

LATVIJAS UNIVERSITĀTE
DATORIKAS FAKULTĀTE

ATSKAIŠU SISTĒMA KOKVEDĒJA VADĪTĀJIEM

KVALIFIKĀCIJAS DARBS

Autors: Reinis Eduards Indriksons

Studenta apliecības Nr.: ri19012

Darba vadītājs: Dr. dat. Jevgēnijs Vihrovs

RĪGA 2022

ANOTĀCIJA

Lietotnes “Atskaišu sistēma kokvedēja vadītājiem” mērķis ir atvieglot loģistikas speciālistu komunikāciju ar kokvedēju vadītājiem, sniedzot minētajiem vadītājiem iespēju ērti sniegt atskaites par veicamā darba statusu. Izstrādātā sistēma apkopo informāciju par katra vadītāja tā brīža pildītajiem darba pienākumiem un ļauj loģistikas speciālistiem plānot un uzdot vadītājiem tālākos darba pienākumus. Sistēma datus glabā datubāzē un tos lietotājam pasniedz viegli pārskatāmā veidā, atvieglojot to pārvaldību. Sistēma sastāv no “Windows Presentation Foundation” lietotāja saskarnes satvara, “Entity Framework” relāciju kartēšanas sistēmas un “Microsoft SQL Server” datu bāzes pārvaldības sistēmas.

Atslēgvārdi: Atskaišu sistēma, loģistika, pārvaldība, WPF, Entity Framework

REPORTING SYSTEM FOR TIMBER TRUCK DRIVERS

ABSTRACT

The purpose of the application "Reporting system for timber truck drivers" is to facilitate the communication between logistics specialists and timber truck drivers, giving them the opportunity to easily report on the status of the work being performed. The developed system collects information about the current job responsibilities of each driver and allows logistics specialists to plan and assign further work to drivers. The system stores the data in a database and presents it to the user in an easy-to-read way, making it easier for users to manage ongoing work. The system consists of "Windows Presentation Foundation" user interface framework, "Entity Framework" relational mapping system, and "Microsoft SQL Server" database management system.

Keywords: Reporting system, logistics, management, WPF, Entity Framework

SATURS

1. DARBA APRAKSTS	7
1.1. Darba mērķis un uzdevumi	7
1.2. Darba struktūra.....	7
1.3. Sistēmas neformāls apraksts	7
1.4. Projekta organizācija	7
1.5. Konfigurāciju pārvaldība	8
1.6. Kvalitātes nodrošināšana.....	8
1.7. Darbietilpības novērtējums	8
2. PROGRAMMATŪRAS PRASĪBU SPECIFIKĀCIJA.....	9
2.1. Ievads	9
2.1.1. Dokumenta nolūks	9
2.1.2. Darbības sfēra.....	9
2.1.3. Saistība ar citiem dokumentiem.....	10
2.2 Vispārīgais apraksts	10
2.2.1. Produkta perspektīva.....	10
2.2.2 Darījumasprasības	10
2.2.3 Sistēmas lietotāji	10
2.2.4. Lietotāja raksturiezīmes	11
2.2.5. Vispārīgie ierobežojumi	11
2.2.6. Pieņēmumi un atkarības	11
2.3. Funkcionālās prasības	11
2.3.1. Lietotāja pieslēgšanās sistēmai	11
2.3.2. Pievienot lietotāju.....	12
2.3.3. Dzēst lietotāju.....	12

2.3.4. Rediģēt lietotāju	13
2.3.5. Izveidot jaunu uzdevumu	13
2.3.6. Pārskatīt uzdevumus un vadītājus	14
2.3.7. Apstiprināt uzdevumu	14
2.3.8. Izpildīt uzdevumu.....	14
2.4. Nefunkcionālās prasības.....	15
2.4.1. Pieejamība	15
2.4.2. Aizsardzība pret nesankcionētu piekļuvi	15
2.4.3. Veiktspēja.....	15
2.4.4. Izmantojamība.....	15
3. PROGRAMMATŪRAS PROJEKTĒJUMA APRAKSTS	16
3.1. Ievads	16
3.1.1. Dokumenta nolūks	16
3.1.2. Darbības sfēra.....	16
3.1.3. Saistība ar citiem dokumentiem	16
3.2. Dekompozīcijas apraksts.....	16
3.2.1. Moduļu dekompozīcija.....	16
3.2.2. Datu dekompozīcija	17
3.3. Datu modelis	18
3.3.1. ER realizācijas modelis	18
3.4. Lietotāja saskarne.....	23
3.4.1. Autentifikācija.....	23
3.4.2. Administratora puse	24
3.4.3. Kokvedēja vadītāja puse	26
3.4.4. Loģistikas speciālista puse	29
3.5. Moduļu projektējums	33
3.5.1. Starpmoduļu atkarības.....	33
3.5.2. Konteksta līmeņa datu plūsmas diagramma.....	33

3.5.3. Pirmā līmeņa datu plūsmas diagramma	33
3.5.4. Otrā līmeņa datu plūsmas diagrammas	35
4. TESTĒŠANAS DOKUMENTĀCIJA	38
4.1. Testēšanas plāns	38
4.2. Testēšanas žurnāls	38
4.2.1. Autentifikācija.....	38
4.2.2. Lietotāju pārvaldība	38
4.2.3. Uzdevumu pārvaldība	39
5. SECINĀJUMI.....	40
6. IZMANTOTĀ LITERATŪRA.....	41
7. PROGRAMMATŪRAS KODS	42
7.1. Ievads	42
7.2 LoginWindow.xaml.cs	42
7.3. MainWindowDriver.xaml.cs.....	44
7.4. AssignmentWindowDriver.xaml.cs	46

1. DARBA APRAKSTS

1.1. Darba mērķis un uzdevumi

Šī kvalifikācijas darba mērķis ir veikt konkrētās sistēmas izstrādi un aprakstīt tās darbību un izstrādes procesu. Darba izstrādes laikā ir veikti sekojoši uzdevumi:

- Apkopotas pret sistēmu izvirzītas prasības vienotā programmatūras prasību specifikācijā,
- Izstrādāts programmatūras projektējuma apraksts,
- Izstrādāta “Windows” tipa lietotne datu reģistriem un apstrādei,
- Veikta sistēmas testēšana.

1.2. Darba struktūra

Kvalifikācijas darba pirmā nodaļa sniedz ieskatu izstrādātajā sistēmā, un darba tapšanas procesā. Tālāk seko izstrādātā programmatūras prasību specifikācija, projektējuma apraksts, un testēšanas dokumentācija. Uzskatāmības nolūkos projektējuma aprakstā iekļautie saskarnes skati gūti no jau izstrādātās sistēmas. Darba beigās iekļauti koda fragmenti. Viss izstrādātās sistēmas kods nav iekļauts, lai ietaupītu lappušu skaitu. Kods tika izstrādāts angļu valodā izstrādātāja preferences dēļ ar mērķi nodrošināt produktivitāti.

1.3. Sistēmas neformāls apraksts

Balstoties uz kravas automobiļa vadītāja darba specifiku, komunikācija ar loģistikas speciālistu ir ļoti svarīgs, bet relatīvi sarežģīts minētā vadītāja darba aspekts. Lai atvieglotu loģistikas speciālista darba pārvaldīšanu, izvairoties no liekas vadītāju apgrūtināšanas, tika nolemts izstrādāt lietotni atbilstošā uzdevuma veikšanai. Jau no paša sākuma tika nolemts, ka lietotne specializēsies tieši kokmateriālu pārvadājumos.

1.4. Projekta organizācija

Pirms kvalifikācijas darbā aprakstītās sistēmas izstrādes tika konsultēti vairāki loģistikas speciālisti un mežu apsaimniekošanā iesaistīti speciālisti, lai noskaidrotu, kādas būtu lietotnes lietotāju prasības, potenciālie ierobežojumi un meža apsaimniekošanai specifiskās problēmas, kuras jāparedz darba izstrādes laikā. Plānojot lietotnes saskarni, iepriekš minētajiem

speciālistiem tika prezentēti potenciālie saskarņu projektējumi. Speciālistu sniegtās atsauksmes tika ņemtas vērā izstrādājot lietotnes saskarni. Tāda pati metode tika pielietota, plānojot datubāzes tabulu izvietojumu un saturu.

Sistēmas izstrāde tika sākta ar saskarnes izveidi un datubāzes tabulu izveidi, balstoties uz programmatūras prasību specifikācijas uzmetumu. Pēc tam tika pakāpeniski realizētas plānotās sistēmas funkcijas.

1.5. Konfigurāciju pārvaldība

Ņemot vērā, ka sistēmu izstrādāja tikai viens programmētājs, nekādu īpašu konfigurācijas rīku netika izmantots. Sistēmas faili regulāri tika kopēti ārējā datu nesējā.

1.6. Kvalitātes nodrošināšana

Lai nodrošinātu kvalitatīvu produkta izstrādi, produkts regulāri tika testēts, īpašu uzmanību vēršot funkcijām, kuras tika realizētas vai mainītas, paralēli vēršot uzmanību funkcijām, kurām bija potenciāls sākt darboties kļūdaini pēc šķietami nenozīmīgām izmaiņām. Kvalitātes un tajā skaitā produktivitātes nodrošināšanai tika pielietots žurnāls, kurā regulāri tika ierakstīti ar sistēmas izstrādi saistītie apsvērumi.

1.7. Darbietilpības novērtējums

Darbietilpības novērtēšanai tika izmantots COCOMO darbietilpības aprēķināšanas modelis. Kalkulatoram [1] tika padoti šādi parametri:

1.7.1 Tabula

Darbietilpības novērtējuma parametru tabula

Parametrs	Vērtība
Aptuvenais koda rindīņu skaits	1000
Programmatūras izstrādes režīms	Organic Mode
Produkta atribūti	
Prasītā uzticamība	N
Datubāzes izmērs	L
Produkta sarežģītība	L
Datora atribūti	
Izpildes laika ierobežojumi	N
Datu glabāšanas ierobežojumi	N
Platformas nepastāvība	N
Datora ātrdarbība	N
Personāla atribūti	
Analītiskās spējas	L

Pieredze aplikāciju izstrādē	L
Programmētāja spējas	N
Platformas pieredze	N
Programmēšanas valodas un rīku pieredze	N
Projekta atribūti	
Moderna programmēšanas prakse	N
Programmatūras rīku izmantošana	N
Nepieciešamais izstrādes grafiks	N

Rezultāts: 3.583

Lai arī aprēķinātā darbietilpība aptuveni atbilst izstrādes laikam, COCOMO mēdz būt diezgan neprecīzs darbietilpības aprēķina modelis, īpaši maziem projektiem.

Atskatoties uz paveikto, lielu daļu laika patērēja dokumentācijas izstrāde. Arī pašapmācības process prasīja relatīvi lielu laika daudzumu.

2. PROGRAMMATŪRAS PRASĪBU SPECIFIKĀCIJA

2.1. Ievads

2.1.1. Dokumenta nolūks

Programmatūras prasību specifikācija paredzēta izstrādājamās programmatūras "Atskaišu sistēma kokvedēja vadītājiem" prasību aprakstīšanai, kas būs pamats tālākai sistēmas izstrādei.

2.1.2. Darbības sfēra

Programmatūra "Atskaišu sistēma kokvedēja vadītājiem" paredzēta iekšējai lietošanai uzņēmumā, kurš nodarbojas ar kokmateriālu pārvadājumiem. Lietotne sniedz iespēju loģistikas speciālistiem efektīvāk komunicēt ar kokvedēja vadītājiem, uzdodot viņiem uzdevumus par veicamajiem pārvadājumiem. Tāpat arī kokvedēja vadītāji var sniegt atskaites loģistikas speciālistiem, lai tie spētu sekot līdzi notiekošajam uzņēmuma kravu pārvadājumos.

2.1.3. Saistība ar citiem dokumentiem

Dokuments izstrādāts balstoties uz *LVS 68:1996 „Programmatūras prasību specifikāciju ceļvedi”* [3].

2.2 Vispārīgais apraksts

2.2.1. Produkta perspektīva

Lietotni būtu iespējams izvērst lietošanai arī citu veidu kravu pārvadājumos. Tāpat arī lietotni būtu iespējams izvērst par ne tikai kokvedēju vadītāju atskaitēm, bet gan kopējai uzņēmuma darbinieku veicamo darbu pārvaldīšanai.

2.2.2 Darījumprasības

Šajā sadaļā tiek aprakstītas produktam paredzētās funkcijas.

- Autentificēšanās,
- Pārvadājuma uzdevuma izveide,
- Atskaites sniegšana par pārvadājuma uzdevuma izpildi,
- Aktīvo un pagājušo pārvadājuma uzdevumu apskate,
- Uzdots pārvadājuma uzdevumu apskate,
- Uzņēmuma kokvedēja vadītāju statusa apskate
- Lietotnes lietotāju pievienošana, rediģēšana un dzēšana

2.2.3 Sistēmas lietotāji

Sistēmai ir 3 lietotāju grupas – vadītājs, loģistikas speciālists un sistēmas administrators

Vadītāji izmanto lietotni lai veiktu saņemtu pārvadājumu uzdevumus un veiktu atskaites par tiem.

Loģistikas speciālisti izmanto lietotni lai piekļūtu datiem par notiekošajiem pārvadājumiem, vadītāju darba statusu un lai izveidotu un vadītājiem piešķirtu jaunus pārvadājuma uzdevumus.

Sistēmas administratori ir personas uzņēmumā, kuras atbild par sistēmā reģistrētajiem lietotājiem. Viņi ir tie, kuri sistēmai pievieno jaunus lietotājus, rediģē vai dzēš esošos.

2.2.4. Lietotāja raksturierzīmes

Lietotājiem ir nepieciešama sapratne par kokmateriālu pārvadājumu procesu. Papildus tam lietotājiem ir nepieciešamas pamazināšanas darbā ar datoru. Sistēmas administratoriem jāpārziņa uzņēmuma iekšējo kārtību .

2.2.5. Vispārīgie ierobežojumi

Sistēma tiek atbalstīta tikai uz Windows operētājsistēmām. Tās darbībai nepieciešams datubāzes serveris. Sistēmas darbībai nepieciešams aktīvs interneta savienojums.

2.2.6. Pieņēmumi un atkarības

- Kravas kokvedējs ir aprīkots ar datoru.
- Kravas kokvedējs ir aprīkots ar sistēmai neatkarīgu navigācijas sistēmu, kura atbalsta koordinātu kā galamērķa iestatīšanu.
- Lietotāji saskarnē tiks identificēti pēc uzņēmumā unikāla e-pasta,
- Vienam kokvedēja vadītājam var būt vairāki uzdevumi.

2.3. Funkcionālās prasības

2.3.1. Lietotāja pieslēgšanās sistēmai

Izmantojot iepriekš piešķirtu e-pastu un paroli pieslēgties sistēmai.

Ievade:

Atverot sistēmu atbilstošajos laukos ievadīt e-pastu un paroli.

Ievaddati:

E-pasts – teksts, maksimālais garums 40, obligāts, nepieciešams pareizā formātā,

Parole – teksts, maksimālais garums 40, obligāts,

Apstrāde:

Sistēma sameklē lietotāju datubāzē. Ja lietotājs ir atrasts, atgriež lietotāja lomu sistēmā.

Izvade:

Ja lietotājs ir atrasts, uzsāk lietotnes sesiju atverot lietotāja lomai atbilstošu logu.

Ja lietotājs nav atrasts, tiek izdots paziņojums “Nepareizs e-pasts un/vai parole.”

2.3.2. Pievienot lietotāju

Nodrošina iespēju sistēmu papildināt ar lietotājiem. Funkcijai piekļūt var tikai sistēmas administrators.

Ievade:

Vārds – teksts, maksimālais garums 40, obligāts,

Uzvārds – teksts, maksimālais garums 40, obligāts,

E-pasts – teksts, maksimālais garums 40, obligāts, nepieciešams pareizā formātā,

Parole – teksts, maksimālais garums 40, obligāts,

Tel. nr. – skaitlis, obligāts,

Loma – skaitlis, katram skaitlim atbilst loma sistēmā, obligāts:

1 – sistēmas administrators,

2 – loģistikas speciālists,

3 – kokvedēja vadītājs,

Apstrāde:

Sistēma pievieno jaunu lietotāju datubāzē. Padotā parole tiek šifrēta SHA256 jaučējkodā.

Izvade:

Paziņojums “Lietotājs pievienots!”

2.3.3. Dzēst lietotāju

Nodrošina iespēju sistēmā dzēst lietotājus. Funkcijai piekļūt var tikai sistēmas administrators.

Ievade:

E-pasts - teksts, maksimālais garums 40, obligāts, nepieciešams pareizā formātā.

Apstiprinājums – skaitlis funkcijas izpildes apstiprināšanai, obligāts

1 – Jā;

2 – Nē;

Apstrāde:

Ja administrators funkcijas izpildi apstiprina, sistēma atrod lietotāju datubāzē un dzēš visus datus par viņu.

Izvade:

Paziņojums “Lietotājs dzēsts”

2.3.4. Rediģēt lietotāju

Nodrošina iespēju sistēmā rediģēt lietotājus. Funkcijai piekļūt var tikai sistēmas administrators.

Ievade:

Vārds – teksts, maksimālais garums 40,

Uzvārds – teksts, maksimālais garums 40,

E-pasts – teksts, maksimālais garums 40,

Parole – teksts, maksimālais garums 40,

Tel. nr. – skaitlis,

Loma – skaitlis, katram skaitlim atbilst loma sistēmā,

1 – sistēmas administrators,

2 – loģistikas speciālists,

3 – kokvedēja vadītājs,

Apstrāde:

Ja administrators ir ievadījis kādu no vērtībām, lietotājs tiek atrasts datubāzē un atbilstošās izmaiņas tiek veiktas.

Izvade:

Paziņojums “Lietotājs rediģēts!”

2.3.5. Izveidot jaunu uzdevumu

Nodrošina iespēju sistēmā izveidot jaunu uzdevumu. Funkcijai piekļūt var tikai loģistikas speciālists.

Ievade:

Krautuve – krautuves koordinātas teksts, maksimālais garums 20, obligāts,

Šķirne – koksnes šķirnes teksts, maksimālais garums 20, obligāts,

Sortiments – koksnes sortimenta teksts, maksimālais garums 30, obligāts,

Galamērķis – adreses teksts, maksimālais garums 100, obligāts

Vadītājs – vadītāju identificējošs e-pasts, obligāts

Apstrāde:

Sistēma saņemot datus piešķir uzdevumam datumu un statusu, pievieno datubāzei

Izvade:

Kokvedēja vadītājs saņem jaunu uzdevumu, loģistikas speciālista pusē parādās jauns uzdevums.

2.3.6. Pārskatīt uzdevumus un vadītājus

Nodrošina iespēju loģistikas speciālistam apskatīt uzsāktus un pabeigtus uzdevumus un vadītājus.

Ievade:

Uzdevuma identifikators – unikāls skaitlis, obligāts,

Personas identifikators – unikāls skaitlis, obligāts,

Apstrāde

Sistēma saņemot atbilstošos identifikatorus atrod datubāzē atbilstošos uzdevumus un izdod datus par tiem speciālistam saprotamā veidā.

Izvade:

Jauns logs ar datiem par uzdevumu.

2.3.7. Apstiprināt uzdevumu

Nodrošina iespēju lietotājam apstiprināt viņam uzdotu uzdevumu Funkcijai piekļūt var tikai kokvedēja vadītājs.

Ievade:

Apstiprinājums – skaitlis, obligāts:

1 – Jā,

2 – Nē.

Apstrāde:

Ja vadītājs ir apstiprinājis izvēli, uzdevuma statuss tiek mainīts no “Atvērts” uz “Īesākts”

Izvade:

Atveras vērtību lauki uzdevuma izpildei – datu ievadei.

2.3.8 Izpildīt uzdevumu

Nodrošina vadītājam iespēja izpildīt uzdevumu – ievadīt datus.

Ievade:

Uzkrautais daudzums – skaitlis, obligāts,

Koksnes novērtējums – teksts atbilstošam koksnes novērtējumam, obligāts:

- A,
- B,
- C,
- D.

Paredzētais laiks ceļā – stundas, vesels skaitlis, obligāts,

Patērētais laiks ceļā – stundas, vesels skaitlis, obligāts.

Apstrāde:

Pakāpeniski aizpilda datubāzē esošo uzdevumu.

Izvade:

Uzdevums pazūd no vadītājam piešķirtajiem uzdevumiem. Loģistikas speciālists uzdevumu var apskatīt pie izpildītajiem uzdevumiem.

2.4. Nefunkcionālās prasības

2.4.1. Pieejamība

Sistēmai jābūt pieejamai visu laiku, un katram reģistrētam lietotājam ir jābūt spējīgam sistēmai pievienoties.

2.4.2. Aizsardzība pret nesankcionētu piekļuvi

Katram lietotājiem ir sniegti pieejas dati. Paroles datubāzē tiek šifrētas.

2.4.3. Veiktspēja

Sistēmai jānodrošina vairāku lietotāju piekļuvi reizē, datubāzei jābūt spējīgai turēt lielu daudzumu datu.

2.4.4. Izmantojamība

Sistēmai jābūt pietiekami vienkāršai un saprotamai, lai sekmētu efektīvu sistēmas pielietošanu

3. PROGRAMMATŪRAS PROJEKTĒJUMA APRAKSTS

3.1. Ievads

3.1.1. Dokumenta nolūks

Dokumenta nolūks ir aprakstīt sistēmas “Atskaišu sistēma kokvedēja vadītājiem” Tas ir paredzēts sistēmas izstrādātājam un ir sastādīts programmatūras prasību specifikācijas izvirzītajām prasībām.

3.1.2. Darbības sfēra

Programmatūra "Atskaišu sistēma kokvedēja vadītājiem" paredzēta iekšējai lietošanai uzņēmumā, kurš nodarbojas ar kokmateriālu pārvadājumiem. Lietotne sniedz iespēju loģistikas speciālistiem efektīvāk komunicēt ar kokvedēja vadītājiem, uzdodot viņiem uzdevumus par veicamajiem pārvadājumiem. Tāpat arī kokvedēja vadītāji var sniegt atskaites loģistikas speciālistiem, lai tie spētu sekot līdzi notiekošajam uzņēmuma kravu pārvadājumos.

3.1.3. Saistība ar citiem dokumentiem

Dokuments sastādīts pēc LVS 72:1996[4] standarta „Ieteicamā prakse programmatūras projektējuma aprakstīšanai”.

3.2. Dekompozīcijas apraksts

3.2.1. Moduļu dekompozīcija

1. Autentifikācijas modulis
 - Lietotāja pieteikšanās sistēmā
2. Lietotāju pārvaldības modulis
 - Lietotāja izveide
 - Lietotāja rediģēšana
 - Lietotāja dzēšana
3. Uzdevumu pārvaldības modulis
 - Uzdevuma izveide
 - Uzdevuma izpilde

- Uzdevuma apskate

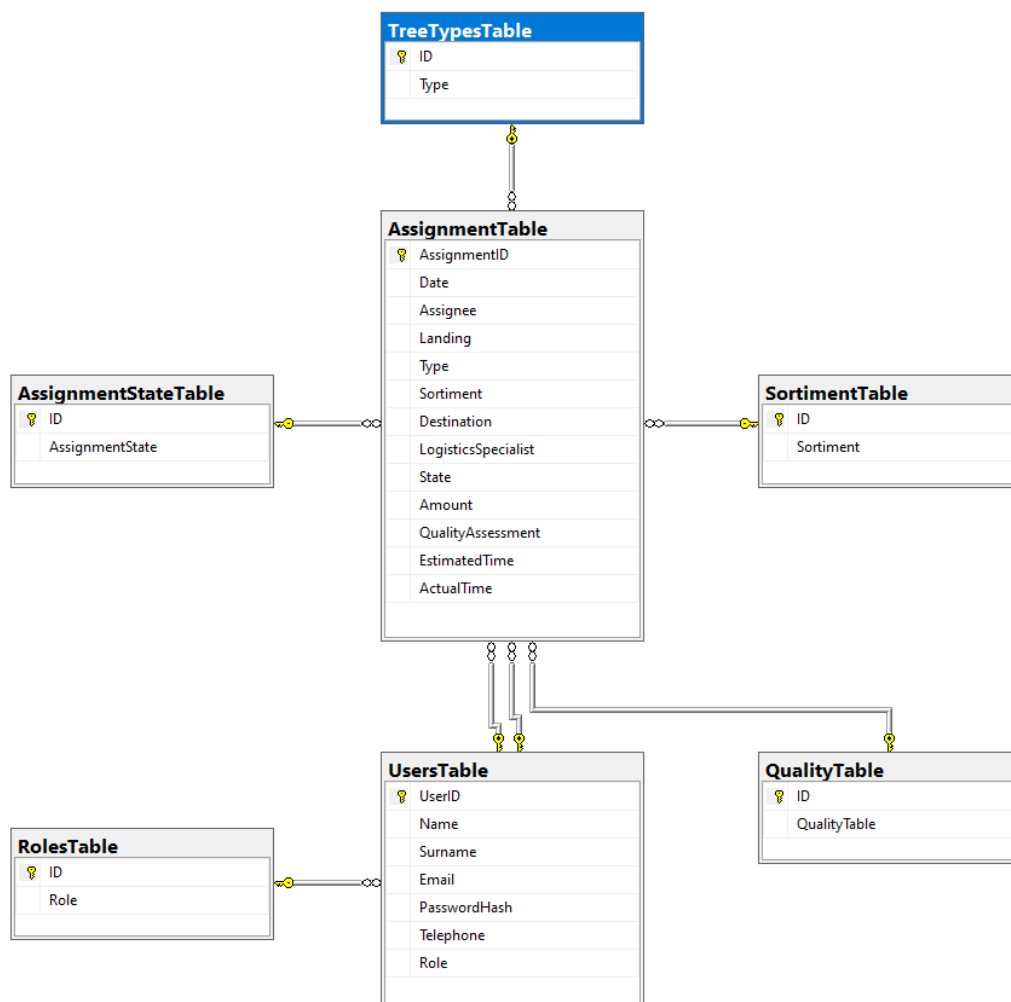
3.2.2 Datu dekompozīcija

Dati tiek glabāti SQL datubāzē. Sistēma sastāv no 7 tabulām:

1. Tabula “UsersTable” – satur datus par lietotājiem,
2. Tabula “RolesTable” – satur datus par lietotāju lomām sistēmā,
3. Tabula “AssignmentsTable” – satur datus par pārvadājumu uzdevumiem,
4. Tabula “TreeTypesTable Table” – satur datus par koku šķirnēm,
5. Tabula “SortimentTable” – satur datus par koksnes iedalījumu pēc sortimenta,
6. Tabula “AssignmentStateTable” – satur datus par pārvadājumu uzdevumu statusiem,
7. Tabula “QualityTable” – satur datus par koksnes kvalitātes novērtējumiem.

3.3. Datu modelis

3.3.1. ER realizācijas modelis



3.3.1.1. att. Fiziskais ER modelis

3.3.2. Datu tabulas

3.3.2.1. Tabula UsersTable

Tabula paredzēta lietotāju glabāšanai sistēmā. Rinda “Role” atsaucas uz tabulu “RolesTable”

Pašā sākumā tabula saturēs noklusētā administratora datus. Pēc autentificēšanās ar to būtu stringri ieteicams izveidot jaunu administratora lietotāju, un dzēst noklusēto.

3.3.2.1.1. Tabula

Datu tabula UsersTable

Lauks	Tips	Atslēga	Tukšs	Apraksts
UserID	INT	PK	Nē	Lietotāja identifikācijas numurs
Name	NVARCHAR (40)		Nē	Lietotāja vārds
Surname	NVARCHAR (40)		Nē	Lietotāja uzvārds
Email	NVARCHAR (40)		Nē	Uzņēmuma piešķirts e-pasts
PasswordHash	NVARCHAR (MAX)		Nē	Parole šifrēta SHA256
Telephone	INT		Nē	Telefona numurs
Role	INT	FK	Nē	Lietotāja loma sistēmā

3.3.2.2. Tabula RolesTable

Tabula “RolesTable” satur iespējamās lietotāju lomas sistēmā:

PK1 Administrators

PK2 Loģistikas speciālists

PK3 Kokvedēja vadītājs

3.3.2.2.1. Tabula

Datu tabula RolesTable

Lauks	Tips	Atslēga	Tukšs	Apraksts
ID	INT	PK	Nē	Lietotāja identifikācijas numurs
Role	NVARCHAR (20)		Nē	Lomas nosaukums

3.3.2.3. Tabula AssignmentTable

Tabula "AssignmentTable" satur sistēmā saglabātos pārvadājumu uzdevumus. "Assignee" un "LogisticsSpecialist" atsaucas uz tabulu "UsersTable". "Type", "Sortiment", "State", "QualityAssessment" atbilstoši atsaucas uz tabulām "TreeTypesTable", "AssignmentSortimentTable", "StateTable", "QualityTable". Vērtības, kuras drīkst būt tukšas tādas ir tādēļ, ka tās tiek aizpildītas pēc uzdevuma izveides.

3.3.2.3.1. Tabula

Datu tabula AssignmentTable

Lauks	Tips	Atslēga	Tukšs	Apraksts
AssignmentID	INT	PK	Nē	Uzdevuma identifikācijas numurs
Date	NVARCHAR (10)		Nē	Datums formātā dd/mm/yyyy
Assignee	INT	FK	Nē	Vadītājs, kuram uzdevums piešķirts
Landing	NVARCHAR (20)		Nē	Krautuves koordinātas
Type	INT	FK	Nē	Koksnes tips
Sortiment	INT	FK	Nē	Koksnes sortiments
Destination	NVARCHAR (100)		Nē	Galamērķis
LogisticsSpecialist	INT	FK	Nē	Atbildīgais loģistikas speciālists
State	INT	FK	Nē	Uzdevuma statuss
Amount	INT		Jā	Uzkrautais koksnes daudzums
QualityAssessment	INT	FK	Jā	Koksnes kvalitātes novērtējums
EstimatedTime	INT		Jā	Paredzētais laiks ceļā
ActualTime	INT		Jā	Patērētais laiks ceļā

3.3.2.4. Tabula TreeTypesTable

Tabula satur koksnes šķirnes pārvadājuma uzdevuma glabāšanai:

- PK1 Egle
- PK2 Priede
- PK3 Apse
- PK4 Bērzs
- PK5 Ozols
- PK6 Cits

3.3.2.4.1. Tabula

Datu tabula TreeTypesTable

Lauks	Tips	Atslēga	Tukšs	Apraksts
ID	INT	PK	Nē	Šķirnes identifikācijas numurs
Type	NVARCHAR (20)		Nē	Šķirnes nosaukums

3.3.2.5. Tabula SortimentTable

Tabula satur koksnes grupas pēc sortimenta pārvadājuma uzdevuma glabāšanai:

- PK1 Skuju koku zāģbaļķi
- PK2 Finierkluči
- PK3 Papīrmalka
- PK4 Lapu koku tara
- PK5 Skuju koku tara
- PK6 Malka
- PK7 Cits

3.3.2.5.1. Tabula

Datu tabula SortimentTable

Lauks	Tips	Atslēga	Tukšs	Apraksts
ID	INT	PK	Nē	Lietotāja identifikācijas numurs
Sortiment	NVARCHAR (30)		Nē	Koksnes iedalījums

3.3.2.6. Tabula AssignmentStateTable

Tabula satur uzdevuma statusu pārvadājuma uzdevuma glabāšanai:

PK1 Atvērts

PK2 Uzsākts

PK3 Pabeigts

3.3.2.6.1. Tabula

Datu tabula AssignmentStateTable

Lauks	Tips	Atslēga	Tukšs	Apraksts
ID	INT	PK	Nē	Identifikācijas numurs
State	NVARCHAR (20)		Nē	Uzdevuma status

3.3.2.7. Tabula QualityTable

Tabula satur kokvedēja vadītāja izdarītu koksnes novērtējumu. Tiek izdarīts pieņēmums, ka loģistikas speciālistam un kokvedēja vadītājam ir skaidrs, ko katrs novērtējums sevī ietver.

PK1 A

PK2 B

PK3 C

PK4 D

3.3.2.7.1. Tabula

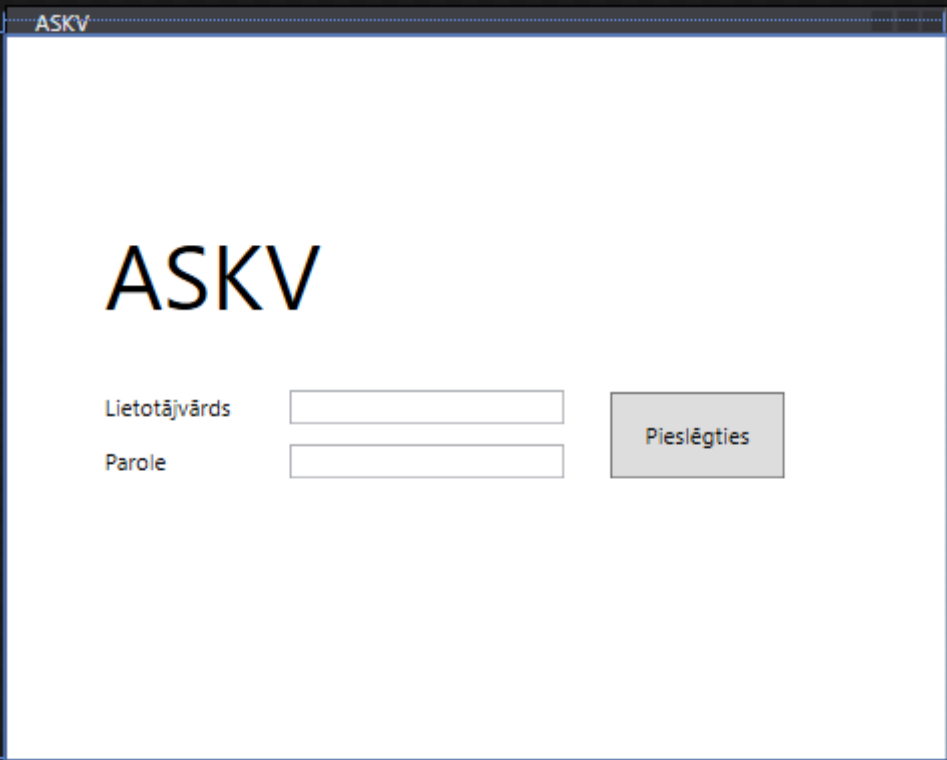
Datu tabula QualityTable

Lauks	Tips	Atslēga	Tukšs	Apraksts
ID	INT	PK	Nē	Identifikācijas numurs
Quality	NVARCHAR (1)		Nē	Kvalitātes novērtējums

3.4. Lietotāja saskarne

3.4.1. Autentifikācija

Pirmais logs ar kuru lietotājs saskaras atverot lietotni ir autentifikācijas logs. Aizpildot laukus ar pareizajiem datiem, lietotājs tiek novirzīts uz vienu no pamata logiem, atkarībā no viņa lomas sistēmā.



The image shows a screenshot of a web application's login page. The window has a title bar with the text 'ASKV'. The main content area features the word 'ASKV' in a large, bold, black font. Below this, there are two input fields: the first is labeled 'Lietotājvārds' (Username) and the second is labeled 'Parole' (Password). To the right of the password field is a grey button with the text 'Pieslēgties' (Login).

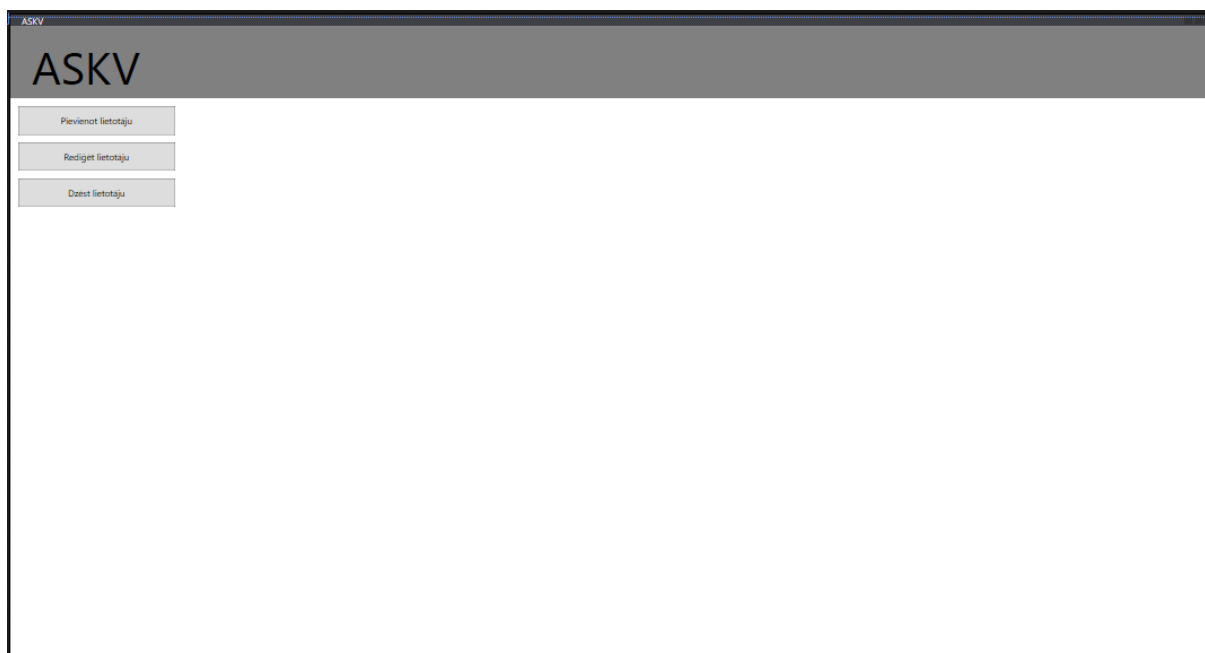
3.4.1.1 att. Lietotāja autentifikācijas logs

Tālāk lietotājs tiek novirzīts uz vienu no 3 pamata logiem, atkarībā pēc lietotāja lomas.

3.4.2. Administratora puse

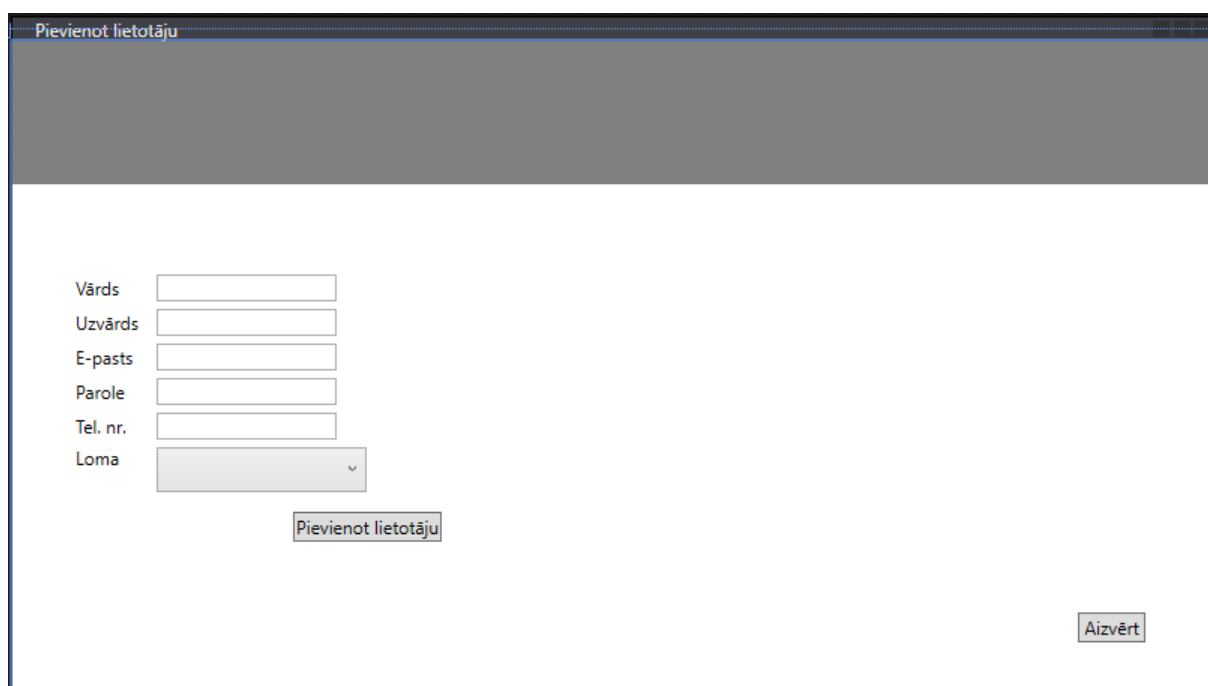
Ja lietotāja loma sistēmā ir administrators, pēc autentifikācijas viņš tiek novirzīts uz administratora pamata loga, kurā viņš var veikt trīs izvēles:

1. Pievienot lietotāju
2. Rediģēt lietotāju
3. Dzēst lietotāju



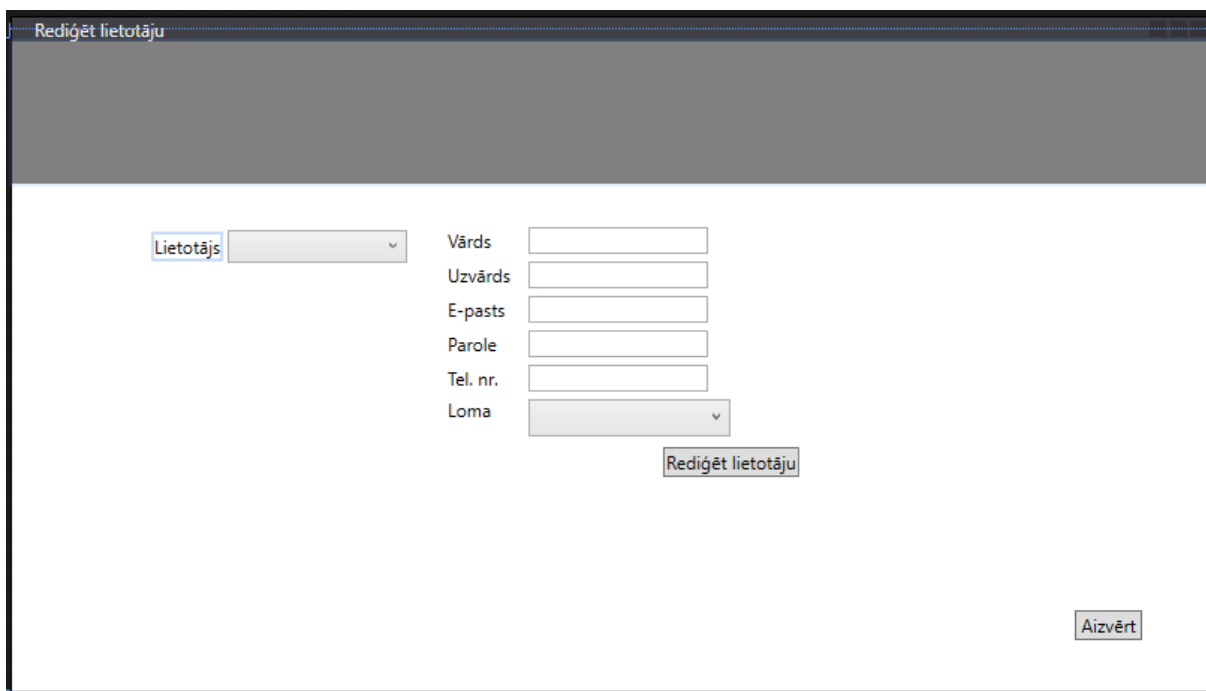
3.4.2.1 att. Administratora pamata logs

Izvēloties pievienot lietotāju, administrators tiek novirzīts uz jaunu logu, kurā pēc lauku aizpildīšanas lietotājs administrators var pievienot lietotāju.

The image shows a form titled 'Pievienot lietotāju' (Add user) in a dark grey header. The form contains several input fields: 'Vārds' (First name), 'Uzvārds' (Last name), 'E-pasts' (Email), 'Parole' (Password), and 'Tel. nr.' (Phone number). Below these is a dropdown menu for 'Loma' (Role). At the bottom of the form, there are two buttons: 'Pievienot lietotāju' (Add user) and 'Aizvērt' (Close).

3.4.2.2 att. Lietotāja pievienošanas logs

Izvēloties rediģēt lietotāju, administrators tiek novirzīts uz jaunu logu, kurā administrators pēc jau eksistējoša lietotāja atlasē var veikt jebkuru izmaiņu sistēmas lietotāja datos.



Rediģēt lietotāju

Lietotājs

Vārds

Uzvārds

E-pasts

Parole

Tel. nr.

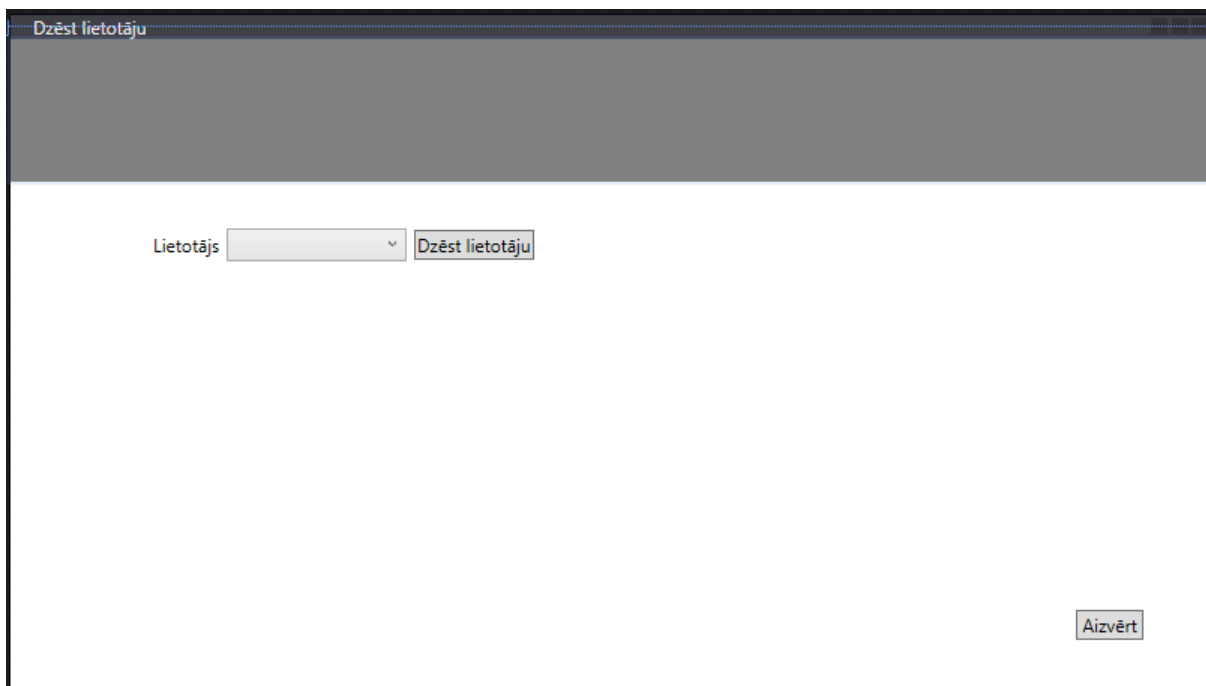
Loma

Rediģēt lietotāju

Aizvērt

3.4.2.3 att. Lietotāja rediģēšanas logs

Izvēloties dzēst lietotāju, administrators tiek novirzīts uz jaunu logu, kurā atlasot jau eksistējošu lietotāju sistēmā viņu var dzēst. Pirms dzēšanas administratoram darbība jāapstiprina.



Dzēst lietotāju

Lietotājs

Dzēst lietotāju

Aizvērt

3.4.2.4 att. Lietotāja dzēšanas logs

3.4.3. Kokvedēja vadītāja puse

Ja lietotāja loma sistemā ir kokvedēja vadītājs, pēc autentifikācijas viņš tiek novirzīts uz vadītāja pamata loga, kurā ja viņam ir piešķirti uzdevumi, viņš tos var apskatīt saraksta veidā un katru no tiem atvērt. Ja lietotājam nav piešķirti uzdevumi, zem teksta “Jūsu uzdevumi” parādās brīdinājums “Šobrīd Jums nav aktīvu uzdevumu.”(skat. 3.4.3.2. att.).



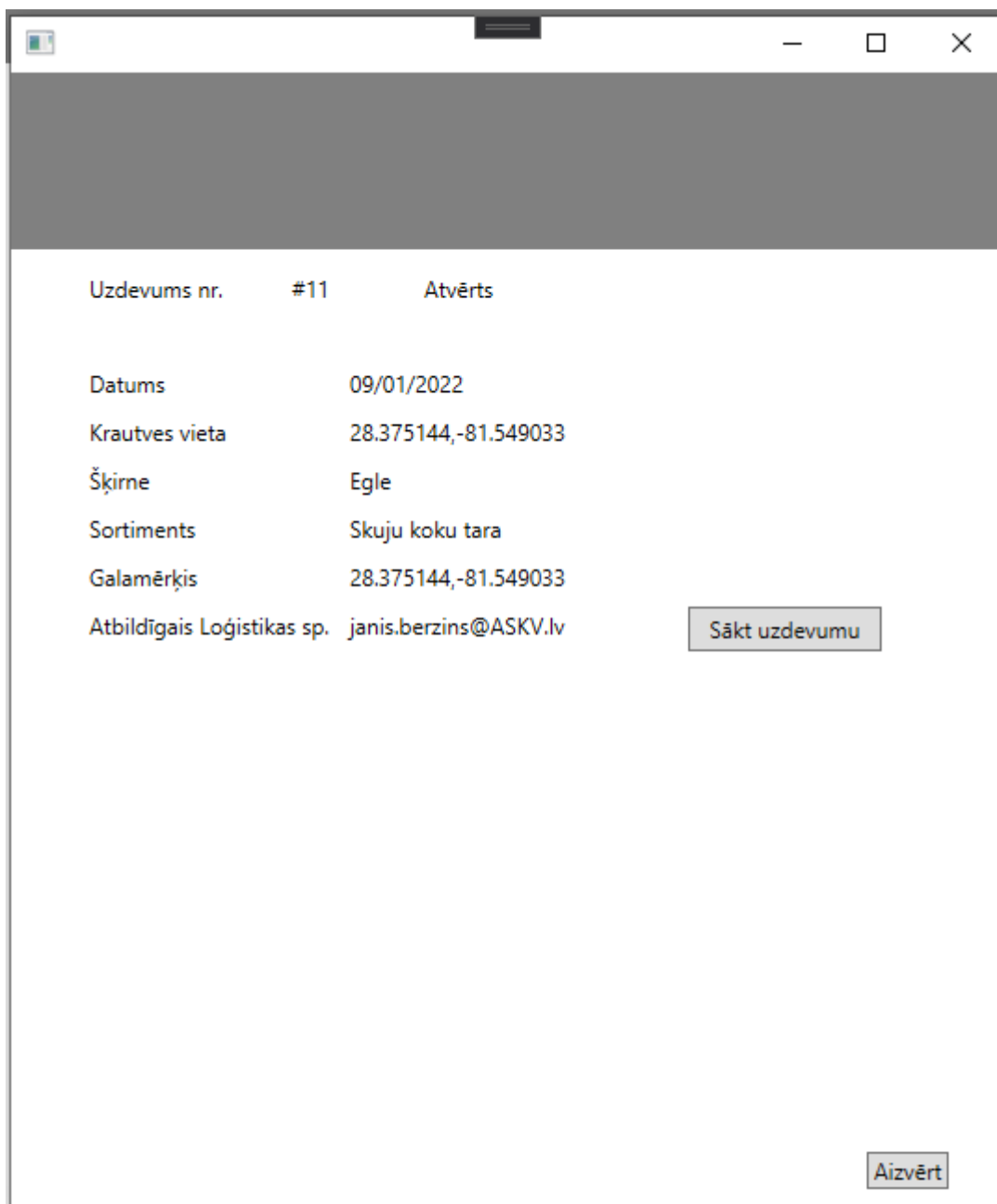
3.4.3.1 att. Vadītāja pamata logs

Jūsu uzdevumi

Šobrīd Jums nav aktīvu uzdevumu.

3.4.3.2 att. Paziņojums par neesošiem uzdevumiem

Izvēloties kādu no sarakstā esošajiem uzdevumiem, lietotājs tiek novirzīts uz pārvadājuma uzdevuma izpildes logu. Izvēloties sākt uzdevumu un apstiprinot veikto izvēli, lietotājs var uzsākt uzdevuma pildīšanu, kas ļauj sākt pievienot detaļas uzdevumam (*skat. 3.4.3.4 att.*).



The screenshot shows a window titled "Vadītāja uzdevuma logs" (Task Manager Log). The window contains a table with task details and two buttons: "Sākt uzdevumu" (Start Task) and "Aizvērt" (Close).

Uzdevums nr.	#11	Atvērts
Datums	09/01/2022	
Krautves vieta	28.375144,-81.549033	
Šķirne	Egle	
Sortiments	Skuju koku tara	
Galamērķis	28.375144,-81.549033	
Atbildīgais Loģistikas sp.	janis.berzins@ASKV.lv	

Sākt uzdevumu

Aizvērt

3.4.3.3 att. Vadītāja uzdevuma logs

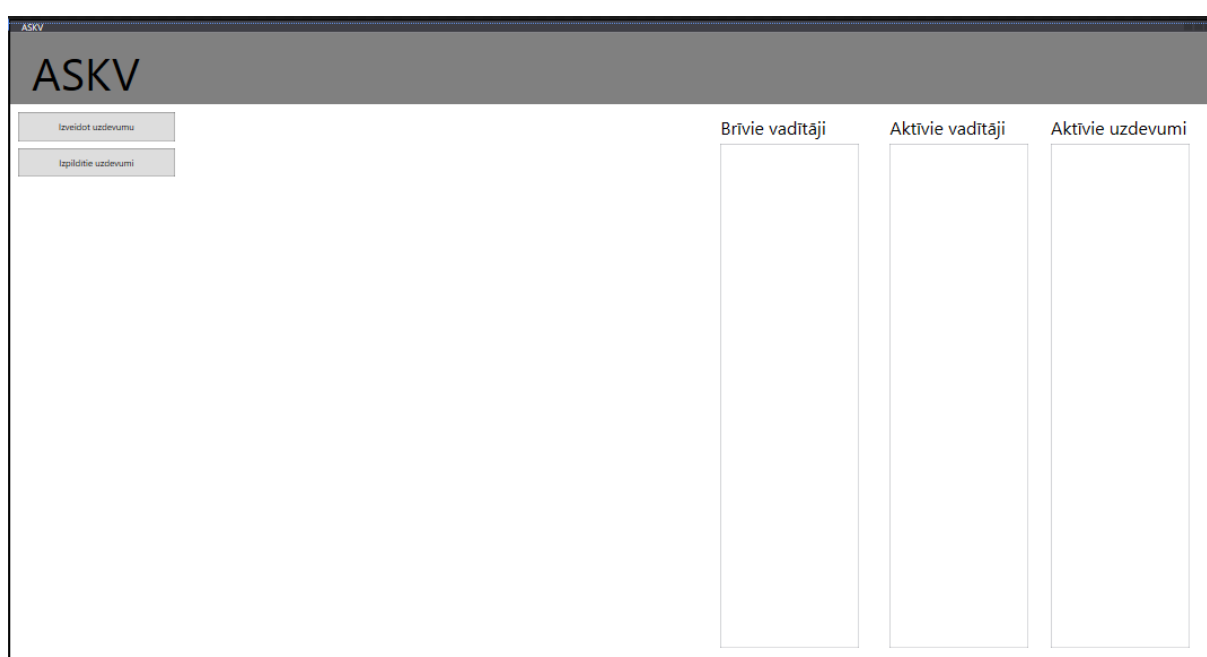
Uzdevums nr.	#11	Uzsākts
Datums	09/01/2022	
Krautves vieta	28.375144,-81.549033	
Šķirne	Egle	
Sortiments	Skuju koku tara	
Galamērķis	28.375144,-81.549033	
Atbildīgais Loģistikas sp.	janis.berzins@ASKV.lv	
Uzkrātais daudzums	<input type="text"/>	m3
Koksnes novērtējums	<input type="text"/>	
Paredzētais laiks ceļā	<input type="text"/>	h
Patērētais laiks ceļā	<input type="text"/>	h
		<input type="button" value="Pabeigt uzdevumu"/>
		<input type="button" value="Aizvērt"/>

3.4.3.4. att. Uzsākta uzdevuma logs

3.4.4. Loģistikas speciālista puse

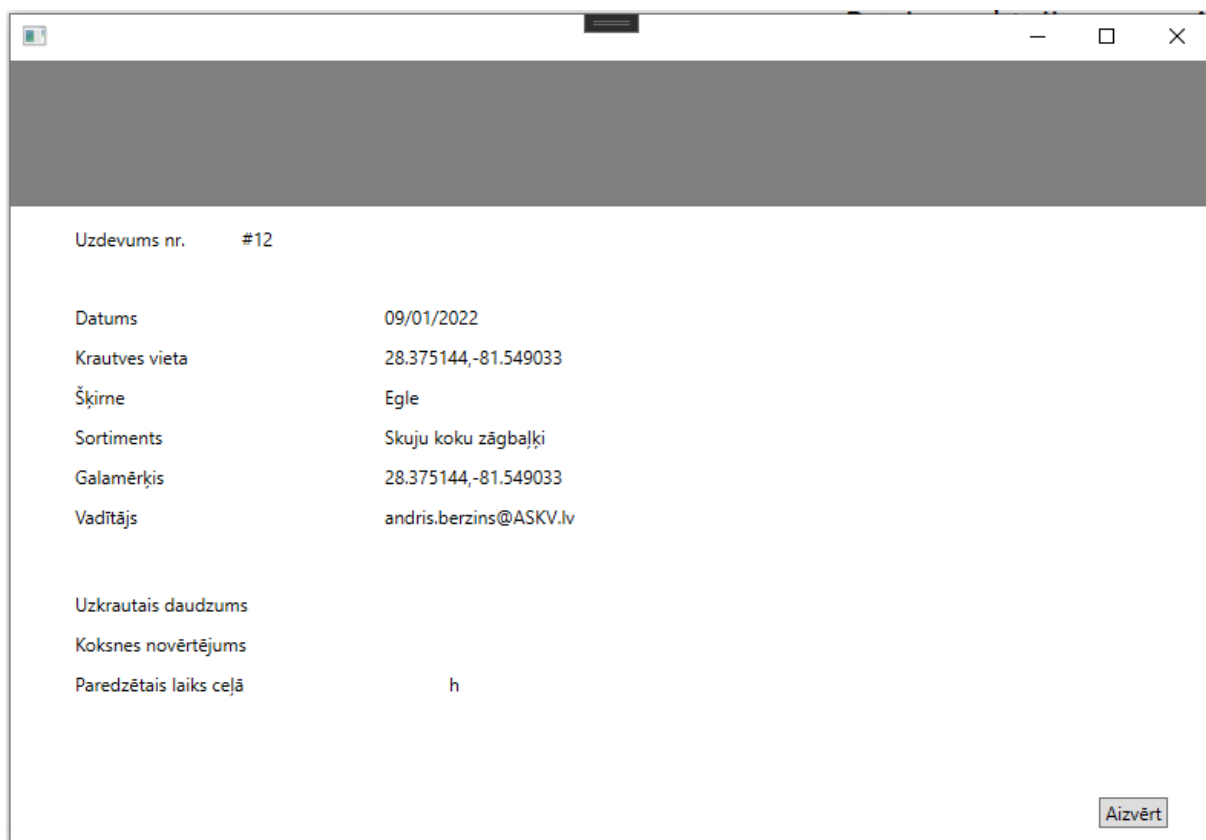
Ja lietotāja loma sistēmā ir loģistikas speciālists, pēc autentifikācijas viņš tiek novirzīts uz loģistikas speciālista pamata loga, kurā viņš var apskatīt sarakstus ar kokvedēja vadītājiem, kuri nav aizņemti un kokvedēja vadītājiem, kuriem ir uzsākts pārvaldījuma uzdevums. Tāpat arī redzams saraksts ar aktīviem uzdevumiem. Izvēloties jebkuru no aktīvo vadītāju identifikatoriem vai aktīvo uzdevumu identifikatoriem sarakstā, lietotājs tiek novirzīts uz atbilstošā pārvaldījuma uzdevuma logu.

Lietotājs no šī loga var arī izvēlēties izveidot uzdevumu vai apskatīt paveiktos uzdevumus.



3.4.4.1. att. Loģistikas speciālista pamata logs

Izvēloties kādu no iepriekš minēto sarakstu elementiem, lietotājs tiek novirzīts uz uzdevuma apskates logu.



The screenshot shows a window titled "Uzdevuma apskate" (Task Review). The window contains the following information:

Uzdevums nr.	#12
Datums	09/01/2022
Krautves vieta	28.375144,-81.549033
Šķirne	Egle
Sortiments	Skuju koku zāgbaļķi
Galamērķis	28.375144,-81.549033
Vadītājs	andris.berzins@ASKV.lv
Uzkrautais daudzums	
Koksnes novērtējums	
Paredzētais laiks ceļā	h

In the bottom right corner of the window, there is a button labeled "Aizvērt" (Close).

3.4.4.2. att. Uzdevuma apskates logs

Izvēloties izveidot jaunu uzdevumu, lietotājs tiek novirzīts uz uzdevuma izveides logu. Tajā, ja visi lauki ir aizpildīti, loģistikas speciālists var izveidot uzdevumu.

Izveidot uzdevumu

Krautuve

Šķirne

Sortiments

Galamērķis

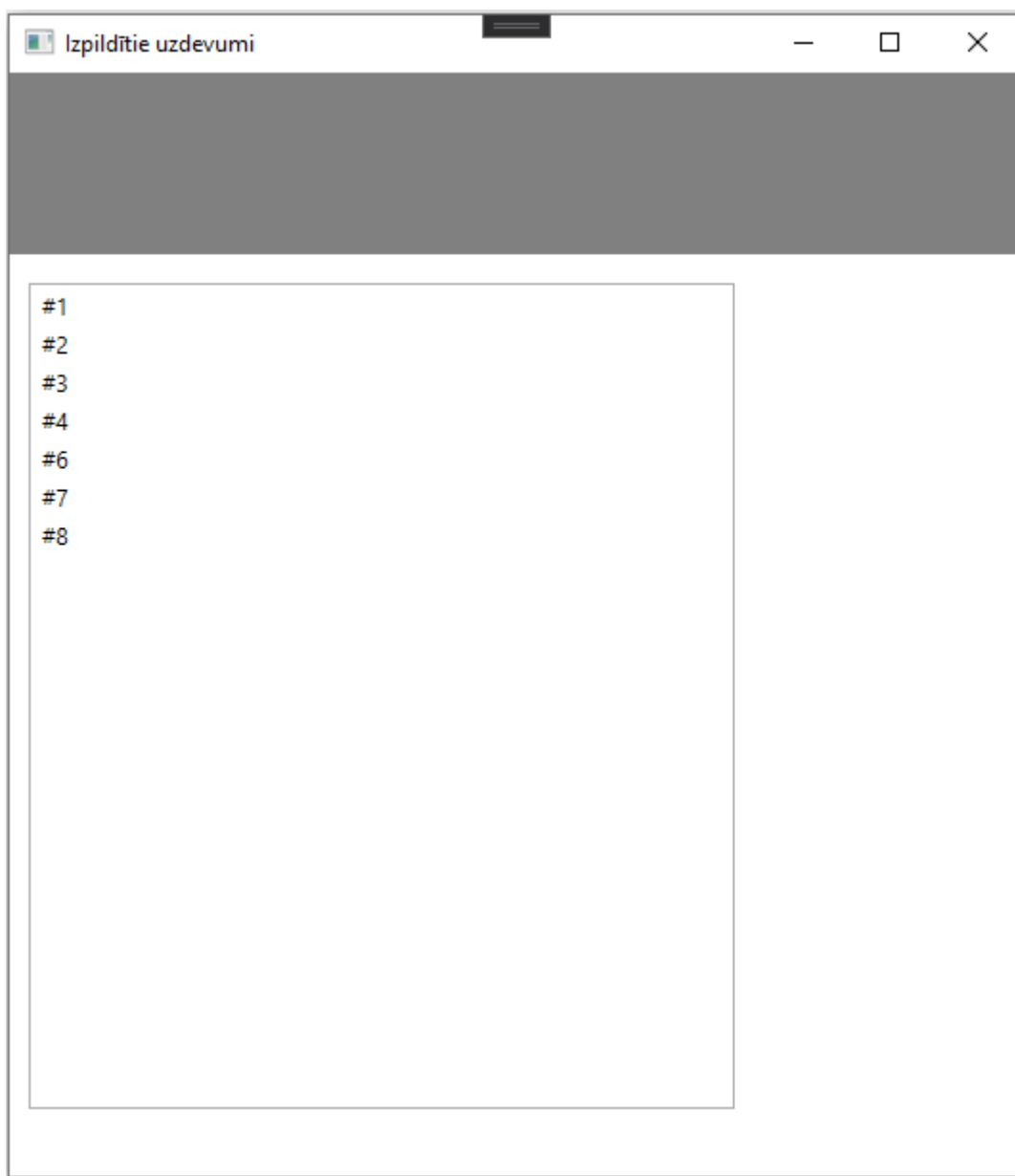
Vadītājs

Izveidot uzdevumu

Aizvērt

3.4.4.3. att. Uzdevuma izveides logs

Pamata logā, izvēloties apskatīt izpildītos uzdevumus, lietotājs nonāk jaunā logā ar izpildīto uzdevumu sarakstu. Līdzīgi kā pamata logā, izvēloties kādu no uzdevumu identifikatoriem, lietotājs tiek novirzīts uz logu, kurā var apskatīt izvēlēta uzdevuma detaļas.



3.4.4.4. att. Izpildīto uzdevumu logs

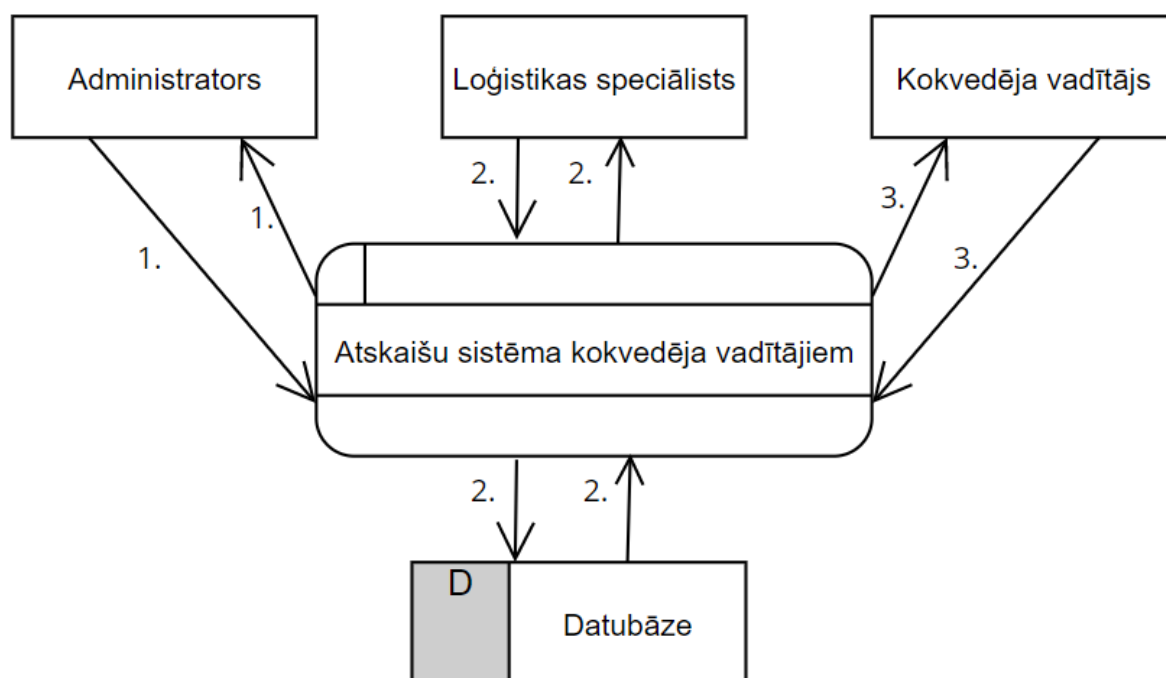
3.5. Moduļu projektējums

3.5.1. Starpmoduļu atkarības

Šajā nodaļā tiek apskatītas atkarības starp sistēmas moduļiem.

3.5.2. Konteksta līmeņa datu plūsmas diagramma

Šajā līmenī tiek apskatīta informācijas aprīte starp sistēmas lietotājiem un pašu sistēmu. Katram lietotājam tiek iedalīta sava loma sistēmā.

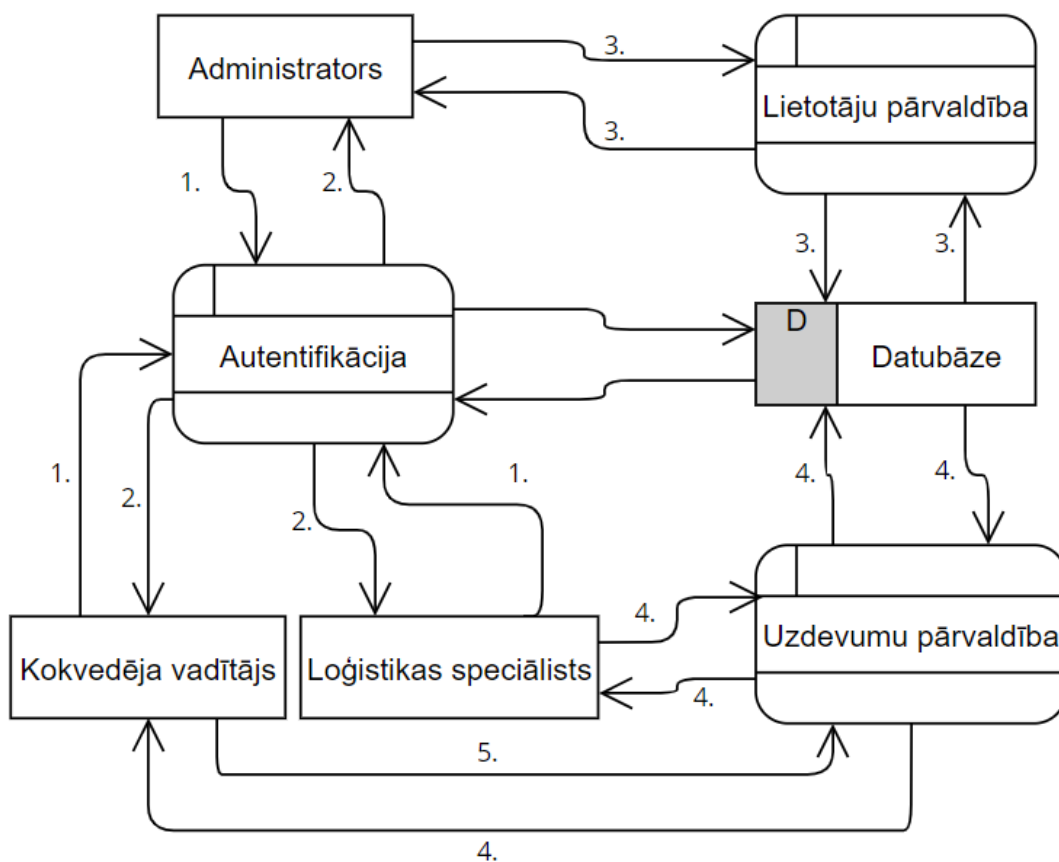


3.5.2.1. att. Konteksta līmeņa datu plūsmas diagramma

1. Dati par sistēmas lietotājiem;
2. Dati par pārvadājumu uzdevumiem, kokvedēja vadītājiem;
3. Dati par pārvadājumu uzdevumiem.

3.5.3. Pirmā līmeņa datu plūsmas diagramma

Šajā līmenī tiek apskatītas sistēmas galvenās funkcijas, datu plūsmas starp minētajām funkcijām un lietotājiem.

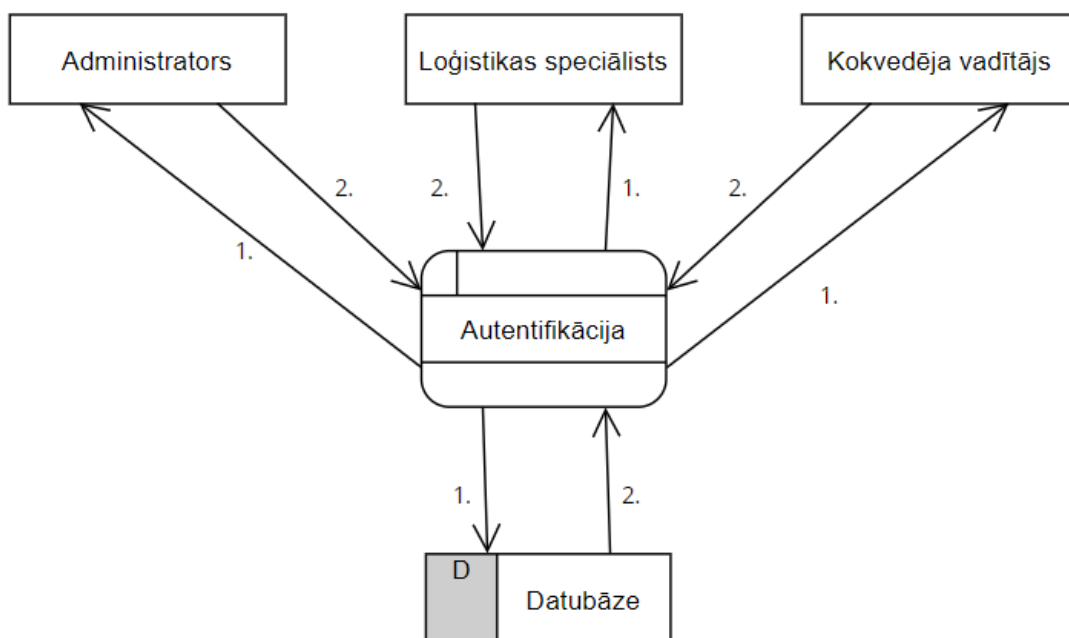


3.5.3.1. att. Pirmā līmeņa datu plūsmas diagramma

1. E-pasts, parole;
2. Lietotāja loma sistēmā;
3. Lietotāja identifikators, vārds, uzvārds, e-pasts, loma;
4. Uzdevuma identifikators, datums, lietotāja identifikators, krautuves koordinātas; koksnes tips, koksnes iedalījumu pēc sortimenta, galamērķis, uzdevuma statuss, uzkrātais daudzums, kvalitātes novērtējums, paredzētais laiks, patērētais laiks.
5. Uzkrātais daudzums, kvalitātes novērtējums, paredzētais laiks, patērētais laiks.

3.5.4. Otrā līmeņa datu plūsmas diagrammas

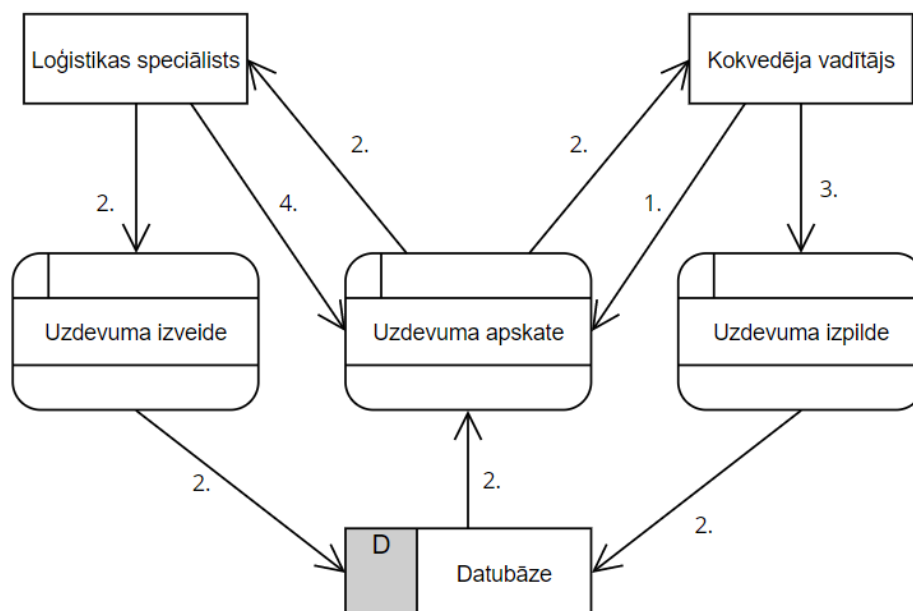
Diagrammā (*skat. 3.5.4.1 att.*) tiek apskatītas sistēmas funkcijas, kuras pieejamas funkcijai “Autentifikācija”.



3.5.4.1. att. Otrā. līmeņa datu plūsmas diagramma funkcijai “Autentifikācija”

1. E-pasts, parole;
2. Lietotāja loma

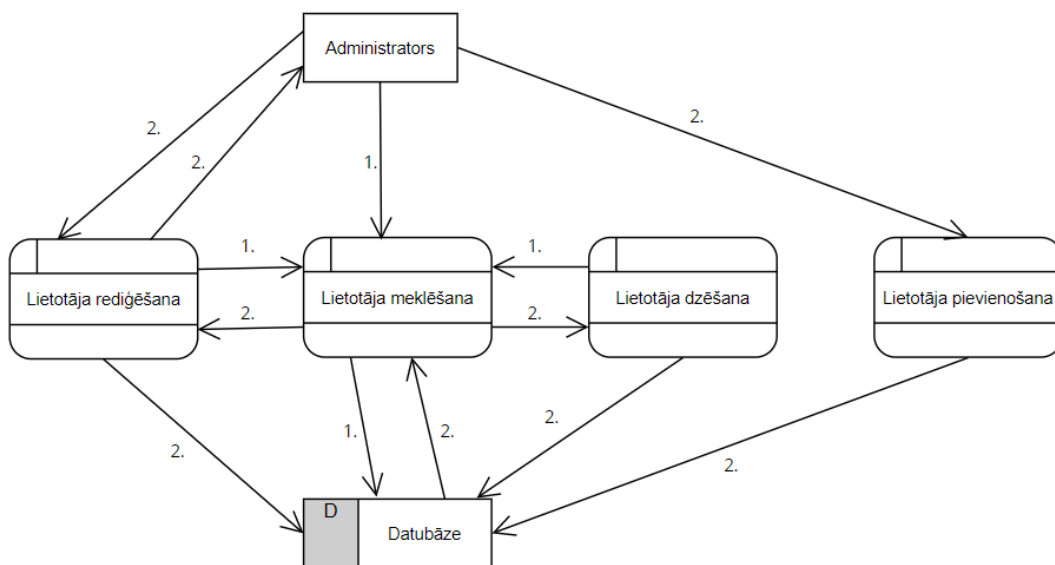
Diagrammā (skat. 3.5.4.2 att.) tiek apskatītas sistēmas funkcijas, kuras pieejamas funkcijai “Uzdevuma pārvaldība”.



3.5.4.2. att. Otrā līmeņa datu plūsmas diagramma funkcijai “Uzdevumu pārvalde”

1. Pārvaldājuma uzdevuma identifikators;
2. Uzdevuma identifikators, datums, lietotāja identifikators, krātuves koordinātas; koksnes tips, koksnes iedalījumu pēc sortimenta, galamērķis, uzdevuma statuss, uzkrātais daudzums, kvalitātes novērtējums, paredzētais laiks, patērētais laiks.
3. Uzkrātais daudzums, kvalitātes novērtējums, paredzētais laiks, patērētais laiks.
4. Pārvaldājuma uzdevuma identifikators, lietotāja identifikators.

Diagrammā (skat. 3.5.4.3 att.) tiek apskatītas sistēmas funkcijas, kuras pieejamas funkcijai “Lietotāja pārvaldība”. Šī funkcija pieejama tikai lietotāja lomai “Administrators”



3.5.4.3. att. Otrā līmeņa datu plūsmas diagramma funkcijai “Lietotāju pārvalde”

1. Lietotāja identifikators,
2. Lietotāja identifikators, vārds, uzvārds, e-pasts, loma.

4. TESTĒŠANAS DOKUMENTĀCIJA

4.1. Testēšanas plāns

Testēšanai tiks izmantota melnās kastes testēšanas metode.

Testēti tiks visi sistēmas moduļi.

Pēc katras funkcijas realizācijas tā uzreiz tika pārbaudīta un kļūdas gadījumā izlabota.

Tas pats tik darīts pēc tam, kad bija izstrādātas visas sistēmas funkcijas, tādēļ veiktie testi balstās uz grūtāk ieraugāmām kļūdām sistēmas darbībā.

4.2. Testēšanas žurnāls

4.2.1. Autentifikācija

4.2.1.1. Tabula

Testēšanas tabula "Autentifikācija"

Testa nr.	Testa apraksts	Sagaidāmais rezultāts	Statuss
1.	Pieslēgšanās – Nav aizpildīti visi lauki.	Kļūdas paziņojums par nekorektiem datiem.	ok
2.	Pieslēgšanās – E-pasts nepareizā formā.	Kļūdas paziņojums par nekorektiem datiem.	ok
3.	Pieslēgšanās – Lietotājs nav atrasts.	Paziņojums par neizdevušos meklēšanu.	ok

4.2.2. Lietotāju pārvaldība

4.2.2.1. Tabula

Testēšanas tabula "Lietotāju pārvaldība"

Testa nr.	Testa apraksts	Sagaidāmais rezultāts	Statuss
1.	Lietotāja izveide – Nav aizpildīti visi lauki.	Kļūdas paziņojums par nekorektiem datiem.	ok
2.	Lietotāja izveide – E-pasts nepareizā formā.	Kļūdas paziņojums par nekorektiem datiem.	ok
3.	Lietotāja izveide – Ievadītā vērtība pārāk gara.	Kļūdas paziņojums par nekorektiem datiem.	ok
4.	Lietotāja izveide – Ievadītais vārds ir skaitlis	Kļūdas paziņojums par nekorektiem datiem.	ok
5.	Lietotāja izveide – Ievadītais uzvārds ir skaitlis	Kļūdas paziņojums par nekorektiem datiem.	ok
6.	Lietotāja rediģēšana – Nav izvēlēts lietotājs	Kļūdas paziņojums par nekorektiem datiem.	ok

7.	Lietotāja rediģēšana – E-pasts nepareizā formā.	Kļūdas paziņojums par nekorektiem datiem.	ok
8.	Lietotāja rediģēšana – Ievadītā vērtība pārāk gara.	Kļūdas paziņojums par nekorektiem datiem.	ok
9.	Lietotāja rediģēšana – Ievadītais vārds ir skaitlis	Kļūdas paziņojums par nekorektiem datiem.	ok
10.	Lietotāja rediģēšana – Ievadītais uzvārds ir skaitlis	Kļūdas paziņojums par nekorektiem datiem.	ok
11.	Lietotāja dzēšana – Nav izvēlēts lietotājs.	Kļūdas paziņojums par nekorektiem datiem.	ok

4.2.3. Uzdevumu pārvaldība

4.2.3.1. Tabula

Testēšanas tabula “Uzdevumu pārvaldība”

Testa nr.	Testa apraksts	Sagaidāmais rezultāts	Statuss
1.	Uzdevuma izveide – Nav aizpildīti visi lauki.	Kļūdas paziņojums par nekorektiem datiem.	ok
2.	Uzdevuma izveide – Ievadītā vērtība pārāk gara.	Kļūdas paziņojums par nekorektiem datiem.	ok
3.	Uzdevuma izveide – Šķirne neatbilst sortimentam.	Kļūdas paziņojums par šķirnes neatbilstību sortimentam.	ok
4.	Uzdevuma izpilde – Ievadītā vērtība nav skaitlis.	Kļūdas paziņojums par nekorektiem datiem.	ok
5.	Uzdevuma izpilde – Ievadītā vērtība ir negatīva.	Kļūdas paziņojums par nekorektiem datiem.	ok
6.	Uzdevuma izpilde – Ievadītais kravas izmērs pārsniedz limitu.	Kļūdas paziņojums par pārsniegtu kravas izmēra limitu.	ok

5. SECINĀJUMI

Izstrādājot kvalifikācijas darbu tika realizēta funkcionāla lietotne. Pēc kvalifikācijas darba izstrādes, esmu sapratis, cik liela nozīme programmatūras izstrādē ir programmatūras prasību specifikācijai un darba plānošanai kopumā.

Izstrādājot sistēmu "Atskaišu sistema kokvedēja vadītājiem" ieguvu pieredzi darbā ar C# programmēšanas valodu un "Microsoft SQL Server" datu bāzes pārvaldības sistēmu, kā arī guva ieskatu "Entity Framework" relāciju kartēšanas sistēmas darbībā.

Esmu pārliecināts ka gūtās zināšanas, izstrādājot kvalifikācijas, atstās lielu iespaidu manā turpmākajā izaugsmē.

6. IZMANTOTĀ LITERATŪRA

1. COCOMO Calculator [tiešsaiste] – [atsauce 03.01.2022].:
<https://strs.grc.nasa.gov/repository/forms/cocomo-calculation/>
2. Microsoft Documentation [tiešsaite] – [atsauce 28.12.2021].:
<https://docs.microsoft.com/en-us/documentation/>
3. LVS 68:1996 „Programmatūras prasību specifikācijas ceļvedis”
4. LVS 72:1996 „Ieteicamā prakse programmatūras projektējuma aprakstīšanai”

7. PROGRAMMATŪRAS KODS

7.1. Ievads

Šajā nodaļā ir iekļauts programmatūras kods. Lai ietaupītu lappušu skaitu, apskatīts tiek tikai kods, kas saistīts ar kokvedēja vadītāja sistēmas pusi.

7.2 LoginWindow.xaml.cs

Loģika autentifikācijas logam

```
using System.Windows;
using System.ComponentModel.DataAnnotations;
using ASKVProject.Models;

namespace ASKVProject
{
    /// <summary>
    /// Interaction logic for LoginWindow.xaml
    /// </summary>
    public partial class LoginWindow : Window
    {
        DBContext context = new DBContext(); // Jauns datubāzes konteksts darbībām ar
        datubāzes modeļiem
        Calculations calculations = new Calculations(); // Jauna klases calculations
        instance paredzēta lai izsauktu void HashPassword(string password) funkciju
        public LoginWindow()
        {
            InitializeComponent();
            WindowStartupLocation = System.Windows.WindowStartupLocation.CenterScreen;
            // Nodrošina jauna loga centrēšanu
        }

        private void LoginBtn_Click(object sender, RoutedEventArgs e)
        {
            string username = TextBoxUsername.Text; // Lietotāja ievadītais e-pasts
            string password = TextBoxPassword.Password; // Lietotāja ievadītā parole
            string hash = calculations.HashPassword(password); // Pārveido lietotāja
            ievadīto paroli Sha256 šifrējumā
            bool userNotFound = false; // Mainīgais kļūst patiens gadījumā, kad
            UsersTable ir pārstaigāts un lietotājs nav atrasts
            if (checkIsValid(username, password) == true) // Ja patiens, lauki nav
            tukši un e-pasts ir pareizā formātā
            {
                foreach (UsersTable ut in context.UsersTables) // Cikls pārstaigā
                lietotāju tabulu un ja atrod lietotāju datubāzes tabulā, atbilstoši viņa lomai sistēmā
                atver sistēmas pamata logu
                {
                    if (username == ut.Email && hash == ut.PasswordHash)
                    {
                        int userRole = ut.Role;
                        int userId = ut.UserId; // Sistēmas pašreizējā lietotāja
                        identifikācijas numurs
                        if (userRole == 1) // Ja lietotājs ir sistēmas administrators
                        {
                            MainWindowAdmin mw = new MainWindowAdmin(userId);
                            mw.Show();
                            mw.Owner = Owner;
                            Close();
                        }
                        if (userRole == 2) // Ja lietotājs ir loģistikas speciālists
                        {
                            MainWindowLogistics mw = new MainWindowLogistics(userId);
```

```

        mw.Show();
        mw.Owner = Owner;
        Close();
    }
    if (userRole == 3) // Ja lietotājs ir kokvedēja vadītājs
    {
        MainWindowDriver mw = new MainWindowDriver(userId);
        mw.Show();
        mw.Owner = Owner;
        Close();
    }
}
else // Ja lietotājs nav atrasts, izdod atbilstošu paziņojumu
{
    userNotFound = true;
    TxtIncorrectInput.Text = "Lietotājs nav atrasts"; // Teksts
zem ievades laukiem, pēc noklusējuma paslēpts
}
}
if (userNotFound == true) // Parāda lietotājam kļūdas paziņojumu
{
    TxtIncorrectInput.Visibility = Visibility.Visible;
}
}
else // Ja funkcija checkIsValid(string username, string password) atgriež
false, ir notikusi kļūda lietotāja ievadē
{
    TxtIncorrectInput.Visibility = Visibility.Visible; // Parāda
lietotājam kļūdas paziņojumu
}
}
// Funkcija lietotāja ievades pārbaudei
private bool checkIsValid(string username, string password)
{
    bool checkEmail = new EmailAddressAttribute().IsValid(username); //
Mainīgais e-pasta formāta pārbaudei
    if (username == "" || password == "") // Viens vai abi no ievades laukiem
atstāti tukši
    {
        TxtIncorrectInput.Text = "Lūdzu aizpildiet visus laukus";
        return false;
    }
    else if (checkEmail == false) // E-pasts ir nepareizā formātā
    {
        TxtIncorrectInput.Text = "E-pasts ir nepareizā formātā";
        return false;
    }
    else
    {
        return true; // Lietotāja ievadītie dati ir korekti
    }
}
}
}
}

```

7.3. MainWindowDriver.xaml.cs

Loģika kokvedēja vadītāja galvenajam logam

```
using System.Collections.Generic;
using System.Windows;
using System.Windows.Controls;
using ASKVProject.Models;

namespace ASKVProject
{
    /// <summary>
    /// Interaction logic for MainWindowDriver.xaml
    /// </summary>
    public partial class MainWindowDriver : Window
    {
        DBContext context = new DBContext(); // Jauns datubāzes konteksts darbībām ar
        datubāzes modeļiem
        AssignmentTable assignmentTable = new AssignmentTable(); // Jauns datubāzes
        tabulas assignmentTable modelis. Izmantots lai izvēlēto uzdevumu varētu padot uzdevuma
        logam
        List<AssignmentTable> atTable = new List<AssignmentTable>(); // Saraksts, kurš
        satur visus uzdevumu objektus, kuri piešķirti sistēmas aktīvajam lietotājam un nav
        pabeigti
        public MainWindowDriver(int userId) // userId = sistēmas aktīvā lietotāja
        identifikators
        {
            InitializeComponent();
            WindowStartupLocation = System.Windows.WindowStartupLocation.CenterScreen;
            // Nodrošina jaunā loga centrēšanu
            LoadInitialData(userId); // Funkcija datu ielādēšanai un atjaunošanai.
            ļauj uzdotajam uzdevumam pazust no pamata loga, ja uzdevums tiek pabeigts
        }

        private void lstAssignments_SelectionChanged(object sender,
        SelectionChangedEventArgs e)
        {
            string removeHashtag;
            if (lstAssignments.SelectedItem != null) { //Pārbauda vai izvēlētais
            assignment nav null sakarā ar tā pabeigšanu
                removeHashtag = lstAssignments.SelectedItem.ToString();
                removeHashtag = removeHashtag.Remove(0, 1);
                // Noņem simbolu "#" no uzdevuma identifikatora, lai varētu to
                salīdzināt ar datubāzes PK
                int.TryParse(removeHashtag, out int assignmentId); // Parsē string
                tipa mainīgo par int, lai varētu to salīdzināt ar datubāzes PK
                foreach (AssignmentTable assignmentTable in atTable) // Cikls, kurš
                iet cauri uzdevumu sarakstam, līdz atrod to uzdevumu, kuru aktīvais sistēmas lietotājs
                izvēlējās
                {
                    if (assignmentTable.AssignmentId == assignmentId)
                    {
                        var nw = new AssignmentWindowDriver(assignmentTable, context);
                        // Lietotāja izvēlētais uzdevums tiek padots apstrādei vadītāja uzdevuma logā
                        nw.Owner = this;
                        nw.Show();
                        break; // Pārstāj ciklu, jo uzdevums ir atrasts
                    }
                }
            }
        }

        public void LoadInitialData(int uID) // Funkcija notīra pamata logā redzamos
        uzdevumu identifikatorus un ielādē visus uzdevumus, kuri piešķirti sistēmas aktīvajam
        lietotājam un nav pabeigti
    }
}
```

```

    {
        lstAssignments.ItemsSource = null;
        lstAssignments.Items.Clear();
        List<string> assignmentNumbers = new List<string>(); // Saraksts, kurš
        satur visus uzdevumu identifikatorus, kuri piešķirti sistēmas aktīvajam lietotājam un
        nav pabeigti
        string assignmentNumber;
        foreach (AssignmentTable at in context.AssignmentTables) // Cikls, kurš
        pārstaigā uzdevumu tabulu
        {
            if (at.Assignee == uID) // Ja aktīvais sistēmas lietotājs sakrīt ar
            lietotāju, kuram uzdevums piešķirts, notiek tālāka datu pārbaude
            {
                if (at.State != 3) // Ja uzdevums nav pabeigts
                {
                    assignmentNumber = "#" + at.AssignmentId; // Mainīgais, lai
                    uzlabotu uzdevuma identifikatora vizuālo reprezentāciju
                    assignmentNumbers.Add(assignmentNumber); // Pievieno "vizuāli
                    uzlaboto" uzdevuma identifikatoru uzdevumu identifikatoru sarakstam
                    atTable.Add(at); // Saglabā atrasto uzdevumu sarakstā
                }
            }
        }
        if (assignmentNumbers.Count != 0) // Ja ir atrasts kaut viens sistēmas
        aktīvajam lietotājam piešķirts uzdevums, tas tiek izdots lietotājam
        {
            lstAssignments.ItemsSource = assignmentNumbers;
        }

        if (lstAssignments.Items.Count == 0) // Ja nav atrasts neviens sistēmas
        aktīvajam lietotājam piešķirts uzdevums, tas tiek izdots lietotājam, tiek izdots
        paziņojums par neesošiem uzdevumiem
        {
            txtNoAssignments.Visibility = Visibility.Visible;
            lstAssignments.Visibility = Visibility.Collapsed;
        }
    }
}

```

7.4. AssignmentWindowDriver.xaml.cs

Loģika kokvedēja vadītāja uzdevuma logam

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Linq;
using System.Text;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Data;
using System.Windows.Documents;
using System.Windows.Input;
using System.Windows.Media;
using System.Windows.Media.Imaging;
using System.Windows.Shapes;
using ASKVProject.Models;
using ASKVProject.Windows.DriverSide;

namespace ASKVProject
{
    /// <summary>
    /// Interaction logic for MainWindow.xaml
    /// </summary>
    public partial class AssignmentWindowDriver : Window
    {
        DBContext dbctxt = new DBContext(); // Jauns datubāzes konteksts darbībām ar
        datubāzes modeļiem
        AssignmentTable assignment = new AssignmentTable(); // Jauns datubāzes
        uzdevumu tabulas objekts, kurā glabāt objektu no iepriekšējā loga apstrādei
        Calculations calculations = new Calculations(); // Jauna klases calculations
        instance paredzēta lai izsauktu void ForeignKeyToString, lai datubāzes ārējās atslēgas
        lietotājam tiktu padotas saprotamā veidā
        public AssignmentWindowDriver(AssignmentTable at, DBContext context) // No
        iepriekšējā loga saņem uzdevumu tabulas objektu tā apstrādei šajā logā
        {
            InitializeComponent();
            WindowStartupLocation = System.Windows.WindowStartupLocation.CenterScreen;
            // Nodrošina jaunā loga centrēšanu
            assignment = at; // Uzdevumu tabulas objektu saglabā ārpus funkcijas, lai
            pārējās funkcijas varētu veikt darbības ar to
            InitialDataLoader(at); // Atvērtajā logā ielādē datus no datubāzes
            if (at.State == 2) // Pārbauda, vai uzdevums jau uzsākts
            {
                btnStartAssignment.Visibility = Visibility.Collapsed; // Paslēpj pogu
                "Uzsākt darbu", jo uzdevums jau ir uzsākts
                ControlsLoaderBatch1(); // Parāda lietotājam aizpildāmos laukus
                uzdevuma veikšanai
                if (at.Amount != null)
                {
                    ControlsLoaderBatch2(); // Ja Amount != null, kokvedēja vadītājs
                    ir ceļā uz galamērķi, jo krava ir uzkrauta. Tiek parādīti lauki laika ievadei.
                }
                dbctxt = context; // Saglabā no iepriekšējā loga saņemto datubāzes
                kontekstu mainīgajā ārpus funkcijas, lai pārējās funkcijas varētu veikt darbības ar to
            }
            private void btnStartAssignment_Click(object sender, RoutedEventArgs e) //
            Pogas klikšķa event uzdevuma uzsākšanai
            {
                MessageBoxResult messageBoxResult = System.Windows.MessageBox.Show("Uzsākt
                uzdevumu?", "", System.Windows.MessageBoxButton.YesNo); // Uznirstošs logs uzdevuma
                uzsākšanas apstiprināšanai
                if (messageBoxResult == MessageBoxResult.Yes) // Ja lietotājs apstiprina
```

```

        {
            txtAssignmentState.Text = "Uzsākts"; // Atsauce lietotājam par
            uzdevuma statusa maiņu
            btnStartAssignment.Visibility = Visibility.Collapsed; // Paslēpj pogu
            "Uzsākt darbu", jo uzdevums ir uzsākts
            assignment.State = 2; // Uzdevuma statuss mainās uz "Uzsākts"
            dbctxt.SaveChanges(); // Saglabā izmaiņas datubāzē
            ControlsLoaderBatch1(); // Parāda lietotājam aizpildāmos laukus
            uzdevuma veikšanai
        }
    }

    private void btnConfirmChanges_Click(object sender, RoutedEventArgs e) //
    Saglabā lietotāja ievadītos datus datubāzē, string mainīgos parsējot uz int
    {
        int.TryParse(txtBoxAmount.Text, out int amount);
        assignment.Amount = amount;
        if (cmbQuality.Text == "A") { assignment.QualityAssessment = 1; }
        if (cmbQuality.Text == "B") { assignment.QualityAssessment = 2; }
        if (cmbQuality.Text == "C") { assignment.QualityAssessment = 3; }
        if (cmbQuality.Text == "D") { assignment.QualityAssessment = 4; }
        int.TryParse(txtBoxEstimatedTime.Text, out int estimatedTime);
        assignment.ActualTime = estimatedTime;
        if (checkIsValid(amount, estimatedTime)) // Pārbauda vai ievadītie
        skaitļi nav negatīvi un vai uzkrautais daudzums nepārsniedz 40cm3, atgriež false ja
        nav
        {
            if (!txtBoxAmount.Text.All(char.IsNumber) ||
            !txtBoxAmount.Text.All(char.IsNumber)) // Pārbauda, vai lietotāja ievade ir veseli
            skaitļi
            {
                MessageBox.Show("Vērtību lauki drīkst būt tikai veseli
                skaitļi");
            }
            else if (txtBoxAmount.Text == "" || txtBoxEstimatedTime.Text ==
            "") // Pārbauda, vai lauki nav atstāti tukši
            {
                MessageBox.Show("Vērtību lauki nedrīkst būt tukši");
            }
            else
            {
                dbctxt.SaveChanges(); // Ja lietotāja ievadītie dati ir
                korekti, dati tiek saglabāti datubāzē un...
                ControlsLoaderBatch2(); // ... tiek atvērti nākošie lauki
                uzdevuma izpildei - paredzamais laiks ceļā un patērētais laiks ceļā
            }
        }
    }

    private void btnFinish_Click(object sender, RoutedEventArgs e) // Pēdējā poga
    logā. Uzdevuma statuss mainās uz izpildīts, potenciālais un patērētais laiks tiek
    pievienoti uzdevumam
    {
        int.TryParse(txtBoxSpentTime.Text, out int spentTime);
        if (checkIsValid2(spentTime)) // Pārbauda, vai patērētais laiks nav
        negatīvs
        {
            if (!txtBoxAmount.Text.All(char.IsNumber) ||
            !txtBoxAmount.Text.All(char.IsNumber)) // Pārbauda, vai lietotāja ievade ir veseli
            skaitļi
            {
                MessageBox.Show("Vērtību lauki drīkst būt tikai veseli skaitļi");
            }
            else if (txtBoxAmount.Text == "" || txtBoxEstimatedTime.Text == "") //
            Pārbauda, vai lauki nav atstāti tukši

```



```

        btnFinish.Visibility = Visibility.Visible;
    }
    private bool checkIsValid(int amount, int estimatedTime) // Pārbauda vai
ievadītais daudzums un paredzētais laiks nav negatīvi un vai iekrautais daudzums
nepārsniedz 40cm3
    {
        if (amount < 0 || estimatedTime < 0)
        {
            MessageBox.Show("Ievadītais skaitlis nevar būt negatīvs");
            return false;
        }
        if (amount > 40)
        {
            MessageBox.Show("Ievadītais daudzums pārsniedz limitu");
            return false;
        }
        else return true;
    }
    private bool checkIsValid2(int actualTime) // Pārbauda, vai patērētais laiks
nav negatīvs
    {
        if (actualTime < 0)
        {
            MessageBox.Show("Ievadītais skaitlis nevar būt negatīvs");
            return false;
        }
        else return true;
    }
}
}
}

```

Kvalifikācijas darbs „*Atskaišu sistēma kokvedēja vadītājiem*” 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: ***Reinis Eduards Indriksons*** _____ **10.01.2022.**

Rekomendēju darbu aizstāvēšanai

Darba vadītājs: ***Doc., Dr.dat. Jevgēnijs Vihrovs*** _____ **10.01.2022.**

Recenzents: ***Mg.Dat. Kalvis Upītis***

Darbs iesniegts 10.01.2022.

Kvalifikācijas darbu pārbaudījumu komisijas sekretārs: Viesturs Vēzis

Darbs aizstāvēts kvalifikācijas darbu pārbaudījuma komisijas sēdē

____.01.2022. prot. Nr. _____

Komisijas sekretārs(-e): _____