**RĪGAS VALSTS TEHNIKUMS**

DATORIKAS NODAĻA

Izglītības programma: Programmēšana

**PRASĪBU DOKUMENTS**

**“Tīmekļa lietotne - Skriešanas apavu interneta veikals”**

Paskaidrojošais raksts \_\_\_ lpp.

Audzēknis: Kristaps Fedosejevs

Prakses vadītājs: Ilona Demčenko

Nodaļas vadītājs: Normunds Barbāns

**Rīga 2023**

**Saturs**

[Ievads 4](#_Toc153400294)

[1. UZDEVUMA NOSTĀDNE 5](#_Toc153400295)

[1.1. Lietojumgadījuma modelis 6](#_Toc153400296)

[2. PRASĪBU SPECIFIKĀCIJA 7](#_Toc153400297)

[2.1. Ieejas un izejas informācijas apraksts 7](#_Toc153400298)

[2.1.1. Ieejas informācijas apraksts 7](#_Toc153400299)

[2.1.2. Izejas informācijas apraksts 8](#_Toc153400300)

[2.2. Funkcionālās prasības 8](#_Toc153400301)

[2.3. Nefunkcionālās prasības 10](#_Toc153400302)

[3. UZDEVUMA RISINĀŠANAS LĪDZEKĻU IZVĒLES PAMATOJUMS 12](#_Toc153400303)

[4. PROGRAMMATŪRAS PRODUKTA MODELĒŠANA UN PROJEKTĒŠANA 13](#_Toc153400304)

[*4.1.1* *Sistēmas arhitektūra* 13](#_Toc153400305)

[**4.1.2. Entītiju relāciju datu modelis (ER-diagramm)** 14](#_Toc153400306)

[4.2. Funkcionālais sistēmas modelis 16](#_Toc153400307)

[**4.2.1. Datu plūsmu modelis** 16](#_Toc153400308)

[5. DATU STRUKTŪRAS APRAKSTS 19](#_Toc153400309)

[6. SECINĀJUMS 22](#_Toc153400310)

[7. IZMANTOTIE AVOTI 23](#_Toc153400311)

# Ievads

Mūsdienās ir ļoti svarīgi izmantot efektīvas risinājumu sistēmas, kas spēj atbilst mūsdienu prasībām un uzdevumiem. Šajā kontekstā tiek veidots jauns sporta apavu interneta veikals "Quick Step" – inovatīva platforma, kuras mērķis ir pārveidot un uzlabot sporta apavu iegādes pieredzi.

Šī sistēma tiek izstrādāta, ņemot vērā plašas vajadzības un izaicinājumus, ar kuriem saskaras sporta apavu nozares patērētāji. Bez šīs sistēmas situācija, kurā cilvēki meklē piemērotus sporta apavus, būtu ļoti apgrūtināta un nepārskatāma. Lai risinātu šo problēmu, "Quick Step" piedāvās ātrus, ērtus un personalizētus risinājumus sporta apavu iegādei tiešsaistē.

Savā būtībā "Quick Step" ir nākotnes risinājums, kas pārmainīs veidu, kā cilvēki iegādājas sporta apavus. Tas radīs revolūciju ne tikai pirkumu procesā, bet arī piedāvās inovatīvus veidus, kā klienti var sajust un pielāgot savu apavu izvēli. Šī platforma būs ne tikai vienkāršs interneta veikals, bet gan integrēts risinājums, kas uzlabos sporta apavu iegādes pieredzi visiem mūsu klientiem.

Savukārt mērķauditorija – cilvēki ar aktīvu dzīvesveidu, kuriem svarīgi ne tikai izskats, bet arī ērtums un kvalitāte. "Quick Step" piedāvās ne tikai augstas kvalitātes sporta apavus, bet arī piedāvās personalizācijas iespējas, lai katrs pircējs justos īpašs un apmierināts ar savu izvēli.

Šajā jaunajā nākotnes sistēmā slēpjas potenciāls mainīt veidu, kā cilvēki izvēlas un iegādājas sporta apavus tiešsaistē. "Quick Step" – tas ir solis uz priekšu, piedāvājot inovatīvus risinājumus, kas atbilst mūsdienu prasībām un padara sporta apavu iegādi par vienkāršu, ērtu un izbaudāmu pieredzi.

# UZDEVUMA NOSTĀDNE

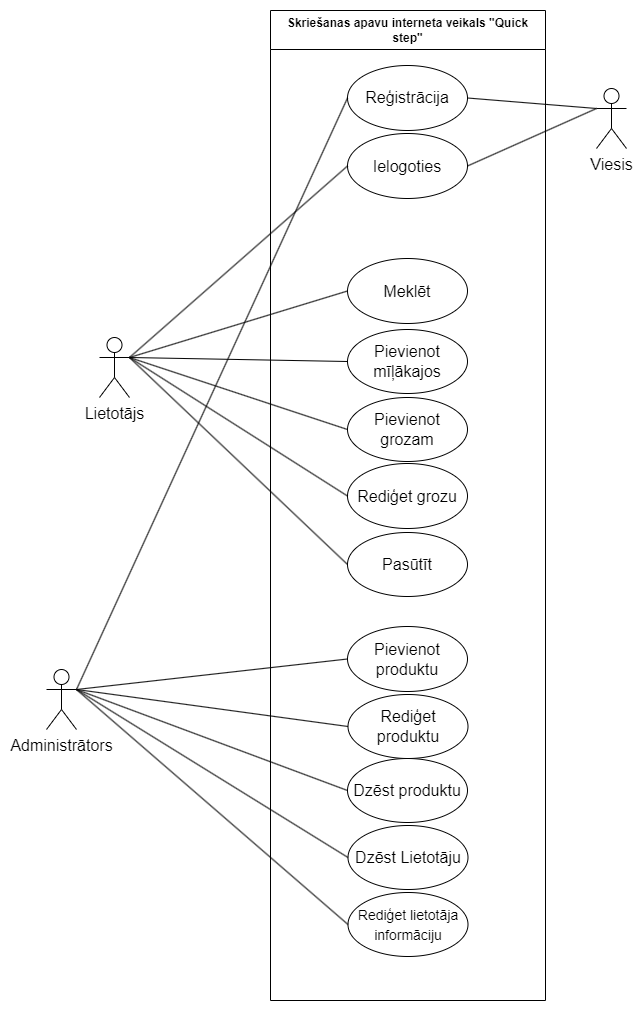
Kvalifikācijas darba uzdevums ir izveidot tīmekļa lietotni - sporta apavu interneta veikalu “Quick step”. Šī lietotne ir paredzēta, lai nodrošinātu ērtu un efektīvu sporta apavu iegādi tiešsaistē. Piedāvāt lietotājam apskatīt un nopirkt kurpes ar dažadu dizainu kādas viņam patiks un ļaut lietotājiem tiešsaistē izveidot pirkuma pasūtījumu, izvēlēties maksājuma veidu un veikt pasūtījumu apmaksu.

Pašreizējā situācija, sporta apavu industrija ir ļoti konkurences spējīga, bet piedāvājot kvalitatīvus produktus, labu klientu apkalpošanu un iespējams arī pielāgotus risinājumus, lai piesaistītu klientus un sasniegtu to lojalitāti. Auditorija, kam domāti šie apavi, varētu būt sporta entuziasti, gan profesionāli skrējēji, gan cilvēki, kuri tikko sāk skriet.

Ir plānotas vairākas funkcijas(skat. att 1.):

* *Lietotājs var meklēt, filtrēt un atrast piemērotu apģerbu;*
* *Pievienot izvēlētās produktus mīļākajos;*
* *Pievienot un dzēst lietas groza;*
* *Sistēma ļauj klientiem novērtēt un atstāt atsauksmes;*
* *Iespēja klientiem sazināties ar klientu atbalsta komandu;*
* *Profila reģistretācija un autorizācija;*
* *Administrātora funkciolitātes.*

## Lietojumgadījuma modelis



1.att. Lietojumgadījuma diagramma

# PRASĪBU SPECIFIKĀCIJA

## Ieejas un izejas informācijas apraksts

### Ieejas informācijas apraksts

Sistēmā tiks nodrošināta šādas ieejas informācijas apstrāde.

1. Informācija par **Lietotājs** sastāvēs no šādiem datiem. (Ievadīta no klaviatūras)

* E-pasts – lietotāja pasts – burtu teksts ar izmēru līdz 50 rakstzīmēm (piem., “kristaps1@gmail.com”).
* Vārds – lietotāja vārds – burtu teksts ar izmēru līdz 20 rakstzīmēm (piem., “kristaps1”).
* Parole – lietotāja parole –burtu teksts ar izmēru līdz 50 rakstzīmēm (piem., “Acs!1123s”).

2. Informācija par **Produkts** sastāvēs no šādiem datiem. (Ievadīta no klaviatūras)

* Nosaukums – produkta nosaukuns – burtu teksts ar izmēru līdz 100 rakstzīmēm. (piem., “Zili skrišana apavi”).
* Sīkāka informācija – burtu teksts ar izmēru līdz 100 rakstzīmēm. (piem., šis modelis ir speciāli izstrādāts intensīvai lietošanai, skrienot pa ceļiem un takām).
* Cena – Produkta cena – vesels skaitlis kas ir vairāk par 0. (piem., “118 €”).
* Attēls – png, jpg formāta faili. . (piem., “Light\_Running\_Shoes\_Black.jpg”).

3. Informācija par **Pasūtīt** sastāvēs no šādiem datiem. (Ievadīta no klaviatūras)

* Nosaukums – Lietotāja vards – burtu teksts ar izmēru līdz 100 rakstzīmēm. (piem., “Zili skrišana apavi”).
* E-pasts – lietotāja pasts – burtu teksts ar izmēru līdz 100 rakstzīmēm. (piem., “kristaps1@gmail.com”).
* Numurs – tālruņa numurs – burtu teksts ar izmēru līdz 20 rakstzīmēm (piem., “25991667”).
* Adrese – Litotāja adrese – burtu teksts ar izmēru līdz 100 rakstzīmēm (piem., “Maskavas iela 23Latgales priekšpilsēta, Rīga, LV-1050”).
* Produktu skaits – kopējo produktu skaits – burtu teksts ar izmēru līdz 100 rakstzīmēm. (piem., “Zili skrišana apavi, zaļi skrišana apavi, sarkani skrišana apavi”).
* Kopējā cena – kopēja pasūtijuma cena – vesels skaitlis kas ir vairāk par 0. (piem., “118+118+118= 354€”).
* Datums – pasūtijuma datums – datums (piem., “21.11.2023”).

4. Informācija par **Grozs** sastāvēs no šādiem datiem. (Ievadīta no klaviatūras)

* Produkta identifikators – Produkta id– burtu teksts ar izmēru līdz 100 rakstzīmēm. (piem., “id 4”).
* Nosaukums – produkta nosaukuns – burtu teksts ar izmēru līdz 100 rakstzīmēm. (piem., “Melni skrišana apavi”).
* Cena – visu produktu kopēja cena – vesels skaitlis kas ir vairāk par 0. (piem., “136€”).
* Attēls – png, jpg formāta faili. . (piem., “Light\_Running\_Shoes\_Black.jpg”).
* Produktu skaits – kopējo produktu skaits – burtu teksts ar izmēru līdz 100 rakstzīmēm. (piem., “2 Melni skrišana apavi”).

### Izejas informācijas apraksts

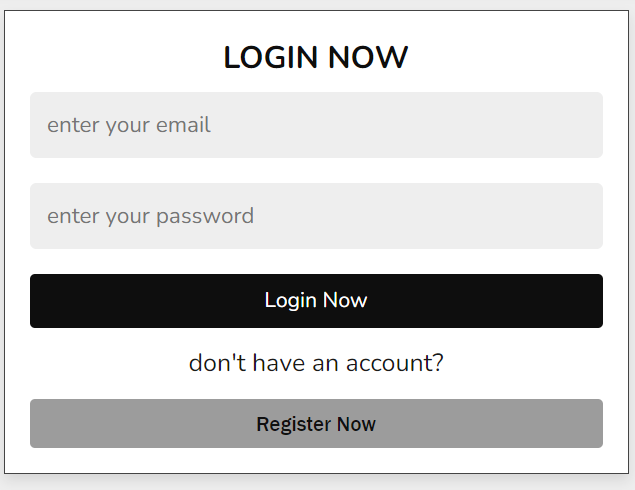
1. E-pasta paziņojums, pēc veiksmīgas pirkuma veikšanas klienti saņems paziņojumu un sīkāku informāciju par veikto pirkumu.
2. E-pasta paziņojums, kas aicina lietotājus atstāt atsauksmi par viņu nesen iegādātajiem skriešanas apaviem.
3. E-pasta paziņojums lietotājiem, kad viņu pasūtījums ir nosūtīts.
4. Paziņojumi par aktuālajām akcijām, jaunajiem produktiem un citiem piedāvājumiem.
5. E-pasta paziņojums, pamatojoties uz klienta pirkšanas vēsturi, periodiski tiks nosūtīti personalizēti piedāvājumi un akcijas.

## Funkcionālās prasības

1. Jānodrošina iespēja reģistrēt jaunu lietotāju.
   1. Jāparedz ieejas informācijas par lietotāju (skat. 3.att.) ievadīšana un pārbaude un formāta pareizību.
   2. Ja kāds no obligātiem laukiem nav ievadīts, tad izvadīt par to kļūdas paziņojumu.
   3. Salīdzināt ievadīto lietotāja e-pasts ar sistēmā jau eksistējošo lietotāju e-pastiem un izvadīt paziņojumu, ja tie sakrīt.
2. Jānodrošina lietotāja autorizācija.
   1. Ja lietotāja eksiste un parole ir pareiza aktīvs sistēmai ir jānodrošina autorizācija, pieslēdzoties ar e-pastu un paroli.
   2. Ja statuss ir neaktīvs, tad sistēmai ir jāieslēdz autorizācijas lapu.
   3. Ja kāds no laukiem nav ievadīts, izvadīt par to paziņojumu.
3. Jānodrošina iespēja pievienot produktus grozam.
   1. Lietotājam jāspēj meklēt produktus pēc dažādiem kritērijiem, piemēram, zīmols, izmērs, krāsa utt.
   2. Pievienojot produktu grozam, jāparāda produkta bilde, nosaukums, cena un daudzums.
   3. Jābūt iespējai mainīt produktu daudzumu vai izņemt to no groza.
   4. Jānodrošina paziņojums par veiksmīgu produktu pievienošanu vai jebkuru kļūdu, ja tāda notiek.
4. Jānodrošina vēlmju saraksta funkcionalitāte.
   1. Lietotājam jāspēj pievienot produktus vēlmju sarakstam.
   2. Jābūt iespējai izdzēst produktu no vēlmju saraksta.
   3. Paziņojums par veiksmīgu produktu pievienošanu vēlmju sarakstam vai jebkuru kļūdu, ja tāda notiek.
5. Jānodrošina pirkuma veikšana.
   1. Lietotājam jāspēj apskatīt grozu pirms pirkuma veikšanas.
   2. Pirkuma laikā jāsniedz informācija par piegādes adresi, maksājuma veidu un citiem nepieciešamajiem datiem.
   3. Jābūt iespējai pārbaudīt pasūtījuma informāciju pirms apstiprinājuma.
   4. Paziņojums par veiksmīgu pirkuma veikšanu vai jebkuru kļūdu, ja tāda notiek.
6. Jānodrošina meklēšanas funkcionalitāte.
   1. Lietotājam jāspēj veikt meklēšanu pēc produktu nosaukumiem, kategorijām vai citiem kritērijiem.
   2. Rezultātos jāparāda attiecīgie produkti ar to bilžu, nosaukumu un cenu.
   3. Paziņojums par veiksmīgu meklēšanu vai jebkuru kļūdu, ja tāda notiek.
7. Jānodrošina lietotājiem apskatīt savu profilu un mainīt datus.
   1. Paziņojums par veiksmīgu datu mainīšānu vai jebkuru kļūdu, ja tāda notiek.
8. Jānodrošina klientu atbalsts.
   1. Lietotājiem jābūt iespējai sazināties ar klientu atbalsta dienestu, ja rodas jautājumi vai problēmas.
   2. Jābūt izveidotam ērtam un efektīvam saziņas kanālam, kurā klienti var saņemt palīdzību un atbildes uz savām jautājumiem.
   3. Jānodrošina paziņojums par saņemto pieprasījumu un tā apstrādes statusu.

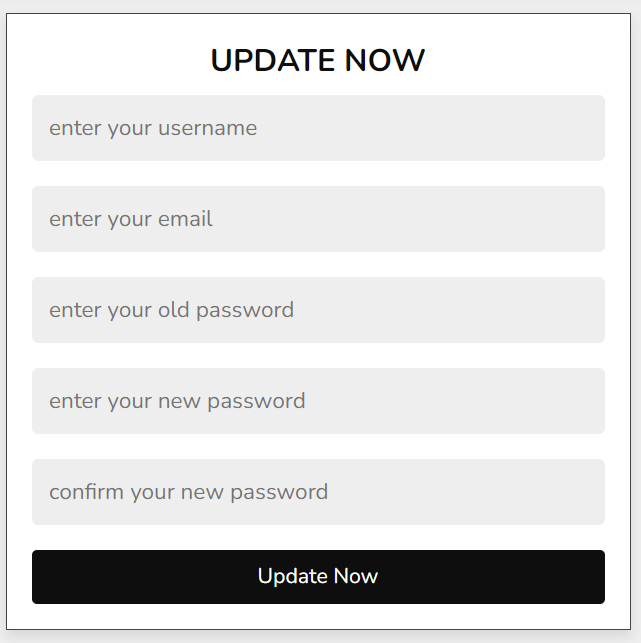
## Nefunkcionālās prasības

1. Sistēmas saskarnes valodai ir jābūt latviešu, angļu un krievu valodas
2. Jānodrošina tīmekļa lietojumprogrammas pielāgošanas ekrāna izmēriem, kas mūsdienās tiek lietoti, lai to varētu izmantot uz dažādiem monitora izmēriem.
3. Dizainam ir jābūt minimalistiska stilā un intuitīvi saprotamam.
4. Tekstam ir jābūt “Nunito” fontā balta, pelēkā un melnā krāsā.
5. Visiem produktiem ir jābūt detalizētiem aprakstiem un augstas kvalitātes attēliem.



2.att. Sistēmas Pieslēgšanās skice

Skice attēlo interfeisu, kur lietotāji var veikt pieslēgšanos vai zemāk reģistrāciju, ievadot e-pastu un paroli, tālāk nospiežt pogu “Login Now” veikt autorizāciju.



2.2.att. Lietotāja datu maiņas skice

Skice attēlo interfeisu, kur lietotāji var datu maiņu, vārda, e-pasta un paroles, tālāk nospiežt pogu “Update Now” veikt mainņu.

# UZDEVUMA RISINĀŠANAS LĪDZEKĻU IZVĒLES PAMATOJUMS

Šī sistēma ir paredzēta lietošanai interneta pārlūkprogrammā, un tās izstrādē tika izmantoti jaunākie programmatūras risinājumi, datu bāzes un citas tehnoloģijas.

XAMPP (XAMP) - Kā pamata servera vidē man ir izvēlējies XAMPP, kas ir krustplatforma, izmantojams servera risinājums. Tas ietver Apache serveri, MySQL datubāzu, PHP interpretatoru un citus komponentus, kas nepieciešami dinamisku un interaktīvu mājas lapu izstrādei. XAMPP sniedz ērtu veidu, kā lokāli izstrādāt un testēt mājas lapas funkcionalitāti pirms to publicēšanas tiešsaistē.

Visual Studio Code - Kā galveno izstrādes vidi es izmantoju Visual Studio Code, kas ir viegls, bet ļoti funkciju bagāts koda redaktors. Tas piedāvā plašu spraudņu klāstu, kas atvieglo kodēšanas procesu, kā arī integrētās atbalsta funkcijas dažādām programmēšanas valodām. Man ļauj viegli organizēt un rediģēt projektu failus, kā arī nodrošina efektīvu kodēšanas pieredzi.

PHP un JavaScript - Programmēšanai es izvēlējos PHP un JavaScript valodas. PHP tiek izmantots servera puses kodam, it īpaši mijiedarbojoties ar datubāzēm un apstrādājot pieprasījumus no klienta puses. JavaScript tiek izmantots klienta puses kodam, lai uzlabotu lietotāja pieredzi, izmantojot dinamiskus efektus un mijiedarbību.

phpMyAdmin - Lai pārvaldītu un uzturētu MySQL datubāzi, es izmantoju phpMyAdmin. Tas ir grafisks lietotāja saskarne datubāzu pārvaldībai, kas nodrošina iespēju izpildīt SQL vaicājumus, importēt un eksportēt datubāzes, kā arī ērti pārvaldīt tabulas un ierakstus.

Operētājsistēma Windows 10 - Kā darba platformu es izmantoju Windows 10 operētājsistēmu, kas nodrošina plašu saderību ar iepriekš minētajiem rīkiem un atvieglo to integrāciju un darbību kopā.

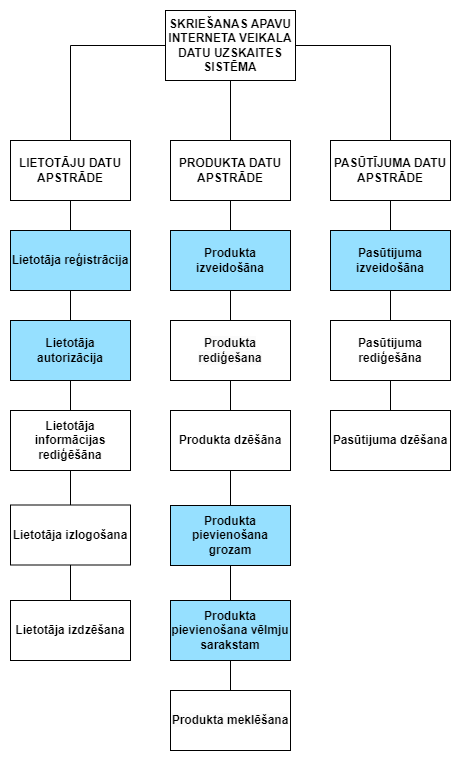
Šie līdzekļi kopā veido efektīvu darba vidi, kas ļauj man izstrādāt dinamisku mājas lapu un nodrošināt to darbību.

# PROGRAMMATŪRAS PRODUKTA MODELĒŠANA UN PROJEKTĒŠANA

* 1. **Sistēmas struktūras modelis**

# *Sistēmas arhitektūra*

Sistēma būs 3 apakšsistēmās (skat. 3. att.), lietotāju datu, produkta datu un pasūtījuma datu kas ir cieši savstarpēji saistītas. Katra no šīm apakšsistēmām atbild par konkrētu funkcionalitāti, nodrošinot integrētu pieeju sistēmas darbībai, ko uzskatāmi parāda funkcionālās dekompozīcijas diagramma.



3.att. Sistēmas arhitektūra

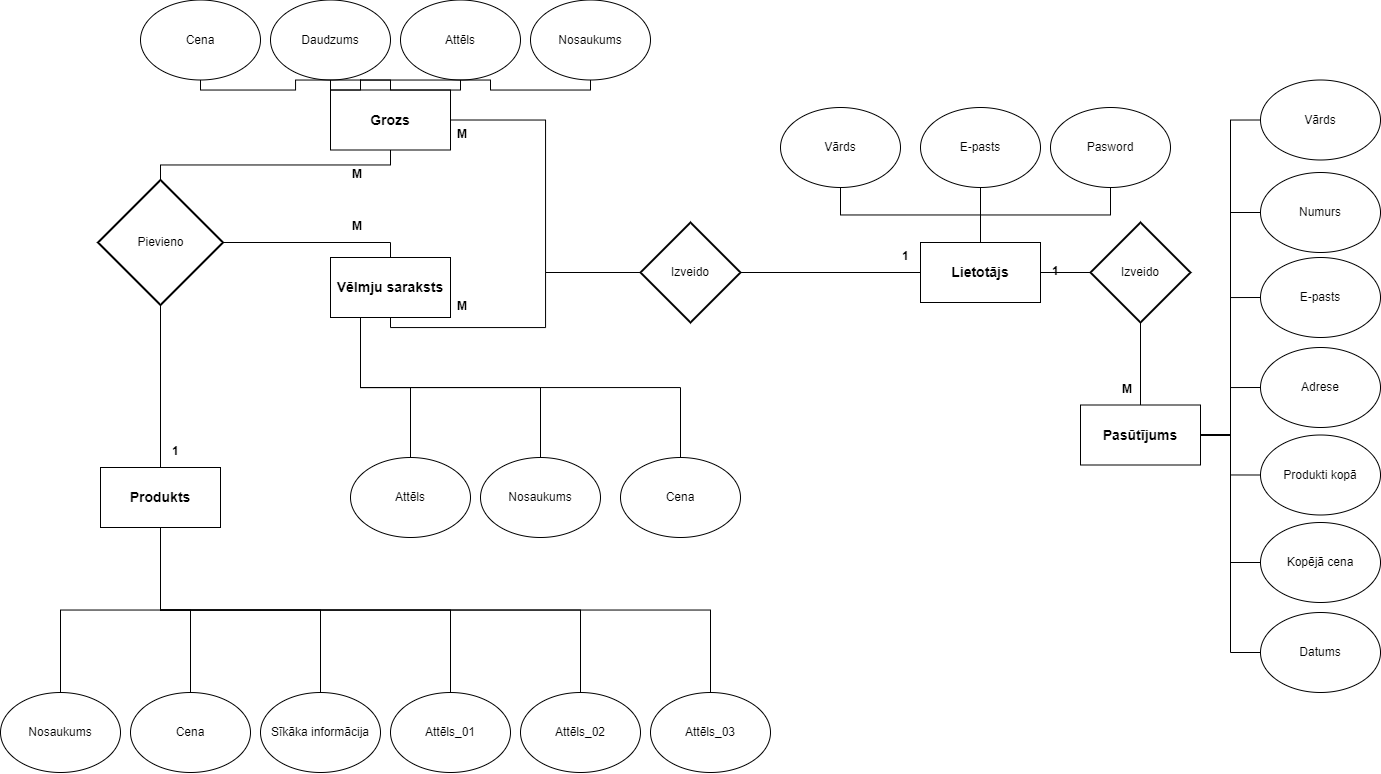
**Lietotāju datu apstrādes apakšsistēma -** ir atbildīga par visiem lietotāju datiem. Tas ietver lietotāja reģistrāciju, autorizāciju, iespēju rediģēt savus datus, izlogošanos. Kā arī nodrošinot efektīvu izmantošanu sistēmas funkcionalitātei un lietotāja pieredzes uzlabošanai.

**Produkta datu apstrādes apakšsistēma -** ir atbildīga par visu produktu informācijas apstrādi un uzturēšanu. Tas ietver jaunu produktu izveidošanu, produktu rediģēšānu, produktu dzēšanu, produktu meklēšanu, pievienošanu grozam un pievienošanu velmju sarakstam.

**Pasūtījumu datu apstrādes apakšsistēma -** ir atbildīga par visu saistīto ar pasūtījumu informācijas apstrādi un pārvaldīšanu. Tas ietver pasūtījumu izveidošanu, pasūtījumu rediģēšanu un pasūtījumu dzēšanu. Kā arī nodrošinot efektīvu pasūtijumu apstrādi un lietotāju pieredzi.

### **4.1.2. Entītiju relāciju datu modelis (ER-diagramm)**

Sistēmas ER-modelis sastāv no 5 entitijām (skat. 2. att.), kas nodrošina pamat informācijas uzglabāšanu un apstrādi. Tie ir "lietotājs", "produkts", "pasūtījums", "grozs", "vēlmju saraksts".



4.att. Sistēmas ER-diagramma

Datu bāzes projektēšanā datu kopu un saišu tām attēlošanai tika letota realitāšu-saišu diagramma. ER modelis sastav no 5 atubāzes tabulas, kas atspoguļo datu aprīti sistēmā:

* **“Lietotajs” -** Satur informāciju par lietotājiem. Tās atribūtu kopums sevī ietver: vārds, parole, e-pasts;
* **“Pasūtījumus” -** Satur informāciju par pasūtījumiem. Tās atribūtu kopums sevī ietver: Vārds, Number, E-pasts, Adrese, Produktu skaits, Kopēja cena, Datums;
* **“Produkts” -** Satur informāciju par produktiem. Tās atribūtu kopums sevī ietver: Nosaukums, Cena, Attēls, Sīkāka informācija;
* **“Groz” -** Satur informāciju par grozu. Tās atribūtu kopums sevī ietver: Nosaukums, Cena, Attēls;
* **“Vēlmju saraksts” -** Satur informāciju par vēlmju sarakstu. Tās atribūtu kopums sevī ietver: Nosaukums, Cena, Attēls;

ER diagrammā starp entītes ir attiecības viens pret daudziem un daudziem pret viens.

Objektū attiecības:

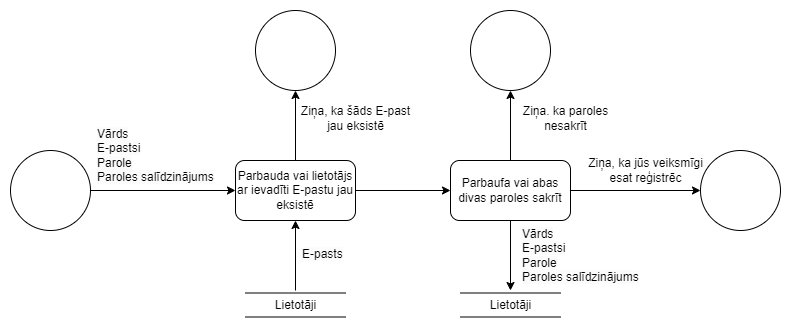
1. **Lietotājs - Grozs**:
   * Viens lietotājs varētu būt saistīts ar vairākiem groziem, bet katrs grozs pieder tikai vienam lietotājam.
   * Šī ir “**daudz vienam”** attiecība no tabulas “**users”** uz tabulu “**cart”**.
2. **Lietotājs - Vēlmju saraksts**:
   * Tāpat kā ar groziem, viens lietotājs varētu būt saistīts ar vairākiem vēlmju sarakstiem, bet katrs vēlmju saraksts pieder tikai vienam lietotājam.
   * Šī ir “**daudz vienam”** attiecība no tabulas “**users”** uz tabulu “**wishlist”**.
3. **Produkts - Grozs**:
   * Vairāki grozi var saturēt vienu un to pašu produktu, bet katrs produkts pieder tikai vienam grozam.
   * Šī ir “**daudz vienam**" attiecība no tabulas “**products”** uz tabulu “**cart”**.
4. **Produkts - Vēlmju saraksts**:
   * Līdzīgi kā ar groziem, vairāki vēlmju saraksti var saturēt vienu un to pašu produktu, bet katrs produkts pieder tikai vienam vēlmju sarakstam.
   * Šī ir “**daudz vienam”** attiecība no tabulas “**products”** uz tabulu “**wishlist”**.
5. **Pasūtījums - Lietotājs**:
   * Vairāki pasūtījumi var būt saistīti ar vienu lietotāju, bet katrs pasūtījums ir saistīts tikai ar vienu lietotāju.
   * Šī ir “**daudz vienam”** attiecība no tabulas “**orders”** uz tabulu **users”**.
6. **Pasūtījums - Produkts**:
   * Viens pasūtījums var ietvert vairākus produktus, bet katrs produkts var būt iekļauts tikai vienā pasūtījumā.
   * Šī ir “**daudz vienam”** attiecība no tabulas “**orders”** uz tabulu **products”**.

# 4.2. Funkcionālais sistēmas modelis

### **4.2.1. Datu plūsmu modelis**

**Lietotāja registrācija** (skat. 5.1. att.).

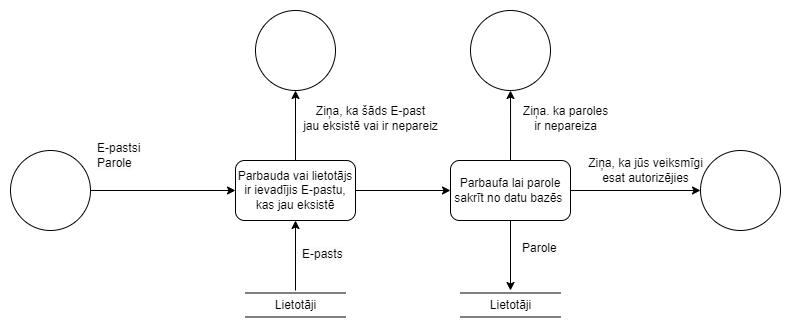
Pēc lietotāja datu ievades sistēma veic divas pārbaudes. Vispirms sistēma pārbauda, vai lietotājs ar tādu e-pastu jau eksistē, un tad pārbauda, vai lietotājam sakrīt paroles. Ja dati veiksmīgi pārbaudīti, tie tiek pievienoti datu bāzei, un sistēma informē lietotāju par veiksmīgu reģistrāciju. Ja kāda no pārbaudēm nav izpildīta, sistēma informē lietotāju par kļūdu un izvada ziņu.



5.1.att. Datu plūsmu diagramma – reģistrācija

**Lietotaja autorizācija**(skat. 5.2. att.).

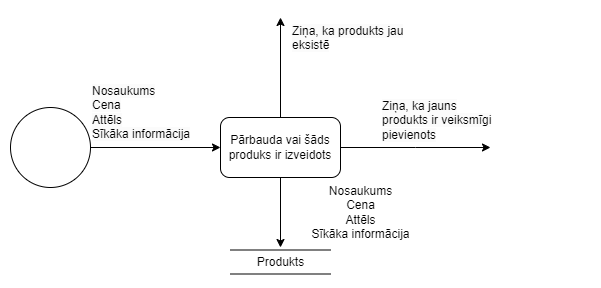
Pēc lietotāja datu ievades tie veic divas pārbaudes. Vispirms sistēma pārbauda, vai eksistē lietotājs ar tādu e-pastu, un vai lietotājs ir ievadījis pareizo paroli . Ja dati ir veiksmīgi pārbaudīti, sistēma informē lietotāju par veiksmīgu autorizāciju. Ja kāda no pārbaudēm nav izpildīta, sistēma informē lietotāju par kļūdu un izvada ziņu.



5.2.att. Datu plūsmu diagramma – autorizācija

**Produkta izveidošana**(skat. 5.3. att.).

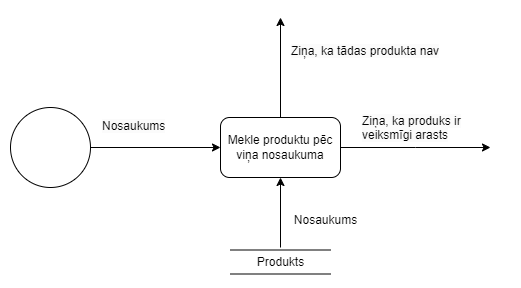
Pēc lietotāja datu ievades tie veic vienu pārbaudi. Vispirms sistēma pārbauda, vai šads produkst vispār eksistē un ir izveidots. Ja dati ir veiksmīgi pārbaudīti, sistēma informē administrātoru par veiksmīgu produkta izveidošanu. Ja kāda no pārbaudēm nav izpildīta, sistēma informē lietotāju par kļūdu un izvada ziņu.



5.3.att. Datu plūsmu diagramma – produkta izveidošana

**Produkta meklēšana** (skat. 5.4. att.).

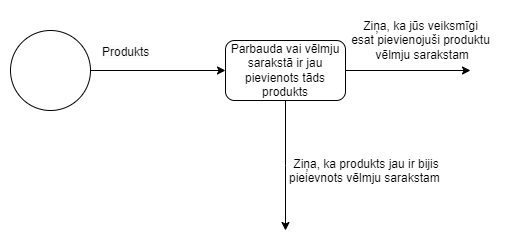
Pēc lietotāja datu ievades tie veic vienu pārbaudi. Vispirms sistēma pārbauda pēc nosaukuma, vai šads produkst vispār ir izveidots. Ja dati ir veiksmīgi pārbaudīti, sistēma informē lietoāju, kas produkts ir veiksmīgi atrasts. Ja kāda no pārbaudēm nav izpildīta, sistēma informē lietotāju par kļūdu un izvada ziņu.



5.4.att. Datu plūsmu diagramma – produkta meklēšana

**Produkta pievienošāna vēlmu sarakstam**(skat. 5.5. att.).

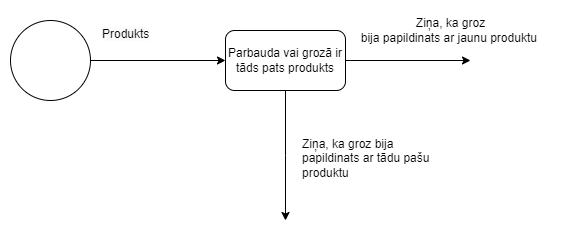
Pēc lietotāja datu ievades tie veic vienu pārbaudi. Vispirms sistēma pārbauda, vai šads produkst jau ir pievienots vēlmju sarakstam. Ja dati ir veiksmīgi pārbaudīti, sistēma informē lietoāju, kas produkts ir veiksmīgi pievienot pie vēlmju saraksta. Ja kāda no pārbaudēm nav izpildīta, sistēma informē lietotāju par kļūdu un izvada ziņu.



5.5.att. Datu plūsmu diagramma – produkta pievienošāna vēlmu sarakstam

**Preces pievienošāna grozam**(skat. 5.6. att.).

Pēc lietotāja datu ievades tie veic vienu pārbaudi. Vispirms sistēma pārbauda, vai grozā ir peivienots šads. Ja dati ir veiksmīgi pārbaudīti, sistēma informē lietoāju, ka groza ir papildināts ar tādu pašu preci vai jauns produkts ir veiksmīgi pievienots. Ja kāda no pārbaudēm nav izpildīta, sistēma informē lietotāju par kļūdu un izvada ziņu.



5.6.att. Datu plūsmu diagramma – produkta pievienošāna grozam

# DATU STRUKTŪRAS APRAKSTS

* 1. **Datu bāzes fiziskā struktūra**

Datu bāzes projektēšanas rezultātā tika veidota vairākas tabulas un starp tām tika definēta tabulu relācija jeb saistība kas norāda, ka abās tabulās saistītajos laukos ir vienādi dati. Saites piemeri: viens-pret-daudziem.

Datu bāze sastāv no 5 tabulām, kas satur informāciju par lietotāju, pasūtijumu, grozu, produktu un vēlmju sarakstu. Tabulu saišu shēma ir (skat. 6. att.).

Tabulas datu bāzē:

* Tabula "**Lietotajs**” šī tabula glabā informāciju par veikala lietotājiem, ieskaitot lietotāja identifikatoru (id), lietotājvārdu (name), e-pasta adresi (email) un šifrētu paroli (password).
* Tabula “**Grozs**” šī tabula glabā informāciju par lietotāju groziem, ieskaitot groza identifikatoru (id), lietotāja identifikatoru (user\_id), produktu identifikatoru (pid), produktu nosaukumu
* Tabula “**Pasūtijums**” šī tabula glabā informāciju par veikala pasūtījumiem, ieskaitot pasūtījuma identifikatoru (id), lietotāja identifikatoru (user\_id), pasūtītāja vārdu (name), kontakttālruni (number), e-pasta adresi (email), piegādes adresi (address), sarakstu ar pasūtītajiem produktiem (total\_products), kopējo cenu (total\_price), pasūtījuma datumu (placed\_on), maksājuma statusu (payment\_status).
* Tabula “**Vēlmju saraksts**” šī tabula glabā informāciju par lietotāju vēlmju sarakstiem, ieskaitot saraksta identifikatoru (id), lietotāja identifikatoru (user\_id), produktu identifikatoru (pid), produktu nosaukumu (name), cenu (price) un attēla faila nosaukumu (image).
* Tabula “**Produkts**” šī tabula satur informāciju par pieejamajiem veikala produktiem, ieskaitot produktu identifikatoru (id), nosaukumu (name), detaļām (details), cenu (price) un saistītos attēlus (image\_01, image\_02, image\_03).

Tabula “**Lietotajs**” saistīta ar tabulam “**Pasūtijums**”, “**Grozs**”, “**Vēlmju saraksts**” un “**Produkts**”

1.tabula

Tabulas **“Lietotajs”** sturktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr.** | **Nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| 1. | id | int | 100 | Unikālāis identifikātors |
| 2. | name | varchar | 20 | Lietotāja vārds |
| 3. | password | varchar | 50 | Lietotāja parole |
| 4. | email | varchar | 50 | Lietotāja pasts |

Tabula “**Pasūtijums**” satur informāciju par pasūtijuma tabulas sastāvdaļām.

2.tabula

Tabulas **“Pasūtijums”** struktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr.** | **Nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| 1. | id | int | 100 | Unikālāis identifikātors |
| 2. | name | varchar | 20 | Lietotāja vārds |
| 3. | number | varchar | 10 | Telefona numurs |
| 4. | email | varchar | 50 | Lietotāja pasts |
| 5. | method | varchar | 50 | Apmaksas veids |
| 6. | address | varchar | 500 | Lietotāja adrese |
| 7. | total\_products | varchar | 1000 | Preču saraksts un to daudzums |
| 8. | total\_price | int | 100 | Kopēja produktu cena |
| 9. | placed\_on | date |  | Pasūtijuma Datums |
| 10. | payment\_status | varchar | 20 | Maksājuma status |
| 11. | user\_id | int | 100 | Lietotāja unikālāis identifikātors |

Tabula “**Grozs**” satur informāciju par groza tabulas sastāvdaļām.

3.tabula

Tabulas **“Grozs”** struktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr.** | **Nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| 1. | id | int | 100 | Unikālāis identifikātors |
| 2. | name | varchar | 100 | Produckta nosaukums |
| 3. | image | varchar | 100 | Attēls |
| 4. | price | int | 100 | Produktu cena |
| 5. | quantity | int | 100 | Produktu daudzums |
| 6. | pid | int | 100 | Produkta unikālāis identifikātors |
| 7. | user\_id | int | 100 | Lietotāja unikālāis identifikātors |

Tabula “**Vēlmju saraksts**” satur informāciju par vēlmju saraksts tabulas sastāvdaļām.

4.tabula

Tabulas “**Vēlmju saraksts**” struktūra

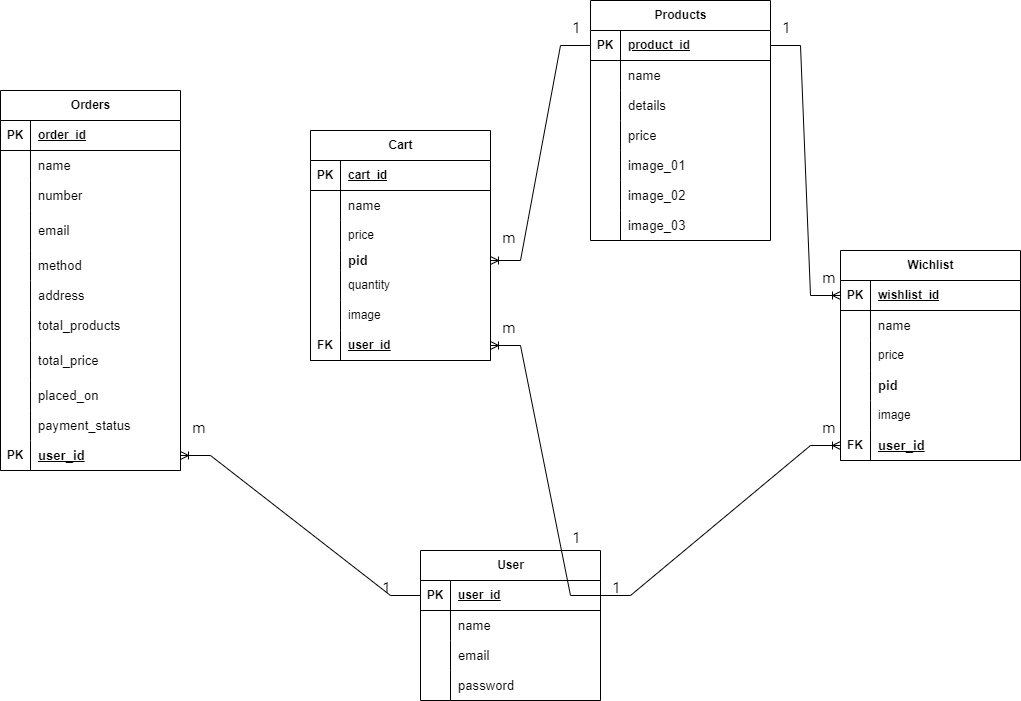
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr.** | **Nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| 1. | id | int | 100 | Unikālāis identifikātors |
| 2. | name | varchar | 100 | Produckta nosaukums |
| 3. | price | int | 100 | Produkta cena |
| 4. | image | varchar | 100 | Attēls |
| 5. | pid | int | 100 | Produkta unikālāis identifikātors |
| 6. | user\_id | int | 100 | Lietotāja unikālāis identifikātors |

Tabula “**Produkts**” satur informāciju par produktu tabulas sastāvdaļām.

5.tabula

Tabulas “**Produkts**” struktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr.** | **Nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| 1. | id | int | 100 | Unikālāis identifikātors |
| 2. | name | varchar | 100 | Produckta nosaukums |
| 3. | price | int | 100 | Produkta cena |
| 4. | image\_01 | varchar | 100 | Ceļš uz 1 produktu attēlu |
| 5. | image\_02 | varchar | 100 | Ceļš uz 2 produktu attēlu |
| 6. | image\_03 | varchar | 100 | Ceļš uz 3 produktu attēlu |
| 7. | details | varchar | 500 | Produkta sikāka informācija |

6.att. Tabulu saišu shēma