

LATVIJAS UNIVERSITĀTE
DATORIKAS FAKULTĀTE

**DETALIZĒTO (DRILL-DOWN) ATSKAIŠU MODUĻA
IZSTRĀDE SISTĒMAI PROIS**

Kvalifikācijas darbs

Autors: Atis Ēriks Dumpis
Studenta apliecības Nr: ad16035
Darba vadītājs: M. dat. Valts Grigāns

Rīga 2018

ANOTĀCIJA

Darbā ir aprakstīts sistēmas PROIS detalizēto atskaišu moduļa izstrādes process. Funkcionalitāte ir paredzēta sistēmas “PROIS” lietotājiem, savukārt dokumentācija vairāk paredzēta izstrādātāju un pasūtītāju auditorijai.

Sistēmas uzlabojumi palīdzēs Prokuratūras darbiniekiem ikdienas procesos tādos kā darbinieku darba izvērtēšanā, balstoties uz konkrētiem datiem no lietu žurnāliem un individuālās darba uzskaites reģistra, vai vieglāku un strukturētāku darbu izskates procesu, balstoties uz informāciju, kas būs pieejama. Ar sistēmas jaunizveidoto daļu būs iespēja attēlot atskaišu detalizāciju lietotāja saskarnē un arī lejupielādēt kā Microsoft Excel dokumentu.

Atslēgvārdi: Prokuratūra, atskaite, detalizācijas, Microsoft Visual Studio, Oracle PL/SQL, ūdenskrituma modelis.

ABSTRACT

DETAILED (DRILL-DOWN) REPORT MODULE DEVELOPMENT FOR THE SYSTEM OF PROIS

The work describes the process of developing a detailed report module for the PROIS system. Functionality is designed for users of the PROIS system, while the documentation is more intended for the audience and customer audience.

Systems upgrades will help the employees of the prosecutor's office in everyday processes such as evaluating the effectiveness of individual worker using the specific data from case registers or more efficient way of overlooking data. With the newly created part of the system, users will be able to display the report details in the user interface and download it as a Microsoft Excel document.

Keywords: prosecutor's office, report, detailing, Microsoft Visual Studio, Oracle PL/SQL, Waterfall module.

SATURA RĀDĪTĀJS

1	APZĪMĒJUMU SARAKSTS	8
2	IEVADS	9
2.1	Tēmas izvēle	9
2.2	PROIS informācijas sistēma	9
2.3	Dokumenta sadaļu kārtojums.....	9
3	PROGRAMMATŪRAS PRASĪBU SPECIFIKĀCIJA	10
3.1	Ievads	10
3.1.1	Dokumenta nolūks	10
3.1.2	Darbības sfēra.....	10
3.1.3	Dokumentā lietotie termini un saīsinājumi	10
3.1.4	Saistība ar citiem dokumentiem	11
3.1.5	Trasējamības tabula.....	11
3.2	Vispārējs apraksts	12
3.2.1	Produkta perspektīva	12
3.2.2	Produkta funkcijas/prasības	12
3.3	Funkcionālās prasības	14
3.3.1	Detalizētās atskaites skata atvēršana	14
3.3.2	Atskaites tabulas lauku attēlošana.....	15
3.3.3	Atskaites lauku specifikas, dialoga logu izveidošana	17
3.3.4	Atskaites tabulas lauka specifikas lejupielādēšana	20
3.3.5	Pogu izveide, ar kurām pārslēgties starp detalizācijām	21
3.3.6	Atskaites laukos, kur vērtību reprezentē vairākas skaitļu vērtības, atskaiti var ielādēt par katru vērtību atsevišķi.....	22
3.4	Darba konceptuālais modelis	23
3.5	Nefunkcionālās prasības	25
3.5.1	Veiktspējas prasības	25

3.5.2	Drošības prasības	25
3.5.3	Citas prasības	25
4	PROGRAMMATŪRAS PROJEKTĒJUMA APRAKSTS	26
4.1	Ievads	26
4.1.1	Nolūks	26
4.1.2	Darbības sfēra.....	26
4.2	Dekompozīcijas apraksts	26
4.2.1	Moduļu dekompozīcija.....	26
4.2.2	Procesu dekompozīcija.....	26
4.2.3	Datu dekompozīcija	27
4.3	Datu plūsmas diagrammas	28
4.3.1	Nulles līmeņa datu plūsmas modelis.....	28
4.3.2	Pirmā līmeņa datu plūsmas modelis Detalizēto atskaišu modulim.....	29
4.3.3	Otrā līmeņa datu plūsmas modeļi.....	30
4.4	Datu bāzes diagrammas	32
4.4.1	Datu bāzes tabulu projektējums un apraksts	32
4.5	Lietotāja ekrānformu projektējums.....	35
4.5.1	Vispārējs apraksts Ekrāna formu izveidē.....	35
4.5.2	Lietotāja pieejas kontrole saglabā tādas pašas tiesību piešķiršanas kā tās, kas piešķirtas atskaišu modulī.	35
4.5.3	Ekrāna formu apraksti	35
5	PROJEKTA ORGANIZĀCIJA.....	40
6	DARBIETILPĪBA.....	41
6.1	Dokumenta Darbietilpības novērtējums	41
7	KONFIGURĀCIJU PĀRVALDĪBA	43
7.1	Izstrādes vide	43
7.2	Testa vide.....	43
7.3	Akcepttestēšanas vide	43

7.4	Klienta akcepttestēšanas (pirmsprodukcijas) vide	43
7.5	Produkcijas vide.....	43
7.6	Programmkoda konfigurācijas pārvaldība	43
8	KVALITĀTES NODROŠINĀŠANA.....	46
9	TESTĒŠANAS DOKUMENTĀCIJA.....	47
9.1	Ievads	47
9.2	Testēšanas žurnāls.....	47
9.2.1	Testēšanas plāns	47
9.2.2	Testa rezultātu apzīmējumi	47
9.2.3	Vienībtestu apraksts	48
9.2.4	Funkcionālie saskarnes testi.....	51
10	SECINĀJUMI	60
11	IZMANTOTĀ LITERATŪRA	61
12	PIELIKUMI.....	62

ATTĒLU SARAKSTS

3.2.2.1. Att. Detalizēto atskaišu moduļa funkcionalitāšu blokshēma.....	13
3.3.1.1. Att. Detalizācijas poga pie atskaitēm	15
3.3.2.1. Att. Tabula ar atskaites laukiem lietotāja saskarnes skatā.....	17
3.3.3.1. Att. sarunas dialoga logs.....	19
3.3.3.2. Att. detalizācijas tabula.....	20
3.3.4.1. Att. Drukāšanas poga lauka detalizācijas logā	21
3.3.5.1. Att. Bultiņu pogas detalizācijas logā	22
3.4.1. Att. Konceptuālais modelis	24
4.4.2.1. Att. Konceptuālais ER modelis	27
4.4.3.1. Att. Realizācijas ER modelis	28
4.3.1.1. Att. Nulles līmeņa datu plūsmas diagramma.....	28
4.3.2.1. Att. Pirmā līmeņa datu plūsmas diagramma.....	29
4.3.3.1.1. Att. Otrā līmeņa datu plūsmas diagramma	30
4.3.3.2.1. Att. Otrā līmeņa datu plūsmas diagramma	30
4.3.3.3.1. Att. Otrā līmeņa datu plūsmas diagramma	31
4.5.2.1.1. Att. Poga detalizēts.....	35
4.5.2.2.1. Att. Neaizpildīta atskaites izveide	36
4.5.2.3.1. Att. Attēlota atskaite lietotāja saskarnes vidē.....	37
4.5.2.4.1. Att. Tabulas lauka detalizācija	38
5.5.2.5.1. Att. Lejupielādētā atskaites detalizācija	39
7.6.1. Att. TortoiseGit kļūdu labojuma pievienošana.....	44
7.6.2. Att. Programmkoda pievienošanas vēstures fragments.....	45

1 APZĪMĒJUMU SARAKSTS

Termins	Skaidrojums
AI	Lauka vērtībā tiek piešķirta automātiski radot unikālu vērtību priekš šī lauka (AUTO INCREMENT)
DB	Datu bāze
DU	Darba uzdevums
DVS	Dokumentu vadības sistēma
FK	Lauka vērtība kas ir atsauce uz citu tabulu (FOREIGN KEY)
Jenkins	Atvērtās piekļuves automatizācijas serveris
NULLABLE	Apzīmējums laukam kurā var pastāvēt nulles vērtība
PPA	Programmatūras projektējuma apraksts
PK	Unikāli identificēts lauks (PRIMARY KEY)
PPS	Programmatūras prasību specifikācija
PROIS	Prokuratūras informācijas sistēma
Redmine	Projektu kontroles instruments
XLSX	Microsoft Excel Atvērtais XML formāts

2 IEVADS

2.1 Tēmas izvēle

Tēma tika izvēlēta sarunā ar PROIS projekta vadītāju un kvalifikācijas darba vadītāju. Tika apspriests, kas būtu nepieciešamskopējās programmatūras uzlabojumiem un jaunajām funkcionalitātēm. Tika izvēlēta detalizēto atskaišu moduļa izveide kā adekvāts darba uzdevums kvalifikācijas darba izveidei.

2.2 PROIS informācijas sistēma

PROIS informācijas sistēma ir Latvijas Republikas prokuratūras visās struktūrvienībās izmantotā informācijas sistēma. Ikdienā šo sistēmu lieto vidēji 800 prokuratūras darbinieki. Iekļauj un nodrošina savstarpējo integrāciju tādām datu kopām kā:

- DVS pilna funkcionalitāte ar augstu automatizācijas līmeni;
- Specifisko pamatdarbības lietu uzskaitē (krimināllietas, pārbaudes lietas, administratīvās un civillietas, prokuroru novērtēšana);
- Pamatdarbības analīze (atskaišu modulis);
- Atbalsta procesu (iepirkumi, saimnieciskais atbalsts, personāls) elektronisku uzskaiti ;

2.3 Dokumenta sadaļu kārtojums

Kvalifikācijas darba dokumentācija sastāv no sekojošām daļām:

- PPS, kurā katrs DU tiek aprakstīts kā atsevišķa funkcionalitāte;
- PPA, kurā aprakstīts, kā realizēta moduļa izveide, kas ir izveidots un kāda informācija pievienota;
- Projekta organizācija, kurā vispārīgi aprakstīts projekta īstenošanas process;
- Darbietilpības novērtējums, kurā aprakstīts laika patēriņš darba izstrādei;
- Konfigurācijas pārvaldība, kurā aprakstīta projekta izstrādes vide;
- Kvalitātes nodrošināšana, kurā aprakstīti kvalitātes nodrošināšanas principi darba izstrādei;
- Testēšanas dokumentācija, kurā aprakstīti veiktie vienībtesti un arī saskarnes funkcionalitāšu testi;
- Secinājumi;
- Pielikumi – komentēta koda paraugs, atskaites XLSX datnes paraugs un tabulas drilldown funkcijas lauka ieraksta piemērs;

3 PROGRAMMATŪRAS PRASĪBU SPECIFIKĀCIJA

3.1 Ievads

3.1.1 Dokumenta nolūks

Šis dokuments ir lietojumprogrammas PROIS detalizēto atskaišu moduļa apraksts. Dokumenta nolūks ir radīt priekšstatu par to, ko saturēs minētais modulis un kādas ir tā paredzamās funkcijas. Šis dokuments izstrādāts saistībā ar Latvijas Universitātes profesionālās izglītības Kvalifikācijas darbu un uzņēmumu SIA ZZ Dats.




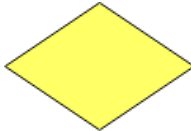

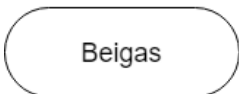
3.1.2 Darbības sfēra

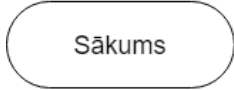



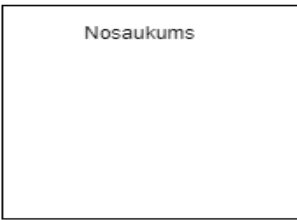
Ar šī moduļa izveidi ir paredzēts uzlabot ikdienas prokuratūras procesus gan darbinieku darba izvērtēšanā, gan efektīvāku darbību veikšanā. Programmatūru plānots izmantot detalizētas informācijas ieguvē, lai varētu vieglāk izsekot līdz prokuratūras procesiem, uzsāktajām un pabeigtajām lietām. Programmatūras sniegtajai informācijai paredzēts piekļūt tikai virsprokuroru grupai, kur kā nosacījums arī jābūt piešķirtai piekļuvei sistēmas daļā.

3.1.3 Dokumentā lietotie termini un saīsinājumi

Definīcijas un saīsinājumi sīkāk aprakstīti sadaļā 1. „Apzīmējumu saraksts”.

3.1.3.1 Lietotie apzīmējumi

Paskaidrojums	Apzīmējums
Modelis vai funkcionalitāte	
Datu bāze	
Sistēmas lietotāja apzīmējums	
Lēmuma pieņemšana	
Iekšējais process	
Procesa beigas	

Procesa sākums	
Datu ievade	
Datu izvade	
Datu plūsma un virziens	
Aptverošā sistēma	

3.1.4 Saistība ar citiem dokumentiem

Izmantotie dokumenti norādīti sadaļā 11 “*Izmantotā literatūra*”.

3.1.5 Trasējamības tabula

3.1.5.1. tab. *Trasējamības tabula*

DU numurs	DU apraksts	PPS
#4675	Pievienot atskaišu izveides skatam pogu, kas ved uz detalizāciju skatu	3.3.1.
#4712	Atskaites “Ikmēneša pārskats par darbu pirmstiesas kriminālprocesā” izveide Detalizēto atskaišu skatā	3.3.2.
#4715	Rādītāju detalizācijas pārskatu izvades saskarne	3.3.3.
#4725	Detalizētās atskaites par individuālajiem laukiem var drukāt kā XLSX datni	3.3.4.
#4720	Pogas pāriet uz nākamo vai iepriekšējo atskaites detalizēto lauku	3.3.5.
#4731	Dalītās informācijas ielāde par atsevišķām vērtībām	3.3.6.

3.2 Vispārējs apraksts

3.2.1 Produkta perspektīva

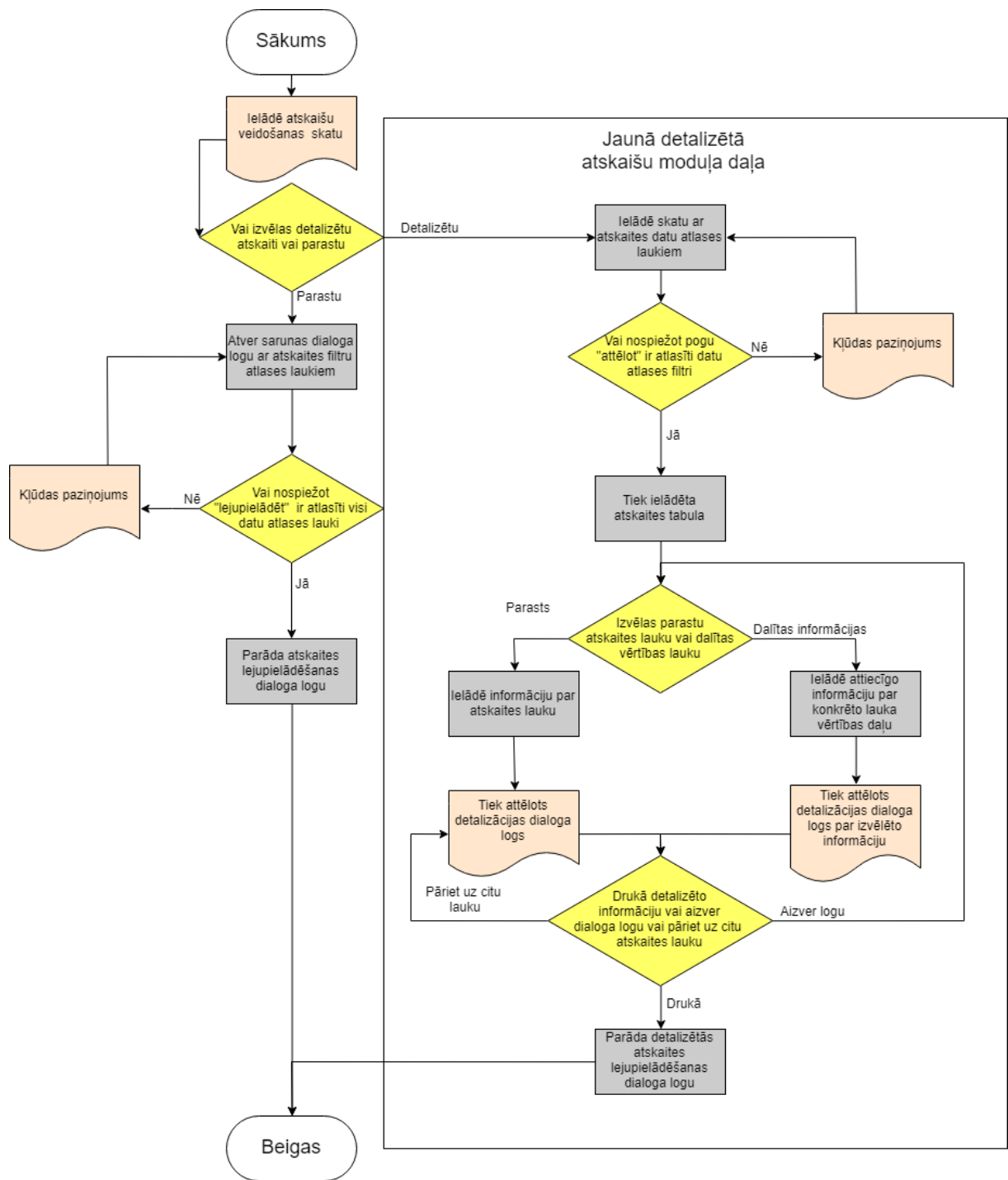
Lai padarītu ikdienas darbu prokuratūrā efektīvāku un lai piekļuve datiem būtu uzskatāmāka. Tiks dota iespēja piekļūt datiem, kas paskaidros parastas atskaites (skatīt 1. pielikumā) sniegto informāciju, izvēršot pašreizējās atskaitēs pieejamo lauku vērtības. Par atskaites lauku un tā vērtību tiks sniegta paplašināta informācija, kas ļaus izsekot konkrētiem prokuratūras procesiem.

3.2.2 Produkta funkcijas/prasības

Moduļa izveidē ir paredzēta vairāku funkcionalitāšu ieviešana esošajā programmatūrā. Modulis tiks izveidots kā apakšmodulis jau esošajam atskaišu moduļim.

- Papildināt atskaišu skatu ar pogu, kas ielādē jauno detalizētās atskaites skatu. Skats būs atverams tikai atskaitēm, kam sagatavoti detalizācijas lauku apraksti datu bāzē;
- Izveidojot detalizētās atskaites skata tabulu, var atlasīt datus tāpat, kā to varētu darīt veidojot atskaites XLSX datni un tabula, kas reprezentē atskaites tabulu, ir veidota maksimāli līdzīga lejupielādētai tabulai;
- Nospiežot uz atskaites lauka tiek ielādēts sarunas dialoga logs ar detalizāciju, kas ir atbilstoša, lai paskaidrotu sīkāk par datiem, kas pieejami no parastās atskaites, piemēram, laukā, kur informācija ir par “iesāktām lietām šajā mēnesī”, tiks atlasīts lietas numurs, struktūrvienība, kas to izveidojusi, un datums, kurā tieši uzsākta lieta;
- Lejupielādējot atskaites datnes, tām ir jābūt XLSX formātā;
- Atskaites laukos, kur lauka vērtība atskaitē ir reprezentēta kā vairāki skaitļi, informāciju var atlasīt par katru skaitļa vērtību atsevišķi, lai neradītu lieku datu sajaušanos;

Jaunā moduļa blokshēmas modulis parāda izveidojamās programmatūras struktūru un to, kādā secībā norit moduļa procesi.



3.2.2.1. Att. Detalizēto atskaišu moduļa funkcionalitāšu blokhēma

3.3 Funkcionālās prasības

3.3.1 Detalizētās atskaites skata atvēršana

Mērķis

Izveidot pogu “Detalizēts” Atskaišu moduļa jauno atskaišu izveides skatā. Poga parādīsies tikai pie tām atskaitēm, kurām ir izveidoti detalizēto atskaites lauku tabulas ieraksti. Nospiežot šo pogu, varēs nokļūt jaunā skatā, kur būs jau ielādēti datu atlases lauki Atbilstoši tiem, kuri ir vajadzīgi parastas atskaites (skatīt 1. pielikumā) izveidei.

Tiesības

Šī funkcionalitāte pieejama tikai situācijās, kur sistēmas lietotājam ir piešķirtas attiecīgās tiesības. Ir nepieciešamas tiesības piekļūt atskaišu rīkam, lai būtu, iespējams tikt pie detalizēto atskaišu veidošanas, un katram atskaites tipam ir atbilstoši vēl savas piekļuves tiesības, kas nozīmē, ka lietotājam būtu jābūt arī piekļuvei pie attiecīgās atskaites veidošanas, lai varētu tikt pielaists arī pie detalizētās atskaites veidošanas.

Ievaddati

Ievaddati tiek padoti, nospiežot pogu “Detalizēts”, bez sistēmas lietotāja alternatīvu datu padošanas iespējām.

3.3.1.1. tab. Ievaddatu tabula

Nr.	Lauks	Tips	Obligāts	Apraksts
1.	Atskaites veids	Poga pie atskaitēm	jā	Nospiežot pogu, tiek padots Id, kas vajadzīgs pareizo datu atlases lauku attēlošanai jaunajā skatā.

Apstrāde

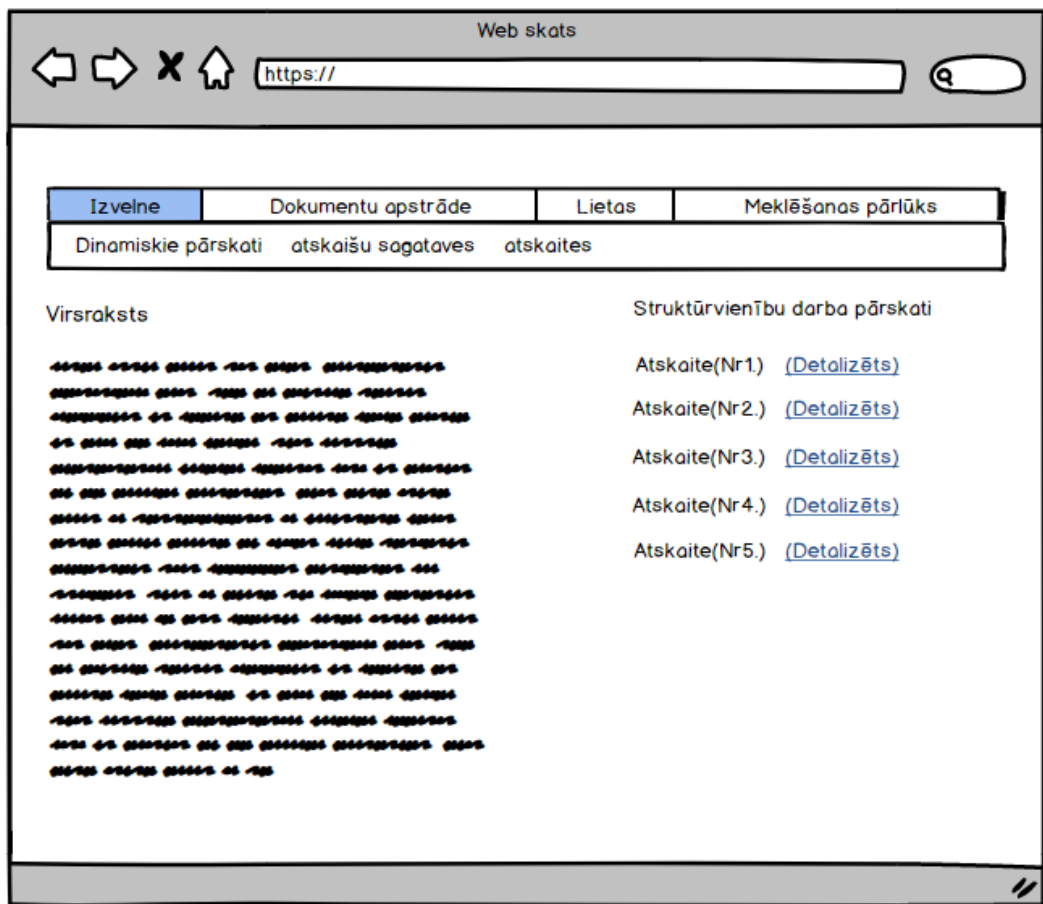
Izmantojot java script, tiek atlasīti lauki, ko attēlot jaunajā skatā.

Izvaddati

Tiek atvērta jauns skats, kurā ir attēloti visi lauki, kas neieciešami attiecīgās atskaites detalizācijas izveidei.

Saskarnes piemērs

Attēlā redzams pašreizējais atskaišu skata izkārtojums ar iecerēto detalizācijas pogu.



3.3.1.1. Att. Detalizācijas poga pie atskaitēm

3.3.2 Atskaites tabulas lauku attēlošana

Mērķis

Tiek ielādēta atskaites tabula lietotāja saskarnes vidē, un šajā tabulas attēlojumā ir iekļauti visi tie paši dati, kas būtu redzami šīs pašas atskaites lejupielādētā XLSX datnes formāta atskaitē.

Tiesības

Ir nepieciešamas tiesības piekļūt atskaišu rīkam un attiecīgajai atskaitei, lai būtu iespēja izmantot funkcionalitāti kas varēs izveidot tabulas attēlojumu.

Ievaddati

3.3.2.1. tab. Ievaddatu tabula

Nr.	Lauks	Tips	Obligāts	Apraksts
1.	Atskaites veids	Izkrītošā izvēlne	Jā	Atskaites tips kas ir izvēlēts
2.	Departaments	Izkrītošā izvēlne	Jā	Departaments, par kuru tiek veidota atskaite

3.	Darbinieks kurš veic pieprasījumu	Tiek iegūta informācija no lietotāja datiem	Jā	Darbinieks, kurš veido atskaites detalizāciju
4.	Laika perioda tips	Radiopogu grupa	Radiopogu grupa	Laika atlases izvēlētais tips
5.	Gads	Datuma izvēles lauks	Jā, ja ir izvēlēts attiecīgais laika perioda tips.	Gads, par kuru veidot atskaiti
6.	Mēnesis	Datuma izvēles lauks	Jā, ja ir izvēlēts attiecīgais laika perioda tips.	Mēnesis, par kuru veidot atskaiti
7.	Perioda sākuma datums	Datuma izvēles lauks	Jā, ja ir izvēlēts attiecīgais laika perioda tips.	Datums, no kura sākt atlasīt datus
8.	Perioda beigu datums	Datuma izvēles lauks	Jā, ja ir izvēlēts attiecīgais laika perioda tips.	Datums, līdz kuram atlasīt datus
9.	Lappušu numurs	Datuma izvēles lauks	Nē	Lappušu skaits dokumentam (svarīgs tikai 19 atskaitēm)

Apstrāde

Dati tiek apstrādāti un pārsūtīti vairākos soļos

1. Tiks pārbaudīts vai struktūrvienība, datuma atlasīšanas tips, gads, mēnesis, datums no un datums līdz ir ievadīti un vai tas izdarīts pareizi pēc to tipiem.
2. Tiks izsaukta metode no detalizēto atskaišu sistēmas daļas, lai iegūtu datus no datu bāzes.
3. Datu bāze atgriezīs datu kopu ar atskaites tabulas informāciju par laukiem un to detalizācijas izveides datiem.

Izvaddati

Ja dati pirms attēlošanas nav atlasīti vai atlasīti nepareizi, tiks parādīts kļūdu paziņojums:

1. Ja nav ievadīta struktūrvienība, tiks par to paziņots sistēmas lietotājam;

2. Ja nav ievadīts datuma atlasīšanas tips, tiks par to paziņots sistēmas lietotājam;
3. Ja nav ievadīts gads, tiks par to paziņots sistēmas lietotājam;
4. Ja nav ievadīts mēnesis, tiks par to paziņots sistēmas lietotājam;
5. Ja nav ievadīts datums, no kura tiks veidota atskaite, tiks par to paziņots sistēmas lietotājam;
6. Ja nav ievadīts datums, līdz kuram veidot atskaiti, tiks par to paziņots sistēmas lietotājam.

Visu datu pareizības gadījumā skata sektorā, kas ir zem attēlot pogas, tiks attēlota tabula, kas līdzināsies lejupielādētai atskaites tabulai.

Saskarnes piemērs

Zem datu atlasē logiem attēlota atskaites tabula

Nr.p.k.	Atskaites tabulas lauki	Vērtība
1.	Atskaites tabulas lauks	5
2.	Atskaites tabulas lauks	0
3.	Atskaites tabulas lauks/Atskaites tabulas lauks	1/3
4.	Atskaites tabulas lauks	

3.3.2.1. Att. Tabula ar atskaites laukiem lietotāja saskarnes skatā

3.3.3 Atskaites lauku specifikas, dialoga logu izveidošana

Mērķis

Nospiežot uz jebkura no atskaites laukiem, tiek ielādēts sarunas dialogs, kurā ir attēlota informācija par šī specifiskā lauka datiem tabulas formātā.

Ievaddati

3.3.3.1. tab. Ievaddatu tabula

Nr.	Lauks	Tips	Obligāts	Apraksts
1.	Atskaites lauks	Tabulas lauks kas strādā kā poga	Jā	Lauks, par kuru paredzēts izveidot atskaiti
2.	Atskaite	Izkrītošā izvēlne	Jā	Atskaites tips
3.	Departaments kurš izvēlēts	Izkrītošā izvēlne	Jā	Departaments, par kuru tiek veidota atskaite
4.	Darbinieks kas lieto sistēmu	Tiek iegūta informācija no lietotāja datiem	Jā	Darbinieks, kurš veido atskaites detalizāciju
5.	Laika perioda tips	Radiopogu grupa	Jā	Laika atlasē izvēlētais tips
6.	Gads	Datuma izvēles lauks	Jā, ja ir izvēlēts attiecīgais laika perioda tips.	Gads, par kuru veidot atskaiti
7.	Mēnesis	Datuma izvēles lauks	Jā, ja ir izvēlēts attiecīgais laika perioda tips.	Mēnesis, par kuru veidot atskaiti
8.	Perioda sākuma datums	Datuma izvēles lauks	Jā, ja ir izvēlēts attiecīgais laika perioda tips.	Datums, no kura sākt atlasīt datus
9.	Perioda beigu datums	Datuma izvēles lauks	Jā, ja ir izvēlēts attiecīgais laika perioda tips.	Datums, līdz kuram atlasīt datus
10.	Lappušu numurs	Datuma izvēles lauks	Nē	Lapušu skaits dokumentam (svarīgs tikai 19 atskaitēm)

Apstrāde

Dati tiks apstrādāti un pārsūtīti no viena procesa uz nākošo

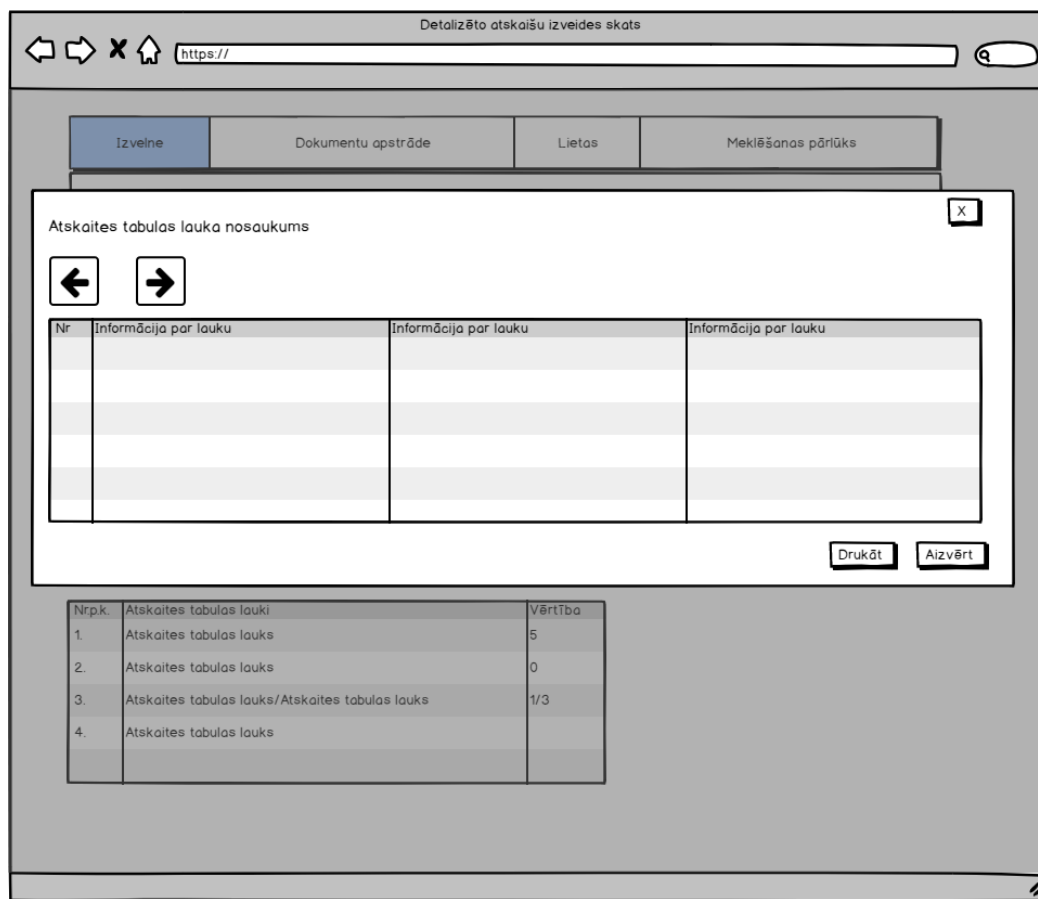
1. Tiks pārbaudīts, vai padodamie dati ir aizpildīti un padodamo vērtību tipi atbilst paredžtajiem datu tiptiem.
2. Tiks izsaukta metode no detalizēto atskaišu sistēmas daļas, lai iegūtu datus no datu bāzes.
3. Datu bāze atgriezīs datu kopu ar detalizētā atskaites lauka datu sarakstu kā teksta rindu un visbeidzot to sašķels atbilstoši detalizācijas tabulai (3.3.3.2. Att).

Izvaddati

Tiks ielādēts sarunas dialogs, kurā būs redzams gan atskaites tabulas lauka nosaukums, gan arī detalizētā informācija par šo lauku, ja tāda pieejama.

Saskarnes piemērs

Attēlā redzams detalizēts atskaites lauks savā dialoga logā.



3.3.3.1. Att. sarunas dialoga logs

Rindas Nr.	Lietas Nr.	Lieta Saņemta	Prokuratūra	Saņemšanas veids
1	123123123	01.05.2017	Metodikas nodaļa	Cits

3.3.3.2. Att. detalizācijas tabula

3.3.4 Atskaites tabulas lauka specifikas lejupielādēšana

Mērķis

Poga detalizētā atskaites lauka, sarunas dialoga logā, kas dod iespēju lejupielādēt detalizēto atskaiti kā XLSX datni tā informācijas vieglākai piekļuvei.

Tiesības

Ir nepieciešamas tiesības piekļūt atskaišu rīkam un attiecīgajai atskaites izveidei, lai būtu iespēja izmatot funkcionalitāti, kas attēlo detalizācijas tabulas laukiem.

Ievaddati

3.3.4.1. tab. Ievaddatu tabula

Nr.	Lauks	Tips	Obligāts	Apraksts
1.	Atskaites lauks	Tabulas lauks kas strādā kā poga	Jā	Lauks, par kuru ir izveidota detalizētā atskaite
2.	Datu kopa ar detalizāciju	Dati iegūti detalizācijas iegūšanas solī	Jā	Detalizētā informācija par konkrēto atskaites lauku

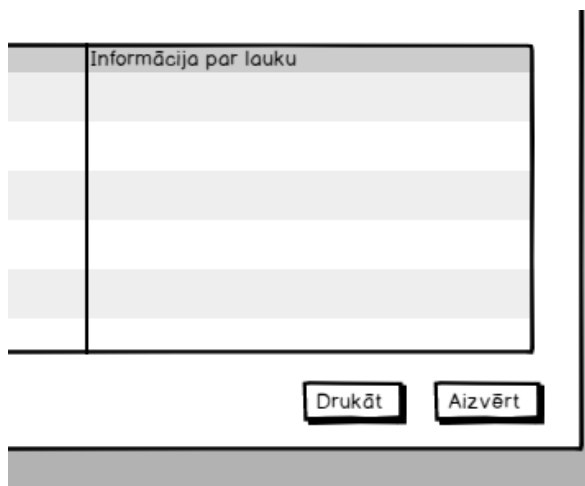
Apstrāde

Ievaddati par detalizāciju tiek padoti uz sistēmu, kur tiek sagatavoti XLSX datnes izveidei. Kad dati ir atgriezti no datu bāzes, tie tiek izmantoti atbilstošas datnes izveidei un lejupielādēšanai.

Izvaddati

Tiek parādīts lejupielādes sarunas dialogs ar datnes nosaukumu, un tiek lejupielādēta XLSX datne ar detalizēto atskaiti.

Saskarnes piemērs



3.3.4.1. Att. Drukāšanas poga lauka detalizācijas logā

3.3.5 Pogu izveide, ar kurām pārslēgties starp detalizācijām

Mērķis

Lai nebūtu katreiz jāiziet no sarunas dialoga, lai atvērtu citu atskaites lauku, tiks izveidotas pogas, ar kurām varēs pārslēgties uz nākošo vai iepriekšējo atskaites lauku.

Ievaddati

Nr.	Lauks	Tips	Obligāts	Apraksts
1.	Atskaites lauks	Tabulas lauks kas strādā kā poga	Jā	Lauks, par kuru ir izveidota detalizētā atskaite

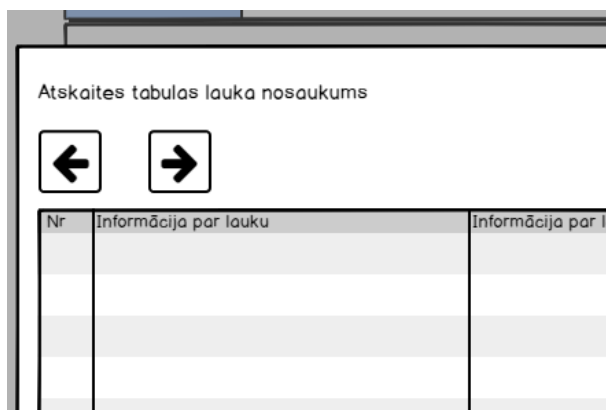
Apstrāde

Nospiežot pogu, tiks pārlādēta informācija padod detalizācijas izveidei vai nu iepriekšējā, vai nākamā lauka Id.

Izvaddati

Tiek ielādēta attiecīgā atskaite, ja tas iespējams. Nav iespējams situācijās, kad lauks ir pirmais vai pēdējais attiecīgajā atskaitē.

Saskarnes piemērs



3.3.5.1. Att. Bultiņu pogas detalizācijas logā

3.3.6 Atskaites laukos, kur vērtību reprezentē vairākas skaitļu vērtības, atskaiti var ielādēt par katru vērtību atsevišķi

Mērķis

Lai varētu iegūt konkrētus datus, kas nebūtu sajaucami, ielādējot atskaites lauka detalizāciju, būs iespējams ielādēt informāciju par konkrēto skaitli atskaites laukos ar vairākām vērtībām.

Ievaddati

3.3.6.1. tab. Ievaddatu tabula

Nr.	Lauks	Tips	Obligāts	Apraksts
1.	Atskaites lauks	Tabulas lauks kas strādā kā poga	Jā	Lauks, par kuru paredzēts izveidot atskaiti
2.	Izvēlētās vērtības numurs	Vērtību laukā izvēlētais skaitlis	Jā	No atskaites vērtības lauka ar vairākām vērtībām izvēlēta skaitļa numurs.
3.	Atskaite	Izkrītošā izvēlne	Jā	Atskaites tips kurš izvēlēts
4.	Departaments	Izkrītošā izvēlne	Jā	Departaments, par kuru tiek veidota atskaite

5.	Darbinieks	Tiek iegūta informācija no lietotāja datiem	Jā	Darbinieks, kurš veido atskaites detalizāciju
6.	Laika perioda tips	Radiopogu grupa	Jā	Laika atlasē izvēlētais tips
7.	Gads	Datuma izvēles lauks	Jā, ja ir izvēlēts attiecīgais laika perioda tips.	Gads, par kuru veidot atskaiti
8.	Mēnesis	Datuma izvēles lauks	Jā, ja ir izvēlēts attiecīgais laika perioda tips.	Mēnesis, par kuru veidot atskaiti
9.	Perioda sākuma datums	Datuma izvēles lauks	Jā, ja ir izvēlēts attiecīgais laika perioda tips.	Datums, no kura sākt atlasīt datus
10.	Perioda beigu datums	Datuma izvēles lauks	Jā, ja ir izvēlēts attiecīgais laika perioda tips.	Datums, līdz kuram atlasīt datus
11.	Lappušu numurs	Datuma izvēles lauks	Nē	Lappušu skaits dokumentam (svarīgs tikai 19 atskaitēm)

Apstrāde

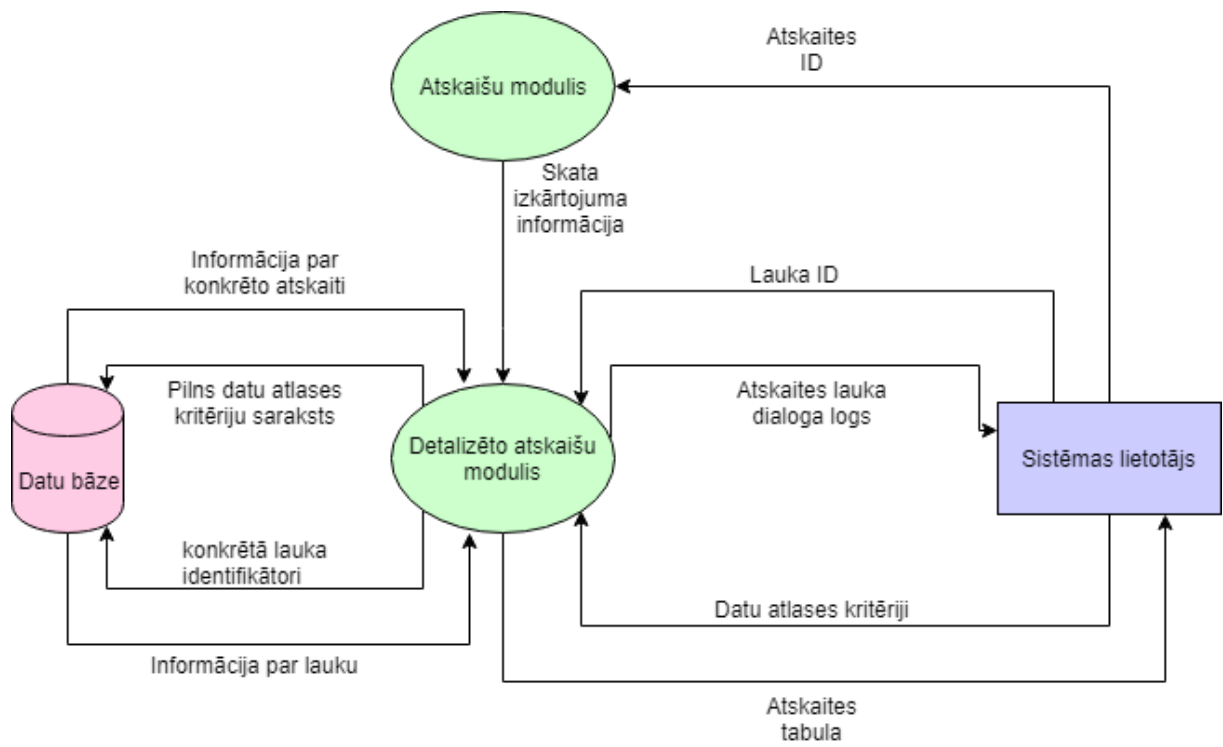
Nospiežot uz konkrētā skaitļa, tiks ielādēts sarunas dialoga logs, bet tikai virsraksta daļā būs nosaukuma daļa, kas attiecas uz izvēlēto skaitli, un ielādētā informācija ir konkrētā par tieši izvēlēto vērtību.

Izvaddati

Tiek atvērts dialoga logs ar informāciju tikai par izvēlēto lauka vērtības daļu.

3.4 Darba konceptuālais modelis

Darba konceptuālais modelis pirmajā izstrādes posmā.



3.4.1. Att. Konceptuālais modelis

3.5 Nefunkcionālās prasības

3.5.1 Veiktspējas prasības

- Programmatūrai jānodrošina darbs vismaz 70 lietotāju vienlaicīgām sesijām;
- Programmatūrai jānodrošina informācijas atlasē laiks, norādot tās meklēšanas parametrus, 90% gadījumu līdz 5 sekundēm.

3.5.2 Drošības prasības

- Sistēmas kļūdu reģistrācijas mehānisms;
- Tiek nodrošināta lietotāja pieslēgumu žurnālēšana;
- Tiek izmantots kopējais tiesību pārbaudes mehānisms.

3.5.3 Citas prasības

Izstrādātajai programmatūrai ir jādarbojas uz aktuālās Firefox versijas;

- Lietojumprogrammatūras ziņojumiem jābūt saprotamā valodā, jāskaidro radušos problēmu būtība;
- Saskaņai jābūt viegli saprotamai un pašpaskaidrojošai;
- Lietojumprogrammatūras procesam jābūt latviešu valodā.

4 PROGRAMMATŪRAS PROJEKTĒJUMA APRAKSTS

4.1 Ievads

4.1.1 Nolūks

Dokuments apraksta projektējumu detalizētu atskaišu izveides, specifiskas informācijas izgūšanas funkcionalitāšu un detalizēto atskaišu lejupielādēšanu, kā XLSX datni, izveidei PROIS programmatūrā.

4.1.2 Darbības sfēra

Dokuments ir darba nodefinēšanai programmētājiem, kā izveidota un strukturēta ir izstrādātā programmatūra.

4.2 Dekompozīcijas apraksts

4.2.1 Moduļu dekompozīcija

4.2.1.1 Atskaites tabulas izveides modulis (GenerateDetails)

Modulis paredzēts, lai sistēmas lietotājs iegūtu vizuālu reprezentāciju atskaites tabulai. Tabula tiks izveidota tikai situācijā, kur visi datu atlasas lauki ir aizpildīti ar atbilstošu informāciju.

4.2.1.2 Detalizētās informācijas atlasas modulis (FieldDetails)

Modulis paredzēts, lai informāciju, kas ir apzīmēta ar vērtības lauka skaitli, varētu attēlot saskarnes skatā kā dialoga logu. Šī moduļa ietvaros tiek izgūta informācija no datu bāzes. No viena detalizācijas lauka uz nākošo vai iepriekšējo var pārslēgties ar bultiņu pogām.

4.2.1.3 Detalizācijas lejupielādēšanas modulis (PrintField)

Modulis paredzēts, lai detalizācijas informāciju varētu iegūt ne tikai saskarnes skata vidē, bet arī lejupielādēta XLSX dokumenta formātā. Atskaites detalizācijas XLSX datne tiek nosaukta par “Detalizētais_atskaites_lauks_<ddmmgggg>”, kur <ddmmgggg> ir atskaites veidošanas datums.

4.2.2 Procesu dekompozīcija

4.2.2.1 Atskaites lauku apkopotājprocess (ReportFields)

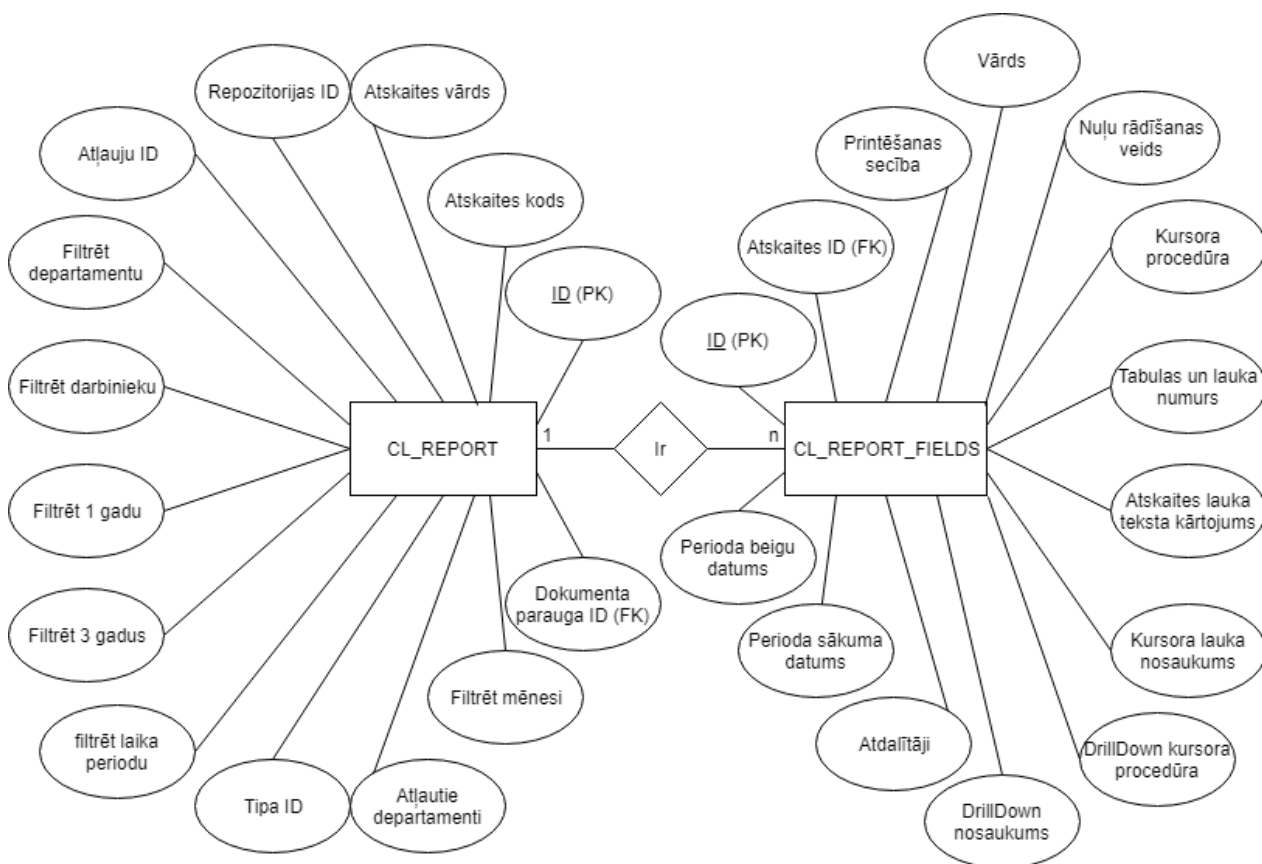
Process iekļauj divu līmeņu darbības: pirmajā tiek izgūti dati no lauku detalizācijas tabulas, kas korespondē ar datu atlasas datiem, tādiem kā atskaites veidu un tiesībām, un otrajā līmenī tā izgūst datus par atskaites lauku vērtībām. Abas datu kopas tiek atgrieztas kā rezultāts.

4.2.2.2 Lauka detalizācijas ieguve (FieldDetails)

Šajā procesā sākotnēji tiek pārbaudīti dati, kas saistīti ar datu atgriešanu, tādi kā lietotāja identifikācija, pareizu datu lietošana un nepieciešamās informācijas esamība. Kad pārbaudīti dati, tiek izmantots atskaites DrillDown procedūra. Tekstuālajam mainīgajam tiek aizstāti teksta fragmenti ar to lokācijai korespondējošajām vērtībām, kā arī komplicētu struktūru gadījumā tiek vērtība sašķelta pa atbilstošajiem fragmentiem to individuālai izsaukšanai.

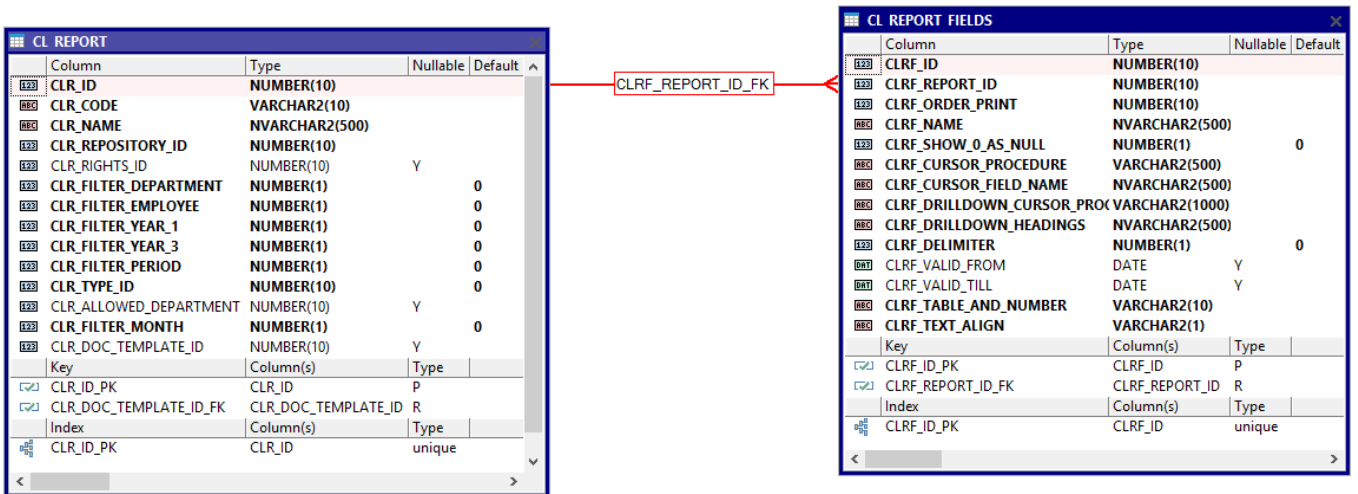
4.2.3 Datu dekompozīcija

4.2.3.1 Konceptuālais entītiņu relāciju (ER) modelis



4.4.2.1. Att. Konceptuālais ER modelis

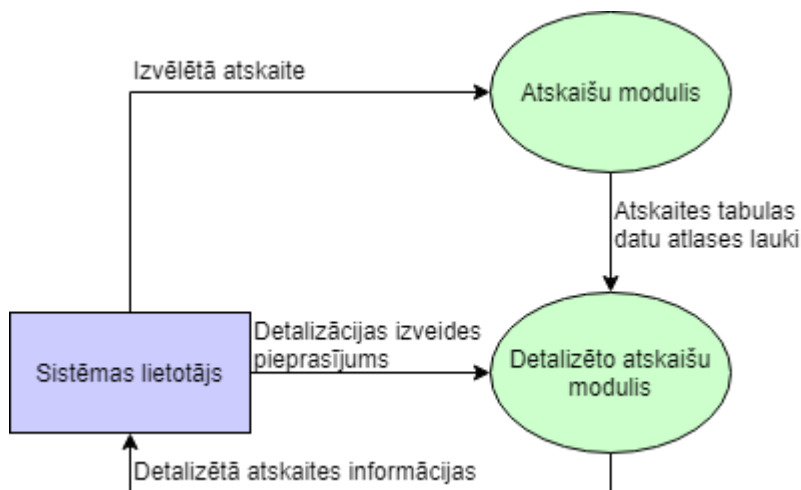
4.2.3.2 Realizācijas entītiņu relāciju (ER) modelis



4.4.3.1. Att. Realizācijas ER modelis

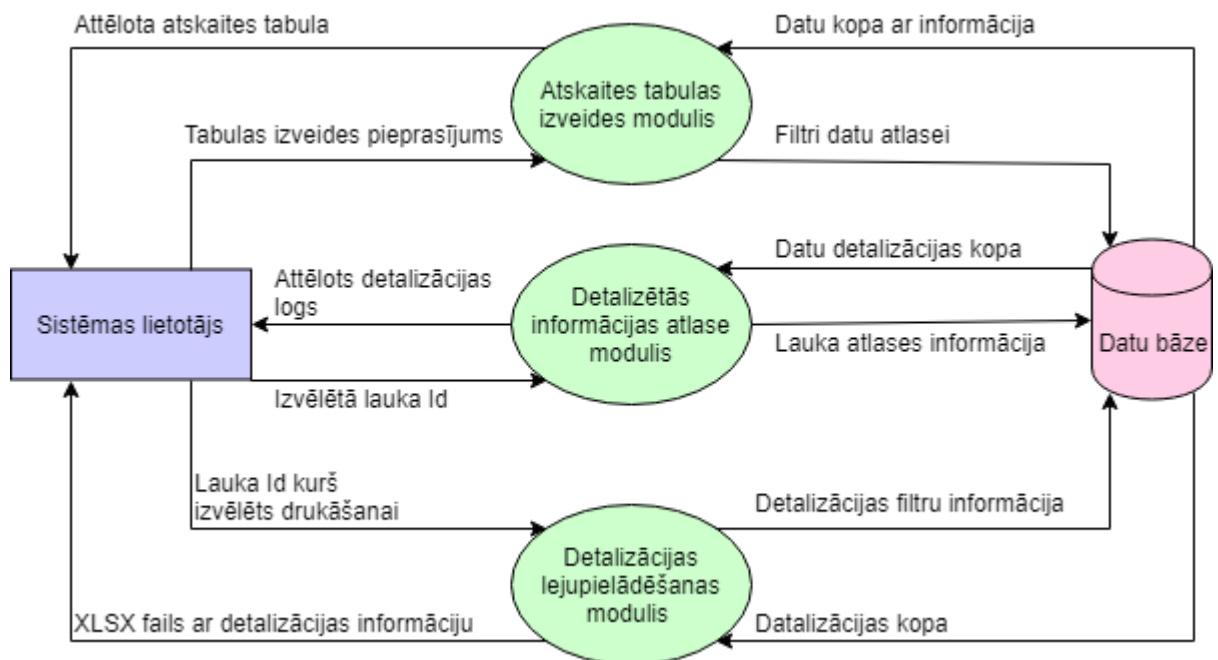
4.3 Datu plūsmas diagrammas

4.3.1 Nulles līmeņa datu plūsmas modelis



4.3.1.1. Att. Nulles līmeņa datu plūsmas diagramma

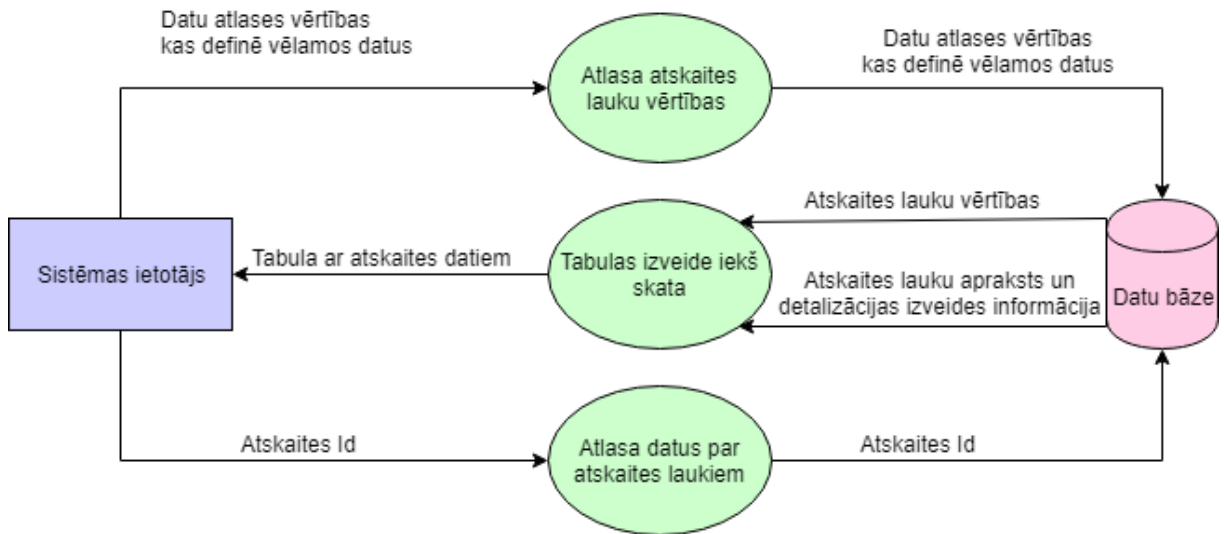
4.3.2 Pirmā līmeņa datu plūsmas modelis Detalizēto atskaišu moduļim



4.3.2.1. Att. Pirmā līmeņa datu plūsmas diagramma

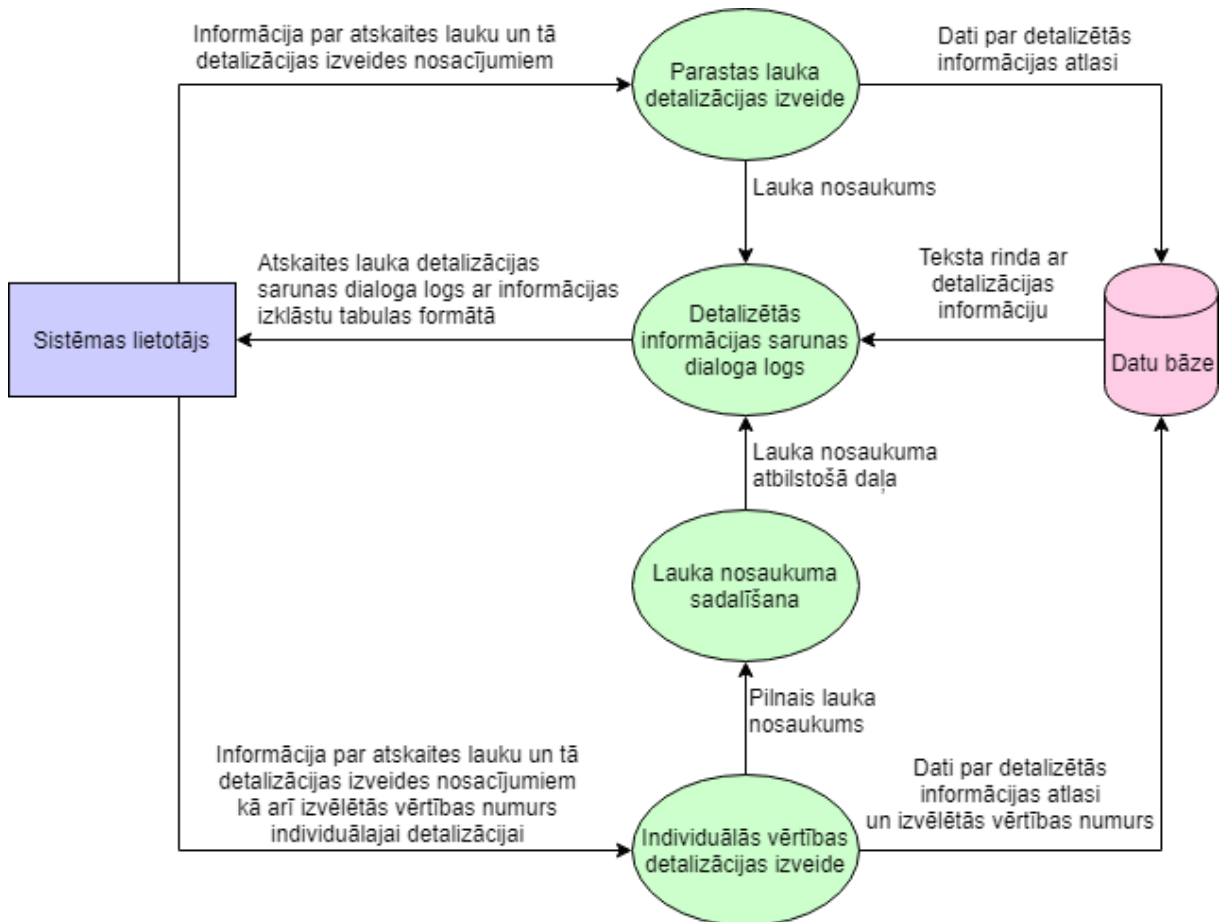
4.3.3 Otrā līmeņa datu plūsmas modeļi

4.3.3.1 Otrā līmeņa datu plūsmas modelis Atskaites tabulas izveides funkcionalitātei



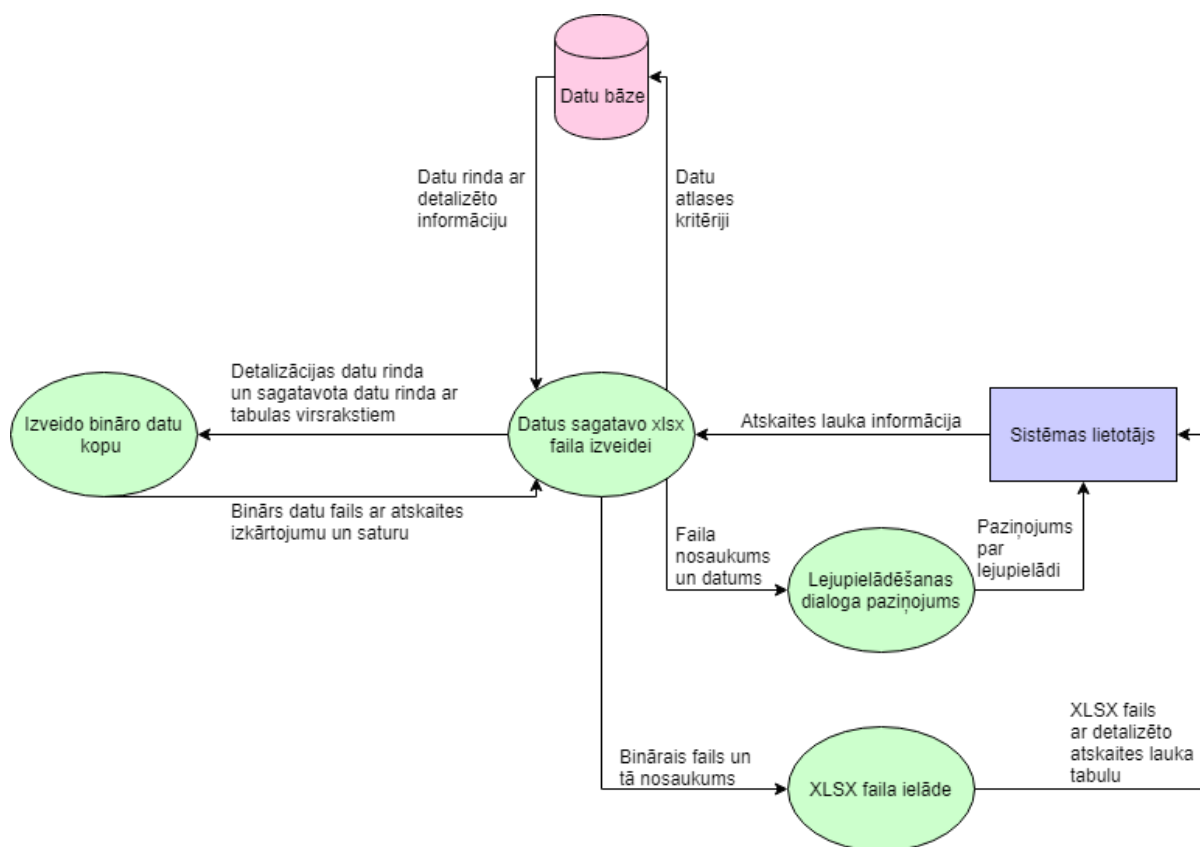
4.3.3.1.1. Att. Otrā līmeņa datu plūsmas diagramma

4.3.3.2 Otrā līmeņa datu plūsmas modelis detalizētās informācijas dialoga loga izveides funkcionalitātei



4.3.3.2.1. Att. Otrā līmeņa datu plūsmas diagramma

4.3.3.3 Otrā līmeņa datu plūsmas modelis Atskaites detalizācijas XLSX datnes lejupielādes funkcionalitātē



4.3.3.3.1. Att. Otrā līmeņa datu plūsmas diagramma

4.4 Datu bāzes diagrammas

4.4.1 Datu bāzes tabulu projektējums un apraksts

4.4.1.1 Tabula CL_REPORT

Atskaišu tabula nav autora veidota, bet gan bija jau PROIS sistēmā atskaišu moduļa datu bāzes struktūrā un tika iekļauta šajā darbā, jo tika izmantota datu atlasei atskaišu tabulu lietotāja saskarnes formas izveidē, un datu atlasei. Šajā tabulā tiek glabāta informācija par atskaišu veidiem un to, kādi kritēriji tiek ņemti vērā, veidojot konkrēto atskaiti. Šajā tabulā jaunus ierakstus var veidot tikai sistēmas izstrādātāji/administratori no datu bāzes puses.

4.4.1. tab. CL_REPORT

Kolonnas nosaukums	Datu tips	Papildus informācija	Apraksts
CLR_ID	NUMBER(10)	PK / AI	Identifikātors
CLR_CODE	VARCHAR2(10)		Atskaites veida kods
CLR_NAME	NVARCHAR2 (500)		Atskaites veida nosaukums
CLR_REPOSITORY_ID	NUMBER(10)		Atskaites Excel sagataves Repozitorija Id
CLR_RIGHTS_ID	NUMBER(10)	NULLABLE	Atskaites veidošanas tiesību Id
CLR_FILTER_DEPARTMENT	NUMBER(1)	DEFAULT '0'	Filtrs, kas attēlo, vai vajadzēs izvēlēties departamentu atskaites izveidei
CLR_FILTER_EMPLOYEE	NUMBER(1)	DEFAULT '0'	Filtrs, kas attēlo, vai vajadzēs izvēlēties darbinieku atskaites izveidei
CLR_FILTER_YEAR_1	NUMBER(1)	DEFAULT '0'	Filtrs, kas attēlo, ja atskaite paredzēta viena gada laika periodam
CLR_FILTER_YEAR_3	NUMBER(1)	DEFAULT '0'	Filtrs, kas attēlo, ja atskaite paredzēta trīs gadu laika periodam

CLR_FILTER_PERIOD	NUMBER(1)	DEFAULT '0'	Filtrs, kas attēlo, ka atskaiti var veidot ar laika formātu no datuma līdz datumam
CLR_TYPE_ID	NUMBER(10)	DEFAULT '0'	Atskaites veids attēlošanai sarakstā
CLR_ALLOWED_DEPARTMENT	NUMBER(10)	NULLABLE	Lauks, kur norāda, kuras struktūrvienības darbiniekus varēs rādīt atskaites ģenerēšanas darbinieku izvēles logā
CLR_FILTER_MONTH	NUMBER(1)	DEFAULT '0'	Filtrs, kas attēlo, ka atskaiti var veidot par konkrētu mēnesi
CLR_DOCUMENT_TEMPLATE_ID	NUMBER(10)	FK / NULLABLE	Ārējā atslēga, kas piešķir to, kāda veida dokuments tiks veidots

4.4.1.2 Tabula CL_REPORT_FIELDS

Tabula satur informāciju par atskaites laukiem un lauku detalizācijām. Šajā tabulā jaunus ierakstus var veikt tikai sistēmas izstrādātāji/administratori no datu bāzes puses.

4.4.2. tab. CL_REPORT_FIELDS

Kolonnas nosaukums	Datu tips	Papildus informācija	Apraksts
CLRF_ID	NUMBER(10)	PK / AI	Identifikators
CLRF_REPORT_ID	NUMBER(10)	FK	Ārējā atslēga, kas norāda, kurai atskaiti šis lauks pieder
CLRF_ORDER_PRINT	NUMBER(10)		Lauku vērtības tiek izmantotas kā lauku secības numuri, sākot no mazākā un pakāpeniski pieaugot vērtībā.
CLRF_NAME	NVARCHAR2(500)		Atskaites lauka nosaukums
CLRF_SHOW_0_AS_NULL	NUMBER(1)	DEFAULT '0'	Filtrs, kas parāda, kādā veidā lauku ar nulles vērtību vajadzēs attēlot

CLRF_CURSOR _PROCEDURE	VARCHAR2(500)		Atskaites datu ģenerēšanas funkcijas izsaukums
CLRF_CURSOR _FIELD_NAME	NVARCHAR2(500)		Lauka atskaites vietai tabulā nosaukums
CLRF_DRILLD OWN_CURSOR _PROC	VARCHAR2(1000)		Saglabāts specifisks funkciju izsaukums teksta formātā datu atlasei par attiecīgo atskaites lauku, kurā koda vārdus aizstāj ar padotajām vērtībām (skatīt 3. pielikumā)
CLRF_DRILLD OWN _HEADINGS	NVARCHAR2 (500)		Atskaites lauka parādītās informācijas virsraksti
CLRF_DELIMIT ER	NUMBER(1)	DEFAULT '0'	Laukā, kurā ir vairākas vērtības, tiek norādīts, cik ir atdalītāju
CLRF_VALID_F ROM	DATE	NULLABLE	Norāda, no kura datuma lauka vērtība ir pieejama
CLRF_VALID_T ILL	DATE	NULLABLE	Norāda, līdz kuram datumam lauka vērtība ir pieejama
CLRF_TABLE_ AND_NUMBER	VARCHAR2(10)		Atskaites lauka numurs un tabulas piederības numurs
CLRF_TEXT_A LIGN	VARCHAR2(1)		Atskaites lauka novietojums laukā ar kārtojumu pēc labās vai kreisās puses, vai pa vidu.

4.5 Lietotāja ekrānformu projektējums

4.5.1 Vispārējs apraksts Ekrāna formu izveidē

Ekrānformas tika projektētas, lai iekļautos kopējā stilā, un konsultējoties ar uzņēmuma pārstāvi. Formas tika projektētas, lai atsauktos uz parastās atskaites (skatīt 1. pielikumā) izveides procesu un tuvi līdzinātos pašas atskaites izskatam, attēlojot to lietotāja saskarnes skatā.

Lietotāju ievadīto datu pārbaude notiek gan lietotāja pusē (Angular JavaScript), gan servera pusē.

4.5.2 Lietotāja pieejas kontrole saglabā tādas pašas tiesību piešķiršanas kā tās, kas piešķirtas atskaišu modulī.

4.5.3 Ekrāna formu apraksti

4.5.3.1 Nokļūšana līdz detalizēto atskaišu izveides skatam

4.5.2.1.1. Att. Poga detalizēts

Šajā cilnē ir atskaišu izveides skats (4.5.2.1.1.att 3.), kurā ir redzams saraksts ar atskaišu veidiem, ko ir iespējams izveidot (4.5.2.1.1.att 2.). Pie tām atskaitēm kurām ir sagatavotas lauku detalizācijas ir poga (Detalizēts) (4.5.2.1.1.att 1.).

4.5.3.2 Detalizēto atskaišu izveides skats

1. Attēlot Drukāt Saglabāt atskaiti kā dokumentu Piesaisīt atskaiti dokumentam

2. Struktūrvienība: Izvēlētās struktūrvienību



4.5.2.2.1. Att. Neaizpildīta atskaites izveide

Skatā redzams jaunais detalizētās atskaites skats vēl bez veiktām darbībām. Ir attēlots tas pats datu atlasē logs (4.5.2.2.1.att. 2.), kas būtu izmantots parastas atskaites (skatīt 1. pielikumā) izveidošanai. Atšķirībā no parastas atskaites izveides loga ir pievienota poga "Attēlot" (4.5.2.2.1.att 1.), kuru, nospiežot pēc attiecīgo datu atlasē lauku informācijas, ievades tiks attēlota atskaites tabula.

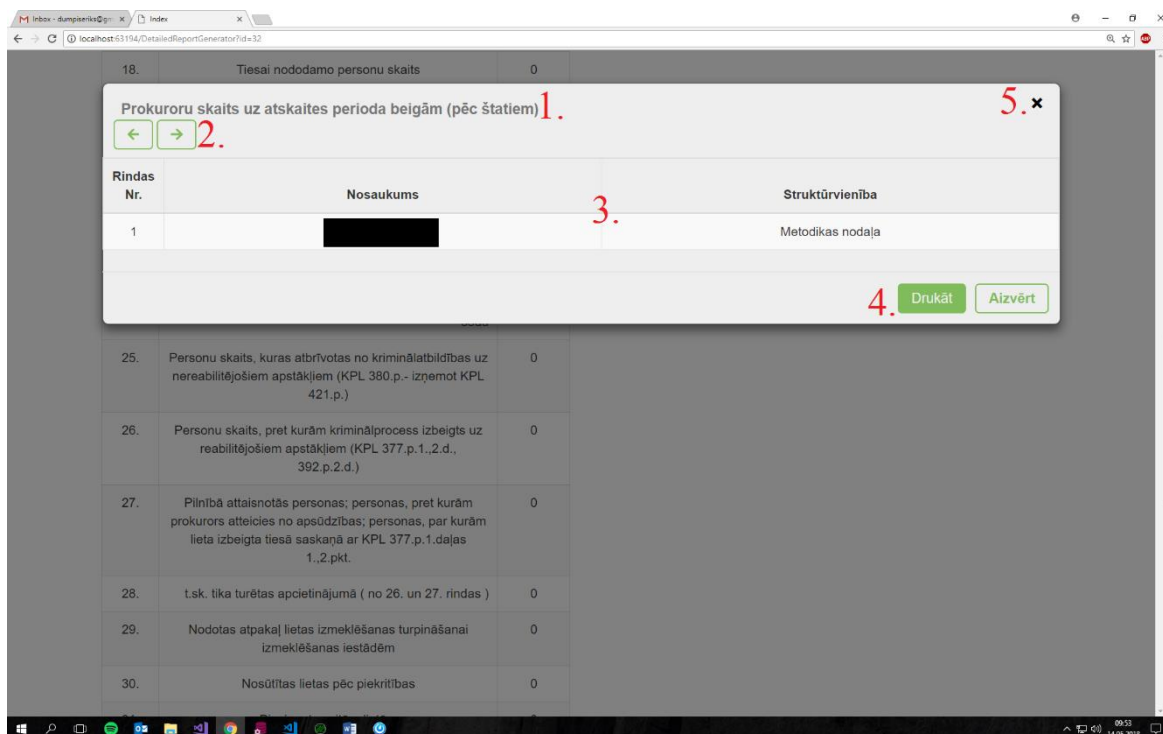
4.5.3.3 Skats ar attēlotu atskaites tabulu lietotāja saskarnes vidē

Nr.p.k.	Nosaukums	Skaitis
1.	Nepabeigto lietu atlikums uz gada sākumu	1
2.	Pieņemtas izmeklēšanā lietas	1
3.	lietas, kuras saņemtas kriminālvajāšanas uzsākšanai / t.sk. kurās izmeklēšana pabeigta 1 mēn. laikā no kriminālprocesa uzsākšanas brīža	1 / 0
4.	lietu skaits pēc sadalīšanas / izmeklēšanas stadijā KPL 38.p.3.d. kārt.	0 / 0
5.	lietas, kuras saņemtas no tiesas trūkumu novēršanai	0
6.	lietas, kuras saņemtas no tiesas pārkāpumu novēršanai	0
7.	pārņemtas izmeklēšanai KPL 37.p.2.d. 7.pkt. kārt.	0
8.	lietas, kuras saņemtas no citām prokuratūras iestādēm	0
9.	atjaunotas no agrāk izbeigtajām vai apturētajām	0
10.	t.sk. no izbeigtajām, nosacīti atbrīvojot no kriminālatbildības	0
11.	Tiesai nodoto lietu skaits	0
12.	vispārējā kārtībā	0
13.	saisinātā procesa kārtībā	0

4.5.2.3.1. Att. Attēlota atskaite lietotāja saskarnes vidē

Pēc tam, kad atskaite ir attēlota skatā kā tabula (4.5.2.3.1.att 1.), kas ir veidota pēc parastas atskaites (skatīt 1. pielikumā) līdzības. Tabula sastāv no trīs kolonnām – Rindas numura, atskaites lauka nosaukuma, ko tas reprezentē, un skaita kolonu (4.5.2.3.1.att 5. 2. 6.). Kolonas vērtība var būt attēlota kā parasta vērtība (4.5.2.3.1.att 3.) tādā gadījumā, nospiežot uz paša lauka vai vērtības, detalizācijas logā nebūs nekādu atšķirību, bet laukos, kuros ir vairākas vērtības atdalītas ar slīpsvītru (4.5.2.3.1.att 4.), nospiežot uz katras no vērtībām, tiks attēlots detalizācijas dialogs par attiecīgo vērtību.

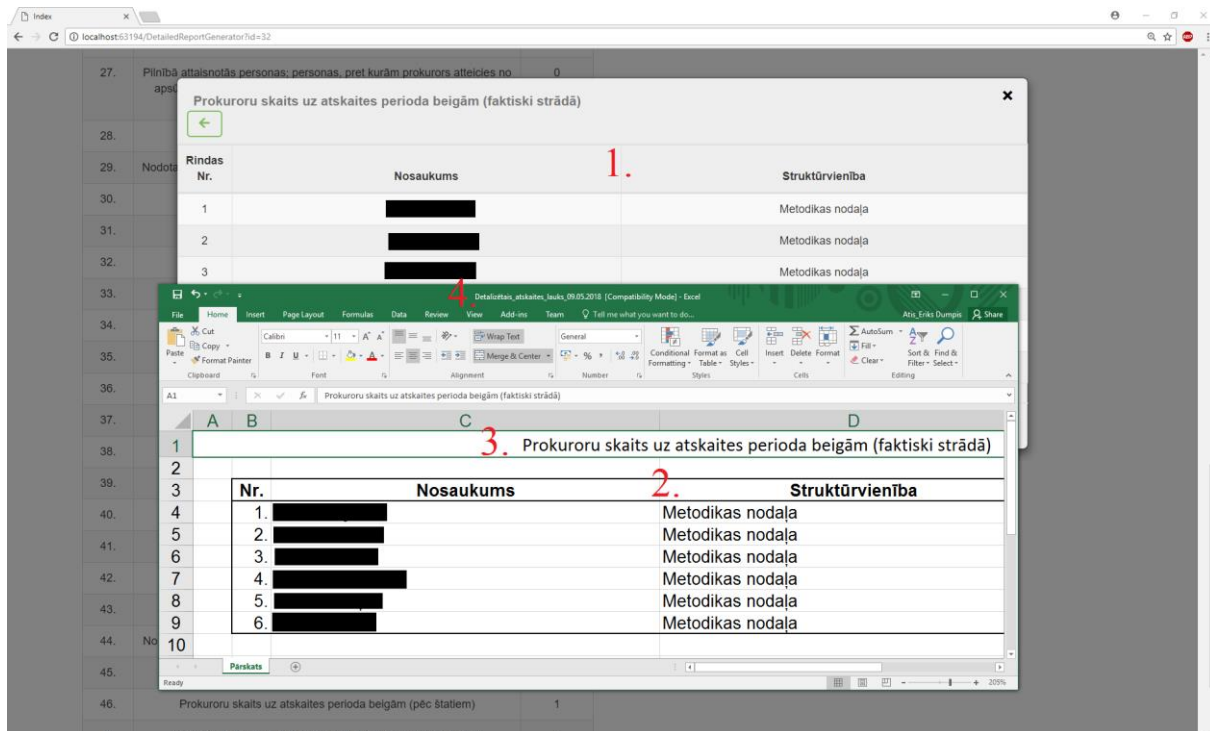
4.5.3.4 Ielādēts detalizācijas dialoga logs



4.5.2.4.1. Att. Tabulas lauka detalizācija

Skatā ir jau attēlots detalizācijas dialoga logs (4.5.2.4.1.att 5.). Šajā logā ir izvēlētā lauka nosaukums vai nosaukuma daļa, ko reprezentē individuālā vērtība laukā ar vairākām vērtībām (4.5.2.4.1.att 1.). Zem virsraksta ir divas pogas, uz vienas no tām ir bulta, kas vērsta pa kreisi, un uz otras ir bulta, kas vērsta pa labi (4.5.2.4.1.att 2.), šīs pogas ļauj pārslēgties no viena lauka uz nākošo vai iepriekšējo, neizverot pašu dialoga logu. Logā ir tabula ar detalizācijas informāciju (4.5.2.4.1.att 3.), kas sastāv no vairākām kolonām ar informāciju atbilstoši laukam. Lauka apakšējā labajā stūrī ir poga drukāt (4.5.2.4.1.att 4.), kuru nospiežot tiks ielādēts XLSX datne.

4.5.3.5 Lejupielādēts XLSX datne



5.5.2.5.1. Att. Lejupielādētā atskaites detalizācija

Šajā ekrānformā ir attēlota lejupielādēta detalizāciju atskaite (4.5.2.5.1.att 4.). XLSX datnes atskaites informācija sakrīt lietotāja saskarnes atskaites (4.5.2.5.1.att. 1.) informācijai. Lejupielādētā atskaite sastāv no virsraksta (4.5.2.5.1.att 3.), kas ir atbilstošā lauka nosaukums un šī lauka detalizāciju tabulas (4.5.2.5.1.att 2.).

5 PROJEKTA ORGANIZĀCIJA

Kā projekta tapšanas dzīves cikls tika izmantots ūdenskrituma modelis. Uzdevumi, kas paredzēti darba izveidei, tika reģistrēti uzdevumu organizēšanas un projektu vadības sistēmā Redmine un tiek izkārtoti darbu sarakstā. Balstoties uz saņemtajiem uzdevumiem, tika veidota uzdevuma dokumentācija un vēlāk arī īstenošana.

Uzdevumu aprakstos Redmine sistēmā tika sniegta informācija par uzdevumā veicamajām darbībām, tam paredzamo izpildes laiku, darba uzsākšanas datumu un tā gala termiņu, kā arī darba nozīmīgumu.

Izdarot kādu no Redmine uzdevumiem, tas ir jāatzīmē kā izpildīts un jānorāda patērētais laiks risinājuma izveidei. Tālāk uzdevums nonāk testēšanā, kur tiek pārbaudīta risinājuma atbilstība uzdevuma aprakstam, ātrdarbībai, definētajām prasībām un kļūdu iespējamības. Pēc testēšanas ir divas iespējas, pirmkārt, ja nav fiksēta neviena kļūda, uzdevums tiek nodefinēts kā atrisināts, bet, ja ir kādas nepilnības, tās tiek aprakstītas un izveidots apakšuzdevums testētajam uzdevumam ar labojuma vai problemātiskās situācijas aprakstu.

Projekta izstrādei tika izmantoti divi izstrādes rīki: PL/SQL Developer un Microsoft Visual Studio 2017

Visa datu bāzes puses veiktā izstrāde tika veikta PL/SQL Developer vidē. Tajā tiek veidotas tabulas datu glabāšanai un procedūras, un funkcijas, ko varēs izsaukt no aplikācijas vides ar servisiem. Šīs vides izmaiņas, lai nonāktu testa vidē, ir jāuzstāda pakāpeniski to izveides secībā, un šo izmaiņu pievienošana produkcijas videi, tās ir tādā pašā procesā jāimplementē no testa vides.

Microsoft Visual Studio 2017 vidē tika izmantoti trīs darbības apgabali. Kā pamatprojekts lietotāja saskarnes izveidei tika izmantots Zzdats.ProIS.Report.Mvc.Site, kurā tika veidots vizuālais noformējums un lietotāja saskarne, ko izmantot sistēmas lietotājam. Lai nodrošinātu datu iegūšanu no datu bāzes, tika izmantots Zzdats.ProIS.Report.Services, un, lai kods būtu sakārtots, tika izmantots Zzdats.ProIS.Cdn, kas projektā satur visus konstantos skriptus, css un citus, šī sistēmas daļa palīdz ātrdarbībai. Lai šīs vides veiktās izmaiņas publicētu testēšanas vidē, tās atliek tikai nopublicēt no pašas vides.

6 DARBIETILPĪBA

Darbietilpības aprēķināšanai tika izmantota ekspertu metode. Tika izvērtēti darba pienākumi no darba autora un nozares speciālista, projekta PROIS jomā, viedokļa. Tika izmantota arī informācija, kas apkopota Redmine sistēmā. Pabeidzot darbu un to klasificējot kā pabeigtu, jānorāda darbam patērētais laiks, un līdz ar to kopējai darbietilpības aprēķināšanai nepieciešams visu Redmine reģistrēto uzdevumu kopējā laika summa un dokumentācijas izveidei patērētais laiks.

6.1 Dokumenta Darbietilpības novērtējums

Uzdevuma Nosaukums vai numurs	Patērētais laiks (h) Autors	Patērētais laiks (h) Speciālists	Patērētais laiks (h)
Iepazīšanās ar izstrādes vidēm un projektā pielietojamajām tehnoloģijām	80	80	80
PPS izveide	60	60	60
PPA izveide	80	60	70
4675: poga “(detalizēts)”, kas ved uz jauno skatu	20	8	16
4712: Atskaites izvade Detalizēto atskaišu skatā	32	40	32
4678: Pipelined funkcijau izveide	24	40	18
4679: Datu bāzes tabulu luku izveide katram atskaites laukam	64	60	72
4715: Atskaites lauku detalizācijas logu izveide	20	16	16
4720: Pogas ar kurām pārvietojas starp atskaites laukiem	8	8	8
4724: Papildinājumi atskaites tabulā	8	8	8

4725:Lauku detalizācijas var lejupielādēt	40	60	80
4731: Laukus ar vairākām vērtībām var ielādēt par katru vērtību	16	16	12
Pārējo Kvalifikācijas darba nodaļu izveide	30	40	30
Kopā	482	496	502

Darba novērtējums pirms tā izpildes tika novērtēts gan no darba autora puses, gan speciālista puses un, spriežot par darba paredzēto apjomu salīdzinājumā ar reālo laika patēriņu, var spriest, ka profesionāla viedoklis bija akurātāks.

Cilvēkmēnešu skaitu var iegūt pēc vienkārša aprēķina, kur:

1 cilvēkmēnesis = 20 cilvēkdienas un 20 cilvēkdienas = 160 cilvēkstundas, ja vadās pēc aprēķina, kur pieņemts, ka vidējais darba dienu skaits cilvēkmēnesī ir 20 cilvēkdienas.

Ņemot vērā cilvēkmēnešu aprēķinu un darba uzdevumiem patērēto laiku, var spriest, ka darba izstrādei pavadīti 3.1 cilvēkmēnesis.

7 KONFIGURĀCIJU PĀRVALDĪBA

7.1 Izstrādes vide

Izstrādes vide ir kompilēta izmantojot Microsoft Visual Studio 2017. Izstrādes vidē notiek pirmās kārtas izmaiņu veikšana. Šajā vidē strādā programmētāji, veidojot programmkodu un to versionējot ar versiju kontroles rīku Git, un, lai jaunākās koda versijas nogādātu testa vidē, tad attiecīgā moduļa daļa, kurā veiktas izmaiņas, ir jāpublicē izvēlētajā vidē.

7.2 Testa vide

Testa vide ir jau gatava (nokompilēta) programmatūras versija. Testa videi var piekļūt testētāji no sistēmas izstrādes komandas, kā arī lietotāji, kuri veic jaunās versijas testēšanu. Viena no sistēmām, kas tiek izmantota projekta jaunākās versijas uzturēšanai un atjaunināšanai, ir Jenkins, kas paņem no versiju kontroles sistēmas GIT aktuālo programmatūras kodu. Šī sistēma automātiski palaiž vienībtestus un, ja tie noritējuši pareizi, tad tiek uzkopēta nokompilēto versiju uz testa vides servera.

7.3 Akcepttestēšanas vide

Jau gatavā testa vidē tiek pārskatītas visas funkcionalitātes, pārbaudot, vai tās ir atbilstošas tam, kā tās definētas specifikācijā, izmaiņu pieprasījumā vai kļūdas aprakstā. Ja visas uzdevuma izpildes prasības ir apmierinošas, tad sistēmas daļa tik nodota tālāk.

7.4 Klienta akcepttestēšanas (pirmsprodukcijas) vide

Kad tiek uzlikta jauna versija vai atjauninājums sistēmai un tā tiek nodota klientam, viņu puses darbinieki to notestē arī no savas puses. Lietotāju puses testētāji pārbauda, vai viņu sistēmas versijā visas prasības un sistēmas piedāvātās funkcionalitātes ir funkcionējošas un atbilstošas.

7.5 Produkcijas vide

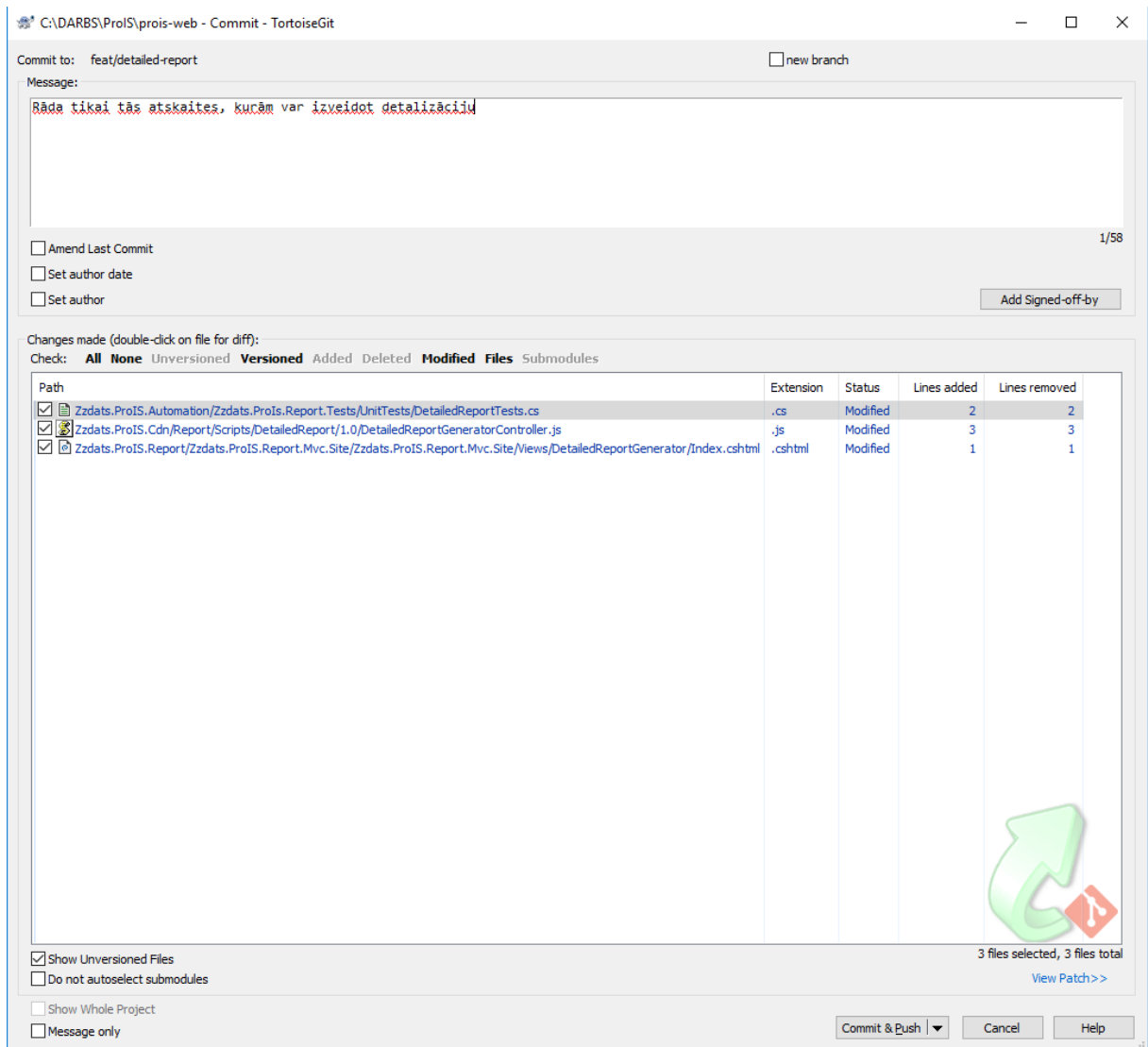
Kad sistēmas funkcionalitātes notestētas no izstrādātāju puses, tā tiek palaista klientu lietošanai, un no tā brīža uzskatāms, ka kods ir lietošanas stadijā.

7.6 Programmkoda konfigurācijas pārvaldība

Kā versiju kontroles sistēma tika izmantota projektā PROIS izmantotā. Sākotnēji, uzsākot darba izstrādi, tika izmantots TFS kā versijkontroles sistēma, bet projekta izstrādes laikā notika






pāreja uz Git visā PROIS sistēmā. Pārejas rezultātā tika izmantotas abas versionēšanas sistēmas. Kopš Git izmantošanas sākuma tika arī izmantota lietojumprogramma TortoiseGit.

Apakšā redzamajā bildē ir kļūdas labojuma pievienošana versijkontroles sistēmai. Bildē ir redzams komentāra lauks ar īsu aprakstu par veiktajām izmaiņām un zemāk ir datņu saraksts, kur ir notikušas koda izmaiņas.



7.6.1. Att. TortoiseGit kļūdu labojuma pievienošana

Apakšējā attēlā redzams versionēšanas sistēmas GIT labojumu un papildinājumu pievienošanas vēstures kataloga fragments. Izmaiņas tiek pievienotas atbilstošajā brīdī, kad tās veiktas, lai neradītu situāciju, kad tiek kods pievienots nepareizā secībā, radot konfliktējošas situācijas vai kodu, kurš ir nefunkcionējošs

Autors	SHA1	Ziņojums	Datums
 atis	ade3559ba7	Rāda tikai tās atskaites, kurām var izveidot detalizāciju	1 stundu atpakaļ
 atis	2df43c9026	Pielikts publish profils gitignore failā	1 nedēļu atpakaļ
 atis	99b867810b	Bultiņas dialoga logā bultiņas pirmajā un pēdējajā atskaites laukā nerādas...	1 nedēļu atpakaļ
 atis	44bee42218	Pievienoti komentāri un pievienota repor type pazīme vai ir detalizēta ats...	3 nedēļas atpakaļ
 atis	8349e3db34	Komentāri koda funkcijām un izdzēsta dublējošā klase.	3 nedēļas atpakaļ

7.6.2. Att. Programmkoda pievienošanas vēstures fragments

8 KVALITĀTES NODROŠINĀŠANA

Darbs tiek izstrādāts uzņēmumā SIA “ZZ Dats”, kuram ir iegūti ISO 9001 un ISO 27001 sertifikāti. Tas ir veidots, ievērojot iekšienē izstrādātas vadlīnijas, kas nodrošina programmēšanas labo praksi.

Darbā tiek ievērotas šādas vadlīnijas iezīmes:

1. Mainīgo nosaukumi tiek veidoti vienotā stilā ar iepriekš izveidotajiem mainīgajiem. Visbiežāk vienā no diviem stiliem – “camelCase” (pirmais vārds nosaukumā sākas ar mazo burtu, pārējie ar lielo) un “PascalCase” (visi vārdi nosaukumā rakstīti ar lielo burtu).
2. Kods ir strukturēts. Mainīgo nosaukumus veido aprakstošus, lai uzreiz saprotams, ko tie reprezentē (piemēram. ”getDetailedField” nevis “button2”).
3. Komentāri tiek rakstīti latviešu valodā, un tie tiek rakstīti pirms katras funkcijas, procesa un citām programmatūras struktūrām. Tiek aprakstīti arī mainīgie, kas nav viegli atpazīstami to satura dēļ vai to neparedzētā pielietojuma dēļ. Komentārus raksta arī pēc autora ieskatiem, paskaidrojot vietas, kas autoram likušās sarežģītas. DB struktūru komentārus raksta pie izmaiņu objekta brīdī, kad tiek pabeigtas izmaiņas.
4. Tiek izmantots versiju kontroles rīks Git un programma TortoiseGit
5. Testēšana tiek veikta trīs līmeņos:
 - a) Pirmajā līmenī programmētājs pats veic testēšanu pēc akcepttestēšanas prasībām, lai pārbaudītu, vai izstrādātā programma strādā bez kļūdām;
 - b) Starp līmeņiem, kur testēšanu veic cilvēks, notiek arī vienībtestu palaišana no automatizācijas sistēmas Jenkins, un, ja visi testi iziet bez problēmām, tad attiecīgā programmatūras daļa tiek nodota testa videi.
 - c) Otrā līmeņa testēšanu veic testētājs, kurš pārbauda programmatūru atbilstoši projekta prasību specifikācijai. Pēc testēšanas, ja tiek konstatētas kļūdas, uzdevums tiek atgriezts programmētājam ar paskaidrotām kļūdām un prasībām, kas nav pilnībā, īstenotas;
 - d) Trešajā līmenī klients pirms jaunās versijas uzlikšanas veic testēšanu atbilstoši akcepttesta prasībām, ja tiek atklāta kāda nepilnība, par to tiek paziņots izstrādes komandai, un atbildīgais programmētājs veic nepilnību labojumus.
6. Kods ir veidots saprotamā veidā, neveidojot pārāk garas koda rindas, tās sadalot pieņemama garuma fragmentos un komentējot visas koda struktūras un mainīgo vērtības, kas nav pirmajā momentā saprotamas.

9 TESTĒŠANAS DOKUMENTĀCIJA

9.1 Ievads

Dokumentā tiek aprakstīti testi, lai atrastu un novērstu programmēšanā radušās kļūdas. Pēc programmatūras daļas izveides tā tiek testēta, vai tā ir atbilstoša programmatūras prasību specifikācijā minētajām prasībām. Konstatētās kļūdas testēšanas procesa laikā tiek novērstas. Pēc labojumu veikšanas tie tiek atkārtoti testēti, lai pārliecinātos par labojuma efektivitāti un pareizību. Programmatūra tika testēta gan ar vienībtestiem, gan ar funkcionālajiem lietotnes testiem.

9.2 Testēšanas žurnāls

9.2.1 Testēšanas plāns

Veicamās darbības:

- Jāpārbauda, vai veidotie vienībtesti veic savu funkcionalitāti, un pārbauda attiecīgās metodes.
- Jāpārbauda, vai lietotāja saskarne reaģē uz lietotāja darbībām tām paredzētajā un aprakstītajā veidā.

9.2.2 Testa rezultātu apzīmējumi

Apakšnodaļās ir sagrupēti izpildāmie testi. Izpildot testus, rezultātu tabulā tiek norādīti testa izpildes rezultāti, izmantojot kādu no apzīmējumiem.

9.2.2.1. tab. testēšanas apzīmējumu skaidrojumi

Apzīmējums	Skaidrojums
OK	Tests izpildīts veiksmīgi, problēmu vai kļūdu nav
DI	Tests izpildās daļēji, kāds no apakšpunktiem, nav izpildīts.
NI	Nav izpildīts, kļūda vai problēma.
NV	Nevar notestēt – nav pieejama atbilstoša testa vide

9.2.3 Vienībtestu apraksts

9.2.3.1 Metodes "DateTimeParser" vienībtesta apraksts

Metode kas pārveido datumus, kas padoti teksta datu tipa formā par datuma datu tipu.

9.2.3.1.1. Tabula

Nr	Pārbaudāmā funkcionalitāte	Ievaddatu specifikācija, veicamās darbības	Testa dati	Sagaidāmais rezultāts	Datums un Statuss
1.	Ja ievadīta pareiza formāta datuma vērtība, metode Atgriez atbilstošo datumu datuma datu tipā.	Ievadīts datums pareizā veidā .	Datums: "03.04.2018"	Pārveidos no teksta tipa mainīgā uz datuma tipa mainīgo: 2018, 04, 03.	11.05.2018 OK
2.	Ja ievadīta nepareiza datuma forma vērtība.	Datums ievadīts ar neiespējamu vērtību.	Datums: "00.00.0000"	Tiks konstatēta kļūda, un tika atgriezta NULL vērtība.	11.05.2018 OK
3.	Ja padodamā vērtība ir padota kā neatbilstoša datu tipa vērtība un nav arī pielīdzināma datuma tipam.	Datuma vērtības vietā ievadīts nepareizs mainīgais.	Datums: "Test"	Tiks konstatēta kļūda, un tika atgriezta NULL vērtība.	11.05.2018 OK
4.	Ja tiek, padota tukša vērtība.	Datuma vērtības vietā nav ievadīts nekas.	Datums: ""	Tiks konstatēta kļūda, un tika atgriezta NULL vērtība.	11.05.2018 OK

9.2.3.2 Metodes “GenerateDetailedFieldContent” vienībtesta apraksts

Metode, kas no detalizācijas datiem izveido bināro datu kopu, kurā ir XLSX datnes saturs.

9.2.3.2.1. Tabula

Nr	Pārbaudāmā funkcionalitāte	Ievaddatu specifikācija, veicamās darbības	Testa dati	Sagaidāmais rezultāts	Datums un Statuss
1.	Mēģina izveidot atskaites datnes satura bin tipa datni, ja ievadīts pareiza formāta detalizācijas lauka nosaukums, tabulas virsrakstu rinda un datu rinda ar detalizācijas informāciju, kura sastāv no viena ieraksta.	Dati ievadīti pareizā formā un detalizācijas sastāv no viena elementa.	Lauka nosaukums: "test" Tabulas virsrakstu rinda: { "Nr.", "a", "b" } Detalizācijas datu rinda: { Detaile = "a s" }.	Tests izvadīs to, ka XLS datne nav tukša, kas nozīmē, ka datne ir veiksmīgi izveidota.	03.05.2018 NI 11.05.2018 OK
2.	Mēģina izveidot atskaites datnes satura bin tipa datni, ja ievadīts pareiza formāta detalizācijas lauka nosaukums, tabulas virsrakstu rinda un datu	Dati ievadīti pareizā formā un detalizācija sastāv no vairākiem elementiem.	Lauka nosaukums: "test" Tabulas virsrakstu rinda: { "Nr.", "a", "b" } Detalizācijas datu rinda: { Detaile = "1 2" }	Tests izvadīs to, ka XLS tipa datne nav tukša, kas nozīmē, ka datne ir veiksmīgi izveidota neatkarīgi no saraksta garuma.	03.05.2018 NI 11.05.2018 OK

	rinda ar detalizācijas informāciju, kura sastāv no viena ieraksta.		{ Detaile = "1 2"} { Detaile = "1 2"} { Detaile = "1 2"} { Detaile = "1 2"} { Detaile = "1 2"}.		
3.	Mēģina izveidot atskaites datnes satura bin tipa datni, ja nav padota informācija par tabulas jauku saturu, bet pārējie dati tiek padoti pareizā formātā.	Dati ievadīti pareizā formātā izņemot detalizācijas saturu.	Lauka nosaukums: "test" Tabulas virsrakstu rinda: { "Nr.", "a", "b"} Detalizācijas datu rinda: NULL.	Tests izvadīs to, ka XLS tipa datne nav tukša, kas nozīmē, ka datne ir veiksmīgi, izveidota pat situācijās, kad pieprasīts izveidot tukšu atskaites tabulu.	03.05.2018 NI 11.05.2018 OK
4.	Mēģina izveidot atskaites datnes satura bin tipa datni, ja nav padota informācija par tabulas jauku saturu un tabulas lauku virsrakstiem, bet nosaukums tiek padots normāli.	Tiek ievadīts tikai tabulas virsraksts un pārējie dati padoti kā tukši.	Lauka nosaukums: "test" Tabulas virsrakstu rinda: NULL Detalizācijas datu rinda: NULL.	Datne netiks izveidota, un tiks izvadīts kļūdu paziņojums: "Notikusi neparedzēta kļūda ģenerējot detalizēto atskaiti!", sakarā ar tabulas izveides elementu trūkumu.	11.05.2018 OK
5.	Mēģina izveidot atskaites datnes satura bin tipa datni,	Netiek padots ne viens no datiem	Lauka nosaukums: NULL Tabulas virsrakstu rinda:	Datne netiks izveidota, un tiks izvadīts kļūdu paziņojums:	11.05.2018 OK

	ja vispār nav padota informācija.		NULL Detalizācijas datu rinda: NULL	"Notikusi neparedzēta kļūda ģenerējot detalizēto atskaiti!", sakarā ar trūkstošajiem datiem datnes izveidei	
--	-----------------------------------	--	---	---	--

9.2.4 Funkcionālie saskarnes testi

9.2.4.1 Pogas detalizētas rādīšanas pie atskaitēm

9.2.4.1. Tabula

Nr.	Funkcionalitāte kas jāpārbauda	Ievaddatu specifikācija, veicamās darbības	Sagaidāmais rezultāts	Datums un Statuss
1.	Atverot atskaišu izveides, skatu, poga "detalizēts" parādās tikai pie atskaites kurai ir izveidoti lauku detalizācijas ieraksti.	Atvērt atskaišu izveides lietotāja saskarni.	Poga "detalizācija" Parādās tikai pie "Ikmēneša pārskats par darbu pirmstiesas kriminālprocesā" atskaites, jo tikai tai ir, izveidota lauku detalizācijas apraksti.	11.05.2018 OK

9.2.4.2 Atskaites satura attēlošana pēc datu atlasēs

9.2.4.2.1. Tabula

Nr.	Funkcionalitāte kas jāpārbauda	Ievaddatu specifikācija, veicamās darbības	Sagaidāmais rezultāts	Datums un Statuss
1.	Aizpildot visus, datu atlasēs, laukus nospiežot, pogu "Attēlot" zem datu atlasēs laukiem tiek attēlota atskaites tabula ar lauku izkārtojumu, kas līdzinās XLSX datnes atskaitēi.	Tiek aizpildīta datu atlasē ar datiem, kas ir adekvāti atskaites izveidei, un nospiež pogu "Attēlot"	Tiks ielādēta atskaites forma zem datu atlasēs laukiem un pogām	11.05.2018 OK

9.2.4.3 Detalizācijas dialoga lauku ielāde

9.2.4.3. Tabula

Nr.	Funkcionalitāte kas jāpārbauda	Ievaddatu specifikācija, veicamās darbības	Sagaidāmais rezultāts	Datums un Statuss
1.	Nospiežot uz jebkura lietotāja saskarnes vides atskaites lauka, būtu jāielādējas dialoga logam ar lauka detalizāciju, ja tāda iespējama. Laukam ar vienu elementu.	Attēlotajā atskaitē tiek nospiests uz tabulas lauka, kura vērtība vienāda ar 1.	Tiks atvērts dialoga logs, kurā būs tabula, kura sastāvēs no virsrakstu rindas un vienas datu rindas.	20.04.2018 DI 11.05.2018 OK

2.	Nospiežot uz jebkura lietotāja saskarnes vides atskaites lauka, būtu jāielādējas dialoga logam ar lauka detalizāciju, ja tāda iespējama. Laukam ar vairākiem elementiem.	Attēlotajā atskaitē tiek nospiests uz tabulas lauka, kura vērtība ir lielāka par 1.	Tiks atvērts dialoga logs, kurā būs tabula, kura sastāvēs no virsrakstu rindas un datu rindām, kas sastāvēs no rindām, kuru skaits ir vienāds ar tabulas lauka vērtības skaitli..	20.04.2018 DI 11.05.2018 OK
3.	Nospiežot uz jebkura lietotāja saskarnes vides atskaites lauka, būtu jāielādējas dialoga logam ar lauka detalizāciju, ja tāda iespējama. Laukam bez neviena elementa.	Attēlotajā atskaitē tiek nospiests uz tabulas lauka, kura vērtība ir 0.	Tiks atvērts dialoga logs, kura detalizācijas tabula būs bez neviena ieraksta. Vienīgi virsrakstu rinda būs redzama no tabulas	20.04.2018 DI 11.05.2018 OK

9.2.4.4 Dialoga loga pogu “nākošais” un “iepriekšējais” rādīšanas atbilstošajās situācijās

9.2.4.4. Tabula

Nr.	Funkcionalitāte kas jāpārbauda	Ievaddatu specifikācija, veicamās darbības	Sagaidāmais rezultāts	Datums un Statuss
1.	Vai atskaites pirmā lauka detalizācijas logā rādās, tikai poga ar bultiņu vērstu	Tiek nospiests uz atskaites pirmā tabulas lauka, un tiek atvērts detalizācijas dialoga skats.	Loga augšējā kreisajā stūrī, zem lauka nosaukuma, būs	11.05.2018 OK

	pa labi un poga ar bultiņu vērtu pa kreisi nav attēlota		tikai poga ar bultiņu vērstu pa labi.	
2.	Vai atskaites laukos, kas nav pirmais vai pēdējais, tiek attēlotas abas bultiņu pogas sakarā ar to, ka visiem tiem laukiem ir atskaites lauks gan pirms tā, gan pēc tā.	Tiek nospiests uz kāda no tabulas laukiem, kas nav ne pirmais, ne pēdējais	Loga augšējā kreisajā stūrī, zem lauka nosaukuma, būs abas bultiņu pogas.	11.05.2018 OK
3.	Vai atskaites pēdējā lauka detalizācijas logā rādās, tikai poga ar bultiņu vērstu pa kreisi un poga ar bultiņu vērstu pa labi nav attēlota.	Tiek nospiests uz atskaites pēdējā tabulas lauka, un tiek atvērts detalizācijas dialoga skats.	Loga augšējā kreisajā stūrī, zem lauka nosaukuma, būs tikai poga ar bultiņu vērstu pa kreisi.	11.05.2018 OK
4.	Vai atskaites detalizācijas dialoga logos bultiņu pogas veic savu funkcionalitāti, ejot pa atskaites laukiem pēc kārtas no pirmā līdz pēdējam.	Tiek atvērts tabulas pirmā lauka detalizācijas logs, un tiek spiests uz pogas, ar pa labi vērstu bultiņu, līdz iziets cauri visiem laukiem.	Katra bultiņa novedīs pie nākamā atskaites lauka, un, kad sasniegs atskaites pēdējo lauku, poga ar pa labi vērsto bultiņu netiks attēlota.	11.05.2018 OK
5.	Vai atskaites detalizācijas dialoga logos bultiņu pogas veic savu funkcionalitāti ejot pa atskaites laukiem pēc kārtas no pēdējā līdz pirmajam.	Tiek atvērts tabulas pēdējā lauka detalizācijas logs, un tiek spiests uz pogas, ar pa labi vērstu bultiņu, līdz iziets cauri visiem laukiem.	Katra bultiņa novedīs pie iepriekšējā atskaites lauka, un, kad sasniegs atskaites pirmo lauku, poga ar pa kreisi vērsto bultiņu netiks attēlota.	11.05.2018 OK

9.2.4.5 Detalizācijas lejupielādes paziņojuma rādīšana

9.2.4.5.1. Tabula

Nr.	Funkcionalitāte kas jāpārbauda	Ievaddatu specifikācija, veicamās darbības	Sagaidāmais rezultāts	Datums un Statuss
1.	Kad notiek detalizētā, lauka XLSX izdrukas lejupielāde jāparādās dialoga laukam ekrāna lietotāja saskarnes lejas daļā, kur rakstīts datnes nosaukums.	Atvērtā detalizācijas dialoga logā nospiež pogu drukāt.	Process norit bez aizķeršanās, un lietotāja saskarnē parādās noklusētais lejupielādes logs ar lejupielādētās datnes nosaukumu, kas sastāv no detalizācijas nosaukuma un attiecīgās dienas datuma.	11.05.2018 OK

9.2.4.6 Individuālās informācijas laukos informācija tiek ielādēta pareizā veidā

9.2.4.6.1. Tabula

Nr.	Funkcionalitāte kas jāpārbauda	Ievaddatu specifikācija, veicamās darbības	Sagaidāmais rezultāts	Datums un Statuss
1.	Atskaites tabulas laukos, kur to vērtība ir izteikta, ka lauks satur vairākas vērtības, ir jāpārliedzinās, ka, nospiežot	Lietotāja saskarnes atskaites attēlojumā laukam ar vairākām vērtībām izvēlas pirmo vērtību un uz tās noklikšķina.	Tiks atvērts dialoga logs, kurā būs dati tikai par pirmo vērtības daļu, kā arī tabulas nosaukums	11.05.2018 OK

	uz vērtības lauka atbilstošo skaitli, tiks ielādēta informācija tikai par to datu daļu. Vai ielādēs datus pareizi, ja izvēlēsies pirmo vērtību.		būs tikai tā daļa, kura attiecās uz pirmo vērtību, kas izvēlēta.	
2.	Atskaites tabulas laukos kur to vērtība ir izteikta, ka lauks satur vairākas vērtības, ka, nospiežot uz vērtības lauka atbilstošās vērtības, tiks ielādēta informācija tikai par to datu daļu. Vai ielādēs datus pareizi, ja izvēlēsies otro vērtību.	Lietotāja saskarnes atskaites attēlojumā laukam ar vairākām vērtībām izvēlas otro vērtību un uz tās noklikšķina.	Tiks atvērts dialoga logs, kurā būs dati tikai par otro vērtības daļu, kā arī tabulas nosaukums būs tikai tā daļa, kura attiecās uz otro vērtību, kas izvēlēta.	11.05.2018 OK

9.2.4.7 Detalizācijas lejupielādētā atskaite atbilst dialoga loga datiem

9.2.4.7.1. Tabula

Nr.	Funkcionalitāte kas jāpārbauda	Ievaddatu specifikācija, veicamās darbības	Sagaidāmais rezultāts	Datums un Statuss
1.	Ja lejupielādē detalizēto lauku kā XLSX datni tad datiem no lietotāja saskarnes, vides būtu jāsakrīt ar tiem, kas jaunajā datnes gan pēc satura, gan izskata.	Atvērtā detalizācijas dialoga logā tiek nospiesta poga "Drukāt". Tiek atvērta lejupielādētā datne.	Datnes saturs sakrītīs ar lietotāja saskarnes detalizācijas informāciju.	08.05.2018 DI 11.05.2018 OK

2.	Ja izvēlēts atskaites lauks, kurš nesatur informāciju, lejupielādētais datne tiks izveidota bet arī tajā izveidotajā tabulā nebūs neviena ieraksta.	Atvērt detalizācijas dialoga logu, kuram vērtība ir nulle, un šim laukam nospiest pogu “Drukāt”. Tiek atvērta lejupielādētajā datnē.	Datne, kura ir lejupielādēta, saturēs tukšu tabulu, kas atbilst lietotāja saskarnes detalizācijas logam.	08.05.2018 DI 11.05.2018 OK
----	---	--	--	--------------------------------------

9.2.4.8 Kļūdu paziņojums par lauku “Struktūrvienība”

9.2.4.8.1. Tabula

Nr.	Funkcionalitāte kas jāpārbauda	Ievaddatu specifikācija, veicamās darbības	Sagaidāmais rezultāts	Datums un Statuss
1.	Ja datu atlasē laukā “Struktūrvienība” ir atlasīta, kāda no izvēlnē minētajām, tad tiks ielādēta atskaites tabula lietotāja saskarnē.	Aizpildot atskaites datu ievades laukus, struktūrvienības laukā tiek izvēlēta “Metodikas nodaļa” Lai procesu pārbaudītu, jānospiež poga “Attēlot”.	Process netiks apturēts, un tiks ielādēta atskaites tabula.	11.05.2018 OK
2.	Ja datu atlasē laukā “Struktūrvienība” netiek atlasīta neviena, no izvēlnē pieejamajām.	Aizpildot atskaites datu ievades laukus, struktūrvienības laukā tiek izvēlēta “Metodikas nodaļa”. Lai procesu pārbaudītu, jānospiež poga “Attēlot”.	Tiks parādīts kļūdas paziņojums “Struktūrvienība ir jānorāda obligāti!”.	11.05.2018 OK

9.2.4.9 Kļūdu paziņojums par lauku “Datums no”

9.2.4.9.1. Tabula

Nr.	Funkcionalitāte kas jāpārbauda	Ievaddatu specifikācija, veicamās darbības	Sagaidāmais rezultāts	Datums un Statuss
1.	Ja laukā “Datums no” ievadīta pareiza informācija, tad tiks ielādēta atskaites tabula lietotāja saskarnē.	Aizpildot atskaites, izveides datu laukus tiek izvēlēts datums pareizā veidā ar vērtību: “01.01.2018”. Lai procesu pārbaudītu, jānospiež poga “Attēlot”.	Process netiks apturēts, un tiks ielādēta atskaites tabula.	11.05.2018 OK
2.	Ja laukā “Datums no” nav ievadīta informācija, tiks parādīts kļūdas paziņojums.	Aizpildot atskaites, izveides datu laukus netiek norādīts lauka “Datums no” vērtība. Lai procesu pārbaudītu, jānospiež poga “Attēlot”.	Tiks parādīts kļūdas paziņojums “Nekorekts laika perioda datums "no"! ”.	11.05.2018 OK
3.	Ja laukā “Datums no” ir ievadīta nepareiza informācija, tiks parādīts kļūdas paziņojums.	Aizpildot atskaites izveides datu laukus “Datums no” vērtības vietā tiek ievadīts nepareizs laika formāts: “1234.56.789”. Lai procesu pārbaudītu, jānospiež poga “Attēlot”.	Tiks parādīts kļūdas paziņojums “Nekorekts laika perioda datums "no"! ”.	11.05.2018 OK

9.2.4.10 Kļūdu paziņojums par lauku “Datums līdz”

9.2.4.10.1. Tabula

Nr.	Funkcionalitāte kas jāpārbauda	Ievaddatu specifikācija, veicamās darbības	Sagaidāmais rezultāts	Datums un Statuss
1.	Ja laukā “Datums līdz” ievadīta pareiza informācija, tad tiks ielādēta atskaites tabula lietotāja saskarnē.	Aizpildot atskaites, izveides datu laukus tiek izvēlēts datums pareizā veidā ar vērtību: “01.01.2018”. Lai procesu pārbaudītu, jānospiež poga “Attēlot”.	Process netiks apturēts, un tiks ielādēta atskaites tabula.	11.05.2018 OK
2.	Ja laukā “Datums līdz” nav ievadīta informācija, tiks parādīts kļūdas paziņojums.	Aizpildot atskaites, izveides datu laukus netiek norādīts lauka “Datums līdz” vērtība. Lai procesu pārbaudītu, jānospiež poga “Attēlot”.	Tiks parādīts kļūdas paziņojums “Laika perioda datums "līdz" ir jānorāda obligāti!”.	11.05.2018 OK
3.	Ja laukā “Datums līdz” ir ievadīta nepareiza informācija, tiks parādīts kļūdas paziņojums.	Aizpildot atskaites izveides datu laukus “Datums līdz” vērtības vietā tiek ievadīts nepareizs laika formāts: “1234.56.789” Lai procesu pārbaudītu, jānospiež poga “Attēlot”.	Tiks parādīts kļūdas paziņojums “Laika perioda datums "līdz" ir jānorāda obligāti!”.	11.05.2018 OK

10 SECINĀJUMI

Veidojot dokumentāciju un arī rakstot programmkodu sistēmā "PROIS", tika veikti vairāki secinājumi. Strādājot pie kvalifikācijas darba, notika komunikācija ar vairākiem nozares pārstāvjiem, kas arī deva vērtīgas atziņas.

- Saziņa ar kolēģiem un viņu dalīšanās ar pieredzi ir viens no visātrākajiem veidiem, kā iejusties jaunajā darba vidē un pierast pie darba izveides struktūras.
- Koda komentāru veidošana ir ļoti svarīga ne tikai tādēļ, lai citi varētu saprast koda fragmentu nozīmi, bet arī koda autoram pašam vieglai koda papildināšanai un kļūdu labošanai.
- Kārtīgi izplānota darbību secība ir nepieciešama, lai veidotu strukturētu un saprotamu programmkodu. Neplānojot darbības var viegli sevi iedzīt stūrī izveidojot kodu, kuru viegli salauzt vai arī grūti implementēt jaunas tā daļas.
- Veidojot programmkodu, pagaidu risinājumi nav īsti risinājumi, jo, ja vajadzēs veikt kādas izmaiņas kodā, tā daļa katreiz būs jāpārveido, lai funkcionētu jaunajā situācijā.

11 IZMANTOTĀ LITERATŪRA

1. Informācijas tehnoloģijas Standartizācijas tehniskā komiteja 1996. LVS 68:1996
“Programmatūras prasību specifikācijas ceļvedis”.
2. Informācijas tehnoloģijas Standartizācijas tehniskā komiteja 1996. LVS 72:1996
“Ieteicamā prakse programmatūras projektējuma aprakstīšanai”;
3. Oracle PL/SQL tiešsaistes kursi [tiešsaiste].-[atsauce 22.05.2018]. –Pieejams:
<https://app.pluralsight.com>
4. HTML tiešsaistes kursi [tiešsaiste].-[atsauce 22.05.2018]. –Pieejams:
<https://www.w3schools.com>
5. Angular JS tiešsaistes kursi [tiešsaiste].-[atsauce 22.05.2018]. –Pieejams:
<https://docs.angularjs.org>
6. Terminoloģijas datu bāze [tiešsaiste].-[atsauce 22.05.2018]. –Pieejams:
<http://termini.lza.lv/term.php>
7. C# tiešsaistes informācijas materiāli [tiešsaiste].-[atsauce 22.05.2018]. –Pieejams:
<https://stackoverflow.com/>
8. Informācijas materiāli no ZZ Dats PROIS projekta – pieejami tikai ZZ Dats darbiniekiem

12 PIELIKUMI

Pielikumu saraksts:

- 1. Pielikums: Ikmēneša pārskats par darbu pirmstiesas kriminālprocesā
- 2. Pielikums: Programmkoda fragments no PL/SQL Developer 12 izstrādes vides
- 3. Pielikums: Lauku detalizācijas tabulas, DrillDown norādes procedūras lauka vērtības piemērs
- 4. Pielikums: Angular funkciju fragments Visual Studio 2017 Izstrādes vidē
- 5. Pielikums: Sistēmas servisa programmkoda fragments Visual Studio 2017 Izstrādes vidē

Pielikums Nr.1.

A	B	C	D	E	F	G	H	I	J	K	L	M		
1	P Ā R S K A T S par darbu pirmstiesas kriminālprocesā						Virsprokurs(e):							
2	_____prokuratūrā par _____mēnešiem 2018.gadā													
3	Nepabeigto lietu atlikums uz gada sākumu		1.				Personu skaits, kuras atbrīvotas no kriminālatbildības uz nereabilitējošiem apstākļiem (KPL 380.p.- izņemot KPL 421.p.)			25.				
4	Pieņemtas izmeklēšanā lietas		2.				Personu skaits, pret kurām kriminālprocess izbeigts uz reabilitējošiem apstākļiem (KPL 377.p.1.2.d. 392.p.2.d.)			26.				
5	to skaitā (no 2. rindas)	Lietas, kuras saņemtas kriminālvajāšanas uzsākšanai / t.sk. kurās izmeklēšana pabeigta 1 mēn. laikā no kriminālprocesa uzsākšanas brīža		3.				Pilnība attaisnotās personas ; personas, pret kurām prokurors atteicies no apsūdzības ; personas, par kurām lieta izbeigta tiesā saskaņā ar KPL 377.p.1.daļas 1.,2.pkt.			27.			
6		lietu skaits pēc sadalīšanas / izmeklēšanas stadijā KPL 38.p.3.d. kārt.		4.				t.sk. tika turētas apcietinājumā (no 26. un 27. rindas)			28.			
7		lietas, kuras saņemtas no tiesas trūkumu novēršanai		5.				Nodotas atpakaļ lietas izmeklēšanas turpināšanai izmeklēšanas iestādēm			29.			
8		lietas, kuras saņemtas no tiesas pārkāpumu novēršanai		6.				Nosūfītas lietas pēc piekrišanas			30.			
9		pārņemtas izmeklēšanai KPL 37.p.2.d. 7.pkt. kārt.		7.				Pievienotas citām lietām			31.			
10		lietas, kuras saņemtas no citām prokuratūras iestādēm		8.				Apturēts kriminālprocess lietās			32.			
11		atjaunotas no agrāk izbeigtajām vai apturētajām		9.				Nepabeigto lietu atlikums uz atskaites perioda beigām			33.			
12		t.sk. no izbeigtajām, nosacīti atbrīvojot no kriminālatbildības		10.				lietas, kurās uzsākta kriminālvajāšana			34.			
13		Tiesai nodoto lietu skaits		11.				lietas ar kriminālvajāšanas termiņu no 2 mēn. līdz 6 mēn.			35.			
14		to skaitā (no 11. rindas)	vispārējā kārtībā		12.				lietas ar kriminālvajāšanas termiņu no 6 mēn. līdz 9 mēn.			36.		
15	saīsinātā procesa kārtībā		13.				lietas ar kriminālvajāšanas termiņu no 9 mēn. līdz 1 gadam			37.				
16	neatliekamības kārtībā		14.				lietas ar kriminālvajāšanas termiņu virs 1 gada			38.				
17	slēdzot vienošanos		15.				Nepabeigtās lietās apcietināto skaits			39.				
18	medicīniska rakstura piespiedu līdzekļu noteikšanai		16.				ar apcietinājuma termiņu līdz 1 mēn.			40.				
19	Tiesai nodoto juridisko personu skaits piespiedu ietekmēšanas līdzekļa piemērošanai		17.				ar apcietinājuma termiņu līdz 4 mēn.			41.				
20	Tiesai nododamo personu skaits		18.				ar apcietinājuma termiņu līdz 6 mēn.			42.				
21	to skaitā - nepilngadīgie		19.				nepilngadīgie			43.				
22	Noziedzīgu nodarījumu skaits tiesai nodotās lietās		20.				No atlikuma, lietas, kuras atrodas izmeklēšanā KPL 38.p.3.d. kārt.			44.				
23	Pabeigti kriminālprocesi (izbeigtas lietas)		21.				Apturēto lietu atlikums uz atskaites perioda beigām			45.				
24	to skaitā	t.sk. 379.p.1.d.2.pkt.,3.pkt., 379.p.2.d.		22.				Prokuroru skaits			pēc štatiem			46.
25		pēc KPL 377.p. 9.pkt. / aiz svītras pēc KPL 379.p.1.d.1.pkt.		23.				uz atskaites perioda beigām			faktiski strādā			47.
26		pēc KPL 415., 415. ¹ p. / aiz svītras ar priekšrakstu par sodu		24.										

Pielikums Nr.2.

```
--<FieldDetails description="Atlasa detalizēto informāciju par vienu
atskaites tabulas lauku">
--<pi_field_code description="Atskaites lauka kods detalizācijas tabulas
izveidei"/>
--<pi_delimiter description="Atskaites lauka ar dalītu informāciju
izvēlētās vērtības numurs"/>
--<pi_report_id description="Atskaites numurs kura ir aktuālā atskaite"/>
--<pi_department_id description="Atbildīgā nodaļa"/>
--<pi_employee_id description="Lietotāja identifikātors"/>
--<pi_periodType description="Kāda veida laika formāts tika izmantots"/>
--<pi_year description="Kurš gads"/>
--<pi_month description="KURš mēnesis"/>
--<pi_dateFrom description="No kura datuma"/>
--<pi_dateTill description="Līdz kuram datumam"/>
--<pi_sheet description="Cik lapas"/>
--<po_cursor1 description="Kursors kas atgriež tekstuālu masīvu ar to no
kā sastāv lauks"/>
--</FieldDetails>
```

```
PROCEDURE FieldDetails (pi_field_code          IN NVARCHAR2,
                        pi_delimiter          IN NUMBER,
                        pi_report_id         IN NUMBER,
                        pi_department_id     IN NUMBER,
                        pi_employee_id      IN NUMBER,
                        pi_periodType       IN NUMBER,
                        pi_year             IN NUMBER,
                        pi_month            IN NUMBER,
                        pi_dateFrom         IN DATE,
                        pi_dateTill        IN DATE,
                        pi_sheet            IN NUMBER,
                        po_cursor1         OUT SYS_REFCURSOR ) is
```

```
v_proceduura varchar2(1000);
v_allldata varchar2(32000);
v_report_code cl_report.clr_code%TYPE;
v_period_tt   period_tt := new period_tt();
v_rights_id   NUMBER(10);
v_functions varchar2_tt := new varchar2_tt();
cache_no varchar2(1000) := null;
```

```
procedure add_to_list(cn varchar2) is
begin
    v_functions.extend();
    v_functions(v_functions.last) := cn;
exception
    when others then
        null;
end add_to_list;
```

```
procedure process_varchars(buf in varchar2) is
    arr apex_application_global.vc_arr2;
begin
    arr := apex_util.string_to_table(buf, '(/)');
    for i in 1..arr.count loop
        if i <= arr.count then
            add_to_list(cache_no || arr(i));
            cache_no := null;
        else
            cache_no := arr(i);
        end if;
    end loop;
end process_varchars;
```

```

BEGIN
U_LOG_PCK.LogInfo('APP_REPORT_PCK.GenerateReport',
' pi_field_code=' || pi_field_code ||
' pi_delimiter=' || pi_delimiter ||
' pi_report_id=' || pi_report_id ||
' pi_department_id=' || pi_department_id ||
' pi_employee_id=' || pi_employee_id ||
' pi_periodType=' || pi_periodType ||
' pi_year=' || pi_year ||
' pi_month=' || pi_month ||
' pi_dateFrom=' || pi_dateFrom ||
' pi_dateTill=' || pi_dateTill ||
' pi_sheet=' || pi_sheet);
-- ' pi_create_employee_id=' || pi_create_employee_id);

BEGIN
SELECT clr.clr_code, clr.clr_rights_id
INTO v_report_code, v_rights_id
FROM cl_report clr
WHERE clr.clr_id = pi_report_id;
EXCEPTION
WHEN no_data_found THEN
RAISE_APPLICATION_ERROR(-20000, 'Sistēmā nav pieejama atskaite ar
id ' || pi_report_id);
END;

CheckRights(v_rights_id);--pārbauda piekļuvi sistēmas lietotājam

select f.CLRF_DRILLDOWN_CURSOR_PROC
into v_proceduura
from CL_REPORT_FIELDS f
where f.CLRF_CURSOR_FIELD_NAME = pi_field_code
and f.clrf_report_id = pi_report_id;

IF pi_periodType = 0 THEN -- 1 gads
v_period_tt.extend();
v_period_tt(1) := GetPeriod(pi_year);
ELSIF pi_periodType = 1 THEN -- iepriekšējie 3 gadi
v_period_tt.extend();
v_period_tt(1) := GetPeriod(pi_year - 1);
v_period_tt.extend();
v_period_tt(2) := GetPeriod(pi_year - 2);
v_period_tt.extend();
v_period_tt(3) := GetPeriod(pi_year - 3);
ELSIF pi_periodType = 2 THEN -- laika periods
v_period_tt.extend();
v_period_tt(1) := new PERIOD_T(pi_dateFrom, pi_dateTill);
ELSIF pi_periodType = 3 THEN -- viena mēneša periods
v_period_tt.extend();
v_period_tt(1) := GetMonthPeriod(pi_year, pi_month);
ELSE
RAISE_APPLICATION_ERROR(-20000, 'Nav norādīts laika periods!');
END IF;

-- Aizvieto teksta mainīgos ar atbilstošajām vērtībām
v_proceduura := replace(v_proceduura, '{DEPARTMENT_ID}',
pi_department_id);--departamenta id tiek ievietots izsaukumā
v_proceduura := replace(v_proceduura, '{PERIOD}', 'period_t('
|| 'to_date('' ||
to_char(v_period_tt(v_period_tt.first).date_from, 'dd.mm.yyyy') || ''',
'dd.mm.yyyy')', '

```

```

|| 'to_date('' ||
to_char(v_period_tt(v_period_tt.last).date_till, 'dd.mm.yyyy') || '',
'dd.mm.yyyy')'
|| ');-- Perioda
vērtība ir ievietota
v_proceduura := replace(v_proceduura, '{DATE_FROM}', 'to_date('' ||
to_char(v_period_tt(1).date_from, 'dd.mm.yyyy') || '', 'dd.mm.yyyy')');
--datums no
v_proceduura := replace(v_proceduura, '{DATE_TILL}', 'to_date('' ||
to_char(v_period_tt(1).date_till, 'dd.mm.yyyy') || '', 'dd.mm.yyyy')');
--datums līdz

--ja ir pieprasīta atsevišķā vērtība
if pi_delimiter != 0 and v_proceduura like '%(/)%' then
process_varchars(v_proceduura);
v_proceduura := v_functions(pi_delimiter);
end if;
--nomaina atlikušos sadalītājus uz simboliem pēc kuru lokācijām varēs
sadalīt selektu
v_proceduura := replace(v_proceduura, '(+)', '(/)');
--Sadala kodu un izgūst informāciju no visiem selektiem
if v_proceduura like '%(/)% ' || ' %(/)%' then
process_varchars(v_proceduura);
v_alldata := '
select t.column_value as details from table(' || v_functions(1) || ') t
union
select k.column_value as details from table(' || v_functions(2) || ') k
union
select l.column_value as details from table(' || v_functions(3) || ') l';

U_LOG_PCK.LogInfo(in_description => 'APP_REPORT_DETAIL_PCK.FieldDetails',
in_data => v_alldata);
open po_cursor1 for v_alldata;
elsif v_proceduura like '%(/)%' then
process_varchars(v_proceduura);
v_alldata := '
select t.column_value as details from table(' || v_functions(1) || ') t
union
select l.column_value as details from table(' || v_functions(2) || ') l';

U_LOG_PCK.LogInfo(in_description => 'APP_REPORT_DETAIL_PCK.FieldDetails',
in_data => v_alldata);

open po_cursor1 for v_alldata;
else
v_alldata := '
select t.column_value as details from table(' || v_proceduura || ') t';
open po_cursor1 for v_alldata;
end if;

--gadījumā ja procesa izpildes laikā bijusi kļūda atgriez kļūdu paziņojumu
EXCEPTION
WHEN too_many_rows THEN
U_LOG_PCK.LogError('APP_REPORT_PCK.GenerateReport', 'Error:' || SQLCODE ||
'. ' || SQLERRM, DBMS_UTILITY.FORMAT_ERROR_BACKTRACE);
RAISE_APPLICATION_ERROR(-20000, 'Notikusi kļūda sagatavojot atskaiti!');
WHEN OTHERS THEN
U_LOG_PCK.LogError('APP_REPORT_PCK.GenerateReport', 'Error:' || SQLCODE ||
'. ' || SQLERRM, DBMS_UTILITY.FORMAT_ERROR_BACKTRACE);
RAISE;
END FieldDetails;

```

Pielikums Nr.3.

```
1  
2  
3 report_generate_detail_pck.GetPersProsEndedCount(pi_department_id => {DEPARTMENT_ID},  
4                                                    pi_period => {PERIOD},  
5                                                    pi_pr_end_article_id => 44)  
6 (+)  
7 report_generate_detail_pck.GetPersProsEndedCount(pi_department_id => {DEPARTMENT_ID},  
8                                                    pi_period => {PERIOD},  
9                                                    pi_pr_end_article_id => 247)  
10 (/)  
11 report_generate_detail_pck.GetPersProsEndedCount(pi_department_id => {DEPARTMENT_ID},  
12                                                    pi_period => {PERIOD},  
13                                                    pi_pr_end_article_id => 43)
```

OK Cancel

Pielikums Nr.4.

```
//Bool mainīgie kas nosaka konkrētu elementu attēlošanu iekš skata
$scope.NoDelimiter = true; //rāda vērtības lauku kā dalītās informācijas lauku
$scope.reportElementsIsLoaded = false; //vērtība kas nosaka vai atskaites
tabula ir redzama vai nē
$scope.FieldDetailShow = false; //Vērtība kas nosaka vai detalizācijas logs ir
redzams

//Atlasa visus atskaites laukus
$scope.GenerateDetails = function () {
    if (!$scope.IsValid()) //izsauc datu pārbaudi
        return;
    Zzdats.Udv.Common.showWaitDialog() //palaiž gaidīšanas dialoga logu
    detailedReportService.GetReportFields($scope.selectedReportTypeId,
    $scope.template.Code, $scope.DepartmentId, $scope.Employee.Id, $scope.PeriodType,
    $scope.Year, $scope.DateFrom, $scope.DateTill, $scope.Month).then(function (response)
    {
        if (response.Result == 0) //pārbauda vai datu atgriežšanas procesā nav
        bijusi kļūda ja ir paziņo par to sistēmas lietotājam
        {
            Zzdats.Udv.Common.hideWaitDialog()
            Zzdats.Udv.Common.showErrorDialog("Atskaites veidošanas procesā
            notikusi kļūda!"); //kļūdas paziņojums par datu neatgriešanu
        }
        else
        {
            $scope.ReportFields = response.ReportFields;

            $scope.firstElem = $scope.ReportFields[0]; //saglabā pirmo datu kopas
            elementu
            $scope.lastElem = $scope.ReportFields[$scope.ReportFields.length - 1];
            //saglabā pēdējo datu kopas elementu

            $scope.reportElementsIsLoaded = true; //Padara redzamu tabulu skatā

            Zzdats.Udv.Common.hideWaitDialog()
        }
    });
}

//Lejupielādē pilnu nedetalizētu atskaiti
$scope.GenerateReport = function () {
    if (!$scope.IsValid()) //izsauc datu pārbaudi
        return;

    var queryStr = $.param({
        templateId: $scope.template.Id,
        templateCode: $scope.template.Code,
        departmentId: $scope.DepartmentId,
        employeeId: $scope.Employee.Id,
        periodType: $scope.PeriodType,
        year: $scope.Year,
        dateFrom: $scope.DateFrom,
        dateTill: $scope.DateTill,
        month: $scope.Month
    });

    $('#FileDownloadContainer').attr('src', urlService.BaseUrl +
    "/ReportGenerator/GenerateReport/" + queryStr);
}

//atlasa informāciju par atskaites lauku
```

```

$scope.FieldDetails = function (field, index, delimiter)
{
    //bultiņu rādīšana iekš dialoga loga, ja ir pirmais vai pēdējais
    $scope.notFirst = true;
    $scope.notLast = true;
    if (field == $scope.firstElem) { $scope.notFirst = false }
    if (field == $scope.lastElem) { $scope.notLast = false }
    if (delimiter != 0)
    {
        $scope.notFirst = false;
        $scope.notLast = false;
        //$scope.delimIndex = delimiter
    }
    $scope.FieldDetailShow = false;

    $scope.delimiterNum = delimiter;

    Zzdats.Udv.Common.showWaitDialog();//parāda gaidīšanas dialogu

    $scope.curIndex = index;
    $scope.FieldIndividual = field;
    $scope.FieldIndividual;
    if (field.Drilldown_cursor_name != 'null') {
        $scope.fieldCursorName = field.Cursor_name;
        detailedReportService.GetReportFieldElements($scope.fieldCursorName,
        $scope.delimiterNum, $scope.selectedReportTypeId, $scope.template.Code,
        $scope.DepartmentId, $scope.Employee.Id, $scope.PeriodType, $scope.Year,
        $scope.DateFrom, $scope.DateTill, $scope.Month).then(function (response) {
            $scope.FieldElements = response.ReportFieldElements;

        });
    }
    else { $scope.FieldElements = [];}
    Zzdats.Udv.Common.hideWaitDialog()
    $scope.FieldDetailShow = true;

    $('#myModal').modal();
}

//Funkcija kas izsauc atskaites lauka detalizācijas dokumenta izveidi
$scope.printField = function (delimiter) {

    var queryStr = $.param({
        nameOfField: $scope.FieldIndividual.Name,
        fieldHeders: $scope.FieldIndividual.Drilldown_headings,

        fieldCode: $scope.fieldCursorName,
        vertNr: $scope.delimiterNum,
        id: $scope.selectedReportTypeId,
        departmentId: $scope.DepartmentId,
        employeeId: $scope.Employee.Id,
        periodType: $scope.PeriodType,
        year: $scope.Year,
        dateFrom: $scope.DateFrom,
        dateTill: $scope.DateTill,
        month: $scope.Month,
    });

    $('#FileDownloadContainer').attr('src', urlService.BaseUrl +
    "/DetailedReportGenerator/GenerateDetailedReportXLSX/?" + queryStr);
}

```

Pielikums Nr.5.

```
/// <summary>
/// Iegūst datus kas nepieciešami lai savienotos ar datu bāzi
/// </summary>
public override string ConnectionString
{
    get { return
        cutionContext.ServiceContext.GetPropertyAsConnectionString("CONNECTIONSTRING").Re
        place(";Unicode=True", ";Pooling=true"); }
}
/// <summary>
///
/// </summary>
/// <param name="executionContext"></param>
public DetailedReportService(ILgpServiceExecutionContext executionContext)
: base(executionContext)
{
    var identity = executionContext.Identity;
    var principal = executionContext.Principal;
    _dbContext = new ReportDbContext(identity.Id, ApplicationId, identity.DistrictId,
    identity.SecondDistrictId, principal.ContextId);
}

/// <summary>
/// Metode kas iegūst datu kopu no datubāzes kas satur atskaites lauku informāciju un
/// datus kas nepieciešami detalizācijas izveidei
/// </summary>
/// <param name="id">Izvēlētās atskaites identifikācijas kods</param>
/// <param name="templateCode">Atskaites Tabulas lejupielādēšanas parauga
/// identifikācijas numurs</param>
/// <param name="departmentId">Departamenta identifikācijas kods</param>
/// <param name="employeeId">Sistēmas lietotāja identifikācijas kods</param>
/// <param name="periodType">Laika perioda tips</param>
/// <param name="year">Gads</param>
/// <param name="dateFrom">Datums no kura sākuma pieprasīti dati</param>
/// <param name="dateTill">Datums līdz kura beigām pieprasīti dati</param>
/// <param name="month">Mēnesis par kuru pieprasīti dati</param>
/// <param name="sheet">Lapu skaits atskaitē Nepieciešams 19atskaites
/// izveidei</param>
/// <returns>Atgriež datu kopu ar informāciju par atskaites laukiem</returns>
public List<DetailedReport> GetDetailedReport(int id, string templateCode, int
departmentId, int? employeeId, int periodType, int year, DateTime? dateFrom, DateTime?
dateTill, int? month, int? sheet)
{
    var DetailedReport = new List<DetailedReport>();
    using (OdpHelper.CreateBatch(ExecutionContext, ConnectionString, _dbContext))
    {
        return OdpHelper.ExecuteInCurrentBatch(ExecutionContext, ConnectionString, cn =>
        {
            _dbContext.Initialize(cn);
            using (var cmd = new OracleCommand("APP_REPORT_DETAIL_PCK.reportFields",
            cn) { CommandType = CommandType.StoredProcedure })
            {
                cmd.Parameters.Add(new OracleParameter("pi_report_id",
                OracleDbType.Int64, id, ParameterDirection.Input));
                cmd.Parameters.Add(new OracleParameter("pi_department_id",
                OracleDbType.Int32, departmentId, ParameterDirection.Input));
                cmd.Parameters.Add(new OracleParameter("pi_employee_id",
                OracleDbType.Int32, employeeId.HasValue ? (object)employeeId.Value :
                DBNull.Value, ParameterDirection.Input));
                cmd.Parameters.Add(new OracleParameter("pi_periodType",
                OracleDbType.Int32, periodType, ParameterDirection.Input));
            }
        }
    }
}
```

```

cmd.Parameters.Add(new OracleParameter("pi_year", OracleDbType.Int32,
year, ParameterDirection.Input));
cmd.Parameters.Add(new OracleParameter("pi_month", OracleDbType.Int32,
month, ParameterDirection.Input));
cmd.Parameters.Add(new OracleParameter("pi_dateFrom",
OracleDbType.Date, dateFrom, ParameterDirection.Input));
cmd.Parameters.Add(new OracleParameter("pi_dateTill",
OracleDbType.Date, dateTill, ParameterDirection.Input));
cmd.Parameters.Add(new OracleParameter("pi_sheet", OracleDbType.Int32,
sheet.HasValue ? (object)sheet.Value : DBNull.Value,
ParameterDirection.Input));
cmd.Parameters.Add(new OracleParameter("po_cursor1 ",
OracleDbType.RefCursor)).Direction = ParameterDirection.Output;
cmd.Parameters.Add(new OracleParameter("po_cursor2",
OracleDbType.RefCursor)).Direction = ParameterDirection.Output;
OracleDataReader dr = cmd.ExecuteReader();
ReportData = new ReportData();
//pirmās datu kopas datus par atskaites lauku vērtībā pievieno
atbilstošajiem laukiem
while (dr.Read())
{
    reportData.Add(dr);
}
dr.NextResult();
Dictionary<string, dynamic> valueDictionary = reportData.Data.First();
//Otrās datu rindai piešķir apzīmētājus
while (dr.Read())
{
    DetailedReport.Add(new DetailedReport()
    {
        Id = dr.GetIntEx("CLRF_ID"),
        Report_id = dr.GetIntEx("CLRF_REPORT_ID"),
        Order_print = dr.GetIntEx("CLRF_ORDER_PRINT"),
        Name = dr.GetStringEx("CLRF_NAME"),
        NULL_or_0 = dr.GetIntNullableEx("CLRF_SHOW_0_AS_NULL") ??
        0,
        Cursor_procedure = dr.GetStringEx("CLRF_CURSOR_PROCEDURE"),
        Cursor_name = dr.GetStringEx("CLRF_CURSOR_FIELD_NAME"),
        Drilldown_cursor_name =
        dr.GetStringEx("CLRF_DRILLDOWN_CURSOR_PROC"),
        Drilldown_headings =
        dr.GetStringEx("CLRF_DRILLDOWN_HEADINGS"),
        Delimiter = dr.GetIntNullableEx("CLRF_DELIMITER") ?? 0,
        Valid_from = dr.GetDateTimeNullableEx("CLRF_VALID_FROM"),
        Valid_till = dr.GetDateTimeNullableEx("CLRF_VALID_TILL"),
        Table_and_field_num =
        dr.GetStringEx("CLRF_TABLE_AND_NUMBER"),
        Text_align = dr.GetStringEx("CLRF_TEXT_ALIGN"),
        Value =
        valueDictionary[dr.GetStringEx("CLRF_CURSOR_FIELD_NAME")]
    }
    );
}
dr.Close();
return DetailedReport;
}
});
}
}

```

Kvalifikācijas darbs „*Detalizēto (drill-down) atskaišu moduļa izstrāde sistēmai PROIS*”
izstrādāts Latvijas Universitātes Datorikas fakultātē.

Ar savu parakstu apliecinu, ka darbs izstrādāts patstāvīgi, izmantoti tikai tajā norādītie
informācijas avoti un iesniegtā darba elektroniskā kopija atbilst izdrukai.

Autors: *Atis Ēriks Dumpis* _____ .05.2018.

Rekomendēju darbu aizstāvēšanai

Darba vadītājs: *M. dat. Valts Grigāns* _____ .05.2018.

Recenzents: *M. dat. Jānis Upītis*

Darbs iesniegts 28.05.2018.

Kvalifikācijas darbu pārbaudījumu komisijas sekretārs: *Darja Solodovņikova* _____

Darbs aizstāvēts kvalifikācijas darbu pārbaudījuma komisijas sēdē

____.06.2018. prot. Nr. _____

Komisijas sekretārs(-e): _____