error or application crash.  Critical = 1,  Recoverable error.  Error = 2,  Noncritical problem.  Warning = 4,  Informational message.  Information = 8,  Debugging trace.  Verbose = 16,  Starting of a logical operation.  Start = 256,  Stopping of a logical operation.  Stop = 512,  Suspension of a logical operation.  Suspend = 1024,  Resumption of a logical operation.  Resume = 2048,  Changing of correlation identity.  Transfer = 4096, |
| 6 | activityId | guid | 1..1 | HL7 pamata ziņojuma identifikators |

Apstrādes algoritms

1. Tiek izsaukta žurnalēšanas bibliotēka ar augstākminētajiem parametriem.

Izejas parametri

Netiek sagaidīts atbildes ziņojums.

Alternatīvie scenāriji

Kļūda auditējot ziņojumu. Tiek atkārtots kļūdas žurnalēšanas mēģinājums, saglabājot datus failā.

## eDoc Service projektējums

### CDA dokumenta izgūšana no eDoc faila

Identifikācija

Funkcija: extractXmlFromEdoc

Tiesības: sistēmas funkcija (tiesības netiek pārbaudītas)

Apraksts

Izmantojot *EDOCLibraries* JAVA bibliotēku tiek validēts *eDoc* fails un veiksmīgas validācijas gadījumā izvērsts un atgriezts CDA dokuments.

Ieejas Parametri

| Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- |
| eDocFileBinary | Baita masīvs | 1..1 | EDoc fails binārajā formātā |

Apstrādes algoritms

Tiek izpildīta šāda darbu secība:

1. Bināram masīvam tiek pārbaudīts garums, un ja tas ir 0, tad tiek atgriezta kļūda.
2. Izmantojot *EDOCBuilder* klases *openEDOC* metodi, tiek atvērta baita masīva plūsma un rezultātā tiek saņemts EDOC objekts.
3. Ar *EDOC.validate* metodes izsaukšanu tiek validēta *EDOC* dokumenta struktūra, metadati, ka arī tiek pārbaudīti visi ciparparaksti.
4. Gadījumā, ja tiek izmests *EDOCException* izņēmums, dokumenta validācija tiek skaitīta par neveiksmīgu un tiek atgriezts kļūdas paziņojums (EVK\_0025 eDoc validācijas kļūda).
5. Pretējā gadījumā tiek izmantots *EDOC.getFiles* lai iegūtu dokumenta failu sarakstu.
6. Ja failu skaits nav vienāds ar 1, tad tiek atgriezta kļūda par nepareizu failu skaitu.
7. Ja saraksts sastāv no viena faila, tiek izvērsts šī faila saturs.
8. Faila saturs tiek validēts uz korektu XML saturu, un gadījumā, ja tam nav korekta XML struktūra, tiek atgriezta kļūda.
9. Faila saturs tiek validēts uz korektu CDA dokumentu, un neveiksmes gadījumā tiek atgriezta kļūda.
10. Pēc visu soļu veiksmīgas izpildīšanas tiek atgriezts CDA dokumenta saturs.

Izejas parametri

| Parametrs | Tips | Skaits | Apraksts |
| --- | --- | --- | --- |
| CDADocument | Teksts | 1..1 | XML CDA dokumenta saturs |

## Administrēšanas modulis

### Pievienot/rediģēt dokumenta veidni

Identifikācija

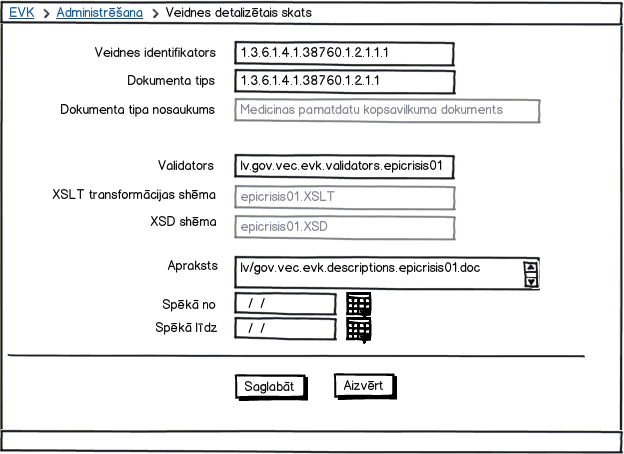
Forma: frmSetTemplate

Tiesības: EvkRghtDocumentTemplateSet

Apraksts

Lietotājam tiek dota iespēja izveidot jauno dokumentu veidni vai rediģēt eksistējošo.

Apstrādes algoritms



68. attēls. frmSetTemplate formas skice

1. Tiek izsaukta funkcija 6.4.2 getDocTemplate (Iegūt dokumenta veidni). Forma tiek aizpildīta ar atgriezta objekta laukiem (vai tiek atvērta tukša, ja lietotājs ir pieprasījis jaunas veidnes reģistrāciju).
2. Lietotājs atver veidnes rediģēšanas formu (izvēloties vienu veidni no saraksta vai pēc pogas „jauns” nospiešanas) un apskata formas saturu.
3. Lietotājs modificē (vai aizpilda pirmoreiz) formu;
4. Lietotājs nospiež „Saglabāt”. Ar laukos ievadītajām vērtībām tiek aizpildīta veidnes datu struktūra (skat. 6.1.8 TemplateDocument (Veidnes struktūra) un tiek izsaukta funkcija 6.4.3 setDocTemplate (Pievienot un papildināt dokumenta veidni). Datu struktūra tiek aizpildīta pēc šāda principa:
   1. Veidnes identifikators – id
   2. Dokumenta tipa kods – code
   3. Dokumenta tipa nosaukums – netiek saglabāts (ir atkarīgs no lauka code)
   4. Validators – validator
   5. Transformācija (XSLT) – XSLTransformation
   6. Shēma (XSD) – Schema
   7. Apraksts – Description
   8. Spēkā no – effectiveTime
   9. Spēkā līdz – availabilityTime

Izejas parametri

Lietotāja forma (nav izejas parametru).

Alternatīvie scenāriji

### Iegūt dokumenta veidņu sarakstu

Identifikācija

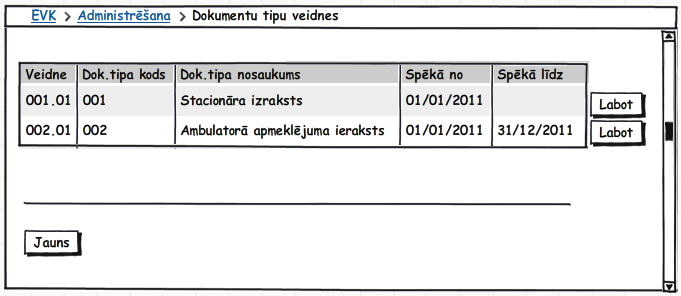
Forma: frmGetTemplates

Tiesības: EvkRghtDocumentTemplateGet

Apraksts

Lietotājam tiek dota iespēja apskatīt dokumentu veidņu sarakstu.

Apstrādes algoritms



69. attēls. frmGetTemplates formas skice

1. Lietotājs aizpilda dokumentu veidņu meklēšanas pieprasījumu un nospiež pogu „Meklēt”. Pieprasījumā var norādīt šādus parametrus:
   1. Dokumenta veidnes statusa kods;
   2. Dokumenta veidnes tipa kods;
   3. Dokumenta veidnes derīguma termiņš.
2. Tiek izsaukta funkcija 6.4.1 getDocumentTemplateList (Iegūt dokumentu veidņu sarakstu).
3. Funkcijas izpildes rezultāts tiek analizēts un uz ekrāna tiek atspoguļots atlasīto veidņu saraksts:
   1. Id (veidnes identifikators);
   2. Code (dokumenta veidnes kods);
   3. Dokumenta veidnes nosaukums (no klasifikatora „Dokumentu veidnes”, atbilstoši laukam Code);
   4. EffectiveTime (dokumenta veidnes reģistrācijas laiks);
   5. AvailabilityTime (dokumenta veidnes darbības laika beigas).
4. Lietotājs iepazīstas ar atlasīto sarakstu.
5. Lietotājs nospiež uz izvēlētas veidnes rindas (vai nospiež pogu „Jauns”) – tiek izsaukta veidnes rediģēšanas forma (skat. 6.12.1 Pievienot/rediģēt dokumenta veidni).

Izejas parametri

Lietotāja forma (nav izejas parametru).

Alternatīvie scenāriji

Nav atrasta neviena dokumentu veidne. Tiek atgriezta tukša forma.

### Iegūt atļauju sarakstu

Identifikācija

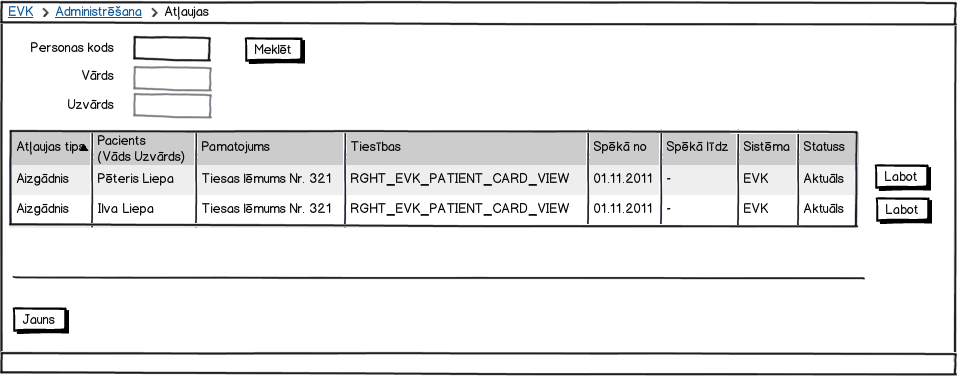
Forma: frmGetDelegations

Tiesības: EvkRghtPatientDelegationsGet

Apraksts

Lietotājam tiek dota iespēja apskatīt personas atļaujas piekļūt pacienta kartēm.

Apstrādes algoritms



70. attēls. frmGetDelegations formas skice

1. Lietotājs izvēlas personas identifikācijas veidu, aizpilda identifikācijas kodu (piem. personas kods) un nospiež pogu „meklēt”.
2. Tiek izsaukta funkcija 6.2.9 getPatientCard (Iegūt pacienta karti);
3. Funkcijas izpildes rezultāts tiek analizēts un uz ekrāna tiek atspoguļots datu struktūras „delegated” saturs.
4. Lietotājs iepazīstas ar izvēlētai personai pieejamām delegācijām.

Alternatīvie scenāriji

Lietotājs neaizpilda visus obligātos laukus; Tiek atgriezta kļūda un izveidots audita ieraksts.

### Pievienot/rediģēt atļaujas

Identifikācija

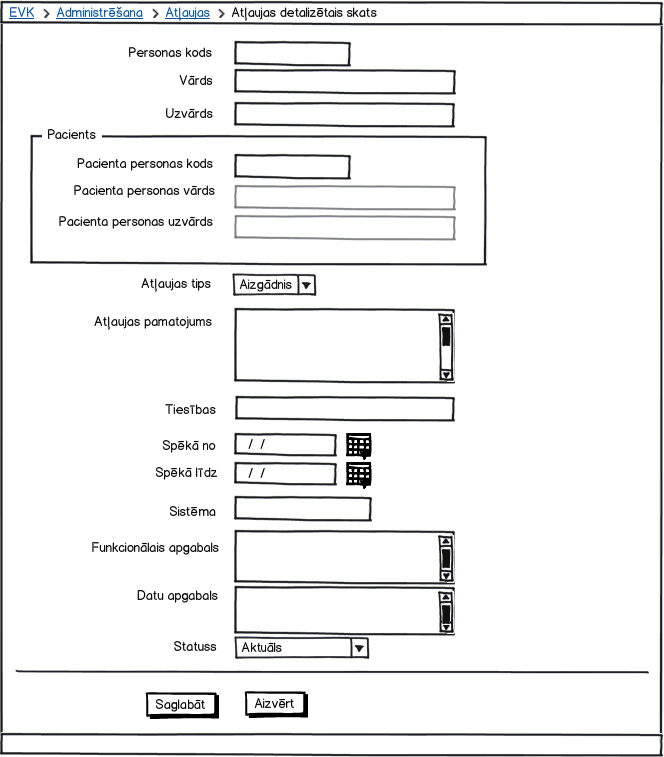
Forma: frmSetDelegations

Tiesības: EvkRghtPatientDelegationSet

Apraksts

Lietotājam tiek dota iespēja uzstādīt personai atļaujas piekļūt pacienta kartēm.

Apstrādes algoritms



71. attēls. frmSetDelegations formas skice

1. Lietotājs izsauc pacienta atļauju meklēšanu (skat. 6.12.3 Iegūt atļauju sarakstu) un apskata rezultātu.
2. Lietotājs izvēlas pievienot sarakstā nepieciešamās papildus atļaujas, uzstādot vērtības šādiem laukiem:
   1. pacienta id;
   2. delegācijas tipu (tiek atļauts uzstādīt „izmeklētāja” vai „aizgādņa” delegācijas);
   3. personu, kam tiek veidota delegācija;
   4. iemeslu;
   5. deleģēto lomu;
   6. delegācijas derīguma laika intervālu (no-līdz). Ja intervāls nav norādīts, delegācija tiek izveidota uz nenoteiktu laiku;
   7. deleģētās sistēmas laukam automātiski tiek uzstādīta vērtība „EVK”
   8. statusa laukam automātiski tiek uzstādīta vērtība „Active”.
3. Ievadītās delegācijas tiek apkopotas pēc pacienta id lauka. Katram pacientam tiek izsaukta funkcija 6.2.11 setPatientCard (Pievienot un papildināt pacienta kartes datus), kurai tiek nodots izmainīts datu komplekts (viena vai vairākas delegācijas).

Alternatīvie scenāriji

Lietotājs neaizpilda visus obligātus laukus. Tiek atgriezta kļūda un izveidots audita ieraksts.

### Uzstādīt personas pilngadības statusu

Identifikācija

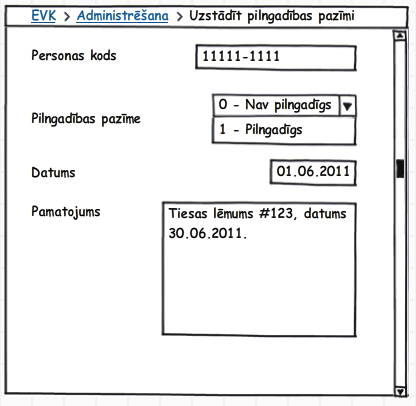
Forma: frmSetMajority

Tiesības: EvkRghtPatientMajoritySet

Apraksts

Lietotājam tiek dota iespēja uzstādīt pacienta pilngadības pazīmi.

Apstrādes algoritms



72. attēls. frmSetMajority formas skice

1. Lietotājs aizpilda personas kodu (vienīgais atļautais identifikācijas veids), uzstāda pilngadības pazīmi un pazīmes uzstādīšanas datumu, ka arī aizpilda iemeslu (visi lauki ir obligāti)
2. Tiek izveidots pieprasījums pacienta kartes datu izmaiņām, atbilstoši datu struktūrai 6.1.2 Person (Pacienta kartes pamatdati). Tiek aizpildīti lauki:
   1. Id
   2. Name
   3. Majority (struktūrai majority tiek uzstādīts updateMode = AR)
3. Tiek izsaukta funkcija 6.2.11 setPatientCard (Pievienot un papildināt pacienta kartes datus).

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | PRPA\_MT201303UV02\_LV01.Person | PRPA\_MT201303UV02\_LV01.Person | 1..1 | Vai lietotājam ir konkrēta tiesība |

Alternatīvie scenāriji

Lietotājs neaizpilda visus obligātus laukus. Tiek atgriezta kļūda un izveidots audita ieraksts.

### Uzstādīt personas aizliegumu

Identifikācija

Forma: frmSetAccessRights

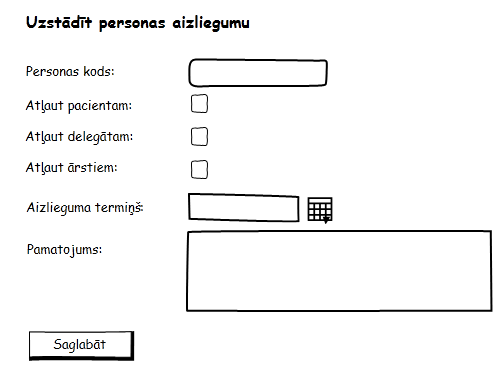
Tiesības: EvkRghtAccessRightsSet

EvkRghtPatientCardAccessRightsSet

Apraksts

Forma nodrošina pacienta kartes tiesību uzstādīšanu no administratora saskarnes.

Apstrādes algoritms



73. attēls. Personas aizliegumu uzstādīšanas formas skice

1. Lietotājs aizpilda personas kodu (vienīgais atļautais identifikācijas veids), uzstāda atļauju līmeni pacientam, delegātiem un ārstiem, ka arī aizpilda iemeslu

Tiek izveidots pieprasījums pacienta kartes pieejas tiesību uzstādīšanai atbilstoši datu struktūrai: 6.1.6.2 ConfidentialityRecord (Aizliegumu struktūra). Tiek aizpildīti lauki:

* 1. confidentialityType = CARD;
  2. confidentialityCode – kombinācija no lauku „Atļaut pacientam”, „Atļaut delegātam” un „Atļaut ārstiem”. Ja ir atzīmēts, tad atbilstoša pozīcija tiek aizpildīta ar 1, citādi – 0;
  3. validTime – lauka „aizlieguma termiņš” vērtība. Ja lauks nav aizpildīts formā, tad validTime vērtība netiek uzstādīta;
  4. confidentialityReason – lauka „Pamatojums” vērtība. Ja lauks nav aizpildīts, tad confidentialityReason vērtība netiek aizpildīta.

1. Tiek izsaukta funkcija 6.2.4 setAccessRights (Mainīt aizliegumus) .

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | RCMR\_MT000002UV02\_LV01.ConfidentialityRecord | RCMR\_MT000002UV02\_LV01.ConfidentialityRecord | 1..1 | Uzstādāmais aizliegumu tiesību komplekts |

Alternatīvie scenāriji

Lietotājs neaizpilda visus obligātos laukus. Tiek atgriezta kļūda un izveidots audita ieraksts.

### GetScreeningConfigurationsList (Iegūt skrīninga konfigurāciju sarakstu)

Identifikācija

Funkcija: GetScreeningConfigurationsList

Tiesības: EvkRghtScreeningConfigurationGet

Apraksts

Iegūt skrīninga konfigurāciju sarakstu vai individuālo skrīninga konfigurācijas ierakstu no datu bāzes.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | GetScreeningConfigurationsList | LVEX\_MT000100UV01.ScreeningConfigurationRequest | 0..1 | Skrīninga konfigurācijas saraksta pieprasījums. Skat. 6.1.19 |

Apstrādes algoritms

Izpildot skrīninga konfigurācijas iegūšanas pieprasījumu, tiek izpildīta šāda darbību secība:

1. Tiek pārbaudītas lietotāja tiesības izsaukt funkciju (skat. 6.5.5 checkUserRight (Pārbaudīt personas tiesību)).
2. No tabulas „ScreeningDefinitions” tiek atlasīta informācija, kas atbilst meklēšanas parametriem.
3. Tiek sagatavota rezultāta datu struktūra, kas aizpildīta ar meklēšanas rezultātā atrasto skrīninga definīciju datiem.
4. Pirms atgriezt rezultātu, tiek izveidots žurnāla ieraksts par veikto darbību.

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | ScreeningConfiguration | LVEX\_MT000100UV01.ScreeningConfiguration | 0..\* | Skrīninga konfigurāciju saraksts. Skat. 6.1.20 |

Alternatīvie scenāriji

Nav identificēti alternatīvie scenāriji. Gadījumā, ja nav atrasti dati, tiek atgriezta tukša atbilde.

### SetScreeningConfiguration (Saglabāt skrīninga konfigurāciju)

Identifikācija

Funkcija: SetScreeningConfiguration

Tiesības: EvkRghtScreeningConfigurationSet

Apraksts

Saglabāt skrīninga konfigurācijas ierakstu datu bāzē.

Ieejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | ScreeningConfiguration | LVEX\_MT000100UV01.ScreeningConfiguration | 0..1 | Skrīninga konfigurāciju saraksts. Skat. 6.1.20 |

Apstrādes algoritms

Izpildot skrīninga konfigurācijas saglabāšanu, tiek izpildīta šāda darbību secība:

1. Tiek pārbaudītas lietotāja tiesības izsaukt funkciju (skat. 6.5.5 checkUserRight (Pārbaudīt personas tiesību)).
2. Atkarībā no ScreeningConfiguration datu struktūrā iekļautā atribūta „UpdateMode”, tiek izpildīta jauna ieraksta pievienošana vai eksistējoša ieraksta rediģēšana.
3. Tabulā ScreeningDefinitions tiek saglabāts jauns skrīninga konfigurācijas ieraksts, kas atbilst saņemtai ScreeningConfiguration datu struktūrai.
4. Tiek sagatavota rezultāta datu struktūra, kas aizpildīta ar saglabātā ieraksta datiem.
5. Pirms atgriezt rezultātu, tiek izveidots žurnāla ieraksts par veikto darbību.

Izejas parametri

| Nr. | Parametrs | Tips | Skaits | Apraksts | |
| --- | --- | --- | --- | --- | --- |
| 1 | ScreeningConfiguration | LVEX\_MT000100UV01.ScreeningConfiguration | 0..1 | Skrīninga konfigurāciju saraksts. Skat. 6.1.20 |

Alternatīvie scenāriji

Nav norādīti pacientu grupas parametri (patientGroup nav pievienota vai nesatur datus) – dati tiks saglabāti skrīninga parametru tabulā, bet netiks pielietoti, veidojot skrīninga ziņojumu.

Nav norādīts neviens no effectiveTime, frequencyMonths vai repetitions – tad sistēma atgriež kļūdu.

Nav atrasta skrininga konfigurācija, kuras parametri tiek atjaunoti.

## Fona uzdevumu projektējums

### Dokumenta apstrādes process

#### SQL Service Broker Services (SBS)

* Sūtītājs: DocumentsValidationRequestService
* Queue: DocumentsValidationRequestQueue
* Saņēmējs: DocumentsValidationService
* Queue: DocumentsValidationQueue

Izmantotais SBS risinājums nodrošina secīgu ienākošo ziņojumu apstrādi, izmantojot standartizētu SBS rindu mehānismu. Servisam „DocumentsValidationRequestService” ir saistītā rinda „DocumentsValidationRequestQueue” un servisam „DocumentsValidationService” ir saistītā rinda „DocumentsValidationQueue”.

Lai nodrošinātu ziņojumu apmaiņu starp šiem servisiem, tiek definēts datu apmaiņas kontrakts „EVK/ DocumentValidationContract” ar vienu ziņojuma tipu [EVK/ Document]. Šis ziņojuma tips sevī ietver CDA dokumentu.

Lai nodrošinātu ziņojuma apstrādi, saņēmēja servisa pusē tiek izmantots Windows serviss „CDADocumentsValidationWinService”. Šis Windows serviss pārbauda saņēmēju, lai pārbaudītu, vai saņēmējs satur ziņojumu, kuru nepieciešams apstrādāt. Atrodot šādu ziņojumu, tiek veiktas darbības (sk. 6.13.1.2Dokumenta apstrādes process”).

#### Dokumenta apstrādes process

Dokumenta apstrāde tiek iniciēta tad, kad starp SBS „DocumentsValidationService” un „DocumentsValidationRequestService” tiek izveidota komunikācija, kurā no sūtītāja „DocumentsValidationRequestService” tiek nodots ziņojums [EVK/ Document] uz saņēmēju „DocumentsValidationService”.

Ziņojuma apstrādei tiek izmantots Windows serviss „CDADocumentsValidationWinService”, kas nodrošina iesūtītā ziņojuma apstrādi atbilstoši funkcijai „6.2.12 processDocument (Pievienot dokumentu –asinhronā daļa)”, kur funkcijai tiek padots ziņojuma saturs. Šajā funkcijā ir aprakstīta CDA dokumenta apstrāde un „Entry” ierakstu apstrādes procesu inicializēšana.

Kad „DocumentsValidationService” pusē tiek pabeigta CDA dokumenta apstrāde, komunikācija starp „DocumentsValidationService” un „DocumentsValidationRequestService” tiek pabeigta, kas nodrošina to, ka ziņojums tiek izņemts no apstrādes procesa rindām.

### Dokumentu ierakstu apstrādes process

#### SQL Service Broker Services (SBS)

* Sūtītājs: TaskRequestService
* Queue: TaskRequestQueue
* Saņēmējs: InternalTaskService
* Queue: InternalTaskQueue
* Saņēmējs: ExternalTaskService
* Queue: ExternalTaskQueue

Datu bāzē tiek izveidoti SBS servisi „TaskRequestService” un divi saņēmējservisi – „InternalTaskService” un „ExternalTaskService”, kur serviss „TaskRequestService” tiks izmantots kā sūtītāja serviss un „InternalTaskService” tiks izmantots kā saņēmēja serviss iekšējo uzdevumu apstrādei un „ExternalTaskService” ārējo ierakstu apstrādei. Izmantotais SBS risinājums nodrošina secīgu ienākošo ziņojumu apstrādi, izmantojot standartizētu SBS rindu mehānismu.

#### Dokumentu ierakstu iekšējās apstrādes process

Ziņojumu apmaiņai starp „TaskRequestService” un „InternalTaskService” tiek definēts datu apmaiņas kontrakts „EVK/InternalTaskContract” ar ziņojuma tipiem „EVK/InternalTask” un „EVK/InternalTaskDone”. Ziņojuma tips „EVK/InternalTask” satur informāciju, kas nepieciešama uzdevuma apstrādei.

Lai nodrošinātu ziņojuma apstrādi, saņēmēja servisa pusē tiek izmantota datu bāzes procedūra „sb\_InternalTaskService”. Šī procedūra pārbauda saņēmēju, lai pārbaudītu vai saņēmējs satur ziņojumu, kuru nepieciešams apstrādāt.

#### Dokumentu ierakstu ārējās apstrādes process

Ziņojumu apmaiņai starp „TaskRequestService” un „ExternalTaskService” tiek definēts datu apmaiņas kontrakts „EVK/ExternalTaskContract” ar ziņojuma tipiem „EVK/InternalTask” un „EVK/ExternalTaskDone”. Ziņojuma tips „EVK/ExternalTask” satur informāciju, kas nepieciešama uzdevuma apstrādei.

Lai nodrošinātu ziņojuma apstrādi, saņēmēja servisa pusē tiek izmantots Windows serviss „ExternalEntryProcessingWinService”. Windows serviss pārbauda saņēmēju, lai pārbaudītu, vai saņēmējs satur ziņojumu, kuru nepieciešams apstrādāt.

### Pacienta kartiņas aizpilde

Identifikācija: processPatientCard

Medicīnisko dokumentu pievienošanas asinhronās apstrādes daļu nodrošina SBS un Windows Services.

#### SQL Service Broker Services (SBS)

* Sūtītājs: ProcessPatientCardRequestService - PatientDataService
* Queue: ProcessPatientCardRequestQueue
* Saņēmējs: ProcessPatientCardService - GetPatientDataService
* Queue: ProcessPatientCardQueue - GetPatientDataQueue

Lai nodrošinātu ziņojumu apmaiņu starp šiem servisiem, tiek definēts datu apmaiņas kontrakts „EVK/PatientDataContract” ar vienu ziņojuma tipu [EVK/PatientData]. Šis ziņojuma tips sevī ietver PersonasGUID un PersonasKodu.

Lai nodrošinātu ziņojuma apstrādi, saņēmēja servisa pusē tiek izmantots Windows serviss „ProcessPatientCardWinService”. Šis Windows serviss pārbauda, vai saņēmējs satur ziņojumu, kuru jāapstrādā. Atrodot šādu ziņojumu, tiek veiktas darbības (sk. 6.13.3.2 Pacienta kartiņas aizpildes apstrādes process).

#### Pacienta kartiņas aizpildes apstrādes process

Dokumenta apstrāde tiek iniciēta tad, kad starp SBS „ProcessPatientCardRequestService” un „ProcessPatientCardService” tiek izveidota komunikācija, kurā tiek nodots ziņojums [EVK/PatientData].

Ziņojuma apstrādei tiek izmantots Windows serviss „ProcessPatientCardWinService”, kas nodrošina iesūtītā ziņojuma apstrādi atbilstoši:

1. Pielasa personas datus no IP, PMLP IR (6.7.1 getPersonData) un aizpilda personas datus pacienta kartiņā;
2. Pielasa VDEĀVK datus no IP, IVIS (6.8.1 getDisabilityData (Iegūt personas invaliditātes datus)) un aizpilda invaliditātes informāciju pacienta kartiņā;
3. Maina pacienta kartiņas statusu NPatients.PersonStatus uz vērtību, kas atbilst statusam „Aktuāls”.

Kad „ProcessPatientCardService” pusē tiek pabeigta pacienta datu apstrāde, komunikācija starp „ProcessPatientCardService” un „ProcessPatientCardRequestService” tiek pabeigta, - ziņojums tiek izņemts no apstrādes procesa rindām.

### Pacienta kartiņu sapludināšana

Identifikācija: mergePatientCard

Medicīnisko dokumentu un pamatdatu apvienošanas asinhronās apstrādes daļu nodrošina SBS Services.

#### SQL Service Broker Services (SBS)

* Sūtītājs: MergePatientCardRequestService
* Queue: MergePatientCardRequestQueue
* Saņēmējs: MergePatientCardService
* Queue: MergePatientCardQueue

#### Apvienot pacienta kartes

Tiek izpildīta šāda darbu secība:

1. Tiek pārbaudīts ieejas ziņojuma formāts.
2. Mērķa pacienta kartei tiek uzstādīts statuss „Inicializācija”.

Katrai avota pacienta kartei tiek izpildīta darbību secība:

* Tiek pārsaistīti pacienta kartes saistītie ieraksti un piesaistītas mērķa pacienta kartei, labojot lauku NPatientID. Izmanto tabulas:

1. PatientIdentifications
2. PatientContacts
3. PatientContactPersons
4. PatientDelegations
5. Documents (visi statusi)
6. PatientNotifications

! Netiek analizēts, vai informācija dublējas.

* Tiek pārsaistīti veselības pamatdatu ieraksti, labojot lauku NPatientID, no tabulām:

1. NPatientAllergies
2. NPatientDiagnosis
3. NPatientMedicalDevices
4. NPatientMedications
5. NPatientWarnings
6. NPatientTransfusions
7. NPatientSurgeries
8. NPatientDisability

Ja veselības pamatdatu ieraksta identificējošā datu kopa mērķa pacienta kartei jau eksistē, tiek labotas sasaistes uz mērķa pacienta kartes atbilstošo veselības pamatdatu ierakstu un avota veselības pamatdatu ieraksts tiek dzēsts, datu izmaiņas fiksējot ēnu tabulās.

* ! Netiek aiztikti avota pacienta kartes dati, kas izvietoti tabulās:

1. PatientFamilyDoctors
2. PatientEhicData
3. PatientProfileSettings

* Avota pacienta kartei tiek uzstādīts statuss „Anulēts”.

Mērķa pacienta kartei tiek uzstādīts statuss „Aktīvs”.

Ja apstrādes laikā tika iegūta kļūda, tad tiek veikta darbību atcelšana (rollback) un kļūda tiek ierakstīta audita žurnālā.

### Personas datu aktualizācija

Identifikācija: updatePersonData

Personas datu aktualizācija, saņemot no DIT datus par personas pamatinformācijas izmaiņām.

#### SQL Service Broker Services (SBS)

* Sūtītājs: ProcessPatientCardRequestService - PatientDataService
* Queue: ProcessPatientCardRequestQueue
* Saņēmējs: ProcessPatientCardService - GetPatientDataService
* Queue: ProcessPatientCardQueue - GetPatientDataQueue

#### Pacienta kartiņas pamatinformācijas izmaiņu apstrādes process

Dati pacienta kartiņas pamatinformācijas aktualizācijai iekļauti grupās, kuru izmērs nav lielāks par sistēmas parametra BatchSize (grupa ProcessCardService) norādīto. Funkcijas izpildes rezultātā visām personām no saraksta tiks atlasīta atjaunināta informācija no PMLP.

Saņemot pieprasījumu par pacienta kartiņas aktualizāciju no PMLP, tiek izpildīta šāda darbību secība:

* Tiek atšifrēts pieprasījumā iekļautais pacienta kartiņas datu veids. Atšifrēšanai izmanto līdzvērtības tabulu, kas tiek glabāta tabulā Parameters, ar datu veidu laukā „Name” un tam atbilstošu kodu – laukā Value. Vērtības tiek glabātas grupā „ProcessCardService”
* Atbilstoši datu pieprasījuma veidam tiek izsaukta viena no LVOR sadaļas tīmekļa pakalpēm, kas nodrošina PMLP generic servisu izsaukumu:
  + GetNextPersonCodeList – Izsaukums generic servisam datu kopas „QMP\_ARI\_IRA\_PKD\_NAKAMAIS” saņemšanai
  + GetPersonBirthDateList - Izsaukums generic servisam datu kopas „QMP\_ARI\_IRA\_DZIM\_DAT” saņemšanai
  + GetPersonDeathOrAbsenceList - Izsaukums generic servisam datu kopas „QMP\_ARI\_IRA\_MIRS\_BEZV\_DAT” saņemšanai
  + GetPreviousPersonCodeList - Izsaukums generic servisam datu kopas „QMP\_ARI\_IRA\_PKD\_IEPR” saņemšanai
  + GetPersonIRStateList - Izsaukums generic servisam datu kopas „QMP\_ARI\_IRA\_STATUSS\_IR” saņemšanai
  + GetPersonLegalCapacityList - Izsaukums generic servisam datu kopas „QMP\_ARI\_IRA\_STATUSS\_RICIBSP” saņemšanai
  + GetParentPersonCodeList - Izsaukums generic servisam datu kopas „QMP\_ARI\_IRA\_VEC\_PKD” saņemšanai
  + GetPersonGenderList - Izsaukums generic servisam datu kopas „QMP\_ARI\_IRA\_DZIMUMS” saņemšanai
  + GetPersonAddressDecriptedList - Izsaukums generic servisam datu kopas „QMP\_ARI\_IRA\_PAMATADR\_IEROBA” saņemšanai
* Tiek analizēta no generic servisa saņemta atbilde, atkarībā no tās uzstādot rindas tabulas (PatientMonitoring) stāvokļa vērtības:
  + NEW – ja ziņojums nav vēl sūtīts PMLP datu pieprasījumam
  + ERROR – ja ziņojums atgriež kļūdu no PMLP (piem. personas kods neeksistē); Šāds ieraksts vairs netiks apstrādāts (nebūs veikti mēģinājumi iegūt datus no PMLP).
  + FAILED – ja ziņojums atgriež tehnisko kļūdu (piem. TimeOut pieprasījuma laikā); Ziņojumam tiek palielināta mēģinājumu skaitītāja vērtība. Kad skaitītājs sasniegs sistēmas parametros norādīto limitu (parametrs BatchMaxRetryCount)
  + IN\_QUEUE – tiek uzstādīts, kamēr ziņojums tiek apstrādāts;
  + SUCCESS – tiek uzstādīts, ja dati veiksmīgi saņemti no PMLP; Ieraksti ar šādu statusu netiek vairāk apstrādāti.
  + NOT\_SENT – tiek uzstādīts, ja pieprasījums nav jāizpilda (piemēram, ja pilngadīgai personai mainās vecāku ieraksts PMLP); Ieraksti ar šādu statusu netiek vairāk apstrādāti.

### Jaundzimušo personas kodu aktualizācija

Identifikācija: updateNewBornPersonCodes

WS: getNewBornPersonCodes

Funkcija: 6.9.1 setNewBornPersonCodes (Iegūt jaundzimušo personas kodus)

Fona procesa izpildīšanas biežums – vienreiz dienā.

### Personas pilngadības aktualizācija

Identifikācija: updatePersonMajority

Funkcija: 6.2.16 setPersonMajority (Uzstādīt personas pilngadības pazīmi)

Fona procesa izpildīšanas biežums – vienreiz dienā.

### sb\_BackgroundScreeningRunStart (Skrīninga fona process)

Identifikācija

Funkcija: sb\_BackgroundScreeningRunStart

Tiesības: fona process, tiek palaists ar SQLServer lietotāja tiesībām

Apraksts

Funkcija tiek palaista pēc regulāriem intervāliem ar SQL Agent Job palīdzību un nodrošina skrīninga fona procesa izpildi.

Ieejas parametri

Fona procesa funkcijai nav ieejas parametru. Fona procesu konfigurācija tiek atlasīta no db tabulām.

Apstrādes algoritms

1. Tiek saskaitīts katram no skrīninga definīciju ierakstiem, cik reizes tas jau tika palaists, un kad notika konkrētās skrīninga definīcijas pēdējā palaišana.
2. Tiek pārbaudīts, kādām skrīninga definīcijām ir nepieciešama izpilde. Tas tiek darīts, atlasot no skrīningu definīcijas tabulas (ScreeningDefinitions) visas reģistrētās aktīvās (StatusCode = Active) definīcijas un turpmāk filtrējot tās pēc laika perioda un atkārtojumu skaita (Mēnešu skaits no pēdējās palaišanas > Skrīninga definīcijā ierakstīts cipars un kopējais atkārtojumu skaits < Skrīninga definīcijā ierakstīts atkārtojumu skaits).
3. Tām definīcijām, kurām nepieciešama palaišana, tiek izpildīta skrīninga ziņojumu veidošanas procedūra sb\_ScreeningRunAll (visas skrīninga procesa palaišanas tiek pievienotas rindā BackgroundRequestQueue ar ziņojuma tipu EVK/Background/ScreeningRun).
4. Apstrādājot šo ziņojumu tipu, Background rindā tiek izpildīta pacientu grupas iegūšana, kas atbilst norādītajiem skrīninga parametriem (izveidojot un izpildot pieprasījumu, kas atbilst skrīninga definīcijas norādītajiem parametriem – funkcija usp\_GetPatientsForScreening).
5. Katram pacientam no izvēlētās grupas tiek izveidots ziņojums (saņemot ziņojuma vērtību no skrīninga definīcijas un aizpildot pacienta detaļas – vārdu %param\_GivenName%, uzvārdu %param\_FamilyName% un personas kodu %param\_IdentificationCode%). Transformēts ziņojums tiek saglabāts tabulā Notifications.

Izejas parametri

Fona process neatgriež informāciju. Stāvoklis tiek paziņots mainot fona procesa konfigurācijas tabulas un datu bāzes žurnāla tabulu DataBaseActivity.

### ProcessNotificationsSendMessage (Notifikāciju sūtīšana)

Identifikācija

Funkcija: ProcessNotificationsSendMessage

Tiesības: fona process

Apraksts

Nosūta pacientam ziņojumu, izmantojot IP notifikācijas servisu.

Ieejas parametri

Fona procesa funkcijai nav ieejas parametru. Nepieciešamā informācija tiek atlasīta no db tabulām.

Apstrādes algoritms

Izpildot notifikāciju sūtīšanu, tiek izpildīta šāda darbību secība:

1. Atbilstoši saņemtajam ziņojuma identifikatoram, no tabulas NotificationRequests tiek atlasīti visi nenosūtītie ziņojumi.
2. Tiek atšifrēts ziņojuma teksts (NotificationText) un iegūts pacienta identifikators (NPatientID), un atlasīta kontaktinformācija un ziņojumu sūtīšanas konfigurācija.
3. Tiek pārbaudīta sistēmas parametra „FilterNotificationsOnPreferences” vērtība. Ja parametrs ir „1”, tad ziņojums tiks izsūtīts, tikai ja pacienta preferencēs (tabulā PatientNotifications) ir ieraksts konkrētam notifikācijas veidam (lauks NotificationCode).
4. Gadījumā, ja ziņojums tiek sūtīts ar atzīmi par nosūtījuma veidošanas nepieciešamību (tabulas Notifications.SendChannel lauka vērtība ir PNIS), tiek iegūts sistēmas parametrs Nosūtītāja darba vietai un izveidots nosūtījums, izsaucot funkciju ReferralSaveAsync, norādot pakalpojuma veidu no skrīninga definīcijas (PNIS ziņojumi var rasties tikai kā skrīninga izpildes rezultāts).
5. Tiek izsaukta IP funkcija SendMessage.

Izejas parametri

Fona process neatgriež informāciju. Stāvoklis tiek paziņots, mainot fona procesa konfigurācijas tabulas un datu bāzes žurnāla tabulu DataBaseActivity.

### \_ (EVK notifikācijas)

Identifikācija

Funkcija: \_

Tiesības: fona process

Apraksts

Apraksts apkopo visus gadījumus, kad no EVK IS tiek izsūtīti paziņojumi pacientam.

Ieejas parametri

Fona procesa funkcijai nav ieejas parametru. Nepieciešamā informācija tiek atlasīta no db tabulām.

Apstrādes algoritms

1. No EVK sistēmas izsūtāmās notifikācijas:
2. Paziņojums par nokavētu vakcināciju;
3. Skat. aprakstu dokumentā [7] sadaļa 4.7.6.8 (Automātiskā atgādinājumu par nenotikušām vakcinācijām nosūtīšana);
4. Paziņojums par nepieciešamo vakcināciju;
5. Paziņojums tiek sūtīts atbilstoši pacienta ieplānotām vakcinācijām. Skat. aprakstu dokumentā [7] sadaļa 4.7.6.7 (Automātiskā vakcināciju atgādinājumu nosūtīšana);
6. Paziņojums par jaunu izveidotu medicīnisko dokumentu.
7. Paziņojums tiek sūtīts, kad pacienta kartē tiek piereģistrēts jauns medicīniskais dokuments un tā statuss ir ‘Actual’. Paziņojums satur informāciju par dokumenta pievienošanas laiku, dokumenta veidu, autoru un numuru.
8. Paziņojums par EVAK kartes derīguma termiņa beigām.
9. Paziņojums par EVAK kartes derīguma termiņa beigām tiek nosūtīts pacientam, ja tabulā PatientEhicData lauka ValidTill vērtība sakrīt ar šodienas datumu. Paziņojums satur informāciju par EVAK numuru (EhicNumber), izsniegšanas datumu (DateOfIssue) un derīguma datumiem no-līdz (ValidFrom; ValidTill).
10. Paziņojums par jaunu piereģistrētu EVAK karti.
11. Paziņojums par jauno EVAK karti tiek nosūtīts pacientam, ja tabulā PatientEhicData tiek piereģistrēts jauns ieraksts (SetPatientEhicData izsaukuma rezultātā). Paziņojums satur informāciju par EVAK numuru (EhicNumber), izsniegšanas datumu (DateOfIssue) un derīguma datumiem no-līdz (ValidFrom; ValidTill)
12. Paziņojums, ka pacientam mainījies ģimenes ārsts.
13. Paziņojums tiek sūtīts pacientam, ja tabulā PatientFamilyDoctors tiek pievienots jauns ieraksts (SetPatientFamilyDoctor). Paziņojums satur informāciju par jaunā ģimenes ārsta vārdu/uzvārdu (informācija no klasifikatora 2.1), organizāciju (no klasifikatora 2.23), specialitāti (no klasifikatora 2.38) un ģimenes ārsta maiņas datumu.
14. Paziņojums par iekļaušanu skrīninga programmā.
15. Paziņojums tiek sūtīts pacientam, ja viņš ir izvēlēts skrīninga programmā kā daļa no skrīninga apstrādes procesa (skat. 6.13.8).
16. Paziņojums par jaunu EVK ietvaros izveidoto audita ierakstu.
17. Paziņojums tiek sūtīts pacientam, kad tiek veidots jauns EVK audita ieraksts. Paziņojums satur informāciju par Avota sistēmu (lauks SourceSystemName), Speciālistu, kas apskatījis datus (AuthorOrPerformerId, AuthorOrPerformerNameGiven, AuthorOrPerformerNameFamily) un audita aprakstu (AuditDescription).
18. Paziņojums, ka sistēmā pēdējais reģistrētais CDA dokuments bija pirms vairāk nekā 3 gadiem.
19. Paziņojumu sūtīšanai izmantojams atsevišķs SQLAgentJob, kas tiek palaists 1 reizi pusgadā un atlasa no dokumentu tabulas (EVK.Documents) visus pacientus, kam nav atzīmēta miršanas pazīme, un pēdējais dokuments jebkurā stāvoklī ir atrodams ne agrāk par sistēmas konfigurācijā norādīto momentu (laiks tiek skaitīts, izmantojot dokumenta lauku EffectiveTime, nevis dokumenta ieraksta reģistrācijas momentu). Paziņojumā tiek iekļauts standarta teksts par nepieciešamību apmeklēt savu ārstu.

Izejas parametri

Fona process neatgriež informāciju. Stāvoklis tiek paziņots mainot fona procesa konfigurācijas tabulas un datu bāzes žurnāla tabulu DataBaseActivity.

### GetMedPersonProfiles (Ārstniecības personu profilu datu atlase)

Identifikācija

Funkcija: GetMedPersonProfiles

Tiesības: fona process

Apraksts

Savāc no IP medicīnas personu profilu datus un saglabā tos EVK IS.

Ieejas parametri

Fona procesa funkcijai nav ieejas parametru. Nepieciešamā informācija tiek atlasīta no db tabulām.

Apstrādes algoritms

1. Palaižot procesu tiek izpildīts šāds algoritms:
2. Atlasa 2.37 (Ārstniecības personu un ārstniecības atbalsta personu darbavietas) klasifikatora pēdējo versiju un apvieno to ar profila datu keštabulu (ConsultationsPersonProfile) ;
3. Sašķiro ierakstus pēc pievienošanas datuma (vecākie ieraksti augšā) un atlasa no konfigurācijas parametra atlasīto skaitu.
4. Katram ierakstam no saraksta izsauc IP servisu GetProfileQuery un ieraksta saņemtus parametrus eConsultation un eConsultationToAll un saglabā tos atbilstošas kolonnās tabulā ConsultationPersonProfile.
5. Gadījumā, ja tādai personai nav profila vai tā nav atrasta IP, tad šo lauku vērtība ir 0 (tas ir AP nesniedz konsultācijas)

Izejas parametri

Fona process neatgriež informāciju. Stāvoklis tiek paziņots, mainot fona procesa konfigurācijas tabulas un datu bāzes žurnāla tabulu DataBaseActivity.

### BackgroundLogExport (Eksportēt kļūdu informāciju)

Identifikācija

Funkcija: BackgroundLogExport

Tiesības: fona process

Apraksts

Eksportē EVK reģistrētas kļūdas.

Ieejas parametri

Fona procesa funkcijai nav ieejas parametru. Nepieciešamā informācija tiek atlasīta no db tabulām.

Apstrādes algoritms

1. Palaižot procesu tiek izpildīts šāds algoritms:
2. No konsoles konfigurācijas tiek atlasīts servera direktorijas nosaukums, kurā tiks veikts kļūdu žurnāla eksports.
3. No tabulas EVLPers.ErrorExport tiek atlasīts laiks, līdz kuram tika veikts pēdējais datu eksports.
4. No tabulas EVLPers.ErrorLog tiek atlasīti visi ieraksti, kam ienākšanas datums ir lielāks par jau eksportētiem.
5. Ieraksti tiek atgriezti izsaucējam xml formātā.
6. Saņemtais xml tiek eksportēts kā fails, atbilstošā direktorijā norādot tam pašreizējo datumu
7. EVLPers.ErrorExport tiek pievienota rinda par veiksmīgo eksportu, norādot ierakstu identifikatorus.

Izejas parametri

Fona process neatgriež informāciju..

## Kļūdas kodu tabula

6.14‑1. tabula. Kļūdas kodu tabula

| Nr. | Kļūdas kods | Nosaukums |
| --- | --- | --- |
| 1 | EVK\_0001 | Nav atrasta vai nav pieejama pacienta karte |
| 2 | EVK\_0008 | Dokuments nav atrasts vai nav pieejams |
| 3 | EVK\_0011 | Karte ar norādīto personas identifikāciju jau eksistē |
| 4 | EVK\_0016 | Nav atļauts veidot pilnvaras pašam sev |
| 5 | EVK\_0018 | Nav tiesības uzstādīt aizliegumu (pacienta karte, dokuments, veselības pamatdati) |
| 6 | EVK\_0019 | Objekta tiesību apraksts neatbilst formātam |
| 7 | EVK\_0025 | eDoc validācijas kļūda |
| 8 | EVK\_0029 | Nav tiesības veikt darbību <RGHT kods> |
| 9 | EVK\_0032 | Dokumenta veidne nav atrasta |
| 10 | EVK\_0033 | Dokumenta veidne jau izveidota |
| 11 | EVK\_0034 | Lauku vērtības CDA XML un ClinicalDocument aploksnē nav vienādas |
| 12 | EVK\_0035 | Nav atrasts aktuāls dokumenta šablons atbilstoši iesniedzamā dokumenta elementam templateid |
| 13 | EVK\_0036 | Kļūda mēģinot atkodēt Base64 tekstu |
| 14 | EVK\_0037 | Veidnes versija (templateId) neatbilst dokumenta tipam |
| 15 | EVK\_0038 | Dokumenta versija nav norādīta korekti |
| 16 | EVK\_0039 | Personas kods nav korekts |
| 17 | EVK\_0042 | Dokumenta versiju nav iespējams anulēt, jo tā jau ir anulēta |
| 18 | EVK\_0044 | Nav aizpildīts obligāts CDA dokumenta elements |
| 19 | EVK\_0047 | Pacienta identifikatora formāts neatbilst identifikācijas tipam. |
| 20 | EVK\_0049 | Parametri norādīti nekorekti |
| 21 | EVK\_0053 | Nekorekta dokumenta identifikācija |
| 22 | EVK\_0054 | Norādītā persona neatbilst pieprasījumam |
| 23 | EVK\_0055 | Norādīts neatbilstošs pacienta identifikācijas tips. |
| 24 | EVK\_0056 | Norādītais objekts neeksistē |
| 25 | EVK\_0057 | Pacients nav satusā „Actual” |
| 26 | EVK\_0083 | Nav iespējams turpināt darbību, dokuments vēl ir apstrādē |

## Datu detalizēts projektējums

### Datubāze: EVKPers

PatientContactPersons

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ContactPersonID | int | 4 | No |  |
| PatientGUID | uniqueidentifier | 16 | No |  |
| ContactPersonTypeCodeSystem | varchar | 50 | No |  |
| ContactPersonTypeCode | nvarchar | 100 | No |  |
| ContactPersonTypeDisplayValue | nvarchar | 100 | yes |  |
| GivenName | nvarchar | 68 | yes | // Nullable, veidojot tukšu kartiņu. Vai def. |
| FamilyName | nvarchar | 68 | yes | // Nullable, veidojot tukšu kartiņu. Vai def. |
| Value | nvarchar | 100 | yes | Kontaktinformācija (telecom). |
| Status | varchar | 15 | yes |  |
| Author | xml | -1 | yes |  |
| ContactType | nvarchar | 100 | yes |  |

PatientContactPersonsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| ContactPersonID | int | 4 | No |  |
| PatientGUID | uniqueidentifier | 16 | No |  |
| ContactPersonTypeCodeSystem | varchar | 50 | No |  |
| ContactPersonTypeCode | nvarchar | 100 | No |  |
| ContactPersonTypeDisplayValue | nvarchar | 100 | yes |  |
| GivenName | nvarchar | 68 | yes | // Nullable, veidojot tukšu kartiņu. Vai def. |
| FamilyName | nvarchar | 68 | yes | // Nullable, veidojot tukšu kartiņu. Vai def. |
| Value | nvarchar | 100 | yes |  |
| Status | varchar | 15 | yes |  |
| Author | xml | -1 | yes |  |
| ContactType | nvarchar | 100 | yes |  |

PatientContacts

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientContactID | int | 4 | No |  |
| PatientGUID | uniqueidentifier | 16 | yes |  |
| RecordStatus | int | 4 | yes | Aktuāls, neaktuāls |
| ContactTypeCodeSystem | varchar | 50 | yes |  |
| ContactTypeCode | varchar | 50 | yes |  |
| ContactTypeDisplayValue | nvarchar | 500 | yes |  |
| Value | nvarchar | 100 | yes | Contactinformation. |
| Author | xml | -1 | yes |  |
| UseForNotifications | bit |  | No | Pazīme, kas norāda, ka kontaktinformācija izmantojama par komunikācijas kanālu paziņojumu sūtīšanai. |

PatientContactsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientsContactsModifID | int | 4 | No |  |
| PatientContactID | int | 4 | No |  |
| PatientGUID | uniqueidentifier | 16 | yes |  |
| RecordStatus | int | 4 | yes | Aktuāls, neaktuāls |
| ContactTypeCodeSystem | varchar | 50 | yes |  |
| ContactTypeCode | varchar | 50 | yes |  |
| ContactTypeDisplayValue | nvarchar | 500 | yes |  |
| Value | nvarchar | 100 | yes |  |
| Author | xml | -1 | yes |  |
| UseForNotifications | bit |  | No |  |

PatientDelegations

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| DelegationID | int | 4 | No | Delegation identification |
| PatientGUID | uniqueidentifier | 16 | yes | Patient whose delegation is stored |
| DelegationType | varchar | 50 | yes | Delegation type (delegation to person or delegation to other entities) |
| GivenName | nvarchar | 68 | yes | Delegate given name |
| FamilyName | nvarchar | 68 | yes | Delegate family name |
| IdentificationType | varchar | 50 | yes | Identification type |
| IdentificationCode | varchar | 50 | yes | Identification code (person code or alternative identification - depends on IdentificationType field value) |
| DelegationReason | nvarchar | 1000 | yes | Reason for delegation (must be entered when delegated to inspector or policeman) |
| DelegatedRight | nvarchar | 200 | yes | Delegated right (uninterpreted charachter string) |
| DelegatedTimeFrom | datetime | 8 | yes | Delegation time start |
| DelegatedTimeTo | datetime | 8 | yes | Delegation time end |
| ViewableDelegation | bit | 1 | No | Is delegation and its audit trail viewable for patient or no |
| DelegatedFunctionality | xml | -1 | yes | Optional; delegated functionality if right alone is not enough |
| DelegatedData | xml | -1 | yes | Optional; delegated data if right alone is not enough |
| DelegatedSystem | nvarchar | 100 | yes | Delegation limited to one of core systems. (System name here) |
| Status | varchar | 15 | yes | Delegation statuss ACTIVE DELETED |
| Author | xml | -1 | yes |  |
| ModifDate | datetime | 8 | yes |  |
| ModifUser | nchar | 20 | yes |  |

PatientDelegationsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| DelegationID | int | 4 | No | Delegation identification |
| PatientGUID | uniqueidentifier | 16 | yes | Patient whose delegation is stored |
| DelegationType | varchar | 50 | yes | Delegation type (delegation to person or delegation to other entities) |
| GivenName | nvarchar | 68 | yes | Delegate given name |
| FamilyName | nvarchar | 68 | yes | Delegate family name |
| IdentificationType | varchar | 50 | yes | Identification type |
| IdentificationCode | varchar | 50 | yes | Identification code (person code or alternative identification - depends on IdentificationType field value) |
| DelegationReason | nvarchar | 1000 | yes | Reason for delegation (must be entered when delegated to inspector or policeman) |
| DelegatedRight | nvarchar | 200 | yes | Delegated right (uninterpreted charachter string) |
| DelegatedTimeFrom | datetime | 8 | yes | Delegation time start |
| DelegatedTimeTo | datetime | 8 | yes | Delegation time end |
| ViewableDelegation | bit | 1 | No | Is delegation and its audit trail viewable for patient or no |
| DelegatedFunctionality | xml | -1 | yes | Optional; delegated functionality if right alone is not enough |
| DelegatedData | xml | -1 | yes | Optional; delegated data if right alone is not enough |
| DelegatedSystem | nvarchar | 100 | yes | Delegation limited to one of core systems. (System name here) |
| Status | varchar | 15 | yes | Delegation statuss ACTIVE DELETED |
| Author | xml | -1 | yes |  |
| ModifDate | datetime | 8 | yes |  |
| ModifUser | nchar | 20 | yes |  |

PatientEhicData

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientsEHICID | int | 4 | No | EHIC = EVAK |
| PatientGUID | uniqueidentifier | 16 | yes |  |
| EhicNumber | nchar | 20 | yes |  |
| DateOfIssue | date | 3 | yes |  |
| IssuingInstitution | nvarchar | 100 | yes |  |
| ValidFrom | date | 3 | yes |  |
| ValidTill | date | 3 | yes |  |
| Status | char | 20 | yes |  |

PatientEhicDataModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientsEHICID | int | 4 | No | EHIC = EVAK |
| PatientGUID | uniqueidentifier | 16 | yes |  |
| EhicNumber | nchar | 20 | yes |  |
| DateOfIssue | date | 3 | yes |  |
| IssuingInstitution | nvarchar | 100 | yes |  |
| ValidFrom | date | 3 | yes |  |
| ValidTill | date | 3 | yes |  |
| Status | char | 20 | yes |  |

PatientFamilyDoctors

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientFamilyDoctorID | int | 4 | No |  |
| PatientGUID | uniqueidentifier | 16 | yes | Jābūt UNIKĀLAM |
| PhysicianId | varchar | 50 | yes |  |
| InstitutionId | varchar | 50 | yes |  |
| SpecialityId | varchar | 50 | yes |  |
| DoctorTimeFrom | datetime | 8 | yes |  |
| DoctorTimeTo | datetime | 8 | yes |  |
| Author | xml | -1 | yes |  |
| ModifDate | datetime | 8 | yes |  |
| ModifUser | nchar | 20 | yes |  |

PatientFamilyDoctorsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientFamilyDoctorID | int | 4 | No |  |
| PatientGUID | uniqueidentifier | 16 | yes | Jābūt UNIKĀLAM |
| PhysicianId | varchar | 50 | yes |  |
| InstitutionId | varchar | 50 | yes |  |
| SpecialityId | varchar | 50 | yes |  |
| DoctorTimeFrom | datetime | 8 | yes |  |
| DoctorTimeTo | datetime | 8 | yes |  |
| Author | xml | -1 | yes |  |
| ModifDate | datetime | 8 | yes |  |
| ModifUser | nchar | 20 | yes |  |

PatientIdentifications

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientIdentificationID | int | 4 | No | Person identification. |
| IdentificationType | varchar | 50 | yes |  |
| IdentificationCode | varchar | 50 | yes |  |
| PatientGUID | uniqueidentifier | 16 | No |  |
| PrimaryIdentificaton | bit | 1 | No |  |
| ActivityGUID | uniqueidentifier |  | yes | Sesijas identifikators |
| ModifDate | datetime |  | No | Pēdējais labošanas datums un laiks |
| Del | bit |  | No | Dzēšanas pazīme (trigerēts lauks) |
| MonitoringOnDate | datetime | 8 | yes |  |
| MonitoringOffDate | datetime | 8 | yes |  |

PatientIdentificationsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | Int | 4 | No |  |
| PatientIdentificationID | int | 4 | No | Person identification. |
| IdentificationType | varchar | 50 | yes |  |
| IdentificationCode | varchar | 50 | yes |  |
| PatientGUID | uniqueidentifier | 16 | No |  |
| PrimaryIdentificaton | bit | 1 | No |  |
| ActivityGUID | uniqueidentifier |  | yes | Sesijas identifikators |
| ModifDate | datetime |  | No | Pēdējais labošanas datums un laiks |
| Del | bit |  | No | Dzēšanas pazīme (trigerēts lauks) |
| MonitoringOnDate | datetime | 8 | yes |  |
| MonitoringOffDate | datetime | 8 | yes |  |

PatientNotifications

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientNotificationID | int |  | No | Primārā atslēga |
| PatientGUID | Uniqueidentifier |  | No | Ārējā atslēga uz personas kartiņu |
| NotificationCode | Varchar(50) |  | Yes | Paziņojumu veida kods |
| NotificationCodeSystem | Varchar(50) |  | Yes | Paziņojumu veidu klasifikatora OID |
| NotificationDisplayValue | Nvarchar(500) |  | Yes | Paziņojuma veida nosaukums |
| ActivityGUID | Uniqueidentifier |  | Yes | Sesijas identifikators |
| ModifDate | Datetime |  | No | Pēdējais labošanas datums un laiks |
| Del | Bit |  | No | Dzēšanas pazīme (trigerēts lauks) |

PatientNotificationsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | Int | 4 | No |  |
| PatientNotificationID | int |  | No | Primārā atslēga |
| PatientGUID | Uniqueidentifier |  | No | Ārējā atslēga uz personas kartiņu |
| NotificationCode | Varchar(50) |  | Yes | Paziņojumu veida kods |
| NotificationCodeSystem | Varchar(50) |  | Yes | Paziņojumu veidu klasifikatora OID |
| NotificationDisplayValue | Nvarchar(500) |  | Yes | Paziņojuma veida nosaukums |
| ActivityGUID | Uniqueidentifier |  | Yes | Sesijas identifikators |
| ModifDate | Datetime |  | No | Pēdējais labošanas datums un laiks |
| Del | Bit |  | No | Dzēšanas pazīme (trigerēts lauks) |

PatientProfileSettings

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ProfileSettingID | int | 4 | No | Lai saglabātu portāla iestatījumus lietotājam |
| PatientGUID | uniqueidentifier | 16 | yes |  |
| Name | nvarchar | 100 | yes |  |
| Value | xml | -1 | yes |  |
| Author | xml | -1 | yes |  |
| ModifDate | datetime | 8 | yes |  |
| ModifUser | nchar | 20 | yes |  |
| System | nchar | 20 | yes |  |

PatientProfileSettingsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| ProfileSettingID | int | 4 | No | Lai saglabātu portāla iestatījumus lietotājam |
| PatientGUID | uniqueidentifier | 16 | yes |  |
| Name | nvarchar | 100 | yes |  |
| Value | xml | -1 | yes |  |
| Author | xml | -1 | yes |  |
| ModifDate | datetime | 8 | yes |  |
| ModifUser | nchar | 20 | yes |  |
| System | nchar | 20 | yes |  |

Patients

| Column name | Type | Length | Nullable | Description |
| --- | --- | --- | --- | --- |
| PatientGUID | uniqueidentifier | 16 | No |  |
| GivenName | nvarchar | 68 | yes | // Nullable, veidojot tukšu kartiņu. Vai def. |
| FamilyName | nvarchar | 68 | yes | // Nullable, veidojot tukšu kartiņu. Vai def. |
| BirthTime | date | 3 | yes | // Nullable, veidojot tukšu kartiņu. Vai def. |
| DateOfDeath | date | 3 | yes |  |
| NPatientID | varbinary | 900 | yes | Šifrēta saite (Non-personified identifier) |
| Village | nvarchar | 40 | yes | Ciems |
| Street | nvarchar | 40 | yes | Iela |
| HouseName | nvarchar | 200 | yes | Ekasnosaukums/ |
| HouseNumber | nvarchar | 38 | yes | EkasNumurs/ |
| HouseBlockNumber | nvarchar | 12 | yes | KorpusaNumurs/ |
| FlatNumber | nvarchar | 38 | yes | Dzīvokļa numurs/ |
| PostalCode | nchar | 18 | yes | PastaIndekss/ |
| AddressAbroad | nvarchar | 600 | yes | AdreseArvalstis/ |
| MajorityDate | datetime | 8 | yes |  |
| MajorityReason | nvarchar | 100 | yes |  |
| Author | xml | -1 | yes |  |
| ModifDate | datetime | 8 | yes |  |
| ModifUser | nvarchar | 100 | yes |  |
| ChangeVersion | int | 4 | yes |  |

PatientsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientGUID | uniqueidentifier | 16 | No |  |
| GivenName | nvarchar | 68 | yes | // Nullable, veidojot tukšu kartiņu. Vai def. |
| FamilyName | nvarchar | 68 | yes | // Nullable, veidojot tukšu kartiņu. Vai def. |
| BirthTime | date | 3 | yes | // Nullable, veidojot tukšu kartiņu. Vai def. |
| DateOfDeath | date | 3 | yes |  |
| NPatientID | varbinary | 900 | yes | Šifrēta saite (Non-personified identifier) |
| Village | nvarchar | 40 | yes | Ciems/ |
| Street | nvarchar | 40 | yes | Iela/ |
| HouseName | nvarchar | 200 | yes | Ekasnosaukums/ |
| HouseNumber | nvarchar | 38 | yes | EkasNumurs/ |
| HouseBlockNumber | nvarchar | 12 | yes | KorpusaNumurs/ |
| FlatNumber | nvarchar | 38 | yes | Dzīvokļa numurs/ |
| PostalCode | nchar | 18 | yes | PastaIndekss/ |
| AddressAbroad | nvarchar | 600 | yes | AdreseArvalstis/ |
| MajorityDate | datetime | 8 | yes |  |
| MajorityReason | nvarchar | 100 | yes |  |
| Author | xml | -1 | yes |  |
| ModifDate | datetime | 8 | yes |  |
| ModifUser | nvarchar | 100 | yes |  |
| ChangeVersion | int | 4 | yes |  |

UpdateList

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientGUID | Uniqueidentifier |  | No | Primārā atslēga |
| ActualizationTime | Datetime |  | No | Pēdējais aktualizācijas datums un laiks |

ErrorExport

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ErrorExportID | int | No | Checked | Primārā atslēga |
| ExportTime | datetime | No | Unchecked | Eksporta laiks |
| FileName | nvarchar(500) | Yes | Unchecked | Izveidotais fails (ar ceļu) |
| Counts | int | No | Unchecked | Eksportētu kļūdu daudzums |
| ServiceType | varchar(20) | No | Unchecked | Eksporta servisa nosaukums |
| Success | bit | No | Unchecked | Vai eksports ir veiksmīgs |
| IdFrom | int | No | Unchecked | Eksportētas kļūdas Id no |
| IdTo | int | No | Unchecked | Eksportētas kļūdas Id līdz |
| IntervalStart | datetime | No | Unchecked | Eksporta laika intervāla sākums |
| IntervalEnd | datetime | No | Unchecked | Eksporta laika intervāla beigas |

ErrorExport

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ErrorLogId | int | No | Checked | Kļūdas žurnāla identifikators |
| OsbMessageID | uniqueidentifier | No | Unchecked | Ziņojuma OSB identifikators |
| OsbCreationTime | varchar(25) | Yes | Unchecked | Ziņojuma laiks atbilstoši OSB ierakstītam |
| MessageTime | datetime | No | Unchecked | Ziņojuma apstrādes laiks uz servera |
| ServiceType | varchar(20) | No | Unchecked | Ziņojumu apstrādājošais serviss |
| ErrorXml | xml | No | Unchecked | Kļūdains xml apgabals |
| Code | varchar(10) | Yes | Unchecked | EVK kļūdas kods |
| ErrorMessage | nvarchar(1000) | No | Unchecked | EVK kļūdas ziņojums |

### Datubāze: EVK – Nepersonificētā

DocumentComments

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| DocumentCommentID | int | 4 | No | Document comment identifier. |
| DocumentID | varchar | 25 | yes | Document instance identifier (CDA documentId). |
| SetID | varchar | 25 | yes | Document identifier that is equal to all version of one document (CDA SetId) |
| Comments | nchar | 20 | yes | Comments. |
| Author | xml | -1 | yes | CDA author |
| ModifDate | datetime | 8 | yes | Last record modifcation date and time. |
| ModifUser | nchar | 20 | yes | Last modification user (SQL/Windows lietotājs, ar ko sistēma pieslēgusies). |
| EntryTemplateOID | varchar | 50 | yes |  |
| EntryID | int | 4 | yes |  |

DocumentErrors

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ErrorID | int | 4 | yes |  |
| DocumentID | varchar | 25 | yes |  |
| ErrorData | nvarchar | -1 | yes |  |
| ErrorCode | nvarchar | 100 | yes |  |

DocumentNumberSequence

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| Custodian | varchar | 25 | No |  |
| MaxNumber | int | 4 | No |  |

DocumentProcessingTaskFields

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| DocumentProcessingTaskFieldID | int | 4 | No |  |
| DocumentProcessingTaskID | int | 4 | No |  |
| Field | varchar | 50 | No |  |
| XPath | varchar | 500 | No |  |

DocumentProcessingTasks

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| DocumentProcessingTaskID | int | 4 | No |  |
| DocumentTemplateOID | varchar | 50 | yes |  |
| SectionTemplateOID | varchar | 50 | yes |  |
| EntryTemplateOID | varchar | 50 | No |  |
| EntryIDXPath | nvarchar | 200 | yes |  |
| NegationIndXPath | nvarchar | 600 | yes |  |
| Task | varchar | 50 | No |  |
| TaskType | varchar | 10 | No |  |
| EntryTransformationXSLForText | xml | -1 | yes |  |
| EntryTransformationXSLForXML | xml | -1 | yes |  |

Documents

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| DocumentID | varchar | 25 | No | Document instance identifier (CDA documentId). |
| AccessRights | tinyint | 1 | No | Access rights: self/delegate/physician |
| SetID | varchar | 25 | yes | Document identifier that is equal to all version of one document (CDA SetId) |
| CDA | varbinary | -1 | yes | CDA XML, encrypted (and maybe compressed). |
| Title | nvarchar | 500 | yes | Document title (CDA Title). |
| VersionNumber | int | 4 | No | Document version number (CDA Version Number). |
| Signature | varbinary | -1 | yes | Signature. |
| HTML | varbinary | -1 | yes | Document html format (generated from XML). Encrypted |
| NPatientID | int | 4 | No | Reference to NPatient (de-personalized patient). |
| EffectiveTime | datetime | 8 | No | Document effective date and time (CDA EffectiveDate). |
| Status | varchar | 15 | yes | Document status (0-[Processing],1-[Actual],2-[Incorrect],3-[Cancelled]) |
| CustodianExtension | varchar | 25 | yes | Medical institution code (CDA custodian assignedCustodian representedCustodianOrganization id extension). |
| AuthorExtension | varchar | 25 | yes | Physician code (CDA author assignedAuthor id extension). |
| Author | xml | -1 | yes | CDA author |
| ModifDate | datetime | 8 | yes | Last record modifcation date and time. |
| ModifUser | nchar | 20 | yes | Last modification user (SQL/Windows lietotājs, ar ko sistēma pieslēgusies). |
| DocumentTemplateOID | varchar | 50 | No | Document template object identifier (CDA TemplateOD). |
| History | xml | -1 | yes | XML lauks, kas satur statusma maiņas vēsturi |

DocumentSections

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| DocumentSectionID | int | 4 | No | Document section identifier. |
| SectionTypeOID | varchar | 50 | No | Document section type identifier. |
| DocumentID | varchar | 25 | yes | Reference to document. |
| SectionXML | xml | -1 | No | Section XML. |
| Author | xml | -1 | yes | CDA author |
| ModifDate | datetime | 8 | yes | Last record modifcation date and time. |
| ModifUser | nchar | 20 | yes | SQL/Windows user. |

DocumentsLOB

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| EffectiveTime | datetime |  | No |  |
| DocumentID | varchar | 25 | No |  |
| CDA | varbinary | max | yes |  |
| Signature | varbinary | max | yes |  |
| HTML | varbinary | max | yes |  |
| ConfidentialityRecords | xml |  | yes |  |

DocumentTemplates

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| DocumentTemplateOID | varchar | 50 | No | Document template object identifier. |
| DocumentTypeOID | varchar | 50 | No | Document type object identifier. Document types is classifier that is kept outside of EVK database (Classification Modul). (CDA Code) |
| Validator | nvarchar | 8000 | yes | Validation class name. Package name from validation server validators cataloque. |
| XSLTransformation | xml | -1 | yes | XSLT transformation file. |
| TemplateSchema | xml | -1 | yes | XSD schema file. |
| Model | varbinary | -1 | yes | MDHT model file, UML. |
| Description | nvarchar | -1 | yes | Template description file. |
| ValidFrom | datetime | 8 | yes | Template effective date from (before this date template is not valid). |
| ValidTill | datetime | 8 | yes | Template effective date till (after this date template is not valid). |
| Version | int | 4 | No | Template version number (? can be detected from DocumentTemplateOID ?). |
| Statuss | varchar | 10 | No | Template version. |
| Author | xml | -1 | yes |  |
| ModifDate | datetime | 8 | yes | Last modification date and time. |
| ModifUser | nchar | 20 | yes | Last modification user (SQL/Windows lietotājs, ar ko sistēma pieslēgusies). |

DocumentTemplatesModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| DocumentTemplateOID | varchar | 50 | No | Document template object identifier. |
| DocumentTypeOID | varchar | 50 | No | Document type object identifier. Document types is classifier that is kept outside of EVK database (Classification Modul). (CDA Code) |
| Validator | nvarchar | 8000 | yes | Validation class name. Package name from validation server validators cataloque. |
| XSLTransformation | xml | -1 | yes | XSLT transformation file. |
| TemplateSchema | xml | -1 | yes | XSD schema file. |
| Model | varbinary | -1 | yes | MDHT model file, UML. |
| Description | nvarchar | -1 | yes | Template description file. |
| ValidFrom | datetime | 8 | yes | Template effective date from (before this date template is not valid). |
| ValidTill | datetime | 8 | yes | Template effective date till (after this date template is not valid). |
| Version | int | 4 | No | Template version number (? can be detected from DocumentTemplateOID ?). |
| Statuss | varchar | 10 | No | Template version. |
| Author | xml | -1 | yes |  |
| ModifDate | datetime | 8 | yes | Last modification date and time. |
| ModifUser | nchar | 20 | yes | Last modification user (SQL/Windows lietotājs, ar ko sistēma pieslēgusies). |

NPatientAllergies

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientAllergyID | int | 4 | No |  |
| NPatientID | int | 4 | No |  |
| AlergyCodeSystem | nvarchar | 100 | No |  |
| AlergyCode | nvarchar | 40 | No | Vai AlergyName |
| AlergyDisplayValue | nvarchar | 100 | No |  |
| Date | date | 3 | No |  |
| EntryXML | xml | -1 | No |  |
| AccessRights | tinyint | 1 | No |  |
| AlergyGroupCodeSystem | nvarchar | 100 | No |  |
| AlergyGroupCode | nvarchar | 40 | No |  |
| AlergyGroupDisplayValue | nvarchar | 100 | No |  |
| History | xml | -1 | No |  |

NPatientAllergiesLinkToDocuments

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientAllergyLinkToDocumentsID | int | 4 | No |  |
| PatientAllergyID | int | 4 | No |  |
| DocumentID | varchar | 25 | yes |  |

NPatientAllergiesLinkToDocumentsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientAllergyLinkToDocumentsID | int | 4 | No |  |
| PatientAllergyID | int | 4 | No |  |
| DocumentID | varchar | 25 | No |  |
| ModifDate | datetime | 8 | No | Faktiskais darbības laiks |
| ModifReason | tinyint | 1 | No |  |

NPatientAllergiesModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientAllergyID | int | 4 | No |  |
| NPatientID | int | 4 | No |  |
| AlergyCodeSystem | nvarchar | 100 | No |  |
| AlergyCode | nvarchar | 40 | No | Vai AlergyName |
| AlergyDisplayValue | nvarchar | 100 | No |  |
| Date | date | 3 | No |  |
| EntryXML | xml | -1 | No |  |
| AccessRights | tinyint | 1 | No |  |
| AlergyGroupCodeSystem | nvarchar | 100 | No |  |
| AlergyGroupCode | nvarchar | 40 | No |  |
| AlergyGroupDisplayValue | nvarchar | 100 | No |  |
| History | xml | -1 | No |  |
| ModifDate | datetime | 8 | No | Faktiskais darbības laiks |
| ModifReason | tinyint | 1 | No |  |

NPatientDiagnosis

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientDiagnosisID | int | 4 | No |  |
| NPatientID | int | 4 | No |  |
| DiagonosisCodeSystem | nvarchar | 100 | No |  |
| DiagonosisCode | nvarchar | 40 | No |  |
| DiagonosisDisplayValue | nvarchar | 500 | No |  |
| Date | date | 3 | No |  |
| EntryXML | xml | -1 | No |  |
| AccessRights | tinyint | 1 | No |  |
| History | xml | -1 | No |  |
| DateActual | datetime | 8 | Yes |  |

NPatientDiagnosisLinkToDocuments

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientDiagnosisLinkToDocumentsID | int | 4 | No |  |
| PatientDiagnosisID | int | 4 | No |  |
| DocumentID | varchar | 25 | No |  |
| EffectiveTime | datetime | 8 | yes |  |
| ActivityGUID | uniqueidentifier | 16 | Yes |  |
| ModifDate | datetime | 8 | No |  |
| Del | bit | 1 | No |  |

NPatientDiagnosisLinkToDocumentsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientDiagnosisLinkToDocumentsID | int | 4 | No |  |
| PatientDiagnosisID | int | 4 | No |  |
| DocumentID | varchar | 25 | No |  |
| EffectiveTime | datetime | 8 | yes |  |
| ActivityGUID | uniqueidentifier | 16 | Yes |  |
| ModifDate | datetime | 8 | No |  |
| Del | bit | 1 | No |  |
| ModifDate | datetime | 8 | No | Faktiskais darbības laiks |
| ModifReason | tinyint | 1 | No |  |

NPatientDiagnosisModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientDiagnosisID | int | 4 | No |  |
| NPatientID | int | 4 | No |  |
| DiagonosisCodeSystem | nvarchar | 100 | No |  |
| DiagonosisCode | nvarchar | 40 | No |  |
| DiagonosisDisplayValue | nvarchar | 500 | No |  |
| Date | date | 3 | No |  |
| EntryXML | xml | -1 | No |  |
| AccessRights | tinyint | 1 | No |  |
| History | xml | -1 | No |  |
| DateActual | datetime | 8 | Yes |  |
| ModifDate | datetime | 8 | No | Faktiskais darbības laiks |
| ModifReason | tinyint | 1 | No |  |

NPatientDisability

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientDisabilityID | int | 4 | No |  |
| NPatientID | int | 4 | No |  |
| InvGroupCode | varchar | 50 | No |  |
| InvGroupDisplayValue | nvarchar | 500 | Yes | **-** |
| DateFrom | datetime | 8 | No |  |
| DateTo | datetime | 8 | Yes |  |
| AccessRights | tinyint | 1 | No |  |
| ActivityGUID | uniqueidentifier | 16 | Yes |  |
| ModifDate | datetime | 8 | No |  |
| Del | bit | 1 | No |  |

NPatientDiagnosisLinkToDocuments

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientDisabilityLinkToDocumentsID | int | 4 | No |  |
| PatientDisabilityID | int | 4 | No |  |
| DocumentID | varchar | 25 | No |  |
| EffectiveTime | datetime | 8 | yes |  |
| ActivityGUID | uniqueidentifier | 16 | Yes |  |
| ModifDate | datetime | 8 | No |  |
| Del | bit | 1 | No |  |

NPatientDisability Modif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientDisabilityID | int | 4 | No |  |
| NPatientID | int | 4 | No |  |
| InvGroupCode | varchar | 50 | No |  |
| InvGroupDisplayValue | nvarchar | 500 | Yes | **-** |
| DateFrom | datetime | 8 | No |  |
| DateTo | datetime | 8 | Yes |  |
| AccessRights | tinyint | 1 | No |  |
| ActivityGUID | uniqueidentifier | 16 | Yes |  |
| ModifDate | datetime | 8 | No |  |
| Del | bit | 1 | No |  |
| ModifDate | datetime | 8 | No | Faktiskais darbības laiks |
| ModifReason | tinyint | 1 | No |  |

NPatientDiagnosisLinkToDocumentsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientDisabilityLinkToDocumentsID | int | 4 | No |  |
| PatientDisabilityID | int | 4 | No |  |
| DocumentID | varchar | 25 | No |  |
| EffectiveTime | datetime | 8 | yes |  |
| ActivityGUID | uniqueidentifier | 16 | Yes |  |
| ModifDate | datetime | 8 | No |  |
| Del | bit | 1 | No |  |
| ModifDate | datetime | 8 | No | Faktiskais darbības laiks |
| ModifReason | tinyint | 1 | No |  |

NPatientMedicalDevices

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientMedicalDeviceID | int | 4 | No |  |
| NPatientID | int | 4 | No |  |
| DeviceCodeSystem | nvarchar | 100 | No |  |
| DeviceCode | nvarchar | 100 | No | kods / DeviceCode |
| DeviceDisplayValue | nvarchar | 500 | No | nosaukums |
| Manufacturer | nvarchar | 500 | yes | razotajs |
| Series | nvarchar | 100 | yes | serija |
| MedicalInstitutionCodeSystem | nvarchar | 100 | yes | iestāde klasif |
| MedicalInstitutionCode | nvarchar | 40 | yes |  |
| MedicalInstitutionDisplayValue | nvarchar | 500 | yes |  |
| Date | date | 3 | No |  |
| EntryXML | xml | -1 | No |  |
| AccessRights | tinyint | 1 | No |  |
| History | xml | -1 | No |  |

NPatientMedicalDevicesLinkToDocuments

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientMedicalDevicesLinkToDocumentsID | int | 4 | No |  |
| PatientMedicalDeviceID | int | 4 | No |  |
| DocumentID | varchar | 25 | No |  |

NPatientMedicalDevicesLinkToDocumentsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientMedicalDevicesLinkToDocumentsID | int | 4 | No |  |
| PatientMedicalDeviceID | int | 4 | No |  |
| DocumentID | varchar | 25 | No |  |
| ModifDate | datetime | 8 | No | Faktiskais darbības laiks |
| ModifReason | tinyint | 1 | No |  |

NPatientMedicalDevicesModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientMedicalDeviceID | int | 4 | No |  |
| NPatientID | int | 4 | No |  |
| DeviceCodeSystem | nvarchar | 100 | No |  |
| DeviceCode | nvarchar | 100 | No | kods / DeviceCode |
| DeviceDisplayValue | nvarchar | 500 | No | nosaukums |
| Manufacturer | nvarchar | 500 | yes | razotajs |
| Series | nvarchar | 100 | yes | serija |
| MedicalInstitutionCodeSystem | nvarchar | 100 | yes | iestāde klasif |
| MedicalInstitutionCode | nvarchar | 40 | yes |  |
| MedicalInstitutionDisplayValue | nvarchar | 500 | yes |  |
| Date | date | 3 | No |  |
| EntryXML | xml | -1 | No |  |
| AccessRights | tinyint | 1 | No |  |
| History | xml | -1 | No |  |
| ModifDate | datetime | 8 | No | Faktiskais darbības laiks |
| ModifReason | tinyint | 1 | No |  |

NPatientMedications

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column name** | **Type** | **Length** | **Nullable** | **Description** |
| PatientMedicationID | int | 4 | No |  |
| NPatientID | int | 4 | No |  |
| MedicationCodeSystem | nvarchar | 100 | No | Klasif/ |
| MedicationCode | nvarchar | 40 | No |  |
| MedicationDisplayValue | nvarchar | 500 | No |  |
| DateFrom | date | 3 | yes |  |
| DateTo | date | 3 | yes |  |
| Notes | nvarchar | 500 | yes |  |
| EntryXML | xml | -1 | No |  |
| RelatedEntryXML | xml | -1 | yes |  |
| Date | date | 3 | No |  |
| AccessRights | tinyint | 1 | No |  |
| History | xml | -1 | No |  |

NPatientMedicationsLinkToDocuments

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientMedicationsLinkToDocumentsID | int | 4 | No |  |
| PatientMedicationID | int | 4 | No |  |
| DocumentID | varchar | 25 | No |  |

NPatientMedicationsLinkToDocumentsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientMedicationsLinkToDocumentsID | int | 4 | No |  |
| PatientMedicationID | int | 4 | No |  |
| DocumentID | varchar | 25 | No |  |
| ModifDate | datetime | 8 | No | Faktiskais darbības laiks |
| ModifReason | tinyint | 1 | No |  |

NPatientMedicationsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientMedicationID | int | 4 | No |  |
| NPatientID | int | 4 | No |  |
| MedicationCodeSystem | nvarchar | 100 | No | Klasif/ |
| MedicationCode | nvarchar | 40 | No |  |
| MedicationDisplayValue | nvarchar | 500 | No |  |
| DateFrom | date | 3 | yes |  |
| DateTo | date | 3 | yes |  |
| Notes | nvarchar | 500 | yes |  |
| EntryXML | xml | -1 | No |  |
| RelatedEntryXML | xml | -1 | yes |  |
| Date | date | 3 | No |  |
| AccessRights | tinyint | 1 | No |  |
| History | xml | -1 | No |  |
| ModifDate | datetime | 8 | No | Faktiskais darbības laiks |
| ModifReason | tinyint | 1 | No |  |

NPatients

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| NPatientID | int | 4 | No |  |
| YearOfBirth | int | 4 | yes | // Nullable, veidojot tukšu kartiņu. |
| YearOfDeath | int | 4 | yes |  |
| LifeExpectancy | int | 4 | yes | DzivesIlgumsDienas |
| Majority | bit | 1 | No |  |
| Status | varchar | 15 | yes |  |
| ICAOCode | varchar | 2 | yes | Country ICAO code |
| ICAOName | nvarchar | 200 | yes |  |
| ARCode | varchar | 9 | yes |  |
| ATUCode1 | varchar | 7 | yes | ATUC = ATVK |
| ATUName1 | nvarchar | 256 | yes |  |
| ATUCode2 | varchar | 7 | yes |  |
| ATUName2 | nvarchar | 256 | yes |  |
| ATUCode3 | varchar | 7 | yes |  |
| ATUName3 | nvarchar | 256 | yes |  |
| ATUCode4 | varchar | 7 | yes |  |
| ATUName4 | nvarchar | 256 | yes |  |
| Author | xml | -1 | yes |  |
| ModifDate | datetime | 8 | yes | Faktiskais darbības laiks |
| ModifUser | nchar | 20 | yes | SQL/Windows lietotājs, ar ko sistēma pieslēgusies |
| GenderValue | varchar | 1 | yes |  |
| GenderName | nvarchar | 20 | yes |  |
| PersonStatus | nvarchar | 100 | yes |  |
| AccessRights | tinyint | 1 | yes |  |
| PatientGUID | varbinary | 900 | yes | Šifrētā saite uz otru pusi. |
| ChangeVersion | int | 4 | yes |  |

NPatientsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| NPatientID | int | 4 | No |  |
| YearOfBirth | int | 4 | yes | // Nullable, veidojot tukšu kartiņu. |
| YearOfDeath | int | 4 | yes |  |
| LifeExpectancy | int | 4 | yes | DzivesIlgumsDienas |
| Majority | bit | 1 | No | Nav pilngadīgs pēc noklusejuma |
| Status | varchar | 15 | yes |  |
| ICAOCode | varchar | 2 | yes | Country ICAO code |
| ICAOName | nvarchar | 200 | yes |  |
| ARCode | varchar | 9 | yes |  |
| ATUCode1 | varchar | 7 | yes | ATUC = ATVK |
| ATUName1 | nvarchar | 256 | yes |  |
| ATUCode2 | varchar | 7 | yes |  |
| ATUName2 | nvarchar | 256 | yes |  |
| ATUCode3 | varchar | 7 | yes |  |
| ATUName3 | nvarchar | 256 | yes |  |
| ATUCode4 | varchar | 7 | yes |  |
| ATUName4 | nvarchar | 256 | yes |  |
| Author | xml | -1 | yes |  |
| ModifDate | datetime | 8 | yes | Faktiskais darbības laiks |
| ModifUser | nchar | 20 | yes | SQL/Windows lietotājs, ar ko sistēma pieslēgusies |
| GenderValue | varchar | 1 | yes |  |
| GenderName | nvarchar | 20 | yes |  |
| PersonStatus | nvarchar | 100 | yes |  |
| AccessRights | tinyint | 1 | yes |  |
| PatientGUID | varbinary | 900 | yes | Šifrētā saite uz otru pusi. |
| ChangeVersion | int | 4 | yes |  |

NPatientWarnings

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientWarningID | int | 4 | No |  |
| NPatientID | int | 4 | No |  |
| WarningsCodeSystem | nvarchar | 100 | No | Klasif/ |
| WarningsCode | nvarchar | 40 | No |  |
| WarningsDisplayValue | nvarchar | 500 | No |  |
| Description | nvarchar | 500 | yes |  |
| Date | date | 3 | No |  |
| EntryXML | xml | -1 | No |  |
| AccessRights | tinyint | 1 | No |  |
| History | xml | -1 | No |  |

NPatientWarningsLinkToDocuments

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientWarningsLinkToDocumentsID | int | 4 | No |  |
| PatientWarningID | int | 4 | No |  |
| DocumentID | varchar | 25 | No |  |

NPatientWarningsLinkToDocumentsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientWarningsLinkToDocumentsID | int | 4 | No |  |
| PatientWarningID | int | 4 | No |  |
| DocumentID | varchar | 25 | No |  |
| ModifDate | datetime | 8 | No | Faktiskais darbības laiks |
| ModifReason | tinyint | 1 | No |  |

NPatientWarningsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientWarningID | int | 4 | No |  |
| NPatientID | int | 4 | No |  |
| WarningsCodeSystem | nvarchar | 100 | No | Klasif/nosaukums no kurienes? |
| WarningsCode | nvarchar | 40 | No |  |
| WarningsDisplayValue | nvarchar | 500 | No |  |
| Description | nvarchar | 500 | yes |  |
| Date | date | 3 | No |  |
| EntryXML | xml | -1 | No |  |
| AccessRights | tinyint | 1 | No |  |
| History | xml | -1 | No |  |
| ModifDate | datetime | 8 | No | Faktiskais darbības laiks |
| ModifReason | tinyint | 1 | No |  |

DocumentProcessingTaskFields

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| DocumentProcessingTaskFieldID | int | 4 | No |  |
| DocumentProcessingTaskID | int | 4 | No |  |
| Field | varchar | 50 | No |  |
| XPath | varchar | 500 | No |  |

DocumentProcessingTasks

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| DocumentProcessingTaskID | int | 4 | No |  |
| DocumentTemplateOID | varchar | 50 | yes |  |
| SectionTemplateOID | varchar | 50 | yes |  |
| EntryTemplateOID | varchar | 50 | No |  |
| EntryIDXPath | nvarchar | 200 | yes |  |
| NegationIndXPath | nvarchar | 600 | yes |  |
| Task | varchar | 50 | No |  |
| TaskType | varchar | 10 | No |  |
| EntryTransformationXSLForText | xml | -1 | yes |  |
| EntryTransformationXSLForXML | xml | -1 | yes |  |

Documents

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| DocumentID | varchar | 25 | No | Document instance identifier (CDA documentId). |
| AccessRights | tinyint | 1 | No | Access rights: self/delegate/physician |
| SetID | varchar | 25 | yes | Document identifier that is equal to all version of one document (CDA SetId) |
| CDA | varbinary | -1 | yes | CDA XML, encrypted (and maybe compressed). |
| Title | nvarchar | 500 | yes | Document title (CDA Title). |
| VersionNumber | int | 4 | No | Document version number (CDA Version Number). |
| Signature | varbinary | -1 | yes | Signature. |
| HTML | varbinary | -1 | yes | Document html format (generated from XML). Encrypted |
| NPatientID | int | 4 | No | Reference to NPatient (de-personalized patient). |
| EffectiveTime | datetime | 8 | No | Document effective date and time (CDA EffectiveDate). |
| Status | varchar | 15 | yes | Document status (0-[Processing],1-[Actual],2-[Incorrect],3-[Cancelled]) |
| CustodianExtension | varchar | 25 | yes | Medical institution code (CDA custodian assignedCustodian representedCustodianOrganization id extension). |
| AuthorExtension | varchar | 25 | yes | Physician code (CDA author assignedAuthor id extension). |
| Author | xml | -1 | yes | CDA author |
| ModifDate | datetime | 8 | yes | Last record modifcation date and time. |
| ModifUser | nchar | 20 | yes | Last modification user (SQL/Windows lietotājs, ar ko sistēma pieslēgusies). |
| DocumentTemplateOID | varchar | 50 | No | Document template object identifier (CDA TemplateOD). |
| History | xml | -1 | yes | XML lauks, kas satur statusma maiņas vēsturi |

DocumentSections

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| DocumentSectionID | int | 4 | No | Document section identifier. |
| SectionTypeOID | varchar | 50 | No | Document section type identifier. |
| DocumentID | varchar | 25 | yes | Reference to document. |
| SectionXML | xml | -1 | No | Section XML. |
| Author | xml | -1 | yes | CDA author |
| ModifDate | datetime | 8 | yes | Last record modifcation date and time. |
| ModifUser | nchar | 20 | yes | SQL/Windows user. |

DocumentTemplates

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| DocumentTemplateOID | varchar | 50 | No | Document template object identifier. |
| DocumentTypeOID | varchar | 50 | No | Document type object identifier. Document types is classifier that is kept outside of EVK database (Classification Modul). (CDA Code) |
| Validator | nvarchar | 8000 | yes | (Pagaidām uzliku uz Nullabe) Validation class name. Package name from validation server validators cataloque. |
| XSLTransformation | xml | -1 | yes | (Pagaidām uzliku uz Nullabe) XSLT transformation file. |
| TemplateSchema | xml | -1 | yes | XSD schema file. |
| Model | varbinary | -1 | yes | MDHT model file, UML. |
| Description | nvarchar | -1 | yes | Template description file. |
| ValidFrom | datetime | 8 | yes | Template effective date from (before this date template is not valid). |
| ValidTill | datetime | 8 | yes | Template effective date till (after this date template is not valid). |
| Version | int | 4 | No | Template version number (? can be detected from DocumentTemplateOID ?). |
| Statuss | varchar | 10 | No | Template version. |
| Author | xml | -1 | yes |  |
| ModifDate | datetime | 8 | yes | Last modification date and time. |
| ModifUser | nchar | 20 | yes | Last modification user (SQL/Windows lietotājs, ar ko sistēma pieslēgusies). |

NPatientAllergies

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientAllergyID | int | 4 | No |  |
| NPatientID | int | 4 | No |  |
| AlergyCodeSystem | nvarchar | 100 | No |  |
| AlergyCode | nvarchar | 40 | No | Vai AlergyName |
| AlergyDisplayValue | nvarchar | 100 | No |  |
| Date | date | 3 | No |  |
| EntryXML | xml | -1 | No |  |
| AccessRights | tinyint | 1 | No |  |
| AlergyGroupCodeSystem | nvarchar | 100 | No |  |
| AlergyGroupCode | nvarchar | 40 | No |  |
| AlergyGroupDisplayValue | nvarchar | 100 | No |  |
| History | xml | -1 | No |  |

NPatientAllergiesLinkToDocuments

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientAllergyLinkToDocumentsID | int | 4 | No |  |
| PatientAllergyID | int | 4 | No |  |
| DocumentID | varchar | 25 | yes |  |

NPatientAllergiesLinkToDocumentsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientAllergyLinkToDocumentsID | int | 4 | No |  |
| PatientAllergyID | int | 4 | No |  |
| DocumentID | varchar | 25 | No |  |
| ModifDate | datetime | 8 | No | Faktiskais darbības laiks |
| ModifReason | tinyint | 1 | No |  |

NPatientAllergiesModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientAllergyID | int | 4 | No |  |
| NPatientID | int | 4 | No |  |
| AlergyCodeSystem | nvarchar | 100 | No |  |
| AlergyCode | nvarchar | 40 | No | Vai AlergyName |
| AlergyDisplayValue | nvarchar | 100 | No |  |
| Date | date | 3 | No |  |
| EntryXML | xml | -1 | No |  |
| AccessRights | tinyint | 1 | No |  |
| AlergyGroupCodeSystem | nvarchar | 100 | No |  |
| AlergyGroupCode | nvarchar | 40 | No |  |
| AlergyGroupDisplayValue | nvarchar | 100 | No |  |
| History | xml | -1 | No |  |
| ModifDate | datetime | 8 | No | Faktiskais darbības laiks |
| ModifReason | tinyint | 1 | No |  |

NPatientDiagnosis

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientDiagnosisID | int | 4 | No |  |
| NPatientID | int | 4 | No |  |
| DiagonosisCodeSystem | nvarchar | 100 | No |  |
| DiagonosisCode | nvarchar | 40 | No |  |
| DiagonosisDisplayValue | nvarchar | 500 | No |  |
| Date | date | 3 | No |  |
| EntryXML | xml | -1 | No |  |
| AccessRights | tinyint | 1 | No |  |
| History | xml | -1 | No |  |
| DateActual | datetime | 8 | Yes |  |

NPatientDiagnosisLinkToDocuments

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientDiagnosisLinkToDocumentsID | int | 4 | No |  |
| PatientDiagnosisID | int | 4 | No |  |
| DocumentID | varchar | 25 | No |  |
| EffectiveTime | datetime | 8 | yes |  |
| ActivityGUID | uniqueidentifier | 16 | Yes |  |
| ModifDate | datetime | 8 | No |  |
| Del | bit | 1 | No |  |

NPatientDiagnosisLinkToDocumentsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientDiagnosisLinkToDocumentsID | int | 4 | No |  |
| PatientDiagnosisID | int | 4 | No |  |
| DocumentID | varchar | 25 | No |  |
| EffectiveTime | datetime | 8 | yes |  |
| ActivityGUID | uniqueidentifier | 16 | Yes |  |
| ModifDate | datetime | 8 | No |  |
| Del | bit | 1 | No |  |
| ModifDate | datetime | 8 | No | Faktiskais darbības laiks |
| ModifReason | tinyint | 1 | No |  |

NPatientDiagnosisModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientDiagnosisID | int | 4 | No |  |
| NPatientID | int | 4 | No |  |
| DiagonosisCodeSystem | nvarchar | 100 | No |  |
| DiagonosisCode | nvarchar | 40 | No |  |
| DiagonosisDisplayValue | nvarchar | 500 | No |  |
| Date | date | 3 | No |  |
| EntryXML | xml | -1 | No |  |
| AccessRights | tinyint | 1 | No |  |
| History | xml | -1 | No |  |
| DateActual | datetime | 8 | Yes |  |
| ModifDate | datetime | 8 | No | Faktiskais darbības laiks |
| ModifReason | tinyint | 1 | No |  |

NPatientMedicalDevices

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientMedicalDeviceID | int | 4 | No |  |
| NPatientID | int | 4 | No |  |
| DeviceCodeSystem | nvarchar | 100 | No |  |
| DeviceCode | nvarchar | 100 | No | kods / DeviceCode |
| DeviceDisplayValue | nvarchar | 500 | No | nosaukums |
| Manufacturer | nvarchar | 500 | yes | razotajs |
| Series | nvarchar | 100 | yes | serija |
| MedicalInstitutionCodeSystem | nvarchar | 100 | yes | iestāde klasif |
| MedicalInstitutionCode | nvarchar | 40 | yes |  |
| MedicalInstitutionDisplayValue | nvarchar | 500 | yes |  |
| Date | date | 3 | No |  |
| EntryXML | xml | -1 | No |  |
| AccessRights | tinyint | 1 | No |  |
| History | xml | -1 | No |  |

NPatientMedicalDevicesLinkToDocuments

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientMedicalDevicesLinkToDocumentsID | int | 4 | No |  |
| PatientMedicalDeviceID | int | 4 | No |  |
| DocumentID | varchar | 25 | No |  |

NPatientMedicalDevicesLinkToDocumentsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientMedicalDevicesLinkToDocumentsID | int | 4 | No |  |
| PatientMedicalDeviceID | int | 4 | No |  |
| DocumentID | varchar | 25 | No |  |
| ModifDate | datetime | 8 | No | Faktiskais darbības laiks |
| ModifReason | tinyint | 1 | No |  |

NPatientMedicalDevicesModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientMedicalDeviceID | int | 4 | No |  |
| NPatientID | int | 4 | No |  |
| DeviceCodeSystem | nvarchar | 100 | No |  |
| DeviceCode | nvarchar | 100 | No | kods / DeviceCode |
| DeviceDisplayValue | nvarchar | 500 | No | nosaukums |
| Manufacturer | nvarchar | 500 | yes | razotajs |
| Series | nvarchar | 100 | yes | serija |
| MedicalInstitutionCodeSystem | nvarchar | 100 | yes | iestāde klasif |
| MedicalInstitutionCode | nvarchar | 40 | yes |  |
| MedicalInstitutionDisplayValue | nvarchar | 500 | yes |  |
| Date | date | 3 | No |  |
| EntryXML | xml | -1 | No |  |
| AccessRights | tinyint | 1 | No |  |
| History | xml | -1 | No |  |
| ModifDate | datetime | 8 | No | Faktiskais darbības laiks |
| ModifReason | tinyint | 1 | No |  |

NPatientMedications

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientMedicationID | int | 4 | No |  |
| NPatientID | int | 4 | No |  |
| MedicationCodeSystem | nvarchar | 100 | No | Klasif/ |
| MedicationCode | nvarchar | 40 | No |  |
| MedicationDisplayValue | nvarchar | 500 | No |  |
| DateFrom | date | 3 | yes |  |
| DateTo | date | 3 | yes |  |
| Notes | nvarchar | 500 | yes |  |
| EntryXML | xml | -1 | No |  |
| RelatedEntryXML | xml | -1 | yes |  |
| Date | date | 3 | No |  |
| AccessRights | tinyint | 1 | No |  |
| History | xml | -1 | No |  |

NPatientMedicationsLinkToDocuments

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientMedicationsLinkToDocumentsID | int | 4 | No |  |
| PatientMedicationID | int | 4 | No |  |
| DocumentID | varchar | 25 | No |  |

NPatientMedicationsLinkToDocumentsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientMedicationsLinkToDocumentsID | int | 4 | No |  |
| PatientMedicationID | int | 4 | No |  |
| DocumentID | varchar | 25 | No |  |
| ModifDate | datetime | 8 | No | Faktiskais darbības laiks |
| ModifReason | tinyint | 1 | No |  |

NPatientMedicationsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientMedicationID | int | 4 | No |  |
| NPatientID | int | 4 | No |  |
| MedicationCodeSystem | nvarchar | 100 | No | Klasif/ |
| MedicationCode | nvarchar | 40 | No |  |
| MedicationDisplayValue | nvarchar | 500 | No |  |
| DateFrom | date | 3 | yes |  |
| DateTo | date | 3 | yes |  |
| Notes | nvarchar | 500 | yes |  |
| EntryXML | xml | -1 | No |  |
| RelatedEntryXML | xml | -1 | yes |  |
| Date | date | 3 | No |  |
| AccessRights | tinyint | 1 | No |  |
| History | xml | -1 | No |  |
| ModifDate | datetime | 8 | No | Faktiskais darbības laiks |
| ModifReason | tinyint | 1 | No |  |

NPatients

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| NPatientID | int | 4 | No |  |
| YearOfBirth | int | 4 | yes | // Nullable, veidojot tukšu kartiņu. |
| YearOfDeath | int | 4 | yes |  |
| LifeExpectancy | int | 4 | yes | DzivesIlgumsDienas |
| Majority | bit | 1 | No | Nav pilngadīgs pēc noklusejuma |
| Status | varchar | 15 | yes |  |
| ICAOCode | varchar | 2 | yes | Country ICAO code |
| ICAOName | nvarchar | 200 | yes |  |
| ARCode | varchar | 9 | yes |  |
| ATUCode1 | varchar | 7 | yes | ATUC = ATVK |
| ATUName1 | nvarchar | 256 | yes |  |
| ATUCode2 | varchar | 7 | yes |  |
| ATUName2 | nvarchar | 256 | yes |  |
| ATUCode3 | varchar | 7 | yes |  |
| ATUName3 | nvarchar | 256 | yes |  |
| ATUCode4 | varchar | 7 | yes |  |
| ATUName4 | nvarchar | 256 | yes |  |
| Author | xml | -1 | yes |  |
| ModifDate | datetime | 8 | yes | Faktiskais darbības laiks |
| ModifUser | nchar | 20 | yes | SQL/Windows lietotājs, ar ko sistēma pieslēgusies |
| GenderValue | varchar | 1 | yes |  |
| GenderName | nvarchar | 20 | yes |  |
| PersonStatus | nvarchar | 100 | yes |  |
| AccessRights | tinyint | 1 | yes |  |
| PatientGUID | varbinary | 900 | yes | Šifrētā saite uz otru pusi. |
| ChangeVersion | int | 4 | yes |  |

NPatientsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| NPatientID | int | 4 | No |  |
| YearOfBirth | int | 4 | yes | // Nullable, veidojot tukšu kartiņu. |
| YearOfDeath | int | 4 | yes |  |
| LifeExpectancy | int | 4 | yes | DzivesIlgumsDienas |
| Majority | bit | 1 | No | Nav pilngadīgs pēc noklusejuma |
| Status | varchar | 15 | yes |  |
| ICAOCode | varchar | 2 | yes | Country ICAO code |
| ICAOName | nvarchar | 200 | yes |  |
| ARCode | varchar | 9 | yes |  |
| ATUCode1 | varchar | 7 | yes | ATUC = ATVK |
| ATUName1 | nvarchar | 256 | yes |  |
| ATUCode2 | varchar | 7 | yes |  |
| ATUName2 | nvarchar | 256 | yes |  |
| ATUCode3 | varchar | 7 | yes |  |
| ATUName3 | nvarchar | 256 | yes |  |
| ATUCode4 | varchar | 7 | yes |  |
| ATUName4 | nvarchar | 256 | yes |  |
| Author | xml | -1 | yes |  |
| ModifDate | datetime | 8 | yes | Faktiskais darbības laiks |
| ModifUser | nchar | 20 | yes | SQL/Windows lietotājs, ar ko sistēma pieslēgusies |
| GenderValue | varchar | 1 | yes |  |
| GenderName | nvarchar | 20 | yes |  |
| PersonStatus | nvarchar | 100 | yes |  |
| AccessRights | tinyint | 1 | yes |  |
| PatientGUID | varbinary | 900 | yes | Šifrētā saite uz otru pusi. |
| ChangeVersion | int | 4 | yes |  |

NPatientSurgeries

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientSurgeryID | int | 4 | No | Ķirurģiskās iejaukšanās identifikators |
| NPatientID | int | 4 | No | Saite ar pacienta ierakstu |
| DateFirst | datetime | 8 | No | Pirmā konstatētā šī veida iejaukšanās |
| DateLast | datetime | 8 | No | Pēdēja konstatētā šī veida iejaukšanās |
| SurgeryCodeSystem | varchar | 50 | yes | Ķirurģiskās iejaukšanās kodu sistēma |
| SurgeryCode | varchar | 50 | yes | Ķirurģiskās iejaukšanās kods |
| SurgeryDisplayName | nvarchar | 1000 | yes | Ķirurģiskās iejaukšanās lietotājam draudzīgs nosaukums |
| AccessRights | tinyint | 1 | No | Tiesības |
| ActivityGUID | uniqueidentifier | 16 | yes | Originālā ziņojuma message ID no IP |
| ModifDate | datetime | 8 | No | Pēdējās izmaiņas datums |
| Del | bit | 1 | No | Dzēšanas pazīme |

NPatientSurgeriesLinkToDocument

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientSurgeryLinkToDocumentsID | int | 4 | No | Identifikatora lauks |
| PatientSurgeryID | int | 4 | No | Saite ar ķirurģisko iejaukšanās tabulu |
| DocumentID | varchar | 25 | yes | Saite ar dokumenta ierakstu |
| EffectiveTime | datetime | 8 | yes | Dokumenta laiks |
| ActivityGUID | uniqueidentifier | 16 | yes | Ziņojuma messageID no IP |
| Del | bit | 1 | No | Dzēšanas pazīme |
| ModifDate | datetime | 8 | No | Pēdējo izmaiņu datums |

NPatientSurgeriesLinkToDocumentsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 |  |  |
| PatientSurgeryLinkToDocumentsID | int | 4 | No | Identifikatora lauks |
| PatientSurgeryID | int | 4 | No | Saite ar ķirurģiskās iejaukšānās tabulu |
| DocumentID | varchar | 25 | yes | Saite ar dokumenta ierakstu |
| EffectiveTime | datetime | 8 | yes | Dokumenta laiks |
| ActivityGUID | uniqueidentifier | 16 | yes | Ziņojuma messageID no IP |
| Del | bit | 1 | No | Dzēšanas pazīme |
| ModifDate | datetime | 8 | No | Pēdējo izmaiņu datums |

NPatientSurgeriesModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientSurgeryID | int | 4 | No | Ķirurģiskās iejaukšanās identifikators |
| NPatientID | int | 4 | No | Saite ar pacienta ierakstu |
| DateFirst | datetime | 8 | No | Pirmā konstatētā šī veida iejaukšanās |
| DateLast | datetime | 8 | No | Pēdējā konstatētā šī veida iejaukšanās |
| SurgeryCodeSystem | varchar | 50 | yes | Ķirurģiskās iejaukšanās kodu sistēma |
| SurgeryCode | varchar | 50 | yes | Ķirurģiskās iejaukšanās kods |
| SurgeryDisplayName | nvarchar | 1000 | yes | Ķirurģiskās iejaukšanās lietotājam draudzīgs nosaukums |
| AccessRights | tinyint | 1 | No | Tiesības |
| ActivityGUID | uniqueidentifier | 16 | yes | Originālā ziņojuma message ID no IP |
| ModifDate | datetime | 8 | No | Pēdējās izmaiņas datums |
| Del | bit | 1 | No | Dzešanas pazīme |

NPatientTransfusions

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientTransfusionID | int | 4 | No | Asins pārliešanas identifikators |
| NPatientID | int | 4 | No | Saite ar pacienta ierakstu |
| DateStart | datetime | 8 | No | Pārliešanas sākums |
| DateFinish | datetime | 8 | No | Pārliešanas beigas |
| TransfusionVolume | int | 4 | No | Pārliešanas apjoms, ml |
| BloodComponentCodeSystem | varchar | 50 | yes | Komponentes koda sistēma |
| BloodComponentCode | varchar | 50 | yes | Komponentes kods |
| BloodComponentDisplayName | nvarchar | 1000 | yes | Komponentes lietotājam draudzīga vērtība |
| BloodGroupCodeSystem | varchar | 50 | yes | Komponentes asins grupas koda sistēma |
| BloodGroupCode | varchar | 50 | yes | Komponentes asins grupas kods |
| BloodGroupDiplayName | nvarchar | 1000 | yes | Komponentes asins grupas lietotājam draudzīga vērtība |
| Complications | nvarchar | 1000 | yes | Pārliešanas laikā konstatētās blaknes |
| AccessRights | tinyint | 1 | No | Tiesības |
| AcitivityGUID | uniqueidentifier | 16 | yes | Originālā ziņojuma message ID no IP |
| ModifDate | datetime | 8 | No | Pēdējās izmaiņas datums |
| Del | bit | 1 | No |  |

NPatientTransfusionsLinkToDocument

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientTransfusionLinkToDocumentsID | int | 4 | No | Identifikatora lauks |
| PatientTransfusionID | int | 4 | No | Saite ar asins pārliešanas tabulu |
| DocumentID | varchar | 25 | yes | Saite ar dokumenta ierakstu |
| EffectiveTime | datetime | 8 | yes | Dokumenta efektīvais laiks |
| ActivityGUID | uniqueidentifier | 16 | yes | Ziņojuma messageID no IP |
| Del | bit | 1 | No | Dzēšanas pazīme |
| ModifDate | datetime | 8 | No | Pēdējo izmaiņu datums |

NPatientTransfusionsLinkToDocumentsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientTransfusionLinkToDocumentsID | int | 4 | No | Identifikatora lauks |
| PatientTransfusionID | int | 4 | No | Saite ar asins pārliešanas tabulu |
| DocumentID | varchar | 25 | yes | Saite ar dokumenta ierakstu |
| EffectiveTime | datetime | 8 | yes | Dokumenta efektīvais laiks |
| ActivityGUID | uniqueidentifier | 16 | yes | Ziņojuma messageID no IP |
| Del | bit | 1 | No | Dzēšanas pazīme |
| ModifDate | datetime | 8 | No | Pēdējo izmaiņu datums |

NPatientTransfusionsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientTransfusionID | int | 4 | No | Asins pārliešanas identifikators |
| NPatientID | int | 4 | No | Saite ar pacienta ierakstu |
| DateStart | datetime | 8 | No | Pārliešanas sākums |
| DateFinish | datetime | 8 | No | Pārliešanas beigas |
| TransfusionVolume | int | 4 | No | Pārliešanas apjoms, ml |
| BloodComponentCodeSystem | varchar | 50 | yes | Komponentes koda sistēma |
| BloodComponentCode | varchar | 50 | yes | Komponentes kods |
| BloodComponentDisplayName | nvarchar | 1000 | yes | Komponentes lietotājam draudzīga vērtība |
| BloodGroupCodeSystem | varchar | 50 | yes | Komponentes asins grupas koda sistēma |
| BloodGroupCode | varchar | 50 | yes | Komponentes asins grupas kods |
| BloodGroupDiplayName | nvarchar | 1000 | yes | Komponentes asins grupas lietotājam draudzīga vērtība |
| Complications | nvarchar | 1000 | yes | Pārliešanas laikā konstatētas blaknes |
| AccessRights | tinyint | 1 | No | Tiesības |
| AcitivityGUID | uniqueidentifier | 16 | yes | Origināla ziņojuma message ID no IP |
| ModifDate | datetime | 8 | No | Pēdējas izmaiņas datums |
| Del | bit | 1 | No |  |

NPatientWarnings

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientWarningID | int | 4 | No |  |
| NPatientID | int | 4 | No |  |
| WarningsCodeSystem | nvarchar | 100 | No | Klasif/ |
| WarningsCode | nvarchar | 40 | No |  |
| WarningsDisplayValue | nvarchar | 500 | No |  |
| Description | nvarchar | 500 | yes |  |
| Date | date | 3 | No |  |
| EntryXML | xml | -1 | No |  |
| AccessRights | tinyint | 1 | No |  |
| History | xml | -1 | No |  |

NPatientWarningsLinkToDocuments

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| PatientWarningsLinkToDocumentsID | int | 4 | No |  |
| PatientWarningID | int | 4 | No |  |
| DocumentID | varchar | 25 | No |  |

NPatientWarningsLinkToDocumentsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientWarningsLinkToDocumentsID | int | 4 | No |  |
| PatientWarningID | int | 4 | No |  |
| DocumentID | varchar | 25 | No |  |
| ModifDate | datetime | 8 | No | Faktiskais darbības laiks |
| ModifReason | tinyint | 1 | No |  |

NPatientWarningsModif

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ModifID | int | 4 | No |  |
| PatientWarningID | int | 4 | No |  |
| NPatientID | int | 4 | No |  |
| WarningsCodeSystem | nvarchar | 100 | No | Klasif/ |
| WarningsCode | nvarchar | 40 | No |  |
| WarningsDisplayValue | nvarchar | 500 | No |  |
| Description | nvarchar | 500 | yes |  |
| Date | date | 3 | No |  |
| EntryXML | xml | -1 | No |  |
| AccessRights | tinyint | 1 | No |  |
| History | xml | -1 | No |  |
| ModifDate | datetime | 8 | No | Faktiskais darbības laiks |
| ModifReason | tinyint | 1 | No |  |

Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column name | Type | Length | Nullable | Description |
| ParameterID | int | 4 | No |  |
| Name | varchar | 25 | No |  |
| Group | varchar | 25 | No |  |
| Value | varbinary | -1 | No |  |

# Konfigurācijas projektējums

## Klasifikatori

7.1‑1. tabula. Klasifikatori

| # | Nosaukums | Struktūra | Sākotnējie dati |
| --- | --- | --- | --- |
| 1 | Ārstniecības personas | * Identifikācija * Vārds * Uzvārds * Kontaktinformācija * Profesijas (Saraksts) |  |
| 3 | Atļaujas veids | * Kods * Vērtība | |  |  | | --- | --- | | Viens no vecākiem | "PARENT" | | Aizbildnis | "GUARDIAN" | | Aizgādnis | "TRUSTEE" | | Pilnvarojums | "MANDATE" | | Izmeklētājs | "INVESTIGATOR" | |
| 4 | Brīdinājuma tips | * Kods * Vērtība | 1. Medikamentozā blakne  2. Augsts medicīnisks risks |
| 5 | Alerģijas grupa | * Kods * Vērtība |  |
| 6 | Medikamenti | * Kods * Vērtība |  |
| 7 | Ārstniecības iestādes | * Kods * Vērtība |  |
| 8 | eVeselības sistēmas | * Kods * Vērtība |  |
| 9 | Identifikācijas klasifikācija | * Kods * Vērtība | 1. Personas kods  2. Jaundzimušais |
| 10 | Meklēšanas veidi | * Kods * Vērtība | ACTUAL. Aktuālie dati  ALL. Dati ar vēsturi |
| 11 | Objektu pieejas tiesības | * Kods * Vērtība | 001 – Pieeja tikai ārstiem  010 – Pieeja tikai delegātiem  100 – Pieeja tikai pacientam  101 – Pieeja ārstiem un pacientam  110 – Pieeja pacientam un delegātiem  111 – Pieeja visiem |
| 12 | Objektu pieejas tips | * Kods * Vērtība | ENTRY – Pamatdatu elements  DOC – Dokuments  CARD – Pacienta karte |
| 13 | Medicīnas pamatdatu veidi | * Kods * Vērtība | ALLERGY – Alerģijas  MEDICAL\_DEVICE – Medicīnas iekārtas  WARNING – Brīdinājumi  DIAGNOSIS – Diagnozes  MEDICATION - Medikamenti |
| 14 | Lietotāju (visu) lomas | * Kods * Vērtība * Tiesību apraksts (String) (UserRightDescriptor) | |  |  |  | | --- | --- | --- | | Loma | Vērtība | Tiesību apr. | | P1 | Rīcībnespējīgs vai bērns (<14) | 100 000 | | P2 | Daļēja rīcībspēja (14<pilngadība) | 100 010 | | P3 | Pilngadīgs | 100 011 | | P4 | Vecāki, Aizbildnis (bērns<14) | 010 101 | | P5 | Vecāki, Aizbildnis (14<pilngadība) | 010 001 | | P7 | Aizgādnis | 010 101 | |  | Ārsts vai ārstniecības atbalsta persona | 001  110 | |  | Izmeklētājs | 111 000 | |
| 15 | Kontaktpersonu tipi | * Kods * Vērtība |  |
| 16 | Pacienta karšu operācijas. | * Kods * Vērtība | CREATE: Izveidot jauno pacienta karti  MERGE: Apvienot divas kartes, kas pieder vienam cilvēkam |
| 17 | Pacienta karšu operācijas parametri | * Kods * Vērtība | NEW: Jaunās kartes izveidošanas pacienta identifikācija  SOURCE: Karte, kura tiek izmantota ka saplūdināšanas avots  TARGET: Pacienta karte, kura tiek izmantota, ka saplūdināšanas mērķis |
| 18 | Dokumentu veidnes | * Kods * Vērtība (nosaukums) * Tips [Klasifikators] |  |
| 19 | Dokumentu tipi | * Kods * Vērtība (nosaukums) * Grupa [Klasifikators] |  |
| 20 | Dokumentu grupas | * Kods * Vērtība (nosaukums) |  |
| 21 | Dokumentu statusi | * Kods * Vērtība | PROCESSING: Apstrāde  ACTUAL: Aktuāls  INCORRECT: Kļūdains  CANCELLED: Anulēts |
| 22 | Veidņu statusi | * Kods * Vērtība | ACTUAL: Aktuāls  CANCELLED: Anulēts |
| 23 | Kontaktinformācijas veidi | * Kods * Vērtība | EMAIL: E-pasts  TEL: Tālruņa numurus |

\* Nav specifisku prasību.

## Sistēmas elementāro tiesību komplekts

Tabulā pārskaitītas sistēmā izmantojamas tiesības un tīmekļa pakalpes, kuru ietvaros tās tiek izmantotas.

7.2‑1. tabula. Tiesību komplekts

| Nr. | | Nosaukums | Apraksts | Izmantošana (WS) | Primārās vai saistītās tiesības | |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | EvkRghtDocumentAdd | Pievienot dokumentu | addDocument | P |
| 1.1 | EvkRghtDocumentCdaAdd | Pievienot jaunu CDA dokumentu vai tā versiju | addDocument | S |
| 1.2 | EvkRghtDocumentCcdAdd | Papildināt un mainīt veselības pamatdatus | addDocument | S |
| 2 | EvkRghtDocumentCommentAdd | Pievienot dokumentam piezīmes | addDocumentComments | P |
| 3 | EvkRghtDocumentStatusSet | Anulēt dokumentu | setDocumentStatus | P |
| 5 | EvkRghtDocumentsGet | Iegūt dokumentu | getDocumentList, getDocument | P |
| 5.1 | EvkRghtDocumentsActualGet | Iegūt aktuālo dokumentu sarakstu,  Iegūt aktuālu dokumentu | getDocumentList  getDocument | S |
| 5.2 | EvkRghtDocumentsAllGet | Iegūt jebkura statusa dokumentu sarakstu | getDocumentList, getDocument | S |
| 5.3 | EvkRghtDocumentCcdGet | Iegūt veselības pamatdatu dokumentu | getDocument | S |
| 5.4 | EvkRghtDocumentCommentsGet | Iegūt dokumenta piezīmes | getDocument | S |
| 6 | EvkRghtPatientCardGet | Iegūt pacienta karti | getPatientCard | P |
| 6.1 | EvkRghtPatientPersonDataGet | Iegūt personas datus | getPatientCard | S |
| 6.2 | EvkRghtPatientContactsGet | Iegūt kontaktinformāciju | getPatientCard | S |
| 6.3 | EvkRghtPatientContactPersonsGet | Iegūt kontaktpersonu informāciju | getPatientCard | S |
| 6.4 | EvkRghtPatientEhicGet | Iegūt informāciju par EVAK | getPatientCard | S |
| 6.5 | EvkRghtPatientFamilyDoctorGet | Iegūt informāciju par ģimenes ārstu | getPatientCard | S |
| 6.6 | EvkRghtPatientDelegationsGet | Iegūt pilnvarojumu sarakstu | getPatientCard | S |
| 6.7 | EvkRghtPatientPermissionsGet | Iegūt atļauju sarakstu | getPatientCard | S |
| 7 | EvkRghtPatientCardSet | Pievienot un papildināt pacienta kartes datus | setPatientCard | P |
| 7.1 | EvkRghtPatientPersonDataSet | Labot personas datus | setPatientCard | S |
| 7.2 | EvkRghtPatientContactsSet | Pievienot un labot kontaktinformāciju | setPatientCard | S |
| 7.3 | EvkRghtPatientContactPersonSet | Pievienot un labot kontaktpersonu informāciju | setPatientCard | S |
| 7.4 | EvkRghtPatientDelegationSet | Pievienot un labot pilnvarojumus | setPatientCard | S |
| 7.5 | EvkRghtPatientPermissionSet | Pievienot un labot atļaujas | setPatientCard | S |
| 7.6 | EvkRghtPatientMajoritySet | Labot personas pilngadības statusu | setPatientCard | S |
| 8 | EvkRghtPatientCardCreate | Izveidot pacienta karti | createPatientCard | P |
| 9 | EvkRghtAccessRightsSet | Uzstādīt un mainīt aizliegumus | setAccessRights | P |
| 9.1 | EvkRghtPatientCardAccessRightsSet | Pievienot un labot „pilno” aizliegumu | setPatientCard | S |
| 9.2 | EvkRghtDocumentAccessRightsSet | Pievienot un labot dokumenta aizliegumu | setAccessRights | S |
| 9.3 | EvkRghtHealthBasicDataAccessRightsSet | Pievienot un labot veselības pamatdatu aizliegumu | setAccessRights | S |
| 10 | EvkRghtPatientCardsMerge | Sapludināt pacientu kartes | mergePatientCard | P |
| 11 | EvkRghtDocumentTemplateSet | Rediģēt dokumenta šablonu | setDocTemplate | P |
| 12 | EvkRghtDocumentTemplateGet | Iegūt dokumentu veidņu sarakstu,  Iegūt dokumenta veidnes datus | getDocTemplateList, getDocTemplate | P |
| 13 | EvkRghtPatientFamilyDoctorSet | Mainīt pacienta ģimenes ārsta datus | setPatientFamilyDoctorChanges | P |
| 14 | EvkRghtPatientEhicSet | Mainīt pacienta EVAK datus | setPatientEhicChanges | P |
| 15 | EvkRghtCheckDetailedRole | Iegūt detalizēto lomu | checkDetailedRole | P |
| 16 | EvkRghtNotificationConfigurationSet |  | setNotificationConfiguration | P |
| 17 | EvkRghtNotificationConfigurationGet |  | getNotificationConfiguration | P |
| 18 | EvkRghtDocumentNumberGet |  | getNextDocumentNumber | P |
| 19 | EvkRghtEVKEpacParametersGet |  | getEVKEpacParameters | P |
| 20 | EvkRghtProfileSettings | Profilu uzstādījumu atlase/uzstādīšana | getProfileSettings, setProfileSettings | P |
| 21 | EvkRghtClassifiers | EVK saglabāto klasifikatoru kopiju izgūšana | getClassifierValues | P |

## Sistēmas audita ziņojumu nodošana Syslog serverim

Infrastruktūras stāvokļa audita ziņojumu nodošana tiks nodrošināta ar Windows event loga eksporta palīdzību. Uz sistēmas serveriem tiks uzinstalēta eventlog-to-syslog servisa lietotne, ar kuras palīdzību Windows eventlog informācija tiks nodota uz NVD norādīto centrālo syslog serveri.

# Sākotnējās datu ielādes projektējums

## Sākotnējās datu ielādes komponente

Identifikācija: EVKCDAImport

Sākotnējā datu ielāde tiks veikta, izmantojot konsoles lietojumprogrammu „EVKCDAImport”.

**Ievads:** lietojumprogramma tiek iedarbināta, palaižot to no konsoles un norādot ieejas parametrus.

Ieejas parametri:

* „/?”: Palaižot lietojumprogrammu ar šo parametru, komandrindas konsolē tiek izvadīta lietojumprogrammas īsa lietošanas pamācība;
* „/source:<value>”: Datu avots – fails, kas satur importējamos datus. Pilns ceļš līdz failam, vai relatīvais ceļš līdz failam;
* „/sourcetype:<value>”: Datu avota tips, kas identificē importējamo datu kopu, datu kopas ir identificētas un uzskaitītas „8.1.1 Datu kopas”;
* „/user:<value>”: lietotājs, ar ko tiks veikta autentifikācijas integrācijas platformā, un kas tiks uzskatīts par dokumenta autoru;
* „/pwd:<value>”: lietotāja parole.

### Datu kopas

Datu kopas nosaka komponentes uzvedību. Katra datu kopa identificē paredzamo ieejas datu formātu, datu apstrādes loģiku atbilstoša CDA dokumenta reģistrācijai.

Tiek atpazītas šādas datu kopas:

* Cukura diabēta reģistra diagnozes (kods: cdd);
* Cukura diabēta reģistra medikamenti (kods: cdm);
* Onkoloģisko slimību reģistra diagnozes (kods: vrd);
* Multiplās sklerozes slimību reģistra diagnozes (kods: msd);
* Tuberkulozes slimību reģistra diagnozes (kods: tbcd);
* Iedzimto anomāliju reģistra diagnozes (kods: ierd);
* Psihisko slimību reģistra diagnozes (kods: psihd);
* Narkoloģijas reģistra diagnozes (kods: narkd);
* NVD VIS Stacionāra taloni (kods: vncst);

### Konfigurācija

Lietojumprogramma satur konfigurācijas elementus, kas ir kopīgi visām apstrādājamajām datu kopām:

* Dokumenta repozitorija pakalpes metode, kas nodrošina CDA dokumentu pievienošanu EVK. Metodes apraksts „6.2.1 addDocument ”;
* Nepieciešamo konfigurāciju, lai varētu veikt autentificētu pieprasījumu izmantojot integrācijas platformas drošības talona principu.

## Pacientu reģistri (PREDA)

Identifikācija: PREDADataImport

**Apraksts:** PREDA reģistru datiem tiek veidoti medicīniskie dokumenti. Tiks apstrādāta tikai pamata informācija pacienta kartes veselības pamatdatiem.

Sistēma veic vienreizēju datu importu personām, par kurām PREDA sistēmas turētāji ir sagatavojuši reģistra datus.

Ir nepieciešams sagatavot vairākas datu kopas:

* Cukura diabēta reģistra diagnozes (datu kopas saīsinājums, kas tiks lietots importā identificējot datu kopas tipu: cdd);
* Cukura diabēta reģistra medikamenti (datu kopas saīsinājums, kas tiks lietots importā identificējot datu kopas tipu: cdm);
* Onkoloģisko slimību reģistra diagnozes (datu kopas saīsinājums, kas tiks lietots importā identificējot datu kopas tipu: vrd);
* Multiplās sklerozes slimību reģistra diagnozes (datu kopas saīsinājums, kas tiks lietots importā identificējot datu kopas tipu: msd);
* Tuberkulozes slimību reģistra diagnozes (datu kopas saīsinājums, kas tiks lietots importā identificējot datu kopas tipu: tbcd);
* Iedzimto anomāliju reģistra diagnozes (datu kopas saīsinājums, kas tiks lietots importā identificējot datu kopas tipu: ierd);
* Psihisko slimību reģistra diagnozes (datu kopas saīsinājums, kas tiks lietots importā identificējot datu kopas tipu: psihd);
* Narkoloģijas reģistra diagnozes (datu kopas saīsinājums, kas tiks lietots importā identificējot datu kopas tipu: narkd);

Sagatavojot datu kopas, ir jāņem vērā kopējs princips visām datu kopām:

* Datu kopas formāts XML;
* Personai aizpildīts personas kods;
* Personas statuss nenorāda, ka persona ir mirusi.

Ielādējot datu kopas, tiek ņemts vērtā kopējs princips visām datu kopām:

* Katrai personai tiek sagatavots CDA dokuments atbildoši pamatdatu dokumenta modelim „1.3.6.1.4.1.38760.1.2.1.1.1”;
* CDA dokuments tiek sūtīts uz pakalpi „6.2.1 addDocument ”, kas nodrošina CDA dokumentu reģistrāciju un apstrādi. Atbilstoši CDA dokumentu sekciju ierakstu apstrādes uzdevumam „6.2.12.7 documentEntryProcessing (dokumenta ierakstu apstrāde)”, CDA dokumentā iekļautie dati tiks izgūti no CDA dokumenta sekciju ierakstiem un ievietoti personas pamatdatu tabulās.

Datu pirmreizējā importa nodrošināšanai tiek izveidota programmatūras vienība, kas aprakstīta: „8.1 Sākotnējās datu ielādes komponente”.

Darbinot lietojumprogrammu, ir nepieciešams norādīt ieejas parametrus:

* „/source:<value>”: Datu avots – XML fails, kas satur importējamos datus;
* „/sourcetype:<value>”: Datu avota tips, kas atbilst importējamajai datu kopai „8.1.1 Datu kopas”;
* „/user:<value>”: lietotājs, ar ko tiks veikta autentifikācijas integrācijas platformā, un kas tiks uzskatīts par dokumenta autoru;
* „/pwd:<value>”: lietotāja parole.

### Ar noteiktām slimībām slimojošu pacientu reģistrs par pacientiem, kuriem diagnosticēts cukura diabēts

Apraksts:

Importējamā datu kopa: Cukura diabēta reģistra diagnozes

Importējamās datu kopas kods: cdd

Datu atlase: Katrai personai, kas pašreiz ir reģistra uzskaitē, ir jāatlasa viens ieraksts, un jānoformē atbilstoši datu shēmai „cddShema.xsd”, piemērs „cddExample.xml”.

8.2‑1. tabula. Datu atlases XSD shēmas lauku atbilstība

| Nr. | XSD lauks | Reģistra lauks | Atlases apraksts |
| --- | --- | --- | --- |
| 1. | PersonasKods | XPersonasDati.PersonasKods |  |
| 2. | SSKDgKods | CDPacientaKarte.SSKDgKods | Personai saistītā pacienta karte, lauks CDPacientaKarte.SSKDgKods |
| 3. | DiabetsDiagnosticetsGads | CDPacientaKarte.DiabetsDiagnosticetsGads | Personai saistītā pacienta karte, lauks CDPacientaKarte.DiabetsDiagnosticetsGads |

Tiek ģenerēts CDA dokuments pēc šāda algoritma :

1. Aizpilda galveni (header) pēc šādas tabulas informācijas :

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | templateId | [root] 1.3.6.1.4.1.38760.1.2.1.1  [extension] 1.3.6.1.4.1.38760.1.2.1.1.1 |
| 2 | TypeId | [root] 2.16.840.1.113883.1.3  [extension] POCD\_HD000040 |
| 3 | Id | [root] 1.3.6.1.4.1.38760.3.4.5.1  [extension] PREDA.[KeyWords].NewDocumentNo.versionNumber.. |
| 4 | Code | [code] 1.3.6.1.4.1.38760.1.2.1.1  [codeSystem] 1.3.6.1.4.1.38760.1.2.1  [displayName] Medicīnas pamatdatu kopsavilkuma dokuments |
| 5 | Title | „Medicīnas pamatdatu kopsavilkuma dokuments cukura diabēta reģistrs” |
| 6 | effectiveTime | Ģenerēšanas datums un laiks |
| 7 | confidentialityCode | „N” |
| 8 | languageCode | „LV” |
| 9 | setId | [root] 1.3.6.1.4.1.38760.3.4.5.4  [extension] VDEAVK.[KeyWords].NewDocumentNo |
| 10 | versionNumber | 1 |
| 11 | recordTarget.patientRole.id | [root] 1.3.6.1.4.1.38760.3.1.1  [extension] PersonasKods |
| 12 | author | author.time  [Value] = Ģenerēšanas datums un laiks  Author.assignedAuthor.id  [extension] = „Preda”  [root]=1.3.6.1.4.1.38760.3.4.6.1  Author.assignedAuthor.assignedPerson.name  [given]=PREDA  [family]=PREDA  Author.assignedAuthor.assignedPerson. representedOrganization.id  [extension]=”PREDA”  [root]=”1.3.6.1.4.1.38760.3.4.6.1” |

Pamatdatu tipam veido sekciju (section), kuru aizpilda pēc atbilstošas tabulas :

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | Section | [code.code] = „1.3.6.1.4.1.38760.1.2.2.2”  [code.codesystem] = „1.3.6.1.4.1.38760.2.21”  [code.codeSystemName] = „OID reģistrs”  [Code.displayname] = „Pacienta pamatdatu diagnozes sekcija”  [title] = „Diagnozes”  [text] = „Pacienta diagnoze ir” SSKDgKods pacienta kartes informācija„ un diagnozes uzstādīšanas datums ir ” DiabetsDiagnosticetsGads” |

Pamatdatu tipam veido ierakstu (entry), kuru aizpilda pēc atbilstošas tabulas :

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | Entry | [code.code] = „1.3.6.1.4.1.38760.1.2.3.2”  [code.codeSystem] = „1.3.6.1.4.1.38760.2.21”  [code.codeSystemName] = „OID reģistrs”  [code.displayname] = „Pacienta diagnoze”  [value.code] = „E11”  [value.displayName] = „Insulīnneatkarīgs cukura diabēts”  [value.codeSystem] = „1.3.6.1.4.1.38760.2.159”  [effectiveTime.value] = Ģenerēšanas datums un laiks |

Datu importa rezultātā tiek veidots CDA dokuments atbilstoši šablona identifikatoram 1.3.6.1.4.1.38760.1.2.1.1.1”, un reģistrēts sistēmā izmantojot metodi „6.2.1 addDocument ”.

Atbilstoši CDA dokumentu sekciju ierakstu apstrādes uzdevumam „6.2.12.7 documentEntryProcessing (dokumenta ierakstu apstrāde)”, katram diagnozes ierakstam reģistrētajā CDA dokumentā tiek izveidots atbilstošais diagnozes ieraksts pamatdatu tabulā „NPatientDiagnosis”, norādot gan datu autoru, gan datu avotu (CDA dokuments). Katrai importētai diagnozei tiek reģistrēts viens piezīmes ieraksts izsaucot funkciju 6.2.2 addComments (Pievienot piezīmes), kas norāda uz to, ka diagnozes datums nav pilnībā precīzs, diagnozes datums tiek ģenerēts balstoties uz lauku, kas satur gadu „DiabetsDiagnosticetsGads”. Piezīmes ieraksts satur šādu tekstu: „Diagnozes uzstādīšanas datums satur informāciju par diagnozes uzstādīšanas gadu. Mēnesis un diena nav precīzi sākotnējo datu ielādes ierobežotības dēļ.”.

8.2‑2. tabula. Diagnozes datu kopa no cukura diabēta reģistra

| Nr. | Objekts | Lauks | Atbilstošais lauks XML datu avotā |
| --- | --- | --- | --- |
| 1. | Diagnoze | Pacienta ID | Nosaka pēc personas koda |
| 2. | Diagnoze | Diagnoze | Lauks CCD.SSKDgKods |
| 3. | Diagnoze | Diagnozes datums | Lauks CCD.DiabetsDiagnosticetsGads, gadu transformējot uz 1.janvāra datumu 01.01.GADS, kur:   * 01 – datums * 01 – mēnesis * GADS – gada vērtība no reģistra |

Cukura diabēta reģistrā ir aptuveni 75 000 dzīvu personu.

Importējamā datu kopa: Cukura diabēta reģistra medikamenti

Importējamās datu kopas kods: cdm

Datu atlase: Katrai personai, kas pašreiz ir reģistra uzskaitē, ir jāreģistrē viens vai vairāki ieraksti medikamentu sarakstītā, datus ņemot no pēdējās uzskaites kartes, kārtojot pēc lauka CDUzskaitesKarte.Datums. Datu sagatavošana ir jāveic atbilstoši shēmai „cdmShema.xsd”, piemērs „cdmExample.xml”.

8.2‑3. tabula. Datu atlases XSD shēmas lauku atbilstība

| Nr. | XSD lauks | Reģistra lauks | Atlases apraksts |
| --- | --- | --- | --- |
| 1. | PersonasKods | XPersonasDati.PersonasKods |  |
| 2. | Medikaments | CDMedikaments.Medikaments | Personai saistītais medikaments caur personas karti, uzskaites karti un uzskaites kartes detaļu tabulu, sasaistes lauks CDUzskaitesKarteDetalas.MedikamentsID |
| 3. | Deva | CDUzskaitesKarteDetalas.Deva | Personai saistītā medikamenta detaļas ieraksts caur personas karti, uzskaites karti |
| 4. | Datums | CDUzskaitesKarte.Datums | Datu atlases procesā atrastās jaunākās uzskaites kartes datuma lauks |

Tiek ģenerēts dokuments pēc šāda algoritma :

1. Aizpilda galveni (header) pēc šādas tabulas informācijas :

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | templateId | [root] 1.3.6.1.4.1.38760.1.2.1.1  [extension] 1.3.6.1.4.1.38760.1.2.1.1.1 |
| 2 | TypeId | [root] 2.16.840.1.113883.1.3  [extension] POCD\_HD000040 |
| 3 | Id | [root] 1.3.6.1.4.1.38760.3.4.5.1  [extension] PREDA.[KeyWords].NewDocumentNo.versionNumber.. |
| 4 | Code | [code] 1.3.6.1.4.1.38760.1.2.1.1  [codeSystem] 1.3.6.1.4.1.38760.1.2.1  [displayName] Medicīnas pamatdatu kopsavilkuma dokuments |
| 5 | Title | „Medicīnas pamatdatu kopsavilkuma dokuments cukura diabēta reģistra medikamentiem” |
| 6 | effectiveTime | Ģenerēšanas datums un laiks |
| 7 | confidentialityCode | „N” |
| 8 | languageCode | „LV” |
| 9 | setId | [root] 1.3.6.1.4.1.38760.3.4.5.4  [extension] PREDA.[KeyWords].NewDocumentNo |
| 10 | versionNumber | 1 |
| 11 | recordTarget.patientRole.id | [root] 1.3.6.1.4.1.38760.3.1.1  [extension] PersonasKods |
| 12 | author | author.time  [Value] = Ģenerēšanas datums un laiks  Author.assignedAuthor.id  [extension] = „Preda”  [root]=1.3.6.1.4.1.38760.3.4.6.1  Author.assignedAuthor.assignedPerson.name  [given]=PREDA  [family]=PREDA  Author.assignedAuthor.assignedPerson.repsresentedOrganization.id  [extension]=”PREDA”  [root]=”1.3.6.1.4.1.38760.3.4.6.1” |

Pamatdatu tipam veido sekciju (section), kuru aizpilda pēc atbilstošas tabulas:

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | Section | [code.code] = „1.3.6.1.4.1.38760.1.2.2.4”  [code.codesystem] = „1.3.6.1.4.1.38760.2.21”  [code.codeSystemName] = „OID reģistrs”  [code.displayName] = „Pacienta pamatdatu medikamenta sekcija”  [title] = „Medikamenti”  [text] = „Pacienta medikaments ir ” [Medikaments] lauka informācija”, medikamenta deva ir „[Deva] lauka informācija” un diagnozes uzstādīšanas datums ir ” [Datums]” |

Pamatdatu tipam veido ierakstu (entry), kuru aizpilda pēc atbilstošas tabulas:

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | Entry | [code.code] = „1.3.6.1.4.1.38760.1.2.3.4”  [code.codeSystem] = „1.3.6.1.4.1.38760.2.21”  [code.codeSystemName] = „OID reģistrs”  [code.displayName] = „Pacienta pamatdatu ieraksts medikamentam”  [value.code] = „A02AD04”  [value.displayName] = „Hydrotalcitum”  [value.codeSystem] = „1.3.6.1.4.1.38760.2.136”  [effectiveTime.value] = Ģenerēšanas datums un laiks |

Datu importa rezultātā tiek veidots CDA dokuments atbilstoši šablona identifikatoram „1.3.6.1.4.1.38760.1.2.1.1.1”, un reģistrēts sistēmā izmantojot metodi „6.2.1 addDocument ”. Atbilstoši CDA dokumentu ierakstu apstrādes uzdevumam „6.2.12.7 documentEntryProcessing (dokumenta ierakstu apstrāde)”, katram medikamenta ierakstam reģistrētajā CDA dokumentā tiek izveidots atbilstošais medikamenta ieraksts pamatdatu tabulā „NPatientMedications”, norādot gan datu autoru, gan datu avotu (CDA dokuments).

8.2‑4. tabula. Medikamenta datu kopa no cukura diabēta reģistra

| Nr. | Objekts | Lauks | Atbilstošais lauks XML datu avotā |
| --- | --- | --- | --- |
| 1. | Medikaments | Pacienta ID | Nosaka pēc personas koda |
| 2. | Medikaments | Medikaments | CDM.Medikaments |
| 3. | Medikaments | Piezīmes | CDM.Deva |
| 4. | Medikaments | Datums no | CDM.Datums |

Cukura diabēta reģistrā ir aptuveni 100 000 importējamu medikamentu ierakstu.

### Ar noteiktām slimībām slimojošu pacientu reģistrs par pacientiem, kuriem diagnosticēta onkoloģiska slimība

**Apraksts:**

Importējamā datu kopa: Onkoloģisko slimību reģistra diagnozes

Importējamās datu kopas kods: vrd

Datu atlase: Katrai personai, kas pašreiz ir reģistra uzskaitē, ir jāsameklē saistītās uzskaites kartes un katrai atrastai uzskaites kartei ir jāiegūst viens ieraksts. Iegūto rezultātu jānoformē atbilstoši datu shēmai „vrdShema.xsd”, piemērs „vrdExample.xml”.

8.2‑5. tabula. Datu atlases XSD shēmas lauku atbilstība

| Nr. | XSD lauks | Reģistra lauks | Atlases apraksts |
| --- | --- | --- | --- |
| 1. | PersonasKods | XPersonasDati.PersonasKods |  |
| 2. | SSKDgKods | VRUzskaitesKarte.SSKDgKods | Personai saistītā uzskaites karte. |
| 3. | DiagnozesDatums | VRUzskaitesKarte.DiagnozesDatums | Personai saistītā uzskaites karte. |

Tiek ģenerēts dokuments pēc šāda algoritma :

1. Aizpilda galveni (header) pēc šādas tabulas informācijas:

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | templateId | [root] 1.3.6.1.4.1.38760.1.2.1.1  [extension] 1.3.6.1.4.1.38760.1.2.1.1.1 |
| 2 | TypeId | [root] 2.16.840.1.113883.1.3  [extension] POCD\_HD000040 |
| 3 | Id | [root] 1.3.6.1.4.1.38760.3.4.5.1  [extension] PREDA.[KeyWords].NewDocumentNo.versionNumber |
| 4 | Code | [code] 1.3.6.1.4.1.38760.1.2.1.1  [codeSystem] 1.3.6.1.4.1.38760.1.2.1  [displayName] Onkoloģisko slimību reģistra diagnozes |
| 5 | Title | „Medicīnas pamatdatu kopsavilkuma dokuments onkoloģisko slimību reģistra diagnozēm” |
| 6 | effectiveTime | Ģenerēšanas datums un laiks |
| 7 | confidentialityCode | „N” |
| 8 | languageCode | „LV” |
| 9 | setId | [root] 1.3.6.1.4.1.38760.3.4.5.4  [extension] PREDA.[KeyWords].NewDocumentNo |
| 10 | versionNumber | 1 |
| 11 | recordTarget.patientRole.id | [root] 1.3.6.1.4.1.38760.3.1.1  [extension] PersonasKods |
| 12 | author | author.time  [Value] = Ģenerēšanas datums un laiks  Author.assignedAuthor.id  [extension] = „Preda”  [root]=1.3.6.1.4.1.38760.3.4.6.1  Author.assignedAuthor.assignedPerson.name  [given]=PREDA  [family]=PREDA  Author.assignedAuthor.assignedPerson.repsresentedOrganization.id  [extension]=”PREDA”  [root]=”1.3.6.1.4.1.38760.3.4.6.1” |

Pamatdatu tipam veido sekciju (section), kuru aizpilda pēc atbilstošas tabulas:

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | Section | [code.code] = „1.3.6.1.4.1.38760.1.2.2.2”  [code.codeSystem] = „1.3.6.1.4.1.38760.2.21”  [code.codeSystemName] = „OID reģistrs”  [code.displayName] = „Pacienta pamatdatu diagnozes sekcija”  [title] = „Diagnozes”  [text] = „Pacienta diagnozes ir ” [SSKDgKods ] lauka attiecīgās kartes informācija”, un diagnozes uzstādīšanas datums ir [DiagnozesDatums] |

Pamatdatu tipam veido ierakstu (entry), kuru aizpilda pēc atbilstošas tabulas :

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | Entry | [code.code] = „1.3.6.1.4.1.38760.1.2.3.2”  [code.codeSystem] = „1.3.6.1.4.1.38760.2.21”  [code.codeSystemName] = „OID reģistrs”  [code.displayName] = „Pacienta diagnoze”  [value.code] = „”  [value.displayName] = „”  [value.codeSystem] = „1.3.6.1.4.1.38760.2.159”  [effectiveTime.value] = Ģenerēšanas datums un laiks |

Datu importa rezultātā tiek veidots CDA dokuments atbilstoši šablona identifikatoram „1.3.6.1.4.1.38760.1.2.1.1.1”, un reģistrēts sistēmā, izmantojot metodi „6.2.1 addDocument ”. Atbilstoši CDA dokumentu sekciju ierakstu apstrādes uzdevumam „6.2.12.7 documentEntryProcessing (dokumenta ierakstu apstrāde)”, katram diagnozes ierakstam reģistrētajā CDA dokumentā tiek izveidots atbilstošais diagnozes ieraksts pamatdatu tabulā „NPatientDiagnosis”, norādot gan datu autoru, gan datu avotu (CDA dokuments).

8.2‑6. tabula. Diagnozes datu kopa no onkoloģisko slimību reģistra

| Nr. | Objekts | Lauks | Atbilstošais lauks XML datu avotā |
| --- | --- | --- | --- |
| 1. | Diagnoze | Pacienta ID | Nosaka pēc personas koda |
| 2. | Diagnoze | Diagnoze | Lauks VRD.SSKDGKods |
| 3. | Diagnoze | Diagnozes datums | Lauks VRD.DiagnozesDatums |

Onkoloģisko slimību reģistrā ir aptuveni 70 000 dzīvu personu uzskaites kartes.

### Ar noteiktām slimībām slimojošu pacientu reģistrs par pacientiem, kuriem diagnosticēta multiplā skleroze

**Apraksts:**

Importējamā datu kopa: Multiplās sklerozes slimību reģistra diagnozes

Importējamās datu kopas kods: msd

Datu atlase: Katrai personai, kas pašreiz ir reģistra uzskaitē, ir jāsameklē saistītās uzskaites kartes (MSVesture). Datus jāņem tikai no tās uzskaites kartes, kur lauka MSVesture.Datums vērtība ir mazākā. Rezultātā jāiegūst datu masīvs, kur katrai personai ir atbilstoši viens diagnozes ieraksts, kas jānoformē atbilstoši datu shēmai „msdShema.xsd”, piemērs „msdExample.xml”.

8.2‑7. tabula. Datu atlases XSD shēmas lauku atbilstība

| Nr. | XSD lauks | Reģistra lauks | Atlases apraksts |
| --- | --- | --- | --- |
| 1. | PersonasKods | XPersonasDati.PersonasKods |  |
| 2. | Diagnoze | MSVesture.Diagnoze | Personai saistītā uzskaites karte |
| 3. | Datums | MSVesture.Datums | Personai saistītā uzskaites karte |

Tiek ģenerēts dokuments pēc šāda algoritma:

1. Aizpilda galveni (header) pēc šādas tabulas informācijas :

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | templateId | [root] 1.3.6.1.4.1.38760.1.2.1.1  [extension] 1.3.6.1.4.1.38760.1.2.1.1.1 |
| 2 | TypeId | [root] 2.16.840.1.113883.1.3  [extension] POCD\_HD000040 |
| 3 | Id | [root] 1.3.6.1.4.1.38760.3.4.5.1  [extension] PREDA.[KeyWords].NewDocumentNo.versionNumber |
| 4 | Code | [code] 1.3.6.1.4.1.38760.1.2.1.1  [codeSystem] 1.3.6.1.4.1.38760.1.2.1  [displayName] Multiplās sklerozes slimību reģistra diagnozes |
| 5 | Title | „Pacienta pamatdatu kopsavilkuma dokuments no multiplās sklerozes slimību reģistra”. |
| 6 | effectiveTime | Ģenerēšanas datums un laiks |
| 7 | confidentialityCode | „N” |
| 8 | languageCode | „LV” |
| 9 | setId | [root] 1.3.6.1.4.1.38760.3.4.5.4  [extension] PREDA.[KeyWords].NewDocumentNo |
| 10 | versionNumber | 1 |
| 11 | recordTarget.patientRole.id | [root] 1.3.6.1.4.1.38760.3.1.1  [extension] PersonasKods |
| 12 | author | author.time  [Value] = Ģenerēšanas datums un laiks  Author.assignedAuthor.id  [extension] = „Preda”  [root]=1.3.6.1.4.1.38760.3.4.6.1  Author.assignedAuthor.assignedPerson.name  [given]=PREDA  [family]=PREDA  Author.assignedAuthor.assignedPerson.repsresentedOrganization.id  [extension]=”PREDA”  [root]=”1.3.6.1.4.1.38760.3.4.6.1” |

Pamatdatu tipam veido sekciju (section), kuru aizpilda pēc atbilstošas tabulas:

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | Section | [code.code] = „1.3.6.1.4.1.38760.1.2.2.2”  [code.codeSystem] = „1.3.6.1.4.1.38760.2.21”  [code.codeSystemName] = „OID reģistrs”  [code.displayName] = „Pacienta pamatdatu diagnozes sekcija”  [title] = „Diagnozes”  [text] = „Pacienta diagnozes ir ” [PersonasKods] lauka attiecīgās kartes informācija”, un diagnozes uzstādīšanas datums ir [Datums] |

Pamatdatu tipam veido ierakstu (entry), kuru aizpilda pēc atbilstošas tabulas:

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | Entry | [code.code] = „1.3.6.1.4.1.38760.1.2.3.2”  [code.codesystem] = „1.3.6.1.4.1.38760.2.21”  [code.codeSystemName] = „OID reģistrs”  [code.displayName] = „Pacienta diagnoze”  [value.code] =  [value.displayName] =  [value.codeSystem] = „1.3.6.1.4.1.38760.2.159”  [effectiveTime.value] = Ģenerēšanas datums un laiks |

Datu importa rezultātā tiek veidots CDA dokuments atbilstoši šablona identifikatoram „1.3.6.1.4.1.38760.1.2.1.1.1”, un reģistrēts sistēmā, izmantojot metodi „6.2.1 addDocument ”. Atbilstoši CDA dokumentu sekciju ierakstu apstrādes uzdevumam „6.2.12.7 documentEntryProcessing (dokumenta ierakstu apstrāde)”, katram diagnozes ierakstam reģistrētajā CDA dokumentā tiek izveidots atbilstošais diagnozes ieraksts pamatdatu tabulā „NPatientDiagnosis”, norādot gan datu autoru, gan datu avotu (CDA dokuments).

8.2‑8. tabula. Diagnozes datu kopa no onkoloģisko slimību reģistra

| Nr. | Objekts | Lauks | Atbilstošais lauks XML datu avotā |
| --- | --- | --- | --- |
| 1. | Diagnoze | Pacienta ID | Nosaka pēc personas koda |
| 2. | Diagnoze | Diagnoze | Lauks MSD.Diagnoze |
| 3. | Diagnoze | Diagnozes datums | Lauks MSD.Datums |

Multiplās sklerozes slimību reģistrā ir aptuveni 2000 dzīvu personu uzskaites kartes.

### Ar noteiktām slimībām slimojošu pacientu reģistrs par pacientiem, kuriem diagnosticēta tuberkuloze

**Apraksts:**

Importējamā datu kopa: Tuberkulozes slimību reģistra diagnozes

Importējamās datu kopas kods: tbcd

Datu atlase: Katrai personai, kas pašreiz ir reģistra uzskaitē, ir jāsameklē saistītās uzskaites kartes (TBCUzskaitesKarte). Šīs saistītās uzskaites kartes tiek grupētas pēc diagnozes un iegūts minimālais datums, pie katras diagnozes (tiek sameklētas unikālās diagnozes ar datumu, kad šī diagnoze pirmo reizi ir fiksēta). Iegūtie jānoformē atbilstoši datu shēmai „tbcdShema.xsd”, piemērs „tbcdExample.xml”.

8.2‑9. tabula. Datu atlases XSD shēmas lauku atbilstība

| Nr. | XSD lauks | Reģistra lauks | Atlases apraksts |
| --- | --- | --- | --- |
| 1. | PersonasKods | XPersonasDati.PersonasKods |  |
| 2. | SSKDgKods | TBCUzskaitesKarte.SSKDgKods | Personai saistītā uzskaites karte |
| 3. | UzskaitesDatums | TBCUzskaitesKarte.UzskaitesDatums | Personai saistītā uzskaites karte |

Tiek ģenerēts dokuments pēc šāda algoritma :

1. Aizpilda galveni (header) pēc šādas tabulas informācijas :

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | templateId | [root] 1.3.6.1.4.1.38760.1.2.1.1  [extension] 1.3.6.1.4.1.38760.1.2.1.1.1 |
| 2 | TypeId | [root] 2.16.840.1.113883.1.3  [extension] POCD\_HD000040 |
| 3 | Id | [root] 1.3.6.1.4.1.38760.3.4.5.1  [extension] PREDA.[KeyWords].NewDocumentNo.versionNumber |
| 4 | Code | [code] 1.3.6.1.4.1.38760.1.2.1.1  [codeSystem] 1.3.6.1.4.1.38760.1.2.1  [displayName] Tuberkulozes slimību reģistra diagnozes |
| 5 | Title | „Pacienta pamatdatu kopsavilkuma dokuments no tuberkulozes slimību reģistra”. |
| 6 | effectiveTime | Ģenerēšanas datums un laiks |
| 7 | confidentialityCode | „N” |
| 8 | languageCode | „LV” |
| 9 | setId | [root] 1.3.6.1.4.1.38760.3.4.5.4  [extension] PREDA.[KeyWords].NewDocumentNo |
| 10 | versionNumber | 1 |
| 11 | recordTarget.patientRole.id | [root] 1.3.6.1.4.1.38760.3.1.1  [extension] PersonasKods |
| 12 | author | author.time  [Value] = Ģenerēšanas datums un laiks  Author.assignedAuthor.id  [extension] = „Preda”  [root]=1.3.6.1.4.1.38760.3.4.6.1  Author.assignedAuthor.assignedPerson.name  [given]=PREDA  [family]=PREDA  Author.assignedAuthor.assignedPerson.repsresentedOrganization.id  [extension]=”PREDA”  [root]=”1.3.6.1.4.1.38760.3.4.6.1” |
|  |  |  |

Pamatdatu tipam veido sekciju (section), kuru aizpilda pēc atbilstošas tabulas:

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | Section | [code.code] = „1.3.6.1.4.1.38760.1.2.2.2”  [code.codeSystem] = „1.3.6.1.4.1.38760.2.21”  [code.codeSystemName] = „OID reģistrs”  [code.displayName] = „Pacienta pamatdatu diagnozes sekcija”  [title] = „Diagnozes”  [text] = „Pacienta diagnozes ir ” [PersonasKods] lauka attiecīgās kartes pēc lauka [SSKDgKods] informācija”, un diagnozes uzstādīšanas datums ir [UzskaitesDatums]. |

Pamatdatu tipam veido ierakstu (entry), kuru aizpilda pēc atbilstošas tabulas :

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | Entry | [code.code] = „1.3.6.1.4.1.38760.1.2.3.2”  [code.codesystem] = „1.3.6.1.4.1.38760.2.21”  [code.codeSystemName] = „OID reģistrs”  [Code.displayname] = „Pacienta diagnoze”  [value.code] = „”  [value.displayName] = „”  [value.codeSystem] = „1.3.6.1.4.1.38760.2.159”  [effectiveTime.value] = Ģenerēšanas datums un laiks |

Datu importa rezultātā tiek veidots CDA dokuments atbilstoši šablona identifikatoram „1.3.6.1.4.1.38760.1.2.1.1.1”, un reģistrēts sistēmā, izmantojot metodi „6.2.1 addDocument ”. Atbilstoši CDA dokumentu ierakstu apstrādes uzdevumam „6.2.12.7 documentEntryProcessing (dokumenta ierakstu apstrāde)”, katram diagnozes ierakstam reģistrētajā CDA dokumentā tiek izveidots atbilstošais diagnozes ieraksts pamatdatu tabulā „NPatientDiagnosis”, norādot gan datu autoru, gan datu avotu (CDA dokuments).

8.2‑10. tabula. Diagnozes datu kopa no tuberkulozes slimību reģistra

| Nr. | Objekts | Lauks | Atbilstošais lauks XML datu avotā |
| --- | --- | --- | --- |
| 1. | Diagnoze | Pacienta ID | Nosaka pēc personas koda |
| 2. | Diagnoze | Diagnoze | Lauks TBCD.SSKDgKods |
| 3. | Diagnoze | Diagnozes datums | Lauks TBCD.UzskaitesDatums |

### Ar noteiktām slimībām slimojošu pacientu reģistrs par pacientiem, kuriem diagnosticēta iedzimta anomālija

**Apraksts:**

Importējamā datu kopa: Iedzimto anomāliju reģistra diagnozes

Importējamās datu kopas kods: ierd

Datu atlase: Katrai personai tiek iegūta saistītā uzskaites (IERUzskaitesKarte) karte un šīs uzskaites kartes defekta diagnozes (IERIedzimtaDefektaDiagnoze). Iegūtais datu masīvs tiek noformēts atbilstoši datu shēmai „ierdShema.xsd”, piemērs „ierdExample.xml”.

8.2‑11. tabula. Datu atlases XSD shēmas lauku atbilstība

| Nr. | XSD lauks | Reģistra lauks | Atlases apraksts |
| --- | --- | --- | --- |
| 1. | PersonasKods | XPersonasDati.PersonasKods |  |
| 2. | SSKDgKods | IERIedzimtaDefektaDiagnoze.SSKDgKods | Visas personai saistītās defekta diagnozes. |
| 3. | DiagnozesUzstadisanasDatums | IERUzskaitesKarte.DiagnozesUzstadisanasDatums | Ja diagnozes uzstādīšanas datums nav aizpildīts, tad diagnozes datums tiek atstāts tukšs (NULL) |

Tiek ģenerēts dokuments pēc šāda algoritma :

1. Aizpilda galveni (header) pēc šādas tabulas informācijas:

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | templateId | [root] 1.3.6.1.4.1.38760.1.2.1.1  [extension] 1.3.6.1.4.1.38760.1.2.1.1.1 |
| 2 | TypeId | [root] 2.16.840.1.113883.1.3  [extension] POCD\_HD000040 |
| 3 | Id | [root] 1.3.6.1.4.1.38760.3.4.5.1  [extension] PREDA.[KeyWords].NewDocumentNo.versionNumber |
| 4 | Code | [code] 1.3.6.1.4.1.38760.1.2.1.1  [codeSystem] 1.3.6.1.4.1.38760.1.2.1  [displayName] Iedzimto anomāliju slimību reģistra diagnozes |
| 5 | Title | „Pacienta pamatdatu kopsavilkuma dokuments no iedzimto anomāliju reģistra”. |
| 6 | effectiveTime | Ģenerēšanas datums un laiks |
| 7 | confidentialityCode | „N” |
| 8 | languageCode | „LV” |
| 9 | setId | [root] 1.3.6.1.4.1.38760.3.4.5.4  [extension] PREDA.[KeyWords].NewDocumentNo |
| 10 | versionNumber | 1 |
| 11 | recordTarget.patientRole.id | [root] 1.3.6.1.4.1.38760.3.1.1  [extension] PersonasKods |
| 12 | author | author.time  [Value] = Ģenerēšanas datums un laiks  Author.assignedAuthor.id  [extension] = „Preda”  [root]=1.3.6.1.4.1.38760.3.4.6.1  Author.assignedAuthor.assignedPerson.name  [given]=PREDA  [family]=PREDA  Author.assignedAuthor.assignedPerson.repsresentedOrganization.id  [extension]=”PREDA”  [root]=”1.3.6.1.4.1.38760.3.4.6.1” |
|  |  |  |

Pamatdatu tipam veido sekciju (section), kuru aizpilda pēc atbilstošas tabulas:

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | Section | [code.code] = „1.3.6.1.4.1.38760.1.2.2.2”  [code.codeSystem] = „1.3.6.1.4.1.38760.2.21”  [code.codeSystemName] = „OID reģistrs”  [code.displayName] = „Pacienta pamatdatu diagnozes sekcija”  [title] = „Diagnozes”  [text] = „Pacienta diagnozes ir ” [PersonasKods] lauka attiecīgās kartes pēc lauka [SSKDgKods] informācija”, un diagnozes uzstādīšanas datums ir [DiagnozesUzstadisanasDatums]. |

Pamatdatu tipam veido ierakstu (entry), kuru aizpilda pēc atbilstošas tabulas:

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | Entry | [code.code] = „1.3.6.1.4.1.38760.1.2.3.2”  [code.codesystem] = „1.3.6.1.4.1.38760.2.21”  [code.codeSystemName] = „OID reģistrs”  [Code.displayname] = „Pacienta diagnoze”  [value.code] = „”  [value.displayName] = „”  [value.codeSystem] = „1.3.6.1.4.1.38760.2.159”  [effectiveTime.value] = Ģenerēšanas datums un laiks |

Datu importa rezultātā tiek veidots CDA dokuments atbilstoši šablona identifikatoram „1.3.6.1.4.1.38760.1.2.1.1.1”, un reģistrēts sistēmā izmantojot metodi „6.2.1 addDocument ”. Atbilstoši CDA dokumentu sekciju ierakstu apstrādes uzdevumam „6.2.12.7 documentEntryProcessing (dokumenta ierakstu apstrāde)”, katram diagnozes ierakstam reģistrētajā CDA dokumentā tiek izveidots atbilstošais diagnozes ieraksts pamatdatu tabulā „NPatientDiagnosis”, norādot gan datu autoru, gan datu avotu (CDA dokuments). Katrai importētai diagnozei tiek reģistrēts viens piezīmes ieraksts, kas norāda uz to, kad diagnozes datums nav zināms, diagnozes datums tiek atstāts ar vērtību NULL.

8.2‑12. tabula. Diagnozes datu kopa no iedzimto anomāliju reģistra

| Nr. | Objekts | Lauks | Atbilstošais lauks XML datu avotā |
| --- | --- | --- | --- |
| 1. | Diagnoze | Pacienta ID | Nosaka pēc personas koda |
| 2. | Diagnoze | Diagnoze | Lauks IERD.SSKDgKods |
| 3. | Diagnoze | Diagnozes datums | Lauks IERD.DiagnozesUzstadisanasDatums |

### Ar noteiktām slimībām slimojošu pacientu reģistrs par pacientiem, kuriem diagnosticēti psihiski un uzvedības traucējumi

**Apraksts:**

Importējamā datu kopa: Psihisko slimību reģistra diagnozes

Importējamās datu kopas kods: psihd

Datu atlase: Katrai personai, kas datu eksporta brīdī ir reģistra uzskaitē, ir jāizveido viens diagnozes ieraksts. Diagnozes ieraksta pamatā ir personas uzskaites karte ar jaunāko kartes datumu - lauks „PSIHUzskaitesKarte.KartesDatums”. Iegūtais datu masīvs tiek noformēts atbilstoši datu shēmai „psihdShema.xsd”, piemērs „psihdExample.xml”.

8.2‑13. tabula. Datu atlases XSD shēmas lauku atbilstība

| Nr. | XSD lauks | Reģistra lauks | Atlases apraksts |
| --- | --- | --- | --- |
| 1. | PersonasKods | XPersonasDati.PersonasKods |  |
| 2. | PamatDgKods | PSIHUzskaitesKarte.PamatDgKods | Visas personai saistītās uzskaites kartes |
| 3. | KartesDatums | PSIHUzskaitesKarte.KartesDatums | Visas personai saistītās uzskaites kartes |

Tiek ģenerēts dokuments pēc šāda algoritma :

1. Aizpilda galveni (header) pēc šādas tabulas informācijas:

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | templateId | [root] 1.3.6.1.4.1.38760.1.2.1.1  [extension] 1.3.6.1.4.1.38760.1.2.1.1.1 |
| 2 | TypeId | [root] 2.16.840.1.113883.1.3  [extension] POCD\_HD000040 |
| 3 | Id | [root] 1.3.6.1.4.1.38760.3.4.5.1  [extension] PREDA.[KeyWords].NewDocumentNo.versionNumber |
| 4 | Code | [code] 1.3.6.1.4.1.38760.1.2.1.1  [codeSystem] 1.3.6.1.4.1.38760.1.2.1  [displayName] Psihisko slimību reģistra diagnozes |
| 5 | Title | „Pacienta pamatdatu kopsavilkuma dokuments no psihisko slimību reģistra” |
| 6 | effectiveTime | Ģenerēšanas datums un laiks |
| 7 | confidentialityCode | „N” |
| 8 | languageCode | „LV” |
| 9 | setId | [root] 1.3.6.1.4.1.38760.3.4.5.4  [extension] PREDA.[KeyWords].NewDocumentNo |
| 10 | versionNumber | 1 |
| 11 | recordTarget.patientRole.id | [root] 1.3.6.1.4.1.38760.3.1.1  [extension] PersonasKods |
| 12 | author | author.time  [Value] = Ģenerēšanas datums un laiks  Author.assignedAuthor.id  [extension] = „Preda”  [root]=1.3.6.1.4.1.38760.3.4.6.1  Author.assignedAuthor.assignedPerson.name  [given]=PREDA  [family]=PREDA  Author.assignedAuthor.assignedPerson.repsresentedOrganization.id  [extension]=”PREDA”  [root]=”1.3.6.1.4.1.38760.3.4.6.1” |

Pamatdatu tipam veido sekciju, kuru aizpilda pēc atbilstošas tabulas:

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | Section | [code.code] = „1.3.6.1.4.1.38760.1.2.2.2”  [code.codeSystem] = „1.3.6.1.4.1.38760.2.21”  [code.codeSystemName] = „OID reģistrs”  [code.displayName] = „Pacienta pamatdatu diagnozes sekcija”  [title] = „Diagnozes”  [text] = „Pacienta diagnozes ir ” [PersonasKods] lauka attiecīgās kartes pēc lauka [PamatDgKods] informācija”, un diagnozes uzstādīšanas datums ir [KartesDatums]. |

Pamatdatu tipam veido sekciju, kuru aizpilda pēc atbilstošas tabulas :

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | Entry | [code.code] = „1.3.6.1.4.1.38760.1.2.3.2”  [code.codesystem] = „1.3.6.1.4.1.38760.2.21”  [code.codeSystemName] = „OID reģistrs”  [Code.displayname] = „Pacienta diagnoze”  [value.code] = „”  [value.displayName] = „”  [value.codeSystem] = „1.3.6.1.4.1.38760.2.159”  [effectiveTime.value] = Ģenerēšanas datums un laiks |

Datu importa rezultātā tiek veidots CDA dokuments atbilstoši šablona identifikatoram „1.3.6.1.4.1.38760.1.2.1.1.1”, un reģistrēts sistēmā, izmantojot metodi „6.2.1 addDocument ”. Atbilstoši CDA dokumentu sekciju ierakstu apstrādes uzdevumam „6.2.12.7 documentEntryProcessing (dokumenta ierakstu apstrāde)”, katram diagnozes ierakstam reģistrētajā CDA dokumentā tiek izveidots atbilstošais diagnozes ieraksts pamatdatu tabulā „NPatientDiagnosis”, norādot gan datu autoru, gan datu avotu (CDA dokuments).

8.2‑14. tabula. Diagnozes datu kopa no psihisko slimību reģistra

| Nr. | Objekts | Lauks | Atbilstošais lauks XML datu avotā |
| --- | --- | --- | --- |
| 1. | Diagnoze | Pacienta ID | Nosaka pēc personas koda |
| 2. | Diagnoze | Diagnoze | Lauks PSIHD.PamatDgKods |
| 3. | Diagnoze | Diagnozes datums | Lauks PSIHD.KartesDatums |

### Ar noteiktām slimībām slimojošu pacientu reģistrs par narkoloģiskajiem pacientiem

**Apraksts:**

Importējamā datu kopa: Narkoloģijas reģistra diagnozes

Importējamās datu kopas kods: narkd

Datu atlase: Katrai personai, kas uz datu eksporta brīdi ir reģistra uzskaitē, ir jāizveido viens diagnozes ieraksts. Diagnozes ieraksta pamatā ir personas uzskaites karte ar jaunāko kartes datumu - lauks „NARKUzskaitesKarte.KartesDatums”. Iegūtais datu masīvs tiek noformēts atbilstoši datu shēmai „narkdShema.xsd”, piemērs „narkdExample.xml”.

8.2‑15. tabula. Datu atlases XSD shēmas lauku atbilstība

| Nr. | XSD lauks | Reģistra lauks | Atlases apraksts |
| --- | --- | --- | --- |
| 1. | PersonasKods | XPersonasDati.PersonasKods |  |
| 2. | SSKDgKods | NARKUzskaitesKarte.SSKDgKods | Visas personai saistītās uzskaites kartes |
| 3. | DgDatums | NARKUzskaitesKarte.DgDatums | Visas personai saistītās uzskaites kartes |

Tiek ģenerēts dokuments pēc šāda algoritma :

1. Aizpilda galveni (header) pēc šādas tabulas informācijas :

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | templateId | [root] 1.3.6.1.4.1.38760.1.2.1.1  [extension] 1.3.6.1.4.1.38760.1.2.1.1.1 |
| 2 | TypeId | [root] 2.16.840.1.113883.1.3  [extension] POCD\_HD000040 |
| 3 | Id | [root] 1.3.6.1.4.1.38760.3.4.5.1  [extension] PREDA.[KeyWords].NewDocumentNo.versionNumber |
| 4 | Code | [code] 1.3.6.1.4.1.38760.1.2.1.1  [codeSystem] 1.3.6.1.4.1.38760.1.2.1  [displayName] = „Narkoloģijas reģistra diagnozes” |
| 5 | Title | „Pacienta pamatdatu kopsavilkuma dokuments no narkoloģijas reģistra”. |
| 6 | effectiveTime | Ģenerēšanas datums un laiks |
| 7 | confidentialityCode | „N” |
| 8 | languageCode | „LV” |
| 9 | setId | [root] 1.3.6.1.4.1.38760.3.4.5.4  [extension] PREDA.[KeyWords].NewDocumentNo |
| 10 | versionNumber | 1 |
| 11 | recordTarget.patientRole.id | [root] 1.3.6.1.4.1.38760.3.1.1  [extension] PersonasKods |
| 12 | author | author.time  [Value] = Ģenerēšanas datums un laiks  Author.assignedAuthor.id  [extension] = „Preda”  [root]=1.3.6.1.4.1.38760.3.4.6.1  Author.assignedAuthor.assignedPerson.name  [given]=PREDA  [family]=PREDA  Author.assignedAuthor.assignedPerson.repsresentedOrganization.id  [extension]=”PREDA”  [root]=”1.3.6.1.4.1.38760.3.4.6.1” |

Pamatdatu tipam veido sekciju (section), kuru aizpilda pēc atbilstošas tabulas:

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | Section | [code.code] = „1.3.6.1.4.1.38760.1.2.2.2”  [code.codeSystem] = „1.3.6.1.4.1.38760.2.21”  [code.codeSystemName] = „OID reģistrs”  [code.displayName] = „Pacienta pamatdatu diagnozes sekcija”  [title] = „Diagnozes”  [text] = „Pacienta diagnozes ir ” [PersonasKods] lauka attiecīgās kartes pēc lauka [SSKDgKods] informācija”, un diagnozes uzstādīšanas datums ir ‘[DgDatums]’ |

Pamatdatu tipam veido ierakstu (entry), kuru aizpilda pēc atbilstošas tabulas :

| Nr. | Elements | Apraksts | |
| --- | --- | --- | --- |
| 1 | Entry | [code.code] = „1.3.6.1.4.1.38760.1.2.3.2”  [code.codesystem] = „1.3.6.1.4.1.38760.2.21”  [code.codeSystemName] = „OID reģistrs”  [Code.displayname] = „Pacienta diagnoze”  [value.code] = „E11”  [value.displayName] = „Insulīnneatkarīgs cukura diabēts”  [value.codeSystem] = „1.3.6.1.4.1.38760.2.159”  [effectiveTime.value] = Ģenerēšanas datums un laiks |

Datu importa rezultātā tiek veidots CDA dokuments atbilstoši šablona identifikatoram „1.3.6.1.4.1.38760.1.2.1.1.1”, un reģistrēts sistēmā, izmantojot metodi „6.2.1 addDocument ”. Atbilstoši CDA dokumentu sekciju ierakstu apstrādes uzdevumam „6.2.12.7 documentEntryProcessing (dokumenta ierakstu apstrāde)”, katram diagnozes ierakstam reģistrētajā CDA dokumentā tiek izveidots atbilstošais diagnozes ieraksts pamatdatu tabulā „NPatientDiagnosis”, norādot gan datu autoru, gan datu avotu (CDA dokuments).

8.2‑16. tabula. Diagnozes datu kopa no narkoloģijas reģistra

| Nr. | Objekts | Lauks | Atbilstošais lauks reģistrā |
| --- | --- | --- | --- |
| 1. | Diagnoze | Pacienta ID | Nosaka pēc personas koda |
| 2. | Diagnoze | Diagnoze | Lauks NARKD.SSKDgKods |
| 3. | Diagnoze | Diagnozes datums | Lauks NARKD.DgDatums |

## NVD VIS

Identifikācija: VNCDataImport

**Apraksts:**

Sistēma veic vienreizēju datu importu personām, par kurām NVD VIS sistēmas turētāji ir sagatavojuši reģistra datus.

Ir nepieciešams sagatavot vienu datu kopu:

* NVD VIS Stacionāra taloni (datu kopas saīsinājums, kas tiks lietots importā, identificējot datu kopas tipu: vncst);

Sagatavojot datu kopu, ir jāņem vērā datu atlases princips:

* Datu kopas formāts XML atbilstoši datu shēmai „vncstShema.xsd”, piemērs: „vncstExample.xml”;
* Personai aizpildīts personas kods;
* Personas statuss nenorāda, ka persona ir mirusi;
* Personai eksistē vismaz viens Stacionārais talons.

Veicot datu ielādi, katrai personai, tiek:

* Nolasīta personas informācija no datu avota;
* Sagatavots CDA dokuments atbilstoši stacionāra kartes dokumenta modelim „1.3.6.1.4.1.38760.1.2.1.1.1”;
* CDA dokuments tiek sūtīts uz pakalpi’ „6.2.1 addDocument ”, kas nodrošina CDA dokumentu reģistrāciju un apstrādi.

Datu pirmreizējā importa nodrošināšanai tiek izveidota programmatūras vienība, kura aprakstīta „8.1 Sākotnējās datu ielādes komponente”, tā pati lietojumprogramma, kas nodrošina pirmreizējo datu ielādi no PREDA reģistriem.

Lietojumprogramma tiek darbināta izmantojot komandrindu norādot lietojumprogrammu, un programmas darbināšanas parametrus:

* „/source:<value>”: Datu avots – XML fails, kas satur importējamos datus;
* „/sourcetype:<value>”: Datu avota tips, kas identificē importējamo datu kopu;
* „/user:<value>”: lietotājs, ar ko tiks veikta autentifikācijas integrācijas platformā, un kas tiks uzskatīts par dokumenta autoru;

„/pwd:<value>”: lietotāja parole.

Datu ielāde jānodrošina pēc šāda scenārija:

* Apstrādā secīgi katru ierakstu, kas atrodas VNC sniegtajos datos;
* Katram ierakstam izpilda secīgās darbības:
* Katram iegūtajam stacionāra talonam tiek veidots CDA dokuments atbilstoši stacionāra kartes dokumenta modelim „1.3.6.1.4.1.38760.1.2.1.1.1”;
* Šis dokuments tiek nosūtīts uz EVK dokumentu repozitoriju, izmantojot EVK pakalpes metodi WS::addDocument.

Stacionārā talona datu kopa, kas jātransformē medicīniskā dokumenta CDA formātā:

| Nosaukums | Datu lauka numurs stacionāra kartē |
| --- | --- |
| Kartes numurs | 4 |
| Pielikums | 5 |
| Pacienta personas kods | 6 |
| Pacienta vārds, uzvārds | 7 |
| Saslimšanas sākuma datums | 15 |
| Nosūtījuma diagnoze | 16 |
| Nosūtītāja ārstniecības iestāde | 17 |
| Nosūtītāja ārstniecības persona | 18 |
| Personas grupa | 19 |
| Neatliekamās medicīniskās palīdzības periods | 20 |
| Pacienta kustība: kods | 21.1 -21.5 |
| Pacienta kustība: datums | 21.1 -21.5 |
| Pacienta kustība: pamata diagnoze | 21.1 -21.5 |
| Pacienta kustība: blakus diagnoze | 21.1 -21.5 |
| Pacienta kustība: sarežģījums | 21.1 -21.5 |
| Pacienta kustība: Ārsta vārds, uzvārds, personas kods | 21.1 -21.5 |
| Pacienta kustība: Patologanatomiskā pamata diagnoze | 21.6 |
| Pacienta kustība: Patologanatomiskā blakus diagnoze | 21.6 |
| Pacienta kustība: Patologanatomiskā diagnoze: sarežģījums | 21.6 |
| Pacienta kustība: Patologanatomiskā diagnoze: Ārsta vārds, uzvārds, personas kods | 21.6 |
| Anestēzijas kods | 22.1, 22.3 |
| Anestēzijas datums un laiks: no | 22.1, 22.3 |
| Anestēzijas datums un laiks: līdz | 22.1, 22.3 |
| Anesteziologa vārds, uzvārds, personas kods | 22.1, 22.3 |
| Operācijas nosaukums | 22.2, 22.4 |
| Operācijas kods | 22.2, 22.4 |
| Operācijas datums un laiks: no | 22.2, 22.4 |
| Operācijas datums un laiks: līdz | 22.2, 22.4 |
| Ķirurga vārds, uzvārds, personas kods | 22.2, 22.4 |
| Manipulācijas kods | 23,24 |
| Manipulāciju skaits | 23,24 |
| Manipulācijas datums | 23,24 |
| Ārstniecības personas kods, vārds, uzvārds, kura veikusi manipulāciju | 23,24 |
| Ārstniecības iestādes kods, nosaukums, kurā veikta manipulācija | 23,24 |
| Ārsta vārds, uzvārds, personas kods | 25 |

NVD VIS talonu informācija tiek uzkrāta kopš 2004.gada. Stacionāro talonu skaits vidēji ir 300 000 gadā.

## VDEĀVK IS

Identifikācija: VDEAVKDataImport

**Apraksts:**

Sistēma veic vienreizēju datu importu personām, par kurām VDEĀVK IS turētāji ir sagatavojuši personu, kurām ir piešķirta invaliditāte, sarakstu.

Veicot datu ielādi, katrai personai, tiek:

* Nolasīts personas kods no datu avota;
* Izpildīta procedūra getDisabilityData (skat. 6.8.1getDisabilityData (Iegūt personas invaliditātes datus)), kā ieejas parametru norādot personas kodu;

VDEĀVK IS ir reģistrētas invaliditātes grupas aptuveni 150 000 personām.

# Prasību trasējamība

## EVK IS 1.kārtas prasību trasējamība

EVK IS 1.kārtas prasību trasējamību skatīt 1.kārtas izstrādes dokumentos.

## EVK IS 2.kārtas prasību trasējamība

9.2‑1. tabula. EVK IS 2.kārtas prasību trasējamības tabula

|  |  |  |
| --- | --- | --- |
| Prasības ID | PPS prasība | Nosaukums |
| EVK-04 | FUN-00005 | Izveidot pacienta karti |
| EVK-04 | FUN-00010 | Iegūt pacienta karti |
| EVK-04 | FUN-00110 | Sapludināt pacientu kartes |
| EVK-04 | PDA-00010 | Personificētie dati |
| Validācijas | FUN-00235 | Validēt dokumentu |

# Pielikumi

## Lietotāju lomas un tiesības

### Ārstniecības persona

* EvkRghtDocumentAdd

1. EvkRghtDocumentCdaAdd
2. EvkRghtDocumentCcdAdd

* EvkRghtDocumentCommentAdd
* EvkRghtDocumentStatusSet
* EvkRghtDocumentsGet

1. EvkRghtDocumentsActualGet
2. EvkRghtDocumentCcdGet
3. EvkRghtDocumentCommentsGet
4. EvkRghtDocumentsAllGet

* EvkRghtPatientCardGet

1. EvkRghtPatientContactsGet
2. EvkRghtPatientContactPersonsGet
3. EvkRghtPatientEhicGet
4. EvkRghtPatientFamilyDoctorGet
5. EvkRghtPatientPersonDataGet

* EvkRghtPatientCardSet

1. EvkRghtPatientContactsSet
2. EvkRghtPatientContactPersonSet
3. EvkRghtDocumentAccessrightsSet

* EvkRghtPatientCardCreate
* EvkRghtPatientCardsMerge
* EvkRghtAccessRightsSet

1. EvkRghtDocumentAccessRightsSet

### Izmeklētājs

* EvkRghtDocumentsGet
* EvkRghtDocumentsActualGet
* EvkRghtDocumentsAllGet
* EvkRghtDocumentCcdGet
* EvkRghtDocumentCommentsGet
* EvkRghtPatientCardGet
* EvkRghtPatientContactsGet
* EvkRghtPatientContactPersonsGet
* EvkRghtPatientEhicGet
* EvkRghtPatientFamilyDoctorGet
* EvkRghtPatientDelegationsGet
* EvkRghtPatientPermissionsGet
* EvkRghtPatientPersonDataGet

### Pacients – P1

* EvkRghtDocumentsGet
* EvkRghtDocumentsActualGet
* EvkRghtDocumentCcdGet
* EvkRghtDocumentCommentsGet
* EvkRghtPatientCardGet
* EvkRghtPatientContactsGet
* EvkRghtPatientContactPersonsGet
* EvkRghtPatientEhicGet
* EvkRghtPatientFamilyDoctorGet
* EvkRghtPatientPersonDataGet
* EvkRghtProfileSettings
* EvkRghtClassifiers

### Pacients – P2

* EvkRghtDocumentsGet
* EvkRghtDocumentsActualGet
* EvkRghtDocumentCcdGet
* EvkRghtDocumentCommentsGet
* EvkRghtPatientCardGet
* EvkRghtPatientPersonDataGet
* EvkRghtPatientContactsGet
* EvkRghtPatientContactPersonsGet
* EvkRghtPatientEhicGet
* EvkRghtPatientFamilyDoctorGet
* EvkRghtPatientCardSet
* EvkRghtPatientContactsSet
* EvkRghtPatientContactPersonSet
* EvkRghtProfileSettings
* EvkRghtClassifiers

### Pacients – P3, P4, P5, P7

* EvkRghtDocumentCommentAdd
* EvkRghtDocumentsGet
* EvkRghtDocumentsActualGet
* EvkRghtDocumentCcdGet
* EvkRghtDocumentCommentsGet
* EvkRghtPatientCardGet
* EvkRghtPatientPersonDataGet
* EvkRghtPatientContactsGet
* EvkRghtPatientContactPersonsGet
* EvkRghtPatientEhicGet
* EvkRghtPatientFamilyDoctorGet
* EvkRghtPatientDelegationsGet
* EvkRghtPatientCardSet
* EvkRghtPatientContactsSet
* EvkRghtPatientContactPersonSet
* EvkRghtPatientDelegationSet
* EvkRghtPatientCardCreate
* EvkRghtAccessRightsSet
* EvkRghtPatientCardAccessrightsSet
* EvkRghtDocumentAccessrightsSet
* EvkRghtProfileSettings
* EvkRghtClassifiers

### Administrators

* EvkRghtPatientCardGet
* EvkRghtPatientPermissionsGet
* EvkRghtPatientCardSet
* EvkRghtPatientMajoritySet
* EvkRghtPatientPermissionSet
* EvkRghtDocumentTemplateSet
* EvkRghtDocumentTemplateGet