**RĪGAS VALSTS TEHNIKUMS**

DATORIKAS NODAĻA

Izglītības programma: Programmēšana

**PRASĪBU DOKUMENTS**

**Tīmekļa lietojumprogramma autoservisa pārvaldībai un klientu apkalpošanai**

Audzēknis: Jānis Krišjānis Gailis

Prakses vadītājs: Oksana Roslova

Nodaļas vadītājs: Normunds Barbāns

**Rīga 2024**

# Anotācija

**Saturs**

[ievads 1](#_Toc190161143)

[1. UZDEVUMA NOSTĀDNE 2](#_Toc190161144)

[2. PRASĪBU SPECIFIKĀCIJA 4](#_Toc190161145)

[2.1. Ieejas un izejas informācijas apraksts 4](#_Toc190161146)

[2**.**1.1. Ieejas informācijas apraksts 4](#_Toc190161147)

[2.1.2. Izejas informācijas apraksts 4](#_Toc190161148)

[2.2. Funkcionālās prasības 5](#_Toc190161149)

[2.3. Nefunkcionālās prasības 5](#_Toc190161150)

[3. UZDEVUMA RISINĀŠANAS LĪDZEKĻU IZVĒLES PAMATOJUMS 5](#_Toc190161151)

[4. PROGRAMMATŪRAS PRODUKTA MODELĒŠANA UN PROJEKTĒŠANA 6](#_Toc190161152)

[4.1. Sistēmas struktūras modelis 6](#_Toc190161153)

[4.1.1. Sistēmas arhitektūra 6](#_Toc190161154)

[4.1.2. Sistēmas ER modelis 6](#_Toc190161155)

[4.2. Funkcionālais sistēmas modelis 8](#_Toc190161156)

[4.2.1. Datu plūsmu modelis 8](#_Toc190161157)

[5. DATU STRUKTŪRU APRAKSTS 8](#_Toc190161158)

[6. LIETOTĀJA CEĻVEDIS 8](#_Toc190161159)

[6.1. Sistēmas prasības aparatūrai un programmatūrai 8](#_Toc190161160)

[6.2. Sistēmas instalācija un palaišana 8](#_Toc190161161)

[6.3. Programmas apraksts 8](#_Toc190161162)

[7. PROGRAMMATŪRAS LIETOJAMĪBAS TESTĒŠANA 8](#_Toc190161163)

[NOBEIGUMS 8](#_Toc190161164)

[INFORMĀCIJAS AVOTI 9](#_Toc190161165)

[PIELIKUMI 10](#_Toc190161166)

# ievads

Mūsdienās gandrīz katram ir sava automašīna un nepieciešams remonts, bet Latvijā ir maz pieejamas mājaslapas autoservisiem, kuras tiek aktīvi atjauninātas. Eksistējošajām mājas lapām ir slikts dizains un maz funkciju. Ideja ir veidot sistēmu, kura piedāvā ikdienas cilvēkiem ērtu, godīgu un ātru mašīnremontu.

Galvenie konkurenti ir sistēmas, kuras ļauj klientiem apskatīt autoservisu informāciju un sniedz klientiem kontaktinformāciju. Stiprās puses šīm sistēmām ir vienkāršs dizains, maz nepieciešamu funkciju, nav vajadzīga klientu profilu izveide, kā arī tās neizmanto daudz resursus, jo tikai uzrāda informāciju. Vājās puses konkurentiem, kuras šis projekts atrisinās, ir tas, ka liela daļa no konkurentiem izmanto novecojušus dizainus vai tādus, kurus klientiem grūti izprast.

Projekts piedāvās arī papildus funkcijas, ar klientu reģistrācijas palīdzību spēsim piedāvāt remontu kalendāru, kas padarīs zvanīšanu nevajadzīgu, ietaupot laiku visiem. Citas funkcijas, kuras šo projektu atšķirs no konkurentiem ir cenrādis, kurš parāda aptuvenas cenas klientam uzreiz, jo klienti labi vērtē atklātību.

Šīs sistēmas aktualitāte tikai aug, jo arvien vairāk cilvēki izmanto mašīnas transportam un tām bieži nepieciešams remonts. Tieši šāda sistēma ir nepieciešama, jo ar tās palīdzību iespējams ietaupīt laiku gan klientiem, gan mehāniķiem.

# 1. UZDEVUMA NOSTĀDNE

Noslēguma darba uzdevums ir autoservisa datu bāzes sistēmas izveide, kurā ietilpst lietotāju reģistrācijas sistēma, lai piešķirtu unikālas funkcijas tieši reģistrētiem lietotājiem un laika tabulas sistēmas izveide, kurā klienti var izvēlēties kalendārā redzamo brīvo laiku, lai pieteiktu savas mašīnas remontu.

Sistēmu nepieciešams veidot, lai ietaupītu laiku, gan darbiniekiem, gan klientiem un veidotu atklātību ar klientiem, arī palielinātu klientu skaitu autoservisam.

Ir plānotas vairākas funkcijas (skat. 1.1.att.):

A diagram of a diagram

AI-generated content may be incorrect.

1.1.att “Lietojumgadījuma diagramma”

* Viesiem būs iespējams:
  + Piekļūt informācijai par autoservisu, tā atrašanās vietu, piedāvātajiem servisiem, darba laiku, jaunumiem u.c.
  + Izveidot kontu;
* Klientiem būs iespējams:
  + Piekļūt informācijai par autoservisu, tā atrašanās vietu, piedāvātajiem servisiem, darba laiku, jaunumiem u.c.
  + Apskatīt cenrādi un aprēķināt aptuveno remontdarba cenu;
  + Rediģēt profilu, kā arī iestatīt paziņojumu iestatījumus;
  + Pieteikt remontu un saņemt atgādnes paziņojumus par to;
  + Saņemt paziņojumus par dažādiem piedāvājumiem;
  + Apskatīt remonta statusu, vai to atcelt;
* Darbiniekiem būs iespējams:
  + Apskatīt klientu pieteiktos remontus;
  + Rediģēt remontu statusus un pabeigšanas laikus, kas automātiski arī informē klientu par izmaiņām;
  + Sūtīt klientiem paziņojumus par piedāvājumiem;
* Administratoram būs iespējams:
  + Apskatīt klientu pieteiktos remontus;
  + Rediģēt remontu statusus un pabeigšanas laikus, kas automātiski arī informē klientu par izmaiņām;
  + Sūtīt klientiem paziņojumus par piedāvājumiem;
  + Pievienot jaunus darbiniekus sistēmai;
  + Rediģēt informāciju par darbiniekiem;
  + Dzēst kontus un remontus.

# 2. PRASĪBU SPECIFIKĀCIJA

## 2.1. Ieejas un izejas informācijas apraksts

### 2.1.1. Ieejas informācijas apraksts

Sistēmā tiks nodrošināta šādas ieejas informācijas apstrāde.

Informāciju ievadīs lietotājs.

1. Informācija par **lietotājiem** sastāvēs no šādiem datiem.

* Vārds – klienta – burtu teksts ar izmēru līdz 15 rakstzīmēm (piem., “Jānis”).
* Lietotājvārds – klienta lietotājvārds – burtu teksts ar izmēru līdz 20 rakstzīmēm (piem., “Rainis”).
* Ēpasts – klienta ēpasta adrese – burtu teksts, ar izmēru līdz 45 rakstzīmēm (piem., “JanisRainis@gmail.com”).
* Parole – klienta profila parole – burtu teksts līdz 45 rakstzīmēm (piem., “Parole123!”).

2. Informācija par **projektiem** sastāvēs no šādiem datiem.

* Sākuma datums – datums un laiks, kurā sāksies projekts – teksts ar izmēru līdz 45 rakstzīmēm (piem., “2024-11-01T07:00:00.000Z”).
* Beigu datums – datums un laiks, kurā beigsies projekts – teksts ar izmēru līdz 45 rakstzīmēm (piem., “2024-11-01T07:00:00.000Z”).
* Informācija – papildus informācija par projektu – burtu teksts ar izmēru līdz 255 rakstzīmēm (piem.,”Vispārējā Ford Focus pārbaude un regulēšana ).

3. Informācija par **Darbinieiem** sastāvēs no šādiem datiem.

* Amats – darbinieka amats – teksts ar izmēru līdz 45 rakstzīmēm (piem., “Galvenais mehāniķis”).
* Darba sākšanas datums – datums, kurā darbinieks sāka strādāt – datums pēc eiropas standarta (piem., “2024-01-01”).

### 2.1.2. Izejas informācijas apraksts

1. **Projektu statistikas izvade administratoram.** Administratoram bus pieeja statistikas lapai, kur būs divas shēmas, kurās uzrādīts 1. Projektu skaitu attiecība pret to, cik klientiem ir šāds remontu skaits, 2. Projektu skaits katrā kategorijā.
2. **E-pasta atgādinājums** lietotājiem. Lietotāji pirms projekta saņems atgādinājumu par projektu, kurā būs sniegta informācija par projektu un tā izvadīta ar paziņojumu, cik ilgi līdz projekta sākuma laika.
3. **Kalendārs.** Lietotājam piesakot jaunu projektu, jāizvēlas datums un laiks, šo datumu izvēlas atzīmējot brīvos laikus no kalendāra, tātad tiek izvadīts kalendārs, kurš apkopo aizņemtos laikus un neļauj lietotājam tos atzīmēt, kā sava projekta laiku.

## 2.2. Funkcionālās prasības

1. Lietotāja reģistrācija

1.1. Jānodrošina iespēja lietotājam ievadīt e-pastu, vārdu, lietotājvārdu un paroli.

1.2. Sistēmai ir jāpārbauda, vai ievadītais e-pasts jau eksistē sistēmā. Ja tā, jāizvada kļūdas paziņojums “Email already registered”.

1.3. Sistēmai ir jāpārbauda, vai ievadītais lietotājvārds jau eksistē sistēmā. Ja tā, jāizvada kļūdas paziņojums “Username is taken”.

1.4. Sistēmai ir jāpārbauda paroles drošības prasības (vismaz 5 rakstzīmes, 1 lielais burts, 1 speciālais simbols).

1.5. Lietotājam jāpārbauda paroles atkārtojums, lai novērstu kļūdas.

1.6. Pēc veiksmīgas reģistrācijas sistēmai jāinformē lietotājs par sekmīgu reģistrāciju.

2. Lietotāja autorizācija

2.1. Lietotājam jābūt iespējai autorizēties ar e-pastu vai lietotājvārdu un paroli.

2.2. Sistēmai jāpārbauda, vai ievadītie dati sakrīt ar esošo lietotāju informāciju.

2.3. Ja e-pasts vai lietotājvārds nesakrīt, jāizvada kļūdas paziņojums “Credentials don’t match any accounts”.

2.4. Ja parole nesakrīt, jāizvada kļūdas paziņojums “Password is incorrect”.

2.5. Pēc veiksmīgas autorizācijas lietotājam jāsaņem paziņojums par veiksmīgu pieslēgšanos.

3. Remonta pieteikšana

3.1. Lietotājam jābūt iespējai izvēlēties pakalpojuma veidu (piem., ātrs remonts, eļļas maiņa).

3.2. Sistēmai jāpielāgo pieejamie laiki atkarībā no pakalpojuma veida (piem., vidēja ilguma pakalpojumiem jānoņem laiki, kur brīvā vieta ir mazāka par 2 dienām).

3.3. Lietotājam jābūt iespējai izvēlēties laiku kalendārā un sniegt remonta aprakstu.

3.4. Sistēmai jāpārbauda, vai izvēlētais laiks ir brīvs. Ja nav, jāizvada kļūdas paziņojums “Invalid choice”.

3.5. Sistēmai jāpaziņo gan klientam, gan darbiniekam par veikto pasūtījumu.

4. Remonta datuma vai statusa maiņa (Administrators)

4.1. Administratoram jābūt iespējai redzēt visus projektus pārvaldības panelī.

4.2. Administratoram jābūt iespējai atvērt remonta informāciju un veikt izmaiņas.

4.3. Mainot remonta beigu datumu, sistēmai jāpārbauda, vai norādītais laiks ir brīvs. Ja nav, jāizvada kļūdas paziņojums “Conflicting repair times”.

4.4. Ja laiks ir pieejams, sistēmai jāatjauno informācija par remontu un jāinformē klients.

5. Remonta cenas aprēķināšana

5.1. Lietotājam jābūt iespējai atvērt sadaļu ar pakalpojumu cenām.

5.2. Lietotājam jābūt iespējai izvēlēties nepieciešamos pakalpojumus (piem., izvēlēties mašīnas daļas, kuras nepieciešams krāsot).

5.3. Sistēmai jāaprēķina kopējā cena, pamatojoties uz izvēlētajiem pakalpojumiem.

6. Remonta statusa maiņa uz “Neieradās” (Darbinieks)

6.1. Darbiniekam jābūt iespējai atvērt remonta projekta sadaļu.

6.2. Darbiniekam jābūt iespējai izvēlēties remontu un mainīt tā statusu uz “Neieradās”.

6.3. Sistēmai jāatjauno remonta statuss un jāizvada jaunā informācija.

7. Vispārīgās funkcijas

7.1. Viesim jābūt iespējai apskatīt pakalpojumus un to cenas.

7.2. Reģistrētam lietotājam jābūt iespējai apskatīt aktīvos remontus un saņemt paziņojumus par atlaidēm un remontiem.

7.3. Administratoram jābūt iespējai pievienot jaunus pakalpojumus un dzēst esošos.

7.4. Sistēmai jābūt viegli lietojamai un pieejamai visiem lietotājiem.

## 2.3. Nefunkcionālās prasības

1. Valoda un lietotāja saskarne
   1. Sistēmas saskarnes valodai ir jābūt angļu valodai, lai nodrošinātu lietotāju ērtības.
   2. Lietotāja saskarnei jābūt intuitīvai, viegli lietojamai, un tai jāatbilst mūsdienu tīmekļa lietojumprogrammu standartiem.
2. Responsivitāte un pielāgošanās ekrāniem
   1. Jānodrošina tīmekļa lietojumprogrammas responsivitāte, lai tā būtu pielāgota dažādiem ekrāna izmēriem, tostarp mobilajām ierīcēm, planšetdatoriem un darbvirsmas monitoriem.
   2. Sistēmai jāspēj nodrošināt pilnīgu funkcionalitāti neatkarīgi no izmantotās ierīces vai tās izšķirtspējas.
3. Dizains un vizuālais noformējums
   1. Dizainam jābūt organizācijas logotipa krāsās, lai saglabātu vizuālo identitāti.
   2. Tekstam jābūt Arial fontā tumšā krāsā, kas nodrošina lasāmību uz gaiša fona.
4. Drošība un datu aizsardzība
   1. Sistēmai jānodrošina lietotāju datu drošība, ievērojot Vispārīgās datu aizsardzības regulas (GDPR) prasības.
   2. Parolēm jābūt šifrētām, izmantojot modernus kriptogrāfijas algoritmus.
   3. Jānodrošina aizsardzība pret nesankcionētu piekļuvi, tostarp droša autentifikācija un autorizācija.
5. Ātrdarbība un veiktspēja
   1. Sistēmai jābūt ātrai un efektīvai, ar reakcijas laiku ne vairāk kā 2 sekundes 90% no pieprasījumiem.
   2. Jānodrošina stabila veiktspēja arī tad, ja sistēmu izmanto vairāk nekā 1000 aktīvo lietotāju vienlaikus.
6. Pieejamība
   1. Sistēmai jāatbilst pieejamības standartiem (piem., WCAG 2.1), lai to varētu izmantot arī cilvēki ar invaliditāti.
   2. Jānodrošina iespēja mainīt fonta izmēru un kontrasta režīmu lietotāja ērtībai.
7. Tehnoloģiskās prasības
   1. Sistēmai jādarbojas uz visām mūsdienu tīmekļa pārlūkprogrammām, piemēram, Google Chrome, Mozilla Firefox, Microsoft Edge un Safari.
   2. Sistēma jāizstrādā, izmantojot stabilas un pārbaudītas tehnoloģijas, lai nodrošinātu ilgtspēju un vieglu uzturēšanu.
8. Lietotāja interfeisa prasības
   1. Pieslēgšanās ekrānloga skice jāizskatās sekojoši(skat. 2.1.att.)

A screenshot of a computer screen

AI-generated content may be incorrect.

2.1.att. “Pieslēgšanās formas skice”

Šī skice attēlo lietotāja pieslēgšanās interfeisu, kurā tiek prasīts ievadīt lietotājvārdu un paroli. Tā nodrošina autentifikāciju, lai piekļūtu sistēmai un izmantotu tās funkcionalitāti.

* 1. Reģistrācijas ekrānloga skice jāizskatās sekojoši (skat. 2.2.att.)

A screenshot of a computer screen

AI-generated content may be incorrect.

2.2.att. “Reģistrēšanās formas skice”

Šī skice attēlo jauna lietotāja reģistrācijas interfeisu, kurā nepieciešams ievadīt personīgos datus, šajā gadijumā, vārdu, uzvārdu , e-pasta adresi un paroli. Šis process ļauj lietotājiem izveidot kontu un iegūt piekļuvi sistēmai.

* 1. Paziņojumu iestatījumu ekrānloga skice jāizskatās sekojoši (skat. 2.3.att.)

A screenshot of a computer screen

AI-generated content may be incorrect.

2.3.att. “Paziņojumu iestatījšanas formas skice”

Šī skice attēlo paziņojumu iestatījumu interfeisu, kurā lietotāji var pielāgot paziņojumu saņemšanas veidu un biežumu. Tas ļauj personalizēt sistēmas paziņojumus un informāciju atbilstoši lietotāja vēlmēm.

# 3. UZDEVUMA RISINĀŠANAS LĪDZEKĻU IZVĒLES PAMATOJUMS

Sistēma ir paredzēta pārlūkprogrammai, tādēļ to realizācijai tiek izmantoti jaunākie programmatūras risinājumi.

**Datu bāzes izstrādei** un administrācijas panelim tika izmantota **MySQL Workbench** 8.0.36 CE, 2024. gada versija, jo MySQL Workbench ir plaši izmantots vizuālās datu bāzes dizaina rīks un administrēšanas rīks. Tas piedāvā lietotājam draudzīgu saskarni **MySQL** datu bāzu projektēšanai, pārvaldībai un mijiedarbībai ar tām.

Datu bāzes savienošanai ar tīmekļa vietni tiek izmantots **Node.js** (v20.11.0), galvenokārt izmantots, jo dod iespēju izmantot JavaScript gan klienta, gan servera pusē, kas atvieglo koda koplietošanu un paātrina izstrādi.

**Serverim** tika izmantots **Express.js** (v.4.18.2), jo tas ir tīmekļa lietojumprogrammu ietvars Node.js, kas nodrošina spēcīgu funkciju kopumu tīmekļa izveidei un tas ir vienkāršs un elastīgs. Express starpprogrammatūras sistēma ļauj viegli integrēt trešo pušu moduļus, ļaujot lietojumprogrammai pievienot dažādas funkcijas un tā nodrošina vienkāršu un efektīvu maršrutēšanas sistēmu HTTP pieprasījumu apstrādei.

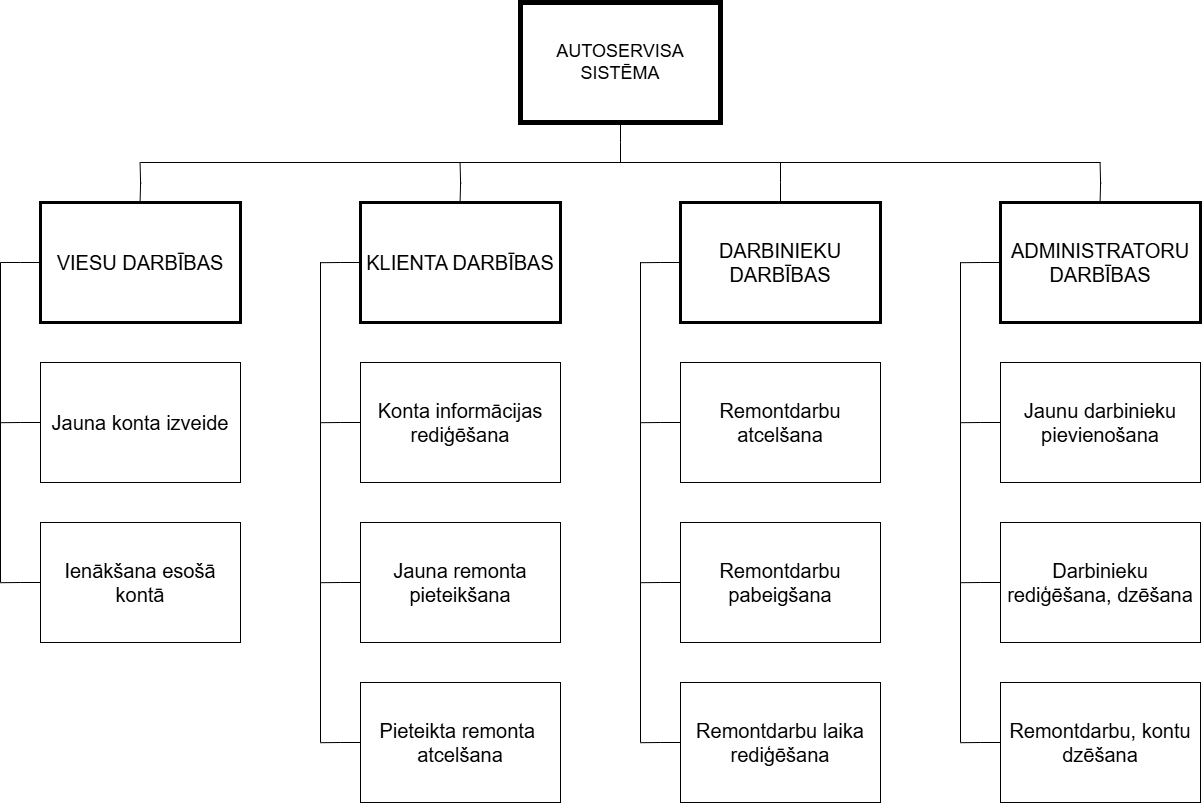
Mājaslapas izstrādei tika izmantotas **JavaScript** un **CSS** valodas, darbojoties **VSCode** un **VS** vidēs. Tika izvēlētas šīs valodas un vides, jo tās labi darbojas kopā ar Express.js un nodrošina plašu saderību ar visām mūsdienu pārlūkprogrammām. Turklāt JavaScript un CSS piedāvā lielu elastību un plašu bibliotēku atbalstu, kas ļauj ātri attīstīt un pielāgot mājaslapu lietotāju vajadzībām

# 4. PROGRAMMATŪRAS PRODUKTA MODELĒŠANA UN PROJEKTĒŠANA

## 4.1. Sistēmas struktūras modelis

### 4.1.1. Sistēmas arhitektūra

Sistēma tiks veidota no četrām apakšsisēmām (skat. 4.1. att.): Viesu darbības apakšsistēmas, klientu darbības apakšsistēmas, darbinieku darbības apakšsistēmas un administratoru darbības apakšsistēmas.



4.1. att “Functionālās dekompozīcijas diagramma”

Pēc diagrammas (skat 4.1.att) varam noteikt, ja viesi var izveidot jaunu kontu un ienākt esošā kontā. Klienti var rediģēt konta informāciju, pieteikt vai atcelt remontu. Darbinieki var atcelt, pabeigt un rediģēt remontdarbu izpildes laiku. Administratori var pievienot, rediģēt un dzēst darbiniekus, kā arī dzēst remontdarbus un kontus.

### 4.1.2. Sistēmas ER modelis

Sistēmas ER-modelis sastāv no piecām entītijām, kas nodrošina pamata informācijas uzglabāšanu un apstrādi (skat. 4.2. att).

Datu bāzes entītijas ir savstarpēji saistītas ar dažādām binārām attiecībām, kas nosaka to savstarpējo kardinalitāti:

* Projekta statusu un projektus saista attiecība "viens pret daudziem", jo vienam projekta statusam var būt vairāki projekti, bet katram projektam var būt tikai viens statuss.
* Projektus un lietotājus saista attiecība "viens pret daudziem", jo viens projekts var piederēt tikai vienam lietotājam, bet viens lietotājs var veikt vairākus pasūtījumus.
* Lietotājus un Paziņojumu iestatījumus saista attiecības "viens pret vienu", jo vienam lietotājam sistēmā var būt tikai viena paziņojumu iestatījumu kopa.
* Lietotājus un darbiniekus saista attiecība "viens pret vienu", jo viens lietotājs sistēmā var būt tikai viens darbinieks un otrādi.
* Administratorus un darbiniekus saista attiecība "viens pret vienu", jo katrs administrators sistēmā tiek uzskatīts par vienu darbinieku, un darbinieks var būt tikai viens administrators.

A diagram of a company

AI-generated content may be incorrect.

4.2. att. “ER-diagramma”

## 4.2. Funkcionālais sistēmas modelis

### 4.2.1. Datu plūsmu modelis

* **Jauna konta izveide** (skat 4.3. att.)

Lietotājs ievada vārdu, ēpastu, Lietotājvārdu, paroli un atkārtotu paroli. Tad tiek pārbaudīts vai ievadītais lietotājvārds ir derīgs. Pēc tam tiek pārbaudīts vai Ēpasts ievadīts pareizā formātā, tad tiek pārbaudīts vai parole un atkārtotā parole sakrīt un vai tā ir droša. Tad tiek pārbaudīts vai Ēpasta adrese un Lietotājvārds ir unikāli un nesakrīt ar jau esošajiem kontiem sistēmā. Ja kāds no procesiem bija neveiksmīgs, tiek izvadīts kļūdas ziņojums, citādi tiek izveidots jaunais konts.

A diagram of a diagram

AI-generated content may be incorrect.

4.3. att. “Konta izveides moduļa datu plūsmas diagramma”

* **Ienākšana esošā kontā** (skat 4.4. att.)

Lietotājs ievada ēpastu vai lietotājvārdu un paroli. Tad attiecīgi tiek pārbaudīts vai ēpasts vai lietotājvārds ir pareizi formatēts, pēc tam pārbauda vai tas sakrīt ar kādu ēpastu sistēmā. Tad tiek pārbaudīts vai parole sakrīt ar sistēmā esošo paroli šim kontam. Ja kāds no procesiem bija neveiksmīgs, tiek izvadīts kļūdas ziņojums, citādi tiek izvadīts ziņojums par veiksmīgu ienākšanu kontā.

A diagram of a company

AI-generated content may be incorrect.

4.4. att. “Ienākšanas esošā kontā datu plūsmas diagramma”

* **Konta informācijas maiņa** (skat 4.5. att.)

Lietotājs ievada jauno lietotājvārdu. Tad, tiek pārbaudīts vai ievadītais lietotājvārds ir derīgs. Pēc tam pārbauda vai šis lietotājvārds sakrīt ar jau eksistējošu lietotājvārdu. Ja kāds no procesiem bija neveiksmīgs, tiek izvadīts kļūdas ziņojums, citādi tiek pievienots jaunais lietotājvārds lietotājam.

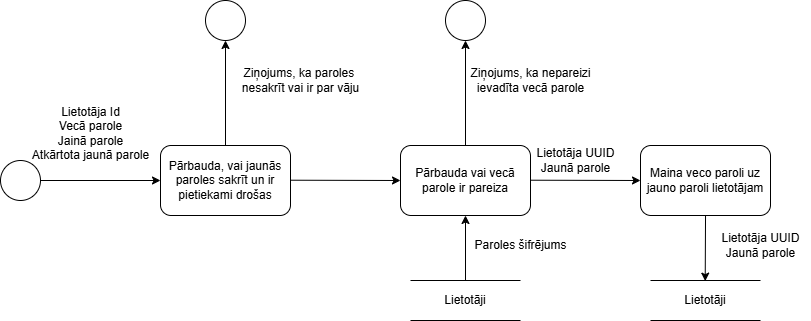
**A diagram of a company

AI-generated content may be incorrect.**

4.5. att. “Konta informācijas maiņas datu plūsmas diagramma”

* **Konta paroles maiņa** (skat 4.6. att.)

Lietotājs ievada veco paroli, jauno paroli un atkārtotu jauno paroli. Tad, tiek pārbaudīts vai ievadītā jaunā parole ir droša un sakrīt ar atkārtoto jauno paroli. Pēc tam pārbauda vai ievadītā vecā parole ir pareizi. Ja kāds no procesiem bija neveiksmīgs, tiek izvadīts kļūdas ziņojums, citādi tiek pievienota jaunā parole lietotājam.



4.6. att. “Konta paroles maiņas datu plūsmas diagramma”

* **Jauna remonta pieteikšana** (skat 4.7. att.)

Lietotājs ievada projekta sākuma datumu un laiku, beigu datumu un laiku, informāciju par automašīnu un informāciju par remontu. Tad, tiek pārbaudīts vai ievadītā informācija par remontu un automašīnu ir derīga. Pēc tam pārbauda vai izvēlētais laika posms ir derīgs. Ja kāds no procesiem bija neveiksmīgs, tiek izvadīts kļūdas ziņojums, citādi tiek izveidots jaunais projekts.

A diagram with text and words

AI-generated content may be incorrect.

4.7. att. “Jauna remonta pieteikšanas datu plūsmas diagramma”

* **Pieteiktā remonta atcelšana** (skat 4.8. att.)

Lietotājs izvēles projektu, kuru atcelt un ievada atcelšanas iemeslu. Tad, tiek pārbaudīts vai ievadītais iemesls ir derīgs. Pēc tam pārbauda vai izvēlētais projekts patiešām ir šī lietotāja projekts un tad pārbauda vai projekts nav sākts, vai pabeigts. Ja kāds no procesiem bija neveiksmīgs, tiek izvadīts kļūdas ziņojums, citādi tas tiek atzīmēts kā atcelts.

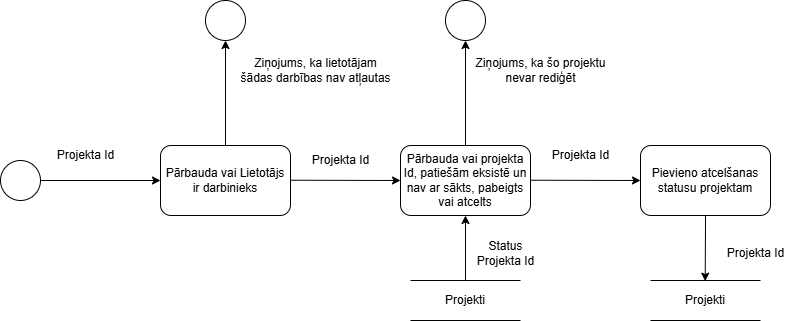
A diagram of a diagram

AI-generated content may be incorrect.

4.8. att. “Pieteiktā remonta atcelšanas datu plūsmas diagramma”

* **Remontdarbu atcelšana** (skat 4.9. att.)

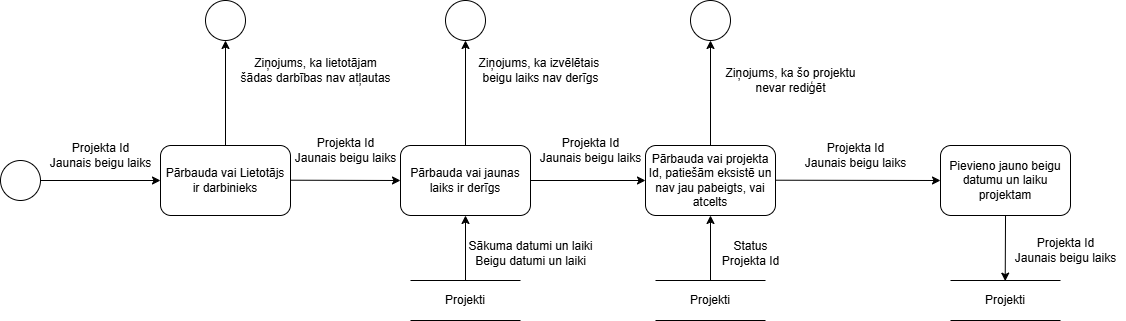
Darbinieks izvēles projektu. Tad, tiek pārbaudīts vai lietotājs patiešam ir darbinieks. Pēc tam pārbauda vai izvēlētais projekts eksistē un nav sākts, atcelts vai pabeigts. Ja kāds no procesiem bija neveiksmīgs, tiek izvadīts kļūdas ziņojums, citādi tas tiek atzīmēts kā atcelts.



4.9. att. “Remontdarbu atcelšanas datu plūsmas diagramma”

* **Remontdarbu laika rediģēšana** (skat 4.10. att.)

Darbinieks izvēles projektu un ievada jaunu beigu laiku. Tad, tiek pārbaudīts vai lietotājs patiešam ir darbinieks. Pēc tam pārbauda vai izvēlētais laiks ir derīgs un vai izvēlētais projekts eksistē un nav sākts, atcelts vai pabeigts. Ja kāds no procesiem bija neveiksmīgs, tiek izvadīts kļūdas ziņojums, citādi pievieno projektam jauno beigu laiku.

****

4.10. att. “Remontdarbu laika rediģēšanas datu plūsmas diagramma”

* **Jauna darbinieka pievienošana** (skat 4.11. att.)

Administrators ievada darbinieka ēpastu un lomu. Tad, tiek pārbaudīts vai lietotājs patiešam ir administrators. Pēc tam pārbauda vai ēpasts sistēmā eksistē un vai lietotājs ar šo ēpastu jau nav darbinieks . Ja kāds no procesiem bija neveiksmīgs, tiek izvadīts kļūdas ziņojums, citādi pievieno lietotāju pie darbinieku tabulas.

A diagram of a diagram

AI-generated content may be incorrect.

4.11. att. “Jauna darbinieka pievienošanas datu plūsmas diagramma”

* **Projektu un kontu dzēšana** (skat 4.12. att.)

Administrators ievada Projekta Id vai lietotāja Ēpastu. Tad, tiek pārbaudīts vai lietotājs patiešam ir administrators. Pēc tam pārbauda vai ievadītais identifikators eksistē sistēmā . Ja kāds no procesiem bija neveiksmīgs, tiek izvadīts kļūdas ziņojums, citādi attiecīgā informācija tiek dzēsta no sistēmas.

A diagram of a project

AI-generated content may be incorrect.

4.12. att. “Projektu un kontu dzēšanas datu plūsmas diagrammas”

# 5. DATU STRUKTŪRU APRAKSTS

Datubāzes izveidošanai ir nepieciešama datubāzes fiziskā struktūra. Tā tika iegūta normalizējot ER diagrammu, kuras rezultātā no entitījām izveidoja relācijus (skat 5.1. att). Kā redzams datubāze sastāv no 6 tabulām, kas satur informāciju par projektu statusu, projektiem, lietotājiem, lietotāju iestatījumiem, darbiniekiem un administratoriem.

* Tabula " **ProjectStatuses**" glabās informāciju par projektu statusiem.
* Tabula "**Projects**" glabās informāciju par projektiem.
* Tabula "**Users**" glabās informāciju par lietotājiem.
* Tabula “**NotificationSettings**” glabās informāciju par lietotāju paziņojumu iestatījumiem
* Tabula "**Workers**" glabās informāciju par darbiniekiem.
* Tabula "**Administrators**" glabās informāciju par administratoriem.

Tabula “ProjectStatuses” satur informāciju par projektu statusies: identifikatoru un nosaukumu.

1.tabula

Tabulas “**ProjectStatuses**” struktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr** | **Lauka nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| 1. | ProjectStatusId | INT | - | Ieraksta idenitifikators |
| 2. | Name | VARCHAR | 15 | Statusa nosaukums |

Tabula “Projects” satur informāciju par remontiem: idenitifikatoru, ārējo atslēgu uz “ProjectStatuses”, ārējo atslēgu uz “Users”, remonta sākuma datumu un laiku, remonta beigu datumu un laiku un aprakstu par remontu.

2.tabula

Tabulas “**Projects**” struktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr** | **Lauka nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| 1. | ProjectId | INT | - | Ieraksta idenitifikators |
| 2. | UserId | INT | - | Ārējā atslēga uz “Users” tabulu |
| 3. | ProjectStatusId | INT |  | Ārējā atslēga uz “ProjectStatuses” tabulu |
| 4. | StartDateTime | DATE |  | Datums un laiks, kad projektam jāsākas |
| 5. | EndDateTime | DATE | - | Datums un laiks, kad projektam jābūt pabeigtam |
| 6 | Description | VARCHAR | 300 | Informācija par remontu, t.i. informācija par automašīnu un remonta tipu |

Tabula “Users” satur informāciju par klientiem: identifikatoru, vārdu, Lietotājvārdu, ēpastu un šifrētu paroli.

3.tabula

Tabulas “**Users**” struktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr** | **Lauka nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| 1. | UserId | INT | - | Ieraksta idenitifikators |
| 2. | Name | VARCHAR | 40 | Klienta vārds |
| 3. | UserName | VARCHAR | 20 | Lietotājvārds ar kuru klients var ienākt kontā |
| 4. | Email | VARCHAR | 100 | Lietotāja ēpasta adrese |
| 5. | PasswordHash | VARCHAR | 300 | Šifrēta klienta parole |

Tabula “NotificationSettings” satur informāciju par klientu iestatījumiem: identifikatoru, ārējo atslēgu uz “Users”, atgādinājumu iestatījumi un piedāvājumu iestatījumiem.

4.tabula

Tabulas “**NotificationSettings**” struktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr** | **Lauka nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| 1. | NotificationId | INT | - | Ieraksta idenitifikators |
| 2. | UserId | INT | - | Ārējā atslēga uz klientu uz kura attiecas šie iestatījumi |
| 3. | AppointmentReminders | VARCHAR | 10 | Atgādinājumu iestatījumi |
| 4. | DealNotifications | VARCHAR | 10 | Piedāvājumu iestatījumi |

Tabula “Workers” staur informāciju par lietotājiem, kuri ir arī darbinieki: identifikatoru, ārējo atslēgu uz “Users”, darbinieka stāžu, amata nosaukumu, un darba sākšanas datumu.

5.tabula

Tabulas “**Workers**” struktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr** | **Lauka nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| 1. | WorkerId | INT | - | Ieraksta idenitifikators |
| 2. | UserId | INT | - | Ārējā atslēga uz darbinieka kontu |
| 3. | Tenure | INT | - | Gadu stāža šajā amatā |
| 4. | StartWorkDate | DATE | - | Datums, kurā darbinieks sāka strādāt |
| 5. | Position | VARCHAR | 50 | Darba amata nosaukums |

Tabula “Administrators” satur informāciju par lietotājiem, kuri ir arī administratori: identifikators, ārējo atslēgu uz “Users”, administratora stāžu un administratora darba sākšanas datumu.

6.tabula

Tabulas “**Administrators**” struktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr** | **Lauka nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| 1. | AdminId | INT | - | Ieraksta idenitifikators |
| 2. | UserId | INT | - | Ārējā atslēga uz darbinieka kontu |
| 3. | AdminTenure | INT | - | Gadu stāža šajā amatā |
| 4. | AdminStartWorkDate | DATE | - | Datums, kurā darbinieks sāka strādāt |

A diagram of a workflow

AI-generated content may be incorrect.

5.1. att. “Datu bāzes tabulu saišu shēma”

# 6. LIETOTĀJA CEĻVEDIS

## 6.1. Sistēmas prasības aparatūrai un programmatūrai

## 6.2. Sistēmas instalācija un palaišana

## 6.3. Programmas apraksts

# 7. PROGRAMMATŪRAS LIETOJAMĪBAS TESTĒŠANA

# NOBEIGUMS

# INFORMĀCIJAS AVOTI

1. **MySQL pamācība. Ģenerētās kolonnas MySQL.** – <https://www.mysqltutorial.org/mysql-basics/mysql-generated-columns/> – (  
   **Resurss apskatīts 15.11.2023. )**
2. **MySQL dokumentācija. Windows komandrindas sākšana. –** <https://dev.mysql.com/doc/refman/8.0/en/windows-start-command-line.html> –  
   **(Resurss apskatīts 02.12.2023.)**
3. **Stack Overflow. Express Cookie atgriež undefined. –** <https://stackoverflow.com/questions/12197053/express-cookie-return-undefined> –  
   **(Resurss apskatīts 20.01.2024).**  
   (Un citas Stack Overflow sadaļas.)
4. **Traversy Media. Iemācieties MERN steku – pilna pamācība. –** <https://www.youtube.com/watch?v=7S_tz1z_5bA> –  
   **(Resurss apskatīts 10.03.2024.)**
5. **Dolthub Blogs. UUID atslēgas.** – <https://www.dolthub.com/blog/2023-10-27-uuid-keys/> –  
   **(Resurss apskatīts 27.10.2023.)**
6. **Dribbble.** – <https://www.dribbble.com>  
   **(Resurss apskatīts 14.11.2023.** **)**
7. **FreeCodeCamp. Lietotāju autentifikācija Node.js lietotnē. –** <https://www.freecodecamp.org/news/authenticate-users-node-app/> –  
   **(Resurss apskatīts 05.11.2023.** **)**
8. **Traversy Media. Iemācieties Passport.js autentifikāciju. –** <https://www.youtube.com/watch?v=AuHNCbnQHBc> –  
   **(Resurss apskatīts 12.12.2023.** **)**
9. **Rīgas Valsts Tehnikums. Programmēšanas kursa resursi. –** <https://e.rvt.lv/course/view.php?id=308&section=2> –  
   **(Resurss apskatīts 18.11.2023.** **)**

# PIELIKUMI

## 1. Pielikums. ER diagramma

Pielikums 1. “ER diagramma”