

**PAR VALSTS INFORMĀCIJAS SISTĒMU SAVIETOTĀJA,
LATVIJAS VALSTS PORTĀLA WWW.LATVIJA.LV UN
ELEKTRONISKO PAKALPOJUMU IZSTRĀDE UN
UZTURĒŠANA
E-PAKALPOJUMU IZSTRĀDE - 2020.GADA PLATFORMA**

VADLĪNIJAS

VRAA-13_7_17_41-VISS_2016-EPAK3-VDL

07.01.2021. versija 1.02

Rīgā 2020

Satura rādītājs

ATTĒLU SARAKSTS	3
1. IEVADS.....	4
1.1. Dokumenta nolūks	4
1.2. Termini un pieņemtie apzīmējumi	4
1.2.1. Termini	4
1.2.2. Apzīmējumi.....	5
1.3. Saistība ar citiem dokumentiem	6
1.4. Dokumenta pārskats	7
2. E-PAKALPOJUMA JĒDZIENS	8
3. E-PAKALPOJUMU IZPILDES VIDES APRAKSTS	10
3.1. E-pakalpojuma izpildes loģika.....	12
3.2. Risinājuma sadalījums slāņos.....	13
4. E-PAKALPOJUMA IZSTRĀDES PROCESS	15
4.1. E-pakalpojuma prasību specifikācijas izveide	15
4.1.1. Posma nodevumi.....	16
4.2. Projektējums.....	16
4.2.1. JSON struktūras	16
4.2.2. IS servisu (API) programmatūras projektējuma apraksta izstrāde.....	17
4.2.3. Posma nodevumi.....	17
4.3. Izstrāde	17
4.3.1. IS servisi (API).....	17
4.3.1.1. IS servisu (API) izstrāde	17
4.3.1.2. IS servisu (API) testēšana	18
4.3.1.3. Reģistrācija API pārvaldniekā	18
4.3.2. E-pakalpojums.....	18
4.3.2.1. Izstrāde.....	18
4.3.2.2. Testēšana	18
4.3.2.3. Piegāde	19
4.3.2.4. E-pakalpojuma reģistrācija	19
4.3.3. Posma nodevumi.....	19
4.4. E-pakalpojuma publicēšana	19
4.4.1. E-pakalpojumu akceptēšanas nosacījumi (check-list).....	19
4.4.2. E-pakalpojuma integrēta testēšana	20
4.4.3. Posma nodevumi.....	20
5. PIELIKUMS.....	21
5.1. Metode “createDocument”	21

Attēlu saraksts

1.attēls. E-pakalpojumu “elektronizācijas līmeņu” diagramma.....	8
2.attēls. E-pakalpojuma sniegšanas scenārijs.....	9
3.attēls. E-pakalpojuma sastāvs un mijiedarbība ar LVP	13
4.attēls. Risinājuma slāņu arhitektūra	14
5.attēls. E-pakalpojuma izstrādes process	15

1. Ievads

Publisko pakalpojumu elektroniskā piegāde būtiski paaugstina valsts institūciju, iedzīvotāju un komercuzņēmumu sadarbības efektivitāti. Veidojot jaunus e-pakalpojumus, valsts institūcijās tiek strādāts pie tā, lai uzlabotu komunikācijas iespējas ar iedzīvotājiem un komercuzņēmumiem. Tā rezultātā tiek veidoti valsts un pašvaldību portāli. Portāli nodrošina vienotus ieejas punktus informācijai un publiskiem pakalpojumiem, optimizējot šo punktu organizāciju atbilstoši lietotāju vajadzībām un neprasot papildu zināšanas par konkrētā pakalpojuma procedūru.

Parasti ikviena iestāde vai komercuzņēmums integrāciju vai e-pakalpojuma izveidi aplūko no savu interešu viedokļa, tomēr jāatzīmē, ka ir ļoti svarīgi, lai šie procesi nenotiktu haotiski, bet gan centralizēti. Valsts informācijas sistēmu savietotājs (VISS) nodrošina kopēju infrastruktūru integrācijas uzdevumu risināšanai un e-pakalpojumu izveidei un pilnveidošanai.

1.1. Dokumenta nolūks

Šī dokumenta mērķis ir aprakstīt vadlīnijas, kuras nosaka un ietekmē e-pakalpojumu izstrādes procesu, izmantojot VISS infrastruktūru, kā arī aprakstīt e-pakalpojumu realizācijas tehniskos aspektus bez konkrētas tehnoloģijas pielietojuma. Dokuments nav paredzēts likumdošanas vai organizatorisko pasākumu aprakstīšanai e-pakalpojumu izstrādes un ieviešanas gaitā.

Šī dokumenta auditoriju veido:

1. VRAA darbinieki, kuri ir atbildīgi par VISS nodevumu un e-pakalpojumu pieņemšanu un izvērtēšanu;
2. E-pakalpojumu portāli, pašvaldību un reģistru mājaslapu uzturētāji;
3. E-pakalpojumu attīstītāji, t.sk., arī:
 - projektu vadītāji un analītiķi – speciālisti, kas nodrošina veidojamā e-pakalpojuma biznesa procesu analīzi un aprakstu, veido sākotnējo e-pakalpojuma specifikāciju, JSON struktūru projektēšanu, balstoties uz šo dokumentu, nodrošina sadarbību starp visiem iesaistīto iestāžu darbiniekiem un VISS uzturētājiem un attīstītājiem;
 - valsts reģistru pārstāvji (lietojumapgabala eksperti) – iestāžu darbinieki, konkrētās iestādes tehniskie un citi speciālisti;
 - izstrādātāji – tehniskie speciālisti, IS servisu, integrācijas procesu un e-pakalpojumu prezentācijas slāņa izstrādātāji.
4. VISS administratori.

Paredzēts, ka šie norādījumi tiks pastāvīgi papildināti ar jauniem ieteikumiem, t.sk., arī no dokumenta lasītāju puses.

1.2. Terminu un pieņemtie apzīmējumi

1.2.1. Terminu

Dokumentā izmantotie termini ir apkopoti 1.tabulā.

1.tabula

Termini

TERMINS	APRAKSTS
E-pakalpojums	E-pakalpojums pakalpojuma sniegšanas veids, kas nodrošina iespēju visus vai atsevišķus pakalpojuma saņemšanas soļus izpildīt attālināti (neklātienē), izmantojot IKT līdzekļus. Katra e-pakalpojuma instance tiek uzsākta un pabeigta vienai konkrētai biznesa situācijas risināšanai (soda apmaksa, izziņas saņemšana, elektroniskās pilnvaru izdošana utt.).
E-pakalpojuma sniegšanas punkts	E-pakalpojumu ieejas punkts, kas ietver, bet neaprobežojas ar: <ul style="list-style-type: none"> • Valsts portālu; • VISS portāls; • pašvaldību mājaslapas; • banku e-banku lietotnes; • privātas un komersantu mājaslapas un portāli.
Web Serviss vai IS serviss	Web Servisi ir apzīmējums tādām lietojumprogrammām, kuras ir paredzētas izsaukumiem no citu lietojumprogrammu puses un kuras saņem datus un sniedz atbildes uz tiem standartizētā formātā (XML, JSON utt.), parasti izmantojot SOAP, REST vai citu tam paredzētu protokolu. Web Servisi ir uz servisiem orientētas arhitektūras (SOA) sastāvdaļu konkrēta realizācija (instance).
Drošības talonu serviss (Security Token Service)	Timekļa serviss, kas izsniedz, atjaunina un dzēš drošības talonus. Drošības talons parasti tiek izsniegts, atbildē uz lietotāja identifikācijas faktu pret pareizo lietotāja vārda un paroles kombināciju vai citu autentifikācijas līdzekli (piemēram, banku internet banku autentifikāciju vai eID). Drošības talons parasti nodrošina lietotāja identitātes apstiprināšanu (autentifikācijas apgalvojumu) vai lietotāja tiesību pierādīšanu (autorizācijas apgalvojums).
Integrācijas IS serviss	Standartizēts informācijas sistēmas pieejas veids (serviss), kas pie informācijas resursa nodrošina ārējo saskarni. Integrācijas IS serviss parasti sastāv no divām daļām: servera daļas un klienta daļas. Servera daļa – IS servisa realizācija iestādes pusē (realizācijas veida izvēle ir iestādes kompetencē, saskarnes aprakstīšanas vēlamais standarts ir WSDL); klienta daļa – standartizētas IS servisa saskarnes realizācija VISS infrastruktūras pusē (IVISRequest/IVISResponse).
Uz servisiem orientēta arhitektūra	Uz servisiem orientēta arhitektūra sastāv no virknes brīvi savienotu, sadalītu, biznesa komponentu, kuras kopā var ietilpt vienā ar e-pakalpojuma sniedzēju saistītā procesā.
JSON datu formāts	Atvieglotais datu apmaiņas standartizēts formāts, kuru cilvēkam viegli rakstīt un lasīt, kā arī datortehnikai viegli pārsēt un ģenerēt.
JSON dokuments	JSON dokuments ir informācijas resurss, kuru apraksta lietojuma/json media tips (CBOR – <i>Concise Binary Object Representation</i>). JSON shēmā, tādi termini kā "JSON dokuments", "JSON teksts" un "JSON vērtība" savstarpēji aizvietojami, jo to nosaka datu modelis.

1.2.2. Apzīmējumi

Dokumentā ir izmantotie apzīmējumi ir apkopoti 2.tabulā.

2.tabula

Apzīmējumi

APZĪMĒJUMS	APRAKSTS
API	Lietojumprogrammu saskarne (<i>Application Programming Interface</i>)
BPEL	<i>Business Process Execution Language</i> integrācijas maršruti
DAIRM	VISS (IVIS) Darbību audita ierakstu reģistrēšanas modulis
DB	Datu bāze (<i>Database</i>)

APZĪMĒJUMS	APRAKSTS
DIT	Datu izplatīšanas tīkls
DIV	Dokumentu integrācijas vide
FAQ	Saraksts "Bieži uzdotie jautājumi" (<i>Frequently asked questions</i>)
HTML	Hiperteksta iezīmēšanas valoda (<i>Hypertext markup language</i>)
IDDV	Iestādes darbinieka darba vieta
IS	Informācijas sistēma
JSON	<i>JavaScript Object Notation</i>
LV	Latvijas Valsts
LVP	Latvijas valsts portāls
PFAS AUTH	Autentifikācijas un autorizācijas modulis
PMLP	Pilsonības un migrācijas lietu pārvalde
PPA	Programmatūras projektējuma apraksts
PPK	Publisko pakalpojumu katalogs
SLA	Pakalpojumu līmeņa vienošanās (<i>Service Level Agreement</i>)
SOA	Servisorientēta arhitektūra (<i>Service Oriented Architecture</i>)
STS	Skat. Drošības talonu serviss
UDDI	<i>Universal Description Discovery and Integration</i>
VISS	(Bij. IVIS – Integrētā valsts informācijas sistēma) Valsts informācijas sistēmu savietotājs
VPA	Vienas pieturas aģentūra
VRAA	(Bij. ĪUMPELS – Īpašu uzdevumu ministra e-pārvaldes lietās sekretariāts) Valsts reģionālās attīstības aģentūra
WS	Web Serviss (<i>Web Service</i>)
WSDL	Tīmekļa servisu pakalpojumu aprakstīšanas valoda (<i>Web Service Definition Language</i>)
XML	Paplašināmās iezīmēšanas valoda (<i>eXtensible Markup Language</i>)
XSD	XML shēma (<i>XML Schema Definition</i>)
XSL	Izvērtējamā stila lapu valoda (<i>Extensible Stylesheet Language</i>), skat. XSLT
XSLT	Paplašināmās stila lapas pārveidošanas valoda (<i>Extensible Stylesheet Language Transformation</i>)
IDS	LVP Identity Server komponente – drošības talonu serviss.

1.3. Saistība ar citiem dokumentiem

Dokuments ir izstrādāts, balstoties uz šādiem dokumentiem:

- [1] E-pakalpojuma izveidošanas ceļvedis Latvija.lv portālam. Programmētāja rokasgrāmata. (VRAA-13_7_17_41-VISS_2016-LVP_WZD_3-PR).
- [2] Standarts LVS 72:1996, leteicamā prakse programmatūras projektējuma aprakstīšanai.
- [3] Valsts informācijas sistēmu savietotāja (VISS) un Vienotā valsts un pašvaldību pakalpojumu portāla www.latvija.lv pilnveidošana un uzturēšana. Latvija.lv portāls. Dizaina vadlīnijas. (VRAA-13_7_19_297-VISS_2016-LVP-DV).
- [4] E-Pakalpojumu arhitektūras izstrāde. Vadlīnijas. (VRAA-13_7_17_41-VISS_2016-EPAK_ARH_3-VDL).

- [5] Standarts LVS 68:1996, Programmatūras prasību specifikācijas (PPS) ceļvedis.
- [6] Standarts LVS testa plāniem, piemēriem.
- [7] Personas koda maiņas realizācija e-pakalpojumos. Rekomendācijas. (VRAA-13_7_14_50-VISS_2014-PKM_EPAK-REK 15.06.2017. versija 1.02).
- [8] Datu apmaiņas izveides vadlīnijas. Vadlīnijas. (VRAA-13_7_17_41-VISS_2016-D_APM-VDL).
- [9] JSON shēmas specifikācija: <https://json-schema.org/>
- [10] JWT standarts, <https://jwt.io>
- [11] OpenApi standarts, <https://swagger.io/specification/>
- [12] Valsts informācijas sistēmu savietotāja (VISS) un Vienotā valsts un pašvaldību pakalpojumu portāla www.latvija.lv pilnveidošana un uzturēšana. XML resursu izstrāde. Vadlīnijas. (VRAA-13_7_17_41-VISS_2016-XML-VDL).
- [13] Valsts informācijas sistēmu savietotāja, Latvijas valsts portāla www.latvija.lv un elektronisko pakalpojumu izstrāde un uzturēšana. E-pakalpojumu katalogs. Administratora rokasgrāmata. (VRAA-6_15_11_58-VISS_2016-EC-AR).
- [14] Valsts informācijas sistēmu savietotāja (VISS) un Vienotā valsts un pašvaldību pakalpojumu portāla www.latvija.lv pilnveidošana un uzturēšana. PFAS AUTH lietotne. Lietotāja rokasgrāmata. (VRAA-6_15_11_58-VISS_2016-PFAS_AUTH-LR)

1.4. Dokumenta pārskats

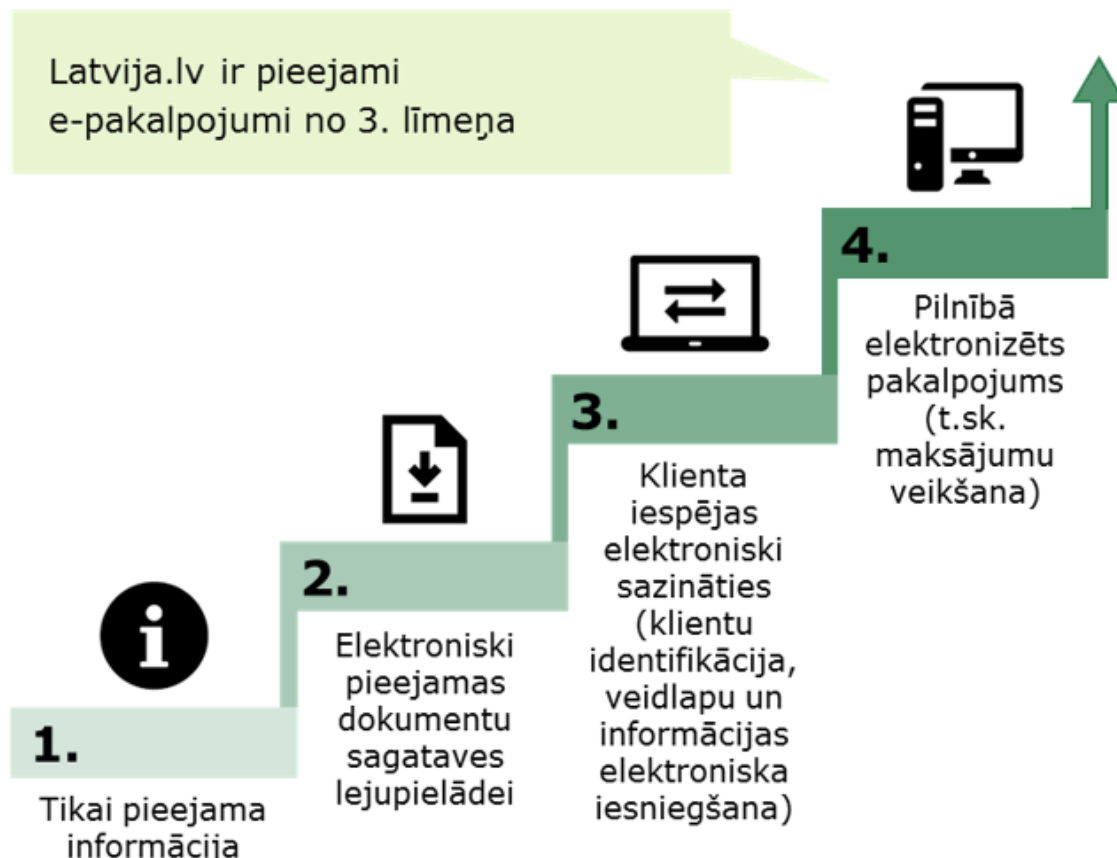
Dokumentu veido 4 nodaļumi:

- 1.nodaļumā – “Ievads” – aprakstīta dokumenta kopējā struktūra, nolūks, izmantotie termini un apzīmējumi, kā arī saistība ar citiem dokumentiem.
- 2.nodaļumā – “E-pakalpojuma jēdziens ” - aprakstīts e-pakalpojuma jēdziens un būtība.
- 3.nodaļumā – “E-pakalpojumu izpildes vides apraksts” – aprakstīta VISS infrastruktūra.
- 4.nodaļumā – “E-pakalpojuma izstrādes process” - aprakstīts e-pakalpojuma izstrādes process.

2. E-pakalpojuma jēdziens

Saskaņā ar e-pakalpojumu definīciju (skat. 1.2.1.sadaļu), par e-pakalpojumu sauc pakalpojumu, kurš tiek sniegts tiešsaistē un kura sniegšanas procesa atbalstam tiek izmantotas IT nodrošinātās automatizācijas iespējas. Tiek izšķirti četri e-pakalpojumu “elektronizācijas līmeņi”:

1. līmenis – Informēšana – informācija par pakalpojumiem ir pieejama internetā.
2. līmenis – Mijiedarbība – veidlapu un dokumentu sagatavju lejupielāde.
3. līmenis – Divvirzienu mijiedarbība – klientu identifikācija, veidlapu un informācijas elektroniska iesniegšana papīra dokumentu vietā.
4. līmenis – Darījumu apstrāde – pakalpojuma pilna apstrāde, ieskaitot lēmuma pieņemšanu, informēšanu, maksājumu kārtošanu.



1.attēls. E-pakalpojumu “elektronizācijas līmeņu” diagramma

Ar e-pakalpojumu LVP vidē saprotam 3. vai 4. līmeņa e-pakalpojumu. No konceptuālā viedokļa e-pakalpojums ir tāds valsts pārvaldes sniegts pakalpojums iedzīvotājiem, uzņēmumiem, kā arī pašai pārvaldei, kura sniegšanai maksimāli tiek samazināta pakalpojuma izmantotāju fiziskās klātbūtnes nepieciešamība un kura sniegšanas procesa atbalstam maksimāli tiek izmantotas informācijas tehnoloģijas. Tipiskie e-pakalpojumu veidi atkarība no mērķauditorijas dalās šādi:

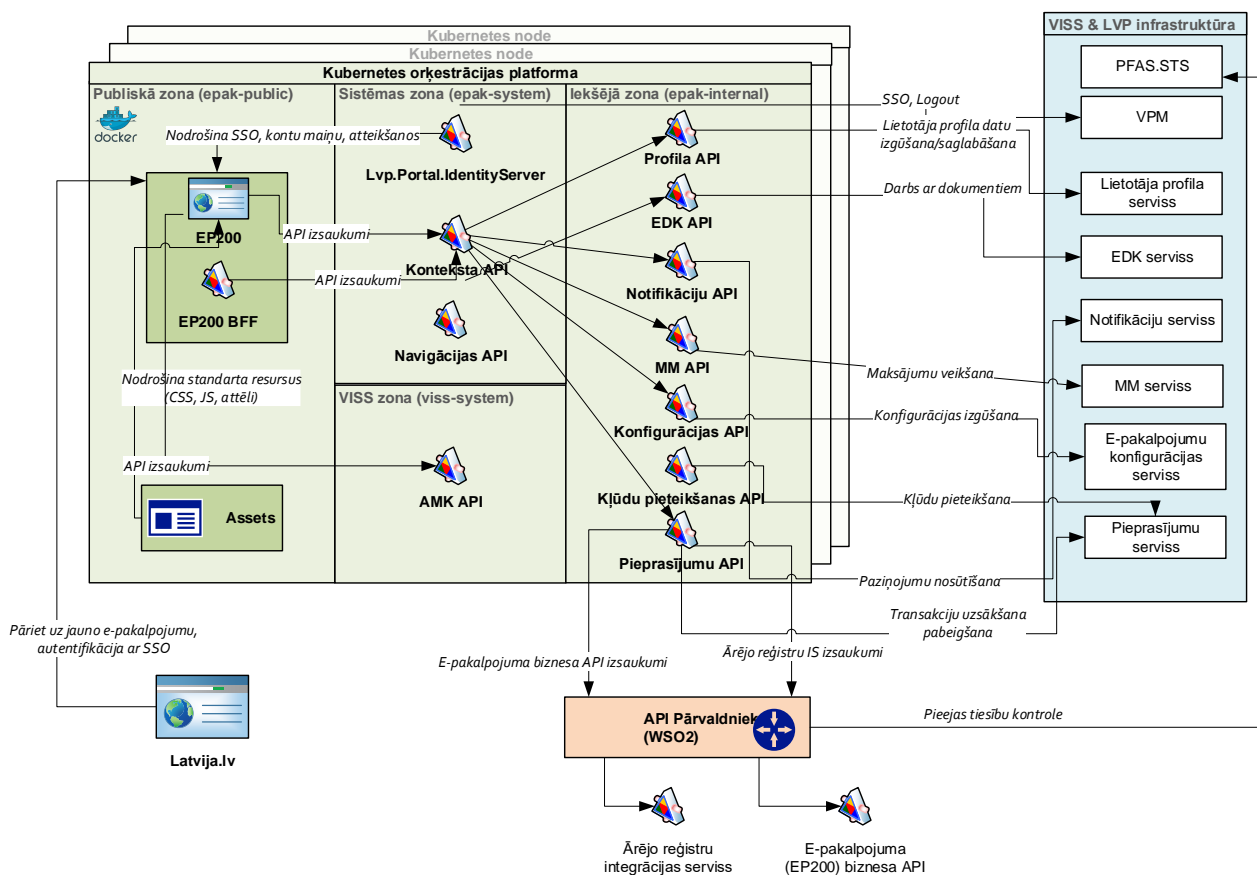
1. e-pakalpojumi iedzīvotājam un vai to pilnvarotajiem;
2. e-pakalpojumi komersantiem (juridiskām personām un/vai to pilnvarotajiem);

E-pakalpojums nevar eksistēt bez konkrētās iestādes publiski sniegtā pakalpojuma. Katram pakalpojumam vienmēr ir apraksts, kurš satur šādu informāciju:

1. pakalpojuma sniegšanas uzsākšanai nepieciešamo dokumentu un citu ieejas nosacījumu apraksts;

2. pakalpojuma soļu apraksts (procesa apraksts);
3. pakalpojuma rezultātu apraksts;
4. cita informācija saskaņā ar publiskā pakalpojuma sākotnējo aprakstu.

Sniedzot pakalpojumu klientam, jāveic darbības noteiktā secībā (biznesa process). Pakalpojumu procesā gandrīz vienmēr ir iesaistītas vairākas iestādes, reti sastopami pakalpojumi, kuru sniegšana notiek vienas iestādes ietvaros. Procesa izpildes gaitas vadība pāriet pie dažādiem iestāžu darbiniekiem, kā arī tiek izmantotas dažādas iestādēs esošās informācijas sistēmas. No tehniskā viedokļa e-pakalpojums ir dažādu servisu, prezentācijas formu, integrācijas un izpildes procesu kopa, kas darbojas saskaņoti un mērķtiecīgi ar nolūku sniegt pakalpojumu tā lietotājam (2.attēls).



2.attēls. E-pakalpojuma sniegšanas scenārijs

VISS arhitektūra nodrošina iespēju izmantot API pārvaldniekā publicētos servissus (API) neierobežotā e-pakalpojumu sniegšanas punktu skaitā, piemēram, Latvijas valsts portālā (LVP), pašvaldību mājaslapas utt. Visi e-pakalpojumu sniegšanas punkti tiek nodrošināti ar katra e-pakalpojuma aprakstu, lietošanas instrukcijām, aizmugursistēmas servisiem un pieeju pie API pārvaldniekā publicētajiem servisiem (API).

3. E-pakalpojumu izpildes vides apraksts

E-pakalpojumu platforma ir izveidota uz divu paaudžu tehnoloģiskā risinājuma pamata – 2014. un 2020. gada.

2014.gada pamatā būvētas LVP un VISS platformas pamatarhitektūra balstās uz šādiem principiem:

- ir izmantota uz servisiem orientēta arhitektūra (SOA);
- primārais sistēmu saskarnes veids ir SOAP Web servisi (Integrācijas IS servisi);
- e-pakalpojumu prezentācijas slāņa izstrāde notiek .NET vidē, bet servisi, kas apkalpo back end var būt izveidoti ar jebkādu tehnoloģiju;
- ir izmantoti vienoti standarti – XML shēmu izveides standarti, IS servisu izveides standarti, e-pakalpojumu izveides standarti.

2020.gada pamatā būvētas LVP platformas pamatarhitektūra balstās uz:

- mikroservisu risinājuma arhitektūras;
- atvērtajiem pasaules standartiem un ietvariem, neierobežojot izstrādātājus ar noteikto tehnoloģiju vai zināšanām e-pakalpojuma vizuālās saskarnes (frontend) un funkcionālajā (backend) izstrādē;
- primāro sistēmu saskarnes veidu, kas ir JSON REST, tiek nodrošināta iespēja arī saukt esošos SOAP integrācijas IS servissus;
- vienotu e-pakalpojuma izstrādes, piegādes un darbināšanas infrastruktūru, pamatojoties uz konteinerizācijas principiem un izmantojot Kubernetes orķestrācijas platformu.

Par primāro tiek uzskatīta 2020.gada pamatā būvēta platforma un visi jaunie e-pakalpojumi ir jārealizē, izmantojot šo ietvaru.

LVP e-pakalpojumu platforma nodrošina:

1. Centralizēto katalogu lietošanu:

- Resursu katalogs. XML shēmu, citu XML resursu krātuve (skat. [12]), kas nodrošina šādas iespējas:
 - jauno XML resursu (shēmu, transformāciju, servisu definīciju) reģistrācija;
 - esošo XML resursu rediģēšana ar versiju kontroli;
 - detalizēta apraksta un hierarhijas informācijas attēlošana;
 - publiskas pieejas nodrošināšana reģistrētajiem XML resursiem ar iespēju veidot atsauces, izmantojot XML shēmu operatorus `include` un `import`.

XML resursu kataloga lietošanu reglamentē VISS XML shēmu izstrādes vadlīniju dokuments.

- API pārvaldnieks. API Pārvaldnieks nodrošina iespēju piekļūt valsts reģistru un pašvaldības informācijas sistēmu sniegtajiem datu servisiem standartizētā veidā. API Pārvaldnieks ir Pieprasījumu servisa tehnoloģiskais pēctecis, un tas jāizmanto kā primārais jaunu e-pakalpojumu izstrādē. API Pārvaldnieks ir uz WSO2 enterprise service bus standarta risinājuma bāzēts risinājums, kas nodrošina:
 - iespēju datu devējiem reģistrēt savus REST un SOAP servissus API publicētāja (API publisher) saskarnē;
 - iespēju datu ņēmējiem izmantot Izstrādātāju portālu (API Store), lai izmantotu publicētos servissus;

- drošības slāni, izmantojot OAuth 2.0 protokolu ar VISS PFAS autentifikāciju. E-pakalpojumu aizmugursistēmas servisi nodrošina automātiskas talonu apmaiņas starp LVP Identity Server (IDS) un PFAS STS.
 - pieprasījumu un atbilžu auditēšanu;
 - iespēju nodot OAuth2 references talonu (access) talonu [10], kas tiek izmantots API publicētājā (API publisher) publicēto servisu (API) izsaukšanai līdz pat biznesa servisam, lai veiktu papildus pieejas kontroli datiem biznesa servisa līmenī;
- Klasifikatoru pārvaldības modulis. Nodrošina VISS un e-pakalpojumu klasifikatoru uzturēšanu un aktualizāciju.
2. E-pakalpojumu ietvara aizmugursistēmas mikroservisi tehnoloģiski ir pieejami no daudziem pieejas punktiem: valsts un pašvaldību portāliem, iestāžu mājaslapām, u.c.;
- Profila mikroserviss – nodrošina informācijas izgūšanu un saglabāšanu lietotāja profilā;
 - Elektronisko dokumentu mikroserviss – nodrošina pieeju pie elektronisko dokumentu krātuves (EDK);
 - Notifikāciju mikroserviss – nodrošina paziņojumu nosūtīšanu uz KDV vai e-pastu;
 - Maksājumu mikroserviss - nodrošina apmaksu plūsmas kontroli, attaisnojuma dokumentu sagatavošanu, maksājumu apmaksas procesa vadību un sasaisti ar apmaksas nodrošinātājiem;
 - Konfigurācijas mikroserviss – nodrošina e-pakalpojuma konfigurācijas izgūšanu no e-pakalpojumu kataloga.
 - Kļūdu pieteikšanas mikroserviss – nodrošina iespēju e-pakalpojumā lietotājiem pieteikt kļūdas par e-pakalpojuma darbību.
 - Pieprasījumu mikroserviss – nodrošina iespēju izsaukt API pārvaldniekā reģistrētos servisos (API) un iespēju uzsākt transakciju.
 - Navigācijas mikroserviss – nodrošina e-pakalpojumu galvenē un kājenē attēlojamus datus.
 - Konteksta mikroserviss – nodrošina iespēju īslaicīgi uzglabāt e-pakalpojuma izpildei nepieciešamos datus (sesijas aizvietotājs).
3. LVP portāla sadaļas, kas atbalsta e-pakalpojumu darbību:
- Klienta darba vieta (KDV) – iedzīvotajā darba vieta e-pakalpojumu izpildes rezultātu apskatei un kontrolei.
 - Publisko pakalpojumu katalogs, kas nodrošina pakalpojumu pieejamību Latvijas valsts portāla lietotājiem;
4. Citu koplietošanas servisu un resursu kopums:
- Adrešu meklēšanas komponente, kas nodrošina unificētu lietotāja saskarni jebkurā web-lietojumā un lietotāja veikto darbību rezultātu atgriešanu e-pakalpojumam;
 - Lietotāju autentifikācijas modulis – LVP Identity Server (IDS), kas nodrošina lietotāju autentifikāciju.
 - Koplietojamie statistiski resursi (Assets) – satur attēlus, CSS un JS e-pakalpojumu darbības nodrošināšanai.
 - PFAS drošības talonu serviss – nodrošina iespēju pārbaudīt drošības talonu no e-pakalpojuma biznesa servisa vai citām backend slāņa komponentēm.

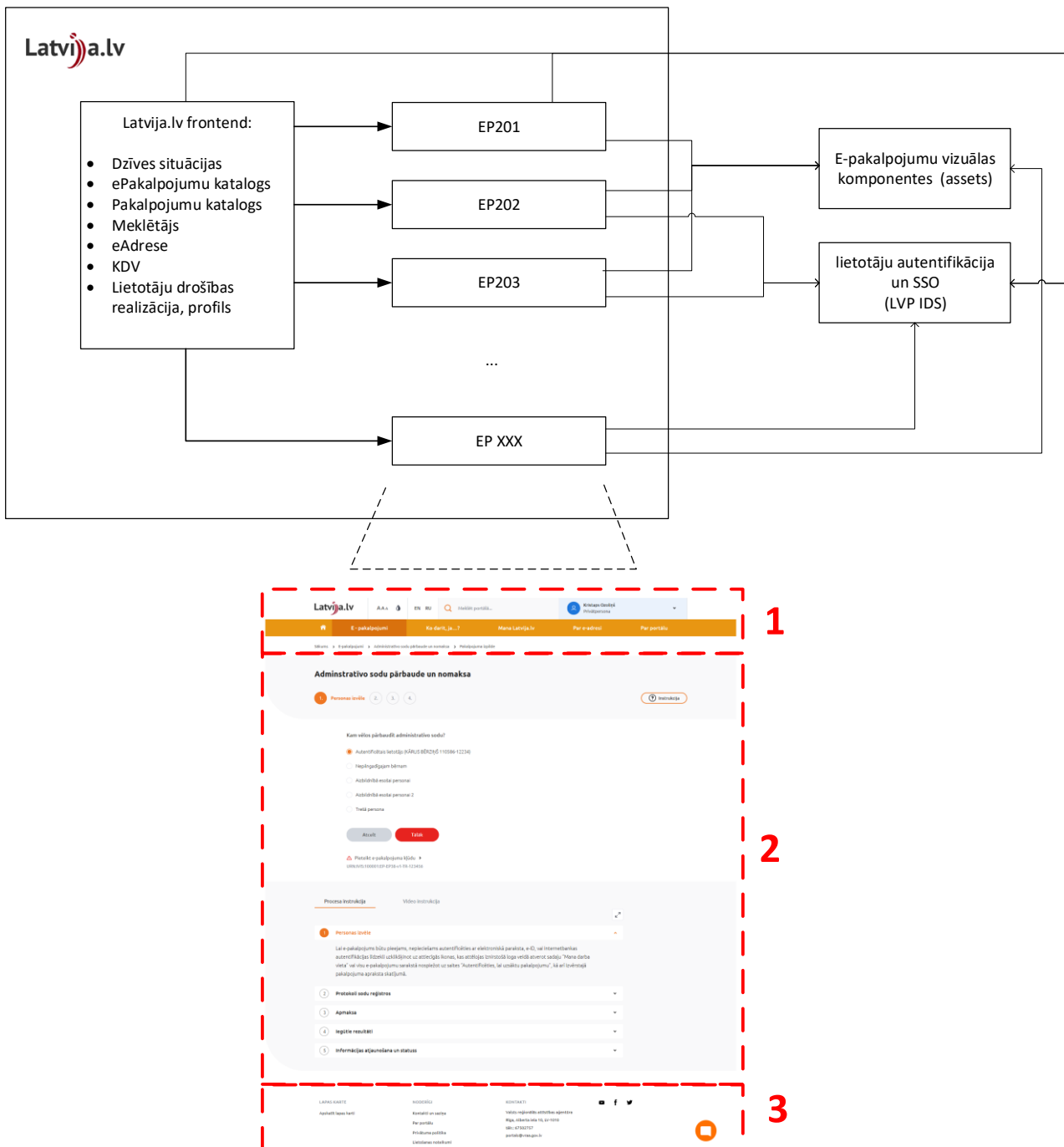
3.1. E-pakalpojuma izpildes loģika

Katrs e-pakalpojums tiek izpildīts neatkarīgi no LVP un tiek darbināts kā neatkarīga interneta vietne. E-pakalpojumi un LVP savā starpā nodrošina SSO un citas kopīgas portāla funkcijas, piemēram: valodu pārslēgšana, lietotāju kontu pārslēgšana, tekstu izmēru maiņa un vājredzīgo režīma pārslēgšana. Visi e-pakalpojumi sastāv no trim daļām (skat. 3.attēlā):

1. Galvenes daļa;
2. Satura daļa (e-pakalpojuma izpildes soļi);
3. Kājenes daļa.

Darba virsmas galvenes un kājenes daļas ir kopīgotas visiem pakalpojumiem un ir integrētas e-pakalpojuma ietvarā, kā datu avots tiek izmantots navigācijas mikroservis, kura saturu pārvalda VRAA.

Saturs ir e-pakalpojuma mainīga daļa, bet tiek būvēts, izmantojot kopējas vizuālas komponentes (SDK) React vai .NET Core MVC.



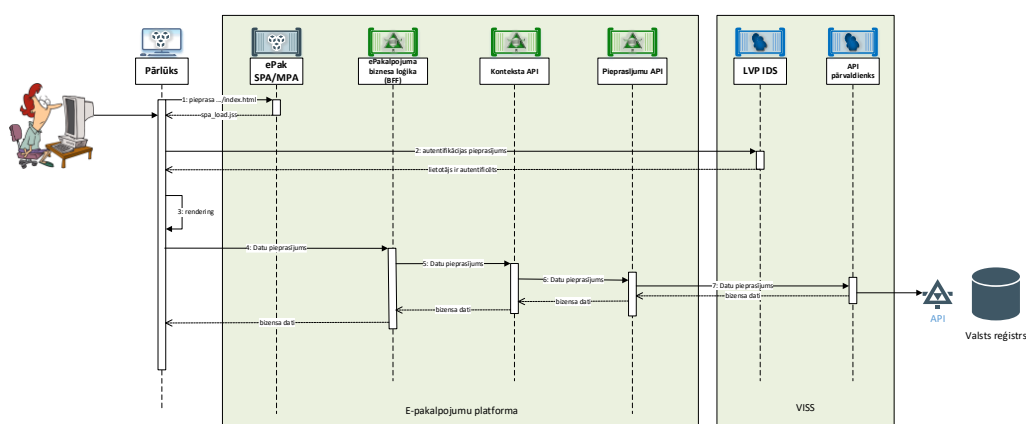
3.attēls. E-pakalpojuma sastāvs un mijiedarbība ar LVP

3.2. Risinājuma sadalījums slāņos

Risinājuma arhitektūra nodrošina iespēju izmantot LVP komponentes un koplietošanas servisu vairākos e-pakalpojumos. 4.attēlā ir demonstrēta tipiskā e-pakalpojuma darbību plūsma un tās dalījums slāņos. Pakalpojuma izpildes posmi ir numurēti, un tiem ir noteikta atbilstoša izpildes secība:

1. Lietotājs no savas darba stacijas, izmantojot pārlūku Latviya.lv portālā, sameklē vēlamo e-pakalpojumu, piekrīt tā lietošanas nosacījumiem un pāriet uz LVP e-pakalpojuma lietojumu.
2. Notiek atbilstoša Docker konteineru izpilde un tiek veikts e-pakalpojumu konfigurācijas mikroservisa izsaukums.

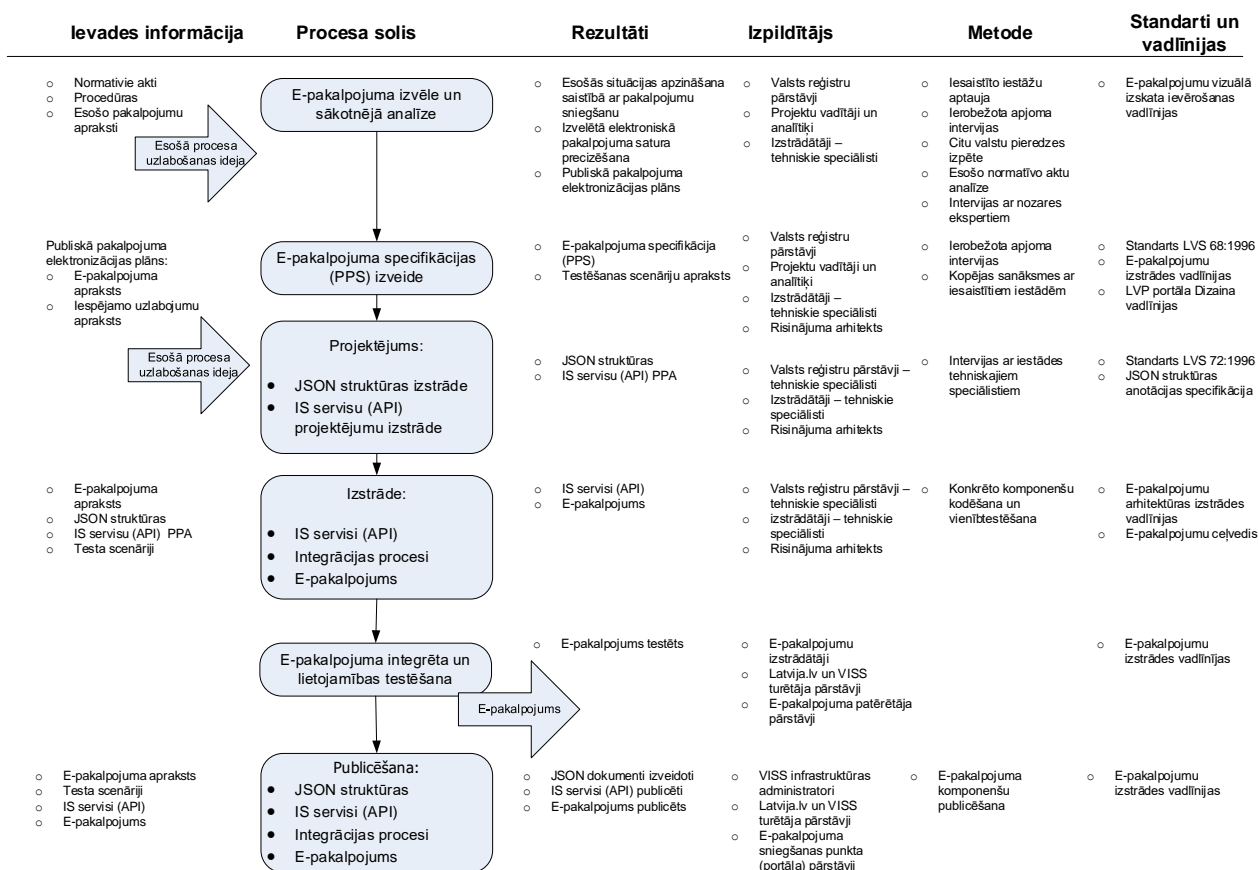
3. Tiek renderēts e-pakalpojums pārlūkā, veikta statisko resursu pieprasīšana no Assets un automātiska autentifikācija LVP IDS ar SSO, ja e-pakalpojums ir paredzēts tikai autentificētiem lietotājiem.
4. BFF (SPA gadījumā) vai servera koda (MPA gadījumā) uzsāk e-pakalpojuma biznesa vai datu devēja servisa (API) izsaukumu izmantojot Konteksta mikroservisu. Pieprasījumā tiek iekļauts IDS drošības talons;
5. Konteksta API mikroserviss SPA gadījumā veic IDS OAuth2 references drošības talona apmaiņu pret IDS OAuth2 JWT drošības talonu un veic Pieprasījumu mikroservisa servisa izsaukumu;
6. Pieprasījumu mikroserviss veic IDS drošības talona apmaiņu pret PFAS AUTH drošības talonu. Un veic API Pārvaldniekā publicētā e-pakalpojuma biznesa vai datu devēja servisa (API) izsaukumu. API Pārvaldnieks pārbauda pieprasījumā norādītā drošības talona derīgumu, vai ir izveidots abonements starp Pieprasījumu API un norādīto biznesa API un vai e-pakalpojumam ir piešķirtas nepieciešamās atļaujas (scopes) biznesa servisa izsaukšanai.
7. API Pārvaldnieks veic e-pakalpojuma biznesa servisa vai datu devēja servisa izsaukumu.



4.attēls. Risinājuma slāņu arhitektūra

4. E-pakalpojuma izstrādes process

E-pakalpojuma izstrādes process ar nodevumu uzskaitījumu parādīts 5.attēlā.



5.attēls. E-pakalpojuma izstrādes process

E-pakalpojuma izstrādes procesa pamatā jāizmanto ūdenskrituma programmatūras izstrādes metodoloģija. Gadījumā, ja tiks izmantota cita metodoloģija (piemēram, Agile), jānodrošina visu minēto nodevumu sagatavošana un standartu ievērošana.

4.1. E-pakalpojuma prasību specifikācijas izveide

Katra e-pakalpojuma veidošanā ir iesaistītas vairākas puses, tāpēc pirms e-pakalpojuma realizācijas ir jāapraksta tā prasības saskaņā ar [3] un [5]. Visām iesaistītajām pusēm ir jāapstiprina šis dokuments, un jānosaka realizācijas laika grafiks.

E-pakalpojuma prasību specifikācija (PPS) ir dokuments, kurā iekļautas šādas nodaļas:

1. E-pakalpojuma apraksts – satur metainformāciju saskaņā ar e-pakalpojuma standartu, kā arī citas e-pakalpojumam izvirzāmās biznesa prasības.
2. E-pakalpojuma procesa modelis – e-pakalpojuma procesa diagramma kādā no biznesa procesu konceptuālās (augsta līmeņa) modelēšanas notācijām. Svarīgi atcerēties, ka e-pakalpojumā elektronizējams process nav obligāts un nevar būt “viens pret viens” pārņemts esošajā iestādes procesā.
3. E-pakalpojuma procesa soļu apraksts – visu e-pakalpojuma procesa soļu tekstuāls apraksts, kā arī norādes uz IS servisiem un VISS nodrošinātiem koplietojamiem moduļiem, kas tiek izmantoti procesa soļu gaitā.

4. E-pakalpojuma prasību apraksts. Katras prasības aprakstam jāietver vismaz šādas nodaļas: identifikators, prasības nosaukums, ievads, ievaddati, prasības apraksts, izvaddati, izsaucamo prasību identifikators un nosaukums.
5. E-pakalpojuma izmantoto JSON dokumentu uzskaitījums.
6. E-pakalpojuma izmantoto IS servisu (API) īss apraksts – satur meta informāciju, kā arī citas biznesa prasības, kas izvirzāmas e-pakalpojuma procesā izmantotajiem IS servisiem.
7. E-pakalpojuma procesa scenāriju apraksts – iespējamie e-pakalpojuma procesa izpildes scenāriji. Jāapraksta gan scenāriji, kas izriet no pakalpojuma procesa, gan arī negaidītie e-pakalpojuma izpildes scenāriji, kuri pakalpojuma procesā nav paredzēti. Procesu scenāriji tiek izmantoti kā ieejas materiāls testa scenāriju veidošanā, lai veiktu e-pakalpojuma akcepttestēšanu.
8. E-pakalpojuma ekrānformu prototipi saskaņā ar [3]. Tiek modelēti izmantojot Microsoft Visio vai citu dizaina rīku, kas atbalsta lietotāja saskarnes formu lauku pievienošanu un pielāgošanu.
9. Nolikuma (izmaiņu pieprasījuma) prasību atbilstības tabula.

4.1.1. Posma nodevumi

Posma realizācija ietver šādus nodevumus:

1. E-pakalpojuma specifikācija (PPS), atbilstoši [5];
2. Testēšanas scenāriju apraksts, atbilstoši [6].

4.2. Projektējums

E-pakalpojuma projektējuma izstrāde ietver šādus posmus:

1. JSON struktūras izstrāde.
2. IS servisu projektējuma izstrāde.

4.2.1. JSON struktūras

JSON struktūras projektējumam un izstrādei jāizmanto JSON-Schema specifikācija [9], lai nodrošinātu vienotu pieeju JSON dokumentu anotāciju un validāciju veikšanai.

JSON ir balstīts uz divām universālām datu struktūrām, kuras atbalsta visas mūsdienu programmēšanas valodas:

- “nosaukums/vērtība” kolekcija (name/value);
- sakārtots vērtību saraksts.

JSON struktūra sastāv no JSON dokumentiem, kuri jāprojektē katram e-pakalpojumam atsevišķi, jo ieejas un izejas dati katram gadījumam var atšķirties.

Iespējama situācija, kad viena JSON struktūra nodrošina vienlaicīgi vairāku e-pakalpojumu izpildi. Šādos gadījumos tas ir jānorāda e-pakalpojuma specifikācijā.

Parasti JSON struktūras projektēšana un izstrāde notiek paralēli, jo JSON dokumentu aprakstīšana pati par sevi ir vienkārša, projektētājam saprotama un ļauj stingri definēt apmaināmos formātus jau projektēšanas stadijā. Parasti netiek veidots speciāls dokuments par JSON struktūras projektējumu, bet citi e-pakalpojumu komponentu projektējumi (IS servisi, e-pakalpojums) tiek papildināti ar atbilstošām sadaļām.

4.2.2. IS servisu (API) programmatūras projektējuma apraksta izstrāde

Pirmā informācija par nepieciešamajiem IS servisiem (API) un to īss apraksts tiek sagatavots E-pakalpojuma prasību specifikācijas izstrādes laikā, skat. 4.1.nodaļu. Šajā posmā par katru nepieciešamo IS servisu (API) ir jāizstrādā IS servisa (API) programmatūras projektējuma apraksts. IS servisu (API) programmatūras projektējumā ir jāapraksta:

1. IS servisu (API) identifikācija un apraksts
2. IS servisu (API) konceptuāla arhitektūra
3. IS servisu (API) saskarnes apraksts.
4. Drošības ierobežojumu apraksts.
5. IS servisu (API) metodes tehniskā un biznesa izmantošanas instrukcija.
6. Servisu kvalitātes un sistēmas noslogojuma novērtējums (SLA).

Projektējumam jāsaturs nodaļas, kas detalizēti apraksta konkrēto IS servisu (API) moduļus, iekļaujot šādas sadaļas – identificējums, apraksts, ievade, apstrāde, izvade un kļūdas.

Ja esošais IS serviss (API) izmanto, daļēji vai pilnīgi balstās uz jau iepriekš API pārvaldniekā reģistrētiem IS servisiem (API), tad jāapraksta tikai nodaļas, kas attiecināmas uz jaunveidojamo IS servisa (API) daļu, uz pārējām – tiek veikta norāde.

Detalizēts IS servisa (API) vienas metodes projektējuma specifikācijas piemērs ir dots 5.1.pielikumā.

4.2.3. Posma nodevumi

Posma realizācija ietver šādus nodevumus:

1. JSON struktūra;
2. IS servisu (API) Programmatūras projektējuma apraksts (PPA).

4.3. Izstrāde

E-pakalpojuma izstrāde ietver šādus posmus:

1. IS servisu (API) izstrāde.
2. E-pakalpojumu prezentācijas slāņa izstrāde (e-pakalpojuma lietotajā saskarne, paziņojumi utt.).

E-pakalpojumu detalizētas izstrādes vadlīnijas ir aprakstītas dokumentā [4]. Šis dokuments satur konkrētus norādījumus par VISS un Latvija.lv izmantojamo platformu un tehnoloģijām.

4.3.1. IS servisi (API)

4.3.1.1. IS servisu (API) izstrāde

Konkrēta IS servisa (API) izstrāde notiek sadarbībā ar iestādes datu sniedzēju un saskaņā ar IS servisu specifikāciju, kas ir 4.2.2.sadaļā aprakstītā posma ieguvums. API jāveido ievērojot Open Api 2.0. vai 3.0 standartu.

Ja paredzēts izvietot vai publicēt IS servisu (API) VISS infrastruktūrā ir jāievēro arhitektūras un izstrādes rekomendācijas [4].

Ja biznesa prasību realizācijai IS servisā tiek veiktas operācijas ar personas kodiem, jāņem vērā rekomendācijas attiecībā uz personas koda maiņu [7].

4.3.1.2. IS servisu (API) testēšana

No testēšanas viedokļa IS servisi (API) ir “melnā kaste” ar noteiktu ieeju un izeju. Katra IS servisa testēšanai jāizveido speciāli testa scenāriji. Ņemot vērā, ka ieejas un izejas parametri ir JSON ziņojumi, scenārijam jāparedz vismaz šādi testi:

1. vairāku pareizi noformētu pieprasījumu apstrāde un rezultāta saņemšana;
2. vairāku nepareizi noformētu pieprasījumu apstrāde un rezultāta saņemšana;
3. pieprasījuma, kas ir ievērojami lielāks, nekā paredzēts (apstrādājamais lielums atkarīgs no konkrēta integrācijas procesa, un tam jābūt specificētam), apstrāde un rezultāta saņemšana;
4. tukša pieprasījuma un pieprasījuma, kas noformēts pēc cita formāta un satura, apstrāde un rezultāta saņemšana;
5. nefunkcionāla testēšana;
6. testēšana pēc apjoma, slodzes testi.

API izstrādei un testēšanai var izmantot speciāli tam paredzēto rīku – Postman. Svarīgi katra API paredzēt saskarnes publicēšanu izmantojot Open Api 2.0. vai 3.0 standartus, sk. [4].

4.3.1.3. Reģistrācija API pārvaldniekā

Pirms IS servisa (API) izmantošanas, nepieciešams veikt servisa reģistrāciju API pārvaldniekā (API publisher) portālā, sk. [8].

IS servisu (API) reģistrāciju API pārvaldniekā testa vidē un produkcijas vidē veic IS servisu (API) uzturētājs. Tiesības uz IS servisu (API) izmantošanai arī piešķir IS servisu (API) uzturētājs atbilstoši iestādes politikai izmantojot VISS portāla PFAS moduli.

4.3.2. E-pakalpojums

4.3.2.1. Izstrāde

E-pakalpojuma lietotnes izstrādei jāizmanto LVP prezentācijas ietvaru [1] un tajā nodrošināto SDK, sk. [4]. Lietotnei jānodrošina visu nepieciešamo e-pakalpojuma formu, klasifikatoru un pārbaudes nosacījumu realizācija.

Visiem e-pakalpojumiem, jāievēro izstrādes vadlīnijas [4]. Vadlīnijas [4] ietver konkrētus norādījumus izstrādātājiem.

Ja biznesa prasību realizācijai e-pakalpojumā tiek veiktas operācijas ar personas kodiem, jāņem vērā rekomendācijas attiecībā uz personas koda maiņu [7].

E-pakalpojumu izstrāde ir iespējama arī daļēju pieslēgumu LVP e-pakalpojumu platformai, sk. [1] sadaļā 7.9.

Lai uzsāktu izstrādi e-pakalpojumam tam jābūt reģistrētam e-pakalpojumu katalogā un PFAS modulī, skatīt 4.3.2.4 nodaļu.

4.3.2.2. Testēšana

E-pakalpojuma testēšana un pārbaude notiek ar pieslēgumu pie Vienotas pieteikšanas moduļa, e-pakalpojumu platformas un citiem izmantotiem palīgserveriem.

E-pakalpojuma testēšanas gaitā jānodrošina analīzes fāzē izveidoto un saskaņoto testa plānu un scenāriju izpilde, kas tiek fiksēts atbilstošā protokolā.

4.3.2.3. Piegāde

E-pakalpojumu izstrādes rezultāts ir vairāki konteineri (Docker'i), kas ietver e-pakalpojuma prezentāciju (SPA vai MPA) un izpildes biznesa loģiku (BFF). E-pakalpojumu konteineru veidošana notiek saskaņā ar [1] sadaļu 8, kas ietver:

- React SDK balstīto e-pakalpojumu konteineru izveide
- .NET core MVC balstīto e-pakalpojumu konteineru izveide
- E-pakalpojumu Helm izmitināšanas skriptu veidošana
- E-pakalpojumu platformas komponentu konfigurēšana

4.3.2.4. E-pakalpojuma reģistrācija

Katram e-pakalpojumam jābūt reģistrētam E-pakalpojumu katalogā (reģistrācijas instrukciju skatīt [13]) un PFAS modulī (instrukciju skatīt [14]). Reģistrāciju nodrošina VISS administrators, bet no e-pakalpojuma izstrādātāja tiek sagaidīta šāda informācija:

1. e-pakalpojuma aprakstošā informācija, lai nodrošinātu autentifikāciju un to reģistrētu PFAS modulī – tiek iesniegta VRAA uzsākot izstrādi;
2. publikācijas informācija par e-pakalpojumu, lai to reģistrētu e-pakalpojumu katalogā – minimālā apjomā tiek iesniegta uzsākot izstrādi;
3. izmantoto IS servisu (API) uzskaitījums.

E-pakalpojuma reģistrācijai e-pakalpojumu katalogā nepieciešamā informācija jāapkopo e-pakalpojuma apraksta šablonā¹, kas satur informāciju e-pakalpojuma uzstādīšanai VISS, kuru pilnā apjomā jāaizpilda un jāiesniedz VRAA kopā ar pirmreizējo e-pakalpojumu laidieni.

4.3.3. Posma nodevumi

Posma realizācija ietver šādus nodevumus, kas ir paredzēti izvietojumam VISS un LVP infrastruktūrā:

1. BFF servisi (API);
2. E-pakalpojuma lietotāju saskarne.

Nodevumi, kas ir paredzēti e-pakalpojumu darbības pārbaudei un publikācijai:

1. BFF servisu (API), e-pakalpojumu testēšanas protokoli;
2. Administratora rokasgrāmata;
3. Publikācijas informācija par e-pakalpojumu.

4.4. E-pakalpojuma publicēšana

Pirms e-pakalpojums kļūst pieejams pakalpojuma portālā, to nepieciešams publicēt E-pakalpojumu katalogā. Pirms publikācijas VISS administratori pārbauda, vai visas e-pakalpojuma komponentes ir reģistrētas un publicētas attiecīgajos katalogos.

4.4.1. E-pakalpojumu akceptēšanas nosacījumi (check-list)

VISS uzturētāju uzdevums ir pārbaudīt katru e-pakalpojuma nodevumu (skat. 4.3.3. sadaļu) atbilstoši šādai specifikācijai:

¹ Pieejams lejupielādei VISS portālā

https://viss.gov.lv/lv/Informacijai/Dokumentacija/Koplietosanas_komponentes/EPAK_izstrades_izpildes_vide

E-PAKALPOJUMA KOMPONENTE	KONTROLES PUNKTS	
JSON struktūras	1. E-pakalpojumu izmantotam API ir izveidotas JSON shēmas	Y
IS servisi (API)	2. Visos IS servisos (API) izmantotas JSON struktūras un to izmantošana notiek korekti.	Y
	3. IS servisa (API) SLA atbilst deklarētajam IS servisu projektējumā, turklāt:	Y
	3.1. Sinhrona IS servisa (API) atbildes laiks nav lielāks par 3 sek.	Y
	3.2. IS servisā (API) pārsūtamais datu apjoms:	Y
	o nav lielāks par 4 MB vai	Y
	o ir lielāks par 4 MB un tiek izmantots EDK	Y
E-pakalpojuma prezentācija	4. IS servisu (API) PPA tiek aprakstītas visas izņēmuma situācijas (notikumi)	Y
	5. IS servisu (API) testēšanas protokols	Y
	6. IS servisi (API) reģistrēti API pārvaldniekā	Y
	7. Atbilstība [3] standarta izvirzītajām prasībām	Y
	8. Tiek noteikti e-pakalpojumu pieturpunkti, kas nodrošinās iedzīvotājiem e-pakalpojuma izpildes gaitas atsekošanas iespējamību	Y
	9. Pieejama publikācijas informācija par e-pakalpojumu (kas atbilst e-pakalpojuma apraksta šablonam)	Y
	10. Pieejama e-pakalpojuma palīga lapa (lietotāja palīgs)	Y
	11. Pieejams e-pakalpojuma testēšanas protokols	Y
	12. Pieejama e-pakalpojuma administratora rokasgrāmata	Y

E-pakalpojums tiks publicēts tikai gadījumā, ja visi E-pakalpojumu akceptēšanas nosacījumi ir izpildīti.

4.4.2. E-pakalpojuma integrēta testēšana

Ar integrēto testēšanu saprot e-pakalpojuma testēšanu portālā, izmantojot nepieciešamos reģistru pieslēgumus. Integrētā testēšana notiek gan pirms e-pakalpojuma publicēšanas, gan arī tiek atkārtota pēc publicēšanas, kad ir nodrošināti visi reģistru pieslēgumi produkcijas vidē.

4.4.3. Posma nodevumi

Posma realizācija ietver šādu nodevumu:

1. E-pakalpojums ir publicēts un ir publiski pieejams.
2. E-pakalpojuma konsolidētā piegāde, kas satur, gan e-pakalpojuma latvija.lv prezentācijas daļu, gan IS servisi (API).
3. Administratora rokasgrāmata, pēc kuras var uzstādīt konsolidēto piegādi.

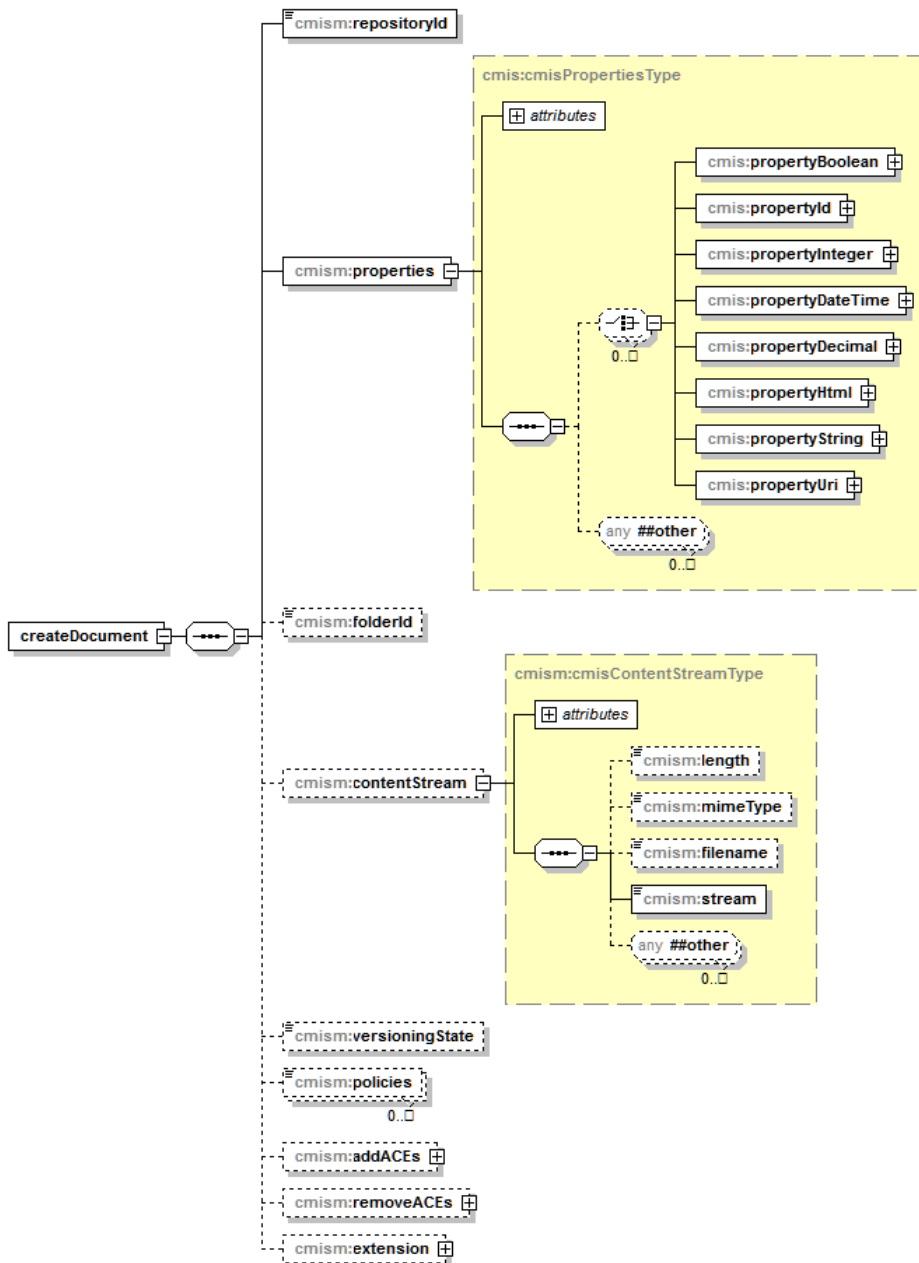
5. Pielikums

5.1. Metode "createDocument"

IDENTIFIKATORS	createDocument
NOSAUKUMS	Izveidot dokumentu
IEVADS	Izveido dokumenta objektu (tips noteikts ievaddatos) ievaddatos norādītajā mapē. Metodi var izmantot, ja nepieciešams izveidot dokumentu bez satura vai ar saturu, kura apmērs ir mazāks nekā 10 MiB.

IEVADDATI

- `repositoryId` – repozitorija identifikatora URN, obligāts;
- `<properties>` – izveidojamā dokumenta īpašības – obligāts; īpašības `cmis:name` vērtība ir reģistrjutīga, tā nevar sākties vai beigties ar atstarpes simbolu, un tā nevar saturēt nevienu no šiem simboliem (uzskaitījumā atdalīti ar komatiem): “\,/,?“,<,>,|”;
- `folderId` – mapes, kurā jāievieto izveidojamais dokuments, identifikators – obligāts;
- `contentStream` – dokumenta binārie dati.
- `versioningState` – dokumenta versijas pazīme, viens no: *none*, *major*, *minor*, noklusētā vērtība: „none”;
- `<addACEs>` – saraksts ar kontroles elementiem, kas jāpievieno objektam;
- `<removeACEs>` – saraksts ar kontroles elementiem, kas jāatņem objektam;
- `extension` – elements papildus datiem, specifiskai repozitorija vai programmatūras realizācijai. Šajā metodē elementa vērtība un/vai tā struktūra tiek ignorēta: *null*.



This is an extension element to hold any repository or vendor-specific extensions

Generated by XMLSpy

www.altova.com

Piemērs:

XML formātā

```
<ns:createDocument xmlns:ns="http://docs.oasis-open.org/ns/cmis/messaging/200908/">
  <ns:repositoryId>REPO-001</ns:repositoryId>
  <ns:properties>
    <ns1:propertyId xmlns:ns1="http://docs.oasis-open.org/ns/cmis/core/200908/"
propertyDefinitionId="cmis:objectId" queryName="cmis:objectId">
      <ns1:value>eds:d:customDocument</ns1:value>
    </ns1:propertyId>
    <ns1:propertyString xmlns:ns1="http://docs.oasis-open.org/ns/cmis/core/200908/"
propertyDefinitionId="cmis:name" queryName="cmis:name">
      <ns1:value>TestDocument.pdf</ns1:value>
    </ns1:propertyString>
    <ns1:propertyString xmlns:ns1="http://docs.oasis-open.org/ns/cmis/core/200908/"
propertyDefinitionId="eds:owner" queryName="eds:owner">
      <ns1:value>AU:100001</ns1:value>
    </ns1:propertyString>
  </ns:properties>
  <ns:folderId>
  <ns:contentStream>
  <ns:versioningState>
  <ns:policies>
  <ns:addACEs>
  <ns:removeACEs>
  <ns:extension>

```

```

<ns1:propertyString xmlns:ns1="http://docs.oasis-open.org/ns/cmisis/core/200908/"
propertyDefinitionId="eds:retentionGroup" queryName="eds:retentionGroup">
  <!-- tiks piešķirta noklusētā vērtība - "nenoteikta" -->
</ns1:propertyString>
</ns:properties>
<ns:folderId>FOLD-1234567</ns:folderId>
<ns:contentStream>
  <ns:filename>TestDocument.pdf</ns:filename>
  <ns:stream>/9j/4AAQSkZJRgABAQEASABI//9k=</ns:stream>
</ns:contentStream>
<ns:versioningState>none</ns:versioningState>
</ns:createDocument>

```

JSON formātā

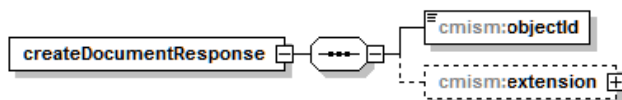
```

{
  "createDocument": {
    "repositoryId": "REPO-001",
    "properties": {
      "propertyId": {
        "value": "eds:d:customDocument",
        "_xmlns:ns1": "http://docs.oasis-open.org/ns/cmisis/core/200908/",
        "_propertyDefinitionId": "cmisis:objectId",
        "_queryName": "cmisis:objectId"
      },
      "propertyString": [
        {
          "value": "TestDocument.pdf",
          "_xmlns:ns1": "http://docs.oasis-open.org/ns/cmisis/core/200908/",
          "_propertyDefinitionId": "cmisis:name",
          "_queryName": "cmisis:name"
        },
        {
          "value": "AU:100001",
          "_xmlns:ns1": "http://docs.oasis-open.org/ns/cmisis/core/200908/",
          "_propertyDefinitionId": "eds:owner",
          "_queryName": "eds:owner"
        },
        {
          "_xmlns:ns1": "http://docs.oasis-open.org/ns/cmisis/core/200908/",
          "_propertyDefinitionId": "eds:retentionGroup",
          "_queryName": "eds:retentionGroup"
        }
      ]
    },
    "folderId": "FOLD-1234567",
    "contentStream": {
      "filename": "TestDocument.pdf",
      "stream": "/9j/4AAQSkZJRgABAQEASABI//9k="
    },
    "versioningState": "none",
    "_xmlns:ns": "http://docs.oasis-open.org/ns/cmisis/messaging/200908/"
  }
}

```

IZVADDATI

- objectId – izveidotā objekta identifikators;
- extension – vienmēr null.



This is an extension element to hold any repository or vendor-specific extensions

Generated by XMLSpy

www.altova.com

Piemērs:

XML formātā

```
<createDocumentResponse xmlns="http://docs.oasis-open.org/ns/cmismessaging/200908/">
  <objectId>DOC-1234567-V1.0</objectId>
  <extension xsi:nil="true"></extension>
</createDocumentResponse>
```

JSON formātā

```
{
  "createDocumentResponse": {
    "objectId": "DOC-1234567-V1.0",
    "extension": {
      "_nil": "true"
    },
    "_xmlns": "http://docs.oasis-open.org/ns/cmismessaging/200908/"
  }
}
```

KĻŪDAS

KĻŪDA	KODS	ZIŅOJUMS	APRAKSTS
<i>permissionDenied</i>	0	"createDocument"	Nav tiesību izsaukt metodi; pēc noklusējuma tas netiek pārbaudīts.
<i>permissionDenied</i>	0	"canCreateDocument.Folder"	Nav tiesību mapē veidot dokumentus.
<i>objectNotFound</i>	0	repositoryId vērtība	Neeksistē norādītais repozitorijs.
<i>objectNotFound</i>	0	folder vērtība	Neeksistē norādītā mape.
<i>invalidArgument</i>	0	"Property, cmis:name"	cmis:name vērtība satur neatļautus simbolus. Neatļautie simboli: '\', '.', '*', '?', '<', '>', ' ', '/', ''.
<i>invalidArgument</i>	0	"filename"	Nav norādīts contentStream filename (obligāts, ja norāda contentStream).
<i>streamNotSupported</i>	0	cmis:objectTypeld vērtība	Tips nenodrošina contentStream saglabāšanu.
<i>constraint</i>	102	cmis:objectId vērtība	Īpašības vērtība pārsniedz tipa definēto max vai min robežu.
<i>constraint</i>	103	cmis:objectId vērtība	Norādīta vairāk kā viena vērtība, lai gan tips nosaka, ka īpašības vērtība var būt ne vairāk kā viena.
<i>constraint</i>	104	cmis:objectId vērtība	Īpašības vērtība ir sistēmas noteikta.
<i>constraint</i>	105	cmis:objectId vērtība	Nav norādīta dokumenta tipa obligātā īpašība, tās vērtība vai īpašība norādīta ar nepareizu datu tipu.
<i>constraint</i>	106	cmis:objectId vērtība	Norādīta neatļauta īpašības vērtība (tips definē atļautās īpašības vērtības).
<i>constraint</i>	107	cmis:objectId vērtība	Norādīta īpašība, kas neietilpst dokumenta tipa definīcijā.
<i>constraint</i>	201	cmis:objectTypeld vērtība	Tips nav repozitorija dokumentu bāzes tips vai tā atvasinājums.
<i>constraint</i>	202	cmis:objectTypeld vērtība	Tipam nevar izveidot instances (objektus).

KĻŪDA	KODS	ZIŅOJUMS	APRAKSTS
<i>constraint</i>	204	cmis:objectTypeld vērtība	Tips nenodrošina versionēšanu, bet ir tiek mēģināts izveidot Minor vai Minor versiju.
<i>constraint</i>	205	cmis:objectTypeld vērtība	Tips nodrošina versionēšanu, bet tiek mēģināts izveidot neverсионējāmu dokumentu.
<i>constraint</i>	206	cmis:objectTypeld vērtība	Tips nenodrošina ACL pārvaldību, bet tiek mēģināts pielāgot ACL ierakstus.
<i>constraint</i>	303	folderId vērtība	Dokumenta tips nav starp mapes atbalstītajiem tipiem.
<i>constraint</i>	304	cmis:objectTypeld vērtība	Nav norādīts contentStream, lai gan tips nosaka, ka tas ir obligāts.
<i>constraint</i>	500	tiesība	Tiesība nav atbalstīta repozitorija konfigurācijā
<i>nameConstraintViolation</i>	0	cmis:name vērtība	cmis:name vērtība nav unikāla mapē, kurā tiek mēģināts izveidot dokumentu.
<i>storage</i>	0	""	Notika kļūda saglabājot dokumenta datni vai dokumentu datubāzē.