**Profesionālās izglītības kompetences centrs**

**„Rīgas Valsts tehnikums”**

Izglītības programma: Programmēšana

**KVALIFIKĀCIJAS DARBS**

**Auto rezerves daļu internetveikals**

Paskaidrojošais raksts 43. lpp.

|  |  |
| --- | --- |
| Audzēknis: | Raivis Bajārs |
| Vadītājs: | Igors Litvjakovs |
| Normu kontrole: | Olga Sabanska |

**Rīga 2021**

**ANOTĀCIJA**

Kvalifikācijas darba ietvaros tika izstrādās auto rezerves daļu internetveikals. Darba autors ir Rīgas Valsts tehnikuma audzēknis Raivis Bajārs. Darba vadītājs ir Rīgas Valsts tehnikuma skolotājs Igors Litvjakovs.

Kvalifikācijas darbā izstrādātā sistēma ir auto rezerves daļu internetveikals, kurā var apskatīt un iegādāties dažādas rezerves daļas priekš savas automašīnas. Sistēma tika izstrādāta ar programmēšanas valodu PHP izmantojot Laravel un Vue JS ietvarus, kā arī tika izmantota MySQL relāciju datubāžu vadības sistēma.

Kvalifikācijas darba paskaidrojošais raksts satur sevī ievadu, uzdevuma nostādni, prasību specifikāciju, uzdevuma risināšanas līdzekļu izvēles pamatojumu, programmatūras produkta modelēšanas un projektēšanas aprakstu, datu struktūru aprakstu, lietotāja ceļvedi un nobeigumu. Ievadā ir aprakstīti kvalifikācijas darba mērķi un riski. Uzdevuma nostādnē ir aprakstīts kvalifikācijas darba galvenais veidošanas mērķis un kādi uzdevumi tam būs jāveic. Prasību specifikācijā ir aprakstītas sistēmas funkcionālās un nefunkcionālās prasības, kā arī sistēmas izejas un ieejas informācija. Uzdevuma risināšanas līdzekļu izvēles pamatojumā ir aprakstīts, kāda programmēšanas valoda, teksta redaktors, datu bāžu vadības sistēma tika izmantota sistēmas izstrādes laikā. Programmatūras produkta modelēšanas un projektēšanas aprakstā ir apskatāmas datu plūsmu diagrammas, ER diagrammas, kā arī detalizēti aprakstīta sistēmas arhitektūra. Datu struktūru apraksts satur sevī visu tabulu struktūru un to aprakstu un arī tabulu relāciju shēmu. Lietotāja ceļvedis attēlo sistēmas informācijas vizuālo izkārtojumu un paskaidro, kā lietot sistēmu. Nobeigumā ir aprakstīts, kas tika izveidots un, kas neizdevās.

**ANNOTATION**

An online car spare parts online store was developed as part of the qualification work. The author of the work is Raivis Bajārs, a student of Riga State Technical School. The supervisor is Igors Litvjakovs, a teacher at the Riga State Technical School.

The system developed in the qualification paper is an online car spare parts store, where you can view and buy various spare parts for your car. The system was developed with the programming language PHP using Laravel and Vue JS frameworks, as well as using the MySQL relational database management system.

The explanatory article of the qualification work includes an introduction, task statement, requirements specification, justification of the choice of task solving means, description of software product modeling and design, description of data structures, user guide and conclusion. The introduction describes the aims and risks of the qualification work. The task statement describes the main goal of the qualification paper and what tasks it will have to perform. The requirements specification describes the functional and non-functional requirements of the system, as well as the system output and input information. The rationale for choosing the means of solving the task describes which programming language, text editor, database management system was used during the system development. The description of the modeling and design of the software product includes data flow diagrams, ER diagrams, as well as a detailed description of the system architecture. The description of data structures includes the structure of all tables and their description, as well as the scheme of table relations. The user guide displays the visual layout of the system information and explains how to use the system. Finally, it describes what was created and what failed.

**SATURS**

[**IEVADS** 5](#_Toc74168193)

[**1.** **UZDEVUMA NOSTĀDNE** 6](#_Toc74168194)

[**2.** **PRASĪBU SPECIFIKĀCIJA** 7](#_Toc74168195)

[**2.1.** **Ieejas un izejas informācijas apraksts** 7](#_Toc74168196)

[***2.1.1.*** ***Ieejas informācijas apraksts*** 7](#_Toc74168197)

[***2.1.2.*** ***Izejas informācijas apraksts*** 8](#_Toc74168198)

[**2.2.** **Funkcionālās prasības** 9](#_Toc74168199)

[**2.3.** **Nefunkcionālās prasības** 10](#_Toc74168200)

[**3.** **UZDEVUMU RISINĀŠANAS IZVĒLES PAMATOJUMS** 11](#_Toc74168201)

[**4.** **PROGRAMMATŪRAS PRODUKTA MODELĒŠANA UN PROJEKTĒŠANA** 12](#_Toc74168202)

[**4.1.** **Sistēmas struktūras modelis** 12](#_Toc74168203)

[***4.1.1.*** ***Sistēmas arhitektūra*** 12](#_Toc74168204)

[***4.1.2.*** ***Sistēmas ER modelis*** 15](#_Toc74168205)

[**4.2.** **Funkcionālais sistēmas modelis** 17](#_Toc74168206)

[***4.2.1.*** ***Datu plūsmu modelis*** 17](#_Toc74168207)

[**5.** **DATU STRUKTŪRAS APRAKSTS** 25](#_Toc74168208)

[**6.** **LIETOTĀJA CEĻVEDIS** 28](#_Toc74168209)

[**6.1.** **Sistēmas prasības aparatūrai un programmatūrai** 28](#_Toc74168210)

[**6.2.** **Sistēmas instalācija un palaišana** 31](#_Toc74168211)

[**6.3.** **Programmas apraksts** 31](#_Toc74168212)

[**NOBEIGUMS** 43](#_Toc74168213)

[**INFORMĀCIJAS AVOTI** 43](#_Toc74168214)

# **IEVADS**

Kvalifikācijas darba mērķis ir izveidot ērtu un intuitīvu auto rezerves daļu internetveikalu. Veidojot internetveikalu ir vairāki riski ar kuriem bieži sastopas, kā arī komersanti, kuri jau vairākus gadus nodarbojas ar šāda veida biznesu.

Pirmkārt, jāapzinās lietotāju vajadzības un vēlmēs, tamdēļ dotajam internetveikalam ir izveidots ērts un intuitīvs dizains, lai uzlabotu lietotāju pieredzi internetveikala izmantošanas laikā. Otrkārt, internetveikalam ir jāpiedāvā pietiekami daudz funkciju, lai viss vajadzīgais būtu vienā vietā. Treškārt, ir jābūt izveidotai datu bāzei, kurā glabātos visas preces, kategorijas, ražotāji, lietotāji, pasūtījumi. Visus šos datus pārskata sistēmas administratori ar viegla un ērta interfeisa palīdzību.

# **UZDEVUMA NOSTĀDNE**

**Kvalifikācijas darba uzdevums** ir izveidot auto rezerves daļu internetveikalu. Lietotājs varēs reģistrēties, autorizēties, apskatīt preces, pievienot preču grozā jaunu preci un nodzēst to, izmainīt preču daudzumu grozā un veikt pasūtījumu. Lai veiktu pirkumu un noformētu pasūtījumu, lietotājam vajadzēs izvēlēties piegādes veidu un saziņas veidu. Lietotāja pasūtījums parādīsies mājaslapas pasūtījumu sadaļā uzreiz pēc veiksmīgas noformēšanas, kā arī visu pasūtījumu vēsture saglabājas pasūtījumu sadaļā.

Auto rezerves daļu internetveikala datu uzskaites automatizētā sistēmā ir paredzēts 2 lietotāju lomas ( reģistrēts lietotājs un sistēmas administrators) ar dažādiem tiesību līmeņiem. Kopumā sistēmai ir jāizpilda vairākas funkcionalitātes:

• lietotāja reģistrēšanu;

• datu pārvaldi – lietotāju, preču, preču kategoriju pievienošanu, labošanu un dzēšanu;

• preču apskatīšanu, meklēšanu, pievienošanu grozam;

• pasūtījumu noformēšanu;

# **PRASĪBU SPECIFIKĀCIJA**

## **Ieejas un izejas informācijas apraksts**

### ***Ieejas informācijas apraksts***

* Informācija par lietotāju, kas sastāv no:
  + paroles, kas ir simbolu virkne, ne garāka par 255 simboliem;
  + lietotāja vārda, kas ir simbolu virkne, ne garāka par 50 simboliem;
  + lietotāja uzvārda, kas ir simbolu virkne, ne garāka par 50 simboliem;
  + e-pasta, kas ir simbolu virkne, ne garāka par 50 simboliem
  + telefona numura, kas ir simbolu virkne, ne garāka par 12 simboliem;
  + valsts, kas ir simbolu virkne, ne garāka par 50 simboliem;
  + pilsētas, kas ir simbolu virkne, ne garāka par 50 simboliem;
  + adreses, kas ir simbolu virkne, ne garāka par 50 simboliem;
  + pasta indekss, kas ir simbolu virkne, ne garāka par 7 simboliem;
  + saziņas veids, kas ir enum, ar divām izvēlēm – Telefons vai E-pasts
  + lietotāja tips, kas ir bool, 1- administrators, 0 - lietotājs
* Informācija par kategoriju, ko ievada administrators un tā sastāv no:
  + kategorijas nosaukuma, kas ir simbolu virkne, ne garāka par 50 simboliem;
  + apakškategorija, kas ir norāde zem kādas kategorijas atrodas dotā kategorija;
  + kategorijas attēls, kas ir šifrēts base64 formātā;
  + redzamība, kas ir enum ar trim izvēlēm, kategorijas, kategorijas\_un\_produkti, produkti
* Informācija par preci, ko ievada administrators un tā sastāv no:
  + preces nosaukuma, kas ir simbolu virkne, ne garāka par 50 simboliem;
  + preces attēla, kas ir šifrēts base64 formātā;
  + ražotāja, kas ir simbolu virkne, ne garāka par 50 simboliem;
  + kategorijas, kas ir simbolu virkne, ne garāka par 50 simboliem;
  + preces cena, kas ir double
  + daudzums, kas ir skaitlis, ne vairāk par 255 – maksimālais
* Informācija par ražotājiem, ko ievada administrators un tā sastāv no:
  + ražotāja nosaukuma, kas ir simbolu virkne, ne garāka par 50 simboliem;
* Informācija par pasūtījumiem, kas tiek pievienota pēc to veiksmīgas noformēšanas:
  + produkti, kas ir datu masīvs;
  + pasūtījuma cena, kas ir double;
  + lietotājs, kas ir norāde uz lietotāju, kas veic pasūtījumu;
  + status, kas ir enum ar divām izvēlēm – Pabeigts vai Nepabeigts;
  + piegādes veida, kas ir enum ar vienu izvēli – Preces saņemšana veikalā

### ***Izejas informācijas apraksts***

Izejas dati ir no datubāzes izgūtie dati, kā atbilde uz lietotāja pieprasījumu.

* Informācija par lietotāju, kas sastāv no:
  + vārda;
  + uzvārda;
  + numura;
  + e-pasta;
  + valsts;
  + pilsētas;
  + adreses;
  + pasta indeksa;
  + saziņas veida;
  + lietotāja tips;
* Informācija par kategoriju:
  + kategorijas nosaukums;
  + apakškategorija;
  + kategorijas attēls;
  + redzamība;
* Informācija par precēm:
  + preces nosaukums;
  + preces attēls;
  + ražotājs;
  + kategorija;
  + preces cena;
  + daudzums;
* Informācija par ražotājiem:
  + ražotāju nosaukums;
* Informācija par pasūtījumiem:
  + produkti;
  + pasūtījuma cena;
  + lietotājs;
  + status;
  + piegādes veids;

## **Funkcionālās prasības**

* Lietotāja reģistrācija:
* obligātie un neobligātie lauki reģistrācijas procesā. Pārbaudes katra lauka pareizumam;
* paroles šifrēšana;
* pēc noklusējuma tiek piešķirta parastā lietotāja loma. Ja rodas vajadzība, ērti
* maināma loma lietotājiem datubāzē;
* Lietotāja profila iespējas:
  + visu profila detaļu labošana, izņemot lietotājvārdu;
  + iziešana no lapas.
* Internetveikals:
  + kategoriju izveide (tikai administratori);
  + preču pievienošana attiecīgajā kategorijā (tikai administratori);
  + preču labošana;
  + kategoriju labošana;
  + kategoriju attēlošana;
  + preču attēlošana;
* Pasūtījumi:
  + statusa glabāšana datubāzē, lai viegli var izsekot, kas ir samaksājis un kas nav.
  + automātiska statusa maiņa pēc veiksmīgas samaksas;

## **Nefunkcionālās prasības**

* Prasības procesam:
  + lapas datu bāzei jābūt izstrādātai atvērtā pirmkoda datu bāžu pārvaldības sistēmā phpMyAdmin – 5.0.3 versijā;
  + lapai jābūt izstrādātai ar PHP – 7.4.11 versijas - servera puses programmēšanas
  + valodu un ar CSS3, HTML5 un JAVASCRIPT lietotāja puses programmēšanas valodām;
  + izstrādē jāizmanto Laravel un VueJS ietvaru;
  + lapai jābūt ātram ielādes laikam un tā nedrīkst pārslogot datora procesoru.
* Prasības produktam:
  + lietotāju saskarnei ar sistēmu jānotiek latviešu valodā;
  + lapai jābūt savietojamai uz visiem ekrānu izmēriem un to izšķirtspējām. Proti, gan uz mobilajiem telefoniem, gan uz planšetēm, gan uz datoriem un citām tamlīdzīgām ierīcēm.
* Prasības lietotāja interfeisam:
  + lapas pamatkrāsai jābūt tumši pelēkai krāsai. Tai jābūt ne pārāk uzkrītošai un neitrālai;
  + visiem lapas galvenajiem virsrakstiem jābūt ieturētā un vienādā stilā;
  + kļūdu paziņojumiem jābūt vienotā stilā;
  + kategorijām ir jābūt skaisti atdalītām no citiem logrīkiem un kolonnu robežām, un tiem jābūt labi pamanāmiem;
  + lapai jābūt ērti pārskatāmai un lietojamai jebkuram interneta tīmekļa izmantotājam.

# **UZDEVUMU RISINĀŠANAS IZVĒLES PAMATOJUMS**

Kvalifikācijas darba ietvaros tika izmantotas vairākas atklātā pirmkoda skriptu valodas. Pirmā ir servera puses programmēšanas valoda – PHP. Izvēlējos šo programmēšanas valodu, jo tā tiek izmantota visas mūsdienu sistēmās un piedāvā ļoti daudzus ietvarus, kā piemēram Laravel, kurš tika izmantots šajā projektā. Izmantota tika PHP 7.4.11 versija.

Lapas pamatā tika izmantota HTML valoda. Izvēlējos šo valodu, jo tas ir standarts bez kura nevar iztikt mājaslapu izstrādē. Izmantota tika HTML5 versija. Kvalifikācijas darbā šī valoda atspoguļo tīmekļa lapas un tās glabājamo informāciju.

Tīmekļa lapas izskata pielāgošanai tika izmantota lapas stilu valoda – CSS. Tika pielietota šīs valodas pēdējā versija – CSS3 un SASS paplašinājums. Kvalifikācijas darbā šī valoda ļauj pielāgot tīmekļa lapas izskatu.

Dinamiskai tīmekļa lapas ģenerēšanai tika izmantota Javascript programmēšanas valoda ar ietvaru – VueJS 2.6.11.

Darbam ar datubāzi tika izmantots phpMyAdmin, kas ir MySQL datubāzes pārvaldības sistēma. Šī sistēma tika izvēlētā, jo tā piedāvā lietotāja saskarni tīmeklī un piedāvā iespēju veidot datubāzes un to tabulas bez SQL komandu ievades. Izmantota tika phpMyAdmin 5.0.3 versija.

Darbībām ar šīm programmēšanas valodām tika izmantots Visual Studio Code teksta redaktors. Izvēlējos šo teksta redaktoru, jo tas atbalsta visas programmēšanas valodas un piedāvā iespaidīgu daudzumu ar paplašinājumiem.

# **PROGRAMMATŪRAS PRODUKTA MODELĒŠANA UN PROJEKTĒŠANA**

## **Sistēmas struktūras modelis**

### ***Sistēmas arhitektūra***

Sistēmā ir realizētas trīs lietotāju lomas. Katrai lietotāja lomai ir pieejamas visas kategorijas, ar atšķirību, ka lietotāja panelī sistēmas administratoram ir iespēja pievienot preces un kategorijas.

Sistēma ir maksimāli vienkāršota, kas lietotājām ļauj ātrāk nonākt uz sev vajadzīgo kategoriju vai preci. Lai gūtu visu sistēmas funkcionalitāti, lietotājam ir jāveic reģistrācija sistēmā.

Lapa sastāv no sekojošajām sistēmām:

**Viesa sistēma** – piedāvā tikai tīmekļa lapas apskatu. Ir pieejams apskatīt tikai daļu no moduļiem. Lai iegūtu visu funkcionalitāti, lietotājam ir jāreģistrē profils. Viesu sistēma sastāv no sekojošiem moduļiem.

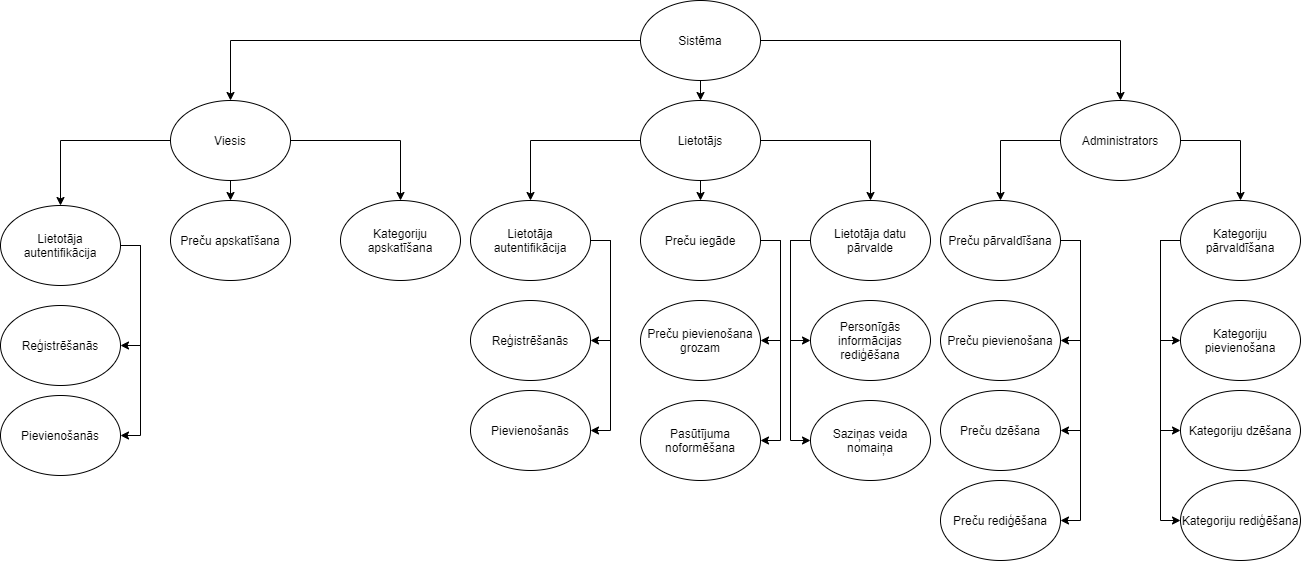
* Modulis “Kategorijas”:
  + Sastāv no 2 apakšmoduļiem – kategorijām un precēm. Viesis var apskatīt kategorijas un preces, bet nevar veikt nekādas darbības ar tām.

**Lietotāja sistēma** – sastāv no visām lietotāju plānotajām funkcijām. Lietotāja funkcijas daļās ir lietotāja autentifikācijas, lietotāja datu pārvaldības un preču iegāde. Lietotāja autentifikācija sastāv no lietotāju reģistrēšanās sistēmā un ja lietotājs jau ir reģistrējies, tad tā sastāv arī no pieslēgšanās sistēmai ar saviem datiem. Lietotāja datu pārvalde sastāv no lietotāja funkcijām, kur lietotājs var pats izmainīt savus datus, piemēram, adresi, tālruni, saziņas veidu. Preču iegādes daļa sastāv no funkcijām, kuras nepieciešamas, lai varētu apskatīt un pievienot preces grozam, un galu galā noformēt pasūtījumu.

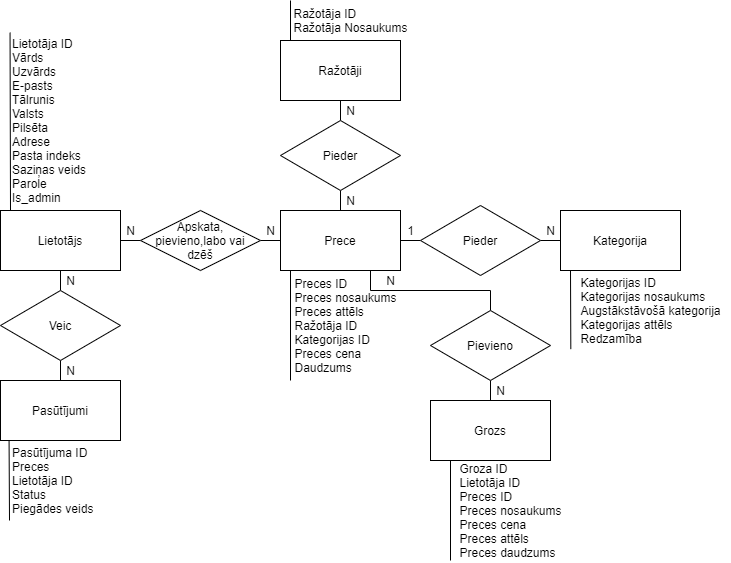
* Modulis “Kategorijas”:
  + Sastāv no 2 apakšmoduļiem – kategorijām un precēm. Lietotājs var apskatīt kategorijas un preces, pievienot preces grozam.
* Modulis “Grozs”:
  + Lietotājs var apskatīt grozā pievienotās preces, izmanīt daudzumu vai izdzēst preci no groza, kā arī noformēt pasūtījumu.
* Modulis “Lietotāja panelis”:
  + Lietotājs var apskatīt savu personīgo informāciju, izmainīt to un, kā arī izvēlēties saziņas veidu.
* Modulis “Pasūtījumi”:
  + Lietotājs var apskatīt savu pasūtījumu vēsturi un to statusu.

**Administratora sistēma** – ir saistīta ar visām sistēmas administratora darbībām. Sistēmas administratoram būs pieeja pārskatīt un pārvaldīt visus ierakstus datu bāzē. Pateicoties šim administrators vienmēr varēs atrisināt jebkādus tehniskos jautājumus saistībā ar sistēmu un viegli rediģēt jebkurus ierakstu ja rodas tāda vajadzība.

* Modulis “Kategorijas”:
  + Sastāv no 2 apakšmoduļiem – kategorijām un precēm. Administrators var apskatīt kategorijas un preces, pievienot preces grozam.
* Modulis “Grozs”:
  + Administrators var apskatīt grozā pievienotās preces, izmanīt daudzumu vai izdzēst preci no groza, kā arī noformēt pasūtījumu.
* Modulis “Lietotāja panelis”:
  + Administrators var apskatīt savu personīgo informāciju, izmainīt to un, kā arī izvēlēties saziņas veidu. Iespēja pievienot preces un kategorijas.
* Modulis “Pasūtījumi”:
  + Administrators var apskatīt savu pasūtījumu vēsturi un to statusu, kā arī izmanīt to.

4.1 att. Sistēmas arhitektūra

### ***Sistēmas ER modelis***



* 1. att. Sistēmas ER diagramma

Datu bāzes projektēšanā datu kopu un saišu starp tām attēlošanai tika lietota realitāšusaišu diagramma, kas sastāv no divu veidu objektiem – entītēm (loģiskais objekts) un relācijām (attiecības).

Datu bāzes ER modelis , kas ir dots 4.2.attēlā, sastāv no sekojošām entītijām:

• „Preces” – apraksta sistēmā pievienotās preces. Tās atribūtu kopums sevī ietver preces ID, preces nosaukumu, preces attēlu, ražotāja ID, kategorijas ID, preces cenu un to pieejamo daudzumu.

• „Ražotāji” – apraksta sistēmā pievienotos ražotājus. Tās atribūtu kopums sevī ietver ražotāja ID un ražotāja nosaukumu.

• „Grozs” – apraksta lietotāju izveidotus grozus sistēmā. Tās atribūtu kopums sevī ietver groza ID, lietotāja ID, preces ID, preces nosaukumu, preces cenu, preces attēlu un preces daudzumu.

• „Kategorijas” – apraksta sistēmā pievienotās kategorijas. Tās atribūtu kopums sevī ietver kategorijas ID, kategorijas nosaukumu, augstākstāvošo kategoriju, kategorijas attēlu, un redzamību.

• „Lietotāji” – apraksta sistēmā reģistrētos lietotājus. Tās atribūtu kopums sevī ietver lietotāja ID, vārdu, uzvārdu, e-pastu, tālruni, valsti, pilsētu, adresi, pasta indeksu, saziņas veidu, paroli un vai lietotājs ir administrators.

• „Pasūtījumi” – apraksta lietotāju izveidotus pasūtījumus sistēmā. Tās atribūtu kopums sevī ietver pasūtījuma ID, preces, lietotāja ID, statusu un piegādes veidu.

Datu bāzes relācijas parada kā savstarpēji ir savienotas divas vai vairākas entītijas:

• starp precēm un grozu attiecība ir daudzi pret daudziem, jo viena prece var atrasties vairākos grozos un vienā grozā var atrasties vairākas preces;

• starp precēm un ražotājiem attiecība ir daudzi pret daudziem, jo vienu preci var ražot vairāki ražotāji un viens ražotājs var ražot vairākas preces;

• starp precēm un kategorijām attiecības ir viens pret daudziem, jo vairākas preces var atrasties vienā kategorijā, bet viena prece nevar atrasties vairākās kategorijās;

• starp grozu un lietotājiem attiecības ir viens pret vienu, jo vienam lietotājam var būt tikai viens grozs;

• starp grozu un pasūtījumiem attiecības ir viens pret vienu, jo no groza var veikt tikai vienu pasūtījumu;

## **Funkcionālais sistēmas modelis**

### ***Datu plūsmu modelis***

**1. Preču pievienošana**

Vispirms tiek pārbaudīts vai lietotājs ir pieslēdzies sistēma. Ja lietotājs ir pieslēdzies sistēmā, tad tiek pārbaudīts vai lietotājs ir sistēmas administrators.

Kad esam pārliecinājušies, ka lietotājs ir administrators, tiek parādīts administratora panelis, ar divām formām – kategoriju pievienošana un preču pievienošana. Lai pievienotu preces ir jāaizpilda visi nepieciešamie lauki preču pievienošanas formā. Gadījumā, ja administrators nav aizpildījis visus nepieciešamos laukus, izvadam kļūdas paziņojumu par konkrētiem neaizpildītiem laukiem un lūdzam lietotāju aizpildīt nepieciešamos laukus, lai turpinātu preču pievienošanas procesu.

Ja visi nepieciešamie lauki sakrīt, laiks pārbaudīt laukus. Pirmo pārbaudam preces nosaukumu. Pārbaudam vai ievadītais preces nosaukums sastāv tikai no burtiem, ja preces nosaukumā ir cipari vai citas rakstzīmes, izvadam kļūdas paziņojumu un informējam lietotāju par to, ka preces nosaukumam jāsastāv tikai no burtiem.

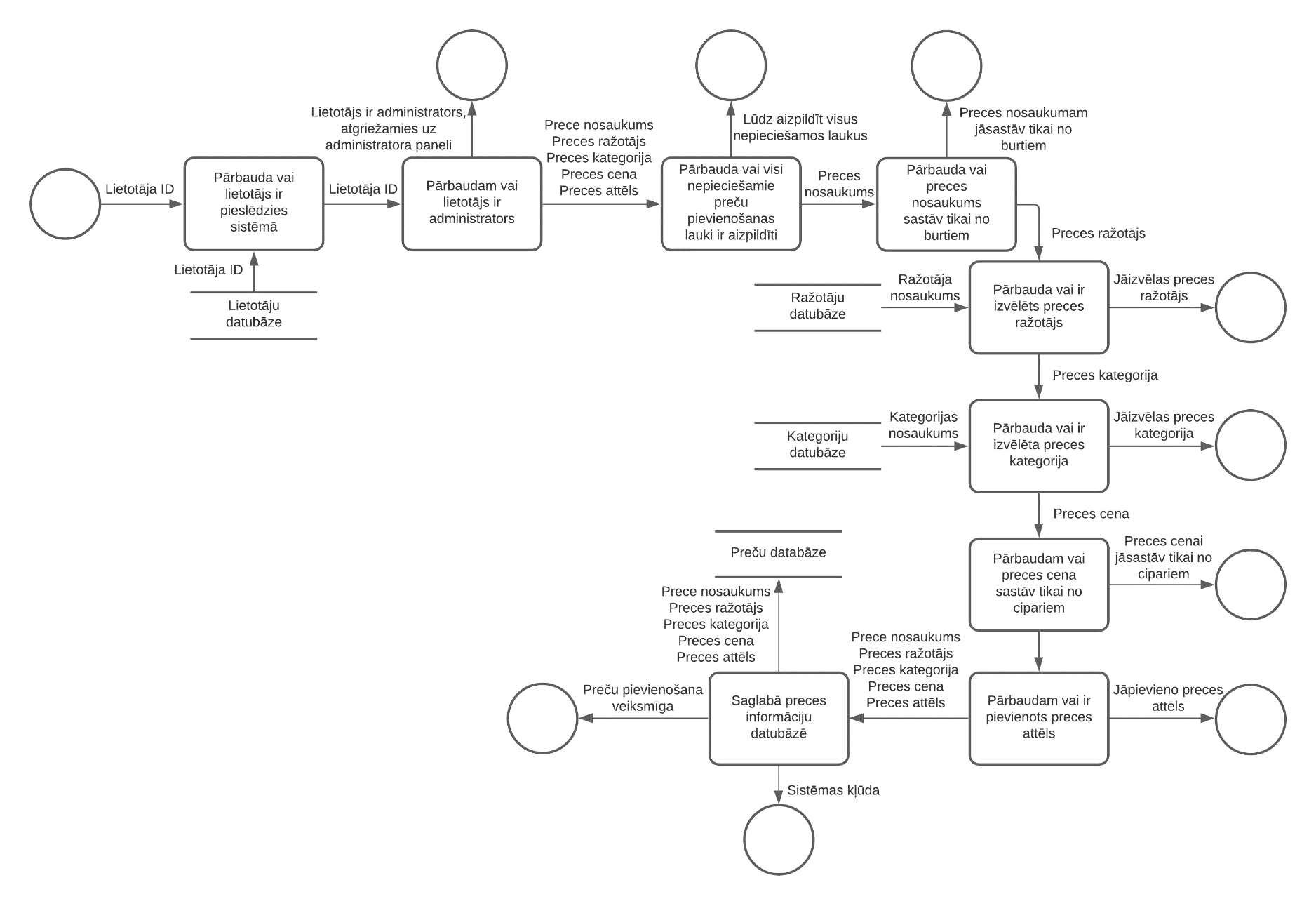
Nākamais lauks ir preces ražotājs. Pārbaudam vai ir izvēlēts preces ražotājs, ja nav izvēlēts preces ražotājs, izvadam kļūdas paziņojumu un informējam lietotāju par to, ka jāizvēlas preces ražotājs.

Pārbaudam kategorijas lauku. Pārbaudam vai ir izvēlēta preces kategorija, ja nav izvēlēta preces kategorija, izvadam kļūdas paziņojumu un informējam lietotāju par to, ka jāizvēlas preces kategorija.

Tālāk pārbaudam preces cenu. Pārbaudam vai ievadītā preces cena sastāv tikai no cipariem, ja ievadītā preces cena sastāv no burtiem vai citām rakstzīmēm, izvadam kļūdas paziņojumu un informējam lietotāju par to, ka preces cenai jāsastāv tikai no cipariem.

Nākamo pārbaudam preces attēlu. Pārbaudam vai ir pievienots preces attēls, ja nav pievienots preces attēls, izvadas kļūdas paziņojumu un informējam lietotāju par to, ka jāpievieno preces attēls.

Visbeidzot saglabājam administratora ievadītos preču datus datubāzē. Ja datu ievietošana datubāzē bijusi neveiksmīga, tiek izvadīts sistēmas kļūdas paziņojums. Citādi izvadam paziņojumu, ka prece pievienota veiksmīgi.

* 1. att. Preču pievienošanas datu plūsmu diagramma

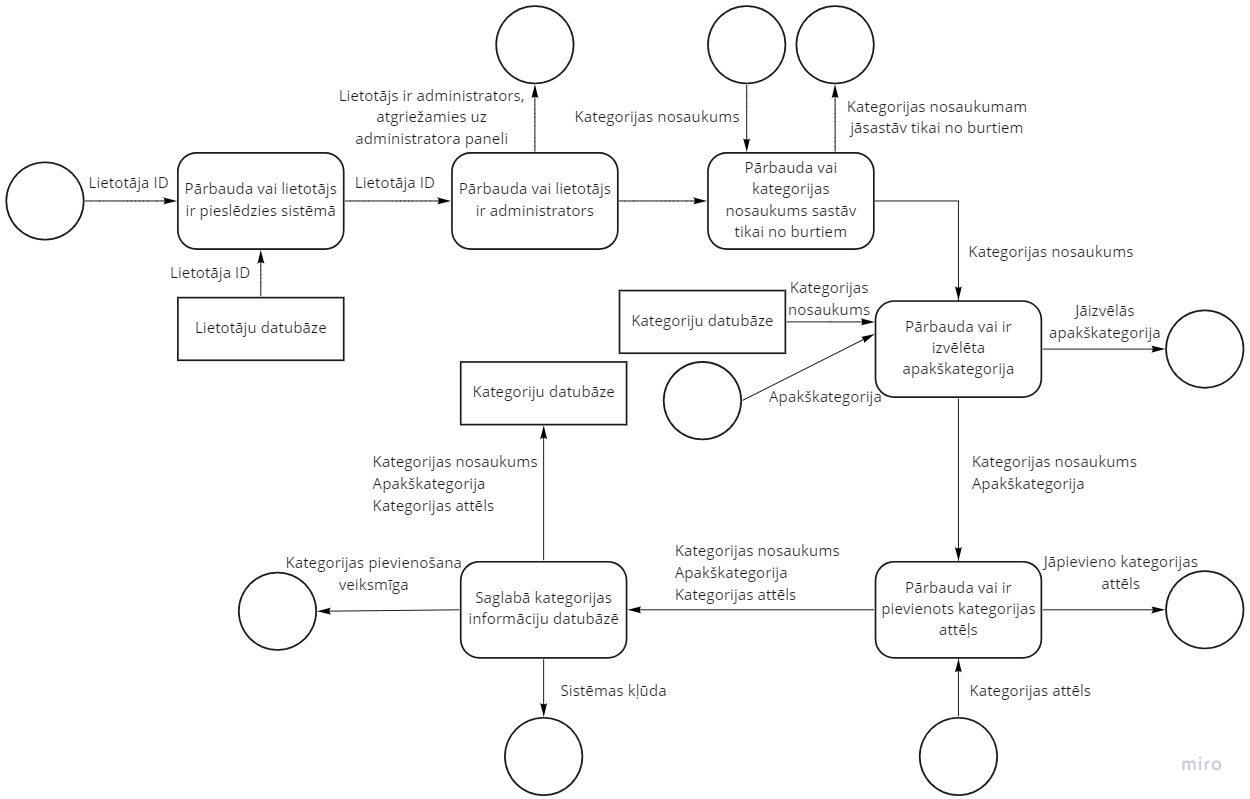
**2. Kategoriju pievienošana**

Vispirms tiek pārbaudīts vai lietotājs ir pieslēdzies sistēma. Ja lietotājs ir pieslēdzies sistēmā, tad tiek pārbaudīts vai lietotājs ir sistēmas administrators.

Kad esam pārliecinājušies, ka lietotājs ir administrators, tiek parādīts administratora panelis, ar divām formām – kategoriju pievienošana un preču pievienošana. Lai pievienotu kategorijas ir jāaizpilda visi nepieciešamie lauki kategorijas pievienošanas formā. Gadījumā, ja administrators nav aizpildījis visus nepieciešamos laukus, izvadam kļūdas paziņojumu par konkrētiem neaizpildītiem laukiem un lūdzam lietotāju aizpildīt nepieciešamos laukus, lai turpinātu kategorijas pievienošanas procesu.

Ja visi nepieciešamie lauki sakrīt, laiks pārbaudīt laukus. Pirmo pārbaudam kategorijas nosaukumu. Pārbaudam vai ievadītais kategorijas nosaukums sastāv tikai no burtiem, ja kategorijas nosaukumā ir cipari vai citas rakstzīmes, izvadam kļūdas paziņojumu un informējam lietotāju par to, ka kategorijas nosaukumam jāsastāv tikai no burtiem.

Nākamais lauks ir apakškategorijas. Pārbaudam vai ir izvēlēta apakškategorija, ja nav izvēlēta apakškategorijas, izvadas kļūdas paziņojumu un informējam lietotāju par to, ka jāizveles apakškategorija.

****Visbeidzot saglabājam administratora ievadītos kategorijas datus datubāzē. Ja datu ievietošana datubāzē bijusi neveiksmīga, tiek izvadīts sistēmas kļūdas paziņojums. Citādi izvadam paziņojumu, ka kategorija pievienota veiksmīgi.

* 1. att. Kategoriju pievienošanas datu plūsmas diagramma

**3. Lietotāja reģistrācija**

Vispirms tiek pārbaudīts vai lietotājs ir pieslēdzies sistēma. Ja lietotājs ir pieslēdzies, atgriežam lietotāju sākumlapā, jo reģistrācija nav vajadzīga.

Kad esam pārliecinājušies, ka lietotājs nav reģistrējies, pārbaucam vai visi nepieciešamies lauki reģistrācijas formā ir aizpildīti. Ja visi nepieciešamie lauki nav aizpildīti, izvadam kļūdas paziņojumu par konkrētiem neaizpildītiem laukiem un lūdzam lietotāju aizpildīt nepieciešamos laukus, lai turpinātu reģistrācijas procesu.

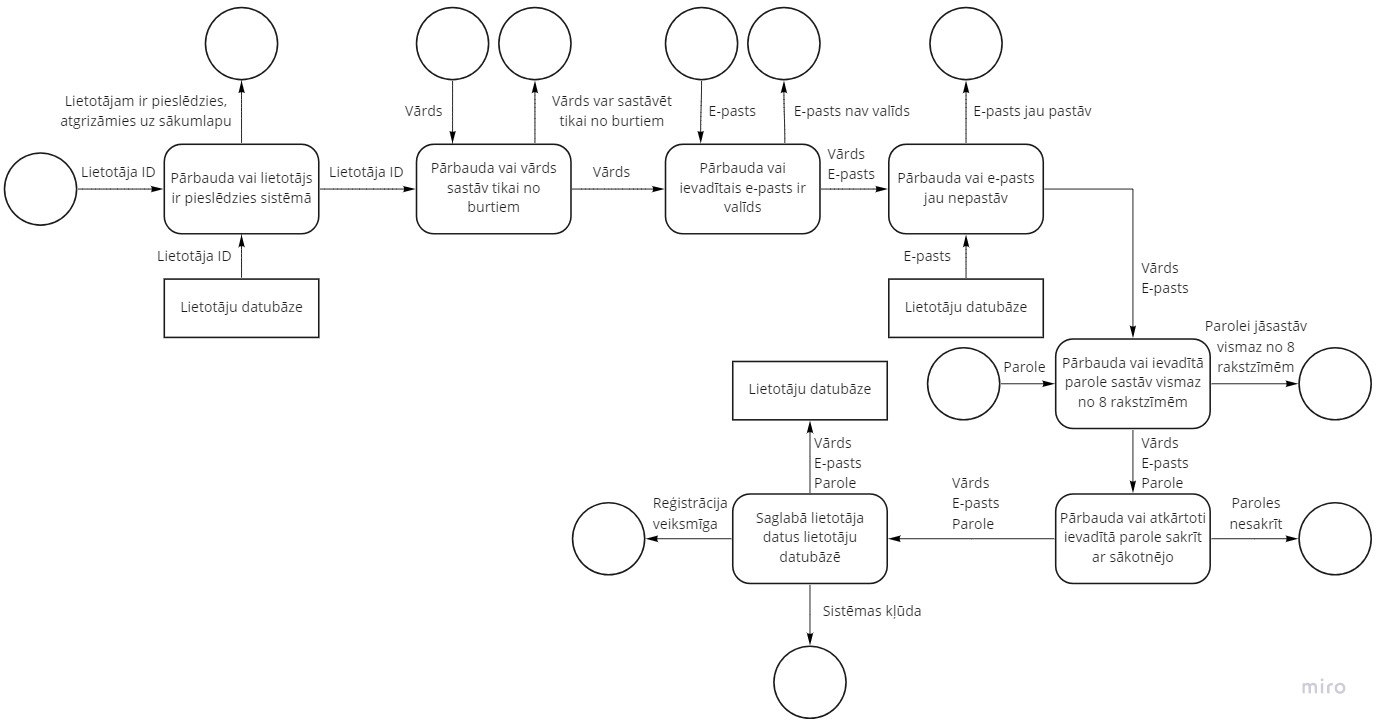
Ja visi nepieciešamie lauki sakrīt, laiks pārbaudīt laukus. Pirmo pārbaudam vārdu. Pārbaudam vai ievadītais vārds sastāv tikai no burtiem, ja vārdā ir cipari vai citas rakstzīmes, izvadam kļūdas paziņojumu un informējam lietotāju par to, ka vārdam jāsastāv tikai no burtiem.

Nākamais lauks ir e-pasts. Pārbaudam vai ievadītais e-pasts ir valīds, ja e-pasts validāciju neiziet veiksmīgi, tad izvadam kļūdas paziņojumu un informējam lietotāju par to, ka ievadīties e-pasts nav valīds. Ja e-pasts ir valīds pārbaudam vai e-pasts jau nav aizņemts, ja tas ir aizņemts izvadam kļūdas paziņojumu un informējam lietotāju par to, ka ievadītais e-pasts jau pastāv..

Pārbaudam paroles lauku. Pārbaudam vai ievadīta parole sastāv vismaz no 8 rakstzīmēm, ja tā ir īsāka par 8 rakstzīmēm, izvadam kļūdas paziņojumu un informējam lietotāju par to, ka parolei jāsastāv vismaz no 8 rakstzīmēm.

Tālāk pārbaudam vai atkārtoti ievadītā parole sakrīt ar vēlamo paroli. Ja ievadītā parole nesakrīt ar vēlamo paroli, izvadam kļūdas paziņojumu un informējam lietotāju par to, ka paroles nesakrīt.

Visbeidzot saglabājam lietotāja ievadītos reģistrācijas datus datubāzē. Ja datu ievietošana datubāzē bijusi neveiksmīga, tiek izvadīts sistēmas kļūdas paziņojums. Citādi izvadam paziņojumu, ka lietotājs veiksmīgi reģistrēts.

* 1. att. Lietotāja reģistrācijas datu plūsmas diagramma

**4. Pasūtījuma noformēšana**

Vispirms tiek pārbaudīts vai lietotājs ir pieslēdzies sistēma. Ja lietotājs ir pieslēdzies sistēmā, tad tiek pārbaudīts vai lietotājs ir sistēmas administrators.

Kad esam pārliecinājušies, ka lietotājs ir pieslēdzies sistēmā. Pārliecinamies, ka visi nepieciešamie lauki pasūtījuma noformēšanas formā ir aizpildīti. Gadījumā, ja lietotājs nav aizpildījis visus nepieciešamos laukus, izvadam kļūdas paziņojumu par konkrētiem neaizpildītiem laukiem un lūdzam lietotāju aizpildīt nepieciešamos laukus, lai turpinātu pasūtījuma noformēšanas procesu.

Ja visi nepieciešamie lauki sakrīt, laiks pārbaudīt laukus. Pirmo pārbaudam vārdu. Pārbaudam vai ievadītais vārds sastāv tikai no burtiem, ja vārdā ir cipari vai citas rakstzīmes, izvadam kļūdas paziņojumu un informējam lietotāju par to, ka vārdam jāsastāv tikai no burtiem.

Nākamais lauks ir uzvārds. Pārbaudam vai ievadītais uzvārds sastāv tikai no burtiem, ja uzvārdā ir cipari vai citas rakstzīmes, izvadam kļūdas paziņojumu un informējam lietotāju par to, ka uzvārdam jāsastāv tikai no burtiem.

Pārbaudam tālruņa lauku. Pārbaudam vai ievadītais tālrunis sastāv tikai no cipariem, ja tālrunī ir burti vai citas rakstzīmes, izvadam kļūdas paziņojumu un informējam lietotāju par to, ka tālrunim jāsastāv tikai no cipariem.

Tālāk pārbaudam e-pastu. Pārbaudam vai ir ievadīts e-pasts, ja nav ievadīts e-pasts, izvadam kļūdas pazinojumu un informējam lietotāju par to, ka jāievada e-pasts.

Nākamo pārbaudam valsti. Pārbaudam vai ir izvēlēta valsts, ja valsts nav izvēlēta, izvadam kļūdas paziņojumu un informējam lietotāju par to, ka jāizvēlas valsts.

Pārbaudam pilsētas lauku. Pārbaudam vai ir ievadīta pilsēta, ja pilsēta nav ievadīta, izvadam kļūdas paziņojumu un informējam lietotāju par to, ka jāievada pilsēta.

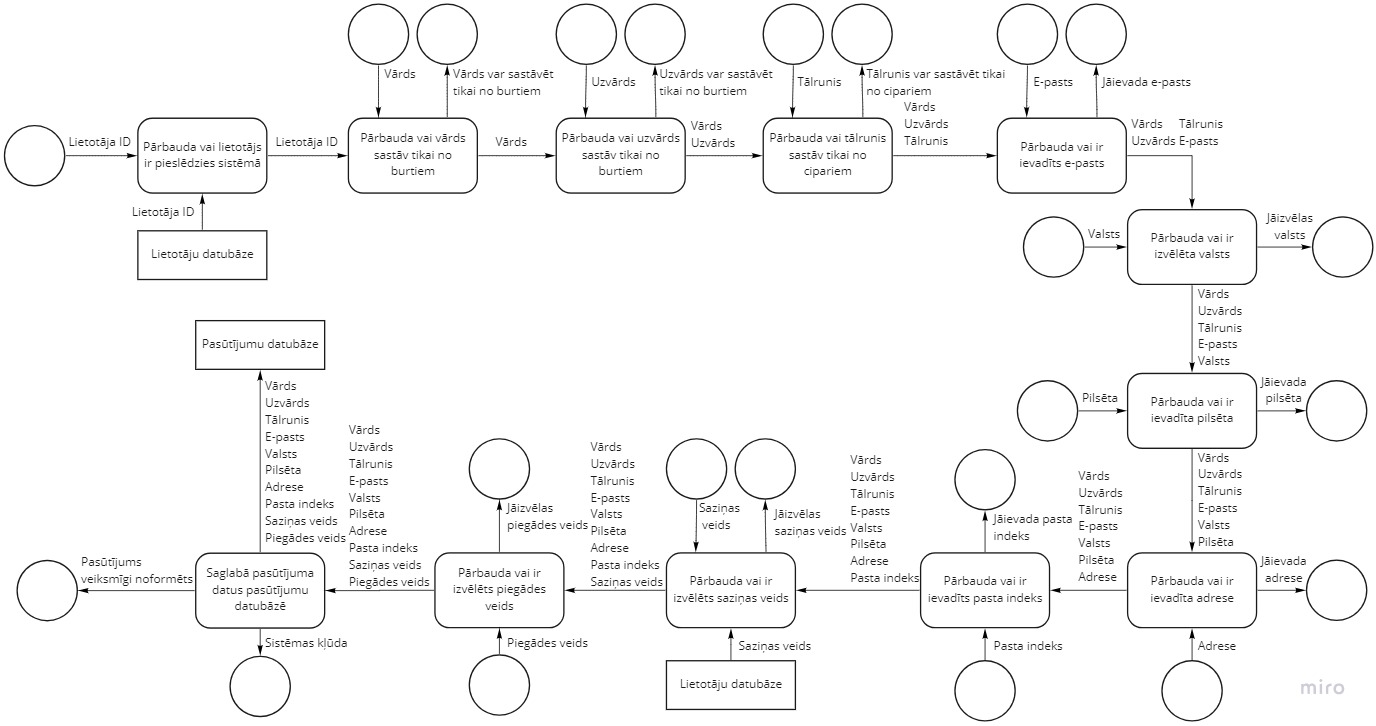
Tālāk pārbaudam adresi. Pārbaudam vai ir ievadīta adrese, ja adrese nav ievadīta, izvadam kļūdas paziņojumu un informējam lietotāju par to, ka jāievada adrese.

Nākamo pārbaudam pasta indeksa lauku. Pārbaudam vai ir ievadīts pasta indeks, ja pasta indeks nav ievadīts, izvadam kļūdas paziņojumu un informējam lietotāju par to, ka jāievada pasta indeks.

Tālāk pārbaudam saziņas veidu. Pārbaudam vai ir izvēlēts saziņas veids, ja saziņas veids nav izvēlēts, izvadam kļūdas paziņojumu un informējam lietotāju par to, ka jāizvēlas saziņas veids.

Pārbaudam piegādes veida lauku. Pārbaudam vai ir izvēlēts piegādes veids, ja piegādes veids nav izvēlēts, izvadam kļūdas paziņojumu un informējam lietotāju par to, ka jāizvēlas piegādes veids.

Visbeidzot saglabājam lietotāja ievadītos pasūtījuma datus datubāzē. Ja datu ievietošana datubāzē bijusi neveiksmīga, tiek izvadīts sistēmas kļūdas paziņojums. Citādi izvadam paziņojumu, ka pasūtījums veiksmīgi noformēts.

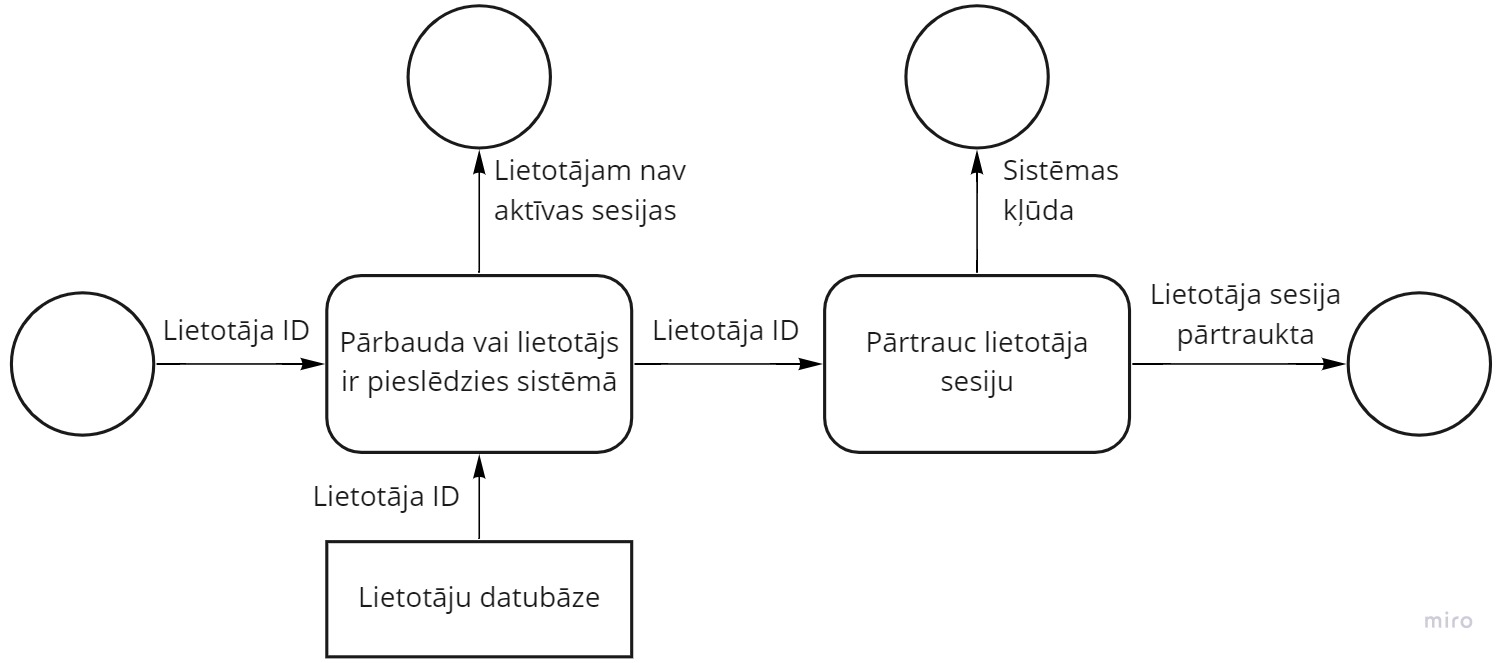
****

* 1. att. Pasūtījuma noformēšanas datu plūsmas diagramma

**5. Lietotāja sesijas pārtraukšana**

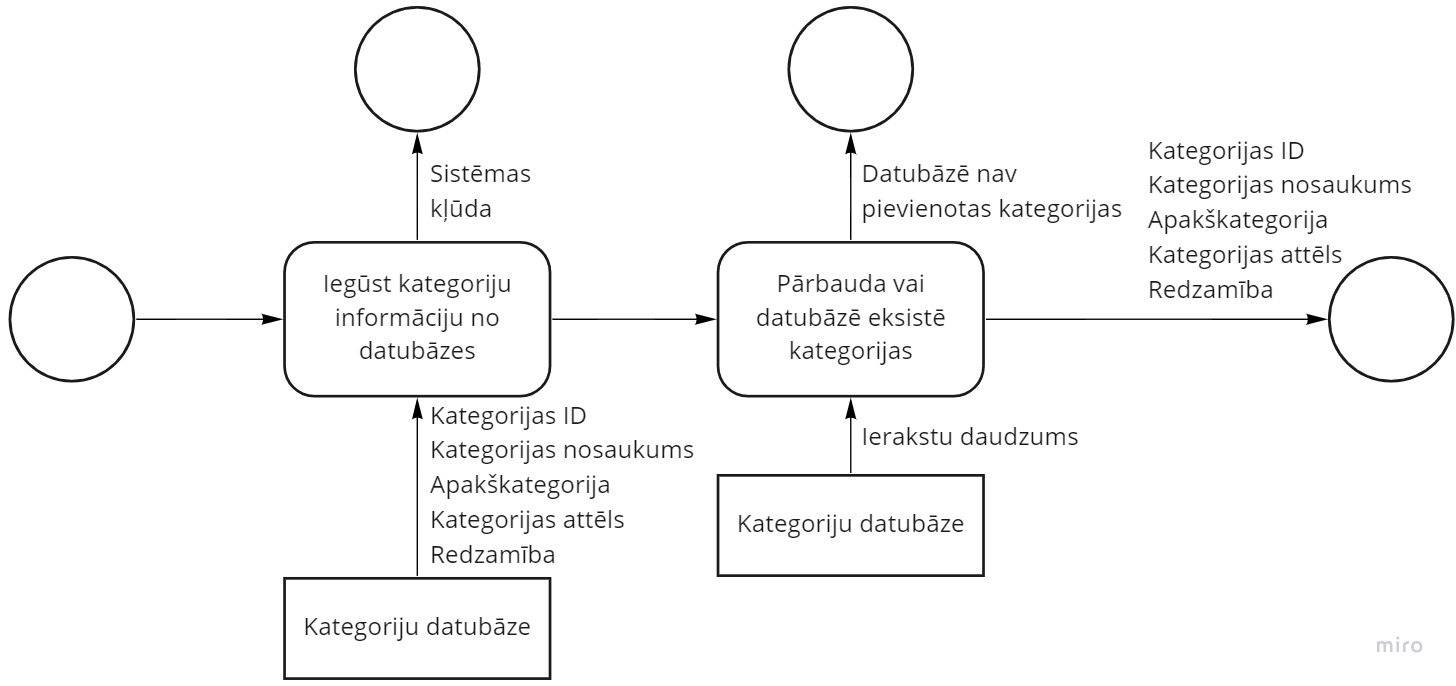
Vispirms tiek pārbaudīts vai lietotājs ir pieslēdzies sistēma. Ja lietotājs nav pieslēdzies sistēmā – sesijas pārtraukšana nav vajadzīga. Lietotājs tiek atgriezts uz sākumlapu.

Kad esam pārliecinājušies, ka lietotājs ir pieslēdzies sistēmā, pārtraucam lietotāja sesiju. Kļūdas gadījumā tiek izvadīta sistēmas kļūda un lietotājs tiek informēts par to. Citādi izvadam, ka lietotāja sesija ir veiksmīgi pārtraukta.

* 1. att. Lietotāja sesijas pārtraukšanas datu plūsmas diagramma

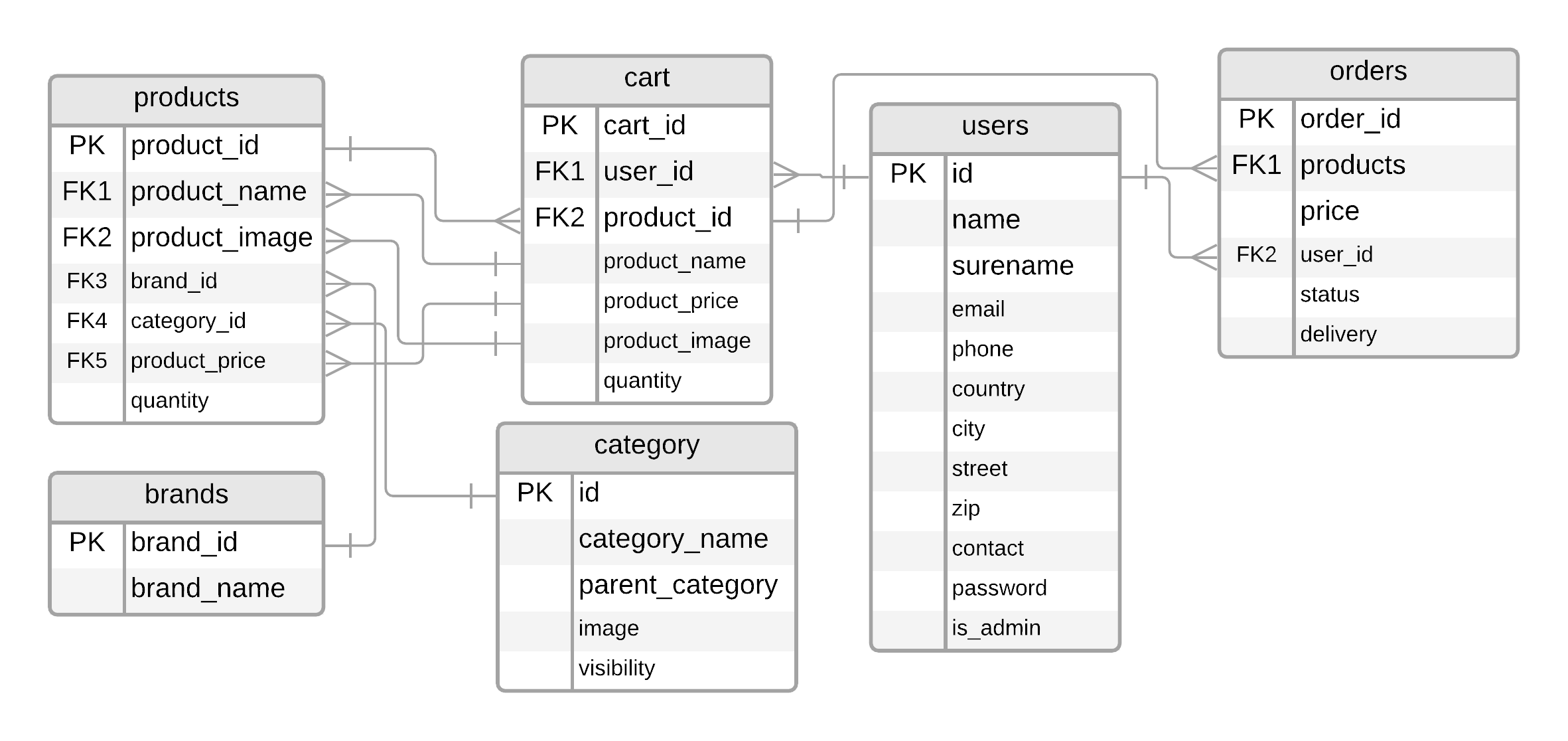
**6. Kategoriju attēlošana**

Vispirms tiek iegūta kategoriju informācija no datubāze. Ja informācijas iegūšanas procesā ir notikusi kļūme, tiek izvadīts sistēmas kļūdas paziņojums un lietotājs tiek informēts par to..

Kad ir iegūta informācija no datubāzes pārbaudam vai datubāzē ir pievienotas kategorijas. Ja netiek atrasts neviens ieraksts datubāzē, izvada paziņojumu informējot lietotāju par to, ka nav nevienas kategorijas. Pretējā gadījumā izvadām kategorijas lietotājam.

* 1. att. Kategoriju attēlošanas datu plūsmas diagramma

# **DATU STRUKTŪRAS APRAKSTS**

Datubāzē sastāv no 6 tabulām, kas satur informāciju par sistēmas lietotājiem, precēm, ražotājiem, kategorijām, pasūtījumiem un lietotāju groza saturu. Preces ir galvenais platformas objekts, kas savienojas ar vairākām tabulām. Zemāk tiek attēlota datubāzes datu relācijas shēma (skat. 5.1. att.).

* 1. att. Datu relācijas shēma

Tabulā “Products” glabājas informācija par administratoru pievienotajām precēm un precēm, kas pieejamas pirkšanai/pasūtīšanai. Informācija par preci, t.sk. produkta id, produkta nosaukums, produkta attēls, ražotāja id, kategorijas id, produkta cena, daudzums.

5.1. tabula

Tabulas “**Products**” struktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr.** | **Nosaukums** | **Tips** | **Garums** | **Piezīmes** |
| 1 | Product\_ID | Auto\_Increment | - | Primārā atslēga |
| 2 | Product\_Name | Varchar | 50 | Produkta nosaukums |
| 3 | Product\_Image | Longtext | - | Produkta attēls |
| 4 | Brand\_ID | Int | 11 | Ražotāja ID |
| 5 | Category\_ID | Int | 11 | Kategorijas ID |
| 6 | Product\_Price | Double | - | Produkta cena |
| 7 | Quantity | Int | 255 | Pieejamais produktu daudzums |
| 8 | Updated\_At | Date | - | Kad prece tika atjaunota |
| 9 | Created\_At | Date | - | Kad prece tika pievienota |

Tabulā “Brands” glabājas informācija par administratoru pievienotajiem ražotājiem. Informācija par ražotājiem, t.sk. ražotāja id, ražotāja nosaukums.

5.2. tabula

Tabulas “**Brands**” struktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr.** | **Nosaukums** | **Tips** | **Garums** | **Piezīmes** |
| 1 | Brand\_ID | Auto\_Increment | - | Primārā atslēga |
| 2 | Brand\_Name | Varchar | 50 | Ražotāja nosaukums |
| 3 | Updated\_At | Date | - | Kad ražotājs tika atjaunots |
| 4 | Created\_At | Date | - | Kad ražotājs tika pievienots |

Tabulā “Cart” glabājas informācija par lietotāju izveidotajiem groziem un to saturu. Informācija par grozu, t.sk. groza id, lietotāja id, produkta id, produkta nosaukums, produkta cena, produkta attēls, daudzums.

5.3. tabula

Tabulas “**Cart**” struktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr.** | **Nosaukums** | **Tips** | **Garums** | **Piezīmes** |
| 1 | Cart\_ID | Auto\_Increment | - | Primārā atslēga |
| 2 | User\_ID | Int | 11 | Lietotāja ID |
| 3 | Product\_ID | Int | 11 | Produkta ID |
| 4 | Product\_Name | Varchar | 50 | Produkta nosaukums |
| 5 | Product\_Price | Double | - | Produkta cena |
| 6 | Product\_Image | Longtext | - | Produkta attēls |
| 7 | Quantity | Int | 255 | Izvēlētā produkta daudzums |
| 8 | Updated\_At | Date | - | Kad grozs tika atjaunots |
| 9 | Created\_At | Date | - | Kad grozs tika pievienots |

Tabulā “Category” glabājas informācija par administratoru pievienotājām kategorijām. Informācija par kategorijām, t.sk. kategorijas id, kategorijas nosaukums, apakškategorija, attēls, redzamība.

5.4. tabula

Tabulas “**Category**” struktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr.** | **Nosaukums** | **Tips** | **Garums** | **Piezīmes** |
| 1 | ID | Auto\_Increment | - | Primārā atslēga |
| 2 | Category\_Name | Varchar | 50 | Kategorijas nosaukums |
| 3 | Parent\_Category | Int | 11 | Apakškategorijas ID |
| 4 | Image | Longtext | - | Kategorijas attēls |
| 5 | Visibility | Enum | - | Kategorijas redzamība |
| 6 | Updated\_At | Date | - | Kad kategorija tika atjaunota |
| 7 | Created\_At | Date | - | Kad kategorija tika pievienota |

Tabulā “Users” glabājas informācija par reģistrētajiem lietotājiem un viņu informāciju. Informācija par lietotājiem, t.sk. lietotāja id, vārdu, uzvārdu, e-pastu, telefonu, valsti, pilsētu, ielu, pasta indeksu, saziņas veidu, paroli, vai lietotājs ir administrators.

5.5. tabula

Tabulas “**Users**” struktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr.** | **Nosaukums** | **Tips** | **Garums** | **Piezīmes** |
| 1 | ID | Auto\_Increment | - | Primārā atslēga |
| 2 | Name | Varchar | 255 | Lietotāja vārds |
| 3 | Surename | Varchar | 50 | Lietotāja uzvārds |
| 4 | Email | Varchar | 255 | Lietotāja e-pasts |
| 5 | Phone | Varchar | 12 | Lietotāja tālrunis |
| 6 | Country | Varchar | 50 | Lietotāja valsts |
| 7 | City | Varchar | 50 | Lietotāja pilsēta |
| 8 | Street | Varchar | 50 | Lietotāja adrese |
| 9 | Zip | Varchar | 7 | Lietotāja pasta indekss |
|  | Contact | Enum | - | Lietotāja saziņas veids |
|  | Password | Varchar | 255 | Lietotāja parole |
|  | Is\_Admin | Tinyint | 1 | Vai lietotājs ir administrators |
|  | Updated\_At | Date | - | Kad lietotājs tika atjaunots |
|  | Created\_At | Date | - | Kad lietotājs tika pievienots |

Tabulā “Orders” glabājas informācija par lietotāja izveidotajiem pasūtījumiem. Informācija par pasūtījumiem, t.sk. pasūtījuma id, produkti, cena, lietotāja id, statuss, piegādes veids.

5.6. tabula

Tabulas “**Orders**” struktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr.** | **Nosaukums** | **Tips** | **Garums** | **Piezīmes** |
| 1 | Order\_ID | Auto\_Increment | - | Primārā atslēga |
| 2 | Products | Longtext | - | Produkti |
| 3 | Price | Int | 255 | Pasūtījuma cena |
| 4 | User\_ID | Int | 11 | Lietotāja ID |
| 5 | Status | Enum | - | Pasūtījuma statuss |
| 6 | Delivery | Enum | - | Piegādes veids |
| 7 | Updated\_At | Date | - | Kad pasūtījums tika atjaunots |
| 8 | Created\_At | Date | - | Kad pasūtījums tika pievienots |

# **LIETOTĀJA CEĻVEDIS**

## **Sistēmas prasības aparatūrai un programmatūrai**

Sistēmas veiksmīgais izmantošanai ir divas nepieciešamas lietas:

* Tīmekļa pārlūkprogramma;
* Serveris, kur ir uzstādīta dotā sistēma.

Tīmekļa pārlūkprogramma ir programmatūra, kas ir veidota speciāli tīmekļa lapu, jeb HTML valodā rakstītu dokumentu apskatei. Tīmekļa pārlūkprogrammas ir dažādas un tām visām ir dažādas nianses, kas atšķiras, tomēr visas no tām ir brīvprogrammatūras un tas nozīme, ka bez liekām problēmām var izmēģināt visas pieejamās pārlūkprogrammas, lai saprastu kura no visām patīk labāk par pašreizējo programmatūru. Uz doto brīdi populārākās tīmekļa pārlūkprogrammas ir Google Chrome, Safari, Mozilla Firefox, Microsoft Edge, Opera. Ir pievērsta uzmanība, lai tīmekļa lapa uz visām pārlūkprogrammu jaunākajām versijām izskatītos un funkcionētu korekti. Zemāk ir attēlots tīmekļa lapas atbalsts uz populārākajām tīmekļa programmatūrām attiecīgajās operētājsistēmās.

6.1 Tabula

Tīmekļa programmatūras atbalsts attiecīgajās operētājsistēmās

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Google Chrome | Safari | Mozilla Firefox | Microsoft Edge | Opera |
| Windows | Atbalstīts | - | Atbalstīts | Atbalstīts | Atbalstīts |
| Mac OS | Atbalstīts | Atbalstīts | Atbalstīts | Atbalstīts | Atbalstīts |
| Linux | Atbalstīts | - | Atbalstīts | - | Atbalstīts |
| Android | Atbalstīts | - | Atbalstīts | Atbalstīts | Atbalstīts |
| IOS | Atbalstīts | Atbalstīts | Atbalstīts | Atbalstīts | Atbalstīts |

Nav ieteicams izmantot vecākas versijas dotajām tīmekļa programmatūrām, jo tajās ir novecojuši standarti un ir iespējams, ka daži tīmekļa lapas elementi var nepareizi attēloties.

Katrai tīmekļa pārlūkprogrammai ir noteiktas savas minimālās prasībās. Zemāk ir attēlotas minimālās prasībās, kas parāda kādas prasības tiek izvirzītas optimālai sistēmas darbībai.

6.2 Tabula

Minimālās prasības tīmekļa programmatūrām

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Google Chrome | Safari | Mozilla Firefox | Microsoft Edge | Opera |
| Windows | Windows 7 | - | Windows 7 | - | Windows 7 |
| Mac | macOS 10.11 | - | macOS 10.12 | - | macOS 10.11 |
| Linux | 64-bit Ubuntu 14.04  Debian 8  openSUSE 13.3+  Fedora Linux 24+ | - | - | - | 64-bit Ubuntu 16.04  OpenSUSE  Fedora |
| Procesors | Intel Pentium 4 | - | Intel Pentium 4 | - | Intel Pentium 4  AMD Athlon 64 |
| Operatīvā atmiņa | 128MB | - | 512MB | - | - |
| Atmiņa cietajā diskā | 100MB | - | 200MB | - | - |

Tukšie lauki apzīmē 6.2 tabulā apzīmē to, ka nav pieejama precīza un oficiāla informācija par minimālajām prasībām.

Sistēmas lietošanai, uzturēšanai un palaišanai ir nepieciešams serveris uz kura ir uzstādīta programmatūra. Sistēmas veiksmīgai darbībai ir nepieciešamas sekojošās uz servera uzstādītās programmatūras:

* PHP 7.4.11 versija:
  + Viena no svarīgākajām sistēmas daļa, jo uz tās bāzējas lielākā sistēmas funkcionalitātes daļa. Ieteicams izmantot tieši PHP 7.4.11 versiju, jo tieši tā tika izmantota sistēmas izstrādē un garantēs sistēmas korektu darbību.
* NodeJS 16.3.0:
  + Sistēmas izstrādes laikā tika izmantots NodeJS 16.3.0, tāpēc ir ieteicams izmantot tieši šo programmatūras versiju,
* Tīmekļa serveris XAMPP 3.2.4:
  + Sistēmas izstrādes laikā tika izmantots tīmekļa serveris XAMPP 3.2.4 un tāpēc ir izmantots tieši šo tīmekļa serveri.
* Visual Studio Code teksta rediģētājs:
  + Pēc manām domām, uz doto brīdi vislabākais teksta rediģētājs. Viegli saprotams interfeis ar daudz dažādiem paplašinājumiem, kas paātrina koda rakstīšanu.
* phpMyAdmin datubāzes pārvaldības rīks:
  + Ieteicams izmantot datubāzes pārvaldības rīku phpMyAdmin, jo tas padara darbības ar datubāzi ļoti vienkāršas, jo tiek izmantota lietotāja saskarne tīmeklī.

Ar programmatūru uzstādīšanu viss neveidzas, vēl ir nepieciešama fiziska vieta kur darbināt sistēmu. Ir pieejamas vairākas iespējams operētājsistēmas ziņa, sistēmas izstrādes laikā tikai izmantots Microsoft Windows, tamdēļ attēlošu minimālās prasības servera programmatūrai uz Microsoft Windows, attēlotās prasības ir pavisam nelielas, kas nozīmē, ka lielākā daļa datoru varēs bez problēmām palaist doto sistēmu.

6.3. Tabula

Minimālās prasības servera programmatūrai

|  |  |
| --- | --- |
| **Procesors** | 1 GHz procesors |
| **Operatīvā atmiņa** | 1 GB |
| **Atmiņa cietajā diskā** | 5 GB |
| **Operētājsistēma** | Windows 10 |
| **Citas ierīces** | Tastatūra, Pele, Monitors |

## **Sistēmas instalācija un palaišana**

Dotajai sistēmai ir divu etapu palaišana. Pirmais etaps ir servera puses palaišana. Šis etaps aktivizē pašu sistēmu un sagatavo serveri lietotāja pusei. Sistēmas sagatavošanai, konfigurēšana un uzstādīšana norit sekojoši:

* Pirmais solis ir sistēmas failu uzlikšana uz fiziskā servera. Failu uzlikšanu uz servera var veikt divos veidos – manuāli lejupielādēt failus un ievietot tos servera /htdocs mapē vai izmantot github, lai klonētu repositoriju.
* Otrais solis ir, jādododas uz sistēmas /backend mapi un .env failā jāuzstāda datubāzes parametri, DB\_HOST – datubāzes ip adrese, DB\_DATABASE – datubāzes nosaukums, DB\_USERNAME – datubāzes lietotājvārds, DB\_PASSWORD – datubāzes parole.
* Nākamais solis ir .sql faila importēšana datubāzē, lai izveidotu visas nepiesiešamās sistēmas datubāzes tabulas.
* Pēdējais solis, lai palaistu sistēmu, ja tiek izmantots Visual Studio Code vienā no termināļiem ir jāievada sekojošās komandas:
  + cd backend
  + php artisan serve

Jātver jauns terminālis un jāievada sekojošās komandas:

* cd frontend
* npm run serve

# **Programmas apraksts**

Lietotājam apmeklējot tīmekļa lapu ir redzama navigācijas josla un galvenais lapas saturs.

**Navigācijas josla**

Navigācijas josla atrodas lapas augšpusē un ir visa lapas platumā. Tajā ir saites uz tīmekļa lapas sākumlapu, kategorijām, grozu, lietotāja paneli, pasūtījumiem. Sākumlapas saite ved uz Home.vue failu. Kategorijas saite ved uz pieejamajām kategorijām, kuras attiecīgi var apskatīt, apskatīt to apakškategoriju un kādas preces ir pieejams dotajā kategorijā. Kā arī iespējams apmeklēt grozu, kurā var apskatīt lietotāja pievienotās preces, izmanīt preču daudzumu, dzēst preces no groza, kā arī noformēt pasūtījumu. Tālāk ir saite uz lietotāja paneli, kurā var apskatīt un rediģēt lietotāja personīgo informāciju, ja lietotājs ir administrators, tad ir iespēja pievienots preces un kategorijas. Nākamā saite ir pasūtījumi, kur var apskatīt lietotāja pasūtījumu vēsturi un to statusus. Pēdējais elements navigācijas joslā ir meklēšana, kurā var meklēt preces un kategorijas, rezultāti ir saites uz respektīvajām precēm vai kategorijām.



6.1. att. Tīmekļa lapas navigācijas josla

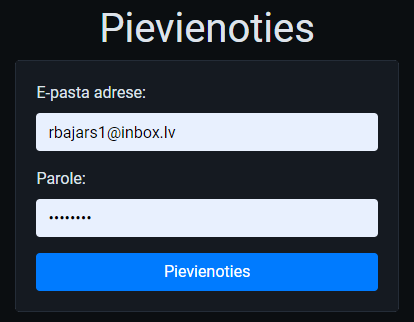
**Reģistrācijas forma**

Reģistrācijas formā ir vairāki ievadlauki, kurus nepieciešams aizpildīt, lai veiksmīgi reģistrētu lietotāja profilu. Zem visiem ievadlaukiem ir poga – reģistrēties. Pēc pogas nospiešanas tiek pārbaudīti ievadītie lauki. Ja nav aizpildīts kāds no laukiem, vai ir nepareizi aizpildīts, tiek atgriezts kļūdas paziņojums un lietotājs tiek informēts par to, ka ir jāaizpilda vai jāizlabo kāds no laukiem. Ja visi lauki ir pareizi aizpildīti tad lietotājs tiek veiksmīgi reģistrēts.

6.2. att. Tīmekļa lapas reģistrācijas forma

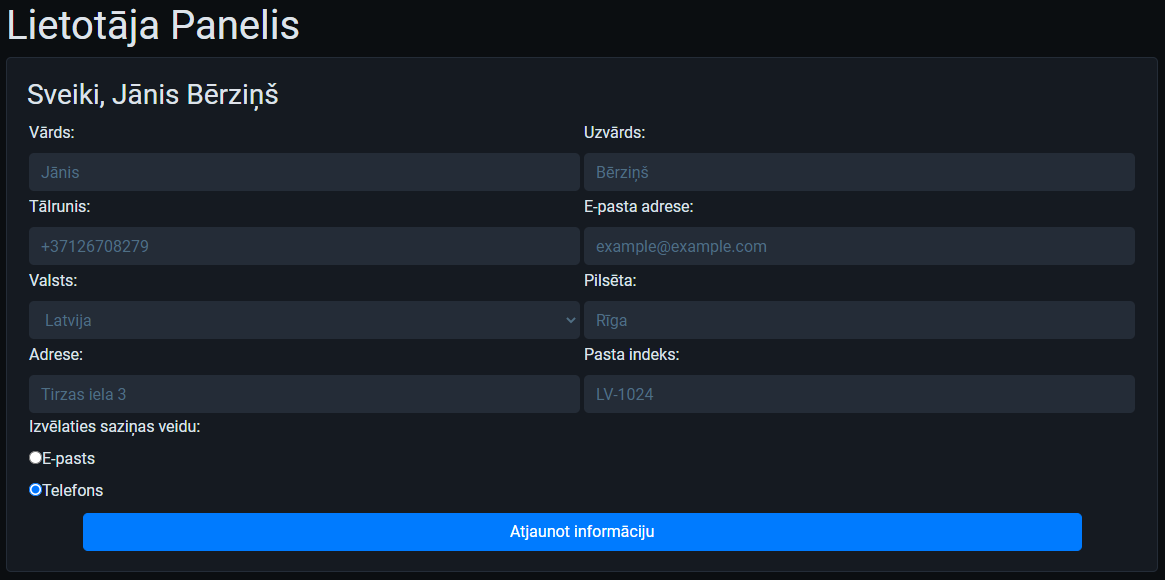
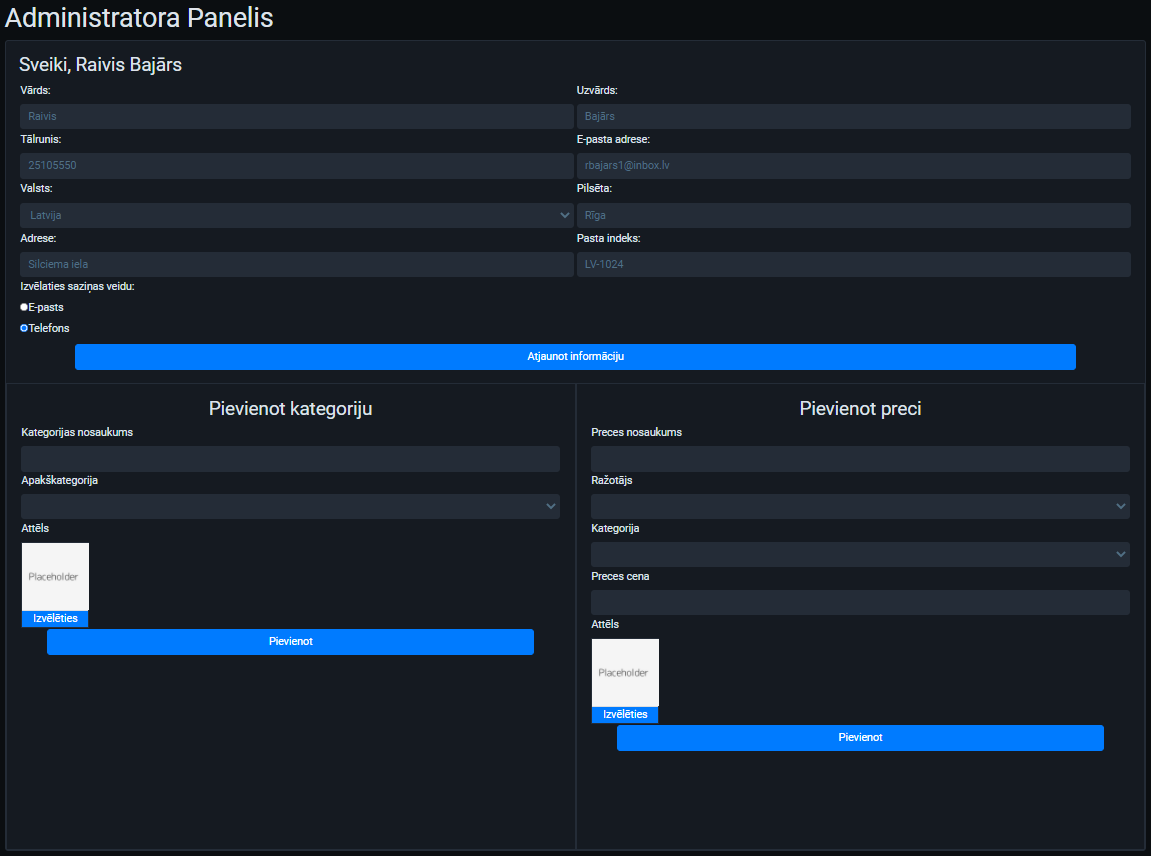
**Pievienošanās forma**

Pievienošanās formā ir divi ievadlauki, kurus nepieciešams aizpildīt, lai veiksmīgi pievienotos sistēmai. Zem ievadlaukiem ir atroda poga – pievienoties. Pēc pogas nospiešanas tiek pārbaudīti ievadītie lauki. Ja nav aizpildīts kāds no laukiem, vai ir nepareizi aizpildīts, tiek atgriezts kļūdas paziņojums un lietotājs tiek informēts par to, ka ir jāaizpilda vai jāizlabo kāds no laukiem. Ja visi lauki ir pareizi aizpildīti un lietotājs eksistē datuibāzē tad lietotājs tiek veiksmīgi pievienots sistēmā.

* 1. att. Tīmekļa lapas pievienošanās forma

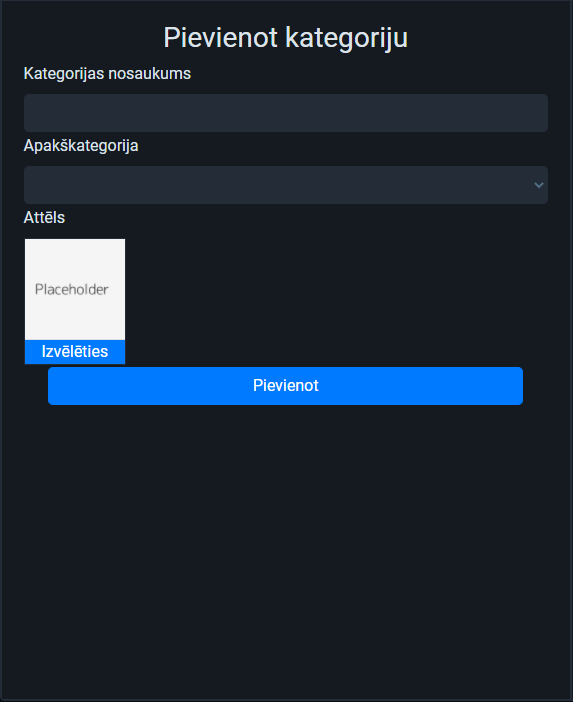
**Lietotāja panelis**

Lietotāja panelis sastāv no vairākiem ievadlaukiem, kuri sastāv no lietotāja personīgās informācijas un vēlamās saziņas veidi. Zem ievadlaukiem atrodas poga – Atjaunot informāciju, kura atjauno lietotāja informāciju datubāzē. Ja nav aizpildīts kāds no laikiem, vai ir aizpildīts nepareizi, tiek atgriezts kļūdas paziņojums un lietotājs tiek informēts par to, ka ir jāaizpilda vai jāizlabo kāds no laukiem. Ja visi lauki ir pareizi aizpildīti tad šī informācija tiek atjaunota lietotāju datubāzē. Šī lietotāja informācija tiek arī izmantota pasūtījumu noformēšanas laikā. Ja lietotājam ir sistēmas administratora status, tad lietotāja panelī parādās vēl divas formas – kategoriju pievienošanas forma un preču pievienošanas forma.

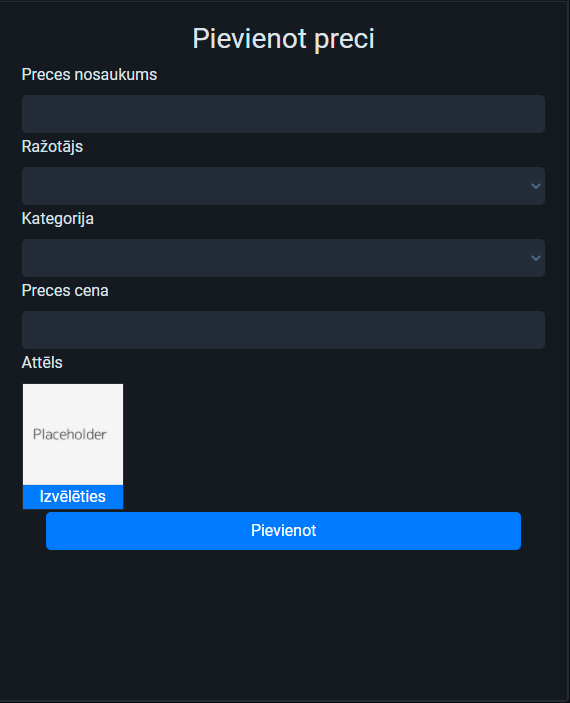
* 1. att. Tīmekļa lapas lietotāja panelis
  2. att. Tīmekļa lapas administratora panelis

**Kategoriju pievienošanas forma**

Kategoriju pievienošanas forma atrodas lietotāja panelī, ja lietotājam ir sistēmas administratora status. Forma sastāv no vairākiem ievadlaukiem kuri jāaizpilda, lai veiksmīgi pievienotu jaunu kategoriju. Kategorija sastāv no nosaukuma, apakškategorijas un kategorijas attēla. Ja nav izvēlēts attēls tad tiek ievietots vietturis attēla vietā. Zem visiem ievadlaukiem atrodas poga – pievienot. Ja kāds no laukiem nav aizpildīts vai ir nepareizi aizpildīts tiek izvadīts kļūdas paziņojums un lietotājs tiek informēts par to, ka ir jāaizpilda vai jāizlabo kāds no ievadlaukiem. Ja visi lauki ir pareizi aizpildīti, tad kategorija tiek veiksmīgi pievienota datubāzē un attēlojas kategoriju skatā.

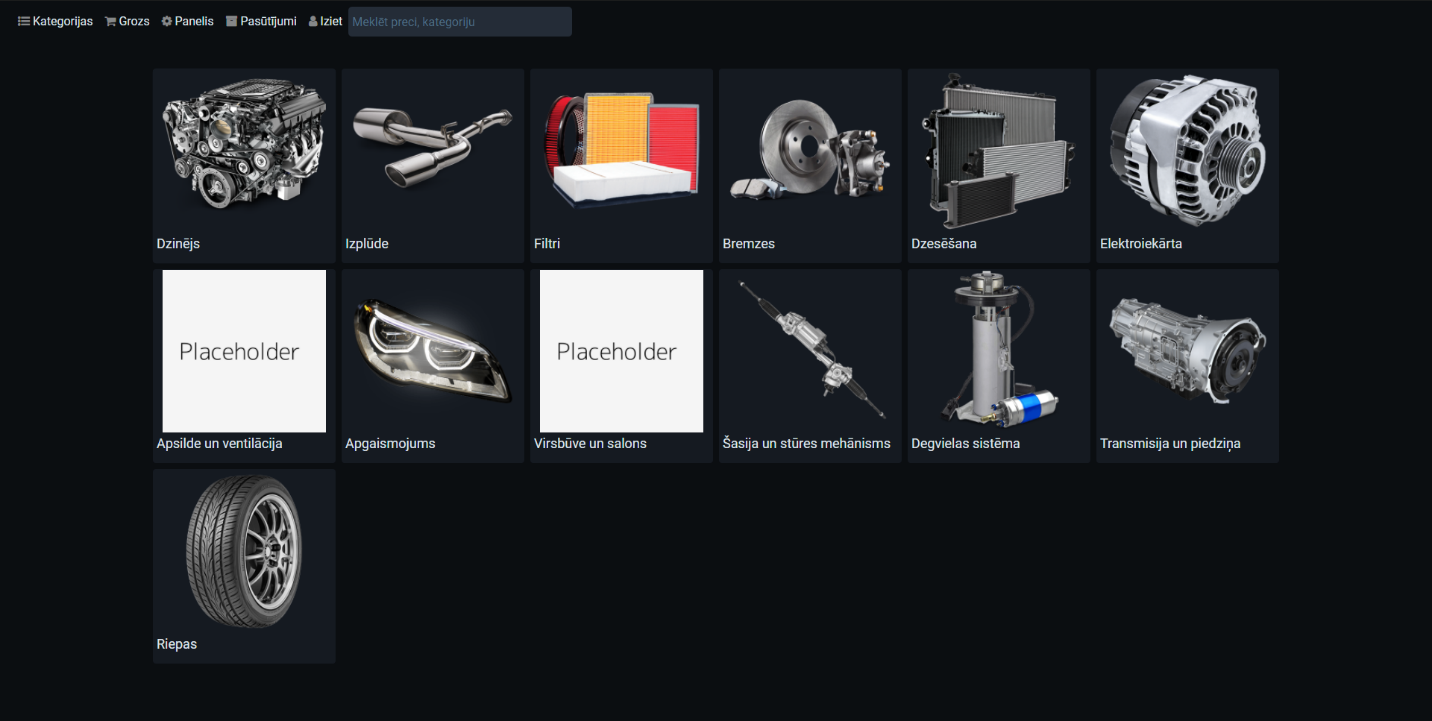
* 1. att. Tīmekļa lapas kategoriju pievienošanas forma

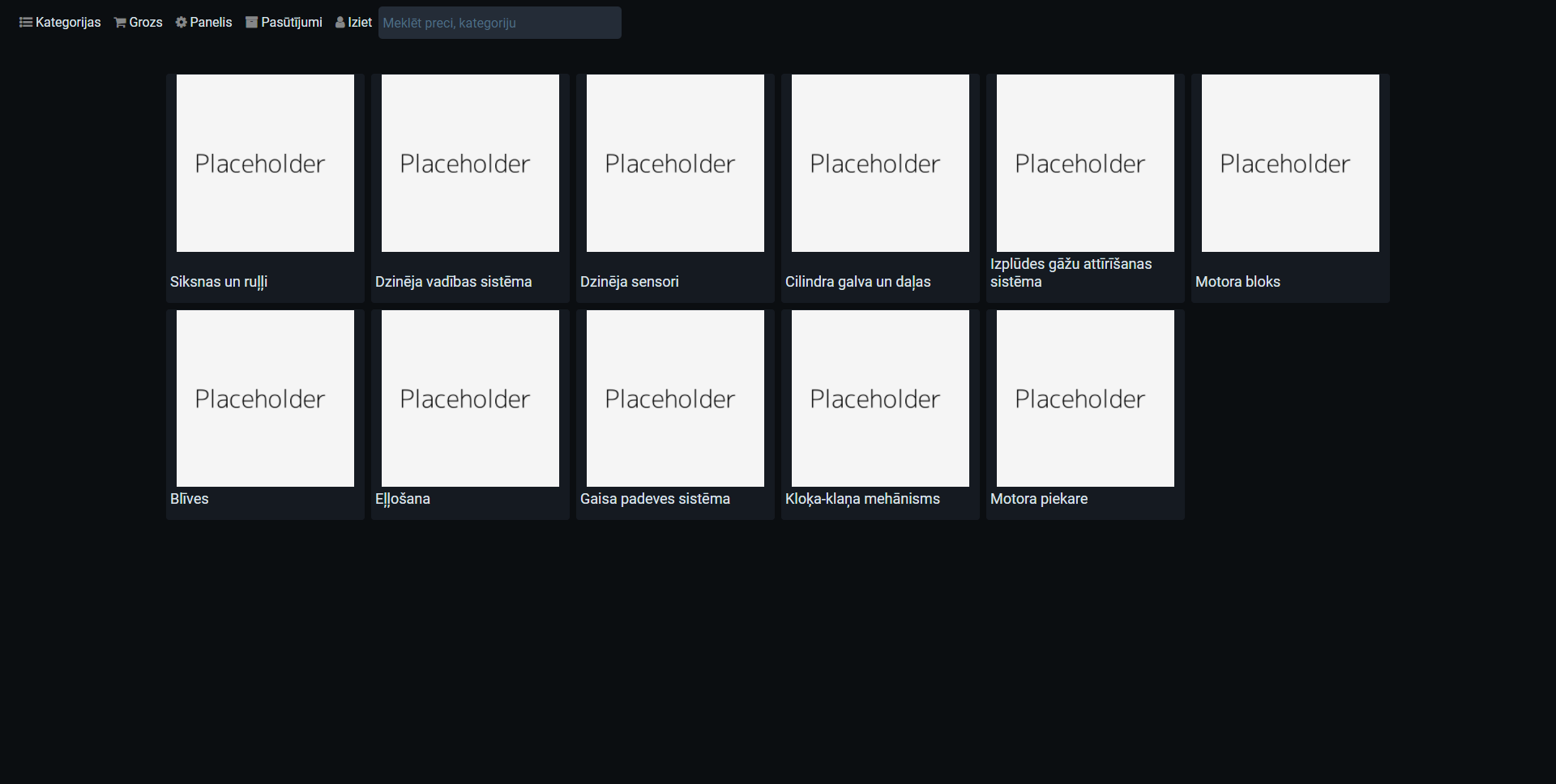
**Preču pievienošanas forma**

Preču pievienošanas forma atrodas lietotāja panelī, ja lietotājam ir sistēmas administratora status. Forma sastāv no vairākiem ievadlaukiem kuri jāaizpilda, lai veiksmīgi pievienotu jaunu preci. Prece sastāv no nosaukuma, ražotāja, kategorijas un preces attēla. Ja nav izvēlēts attēls tad tiek ievietots vietturis attēla vietā. Zem visiem ievadlaukiem atrodas poga – pievienot. Ja kāds no laukiem nav aizpildīts vai ir nepareizi aizpildīts tiek izvadīts kļūdas paziņojums un lietotājs tiek informēts par to, ka ir jāaizpilda vai jāizlabo kāds no ievadlaukiem. Ja visi lauki ir pareizi aizpildīti, tad prece tiek veiksmīgi pievienota datubāzē un attēlojas zem attiecīgās preces kategorijas.

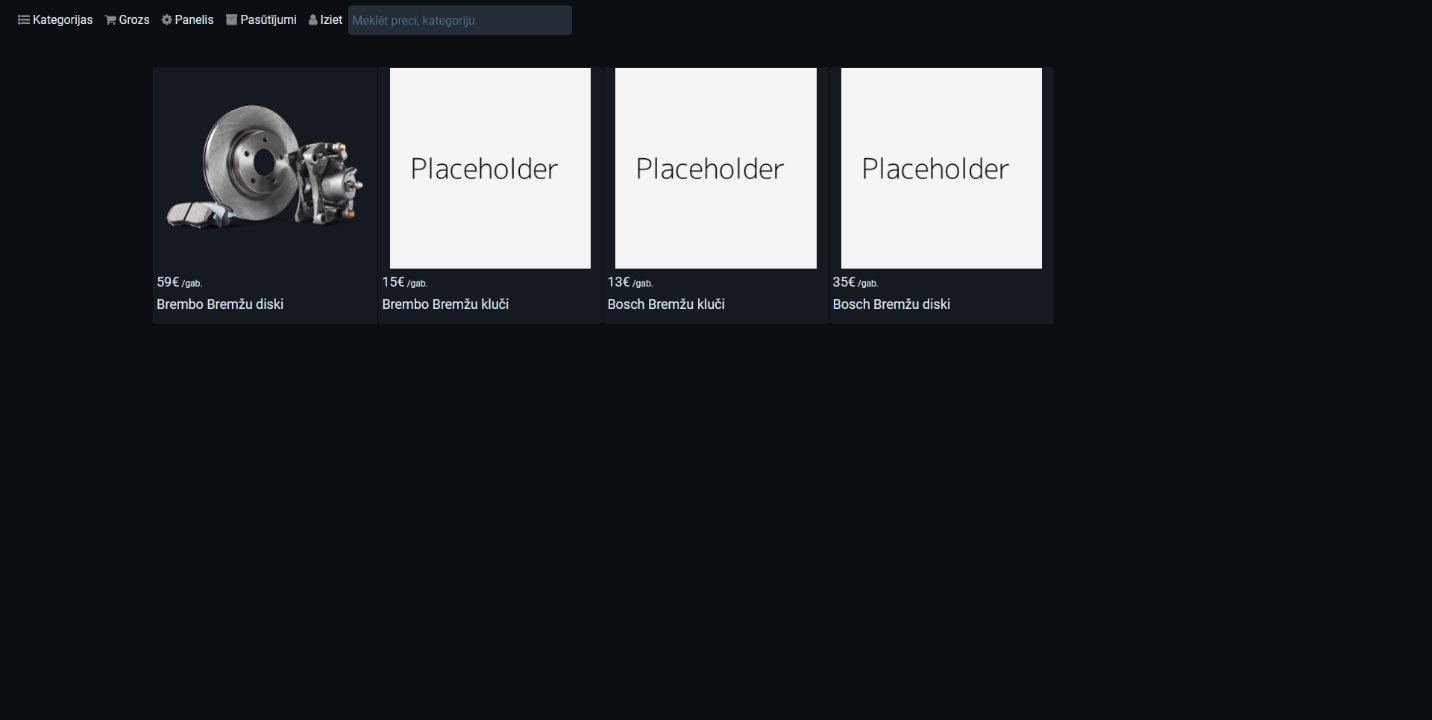
* 1. att. Tīmekļa lapas preču pievienošanas forma

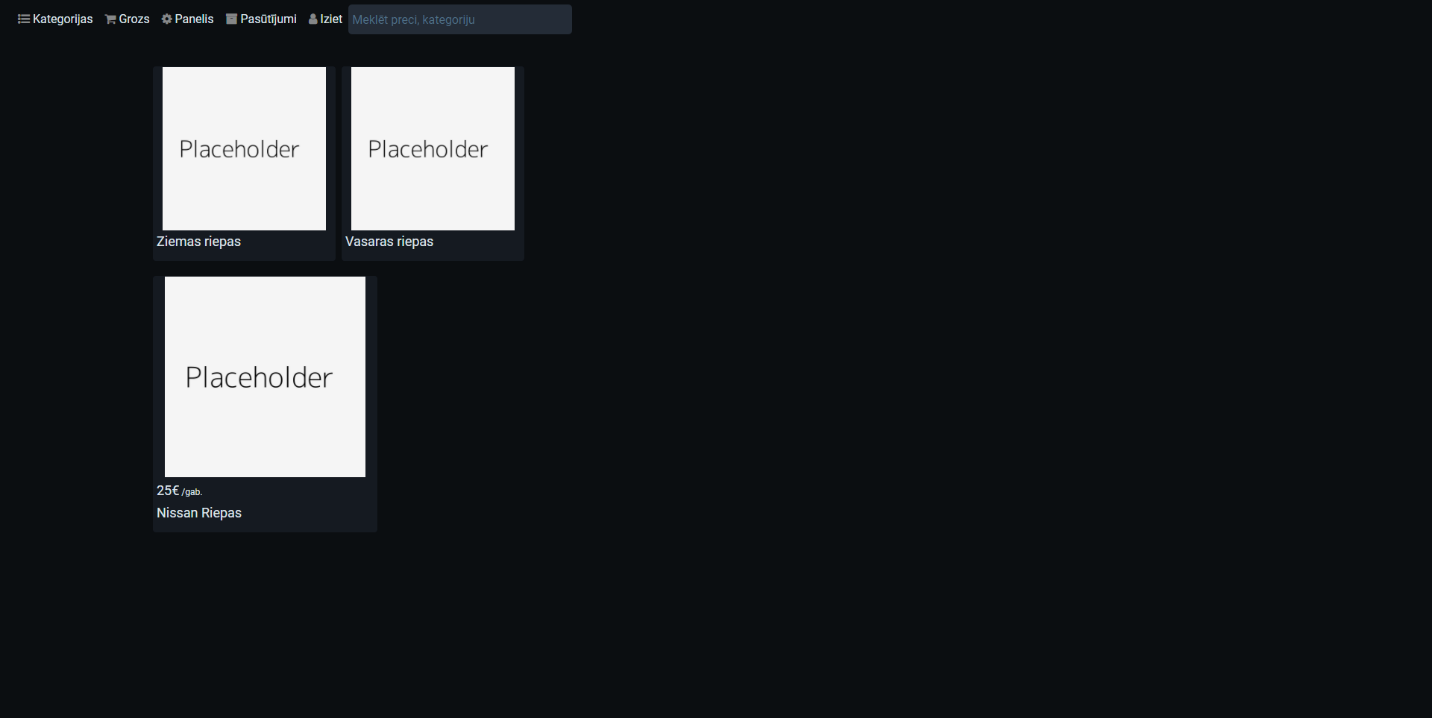
**Kateogriju skats**

Kategoriju skats parāda visas datubāzē pievienotās kategorijas, dotais skaits rādās pat ja lietotājs ir viesis. Atverot kategoriju skatu parādās visas galvenās kategorijas un un uzspiežot uz kādu no dotajām kategorijām atveras attiecīgās kategorijas apakškategorijas.

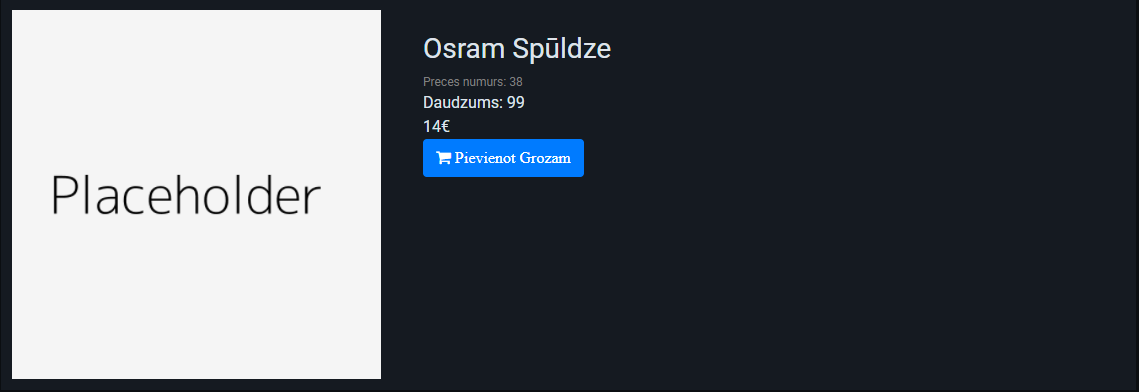
* 1. att. Tīmekļa lapas kategoriju skats
  2. att. Tīmekļa lapas “Dzinējs” kategorijas apakškategorijas

**Preču skats**

Preču skats attēlo datubāzē pievienotās preces. Preces tiek attēlotas izvēloties attiecīgo kategoriju vai apakškategoriju. Preces var arī tikt vienlaicīgi attēlotas kopā ar apakškategorijām.

* 1. att. Tīmekļa lapas “Bremzes” kategorijas un “Bremžu diski un kluči” apakškategorijas preces
  2. att. Tīmekļa lapas piemērs, kur prece vienlaikus atrodas kopā vienā skatā ar apakškategorijām

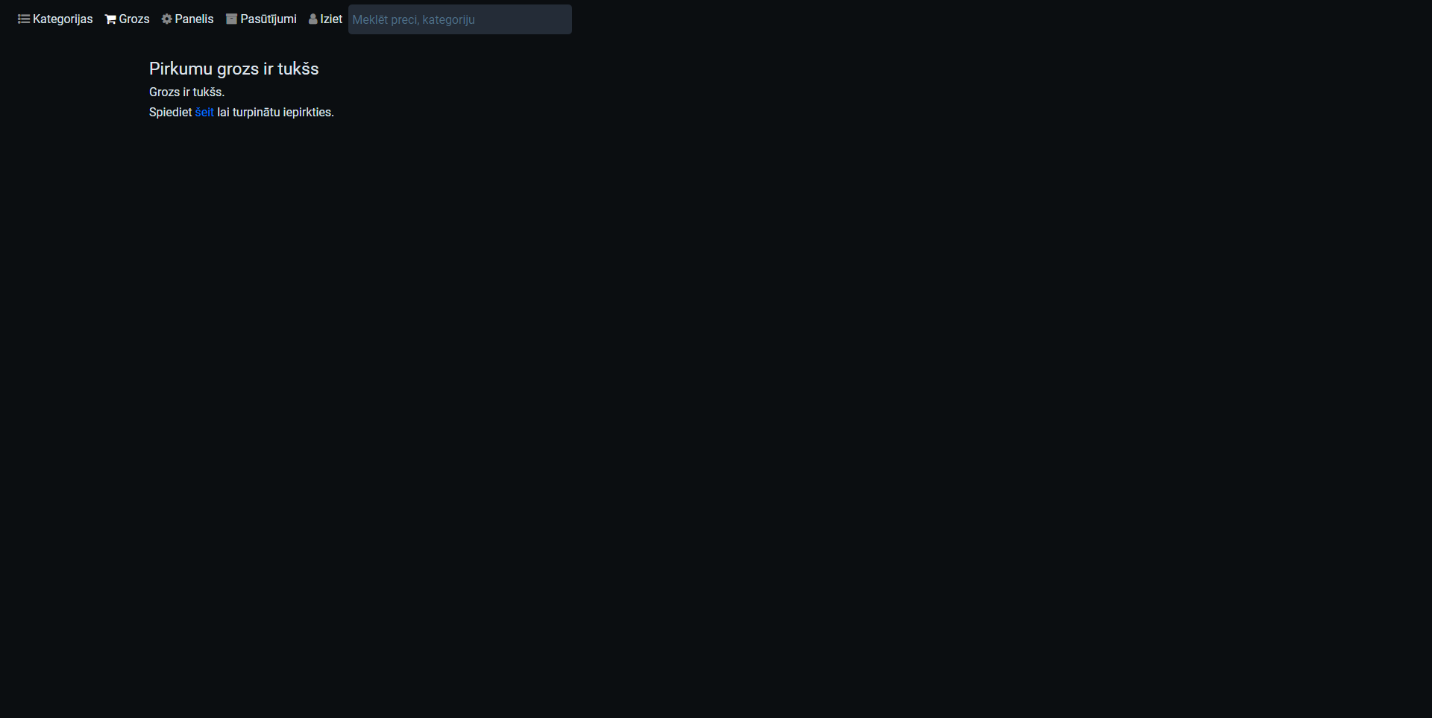
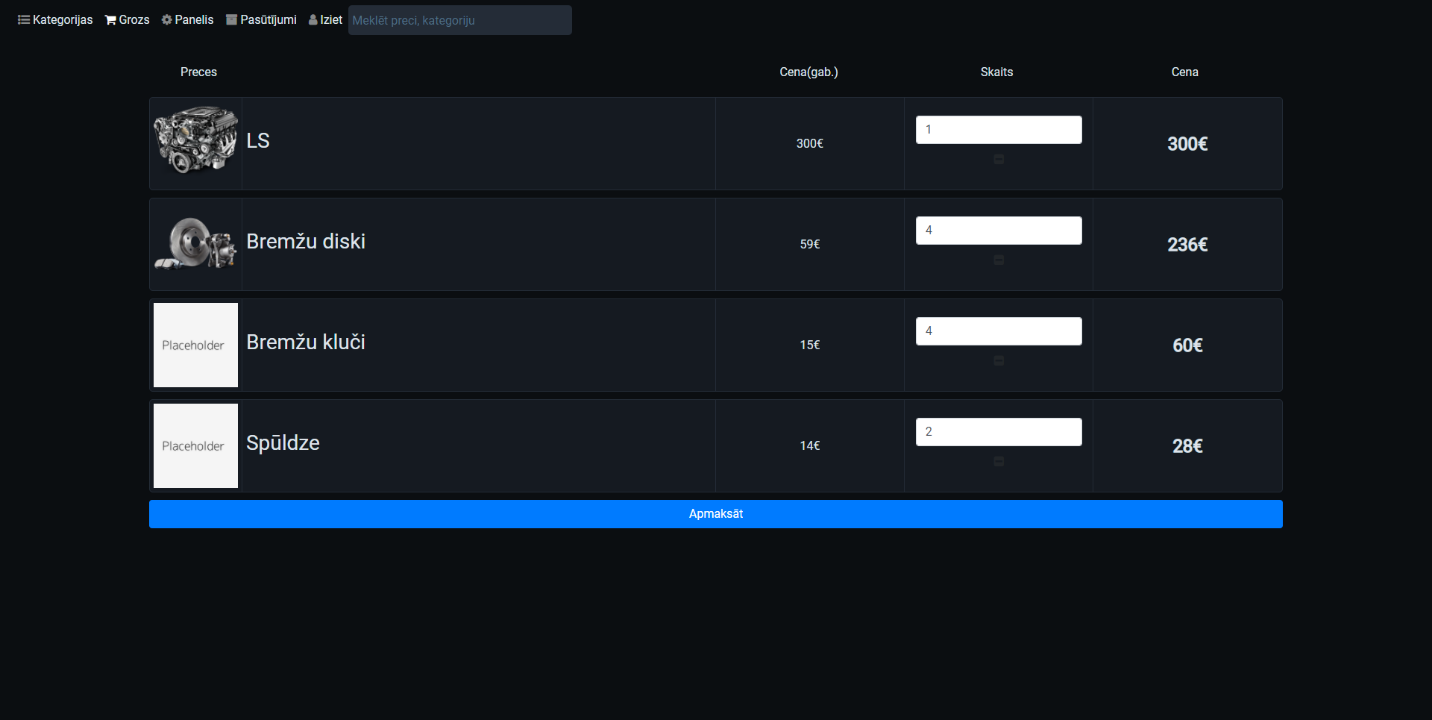
**Preces skats**

Preces skats atspoguļo izvēlēto preci no piedāvāto preču saraksta. Preces skatā iespējams apskatīt informāciju par attiecīgi izvēlēto preci, informācija satur – preces nosaukumu, preces numuru, daudzumu noliktavā, cenu un preces attēlu. Zem visas informācijas atrodas poga – Pievienot grozam. Nospiežot pogu prece tiek pievienota grozam, kur pēctam var to dzēst, mainīt daudzumu vai turpināt ar pasūtījuma noformēšanu.

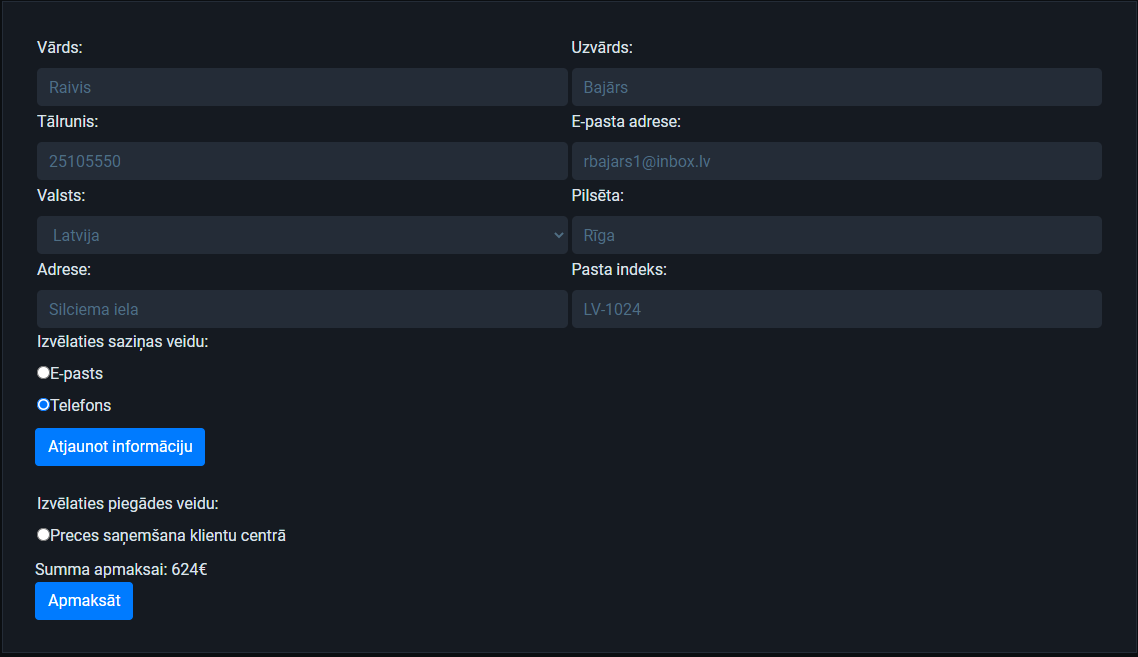
* 1. att. Tīmekļa lapas preces skata piemērs

**Groza skats**

Groza skatā ir iespējams apskatīt lietotāja grozā pievienotās preces. Grozā ir iespējams rediģēt pievienotās preces daudzumu, dzēst preci no groza vai turpināt ar pasūtījuma noformēšanu. Ja grozā nav pievienotas nevienas preces tad tiek parādīts paziņojums, ka grozs ir tukšs. Ja grozā ir pievienotas preces, tad zem tām tiek attēlo poga – apmaksāt, kura atver pasūtījuma noformēšanas formu.

* 1. att. Tīmekļa lapas groza skats, kad grozs ir tukšs
  2. att. Tīmekļa lapas groza skats, kad tajā ir pievienotas preces

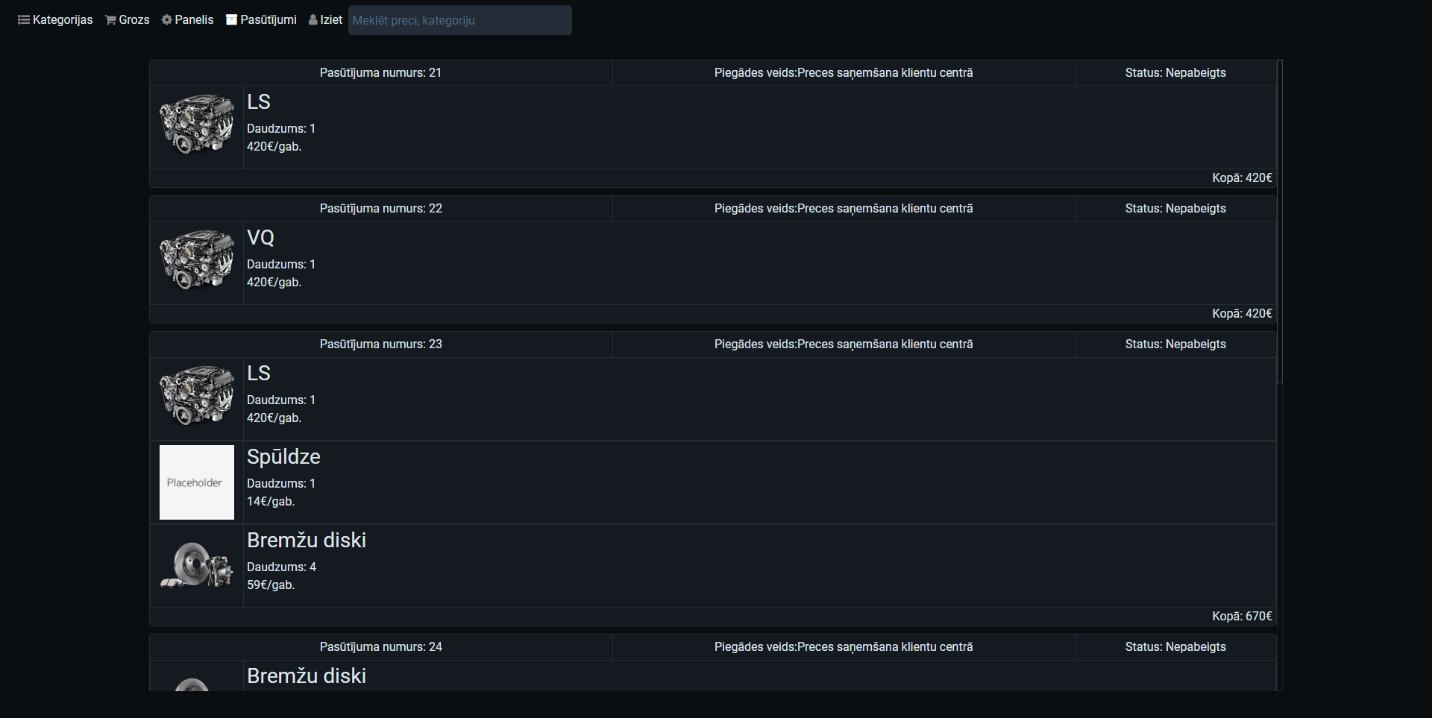
**Pasūtījumu noformēšanas forma**

Pasūtījumu noformēšanas forma attēlojas pēc groza skatā uzspiestās pogas – apmaksāt. Formā atrodas tā pati informācija, kas ir ievadīta lietotāja panelī, bet to var rediģēt pie pasūtījuma noformēšanas. Personīgajai informācijai klāt vēl nāk piegādes veids, uz doto brīdi ir pieejams tikai viens piegādes veids – preces saņemšana klientu centrā. Zem visas šīs informācijas atrodas poga – Apmasāt, pēc pogas nospiešanas, ja visi nepieciešamie lauki ir aizpildīti un ir korekti, pasūtījums tiek pievienots pasūtījumu datubāzē un tiks attēlots pasūtījumu skatā.

* 1. att. Tīmekļa lapas pasūtījumu noformēšanas forma

**Pasūtījumu skats**

Pasūtījumu skats atspogoļo visus lietotāju veiktos pasūtījumus. Pieejama informācija par – pasūtījuma numuru, piegādes veidu, statusu, preču attēliem, preču nosaukumiem, preču daudzumiem, preču cenām un arī kopējo apmaksāto summu.

* 1. att. Tīmekļa lapas pasūtījumu skata piemērs

# **NOBEIGUMS**

Kvalifikācijas darba izvirzītais mērķis ir nelīdz galam sasniegts. Ir izveidota tīmekļa lapa, kurā var apskatīt un iegādāties preces, bet daļa no funkcijām, kuras bija ieplānotas tā arī nesanāca izveidot.

Darba sākumā bija ieplānots izveidot viegli un ērti lietojamu auto rezerves internetveikalu, kur var viegli pēc automašīnas izvēlēs filtrēt preces un kategorijas, bet izveidot šo funkcionālu tā arī neizdevās.

Izstrādājot darbu secināju, ka nevienmēr pēc izskata viegls uzdevums patiesībā tāds ir, saskāros ar to, ka jau izstrādes laikā sapratu, to, ka vēlamo funkcionālu tīmekļa lapai nevarēšu pasniegt, jo nebija viss līdz galam pareizi izplānots un tika nepareizi izdalīts laiks uz sistēmas funkcionāla izstrādi.

# **INFORMĀCIJAS AVOTI**

* Bootstrap 4 ietvara pamācība - <https://www.w3schools.com/bootstrap4/>
* Composer dokumentācija - <https://getcomposer.org/doc/>
* FontAwesome ikonas - <https://fontawesome.com/>
* Diagrammu veidošanas rīks - <https://miro.com/>
* Laravel ietvara dokumentācija - <https://laravel.com/docs/7.x>
* Vue JS ietvara dokumentācija - <https://vuejs.org/v2/guide/>