Profesionālās izglītības kompetences centrs

“Rīgas Valsts tehnikums”

Izglītības programma: Programmēšana

KVALIFIKĀCIJAS DARBS

Interneta veikala datu uzskaites automatizētā sistēma

Paskaidrojošais raksts: 75 lpp

Audzēknis: Deniss Štrombergs

Vadītājs: Igors Litvjakovs

Normu kontrole: Olga Sabanska

Rīga

2021

ANOTĀCIJA

Kvalifikācijas darbā ir aprakstīta interneta veikala automatizētas sistēmas izstrādāšanas process. Sistēma dod iespēju tā lietotājam, apkopot un apstrādāt datus par sava veikala produktiem un pircējiem, konkrēti, var veidot produktus, kuri atspoguļo reālus fiziskus produktus, norādīt produkta atribūtus (cena, krāsa u.c.), kā arī, var redzēt kurus produktus iegādājās pircēji, un kādas atsauksmes tie atstāj. Sistēma tika izstrādāta ar JavaScript programmēšanas valodu, izmantojot ”React” ietvaru, lai realizētu klienta daļu, un ”Node” ar ”PostgreSQL”, “GraphQL” un ”Express”, lai realizētu servera daļu.

Kvalifikācijas darbs ietver ievadu, uzdevumu nostādni, prasību specifikāciju, uzdevuma risināšanas līdzekļu izvēles pamatojumu, programmatūras produkta modelēšanas un projektēšanas aprakstu, datu struktūru aprakstu, lietotāju ceļvedi, nobeigumu un pielikumus.

Kvalifikācijas darba ievadā ir aprakstīta aktuālā problēma un tās risinājumus. Uzdevumu nostādnē ir norādīti uzdevumi, kurus sistēmai būs nepieciešams veikt. Prasību specifikācija sastāv no ieejas un izejas informācijas, un arī no sistēmas funkcionālajām un nefunkcionālajām prasībām. Uzdevuma risināšanas līdzekļu izvēles pamatojumā ir norādīti, kādi līdzekļi būs izmantoti izstrādāšanai un kādiem nolūkiem tie tiek izmantoti. Programmatūras produkta modelēšanas un projektēšanas apraksts sastāv no sistēmas struktūras modeļa, kas ietver sistēmas arhitektūru un ER modeli, un no funkcionālās sistēmas modeļa, kas satur datu plūsmu modeli. Datu struktūru aprakstā tiek parādīta datu bāzes relāciju shēma, kā arī tabulu struktūra ar aprakstu, kas ietver datu tipu un datu garumu norādīšanu. Lietotāja ceļvedī ir norādītas nepieciešamās sistēmas prasības aparatūrai un programmatūrai, sistēmas instalācija un palaišana, kā arī programmas apraksts, kas paskaidro, kā pareizi lietot sistēmu. Testa piemērā ir dots detalizēt produktu pievienošanas apraksts.

Kvalifikācijas darbs sastāv no 75 lappusēm, kurā ietilpst 66 attēli, 15 tabulas un 3 pielikumi. Pielikumi satur sadaļas: ER diagrammu, datu bāzes relāciju shēmu un sistēmas arhitektūru.

ANNOTATION

This qualification project contains the description of online store automatized system development process. The system will give opportunity to user to collect and process the data about their stores products and clients, to be more specific, it is possible to create products, which represent real material entities from the real word, specify product attributes (price, color, etc.). Also it is possible to view which products are bought and how much exactly, and see what kind of review customers write. The system is built using JavaScript with React framework to realize frontend and Node with PostgreSQL, GraphQL and Express library, to realize backend.

The introduction of the qualification project is composed out of following sections: introduction, task definition, requirement specification, used technology justification, product model and design description, data structure description, user guide, conclusion and attachments.

Task definition contains information about the required functionality the system is suppose to provide. The requirement specification will include the data that is coming into the system and the data that is leaving the system, and also information about functional and non-functional needs. The justification for used tools and technology will contain all frameworks, libraries, etc which would be used for development of the software. Software product modeling and design contains system's structural model, which in itself contains the system's architecture and ER model, and also, system's functional model, which in itself contains data flow model. In data structure description there is specified the database relation schema, and also table structure with description, which contains data types and data type length definitions. User guide section contains system requirements, information about setup and launch process, and all necessary information for systems user. In test example, in details there is describe the creation of the product.

Overall qualification workload is 75 pages., which includes 74 images, 15 tables and 3 attachments.

SATURS

IEVADS....6

1 UZDEVUMA NOSTĀDNE....7

2 PRASĪBU SPECIFIKĀCIJA....7

2.1 Ieejas un izejas informācijas apraksts....7

2.1.1 Ieejas informācijas apraksts....7

2.1.2 Izejas informācijas apraksts....10

2.2 Funkcionālās prasības....11

2.3 Nefunkcionālas prasības....13

3 RISINĀŠANAS LĪDZEKĻU IZVĒLES PAMATOJUMS....14

4 PRODUKTA MODELĒŠANA UN PROJEKTĒŠANA....15

4.1 Sistēmas struktūras modelis....15

4.1.1 Sistēmas arhitektūra....15

4.1.2 Sistēmas ER modelis....16

4.2 Funkcionālais sistēmas modelis....18

4.2.1 Datu plūsmu modelis....18

5 DATU STRUKTŪRAS APRAKSTS....31

6 LIETOTĀJA CEĻVEDIS....40

6.1 Sistēmas prasības aparatūrai un programmatūrai....40

6.2 Sistēmas instalācija un palaišana....42

6.3 Programmas apraksts....42

6.4 Testa piemērs....66

NOBEIGUMS....72

INFORMĀCIJAS AVOTI....73

PIELIKUMI....74

ER diagramma....74

Fiziska struktūra....75

Sistēmas arhitektūra....76

IEVADS

Kvalifikācijas darba mērķis ir izveidot interneta veikala sistēmu, līdzīgi jau eksistējošiem variantiem, kā Magento 2, Shopify, BigCommerce un citiem, lai varētu nodrošināt varētu apkopot sava veikala datus vienā sistēma un, lai paplašinātu pircēju loku un pieejamību balstoties uz to, kā veikals ir arī pieejams internetā, kas joprojām ir ļoti aktuāli.

Protams nav jēgās būvēt sistēmu, kura spēs kļūt par konkurent ar iepriekš minētām, jau eksistējošām sistēmā, tās sistēmas ir ļoti lielas un sarežģītas, it īpaši Magento 2. Bet tas ar ko šī interneta veikala sistēma atšķirsies ir ar to, ka tā būs vienkāršāka, un tā kalpos kā pamats jau kaut kādai konkrētai interneta sistēmas būvēšanas. Šīs sistēmas nodrošina visas nepieciešamas bāzes funkcionalitātes, kuras varētu būt nepieciešamas, taču dažādas integrācijas ir neobligātas un nav pēc nosacījuma implementētas, piemēram, “Payment Gateway”, vai “Shipping API”, nebūs pieejami šajā programmā, vismaz pagaidām, tām vajag būt implementētam, bāzējoties uz konkrēta gadījuma nepieciešamībām.

Īsumā, šī sistēma nodrošina pamata funkcionalitāti, taču specifiskām lietām, jau jābūt implementētam balstoties uz konkrētu gadījumu.

1 UZDEVUMA NOSTĀDNE

Kvalifikācijas darba uzdevums ir izveidot interneta veikala sistēmu. Sistēma sastāvēs no 2 galvenām daļām administrators un klients. Administrators, ir kā interneta veikala īpašnieks, tam ir pieeja visiem nepieciešamiem datiem un tas var tos manipulēt un pārskatīt. Klients, ir kā pircējs, tam ir pieeja veikala priekšpusei, tas reģistrēties, likt produktus grozā, un veikt pasūtījumus.

Interneta veikala sistēmai ir jānodrošina sekojošās funkcionalitātes:

1. Viesis
   - Veikala apskatīšana
   - Produktu meklēšana un apskatīšana
   - Reģistrēšanās
2. Klients:
   - Produktu meklēšana
   - Produkta lapas apskatīšana un iespēja pievienot grozā
   - Atsauksmju sistēma
   - Pircēja akaunta veidošana un rediģēšana
   - Adrešu saglabāšana
   - Pasūtījumu pārskats
3. Administrators
   - Dažādu datu (produkti, kategorijas, lapas, atribūti, u.c.) pārvalde
   - Kopējās veikala konfigurācijas uzstādīšana:

2 PRASĪBU SPECIFIKĀCIJA

2.1 Ieejas un izejas informācijas apraksts

2.1.1 Ieejas informācijas apraksts

- Adrese, ko ievada gan administrators, gan klients, kas sastāv no:

    - Vārds – obligāts teksts ar izmēru līdz 255 rakstzīmēm
    - Uzvārds – obligāts teksts ar izmēru līdz 255 rakstzīmēm
    - Telefona numurs – obligāts teksts ar izmēru līdz 32 rakstzīmēm
    - Valsts – obligāts teksts ar izmēru līdz 255 rakstzīmēm
    - Pilsēta – obligāts teksts ar izmēru līdz 255 rakstzīmēm
    - Reģions – obligāts teksts ar izmēru līdz 255 rakstzīmēm
    - Pirmā ielas līnija – obligāts teksts ar izmēru līdz 255 rakstzīmēm
    - Otrā ielas līnija – neobligāts teksts ar izmēru līdz 255 rakstzīmēm
    - Pasta kods – obligāts teksts ar izmēru līdz 16 rakstzīmēm
- Atribūts, ko ievada administrators, kas sastāv no:
    - Unikāls kods – obligāts, unikāls teksts ar izmēru līdz 255 rakstzīmēm
    - Vai ir ieslēgts – neobligāta Bula vērtība
    - Nosaukums – obligāts, teksts ar izmēru līdz 255 rakstzīmēm
    - Datu tips – obligāts skaitītājs ar 5 iespējamiem variantiem (“boolean”, “number”, “string”, “select_number”, “select_string”)
    - Vai tiek izmantots, kā filtrs meklējot produktu – neobligāta Bula vērtība
    - Atribūta opcijas – neobligāts, atkarīgs, no datu tipa lauks ar “JSONB” datu tipu, kurā var ievadīt noteiktas atribūtu opcijas
- Atribūtu sets, ko ievada administrators, kas sastāv no:
    - Unikāls kods - obligāts, unikāls teksts ar izmēru līdz 255 rakstzīmēm
    - Vai ir ieslēgts – neobligāta Bula vērtība
    - Nosaukums - obligāts teksts ar izmēru līdz 255 rakstzīmēm
- Kategorija, ko ievada administrators, kas sastāv no:
    - Unikāls lapas identifikators - obligāts, unikāls teksts ar izmēru līdz 255 rakstzīmēm
    - Vai ir ieslēgta - neobligāta Bula vērtība
    - Nosaukums - obligāts teksts ar izmēru līdz 255 rakstzīmēm

- Vai tiek pievienota pie izvēlnes - neobligāta Bula vērtība

- Pircējs, ko var ievadīt gan administrators, gan klients reģistrējoties, kas sastāv no:
  - Unikāls e-pasts - obligāts, unikāls teksts ar izmēru līdz 255 rakstzīmēm
  - Parole - obligāts teksts ar izmēru līdz 255 rakstzīmēm
  - Vārds - obligāts teksts ar izmēru līdz 255 rakstzīmēm
  - Uzvārds - obligāts teksts ar izmēru līdz 255 rakstzīmēm
- Produkts, ko ievada administrators, kas sastāv no:
  - Unikāls lapas identifikators - obligāts, unikāls teksts ar izmēru līdz 255 rakstzīmēm
  - Krājumu glabāšanas vienība – obligāts, unikāls teksts ar izmēru līdz 255 rakstzīmēm
  - Vai ir ieslēgts - neobligāta Bula vērtība
  - Pamata cena – cena bez nodokļiem, atlaidēm un citiem pieskaitījumiem
  - Daudzums noliktavā – vesels skaitlis ar nenoteiktu diapazonu
  - Atlaides tips (daudzums, procentuāls) – neobligāts skaitītājs ar 2 iespējamām vērtībām (“percentage”, “amount”)
  - Atlaides vērtība - skaitlis ar diapazonu no 0 līdz bezgalības (precīzāk līdz datu tipa ierobežojumam)
  - Nodokļu procents – skaitlis ar diapazonu no 0 līdz bezgalības
  - Īss apraksts - neobligāts, teksts ar izmēru līdz 255 rakstzīmēm
  - Garš apraksts – neobligāts teksts ar nenoteiktu rakstzīmju daudzumu
  - Pamata attēls – neobligāta”JSONB” vērtība, kurā glabājās URL uz galveno produkta attēlu
  - Mazais attēls –neobligāta”JSONB” vērtība, kurā glabājās URL uz mazo produkta attēlu
  - Pārējie attēli –neobligāta”JSONB” vērtība, kurā glabājās URL saraksts uz pārējiem produkta attēliem

    - Atribūtu vērtības – neobligāta ”JSONB” tipa vērtība, kuras lauki sastāv no atribūtu koda un tā vērtības
- Atsauksme, ko ievada klients, kas sastāv no:
    - Status (pieņemta, apstrādā, noliegta) – skaitītājs ar 3 iespējamām vērtībām (“accepted”, “pending”, “rejected”)
    - Nosaukums - obligāts, teksts ar izmēru līdz 255 rakstzīmēm
    - Saturs - obligāts, teksts ar neierobežotu izmēru
    - Reitings no 1 līdz 10 – obligāts skaitlis ar diapazonu no 1 līdz 10

2.1.2 Izejas informācijas apraksts

Izejas dati ir aizgūtie dati, kā atbilde uz lietotāja izveidotu pieprasījumu:

- Meklēšanas rezultātu atspoguļojums, kas sastāv no produktu saraksta, un produktu elementiem, kas sastāv no:
    - Nosaukuma
    - Cenas
    - Galvenā attēla
- Pasūtījuma izveide, satur ģenerētu informāciju par pasūtījuma kopsavilkumu, kas sastāv no:
    - Pasūtījuma references, kas ir unikāla simbolu virkne, kas sastāv no 11 rakstzīmēm
    - Nodokļa cenas
    - Cenas bez nodokļa
    - Cena ar nodokli
    - Pasūtījuma priekšmetiem, kas sastāv no
        - Daudzuma
        - Nodokļa cenas
        - Cenas bez nodokļa
        - Cena ar nodokli
    - Adreses informācijas

    - Pircēja informācijas
- Rēķinu izveide, satur ģenerētu informāciju par rēķinu, kas tiek ģenerēts no pasūtījuma, tāpēc satur tādus pašus laukus, un satur referenci uz pašu pasūtījumus

2.2 Funkcionālās prasības

- Jānodrošina lietotāja autorizācija.
    - Ja kāds lauks nav aizpildīts, vai e-pasts neatbilst formātam, vai e-pasts un parole nesader kopā, sistēmai ir jāizvada paziņojums
    - Lietotājs tiek autorizēts tikai tad, kad visi lauki ir pareizi ievadīti, kad e-pasts un parole ir atrodamas datubāzē
    - Pēc autorizācijas lietotāja dati tiek ielādēti
- Jānodrošina lietotāja reģistrācija.
    - Ja kāds lauks nav aizpildīts, vai e-pasts neatbilst formātam, vai abas paroles nesakrīt, vai, ja lietotājs ar tādu e-pastu jau eksistē, tad sistēmai ir jāizvada paziņojums
    - Paroles tiek šifrēta
- Jānodrošina adreses pievienošana un rediģēšana:
    - Jā kāds lauks nav aizpildīts (izņemot novadu un otru ielas līniju), tad sistēmai ir jāizvada paziņojums, formāta pārbaudījumi uz pasta indeksu netiek veikti
    - Adrese tiek pievienota tikai, tad kad visi lauki ir pareizi ievadīti
    - Adresi var pievienot tikai klients, savā akauntā, zem adrešu nodaļas, vai veicot pasūtījumu
- Jābūt aptauju sistēmai:
    - Aptaujas veidošana, aptaujas var veidot tikai klients, administrators nevar
    - Aptaujas rediģēšana, aptaujas var rediģēt tikai klients savā akauntā, zem atsauksmju nodaļas
    - Aptaujas dzēšana, aptaujas var dzēst gan administrators, gan klients
    - Aptaujas statusa maiņa, administratoram ir iespēja mainīt aptaujas statusu, vai nu akceptēt to vai noraidīt

    - Aptaujas var eksportēt
- Jānodrošina produkta groza rediģēšana:
    - Klients var pievienot produktus grozām
    - Ir iespējams mainīt groza produkta daudzumu, vai nu palielināt to par 1 vai samazināt to par 1 vienību, ja rezultējoša vienība ir vienāda ar 0, tad produkts tiek izņemts no groza
    - Ir iespējams izņemt produktu no groza uzreiz
    - Ir iespējams pāriet uz konkrētu produktu uzspiežot uz tā groza elementus
- Jānodrošina produktu meklēšana:
    - Produktus ir iespējams meklēt balstoties uz sekojošiem datiem: cena diapazons (minimāla cena, maksimāla cena), atribūtu filtriem, kategorijas, pēc nosaukuma, un ir iespējams kārtot produktus balstoties uz dažādām metodēm
    - Produkti ir pieejami saraksta viedā, kas sastāv no lapām un lapai ir noteikts izmērs, ko ir iespējam mainīt
- Jānodrošina pasūtījuma veidošana:
    - Kad produktu grozs ir aizpildīts, lietotājs var veikt pasūtījumu, ir nepieciešams vai nu izveidot vai izvēlēties no eksistējošam adresēm.
    - Izvēlēties piegādes veidu
    - Izvēlēties apmaksas veidu
    - Akceptēt nosacījumus
    - Tikai pēc visām šim darbībām, tiks ģenerēts pasūtījums, pasūtījuma ģenerācijā tiek pārbaudīti dati, adrese un produktu pieejamības
- Jānodrošina produktu pārvalde administratoram:
    - Produktu pievienošana
    - Produktu rediģēšana
    - Produktu dzēšana
    - Jābūt iespējai piešķirt, kura atribūtu setam pieder produkts
    - Jābūt iespējai piešķirt, kurām kategorijām pieder produkts

- Jānodrošina atribūtu pārvalde administratoram:
  - Atribūtu pievienošana
  - Atribūtu rediģēšana
  - Atribūtu dzēšana
- Jānodrošina atribūtu setu pārvalde administratoram:
  - Atribūtu seta pievienošana
  - Atribūtu seta rediģēšana
  - Atribūtu seta dzēšana
  - Jābūt iespējai piešķirt atribūtu atribūtu setam
- Jānodrošina kategoriju pārvalde administratoram
  - Kategoriju pievienošana
  - Kategoriju rediģēšana
  - Kategoriju dzēšana

2.3 Nefunkcionālas prasības

- Prasības procesam:
  - Programmas priekšējai daļai jābūt izstrādātai JavaScript valodā, izmantojot React ietvaru
  - Programmas aizmugurējai daļai jābūt izstrādātai JavaScript valodā, izmantojot Express bibliotēka
  - Servera pieprasījumu rezolūcijai jānotiek, izmantojot GraphQL, pieprasījumu valodu
  - Programmas administratorā daļai, jāizmanto React Admin bibliotēku, par pamatu
  - Datubāzei jābūt izstrādātai PostgreSQL valodā, izmantojot JavaScript bibliotēku.
- Produkta prasības:

- Lietotāju saskarsnei ar sistēmu jānotiek angļu valodā, bet vēlāk sistēma var būt adaptēta citās valodās
- Programmai par pamatu jāatbalsta ekrāna izmēru, kas atbilst datora ekrāna izmēram

- Prasības lietotāja interfeisam
  - Lapas pamatkrāsai jābūt zilai. Tai jābū labi redzamai, lai tā veido labu kontrastu ar tekstu, lai lietojām nebūtu grūtības redzēt lapas saturu
  - Kļūdu paziņojumiem jābūt vienā stilā

3 RISINĀŠANAS LĪDZEKĻU IZVĒLES PAMATOJUMS

- Frontend:
  - JavaScript – ir praktiski vienīga programmēšanas valoda, kas tiek izmantota šajā projektā, un visas darbības ar HTML, SCSS, PostgreSQL tiek veiktas arī ar JavaScript. Izvēles pamatojums ir tāds, kā šī programmēšanas valoda ļoti labi der mājaslapu veidošanai, tagad, ne tikai, lai nodrošinātu lietotāja interfeisu, bet arī, lai nodrošinātu datu apstrādi un datu bāzes manipulācijas. JavasScript ir ļoti populāra un attīstīta programmēšanas valoda, ar lielu daudzumu palīg resursu, tāpēc, izvēle ir ļoti izdevīga, jo atvieglo palīdz paātrināt izstrādes procesu.
  - React – interfeisa komponentu sistēma, lai labu interfeisu, ir nepieciešama laba komponentu sistēma, protams, varēja būvēt savu komponentu sistēmu, bet tas aizņemtu pārāk daudz laika, un resursu, pietam, vēl, nevar paredzē, kādas grūtības rastos, kas tieši būtu nepieciešams, lai sistēma funkcionētu atbilstoši vajadzībām, tāpēc, ir labāk izmantot jau gatavu, pārbaudītu, sistēmu.
  - Material UI – bāzes stils, būtu labi, lai jau uzreiz būtu kaut kāds mājaslapas stils, kuru pēc tām pēc konkrēta lietotāja nepieciešamībām varētu modificēt un pielāgot tā specifiskām vajadzībām.
  - React Admin – administratorā daļas abstrakcija, nodrošina resursu sasaisti starp Backend un Frontend, konrkēti adminstrātora daļai, izvēle, ir pamatota, ar

to, ka ir nepieciešama, kaut kāda bāze, uz kuras būvēt adminstrātora daļu, jo lielāka daļa no operācijām atkārtojās, tāpēc arī šīs komponents ir ļoti lietderīgs.

- Backend:
  - Node – JavaScript bāzēta Backend sistēma, ar to palīdzību tiek nodrošināta visas servera operācijas: GraphQL API un PostgreSQL. Alternatīas izvēles varētu būt PHP, bet šajā gadījumā, lai ērtāk saskaņotu Backend ar frontend, tieši Node der ļoti labi, neskatoties uz tā trūkumiem.
  - Express – bibliotēka serverim, lai nodrošinātu dažādas funkcijas, piemēram ceļu definēšanas (routes), šī bibliotēka nodrošina komunikāciju ar klientu, un iespējo citas sistēmas daļas, piemēram GraphQL
  - PostgreSQL – datu bāzes vadības sistēma, lai nodrošinātu datu uzglabāšanu un manipulācijas. Tieši izvēlējos PostgreSQL un ne MySQL, vai citu SQL tipa datu bāzes vadības sistēmu, tāpēc, ka PostgreSQL satur specifisku funkcionalitāti, kura noderētu projekta realizācijas, konkrēti, JSONB datu tips, ar kura palīdzību var realizēt efektīvu produktu atribūtu sistēmu, protams arī izmantojot MySQL, varētu to iegūt ar EAV paņēmienu.
  - GraphQL – pieprasījumu valoda, kura nodrošina efektīvu datu pieprasījumu rezolūciju. Klients pieprasa konkrētus datus, un severis tieši tos tādus, kurus klients ir pieprasījis, ne vairāk, ne mazāk. Atšķirībā no alternatīvas – Rest API, šī sistēma ir krietni elastīgāka un datu pieprasījums tiek veikts no viena endpoint.

4 PRODUKTA MODELĒŠANA UN PROJEKTĒŠANA

4.1 Sistēmas struktūras modelis

4.1.1 Sistēmas arhitektūra

Manā sistēmā ir trīs lietotāju tipi: viesis, pircējs jeb reģistrēts lietotājs un administrator, tāpēc kopumā interneta veikala sistēma sastāvēs no 3 daļām: viesa daļa, pircēja daļa un administratora daļa, kas tiek attēlotas, kopā ar to funkcionalitātēm uz sistēmas arhitektūras diagrammas (skatīt 3. pielikumā).

Viesa daļa.

Viesis ir lietotājs, kurš nonāk uz mājaslapu pirmo reizi, un tām ir viszemākās privilēģijas, tas var apskatīt tikai veikala lapas, meklēt un apskatīt produktus, un reģistrēties. Tas nevar likt produktus grozā un rediģēt groza saturu, tas nevar veikt pirkumus, kā arī tām nav pieejas nekādai datu pārvaldei, vai nu klienta mērogā, un it īpaši administratorā mērogā.

Klienta daļa.

Klients ir viesis, kurš ir reģistrējies sistēmā, tāpēc tām tagad jau ir vairāk privilēģiju, tas var pievienot produktus grozā un rediģēt groza saturu, rediģēt savu akauntu, pievienot adreses savā adrešu sarakstā, rakstīt atsauksmes uz produktiem, kurus tas ir nopircis, pārskatīt savus pasūtījumus, kā arī veikt pašus pasūtījumus. Taču klientam nav privilēģiju mainīt produkta datus, kategorijas datus, vai citus objektu datus. Un klients nevar dabūt administratorā privilēģijas, as var redzēt un pārvaldīt tikai datus, kuri tam ir redzami un pieder.

Administratora daļa.

Administrators ir lietotājs ar visām iespējamām privilēģijām, šajā gadījumā netiek izdalītas atsevišķās privilēģijas administratoriem, un visiem administratoriem ir vienādas privilēģijas. Administratoram ir pieeja, pie gandrīz visiem datiem un to pārvaldēm, izņemot dažus jūtīgus datus, piemēram, kā lietotāju paroles, bet tās tiek šifrētas. Administrators arī nevar mainīt atsauksmes datus, tikai akceptēt vai noraidīt tās, vai arī dzēst. Bet kopumā administratoram ir visas pieejas, kas ir nepieciešamas, lai pārvaldītu visu veikala sistēmu.

4.1.2 Sistēmas ER modelis

ER diagramma (skatīt 1. pielikumā) sastāv no 12 entitijām, kas atspoguļo datu apriti sistēmā.

- “Produkts” – entitija, kas satur visus nepieciešamus datus, lai attēlotu produktu glabāšanu sistēmā
- “Atribūtu sets” - sets, kas apvieno kaut kādu atribūtu daudzumu.
- “Atribūts” – apraksta konkrētas produkta īpašības
- “Kategorijas” – grupa, kas apvieno vairākas citas kategorijas (apakškategorijas), vai/un produktus.
- “Pircējs” – klients, persona, kura meklē produktus, liek tos grozā un veic pasūtījumus.

- “Addrese” – entitija, kuru var pievienot pircējs pirkuma laikā, un kas ir nepieciešama pasūtījumam, lai nodrošinātu piegādi un apmaksu
- “Groza produkts” – ir produkts, kas tika pievienots grozām, ar papildus datiem, kā daudzumu un cenām.
- “Pasūtījums” – pasūtījums ir pircēja darbības galīgais rezultāts, tas ietver sevī visu nepieciešamo informāciju: kopējie dati: numurs, datums, cenas: cena par piegādes cena, nodokļi, kopējā cena bez nodokļiem, kopējā cena ar nodokļiem, un informāciju par visiem pasūtītiem produktiem un to daudzumu.
- “Pasūtījuma produkts” – pasūtījuma priekšmets, kas pieder noteiktam pasūtījuma, kas saglabā sevī uz to brīdi aktuālo informāciju par produktu
- “Rēķins” - rēķins, ir dokuments, kas satur visu galīgo informāciju par pasūtījuma datiem: kopējie dati: numurs, datums, cenas: cena par piegāi, nodokļi, kopējā cena bez nodokļiem, kopējā cena ar nodokļiem, un referenci uz konkrētu pasūtījumu un tā pasūtījuma produktiem.
- “Atsauksme” – viedoklis, kuru uzraksta pircējs par produktu.

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

- Starp produktiem un kategorijām ir attiecība daudzi pret daudziem, jo vienam produktam var būt vairākas kategorijas, un vienai kategorijai var būt vairāki produkti.
- Starp produktiem un atribūtu setiem ir attiecība daudzi pret vienu, jo produktam var būt tikai viens atribūtu sets, bet atribūtu setiem var būt vairāki produkti.
- Starp produktu un atsauksmi ir attiecība viens pret daudziem, jo vienam produktam var būt vairākas atsauksmes, bet vienai atsauksmei var būt tikai viens produkts.
- Starp pircēju un adresi ir attiecība viens pret daudziem, jo vienam pircējam var būt vairākas adreses, bet vienai adreses var būt tikai viens pircējs.
- Starp pircēju un produktu ir attiecība daudzi pret daudziem, jo viens pircējs var nopirkt vairākus produktus, un viens produkts var būt pārdots vairākiem pircējiem.

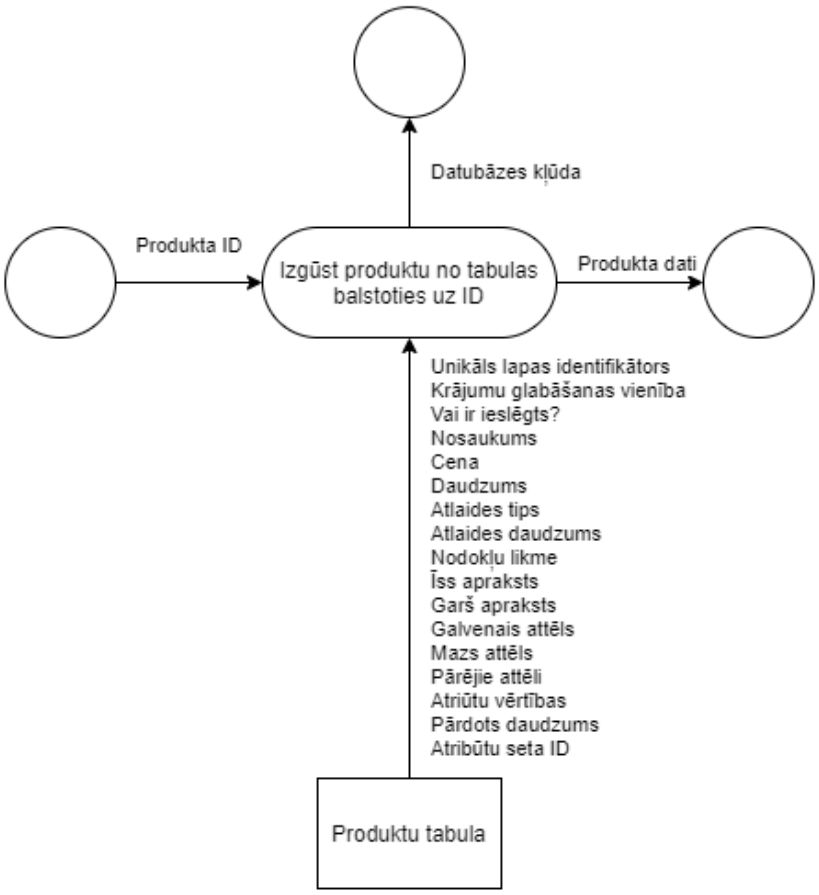
- Starp pircēju un pasūtījumu ir attiecība viens pret daudziem, jo vienam pircējam var būt vairāki pasūtījumi, bet pasūtījumam var būt tikai viens pircējs.
- Starp pircēju un atsauksmi ir attiecība viens pret daudziem, jo vienam pircējam var būt vairākas atsauksmes, bet vienai atsauksmei var būt tikai viens pircējs.
- Starp pircēju un pircēja grupu ir attiecība daudzi pret daudziem, jo vienam pircējam var būt vairākas pircēju grupas, un vienai pircēju grupai var būt vairāki pircēji.
- Starp kategoriju un kategoriju ir attiecība daudzi viens pret daudziem, jo vienai kategorijai var būt vairākas apakškategorijas, bet vienai apakškategorija, var būt tikai viena kategorija.
- Starp pasūtījumu un rēķinu ir attiecība viens pret viens, jo vienam pasūtījumam var būt tikai viens rēķins un vienam rēķinam var būt tikai viens pasūtījums.
- Starp atribūtu setu un atribūtu ir attiecība daudzi pret daudziem, jo vienam atribūtu setam var būt vairāki atribūti, un vienam atribūtam var būt vairāki atribūtu seti.

4.2 Funkcionālais sistēmas modelis

4.2.1 Datu plūsmu modelis

1 **Produkta apskatīšana**

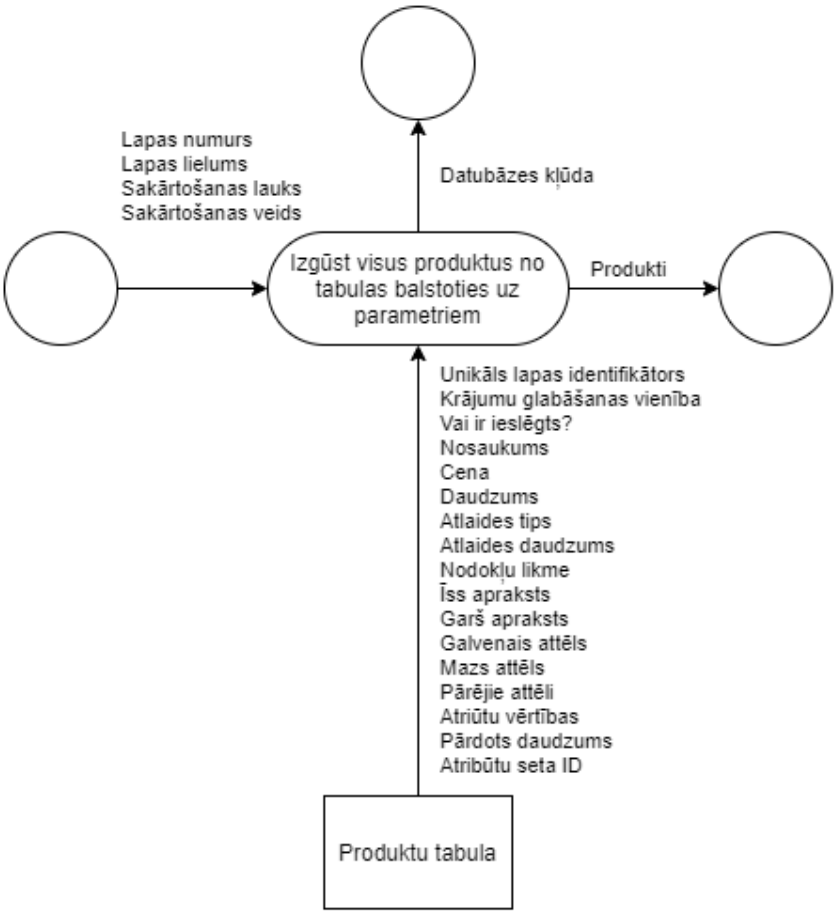
Šajā kontekstā, produkta apskatīšana notiek no administratorā puses, tikai ar administratorā pieeju, produkts tiek izgūsts no produktu tabulas balstoties uz produkta ID. Visi produkta dati tiek pieprasīti un sūtīti lietotājam, kas tos varēs izmainīt un saglabāt.



1. att: Produkta apskatīšanas datu plūsmas diagramma

2 Produktu saraksta attēlošana

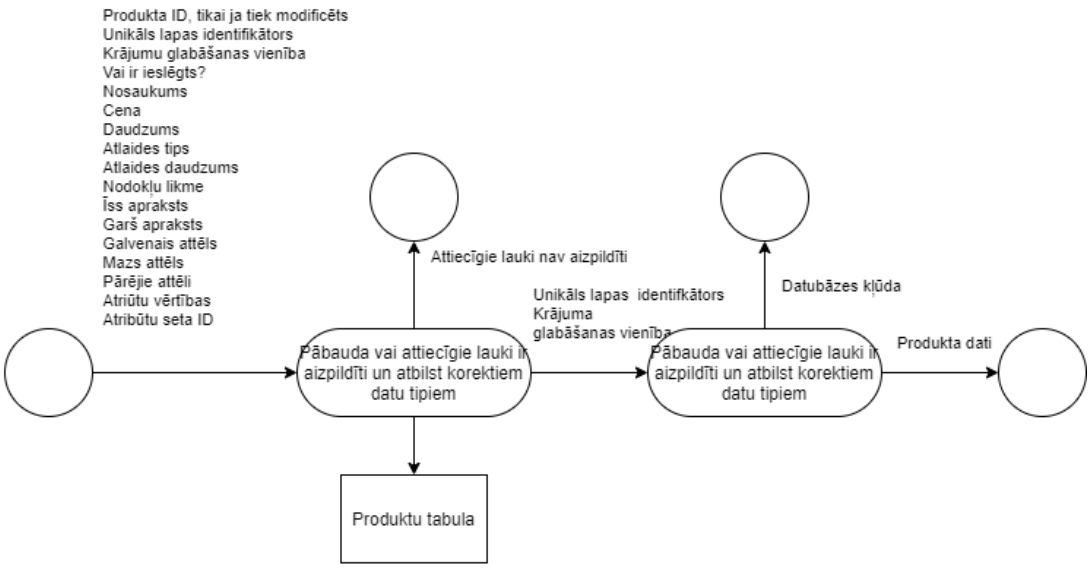
Produkta, kā arī citu entitiju sarakstu attēlošana ir ļoti svarīga administratorā daļai tā, kā lietotājām ir jābūt pieejai visiem produktiem, un iespējai apskatīt katru individuālu produktu, kā arī sakārtot to konkrētā secībā, lai varētu labāk atrastu tos produktus, kuri lietotājam ir nepieciešami. Produktu saraksts tiek veidots no lapām, un lapas izmērs ir nosakāms, var rādīt vairāk produktus vienā lapā pēc izvēles, vai mazāk, ir iespējams norādīt lapas numur, lai varētu kontrolēt, tieši kura lapa ir redzama lietotājam, un ir iespējams kārtot pēc kaut kāda lauka, vai nu augošā vai dilstošā secībā.



2. att: Produktu attēlošanas datu plūsmas diagramma

3 Produkta veidošana vai rediģēšanas

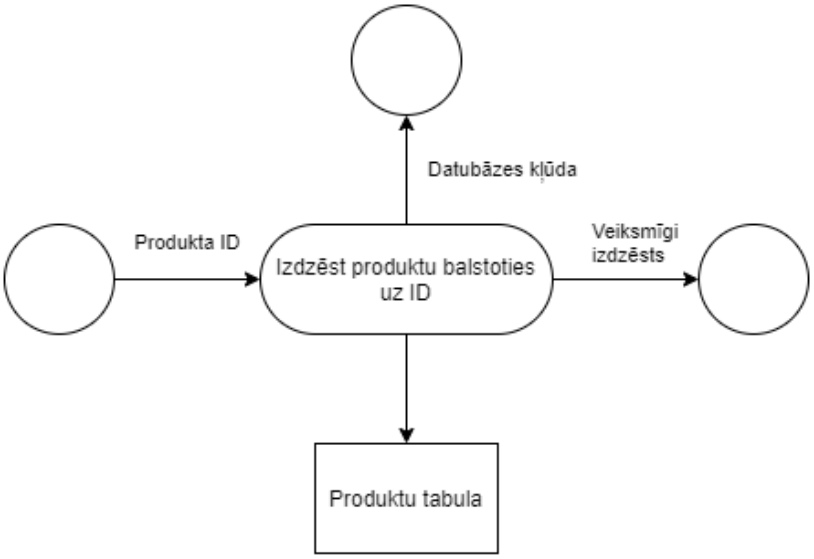
Sistēmai ir arī jānodrošina produktu izveide un rediģēšana, kopumā abas darbības ir vienādas, izņemot ar to, ka lai modificētu produktu sistēmai vajag sūtīt arī paša produkta ID. Produkta saglabāšanas process notiek tā, ka lietotājs ievada attiecīgos produkta datus un tad, kad visi nepieciešamie lauki tika ievadīti, tas var saglabāt produktu.



3. att: Produktu veidošanas un rediģēšanas datu plūsmu diagramma

4 Produkta dzēšanas

Produkta, kā arī citu entitiju dzēšanas ir svarīga, lai lietotājs varētu izdzēst nederīgus produktus.

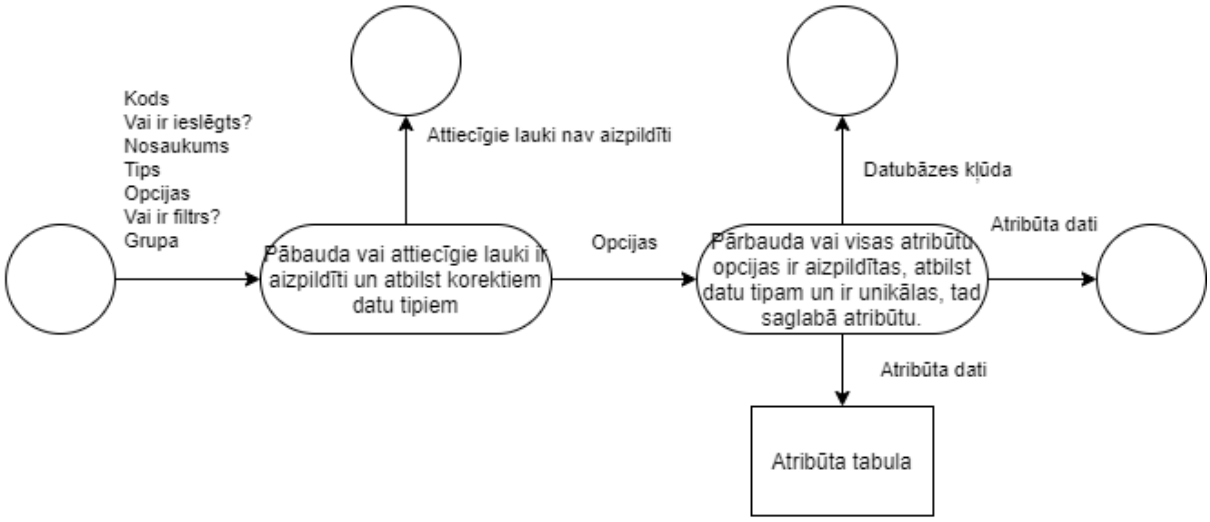


4. att: Produkta dzēšanas datu plūsmu diagramma

5 Atribūta pievienošanas

Produktiem var būt dažādi atribūtu, tāpēc administratoram ir jābūt iespējai pievienot specifisku produkta atribūtus. Visiem atribūta laukiem ir jābūt korekti aizpildītiem, un atribūtu opcijām, ir jābūt visām unikālam un definētām, pretējā gadījumā atribūts netiks saglabāts.

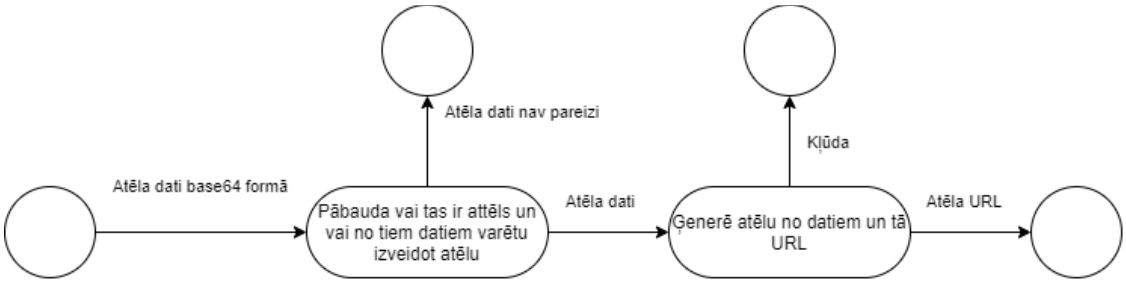
Pārējās atribūtu darbības netiks attēlotas datu plūsmu diagramma, tā kā tās ir ļoti līdzīgas, tam, kas ir produktam, tikai mainās dati.



5. att: Atribūta veidošanas datu plūsmas diagramma

6 Attēla augšupielādēšana

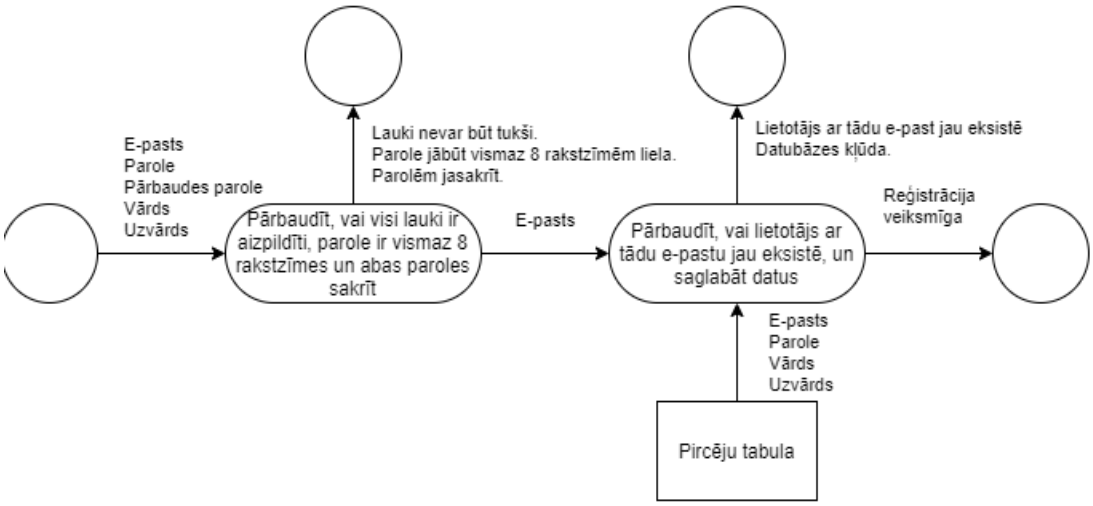
Lai varētu pievienot produktam dažādus attēlus ir jānodrošina attēlu apstrādes process, klients aizsūta attēlu serverim Base64 formātā, tas attēlu datus pārbauda un izveido attēlu ar unikālu nosaukumu, lai nebūtu kolīziju, un nosūta atpakaļ unikālu attēla URL. Šim konkrētam procesam jābūt diezgan nodrošinātam, jo ja kaut kādā viedā tiek pievienoti citi faili, tad var rasties problēma.



6. att: Attēla pievienošanas datu plūsmas diagramma

7 Lietotāja veidošana

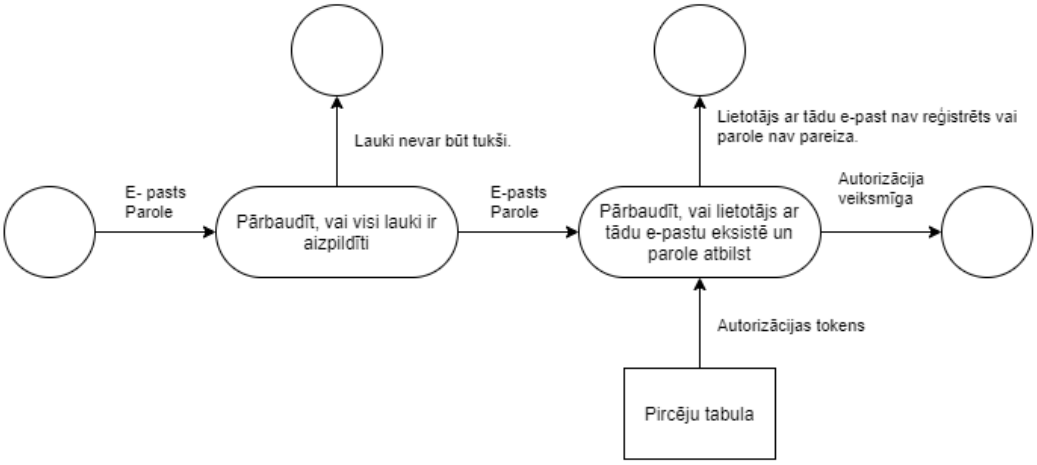
Lietotāja veidošanas ir viens no galvenajiem mehānismiem, kas ir jānodrošina sistēmai. Jā lietotājs nav autorizēts, tad tas nevar veikt noteiktas funkcijas, piemēram, kā pievienot produktu grozām, veikt pasūtījumu, rakstīt produkta atsauksmi un veikt citas darbības. Lietotāju autorizācijas ir svarīga ne tikai, lai atļautu veikt dažādas darbības, bet arī, lai noteiktiem lietotājiem ierobežotu noteiktas darbības, piemēram, parast pircējs nevar rediģēt produkta datus, bet administrators. Šeit konkrēti tiek aprakstīta pircēja lietotāja izveide, nevis administratora. Lai lietotājs varētu reģistrēties, tam ir jāievada attiecīgie dati: e-pasts, parole, pārbaudes parole, vārds, uzvārds. Visi šie lauki ir obligāti, e-pastam ir jābūt unikālam un nereģistrētam, kā arī parolēm jāsakrīt. Pēc reģistrācijas lietotājam tiek atsūtīts reģistrācijas tokens, kurš ļaus veikt noteiktas darbības.



7. att: Lietotāja reģistrācijas datu plūsmu diagramma

8 Lietotāja autorizācija

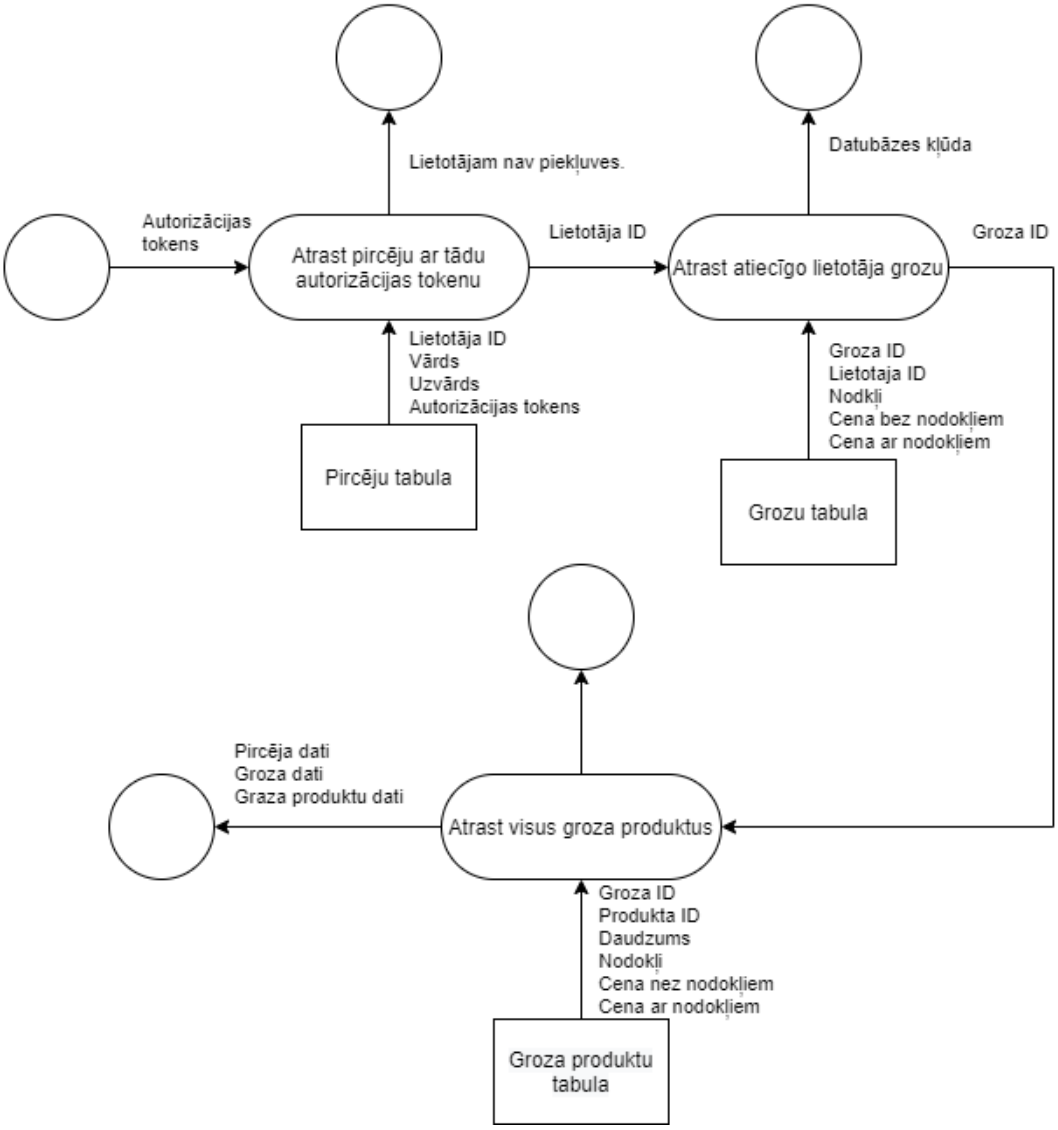
Ja lietotāja reģistrācijas ir nodrošināta, tad arī lietotāja autorizācijas ir jābūt nodrošinātai, lai lietotājas, varētu autorizēties savā akauntā. Process ir ļoti līdzīgs reģistrācijas, tikai šajā gadījumā, tiek pieprasīts e-pasts un parole. Un balstoties uz tām parolēm tiek atrasts lietotājs datu bāzē un ģenerēts autorizācijas tokens, kurš dos atļauju veikt darbības saistībā ar šo lietotāju.



8. att: Lietotāja autorizācijas datu plūsmas diagramma

9 Lietotāja datu ielādēšana

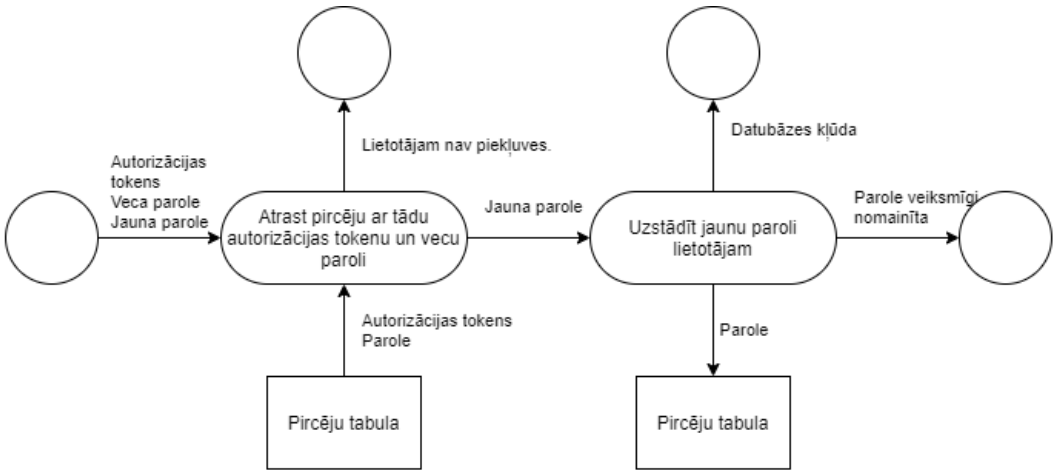
Pēc tam kad lietotājs tiek autorizēts, tam ir nepieciešams iegūt visus savus datus, tas ir visi pircēja personālie dati, groza kopsavilkums un groza saturs. Tas tiek veikts izmantojot autorizācijas tokenu. Izmantojot autorizācijas tokenu tiek atrasts lietotājs, un tad tiek atrasts attiecīgs grozs un tā saturs. Visi dati sagrupēti un aizsūtīti lietotājam.



9. att: Lietotāja datu ielādēšanas datu plūsmas diagramma

10 Paroles maiņa

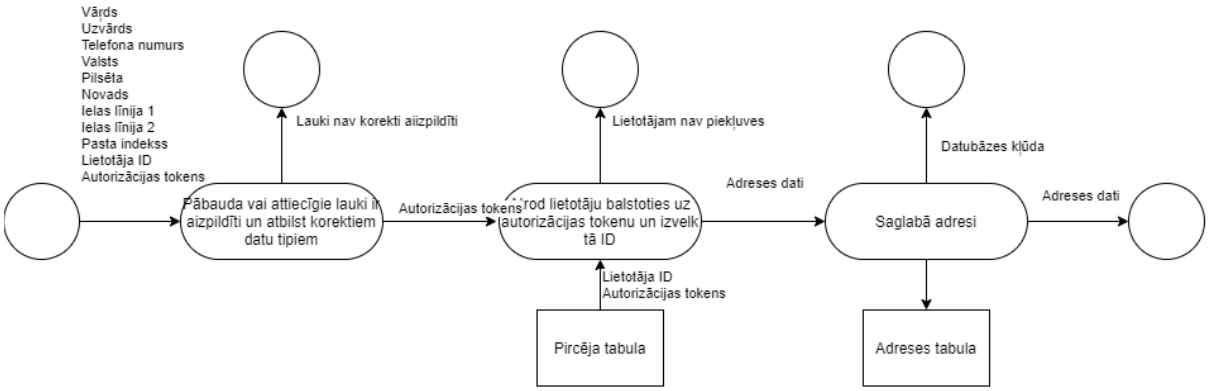
Lietotājam ir jābūt iespējai nomainīt savu paroli, lai to izdarītu tam ir jāievada veca parole un jaunā parole. Tad izmantojot lietotāja autorizācijas tokenu un paroli, varēs atrast lietotāju, un pārliecināties, kad akaunts pieder tieši tām lietotājam, un nomainīt paroli.



10. att: Klienta paroles maiņas datu plūsmas diagramma

11 Adreses veidošana

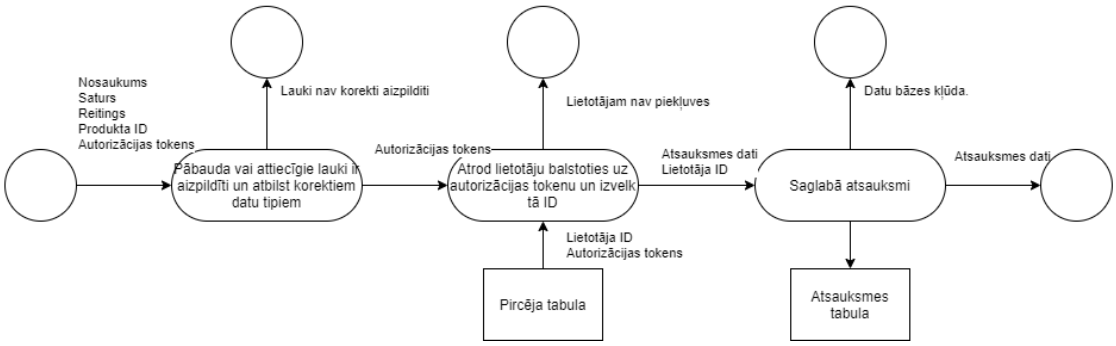
Klientam ir jābūt iespējai pievienot adresi



11. att: Adreses pievienošanas datu plūsmas diagramma

12 Atsauksmes pievienošanas

Klientam ir jābūt iespējai pievienot atsauksmi uz produktu, kuru tas ir nopircis, un klients nevar pievienot vairāk par 1 atsauksmi 1 produktam, tas tiek pārbaudīts izmantojot datubāzes nosacījumus.

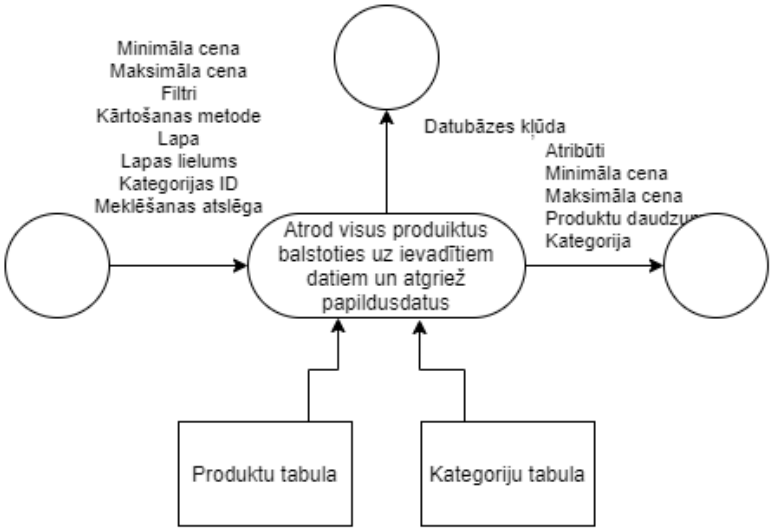


12. att: Atsauksmes pievienošanas datu plūsmas diagramma

13 Produktu meklēšana

Pircējiem ir jābūt iespējai atrasts nepieciešamus produktus. Tāpēc jānodrošina meklēšanas darbība. Produkti tiek attēloti saraksta veidā vairākās lapās, tāpēc, ir nepieciešams norādīt lapas numuru un lapas izmēru, kas nosaka cik daudz produktu būs vienā lapā. Produkti var būt meklēti balstoties uz konkrētiem parametriem, tas ir minimālā cena, maksimālā cena, atribūtu filtri. Produktus var arī kārot balstoties uz vairākām metodēm: pēc nosaukuma augošā/dilstošā secībā, pēc pārdodamības augošā/dilstošā secībā, pēc cenas augošā/dilstošā secībā, vai nekārtot vispār. Var arī norādīt kategoriju, kurai pieder produkti vai konkrētu pieprasījumu, kurš meklēs pēc līdzīgiem produktu nosaukumiem.

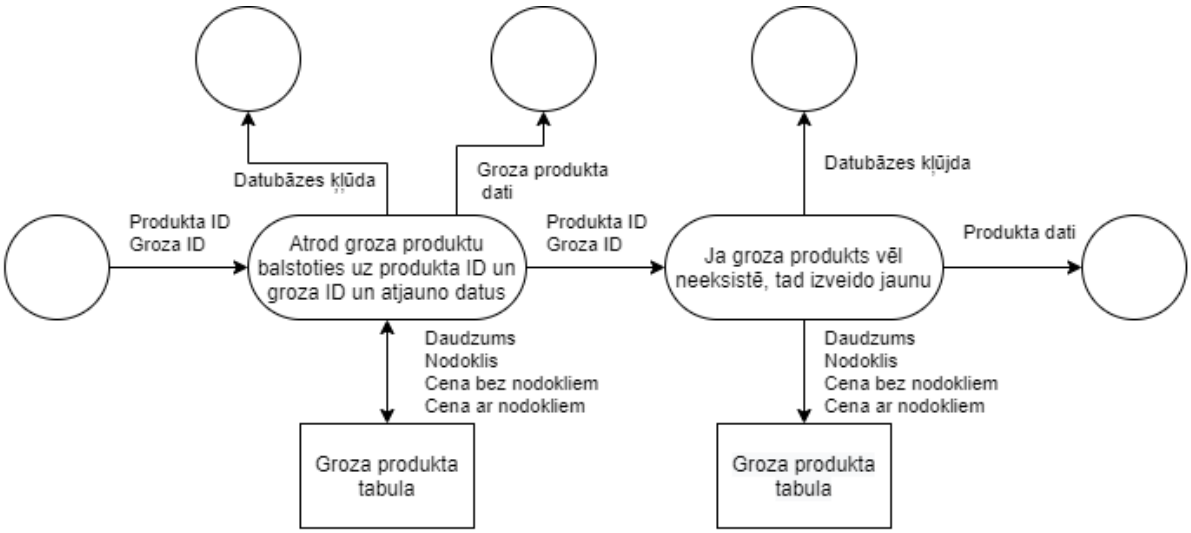
Pēc pieprasījuma izpildes balstoties uz konkrētiem parametriem, tiek atrasti visi produkti, saistīta kategorija, kā arī dažādas agregācijas, piemēram, atribūtu filtri, kuri ir nepieciešami, lai lietotājs varētu filtrēt produktus.



13. att: Produktu meklēšanas datu plūsmas diagramma

14 Produkta pievienošana grozām

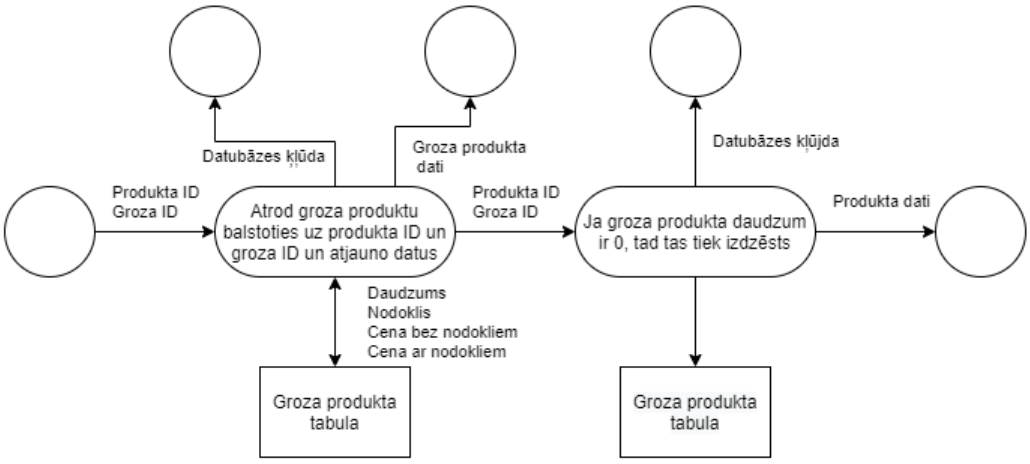
Lai lietotājs varētu veikt pasūtījumu, tam no sākuma ir nepieciešams pievienot produktus grozām. Šajā diagrammā nav attēlots viens solis, tas ir, ka tikai autorizētie pircēji var pievienot produktus grozām. Produkts tiek pievienots grozām, balstoties uz produkta ID un groza ID. Pēc nosacījuma produkts tiek pievienots ar vienu vienību, un uzreiz tiek izrēķināta produkta cena, nodokļi, un kopējā cena. Ja groza produkts neeksistē, tad tas tiek izveidos, ja eksistē, tad tam tiek mainīts daudzums, un tiek pārskaitītā cenas.



14. att: Produkta pievienošana grozām datu plūsmas diagramma

15 Produkta noņemšana no groza

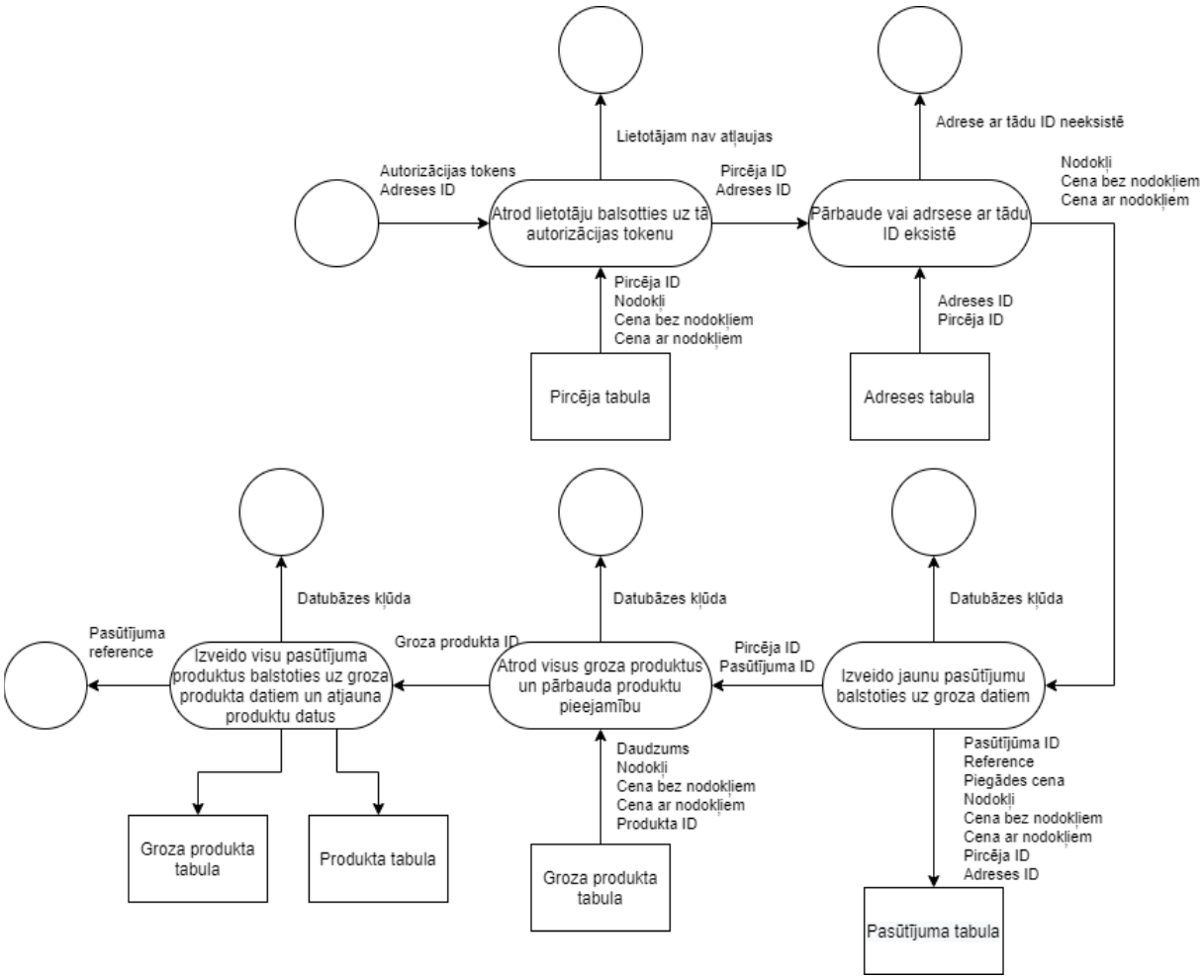
Ir arī jābūt iespējai izņemt produktu no grozā, tas tiek veids arī tāda viedā, ka ja groza produkta daudzums pēc noņemšanas ir lielāks par nulli, tad tas tiek atgriezts ar jaunām pārskaitītām vērtībām, bet ja groza produkta daudzums ir vienāds ar nulli, tad tas tiek izdzēst un atgriezts lietotājam.



15. att: Groza izņemšanas datu plūsmas diagramma

16 Pasūtījuma ģenerēšana

Pēc visu nepieciešamo soļu veikšanas, lietotajam tiek ģenerēts pasūtījums. Lai to ģenerētu, sistēmai, ir jābūt pieejai, pie adreses ID, pircēja ID, un groza ID, tad balstoties uz groza datiem un tā saturu tiek izveidots jauns pasūtījums un pasūtījuma saturs.



16. att: Pasūtījuma veidošanas datu plūsmas diagramma

17 Pārējās datu plūsmas

Vēl varētu attēlot datu plūsmas darbībām ar atribūtu setiem (apskatīšana, pievienošana, rediģēšana, dzēšana), kategorijām (apskatīšana, pievienošana, rediģēšana, dzēšana), pasūtījumiem (apskatīšana, rediģēšana, dzēšanā), rēķiniem (apskatīšana, dzēšana), klientiem (), klientu grupām (apskatīšanā, rediģēšana, dzēšana), atsauksmēm (apskatīšanā, rediģēšana, dzēšana), taču visas tās datu plūsmas būs jau līdzīgas produkta pārvaldes datu plūsmām, tāpēc tiek pārskatītas tikai nepieciešamas.

5 DATU STRUKTŪRAS APRAKSTS

Datu bāzes fiziskā struktūra (skatīt 2. pielikumā) sastāv no 15 tabulām, kas satur informāciju par sistēmas klientiem, klientu grupām, kategorijām, produktiem, pasūtījumiem, rēķiniem, atribūtiem, atribūtu setiem, produktu atsauksmēm un citiem objektiem. Ir divas galvenās tabulas, kurām ir visvairāk savienojumu, tas ir produktu tabula, un klientu tabula, un, acīmredzami, šie divi objekti, kopumā arī definē lielāko daļu no sistēmas, jo klients pērk produktus. Datubāzes realizēšanai izmanto PostgreSQL. Shēma tika ģenerēta (skatīt 2. pielikumā).

Tabulā "Address" glabājas informācija par klienta izveidotajām adresēm.

Šī tabula attiecās uz "Customer" tabulu ar "customer_id" ārējo atslēgu, un viedo "daudzi pret vienu" attiecību – vairākas adreses var piederēt vienam klientam.

1. tabula

Tabulas "Address" struktūra

Numurs	Nosaukums	Tips	Garums	Piezīme
1.	id	integer	-	Primāra atslēga
2.	firstName	varchar	255	Vārds
3.	lastName	varchar	255	Uzvārds
4.	phoneNumber	varchar	32	Telefona numurs
5.	country	varchar	255	Valsts
6.	city	varchar	255	Pilsēta
7.	province	varchar	255	Reģions
8.	street1	varchar	255	Ielas līnija 1
9.	street2	varchar	255	Ielas līnija 2
10.	postalCode	varchar	16	Pasta kods
11.	customer_id	integer	-	Ārēja atslēga uz klientu

Tabulā "Attribute" glabājas informācija par produkta atribūtiem.

Šī tabula attiecās uz "AttributeSet" tabulu un viedo "daudzi pret daudziem" attiecību – vairāki atribūti var piederēt vairākiem atribūtu setiem, tāpēc ir nepieciešama atsevišķā tabula "AttributeSetAttribute", kur ir divas ārēja atslēgas, kas savieno šīs tabulas: "attribute_set_id", "attribute_id".

2. tabula

Tabulas “Attribute” struktūra

Numurs	Nosaukums	Tips	Garums	Piezīme
1.	id	integer	-	Primāra atslēga
2.	code	varchar	255	Unikāls kods
3.	isEnabled	boolean	-	Vai ir ieslēgts
4.	label	varchar	-	Nosaukums
5.	type	varchar	255	Datu tips
6.	attributeOptions	jsonb	-	Iespējamie vērtību varianti
7.	isFilter	boolean	-	Vai tiek izmantots, kā filtrs meklējot produktu
8.	AttributeGroup	Varchar	255	Atribūtu grupa

Tabulā “AttributeSet” glabājas informācija par atribūtu setiem, tās ir kopas, kas apvieno vairākus atribūtus, un produktam var būt viena atribūtu kopa, un uz to kopu bāzēsies tā iespējamie atribūti.

Šī tabula attiecās uz “Attribute” tabulu un viedo “daudzi pret daudziem” attiecību – vairāki atribūti var piederēt vairākiem atribūtu setiem, tāpēc ir nepieciešama atsevišķā tabula “AttributeSetAttribute”, kur ir divas ārēja atslēgas, kas savieno šīs tabulas: “attribute_set_id”, “attribute_id”.

3. tabula

Tabulas “AttributeSet” struktūra

Numurs	Nosaukums	Tips	Garums	Piezīme
1.	id	integer	-	Primāra atslēga
2.	code	varchar	255	Unikāls kods
3.	isEnabled	boolean	-	Vai ir ieslēgts
4.	name	varchar	255	Nosaukums

Tabulā “AttributeSetAttribute” ir tabula, lai realizēta “daudzi pret daudziem” attiecību starp “AttributeSet” un “Attribute” tabulām.

4. tabula

Tabulas “AttributeSetAttribute” struktūra

Numurs	Nosaukums	Tips	Garums	Piezīme

1.	attribute_set_id	integer	-	Ārēja atslēga uz atribūtu setu
2.	attribute_id	integer	-	Ārēja atslēga uz atribūtu

Tabulā "CartItem" glabājas informācija par katru pievienotu produktu pie klienta groza.

Šī tabula ir nepieciešama, lai realizētu "daudzi pret daudziem" attiecību starp "Cart" un "Product" tabulām.

5. tabula

Tabulas "CartItem" struktūra

Nr.	Nosaukums	Tips	Garums	Piezīme
1.	quantity	integer	-	Produkta daudzums
2.	totalTax	double	-	Nodokļi
3.	subtotal	double	-	Kopēja cena bez nodokļiem
4.	total	Double	-	Kopēja cena ar nodokļiem
5.	customer_id	integer	-	Ārēja atslēga uz grozu
6.	product_id	integer	-	Ārēja atslēga uz produktu

Tabulā "Category" glabājas informācija par kategorijām.

Šī tabula attiecās uz "Product" tabulu un viedo "daudzi pret daudziem" attiecību – vairākās kategorijas var piederēt vairākiem produktiem, tāpēc ir nepieciešama atsevišķā tabula "ProductCategory", kur ir divas ārējas atslēgas, kas savieno šīs tabulas: "product_id", "category_id".

Vēl šī tabula attiecās pati uz sevi ar arējo atslēgu "category_id", un veido attiecību viens pret daudziem, jo vienai kategorijai var būt vairākas apakškategorijas.

6. tabula

Tabulas "Category" struktūra

Nr.	Nosaukums	Tips	Garums	Piezīme
1.	id	integer	-	Primāra atslēga
2.	urlKey	varchar	255	Unikāls lapas identifikators

3.	isEnabled	boolean	-	Vai ir ieslēgta
4.	name	varchar	255	Nosaukums
5.	isInMenu	boolean	-	Vai tiek pievienota pie izvēlnes
6.	content	text	-	Kategorijas saturs
7.	category_id	integer	-	Ārēja atslēga uz pašu sevi

Tabulā "Customer" glabājas informācija par klientiem, kā arī informācija par tā groza saturu, netika veidota atsevišķa tabula, kurā glabātos groza, dati un tad tam būtu ārēja atslēja uz klientu tabulu, ar viens pret viens attiecību. Šajā gadījumā ir vienkāršāk glabāt groza datus pašā klienta tabulā, jo jebkurā gadījumā, klientam var būt tikai viens grozs.

7. tabula

Tabulas "Customer" struktūra

Nr.	Nosaukums	Tips	Garums	Piezīme
1.	id	integer	-	Primāra atslēga
2.	email	varchar	255	Unikāls epasts
3.	password	varchart	255	Šifrēta parole
4.	firstName	varchart	255	Vārds
5.	lastName	varchart	255	Uzvārds
6.	totalTax	double	-	Nodokļi
7.	subtotal	double	-	Cena bez nodokļiem
8.	total	double	-	Cena ar nodokļiem
9.	token	text	-	Autorizācijas tokens

Tabulā "CustomerCustomerGroup" glabājas informācija par klientu grupu grupām.

Šī tabula ir nepieciešama, lai realizētu "daudzi pret daudziem" attiecību starp "Customer" un "CustomerGroup" tabulām.

8. tabula

Tabulas "CustomerCustomerGroup" struktūra

Nr.	Nosaukums	Tips	Garums	Piezīme
1.	customer_id	integer	-	Ārēja atslēga uz klientu
2.	customer_group_id	integer	-	Ārēja atslēga uz klienta grupu

Tabulā “CustomerGroup” glabājas informācija par klientu grupām.

Šī tabula attiecās uz “Customer” tabulu un viedo “daudzi pret daudziem” attiecību – vairāki klienti var piederēt vairākām klientu grupām, tāpēc ir nepieciešama atsevišķā tabula “CustomerCustomerGroup”, kur ir divas ārēja atslēgas, kas savieno šīs tabulas: “customer_id”, “customer_group_id”.

9. tabula

Tabulas “CustomerGroup” struktūra

Nr.	Nosaukums	Tips	Garums	Piezīme
1.	id	integer	-	Primāra atslēga
2.	code	varchar	255	Unikāls kods
3.	Name	Varchar	255	Nosaukums

Tabulā “Invoice” glabājas informācija par klienta rēķiniem. Šī tabula glabā gandrīz tādus pašus datus, tikai tie ir galīgie dati, kas tiek veidoti transakcijas beigās, kur glabājas arī galīga piegādes izmaksa.

Šī tabula attiecās uz “Order” tabulu ar unikālu ārēju atslēgu “order_id”, un viedo attiecību “viens pret vienu.

10. tabula

Tabulas “Invoice” struktūra

Nr.	Nosaukums	Tips	Garums	Piezīme
1.	id	integer	-	Primāra atslēga
2.	date	date	-	Datums
3.	totalDelivery	double	-	Pārvešanas izmaksas
4.	totalTax	double	-	Nodokļi
5.	subtotal	double	-	Kopējā cena bez nodokļiem
6.	total	double	-	Kopējā cena
7.	order_id	integer	-	Unikāla ārēja atslēga uz pirkumu

Tabulā “Order” glabājas informācija par pasūtījumiem.

Šī tabula attiecās uz “Customer” tabulu ar “customer_id” atslēgu un viedo attiecību “viens pret daudziem”, jo vienam klientam var būt vairāki pasūtījumi, bet pasūtījumam var būt tikai viens klients.

Šī tabula arī attiecās uz “OrderItem” tabulur un viedo atttiecību “viens pret daudziem”, jo vienam pasūtījuma var būt vairāki pasūtījuma produkti, bet vienam pasūtījuma produktam, var būt tikai viens pasūtījums.

11. tabula

Tabulas “Order” struktūra

Nr.	Nosaukums	Tips	Garums	Piezīme
1.	id	integer	-	Primāra atslēga
2.	reference	varchar	255	Unikāla eference
3.	date	date	-	Datums
4.	status	varchar	255	Status
5.	totalDelivery	double	-	Pārvešanas izmaksas
6.	totalTax	double	-	Nodokļi
7.	subtotal	double	-	Kopēja cena bez nodokļiem
8.	total	double	-	Kopēja cena
9.	customer_id	integer	-	Ārēja atslēga uz klientu
10.	address_id	integer	-	Ārēja atslēga uz adresi

Tabulā “OrderItem” glabājas informācija par pasūtījuma produktiem.

Šī tabula attiecās uz “Order” tabulu ar “order_id” ārēju atslēgu, un veido attiecību “viens pret daudziem”, jo vienam pasūtījuma var būt vairāki pasūtījuma produkti, bet vienam pasūtījuma produktam var būt tikai viens pasūtījums

Šī tabula arī attiecās uz “Product” tabulu ar “product_id” ārēju atslēgu, un veido attiecību “viens pret daudziem”, jo vienām produktam var būt vairāki pasūtījuma produkti, bet vienam pasūtījuma produktam var būt tikai viens produkts.

Kopumā šī tabula kalpo kā starptabula starp produktu un pasūtījumu, lai realizētu “daudzi pret daudziem” attiecību

12. tabula

Tabulas “OrderItem” struktūra

Nr.	Nosaukums	Tips	Garums	Piezīme
1.	id	integer	-	Primāra atslēga
2.	quantity	integer	-	Produkta daudzums
3.	totalTax	dobule	-	Nodokļi

4.	subtotal	double	-	Kopēja cena bez nodokļiem
5.	total	double	-	Kopēja cena
6.	order_id	integer	-	Ārēja atslēga uz pirkumu
7.	product_id	integer	-	Ārēja atslēga uz produktu

Tabulā “Product” glabājas informācija par produktiem.

Šī tabula attiecās uz “AttributeSet” tabulu ar “attribute_set_id” ārēju atslēgu, un veido attiecību “viens pret dauzdiem”, jo vienam produktam var būt tikai viens atribūtu sets, bet vienam atribūtu setam var būt vairāki produkti.

13. tabula

Tabulas “Product” struktūra

Nr.	Nosaukums	Tips	Garums	Piezīme
1.	id	integer	-	Primāra atslēga
2.	urlKey	varchar	255	Unikāls lapas identifikators
3.	sku	varchar	255	Krājumu glabāšanas vienība
4.	isEnabled	boolean	-	Vai ir ieslēgts
5.	name	varchar	255	Nosaukums
6.	price	double	-	Pamata cena
7.	stockQuantity	integer	-	Daudzums noliktavā
8.	soldAmount	integer	-	Pārdots daudzums
9.	specialDiscountType	varchar	255	Atlaides tips (daudzums, procentuāls)
10.	specialDiscountValue	double	-	Atlaides vērtība
11.	specialTaxRate	double	-	Nodokļu procents
12.	shortDescription	varchar	255	Īss apraksts
13.	longDescription	text	-	Garš apraksts
14.	base_image	jsonb	-	Galvenais attēls
14.	thumbnail_image	jsonb		Mazais attēls
15.	other_images	jsonb		Citi attēli
16.	attributeValues	jsonb	-	Atribūtu vērtības
17.	attribute_set_id	integer	-	Ārēja atslēga uz atribūtu

Tabulā “ProductCategory” glabājas informācija par produktu kategorijām.

Šī tabula ir nepieciešama, lai realizētu “daudzi pret daudziem” attiecību starp “Product” un “Category” tabulām.

14. tabula

Tabulas “ProductCategory” struktūra

Nr.	Nosaukums	Tips	Garums	Piezīme
1.	product_id	integer	-	Ārēja atslēga uz produktu
2.	category_id	integer	-	Ārēja atslēga uz kategoriju

Tabulā “Review” glabājas informācija par klienta atsauksmēm uz produktu.

Šī tabula attiecas uz “Customer” tabulu ar “cutsomer_id” āreju atslēgu un veido attiecību “viens pret daudziem”, jo vienam lietotājam var būt vairākas atsauksmes, bet vienai atsauksmei var būt tikai viens lietotājs.

Šī tabula arī attiecās uz “Product” tabulu ar “produkt_id” ārēju atslēgu un veido attiecību “viens pret daudziem”, jo vienam produktam var būt vairākas atsauksmes, bet vienai atsauksmei var būt tikai viens produkts.

Kopumā šī tabula nodrošina “daudzi pret daudziem” attiecību start tabulām “Customer” un “Product”. Primāra atslēga tiek veidota no abām ārējām atslēgām.

, t15. tabula

Tabulas “Review” struktūra

Nr.	Nosaukums	Tips	Garums	Piezīme
1.	id	integer	-	Unikāls identifkātors
2.	status	varchar	255	Status (pieņemta, apstrādā, noliegta)
3.	date	date	-	Datums
4.	title	varchar	255	Nosaukums
5.	content	text	-	Saturs
6.	rating	integer	-	Reitings no 1 līdz 10
7.	customer_id	integer	-	Ārēja atslēga uz klientu, primāra kopā ar “product_id”
8.	product_id	integer	-	Ārēja atslēga uz produktu, primāra kopā ar

				“customer_id”

6 LIETOTĀJA CEĻVEDIS

6.1 Sistēmas prasības aparatūrai un programmatūrai

Nav specifiskas sistēmas aparatūras prasības, un programmu ir iespējams palaist gandrīz uz jebkuras modernās sistēmas, tāpēc, visas prasības attiecās tikai uz nepieciešamo programmatūru. Programma var būt instalēta gan uz Linux, gan uz Windows (labāk izmantot Windows, tāpēc kā tieši uz šīs operētājsistēmas tika izstrādāta programma), taču MacOS netiek atbalstīs šobrīd. Lai instalētu un palaistu to ir nepieciešama noteikta programmatūra:

- Node / NPM
- PostgreSQL

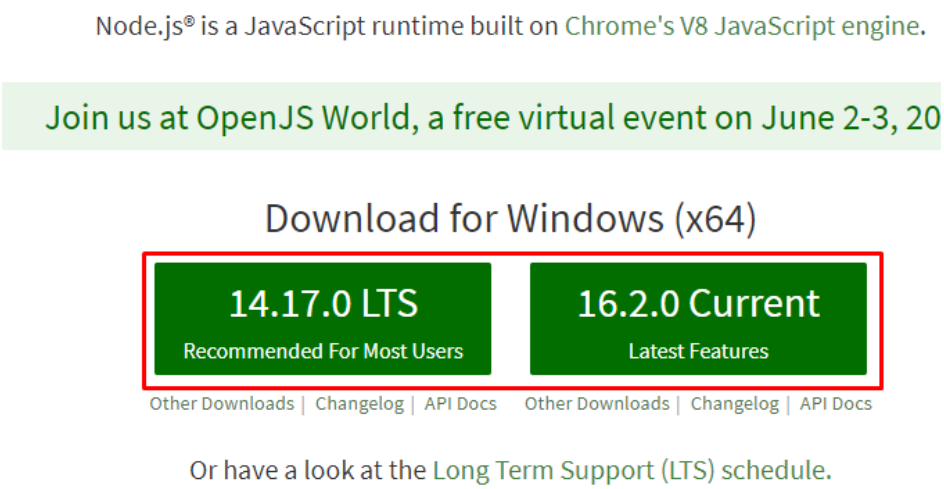
Windows

Lai šo programmu varētu palaist ir nepieciešams uzinstalēt

1 Node / NPM

1.1 Pāriet uz šo adresi: https://nodejs.org/en/

1.2 Tad lejupielādēt 14.17.0 LTS versiiju, taču programma spēj strādāt arī ar visjaunāko Node versiju



17. att. Node / NPM instalācija

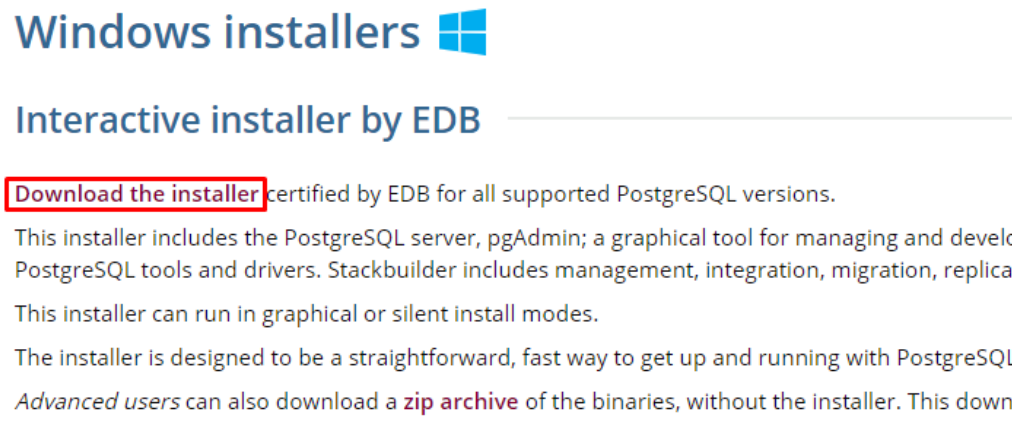
1.3 Palaist instalāciju un izpildīt visas darbības pēc noklusējuma

1.4 Pēc šo darbību izpildes, programma būs veiksmīgi instalēta

2 PostgreSQL

2.1 Pāriet uz šo adresi: https://www.postgresql.org/download/windows/

2.2 Tad lejupielādē jaunāko PostgreSQL versiju



18. att. PostgreSQL instalācija

2.3 Tad izpildīt visas darbības pēc noklusējuma.

2.4 Pēc šo darbību izpildes, programma būs veiksmīgi instalēta.

Linux (Ubuntu)

Lai uzinstalētu visu uz Linux (Ubuntu), ir nepieciešams palaist tikai dažas komandas:

1 Node / NPM

1.1 curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.38.0/install.sh | bash

1.2 nvm install 14.17.0

1.3 Pēc šo darbību izpildes, programma būs veiksmīgi instalēta

2 PostgreSQL

2.1 sudo apt update

2.2 sudo apt install postgresql postgresql-contrib

2.3 Pēc šo darbību izpildes, programma būs veiksmīgi instalēta

6.2 Sistēmas instalācija un palaišana

Vispirms ir nepieciešams strādājošs Node / NPM un PostgreSQL setups, kad tas ir sagatavots, tad ir nepieciešams palaist sekojošas komandas:

1. git clone https://github.com/rvtprog-kval-21/d41-DenissStrombergs-InternetaVeikals.git
2. cd d41-DenissStrombergs-InternetaVeikals
3. cd backend
4. npm start, tas nodrošinās to, ka servera daļa tiks palaista
5. Atvērt jaunu konsoles logu DenissStrombergs-InternetaVeikals direktorijā
6. cd frontend
7. npm start. Tas nodrošinās to, ka klienta daļa tiks palaista

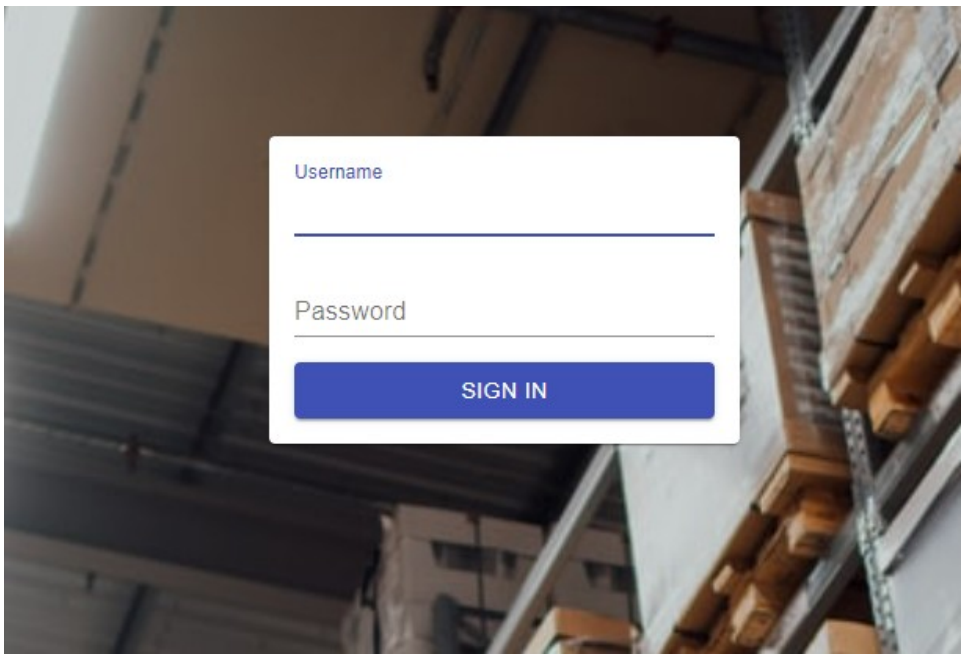
6.3 Programmas apraksts

Administrators

Administratora daļas apraksts:

1 Autorizācija

1.1 Lai autorizēties, kā administrators, ir nepieciešams pāriet uz "/admin" lapu

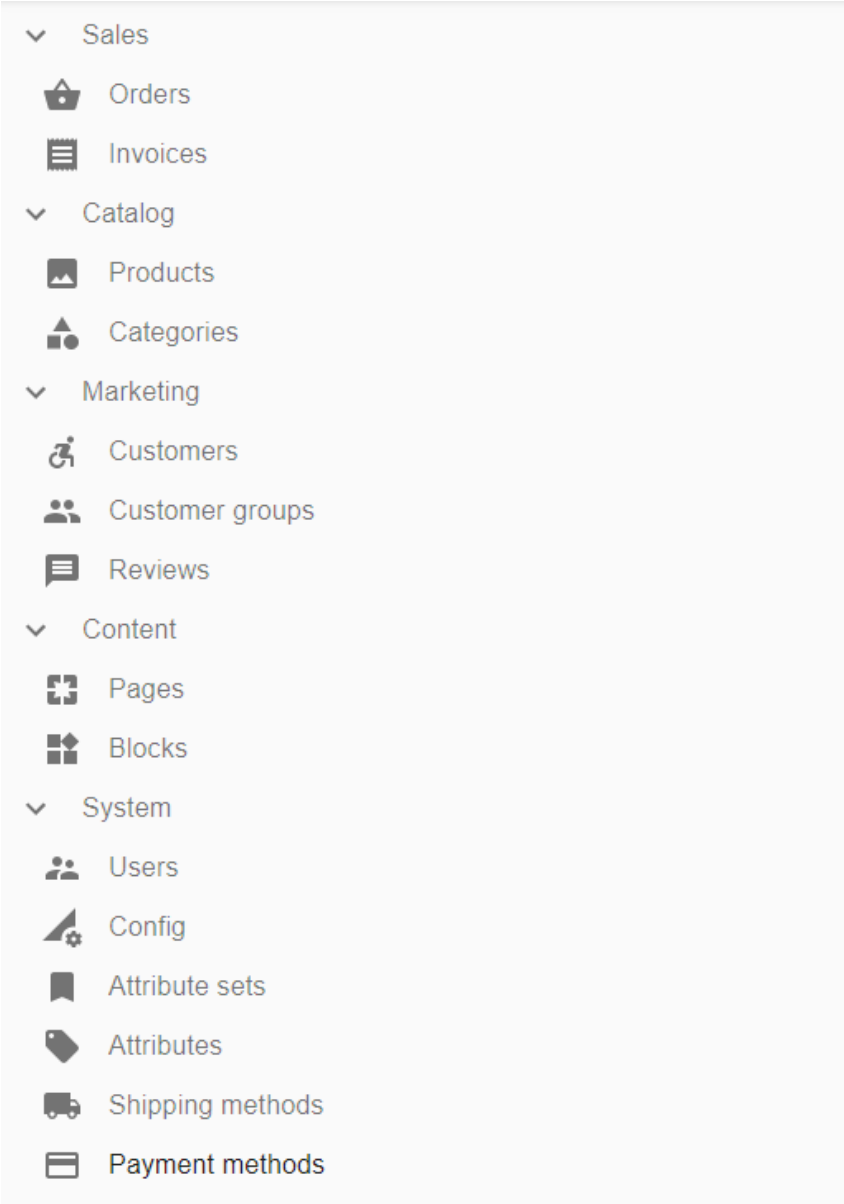


19. att. Autorizācijas forma

1.2 Tad ievadīt lietotāja autorizācijas datus, pēc noklusējuma, lietotāj vārds ir **admin** un parole ir arī **Admin1234**

1.3 Pēc lietotāja datu ievades, ir nepieciešams uzspiest uz “SIGN IN” pogu, un tad, ja pareizi lietotājvārds un parole tika veiksmīgi ievadītas, tad jūs pāriesiet uz administratora lapu

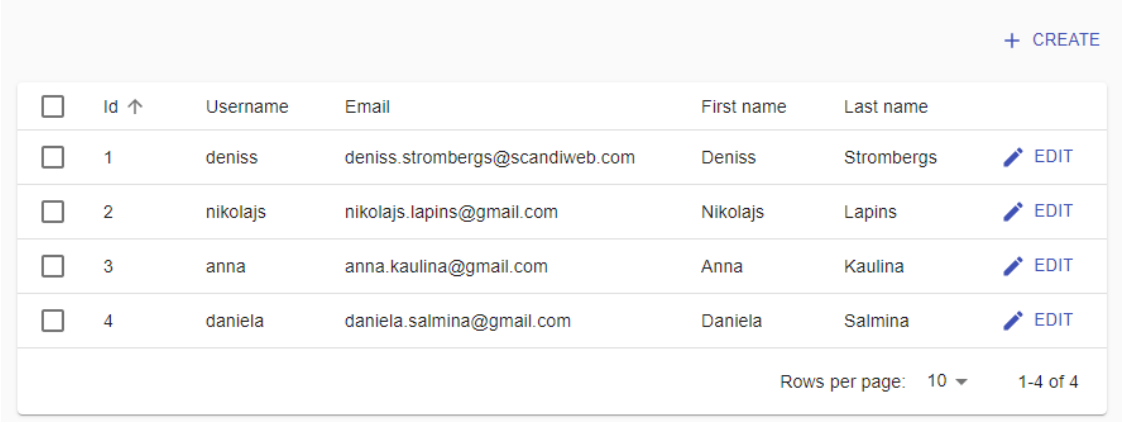
1.4 Pēc autorizācijas var redzēt navigācijas paneli, tajā ir iespējams redzēt, visus administratora pārvaldes sadalījumus, spiežot uz konkrētu sadalījumu lietotājs var pāriet uz noteiktu lapu, kur būs redzami visi atbilstoši dati



20. att. Navigācijas panelis

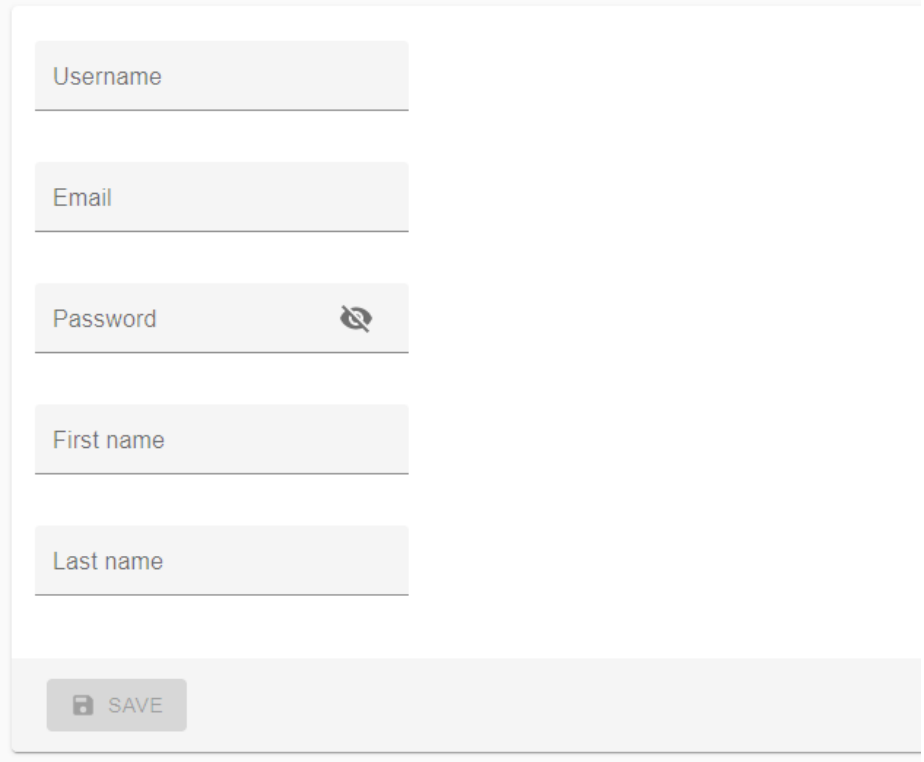
2 Lietotāju pārvalde

2.1 Lai redzētu, kuriem lietotājiem ir piekļuve pie administratora daļas ir nepieciešams pāriet uz "User" sadalījumu, šajā vietā var arī uzspiest pogu "Create", lai pievienotu jaunu lietotāju, vai "Edit", lai apskatītu vai rediģētu jau eksistējošu lietotāju:



21. att. Lietotāju saraksts

2.2 Pārejot vai nu uz “Create” vai uz “Edit”, parādīsies lauki, kuros vajadzēs ierakstīt nepieciešamus datus par lietotāju, lai to varētu saglabāt.



22. att. Lietotāja rediģēšanas lapa

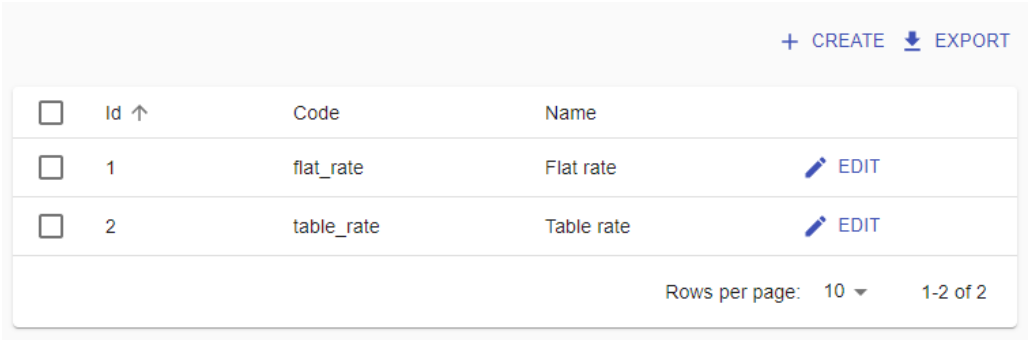
3 Konfigurācijas pārvalde

3.1 Lai redzētu kopējo veikala konfigurāciju, ir nepieciešams pāriet uz “Config” sadaļu.

3.2 Tajā var redzēt visu iestatījumus

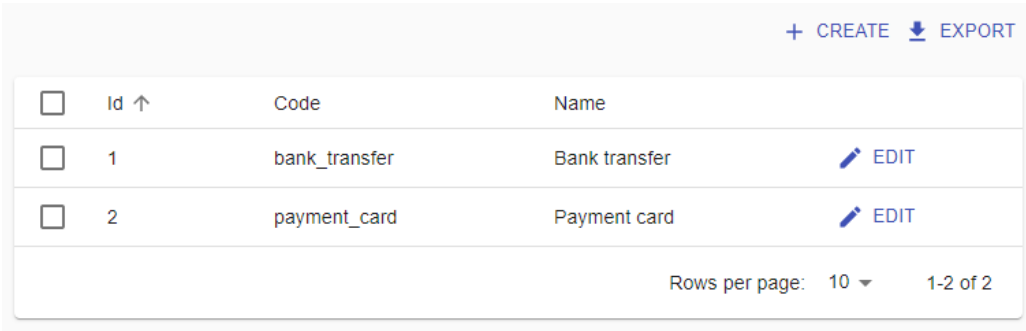
23. att. Konfigurācija

3.3 Lai redzētu visas piegādes veidus ir nepieciešams pāriet uz “Shipping methods” sadaļu, bet šobrīd tiem nav liela pielietojuma, tāpēc īpaši daudz jēgas pievienot tos nav



24. att. Piegādes veidu saraksts

3.4 Lai redzētu visas apmaksas veidus ir nepieciešams pāriet uz “Payment methods”, bet šobrīd tiem nav liela pielietojuma, tāpēc īpaši daudz jēgas pievienot tos nav

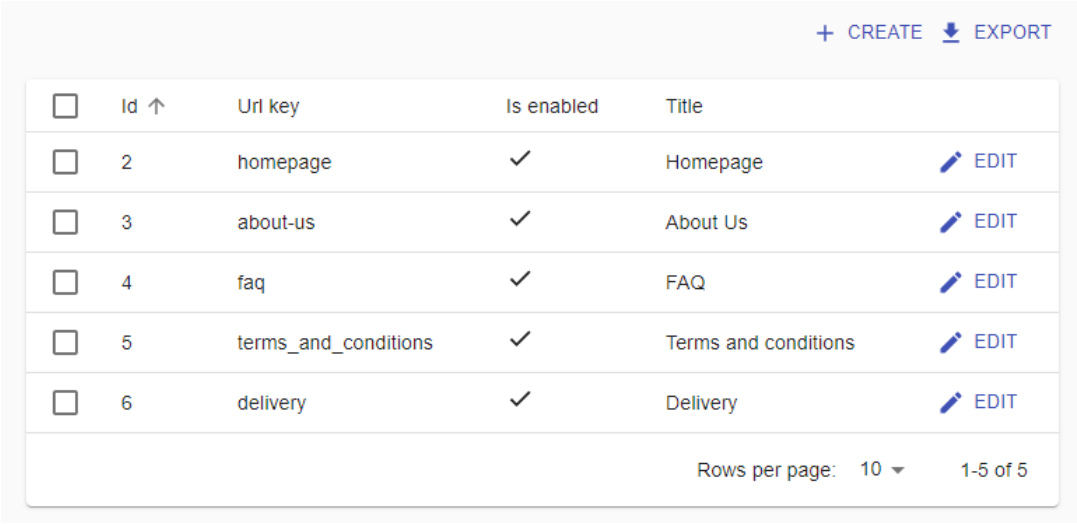


25. att. Apmaksas veidu saraksts

4 Satura pārvalde, tā ir “Content” sadaļa, kas satur 2 apakšsadaļas, “Page”, lai redzētu un rediģētu visas specifiskas lapas un “Block”, lai redzētu un rediģētu

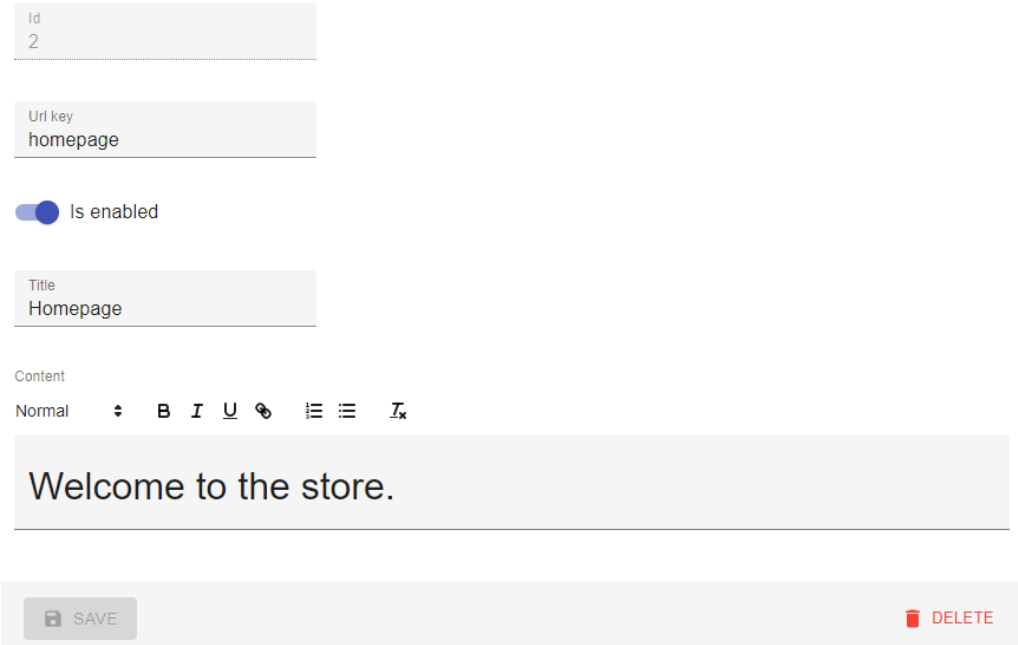
visas atsevišķas lapas komponentus. Taču komponentus šobrīd nevar ievietot lapas iekšā, tikai, ja tas tiek izdarīts programmas kodā.

4.1 Lai redzētu visas lapas ir nepieciešams pāriet uz “Pages”, šajā vietā var arī uzspiest pogu “Create”, lai pievienotu jaunu lapu, vai “Edit”, lai apskatītu vai rediģētu jau eksistējošu lapu:



26. att. Lapu saraksts

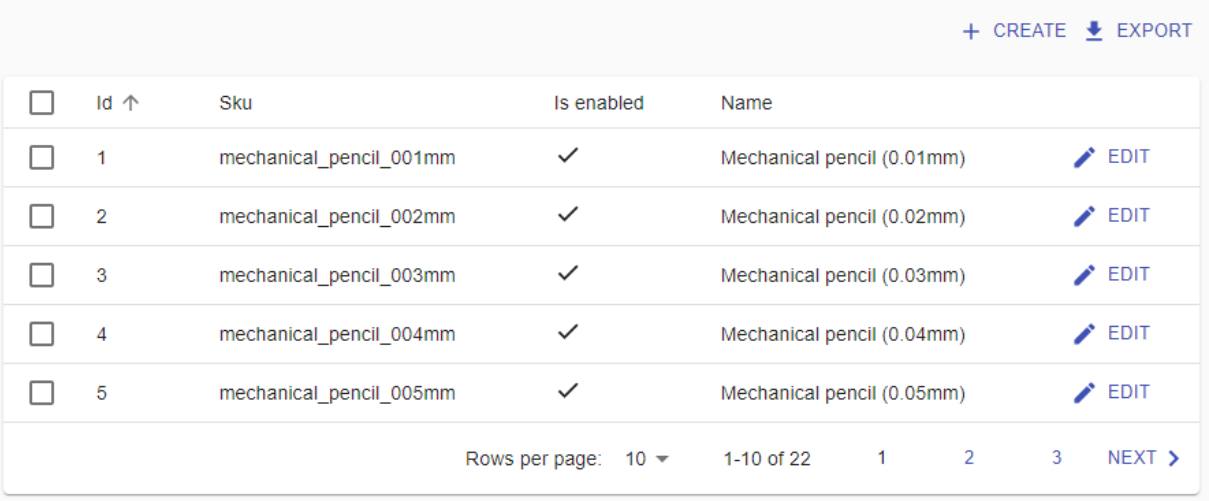
4.2 Pārejot vai nu uz “Create” vai uz “Edit”, parādīsies lauki, kuros vajadzēs ierakstīt nepieciešamus datus par lietotāju, lai to varētu saglabāt.



27. att: Mājaslapas rediģēšana

5 Produktu pārvalde

5.1 Lai redzētu, kuri produkti ir pievienoti pie sistēmas, ir nepieciešams pāriet uz “Products” sadalījumu, šajā vietā var arī uzspiest pogu “Create”, lai pievienotu jaunu produktu, vai “Edit”, lai apskatītu vai rediģētu jau eksistējošu produktu, produkta datus var arī eksportēt uzspiežot uz “Export” pogu:

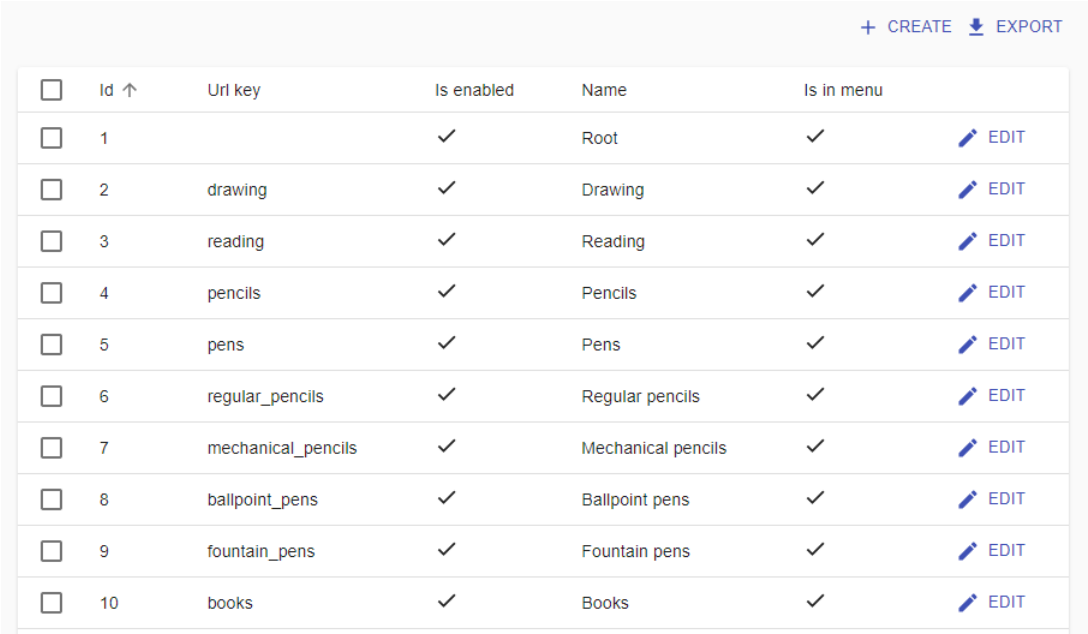


28. att: Produktu saraksts

5.2 Pārejot vai nu uz “Create” vai uz “Edit”, parādīsies lauki, kuros vajadzēs ievadīt nepieciešamus datus par produktu, lai to varētu saglabāt. Produktu var arī izdzēst, vairāk par produktu veidošanu skatīt “Testa piemērs” sadaļā.

6 Kategoriju pārvalde

6.1 Lai redzētu, kuras kategorijas ir pievienotas pie sistēmas, ir nepieciešams pāriet uz “Categories” sadalījumu, šajā vietā var arī uzspiest pogu “Create”, lai pievienotu jaunu kategoriju, vai “Edit”, lai apskatītu vai rediģētu jau eksistējošu kategoriju, kategorijas datus var arī eksportēt uzspiežot uz “Export” pogu:

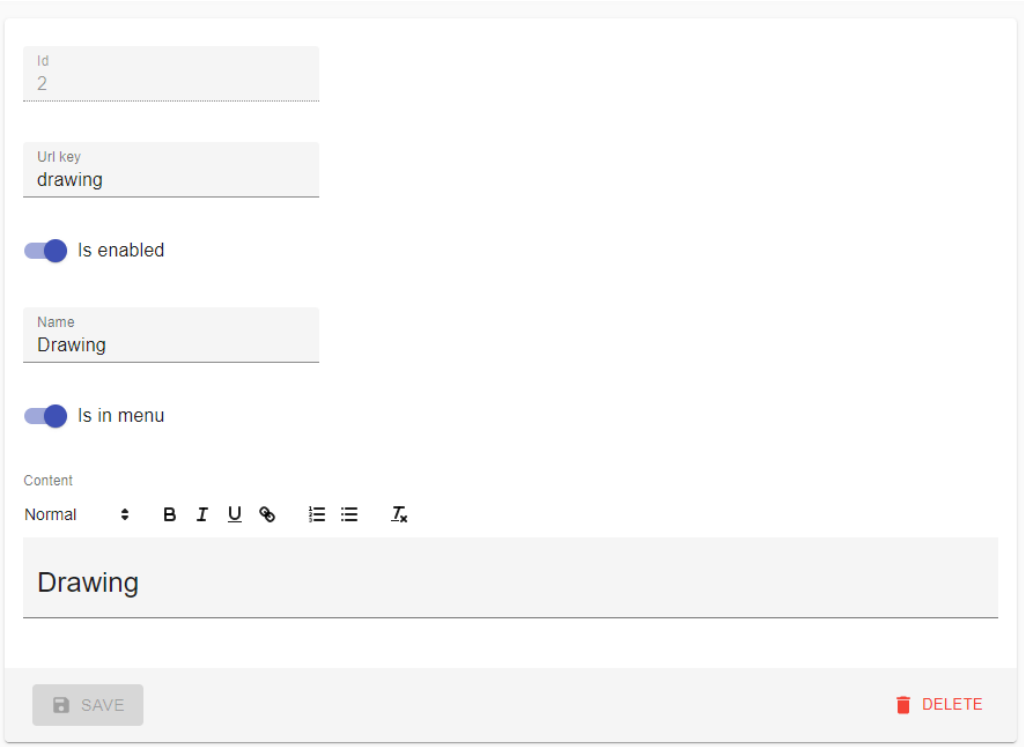


+ CREATE EXPORT

Id	Url key	Is enabled	Name	Is in menu	
1		✓	Root	✓	EDIT
2	drawing	✓	Drawing	✓	EDIT
3	reading	✓	Reading	✓	EDIT
4	pencils	✓	Pencils	✓	EDIT
5	pens	✓	Pens	✓	EDIT
6	regular_pencils	✓	Regular pencils	✓	EDIT
7	mechanical_pencils	✓	Mechanical pencils	✓	EDIT
8	ballpoint_pens	✓	Ballpoint pens	✓	EDIT
9	fountain_pens	✓	Fountain pens	✓	EDIT
10	books	✓	Books	✓	EDIT

29. att: Kategoriju saraksts

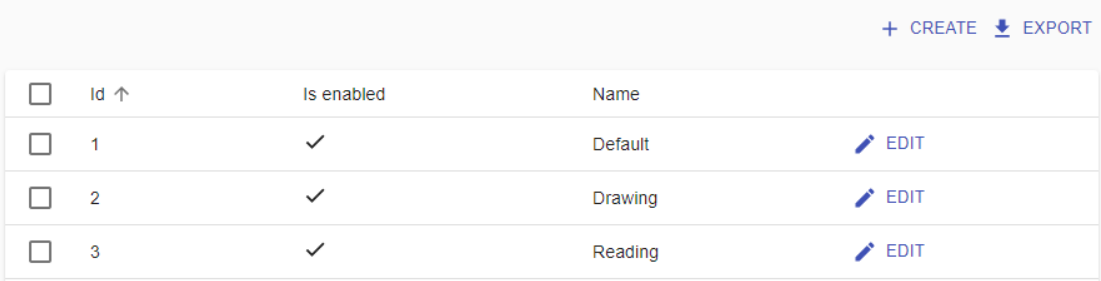
6.2 Pārejot vai nu uz “Create” vai uz “Edit”, parādīsies lauki, kuros vajadzēs ievadīt nepieciešamus datus par produktu, lai to varētu saglabāt. Kategoriju var arī izdzēst.



30. att: Kategorijas rediģēšana

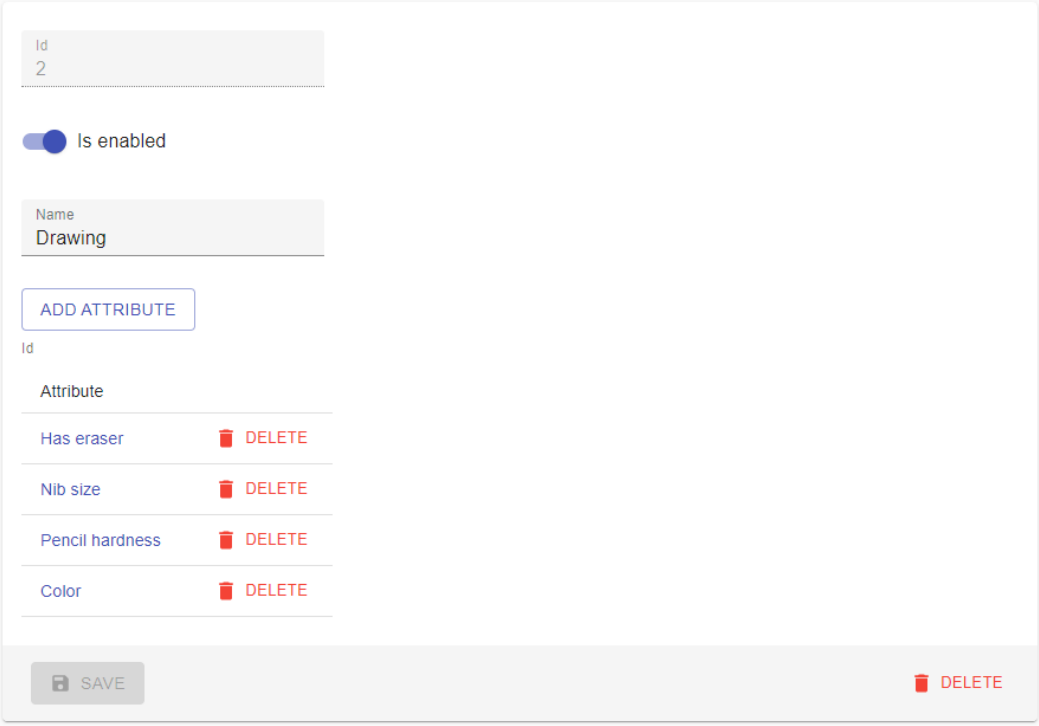
7 Atribūtu setu pārvalde

7.1 Lai redzētu, kuri atributu seti ir pievienoti pie sistēmas, ir nepieciešams pāriet uz “Attribute sets” sadalījumu, šajā vietā var arī uzspiest pogu “Create”, lai pievienotu jaunu produktu, vai “Edit”, lai apskatītu vai rediģētu jau eksistējošu produktu, atributa seta datus.



31. att: Atribūtu setu saraksts

7.2 Pārejot vai nu uz “Create” vai uz “Edit”, parādīsies lauki, kuros vajadzēs ievadīt nepieciešamus datus par atributu setu, lai to varētu saglabāt. Atribūtu setu var arī izdzēst.

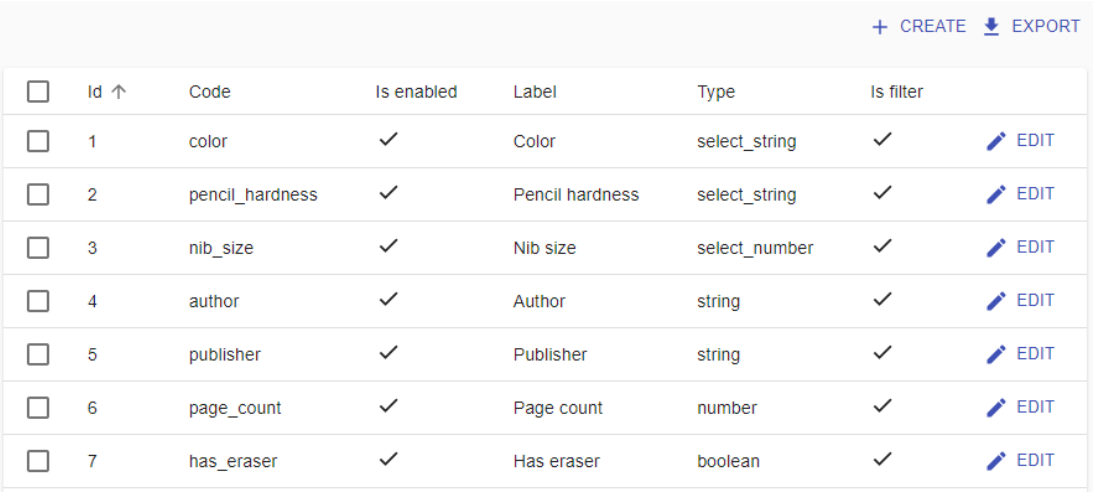


32. att: Atribūtu seta rediģēšana

8 Atribūtu pārvalde

8.1 Lai redzētu, kuri atribūti ir pievienoti pie sistēmas, ir nepieciešams pāriet uz “Attributes” sadalījumu, šajā vietā var arī uzspiest pogu “Create”, lai

pievienotu jaunu atribūtu, vai “Edit”, lai apskatītu vai rediģētu jau eksistējošu atribūtu.

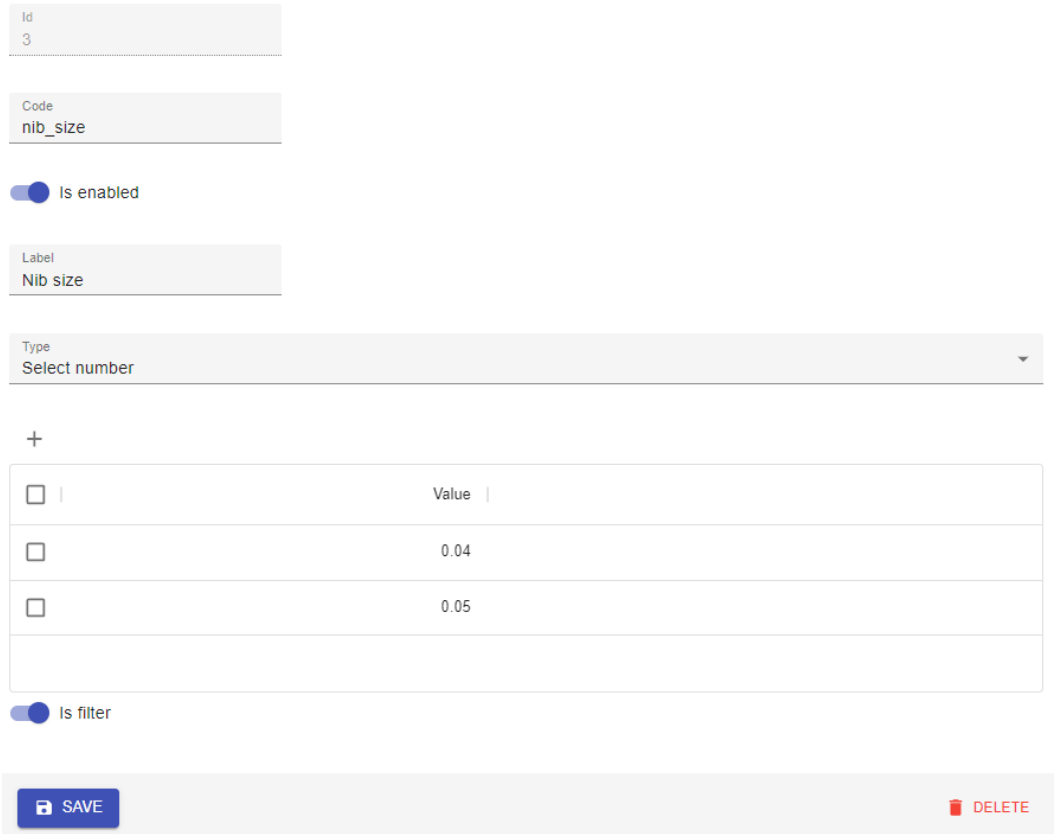


+ CREATE EXPORT

Id	Code	Is enabled	Label	Type	Is filter	
1	color	✓	Color	select_string	✓	EDIT
2	pencil_hardness	✓	Pencil hardness	select_string	✓	EDIT
3	nib_size	✓	Nib size	select_number	✓	EDIT
4	author	✓	Author	string	✓	EDIT
5	publisher	✓	Publisher	string	✓	EDIT
6	page_count	✓	Page count	number	✓	EDIT
7	has_eraser	✓	Has eraser	boolean	✓	EDIT

33. att: Atribūtu saraksts

8.2 Pārejot vai nu uz “Create” vai uz “Edit”, parādīsies lauki, kuros vajadzēs ievadīt nepieciešamus datus par atribūtu, lai to varētu saglabāt. Attribūtu var arī izdzēst.



Id
3

Code
nib_size

Is enabled

Label
Nib size

Type
Select number

+

Value
0.04
0.05

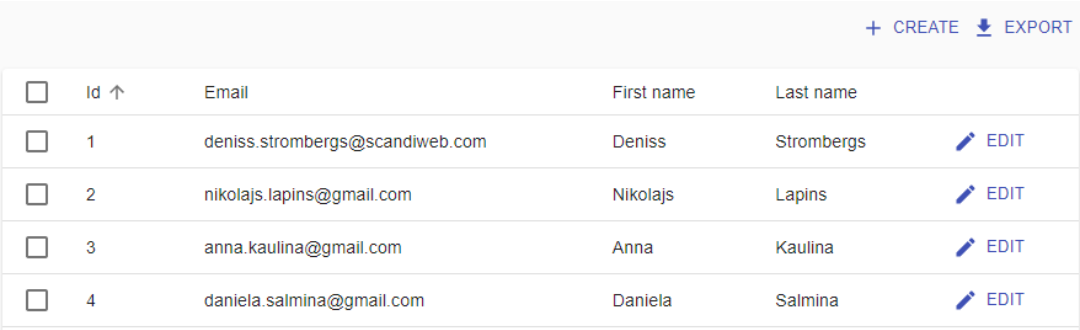
Is filter

SAVE DELETE

34. att: Atribūta rediģēšana

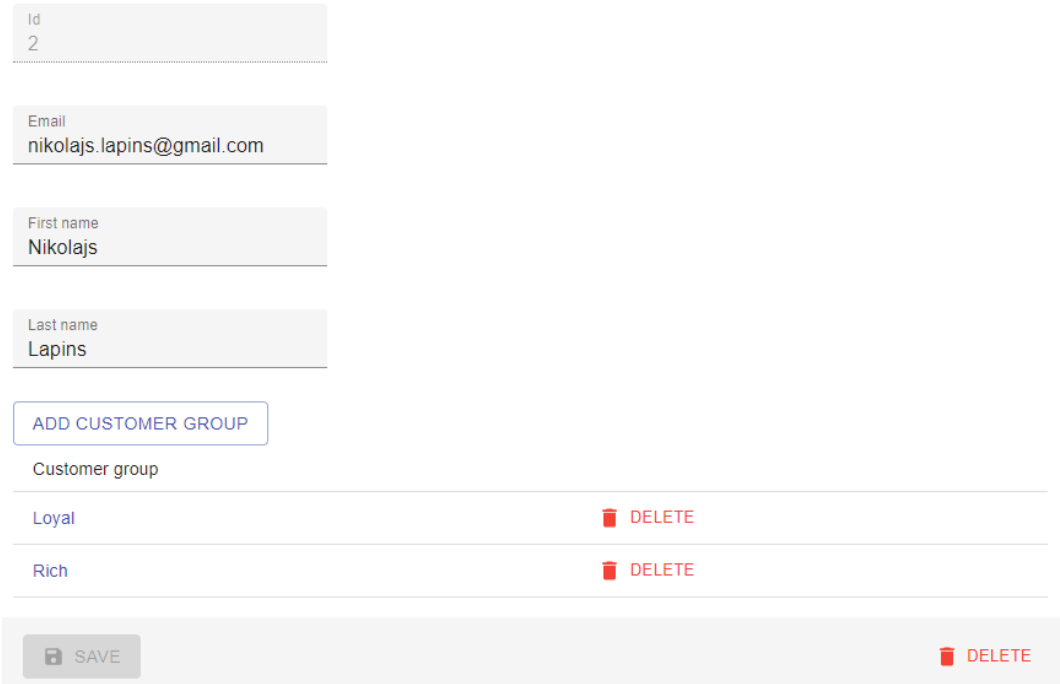
9 Klientu pārvalde

9.1 Lai redzētu, kuri klienti ir reģistrējušies pie sistēmas, ir nepieciešams pāriet uz “Customers” sadalījumu, šajā vietā var arī uzspiest pogu “Create”, lai pievienotu jaunu klientu, vai “Edit”, lai apskatītu vai rediģētu jau eksistējošu klientu.



35. att: Klientu saraksts

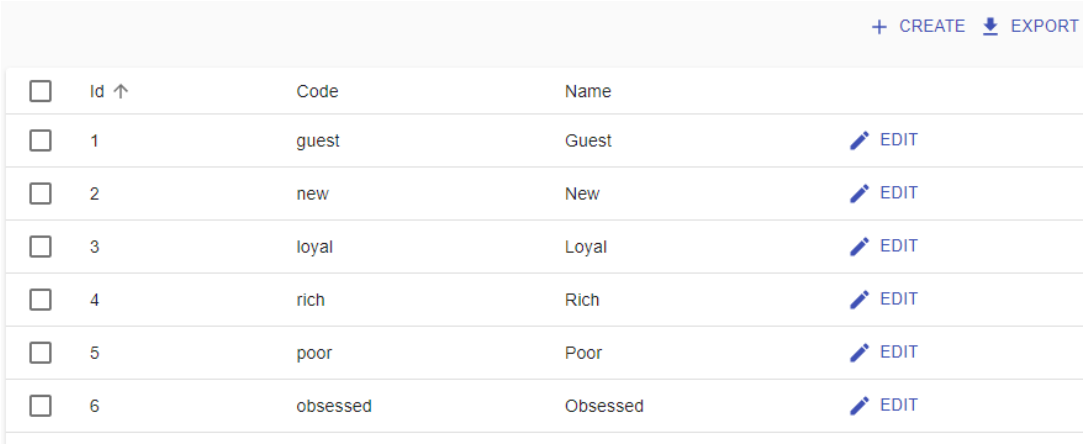
9.2 Pārejot vai nu uz “Create” vai uz “Edit”, parādīsies lauki, kuros vajadzēs ievadīt nepieciešamus datus par klientu, lai to varētu saglabāt. Klientu var arī izdzēst.



36. att: Klienta rediģēšana

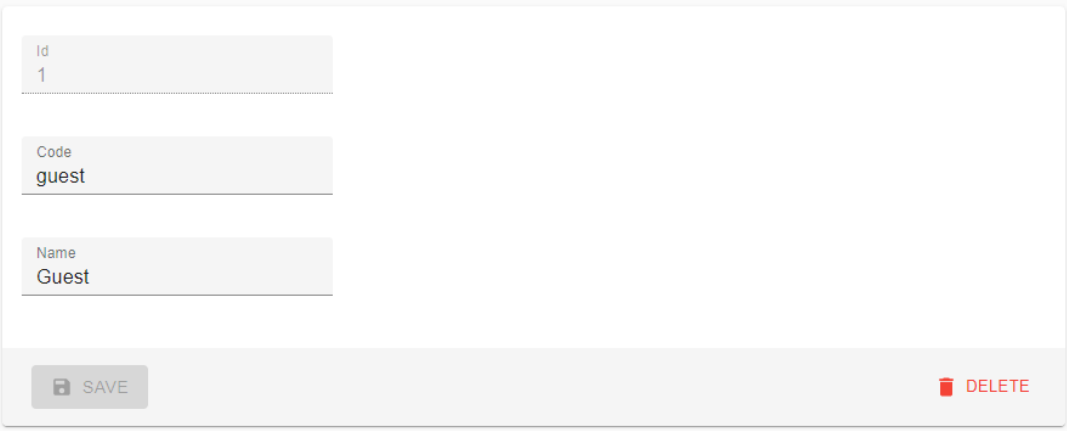
10 Klientu grupu pārvalde

10.1 Lai redzētu, kuras klientu grupas ir pievienotas pie sistēmas, ir nepieciešams pāriet uz “Customer groups” sadalījumu, šajā vietā var arī uzspiest pogu “Create”, lai pievienotu jaunu klientu grupu, vai “Edit”, lai apskatītu vai rediģētu jau eksistējošu klientu grupu.



37. att: Klienta grupu saraksts

10.2 Pārejot vai nu uz “Create” vai uz “Edit”, parādīsies lauki, kuros vajadzēs ievadīt nepieciešamus datus par klientu grupu, lai to varētu saglabāt. Klientu grupu var arī izdzēst.



38. att: Klienta grupas rediģēšana

11 Pasūtījumu pārskats

11.1 Lai redzētu, kuri pasūtījumi ir veikti, ir nepieciešams pāriet uz “Orders” sadalījumu, šajā vietā var arī uzspiest pogu “View”, lai apskatītu

vai rediģētu jau eksistējošu pasūtījumu, pasūtījuma datus var arī eksportēt uzspiežot uz “Export” pogu:

EXPORT

	Id	Reference	Date	Status	Total delivery	Total tax	Subtotal	Total	
	1	9V617LMMJST	4/20/2021	ordered	6.99	21	79	100	EDIT

39. att: Pasūtījumu saraksts

11.2 Pārejot vai nu uz “View”, parādīsies visa informācija par pasūtījumu, tā kopējām izmaksām, un saistītiem produktiem, kā arī informācija par pasūtītāju un tā adresi. Pasūtījumu var arī izdzēst.

Id
1

Reference
9V617LMMJST

Date
20/04/2021

Status
Ordered

Total delivery
6.99

Total tax
21

Subtotal
79

Total
100

Order items

Product	Quantity	Total tax	Subtotal	Total
Mechanical pencil (0.02mm)	2	31.5	15	18.15
Mechanical pencil (0.01mm)	2	4.2	20	24.2

40. att: Pasūtījuma pārskats

12 Rēķinu pārskats

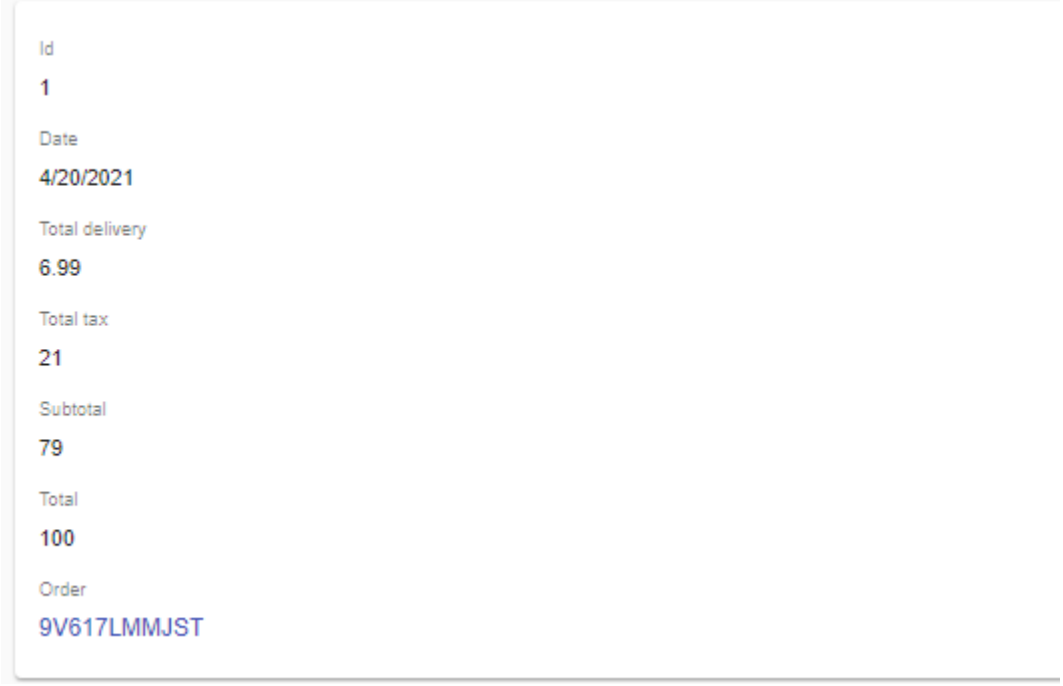
12.1 Lai redzētu, kuri rēķini tika ģenerēti, ir nepieciešams pāriet uz "Invoice" sadalījumu, šajā vietā var arī uzspiest pogu "View", lai apskatītu vai rediģētu jau eksistējošu rēķinu, rēkķina datus var arī eksportēt uzspiežot uz "Export" pogu:

EXPORT

Id	Date	Total delivery	Total tax	Subtotal	Total	
1	4/20/2021	6.99	21	79	100	SHOW

41. att: Rēķinu saraksts

12.2 Pārejot uz “View”, parādīsies visa informācija par rēķinu, tai skaitā, arī informācija par saistīto pasūtījumu, pasūtītāju un tā adresi.



42. att: Rēķina pārskats

13 Atsauksmju pārskats

13.1 Lai redzētu, kuras atsauksmes ir uzrakstītas, ir nepieciešams pāriet uz “Reviews” sadalījumu, šajā vietā var arī uzspiest pogu “View”, lai apskatītu vair rediģētu atsauksmi, atsauksmes datus var arī eksportēt uzspiežot uz “Export” pogu:

EXPORT

Id	Status	Date	Content	Rating	
1	accepted	5/5/2021	Absolute garbage, does not work.	1	EDIT
2	accepted	5/5/2021	Very nice product, I use it every day.	10	EDIT
3	accepted	5/5/2021	Yea its pretty good, but it has its own downsides.	7	EDIT
4	accepted	5/5/2021	Very nice, but I still would prefer to use the other version.	8	EDIT
5	accepted	5/5/2021	It is alright, not too good, not too bad.	5	EDIT
6	accepted	5/5/2021	The best thing I have ever used in my life.	10	EDIT
7	accepted	5/5/2021	Absolute garbage, does not work.	1	EDIT
8	accepted	5/5/2021	This product is not very good, do not recomend it.	5	EDIT

43. att: Atsauksmju saraksts

13.2 Pārejot uz “View”, parādīsies visa informācija par atsauksmi, kā arī par atsauksmes produktu un atsauksmes autoru. Atsauksmi var arī izdzēst.

Id
2

Status
Accepted

Date
05/05/2021

Title
Very nice

Content
Very nice product, I use it every day.

Rating
10

Customer
Nikolajs Lapins

SAVE DELETE

44. att: Atsauksmes pārskats

Klients

Klienta daļas apraksts:

1 Navigācija

45. att: Navigācijas panelis

1.1 Atvērot mājaslapu, lapas augšpusē varēs redzēt navigācijas paneli, tas satur galvenos navigācijas elementus: akaunts, grozs, meklēšana un izvēlne, ir jaatcerās šo elementus lasot nākamos punktus.

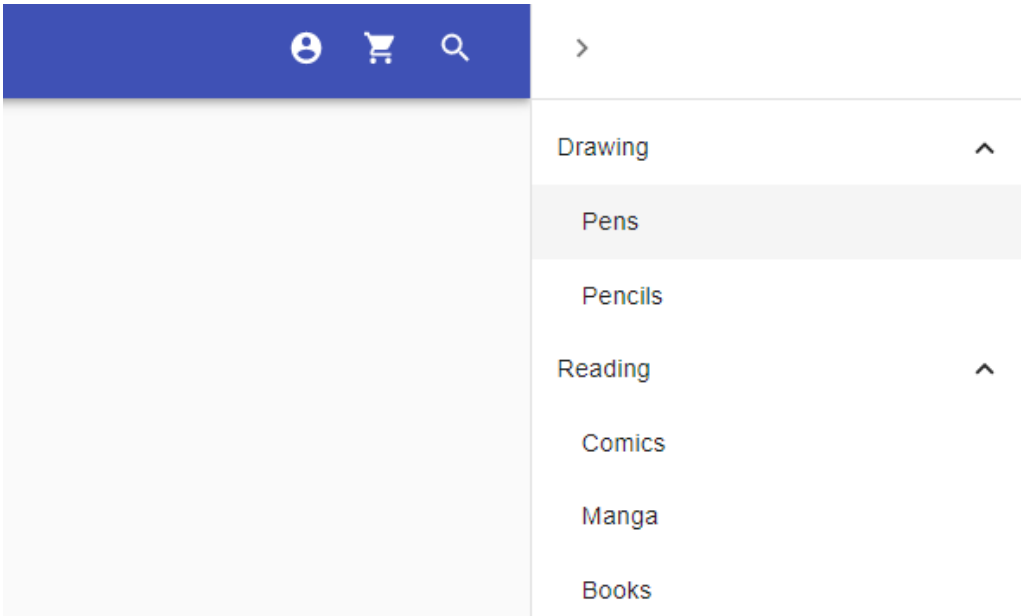
2 Produktu meklēšana

2.1 Meklēt produktus var divos veidos, vai nu pāriet uz konkrētu kategoriju, vai uzrakstīt meklējuma pieprasījumu

2.2 Lai pārietu uz konkrētu kategoriju ir nepieciešams atvērt izvēlni uzspiežot uz izvēlnes ikonu navigācijas panelī.

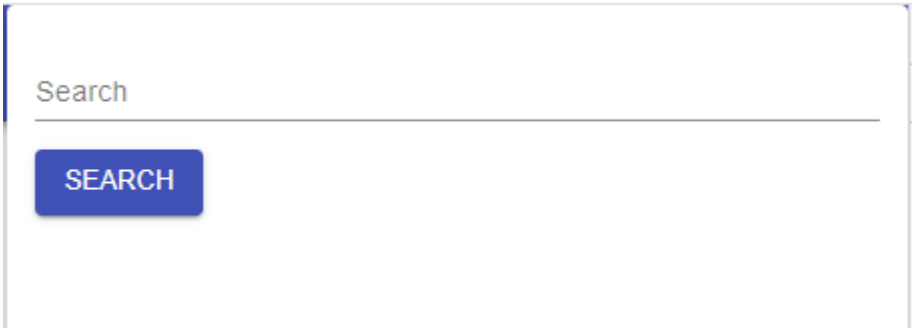
46. att: Izvēlnes ikona

2.3 Pēc izspiešanās parādīsies izvēlne, no kuras varēs pāriet uz konkrētu kategoriju:



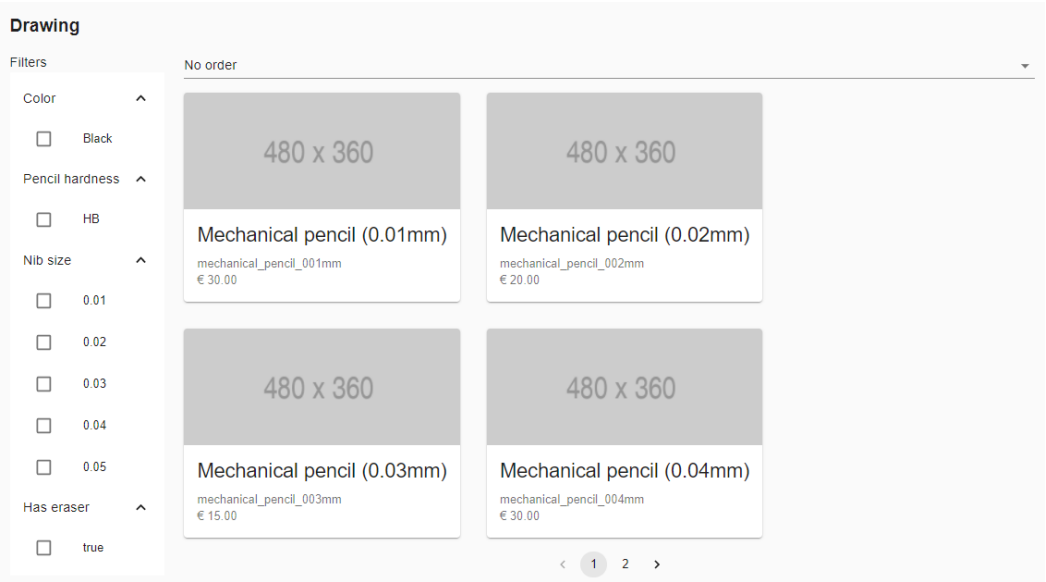
47. att: Izvēlne

2.4 Otrs variants ir atvērt meklēšanu un ievadīt tajā konkrētu pieprasījumu, pieprasījums meklē pēč lidzīgiem produktu nosaukumiem



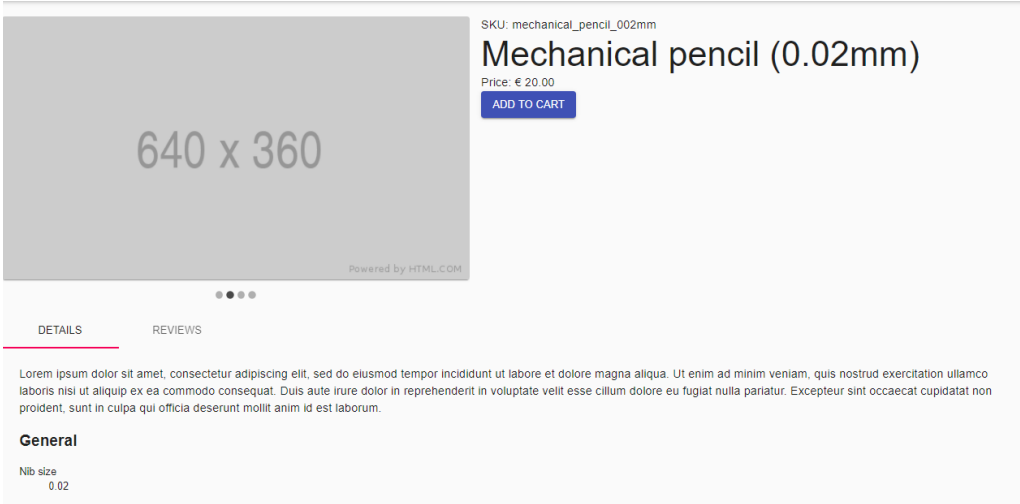
48. att: Meklēšana

2.5 Izvēloties vienu no pieminētām metodēm, var nonākt līdz produktu sarakstam, lai apskatītu konkrētu, produktu, ir nepieciešam uzspiest uz konkrētu produktu sarakstā.



3 Produktu apskatīšana

3.1 Pēc pāriešanas no produktu saraksta uz konkrētu produktu, parādīsies produktu lapa, kur būs visa informācija par produktus



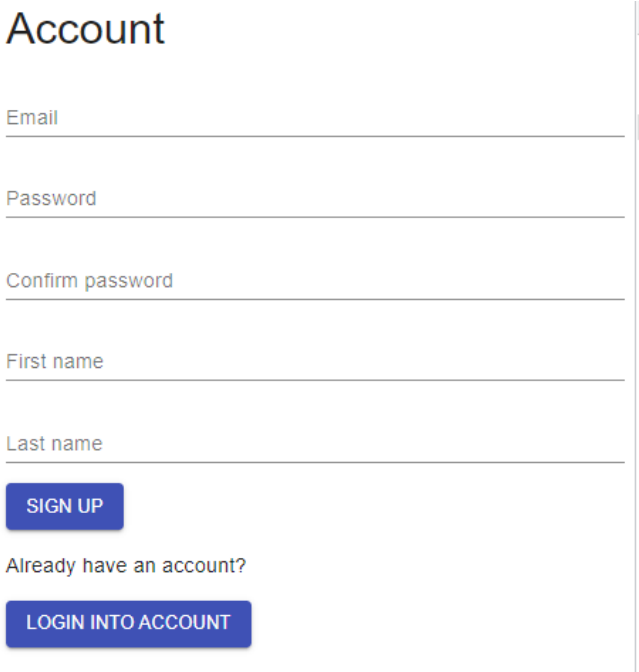
49. att: Produkta lapa

4 Lietotāja reģistrācija uz autorizācija

4.1 Lai piereģistrētu jaunu lietotāju ir nepieciešams atvērt akaunta logu, lai to izdarītu ir nepieciešams uzspiest uz akaunta ikonu:

50. att: Lietotaja ikona

4.2 Uzspiežot uz akaunta ikonu, atvērsies akaunta logs, kurā var vai nu piereģistrēties vai autorizēties jau eksistējoša akauntā

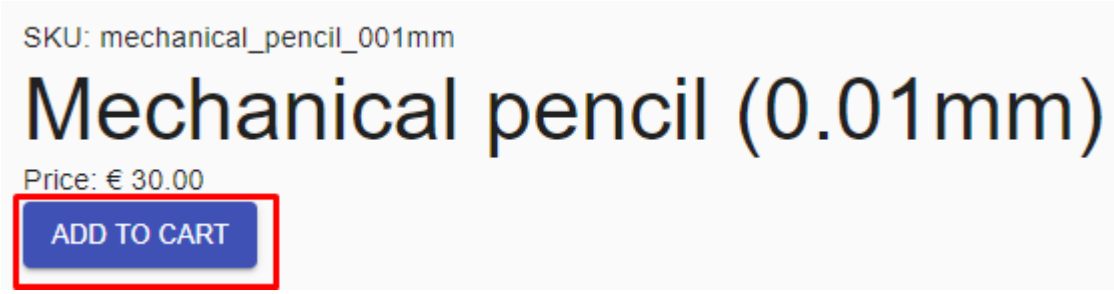


51. att: Akaunta reģistrēšana

4.3 Kad visi lauki ir aizpildīti, ir nepieciešams uzspiest uz “SIGN UP”, un ja viss tika ievadīts korekti, tad jauns lietotājs tiks piereģistrēts.

5 Groza rediģēšana

5.1 Lai pievienotu produktu grozām ir nepieciešams uzspiest uz “Add to cart”, lai to varētu izdarīt lietotājam ir jābūt autorizetam

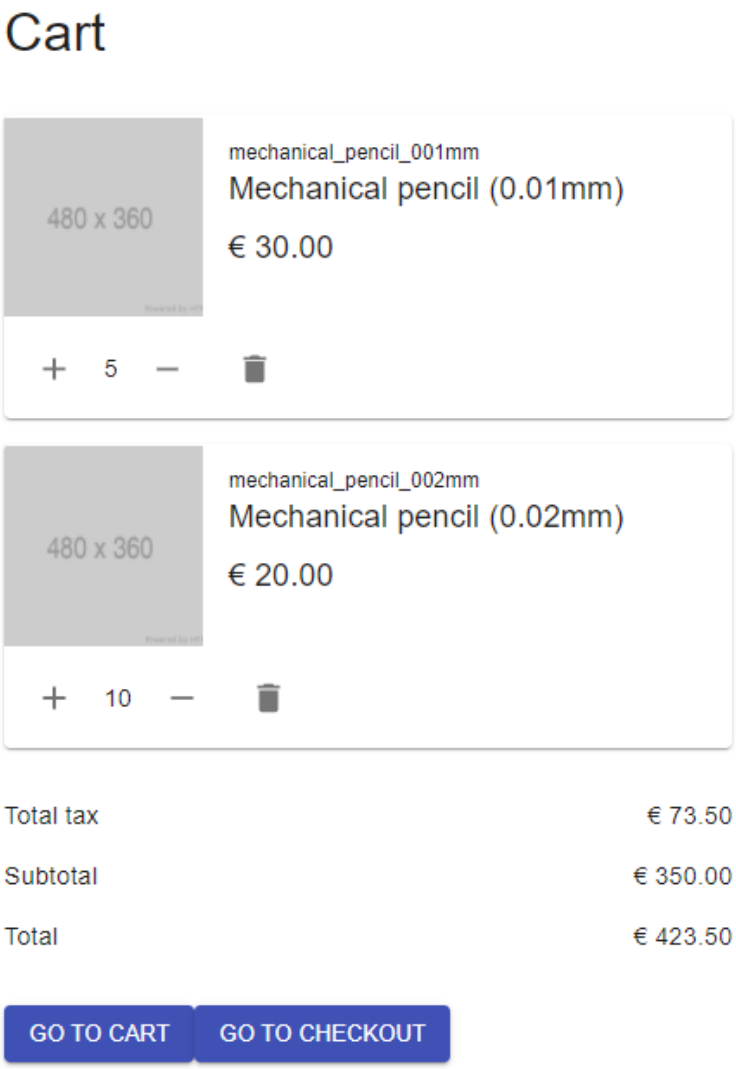


52. att: Produkta pievienošana grozām

5.2 Lai atvertu grozu ir nepieiciešams uzspiest uz groza ikona, kas atrodas navigācijas panelī

53. att: Groza ikona

5.3 Kad grozs atvēries varēs redzē tā saturu: pievienotu produktu sarakstu un cenu kopsavilkumu



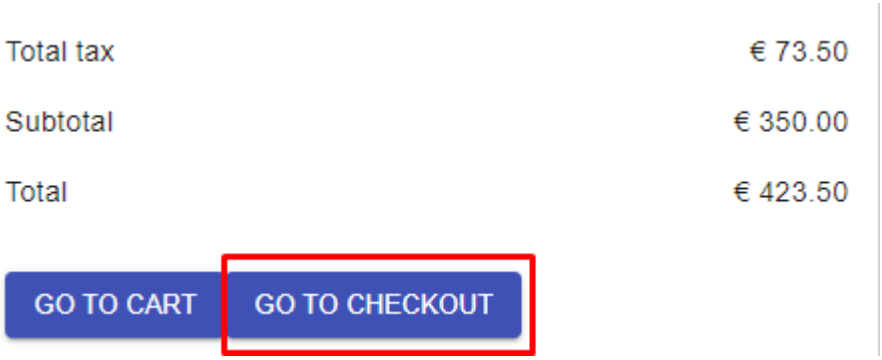
54. att: Groza saturs

5.4 Lai palielinātu produkta daudzumu ir nepieciešams uzspiest uz “+” pogu, lai samazinātu produkta daudzumu ir nepieciešams uzspiest uz “-” pogu, un lai izņemtu produktu no groza ir nepieciešams uzspiest uz pogu ar miskastes ikonu. Starp “+” un “-” pogām var redzēt produkta daudzumu.

55. att: Groza produkta pogas

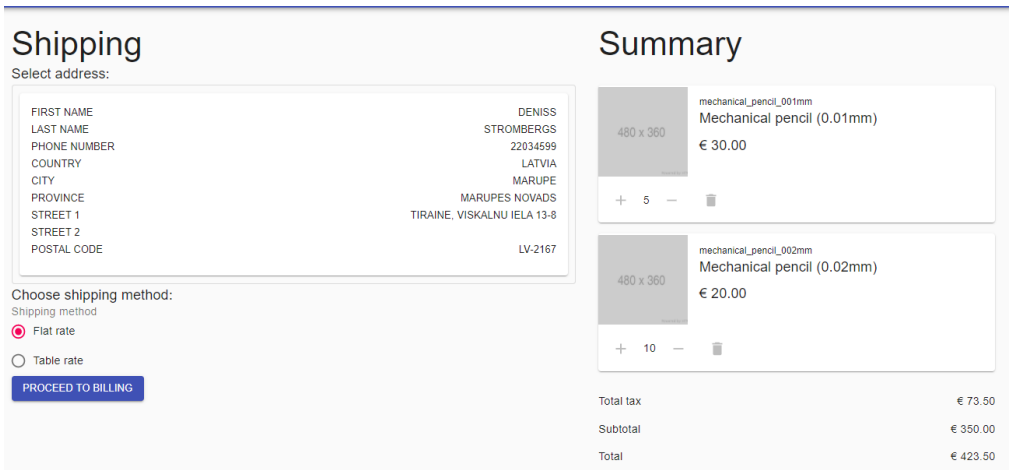
6 Pasūtījumu veikšana

6.1 Lai veiktu pasūtījumu ir nepieciešams pāriet uz “Checkout” lapu, lai to izdarītu ir nepieciešams uzspiest uz “GO TO CHECKOUT ”pogu, kas atrodas grozā zem kopsavilkumiem



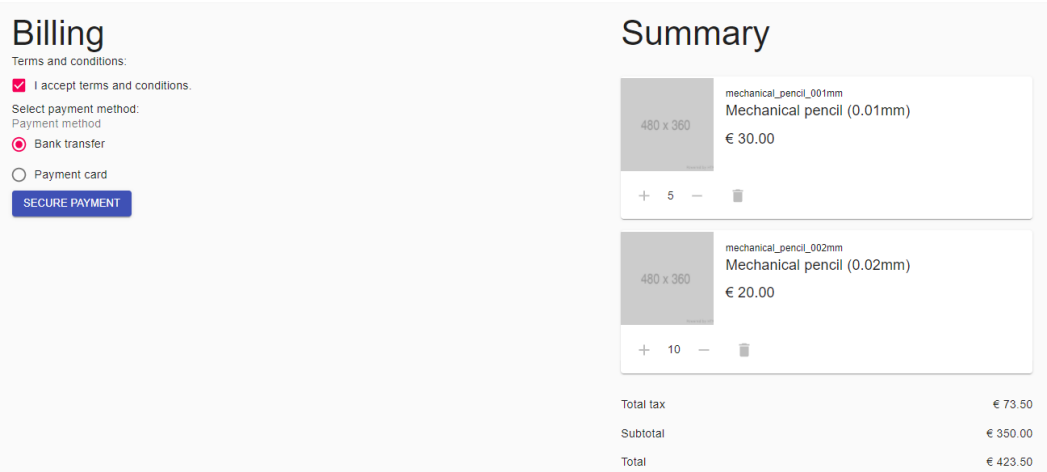
56. att: Pasūtījuma veikšana

6.2 “Checkout” lapā varēs redzēt pasutījuma kopsavilkumu, un būs nepieciešams vai nu ievadīt adresi vai izvēlēties jau eksistējošu adresi, un arī izvēlēties piegādes veidu, lai pārietu uz nākamo soli ir nepieciešams uzspiest uz “PROCEED TO BILLING” pogu



57. att: Pasutījuma piegāde

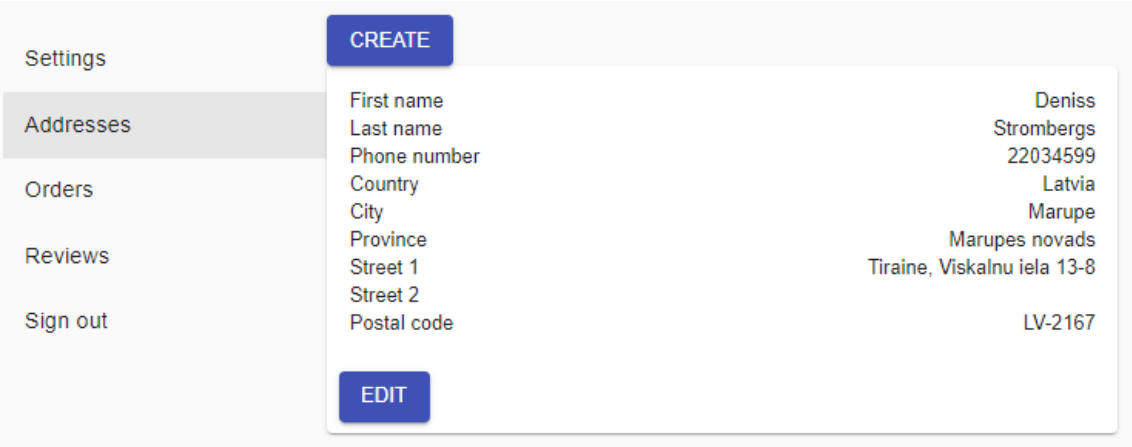
6.3 Nākamajā solī varēs arī varēs redzēt pasūtījuma kopsavilkumu, un vajadzēs izvēlēties pasūtījuma apmaksas veidu un akceptēt pasūtījuma noteikumus.



58. att: Pasūtījuma apmaksa

7 Adrešu pārvalde

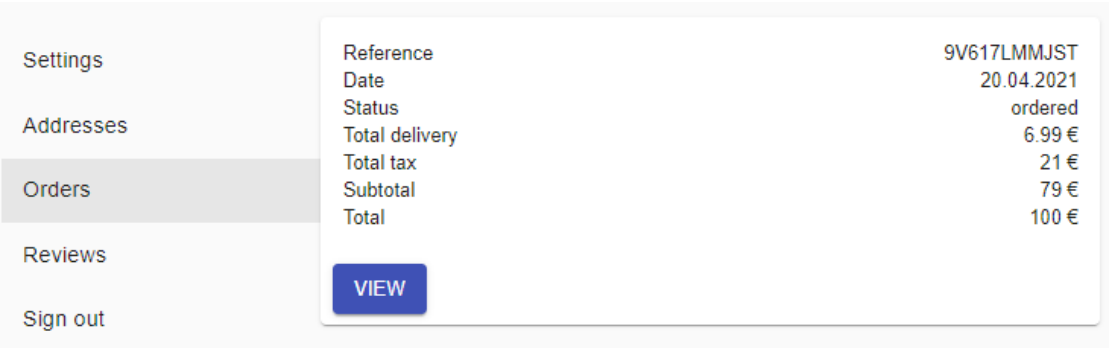
7.1 Adreses var pievienot, rediģēt un dzēst lietotāja akaunta lapā, adrešu sadaļā



59. att: Lietotāja adrešu pārvalde

8 Pasūtījumu pārvalde

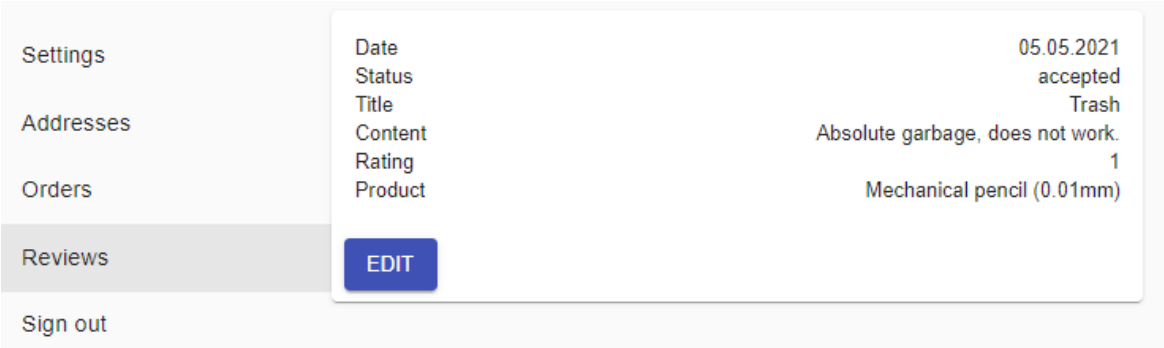
8.1 Pasūtījumus var pārskatīt akaunta lapā, pasūtījum sadaļa



60. att: Lietotāju pasūtījumu pārvalde

9 Atsauksmju pārvalde

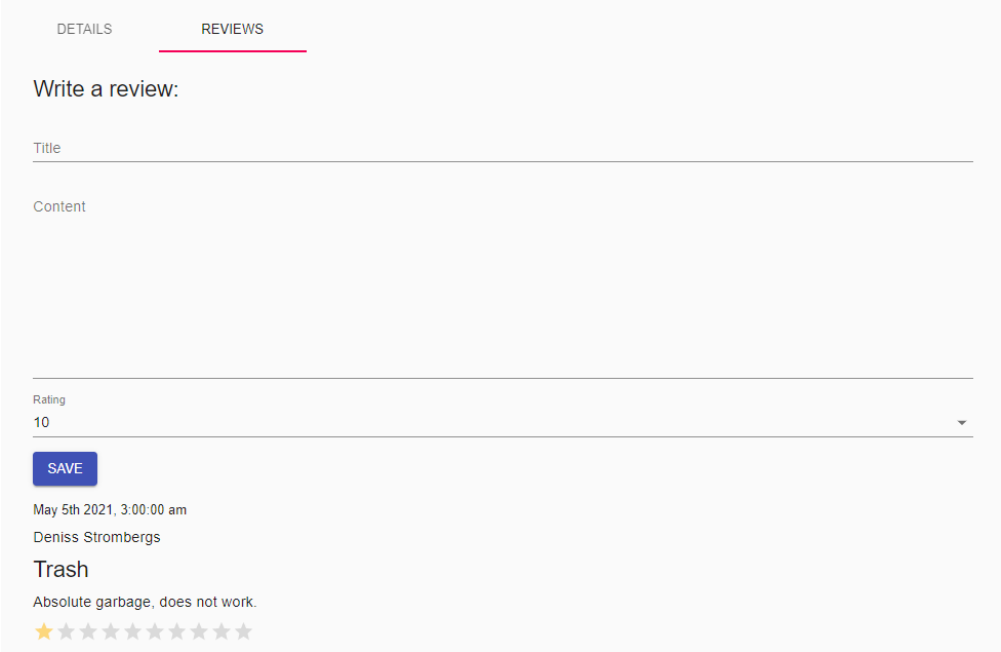
9.1 Atsauksmes par rediģēt un dzēst akaunta lapā, atsauksmju sadaļā



61. att: Lietotāja atsauksmju pārvalde

10 Atsauksmju pievienošana

10.1 Lai pievienotu jaunu atsauksmi produktam, ir nepieciešams pāriet uz produkta lapu un “Reviews” cilni, tur vajadzēs aizpildīt laukus, lai pievienotu atsauksmi



62. att: Atsauksmes pievienošana

6.4 Testa piemērs

Lai apskatītu produktu sarakstu, kā jau iepriekš bija pieminēts ir nepieciešams, pāriet uz “Products” sadaļu, un tur varēs izvēlēties vai nu izveidot jaunu produktu, vai rediģēt jau eksistējošu.

Produkta izveides process un produkta iestatījumi. Uzspiežot uz “Create” vai “Edit” pogu, parādīsies lauki, kurus ir nepieciešams rediģēt:

1 Galvenie iestatījumi

1.1 URL key - unikāls lapas identifikators, šajā lauka ir nepieciešams ievadīt, unikālu identifikatoru, kurš tiks izmantots, lai identificētu produkta lapu interneta veiklā.

1.2 SKU (Stock keeping unit) – (KGV) krājumu glabāšanas vienība, šajā laukā ir nepieciešams ievadīt unikālu identifikatoru, kurš identificēs pašu produktu, kā glabāšanas vienību

1.3 Is enabled - vai ir ieslēgts, šis lauks ir nepieciešams, lai varētu kontrolēt produkta redzamību veikalā, ja tas ir izslēgts, tad produkts nebūs pieejams pircējiem, un ja tas ir ieslēgts, tad būs.

1.4 Price - pamata cena, cena bez nekādām atlaidēm un nodokļiem.

1.5 Quantity - daudzums noliktavā, šis lauks norāda uz to cik

1.6 Special discount type - atlaides tips (daudzums, procentuāls), šis lauks norāda uz to, kāds atlaides tips tiek pielietots, lai izrēķinātu produkta atlaides cenu, ir divi tipi: procentuāls un uz daudzumu bāzēts, procentuāls noņēm no cenas kaut kādu daļu, bet daudzums atņēm konkrētu cenu. Lai nebūtu atlaides vispār, šo lauku var atstāt tukšu.

1.7 Special discount value - atlaides vērtība, šis lauks norāda uz to, cik liela ir atlaide, šis lauks ir atkarīgs no atlaides tipa.

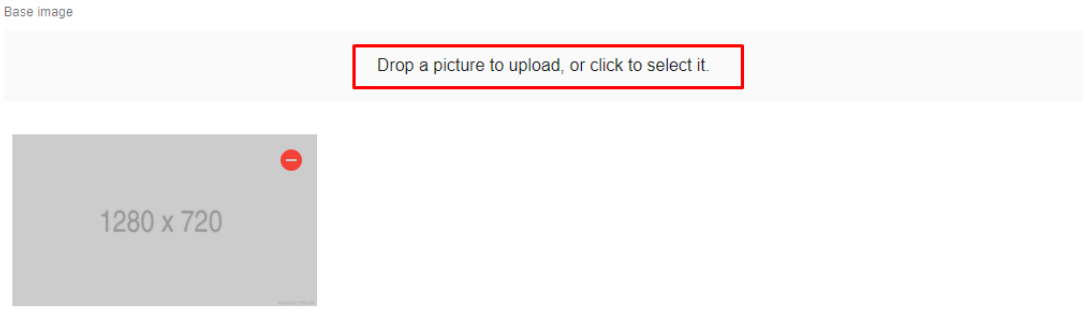
1.8 Special tax rate - nodokļu procents, šeit var norādīt produkta specifisku nodokļā likmi, pēc noklusējuma nodokļa likme atbilst 21%.

1.9 Short description - īss apraksts, šajā laukā var uzrakstīt īsu produkta aprakstu līdz 255 rakstu zīmēm.

1.10 Long description - garš apraksts, šajā laukā var uzrakstīt garu aprakstu, kur var iekļaut dažādas detaļas un attēlus.

1.11 Base image - galvenais attēls, šajā laukā ir iespējams norādīt attēlu, kurš tiks izmantots, kā pirmais attēls uz produkta lapas. Lai pievienot attēlu ir nepieciešams uzspiest uz “Drop a picture to upload, or click to select it.”.

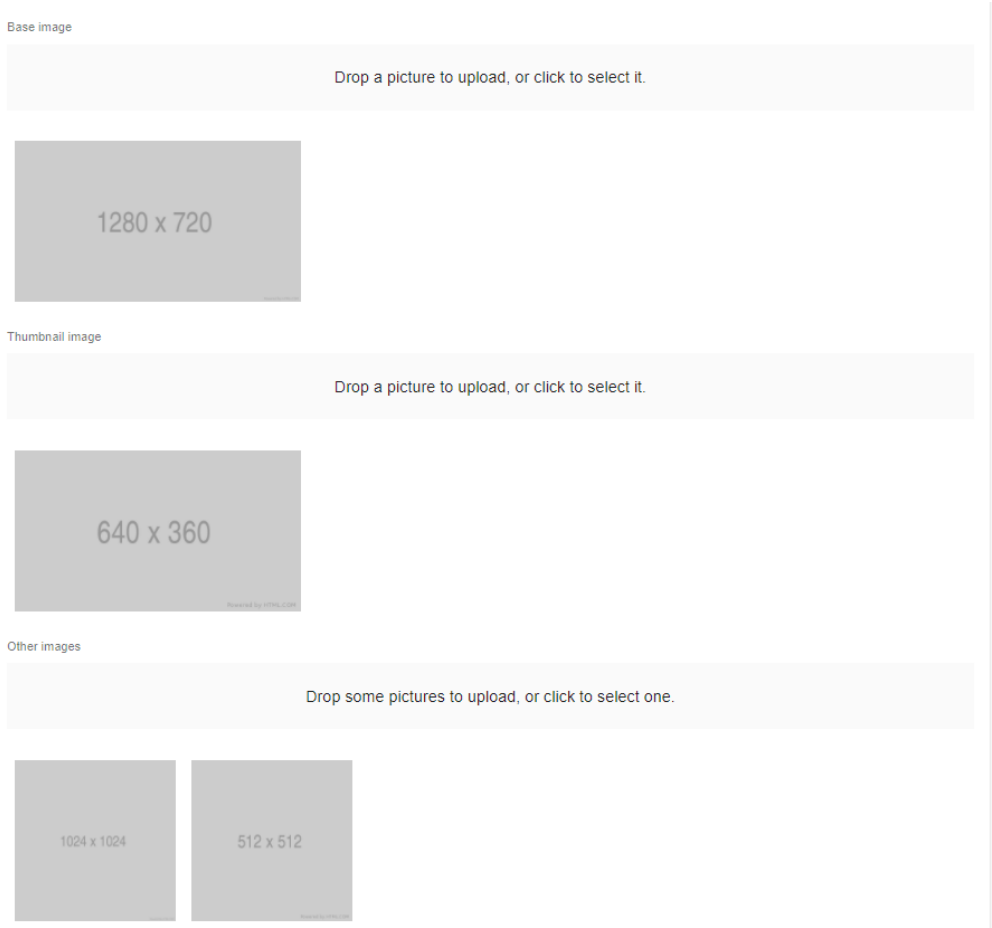
1.11.1 Pievienot attēlu var divos viedos, vai nu pārvietot attēlu uz attēla zonu, vai uzspiest uz “Drop a picture to upload, or click to select it.”, un tad atvērsies dialogs, kurā var izvēlieties attēlu. Var pievienot tikai attēlu, netiek atbalstīts nekāds cits failu tips.



63. att: Attēla pievienošana produktam

1.12 Thumbnail image - mazais attēls, šajā laukā ir iespējams norādīt attēlu, kurš tiks izmantots, lai attēlot atrastus produktu.

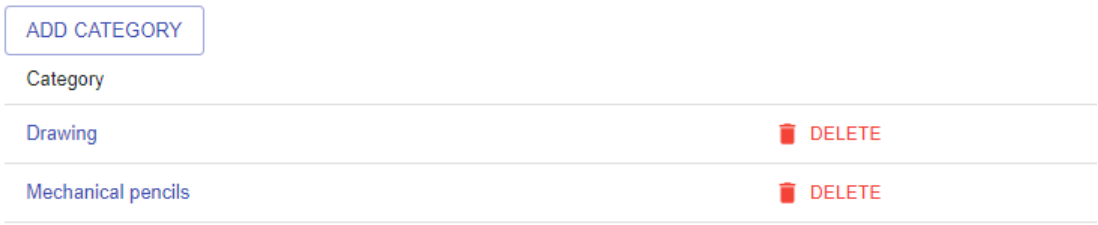
1.13 Other images - citi attēli, šajā laukā var norādīt visu pārējus attēlus, kuru būs parādīti uz produkta lapas



64. att: Produkta attēli

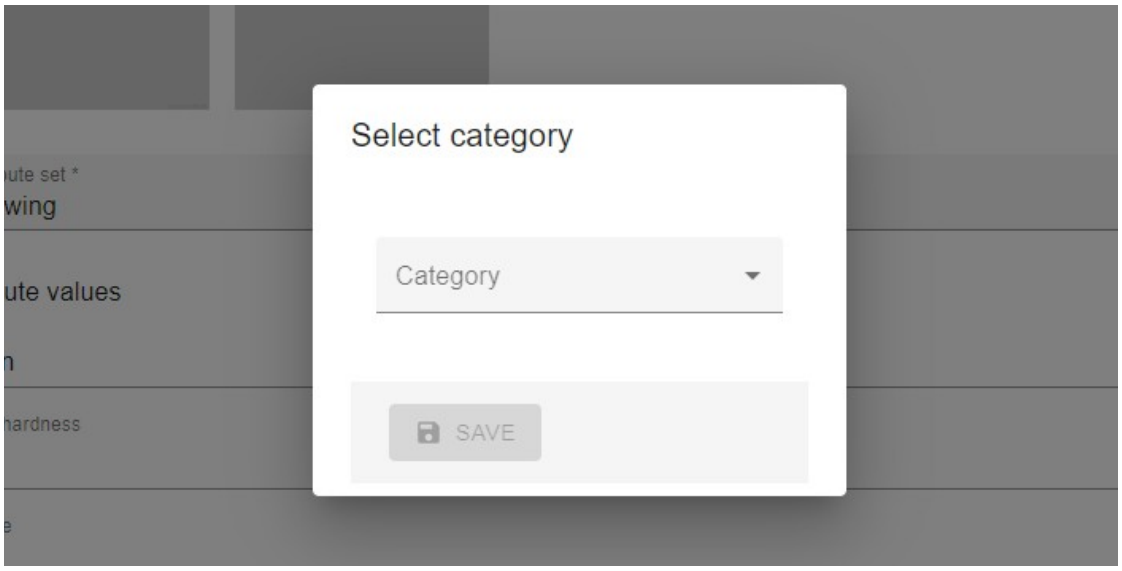
1.14 Attribute set - atribūtu sets, šāja laukā var izvēlēties konkrētu atribūtu setu, uz kura šī produkta atribūti tiks bāzēti, šis lauks ir nepieciešams, lai varētu iestatīt produkta atribūtus.

1.15 Category – kategorija, šajā laukā var pievienot visas kategorijas, pie kurām šis produkts pieder, lai pievienotu kategoriju ir nepieciešams uzspiest uz "ADD CATEGORY". Viens svarīgs nosacījums par, kuru ir jāzina, ir tas, ka kategorijas var pievienoti tikai pēc tam kad produkts ir izveidots, tāpēc no sākuma vajag saglabāt produktu.



65. att: Kategorijas pievienošana

1.16 Tad atvērsies dialogs, kurā var izvēlēties vienu no iespējamām kategorijām



66. att: Kategoriju pievienošana produktam

2 Atribūtu vērtības

2.1 Pēc izvēlētā atribūtu seta, parādīsies atsevišķi lauki, kuros varēs ievadī nepieciešamas produktu atribūtu vērtības, vai izvēlēties tās no iespējamiem variantiem.

2.2 Viens svarīgs nosacījums par, kuru ir jāzina, ir tas, ka atribūtu vērtība var ievadīt tikai pēc tam kad produkts ir izveidots un attiecīgs atribūtu sets tika izvēlēts, tāpēc no sākuma vajag saglabāt produktu.

67. att: Attribūtu vērtības

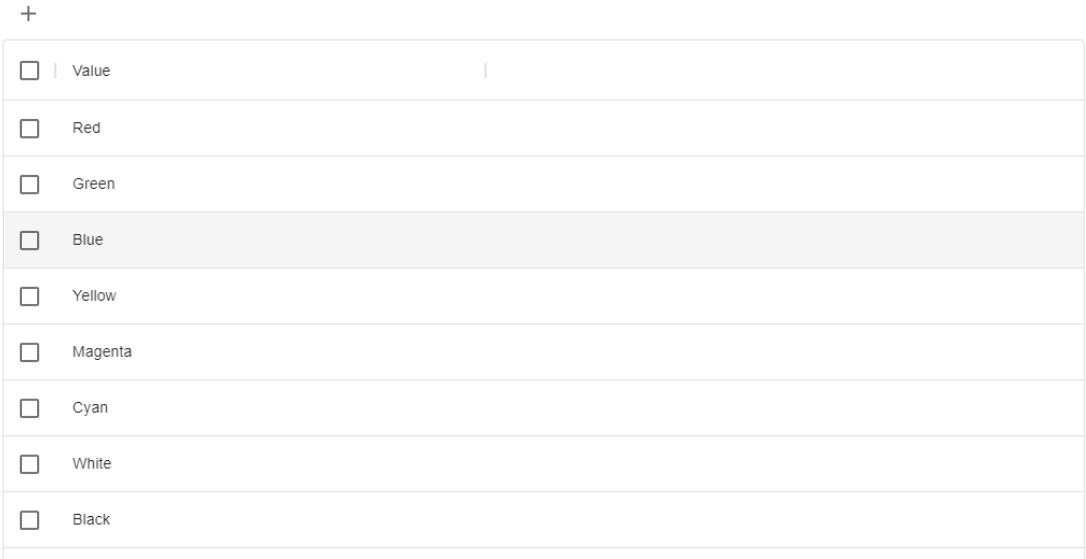
3 Atribūti

3.1 Lai apskatītu produktu atribūtu sarakstu, kā jau iepriekš bija pieminēts ir nepieciešams, pāriet uz "Attributes" sadaļu, un tur varēs izvēlēties vai nu izveidot jaunu atribūtu, vai rediģēt jau eksistējošu.

3.2 Šie ir galvenie atribūta lauki

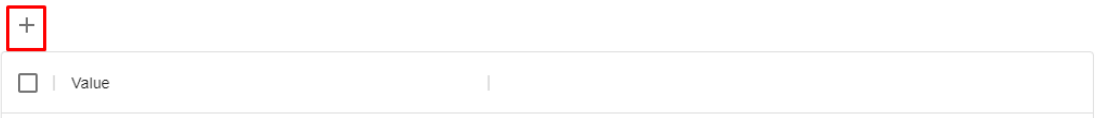
68. att: Galvenie atribūta lauki

3.3 Tad ir viens speciāls lauks – atribūta tips, kas nosaka kāds datu tips tajā būs ievadāms, šobrīd ir 5 datu tipi: Bula vērtība, skaitlis, simbola virkne, skaitļu saraksts, simbola virkņu saraksts. Pirmie trīs datu tipi ir parastie, bet saraksta tipi dod iespēju izvedot sarakstu no tiem datu tipiem. Piemēram var izveidot krāsu sarakstu, kā ir parādīts piemērā.



69. att: Atribūtu opcijas

3.4 Lai pievienotu jaunu atribūta opcija ir nepieciešams uzspiest uz “+”, tad tiks pievienots jauns ieraksts, un kamēr, tas ieraksts netiks aizpildīts, nevarēs pievienot vēl vienu vai saglabāt, jo visām vērtībām ir jābūt saglabāta.



70. att: Atributu opcijas pievienošana

3.5 Lai rediģētu konkrētu atribūta opcija tad ir nepieciešams uztaisīt dubultklikšķi uz atribūta opcijas vērtības lauka, tas tiks saglabāts tikai tad, kad tam būs piešķirta unikāla vērtība, atribūtu opcijas nevar atkārtoties

71. att: Atribūta opcijas vērtības rediģēšana

NOBEIGUMS

Kvalifikācijas darbā izvirzītais mērķis ir daļēji sasniegts. Ir izveidota interneta veikal sistēma līdzīgi, kuru ar dažiem uzlabojumiem varētu pat izmantot par bāzi, kaut kādai konkrētai interneta veikala sistēmas implementācijas.

Darbā izvirzītie uzdevumi ir nevar būt īsti novērtēti uzreiz, jo sistēmas princips, ir tāds, ka tai ir jākalpo par kaut kādu konkrēta risinājuma sākuma punktu, un tikai ar laiku varētu izprast, vai tāda sistēma ir paistam piemērota interneta veikalu veidošanai.

Šis risinājums ir ļoti ierobežots, un tam ir vairāki trūkumi, kurus būtu nepieciešams mazināt, lai šī sistēma varētu konkurēt ar jau eksistējošam sistēmām, piemēram, tādām kā, ”Magento 2”, ”Shopify”, ”Big Commerce” un citām. Šī sistēma nenodrošina nekādu reālu maksājuma veidu, nav SEO optimizācijas, un nav vairāku citu elementu. Vispār šī sistēma ir ļoti jauna un ar lieliem ierobežojumiem.

Taču neskatoties uz visiem trūkumiem, vēl ar dažiem uzlabojumiem, šī sistēma varētu būt spējīga nodrošināt paistam reālu interneta veikala darbību mazajos izmēros, un varbūt pat lielajos izmēros.

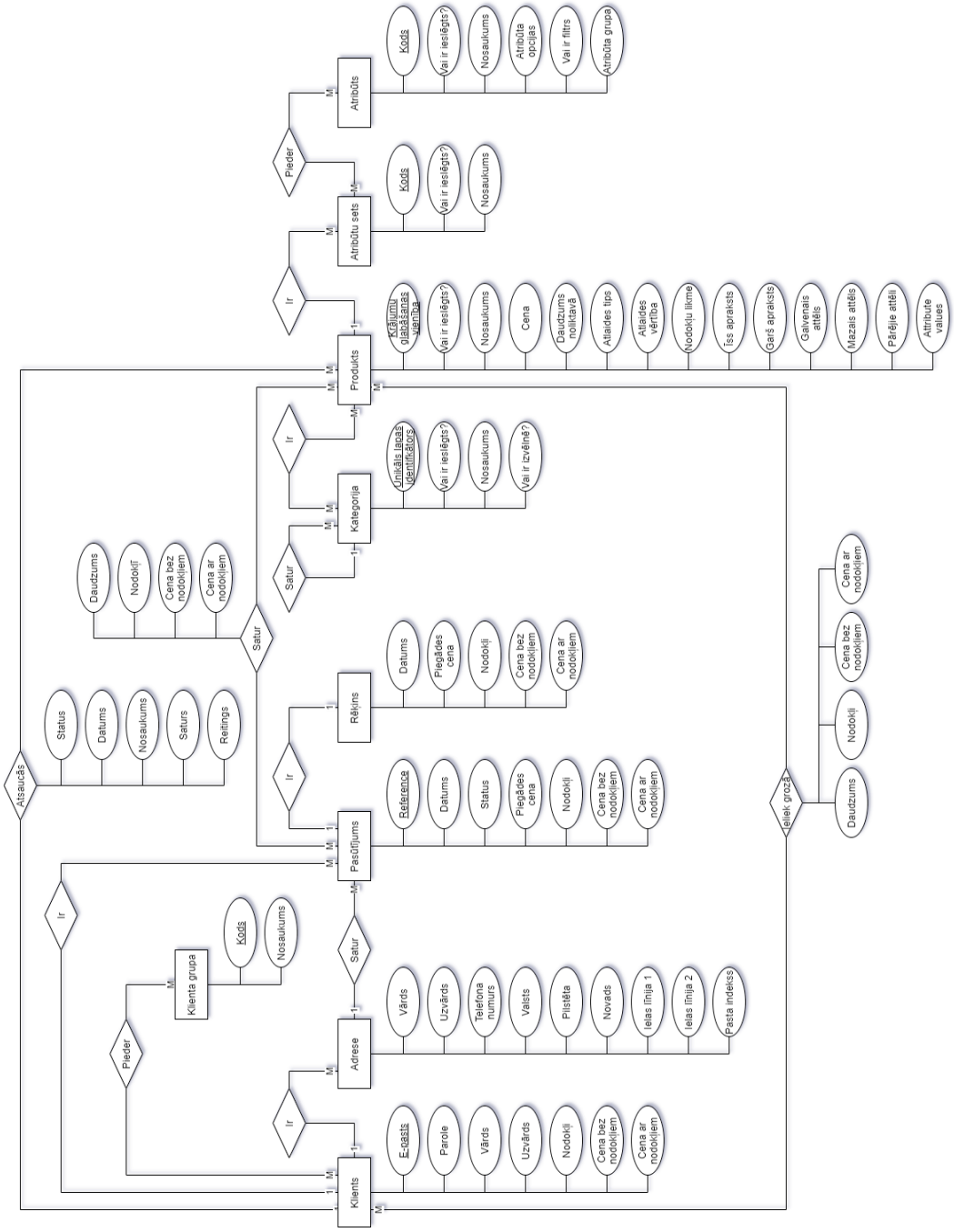
INFORMĀCIJAS AVOTI

1. https://marmelab.com/react-admin/Readme.html
2. https://www.postgresql.org/docs/9.4/datatype-json.html

PIELIKUMI

1. pielikums

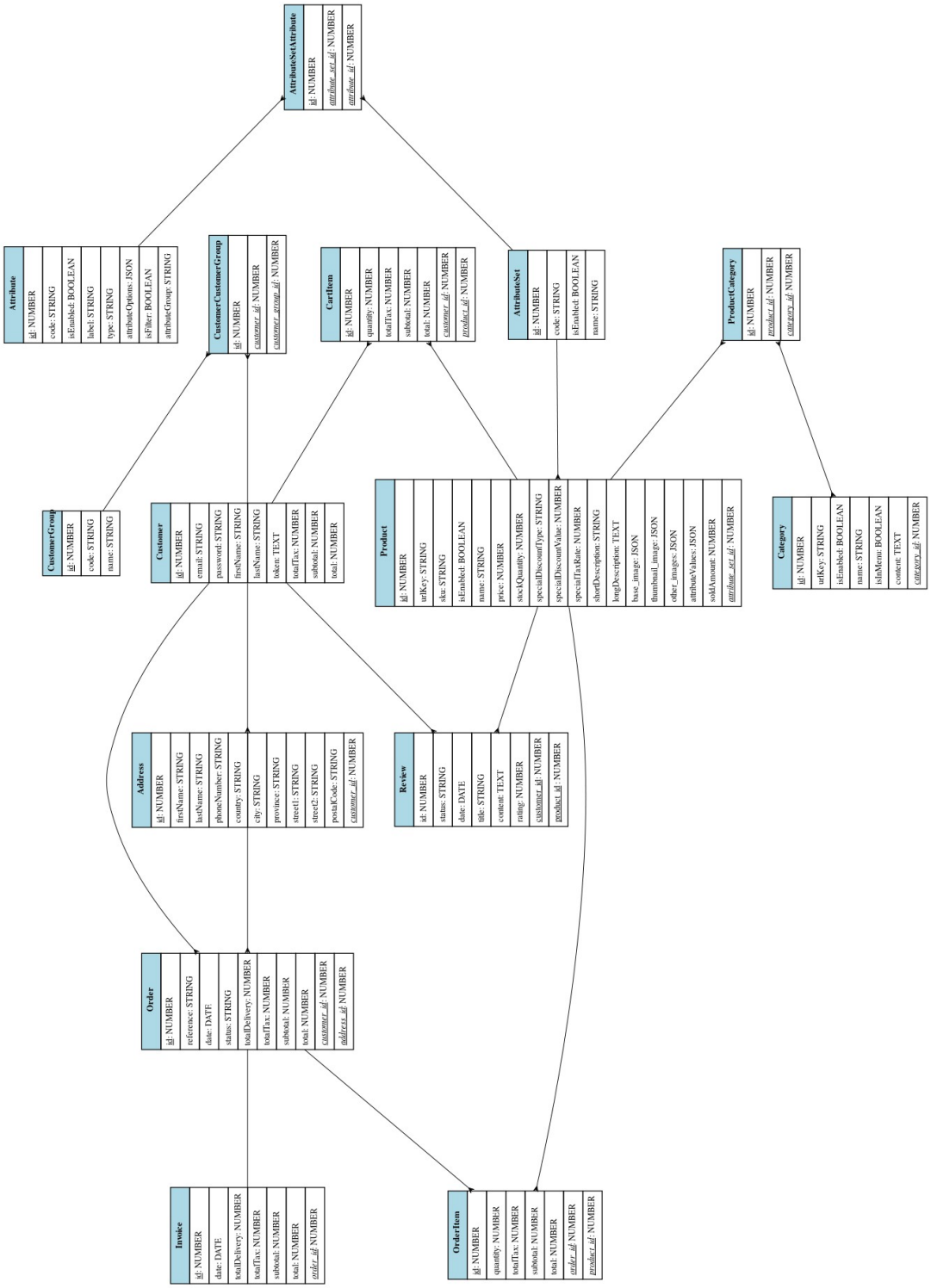
ER diagramma



72. att: ER diagramma

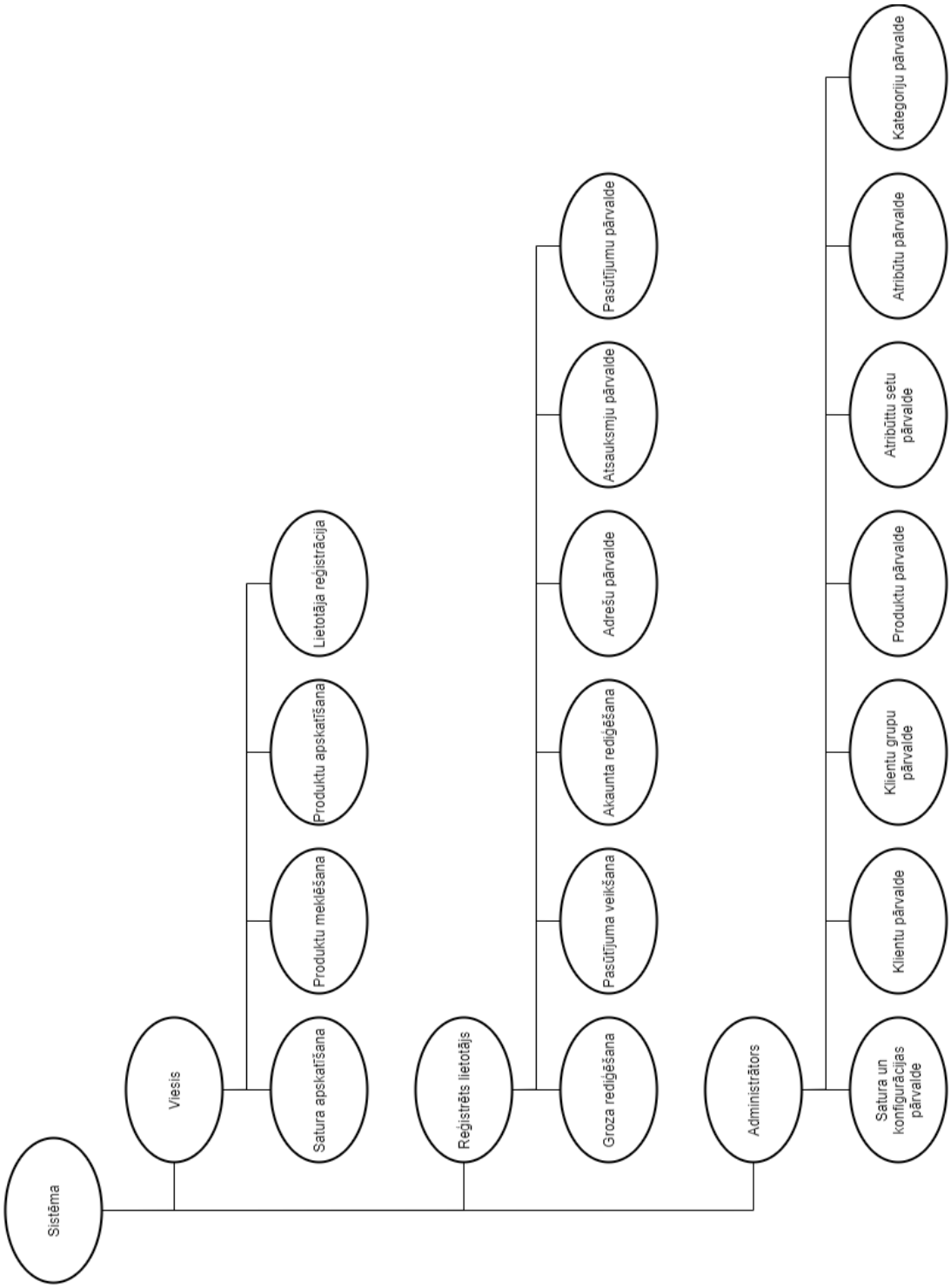
2. pielikums

Fiziska struktūra



73. att: Fiziska struktūra

Sistēmas arhitektūra



74. att: Sistēmas arhitektūra