

|  |  |
| --- | --- |
|  |  |

Valsts informācijas sistēmu savietotāja, Latvijas valsts portāla www.latvija.lv un elektronisko pakalpojumu izstrāde un uzturēšana

3.daļa "VISS un Portāla jaunu un esošo moduļu papildinājumu izstrāde, ieviešana, garantijas apkalpošana un uzturēšana saskaņā ar tehnisko specifikāciju"

E-pakalpojuma izveidošanas ceļvedis Latvija.lv portālam

Programmētāja rokasgrāmata

VRAA-6\_15\_11\_58-VISS\_2010-LVP\_WZD\_2-PR

28.03.2014. versija 1.07

Rīgā 2014

|  |  |  |
| --- | --- | --- |
|  | **IEGULDĪJUMS TAVĀ NĀKOTNĒ** |  |

Dokumenta identifikācija

|  |  |
| --- | --- |
| Dokumenta ID: | VRAA-6\_15\_11\_58-VISS\_2010-LVP\_WZD\_2-PR-V1.07-28.03.2014. |
| Dokumenta nosaukums: | Valsts informācijas sistēmu savietotāja, Latvijas valsts portāla www.latvija.lv un elektronisko pakalpojumu izstrāde un uzturēšana.  3.daļa "VISS un Portāla jaunu un esošo moduļu papildinājumu izstrāde, ieviešana, garantijas apkalpošana un uzturēšana saskaņā ar tehnisko specifikāciju".  E-pakalpojuma izveidošanas ceļvedis Latvija.lv portālam.  Programmētāja rokasgrāmata. |
| Dokumenta kods: | VRAA-6\_15\_11\_58-VISS\_2010-LVP\_WZD\_2-PR |
| Versija: | Versija 1.07, Laidiens 28.03.2014. (saīsināti V1.07 28.03.2014.) |

Saskaņojumi

|  |  |  |  |
| --- | --- | --- | --- |
| Organizācija | Vārds, uzvārds, amats | Datums | Paraksts |
| Valsts reģionālās attīstības aģentūra | Atbildīgā persona no Pasūtītāja puses |  |  |
| SIA "ABC software" | J.Korņijenko, projekta vadītājs par tehniskiem jautājumiem no Izpildītāja puses | 28.03.2014. |  |
| SIA "ABC software" | S.Siksna, programmētājs | 28.03.2014. |  |
| SIA "ABC software" | M.Pētersons, projekta vadītājs par administratīviem jautājumiem no Izpildītāja puses | 28.03.2014. |  |
| SIA "ABC software" | E.Blumberga, projekta kvalitātes kontroles vadītāja | 28.03.2014. |  |

Izmaiņu vēsture

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Versija | Datums | Apraksts | Organizācija | Autors |
| 1.00 | 08.03.2013. | Izveidota dokumenta sākotnējā versija. | SIA „ABC software” | S.Siksna |
| 1.01 | 31.05.2013. | Dokuments atjaunots atbilstoši veiktajām izmaiņām e-pakalpojumos un komentāriem par iepriekšējo dokumenta versiju.  Pievienots uzstādīšanas pakas satura apraksts un komponentu shēmas attēls.  Papildināts biežāk sastopamo kļūdu nodalījums. | SIA „ABC software” | S.Siksna |
| 1.02 | 19.08.2013. | Dokuments atjaunots atbilstoši veiktajām izmaiņām e-pakalpojumos un komentāriem par iepriekšējo dokumenta versiju. | SIA „ABC software” | S.Siksna |
| 1.03 | 04.10.2013. | Dokuments atjaunots atbilstoši veiktajām izmaiņām e-pakalpojumos. | SIA „ABC software” | S.Siksna |
| 1.04 | 06.11.2013. | Precizēta maksājuma moduļa integrācijas instrukcija. | SIA „ABC software” | J.Korņijenko |
| 1.05 | 11.11.2013 | Dokuments atjaunots atbilstoši veiktajām izmaiņām e-pakalpojumos.  Pievienots formu (KDV, IDDV) apraksts. | SIA „ABC software” | S.Siksna |
| 1.06 | 24.01.2014. | Dokuments atjaunots atbilstoši veiktajām izmaiņām e-pakalpojumos (LvpAddressFinder, LvpPopup). | SIA „ABC software” | S.Siksna |
| 1.07 | 28.03.2014 | Dokuments atjaunots atbilstoši veiktajām izmaiņām e-pakalpojumos (LvpTerminateEservice, EDK).  Pievienots automātisko vides uzstādīšanas skriptu apraksts.  Pievienots integrācijas servisu un XML shēmu reģistrācijas VISS apraksts.  Pievienots emulatora autentifikācijas apraksts (tai skaitā juridiskai personai).  Papildināti problēmu novēršanas apraksti. | SIA „ABC software” | S.Siksna |

**Satura rādītājs**

[Attēlu saraksts 8](#_Toc383772081)

[1. Ievads 11](#_Toc383772082)

[1.1. Dokumenta nolūks 11](#_Toc383772083)

[1.2. Darbības sfēra 11](#_Toc383772084)

[1.3. Termini un pieņemtie apzīmējumi 11](#_Toc383772085)

[1.4. Saistība ar citiem dokumentiem 11](#_Toc383772086)

[1.5. Dokumenta pārskats 12](#_Toc383772087)

[2. Risinājuma arhitektūra 13](#_Toc383772088)

[2.1. Autentifikācija 14](#_Toc383772089)

[2.2. Maršrutēšanas serviss 14](#_Toc383772090)

[2.3. Pieprasījumu serviss 14](#_Toc383772091)

[2.4. Integrācijas servisi 16](#_Toc383772092)

[2.5. EDK serviss 16](#_Toc383772093)

[2.6. Konfigurācijas (e-pakalpojumu kataloga) serviss 16](#_Toc383772094)

[2.7. CDN 16](#_Toc383772095)

[2.8. Maksājumu modulis 17](#_Toc383772096)

[2.9. KDV un IDDV formas 17](#_Toc383772097)

[2.10. Emulators un izstrādes vide 18](#_Toc383772098)

[3. E-pakalpojumu piemēri 20](#_Toc383772099)

[3.1. Lvp.Eservice.Example.ComplexUI.Web 20](#_Toc383772100)

[3.2. Lvp.Eservice.Example.ServiceIntegration.Web 20](#_Toc383772101)

[3.3. Lvp.Eservice.Example.IsolationForDevelopment.Web 20](#_Toc383772102)

[3.4. Lvp.Eservice.Example.Infrastructure.Web 20](#_Toc383772103)

[3.5. Lvp.Eservice.Example.Globalization.Web 20](#_Toc383772104)

[3.6. Lvp.Eservice.Example.Sidebar.Web 21](#_Toc383772105)

[4. Izstrādes vides sagatavošana 22](#_Toc383772106)

[4.1. Izstrādes vides prasības 22](#_Toc383772107)

[4.2. Pakotnes saturs 22](#_Toc383772108)

[4.2.1. Lvp.Eservice.Template apakšpakotnes saturs 22](#_Toc383772109)

[4.2.2. Lvp.Eservice.Environment apakšpakotnes saturs 22](#_Toc383772110)

[4.2.2.1. Lvp.Eservice.Certificates apakšpakotnes saturs 22](#_Toc383772111)

[4.2.2.2. Lvp.Eservice.Examples apakšpakotnes saturs 23](#_Toc383772112)

[4.2.3. Setup apakšpakotnes saturs 23](#_Toc383772113)

[4.3. Izstrādes vides uzstādīšana 24](#_Toc383772114)

[4.4. Manuāla izstrādes vides uzstādīšana 24](#_Toc383772115)

[4.4.1. Portāla emulatora uzstādīšana 25](#_Toc383772116)

[4.4.1.1. Sertifikātu pārvaldības rīka atvēršana 25](#_Toc383772117)

[4.4.1.2. Saknes sertifikātu imports 27](#_Toc383772118)

[4.4.1.3. E-pakalpojumu IIS lietojuma (application) izveide 28](#_Toc383772119)

[4.4.1.4. Portāla emulatora apakšlietojuma izveidošana 35](#_Toc383772120)

[4.4.2. Maršrutēšanas servisa uzstādīšana 37](#_Toc383772121)

[4.4.2.1. IIS apakšlietojuma izveide 37](#_Toc383772122)

[4.4.2.2. Sertifikātu importēšana 38](#_Toc383772123)

[4.4.2.3. Žurnalēšanas mapju uzstādīšana 42](#_Toc383772124)

[4.4.3. E-pakalpojumu piemēru uzstādīšana 43](#_Toc383772125)

[4.4.3.1. IIS apakšlietojuma izveide 43](#_Toc383772126)

[4.4.3.2. Sertifikātu importēšana 43](#_Toc383772127)

[4.4.3.3. Žurnalēšanas mapju uzstādīšana 43](#_Toc383772128)

[4.5. Izstrādes vides atjaunošana 44](#_Toc383772129)

[4.6. Izstrādes vides noņemšana (atinstalēšana) 44](#_Toc383772130)

[5. Jauna e-pakalpojuma izveidošana 46](#_Toc383772131)

[5.1. Vienošanās par atbalstīto LVP integrāciju 46](#_Toc383772132)

[5.2. Priekšnosacījumi e-pakalpojuma integrācijai ar EPU vidi 46](#_Toc383772133)

[5.3. E-pakalpojuma projekta izveide un uzstādīšana 46](#_Toc383772134)

[5.4. Manuāla e-pakalpojuma projekta izveide un uzstādīšana 48](#_Toc383772135)

[5.4.1. E-pakalpojuma projekta izveide 48](#_Toc383772136)

[5.4.2. E-pakalpojuma apakšlietojuma izveide IIS 51](#_Toc383772137)

[5.4.3. E-pakalpojuma reģistrēšana emulatorā 51](#_Toc383772138)

[5.5. E-pakalpojuma sertifikāta reģistrēšana 51](#_Toc383772139)

[5.6. E-pakalpojuma īpašas piekļuves apgabala (realm) reģistrēšana 51](#_Toc383772140)

[5.7. Sagatavošanās EDK servisa izmantošanai 51](#_Toc383772141)

[5.8. Sagatavošanās e-paraksta izmantošanai 52](#_Toc383772142)

[5.9. Sagatavošanās lietotāja profila izmantošanai 52](#_Toc383772143)

[5.10. XML shēmas reģistrācijas VISS 52](#_Toc383772144)

[5.11. Integrācijas servisa reģistrācijas VISS 55](#_Toc383772145)

[6. E-pakalpojumu projekts 59](#_Toc383772146)

[6.1. Projekta struktūra 59](#_Toc383772147)

[6.2. Jauna soļa pievienošana 60](#_Toc383772148)

[6.3. IIS darbināta projekta atkļūdošana (debug) 60](#_Toc383772149)

[7. Bibliotēkas e-pakalpojumu izstrādei 62](#_Toc383772150)

[7.1. .NET bibliotēka Lvp.Eservice.FunctionsLib 62](#_Toc383772151)

[7.1.1. LvpContext.Config 62](#_Toc383772152)

[7.1.2. LvpContext.EdkService 63](#_Toc383772153)

[7.1.3. LvpContext.EsigningService 65](#_Toc383772154)

[7.1.4. LvpContext.EsignatureValidationService 66](#_Toc383772155)

[7.1.5. LvpContext.Globalization 67](#_Toc383772156)

[7.1.6. LvpContext.Log 68](#_Toc383772157)

[7.1.7. LogOptions 69](#_Toc383772158)

[7.1.8. LvpContext.OldPayments 70](#_Toc383772159)

[7.1.9. LvpContext.Payments 71](#_Toc383772160)

[7.1.10. LvpContext.RequestService 73](#_Toc383772161)

[7.1.11. RequestServiceException 76](#_Toc383772162)

[7.1.12. ErrorStructure 76](#_Toc383772163)

[7.1.13. LvpContext.User 77](#_Toc383772164)

[7.1.14. LvpContext.UserProfile 77](#_Toc383772165)

[7.1.15. Sesijas izmantošana 79](#_Toc383772166)

[7.1.16. IBodySerializer 80](#_Toc383772167)

[7.1.17. LvpWSFederationAuthenticationModule 80](#_Toc383772168)

[7.2. .NET bibliotēka Lvp.Eservice.Controls 80](#_Toc383772169)

[7.2.1. LvpWizard 81](#_Toc383772170)

[7.2.2. LvpWizardStepContainer 82](#_Toc383772171)

[7.2.3. LvpWizardStepBase 83](#_Toc383772172)

[7.2.4. LvpWizardMenu 85](#_Toc383772173)

[7.2.5. LvpFooter 86](#_Toc383772174)

[7.2.6. LvpErrorReport 87](#_Toc383772175)

[7.2.7. LvpTerminateEservice 87](#_Toc383772176)

[7.2.8. LvpActiveStepManual 88](#_Toc383772177)

[7.2.9. LvpDatePicker 89](#_Toc383772178)

[7.2.10. LvpDateTimePicker 90](#_Toc383772179)

[7.2.11. LvpDatePickerValidator 91](#_Toc383772180)

[7.2.12. Validatori 91](#_Toc383772181)

[7.2.13. LvpPersonCodeValidator 91](#_Toc383772182)

[7.2.14. LvpDisplayState 92](#_Toc383772183)

[7.2.15. LvpMessageBox 92](#_Toc383772184)

[7.2.16. LvpTooltip 93](#_Toc383772185)

[7.2.17. LvpLink 93](#_Toc383772186)

[7.2.18. LvpResolutionLink 95](#_Toc383772187)

[7.2.19. LvpScript 95](#_Toc383772188)

[7.2.20. LvpEsigner 96](#_Toc383772189)

[7.2.21. LvpAddressFinder 97](#_Toc383772190)

[7.2.22. LvpPopup 99](#_Toc383772191)

[7.2.23. LvpExpander 100](#_Toc383772192)

[7.2.24. LvpWizardPageBase 100](#_Toc383772193)

[7.2.25. LvpAboutPage 100](#_Toc383772194)

[7.2.26. LvpAboutPageBase 101](#_Toc383772195)

[7.2.27. LvpUserManualPageBase 101](#_Toc383772196)

[7.2.28. LvpMasterPageBase 101](#_Toc383772197)

[7.2.29. SessionKeepAlive 102](#_Toc383772198)

[7.2.30. WrapperPage 102](#_Toc383772199)

[7.2.31. ErrorPage 102](#_Toc383772200)

[7.2.32. LvpWsFederationRequestValidator 102](#_Toc383772201)

[7.3. JavaScript bibliotēkas 103](#_Toc383772202)

[7.3.1. lvp.portalProxy 104](#_Toc383772203)

[7.3.2. lvp.controlStyle 105](#_Toc383772204)

[7.3.3. lvp.displayState 106](#_Toc383772205)

[7.3.4. lvp.textZoom 107](#_Toc383772206)

[8. E-pakalpojumu konfigurācija 108](#_Toc383772207)

[8.1. lvp/general sekcija 108](#_Toc383772208)

[8.2. lvp/esigner sekcija 108](#_Toc383772209)

[8.3. lvp/addressFinder sekcija 109](#_Toc383772210)

[8.4. lvp/oldPayment sekcija 110](#_Toc383772211)

[8.5. lvp/payment sekcija 111](#_Toc383772212)

[8.6. lvp/errorReport sekcija 112](#_Toc383772213)

[8.7. lvp/messages/message sekcija 112](#_Toc383772214)

[8.8. lvp/nasa sekcija 113](#_Toc383772215)

[8.9. EserviceConfiguration galapunkta sekcija 113](#_Toc383772216)

[9. Portāla emulatora lietotāja ceļvedis 115](#_Toc383772217)

[9.1. E-pakalpojumu saraksta lapa 115](#_Toc383772218)

[9.2. Autentifikācija 117](#_Toc383772219)

[9.3. E-pakalpojuma darbināšanas lapa 119](#_Toc383772220)

[9.4. E-pakalpojuma beigu lapa 120](#_Toc383772221)

[9.5. E-pakalpojuma kļūdu lapa 121](#_Toc383772222)

[9.6. E-pakalpojuma lietotāja instrukcijas lapa 121](#_Toc383772223)

[10. Biežāk sastopamās problēmas un to risinājumi 123](#_Toc383772224)

# Attēlu saraksts

[1.attēls. E-pakalpojuma ielāde 13](#_Toc383772225)

[2.attēls. E-pakalpojuma integrācijas shēma 13](#_Toc383772226)

[3.attēls. Anonīms servisu izsaukums 15](#_Toc383772227)

[4.attēls. Autentificēts (drošs) servisu izsaukums 15](#_Toc383772228)

[5.attēls. Sarežģīta pakalpojuma izpildes plūsmas piemērs 18](#_Toc383772229)

[6.attēls. E-pakalpojumu izstrādes vide 19](#_Toc383772230)

[7.attēls. Veiksmīgas skripta izpildes rezultāts 24](#_Toc383772231)

[8.attēls. MMC palaišana 25](#_Toc383772232)

[9.attēls. Sertifikātu spraudņa pievienošana (1) 25](#_Toc383772233)

[10.attēls. Sertifikātu spraudņa pievienošana (2) 25](#_Toc383772234)

[11.attēls. Sertifikātu spraudņa pievienošana (3) 26](#_Toc383772235)

[12.attēls. Sertifikātu spraudņa pievienošana (4) 26](#_Toc383772236)

[13.attēls. Saknes sertifikāta imports 27](#_Toc383772237)

[14.attēls. Saknes sertifikāta importa vednis (2.solis) 27](#_Toc383772238)

[15.attēls. Saknes sertifikāta importa vednis (3.solis) 28](#_Toc383772239)

[16.attēls. Hosts datnes labošana 28](#_Toc383772240)

[17.attēls. IIS Manager palaišana 29](#_Toc383772241)

[18.attēls. Sertifikātu pārvaldības atvēršana 29](#_Toc383772242)

[19.attēls. Sertifikātu importa uzsākšana 30](#_Toc383772243)

[20.attēls. Sertifikāta imports 30](#_Toc383772244)

[21.attēls. Sertifikāts veiksmīgi importēts 31](#_Toc383772245)

[22.attēls. Jauna lietojuma pievienošanas uzsākšana 31](#_Toc383772246)

[23.attēls. Lietojuma parametri 32](#_Toc383772247)

[24.attēls. Lietojuma pūla atvēršana 32](#_Toc383772248)

[25.attēls. Lietojuma pūla parametri 33](#_Toc383772249)

[26.attēls. Pūla lietotājs 33](#_Toc383772250)

[27.attēls. Saistījumu konfigurācijas atvēršana 34](#_Toc383772251)

[28. attēls. Jauna saistījuma pievienošanas uzsākšana 34](#_Toc383772252)

[29.attēls. Saistījuma parametru ievade 34](#_Toc383772253)

[30. attēls. Jauna saistījuma pievienošanas uzsākšana 35](#_Toc383772254)

[31.attēls. Saistījuma parametru ievade 35](#_Toc383772255)

[32.attēls. Nepieciešamie eservices lietojuma saistījumi 35](#_Toc383772256)

[33.attēls. Apakšlietojuma izveides uzsākšana 36](#_Toc383772257)

[34.attēls. Apakšlietojuma parametri 36](#_Toc383772258)

[35.attēls. Apakšlietojuma parametri 37](#_Toc383772259)

[36.attēls. Apakšlietojuma papildu iestatījumu atvēršana 38](#_Toc383772260)

[37.attēls. net.pipe protokola iespējošana 38](#_Toc383772261)

[38.attēls. Sertifikāta imports 39](#_Toc383772262)

[39.attēls. Sertifikāta importa vednis (2.solis) 40](#_Toc383772263)

[40.attēls. Sertifikāta importa vednis (3.solis) 40](#_Toc383772264)

[41.attēls. Sertifikāta importa vednis (4.solis) 41](#_Toc383772265)

[42.attēls. Sertifikāta privāto atslēgu tiesību pārvaldības atvēršana 41](#_Toc383772266)

[43.attēls. Sertifikāta privāto atslēgu tiesību pārvaldība 42](#_Toc383772267)

[44.attēls. Veiksmīgas skripta izpildes rezultāts 44](#_Toc383772268)

[45.attēls. Veiksmīgas skripta izpildes rezultāts 45](#_Toc383772269)

[46.attēls. E-pakalpojuma numura ievade 47](#_Toc383772270)

[47.attēls. Jauno e-pakalpojumu saturošās mapes ievade 47](#_Toc383772271)

[48.attēls. Veiksmīgas skripta izpildes rezultāts 47](#_Toc383772272)

[49.attēls. Jaunā e-pakalpojuma lapa 47](#_Toc383772273)

[50.attēls. Jaunā e-pakalpojuma lapa 48](#_Toc383772274)

[51.attēls. E-pakalpojuma numura aizstāšana 49](#_Toc383772275)

[52.attēls. E-pakalpojuma numura aizstāšana projekta iestatījumos 49](#_Toc383772276)

[53.attēls. E-pakalpojuma GUID maiņa 50](#_Toc383772277)

[54.attēls. GUID gen rīka izmantošana 50](#_Toc383772278)

[55.attēls. Jaunas XSD shēmas pievienošana 52](#_Toc383772279)

[56.attēls. Resursa datnes ielāde 53](#_Toc383772280)

[57.attēls. XSD shēmas tipa izvēle 53](#_Toc383772281)

[58.attēls. XSD shēmas metadatu apraksta pievienošana 54](#_Toc383772282)

[59.attēls. XSD shēmas metadatu apskats 54](#_Toc383772283)

[60.attēls. Pieslēgšanās VISS sistēmai 55](#_Toc383772284)

[61.attēls. Pieslēgšanās VISS sistēmai 55](#_Toc383772285)

[62.attēls. Pieslēgšanās VISS sistēmai 56](#_Toc383772286)

[63.attēls. Jauna IS servisa pievienošanas uzsākšana 56](#_Toc383772287)

[64.attēls. Servisa datu ievade 57](#_Toc383772288)

[65.attēls. Servisa adreses ievade 58](#_Toc383772289)

[66.attēls. E-pakalpojuma projekta struktūra 59](#_Toc383772290)

[67.attēls. Pieslēgšanās dialoga atvēršana 60](#_Toc383772291)

[68.attēls. Pieslēgšanās IIS procesam 61](#_Toc383772292)

[69.attēls. Bibliotēkas e-pakalpojumu izstrādei 62](#_Toc383772293)

[70.attēls. LvpWizard konteiners 81](#_Toc383772294)

[71.attēls. LvpWizardMenu 85](#_Toc383772295)

[72.attēls. LvpFooter 86](#_Toc383772296)

[73.attēls. LvpErrorReport 87](#_Toc383772297)

[74.attēls. LvpTerminateEservice 87](#_Toc383772298)

[75.attēls. LvpActiveStepManual 88](#_Toc383772299)

[76.attēls. LvpDatePicker 89](#_Toc383772300)

[77.attēls. LvpDateTimePicker 90](#_Toc383772301)

[78.attēls. LVP validatora darbības piemērs 91](#_Toc383772302)

[79.attēls. LvpMessageBox 92](#_Toc383772303)

[80.attēls. LvpTooltip 93](#_Toc383772304)

[81.attēls. LvpEsigner 96](#_Toc383772305)

[82.attēls. LvpAddressFinder 97](#_Toc383772306)

[83.attēls. LvpPopup 99](#_Toc383772307)

[84.attēls. LvpExpander 100](#_Toc383772308)

[85.attēls. LvpAboutPage 101](#_Toc383772309)

[86.attēls. Iframe struktūra e-pakalpojuma un LVP integrācijai 102](#_Toc383772310)

[87.attēls. Piekļuve LVP JS bibliotēkām 103](#_Toc383772311)

[88.attēls. E-pakalpojums ar un bez lvp.controlStyle izmantošanas 106](#_Toc383772312)

[89.attēls. Emulatora sākuma lapa 115](#_Toc383772313)

[90.attēls. Autentifikācijas loga atvēršana 117](#_Toc383772314)

[91.attēls. Autentifikācijas loga atvēršana 118](#_Toc383772315)

[92.attēls. Pārstāvētā uzņēmuma izvēle 118](#_Toc383772316)

[93.attēls. Informācija par autentificēto lietotāju 119](#_Toc383772317)

[94.attēls. Emulatorā atvērts e-pakalpojums 120](#_Toc383772318)

[95.attēls. E-pakalpojuma noslēguma lapa 120](#_Toc383772319)

[96.attēls. E-pakalpojuma neapstrādātas kļūdas lapa 121](#_Toc383772320)

[97.attēls. Lietotāja instrukcijas atvēršana 121](#_Toc383772321)

[98.attēls. Emulatorā atvērta e-pakalpojuma lietotāja instrukcija 122](#_Toc383772322)

[99.attēls. Servisu ziņojumu žurnāla fragments 123](#_Toc383772323)

[100.attēls. Emulatora autentifikācijas žurnāla fragments 124](#_Toc383772324)

[101.attēls. Kļūda, atverot autentifikācijas lapu 126](#_Toc383772325)

[102.attēls. Autentifikācijas iestatījumu atvēršana 127](#_Toc383772326)

[103.attēls. Autentifikācijas iespēju saraksts 127](#_Toc383772327)

# Ievads

Viena no Latvija.lv portāla (turpmāk LVP) funkcijām ir piedāvāt iedzīvotājiem e-pakalpojumus. E-pakalpojumus izstrādā kā patstāvīgus lietojumus, kurus integrē LVP vidē.

## Dokumenta nolūks

Dokumenta nolūks ir iepazīstināt izstrādātājus ar LVP e-pakalpojumu veidošanas principiem, integrāciju ar LVP, piedāvātajām bibliotēkām, kā arī nodrošināt citu e-pakalpojumu izstrādes uzsākšanai nepieciešamo informāciju.

Dokumentā iekļauts arī izstrādes vides apraksts un tās uzstādīšanas instrukcija.

## Darbības sfēra

LVP ir publiska vietne e-pakalpojumu izmitināšanai. Šis dokuments paredzēts e-pakalpojumu izstrādātājiem.

## Termini un pieņemtie apzīmējumi

Apzīmējumu un terminu vārdnīca pieejama dokumentā [1].

Šī dokumenta ietvaros termins CDN tiek lietots, lai aprakstītu tīmekļa lietojumprogrammu, kuras vienīgais uzdevums ir nodrošināt resursus citām lietojumprogrammām (attēli, JS un CSS datnes).

## Saistība ar citiem dokumentiem

Dokuments ir izstrādāts, balstoties uz šādiem dokumentiem:

1. „Valsts informācijas sistēmu savietotāja, Latvijas valsts portāla www.latvija.lv un elektronisko pakalpojumu izstrāde un uzturēšana”. Iepirkuma priekšmeta 3.daļa - VISS un portāla jaunu un esošo moduļu papildinājumu izstrāde, ieviešana, garantijas apkalpošana un uzturēšana saskaņā ar tehnisko specifikāciju. Terminu un saīsinājumu indekss. (VRAA-6\_15\_11\_58-VISS\_2010-TSI).
2. Publiskās pārvaldes dokumentu pārvaldības sistēmu integrācijas vides izveide. Programmatūras projektējuma apraksts (E-Parakstītāja ārējās saskarnes). (VRAA.DIV.EPAR.PPA.ASP.1.1.0.2013.02.15).
3. „Valsts informācijas sistēmu savietotāja, Latvijas valsts portāla www.latvija.lv un elektronisko pakalpojumu izstrāde un uzturēšana”. Iepirkuma priekšmeta 3.daļa - VISS un portāla jaunu un esošo moduļu papildinājumu izstrāde, ieviešana, garantijas apkalpošana un uzturēšana saskaņā ar tehnisko specifikāciju. Elektronisko dokumentu krātuve: ārējās saskarnes. Programmatūras projektējuma apraksts. (VRAA-6\_15\_11\_58-VISS\_2010-EDK\_AS-PPA).
4. Par Valsts informācijas sistēmu savietotāja, Latvijas valsts portāla www.latvija.lv un elektronisko pakalpojumu izstrāde un uzturēšana. 3.daļa „VISS un Portāla jaunu un esošo moduļu papildinājumu izstrāde, ieviešana, garantijas apkalpošana un uzturēšana saskaņā ar tehnisko specifikāciju”. E-pakalpojumu arhitektūras izstrāde. Vadlīnijas. (VRAA-6\_15\_11\_58-VISS\_2010-EPAK\_ARH-VDL).
5. Par Valsts informācijas sistēmu savietotāja, Latvijas valsts portāla www.latvija.lv un elektronisko pakalpojumu izstrāde un uzturēšana. 3.daļa „VISS un Portāla jaunu un esošo moduļu papildinājumu izstrāde, ieviešana, garantijas apkalpošana un uzturēšana saskaņā ar tehnisko specifikāciju”. KDV formu izveidošanas ceļvedis Latvija.lv portālam. Programmētāja rokasgrāmata. (VRAA-6\_15\_11\_58-VISS\_2010-KDV\_FORM-PR).
6. Par Valsts informācijas sistēmu savietotāja, Latvijas valsts portāla www.latvija.lv un elektronisko pakalpojumu izstrāde un uzturēšana. 3.daļa „VISS un Portāla jaunu un esošo moduļu papildinājumu izstrāde, ieviešana, garantijas apkalpošana un uzturēšana saskaņā ar tehnisko specifikāciju”. IDDV formu izveidošanas ceļvedis VISS portālam. Programmētāja rokasgrāmata. (VRAA-6\_15\_11\_58-VISS\_2010-IDDV\_FORM-PR).
7. „Valsts informācijas sistēmu savietotāja, Latvijas valsts portāla www.latvija.lv un elektronisko pakalpojumu izstrāde un uzturēšana”. Iepirkuma priekšmeta 3.daļa - VISS un portāla jaunu un esošo moduļu papildinājumu izstrāde, ieviešana, garantijas apkalpošana un uzturēšana saskaņā ar tehnisko specifikāciju. Adrešu meklēšanas komponente. Programmatūras prasību specifikācija. (VRAA-6\_15\_11\_58-VISS\_2010-AMK-PPS).

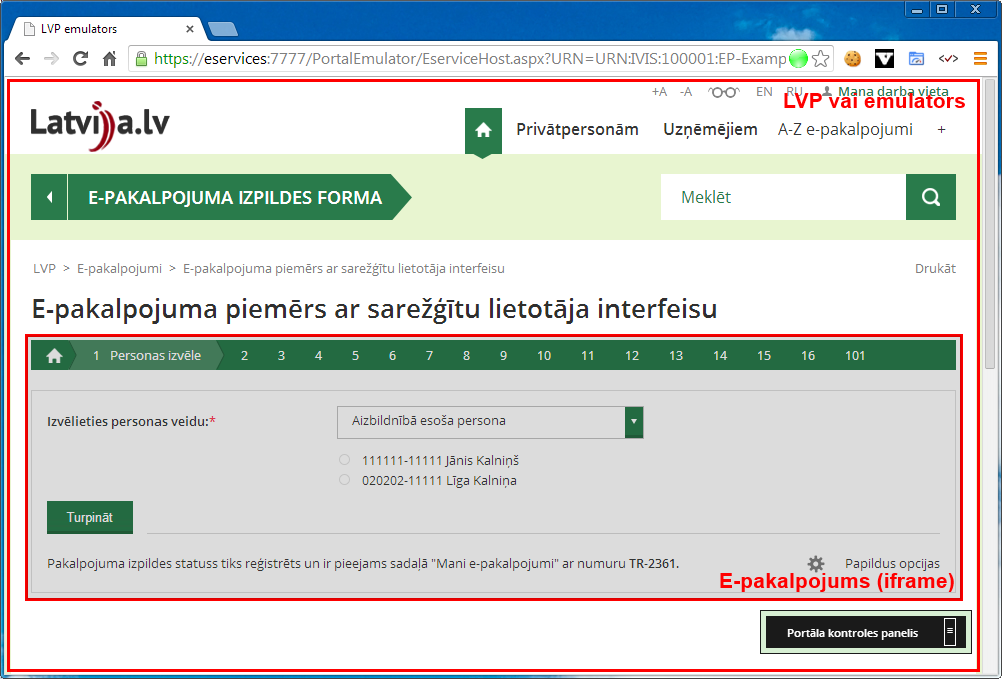
## Dokumenta pārskats

Dokumentu veido šādi nodalījumi:

* „Ievads” – iekļauta informācija par dokumenta vispārējo struktūru, darbības sfēru, nolūku, dokumentā izmantotajiem terminiem un apzīmējumiem, kā arī par saistību ar citiem dokumentiem.
* „Risinājuma arhitektūra” – vispārīgs apraksts par e-pakalpojumu integrāciju ar LVP. Sākumpunkts izstrādātājiem, lai saprastu kopainu.
* „E-pakalpojumu piemēri” – piedāvāto e-pakalpojumu piemēru apraksts.
* „Izstrādes vides sagatavošana” – instrukcija e-pakalpojuma izstrādes vides sagatavošanai.
* „Jauna e-pakalpojuma izveidošana” – instrukcija jauna e-pakalpojuma izstrādes uzsākšanai, izmantojot piedāvāto sagatavi.
* „E-pakalpojumu projekts” – e-pakalpojuma projekta šablona apraksts.
* „Bibliotēkas e-pakalpojumu izstrādei” – izstrādātājiem piedāvāto komponenšu apraksts.
* „E-pakalpojumu konfigurācija” – e-pakalpojuma konfigurācijas sekciju apraksts.
* „Portāla emulatora lietotāja ceļvedis” – ceļvedis emulatoram, kas izstrādes vidē imitē LVP darbību.
* „Biežāk sastopamās problēmas un to risinājumi” – bieži novērotu problēmu simptomi, diagnozes un ārstēšana.

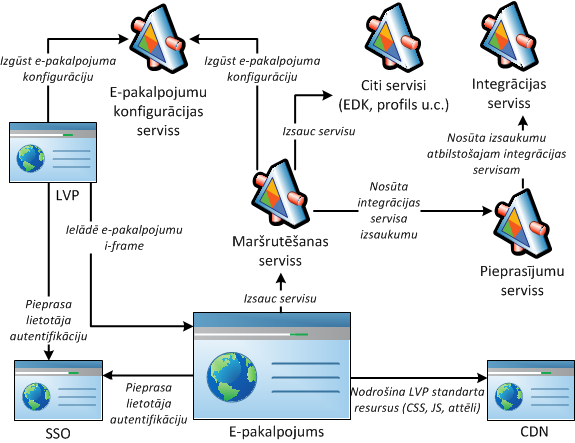
# Risinājuma arhitektūra

E-pakalpojumi tiek darbināti kā atsevišķi ASP.NET tīmekļa lietojumi. Integrācija ar portālu notiek, ielādējot e-pakalpojuma lietojumu *iframe* elementā.



1.attēls. E-pakalpojuma ielāde

E-pakalpojumu infrastruktūra ir redzama 2.attēlā.



2.attēls. E-pakalpojuma integrācijas shēma

## Autentifikācija

Autentifikācijai tiek izmantots SSO. Ja lietotājs ir ielogojies LVP, atverot e-pakalpojumu, notiek automātiska autentifikācija arī e-pakalpojuma lietojumā.

Savukārt, ja lietotājs nav ielogojies LVP, pie e-pakalpojuma atvēršanas caur konfigurācijas servisu tiek noskaidrots, vai e-pakalpojums ir publisks. Ja e-pakalpojums nav publisks, lietotājam tiek attēlots autentifikācijas logs un e-pakalpojums netiek atvērts.

## Maršrutēšanas serviss

Saziņu ar servisiem e-pakalpojums veic caur maršrutēšanas servisu. Maršrutēšanas serviss saņem ziņojumu un atkarībā no izsauktā maršrutēšanas servisa galapunkta (*endpoint*) nosūta to tālāk uz atbilstošo servisu. Nepieciešamības gadījumā tiek veikta lietotāja drošības talona apmaiņa pret servisa izsaukšanai nepieciešamo talonu. Tā kā vienu maršrutēšanas servisu izmanto vairāki e-pakalpojumi, tiek samazināta servisu galapunktu konfigurācijas dublēšanās (galapunktu konfigurācija glabājas maršrutēšanas servisā, nevis katra e-pakalpojuma konfigurācijā), kā arī tiek uzlabota lietotāja drošības talonu kešdarbe (*caching*).

Autentificēta izsaukuma gadījumā maršrutēšanas servisam ir nepieciešams sertifikāts (skat. 4.4.2.2.paragrāfu), lai autentificētos STS.

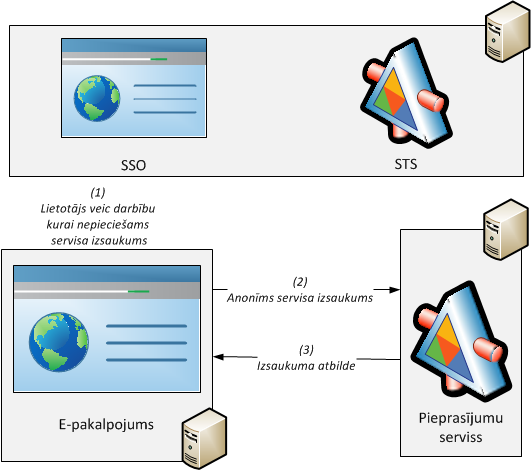
## Pieprasījumu serviss

Saziņa ar integrācijas servisiem notiek caur pieprasījumu servisu (turpmāk PS). Pieprasījumu serviss saņem ziņojumu un pārsūta uz norādītajam galamērķim atbilstošo integrācijas servisu. PS pārbauda ziņojuma struktūras atbilstību pret XSD shēmu, kā arī veic žurnalēšanu.

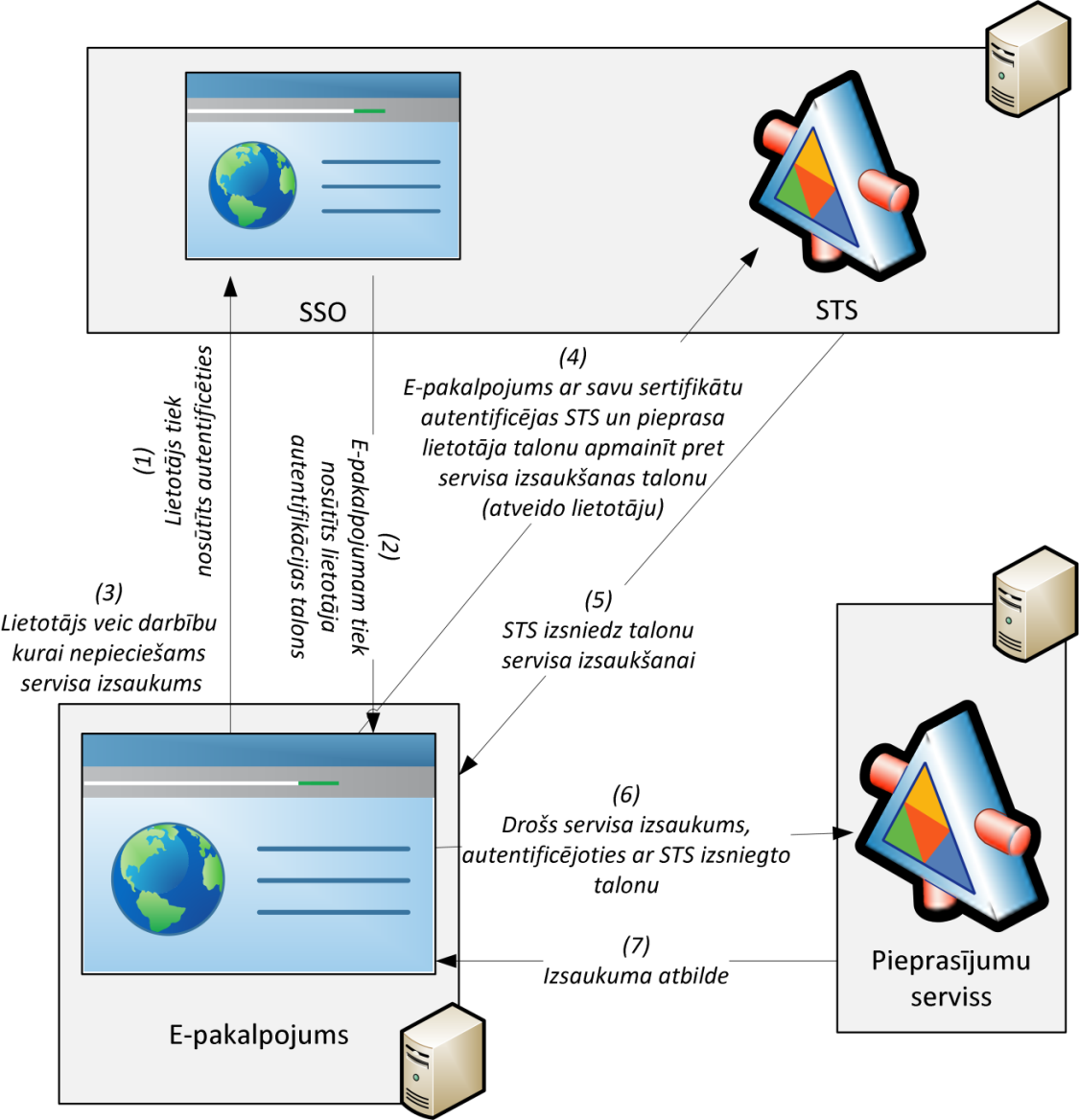
PS izsaukšanai no drošības viedokļa pieejami divi veidi:

* Anonīma izsaukšana (lietojama gadījumos, kad e-pakalpojuma lietotājs nav autentificējies);
* Autentificēta (droša) izsaukšana (izsaucot PS tiek atveidots (impersonalized) e-pakalpojuma lietotājs (izmantota lietotāja identitāte)).

Abu izsaukumu shēmas apskatāmas 3. un 4.attēlos. Lai nesarežģītu struktūru attēlos nav iekļauts maršrutēšanas serviss (tas darbojas kā e-pakalpojuma sastāvdaļa).



3.attēls. Anonīms servisu izsaukums



4.attēls. Autentificēts (drošs) servisu izsaukums

Pirms integrācijas servisu izsaukšanas (Submit, SubmitSync), nepieciešams uzsākt transakciju (StartTransaction). Vienā e-pakalpojuma izpildes reizē (viens lietotājs atver un izpilda e-pakalpojumu) jāveido **viena** transakcija. E-pakalpojuma beigās transakcija jābeidz (EndTransaction). Ja e-pakalpojums ir asinhrons (darbība paredz pāreju uz KDV, IDDV formām vai maksājumu moduli), transakcija jābeidz loģiskajā e-pakalpojuma noslēgumā (pēdējā KDV vai IDDV forma). Kopumā transakcija paredzēta, lai varētu atsekot lietotāja veiktās darbības (integrācijas servisu izsaukumi) viena e-pakalpojuma izpildes laikā.

## Integrācijas servisi

Integrācijas servisi nodrošina integrāciju ar e-pakalpojumu izmantotajām aizmugursistēmām (backend). Parasti integrācijas servisi tiek izstrādāti kopā ar e-pakalpojumiem, kas tos izmanto. Integrācijas servisu izsaukšana notiek caur PS.

## EDK serviss

Elektronisko dokumentu krātuves serviss (skat. [3]) nodrošina datņu glabāšanas un pārsūtīšanas funkcionalitāti starp e-pakalpojumiem un to aizmugursistēmām (back-end). PS nav piemērots šai funkcijai, jo neatbalsta lielus ziņojumu izmērus. Tādēļ datnes jāielādē EDK un caur PS jāsūta tikai EDK dokumenta identifikators.

E-pakalpojumiem pieeja tiek atļauta EDK objektiem, kas atrodas autentificētā lietotāja mapēs (skat. 7.1.2.sadaļu). Anonīma lietotāja gadījumā EDK nav pieejams.

## Konfigurācijas (e-pakalpojumu kataloga) serviss

Emulatorā vai e-pakalpojumu katalogā ievadīto informāciju e-pakalpojums saņem caur konfigurācijas servisu. E-pakalpojumu izstrādātājiem šis serviss nav tieši jāizmanto, tomēr ir vērts zināt par tā darbības principiem, potenciālo infrastruktūras problēmu risināšanai.

Konfigurācija tiek izgūta no minētā servisa un glabāta kešatmiņā. Ja kešatmiņā minētā informācija nav, tā tiek izgūta no servisa un saglabāta.

E-pakalpojuma lietojumam uzsākot darbu, notiek pierakstīšanās uz konfigurācijas izmaiņu saņemšanu (NASA) un izmaiņu klausīšanās servisu inicializācija. Pēc konfigurācijas izmaiņu veikšanas e-pakalpojumu katalogā NASA serviss izsauc e-pakalpojumu (paziņo par izmaiņu faktu). Saņemot izsaukumu, pakalpojums iztīra konfigurāciju no kešatmiņas.

## CDN

Portāla standarta resursi (attēli, JS un CSS datnes) e-pakalpojumam tiek piegādāti, izmantojot atsevišķu tīmekļa lietojumu (CDN).

Izstrādes videi šis lietojums ir iebūvēts emulatorā.

CDN paredzēts e-pakalpojumu koplietojamo (atkārtoti izmantojamo) resursu glabāšanai. Šāds risinājums samazina koda dublēšanu, ļauj veikt resursu optimizāciju (*minification*), kā arī uzlabo tīmekļa pārlūka kešdarbes iespējas.

E-pakalpojumu specifiskos resursus paredzēts glabāt pašā e-pakalpojumā.

CDN satur vairākas resursu versijas. Parasti ar jaunas e-pakalpojumu izstrādātāja pakotnes versijas piegādi tiek izveidota arī jauna CDN resursu versija. Piegādātās CDN versijas netiek labotas, lai izvairītos no problēmām ar pārlūku kešdarbi. Nepieciešamie labojumi un uzlabojumi tiek veikti jaunā (nākošajā) versijā. Izstrādājot e-pakalpojumu, jāizmanto pēdējā versija.

Ir divi versiju zari:

* **0.X**: paredzēts administratīvi migrētajiem e-pakalpojumiem;
* **1.X**: paredzēts no jauna izstrādātajiem vai pilnībā migrētiem e-pakalpojumiem.

Pašlaik ir pieejamas šādas CDN resursu versijas:

* **0.4**: janvāra piegādes versija administratīvi migrētajiem pakalpojumiem;
* **0.5**: marta piegādes versija administratīvi migrētajiem pakalpojumiem;
* **1.3**: janvāra piegādes versija;
* **1.4**: marta piegādes versija.

Ar laiku vecās versijas, kuras e-pakalpojumi vairs neizmanto, tiek dzēstas.

## Maksājumu modulis

Maksājumu modulis (turpmāk MM, detalizēti sk. [4] dokumentā) nodrošina LVP lietotājiem iespēju veikt maksājumus e-pakalpojumu ietvaros.

Ir sinhronais un asinhronais maksājumu veikšanas režīms.

Sinhronā maksājuma veikšanas soļi:

* Caur PS uz MM tiek nosūtīta maksājumu informācija kopā ar e-pakalpojuma specifiskiem datiem. MM atgriež maksājuma identifikatoru.
* E-pakalpojumu ceļvedis nodrošina, ka lietotājs tiek pārvirzīts uz maksājumu lapu, padodot līdzi maksājuma identifikatoru.
* Lietotājs veic maksājumu.
* Pirmajā solī izveidotais pieprasījums tiek pārsūtīts tālāk uz norādīto integrācijas servisu (ResponseEndPoint).
* Lietotājs tiek pāradresēts uz sākotnējā pieprasījumā norādītu adresi. Šai adresei jānorāda vai nu uz konstantu (nemainīgu) LVP lapu ar informāciju par veiksmīgu maksājuma norisi vai arī uz LVP KDV sadaļu[[1]](#footnote-1).

Maksājuma veikšana ir e-pakalpojuma pēdējais solis un pēc maksājuma veikšanas nevajadzētu notikt pārejai atpakaļ uz e-pakalpojumu.

Ar maksājumu saistītā konfigurācija (maksājuma summa, valūta, vienību skaits utt. – skat. 8.5. nodaļu) ir jāsinhronizē ar MM PPK (publisko pakalpojumu katalogs) esošo konfigurāciju. Lai to veiktu, jāsazinās ar EPU (VRAA).

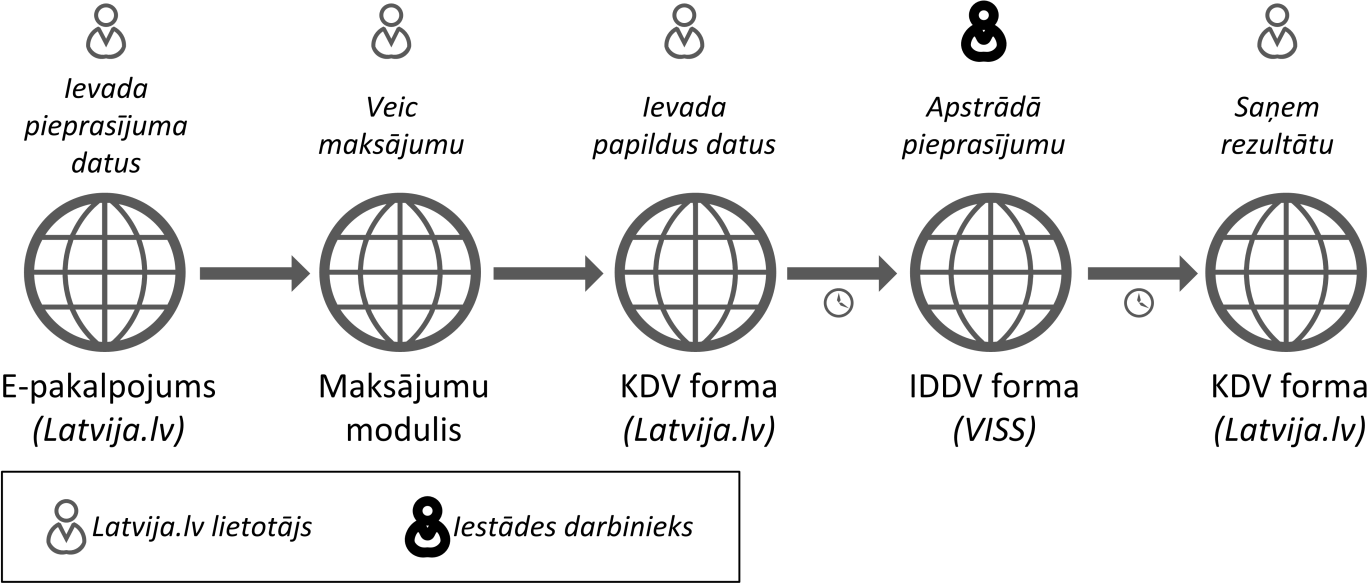
## KDV un IDDV formas

Asinhrona pakalpojuma gadījumā pēc asinhrono darbību veikšanas ieteicams atgriezties nevis uz e-pakalpojumu, bet gan uz formām. Ir šādi formu veidi:

* Klienta darba vietas (KDV) forma – pieejama latvija.lv portālā lietotājam no viņa darba vietas, kad ir saņemts atbilstošs lietotājam adresēts ziņojums.
* Iestādes darbinieka darba vietas (IDDV) forma – pieejama VISS portālā iestāžu darbiniekam, kad saņemts atbilstošs darbiniekam adresēts ziņojums.

Formas līdzīgi e-pakalpojumiem tiek izstrādātas kā atsevišķi lietojumi (skat. [5] un [6] dokumentus).

Zemāk redzams ļoti sarežģīta potenciālā e-pakalpojuma izpildes gaitas piemērs:



5.attēls. Sarežģīta pakalpojuma izpildes plūsmas piemērs

## Emulators un izstrādes vide

E-pakalpojumu izstrādātājiem tiek piedāvāts LVP emulators (skat. 4.2.nodaļu), kas e-pakalpojumam nodrošina tādu pat interfeisu, kāds tiks izmantots saziņai ar īsto LVP.

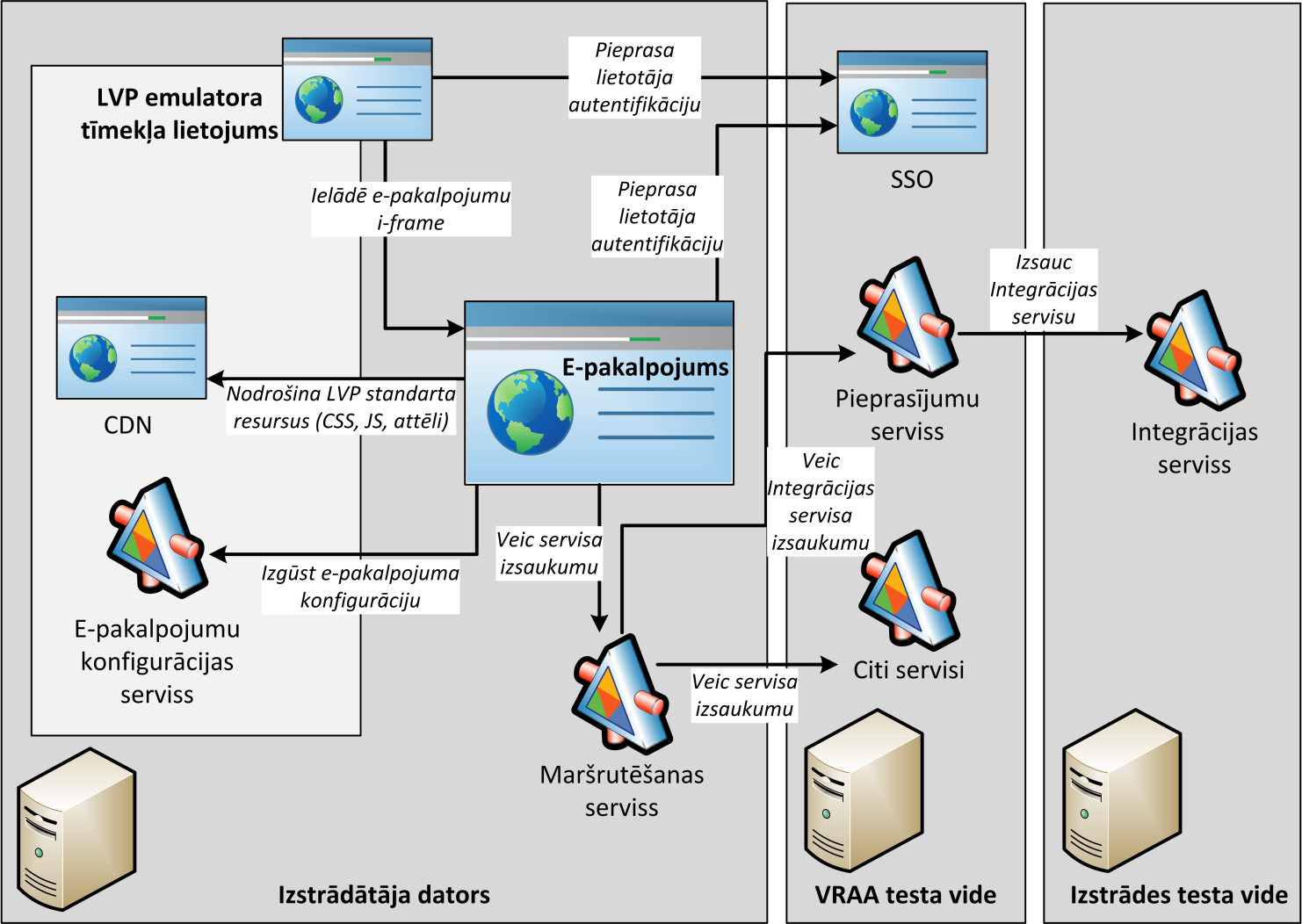
Emulētās portāla infrastruktūras daļas:

* Konteiners, kurā tiek ielādēts e-pakalpojums.
* Portāla standarta resursi (CSS, JS, attēli).
* E-pakalpojuma konfigurācijas serviss.

Lietas, kuras netiek emulētas un kuru darbībai nepieciešama atbilstoša e-pakalpojumu vides uzturētāja (turpmāk EPU) testa vides konfigurācija:

* Lietotāja autentifikācijas mehānisms (SSO).
* Pieprasījumu serviss (PS).
* Citi servisi (EDK, lietotāja profils, e-parakstīšana, NASA utt.).

Savus e-pakalpojumus izstrādātājs var darbināt, izveidojot IIS lietojumu un to piereģistrējot emulatorā (skat. 5.4.2 un 5.4.3.sadaļas). Darbināšana *VS Development Server* nav iespējama, jo e-pakalpojumus nepieciešams izsaukt caur TLS (https) protokolu, lai nodrošinātu SSO drošību. *VS Development Server* neatbalsta TLS.



6.attēls. E-pakalpojumu izstrādes vide

# E-pakalpojumu piemēri

Lai atvieglotu e-pakalpojumu izstrādes uzsākšanu un ļautu vieglāk saprast darbības mehānismus, ir pieejami e-pakalpojumu piemēri. Izstrādātāji var apskatīt to kodu un uzstādīt pakalpojumus lokālajā izstrādes vidē.

Emulatorā šie e-pakalpojumi ir reģistrēti pēc noklusējuma. Ja piemērus nav paredzēts izmantot, no emulatora var dzēst tiem atbilstošo konfigurāciju (skat. 9.1.nodaļu).

## Lvp.Eservice.Example.ComplexUI.Web

Šajā e-pakalpojumā ir apkopoti dažādi e-pakalpojuma kontroļu lietojumi. Tā uzdevums ir parādīt veidus, kā realizēt sarežģītākus lietotāja interfeisa gadījumus un demonstrēt piedāvāto kontroļu pielāgošanas iespējas.

E-pakalpojums minimāli izmanto PS – tikai uzsāk un beidz transakciju.

## Lvp.Eservice.Example.ServiceIntegration.Web

Šajā e-pakalpojumā iespējams apskatīt:

* PS izsaukšanu gan ar sinhrono, gan ar asinhrono metodi (skat. 7.1.8.sadaļu).
* EDK servisu izsaukšanu (skat. 7.1.2.sadaļu).
* Dokumentu e-parakstīšanas un e-paraksta validācijas servisa izsaukšanu (skat. 7.1.3. un 7.1.4.sadaļas).

## Lvp.Eservice.Example.IsolationForDevelopment.Web

Piemērs, kurā PS un SSO integrācija aizstāta ar aizbāžņiem, nodrošinot iespēju e-pakalpojumu darbināt uz viena, no ārpasaules izolēta datora. Šāda pieeja var palīdzēt izstrādes procesa vienkāršošanai un projekta testēšanai, bet nekādā gadījumā **nedrīkst tikt izmantota produkcijas vidē**.

## Lvp.Eservice.Example.Infrastructure.Web

Šajā e-pakalpojumā iespējams apskatīt:

* Piedāvātās žurnalēšanas funkcionalitātes izmantošanu (skat. 7.1.5.sadaļu).
* Pielāgotu XML ziņojumu veidošanu (*serialization*) PS izsaukšanai (skat. 7.1.16.sadaļu).
* Lietotāja profila īpašību izgūšanu un labošanu (skat. 7.1.14.sadaļu).
* Maksājumu moduļa izsaukšanu (skat. 7.1.9.un 7.1.8.sadaļas).

## Lvp.Eservice.Example.Globalization.Web

Piemērs demonstrē iespējas daudzvalodu e-pakalpojumu izstrādei (e-pakalpojuma valoda mainās atbilstoši LVP valodai).

Piedāvātajai realizācijai pamatā tiek izmantots standarta ASP.NET lokalizācijas risinājums.

## Lvp.Eservice.Example.Sidebar.Web

Demonstrē iespējas e-pakalpojumā (iekš *iframe*) izvietot sānjoslas un filtrus, kas atbilst portāla kopējam dizainam.

# Izstrādes vides sagatavošana

## Izstrādes vides prasības

* Operētājsistēma: *Windows*;
* Uzinstalēta *.Net Framework 4.0* komponente;
* Uzinstalēta *IIS 7.0 ASP.NET* komponente darbam ar *.Net Framework 4.0*;
* Uzinstalēta *Visual Studio* (*2010 Service Pack 1* vai jaunāka versija).

## Pakotnes saturs

E-pakalpojumu pakotne (LVP\_EPAK\_WZD\_2.zip) satur e-pakalpojumu izstrādes videi nepieciešamos lietojumus, e-pakalpojumu piemērus un jauna e-pakalpojuma šablonu. Pakotne jāuzstāda uz e-pakalpojumu izstrādātāju datoriem.

Pakotnes saturs:

* ***Lvp.Eservice.Environment*** – izstrādes videi nepieciešamās komponentes;
* ***Lvp.Eservice.Template*** – e-pakalpojuma sagatave (jaunu e-pakalpojumu veidošanai);
* ***Setup*** – skripti e-pakalpojumu vides uzstādīšanai (nav paredzēti tiešai izsaukšanai);
* ***CreateEservice.bat*** – jauna e-pakalpojuma izveides skripts;
* ***InstallEnvironment.bat*** – izstrādes vides sagatavošanas skripts;
* ***UninstallEnvironment.bat*** – izstrādes vides noņemšanas skripts;
* ***UpdateEnvironment.bat*** – izstrādes vides atjaunošanas skripts.

### Lvp.Eservice.Template apakšpakotnes saturs

* ***Lvp.Eservice.EPXXX.Web*** – e-pakalpojuma sagataves projekts, kuru kopējot un pielāgojot paredzēts veidot jaunos e-pakalpojumus (skat. 5.4.1.sadaļu).
* ***Reference Assemblies*** – mape, kurā paredzēts glabāt bibliotēkas, kas nepieciešamas visiem izstrādājamajiem e-pakalpojumiem un kuras projektā tiek iekļautas kā bibliotēkas (dll), nevis kā VS projekti.

### Lvp.Eservice.Environment apakšpakotnes saturs

Pakotnē atrodamas šādas mapes:

* ***Lvp*.*Eservice*.*Certificates*** – sertifikāti;
* ***Lvp*.*Eservice*.*Examples*** – e-pakalpojumu piemēri;
* ***Lvp*.*Eservice*.*PortalEmulator*** – LVP emulators (izstrādes videi);
* ***Lvp*.*Eservice*.*RouterService*** – e-pakalpojumu maršrutēšanas serviss.

#### Lvp.Eservice.Certificates apakšpakotnes saturs

* ***eservices.pfx*** – e-pakalpojumu SSL/TLS sertifikāts (e-pakalpojumu https nodrošināšanai izstrādes vidē).
* **eservices\_*Root*\_*CA*.*cer*** – e-pakalpojumu SSL/TLS sertifikāta CA sertifikāts.
* ***IVIS*\_*Root\_CA.cer*** – VISS CA sertifikāts (vecais).
* ***VISS*\_*Root\_CA.cer*** – VISS CA sertifikāts.
* ***esigning*\_*client*.*pfx*** – e-parakstīšanas servisa izsaukšanai (autentifikācijai) nepieciešams sertifikāts – skat. 7.1.3.sadaļu.
* ***Abc.Epak.Router.pfx*** – maršrutēšanas servisa sertifikāts, lai autentificētos STS servisā.
* ***Abc.Epak.Examples.ComplexUi.pfx*** – Lvp.Eservice.Example.ComplexUI.Web e-pakalpojuma piemēra sertifikāts.
* ***Abc.Epak.Examples.ServiceIntegration.pfx*** - Lvp.Eservice.Example.ServiceIntegration.Web e-pakalpojuma piemēra sertifikāts.
* ***Abc.Epak.Examples.Infrastructure.pfx*** – Lvp.Eservice.Example.Infrastructure.Web e-pakalpojuma piemēra sertifikāts.
* ***Abc.Epak.Examples.Globalization.pfx*** – Lvp.Eservice.Example.Globalization.Web e-pakalpojuma piemēra sertifikāts.
* ***Abc.Epak.Examples.Sidebar.pfx*** - Abc.Epak.Examples.Sidebar e-pakalpojuma piemēra sertifikāts.

#### Lvp.Eservice.Examples apakšpakotnes saturs

* ***Lvp.Eservice.Example.Common*** – e-pakalpojumu piemēros atkārtoti izmantotās funkcionalitātes bibliotēka.
* ***Lvp.Eservice.Example.ComplexUI.Web*** – e-pakalpojuma piemērs (skat. 3.1.nodaļu).
* ***Lvp.Eservice.Example.Globalization.Web*** – e-pakalpojuma piemērs (skat. 3.5.nodaļu).
* ***Lvp.Eservice.Example.Infrastructure.Web*** – e-pakalpojuma piemērs (skat. 3.4.nodaļu).
* ***Lvp.Eservice.Example.IsolationForDevelopment.Web*** – e-pakalpojuma piemērs (skat. 3.3.nodaļu).
* ***Lvp.Eservice.Example.ServiceIntegration.Web*** – e-pakalpojuma piemērs (skat. 3.2.nodaļu).
* ***Lvp.Eservice.Example.Sidebar.Web*** – e-pakalpojuma piemērs (skat. 3.6.nodaļu).
* ***Reference Assemblies*** – mape, kurā tiek glabātas bibliotēkas, kas nepieciešamas visiem e-pakalpojumu piemēriem un kuras projektā tiek iekļautas kā bibliotēkas (dll), nevis kā VS projekti.
* ***BasicCorrectnessRules-custom.ruleset*** – piemēru koda kvalitātes pārbaudes iestatījumi.
* ***CustomDictionary.xml*** – piemēru koda kvalitātes pārbaudes iestatījumi.
* ***Lvp.Eservice.Example.Web.sln*** – VS risinājums (*solution*), kas iekļauj e-pakalpojumu piemēru projektus.

### Setup apakšpakotnes saturs

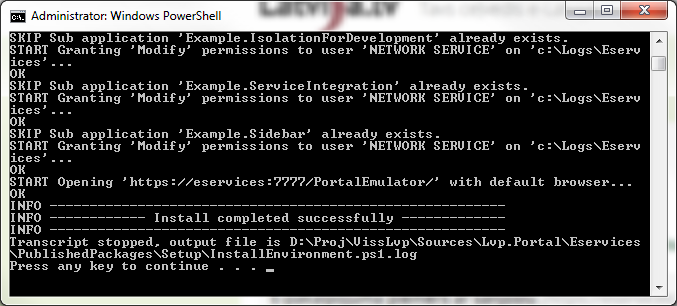
Šeit atrodami uzstādīšanas laikā izmantotie papildu skripti. Skriptus ar *.ps1* paplašinājumu nav paredzēts izpildīt „pa tiešo”. Tie tiek izsaukti, izpildot *.bat* skriptus. Kopumā izstrādātājam nebūtu jāpievērš uzmanība šīs mapes saturam.

## Izstrādes vides uzstādīšana

Lai veiktu vides uzstādīšanu:

1. Jāatarhivē pakotne (LVP\_EPAK\_WZD\_2.zip). Viss atarhivētais saturs jāpārvieto uz brīvi izvēlētu mapi, kurā tiks darbināti e-pakalpojumu lietojumi (emulators, maršrutēšanas serviss, piemēri utt.).
2. Ar administratora tiesībām (Run as administrator) jāpalaiž *InstallEnvironment.bat* datne.
3. Jāievada simbols „y”, lai apstiprinātu uzstādīšanas sākšanu.

Veiksmīgas uzstādīšanas rezultātā tīmekļa pārlūkā ir jāatveras emulatora lapai.



7.attēls. Veiksmīgas skripta izpildes rezultāts

Skripta izpildes žurnāls tiek ne tikai izvadīts konsolē, bet arī saglabāts datnē *\Setup\InstallEnvironment.ps1.log*.

Gadījumā, ja skripta izpilde beidzas ar kļūdu, jāveic manuāla izstrādes vides uzstādīšana (4.4. nodaļa). Vēlams arī sazināties ar EPU (VRAA), lai informētu par kļūdu un nākotnē varētu novērst līdzīgas problēmas.

Skriptu var droši izpildīt atkārtoti. Tajā iestrādātā loģika paredz, ka modifikācijas tiek veiktas tikai, ja tās ir nepieciešamas. Piemēram, ja IIS nav reģistrēts emulators, tas tiek piereģistrēts. Ja nav importēts kāds sertifikāts, tas tiek izdarīts. Tādēļ skripta izpildes kļūdu gadījumos var mēģināt arī manuāli novērst kļūdas iemeslus un skriptu darbināt atkārtoti.

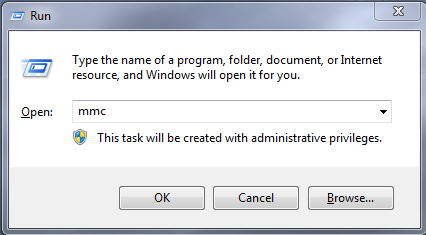
## Manuāla izstrādes vides uzstādīšana

Šajā nodaļā aprakstītās darbības jāveic tikai, ja ir radušās problēmas ar automātiskās uzstādīšanas skriptu (skat. 4.3.nodaļu). Skripts šajā nodaļā aprakstītās darbības veic automātiski.

### Portāla emulatora uzstādīšana

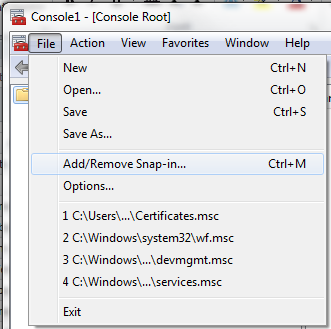
#### Sertifikātu pārvaldības rīka atvēršana

1. *Start -> Run -> mmc*



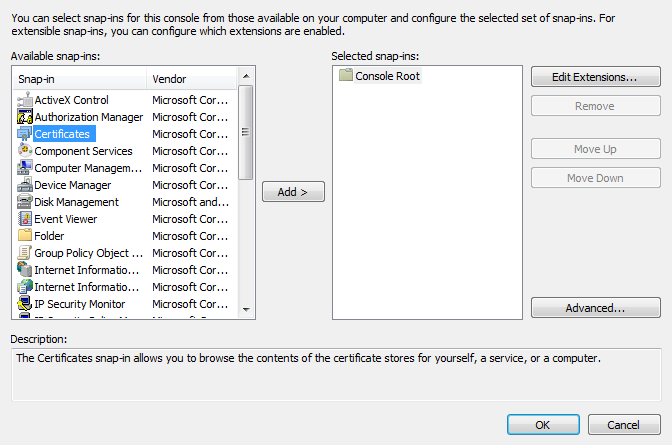
8.attēls. MMC palaišana

1. *File -> Add/Remove Snap-in*



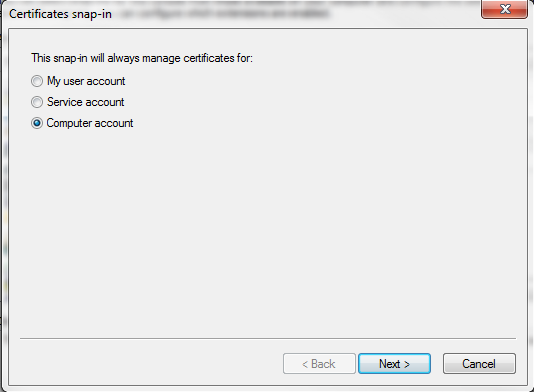
9.attēls. Sertifikātu spraudņa pievienošana (1)

1. *Certificates -> Add*



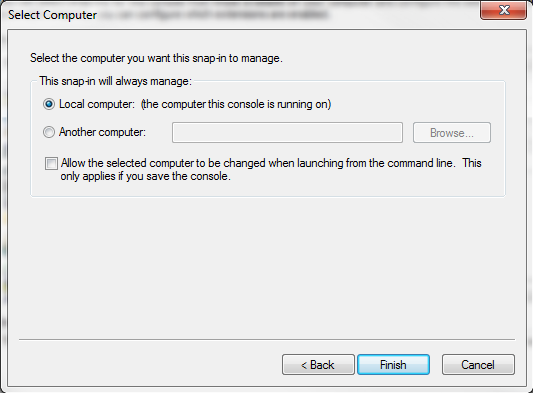
10.attēls. Sertifikātu spraudņa pievienošana (2)

1. *Computer Account*



11.attēls. Sertifikātu spraudņa pievienošana (3)

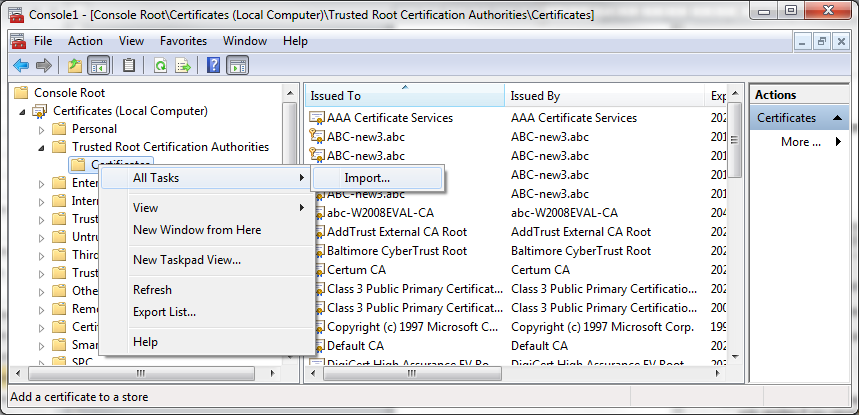
1. *Local computer*



12.attēls. Sertifikātu spraudņa pievienošana (4)

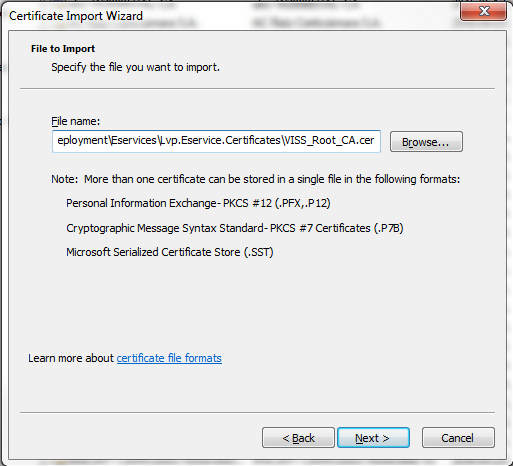
#### Saknes sertifikātu imports

1. *All tasks -> Import* (no *Trusted Root Certification Authorities* mapes)

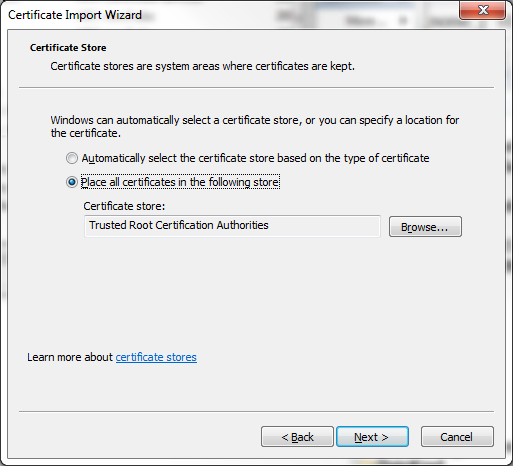


13.attēls. Saknes sertifikāta imports

1. Jāielādē saknes sertifikāta fails *VISS\_Root\_CA.cer*.



14.attēls. Saknes sertifikāta importa vednis (2.solis)



15.attēls. Saknes sertifikāta importa vednis (3.solis)

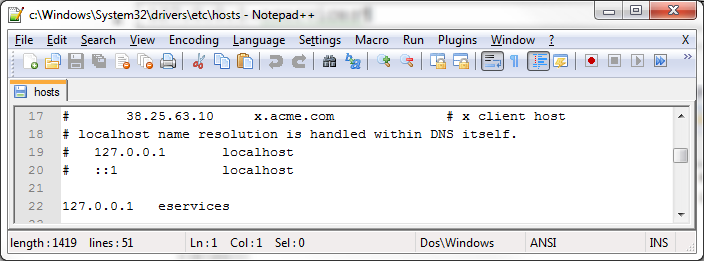
1. Jāielādē saknes sertifikāta fails *eservices\_Root\_CA.cer* (ielādes instrukcijai skatīt iepriekšējo punktu).

#### E-pakalpojumu IIS lietojuma (application) izveide

1. *Hosts* datnē jāpievieno rindiņa:

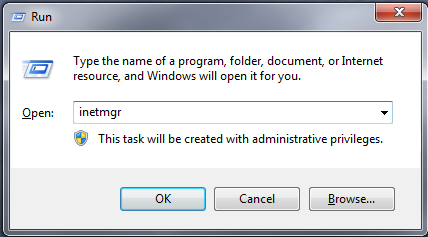
127.0.0.1 eservices

*Hosts* datne atrodama mapē *c:\Windows\System32\drivers\etc* jeb *%windir%\System32\drivers\etc*.



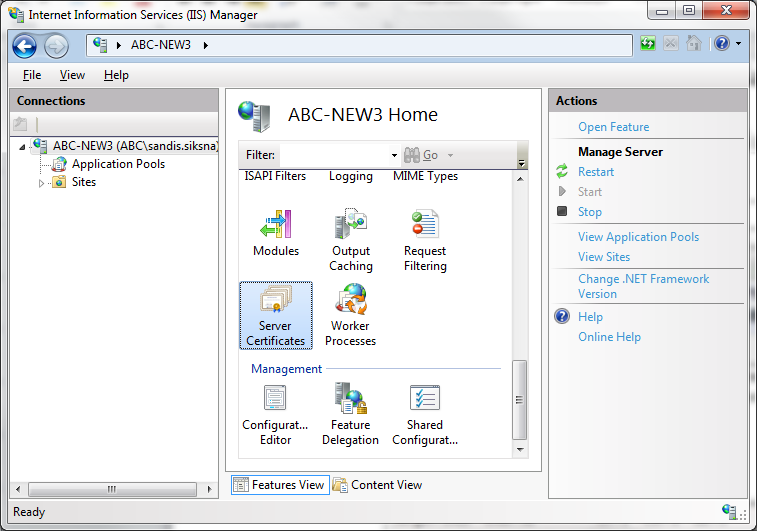
16.attēls. Hosts datnes labošana

1. IIS jāimportē sertifikāts, lai varētu lietojumā nodrošināt https:
   1. Jāatver *IIS Manager (Start -> Run -> inetmgr)*



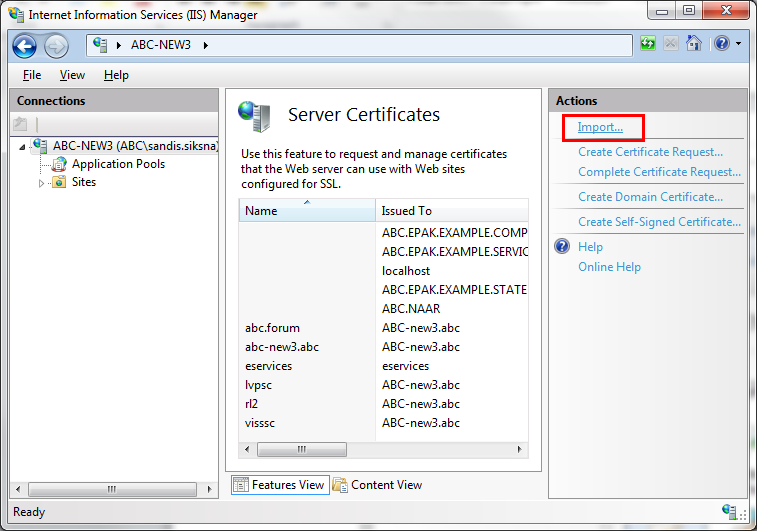
17.attēls. IIS Manager palaišana

* 1. Jāatver IIS sertifikātu pārvaldība *(Server Certificates)*



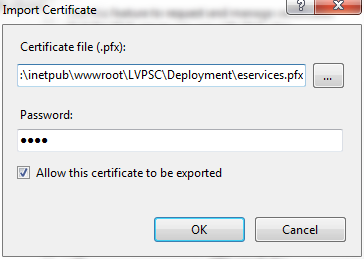
18.attēls. Sertifikātu pārvaldības atvēršana

* 1. Jāuzsāk sertifikāta imports *(Import)*



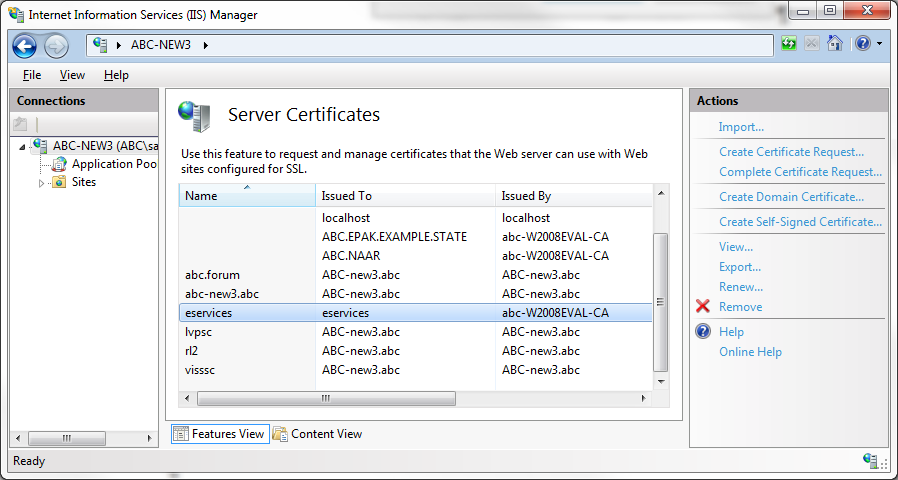
19.attēls. Sertifikātu importa uzsākšana

* 1. Jāielādē *eservices.pfx* datneun jānorāda parole *1234.*



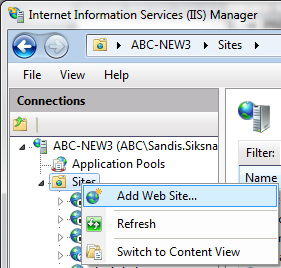
20.attēls. Sertifikāta imports

* 1. Jāpārliecinās, ka sertifikāts ir veiksmīgi importēts.



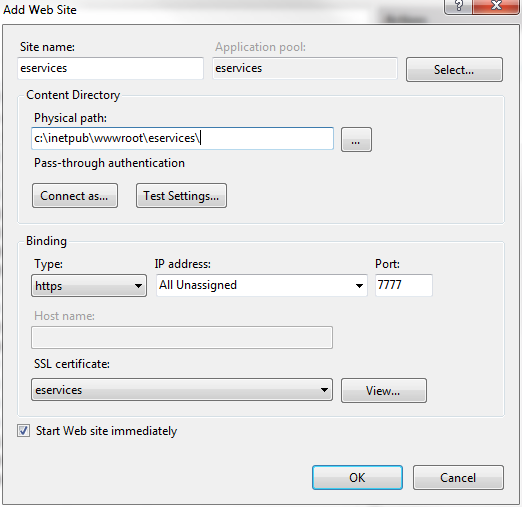
21.attēls. Sertifikāts veiksmīgi importēts

1. IIS jāizveido jauns lietojums.
   1. *Sites -> Add Web Site*



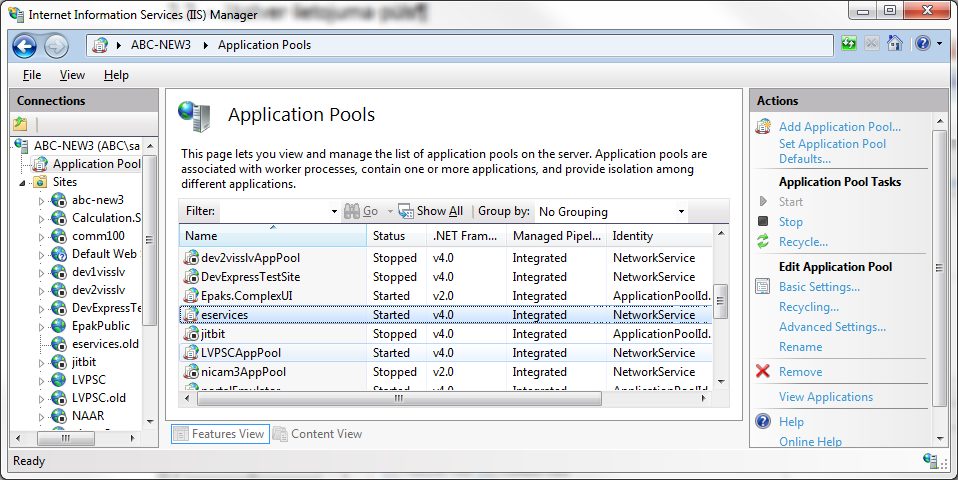
22.attēls. Jauna lietojuma pievienošanas uzsākšana

* 1. Jāievada parametri:
* ***Site name****:* eservices (teorētiski var ievadīt brīvi izvēlētu vērtību, bet tad jārēķinās ar konfigurācijas maiņu vairākās aplikācijās (emulators, e-pakalpojumu piemēri, šablons utt.), tādēļ ļoti ieteicams pieturēties pie vērtības „eservices”).
* ***Physical path***: ceļš uz brīvi izvēlētu mapi (lietojumā nav paredzēts neko darbināt – e-pakalpojumi un emulators tiks uzstādīti kā šī lietojuma apakšlietojumi). Mapei vēlams būt tukšai, jo gadījumā, ja mapē atradīsies konfigurācijas fails (web.config), šī konfigurācijas faila vērtības tiks attiecinātas uz apakšlietojumiem, radot potenciāli grūti saprotamas kļūdas.
* ***Type****:* https
* ***Port****:* 7777
* ***SSL certificate***: eservices



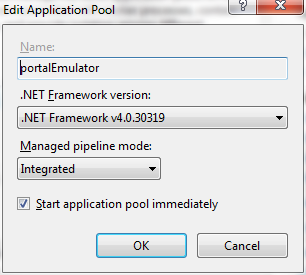
23.attēls. Lietojuma parametri

* 1. Jāatver lietojuma pūls.



24.attēls. Lietojuma pūla atvēršana

* 1. Jāpārliecinās, ka pūlam ir šādi parametri:
* ***.NET Framework version***: v4.0
* ***Managed pipeline mode***: Integrated

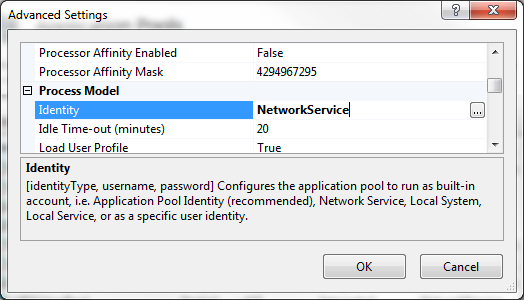


25.attēls. Lietojuma pūla parametri

* 1. Jānorāda pūla lietotājs. Piemērā izmantosim *NetworkService*.

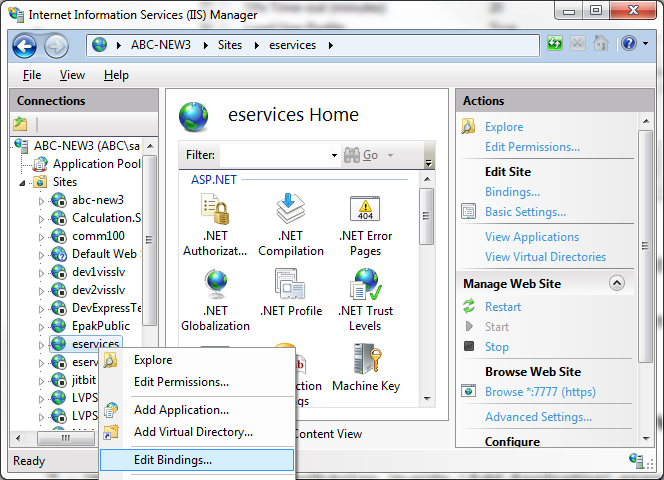
Lietotājs var būt brīvi izvēlēts, vienīgi jārēķinās, ka lietotājam būs jāpiešķir visas nepieciešamās tiesības (failu sistēma, sertifikāti utt.) un šajā dokumentā aprakstītā uzstādīšanas gaita ir pārbaudīta tieši ar minēto lietotāju, tādēļ ieteicams izmantot pārbaudīto NetworkService.

Gadījumā, ja tiek izmantota ApplicationPoolIdentity, uzstādot tiesībās citās vietās (piemēram, failu sistēmā), lietotājs jānorāda formā „iis apppool\pūla\_nosaukums”.

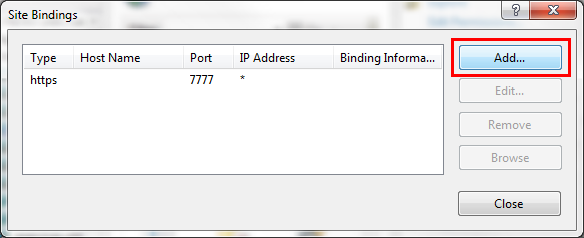


26.attēls. Pūla lietotājs

* 1. Lietojumam jāpievieno saistījums (*binding*):



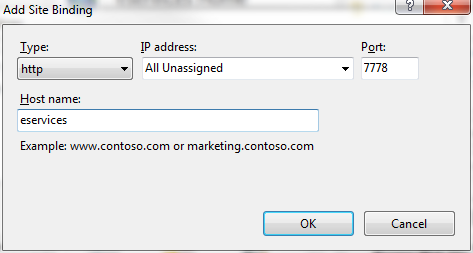
27.attēls. Saistījumu konfigurācijas atvēršana



28. attēls. Jauna saistījuma pievienošanas uzsākšana

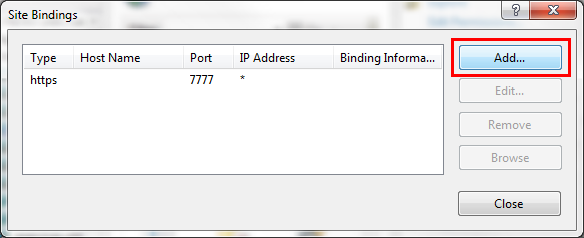
Saistījuma parametri:

* ***Type****:* http
* ***Port****:* 7778
* ***Host name***: eservices



29.attēls. Saistījuma parametru ievade

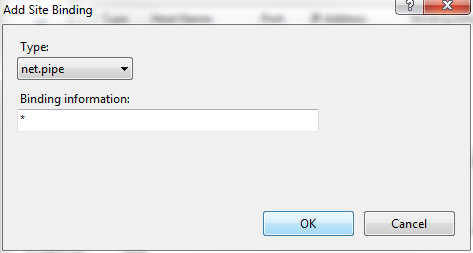
* 1. Lietojumam jāpievieno *net.pipe* saistījums (*binding*):



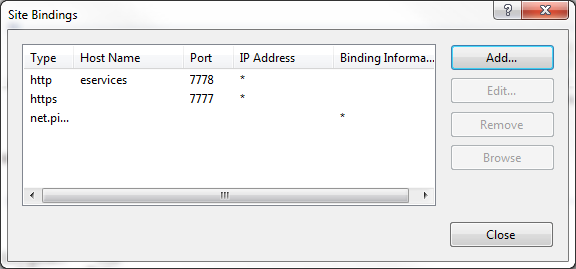
30. attēls. Jauna saistījuma pievienošanas uzsākšana

Saistījuma parametri:

* ***Type****:* net.pipe
* ***Binding information***: \*



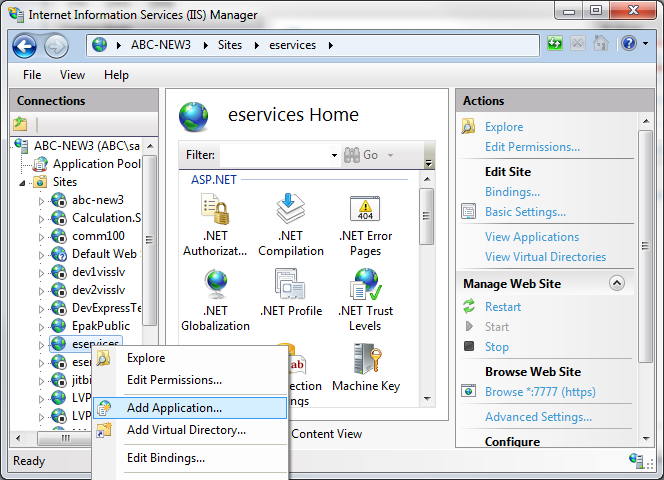
31.attēls. Saistījuma parametru ievade



32.attēls. Nepieciešamie eservices lietojuma saistījumi

#### Portāla emulatora apakšlietojuma izveidošana

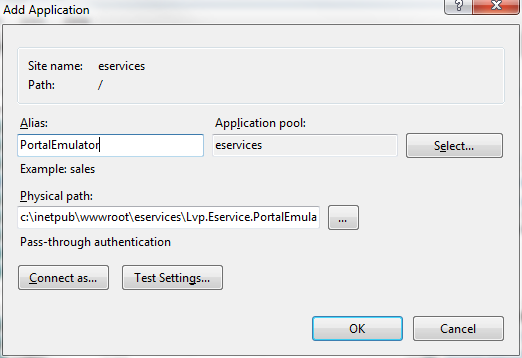
1. No *Lvp.Eservice.Environment* mapes jāpārkopē *Lvp.Eservice.PortalEmulator* mape uz plānoto lietojuma atrašanās vietu.
2. Jāuzsāk jauna apakšlietojuma izveide (*Add Application*) ***eservices*** IIS lietojumam.



33.attēls. Apakšlietojuma izveides uzsākšana

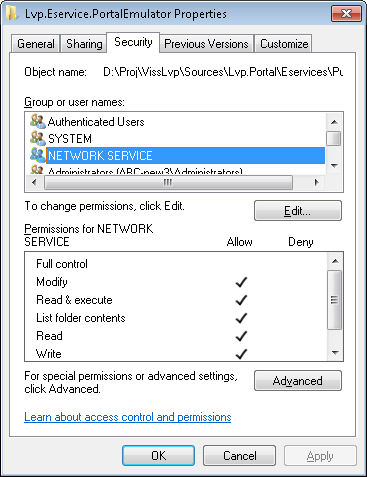
1. Jāpievieno apakšlietojums ar parametriem:

* ***Alias***: PortalEmulator;
* ***Application* *pool***: noklusētā vērtība (eservices lietojuma pūls);
* ***Physical* *path***: ceļš uz 1.solī kopēto mapi.



34.attēls. Apakšlietojuma parametri

1. Jāpārliecinās, ka *eservices* pūla lietotājam ir datņu sistēmas (*file system*) tiesības lasīt un rakstīt apakšlietojuma mapē:



35.attēls. Apakšlietojuma parametri

### Maršrutēšanas servisa uzstādīšana

Maršrutēšanas serviss atrodams *Lvp.Eservice.Environment* pakotnes *Lvp.Eservice.RouterService* mapē. Ir paredzēts, ka vienu servisa eksemplāru izmantos vairāki (visi) izstrādājamie e-pakalpojumi.

Izstrādes vidē maršrutēšanas servisu var uzstādīt lokāli katram izstrādātājam vai arī starp vairākiem izstrādātājiem izmantot vienu eksemplāru uz atsevišķa servera.

#### IIS apakšlietojuma izveide

Priekšnosacījumi: izpildītas 4.4.1.2. un 4.4.1.3.paragrāfu instrukcijas.

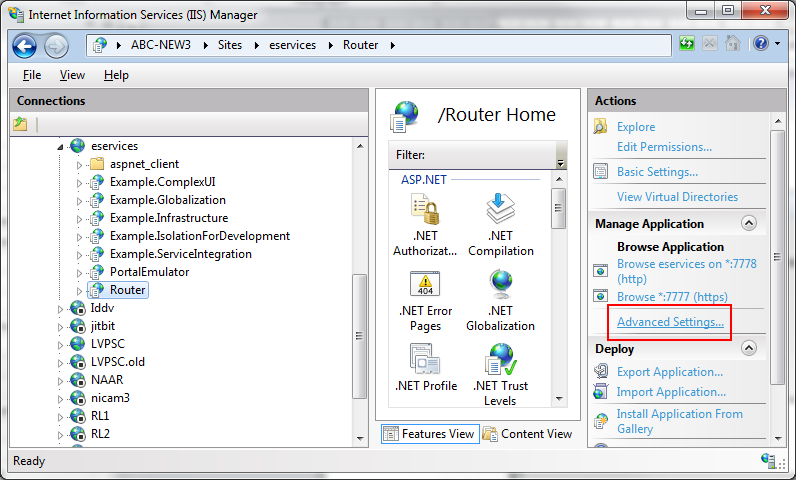
No *Lvp.Eservice.Environment* pakotnes jāpārkopē *Lvp.Eservice.RouterService* mape uz plānoto lietojuma atrašanās vietu.

Jāpievieno jauns apakšlietojums ***eservices*** lietojumam (skat. 4.4.1.4.paragrāfu) ar šādiem parametriem:

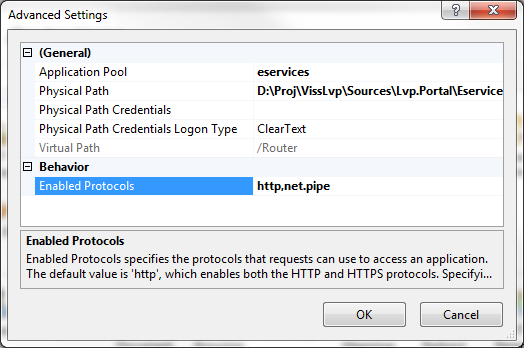
* ***Alias***: Router;
* ***Application* *pool***: noklusētā vērtība (eservices lietojuma pūls);
* ***Physical* *path***: ceļš uz pārkopēto mapi.

Pūla lietotājam jāpiešķir lasīšanas tiesības apakšlietojuma mapē (skat. 4.4.1.4.paragrāfa pēdējo punktu).

Apakšlietojumā jāiespējo *net.pipe* protokols (papildu iestatījumos *Enabled Protocols* jāievada *http,net.pipe*):



36.attēls. Apakšlietojuma papildu iestatījumu atvēršana

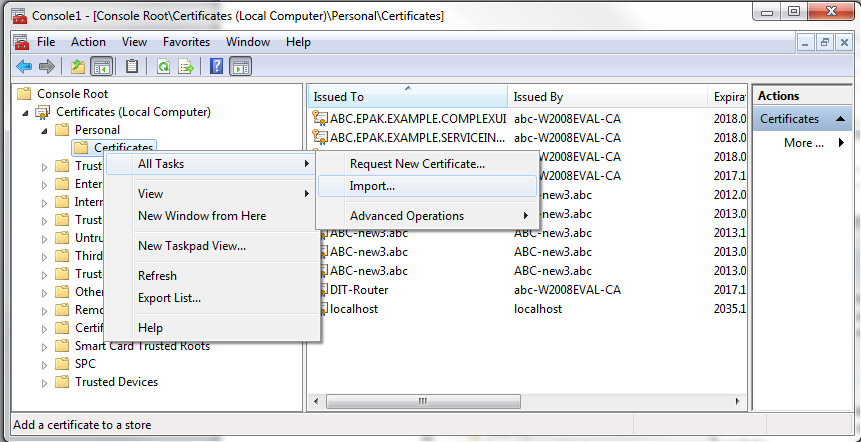


37.attēls. net.pipe protokola iespējošana

#### Sertifikātu importēšana

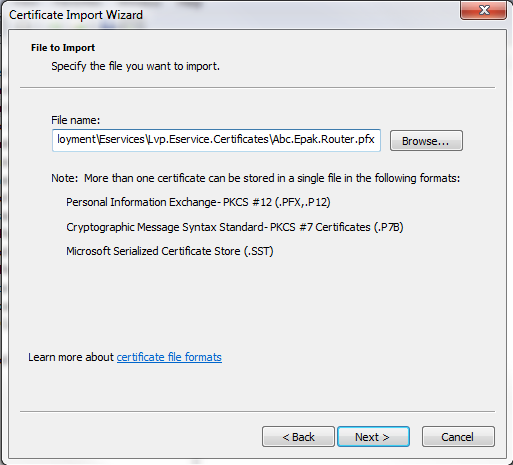
Lai no e-pakalpojumiem varētu izsaukt PS ar autentificētu lietotāju vai citus drošos servisus (EDK, lietotāja profils), ir jāreģistrē *Abc.Epak.Router.pfx* sertifikāts, kas atrodams sertifikātu mapē.

1. Jāatver sertifikātu pārvaldības rīks (skat. 4.4.1.1.paragrāfu).
2. *All* *tasks* -> *Import* (no Personal mapes).



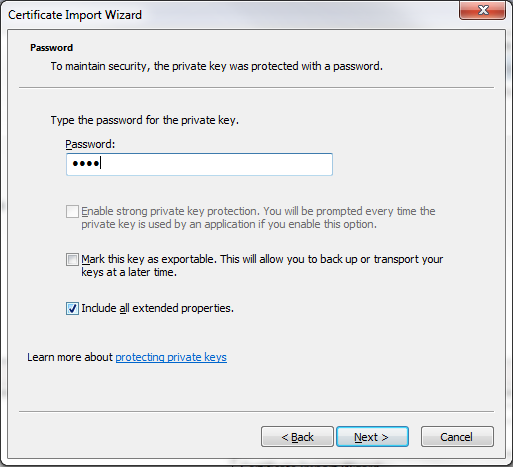
38.attēls. Sertifikāta imports

1. Jāielādē sertifikāta fails Abc.Epak.Router.pfx.



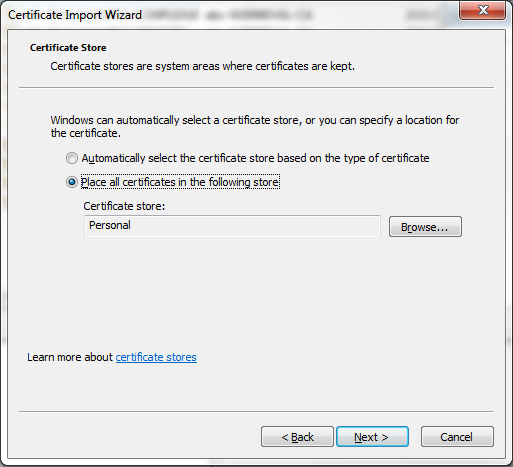
39.attēls. Sertifikāta importa vednis (2.solis)

1. Jāievada sertifikāta parole (1234).



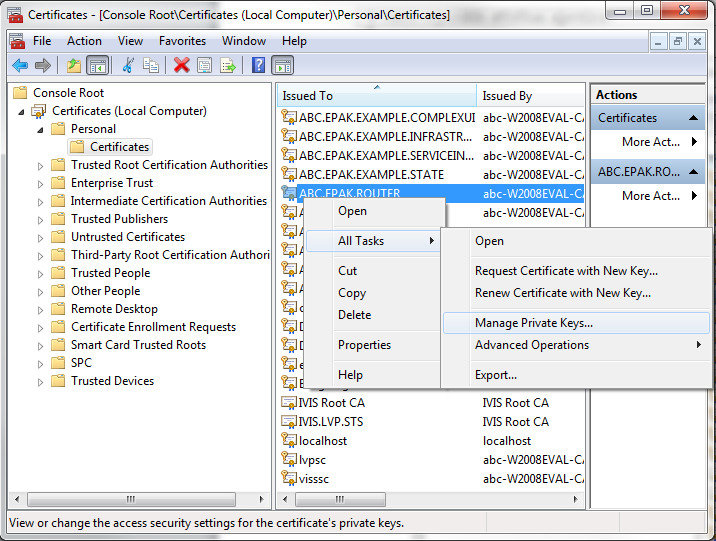
40.attēls. Sertifikāta importa vednis (3.solis)

1. Jāpabeidz vednis.

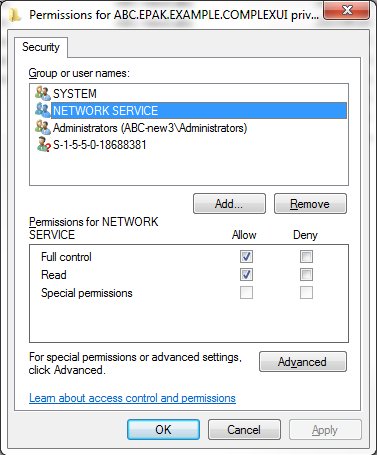


41.attēls. Sertifikāta importa vednis (4.solis)

1. Jāpārliecinās, ka 4.4.1.3.paragrāfā izveidotā tīmekļa lietojuma pūla lietotājam ir tiesības lasīt sertifikāta privāto atslēgu (*All Tasks -> Manage Private Keys*).



42.attēls. Sertifikāta privāto atslēgu tiesību pārvaldības atvēršana



43.attēls. Sertifikāta privāto atslēgu tiesību pārvaldība

1. Jāuzstāda arī sertifikāts e-parakstīšanas servisa autentifikācijai (*esigning\_client.pfx*). Uzstādīšanas soļi ir analoģiski maršrutēšanas servisa sertifikāta uzstādīšanai (iepriekšējie punkti).

#### Žurnalēšanas mapju uzstādīšana

Nepieciešams izveidot mapi žurnalēšanas datnēm un iepriekšējos soļos izveidotā pūla lietotājam jāpiešķir tiesības rakstīt šajā mapē. Pretējā gadījumā žurnalēšanas informācija vienkārši netiks saglabāta.

Pēc noklusējuma mapes ceļš ir „C:\Logs\Eservices\”, bet to var mainīt konfigurācijas datnē (web.config - atrodas maršrutēšanas servisa IIS apakšlietojuma mapē):

<source name="Microsoft.IdentityModel" switchValue="Warning">

<listeners>

<add name="IdentityModelTraceListener"

initializeData="c:\Logs\Eservices\Router.WIFTrace.svclog"

type="System.Diagnostics.XmlWriterTraceListener"

traceOutputOptions="Timestamp"/>

</listeners>

</source>

<source name="System.ServiceModel.MessageLogging"

switchValue="Warning, ActivityTracing">

<listeners>

<add name="ServiceModelMessageLoggingListener"

initializeData="c:\Logs\Eservices\Router.web\_messages.svclog"

type="System.Diagnostics.XmlWriterTraceListener"

traceOutputOptions="Timestamp"/>

</listeners>

</source>

<source name="System.ServiceModel"

switchValue="Warning, ActivityTracing" propagateActivity="true">

<listeners>

<add name="ServiceModelTraceListener"

initializeData="c:\Logs\Eservices\Router.web\_tracelog.svclog"

type="System.Diagnostics.XmlWriterTraceListener"

traceOutputOptions="Timestamp"/>

</listeners>

</source>

<source name="System.ServiceModel.Routing" switchValue="Warning">

<listeners>

<add name="ServiceModelTraceListener"

initializeData="c:\Logs\Eservices\Router.web\_routerlog.svclog"

type="System.Diagnostics.XmlWriterTraceListener"

traceOutputOptions="Timestamp"/>

</listeners>

</source>

### E-pakalpojumu piemēru uzstādīšana

E-pakalpojumu piemēri atrodami *Lvp.Eservice.Environment* pakotnes *Lvp.Eservice.Examples* mapē. Apakšnodaļās aprakstītās darbības jāveic katram piemēram (apakšmape ar nosaukumu, kas beidzas ar „.Web”).

#### IIS apakšlietojuma izveide

Priekšnosacījumi: izpildītas 4.4.1.2. un 4.4.1.3.paragrāfu instrukcijas.

Jāpievieno jauns apakšlietojums ***eservices*** lietojumam (skat. 4.4.1.4.paragrāfu) ar šādiem parametriem:

* ***Alias***: Piemēra nosaukums (daļa mapes nosaukumā starp vārdiem „Eservice” un „Web”, piemēram, Example.ComplexUI);
* ***Application* *pool***: noklusētā vērtība (eservices lietojuma pūls);
* ***Physical* *path***: ceļš uz piemēra mapi.

Jāatceras arī pūla lietotājam piešķirt tiesības piemēru mapēs (lasīšanas tiesības pakalpojuma saknes un apakšmapēs, kā arī rakstīšanas tiesības App\_Data mapē).

#### Sertifikātu importēšana

Jāveic sertifikātu uzstādīšana un privātās atslēgas pieejas tiesību piešķiršana, kā tas ir aprakstīts 4.4.2.2.paragrāfā. Sertifikāti atrodami sertifikātu arhīvā (skat. 4.2.2.1.paragrāfu) un to nosaukums atbilst e-pakalpojuma projekta nosaukumam (piemēram, *Abc.Epak.Examples.ComplexUi.pfx*).

Vienīgais e-pakalpojums bez sertifikāta ir Lvp.Eservice.Example.IsolationForDevelopment.Web, jo tas demonstrē iespēju izstrādes laikā neizmantot PS.

#### Žurnalēšanas mapju uzstādīšana

Nepieciešams izveidot mapi žurnalēšanas datnēm un iepriekšējos soļos izveidotā pūla lietotājam jāpiešķir tiesības rakstīt šajā mapē. Pretējā gadījumā žurnalēšanas informācija vienkārši netiks saglabāta.

Pēc noklusējuma mapes ceļš ir „C:\Logs\Eservices\”, bet to var mainīt konfigurācijas datnē (web.config):

<add name="XMLTraceListener"

type="IVIS.Diagnostics.LogFlatFileTraceListener, IVIS.Diagnostics"

initializeData="**c:\Logs\Eservices\**Example.ComplexUI.txt" />

<add name="MessageListener"

initializeData="**c:\Logs\Eservices\**Example.ComplexUI.Messages.svclog"

type="System.Diagnostics.XmlWriterTraceListener"

traceOutputOptions="Timestamp"/>

Pēc noklusējuma visu e-pakalpojumu piemēru žurnalēšanas datnes tiek glabātas vienā mapē.

## Izstrādes vides atjaunošana

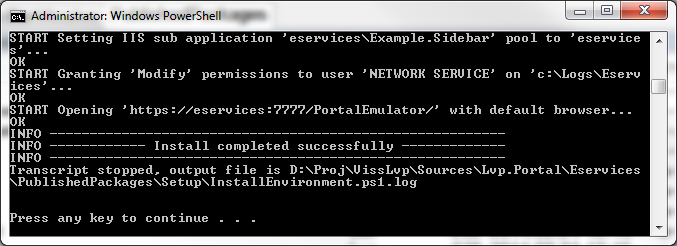
Vides atjaunošanas skripts veic iepriekšējās vides noņemšanu (4.6.nodaļa) un jaunās vides uzstādīšanu (4.3.nodaļa).

Veicot atjaunošanu, tiks saglabāti izstrādātāja veidoto e-pakalpojumu apakšaplikācijas. T.i. noņemti un uzlikti tiks tikai pakotnē esošie (uzstādīšanas skriptos aprakstītie) lietojumi.

Lai veiktu vides atjaunošanu:

1. Jāatarhivē pakotne (LVP\_EPAK\_WZD\_2.zip). Viss atarhivētais saturs jāpārvieto uz brīvi izvēlētu mapi, kurā tiks darbināti e-pakalpojumu lietojumi.
2. Gadījumā, ja iepriekšējā uzstādīšana veikta manuāli, vispirms ieteicams palaist uzstādīšanas skriptu (4.3.nodaļa). Tas ļaus samazināt varbūtību scenārijam, kad vecā izstrādes vide tiek veiksmīgi noņemta, bet jaunās uzstādīšanas laikā notiek kļūda.
3. Ar administratora tiesībām (*Run as administrator*) jāpalaiž *UpdateEnvironment.bat* datne.
4. Jāievada simbols „*y*”, lai apstiprinātu atjaunošanas sākšanu.

Veiksmīgas uzstādīšanas rezultātā tīmekļa pārlūkā jāatveras emulatora sākuma lapai.

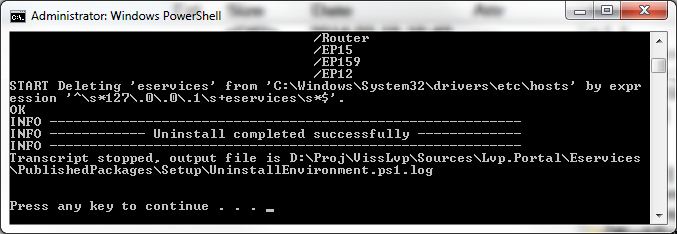


44.attēls. Veiksmīgas skripta izpildes rezultāts

## Izstrādes vides noņemšana (atinstalēšana)

Lai veiktu vides atjaunošanu:

1. Ar administratora tiesībām (Run as administrator) jāpalaiž *UninstallEnvironment*.*bat* datne.
2. Jāievada simbols „y”, lai apstiprinātu noņemšanas sākšanu.



45.attēls. Veiksmīgas skripta izpildes rezultāts

Noņemšanas skripts neveic uzstādīšanas pakā neesošo (manuāli reģistrēto) e-pakalpojumu IIS apakšlietojumu noņemšanu. Piemēram, ja tika izstrādāts e-pakalpojums EP123, tad pēc noņemšanas IIS apakšaplikācijai *eservices/EP123* joprojām jābūt pieejamai.

Netiek veikta arī e-pakalpojumu sertifikātu noņemšana, jo:

1. ir sarežģīti noteikt, vai tos neizmanto vēl kāda aplikācija, kas nenāk no uzstādīšanas pakotnes;
2. pie atjaunošanas (noņemšana un uzstādīšana) tiktu zaudēts manuāli piešķirtās sertifikātu privāto atslēgu lasīšanas tiesības.

# Jauna e-pakalpojuma izveidošana

## Vienošanās par atbalstīto LVP integrāciju

Uzsākot e-pakalpojuma izstrādi, ar pasūtītāju ir jāvienojas par e-pakalpojuma iestatījumiem un e‑pakalpojumā atbalstāmo LVP funkcionalitāti:

* **Mērogojams interfeiss**: Vai e-pakalpojums atbalstīs izmēra maiņu atbilstoši pārlūka loga platumam (tiek piedāvāti pamata stili šādai funkcionalitātei, tomēr visticamāk būs jāveic arī e-pakalpojuma specifiski pielāgojumi).
* **Portāla izkārtojums**: Kāds būs e-pakalpojuma (iframe) platums un kādi LVP komponenti tiks rādīti blakus. Portālā pēc noklusējuma tiek izmantots vienas kolonnas izkārtojums.
* **Vājredzīgo režīms**: Vai e-pakalpojums nodrošinās vājredzīgo režīmiem atbilstošu attēlojumu (tiek piedāvāti pamata stili šādai funkcionalitātei, tomēr visticamāk būs jāveic arī e-pakalpojuma specifiski pielāgojumi).
* **Teksta palielināšana/samazināšana**: Vai e-pakalpojums ļaus lietotājam palielināt un samazināt tā teksta izmēru (šim nolūkam tiek piedāvāta JS bibliotēka – skat. 7.3.4.sadaļu).
* **Daudzvalodu atbalsts**: Vai e-pakalpojums būs pieejams vairākās valodās (vai e-pakalpojuma teksti mainīsies atbilstoši lietotāja izvēlētajai portāla valodai).

Lai labāk saprastu šo parametru nozīmi, ieteicams apskatīt 9.3.nodaļu.

## Priekšnosacījumi e-pakalpojuma integrācijai ar EPU vidi

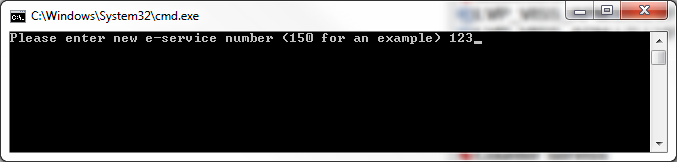
Lai e-pakalpojumu varētu veiksmīgi integrēt ar ārpus izstrādātāja datora esošo infrastruktūru (SSO, STS, PS utt.), nepieciešams veikt šādas darbības:

* Reģistrēt jauno pakalpojumu e-pakalpojumu reģistrā un iegūt e-pakalpojuma URN (iesniegums EPU).
* Reģistrēt e-pakalpojuma sertifikātu Valsts informācijas sistēmu savienotāja (turpmāk VISS) - iesniegums EPU. Pašu sertifikātu var vai nu pieprasīt no EPU vai arī uzģenerēt (pieņemot, ka izstrādātājiem ir pieejams savs sertifikātu izsniedzējs (CA)). Produkcijas videi sertifikātu vienmēr ģenerēs EPU. Papildu informācijai skatīt 5.5.nodaļu.
* Reģistrēt e-pakalpojuma *realm* STS (iesniegums EPU). Papildu informācijai skatīt 5.6.nodaļu.
* Reģistrēt caur PS sūtīto datu shēmas VISS (jāsaņem VISS lietotājs, nosūtot iesniegumu EPU) – skat. 5.10.nodaļu.
* Reģistrēt integrācijas servisus VISS (jāsaņem VISS lietotājs) – skat. 5.11.nodaļu.

## E-pakalpojuma projekta izveide un uzstādīšana

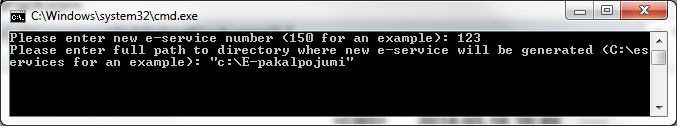
Lai ģenerētu un uzstādītu jaunu e-pakalpojumu:

1. Ar administratora tiesībām (*Run as administrator*) no uzstādīšanas pakotnes jāpalaiž *CreateEservice.bat* skripts.
2. Jāievada e-pakalpojuma numurs (piemēram, „123”):



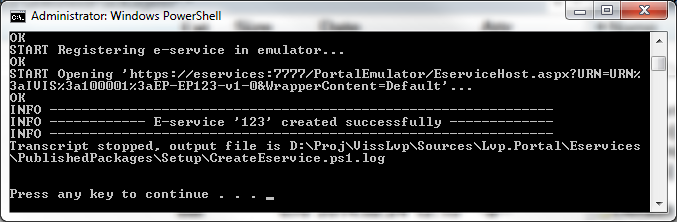
46.attēls. E-pakalpojuma numura ievade

1. Jāievada ceļš uz mapi, kurā darbosies e-pakalpojuma IIS aplikācija un tiks glabāts pakalpojuma kods (piemēram, „C:\E-pakalpojumi”):



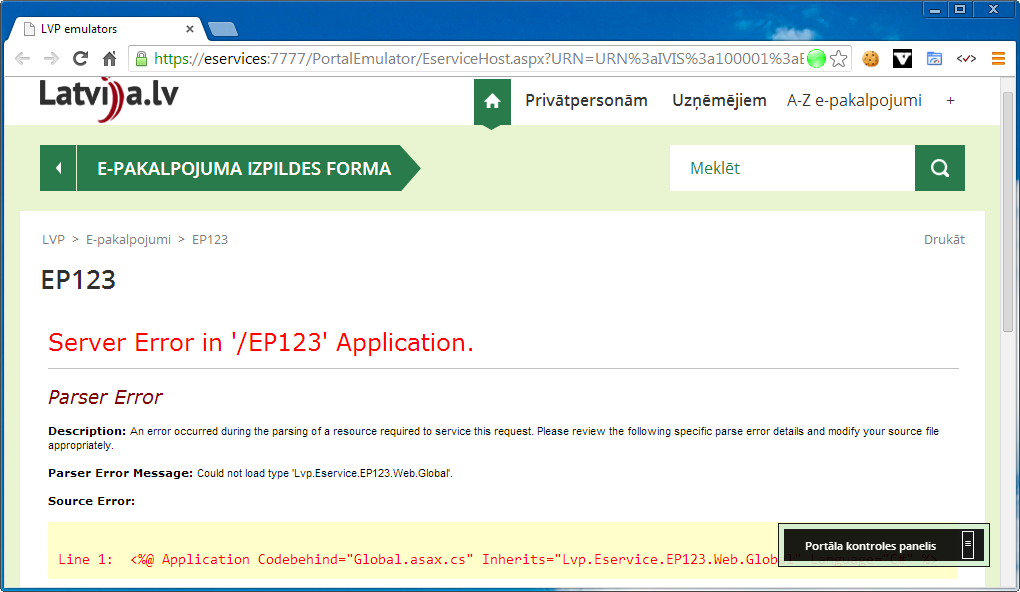
47.attēls. Jauno e-pakalpojumu saturošās mapes ievade

1. Jāsagaida, kamēr skripts izvada veiksmīgas izpildes paziņojumu:



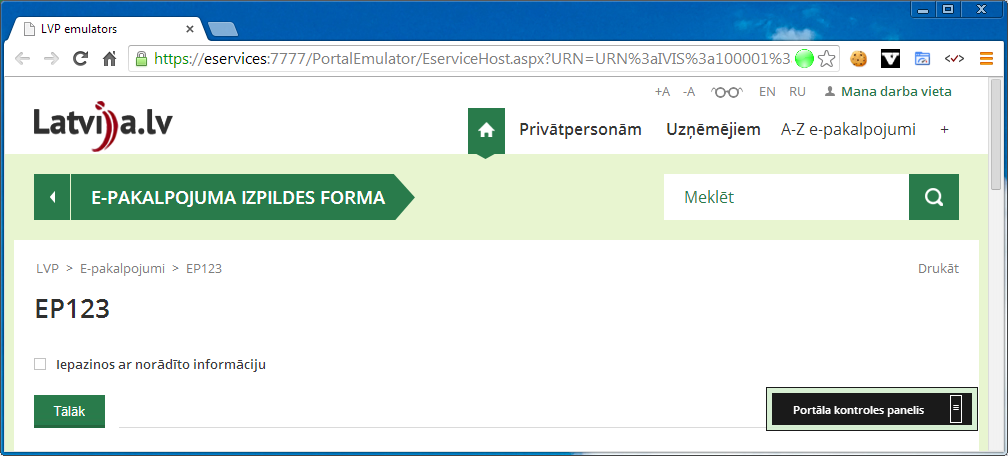
48.attēls. Veiksmīgas skripta izpildes rezultāts

1. Skripta noslēguma fāzē tiek atvērts tīmekļa pārlūka logs ar jauno pakalpojumu. Tā kā pakalpojuma kods šajā brīdī vēl nav nokompilēts, e-pakalpojums attēlos kļūdas paziņojumu.



49.attēls. Jaunā e-pakalpojuma lapa

1. Jāatver e-pakalpojuma projekts (šeit aprakstītā piemēra gadījumā tas atradīsies *c:\E-pakalpojumi\Lvp.Eservice.EP123.Web\Lvp.Eservice.EP123.Web.csproj*) ar *Visual Studio* un jānokompilē.
2. Jāpārlādē tīmekļa pārlūka lapa. Šoreiz ir jāparādās e-pakalpojuma uzsākšanas lapai.



50.attēls. Jaunā e-pakalpojuma lapa

1. Emulatorā jāmaina jaunā e-pakalpojuma iestatījumi (9.1.nodaļa) atbilstoši e-pakalpojuma specifikai (skat. 5.1.nodaļu).
2. Sadarbībā ar EPU, jāveic nepieciešamās konfigurācijas izmaiņas EPU infrastruktūrā (5.5., 5.6.nodaļas).

Ja skripta darbība noslēdzas ar kļūdu, e-pakalpojums jāveido manuāli (5.4.nodaļa).

## Manuāla e-pakalpojuma projekta izveide un uzstādīšana

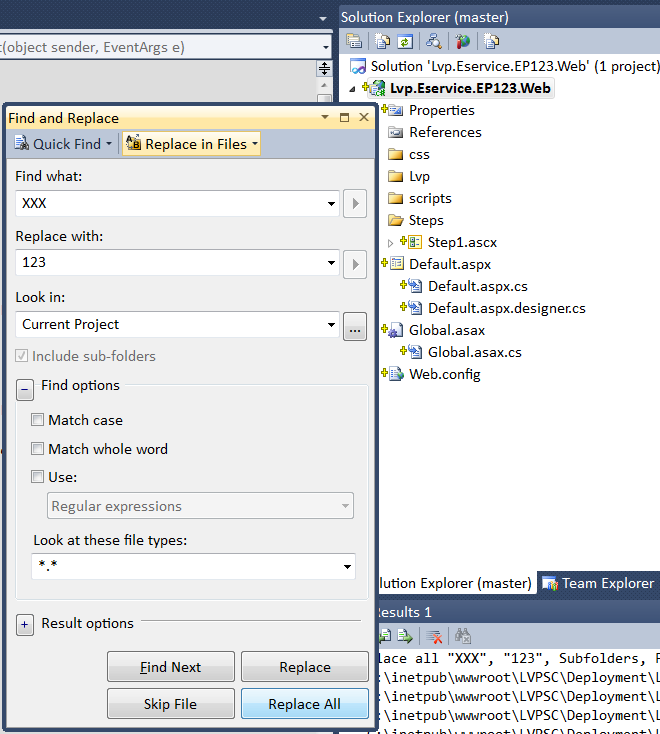
Šajā sadaļā aprakstītās darbības jāveic tikai, ja ir radušās problēmas ar automātisko jauna e-pakalpojuma izveidošanas skriptu (skat. 5.3.nodaļu). Skripts automatizē šajā nodaļā aprakstītās darbības.

### E-pakalpojuma projekta izveide

1. Jāpārkopē *Lvp.Eservice.Template\Lvp.Eservice.EPXXX.Web* mape uz paredzēto aplikācijas atrašanās vietu (līdzi jākopē arī *Reference Assemblies* mape, ja tādas vēl nav mērķa mapē).
2. Jāpārsauc projekta fails (*Lvp.Eservice.EPXXX.Web.csproj*) un mape, kurā tas atrodas, aizstājot *XXX* ar e-pakalpojuma numuru, ko izsniedzis EPU (piemēram, *Lvp.Eservice.EP123.Web.csproj*).

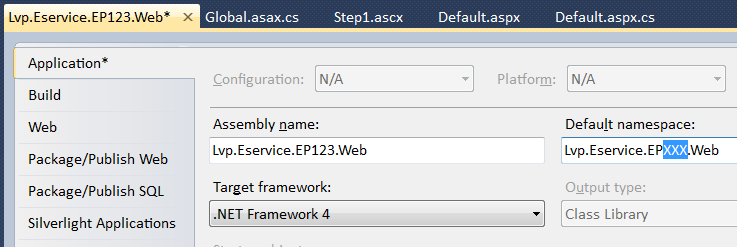
Ja Lvp.Eservice.EPXXX.Web mape tiek pārnesta uz citu mapi, līdzi jāpārnes arī Reference Assemblies mape, jo šablona projektā ir saites uz Reference Assemblies saturu. Citiem vārdiem - Lvp.Eservice.EPXXX.Web un Reference Assemblies jāatrodas blakus vai arī jāmaina Lvp.Eservice.EPXXX.Web projekta dll bibliotēku saites (references).

1. Ar Visual Studio jāatver iepriekš minētais projekts.
2. Projektā jāveic globāla teksta aizstāšana (*replace*), XXX vietā liekot e-pakalpojuma numuru.



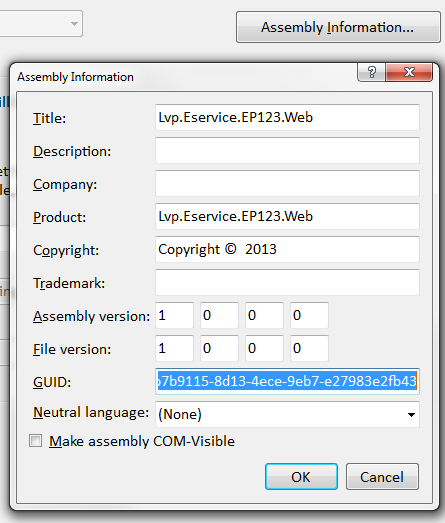
51.attēls. E-pakalpojuma numura aizstāšana

1. Jāveic izmaiņas projekta uzstādījumos, ievietojot e-pakalpojuma numuru *Assembly Name* un *Default Namespace* laukos.



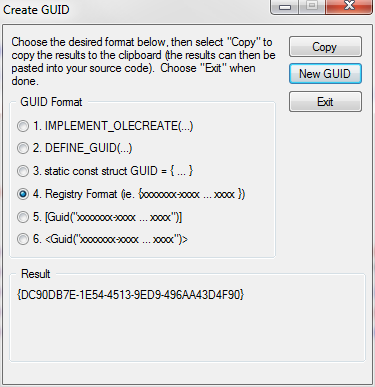
52.attēls. E-pakalpojuma numura aizstāšana projekta iestatījumos

1. Jāuzģenerē un jānomaina projekta identifikators (*GUID*).



53.attēls. E-pakalpojuma GUID maiņa

Jauno identifikatoru var ģenerēt ar VS *GUID gen* rīku.



54.attēls. GUID gen rīka izmantošana

### E-pakalpojuma apakšlietojuma izveide IIS

Priekšnosacījumi: izpildītas 4.4.1.2. un 4.4.1.3.paragrāfu instrukcijas.

Jāpievieno jauns apakšlietojums ***eservices*** lietojumam (skat. 4.4.1.4.paragrāfu) ar šādiem parametriem:

* ***Alias****:* apakšadrese, caur kuru e-pakalpojums būs pieejams, piemēram, EP123;
* ***Application pool***: noklusētā vērtība (*eservices* lietojuma pūls);
* ***Physical path****:* ceļš uz e-pakalpojuma projektu.

Pūla lietotājam jāpiešķir lasīšanas tiesības e-pakalpojuma mapēs un rakstīšanas tiesības e-pakalpojuma *App*\_*Data* mapē.

### E-pakalpojuma reģistrēšana emulatorā

Jauno e-pakalpojumu nepieciešams reģistrēt emulatorā (skatīt 9.1.nodaļu).

## E-pakalpojuma sertifikāta reģistrēšana

Katram ne publiskajam e-pakalpojumam ir nepieciešams ģenerēt sertifikātu. Šī sertifikāta publiskajai atslēgai jābūt reģistrētai arī VISS (jāsazinās ar EPU).

Jāveic sertifikātu uzstādīšana un privātās atslēgas pieejas tiesību piešķiršana, kā tas ir aprakstīts 4.4.2.2.paragrāfā.

## E-pakalpojuma īpašas piekļuves apgabala (realm) reģistrēšana

Lai darbotos SSO, VISS un LVP vides STS konfigurācijā nepieciešams e-pakalpojumam reģistrēt īpašas piekļuves apgabalu (*realm*).

EPU reģistrēto apgabalu nepieciešams piefiksēt e-pakalpojuma konfigurācijas failā (*web.config*):

<microsoft.identityModel>

<service saveBootstrapTokens="true">

…

<audienceUris>

<add value="https://epakvisstv.vraa.gov.lv/Eservices/EP123/1.0"/>

</audienceUris>

…

<federatedAuthentication>

<wsFederation

issuer="https://epakvisstv.vraa.gov.lv/LVP/LVP.STS/v1-1/Default.aspx"

realm="https://epakvisstv.vraa.gov.lv/Eservices/EP123/1.0"/>

</federatedAuthentication>

…

</service>

</microsoft.identityModel>

## Sagatavošanās EDK servisa izmantošanai

Šīs nodaļas izpilde nepieciešama tikai, ja e-pakalpojumā nepieciešams izmantot EDK funkcionalitāti (skat. 7.1.2.sadaļu).

E-pakalpojuma projektam nepieciešams piesaistīt papildu bibliotēku (Add Reference) *Abc.Edk.Client.dll*, kas atrodama Reference Assemblies mapē.

## Sagatavošanās e-paraksta izmantošanai

Šīs nodaļas izpilde nepieciešama tikai, ja e-pakalpojumā nepieciešams iekļaut dokumentu e‑parakstīšanas funkcionalitāti (skat. 7.1.3.sadaļu).

Jāpārliecinās, ka ir uzstādīts e-paraksta sertifikāts un piešķirtas tā privātās atslēgas lasīšanas tiesības (4.4.2.2.paragrāfa pēdējais solis).

Nepieciešams uzstādīt viedkartes izmantošanai nepieciešamos dziņus (drivers), kā tas ir aprakstīts www.eparaksts.lv. Papildus nepieciešams uzstādīt arī GemSafe draiveru programmatūru.

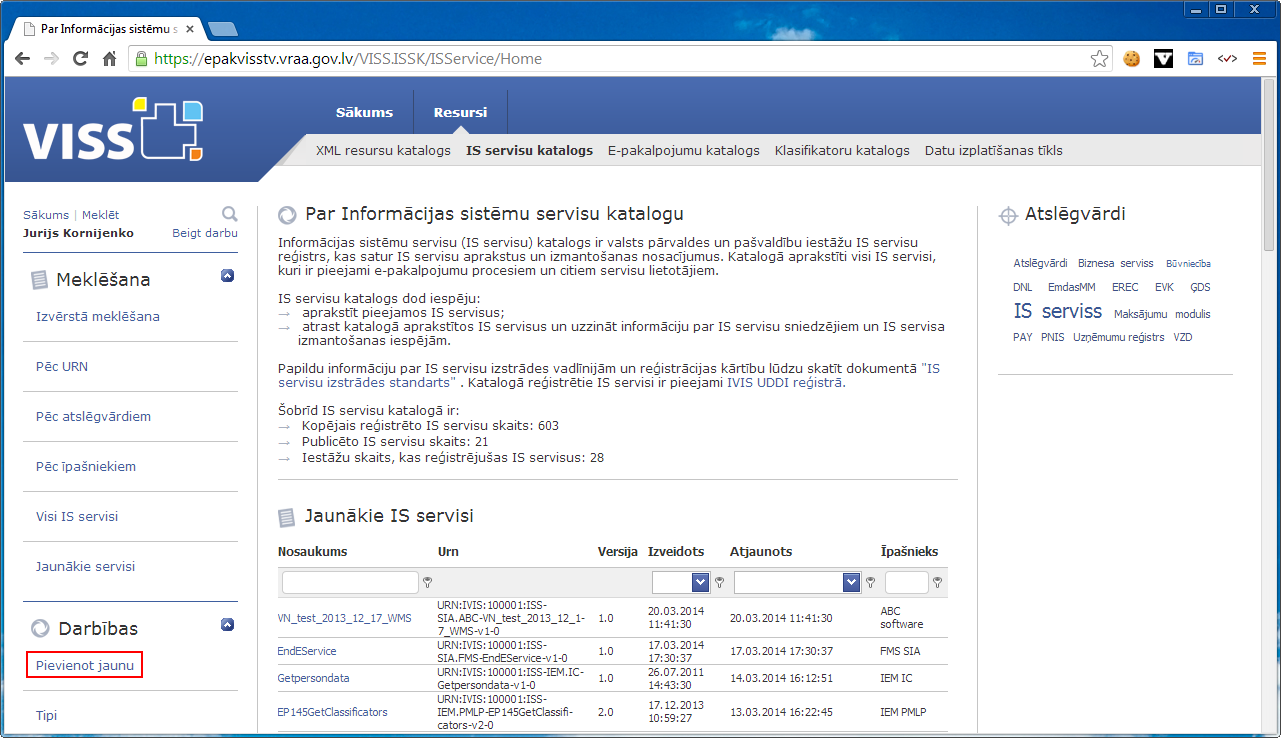
## Sagatavošanās lietotāja profila izmantošanai

Šīs nodaļas izpilde nepieciešama tikai, ja e-pakalpojumā nepieciešams izmantot lietotāja vai lietotāja pārstāvētā uzņēmuma profila informāciju (skat. 7.1.14.sadaļu).

E-pakalpojuma projektam nepieciešams piesaistīt papildu bibliotēkas (Add Reference) *Abc.UserProfile.Base.dll, Abc.UserProfile.Client.dll*, kas atrodamas *Reference Assemblies* mapē.

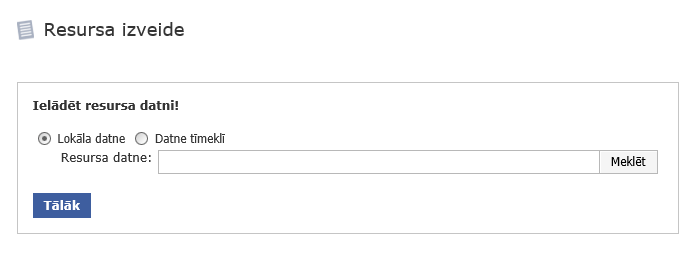
## XML shēmas reģistrācijas VISS

1. Jāpieslēdzas VISS **produkcijas** videi (<https://ivis.eps.gov.lv/RC.WebApp>) ar EPU izsniegto lietotāja vārdu un paroli.
2. Jāaiziet uz sadaļu Resursi -> XML resursu katalogs.
3. Jāizvēlas darbības „Pievienot jaunu”:



55.attēls. Jaunas XSD shēmas pievienošana

1. Jāizvēlas XML shēmu:



56.attēls. Resursa datnes ielāde

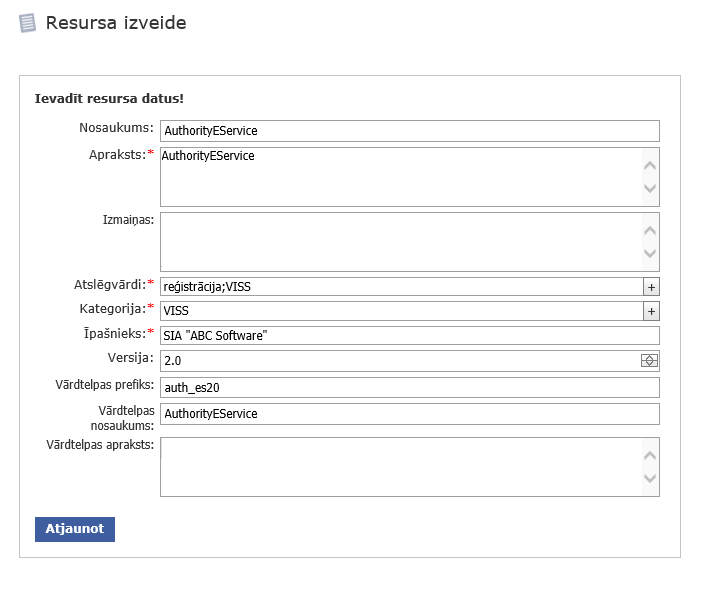
1. Jāizvēlas XML shēmas tipu, parasti „IVIS XML shēma”:



57.attēls. XSD shēmas tipa izvēle

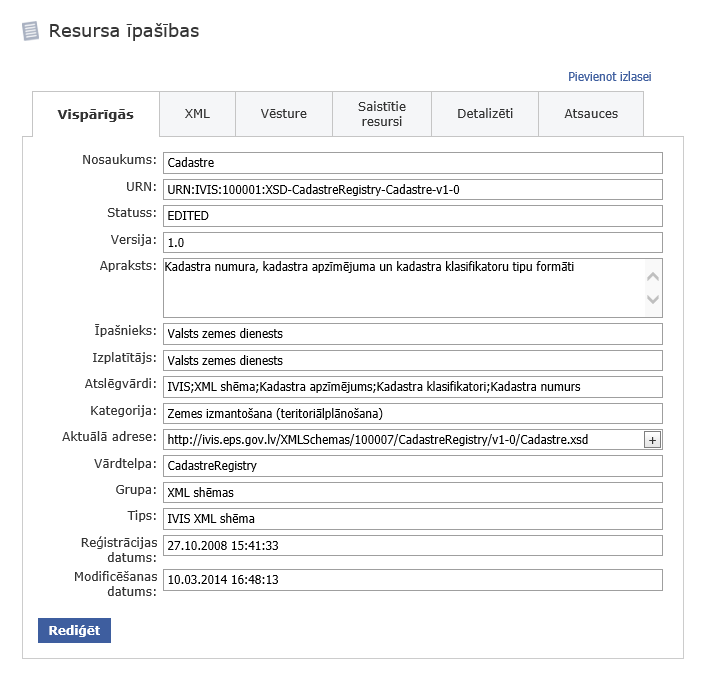
1. Aizpildāmi visi shēmas metadati, tai skaitā:

* Apraksts – shēmas apraksts latviešu valodā;
* Izmaiņas – veiktas izmaiņas, ja notiek shēmas atkārtota pievienošana;
* Atslēgvārdi – shēmu aprakstošie atslēgvārdi, jāizvēlas no esošiem vai jāievieno jauni;
* Kategorija – shēmas piederība kādai kategorijai;
* Versija – shēmas versija. Shēmas tipam „IVIS XML shēma” versija ir vārdtelpas sastāvdaļa. Tāpēc, ja notiek atkārtota shēmas pievienošana, vēlams saglabāt eksistējošo versiju.
* Vārdtelpas apraksts – shēmu hierarhijas nosaukums.



58.attēls. XSD shēmas metadatu apraksta pievienošana

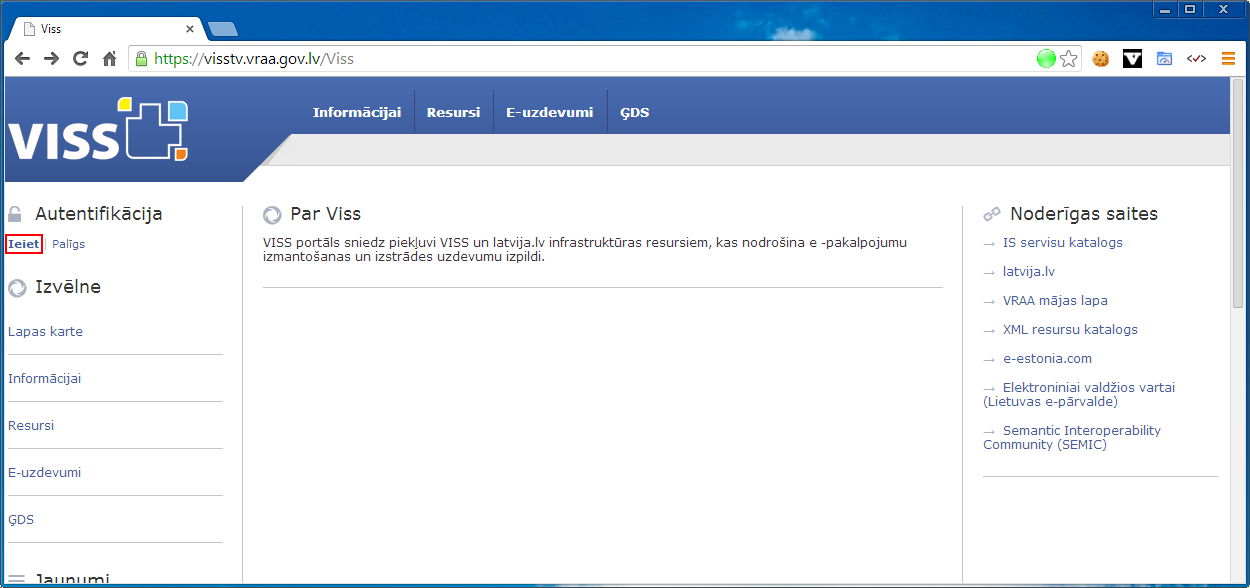
Pēc metadatu aizpildīšanas jānospiež poga „Pievienot” vai „Atjaunot”. Tālāk shēmas pieejamas *online* režīmā pēc norādītās, aktuālās adreses:



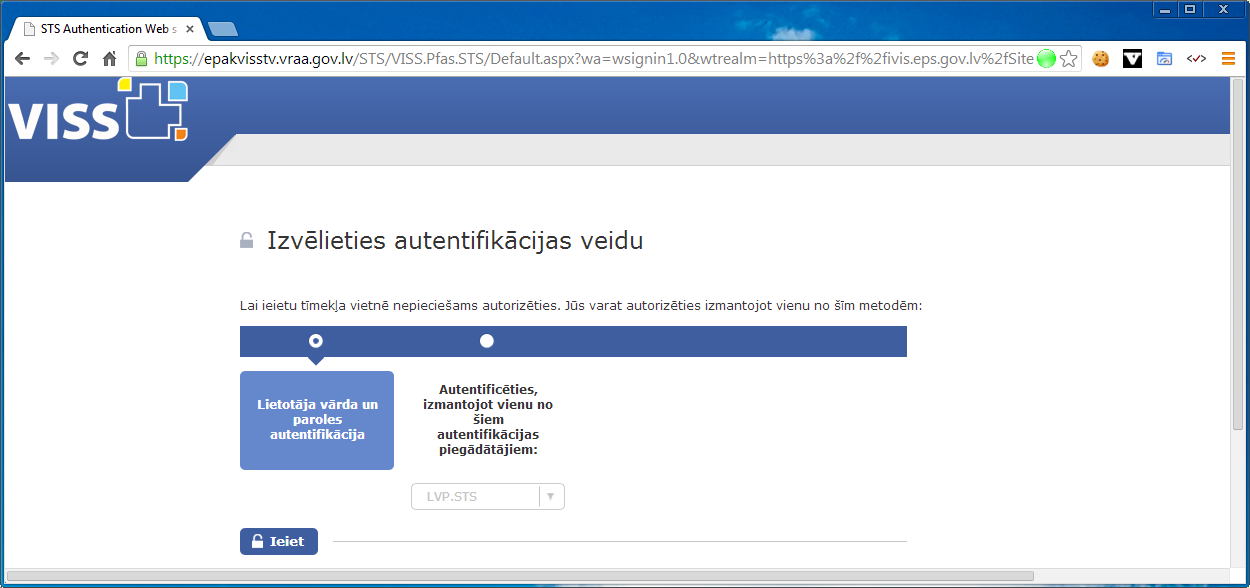
59.attēls. XSD shēmas metadatu apskats

## Integrācijas servisa reģistrācijas VISS

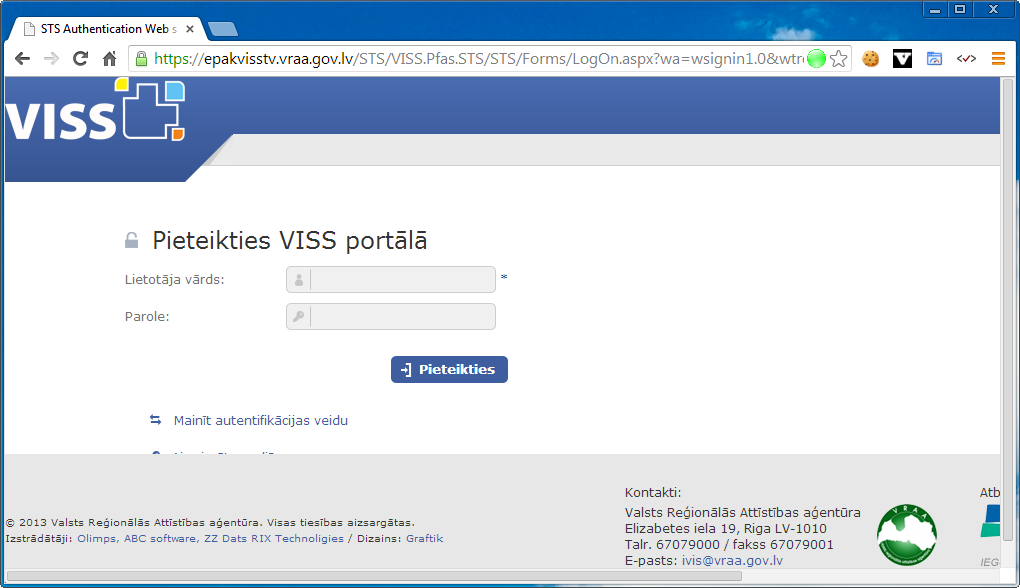
1. Jāpieslēdzas VISS **testa** videi (<https://visstv.vraa.gov.lv/Viss>) ar EPU izsniegto lietotāja vārdu un paroli.



60.attēls. Pieslēgšanās VISS sistēmai

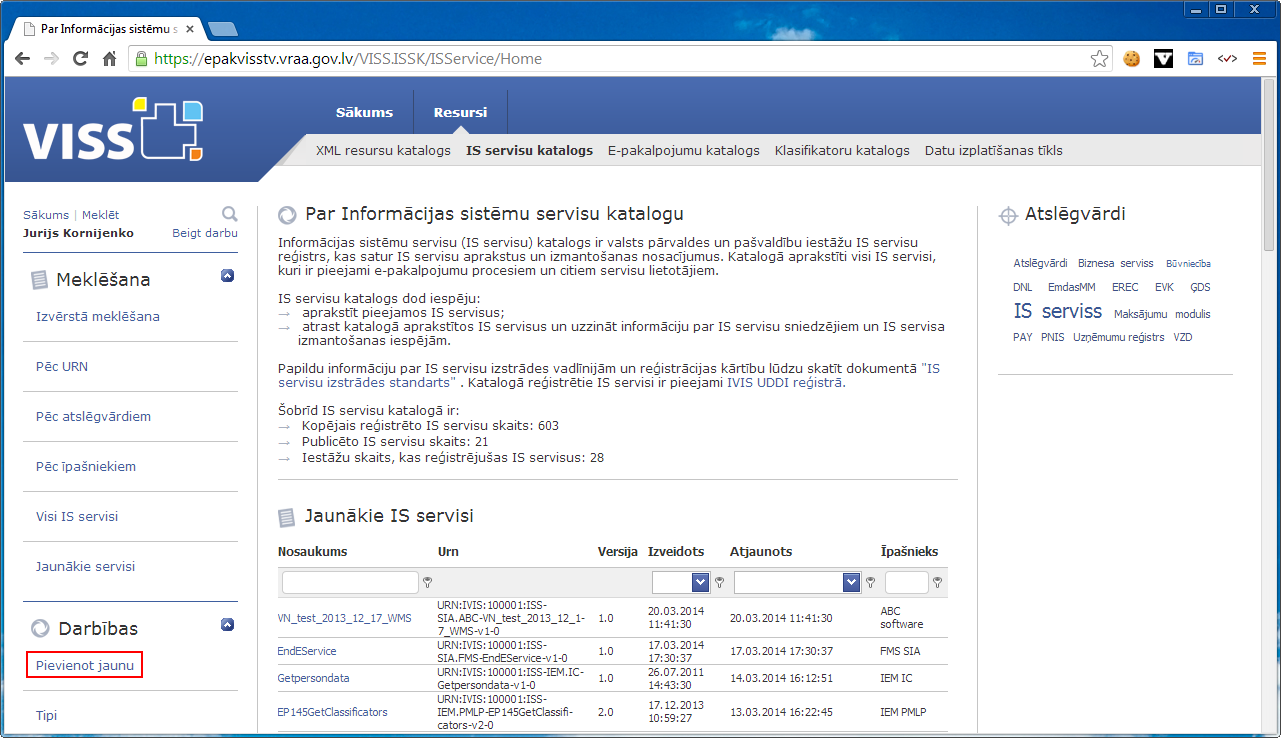


61.attēls. Pieslēgšanās VISS sistēmai



62.attēls. Pieslēgšanās VISS sistēmai

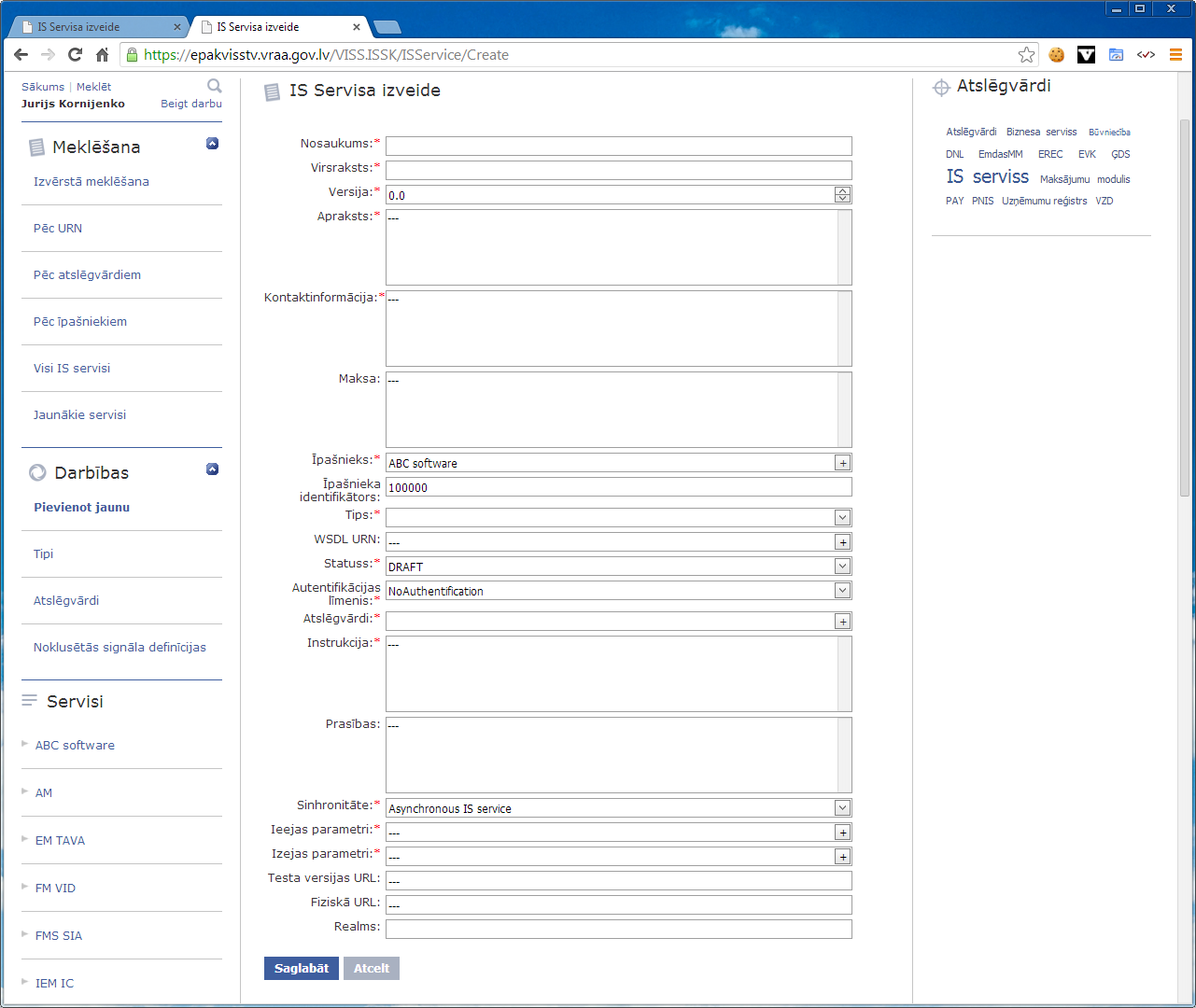
1. Jāaiziet uz sadaļu Resursi -> IS servisu katalogs.
2. Jāizvēlas darbības „Pievienot jaunu”.



63.attēls. Jauna IS servisa pievienošanas uzsākšana

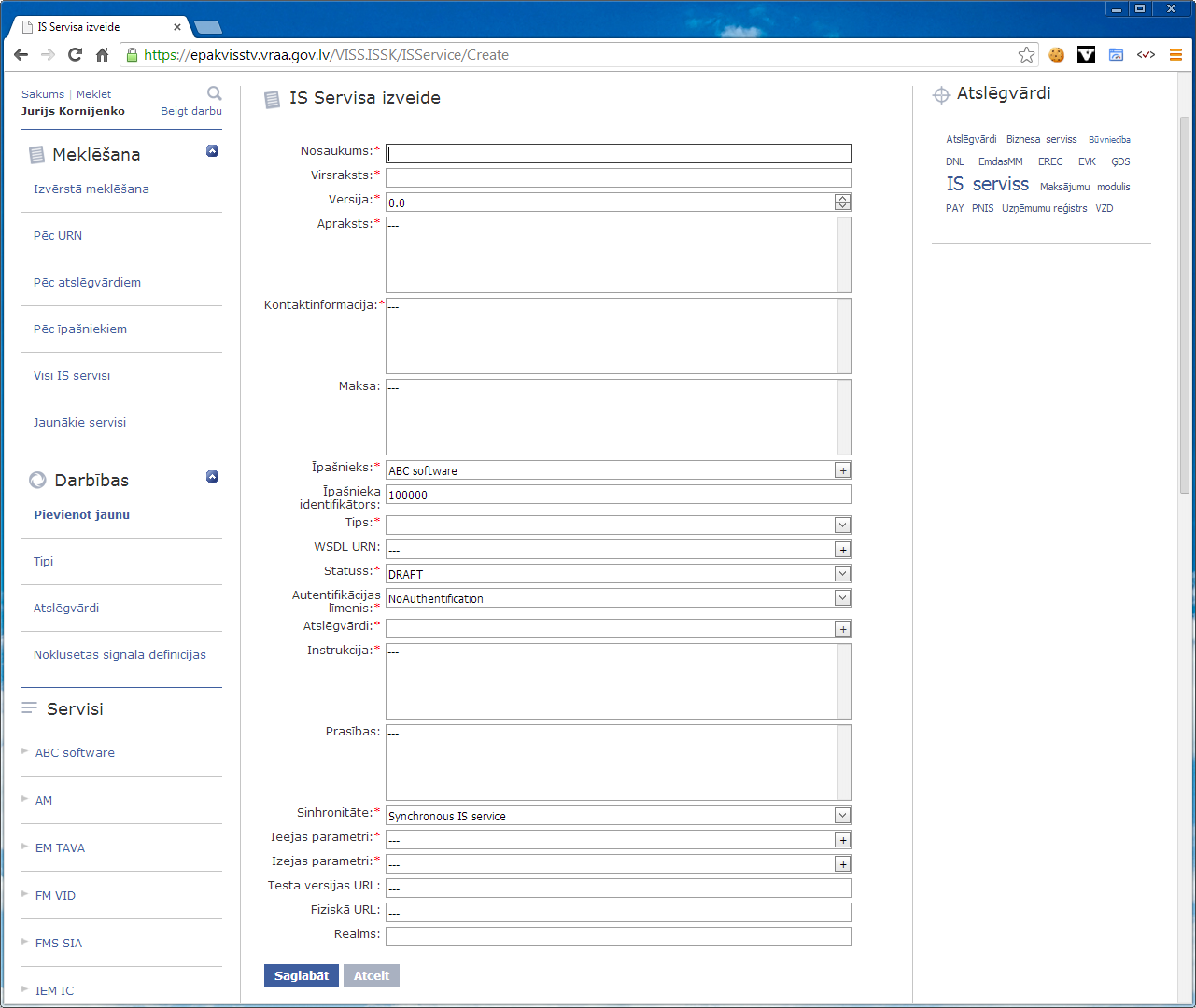
1. Jāaizpilda obligātie lauki. Jānospiež poga „Saglabāt”. Korektai IS izsaukšanai svarīgākie ir šādi lauki:

* **Nosaukums** – servisa nosaukums angļu valodā (ietilps servisa URN);
* **Versija** – parasti 1.0 (ietilps servisa URN);
* **Īpašnieks** – īpašnieka kods ietilps servisa URN;
* **Tips** – parasti izsaucamā servisa tips būs „Integrācijas IS serviss V2”;
* **Autentifikācijas līmenis** – atbilstoši servisa izmantošanas nosacījumiem;
* **Sinhronitāte**;
* **Ieejas parametri** – jābūt norādītai korektajai servisa pieprasījuma XSD shēmai (visi pieprasījumi tiks validēti pēc šīs shēmas);
* **Izejas parametri** – jābūt norādītai korektajai servisa atbildes XSD shēmai (visas atbildes tiks validētas pēc šīs shēmas).



64.attēls. Servisa datu ievade

1. Jānospiež poga „Rediģēt”.
2. Jāievada servisa adrese (Testa versijas URL, Fiziskā URL) un jāsaglabā izmaiņas.

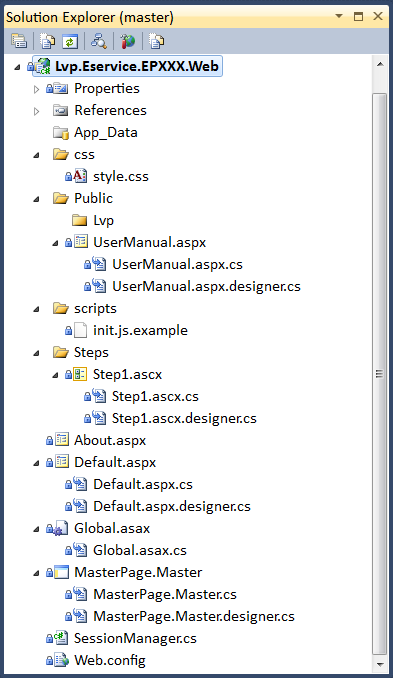


65.attēls. Servisa adreses ievade

1. Pēc aptuveni 5 minūtēm informācijai par IS konfigurāciju būtu jānonāk līdz PS.

# E-pakalpojumu projekts

## Projekta struktūra



66.attēls. E-pakalpojuma projekta struktūra

* **App\_Data:** mape e-pakalpojuma datu datnēm (tai skaitā NASA pierakstīšanās datnei).
* **css\style.css**: vieta e-pakalpojuma CSS stiliem.
* **Public**: mape, kurai piekļuve vienmēr būs atļauta arī neautentificētiem lietotājiem neatkarīgi no e-pakalpojumam nepieciešamā autentifikācijas līmeņa.
* **Public\Lvp**: mape, kurā tiek reģistrēti pieprasījumu apdarinātājai (skat. 7.2.27., 7.2.30. un 7.2.31.sadaļas). Mapei jābūt tukšai.
* **Public\UserManual.aspx:** Lapa ar e-pakalpojuma lietotāja instrukciju. Lapas nosaukumu vai atrašanās vietu mainīt nedrīkst.
* **scripts**: vieta e-pakalpojuma JS skriptiem.
* **scripts\init.js.example**: Piemērs skriptam ar pamata darbībām LVP JS bibliotēku inicializācijai. Pēc noklusējuma netiek izmantots, jo analoģiska skripta versija ir pieejama CDN (*eservice/scripts/lvp-init.js*). Gadījumā, ja e-pakalpojumā nepieciešams apstrādāt papildu portāla notikumus, var modificēt *init.js.example* (pārsaucot par *init.js*) un neizmantot *lvp-init.js* no CDN.
* **Steps**: Mape, kurā var ievietot e-pakalpojuma soļus.
* **Steps\Step1.ascx**: E-pakalpojuma soļa piemērs.
* **About.aspx**: Informācija, kas tiek parādīta lietotājam pirms e-pakalpojuma uzsākšanas un kuru lietotājam ir jāapstiprina (jāpiekrīt). Lapas nosaukumu vai atrašanās vietu mainīt nedrīkst.
* **Default.aspx**: E-pakalpojuma galvenā lapa, kas satur visus pārējos e-pakalpojuma elementus. Lapas nosaukums jāatstāj nemainīgs, lai tā tiktu atvērta, pieprasot e-pakalpojuma saknes URL. Lapas nosaukumu vai atrašanās vietu mainīt nedrīkst.
* **Global.asax**: Lietojuma globālo notikumu apstrādātājs.
* **MasterPage.Master**: Lappušu šablons, kuru izmanto pārējās e-pakalpojuma lapas.
* **SessionManager.cs**: Klase, kura jāizmanto piekļuvei pie ASP.NET sesijas.
* **Web.config**: Lietojuma konfigurācijas datne.

## Jauna soļa pievienošana

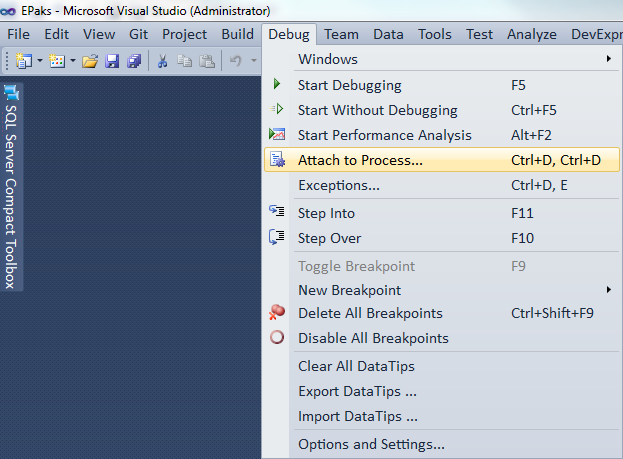
1. Projektam jāpievieno jauns lietotāja kontrolis (\*.ascx).
2. Pievienotā kontroļa bāzes klase jānomaina no UserControl uz LvpWizardStepBase.
3. *Default.aspx* lapā abc:LvpWizard kontrolim jāpievieno jauns solis (abc:LvpWizardStepContainer), norādot:

* **Title**: soļa nosaukums (tiek attēlots augšējā izvēlnē).
* **Src**: ceļš uz izveidoto kontroli.

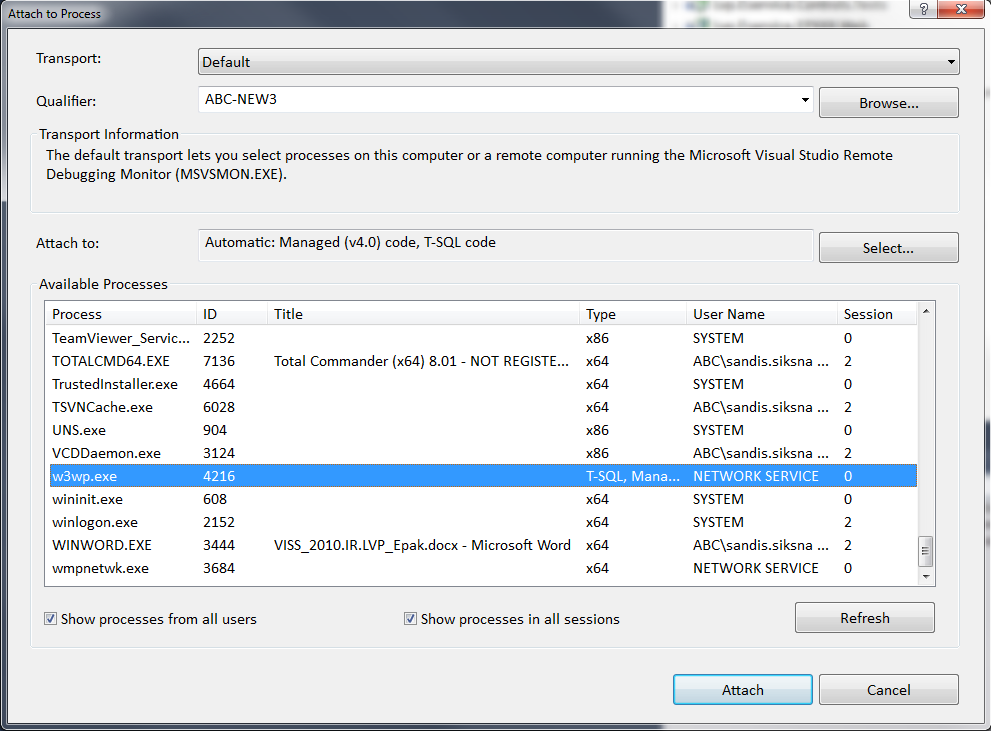
1. Parasti jāpārlabo (*override*) kāda no LvpWizardStepBase metodēm (OnFirstOpen, OnOpen, OnNextStepClick), lai realizētu soļa funkcionalitāti.

## IIS darbināta projekta atkļūdošana (debug)

IIS darbinātu tīmekļa lietojumu var atkļūdot, pieslēdzoties *w3wp.exe* procesam:

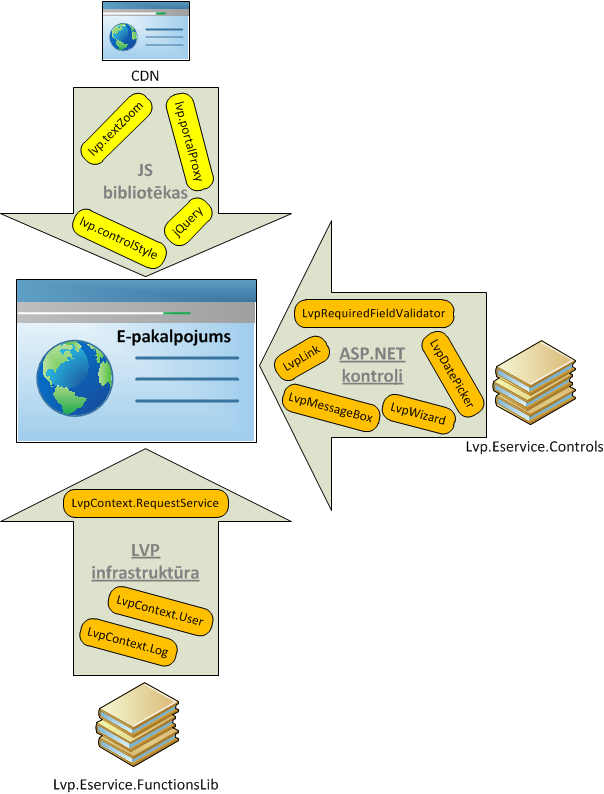


67.attēls. Pieslēgšanās dialoga atvēršana



68.attēls. Pieslēgšanās IIS procesam

# Bibliotēkas e-pakalpojumu izstrādei



69.attēls. Bibliotēkas e-pakalpojumu izstrādei

## .NET bibliotēka Lvp.Eservice.FunctionsLib

Pamata bibliotēkas funkcionalitāte ir pieeja caur *LvpContext* klases statiskajām metodēm. *LvpContext* klase tiek inicializēta lietojuma darbības uzsākšanas sākumā (*Application\_Start*).

### LvpContext.Config

Piedāvā e-pakalpojuma konfigurācijas parametrus no konfigurācijas faila. Pamata parametri tiek izmantoti citu *FunctionsLib* komponentu darbībai. Konfigurācijas parametru aprakstu skatīt 8.nodalījumā.

### LvpContext.EdkService

Nodrošina EDK servisa izsaukšanu (skat. [3]). Visām metodēm pieeja tiek atļauta tikai objektiem, kas atrodas autentificēta lietotāja mapēs. Anonīmiem lietotājiem EDK nav pieejams.

Priekšnosacījumi izmantošanai aprakstīti 5.7.nodaļā.

Izmantošanas piemērs:

string documentId = "URN:IVIS:100266:DOC-2265100-V0.0";

cmisObjectType cmisObject =

LvpContext.EdkService.GetDocumentProperties(documentId);

string name = cmisObject.GetStringPropertyValue(PropertyIds.Name);

string changeToken = cmisObject.GetStringPropertyValue(

PropertyIds.ChangeToken);

cmisContentStreamType content =

LvpContext.EdkService.GetDocumentContent(documentId);

1.tabula

LvpContext.EdkService metodes

| Īpašība vai metode | Apraksts |
| --- | --- |
| cmisObjectType GetDocumentProperties(  string documentId); | Izgūst dokumentu aprakstošo informāciju (laukus atbilstoši objekta tipa definīcijai).  Atbilstošā EDK servisa metode ir getObject.   * documentId: dokumenta identifikators (URN). * Atgriež: sarakstu ar objekta īpašībām (bez pašas dokumenta datnes). |
| string CreateDocument(  Folder folder,  cmisPropertiesType  documentProperties,  cmisContentStreamType  documentContent); | Izveido jaunu dokumentu.  Atbilstošā EDK servisa metode ir createDocument.   * folder: Mape, kurā tiks veidots objekts. E-pakalpojumiem ir pieejamas trīs mapes: * *Temporary*: lietotāja īslaicīgās glabāšanas mape (e-pakalpojumam ir jārūpējas par šīs mapes tīrīšanu pēc tam, kad datne vairs nav nepieciešama); * *Inbox*: lietotāja ieejas mape; * *Outbox*: lietotāja izejas mape. * documentProperties: dokumenta īpašību saraksts. * documentContent: dokumenta datne (nav obligāta). * Atgriež izveidotā dokumenta identifikatoru (URN). |
| void UpdateDocumentProperties(  ref string documentId,  ref string documentChangeToken,  cmisPropertiesType  documentProperties); | Labo dokumenta aprakstošo informāciju.  Atbilstošā EDK servisa metode ir updateProperties.   * documentId: dokumenta identifikators. * documentChangeToken: dokumenta izmaiņu talons jeb pēdējā lauka cmis:changeToken vērtība (var izgūt ar GetDocumentProperties metodi). Minētā vērtība tiek automātiski mainīta pie katras dokumenta manipulācijas. Gadījumā, ja labojot tiek nodots vecs talons (cits lietotājs jau ir paspējis labot dokumentu), serviss atgriež izņēmumu (exception). * documentProperties: laboto dokumenta īpašību saraksts. Īpašības, kuras nav mainītas, padot nedrīkst. |
| void DeleteDocument(  string documentId); | Dzēš dokumentu (pārvieto uz Trash mapi).  Atbilstošā EDK servisa metode ir deleteObject.   * documentId: dokumenta identifikators. |
| cmisContentStreamType  GetDocumentContent(  string documentId); | Izgūst dokumenta datni.  Atbilstošā EDK servisa metode ir getContentStream.   * documentId: dokumenta identifikators. * Atgriež: dokumenta datni ar standarta aprakstošo informāciju. |
| void UploadDocumentContent(  ref string documentId,  ref string documentChangeToken,  cmisContentStreamType  documentContent); | Pievieno vai mana dokumenta datni.  Atbilstošā EDK servisa metode ir setContentStream.   * documentId: dokumenta identifikators. * documentChangeToken: dokumenta izmaiņu talons – pēdējā lauka cmis:changeToken vērtība. * documentContent: dokumenta datne ar standarta aprakstošo informāciju. |
| void DeleteDocumentContent(  ref string documentId,  ref string documentChangeToken); | Dzēš dokumenta datni.  Atbilstošā EDK servisa metode ir deleteContentStream.   * documentId: dokumenta identifikators. * documentChangeToken: dokumenta izmaiņu talons – pēdējā lauka cmis:changeToken vērtība. |
| cmisObjectInFolderContainerType[] GetObjectsInFolder(  Folder folder,  string relativePath = null); | Izgūst mapes saturu (mapes un objekti). Jāņem vērā, ka parasti trīs piedāvātās mapes EDK tiek dalītas vēl sīkāk pēc dokumentu izveidošanas laika (pie dokumenta veidošanas šī nianse tiek slēpta, automātiski ievietojot objektu tekošajam laikam atbilstošajā mapē).  Atbilstošā EDK servisa metode ir getDescendants.   * folder: mape, kuras saturs vai kuras apakšmapju saturs tiks izgūts. * relativePath: relatīvais ceļš zem folder parametrā norādītās mapes. * Atgriež: mapē esošo objektu un mapju sarakstu. |
| void ShareDocumentInFolder(  string documentId,  string folderId); | Pievieno dokumentu vēl kādai mapei. Pievienota tiek norāde uz objektu – pats objekts netiek kopēts. Norādi no paša objekta nevar atšķirt.  Caur šo metodi ir plānots realizēt, piemēram, dokumentu apmaiņu starp e-pakalpojumu un aizmugursistēmu (back-end). E-pakalpojums izveido dokumentu lietotāja mapē, pievieno dokumentu aizmugursistēmas mapei un caur PS nosūta dokumenta identifikatoru aizmugursistēmai. Aizmugursistēma pēc identifikatora atrod un lejupielādē dokumentu.  Atbilstošā EDK servisa metode ir addObjectToFolder.   * documentId: pievienojamā dokumenta identifikators. * folderId: mapes, kurai objekts tiks pievienots, identifikators (URN). Pieejas tiesības mapei netiek pārbaudītas. |
| void StopSharingDocumentInFolder(  string documentId,  string folderId); | Izņem objektu no mapes. Piemēram, ja objekts ir bijis ievietots divās mapēs, izpildot šo metodi, tas paliks vairs tikai vienā mapē.  Atbilstošā EDK servisa metode ir removeObjectFromFolder.   * documentId: pievienojamā dokumenta identifikators. * folderId: mapes, no kuras objekts tiks izņemts, identifikators (URN). Pieejas tiesības mapei netiek pārbaudītas. |

### LvpContext.EsigningService

Nodrošina e-parakstīšanas servisa izsaukšanu (skat. [2] atsauci). Parasti šis elements ir jālieto kopā *ar LvpEsigner* kontroli, kas nodrošina lietotāja interfeisu (skat. 7.2.20.sadaļu).

Priekšnosacījumi izmantošanai aprakstīti 5.8.nodaļā.

Izmantošanas piemērs:

string sessionId = LvpContext.EsigningService.CreateSession(

configurationFileId: null,

userOrSystemName: null,

personCode: LvpContext.User.IsAuthenticated

? LvpContext.User.PersonCode : null,

authenticationLevel: null,

paymentBy: null);

byte[] documentContent =

Encoding.Unicode.GetBytes("Parakstāmās datnes saturs.");

using (var stream = new MemoryStream(documentContent))

{

var documentToSign = new AddDocInMsg(

"Parakstāmais dokuments",

"txt",

"First.txt",

sessionId,

documentContent.Length,

stream);

LvpContext.EsigningService.AddDocument(documentToSign);

}

DivSessionStatus status =

LvpContext.EsigningService.GetSessionStatus(sessionId);

List<DivDocument> documents =

LvpContext.EsigningService.GetDocuments(sessionId);

GetFileOutMsg document =

LvpContext.EsigningService.GetDocument(documentId, sessionId);

LvpContext.EsigningService.CloseSession(sessionId);

2.tabula

LvpContext.EsigningService metodes

| Īpašība vai metode | Apraksts |
| --- | --- |
| string CreateSession(  long? configurationFileId,  string userOrSystemName,  string personCode,  string authenticationLevel,  string paymentBy); | Veic sesijas izveidi un sesijas identifikatora atgriešanu pieprasītājam – jālieto pirms darba uzsākšanas ar e-parakstītāju.   * configurationFileId: izvēlētā konfigurācijas faila ID. Nav obligāts. Ja tiek izmantots specifisks (jauns) konfigurācijas fails, tā definīciju nepieciešams ielādēt ar AddConfigFile metodi (metode netiek piedāvāta caur EsigningService, jo tās izmantošana ir vienreizēja un nav paredzēta e-pakapojuma izpildes laikā). * userOrSystemName: lietotājs vai IS, kas veikusi sesijas izveidi. * personCode: personas kods (izmanto virtuālajam parakstam). * authenticationLevel: autorizācijas līmenis (izmanto virtuālajam parakstam): „trusted”. * paymentBy: kurš maksā (izmanto virtuālajam parakstam): „user”. * Atgriež: Sesijas identifikators. |
| long AddDocument(  AddDocInMsg document); | Veic faila augšupielādi e-parakstītāja DB – jālieto sākotnējo datņu saraksta (parakstāmo failu saraksta) izveidei.   * document: Pievienojamā faila saturs un aprakstošā informācija. * Atgriež: izveidotā dokumenta identifikators. |
| DivSessionStatus GetSessionStatus(  string sessionId); | Veic sesijas statusa atgriešanu – jālieto, ja ir nepieciešams noskaidrot e-parakstītāja stāvokli uz pieprasījuma brīdi.   * sessionId: sesijas identifikators. * Atgriež: objekts ar dažādiem parakstāmo dokumentu statusiem. |
| List<DivDocument> GetDocuments(  string sessionId); | Veic sesijas failu izgūšanu – jālieto pēc e-parakstītāja sesijas darba sesijas izbeigšanas – parakstīto failu saraksta informācijas saņemšanai.   * sessionId: sesijas identifikators. * Atgriež: saraksts ar dokumentiem, kas pieejami parakstītājā. |
| GetFileOutMsg GetDocument(  long documentId,  string sessionId); | Veic faila datnes lejupielādi – jālieto faila datnes izgūšanai no e-parakstītāja DB.   * documentId: dokumenta identifikators. * sessionId: sesijas identifikators. * Atgriež: datne ar pamata aprakstošo informāciju. |
| void CloseSession(  string sessionId); | Veic sesijas dzēšanas pieprasījuma apstrādi – jālieto pēc darba sesijas ar e-parakstītāju izbeigšanas – e-parakstītāja DB resursu atbrīvošanai.   * sessionId: sesijas identifikators. |

### LvpContext.EsignatureValidationService

Nodrošina iespēju e-pakalpojumiem pārbaudīt parakstītus dokumentus un izgūt to metadatus (skat. [2] atsauci).

Izmantošanas piemērs:

DocumentIdentifier identifier;

using (var stream = new MemoryStream(uxUpload.FileBytes))

{

var edoc = new AddEdocInMsg

{

Name = uxUpload.FileName,

Extension = uxUpload.FileName.Split('.').Last(),

Size = uxUpload.FileBytes.Length,

Data = stream

};

identifier = LvpContext.EsignatureValidationService.AddEdoc(edoc);

}

GetFileOutMsg document =

LvpContext.EsignatureValidationService.GetDocument(identifier);

3.tabula

LvpContext.EsignatureValidationService metodes

| Īpašība vai metode | Apraksts |
| --- | --- |
| DocumentIdentifier AddEdoc(  AddEdocInMsg edoc); | Ļauj uzsākt dokumenta validācijas anonīmo procesu bez faila ielādes e-parakstītājā.   * edoc: pievienojamā faila saturs un aprakstošā informācija. * Atgriež: dokumenta un sesijas identifikatoru. |
| List<BrowseAndValidateOutMsg>  BrowseAndValidate(  DocumentIdentifier identifier); | Ļauj validēt pievienotu dokumentu un izgūt dokumenta metadatus.   * identifier: dokumenta un sesijas identifikators. * Atgriež: dokumenta parakstīšanas informāciju un citus metadatus. |
| GetFileOutMsg GetDocument(  DocumentIdentifier identifier); | Veic faila datnes lejupielādi.   * identifier: dokumenta un sesijas identifikators. * Atgriež: datni ar pamata aprakstošo informāciju. |

### LvpContext.Globalization

Nodrošina informāciju par tekošo lietotāja interfeisa valodu.

Izmantošanas piemērs:

uxCurrentLanguage.Text = LvpContext.Globalization.Culture.DisplayName;

Daudzvalodu atbalsta nodrošināšanai tiek izmantots standarta ASP.NET risinājums – *CurrentCulture* uzstādīšana un valodām atbilstošas resursu datnes ar attēlojamo tekstu.

4.tabula

LvpContext.Globalization metodes

|  |  |
| --- | --- |
| Īpašība vai metode | Apraksts |
| CultureInfo Culture { get; } | Informācija par tekošo lietojuma valodu. |

### LvpContext.Log

Nodrošina žurnalēšanas funkcionalitāti – dažādu lietojuma notikumu pierakstīšanu žurnāla failā vai citās informācijas krātuvēs. Komponenti var konfigurēt, izmantojot <system.diagnostics> sekciju konfigurācijas failā.

Izmantošanas piemērs:

LvpContext.Log.Audit("start", 0, "Lietotājs uzsāka e-pakalpojuma izpildi.");

var ex = new ArgumentException("Šim izņēmumam būtu jānāk no catch daļas.");

LvpContext.Log.Exception(ex, "{0}: {1}", ex.GetType().Name, ex.Message);

LvpContext.Log.GlobalException(Request, Server);

LvpContext.Log.Info("Tika atvērts {0}. ceļveža solis.", 5);

LvpContext.Log.Info(

"Anonīms paziņojums, kura daļa pēc 100. simbola tiek nogriezta.",

LogOptions.WithoutUser,

LogOptions.Shorten(maxLength: 100));

5.tabula

LvpContext.Log metodes

| Metode | Apraksts |
| --- | --- |
| void Audit(  string actionCode,  int eventId  string message); | Fiksē notikumu sistēmā. Tiek piefiksēta arī informācija par lietotāju, kas notikumu izraisījis. Informācija tiek rakstīta Audit avotā. (<source name="Audit" sekcija konfigurācijas failā). Pēc noklusētās konfigurācijas informācija tiek saglabāta auditācijas DB (tā ir atšķirība no Info un Exception metodēm, kas pēc noklusējuma informāciju raksta datnē).   * actionCode: darbības kods. * eventId: notikuma identifikators. * message: paziņojums. |
| void Audit(  string actionCode,  int eventId  string messageFormat  params object[]  messageArguments); | Fiksē notikumu sistēmā.   * actionCode: darbības kods. * eventId: notikuma identifikators. * messageFormat: paziņojums ar vietturiem argumentu ievietošanai (analoģiski string.Format funkcijai). * messageArguments: argumenti, kuru vērtības ievieto paziņojumā (analoģiski string.Format funkcijai). |
| void Audit(  string actionCode,  int eventId,  string message,  params ILogTransform[]  messageTransformations); | Fiksē notikumu sistēmā.   * actionCode: darbības kods. * eventId: notikuma identifikators. * message: paziņojums. * messageTransformations: papildu transformācijas, kas jāveic ar ziņojumu (piemēram, teksta maksimālā garuma ierobežošana). Parasti vērtības tiek aizpildītas, izmantojot LogOptions īpašības, piemēram, LogOptions.Shorten(50). Visiem ziņojumiem veicamās transformācijas iespējams norādīt *LvpContext* inicializācijas brīdī (pēc noklusējuma tiek pievienotas transformācijas LogOptions.WithUser un LogOptions.WithTimestamp). |
| void Exception(  Exception ex,  string message) | Žurnalē informāciju par kļūdu. Informācija tiek rakstīta Error avotā, kas pēc noklusējuma veic izvadi teksta datnē.   * ex: izņēmums, kas izraisījis kļūdu. * message: kļūdas paziņojums. |
| void Exception(  Exception ex,  string messageFormat,  params object[]  messageArguments); | Žurnalē informāciju par kļūdu.   * ex: izņēmums, kas izraisījis kļūdu. * messageFormat: kļūdas paziņojums ar vietturiem argumentu ievietošanai (analoģiski string.Format funkcijai). * messageArguments: argumenti, kuru vērtības ievieto paziņojumā (analoģiski string.Format funkcijai). |
| void Exception(  Exception ex,  string message,  params ILogTransform[]  messageTransformations); | Žurnalē informāciju par kļūdu.   * ex: izņēmums, kas izraisījis kļūdu. * message: kļūdas paziņojums. * messageTransformations: papildu transformācijas, kas jāveic ar ziņojumu (piemēram, teksta maksimālā garuma ierobežošana). Parasti vērtības tiek aizpildītas, izmantojot LogOptions īpašības, piemēram, LogOptions.Shorten(50). Visiem ziņojumiem veicamās transformācijas iespējams norādīt *LvpContext* inicializācijas brīdī (pēc noklusējuma tiek pievienotas transformācijas LogOptions.WithUser un LogOptions.WithTimestamp). |
| void GlobalException(  HttpRequest request,  HttpServerUtility server); | Žurnalē informāciju par kļūdu. Informācija tiek rakstīta Error un Audit avotos. Metode paredzēta tikai neapstrādāto e-pakalpojuma kļūdu pierakstīšanai (Application\_Error notikums *Global.asax* datnē).   * request: aktīvais pieprasījums. * server: aktīvā lietotne. |
| void Info(  string message); | Žurnalē informatīvu paziņojumu. Informācija tiek rakstīta Info avotā, kas pēc noklusējuma veic izvadi teksta datnē.   * message: paziņojums. |
| void Info(  string message  params object[]  messageArguments); | Žurnalē informatīvu paziņojumu.   * messageFormat: paziņojums ar vietturiem argumentu ievietošanai (analoģiski string.Format funkcijai). * messageArguments: argumenti, kuru vērtības ievieto paziņojumā (analoģiski string.Format funkcijai). |
| void Info(  string message,  params ILogTransform[]  messageTransformations); | Žurnalē informatīvu paziņojumu.   * message: paziņojums. * messageTransformations: papildu transformācijas, kas jāveic ar ziņojumu (piemēram, teksta maksimālā garuma ierobežošana). Parasti vērtības tiek aizpildītas, izmantojot LogOptions īpašības, piemēram, LogOptions.Shorten(50). Visiem ziņojumiem veicamās transformācijas iespējams norādīt *LvpContext* inicializācijas brīdī (pēc noklusējuma tiek pievienotas transformācijas LogOptions.WithUser un LogOptions.WithTimestamp). |

### LogOptions

Nodrošina piekļuvi ziņojumu apstrādātājiem (izmantojami LvpContext.Log metodēs). Ziņojumu apstrādātāju uzdevums ir modificēt žurnalējamo notikumu tekstu (piemēram, pievienot informāciju par aktīvo lietotāju).

6.tabula

LogOptions īpašības un metodes

|  |  |
| --- | --- |
| Metode | Apraksts |
| ILogTransform WithUser {get;} | Pirms ziņojuma teksta pievieno informāciju par tekošo lietotāju (vārds, uzvārds, personas kods). |
| ILogTransform WithoutUser  { get; } | Apstādina lietotāja informācijas pievienošanu ziņojumam, ja šī opcija ir pievienota pēc WithUser. |
| ILogTransform WithTimestamp  {get;} | Pirms ziņojuma teksta pievieno informāciju par esošo laiku. |
| ILogTransform Shorten(  int maxLength) | Nogriež beigas ziņojumam, ja tā garums pārsniedz norādīto maksimumu.   * maxLength: maksimālais ziņojuma garums. |
| ILogTransform ShortenStart(  int maxLength) | Nogriež sākumu ziņojumam, ja tā garums pārsniedz norādīto maksimumu.   * maxLength: maksimālais ziņojuma garums. |

### LvpContext.OldPayments

Nodrošina ar MM izsaukšanu saistīto funkcionalitāti migrētajiem e-pakalpojumiem. Paredzēts darbam ar MM v1.0 versiju. Nav paredzēts izmantot jaunu e-pakalpojumu izstrādē.

Migrācijā parasti jāizmanto vecā versija, jo ziņojumi caur maksājumu moduli tiek nosūtīti tālāk uz integrācijas servisiem. Tā kā vecajā un jaunajā versijā atšķiras izmantotās pieprasījumu bāzes shēmas (xsd), jaunās versijas izmantošana novestu pie izmaiņu veikšanas integrācijas servisos.

Izmantošanas piemērs:

var requestItem = new IVISServiceExStructure();

requestItem.EServiceID = LvpContext.Config.General.EserviceUrn;

requestItem.ServiceInstanceID = LvpContext.RequestService.TransactionId;

requestItem.Receiver = new ParticipantStructure

{

SystemID = "URN:IVIS:100003:IDDV",

Item = new EmployeePoolStructure

{

AuthorityID = "100026",

AuthorityName = "SIA 'ZZ DATS'",

RoleName = "Administrators",

RoleID = "AUTHORITYEMPLOYER"

}

};

requestItem.Initiator =

LvpContext.OldPayments.NewInitiator<ParticipantStructure>();

requestItem.PaymentInfo =

LvpContext.OldPayments.NewPaymentInfo<PaymentInfoStructure>();

var request = new EP10DataFullStructure();

request.Item = requestItem;

// Aizpilda e-pakalpojumam specifiskus datus

request.EP10Data = CreateEP10Data();

Document response =

LvpContext.RequestService.SubmitSync<EP10DataFullStructure, Document>(

request, "SyncOldPaymentRequest");

LvpContext.OldPayments.OpenPaymentPage(

LvpContext.Config.OldPayment.PaymentUrl,

response.DocumentID.ToString());

7.tabula

LvpContext.OldPayments īpašības un metodes

| Īpašība vai metode | Apraksts |
| --- | --- |
| TParticipantStructure NewInitiator<  TParticipantStructure>(); | Izveido struktūru MM pieprasījuma IVISServiceStructure.Initiator īpašības aizpildīšanai.  Struktūrā pamatā tiek iekļauta informācija par tekošo lietotāju.  Paredzēta kā noklusētā vērtība, kuru e-pakalpojuma kods var tālāk pielāgot savām vajadzībām (uzstādīt citas īpašību vērtības).   * *TParticipantStructure*: tips e-pakalpojuma specifiskai klasei, kura ir ģenerēta no XSD shēmas ParticipantStructure definīcijas. |
| TPaymentInfoStructure NewPaymentInfo<  TPaymentInfoStructure>(  bool getUserInfoFromProfile=true); | Izveido struktūru MM pieprasījuma IVISServiceStructure.PaymentInfo īpašības aizpildīšanai.  Struktūrā pārsvarā tiek iekļauta lvp/payment sekcijā norādītā informācija.  Paredzēta kā noklusētā vērtība, kuru e-pakalpojuma kods var tālāk pielāgot savām vajadzībām (uzstādīt citas īpašību vērtības).   * *TParticipantStructure*: tips e‑pakalpojuma specifiskai klasei, kura ir ģenerēta no XSD shēmas PaymentInfoStructure definīcijas. * getUserInfoFromProfile: nosaka, vai aizpildīt laukus, kuru vērtības tiek izgūtas no lietotājprofila. |
| void OpenPaymentPage(  string url,  string paymentRequestId); | Atver maksājumu veikšanas lapu.   * url: saite uz maksājumu veikšanas lapu. * paymentRequestId: maksājuma identifikators. |

### LvpContext.Payments

Nodrošina ar MM izsaukšanu saistīto funkcionalitāti, detalizēti sk. [4] dokumentā. Paredzēts darbam ar MM v2.0 versiju.

Izmantošanas piemērs:

var requestItem = new IVISServiceStructure();

requestItem.EServiceID = LvpContext.Config.General.EserviceUrn;

requestItem.ServiceInstanceID = LvpContext.RequestService.TransactionId;

requestItem.Receiver = new ParticipantStructure

{

SystemID = "URN:IVIS:100003:IDDV",

Item = new EmployeePoolStructure

{

AuthorityID = "100026",

AuthorityName = "SIA 'ZZ DATS'",

RoleName = "Administrators",

RoleID = "AUTHORITYEMPLOYER"

}

};

requestItem.Initiator =

LvpContext.Payments.NewInitiator<ParticipantStructure>();

requestItem.PaymentInfo =

LvpContext.Payments.NewPaymentInfo<PaymentInfoStructure>();

var request = new IVISServiceDataStructure();

request.Item = requestItem;

PaymentRequestResponseStructure response =

LvpContext.RequestService

.SubmitSync<IVISServiceDataStructure, PaymentRequestResponseStructure>(

request, "SyncPaymentRequest");

LvpContext.Payments.OpenPaymentPage(response.PaymentRequestUrl);

Piemērā tiek izmantota IVISServiceStructure klase, kas ir bāzes struktūra. E-pakalpojumos ir jāveido shēmas, kas manto no šīs struktūras un jāizmanto šīs mantojošās struktūras, lai pēc maksājuma varētu pārsūtīt nepieciešamos datus uz integrācijas servisu.

8.tabula

LvpContext.Payments īpašības un metodes

|  |  |
| --- | --- |
| Īpašība vai metode | Apraksts |
| TParticipantStructure NewInitiator<  TParticipantStructure>(); | Izveido struktūru MM pieprasījuma IVISServiceStructure.Initiator īpašības aizpildīšanai.  Struktūrā pamatā tiek iekļauta informācija par tekošo lietotāju.  Paredzēta kā noklusētā vērtība, kuru e-pakalpojuma kods var tālāk pielāgot savām vajadzībām (uzstādīt citas īpašību vērtības).   * *TParticipantStructure*: tips e-pakalpojuma specifiskai klasei, kura ir ģenerēta no XSD shēmas ParticipantStructure definīcijas. |
| TPaymentInfoStructure NewPaymentInfo<  TPaymentInfoStructure>(  bool getUserInfoFromProfile=true); | Izveido struktūru MM pieprasījuma IVISServiceStructure.PaymentInfo īpašības aizpildīšanai.  Struktūrā pārsvarā tiek iekļauta lvp/payment sekcijā norādītā informācija.  Paredzēta kā noklusētā vērtība, kuru e-pakalpojuma kods var tālāk pielāgot savām vajadzībām (uzstādīt citas īpašību vērtības).   * *TParticipantStructure*: tips e‑pakalpojuma specifiskai klasei, kura ir ģenerēta no XSD shēmas PaymentInfoStructure definīcijas. * getUserInfoFromProfile: nosaka, vai aizpildīt laukus, kuru vērtības tiek izgūtas no lietotājprofila. |
| void OpenPaymentPage(string url); | Atver maksājumu veikšanas lapu.   * url: saite uz maksājumu veikšanas lapu. Saitei jāsatur maksājuma identifikators. |

### LvpContext.RequestService

Nodrošina PS izsaukšanu.

Izmantošanas piemērs:

// Jāizsauc pirms pirmā integrācijas servisu izsaukuma.

LvpContext.RequestService.StartTransaction();

try

{

ResultStructure result = LvpContext.RequestService

.SubmitSync<CalculationStructure,ResultStructure>(

request,

"Calculate-Sync");

base.ShowSuccess("Servisa izsaukums veiksmīgi noslēdzies.");

}

catch (RequestServiceException ex)

{

// Pieprasījumu serviss atgrieza kļūdu

}

// Jāizsauc, lietotājam beidzot e-pakalpojumu

LvpContext.RequestService.EndTransaction();

9.tabula

LvpContext.RequestService īpašības un metodes

| **Īpašība vai metode** | **Apraksts** |
| --- | --- |
| string TransactionId { get; } | Aktīvās PS transakcijas identifikators. |
| bool HasTransaction { get; } | Pazīme, vai e-pakalpojums ir uzsācis PS transakciju. |
| void StartTransaction(); | Uzsāk PS transakciju. Parasti transakcija ir jāuzsāk vienu reizi pakalpojuma laikā - pirms pirmā SubmitSync vai SubmitAsync izsaukuma. |
| void EndTransaction(); | Izbeidz PS transakciju. Parasti transakcija jāizbeidz vienu reizi – e-pakalpojuma beigās. |
| TResponse SubmitSync<TRequest, TResponse>(  TRequest request,  string messageNameInConfig); | Veic sinhrono PS izsaukšanu. Gadījumos, kad PS atgriež kļūdu sarakstu, kurā vismaz vienam no elementiem EventType ir error, tiek izraisīts RequestServiceException. Gadījumos, kad ir nepieciešama specifiskāka atgriezto kļūdu apstrāde, ir iespējams pārlabot (override) RequestService HandleErrors metodi.   * *TRequest*: pieprasījuma datu tips (Body daļai). * *TResponse*: atbildes datu tips (Body daļai). * request: pieprasījuma dati (pieprasījuma Body daļai). * messageNameInConfig: nosaukums ziņojuma konfigurācijai (<message name="…" konfigurācijas datnē), kura tiek izmantota integrācijas servisa URN, pieprasījuma un atbildes tipa URN un citu pieprasījuma parametru aizpildīšanai. * Atgriež: PS atbildi (Body daļu). |
| TResponse SubmitSync<TRequest, TResponse>(  TRequest request,  string messageNameInConfig,  out ErrorStructure[]  errorList); | Veic sinhrono PS izsaukšanu. Gadījumos, kad PS atgriež kļūdu sarakstu, šis saraksts tiek atgriezts errors parametrā. Netiek izraisīts izņēmums, ja kļūdu izraisījis integrācijas serviss (saņemot kļūdu, ko izraisījis pieprasījumu serviss, izņēmums joprojām tiek izraisīts).   * *TRequest*: pieprasījuma datu tips (Body daļai). * *TResponse*: atbildes datu tips (Body daļai). * request: pieprasījuma dati (pieprasījuma Body daļai). * messageNameInConfig: nosaukums ziņojuma konfigurācijai (<message name="…" konfigurācijas datnē), kura tiek izmantota integrācijas servisa URN, pieprasījuma un atbildes tipa URN un citu pieprasījuma parametru aizpildīšanai. * errorList: Kļūdu sarakstu no PS atbildes <c>Head</c> daļas. * Atgriež: PS atbildi (Body daļu). |
| TResponse SubmitSync<TRequest, TResponse>(  TRequest request,  string messageNameInConfig,  out ResponseHeaderStructure  head); | Veic sinhrono PS izsaukšanu. Gadījumos, kad PS atgriež kļūdu sarakstu, šis saraksts tiek atgriezts head parametrā. Netiek izraisīts izņēmums, ja kļūdu izraisījis integrācijas serviss (saņemot kļūdu, ko izraisījis pieprasījumu serviss, izņēmums joprojām tiek izraisīts).   * *TRequest*: pieprasījuma datu tips (Body daļai). * *TResponse*: atbildes datu tips (Body daļai). * request: pieprasījuma dati (pieprasījuma Body daļai). * messageNameInConfig: nosaukums ziņojuma konfigurācijai (<message name="…" konfigurācijas datnē), kura tiek izmantota integrācijas servisa URN, pieprasījuma un atbildes tipa URN un citu pieprasījuma parametru aizpildīšanai. * head: PS atbildes Head daļa. * Atgriež: PS atbildi (Body daļu). |
| string SubmitAsync<TRequest>(  TRequest request,  string messageNameInConfig); | Nosūta PS asinhrono pieprasījumu.   * *TRequest*: pieprasījuma datu tips. * request: pieprasījuma dati (pieprasījuma Body daļai). * messageNameInConfig: nosaukums ziņojuma konfigurācijai konfigurācijas datnē. * Atgriež: CorrelationID, kuru jāizmanto asinhronās atbildes izgūšanai. |
| string SubmitAsync<TRequest>(  TRequest request,  string messageNameInConfig,  Guid correlationId); | Nosūta PS asinhrono pieprasījumu.   * *TRequest*: pieprasījuma datu tips. * request: pieprasījuma dati (pieprasījuma Body daļai) * messageNameInConfig: nosaukums ziņojuma konfigurācijai konfigurācijas datnē. * correlationId: Pieprasījuma Head daļas CorrelationID lauka vērtība. * Atgriež: CorrelationID, kuru jāizmanto asinhronās atbildes izgūšanai. |
| string SubmitAsync<TRequest>(  TRequest request,  string messageNameInConfig,  out ErrorStructure[]  errorList); | Nosūta PS asinhrono pieprasījumu.   * *TRequest*: pieprasījuma datu tips. * request: pieprasījuma dati (pieprasījuma Body daļai) * messageNameInConfig: nosaukums ziņojuma konfigurācijai konfigurācijas datnē. * errorList: Kļūdu sarakstu no PS atbildes Head daļas. * Atgriež: CorrelationID, kuru jāizmanto asinhronās atbildes izgūšanai. |
| string SubmitAsync<TRequest>(  TRequest request,  string messageNameInConfig,  Guid correlationId,  out ErrorStructure[]  errorList); | Nosūta PS asinhrono pieprasījumu.   * *TRequest*: pieprasījuma datu tips. * request: pieprasījuma dati (pieprasījuma Body daļai) * messageNameInConfig: nosaukums ziņojuma konfigurācijai konfigurācijas datnē. * correlationId: Pieprasījuma Head daļas CorrelationID lauka vērtība. * errorList: Kļūdu sarakstu no PS atbildes Head daļas. * Atgriež: CorrelationID, kuru jāizmanto asinhronās atbildes izgūšanai. |
| string SubmitAsync<TRequest>(  TRequest request,  string messageNameInConfig,  out ResponseHeaderStructure  head); | Nosūta PS asinhrono pieprasījumu.   * *TRequest*: pieprasījuma datu tips. * request: pieprasījuma dati (pieprasījuma Body daļai) * messageNameInConfig: nosaukums ziņojuma konfigurācijai konfigurācijas datnē. * head: PS atbildes Head daļa. * Atgriež: CorrelationID, kuru jāizmanto asinhronās atbildes izgūšanai. |
| string SubmitAsync<TRequest>(  TRequest request,  string messageNameInConfig,  Guid correlationId,  out ResponseHeaderStructure  head); | Nosūta PS asinhrono pieprasījumu.   * *TRequest*: pieprasījuma datu tips. * request: pieprasījuma dati (pieprasījuma Body daļai) * messageNameInConfig: nosaukums ziņojuma konfigurācijai konfigurācijas datnē. * correlationId: Pieprasījuma Head daļas CorrelationID lauka vērtība. * head: PS atbildes Head daļa. * Atgriež: CorrelationID, kuru jāizmanto asinhronās atbildes izgūšanai. |
| string SubmitAsyncWithinNewTransaction(  TRequest request,  string messageNameInConfig,  IMessageSettings  messageSettings,  string systemId | Metode nav paredzēta izmantošanai e-pakalpojumos (tikai iekšējai *FunctionsLib* lietošanai). |
| TResponse Poll<TResponse>(  string correlationId,  string messageNameInConfig); | Izgūst asinhronā pieprasījuma atbildi.   * *TResponse*: atbildes datu tips. * correlationId: identifikators, kas tika atgriezts, nosūtot asinhrono pieprasījumu. * messageNameInConfig: nosaukums ziņojuma konfigurācijai konfigurācijas datnē. * Atgriež: PS atbildi (body daļu). |
| TResponse Poll<TResponse>(  string correlationId,  string messageNameInConfig,  out ErrorStructure[]  errorList); | Izgūst asinhronā pieprasījuma atbildi. Gadījumos, kad PS atgriež kļūdu sarakstu, šis saraksts tiek atgriezts errorList parametrā. Netiek izraisīts izņēmums, ja kļūdu izraisījis integrācijas serviss (saņemot kļūdu, ko izraisījis pieprasījumu serviss, izņēmums joprojām tiek izraisīts).   * *TResponse*: atbildes datu tips. * correlationId: identifikators, kas tika atgriezts, nosūtot asinhrono pieprasījumu. * messageNameInConfig: nosaukums ziņojuma konfigurācijai konfigurācijas datnē. * errorList: Kļūdu sarakstu no PS atbildes Head daļas. * Atgriež: PS atbildi (body daļu). |
| TResponse Poll<TResponse>(  string correlationId,  string messageNameInConfig,  out ResponseHeaderStructure  head); | Izgūst asinhronā pieprasījuma atbildi. Gadījumos, kad PS atgriež kļūdu sarakstu, šis saraksts tiek atgriezts errorList parametrā. Netiek izraisīts izņēmums, ja kļūdu izraisījis integrācijas serviss (saņemot kļūdu, ko izraisījis pieprasījumu serviss, izņēmums joprojām tiek izraisīts).   * *TResponse*: atbildes datu tips. * correlationId: identifikators, kas tika atgriezts, nosūtot asinhrono pieprasījumu. * messageNameInConfig: nosaukums ziņojuma konfigurācijai konfigurācijas datnē. * head: PS atbildes Head daļa. * Atgriež: PS atbildi (body daļu). |
| RequestStatus GetAsyncStatus(  string correlationId); | Izgūst asinhronā pieprasījuma statusu.   * correlationId: identifikators, kas tika atgriezts, nosūtot asinhrono pieprasījumu. * Atgriež: statusu. |
| bool DisposeAsyncRequest(  string correlationId); | Padara asinhrono pieprasījumu neaktīvu.   * correlationId: identifikators, kas tika atgriezts, nosūtot asinhrono pieprasījumu. * Atgriež: true veiksmīgas operācijas gadījumā vai false neveiksmes gadījumā. |

### RequestServiceException

Izņēmums, kas tiek izraisīts, no PS saņemot kļūdas atbildes Head daļā (sīkākam izraisīšanas aprakstam skat. 7.1.8.sadaļu).

10.tabula

RequestServiceException īpašības

|  |  |
| --- | --- |
| Īpašība | Apraksts |
| List<ErrorStructure> Errors  { get; } | Saraksts ar kļūdām (no atbildes Head daļas). |
| bool ContainsRequestServiceError  { get; } | Vai kļūdu saraksts satur vismaz vienu kļūdu, ko izraisījis PS. |
| bool ContainsIntegrationServiceErrors  { get; } | Vai kļūdu saraksts satur vismaz vienu kļūdu, kas saņemta no integrācijas servisa (caur PS). |

### ErrorStructure

Kļūdu struktūra no PS ziņojumu Head daļā.

11.tabula

ErrorStructure papildu īpašības, kas nav XSD shēmā

|  |  |
| --- | --- |
| Īpašība | Apraksts |
| ErrorSource Source { get; } | Pazīme, vai kļūda saņemta no PS vai no integrācijas servisa. |
| int ErrorId { get; } | Kļūdas numurs, kas izgūts no kļūdas koda (URN). |
| RequestServiceErrorId RequestServiceErrorId  { get; } | Kļūdas numurs, kas izgūts no kļūdas koda (URN). Atšifrē tikai PS kļūdas. |

### LvpContext.User

Nodrošina informāciju par aktīvo lietotāju. Informācija pārsvarā tiek izgūta no drošības talona.

Izmantošanas piemērs:

string halloUser = string.Format(

"Esi sveicināts {0} {1}!",

LvpContext.User.FirstName,

LvpContext.User.LastName);

12.tabula

LvpContext.User īpašības un metodes

| Īpašība | Apraksts |
| --- | --- |
| string FirstName { get; } | Lietotāja vārds. |
| string LastName { get; } | Lietotāja uzvārds. |
| string PersonCode { get; } | Lietotāja personas kods. |
| string AuthenticationProviderId  { get; } | Identifikators sistēmai, kura veikusi lietotāja autentifikāciju (piemēram, e-ID vai Swedbank). |
| AuthProviderType AuthenticationProviderType{ get; } | Lietotāja autentifikāciju veikušās sistēmas veids.  Rezultātam iespējamas šādas vērtības:   * *Unavailable* – veids nav nosakāms (lietotājs nav autentificējies). * *Undeclared* – autentifikācija ar e-pastu (piemēram, latvija.lv lietotājs). * *Bank* – autentifikāciju veikusi banka. * *Esignature* – autentifikācija ar e-parakstu. |
| string NameIdentifier { get; } | Lietotāja identifikators. |
| UserType UserType { get; } | Lietotāja (autentifikācijas) veids. |
| bool IsAuthenticated { get; } | Pazīme, vai lietotājs ir autentificēts. |
| SecurityToken Token { get; } | Īpašība nav paredzēta tiešai izmantošanai e-pakalpojumos. Lietotāja drošības talons, kas ar STS starpniecību var tikt apmainīts pret citiem taloniem, piemēram, servisu izsaukšanai. |
| bool IsUserAuthenticationRequired(  HttpContext context); | Metode nav paredzēta tiešai izmantošanai e-pakalpojumos. Atbilstoši e-pakalpojuma konfigurācijai nosaka, vai ir atļauta anonīma piekļuve.   * *context*: tekošā HTTP pieprasījuma konteksts. * Atgriež: true, ja lietotāju nepieciešams pārvirzīt uz autentifikācijas lapu vai false, ja lietotājs var turpināt darbu ar e-pakalpojumu. |
| bool IsAuthenticationAccepted(); | Metode nav paredzēta tiešai izmantošanai e-pakalpojumos.  Nosaka, vai autentifikācija ir atbilstoša (pieņemama) e-pakalpojuma konfigurācijai. |

### LvpContext.UserProfile

Nodrošina informāciju par aktīvā lietotāju profilu. Informācija pieejama tikai autentificētiem lietotājiem.

Priekšnosacījumi izmantošanai aprakstīti 5.9.nodaļā.

Izmantošanas piemērs:

string email = LvpContext.UserProfile.GetProperty("Email") as string;

ProfilePropertiesList properties = LvpContext.UserProfile.GetProperties(

ProfilePropertyName.Street,

ProfilePropertyName.FlatNumber);

string street =

properties[ProfilePropertyName.Street.ToString()].Value as string;

int? number =

properties[ProfilePropertyName.FlatNumber.ToString()].Value as int?;

string companyName = LvpContext.UserProfile.GetProperty(

ProfileType.Company,

ProfilePropertyName.CompanyName) as string;

LvpContext.UserProfile.SetProperties(

new ProfileProperty(ProfilePropertyName.Email.ToString(), "mail@box.com"),

new ProfileProperty(ProfilePropertyName.GetInfoOnEmail.ToString(), true));

13.tabula

LvpContext.User metodes

| Īpašība | Apraksts |
| --- | --- |
| object GetProperty(  string propertyName); | Izgūst vienas profila īpašības vērtību. Ja zināms, ka būs nepieciešams izgūt vairākas īpašības, veiktspējas apsvērumu dēļ labāk izmantot metodes, kas izgūst īpašību sarakstu (skatīt zemāk).   * propertyName: īpašības nosaukums. * Atgriež: īpašības vērtību. Katrai īpašībai ir savs datu tips. Iespējamie vērtības datu tipi ir bool, string, int un DateTime. |
| object GetProperty(  ProfilePropertyName propertyName); | Izgūst vienas profila īpašības vērtību.   * propertyName: īpašības nosaukums. * Atgriež: īpašības vērtību. |
| object GetProperty(  ProfileType? profileType,  string propertyName); | Izgūst vienas profila īpašības vērtību.   * profileType: norāda, vai izgūt paša lietotāja (fiziska persona) vai arī lietotāja pārstāvētā uzņēmuma profilu (tikai juridisko personu lietotājiem). * propertyName: īpašības nosaukums. * Atgriež: īpašības vērtību. |
| object GetProperty(  ProfileType? profileType,  ProfilePropertyName propertyName); | Izgūst vienas profila īpašības vērtību.   * profileType: norāda, vai izgūt paša lietotāja (fiziska persona) vai arī lietotāja pārstāvētā uzņēmuma profilu (tikai juridisko personu lietotājiem). * propertyName: īpašības nosaukums. * Atgriež: īpašības vērtību. |
| ProfilePropertiesList GetProperties(  params string[] propertyNames); | Izgūst vairāku profila īpašību vērtības. Piedāvā labāku veiktspēju salīdzinājumā ar vairāku īpašību izgūšanu pa vienai (metodes augstāk).   * propertyNames: īpašību nosaukumi. * Atgriež: īpašību sarakstu. Atgrieztais datu tips ir List<ProfileProperty> atvasinājums, kuram pievienota iespēja izgūt saraksta vienumu pēc īpašības nosaukuma. |
| ProfilePropertiesList GetProperties(  params ProfilePropertyName[]  propertyNames); | Izgūst vairāku profila īpašību vērtības.   * propertyNames: īpašību nosaukumi. * Atgriež: īpašību sarakstu. |
| ProfilePropertiesList GetProperties(  ProfileType? profileType,  params string[] propertyNames); | Izgūst vairāku profila īpašību vērtības.   * profileType: norāda, vai izgūt lietotāja vai arī lietotāja pārstāvētā uzņēmuma profilu. * propertyNames: īpašību nosaukumi. * Atgriež: īpašību sarakstu. |
| ProfilePropertiesList GetProperties(  ProfileType? profileType,  params ProfilePropertyName[]  propertyNames); | Izgūst vairāku profila īpašību vērtības.   * profileType: norāda, vai izgūt lietotāja vai arī lietotāja pārstāvētā uzņēmuma profilu. * propertyNames: īpašību nosaukumi. * Atgriež: īpašību sarakstu. |
| void SetProperty(  ProfileProperty profileProperty); | Uzstāda profila īpašības vērtību.   * profileProperty: īpašības nosaukums un uzstādāmā vērtība. |
| void SetProperty(  ProfileType? profileType,  ProfileProperty profileProperty); | Uzstāda profila īpašības vērtību.   * profileType: norāda, vai modificēt lietotāja vai arī lietotāja pārstāvētā uzņēmuma profilu. * profileProperty: īpašības nosaukums un uzstādāmā vērtība. |
| void SetProperties(  params ProfileProperty[]  profileProperties); | Uzstāda vairāku profila īpašību vērtības. Labāka veiktspēja salīdzinājumā ar vairākiem SetProperty izsaukumiem.   * profileProperty: uzstādāmo īpašību saraksts. |
| void SetProperties(  ProfileType? profileType,  params ProfileProperty[]  profileProperties); | Uzstāda vairāku profila īpašību vērtības.   * profileType: norāda, vai modificēt lietotāja vai arī lietotāja pārstāvētā uzņēmuma profilu. * profileProperty: uzstādāmo īpašību saraksts. |

### Sesijas izmantošana

Piekļuve ASP.NET sesijai jāveic caur katrā e-pakalpojumā definētu, tipizētu klasi, kas manto no SessionManagerBase.

Tipizētas sesijas datu glabāšanas klases piemērs:

public class SessionManager : SessionManagerBase<SessionManager>

{

public string SelectedPersonDisplayName { get; set; }

public bool SkipToHiddenStep { get; set; }

public List<Person> Friends { get; private set; }

public SessionManager()

{

this.Friends = new List<Person>();

}

}

Sesijas izmantošana:

SessionManager.Current.SkipToHiddenStep = false;

14.tabula

SessionManagerBase<T> īpašības

|  |  |
| --- | --- |
| Īpašība vai metode | Apraksts |
| T Current { get; } | Izgūst sesijas objektu. |
| void Reset(); | Notīra sesijas objektu. |

### IBodySerializer

Gadījumiem, kad nepieciešams specifiski serializēt PS pieprasījuma vai atbildes Body daļu, paredzēts izmantot IBodySerializer implementējošu klasi. Šāda situācija var būt novērojama gadījumos, kad pārsūtāmo datu XSD shēmai jāatbilst kādam standartam. Lai pielāgotais serializators tiktu izmantots, tā tipa nosaukumu nepieciešams norādīt konfigurācijas faila lvp/messages/message/bodySerializer elementā (skat. 8.3.nodaļu).

15.tabula

IBodySerializer īpašības un metodes

|  |  |
| --- | --- |
| Īpašība | Apraksts |
| void Serialize(  XmlDictionaryWriter writer,  object body); | No objekta izveido *Body* daļas XML.   * writer: rakstītājs, ar kura starpniecību jāveido XML. * body: serializējamais objekts. |
| object Deserialize(  XmlDictionaryReader reader,  Type bodyType); | No *Body* daļas XML izveido objektu, kas atbilstu norādītajam tipam.   * reader: lasītājs, ar kura starpniecību jānolasa XML. * bodyType: deserializējamā objekta datu tips. * Atgriež: deserializēto objektu. |

### LvpWSFederationAuthenticationModule

Autentifikācijas modulis, kas kontrolē lietotāju pieeju e-pakalpojumam atbilstoši e-pakalpojuma konfigurācijai emulatorā vai e-pakalpojumu katalogā. Tiek reģistrēts e-pakalpojuma konfigurācijas datnē un nav paredzēts tiešai (no koda) e-pakalpojumu izstrādātāju izmantošanai.

## .NET bibliotēka Lvp.Eservice.Controls

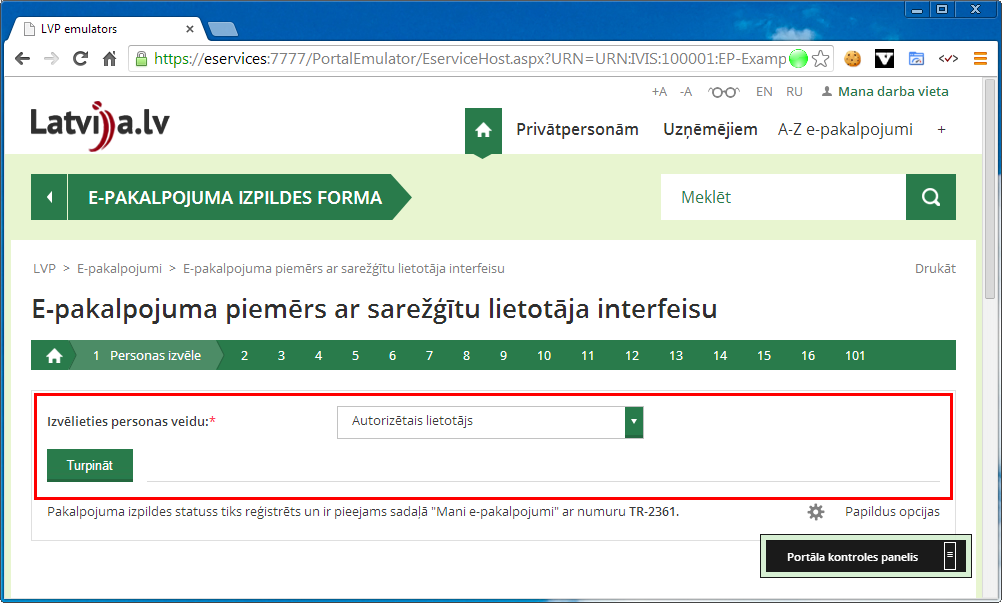
Bibliotēka nodrošina ASP.NET kontroļus un citus elementus e-pakalpojumu lietotāja interfeisa standartizēšanai un izstrādes atvieglošanai.

Visiem šīs bibliotēkas kontroļiem nosaukums sākas ar „Lvp”. Pēc noklusējuma kontroļi ir reģistrēti e-pakalpojuma lietojumā ar birkas prefiksu (tagPrefix) „abc”.

Lielākā daļa kontroļu ir standarta ASP.NET kontroļu atvasinājumi, kas pielāgoti LVP dizainam un lietojumam. Pirms lietot kādu ASP.NET kontroli var pameklēt, vai šī bibliotēka nepiedāvā pielāgotu tā versiju (ar LVP prefiksu).

Piemēram, RequiredFieldValidator vietā ieteicams lietot *LvpRequiredFieldValidato*r. Savukārt, *LvpTextBox* kontrolis neeksistē, tādēļ jālieto *TextBox*.

### LvpWizard



70.attēls. LvpWizard konteiners

Kontroļa uzdevums ir nodrošināt e-pakalpojuma soļu maiņu un citiem kontroļiem nepieciešamo infrastruktūru (piemēram, LvpWizardMenu un LvpWizardStepBase). Varētu teikt, ka tas ir e-pakalpojuma lietotāja interfeisa mugurkauls.

*LvpWizard* bāzes klase ir ASP.NET Wizard.

16.tabula

LvpWizard īpašības un metodes

| Īpašība vai metode | Apraksts |
| --- | --- |
| string MessageBoxId {get;set;} | E-pakalpojuma noklusētā paziņojumu kontroļa (LvpMessageBox) identifikators. |
| IMessageBox MessageBox{ get; } | Atribūts noklusētā paziņojumu kontroļa izgūšanai. Ja ir norādīts MessageBoxId atribūts, meklēšana notiek pēc tā. Pretējā gadījumā lapā tiek meklēts pirmais kontrolis ar tipu IMessageBox. |
| Button StartNextButton { get; } | Ļauj piekļūt e-pakalpojuma pirmā (*Start*) soļa *Next* pogai.  Var būt problēmas ar navigācijas pogu Visible atribūta vērtību uzstādīšanu, jo pogu redzamību uzstāda arī ASP.NET. Tādēļ, piemēram, soļa atvēršanas (OnOpen) notikumā uzstādīta Visible vērtība tiek pārrakstīta. |
| Button StepNextButton { get; } | Ļauj piekļūt e-pakalpojuma vidējo (ne pirmā un pēdējā) soļu *Next* pogai. |
| Button StepPreviousButton  { get; } | Ļauj piekļūt e-pakalpojuma vidējo (ne pirmā un pēdējā) soļu *Previous* pogai. |
| Button FinishCompleteButton  { get; } | Ļauj piekļūt e-pakalpojuma pēdējā soļa *Next* pogai. |
| Button FinishPreviousButton  { get; } | Ļauj piekļūt e-pakalpojuma pēdējā soļa *Previous* pogai. |
| void ClickNext(  object sender,  EventArgs e) | Metode, kura jāizsauc uz pogas „Tālāk” nospiešanu gadījumos, ja tiek lietots nestandarta navigācijas šablons (StepNavigationTemplate). |
| void ClickPrevious(  object sender,  EventArgs e) | Metode, kura jāizsauc uz pogas „Atpakaļ” nospiešanu gadījumos, ja tiek lietots nestandarta navigācijas šablons (StepNavigationTemplate). |
| void ClickFinish(  object sender,  EventArgs e) | Metode, kura jāizsauc uz pogas „Beigt” nospiešanu gadījumos, ja tiek lietots nestandarta navigācijas šablons (StepNavigationTemplate). |
| void ChangeContainerVisibility(  object sender,  EventArgs e) | Metode, kura jāizsauc pirms lapas renderēšanas (OnPreRender) gadījumos, ja tiek lietots nestandarta navigācijas šablons (StepNavigationTemplate). |
| List<StepMenuInfo> GetStepsMenuInfo() | Izgūst soļu izvēlnei (LvpWizardMenu) nepieciešamo informāciju par vedņa soļiem.   * Atgriež: informāciju par attēlojamajiem soļiem. |
| void OpenFirstVisibleStep() | Atver pirmo soli, ja tāds eksistē. |
| T FindStep<T>() | Atrod soļa kontroli pēc tā datu tipa.   * *T*: meklētā kontroļa tips. * Atgriež: atrasto kontroli vai null, ja kontrolis nav atrasts. |
| void OpenPage(  string absoluteUrl,  string infoText = null) | Veic pāreju uz citu (ārēju) lapu. Pārvirzīta tiek visa lapa (nevis tikai *iframe* saturs).   * absoluteUrl: atveramās lapas adrese. * infoText: teksts, ko attēlot *iframe*, kamēr notiek pāreja. |
| void OpenPageWithPost(  string absoluteUrl,  Dictionary<string, string>  postParams = null,  string infoText = null) | Veic pāreju uz citu (ārēju) lapu ar *HTTP POST* pieprasījumu. Pārvirzīta tiek visa lapa (nevis tikai *iframe* saturs).   * absoluteUrl: atveramās lapas adrese. * postParams: ar POST pieprasījumu nododamie dati. * infoText: teksts, ko attēlot *iframe*, kamēr notiek pāreja. |

### LvpWizardStepContainer

Šī kontroļa uzdevums ir nodrošināt iespēju katru e-pakalpojuma soli veidot kā atsevišķu lietotāja kontroli (\*.ascx datne). Tas tiek paveikts, dodot iespēju caur Src atribūtu norādīt ceļu uz lietotāja kontroli, kas tiks ielādēts izpildes gaitā.

Tādējādi katrs lietotāja kontrolis nav jāreģistrē lapas sākumā (<%@ Register).

*LvpWizardStepContainer* bāzes klase ir *ASP.NET WizardStepBase*.

17.tabula

LvpWizardStepContainer īpašības

|  |  |
| --- | --- |
| Īpašība | Apraksts |
| string Src { get; set; } | Ceļš uz ielādējamo lietotāja kontroli (analoģiski, kā būtu, norādot ar <%@ Register birku). |

### LvpWizardStepBase

Bāzes klase visiem e-pakalpojuma soļiem. Nodrošina, dažādu biežāk lietojamo soļu funkcionalitāti, notikumus (*event*), kā arī integrāciju ar citiem kontroļiem (piemēram, soļu izvēlne).

*LvpWizardStepBase* bāzes klase ir *ASP.NET UserControl*.

Soļa notikumu izmantošanas karkass:

public partial class Step1Example : LvpWizardStepBase

{

protected override void OnFirstOpen()

{

// Inicializācija.

// \*.aspx lapas analoģija būtu

// OnLoad notikums ar Page.IsPostBack == false.

// (Šis notikums tiek izsaukts pirms OnOpen).

}

protected override void OnOpen()

{

// Darbības, kas jāveic katru reizi atverot soli.

// \*.aspx lapas analoģija būtu

// OnLoad notikums.

}

public override void OnNextStepClick()

{

if (!Page.IsValid)

{

return;

}

// Saglabāšanas darbības, kas jāveic pirms pārejas uz nākošo soli.

base.OpenNextStep();

}

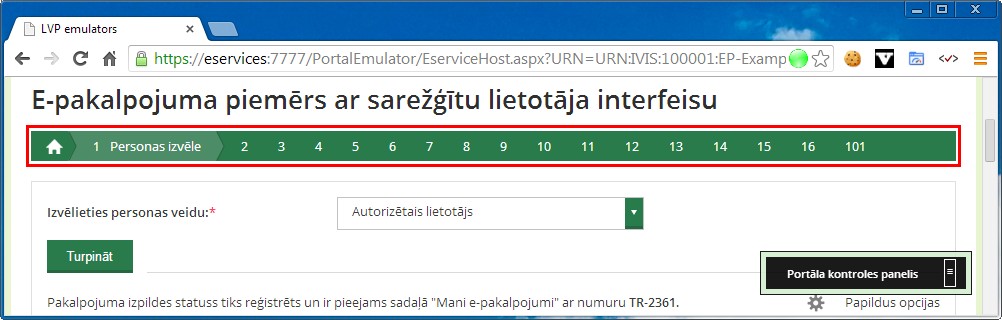
}

18.tabula

LvpWizardStepBase īpašības un metodes

| Īpašība vai metode | Apraksts |
| --- | --- |
| bool HasBeenOpened { get; } | Nosaka, vai solis jau ir bijis atvērts. |
| IWizardControl Wizard { get; } | Nodrošina piekļuvi vednim, kurā solis ir iekļauts. |
| string Title { get; } | Soļa nosaukums, kas tiek attēlots izvēlnē. Pēc noklusējuma tiek ņemta konteinera *Title* atribūta vērtība. Pārlabojot (*override*) iespējams norādīt citu vērtību. |
| string MenuTooltip { get; } | Soļa ekrāna padoms, kas tiek attēlots izvēlnē. Pēc noklusējuma tiek izmantota Title vērtība. Pārlabojot (*override*) iespējams norādīt citu vērtību. |
| int? MenuNumber { get; } | Soļa numurs, kas tiek attēlots izvēlnē. Pēc noklusējuma tiek ņemts soļa kārtas numurs. Pārlabojot (*override*) iespējams norādīt citu vērtību. |
| StepMenuDisplay MenuDisplay  {get;} | Nosaka soļa pieejamību izvēlnē (vai var atvērt). Pēc noklusējuma tiek lietota vērtība StepMenuDisplay.EnabledIfPrevious. Pārlabojot (*override*) iespējams norādīt citu vērtību.  StepMenuDisplay vērtības:   * Hidden: solis izvēlnē netiek rādīts. * Enabled: soli iespējams atvērt no izvēlnes. * EnabledIfPrevious: soli, iespējams, atvērt no izvēlnes, ja tas atrodas pirms pašlaik atvērtā soļa. * Disabled: soli nav iespējams atvērt no izvēlnes. |
| bool HideNavigationButtons{get;} | Vai vednī slēpt navigācijas pogas. Pēc noklusējuma false, bet pārlabojot iespējams norādīt citu vērtību. |
| void OnFirstOpen(); | Notikums, kas tiek izsaukts, pirmo reizi atverot soli. \*.aspx lapas analoģija būtu OnLoad notikums ar Page.IsPostBack == false**.** |
| void OnOpen(); | Notikums, kas tiek izsaukts, katru reizi atverot soli. \*.aspx lapas analoģija būtu OnLoad notikums. |
| void OnNextStepClick(); | Notikums, kas tiek izsaukts, aktīvajam solim, kad tiek nospiesta vedņa poga „Tālāk”. |
| void OnPreviousStepClick(); | Notikums, kas tiek izsaukts, aktīvajam solim, kad tiek nospiesta vedņa poga „Atpakaļ”. |
| void OnFinishClick(); | Notikums, kas tiek izsaukts, aktīvajam solim, kad tiek nospiesta vedņa poga „Beigt” (pēdējā solī). |
| void OpenNextStep(  int offset = 1); | Atver nākošo soli.   * offset: nosaka, cik soļus uz priekšu atvērt (pēc noklusējuma 1 – atvērt nākošo soli). |
| void OpenPreviousStep(  int offset = 1); | Atver iepriekšējo soli.   * offset: nosaka, cik soļus uz atpakaļu „atkāpties” (pēc noklusējuma 1). |
| void OpenStep(int stepIndex); | Atver soli ar norādīto kārtas numuru.   * stepIndex: atveramā soļa indekss. |
| void OnStepOpened(); | Iekšēja metode, kas tiek izmantota, lai solim paziņotu, ka tas ir atvērts. |
| void ShowError(  string messageFormat,  params object[]  messageArguments); | Parāda kļūdas paziņojumu vedņa noklusētajā ziņojumu kontrolī.   * messageFormat: paziņojums ar vietturiem argumentu ievietošanai (analoģiski string.Format funkcijai). * messageArguments: argumenti, kuru vērtības ievieto paziņojumā (analoģiski string.Format funkcijai). |
| void ShowInfo(  string messageFormat,  params object[]  messageArguments); | Parāda informatīvu paziņojumu vedņa noklusētajā ziņojumu kontrolī.   * messageFormat: paziņojums ar vietturiem argumentu ievietošanai (analoģiski string.Format funkcijai). * messageArguments: argumenti, kuru vērtības ievieto paziņojumā (analoģiski string.Format funkcijai). |
| void ShowSuccess(  string messageFormat,  params object[]  messageArguments); | Parāda paziņojumu par veiksmīgu darbību vedņa noklusētajā ziņojumu kontrolī.   * messageFormat: paziņojums ar vietturiem argumentu ievietošanai (analoģiski string.Format funkcijai). * messageArguments: argumenti, kuru vērtības ievieto paziņojumā (analoģiski string.Format funkcijai). |
| void ShowMessage(  MessageType type,  string message,  string title = null,  bool htmlEncode = true); | Parāda paziņojumu vedņa noklusētajā ziņojumu kontrolī.   * type: ziņojuma veids (informācija, kļūda, veiksme). * message: ziņojuma teksts. * Title: ziņojuma virsraksts. * htmlEncode: nosaka, vai ziņojuma tekstu kodēt (aizstāt HTML rezervētos simbolus) vai arī izvadīt nemodificējot. |
| T FindStep<T>(); | Atrod citu soli pēc tā tipa. Paredzēts, lai realizētu datu apmaiņu starp soļiem.   * T: meklētā soļa datu tips. * Atgriež: atrasto kontroli vai null, ja kontrolis nav atrasts. |
| void OpenPage(  string absoluteUrl,  string infoText = null) | Veic pāreju uz citu (ārēju) lapu. Pārvirzīta tiek visa lapa (nevis tikai *iframe* saturs).   * absoluteUrl: atveramās lapas adrese. * infoText: teksts, ko attēlot *iframe*, kamēr notiek pāreja. |
| void OpenPageWithPost(  string absoluteUrl,  Dictionary<string, string>  postParams = null,  string infoText = null) | Veic pāreju uz citu (ārēju) lapu ar *HTTP POST* pieprasījumu. Pārvirzīta tiek visa lapa (nevis tikai *iframe* saturs).   * absoluteUrl: atveramās lapas adrese. * postParams: ar POST pieprasījumu nododamie dati. * infoText: teksts, ko attēlot *iframe*, kamēr notiek pāreja. |

### LvpWizardMenu



71.attēls. LvpWizardMenu

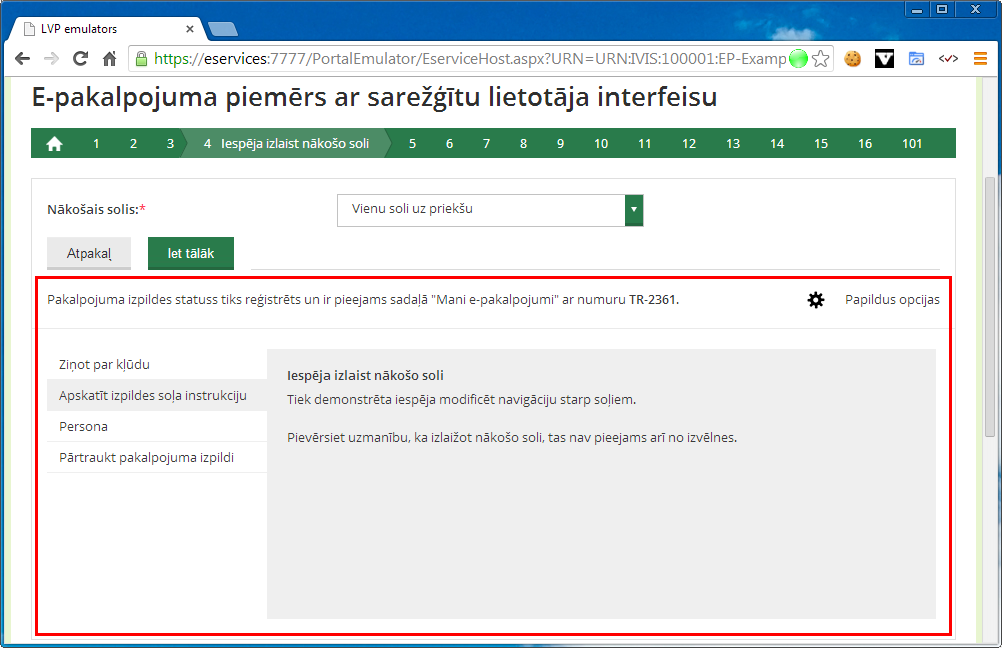
Kontrolis attēlo soļu izvēlni.

19.tabula

LvpWizardMenu īpašības un metodes

| Īpašība vai metode | Apraksts |
| --- | --- |
| string WizardId { get; set; } | Vedņa kontroļa ID. Ja šī vērtība nav norādīta, vednis tiek meklēts pēc datu tipa (IWizardControl). |
| bool HideStepsAfterActive  { get; set; } | Piedāvā iespēju nerādīt soļus, kas ir pēc aktīvā soļa (redzami tikai jau izpildītie un aktīvais solis). Pēc noklusējuma false (vairāk paredzēts veco e-pakalpojumu migrācijai). |
| bool HomeEnabled { get; set; } | Nosaka, vai mājiņas saite kreisajā pusē ir aktīvs solis. Pēc noklusējuma aktīvs. Nospiežot uz minēto saiti, notiek pāriešana uz pirmo soli. |
| event EventHandler OpenHome; | Iespēja pierakstīties uz mājiņas saites atvēršanas notikumu. |
| void LoadStepsManually(  List<StepMenuInfo> stepsInfo) | Piedāvā iespēju soļu informāciju nevis ņemt no saistītā vedņa, bet gan pārlādēt uz katru lapas pārlādi. Paredzēts veco e-pakalpojumu migrācijai.   * stepsInfo: izvēlnē attēlojamais saturs. |

### LvpFooter



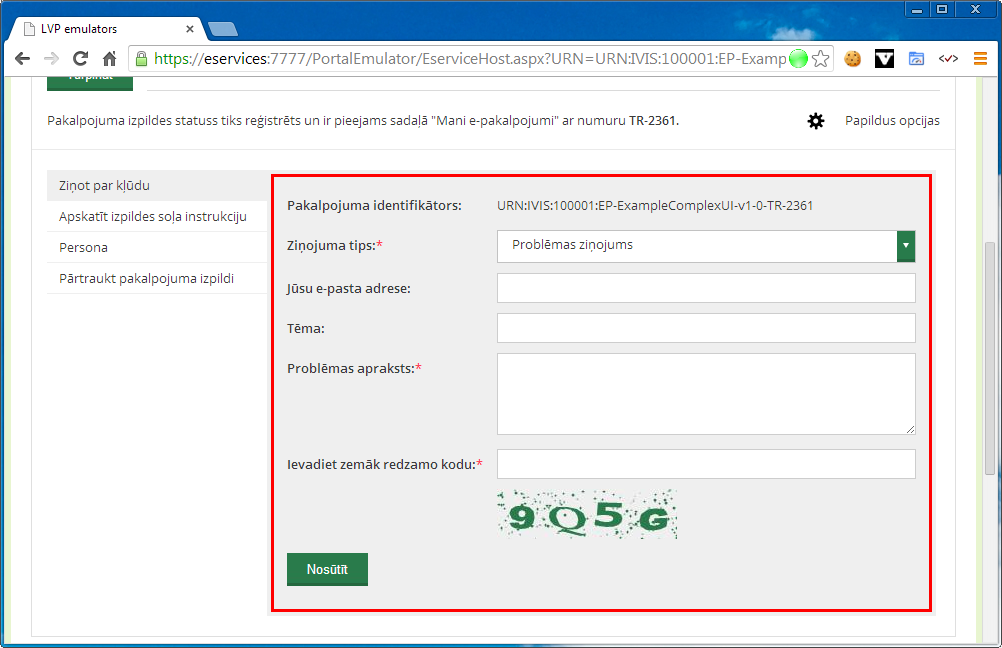
72.attēls. LvpFooter

Kontrolis attēlo lapas apakšējo daļu un ļauj tajā ievietot sadaļas, kas ir pieejamas visos e-pakalpojuma soļos.

*LvpFooter* bāzes klase ir ASP.NET *Wizard*.

Gadījumos, ja solis nav pieejams visu pakalpojuma izpildes laiku, tā redzamību kreisās puses izvēlnē var kontrolēt, kā kontroļa bāzes klasi norādot LvpFooterStepBase un atgriežot atbilstošu vērtību ar pārrakstītu (*override*) VisibleInSidebar īpašību.

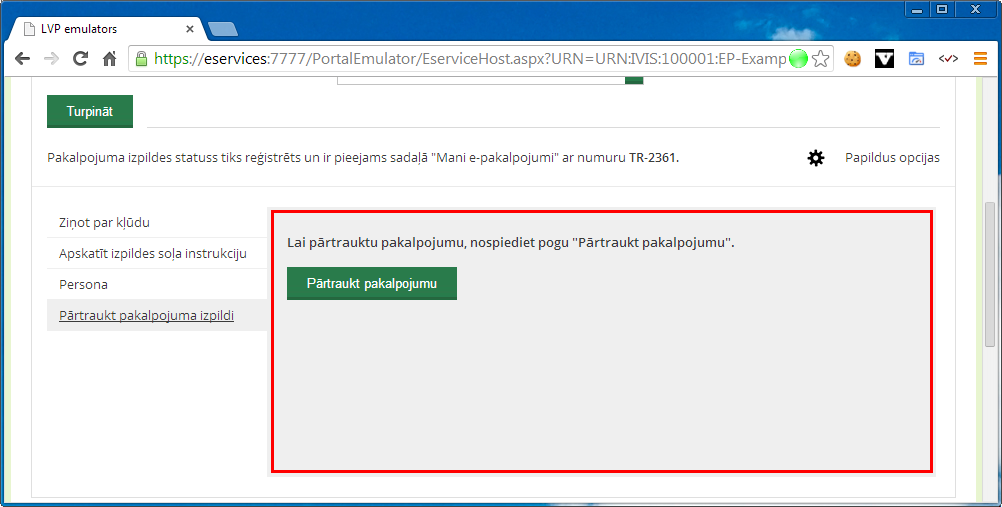
### LvpErrorReport



73.attēls. LvpErrorReport

Kontrolis nodrošina kļūdu iesniegšanas formu. Šī ir viena no obligātajām *LvpFooter* sadaļām. Tiek attēlota tikai, kamēr ir aktīva PS transakcija.

### LvpTerminateEservice



74.attēls. LvpTerminateEservice

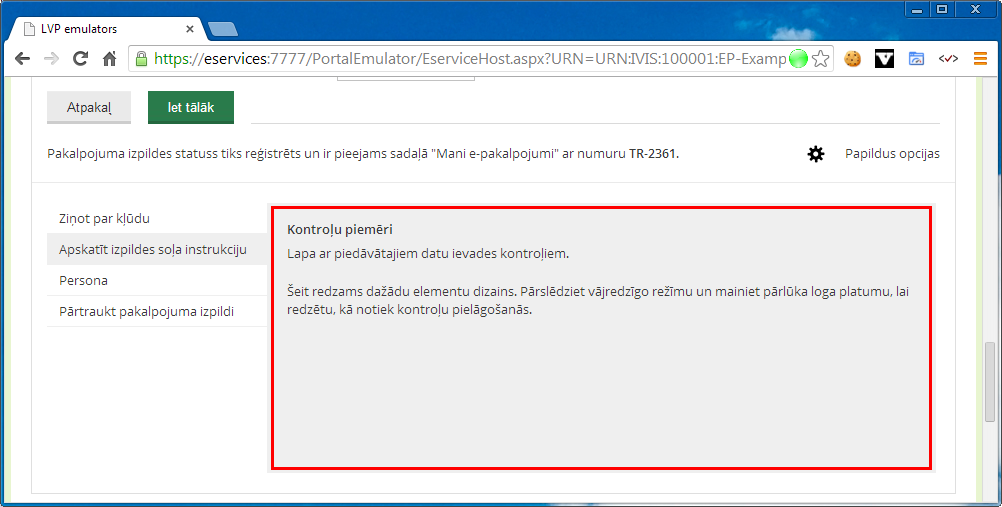
Kontrolis nodrošina iespēju pārtraukt e-pakalpojumu, neaizejot līdz pēdējam solim. Šī ir viena no obligātajām *LvpFooter* sadaļām.

20.tabula

LvpTerminateEservice īpašības

|  |  |
| --- | --- |
| Īpašība | Apraksts |
| event EventHandler TerminateEservice | Piedāvā iespēju pierakstīties uz e-pakalpojuma pārtraukšanas notikumu. Parasti šis notikums jāizmanto, lai pārtrauktu PS transakciju.  ASP.NET sesijas izbeigšana šajā notikumā nav nepieciešama, jo tā tiek veikta automātiski kontroļa Render notikumā. |

### LvpActiveStepManual



75.attēls. LvpActiveStepManual

Kontrolis piedāvā iespēju e-pakalpojuma lejas daļā attēlot solim atbilstošo informāciju no e-pakalpojuma lietotāja instrukcijas.

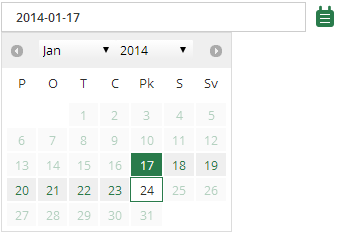
Tiek ņemts aktīvā soļa indekss vednī. No šī indeksa tiek ģenerēts atbilstošā kontroļa identifikators lietotāja instrukcijas kontrolī. Tālāk tiek ņemta lietotāja instrukcija, tajā paslēpts viss, izņemot kontroli ar norādīto ID.

21.tabula

LvpActiveStepManual īpašības

|  |  |
| --- | --- |
| Īpašība | Apraksts |
| string VisibleControlIdPattern  { get; set; } | Nosaka, kāds izskatīsies lietotāja instrukcijā meklētā kontroļa identifikators. Identifikators tiek ģenerēts šajā īpašībā norādītajā vērtībā aizstājot tekstu {{ActiveStepIndex}} ar vedņa aktīvā soļa identifikatoru. |
| string WizardId { get; set; } | Identifikators vednim, kura aktīvā soļa indekss tiek ņemts. Ja vērtība nav norādīta, vednis tiek meklēts pēc kontroļa tipa (IWizardControl). |

### LvpDatePicker



76.attēls. LvpDatePicker

Kontrolis datuma ievadei. Vienmēr jālieto kopā ar *LvpDatePickerValidator*, lai nodrošinātu korektu datuma formātu un intervālu.

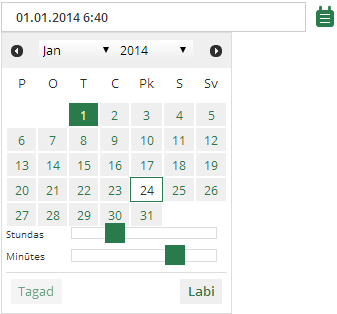
Bāzes klase ir ASP.NET *TextBox.*

22.tabula

LvpDatePicker īpašības

| Īpašība | Apraksts |
| --- | --- |
| DateTime? Date { get; set; } | Ievadītā datuma vērtība. null, ja vērtībai (teksts) nav korekts formāts. |
| bool ValueHasValidFormat {get;} | Vai lauka vērtība (teksts) ir ievadīta korektā formātā. |
| bool ValueIsInRange { get; } | Vai ievadītais datums atbilst norādītajam datuma intervālam. |
| DateTime MinDate { get; set; } | Minimālais ievadāmais datums. Pēc noklusējuma 100 gadi pagātnē. |
| string MinDateString {get; set;} | Minimālais ievadāmais datums kā teksts (izmantojams, norādot vērtību ASP *markup* (\*.aspx, \*.ascx) datnē. |
| int MaxYearsInPast { get; set; } | MinDate uzstāda vērtību, kas aprēķināta, šodienas datumam atņemot norādīto gadu skaitu. |
| int MaxMonthsInPast {get; set;} | MinDate uzstāda vērtību, kas aprēķināta, šodienas datumam atņemot norādīto mēnešu skaitu. Piemēram, lai norādītu 1 gadu un 6 mēnešus, jāuzstāda vērtība 18. |
| int MaxDaysInPast { get; set; } | MinDate uzstāda vērtību, kas aprēķināta, šodienas datumam atņemot norādīto dienu skaitu. |
| DateTime MaxDate { get; set; } | Maksimālais ievadāmais datums. Pēc noklusējuma 50 gadi nākotnē. |
| string MaxDateString {get; set;} | Maksimālais ievadāmais datums kā teksts (izmantojams, norādot vērtību ASP *markup* (\*.aspx, \*.ascx) datnē. |
| int MaxYearsInFuture {get; set;} | MaxDate uzstāda vērtību, kas aprēķināta, šodienas datumam pieskaitot norādīto gadu skaitu. |
| int MaxMonthsInFuture {get;set;} | MaxDate uzstāda vērtību, kas aprēķināta, šodienas datumam pieskaitot norādīto mēnešu skaitu. Piemēram, lai norādītu 1 gadu un 6 mēnešus nākotnē, jāuzstāda vērtība 18. |
| int MaxDaysInFuture {get; set;} | MaxDate uzstāda vērtību, kas aprēķināta, šodienas datumam pieskaitot norādīto dienu skaitu. |
| string ParseFormat {get; set;} | Formāts, pēc kura pārveidot lietotāja ievadīto tekstu par datumu. Pēc noklusējuma d.M.yyyy. |
| string Format {get; set;} | Formāts, kādā izvadīt datumu. Pēc noklusējuma dd.MM.yyyy. |
| string ClientFormat {get; set;} | Datuma formāts kalendāra *JavaScript* bibliotēkām (*jQuery* *UI*). Pēc noklusējuma dd.mm.yy. |

### LvpDateTimePicker



77.attēls. LvpDateTimePicker

Kontrolis datuma un laika ievadei. Vienmēr jālieto kopā ar *LvpDatePickerValidator*, lai nodrošinātu korektu datuma formātu un intervālu.

Bāzes klase ir ASP.NET *TextBox*.

23.tabula

LvpDateTimePicker īpašības

| Īpašība | Apraksts |
| --- | --- |
| DateTime? DateTime { get; set; } | Ievadītā datuma un laika vērtība. null, ja vērtībai (teksts) nav korekts formāts. |
| bool ValueHasValidFormat {get;} | Vai lauka vērtība (teksts) ir ievadīta korektā formātā. |
| bool ValueIsInRange { get; } | Vai ievadītais datums atbilst norādītajam datuma intervālam. |
| DateTime MinDate { get; set; } | Minimālais ievadāmais datums. Netiek ņemts vērā laiks – nozīme ir tikai datumam. Pēc noklusējuma 100 gadi pagātnē. |
| string MinDateString {get; set;} | Minimālais ievadāmais datums kā teksts (izmantojams, norādot vērtību ASP *markup* (\*.aspx, \*.ascx) datnē. |
| int MaxYearsInPast { get; set; } | MinDate uzstāda vērtību, kas aprēķināta, šodienas datumam atņemot norādīto gadu skaitu. |
| int MaxMonthsInPast {get; set;} | MinDate uzstāda vērtību, kas aprēķināta, šodienas datumam atņemot norādīto mēnešu skaitu. Piemēram, lai norādītu 1 gadu un 6 mēnešus, jāuzstāda vērtība 18. |
| int MaxDaysInPast { get; set; } | MinDate uzstāda vērtību, kas aprēķināta, šodienas datumam atņemot norādīto dienu skaitu. |
| DateTime MaxDate { get; set; } | Maksimālais ievadāmais datums. Netiek ņemts vērā laiks – nozīme ir tikai datumam. Pēc noklusējuma 50 gadi nākotnē. |
| string MaxDateString {get; set;} | Maksimālais ievadāmais datums kā teksts (izmantojams, norādot vērtību ASP *markup* (\*.aspx, \*.ascx) datnē. |
| int MaxYearsInFuture {get; set;} | MaxDate uzstāda vērtību, kas aprēķināta, šodienas datumam pieskaitot norādīto gadu skaitu. |
| int MaxMonthsInFuture {get;set;} | MaxDate uzstāda vērtību, kas aprēķināta, šodienas datumam pieskaitot norādīto mēnešu skaitu. Piemēram, lai norādītu 1 gadu un 6 mēnešus nākotnē, jāuzstāda vērtība 18. |
| int MaxDaysInFuture {get; set;} | MaxDate uzstāda vērtību, kas aprēķināta, šodienas datumam pieskaitot norādīto dienu skaitu. |
| string ParseFormat {get; set;} | Formāts, pēc kura pārveidot lietotāja ievadīto tekstu par datumu. Pēc noklusējuma d.M.yyyy H:m. |
| string Format {get; set;} | Formāts, kādā izvadīt datumu. Pēc noklusējuma dd.MM.yyyy H:mm. |
| string ClientFormat {get; set;} | Datuma formāts kalendāra *JavaScript* bibliotēkām (*jQuery* *UI*). Pēc noklusējuma dd.mm.yy. |
| string ClientTimeFormat  {get; set;} | Laika formāts kalendāra *JavaScript* bibliotēkām. Pēc noklusējuma H:mm. |
| string ClientDateTimeSeparator  {get; set;} | Datuma un laika atdalītāja simbols kalendāra *JavaScript* bibliotēkām. Pēc noklusējuma „ ” (tukšuma simbols). |

### LvpDatePickerValidator

Nodrošina datuma formāta un intervāla validāciju *LvpDatePicker* un *LvpDateTimePicker* kontroļiem.

Bāzes klase ir ASP.NET *CustomValidator*.

### Validatori



78.attēls. LVP validatora darbības piemērs

Datu validācijai ir jāizmanto LVP validatori. Tie ir no ASP.NET validatoriem atvasināti kontroļi, kuru attēlojums ir pielāgots portālam (uzstādītas atbilstošas noklusējuma vērtības un pievienota papildus funkcionalitāte, piemēram, saistītā lauka teksta iekrāsošana).

24.tabula

LVP validatoru īpašības

| Īpašība | Apraksts |
| --- | --- |
| string IsThisSpecialTypeValidator  { get; set; } | Vai darbināt saistītā lauka iekrāsošanu. Pēc noklusējuma true – darbināt. |
| string RegisterOwnQueryScript  { get; set; } | Vai kontrolim lapā iekļaut *jQuery* bibliotēku. Pēc noklusējuma false, jo *jQuery* jau ir iekļauts CDN skriptos. |
| string TargetControlErrorClass  { get; set; } | CSS klase, kura jāpiešķir saistītajam laukam, ja tas ir nekorekts. |
| string TooltipControlId  { get; set; } | ID no *LvpTooltip* kontroļa, kura attēlojums jāmaina, ja lauka vērtība ir nekorekta. |

### LvpPersonCodeValidator

Nodrošina iedzīvotāja personas koda validāciju.

Bāzes klase ir *ASP.NET CustomValidator*.

### LvpDisplayState

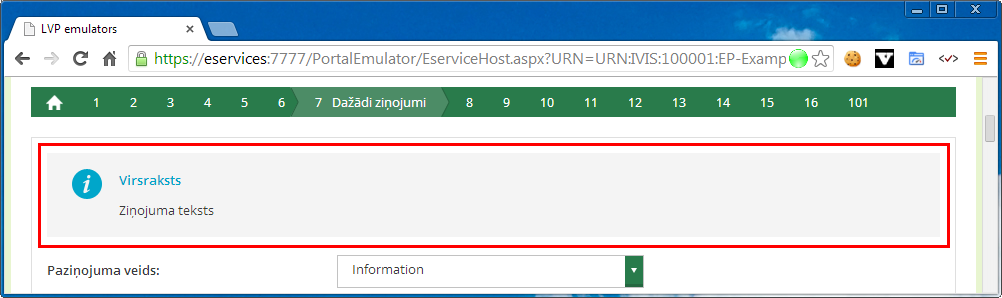
Neredzams kontrolis, kas nodrošina piekļuvi LVP tekošā attēlojuma informācijai (vājredzīgo režīms, teksta palielinājums, ekrāna platums), kā arī piedāvā ASP.NET iespēju pierakstīties uz LVP attēlojuma maiņas notikumiem.

25.tabula

LvpDisplayState īpašības un notikumu apstrādātāji

| Īpašība | Apraksts |
| --- | --- |
| BlindMode BlindMode { get; set; } | Aktīvais vājredzīgo režīms. |
| bool AutoPostBackOnBlindModeChanged  { get; set; } | Vai veikt pieprasījumu uz serveri (post-back) pie vājredzīgo režīma maiņas. Pēc noklusējuma true. |
| EventHandler BlindModeChanged | Notikums, kas tiek izraisīts pie vājredzīgo režīma maiņas. |
| int TextZoom { get; set; } | Aktīvais teksta palielinājums. |
| bool AutoPostBackOnTextZoomChanged  { get; set; } | Vai veikt pieprasījumu serverim (post-back) pie teksta palielinājuma maiņas. Pēc noklusējuma false. |
| EventHandler TextZoomChanged | Notikums, kas tiek izraisīts pie teksta izmēra maiņas. |
| Resolution? Resolution { get; } | Aktīvais platums. Vērtība pieejama tikai pēc lapas atkārtotas ielādes (*post-back)*. |
| string OnClientResolutionChanged  { get; set; } | *JavaScript* funkcija, kuru izsaukt pie loga platuma maiņas. Funkcijai ir pieejams parametrs size. |
| bool AutoPostBackOnResolutionChanged  { get; set; } | Vai veikt pieprasījumu serverim (*post-back*) pie loga platuma maiņas. Pēc noklusējuma false. |
| EventHandler ResolutionChanged | Notikums, kas tiek izraisīts pie loga platuma maiņas. |
| bool AutoPostBackBeforeLanguageChange  { get; set; } | Vai veikt pieprasījumu serverim (*post-back*) pirms portāla valodas maiņas (e-pakalpojuma *iframe* pārlādes).  Pēc noklusējuma true. |
| EventHandler BeforeLanguageChange | Notikums, kas tiek izraisīts pirms valodas maiņas un e-pakalpojuma pārlādes. |

### LvpMessageBox



79.attēls. LvpMessageBox

Kontrolis piedāvā iespēju izvadīt dažādus ziņojumus.

26.tabula

LvpMessageBox īpašības un metodes

|  |  |
| --- | --- |
| Īpašība vai metode | Apraksts |
| bool AutoHide { get; set; } | Vai ziņojums ir automātiski jāpaslēpj pēc lapas pārlādes. Pēc noklusējuma true. |
| string CssClass { get; set; } | Iespēja ziņojumam norādīt papildu CSS klasi. |
| string Text { get; set; } | Paziņojuma teksts. |
| string Title { get; set; } | Paziņojuma virsraksts. |
| void ShowMessage(  MessageType messageType,  string message,  string title,  bool htmlEncode); | Ziņojuma parādīšana.   * messageType: paziņojuma tips (informācija, kļūda, veiksmīga darbība). * message: paziņojuma teksts. * title: paziņojuma virsraksts. * htmlEncode: vai kodēt paziņojuma tekstu (aizstāt HTML rezervētos simbolus). |
| void HideMessage(); | Ziņojuma paslēpšana. |

### LvpTooltip



80.attēls. LvpTooltip

Kontrolis piedāvā iespēju laukiem pievienot paskaidrojošu informāciju.

27.tabula

LvpTooltip īpašības un metodes

| Īpašība vai metode | Apraksts |
| --- | --- |
| string Text { get; set; } | Paskaidrojošās informācijas teksts. |
| string CssClass { get; set; } | Iespēja ziņojumam norādīt CSS klasi. |
| string RegisterOwnQueryScript  { get; set; } | Vai kontrolim iekļaut *JQuery* bibliotēku. Pēc noklusējuma false. |

### LvpLink

Kontrolis CSS failu iekļaušanai dokumentā (ģenerē HTML <link> elementus). Katra jauna rindiņa no Href atribūta tiek pārveidota par jaunu <link> elementu (ja rindiņa nav tukša). Aizstāj Href atribūtā norādītās vērtības vietturus ar atbilstošu konfigurācijas vai attēlojuma režīma vērtībām.

Viettura formāts:

**{**papildu teksts**{**viettura nosaukums**}**papildu teksts**}**

Papildu teksts tiek ievietots tikai gadījumos, ja viettura vērtība nav tukša. Viettura nosaukums nav reģistra jūtīgs (*case-insensitive*).

28.tabula

Vietturu nosaukumi

|  |  |
| --- | --- |
| Vietturis | Apraksts |
| CDN | Tiek aizstāts ar CDN adresi no konfigurācijas datnes. |
| BlindMode | Tiek aizstāts ar aktīvo vājredzīgo režīmu. BlindMode.None tiek uztverts kā tukšums (””). |
| Resolution | Tiek aizstāts ar aktīvo e-pakalpojuma loga platumu. |

Piemērs ar gaišo vājredzīgo režīmu:

* Href: {{cdn}}/eservice/css/lvp-eservice{-{blindMode}}-full-jquery-ui.css
* CDN: http://www.latvija.lv/cdn/v1.0
* BlindMode: BlindMode.Bright
* Rezultāts:

<link href="**http://www.latvija.lv/cdn/v1.0**/eservice/css/lvp-eservice**-bright**-full-jquery-ui.css" rel="stylesheet" type="text/css">

Piemērs bez vājredzīgo režīmu:

* Href: {{cdn}}/eservice/css/lvp-eservice{-{blindMode}}-full-jquery-ui.css
* CDN: http://www.latvija.lv/cdn/v1.0
* BlindMode: BlindMode.None
* Rezultāts:

<link href="**http://www.latvija.lv/cdn/v1.0**/eservice/css/lvp-eservice-full-jquery-ui.css" rel="stylesheet" type="text/css">

Piemērs ar tumšo vājredzīgo režīmu un vairākām rindiņām (rezultātā redzamais *virtual-directory* ir e-pakalpojuma virtuālās direktorijas apzīmējums):

* Href:

~/css/style.css

{~/css/style-{blindMode}.css}

* CDN: http://www.latvija.lv/cdn/v1.0
* BlindMode: BlindMode.Dark
* Rezultāts:

<link href="/virtual-directory/css/style.css" rel="stylesheet"

type="text/css"/>

<link href="**/virtual-directory/css/style-dark.css"** rel="stylesheet"

type="text/css" />

Piemērs bez vājredzīgo režīma ar vairākām rindiņām:

* Href:

~/css/style.css

{~/css/style-{blindMode}.css}

* CDN: http://www.latvija.lv/cdn/v1.0
* BlindMode: BlindMode.None
* Rezultāts:

<link href="/virtual-directory/css/style.css" rel="stylesheet"

type="text/css"/>

*LvpLink* bāzes klase ir ASP.NET *HtmlLink*.

29.tabula

LvpLink īpašības

|  |  |
| --- | --- |
| Īpašība | Apraksts |
| string DisplayStateId  { get; set; } | LvpDisplayState kontroļa ID. Ja nav norādīts, minētais kontrolis tiek meklēts pēc tipa (IDisplayState). |
| string Href { get; set; } | link elementu saites formāts ar vietturiem. Pēc noklusējuma aizpildīts ar formātu noklusētajiem CDN stiliem. |

### LvpResolutionLink

Paredzēts CSS stilu iekļaušanai lapā atkarībā no e-pakalpojuma platuma. URL būvēšanai tiek izmantots tas pats vietturu mehānisms, kas aprakstīts 7.2.17.sadaļā.

Kontrolis ģenerē saites visiem platuma veidiem. Ar CDN piegādātais JS skripts šīs vērtības apstrādā un dokumentam pievieno tikai tos CSS, kas atbilst tekošajam platumam. Lietotājam mainot pārlūka loga izmērus, dinamiski tiek mainīti iekļautie CSS faili.

30.tabula

LvpResolutionLink īpašības

|  |  |
| --- | --- |
| Īpašība | Apraksts |
| string DisplayStateId  { get; set; } | LvpDisplayState kontroļa ID. Ja nav norādīts, minētais kontrolis tiek meklēts pēc tipa (IDisplayState). |
| string Href { get; set; } | CSS datņu saites formāts ar vietturiem. |

### LvpScript

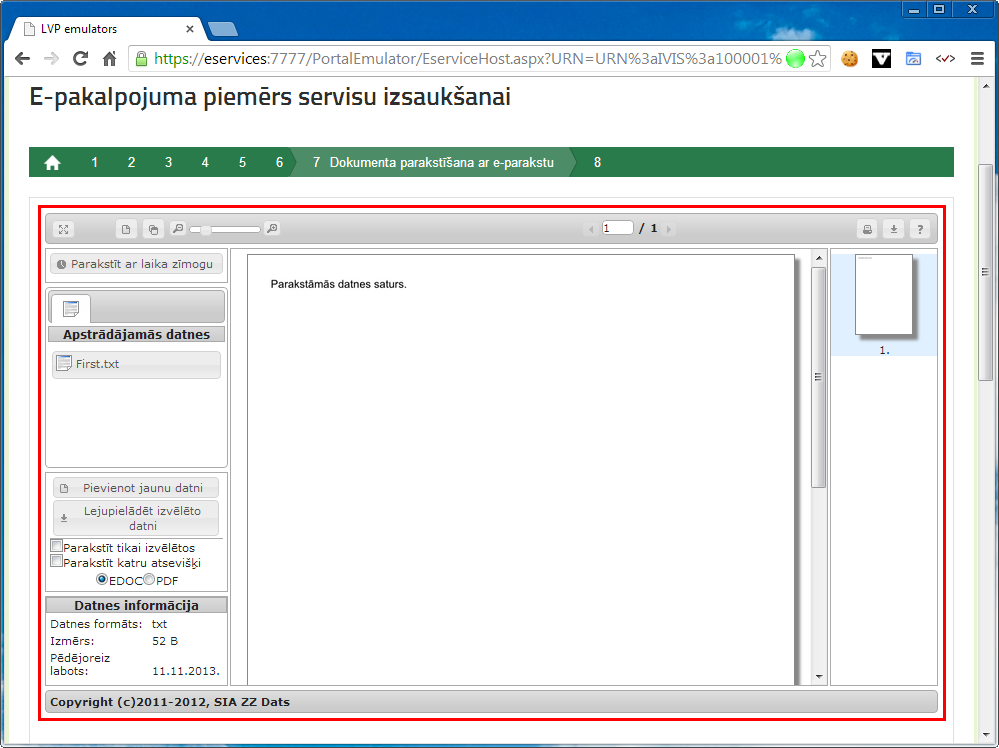
Paredzēts JS skriptu iekļaušanai lapā atkarībā no konfigurācijas parametriem. URL būvēšanai tiek izmantots tas pats vietturu mehānisms, kas aprakstīts 7.2.17.sadaļā.

31.tabula

LvpScript īpašības

|  |  |
| --- | --- |
| Īpašība | Apraksts |
| string DisplayStateId  { get; set; } | LvpDisplayState kontroļa ID. Ja nav norādīts, minētais kontrolis tiek meklēts pēc tipa (IDisplayState). |
| string Src { get; set; } | script elementu saites formāts ar vietturiem. Pēc noklusējuma aizpildīts ar formātu noklusētajiem CDN skriptiem. |

### LvpEsigner



81.attēls. LvpEsigner

*LvpEsigner* kontrolis ir ASP.NET konteiners (wrapper) *DocumentViewer* JavaScript/Java kontrolim (skat. [2] atsauci).

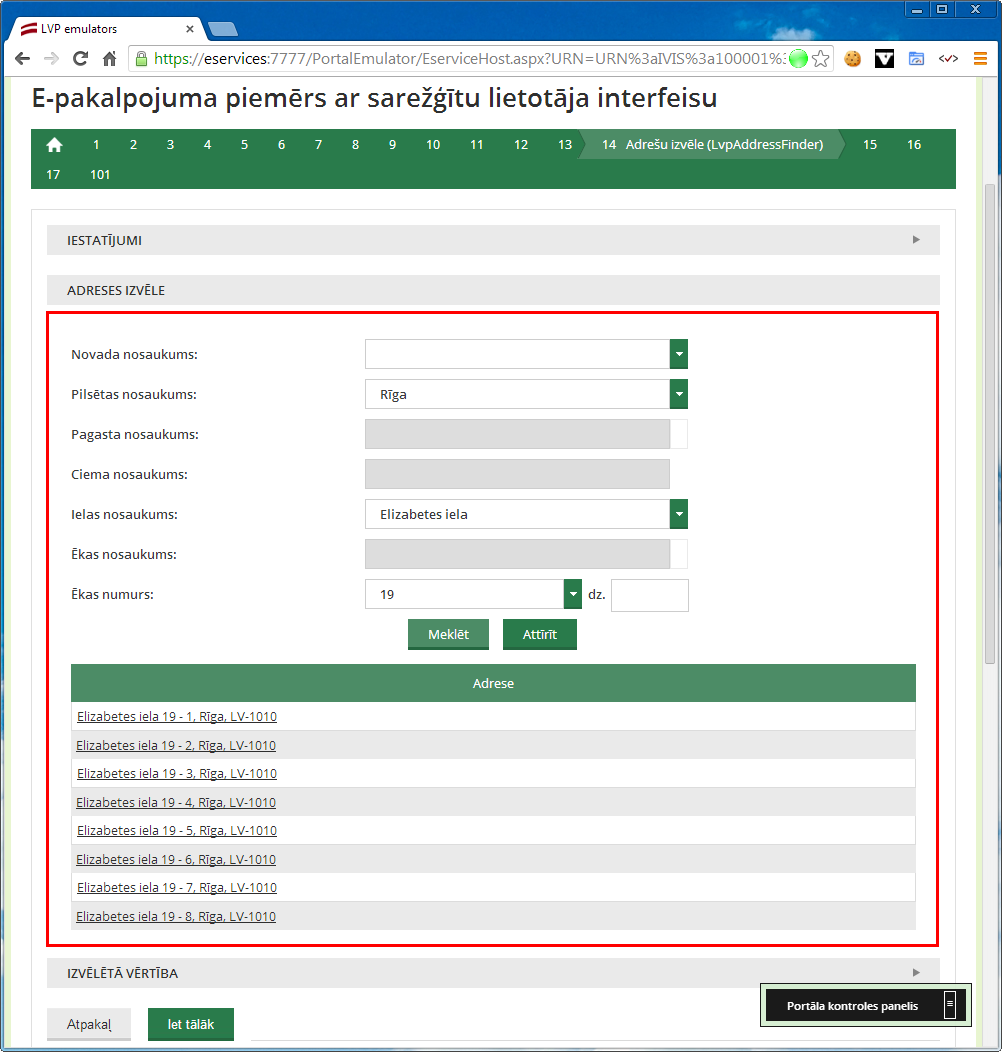
Kontroļa izmantošanai konfigurācijā nepieciešams aizpildīt konfigurācijas sekciju, kas aprakstīta 8.2.nodaļā.

32.tabula

LvpEsigner īpašības un metodes

| Īpašība vai metode | Apraksts |
| --- | --- |
| string SessionId { get; } | Kontrolī ielādētās sesijas identifikators. |
| bool HasSession{ get; } | Pazīme, vai kontrolī ir ielādēta sesija. |
| long? ThumbnailsPaneWidth  { get; set; } | DocumentViewer komponentes ThumbnailsPaneWidth vērtība (skat. [2] atsauci). |
| long? LeftPaneWidth { get; set; } | DocumentViewer komponentes LeftPaneWidth vērtība (skat. [2] atsauci). |
| long? Height { get; set; } | Kontroļa augstums. Ja norādītā vērtība būs mazāka par 500px, tā netiks ņemta vērā (kontroļa minimālais izmērs ir 500px x 500px). |
| long? HeaderLinkInjectionControlId  { get; set; } | Identifikators Literal kontrolim lapas *Head* daļā, kurā tiks ielādēta DocumentViewer *JavaScript* saite. Ja vērtība nav norādīta, saite tiks pievienota *Head* daļas beigās. Vērtība obligāti jānorāda, ja *Head* daļā tiek izmantotas <%= … %> konstrukcijas. |
| string OnClientUiLoaded  { get; set; } | Piedāvā iespēju izsaukt *JavaScript* funkciju pēc tam, kad ir ielādējies e-parakstīšanas kontrolis. |
| void Open(string sessionId) | Atver kontroli un ielādē sesijas identifikatoru.   * sessionId: sesijas identifikators. |
| void Close() | Slēpj kontroli un notīra sesijas identifikatoru. |
| event EventHandler SignComplete; | Ļauj pierakstīties uz e-parakstīšanas beigšanas notikumu. |

### LvpAddressFinder



82.attēls. LvpAddressFinder

*LvpAddressFinder* kontrolis ir ASP.NET konteiners (wrapper) *DocumentFinder* JavaScript kontrolim (skat. [7] atsauci).

Izmantotās bibliotēkas JavaScript kods atrodams window.Vraa.AddressFinder objektā.

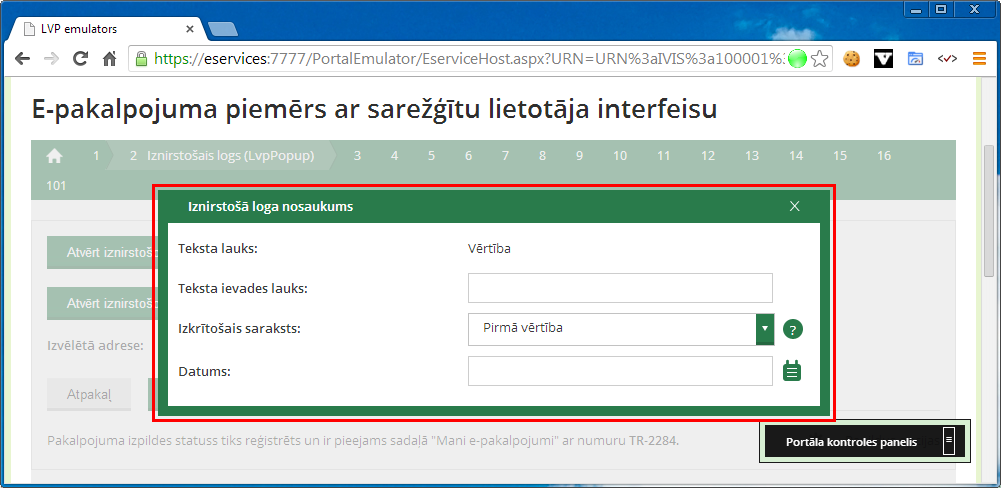
Kontroļa izmantošanai konfigurācijā nepieciešams aizpildīt konfigurācijas sekciju, kas aprakstīta 8.3.nodaļā.

33.tabula

LvpAddressFinder īpašības un metodes

| Īpašība vai metode | Apraksts |
| --- | --- |
| int? MaxTableRows { get; set; } | Atlasāmais maksimālais rindu skaits tabulā, 0 – atslēgt, noklusētā vērtība – 100 (null), max - 10000. |
| bool EnableCodeFind { get; set; } | Vai rādīt meklēšanas lauku „Klasifikatora kods”, noklusētā vērtība ir false. |
| bool EnableCodeCol { get; set; } | Vai rādīt rezultātu saraksta tabulā kolonu „Klasifikatora kods”, noklusētā vērtība ir false. |
| bool EnableAddrTypeCol {get; set;} | Vai rādīt rezultātu saraksta tabulā kolonnu „Adreses veids”, noklusētā vērtība ir false. |
| SearchType? SearchType {get; set;} | Adrešu meklēšanas veids. Noklusētā vērtība Default (null). |
| int? RowsInPage { get; set; } | Attēlojamais rindu skaits vienā lapā, 0 – atslēgts, noklusētā vērtība 10 (null), max - 100. |
| TableSelectType TableSelectType  { get; set; } | Adrešu izvēles veids tabulā. Noklusētā vērtība Link. |
| long? AutoFillId { get; set; } | Identifikators, pēc kura aizpildīt adrešu komponentes laukus pie inicializācijas.  Vērtība var būt vietas kods, mājas kods vai dzīvokļa kods.  Noklusētā vērtība ir null. |
| AutoFillBlock AutoFillBlock  { get; set; } | Vai bloķēt laukus pēc automātiskās lauku aizpildes. Noklusētā vērtība None. |
| bool EnableStreetButton  { get; set; } | Rādīt pogu „visi” pie ielu meklēšanas lauka, noklusētā vērtība true. |
| bool EnableBuildingButton  { get; set; } | Rādīt pogu „visi” pie ēku nosaukuma meklēšanas lauka, noklusētā vērtība true. |
| bool AutoFillParentChild  { get; set; } | Vai aizpildīt pēc lauku izvēles augšēji pakārtotos laukus sākot no ielu meklēšanas un vai nobloķēt apakšējos laukus, kuriem nav iespējamas vērtības.  Noklusētā vērtība – false. |
| string HeaderLinkInjectionControlId  { get; set; } | Identifikators Literal kontrolim lapas *Head* daļā, kurā tiks ielādēta DocumentViewer *JavaScript* saite. Ja vērtība nav norādīta, saite tiks pievienota *Head* daļas beigās. Vērtība obligāti jānorāda, ja *Head* daļā tiek izmantotas <%= … %> konstrukcijas. |
| string OnClientAddressSelected  { get; set; } | Piedāvā iespēju izpildīt *JavaScript* komandas pēc tam, kad lietotājs ir izvēlējies adresi. |
| string OnClientSearched  { get; set; } | Piedāvā iespēju izpildīt *JavaScript* komandas pēc tam, kad lietotājs ir veicis adrešu meklēšanu. |
| string OnClientCleared {get; set;} | Piedāvā iespēju izpildīt *JavaScript* komandas pēc tam, kad lietotājs ir attīrījis adrešu meklēšanas kritērijus. |
| string OnClientError { get; set; } | Piedāvā iespēju izpildīt *JavaScript* komandas pēc tam, kad *JavaScript* kontrolī ir atgadījusies kļūda. |
| Address SelectedAddress{ get; } | Lietotāja izvēlētās adreses izgūšana.  Adreses uzstādīšanai jāizmanto AutoFillId atribūts. |
| event EventHandler  AddressSelected; | Ļauj pierakstīties uz adreses izvēles notikumu. |

### LvpPopup



83.attēls. LvpPopup

*LvpPopup* kontrolis paredzēts satura izvadīšanai iznirstošā logā.

Kontrolis piedāvā servera puses risinājumu. Lai logu rādītu vai slēptu klienta pusē, izstrādātājam loga parādīšana un slēpšana jārealizē ar savām *JavaScript* komandām. Piemēram, $(‘.popup’).hide() un $(‘.popup’).show().

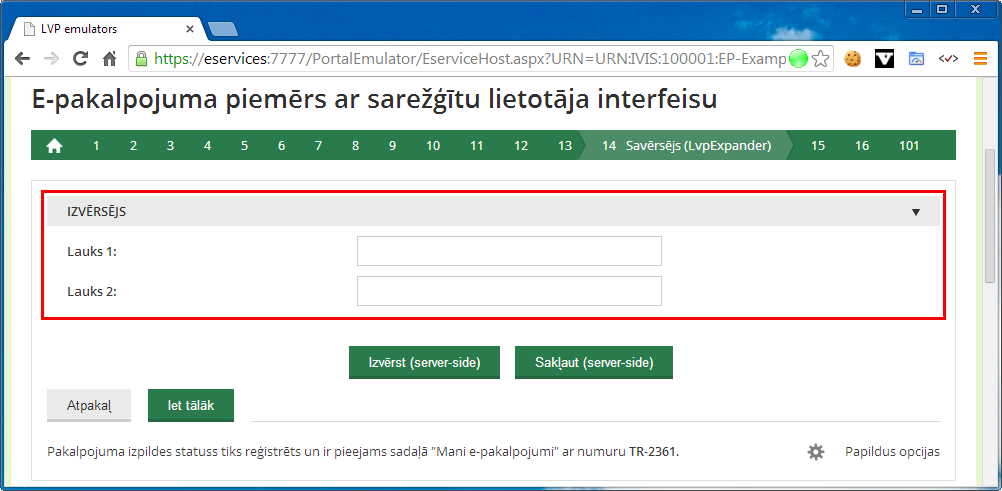
Iznirstošais logs lietotājam nepiedāvā iespēju mainīt loga atrašanās vietu vai izmēru.

34.tabula

LvpExpander īpašības

|  |  |
| --- | --- |
| Īpašība vai metode | Apraksts |
| string Header { get; set; } | Virsraksta teksts. |
| string CssClass { get; set; } | Papildus CSS klašu nosaukumi. |
| string OnClientClosed{ get; set; } | Piedāvā iespēju izpildīt *JavaScript* komandas pēc tam, kad lietotājs ir nospiedis loga aizvēršanas pogu. |
| event EventHandler Closed; | Ļauj pierakstīties uz loga aizvēršanas notikumu. |

### LvpExpander



84.attēls. LvpExpander

*LvpExpander* kontrolis ir savēršams un izvēršamas lapas elementu konteiners ar virsrakstu. Paredzēts vizuālai e-pakalpojuma lapas daļu grupēšanai.

35.tabula

LvpExpander īpašības

|  |  |
| --- | --- |
| Īpašība vai metode | Apraksts |
| string Header { get; set; } | Virsraksta teksts. |
| string CssClass { get; set; } | Papildus CSS klašu nosaukumi. |
| bool Collapsed { get; set; } | Pazīme, vai konteiners ir savērsts vai izvērsts. |

### LvpWizardPageBase

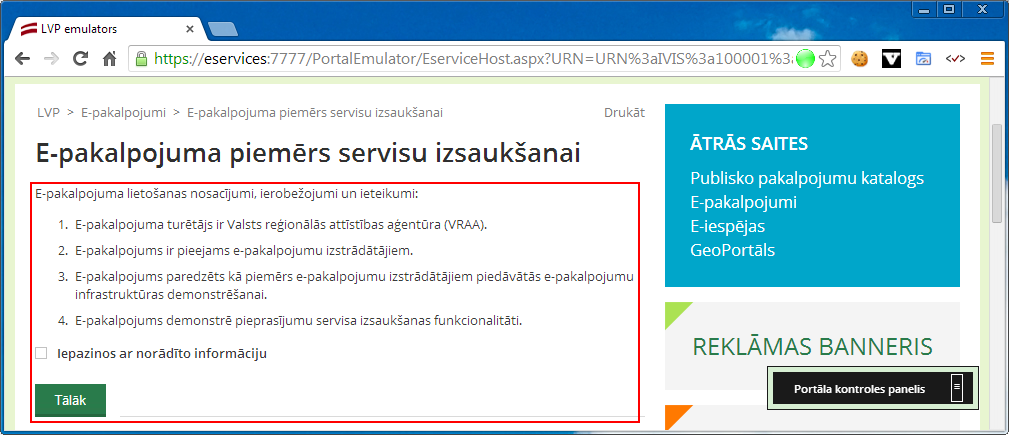
Bāzes klase e-pakalpojuma galvenajai lapai nodrošina, piemēram, apraksta lapas attēlošanu pirms e-pakalpojuma uzsākšanas un abilstošu tekošās kultūras (*CurrentCulture*) informācijas uzstādīšanu.

*LvpWizardPageBase* bāzes klase ir standarta ASP.NET *Page*.

### LvpAboutPage

E-pakalpojuma uzsākšanas apraksta lapas (*About.aspx*) noklusētā koda (*code-behind*) klase. Nodrošina e-pakalpojuma apraksta attēlošanu un lietotāja apstiprinājuma funkcionalitāti.

*LvpAboutPage* bāzes klase ir *LvpAboutPageBase*.



85.attēls. LvpAboutPage

### LvpAboutPageBase

E-pakalpojuma uzsākšanas apraksta lapas (*About.aspx*) bāzes klase. Paredzēts izmantot, ja nepieciešams izveidot pielāgotu sākuma lapu (piemēram, ar vairākām apstiprināšanas iespējām).

*LvpAboutPageBase* bāzes klase ir standarta ASP.NET *Page*.

36.tabula

LvpAboutPageBase īpašības un metodes

|  |  |
| --- | --- |
| Īpašība vai metode | Apraksts |
| static readonly string  DefaultNotAcceptedMessage | Noklusētais kļūdas paziņojuma teksts, kas tiek attēlots, ja lietotājs nav apstiprinājis savu piekrišanu. |
| bool AgreementHasBeenAccepted  { get; set; } | Pazīme, vai lietotājs ir piekritis paziņojuma tekstam. |
| string GetEserviceAboutHtml() | Izgūst e-pakalpojuma konfigurācijā norādīto ziņojuma tekstu. |

### LvpUserManualPageBase

E-pakalpojuma lietotāja instrukcijas lapas (*UserManual.aspx*) bāzes klase.

*LvpUserManualPageBase* bāzes klase ir standarta ASP.NET *Page*.

37.tabula

LvpUserManualPageBase metodes

|  |  |
| --- | --- |
| Īpašība vai metode | Apraksts |
| string GetUserManualHtml() | Izgūst emulatorā vai e-pakalpojumu katalogā ievadīto pakalpojuma lietotāja instrukcijas tekstu (HTML). |

### LvpMasterPageBase

E-pakalpojuma lappušu šablona (*MasterPage.Master*) bāzes klase.

*LvpMasterPageBase* bāzes klase ir standarta ASP.NET *MasterPage*.

### SessionKeepAlive

*SessionKeepAlive* ir pieprasījumu apdarinātājs (*HttpHandler*), kas atbildīgs par e-pakalpojuma sesijas atjaunošanu. No LVP var tikt sūtīti pieprasījumi pagarināt e-pakalpojuma sesiju. Šie pieprasījumi caur AJAX tiek novirzīti uz šo apdarinātāju.

Apdarinātājs ir reģistrēts e-pakalpojuma apakšadresē *Public/Lvp/SessionKeepAlive*. Šis parametrs nav modificējams.

### WrapperPage

*WrapperPage* ir pieprasījumu apdarinātājs (*HttpHandler*), kas atbildīgs par:

1. JS kanāla uzturēšanu starp e-pakalpojumu un LVP;
2. LVP *iframe* garuma maiņu atbilstoši e-pakalpojuma lapas augstumam;
3. Lietotāju autentifikācijas atbalstu.

E-pakalpojuma lapa patiesībā tiek darbināta iekš vēl viena *iframe*, kuru veido minētā klase.



86.attēls. Iframe struktūra e-pakalpojuma un LVP integrācijai

Apdarinātājs ir reģistrēts e-pakalpojuma apakšadresē */Public/Lvp/Wrapper*. Šis parametrs nav modificējams.

### ErrorPage

Pieprasījumu apdarinātājs (*HttpHandler*), kas paziņo portālam, ka e-pakalpojumā ir notikusi globāla (neapstrādāta) kļūda.

Neapstrādātas kļūdas situācijā e-pakalpojumā ar standarta *ASP.NET* konfigurāciju tiek panākta pāreja uz šo apdarinātāju. Apdarinātājs paziņoju portālam, ka e-pakalpojumā ir notikusi kļūda. Portāls, savukārt, veic pāreju uz globālo kļūdu lapu.

Apdarinātājs ir reģistrēts e-pakalpojuma apakšadresē *Public/Lvp/Error*.

### LvpWsFederationRequestValidator

ASP.NET pieprasījumu validators, kas pielāgots SSO izmantošanai. Aizsargā e-pakalpojumu pret starpvietņu skriptošanu, JavaScript injekcijām.

Manto no standarta ASP.NET pieprasījumu validatora (versija 4.0).

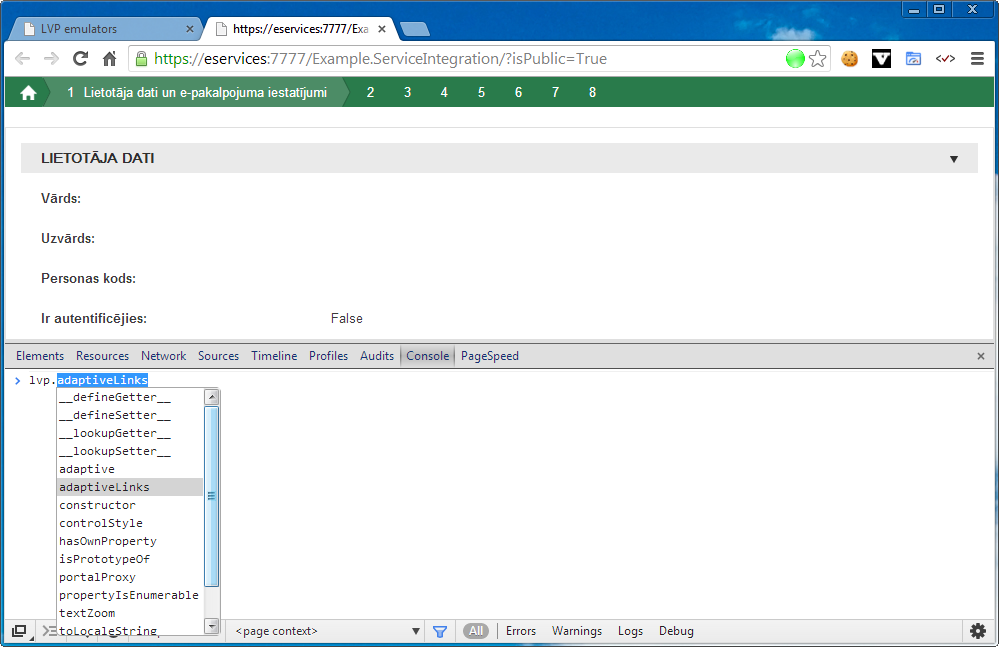
## JavaScript bibliotēkas

Caur CDN e-pakalpojumam tiek piegādāta virkne JS bibliotēku. Visas bibliotēkas tiek iekļautas vienā datnē, kas produkcijas vidē būs minificēta.

Trešo pušu bibliotēkas:

* **jQuery** – <http://jquery.com>;
* **jQuery UI** – <http://jqueryui.com> (tiek piegādāta pilna bibliotēkas versija ar pielāgotu tēmu (stiliem);
* **Adapt.js** – <http://adapt.960.gs>;
* **jQuery timepicker addon** <http://trentrichardson.com/examples/timepicker>.

LVP bibliotēkas ir pieejamas lvp nosaukumvietā (*namespace*).



87.attēls. Piekļuve LVP JS bibliotēkām

### lvp.portalProxy

Bibliotēka nodrošina metodes ziņojumu sūtīšanai uz un saņemšanai no LVP.

Piemērs:

$(document).ready(function () {

lvp.portalProxy.prolongPortalSession();

$("#uxWizardFinishButton").click(function () {

lvp.portalProxy.completeEservice();

});

lvp.portalProxy.onPrintClicked(function () {

self.focus();

self.print();

});

});

38.tabula

lvp.portalProxy metodes

| Metode | Apraksts |
| --- | --- |
| prolongPortalSession() | Nosūta pieprasījumu pagarināt LVP sesiju. |
| onTextZoomChanged(func) | Piereģistrē funkciju, kas tiks izsaukta, lietotājam LVP mainot teksta palielinājumu.   * func: notikumu apstrādājošā funkcija, kurai tiek padots viens parametrs (size - jaunais teksta palielinājums). |
| onBlindModeChanged(func) | Piereģistrē funkciju, kas tiks izsaukta, lietotājam LVP mainot teksta palielinājumu.   * func: notikumu apstrādājošā funkcija, kurai tiek padots viens parametrs (newMode – aktīvais vājredzīgo režīms). |
| onPrintClicked(func) | Piereģistrē funkciju, kas tiks izsaukta, lietotājam LVP nospiežot saiti „Drukāt”.   * func: notikumu apstrādājošā funkcija. |
| onBeforeLanguageChange(func) | Piereģistrē funkciju, kas tiks izsaukta pirms LVP valodas maiņas.   * func: notikumu apstrādājošā funkcija. |
| onPortalCommandReceived(commandName, func) | Piereģistrē funkciju, kas tiks izsaukta, kādai nākotnē potenciāli izstrādājamai LVP komponentei sūtot ziņojumu e-pakalpojumam. Aktuāla gadījumos, ja e-pakalpojums tiek izstrādāts kopā ar LVP komponenti.   * commandName: komandas nosaukums. * func: notikumu apstrādājošā funkcija. |
| sendCommandToPortal(commandName, commandArguments) | Uz LVP nosūta komandu un tās argumentus. Paredzēts lietot gadījumos, kad kopā ar e-pakalpojumu tiek izstrādāta arī LVP komponente (darbojas LVP vidē). |
| completeEservice | Nosūta LVP ziņojumu, ka e-pakalpojuma izpilde ir beigusies un var notikt pāreja uz citu LVP lapu. |
| showLoading | Liek LVP rādīt ielādes simbolu (loading) un bloķē lietotāja interfeisu. |
| hideLoading | Liek LVP paslēpt ielādes simbolu (loading) un atbloķēt lietotāja interfeisu. |
| redirect(url, usePost, postValues) | Liek LVP pāriet uz citu lapu.   * url: saite, uz kuru pāriet. * usePost: vai pāriet, izmantojot HTTP POST pieprasījumu (pēc noklusējuma tiek veikts GET pieprasījums). * postValues: *JavaScript* objekts ar POST pieprasījumā sūtāmajām vērtībām (tiek atbalstīti tikai objekti vienā līmenī). |
| errorOccured(query) | Paziņo LVP par kļūdas notikumu e-pakalpojumā. Metode nav paredzēta e-pakalpojumu izstrādātāju izmantošanai.   * query: *queryString* vērtības, kas tika padotas uz e-pakalpojuma kļūdu lapu. |

### lvp.controlStyle

Bibliotēkas uzdevums ir nodrošināt datu ievades elementiem vienoto LVP attēlojumu.

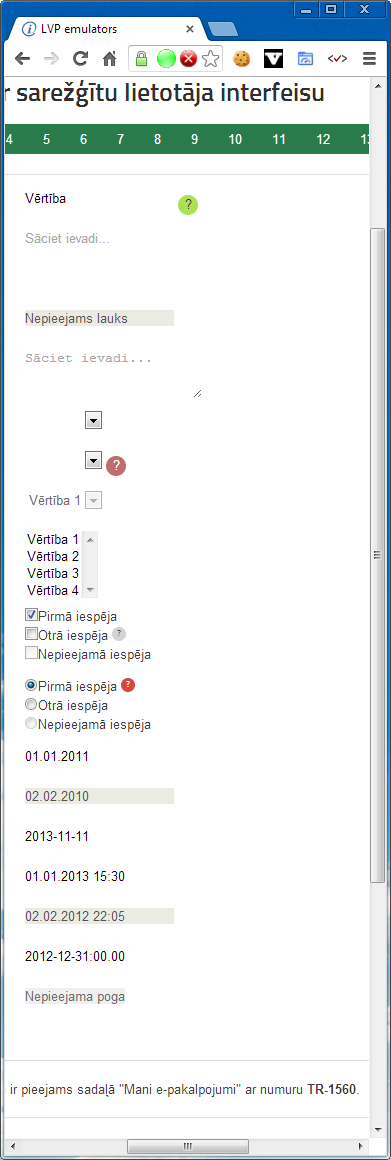
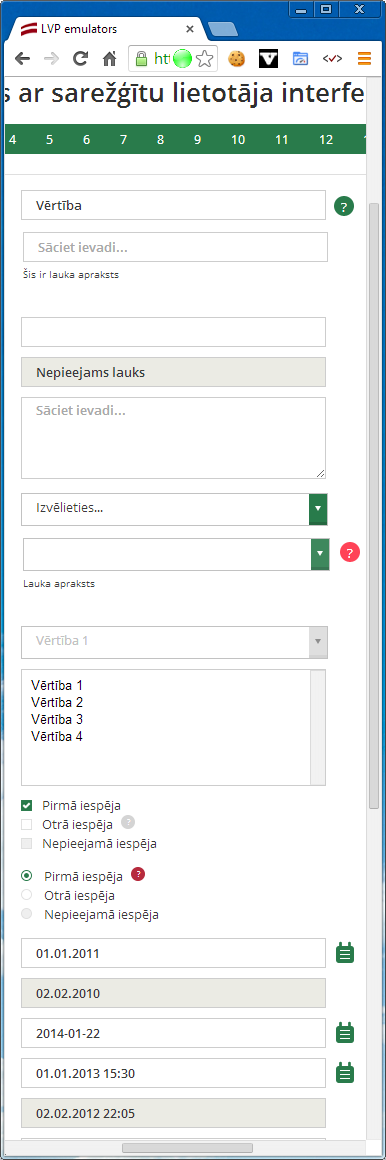
Piemērs:

lvp.controlStyle.applyStyle();

39.tabula

lvp.controlStyle metodes

|  |  |
| --- | --- |
| Metode | Apraksts |
| applyStyle(container) | Pievieno konteinera elementiem LVP stilu. Stils tiek piemērots tikai tiem elementiem, kas atrodas elementos ar CSS klasi auto-style, un paša elementa CSS klase nav no-style.   * container: HTML elements, kurā esošajiem elementiem stils jāpiemēro. Nenorādot parametru, tiek pārmeklēti visi lapas elementi. |
| dateValidationFunction(sender, args) | Funkcija, kas tiek izmantota *LvpDatePickerValidator* klienta puses (JS) validācijai. |
| refresh(item) | Ļauj atjaunot HTML elementu stilu pēc disabled (lielākajai daļai datu ievades elementu) vai *checked* (izvēles rūtiņām un radio pogām) īpašību maiņas, kas veikta pēc lapas ielādes ar *JavaScript*. Izmantošanas piemēram veltīts viens no *ComplexUi* e-pakalpojuma soļiem.   * item: HTML elements, kur stils jāatjauno. Nepadodot šo parametru, atjaunoti tiek visi lapas elementi. |
| monitorPopupOverflow() | Funkcija, kas tiek izmantota *LvpPopup* kontroļa garuma apstrādei. Novērš situācijas, kad iznirstošā loga lejas daļai atrodoties zemāk par dokumenta (iframe) apakšējo malu, šī daļa pazūd. |



88.attēls. E-pakalpojums ar un bez lvp.controlStyle izmantošanas

### lvp.displayState

Nodrošina piekļuvi LVP attēlojuma iestatījumiem ar JS.

Šī bibliotēka netiek piegādāta caur CDN, bet to ģenerē *LvpDisplayState* kontrolis (bibliotēkas darbībai ir nepieciešami citi CDN skripti).

40.tabula

lvp.displayState metodes

| Metode | Apraksts |
| --- | --- |
| getCurrentBlindMode() | Atgriež aktīvo vājredzīgo režīmu. Atgrieztā vērtība ir teksts, kuram jāsakrīt ar C# *BlindMode* *enum* vērtībām (piemēram, „None”). |
| getCurrentTextZoom() | Atgriež aktīvo teksta palielinājumu (skaitlis). |
| getCurrentResolution() | Atgriež aktīvo loga platumu (skaitlis no 0 līdz 3). |

### lvp.textZoom

Bibliotēka piedāvā iespēju palielināt vai samazināt lapas teksta izmēru. Bibliotēka izmantojama, ja tiek izlemts, ka e-pakalpojums atbalstīs LVP teksta palielināšanu/samazināšanas funkcionalitāti.

Piemērs:

$(document).ready(function () {

lvp.textZoom.setOptions({

selector: ".wizard-block, " +

".wizard-block input, " +

".wizard-block select, " +

".wizard-block textarea",

zoomStep: 1.1,

minLevel: -4,

maxLevel: 10

});

lvp.textZoom.zoom(lvp.displayState.getCurrentTextZoom());

lvp.portalProxy.onTextZoomChanged(function (level) {

lvp.textZoom.zoom(level);

});

});

41.tabula

lvp.textZoom metodes

| Metode | Apraksts |
| --- | --- |
| setOptions(newOptions) | Konfigurē teksta izmēra maiņas bibliotēku. newOptions argumenta lauki:   * selector: *jQuery* selektors teksta maiņai pakļauto elementu atrašanai. Pēc noklusējuma nav aizpildīts. * zoomStep: teksta palielināšanas/samazināšanas koeficients (piemēram, 1 – palielināšana/samazināšana netiek veikta, 2 – teksts katru reizi tiek palielināts/samazināts divas reizes. Pēc noklusējuma 1.2. * maxLevel: maksimālais lietotājam pieejamais palielināšanas reižu skaits (cik reizes var tikt pielietots zoomStep koeficients). Pēc šī līmeņa sasniegšanas tālāka teksta palielināšana vairs nenotiek. Pēc noklusējuma 10. * minLevel: minimālais lietotājam pieejamais samazināšanas reižu skaits. Pēc šī līmeņa sasniegšanas tālāka teksta samazināšana vairs nenotiek. Pēc noklusējuma – 10. |
| zoom(level) | Palielina vai samazina tekstu līdz norādītajam level (ja tas iekļaujas minLevel - maxLevel intervālā). Piemēram, 0 nozīmē, ka teksts tiek atgriezts sākotnējā izmērā. Savukārt, 3 nozīmē, ka teksts tiks palielināts zoomStep \* zoomStep \* zoomStep reizes. |

# E-pakalpojumu konfigurācija

## lvp/general sekcija

Konfigurācijas sekcijā *lvp/general* tiek norādīti e-pakalpojuma pamata iestatījumi (skat 42.tabulu).

<lvp>

<general eserviceUrn="URN:IVIS:100001:EP-ExampleServiceIntegration-v1-0"

cdnUrl="https://eservices:7777/PortalEmulator/cdn/{{version}}"

routerServiceUrl="https://eservices:7777/Router/RouterService.svc" />

…

</lvp>

42.tabula

lvp.general elementu uzskaitījums

|  |  |
| --- | --- |
| **Iestatījums** | **Apraksts** |
| eserviceUrn | E-pakalpojuma identifikators jeb URN (izsniedz EPU). |
| cdnUrl | CDN saknes URL (tiek izmantots CSS un JS piegādāšanai e-pakalpojumam). Ja ir norādīts vietturis {{version}}, tas tiek aizstāts ar vērtību no emulatora vai e-pakalpojumu kataloga (lauks CDN versija). |
| routerServiceUrl | Maršrutēšanas servisa URL. |
| userProfileApplicationName | Identificē sistēmu, kuras lietotāju profila dati tiek pieprasīti. Parasti nevajag norādīt, jo noklusētā vērtība ir atbilstoša. |
| edkRepositoryId | Nosaka EDK izmantojamo repozitoriju. Paredzēts izmantot tikai vienu repozitoriju, tādēļ parasti vērtību nevajag norādīt, jo pēc noklusējuma tā ir atbilstoša. |
| portalConfigCacheTimeoutInSeconds | Nosaka, cik ilgi kešatmiņā tiek glabāti no e-pakalpojumu konfigurācijas servisa izgūtie iestatījumi. Pēc noklusējuma 1 gads. Norādot vērtību 0, rezultāti kešatmiņā glabāti netiek. Produkcijā vēlams šo iestatījumu nenorādīt (noklusētā vērtība), jo e-pakalpojumiem tiek sūtīti ziņojumi par konfigurācijas izmaiņām. Saņemot šādu ziņojumu, kešatmiņa tiek tīrīta. |
| manageUiCulture | Nosaka, vai e-pakalpojuma lapām automātiski uzstādīt UI kultūras iestatījumus (Thread.CurrentThread.CurrentCulture). Pēc noklusējuma True. |
| speedLogEnabled | Nosaka, vai veikt servisu izsaukumu laika trasēšanu. Pēc noklusējuma True. |

## lvp/esigner sekcija

Konfigurācijas sekciju *lvp/esigner* nepieciešams norādīt tikai, ja e-pakalpojumā tiek izmantots e-parakstīšanas kontrolis (skat. 7.2.20.sadaļu).

<lvp>

…

<esigner

javaScriptLibraryUrl="https://signetest.vraa.gov.lv/WebStatic/Scripts/

Zzdats.Div.DocumentViewer.min.js"

statusServerUrl="https://signetest.vraa.gov.lv/WebApi/"

staticDataBaseUrl="https://signetest.vraa.gov.lv/WebStatic/" />

…

<lvp>

43.tabula

lvp.esigner elementu uzskaitījums

|  |  |
| --- | --- |
| Iestatījums | Apraksts |
| javaScriptLibraryUrl | Saite, no kurienes ielādēt DocumentViewer *JavaScript* e-parakstīšanas komponenti (skat. [2] atsauci). |
| statusServerUrl | DocumentViewer komponentes inicializācijai padodamā statusServerUrl parametra vērtība (skat. [2] atsauci). |
| staticDataBaseUrl | DocumentViewer komponentes inicializācijai padodamā staticDataBaseUrl parametra vērtība (skat. [2] atsauci). |

## lvp/addressFinder sekcija

Konfigurācijas sekciju *lvp/addressFinder* nepieciešams norādīt tikai, ja e-pakalpojumā tiek izmantots adrešu meklēšanas kontrolis (skat. 7.2.21.sadaļu).

<lvp>

…

<addressFinder

javaScriptLibraryUrl="https://tv-vraa.zzdats.lv/sites/Vraa.AddressFinder.Web/Scripts/Vraa.AddressFinder.js"

statusServerUrl="https://tv-vraa.zzdats.lv/services/Vraa.AddressFinder.WebApi/"

staticDataBaseUrl="https://tv-vraa.zzdats.lv/sites/Vraa.AddressFinder.Web/"/>

…

<lvp>

44.tabula

lvp.addressFinder elementu uzskaitījums

|  |  |
| --- | --- |
| Iestatījums | Apraksts |
| javaScriptLibraryUrl | Saite, no kurienes ielādēt AddressFinder *JavaScript* adrešu izvēles komponenti (skat. [7] atsauci). |
| statusServerUrl | AddressFinder komponentes inicializācijai padodamā statusServerUrl parametra vērtība (skat. [7] atsauci). |
| staticDataBaseUrl | AddressFinder komponentes inicializācijai padodamā staticDataBaseUrl parametra vērtība (skat. [7] atsauci). |

## lvp/oldPayment sekcija

Konfigurācijas sekciju nepieciešams norādīt tikai, ja e-pakalpojumā tiek izmantota maksājumu moduļa vecā versija (skat. 7.1.8).

Pamatā elementi tiek izmantoti MM moduļa noklusētā PS pieprasījuma veidošanai. Ja konfigurācijā ir norādīta vērtība, tā tiek izmantota atbilstošā pieprasījuma lauka aizpildīšanai. Ja vērtība nav norādīta, lauks netiek aizpildīts.

<oldPayment paymentUrl="https://tv-vraa.zzdats.lv/sites/VRAA.VISS2.MM/WebUI/Payments/Transfer"

taxPmtFlg="N"

docNo="1"

ccy="LVL"

pmtInfo="Payment info">

<benSets>

<benSet amt="5.99"

benAccNo="LV58HABA0551017432589"

benName="SIA Qwerty"

benLegalId="40000000009"

bbName="Swedbank"

bbSwift="HABALV22"

benCountry="LV"/>

</benSets>

</oldPayment>

45.tabula

lvp.oldPayment elementu uzskaitījums

| Iestatījums | Apraksts |
| --- | --- |
| paymentUrl | Saite uz maksājumu lapu. |
| taxPmtFlg | Vērtība laukam PaymentRequest.TaxPmtFlg. |
| docNo | Vērtība laukam PaymentRequest.DocNo. |
| ccy | Vērtība laukam PaymentRequest.Ccy. |
| pmtInfo | Vērtība laukam PaymentRequest.PmtInfo. |
| bankInfo | Vērtība laukam PaymentRequest.BankInfo. |
| extId | Vērtība laukam PaymentRequest.ExtId. |
| institutionId | Vērtība laukam PaymentRequest.InstitutionID. |
| payerAcceptStatus | Vērtība laukam PaymentRequest.PayerAcceptStatus. |
| paymentId | Vērtība laukam PaymentRequest.PaymentID. |
| regDate | Vērtība laukam PaymentRequest.RegDate. |
| termOfPayment | Vērtība laukam PaymentRequest.TermOfPayment. |
| reminder | Vērtība laukam PaymentRequest.Reminder. |
| benSets\amt | Vērtība laukam PaymentRequest.BenSet.Amt. |
| benSets\benAccNo | Vērtība laukam PaymentRequest.BenSet.BenAccNo. |
| benSets\benName | Vērtība laukam PaymentRequest.BenSet.BenName. |
| benSets\benLegalId | Vērtība laukam PaymentRequest.BenSet.BenLegalId. |
| benSets\bbName | Vērtība laukam PaymentRequest.BenSet.BBName. |
| benSets\bbSwift | Vērtība laukam PaymentRequest.BenSet.BBSwift. |
| benSets\benCountry | Vērtība laukam PaymentRequest.BenSet.BenCountry. |
| benSets\amkSets\amt | Vērtība laukam PaymentRequest.BenSet.AmkSet.Amt. |
| benSets\amkSets\opc | Vērtība laukam PaymentRequest.BenSet.AmkSet.Opc. |
| benSets\bbAddress | Vērtība laukam PaymentRequest.BenSet.BBAddress. |
| benSets\bbCodeCodeType | Vērtība laukam PaymentRequest.BenSet.BBCode.CodeType. |
| benSets\bbCodeValue | Vērtība laukam PaymentRequest.BenSet.BBCode.Value. |
| benSets\benAccIbanFlg | Vērtība laukam PaymentRequest.BenSet.BenAccIbanFlg. |
| benSets\benAddress | Vērtība laukam PaymentRequest.BenSet.BenAddress. |
| benSets\benExtId | Vērtība laukam PaymentRequest.BenSet.BenExtId. |
| benSets\benSubAccNo | Vērtība laukam PaymentRequest.BenSet.BenSubAccNo. |
| benSets\budgCode | Vērtība laukam PaymentRequest.BenSet.BudgCode. |
| benSets\cbAcc | Vērtība laukam PaymentRequest.BenSet.CBAcc. |
| benSets\cbCodeCodeType | Vērtība laukam PaymentRequest.BenSet.BCode.CodeType. |
| benSets\cbCodeValue | Vērtība laukam PaymentRequest.BenSet.BCode.Value. |
| benSets\cbName | Vērtība laukam PaymentRequest.BenSet.CBName. |
| benSets\cbSwift | Vērtība laukam PaymentRequest.BenSet.CBSwift. |
| benSets\comm | Vērtība laukam PaymentRequest.BenSet.Comm. |
| benSets\priority | Vērtība laukam PaymentRequest.BenSet.Priority. |

## lvp/payment sekcija

Konfigurācijas sekciju nepieciešams norādīt tikai, ja e-pakalpojumā tiek izmantots maksājumu modulis (skat. 7.1.9.sadaļu).

Elementi tiek izmantoti, MM moduļa noklusētā PS pieprasījuma veidošanai. Ja konfigurācijā ir norādīta vērtība, tā tiek izmantota atbilstošā pieprasījuma lauka aizpildīšanai. Ja vērtība nav norādīta, lauks netiek aizpildīts.

Lielākā daļa konfigurācijas (*allowedBanks, serviceStep, swiftCode, price* u.c.) ir tieši atkarīga no MM konfigurācijas PPK sistēmā. Tādēļ, veicot konfigurāciju, parasti ir jāsazinās ar EPU (VRAA).

<payment allowedBanks="DEMO0000,HABALV22"

expirationIntervalInMinutes="20160"

noInvoice="false"

paymentObjectType="Payment Object"

responseUrl= "https://eservices:7777/PortalEmulator/EserviceHost.aspx?URN=URN%3aIVIS%3a100001%3aEP-ExampleInfrastructure-v1-0&amp;WrapperContent=Default&amp;paymentId="

serverResponseUrl="http://tv-vraa.zzdats.lv"

serviceStep="02Y70"

swiftCode="DEMO0000" >

<prices>

<price position="53"

item="1"

itemElementName="Quantity"/>

<price position="54"

item="12.12"

itemElementName="Amount"/>

</prices>

</payment>

46.tabula

lvp.payment elementu uzskaitījums

| Iestatījums | Apraksts |
| --- | --- |
| allowedBanks | Vērtība laukam PaymentRequest.AllowedBanks.  Vērtību saraksts. Jāatdala ar komatiem. Norādot tukšu vērtību, tiks piedāvātas visas bankas. |
| expirationIntervalInMinutes | Vērtība laukam PaymentRequest.ExpiryDate.  Lauka vērtība tiek uzstādīta kā šī brīža datums + norādītais minūšu skaits. |
| noInvoice | Vērtība laukam PaymentRequest.NoInvoice. |
| paymentObjectType | Vērtība laukam PaymentRequest.PaymentObject. |
| responseUrl | Vērtība laukam PaymentRequest.ResponseUrl. |
| serverResponseUrl | Vērtība laukam PaymentRequest.ServerResponseUrl. |
| serviceStep | Vērtība laukam PaymentRequest.ServiceStep. |
| swiftCode | Vērtība laukam PaymentRequest.SwiftCode. |
| prices/price/position | Vērtība laukam PaymentRequest.PriceList.Position. |
| prices/price/item | Vērtība laukam PaymentRequest.PriceList.Item. |
| prices/price/itemElementName | Vērtība laukam PaymentRequest.PriceList.ItemElementName. |

## lvp/errorReport sekcija

Elements tiek izmantots kļūdu pieteikšanas funkcionalitātes konfigurācijas nodrošināšanai (skat. 7.2.6.sadaļu). Parasti šīs sekcijas modificēšana nav nepieciešama (noklusētās vērtības būs korektas).

## lvp/messages/message sekcija

Konfigurācijas sekcijā *lvp/messages* tiek norādīti PS izsaukšanai nepieciešamie iestatījumi (skat 47.tabulu). Katram izsaukuma veidam jābūt reģistrētam šajā sekcijā.

<lvp>

…

<messages>

<message

name="Calculate-Sync"

integrationServiceUrn="URN:IVIS:100001:ISS-SIA.ABC-CalculationDataSync-v1-0"

messageType="URN:IVIS:100001:XSD-Testing-TestISServise-v1-0-TYPE-Calculation"

useTestService="true"/>

</lvp>

47.tabula

lvp.messages.message elementu uzskaitījums

| Iestatījums | Apraksts |
| --- | --- |
| name | Konfigurācijas sekcijas identifikators. Veicot PS izsaukumu, pēc šīs vērtības tiek meklēta atbilstošā izsaukuma konfigurācija. |
| integrationServiceUrn | Izsaucamā integrācijas servisa identifikators (pieprasījuma Header.Destination lauks). |
| messageType | Pieprasījuma tipa identifikators (pieprasījuma Header.MessageType lauks). Tiek izmantots, lai atrastu XSD shēmu, pret kuru validēt pieprasījuma saturu. |
| milestoneUrn | Pieturpunkta identifikators (pieprasījuma Header.MilestoneID lauks). |
| nextMilestoneUrn | Nākamā pieturpunkta identifikators (pieprasījuma Header.NextMilestoneID lauks). |
| responseEndpoint | Pieprasījuma Header.ResponseEndPoint lauks. |
| useTestService | Vai izmantot PS testa versiju (pieprasījuma Header.Test lauks). |
| bodySerializer | Pieprasījuma Body daļas serializatora tipa nosaukums. Ja vērtība nav norādīta, tiek lietots DefaultBodySerializer, kas izmanto standarta .NET XmlSerializer. Norādītajam tipam ir jāimplementē interfeiss IBodySerializer. |

## lvp/nasa sekcija

Apraksta ar NASA notikumu klausīšanos saistīto konfigurāciju. NASA tiek izmantota, lai no e-pakalpojumu kataloga saņemtu ziņojumu par e-pakalpojuma iestatījumu maiņu. Vienīgais parametrs, kuru varētu būt nepieciešams mainīt izstrādes laikā, ir subscribeToNotifications.

<lvp>

  …

  <nasa subscribeToNotifications="false" />

  …

</lvp>

48.tabula

lvp.nasa elementu uzskaitījums

| Iestatījums | Apraksts |
| --- | --- |
| topic | NASA pieraksta temats (pēc noklusējuma „EserviceConfigChange”). |
| filterXPathTemplate | NASA notikumu filtra šablons. |
| filterNamespacePrefix | filterXPathTemplate izmantotās vārdu telpas apzīmējums (prefikss). |
| filterNamespace | filterXPathTemplate izmantotā vārdu telpa, kas tiek apzīmēta ar filterNamespacePrefix. |
| subscriptionLengthInMinutes | NASA pieraksta ilgums. Pēc noklusējuma 20 dienas. |
| subscribeToNotifications | Nosaka, vai veikt pierakstīšanos uz NASA notikumiem.  Izstrādes vidē e-pakalpojuma konfigurācija tiek ņemta no emulatora, nevis e-pakalpojumu kataloga. Līdz ar to pierakstam uz notifikācijām nav nozīmes.  Savukārt, produkcijas vidē pieraksts uz NASA notikumiem ir nepieciešams.  Pēc noklusējuma true. |

## EserviceConfiguration galapunkta sekcija

Gadījumā, ja e-pakalpojuma konfigurācijas datnē ir WCF galapunkta (*endpoint*) sekcija ar nosaukumu EserviceConfiguration, tā tiek izmantota, lai pieslēgtos e-pakalpojumu konfigurācijas servisam. Ja sekcijas ar minēto nosaukumu nav, konfigurācijas servisa izsaukšana notiek caur maršrutēšanas servisu.

Tādēļ gadījumos, kad e-pakalpojumu izstrādā vairāki cilvēki, un tiek izmantots viens maršrutēšanas serviss, šai sekcijai ir jāpaliek e-pakalpojuma konfigurācijā, lai nodrošinātu, ka e-pakalpojuma uzstādījumi tiek ņemti no katra izstrādātāja emulatora. Savukārt, ja maršrutēšanas serviss ir uzstādīts pie katra izstrādātāja vai arī e-pakalpojums tiek darbināts produkcijas vidē, šo sekciju no e-pakalpojuma konfigurācijas var dzēst un paļauties uz maršrutēšanas servisa iestatījumiem.

<endpoint

name="EserviceConfiguration"

address="https://eservices:7777/PortalEmulator/Services/ EserviceConfiguration.svc/soap12"

binding="customBinding"

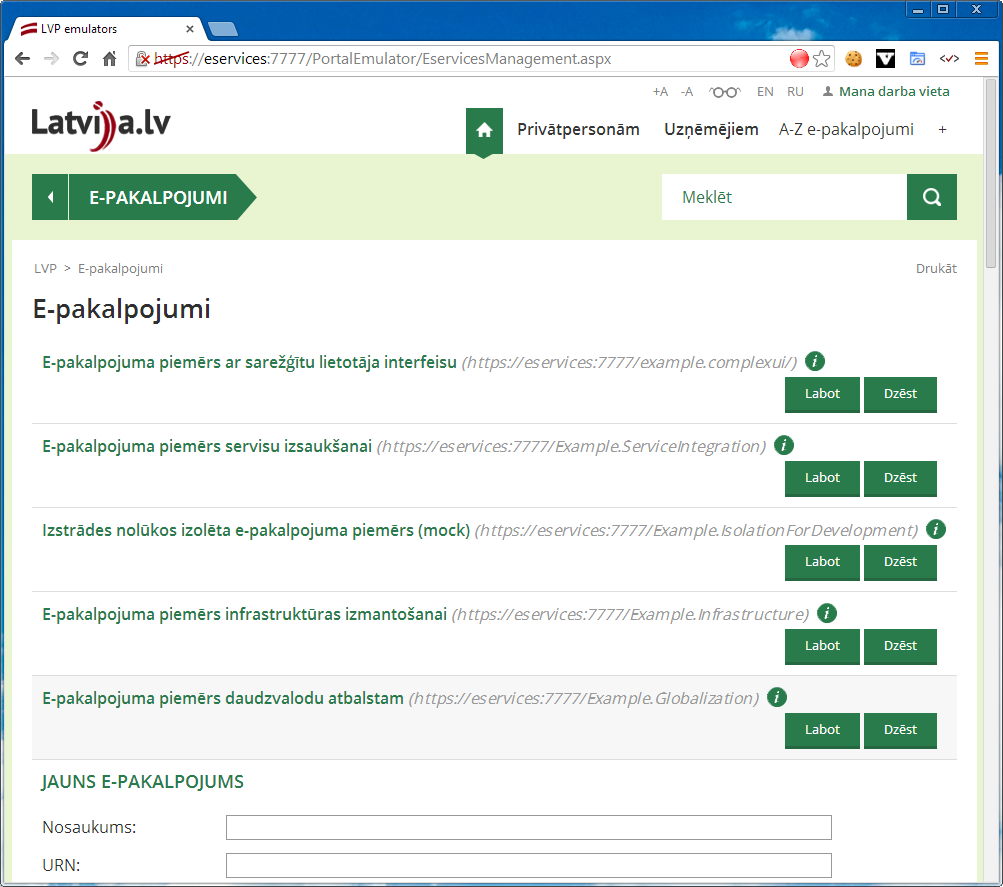
bindingConfiguration="EserviceConfigurationBinding"

contract= "Viss.EserviceCatalogue.DataServices.Base.Configuration.IEserviceConfigurationService" />

# Portāla emulatora lietotāja ceļvedis

Pēc noklusētās konfigurācijas emulators pieejams adresē [*https://eservices:7777/PortalEmulator*](https://eservices:7777/PortalEmulator).

## E-pakalpojumu saraksta lapa



89.attēls. Emulatora sākuma lapa

No sākuma lapas, iespējams, atvērt vai labot kādu no reģistrētajiem e-pakalpojumiem, kā arī, piereģistrēt jaunu e-pakalpojumu.

Reģistrētie e-pakalpojumi glabājas lokālā datu bāzē (Lvp.Eservice.PortalEmulator\App\_Data\Portal.sdf). Ja minētais fails neeksistē, pie portāla atvēršanas tas tiek izveidots no jauna un aizpildīts ar piemēru e-pakalpojumu datiem.

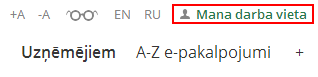
E-pakalpojumu reģistrācijas lauki:

* **Nosaukums**: informatīvs teksts, kas tiks attēlots e-pakalpojuma virsrakstā.
* **URN**: e-pakalpojuma identifikators (obligāti jāsakrīt ar e-pakalpojuma konfigurācijas failā norādīto (*configuration/lvp/general/eserviceUrn*).
* **Iestādes kods**: e-pakalpojuma turētājas iestādes kods. Tiek izmantots, piemēram, e-pakalpojumu meklēšanā latvija.lv portālā.
* **Saite**: e-pakalpojuma izmitināšanas URL (tikai saknes daļa – bez konkrēta lapas nosaukuma).
* **Portāla izkārtojums**: norāda, cik plats būs e-pakalpojumam atvēlētais konteiners (iframe). LVP būs iespējams konfigurēt sadaļas, kas tiks attēlotas e-pakalpojuma kreisajā pusē. No tā būs atkarīgs, cik daudz vietas paliek pašam e-pakalpojumam. Pirms izstrādes uzsākšanas ar klientu ir jāvienojas, kādā izkārtojumā e-pakalpojums tiks darbināts LVP. Lielākā daļa e-pakalpojumu izmanto vienas kolonnas izkārtojumu.
* **CDN versija**: norāda e-pakalpojuma izmantoto CDN versiju. Tiek izmantota emulatorā pakalpojuma ielādei, kā arī e-pakalpojumā, ja konfigurācijā norādīts vietturis {{version}}.
* **Mērogojams interfeiss**: nosaka, vai e-pakalpojums atbalsta lapas izmēra maiņu atbilstoši pārlūka izmēram (izmēra maiņa uz pārlūka loga platuma palielināšanu/samazināšanu). Iestatījums attiecas tikai uz šaurajiem (mobilajiem) skatiem. Divi platākie režīmi mainās vienmēr, neatkarīgi no šī iestatījuma. Pirms izstrādes uzsākšanas jāvienojas, vai e-pakalpojums atbalstīs šādu funkcionalitāti.
* **Anonīms**: Nosaka, vai e-pakalpojums ir pieejams visiem lietotājiem (tai skaitā anonīmajiem). Ja ir atzīmēta šī pazīme, autentifikāciju (Fiziskas personas autentifikācija u.c.) lauku vērtības tiek ignorētas.
* **Nedeklarētas identitātes autentifikācijas**: Nedeklarētās identitātes lietotāju atļauto identitātes sniedzēju saraksts. Ja lietotājs ir pieslēdzies ar nedeklarētu identitāti, e-pakalpojums būs pieejams tikai, ja lietotāja identitātes sniedzējs būs šajā sarakstā. Jānorāda latvija.lv lietotāja identitātes sniedzējs. Visos autentifikācijas laukos ir pieejamas šādas vērtības:
* URN:IVIS:111111:AM-ALLBANKS – ar visu banku (SEB, Swedbank, DNB, Nordea u.c.) lietotājiem, šī ir autentifikāciju grupa;
* URN:IVIS:111111:AM-ALLESIGNATURES – ar visiem e-parakstiem (eID, e-me utt.), šī ir autentifikāciju grupa;
* URN:IVIS:100001:AM.BANK-DNB – ar DNB bankas lietotāju;
* URN:IVIS:100001:AM.SIGN-EID – ar eID e-parakstu;
* URN:IVIS:100001:AM.BANK-SWED – ar Swedbank lietotāju;
* URN:IVIS:100001:AM.SIGN-EME – ar e-me e-parakstu;
* URN:IVIS:100001:AM.NI-LVP – ar latvija.lv lietotāju (nedeklarētā identitāte);
* URN:IVIS:100001:AM.BANK-NORDEA – ar Nordea bankas lietotāju;
* URN:IVIS:100001:AM.BANK-NORVIKBANKA – ar Norvik bankas lietotāju;
* URN:IVIS:100001:AM.BANK-CITADELE – ar Citadele bankas lietotāju;
* URN:IVIS:100001:AM.BANK-SEB – ar SEB bankas lietotāju;
* URN:IVIS:100001:AM-IDP40-WIF– ar testa vides autentifikāciju.
* **Fiziskas personas autentifikācijas**: Fiziskās personas lietotāju atļauto identitātes sniedzēju saraksts. Ja lietotājs ir pieslēdzies ar privātpersonas identitāti, e-pakalpojums būs pieejams tikai, ja lietotāja identitātes sniedzējs būs šajā sarakstā. Parasti jānorāda visas bankas un visu e-parakstu grupas.
* **Juridiskas personas autentifikācijas**: Juridiskās personas lietotāju atļauto identitātes sniedzēju saraksts. Ja lietotājs ir pieslēdzies ar juridiskās personas identitāti, e-pakalpojums būs pieejams tikai, ja lietotāja identitātes sniedzējs būs šajā sarakstā. Parasti jānorāda visas bankas un visu e-parakstu grupas.
* **Atbalsta daudzvalodību**: pazīme, vai e-pakalpojums ir pieejams vairākās valodās.
* **Nerādīt valodas maiņas brīdinājumu**: pazīme ļauj nerādīt lietotājam brīdinājumu pirms valodas maiņas. Valodas maiņai LVP ir nepieciešama pilna lapas pārlāde, kas, savukārt, nozīmē e-pakalpojuma *iframe* pārlādi un *View-state* saglabāto vērtību zaudēšanu. Šī opcija būtu jāatzīmē tikai gadījumos, ja e-pakalpojums minēto lapas pārlādi spēj „pārdzīvot”, nezaudējot lietotāju ievadītos datus, un tekošo atvērto soli.
* **Nodot pakalpojumam query string parametrus**: Vai pārsūtīt e-pakalpojumam *query string* parametrus, kas saņemti, atverot emulatora/portāla pakalpojuma lapu. Jāatzīmē tikai, ja pakalpojums izmanto šādu funkcionalitāti. Tālāk tiek padoti tikai tie parametri, kuru nosaukumi nesakrīt ar portāla/emulatora – pakalpojuma saziņā izmantotajiem (predefinētajiem).
* **Apraksts, uzsākot pakalpojumu (latviski, angliski, krieviski)**: ziņojuma teksts, kas tiek rādīts pirms e-pakalpojuma uzsākšanas un kuram lietotājam ir jāpiekrīt darba turpināšanai. Iespēja ievadīt vairākas valodas. Ja e-pakalpojums neatbalsta daudzvalodību, tiek izmantota tikai latviešu versija.
* **Lietotāja instrukcija (latviski, angliski, krieviski)**: Teksts, kas tiks iekļauts e-pakalpojuma lietotāja instrukcijas lapā.
* **Lasāmie lietotāja profila atribūti**: lietotāja (fiziska persona) profila atribūti, kurus e-pakalpojumam būs atļauts lasīt.
* **Rakstāmie lietotāja profila atribūti**: lietotāja (fiziska persona) profila atribūti, kurus e-pakalpojums drīkstēs mainīt.
* **Lasāmie uzņēmuma profila atribūti**: lietotāja pārstāvētā uzņēmuma profila atribūti, kurus e-pakalpojumam būs atļauts lasīt.
* **Rakstāmie uzņēmuma profila atribūti**: lietotāja pārstāvētā uzņēmuma profila atribūti, kurus e-pakalpojumam būs atļauts mainīt.

Emulatorā tiek reģistrēta tikai informācija, kas nepieciešama e-pakalpojuma darbības emulēšanai. Konfigurācija LVP var ievērojami atšķirties.

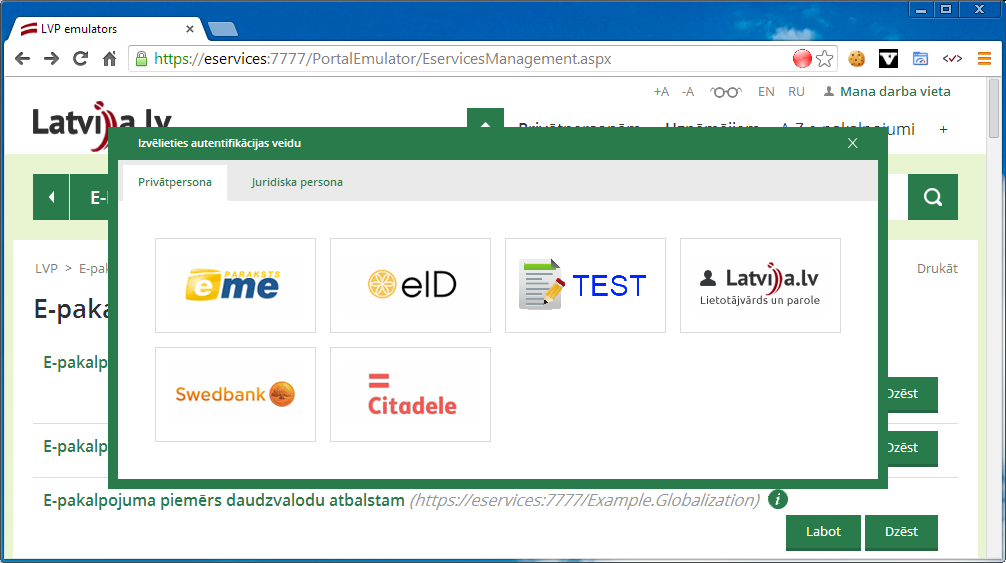
## Autentifikācija

Lai autentificētos emulatorā (un automātiski arī e-pakalpojumos), jānospiež lapas augšējā kreisajā stūrī esošā saite „Mana darba vieta”.



90.attēls. Autentifikācijas loga atvēršana

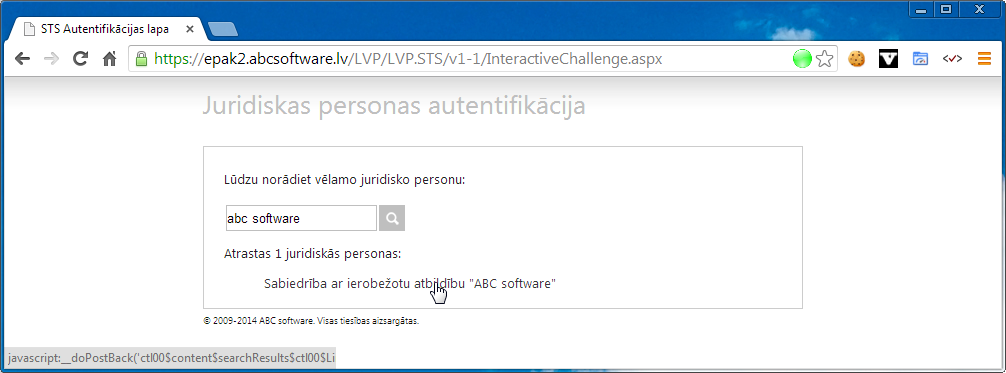
Autentifikācijas logā jāizvēlas autentifikācijas veida cilne (Privātpersona / Juridiska persona) un jānospiež uz izvēlēto autentifikācijas sniedzēju.



91.attēls. Autentifikācijas loga atvēršana

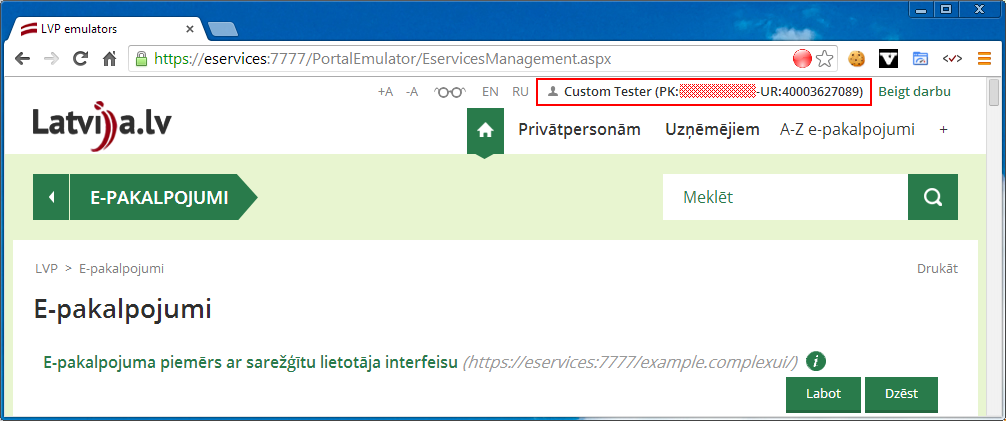
Notiek pāreja pie autentifikācijas sniedzēja, kur jāievada autentifikācijai nepieciešamie dati.

Ja tika izvēlēta juridiskās personas autentifikācija, pirms pārejas atpakaļ uz emulatoru tiek pieprasīts ievadīt uzņēmumu, kuru persona pārstāvēs. Lai uzņēmums būtu pieejams, lietotāja personas kodam jābūt reģistrētam kā uzņēmuma paraksttiesīgai personai Uzņēmumu reģistrā. Ja uzņēmums lietotājam nav pieejams, netiek veikta juridiskas personas autentifikācija un lietotājs tiek autentificēts kā privātpersona.



92.attēls. Pārstāvētā uzņēmuma izvēle

Pēc veiksmīgas autentifikācijas emulatora augšējā daļā tiek attēlots lietotāja vārds, uzvārds un identifikators (NameIdentifier). Juridiskas personas gadījumā identifikators sastāv no lietotāja personas koda (*PK:* daļa) un uzņēmuma reģistrācijas numura (*UR:* daļa). Ja kods sastāv tikai no personas koda daļas, lietotājs ir autentificējies kā privātpersona.



93.attēls. Informācija par autentificēto lietotāju

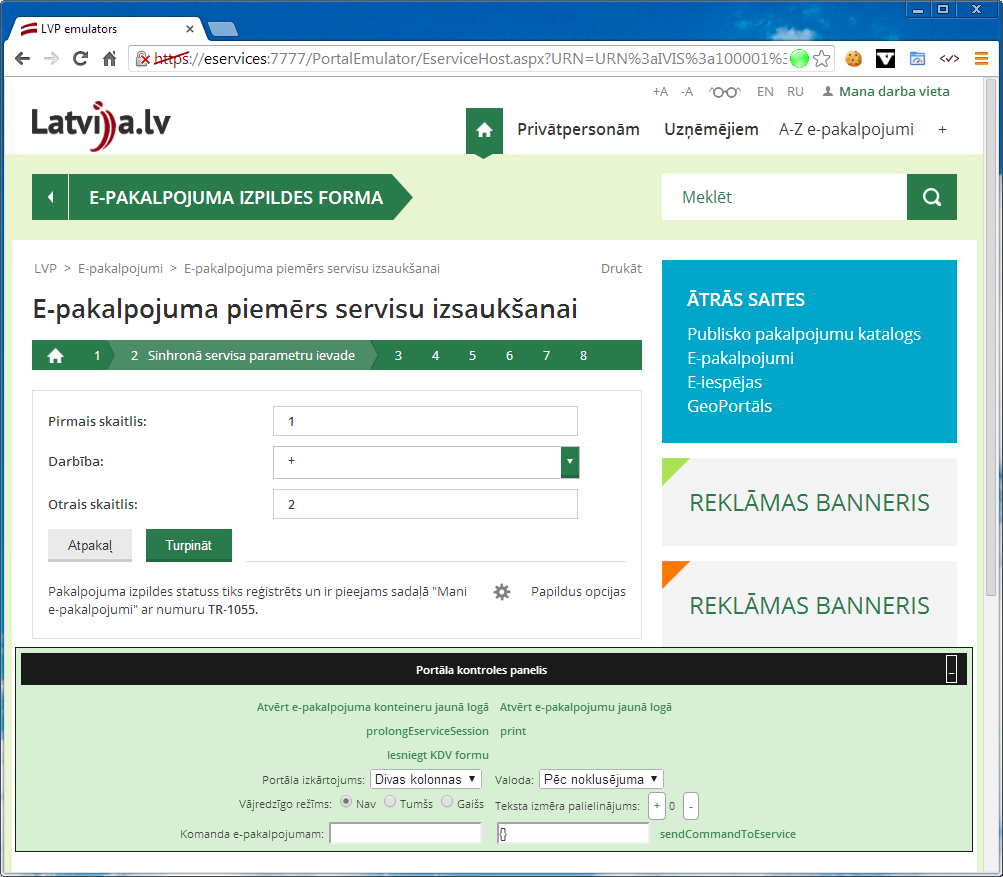
## E-pakalpojuma darbināšanas lapa

Noklikšķinot uz e-pakalpojuma nosaukuma emulatora sākuma lapā, tiek ielādēts e-pakalpojums.

Kreisajā daļā redzams e-pakalpojums. Labajā pusē atkarībā no izvēlētā izkārtojuma tiek attēloti paraugi komponentēm, kas LVP varētu tikt attēloti kopā ar e-pakalpojumu (pēc noklusējuma LVP gan šīs komponentes netiek attēlotas, nodrošinot pilnu platumu pakalpojumam).

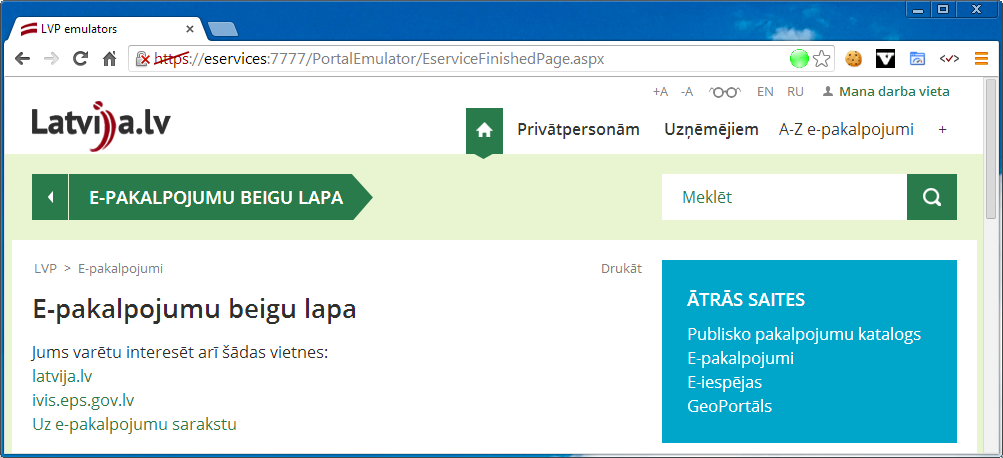
Lapas apakšdaļā redzams kontroles panelis, kas ļauj imitēt lietotāja portālā veiktās darbības un veikt dažas citas operācijas:

* **Atvērt e-pakalpojuma konteineru jaunā logā**: atver e-pakalpojuma *wrapper l*apu (ārējo *iframe* – skat. 7.2.30.sadaļu).
* **Atvērt e-pakalpojumu jaunā logā**: atver e-pakalpojumu (iekšējo *iframe*). Šī funkcionalitāte var būt noderīga, piemēram, lai no pārlūka konsoles izsauktu kādu no e-pakalpojuma JS funkcijām.
* **prolongEserviceSession**: nosūta e-pakalpojumam sesijas pagarināšanas pieprasījumu.
* **print**: nosūta e-pakalpojumam drukāšanas pieprasījumu (tas pats, kas portālā nospiest saiti „drukāt”).
* **Iesniegt KDV formu**: KDV formas nosūtīšanas pieprasījums. Nav izmantojams e-pakalpojumu izstrādē, bet ir nepieciešams KDV formu izstrādei.
* **Portāla izkārtojums**: Ļauj mainīt e-pakalpojumam pieejamo platumu. LVP šāda opcija lietotājiem noteikti nebūs pieejama un glabāsies konfigurācijā. Pievienota tikai izstrādes nolūkiem, lai varētu redzēt, kā mainās e-pakalpojuma attēlojums dažādos platumos.
* **Valoda**: Ļauj mainīt tekošo valodu (tas pats, kas valodas maiņu pogu nospiešana). Emulatorā nav realizēta portāla (ārējā daļa ārpus *iframe*) tekstu maiņa, bet LVP visi portāla teksti tiks tulkoti atbilstoši izvēlētajai valodai.
* **Vājredzīgo režīms**: Ļauj lietotājam portālā ieslēgt vājredzīgo režīmu. Pirms e-pakalpojuma izstrādes jāvienojas ar pasūtītāju, vai e-pakalpojums atbalstīs šādu funkcionalitāti.
* **Teksta izmēra palielinājums**: Ļauj lietotājam mainīt teksta palielinājumu (analoģiski +A un – A ikonām lapas augšdaļā). Pirms e-pakalpojuma izstrādes jāvienojas ar pasūtītāju, vai e-pakalpojums atbalstīs šādu funkcionalitāti.
* **Komanda e-pakalpojumam**: Ļauj sūtīt *JavaScript* komandu uz e-pakalpojumu. Paredzēta gadījumiem, ja kopā ar e-pakalpojumu tiks izstrādāta arī kāda LVP (*Sitecore*) komponente, kurai būs nepieciešama saziņa ar pakalpojumu. Pirmajā laukā ievadāms komandas nosaukums, bet otrajā – komandai padotie dati (JSON).



94.attēls. Emulatorā atvērts e-pakalpojums

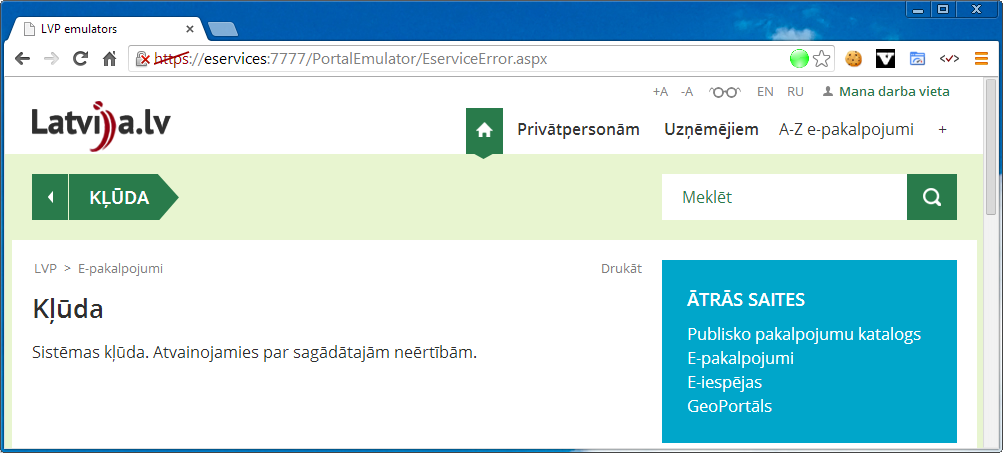
## E-pakalpojuma beigu lapa



95.attēls. E-pakalpojuma noslēguma lapa

Pēc e-pakalpojuma pabeigšanas notiek pāreja uz portāla lapu. LVP šajā lapā tiek attēlots e-pakalpojumu saraksts. Emulatorā – vienkāršs saišu saraksts.

## E-pakalpojuma kļūdu lapa

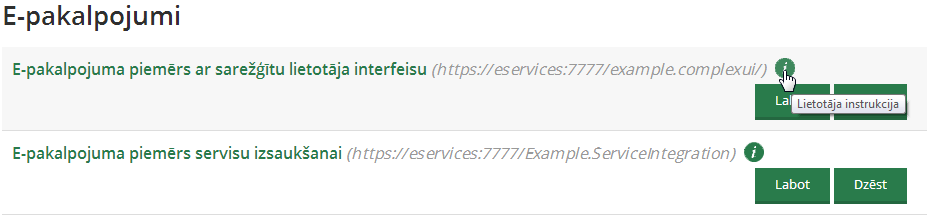


96.attēls. E-pakalpojuma neapstrādātas kļūdas lapa

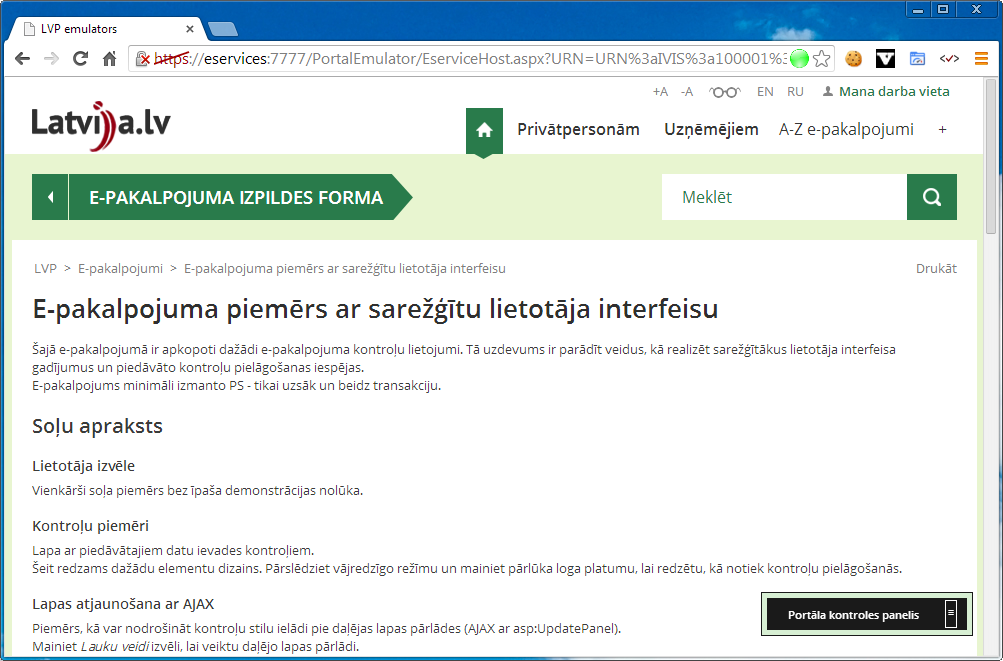
Gadījumā, ja e-pakalpojumā notiek neapstrādāta kļūda, lietotājs tiek pārsūtīts uz portāla kļūdas lapu. Izstrādes vidē gan pēc noklusētās e-pakalpojuma konfigurācijas šāda pāreja notiek tikai gadījumos, kad piekļuvi e-pakalpojumam veic no ārēja datora (<customErrors mode="RemoteOnly">).

## E-pakalpojuma lietotāja instrukcijas lapa

Noklikšķinot informācijas ikonu e-pakalpojumu sarakstā portāla sākuma lapā, tiek ielādēta e-pakalpojuma lietotāja instrukcijas lapa. Šī lapa vienmēr būs pieejama arī neautentificētiem lietotājiem.



97.attēls. Lietotāja instrukcijas atvēršana



98.attēls. Emulatorā atvērta e-pakalpojuma lietotāja instrukcija

# Biežāk sastopamās problēmas un to risinājumi

49.tabula

Žurnalēšanas ierakstu apskate

|  |  |
| --- | --- |
| Apraksts | E-pakalpojuma problēmu risināšanā parasti ļoti vērtīga ir žurnālos saglabātā informācija. Žurnalējamās informācijas izvade tiek konfigurēta ar e-pakalpojumu, maršrutēšanas servisa, kā arī emulatora *web.config* datņu *system.diagnostics* sadaļu.  Nākamajās tabulas rindiņās doti noderīgāko žurnālu apraksti. |
| E-pakalpojuma žurnāls  (EP123.txt) | Glabā ierakstus par e-pakalpojumā veiktajām darbībām (piemēram, PS izsaukumi), kā arī neapstrādātajiem (unhandled) izņēmumiem. |
| Maršrutēšanas servisa ziņojumu žurnāls  (Router.web\_messages.svclog) | Satur visu maršrutēšanas servisa saņemto un sūtīto ziņojumu pilnu saturu.  Pēc noklusējuma ziņojumi, kuru izmērs pārsniedz 256KB netiek saglabāti žurnālā. Izmēra ierobežojumu var mainīt ar system.serviceModel/diagnostics/messageLogging/maxSizeOfMessageToLog atribūta vērtību.    99.attēls. Servisu ziņojumu žurnāla fragments  Attēlā redzams piemērs, kā žurnālā izskatās PS transakcijas uzsākšanas pieprasījums:   1. <http://ivis.eps.gov.lv/ISS/RequestServiss/v1-0/StartTransaction>: Ziņojums, ko maršrutēšanas serviss saņēma no e-pakalpojuma transakcijas uzsākšanai. 2. <http://docs.oasis-open.org/ws-sx/ws-trust/200512/RST/Issue>: Ziņojums, ko maršrutēšanas serviss nosūtīja uz STS, lai lietotāja drošības talonu apmainītu pret PS izsaukšanai nepieciešamo talonu. 3. <http://docs.oasis-open.org/ws-sx/ws-trust/200512/RSTRC/IssueFinal>: STS atbildes ziņojums. 4. <http://ivis.eps.gov.lv/ISS/RequestServiss/v1-0/StartTransaction>: Ziņojums, ko maršrutēšanas serviss nosūtīja PS. 5. <http://ivis.eps.gov.lv/ISS/RequestServiss/v1-0/StartTransactionResponse>: PS atbildes ziņojums. 6. <http://ivis.eps.gov.lv/ISS/RequestServiss/v1-0/StartTransactionResponse>: Ziņojums, ko maršrutēšanas serviss nosūtīja kā atbildi e-pakalpojumam.   Sarkanajā rāmī iezīmēts 4.ziņojuma saturs (*body*). |
| Emulatora autentifikācijas žurnāls  (Emulator\WIFTrace.svclog) | Šajā žurnālā var ieraudzīt ar lietotāja autentifikāciju emulatorā saistīto informāciju. Visai noderīga informācija ir lietotājam izsniegtā drošības talona saturs (tiek fiksēts, ja *web.config* *system.diagnostics/sources/source[@name=’ Microsoft.IdentityModel’]/switchValue* atribūtā norādīts „*Warning, Information*”).    100.attēls. Emulatora autentifikācijas žurnāla fragments |

50.tabula

Izņēmums „There was no endpoint listening at …”

|  |  |
| --- | --- |
| Pazīmes | Mēģinot izsaukt kādu no servisiem notiek kļūda:  There was no endpoint listening at … that could accept the message. This is often caused by an incorrect address or SOAP action. See InnerException, if present, for more details.  Exception Details:  ystem.ServiceModel.FaultException`1[[System.ServiceModel.ExceptionDetail, System.ServiceModel, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089]]: There was no endpoint listening at … that could accept the message. This is often caused by an incorrect address or SOAP action. See InnerException, if present, for more details. |
| Cēlonis | Izstrādātāja mašīnai nav piešķirtas piekļuves tiesības kādai servisus izmitinošai videi.  Var gadīties arī situācijas, ka serviss nedarbojas (ir „nobrucis”) izmitinošajā vidē. |
| Novēršana | Jāpārliecinās, ka ar pārlūku var atvērt tīmekļa servisu adreses, kas atrodamas maršrutēšanas servisa konfigurācijas datnē (Web.config). Ja tiek izmantots *proxy*, jāpārliecinās, ka pārlūka *proxy* iestatījumi ir tādi paši, kā .net klienta izmantotie.  Ja nevar atvērt kādu no adresēm:  <https://epakvisstv.vraa.gov.lv/STS/VISS.Pfas.STS/STS/Issue.svc>  <https://ausmatest8.vraa.gov.lv/Request.WebService/v1-8/RequestWcfService.svc>  <https://ausmatest8.vraa.gov.lv/EDK/Logic/NavigationService.svc>  <https://ausmatest8.vraa.gov.lv/EDK/Logic/ObjectService.svc>  <https://ausmatest8.vraa.gov.lv/EDK/Logic/MultiFilingService.svc>  <https://ausmatest8.vraa.gov.lv/VISS.UserProfile.Service/UserProfileService.svc>  jāpieprasa pieejas tiesības EPU (VRAA) testa videi (iesniegums EPU).  Ja nevar atvērt kādu no adresēm:  <https://signetest.vraa.gov.lv/VRAA.DIV.Services.DivDocuments.Endpoint.WCFX.509/DivDocumentsWS.svc>  <https://signetest.vraa.gov.lv/VRAA.DIV.Services.EdocBrowser.Endpoint/EdocBrowserWS.svc>  jāpārliecinās, ka ir uzstādīts e-parakstīšanas sertifikāts (skat. 4.4.2.2.paragrāfa pēdējo punktu). Ja sertifikāts ir uzstādīts korekti, jāpieprasa pieejas tiesības EPU testa videi (iesniegums EPU).  Ja nevar atvērt adresi:  <https://eservices:7777/PortalEmulator/Services/EserviceConfigurationService.svc>  jāpārliecinās, ka emulators ir korekti uzstādīts. |

51.tabula

Izņēmums „The HTTP request was forbidden…”

|  |  |
| --- | --- |
| Pazīmes | Mēģinot izsaukt kādu no servisiem, notiek kļūda:  The HTTP request was forbidden with client authentication scheme 'Anonymous'.  Exception Details: System.ServiceModel.FaultException`1[[System.ServiceModel.ExceptionDetail, System.ServiceModel, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089]]: The HTTP request was forbidden with client authentication scheme 'Anonymous'. |
| Novēršana | Skatīt 50.tabula. |

52.tabula

Izņēmums „Keyset does not exist”

|  |  |
| --- | --- |
| Pazīmes | Atverot e-pakalpojumu, ar autentificētu lietotāju pie pirmā PS izsaukuma notiek neapstrādāta kļūda:  Keyset does not exist  Description: An unhandled exception occurred during the execution of the current web request. Please review the stack trace for more information about the error and where it originated in the code.  Exception Details: System.Security.Cryptography.CryptographicException: Keyset does not exist |
| Cēlonis | Maršrutēšanas servisa lietojuma pūla lietotājiem nav tiesības lasīt maršrutēšanas servisa sertifikāta privāto atslēgu vai arī sertifikāts vispār nav uzstādīts. |
| Novēršana | Jāpiešķir pūla lietotājiem tiesības lasīt sertifikāta privāto atslēgu (skat. 4.4.2.2.paragrāfa noslēgumu). |

53.tabula

Kļūda atverot SSO pieslēgšanās lapu

|  |  |
| --- | --- |
| Pazīmes | Atverot SSO pieslēgšanās lapu, parādās kļūda:    101.attēls. Kļūda, atverot autentifikācijas lapu |
| Novēršana | Jāiztīra pārlūka sīkfaili (*cookies*).  Ja tas nepalīdz, iespējams, ka e-pakalpojumā norādītais *realm* (*microsoft.identityModel* sekcija*)* nav reģistrēts STS. Jāsazinās ar EPU, lai pārbaudītu aizdomas. |

54.tabula

Kļūda „invalid method (HTTP verb) is being used”

|  |  |
| --- | --- |
| Pazīmes | Atverot e-pakalpojumu ar autentificētu lietotāju (vai arī pēc autentifikācijas veikšanas e-pakalpojuma logā), parādās IIS kļūda:  HTTP Error 405.0 - Method Not Allowed  The page you are looking for cannot be displayed because an invalid method (HTTP verb) is being used. |
| Cēlonis | SSO STS konfigurācijā norādīts nekorekts e-pakalpojuma URL. URL obligāti jābeidzas ar „Public/Lvp/Wrapper” (jāatver nevis sākuma lapa (Default.aspx), bet gan apdarinātājs (skat. 7.2.30.sadaļu)).  Piemēram, ja e-pakalpojums izvietots adresē:  *https://eservices:7777/Example.Infrastructure*  STS konfigurācijā jābūt norādītai adresei:  *https://eservices:7777/Example.Infrastructure/Public/Lvp/Wrapper* |
| Novēršana | Jāsazinās ar EPU, lai mainītu STS konfigurāciju, norādot korektu URL. |

55.tabula

Emulators vienmēr prasa pieslēgties

|  |  |
| --- | --- |
| Pazīmes | Atverot emulatora lapu, uzreiz notiek automātiska pārsūtīšana (redirect) uz SSO pieslēgšanās lapu. |
| Cēlonis | Iespējams, emulatora lietojumam nav atļauta anonīma piekļuve. |
| Novēršana | Jāatļauj anonīmā piekļuve:   * Jāatver IIS Manager. * Jāizvēlas lietojums. * Jāatver lietojuma autentifikācijas iestatījumi (Authentication).     102.attēls. Autentifikācijas iestatījumu atvēršana   * Jāpārliecinās, ka ir iespējota anonīma piekļuve:     103.attēls. Autentifikācijas iespēju saraksts |

56.tabula

Nedarbojas e-pakalpojuma iframe automātiskā augstuma maiņa

|  |  |
| --- | --- |
| Pazīmes | E-pakalpojuma kreisajā malā redzama ritjosla – e-pakalpojums nav redzams visā augstumā. Otrs variants – e-pakalpojuma apakšējā daļa nav redzama (tiek „nogriezta”). |
| Cēlonis | E-pakalpojuma body *tag* izmērs tiek izrēķināts mazāks nekā tā satura kopējais izmērs. |
| Novēršana | Jāpārliecinās, ka:   * e-pakalpojuma ielādes laikā nenotiek JavaScript kļūdas (tās var novest pie skriptu izpildes apturēšanas, kā rezultātā augstuma maiņas skripts netiek izpildīts). * e-pakalpojuma *body tag* augstums ir uzstādīts kā auto (problēmu parasti rada vērtība 100%). * gan *body*, gan html tagiem *overflow-y* uzstādīts kā *hidden*. |

57.tabula

Pēc UpdatePanel pārlādes pazūd kontroļu stili

|  |  |
| --- | --- |
| Pazīmes | Pēc UpdatePanel pārlādes pazūd LVP raksturīgais kontroļu izskats. |
| Cēlonis | Stilu veidošanas *JavaScript* funkciju nepieciešams izsaukt katru reizi, kad lapā parādās jauni HTML datu ievades elementi. |
| Novēršana | Pie lapas ielādes uz UpdatePanel ielādes notikumu (Sys.WebForms.PageRequestManager.getInstance().add\_endRequest) jāizsauc lvp.controlStyle.applyStyle(); *JavaScript* funkcija.  Jāpārliecinās, ka add\_endRequest notikumam e-pakalpojumā tiek piesaistīta tikai viena funkcija (vairāku piesaistījumu gadījumā visticamāk tiks izpildīta tikai pēdējā no piesaistītajām funkcijām).  Piemērs šai funkcionalitātei atrodams *ComplexUI* e-pakalpojumu piemēra 3.solī. |

58.tabula

Nepieciešams apmainīties ar datiem starp e-pakalpojuma soļiem

|  |  |
| --- | --- |
| Pazīmes | Nepieciešams apmainīties ar datiem starp e-pakalpojuma soļiem. |
| Risinājums | LvpWizardStepBase piedāvā FindStep<TStepType> funkciju, kas ļauj viegli atrast soli pēc tā tipa un piekļūt soļa īpašībām.  Izmantošanas piemērs atrodams *ServiceIntegration* e-pakalpojumu piemēra 3. solī. |

59.tabula

No PS tiek saņemta kļūda

|  |  |
| --- | --- |
| Pazīmes | No PS tiek saņemta kļūda ErrorList struktūrā (tiek izraisīts RequestServiceException). |
| Cēlonis | Saņemtās kļūdas var iedalīt divās daļās:   * Kļūdas, kuras izraisījis PS. Šajā gadījumā bieži vien uz PS no e-pakalpojuma vai integrācijas servisa tiek sūtīti dati, kas neatbilst XSD shēmai vai arī VISS ir nekorekta integrācijas servisu konfigurācija. Šādas kļūdas var atpazīt pēc fragmenta „-REQ” kļūdas kodā (piemēram, „URN:IVIS:100001:ISS-EPS-REQ-v1-8-ER-16”). * Kļūdas, kuras izraisījis integrācijas serviss (PS tās vienkārši pārsūta tālāk). Šajā gadījumā kļūdas cēloņi ir specifiski izmantotajam integrācijas servisam.   Saņemtās kļūdas struktūras aprakstu skatīt 7.1.11. un 7.1.12.sadaļās.  Zemāk uzskaitīti PS kļūdu numuri un apraksti. Parasti apraksti tiek iekļauti arī saņemtajā kļūdas struktūrā. Kļūdas struktūrā minētais apraksts ir uzticamāks nekā zemāk minētais, turklāt tajā var būt papildus informācija, piemēram, servisu, transakciju u.c. identifikatori.   * 0 (atbilst, piemēram, kodam URN:IVIS:100001:ISS-EPS-REQ-v1-8-ER-0): Nav norādīts *MessageID*. * 1: Nav uzdots korelācijas numurs. * 2: Atbilde ar norādīto *CorrelationID* neeksistē vai ir nekorekti uzdota. * 3: Nepareizs transakcijas identifikators. * 4: Transakcija '{0}' nav uzsākta. * 5: *EmployeePoll* nevar būt *ivisRequest* sūtītājs. * 6: *Destination* nav norādīts. * 7: Nepareizi norādīts *Destination* vai *ResponseEndPoint*. Jābūt UNT:IVIS ISService vai *IDDVForm* identifikatoram. * 8: Norādītais *Destination* neeksistē. * 9: ISServiss nav reģistrēts kā integrācijas Serviss. * 10: *Destination* nav reģistrēts ISServisu katalogā kā serviss. * 11: Lietotāja tiesības ir pārāk mazas, lai izsauktu ISServisu. * 12: ISServisu katalogā nav atrasts *Destination* URL. * 13: ISServisu katalogā nav atrasts *Destinatoin* testa URL. * 14: Uzstādīts *Body*, bet nav uzstādīts *MessageType*. * 15: Uzstādīts *MessageType*, bet nav uzstādīts *Body*. * 16: *Body* serializācijas kļūda. * 17: Ziņojuma kopējais izmērs pārsniedz atļauto garumu. * 18: *MessageType* nav URN:IVIS shēmas tipa identifikators. * 19: Ziņojuma *Body* elements neatbilst *MessageType* norādītai shēmai. * 20: XML shēma XML shēmu katalogā neeksistē. * 21: Nepareizs pieturpunkta identifikators. * 22: Tāds *Milestone* identifikators nav reģistrēts. * 23: Atbildes *correlationID* tiek aizvietots ar pieprasījuma *correlationID*. * 24: Atbildes *transactionID* tiek aizvietots ar pieprasījuma *transactionID*. * 25: Tika noģenerēts jauns *messageID*. * 26: Tika noģenerēts jauns *corellationID*. * 27: IS Servisam jābūt uzdotam *realm* atribūtam. * 100: Pieprasījums vai pieprasījuma *Header* struktūra ir tukši. * 101: Atbilde vai atbildes *Header* struktūra ir tukši. * 102: Kļūda, saglabājot pieprasījumu datubāzē. * 103: Kļūda, saglabājot atbildi datubāzē. * 104: Kļūda, saglabājot statusu datubāzē. * 105: Kļūda, izsaucot ISServisu vai IDDV formu. * 106: Kļūda, atjaunojot sūtītāju. * 107: Kļūda, atjaunojot pieturpunktu. * 108: Lietotājam nav tiesību izpildīt operāciju. |
| Novēršana | Jānovērš problēmas, par kurām informē kļūda. Piemēram, pārsūtāmās datu struktūras jāaizpilda tā, lai tās atbilstu XSD shēmai, jālabo XSD shēma, VISS jāmaina izsaucamo integrācijas servisu adrese, jāizpilda integrācijas servisa specifiskie nosacījumi vai kāda cita darbība atbilstoši kļūdai. |

60.tabula

No PS tiek saņemta kļūda #19 „IvisRequest pieprasījuma Body elements neatbilst MessageType laukā norādītajai shēmai”

|  |  |
| --- | --- |
| Pazīmes | No PS tiek saņemta kļūda ErrorList struktūrā (tiek izraisīts RequestServiceException), kas satur ziņojumu „IvisRequest pieprasījuma Body elements neatbilst *MessageType* laukā norādītajai shēmai”. |
| Cēlonis | Iespējams e-pakalpojuma konfigurācijā nekorekti norādīta *lvp/messages/message/messageType* elementa vērtība. |
| Novēršana | Jāpārbauda *messageType* atribūta vērtība. Tai jābūt norādītai formātā „*XSD:shēmas:urn****-TYPE-****shēmasElements*”.  Bieži vien problēmas rada *shēmasElementa* sadaļa. Tajā ir jānorāda nevis izmantotā datu tipa nosaukums (<xs:complexType name=”…”>), bet gan XSD shēmas elementa ar atbilstošo tipu nosaukums (<xs:element name=”…”>).  Piemēram šādai shēmai, kas resursu katalogā ir reģistrēta ar URN „*URN:IVIS:100001:XSD-Testing-TestISServise-v1-0*”:  <xs:schema>    <xs:element name="Calculation" type="CalculationStructure">      <xs:annotation>        <xs:documentation xml:lang="lv">Ieejas elementi</xs:documentation>      </xs:annotation>    </xs:element>    ...    <xs:complexType name="CalculationStructure">      <xs:sequence>        ...      </xs:sequence>    </xs:complexType>    ...  </xs:schema>  e-pakalpojuma konfigurācijā *messageType* ir jānorāda kā „*URN:IVIS:100001:XSD-Testing-TestISServise-v1-0-TYPE-****Calculation***” (nevis „URN:IVIS:100001:XSD-Testing-TestISServise-v1-0*-TYPE-* *CalculationStructure*”). |

61.tabula

Kļūda „Could not establish trust relationship for the SSL/TLS secure channel with authority '…'.”

|  |  |
| --- | --- |
| Pazīmes | Mēģinot izsaukt maršrutēšanas servisu caur https savienojumu, tiek atgriezta kļūda „Could not establish trust relationship for the SSL/TLS secure channel with authority '…'.” No maršrutēšanas servisa tālāki izsaukumi (uz PS, lietotāja profila u.c. servisiem) netiek veikti. |
| Cēlonis | Maršrutēšanas servisa aplikācijas IIS norādītais SSL sertifikāts nav korekts (neatbilst aplikācijas adresei vai nav izdots no uzticamas sertificēšanas iestādes (CA)). |
| Novēršana | Ja maršrutēšanas serviss ir reģistrēts noklusētajā adresē (<https://eservices:7777/>...), jāpārliecinās, ka izsaucēja datorā pie *Trusted Root Certification Authorities* ir importēts abc-W2008EVAL-CA sertifikāts (eservices\_Root\_CA.cer) – skat. 4.4.1.2.paragrāfu.  Ja maršrutēšanas serviss ir uzstādīts kādā citā adresē, izstrādātājiem pašiem jāģenerē SSL sertifikāts izmantotajai adresei un jāveic nepieciešamā konfigurācija, lai sertifikāts būtu uzticams.  Ja maršrutēšanas serviss ir uzstādīts tajā pašā datorā, kur izsaucējs (e-pakalpojums), var izmantot *net.pipe* protokolu. *Net*.*pipe* protokols neizmanto sertifikātus. Lai pārslēgtos uz *net*.*pipe*, e-pakalpojuma konfigurācijā *lvp/general/routerServiceUrl* https:// daļa jāaizstāj ar net.pipe:// . |

62.tabula

E-pakalpojumā nav pieejams lietotāja drošības talons

|  |  |
| --- | --- |
| Pazīmes | Lietojot e-pakalpojumu ar autentificētu lietotāju, notiek neapstrādāta kļūda ArgumentNullException ar ziņojumu „Parameter name: user.Token”. |
| Cēlonis | Konfigurācijā nav norādīta pazīme, ka e-pakalpojumam ir jāglabā sākotnēji saņemtais lietotāja talons. |
| Novēršana | E-pakalpojuma *web*.*config* datnē *microsoft.identityModel/service* sekcijas atribūtam *saveBootstrapTokens* jāuzstāda vērtība true.  Pēc tam jāatslēdzas (log-out) un jāpieslēdzas (log-in) e-pakalpojumā no jauna. |

63.tabula

E-pakalpojumu vides uzstādīšanas skripta kļūda

|  |  |
| --- | --- |
| Pazīmes | Skripta izpilde beidzas ar izņēmumu: „Process should have elevated statuss to …”. |
| Cēlonis | Skripta izpildei nav nodrošinātas pietiekamas lietotāja tiesības. |
| Novēršana | Skripts jāpalaiž ar administratora tiesībām (labais klikšķis uz \*.bat datni -> Run as administrator). |

1. Izņēmuma situācijā tā varētu būt arī e-pakalpojuma lapa (tad URL var norādīt arī citus parametrus, kas identificē e-pakalpojuma transakciju), bet par pakalpojuma sesijas atjaunošanu ir jārūpējas pašam pakalpojumam. [↑](#footnote-ref-1)