**RĪGAS VALSTS TEHNIKUMS**

DATORIKAS NODAĻA

Izglītības programma: Programmēšana

**KVALIFIKĀCIJAS DARBS**

**“Internetveikals sporta apavu Quick Step”**

Paskaidrojošais raksts \_\_\_ lpp.

Audzēknis: Kristaps Fedosejevs

Prakses vadītājs: Ilona Demčenko

Nodaļas vadītājs: Normunds Barbāns

**Rīga 2024**

**ANOTĀCIJA**

Kvalifikācijas darbā ir aprakstīts par sporta apavu internetveikalu “Quick Step”. Sistēma nodrošina iespēju vietnes lietotājiem veikt pirkumus tiešsaistē, pārvaldīt savus pasūtījumus, apskatīt un rediģēt savu profilu, kā arī izmantot vēlmju saraksta un groza funkcionalitāti. Sistēma piedāvā ērtu un intuitīvu lietotāja saskarni, kas ļauj viegli atrast un iegādāties sporta apavus, kā arī nodrošina administratoram nepieciešamos rīkus produktu pārvaldībai un lietotāju administrēšanai. Sistēma tika izstrādāta, izmantojot PHP programmēšanas valodu, MySQL datubāzu vadības sistēmu, un tā ietver vairākas mūsdienīgas tīmekļa tehnoloģijas, piemēram, HTML, CSS un JavaScript..

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 sporta spēles derībasv vietnes aktuālā problēma un tās risinājums. Uzdevumu nostādnē ir norādīti uzdevumi, kurus sistēmā būs nepieciešams veikt. Prasību specifikācija sastāv no ieejas un izejas informācijas, kā arī no sistēmas funkcionālajām un nefunkcionālajām prasībām. Uzdevuma risināšanas līdzekļu izvēles pamatojumā ir norādīti, kādi līdzekļi tiks izmantoti izstrādāšanai un kādiem nolūkiem tie tiek izmantoti. Programmatūras produkta modelēšanas un projektēšanas apraksts sastāv no sistēmas struktūra modeļa, kas ietver sistēmas arhitektūru un ER modeli, un funkcionālās sistēmas modeļa, kas satur datu plūsmas modeli. Datu struktūru aprakstā tiek parādīta datu bāzes relāciju shēma, kā arī tabulu struktūra ar aprakstu, kas ietver datu tipu un datu garumu norādīšanu. Lietotāju ceļvedī ir norādītas nepieciešamās sistēmas prasības aparatūrai un programmatūrai, sistēmas instalācija un palaišana, kā arī programmas apraksts, kas paskaidro, kā pareizi lietot sistēmu. Testa piemēri ir dots detalizēt likems likšanas apraksts ar vizuāliem attēliem.

Kvalifikācijas darbs sastāv no 38 lappusēm, kurā ietilpst 8 attēli, 6 tabulas un 3 pielikumi. Pielikumi satur saišu diagrammu, ER diagrammu un use-case diagramu.

**ANNotation**

The qualification work describes the sports shoe online store "Quick Step". The system enables website users to make online purchases, manage their orders, view and edit their profile, and use the wish list and cart functionality. The system offers a convenient and intuitive user interface that makes it easy to find and buy sports shoes, and also provides the administrator with the necessary tools for product management and user administration. The system was developed using the PHP programming language, MySQL database management system and includes several modern web technologies such as HTML, CSS and JavaScript.

The qualification work includes an introduction, statement of tasks, specification of requirements, rationale for choosing tools for solving the task, description of software product modeling and design, description of data structures, user's guide, conclusion and appendices. The introduction of the qualification paper contains the actual problem of the sports betting site and its solution. The task statement contains the tasks that will need to be performed in the system. The requirements specification consists of input and output information, as well as the functional and non-functional requirements of the system. The justification for choosing means of solving the task indicates which means will be used for development and for what purposes they are used. The modeling and design description of a software product consists of a system structure model, which includes the system architecture and an ER model, and a functional system model, which includes a data flow model. The description of data structures shows the relational schema of the database, as well as the table structure with a description that includes specifying data types and data lengths. The User's Guide provides the necessary system requirements for hardware and software, system installation and startup, and a program description that explains how to properly use the system. Test examples are given to detail the description of placing likems with visual images. The qualification paper consists of 38 pages, which includes 8 figures, 6 tables and 3 appendices. The appendices contain a link diagram, an ER diagram, and a use-case diagram.

**Saturs**

[Ievads 4](#_Toc167930623)

[1. UZDEVUMA NOSTĀDNE 5](#_Toc167930624)

[2. PRASĪBU SPECIFIKĀCIJA 7](#_Toc167930625)

[2.1. Ieejas un izejas informācijas apraksts 7](#_Toc167930626)

[*2.1.1. Ieejas informācijas apraksts 7*](#_Toc167930627)

[*2.1.2. Izejas informācijas apraksts 8*](#_Toc167930628)

[2.2. Funkcionālās prasības 8](#_Toc167930629)

[2.3. Nefunkcionālās prasības 10](#_Toc167930630)

[3. UZDEVUMA RISINĀŠANAS LĪDZEKĻU IZVĒLES PAMATOJUMS 12](#_Toc167930631)

[4. PROGRAMMATŪRAS PRODUKTA MODELĒŠANA UN 13](#_Toc167930632)

[4.1. Sistēmas struktūras modelis 13](#_Toc167930633)

[*4.1.1 Sistēma arhitektūra* 13](#_Toc167930634)

[*4.1.2. Entītiju relāciju datu modelis (ER-diagramm)* 14](#_Toc167930635)

[4.2. Funkcionālais sistēmas modelis 15](#_Toc167930636)

[*4.2.1. Datu plūsmu modelis* 15](#_Toc167930637)

[5. DATU STRUKTŪRAS APRAKSTS 19](#_Toc167930638)

[6. DATU STRUKTŪRAS 22](#_Toc167930639)

[6.1. Sistēmas prasības 22](#_Toc167930640)

[6.2. Sistēmas instalācija un palaišana 23](#_Toc167930641)

[6.3. Programmas apraksts 24](#_Toc167930642)

[6.4. Testa peimērs 24](#_Toc167930643)

[NOBEIGUMS 25](#_Toc167930644)

[INFORMĀCIJAS AVOTI 25](#_Toc167930645)

# 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.

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 iespēju katram pircējam justies īpašam un apmierinātam ar savu izvēli. Šajā jaunajā nākotnes sistēmā slēpjas potenciāls mainīt veidu, kā cilvēki izvēlas un iegādājas sporta apavus tiešsaistē.

Tirgus izpēte rāda, ka pastāv vairākas alternatīvas sistēmas, piemēram, "Nike", "Adidas", "Zappos" un "Foot Locker". Tomēr šīs platformas bieži vien piedāvā ierobežotu produktu klāstu vai koncentrējas uz konkrētiem zīmoliem, nesniedzot pietiekami daudz informācijas par katra produkta tehniskajiem raksturlielumiem un piemērotību dažādām aktivitātēm. "Quick Step" mērķis ir aizpildīt šo tukšumu tirgū, piedāvājot plašu izvēli, detalizētu informāciju un personalizētus ieteikumus, padarot sporta apavu iegādi par vienkāršu, ērtu un izbaudāmu pieredzi visiem klientiem.

Situācija, kas pastāvēja pirms "Quick Step" izveides, bija saistīta ar to, ka klienti bieži vien nespēja ātri un efektīvi atrast nepieciešamos sporta apavus. Fiziskie veikali piedāvā ierobežotu klāstu, un tradicionālās interneta platformas nespēja nodrošināt personalizētus ieteikumus un pietiekami daudz informācijas par produktiem. "Quick Step" ir veidots, lai risinātu šīs problēmas un nodrošinātu klientiem labāko iespējamo iepirkšanās pieredzi. Kopumā "Quick Step" ir solis uz priekšu, piedāvājot inovatīvus risinājumus, kas atbilst mūsdienu prasībām un padara sporta apavu iegādi par vienkāršu, ērtu un izbaudāmu pieredzi.

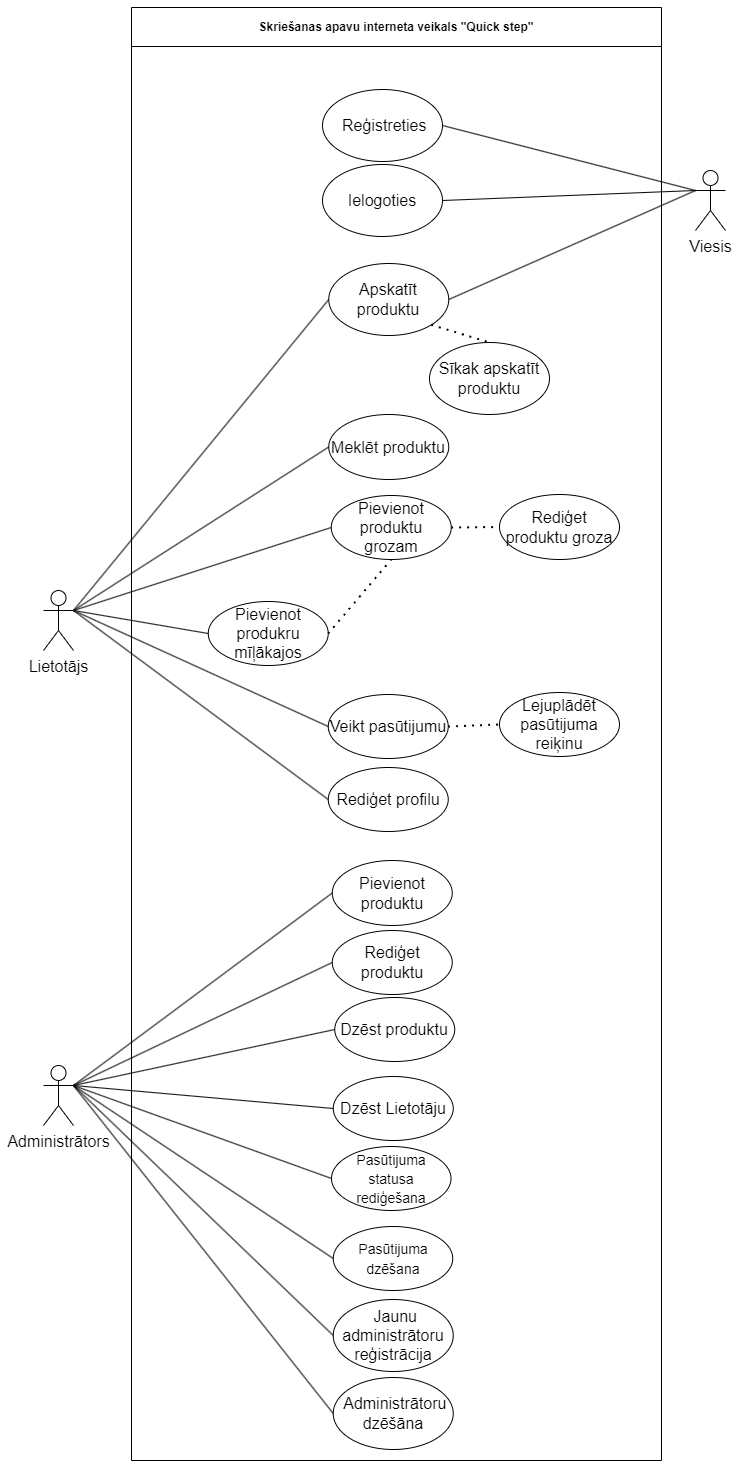
# UZDEVUMA NOSTĀDNE

Kvalifikācijas darba uzdevums ir izveidot 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. 1 att.).

* Lietotāja reģistrācija un pieteikšanās.
* Lietotāja profila pārvaldība, profila datu atjauninājums.
* Produktu meklēšana.
* Produkta detalizēts skatījums.
* Grozā produkta pevienošāna, dzēšāna un dauzdzuma rediģešana.
* Vēlmju sarakstā produkta pevienošāna un dzēšāna, tālāk pevienošāna grozam.
* Pasūtījumu pārvaldība.
* Administratīvā paneļa funkcijas.
* Drošības funkcijas.



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., “Kristaps”).
* Uzvārds – lietotāja uzvārds – burtu teksts ar izmēru līdz 50 rakstzīmēm (piem., “Fedosejevs”).
* Parole – lietotāja parole – burtu teksts ar izmēru līdz 50 rakstzīmēm (piem., “Acs!1123s”).
* Adrese – lietotāja adrese – burtu teksts ar izmēru līdz 255 rakstzīmēm (piem., “Gaismas iela, Stūnīši, Olaines pagasts”).
* Tālruņa numurs – lietotāja telefona numurs – burtu teksts ar izmēru līdz 20 rakstzīmēm (piem., “25991779”).

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

* Nosaukums – produkta nosaukuns – burtu teksts ar izmēru līdz 100 rakstzīmēm. (piem., “running shoes”).
* Sīkāka informācija – burtu teksts ar izmēru līdz 100 rakstzīmēm. (piem These running shoes are designed for optimal performance and comfort).
* Cena – Produkta cena – vesels skaitlis kas ir vairāk par 0. (piem., “100 €”).
* Attēls\_01 – produkta attēls – png, jpg formāta faili. (piem., “Light\_Running\_Shoes\_Black1.jpg”).
* Attēls\_02 – produkta attēls –png, jpg formāta faili. (piem., “Light\_Running\_Shoes\_Black2.jpg”).
* Attēls\_03 – produkta attēls – png, jpg formāta faili. (piem., “Light\_Running\_Shoes\_Black3.jpg”).
* Izmēri – produkta izmēri – burtu teksts ar izmēru līdz 50 rakstzīmēm (piem., “48”).
* Modelis: Produkta modelis – teksts ar izmēru līdz 50 rakstzīmēm (piem., "Model XYZ").
* Ārējais materiāls: Produkta ārējais materiāls – teksts ar izmēru līdz 50 rakstzīmēm (piem.,"Synthetic").
* Iekšējais materiāls: Produkta iekšējais materiāls – teksts ar izmēru līdz 50 rakstzīmēm (piem.,"Textile").
* Zole: Produkta zole – teksts ar izmēru līdz 50 rakstzīmēm (piem., "Synthetic").
* Krāsa: Produkta krāsa – teksts ar izmēru līdz 30 rakstzīmēm (piem., "Black").
* Dizainers: Produkta dizainers – teksts ar izmēru līdz 255 rakstzīmēm (piem., "NIKE").
* Artikula numurs: Produkta artikula numurs – teksts ar izmēru līdz 30 rakstzīmēm (piem.,"XYZ123").

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

* Metode– maksāšānas metode – burtu teksts ar izmēru līdz 50 rakstzīmēm. (piem., “PayPal”).
* Adrese – lietotāja adrese – burtu teksts ar izmēru līdz 255 rakstzīmēm (piem., “Gaismas iela, Stūnīši, Olaines pagasts”).
* Kopējā cena – kopējā cena pasūtijuma – burtu teksts ar izmēru līdz 100 rakstzīmēm (piem., “Grand total : €355”).
* Datums – pasūtijuma izveidošanas datums– burtu teksts ar izmēru līdz 100 rakstzīmēm (piem., “Maskavas iela 23Latgales priekšpilsēta, Rīga, LV-1050”).
* Maksājuma status – pasūtijuma status – formātas date. (piem., “Status: Pending”).

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

* Produktu daudzums – kopējo produktu skaits – burtu teksts ar izmēru līdz 100 rakstzīmēm. (piem., “2”).
* Izmērs – produkta izmērs – burtu teksts ar izmēru līdz 10 rakstzīmēm (piem., “44”).

5. Informācija par **Vēlmju saraksts** sastāvēs no šādiem datiem. (Ievadīta no klaviatūras)

* Izmērs – produkta izmēri – burtu teksts ar izmēru līdz 10 rakstzīmēm. (piem., “default”).
* 6. Informācija par **Pasūtījuma pozīcijas** sastāvēs no šādiem datiem. (Ievadīta no klaviatūras)
* Izmērs – produkta izmēri – burtu teksts ar izmēru līdz 10 rakstzīmēm. (piem., “42”).
* Produktu daudzums – kopējo produktu skaits – burtu teksts ar izmēru līdz 100 rakstzīmēm. (piem., “1”).
* Cena – Produkta cena – vesels skaitlis kas ir vairāk par 0. (piem., “130 €”).

### **Izejas informācijas apraksts**

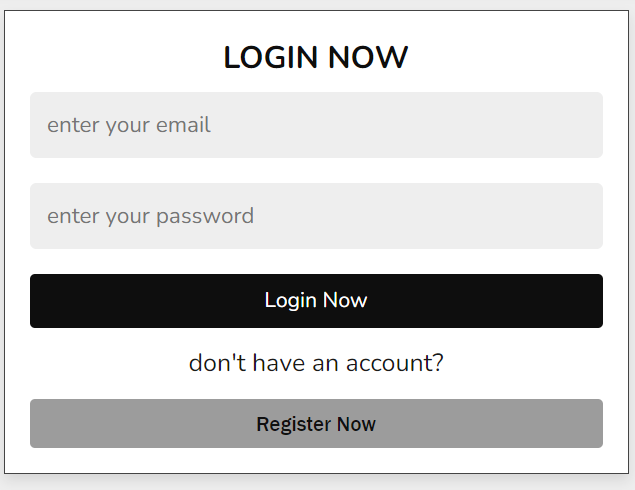
1. **PDF fails ar pasūtijuma reiķina informāciju.** Pasūtijuma kreisajā augšā.
2. **PDF fails ar administrātora ststistiku.** Pavidu ekrānam.
3. **Ekrāna paziņojumi par pasūtījuma veiksmīgu apstiprināšanu.** Ekrāna augšā.
4. **Ekrāna paziņojumi par produkta pievienošanu vēlmju sarakstam.** Ekrāna augšā.
5. **Ekrāna paziņojumi par produkta pievienošanu grozam.** Ekrānaaugšā**.**
6. **Ekrāna paziņojumi par veiksmīgu produkta izveidošanu.** Ekrānaaugšā**.**
7. **Ekrāna paziņojumi par veiksmīgu produkta atjaunišanu.** Ekrānaaugšā**.**

## Funkcionālās prasības

1. Jānodrošina iespēja reģistrēt jaunu lietotāju.
   1. Jāparedz ieejas informācijas par lietotāju ievadīšana un pārbaude un formāta pareizību.
   2. Ja ievadītā e-pasta adrese jau ir reģistrēta, sistēmai jāinformē lietotājs par esošo reģistrāciju un jāiesaka atjaunot paroli..
   3. Visi obligātie reģistrācijas formas lauki jāaizpilda. Ja lauks nav aizpildīts, sistēmai jāizdod attiecīgs kļūdas ziņojums.
   4. Sistēmai jāpārbauda ievadītā e-pasta un paroles formāta pareizība. Ja formāts ir nepareizs, sistēmai jāizdod attiecīgs kļūdas ziņojums
2. Jānodrošina lietotāja autorizācija. (skat. 2.1.att.)
   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 lietotāja izrakstīšanās.
   1. Profila panelī jābūt pieejamai "Logout" pogai autorizētiem lietotājiem..
4. Jānodrošina iespēja pievienot produktus grozam.
   1. Lietotājam jābūt iespējai pievienot produktus grozam.
   2. Grozā jāparāda produkta nosaukums, cena, attēli un citi būtiski parametri.
   3. Lietotājam jābūt iespējai rediģēt produktu daudzumu grozā un izdzēst produktus no vēlmju saraksta.
5. Jānodrošina vēlmju saraksta funkcionalitāte.
   1. Lietotājam jāspēj pievienot produktus vēlmju sarakstam.
   2. Vēlmju sarakstā jāparāda produkta nosaukums, cena, attēli un citi būtiski parametri.
6. Jānodrošina pirkuma veikšana.
   1. Sistēmai jānodrošina iespēja veikt pasūtījumus, izvēloties produktus, norādot piegādes adresi un maksājuma veidu.
   2. Pirkuma laikā jāsniedz informācija par piegādes adresi, maksājuma veidu un citiem nepieciešamajiem datiem.
   3. Pasūtījuma detaļās jāparāda visi pasūtītie produkti, kopējā cena, piegādes adrese un maksājuma statuss.
7. 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.
8. Jānodrošina lietotājiem profila pārvaldība. (skat. 2.2.att.)
   1. Lietotājam jābūt iespējai skatīt un rediģēt savu profilu, tostarp vārdu, uzvārdu, e-pasta adresi un adresi.
   2. Lietotājam jābūt iespējai mainīt paroli, ievadot pašreizējo un jauno paroli.
9. Jānodrošina produktu meklēšana.
   1. Sistēmai jānodrošina meklēšanas funkcija, lai lietotājs varētu meklēt produktus pēc dažādiem kritērijiem, piemēram, nosaukuma, kategorijas, zīmola, izmēra un krāsas.
   2. Meklēšanas rezultātos jāparāda atbilstošie produkti ar to nosaukumu, cenu un attēliem.
10. Jānodrošina administratora funkcionalitāte.
    1. Administratoram jābūt iespējai piekļūt visiem lietotāju kontiem un rediģēt tos.
    2. Administratoram jābūt iespējai skatīt un pārvaldīt visus pasūtījumus.
    3. Administratoram jābūt iespējai pievienot, rediģēt un dzēst produktus.
    4. Administratoram jābūt iespējai ģenerēt statistikas pārskatus par pārdošanu, lietotāju aktivitāti un citiem parametriem.
11. Jānodrošina paziņojumi funkcionalitāte.
    1. Paziņojumi par veiksmīgu reģistrāciju.
    2. Paziņojumi par veiksmīgu autorizāciju.
    3. Paziņojumi par paroles atjaunošanu.
    4. Paziņojumi par veiksmīgu pasūtījuma veikšanu.
    5. Paziņojumi par pasūtījuma statusa izmaiņām.
    6. Paziņojumi par produktu pievienošanu grozam.

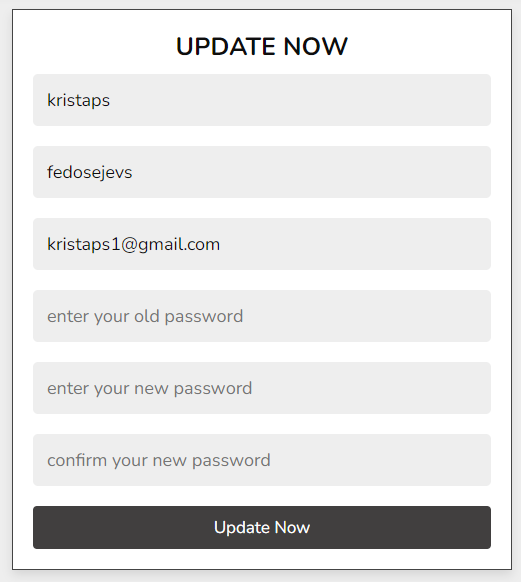
## Nefunkcionālās prasības

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.



2.1.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, nodrošinot lietotājiem ērtu un efektīvu sporta apavu iegādes pieredzi. Sistēma sastāv no divām daļām: lietotāja puses (frontend) un servera puses (backend). Lietotāja puse ir atbildīga par vizuālo saskarni un mijiedarbību ar lietotāju, savukārt servera puse apstrādā datus, veic darījumus un uztur datu bāzi.

**XAMPP 8.0.28** - XAMPP ir krustplatformas servera risinājums, kas ietver Apache serveri, MySQL datu bāzi, 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.

**Visual Studio Code 1.70** - 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. Visual Studio Code nodrošina efektīvu kodēšanas pieredzi un atbalsta dažādu valodu sintaksi un kļūdu pārbaudi.

**PHP 8.0.28** - PHP tiek izmantots servera puses kodam, it īpaši mijiedarbojoties ar datu bāzēm un apstrādājot pieprasījumus no klienta puses. PHP ir populāra skriptu valoda, kas ir piemērota dinamisku tīmekļa lietojumprogrammu izstrādei. PHP kods tiek izpildīts uz servera, ģenerējot HTML, kas tiek nosūtīts uz klienta pusi.

**JavaScript (ES6)** - JavaScript tiek izmantots klienta puses kodam, lai uzlabotu lietotāja pieredzi, izmantojot dinamiskus efektus un mijiedarbību. JavaScript ir svarīgs rīks, lai veidotu interaktīvas un atsaucīgas tīmekļa lapas. ES6 (ECMAScript 6) versija piedāvā daudzus uzlabojumus un jaunas iespējas, kas padara kodēšanu efektīvāku un lasāmāku.

**phpMyAdmin 5.2.1** - Lai pārvaldītu un uzturētu MySQL datu bāzi, es izmantoju phpMyAdmin. Tas ir grafisks lietotāja saskarne datu bāzu pārvaldībai, kas nodrošina iespēju izpildīt SQL vaicājumus, importēt un eksportēt datu bāzes, kā arī ērti pārvaldīt tabulas un ierakstus. phpMyAdmin ir ļoti noderīgs rīks datu bāzes administrēšanai.

**HTML5** - HTML5 ir jaunākā HTML standarta versija, kas tiek izmantota tīmekļa satura struktūras veidošanai. HTML5 piedāvā daudzas jaunas iespējas un elementus, kas padara tīmekļa lapas vieglākas, ātrākas un funkcionālākas.

**CSS3** - CSS3 tiek izmantots, lai stilizētu un izkārtotu tīmekļa lapas saturu. CSS3 piedāvā daudzus jauninājumus, kas ļauj veidot sarežģītas animācijas, pārejas un citus vizuālus efektus, kas uzlabo lietotāja pieredzi.

**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.

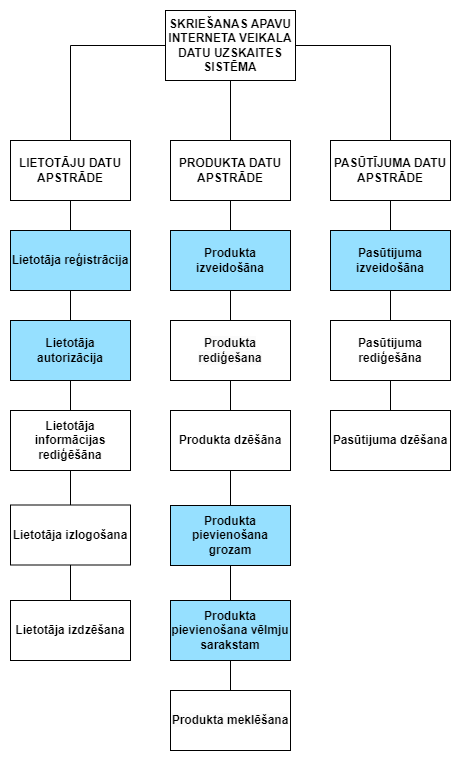
**TCPDF (PDF ģenerēšana)** - TCPDF ir PHP klase, kas ļauj ģenerēt PDF dokumentus tieši no PHP koda bez nepieciešamības pēc papildu bibliotēkām. Tas tiek izmantots, lai ģenerētu PDF failus ar pasūtījumu rēķiniem un administratora statistiku. Atsauce

# PROGRAMMATŪRAS PRODUKTA MODELĒŠANA UN

# 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

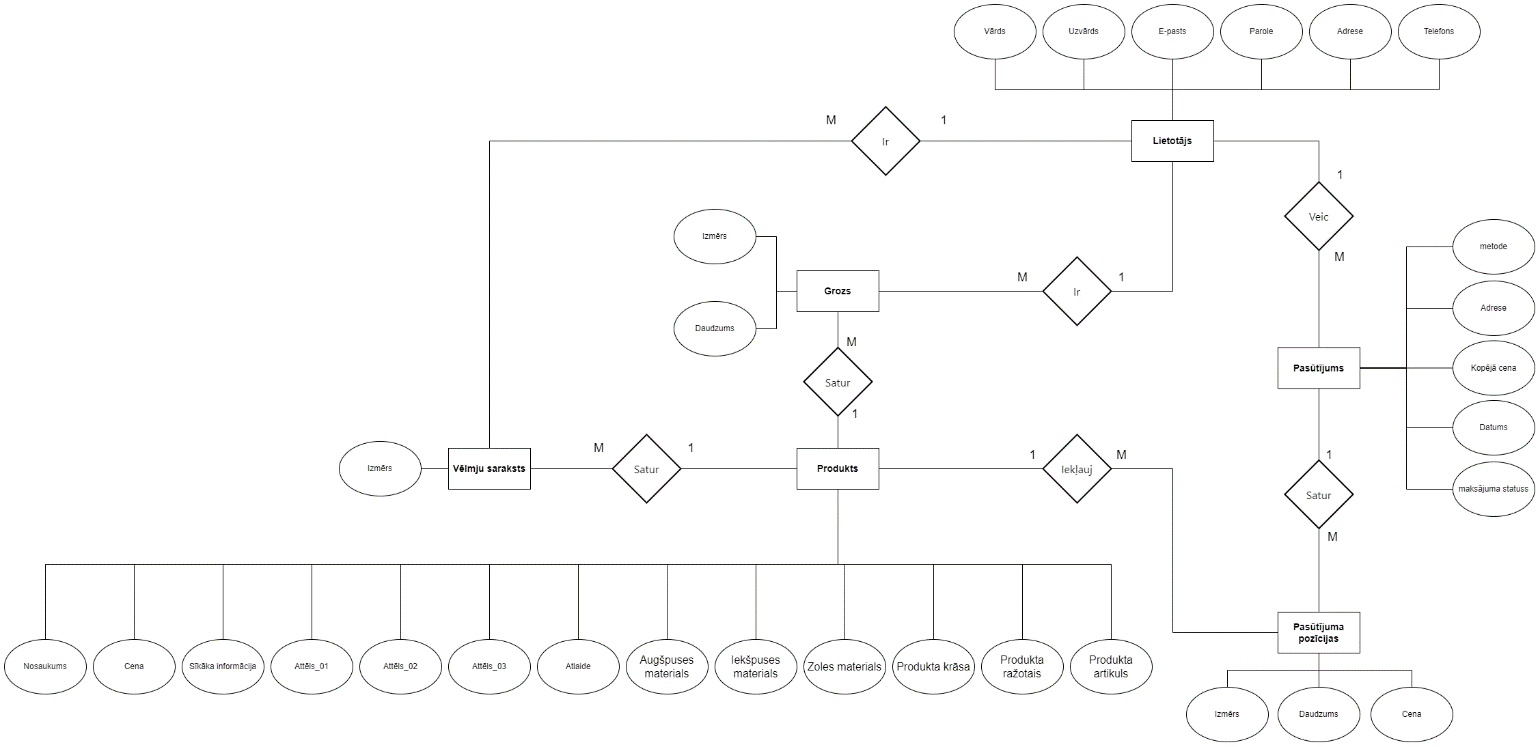
**Lietotāju datu apstrādes apakšsistēma -** ir atbildīga par visiem lietotāju datiem. Tas ietver lietotāja reģistrāciju, autorizāciju, iespēju rediģēt savus datus, izlogošanos. Kā arī nodrošinot efektīvu izmantošanu sistēmas funkcionalitātei un lietotāja pieredzes uzlabošanai.

**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. Sistēmas ER modelis**

Sistēmas ER-modelis sastāv no 6 entitijām (skat. 4.att.), kas nodrošina pamatinformācijas uzglabāšanu un apstrādi. Tie ir "ORDER\_ITEMS", "USER", "PRODUCT", "ORDER", "CART", "WISHLIST"".



4.att. Sistēmas ER-diagramma

**"USER"**

* + Starp "**USER**" un "**ORDER**" ir attiecības viens pret daudziem, jo viens lietotājs var veikt vairākus pasūtījumus, bet katrs pasūtījums pieder tikai vienam lietotājam.
  + Starp "**USER**" un "**CART**" ir attiecības viens pret daudziem, jo viens lietotājs var pievienot vairākus grozus, bet katrs grozs pieder tikai vienam lietotājam.
  + Starp "**USER**" un "**WISHLIST**" ir attiecības viens pret daudziem, jo viens lietotājs var pievienot vairākus vēlmju sarakstus, bet katrs vēlmju saraksts pieder tikai vienam lietotājam.

**"PRODUCT"**

* Starp "**PRODUCT**" un "**ORDER\_ITEMS**" ir attiecības viens pret daudziem, jo vienu produktu var iekļaut vairākos pasūtījuma pozīciju ierakstos, bet katrs pasūtījuma pozīcijas ieraksts attiecas tikai uz vienu produktu.
* Starp "**PRODUCT**" un "**CART**" ir attiecības daudzi pret daudziem, jo vienu produktu var pievienot vairākiem groziem, un katrā grozā var būt vairāki produkti.
* Starp "**PRODUCT**" un "**WISHLIST**" ir attiecības daudzi pret daudziem, jo vienu produktu var pievienot vairākiem vēlmju sarakstiem, un katrā vēlmju sarakstā var būt vairāki produkti.

**"ORDER"**

* Starp "**ORDER**" un "**ORDER\_ITEMS**" ir attiecības viens pret daudziem, jo vienam pasūtījumam var būt vairāki pasūtījuma pozīciju ieraksti, bet katrs pasūtījuma pozīcijas ieraksts attiecas tikai uz vienu pasūtījumu.
* Starp "**ORDER**" un "**USER**" ir attiecības daudzi pret vienu, jo vienam lietotājam var būt vairāki pasūtījumi, bet katrs pasūtījums ir saistīts tikai ar vienu lietotāju.

**"CART"**

* Starp "**CART**" un "**PRODUCT**" ir attiecības daudzi pret daudziem, jo vienu produktu var pievienot vairākiem groziem, un katrā grozā var būt vairāki produkti.
* Starp "**CART**" un "**USER**" ir attiecības daudzi pret vienu, jo vienam lietotājam var būt vairāki grozi, bet katrs grozs pieder tikai vienam lietotājam.

**"WISHLIST"**

* Starp "**WISHLIST**" un "**PRODUCT**" ir attiecības daudzi pret daudziem, jo vienu produktu var pievienot vairākiem vēlmju sarakstiem, un katrā vēlmju sarakstā var būt vairāki produkti.
* Starp "**WISHLIST**" un "**USER**" ir attiecības daudzi pret vienu, jo vienam lietotājam var būt vairāki vēlmju saraksti, bet katrs vēlmju saraksts pieder tikai vienam lietotājam.

**"ORDER\_ITEMS"**

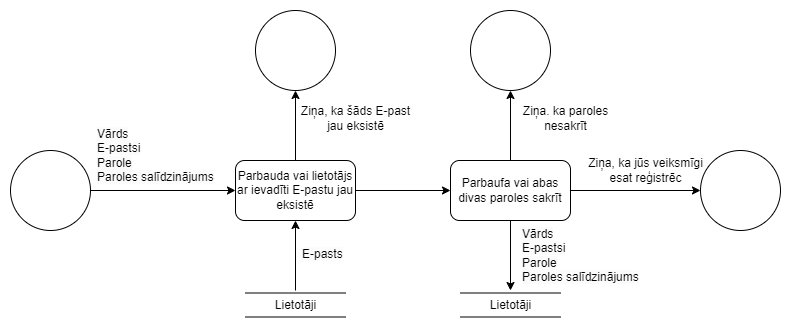
* Starp "**ORDER\_ITEMS**" un "**PRODUCT**" ir attiecības daudzi pret vienu, jo vienu produktu var iekļaut vairākos pasūtījuma pozīciju ierakstos, bet katrs pasūtījuma pozīcijas ieraksts attiecas tikai uz vienu produktu.
* Starp "**ORDER\_ITEMS**" un "**ORDER**" ir attiecības daudzi pret vienu, jo vienam pasūtījumam var būt vairāki pasūtījuma pozīciju ieraksti, bet katrs pasūtījuma pozīcijas ieraksts attiecas tikai uz vienu pasūtījumu**.**

# 4.2. Funkcionālais sistēmas modelis

### **4.2.1. Datu plūsmu modelis**

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

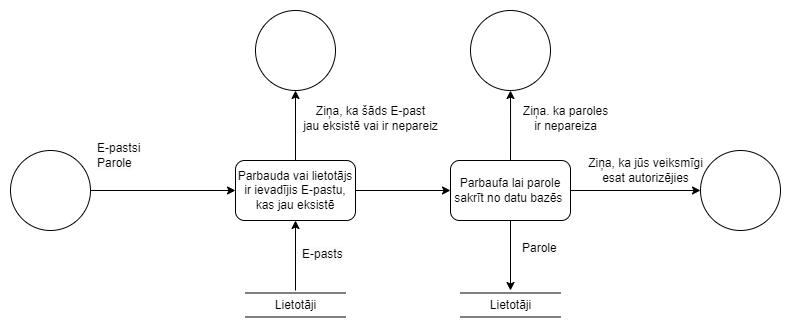
Pēc lietotāja datu ievades sistēma veic divas pārbaudes. Vispirms sistēma pārbauda, vai lietotājs ar tādu e-pastu jau eksistē, un tad pārbauda, vai lietotājam sakrīt paroles. Ja dati veiksmīgi pārbaudīti, tie tiek pievienoti datu bāzei, un sistēma informē lietotāju par veiksmīgu reģistrāciju. Ja kāda 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.).

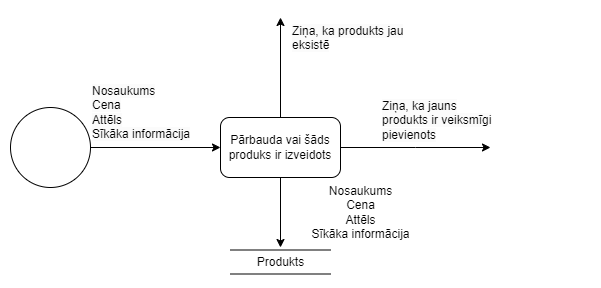
Pēc lietotāja datu ievades tie veic divas pārbaudes. Vispirms sistēma pārbauda, vai eksistē lietotājs ar tādu e-pastu, un vai lietotājs ir ievadījis pareizo paroli . Ja dati ir veiksmīgi pārbaudīti, sistēma informē lietotāju par veiksmīgu autorizāciju. Ja kāda no pārbaudēm nav izpildīta, sistēma informē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



5.2.att. Datu plūsmu diagramma – autorizācija

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

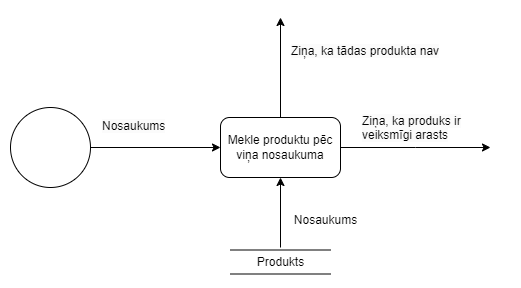
Pēc lietotāja datu ievades tie veic vienu pārbaudi. Vispirms sistēma pārbauda, vai šads produkst vispār eksistē un ir izveidots. Ja dati ir veiksmīgi pārbaudīti, sistēma informē administrātoru par veiksmīgu produkta izveidošanu. Ja kāda no pārbaudēm nav izpildīta, sistēma informē lietotāju par kļūdu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



5.3.att. Datu plūsmu diagramma – produkta izveidošana

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

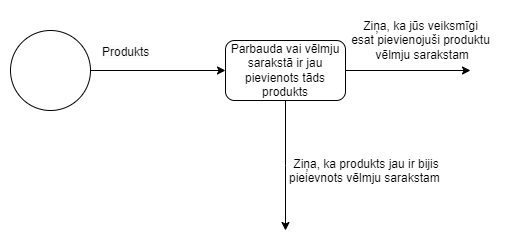
Pēc lietotāja datu ievades tie veic vienu pārbaudi. Vispirms sistēma pārbauda pēc nosaukuma, vai šads produkst vispār ir izveidots. Ja dati ir veiksmīgi pārbaudīti, sistēma informē lietoāju, kas produkts ir veiksmīgi atrasts. Ja kāda no pārbaudēm nav izpildīta, sistēma informē lietotāju par kļūdu un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



5.4.att. Datu plūsmu diagramma – produkta meklēšana

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

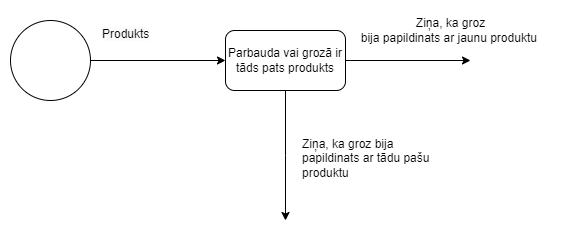
Pēc lietotāja datu ievades tie veic vienu pārbaudi. Vispirms sistēma pārbauda, vai šads produkst jau ir pievienots vēlmju sarakstam. Ja dati ir veiksmīgi pārbaudīti, sistēma informē lietoāju, kas produkts ir veiksmīgi pievienot pie vēlmju saraksta. Ja kāda no pārbaudēm nav izpildīta, sistēma informē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

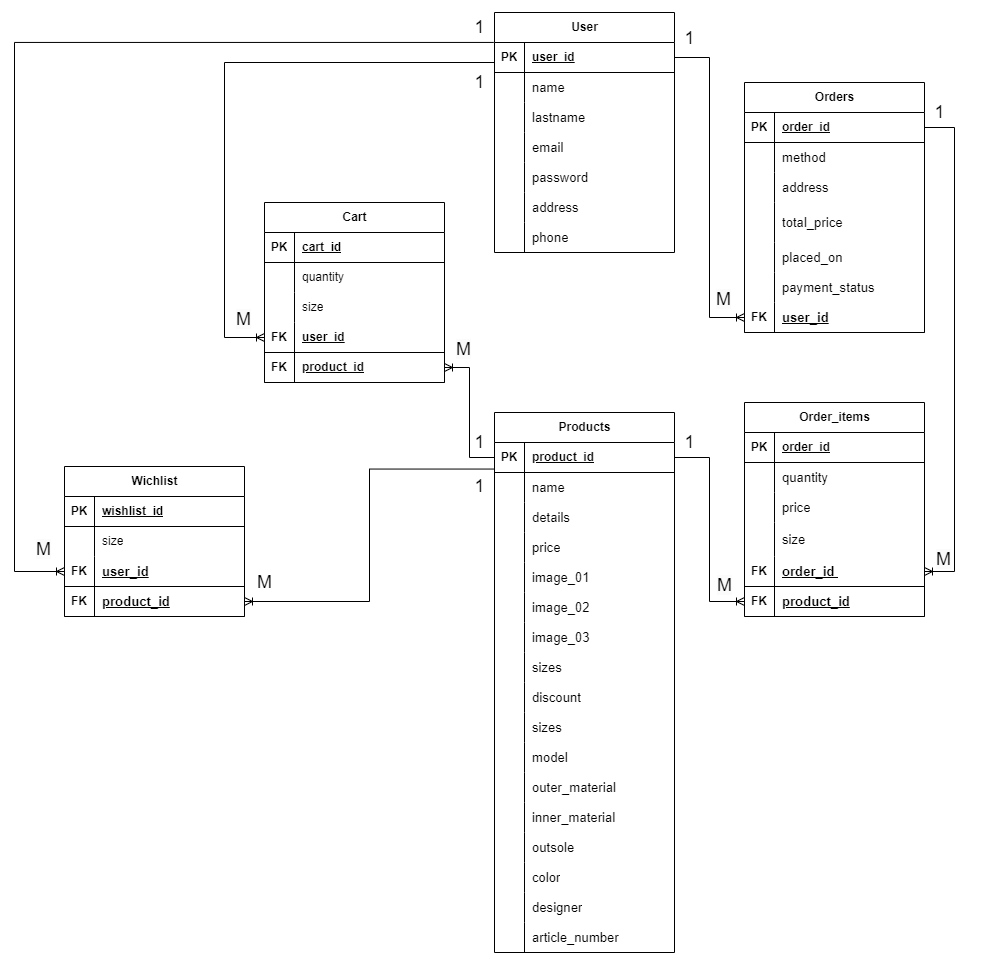


5.6.att. Datu plūsmu diagramma – produkta pievienošāna grozam

# DATU STRUKTŪRAS APRAKSTS

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 6 tabulām, kas satur informāciju par lietotāju, pasūtijumu, grozu, produktu un vēlmju sarakstu. Tabulu saišu shēma ir (skat. 6. att.).



6.att. Tabulu saišu shēma

Tabulas datu bāzē:

* Tabula "**Users**” šī tabula glabā informāciju par veikala lietotājiem.
* Tabula “**Cart**” šī tabula glabā informāciju par produktiem grozā.
* Tabula “**Orders**” šī tabula glabā informāciju par veikala pasūtījumiem.
* Tabula “**Wishlist** ” šī tabula glabā informāciju par produktiem vēlmju sarakstiu.
* Tabula “**Order\_items** ” šī tabula glabā informāciju par katru atsevišķu pasūtījuma vienību.
* Tabula “**Produkts**” šī tabula satur informāciju par produktiem.

1.tabula

Tabulas **“Users”** sturktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr.** | **Nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| 1. | id | int | 100 | Unikālāis identifikātors |
| 2. | name | varchar | 20 | Lietotāja vārds |
| 3. | lastname | varchar | 50 | Lietotāja uzvārds |
| 4. | email | varchar | 50 | Lietotāja pasts |
| 5. | password | varchar | 50 | Lietotāja parole |
| 6. | address | varchar | 255 | Lietotāja adrese |
| 7. | phone | varchar | 20 | Lietotāja telefona numurs |

Tabula ”**Users**” saistīta ar tabulam “**Order**”, “**Cart**”, “**Wishlist**” caur lauku user\_id.

2.tabula

Tabulas **“Order”** struktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr.** | **Nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| 1. | id | int | 100 | Unikālāis identifikātors |
| 2. | method | varchar | 50 | Apmaksas veids |
| 3. | address | varchar | 500 | Lietotāja apmaksas adrese |
| 4. | total\_price | int | 100 | Kopēja produktu cena |
| 5. | placed\_on | date |  | Pasūtijuma Datums |
| 6. | payment\_status | varchar | 20 | Maksājuma status |
| 7. | user\_id | int | 100 | Lietotāja unikālāis identifikātors |

Tabula ”**Order**” saistīta ar tabulam “**Users**” caur lauku user\_id, “**Order\_Items** caur id kā ārējā atslēga uz tabulu.

3.tabula

Tabulas **“Cart”** struktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr.** | **Nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| 1. | id | int | 100 | Unikālāis identifikātors |
| 2. | quantity | int | 10 | Produktu daudzums |
| 3. | size | varchar | 10 | Produkta izmērs |
| 4. | product\_id | int | 100 | Produkta unikālāis identifikātors |
| 5. | user\_id | int | 100 | Lietotāja unikālāis identifikātors |

Tabula ”**Cart**” saistīta ar tabulam “**Users**”, “**Products**” caur lauku user\_id un caur lauku product\_id.

4.tabula

Tabulas “**Wishlist**” struktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr.** | **Nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| 1. | id | int | 100 | Unikālāis identifikātors |
| 2. | size | varchar | 10 | Produkta izmērs |
| 3. | product\_id | int | 100 | Produkta unikālāis identifikātors |
| 4. | user\_id | int | 100 | Lietotāja unikālāis identifikātors |

Tabula ”**Wishlist**” saistīta ar tabulam “**Users**”, “**Products**” caur lauku user\_id un caur lauku product\_id.

5.tabula

Tabulas “**Products**” struktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr.** | **Nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| 1. | id | int | 100 | Unikālāis identifikātors |
| 2. | name | varchar | 100 | Produckta nosaukums |
| 3, | details | varchar | 500 | Produkta sikāka informācija |
| 4. | price | int | 100 | Produkta cena |
| 5. | image\_01 | varchar | 100 | Ceļš uz 1 produktu attēlu |
| 6. | image\_02 | varchar | 100 | Ceļš uz 2 produktu attēlu |
| 7. | image\_03 | varchar | 100 | Ceļš uz 3 produktu attēlu |
| 8. | sizes | varchar | 50 | Produktu izmēri |
| 9. | discount | decimal | 5,2 | Produkta atlaide |
| 10. | model | varchar | 50 | Produkta modelis |
| 11. | outer\_material | varchar | 50 | Produkta augšpuses materials |
| 12. | inner\_material | varchar | 50 | Produkta iekšpuses materials |
| 13. | outsole | varchar | 50 | Produkta zoles materials |
| 14. | color | varchar | 30 | Produkta krasa |
| 15. | designer | varchar | 255 | Produkta ražotais |
| 16. | article\_number | varchar | 30 | Produkta artikuls |

Tabula ”**Produkts**” saistīta ar tabulam “**Cart**”, “**Order\_items**” ” **Wishlist**” caur lauku product\_id

6.tabula

Tabulas **“Order\_items”** struktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr.** | **Nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| 1. | id | int | 100 | Unikālāis identifikātors |
| 2. | quantity | int | 10 | Produktu daudzums |
| 3. | price | int | 10 | Produkta prece |
| 4. | size | varchar | 10 | Produkta izmērs |
| 5. | product\_id | int | 100 | Produkta unikālāis identifikātors |
| 6. | order\_id | int | 100 | Pasūtijuma unikālāis identifikātors |

Tabula ”**Order\_items**” saistīta ar tabulam “**Order**” ”**Products**” caur lauku order\_id un product\_id.

# DATU STRUKTŪRAS

# *Sistēmas prasības*

Sistēma ir realizēta kā interneta tīmekļa vietne, kuras lietošana neprasa specifisku programmu instalāciju, kā arī var tikt palaista no jebkuras ierīces, kurai ir:

* Viena no jaunākajām pārlūkprogrammu versijām:
  + Safari — 14. versija vai jaunāka
  + Microsoft Edge — 113.0.0.0 vai jaunāka
  + Google Chrome — 113.0.5672.127 vai jaunāka
  + Opera — 99.0.4788.77 vai jaunāka
  + Mozilla Firefox — 113.0.1 vai jaunāka
* Stabils interneta savienojums – jo labāks savienojums, jo veiksmīgāk noritēs programmas darbība;
* Aparatūra, lai uz tās varētu darbināt iepriekš šajā sadaļā minētās pārlūkprogrammas.

# *Sistēmas* *instalācija un palaišana*

Šajā sadaļā tiks aprakstītas darbības, kas nepieciešamas, lai instalētu un palaistu sistēmu. Ja programmas nodošanas periodā sistēma būs pieejama tīmekļa domēnā, tiks norādīta arī konkrētā adrese, kā arī pieejas dati no visām sistēmā izmantojamajām lomām.

1. Prasības
   * Serveris: XAMPP (Apache, MySQL, PHP)
   * PHP: Versija 7.4 vai jaunāka
2. XAMPP instalēšana
3. Lejupielādējiet un instalējiet XAMPP no oficiālās vietnes: https://www.apachefriends.org/index.html
4. Palaidiet Apache un MySQL, izmantojot XAMPP kontrolpaneli.
5. Koda iegūšana
6. Lejupielādējiet projekta avota kodu no GitHub repozitorija.
7. Izpakojiet arhīvu un ielieciet projekta mapi “htdocs” direktorijā, kas atrodas XAMPP instalācijas direktorijā (parasti C:\xampp\htdocs).
8. Datubāzes iestatīšana
9. Atveriet phpMyAdmin, ievadot URL: http://localhost/phpmyadmin/
10. Izveidojiet jaunu datubāzi, piemēram, “shop\_db”.
11. Importējiet datubāzes struktūru un sākotnējos datus no “shop\_db.sql” faila:
    * + Atveriet izveidoto datubāzi phpMyAdmin.
      + Izvēlieties cilni "Import".
      + Augšupielādējiet shop\_db.sql failu un spiediet " Import".
12. Projekta palaišana
13. Pārliecinieties, ka Apache serveris ir palaists XAMPP kontrolpanelī.
14. Atveriet pārlūkprogrammu un ievadiet URL: (http://localhost/quickstep), lai pārbaudītu, vai sistēma darbojas pareizi.

**Piekļuves dati**

Lai pieslēgtos sistēmai, izmantojiet sekojošos piekļuves datus:

* Administratora loma:
* URL: http://localhost/your\_project\_directory/admin\_login.php
* Lietotājvārds: admin
* Parole: 1234
* Lietotāja loma:
  + URL: http://localhost/your\_project\_directory/user\_login.php
  + Lietotājvārds: test@gmail.com
  + Parole: 1234

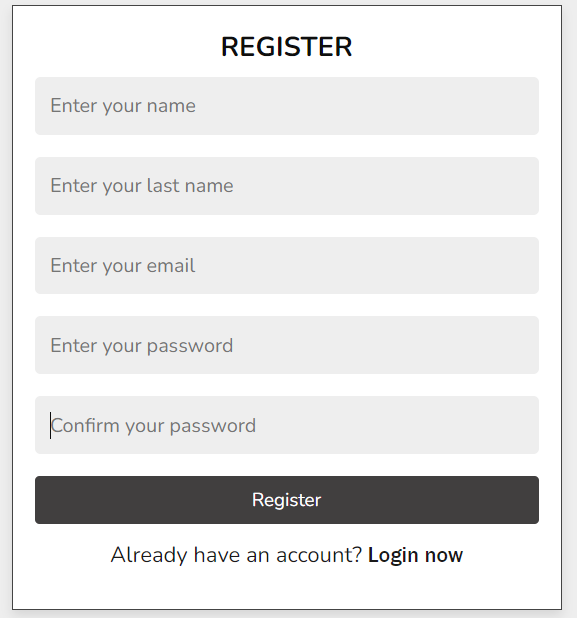
# *Programmas apraksts*

# Reģistrācija

Reģistrācijas process sākas ar piekļuvi reģistrācijas lapai, kur lietotājam tiek lūgts aizpildīt veidlapu ar nepieciešamo informāciju. Šī lapa ir pieejama, nospiežot uz "Reģistrēties" pogas galvenajā lapā. Reģistrācijas veidlapā ir šādi lauki:

* + Vārds: Teksta lauks, kurā lietotājam jāievada savs lietotājvārds. (Piemērs: "Kristaps").
  + Uzvārd: Teksta lauks, kurā lietotājam jāievada savs lietotājvārds. (Piemērs: "Fedosejevs").
  + E-pasts: Teksta lauks, kurā lietotājam jāievada savs e-pasta adrese. (Piemērs: "kristaps1@gmail.com").
  + Parole: Teksta lauks, kurā lietotājam jāievada parole. (Piemērs: "1234").
  + Apstiprināt paroli: Teksta lauks, kurā lietotājam atkārtoti jāievada parole, lai pārliecinātos par tās pareizību. (Piemērs: "1234").

Kad visi lauki ir aizpildīti, lietotājam jānospiež "Reģistrēties" poga, lai iesniegtu veidlapu. Ja visi ievadītie dati ir pareizi, lietotājs saņem paziņojumu par veiksmīgu reģistrāciju. Ja ir kādas kļūdas, piemēram, e-pasta formāts ir nepareizs vai parole nesakrīt, lietotājs saņems attiecīgu paziņojumu par kļūdu.



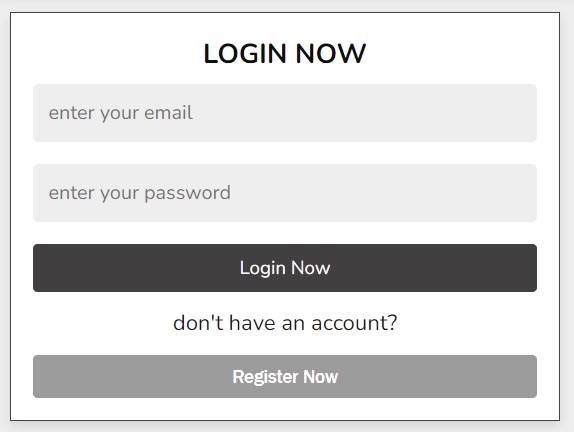
7.att. registrācija

# Autorizācija

Autorizācijas process ļauj lietotājam piekļūt savam kontam, ievadot pareizu e-pasta adresi un paroli. Šo procesu uzsāk, nospiežot "Autorizēties" pogu galvenajā lapā. Autorizācijas veidlapā ir šādi lauki::

* E-pasts: Teksta lauks, kurā lietotājam jāievada reģistrētā e-pasta adrese. (Piemērs: "kristaps1@gmail.com").
* Parole: Teksta lauks, kurā lietotājam jāievada reģistrētā parole.

.(Piemērs: "1234").



8.att. autorizācija

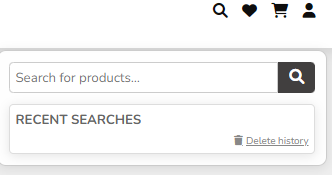
Pēc šo lauku aizpildīšanas lietotājam jānospiež "Pieslēgties" poga. Ja ievadītā informācija ir pareiza, lietotājs tiek novirzīts uz galveno lapu. Ja e-pasts vai parole ir nepareizi, lietotājam tiek parādīts paziņojums par kļūdu.

# Meklēšanas josla

Autorizācijas process ļauj lietotājam piekļūt savam kontam, ievadot pareizu e-pasta adresi un paroli. Šo procesu uzsāk, nospiežot "Autorizēties" pogu galvenajā lapā. Autorizācijas veidlapā ir šādi lauki:

* Meklēšanas josla: Meklēšanas lauks, kurā lietotājs var ievadīt meklējamo produktu nosaukumu vai kategoriju.

Lietotājs var meklēt produktus, izmantojot meklēšanas joslu galvenajā lapā. Meklēšanas rezultāti parādās sarakstā ar produktu attēliem, nosaukumiem, cenām un pieejamajiem izmēriem. Lietotājs var izmantot filtrus, lai sašaurinātu meklēšanas rezultātus pēc kategorijām, zīmoliem vai cenām.



8.att. Meklēšanas josla

# 

# *Testa peimērs*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nr.** | **Prasības**  **numurs** | **Prasības nosaukums** | **Ievaddati/situācijas apraksts** | **Sagaidāmais rezultāts** | **Statuss** |
| 1. | 1.1 | Jauna lietotāja reģistrācija | Pareizi dati: lietotāja vārds, lietotāja uzvrds, lietotāja e-pasts, 1. parole, 2. parole | Datu bāzē tiek pievienots ieraksts, Saskarē parādās paziņojums par veiksmīgu reģistrāciju | Pareizi |
|  |  |  | 1. parole, 2. parole  1234, 1235 | Paziņojums par nepareiziem datiem | Pareizi |
|  |  |  | Lietotāja e-pasts: janis#gmail.com | Paziņojums par lietotāja vārda nepareizo formātu | Pareizi |
| 2. | 2.1 | Lietotāja autorizācija | Pareizi dati: e-pasts, parole E-pasts: janis@gmail.com  Parole: Parole123 | Autorizācija veiksmīga, lietotājs tiek novirzīts uz galveno lapu | Pareizi |
|  |  |  | Nepareizi dati:  E-pasts: janis@gmail.com Parole: nepareizaParole | Paziņojums par nepareizu e-pasta vai paroles kombināciju | Pareizi |
| 3. | 3.1 | Produktu pievienošana grozam | Pareizi dati: Preces pievienošana grozam ar izmēra izvēliProdukta ID: 101 Izmērs: 42 | Paziņojums par nepieciešamību izvēlēties izmēru | Pareizi |
| 4. | 4.3 | Preces dzēšana | Pareizi dati: Pelēs klikšķis uz pogas "Dzēst" Produkta ID: 101 | Ieraksts tiek dzēsts no datubāzes un izņemts no groza, paziņojums "Prece ir dzēsta" | Pareizi |

Testēšana bija veiksmīga.

# NOBEIGUMS

"Projekta "Tīmekļa vietne sporta apavu interneta veikals Quick Step" izveide šobrīd ir pabeigta par 85%. Realizētās funkcijas ietver reģistrāciju un autorizāciju, produktu meklēšanu, izveidošanu, rediģēšanu un dzēšanu, lietotāja datu pārvaldību, produktu pievienošanu, dzēšanu un daudzuma rediģēšanu groza un vēlmju sarakstā, kā arī pasūtījumu izveidi un administrēšanas funkcionalitāti. Šīs funkcijas ir izstrādātas, izmantojot tādas tehnoloģijas kā PHP, JavaScript, MySQL un Visual Studio Code..

Šobrīd sistēma tiek izmantota lokālajā vidē, lai veiktu testēšanu un kļūdu novēršanu. Pēc pilnīgas funkcionalitātes pabeigšanas un rūpīgas testēšanas tā tiks palaista produkcijā, lai apkalpotu plašāku lietotāju loku. Nākotnē sistēmu varēs attīstīt un paplašināt, pievienojot jaunus moduļus un funkcijas, kas padarīs to par vēl pilnīgāku un efektīvāku e-komercijas platformu. Sistēmas modularitāte un izmantojamās tehnoloģijas ļauj viegli pielāgoties jaunām prasībām un attīstības virzieniem, padarot to piemērotu ilgtermiņa lietošanai un attīstībai..

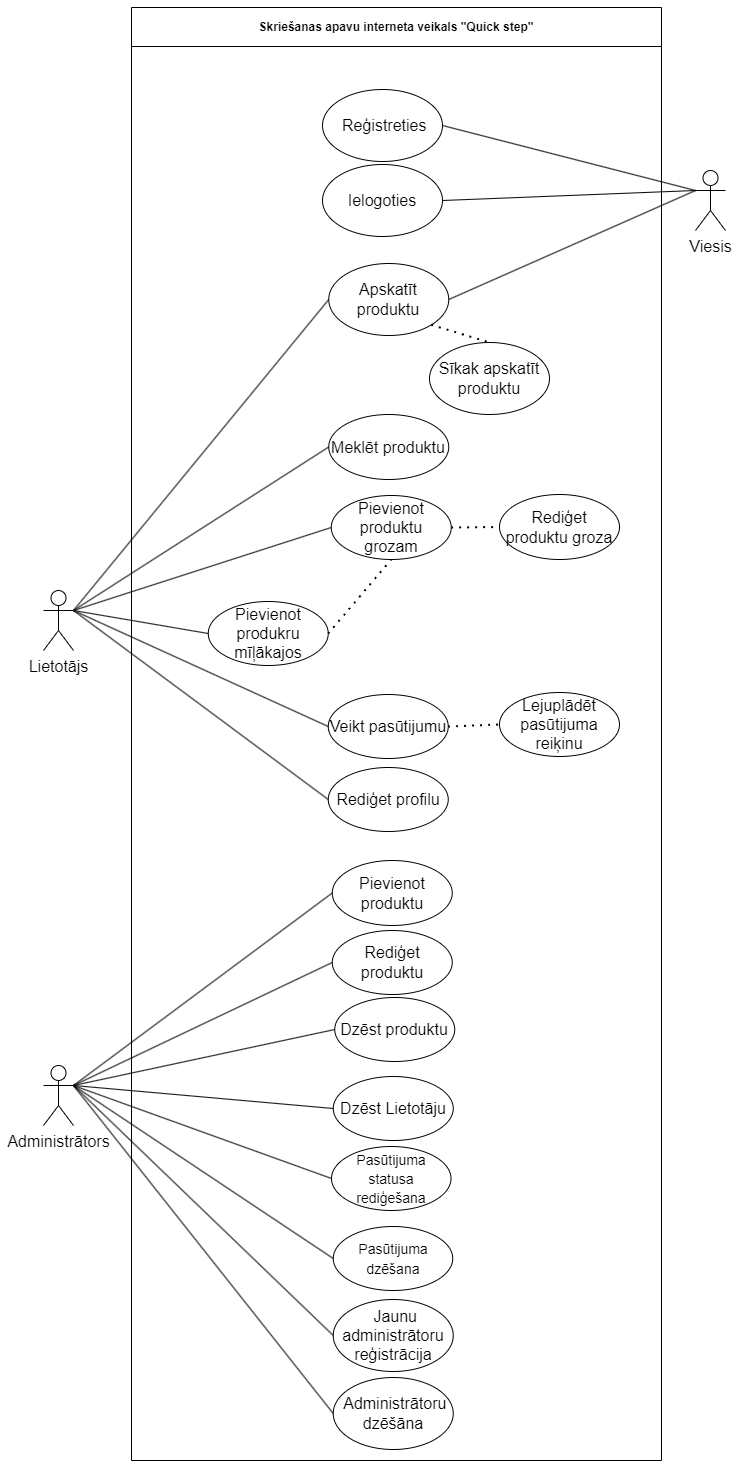
# INFORMĀCIJAS AVOTI

1. Javascript dokumentācija – <https://devdocs.io/javascript/> (Resurss apskatīts 04.11.2023)
2. CSS ievads - <https://www.w3schools.com/css/css_intro.asp>(Resurss apskatīts 04.11.2023)
3. MDN Web Docshttps HTML dokumtacija- <https://developer.mozilla.org/en-US/docs/Web/HTML>
4. PHP rokasgrāmata– [https://www.php.net/manual/en/index.php](https://www.php.net/manual/en/index.php%20) (Resurss apskatīts 05.11.2023)
5. MDN Web Docs ir dokumentācijas krātuve - <https://developer.mozilla.org/ru/> (Resurss apskatīts 01.10.2023)
6. <https://github.com/> (Resurss apskatīts 09.10.2023)
7. phpMyAdmin dokumentācija – <https://www.phpmyadmin.net/docs/> (Resurss apskatīts 02.10.2023)
8. XAMPP apmācība iesācējiem - <https://www.youtube.com/watch?v=r0lDDeVkaks> (Resurss apskatīts 19.09.2023)
9. TCPDF - PDF dokumentu ģenerēšanai - <https://tcpdf.org/> (Resurss apskatīts 19.05.2024)

**PIELIKUMI**

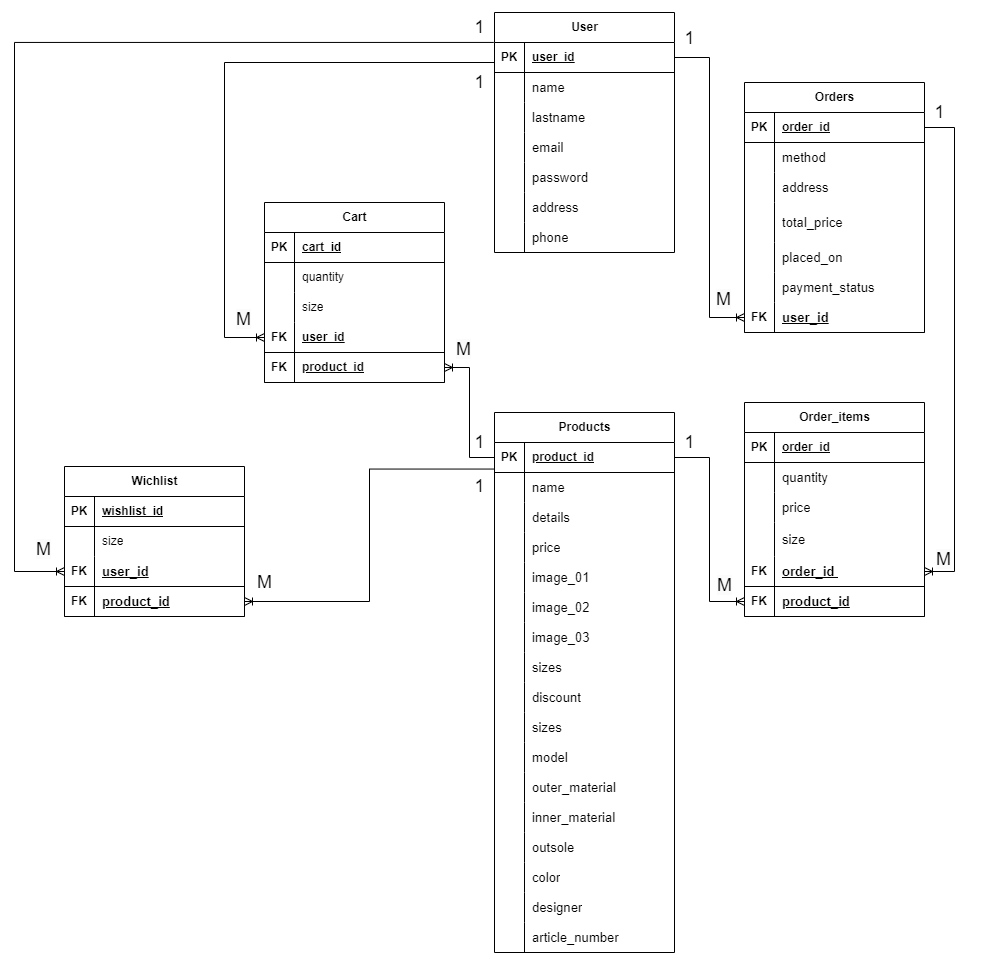
1. Pielikums

**use-case diagram**



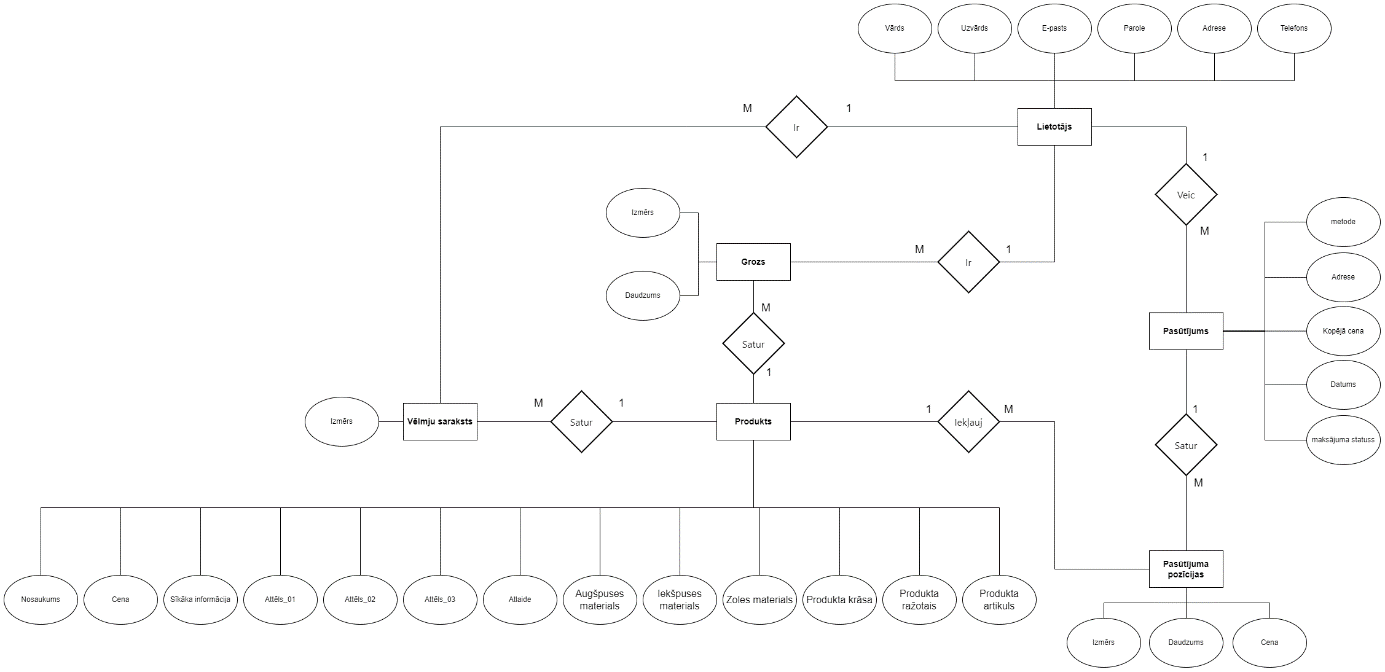
1. Pielikums

**saišu shēma**

****

1. Pielikums

**ER-diagramma**

****