

Liepājas Valsts tehnikums

Ģitāru tirdzniecības interneta veikals un tā administrēšanas sistēma

Kvalifikācijas eksāmena praktiskās daļas tehniskā dokumentācijas

|  |  |
| --- | --- |
| Izglītības programma | **33484011 Programmēšana** |
|  |  |
| Profesionālā kvalifikācija | **Programmēšanas tehniķis** |

|  |  |
| --- | --- |
| Projekta izstrādātājs | Endijs Dārznieks‎‎‎‎‎‎‎‎‎‎‎‎‎ ‎‎‎ ‎ ‎ ‎ ‎ ‎ ‎ ‎ ‎ ‎‎ ‎ ‎ ‎ ‎ ‎ ‎ ‎ ‎‎‎ ‎ ‎ ‎ ‎ ‎ ‎ ‎‎ ‎ ‎ ‎ ‎ ‎ |
|  | /vārds, uzvārds, paraksts/ |

Eksāmena datums 2023.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Liepāja 2023

Saturs

[Ievads 1](#_Toc137044815)

[1. Uzdevuma formulējums 2](#_Toc137044816)

[2. Programmatūras prasību specifikācija 3](#_Toc137044817)

[2.1. Produkta perspektīva 3](#_Toc137044818)

[2.2. Sistēmas funkcionālās prasības 4](#_Toc137044819)

[2.3. Sistēmas nefunkcionālās prasības 8](#_Toc137044820)

[2.4. Gala lietotāja raksturiezīmes 8](#_Toc137044821)

[3. Izstrādes līdzekļu, rīku apraksts un izvēles pamatojums 10](#_Toc137044822)

[3.1. Izvēlēto risinājuma līdzekļu un valodu apraksts 10](#_Toc137044823)

[3.2. Alternatīvo risinājuma līdzekļu un valodu apraksts 11](#_Toc137044824)

[4. Sistēmas modelēšana un projektēšana 13](#_Toc137044825)

[4.1. Sistēmas struktūras modelis 13](#_Toc137044826)

[4.1.1. Klašu diagramma / ER diagramma 13](#_Toc137044827)

[4.2. Funkcionālais un dinamiskais sistēmas modelis 14](#_Toc137044828)

[4.2.1. Aktivitāšu diagrammas 14](#_Toc137044829)

[4.2.2. Lietojumgadījumu diagrammas 16](#_Toc137044830)

[4.2.3. Sistēmas moduļu apraksts un algoritmu shēmas 18](#_Toc137044831)

[5. Lietotāju ceļvedis 25](#_Toc137044832)

[6. Testēšanas dokumentācija 26](#_Toc137044833)

[6.1. Izvēlētās testēšanas metodes, rīku apraksts un pamatojums 26](#_Toc137044834)

[6.2. Alternatīvās testēšanas metodes un rīki 26](#_Toc137044835)

[6.3. Testpiemēru kopa 26](#_Toc137044836)

[6.4. Testēšanas žurnāls 26](#_Toc137044837)

[Secinājumi 27](#_Toc137044838)

[Lietoto saīsinājumu un terminu skaidrojums 28](#_Toc137044839)

[Literatūras un informācijas avotu saraksts 29](#_Toc137044840)

[Pielikumi 30](#_Toc137044841)

Ievads

Es izvēlējos izveidot ģitāru tirdzniecības interneta veikalu un tā administrēšanas sistēmu. Šis projekts būtu noderīgs jaunam uzņēmumam, kurš vēlas izplatīties un pārdot savu preci, šajā gadījumā ģitāras, ar interneta palīdzību, kas paplašinātu pieejamo klientu skaitu un palielinātu uzņēmuma popularitāti, skatoties uz to ka mūsdienās cilvēki lielākoties izmanto visādus interneta pakalpojumus, lai dabūtu sev vajadzīgo. Kā arī šī sistēma būtu noderīga, dēļ iepriekš minētiem iemesliem, jau izveidotam uzņēmumam, kurš nodarbojas ar preču pārdošanu, bet nepārdod preces ar interneta palīdzību un pārdod tās tikai klātienē. Mana izveidotā sistēma pilnīgi modernizētu uzņēmumu ar mājaslapas priekšējo daļu, kur klienti varētu apskatīt pieejamās preces, izpētīt un iegūt vairāk informāciju par pašu uzņēmumu un nopirkt sev vajadzīgo. Un ar mājaslapas administratoru pusi strādātu administratori, kuri varētu pārvaldīt sistēmas datubāzi, veicot tās uzturēšanu. Pielāgojot šo sistēmu, tā varētu derēt jebkuram uzņēmumam ar interneta veikala nepieciešamību. Mana sistēma atšķirsies no citām līdzīgām sistēmām ar dizaina un funkcionālām īpašībām, piemēram, manā sistēmā būs iespēja nomainīt motīvu, šādu opciju es neesmu redzējis nevienā interneta veikalā, kā arī kategoriju izvēlne, piemēram, būs ļoti unikāla un nekur iepriekš neredzēta. Tālāk šajā dokumentācijā tiks aprakstīts, paskaidrots un attēlots viss mans projekts līdz mazākajai detaļai, kur varēs apskatīties daudz ko par pašas sistēmas uzbūvi, tās testēšanu un pat kā to izmantot no lietotāju un administratoru pusēm.

Iepazīties ar dokumentācijas elektronisko versiju, pašu sistēmu un tās datubāzi var šajā GitHub saitē: <https://github.com/EndijsD/projekts>.

1. Uzdevuma formulējums

Mana projekta uzdevuma formulējums ir izveidot funkcionālu un bez nekādām kļūdām, problēmām palaižamu un lietojamu, tirdzniecības interneta veikalu un tā administrēšanas sistēmu, kurš šajā gadījumā būs domāts ģitāru pārdošanai. Šis mērķis tiks sasniegts pielietojot visas manas prakses un skolas laikā iegūtās zināšanas un prasmes, ar skolotāja un prakses mentora palīdzību, kā arī pielietojot visādus citus resursus un izmantojot visādas mūsdienīgas tehnoloģijas un servisus priekš šī projekta izveides. Mērķis tiks sasniegts, kad būs izveidota sistēmai piemērota datubāze ar vairākiem ierakstu piemēriem datubāžu tabulās, pats projekts, kur mājaslapa sastāvēs no administratora un klienta puses un tur būs pieejamas vispopulārākās funkcijas šādās sistēmās un arī kāda retāk sastopama, kā arī API serveris, priekš sazināšanās ar datubāzi. Un protams, visam projektam, ir jābūt detalizēti aprakstītam dokumentācijā. Par šīs sistēmas nepieciešamību, es varu atsaukties uz ievadā teikto, ka jauniem vai jau izveidotiem uzņēmumiem, kuri nepārdod savas preces ar interneta palīdzību, šī sistēma var būt ļoti noderīga, lai paplašinātu savu klientu klāstu. Un, manuprāt, šie uzņēmumi vispār gribētu izvēlēties manu sistēmu, jo jau aktuālie un lietotie interneta veikali ir ļoti piebāzti ar visādām nevajadzīgām lietām, tāpēc es savu interneta veikalu veidošu minimalistisku un modernu, kā arī izveidošu vismaz vienu iepriekš neredzētu funkciju interneta veikalā, piemēram, motīva pārmainīšanas funkciju. Kā arī ar šīs sistēmas administrācijas pusi, internetveikalu būs viegli pārvaldīt un darbiniekiem būs viegli strādāt ar to, tāpēc ne tikai internetveikala lietotāji bet arī darbinieki būtu apmierināti ikdienā izmantot šo sistēmu.

1. Programmatūras prasību specifikācija

Šajā nodaļā tiks veikta programmatūras prasību specifikācijas apraksta izveide. Šī nodaļa sevī iekļauj četras apakšnodaļas, pēc kuru izlasīšanas jūs uzzināsiet daudz vairāk par pašu sistēmu un tās mērķi. Produkta perspektīvas apakšnodaļā tiks aprakstīts tāds kā trešās personas skats uz šo produktu, funkcionālās prasības aprakstīs pamata sistēmas funkcijas, kuras būs iespējams izmantot attiecīgi administratoriem un klientiem, nefunkcionālās prasības aprakstīs vispārējas nefunkcionāli saistītas lietas par pašu projektu un gala lietotāja raksturiezīmju apakšnodaļa aprakstīs šīs sistēmas pašu lietotāju aspektu.

* 1. Produkta perspektīva

Vispirms īsumā aprakstīšu savu produktu un tad aprakstīšu produkta perspektīvu no vairākiem skata punktiem. Mans piedāvātais produkts ir interneta veikals un tā administrēšanas sistēma, kur administrācijas sistēma atļaus administratoriem veikt datubāzes pārvaldīšanu un interneta veikals veikt preču pasūtīšanu, pirkšanu un informācijas uzzināšanu tā lietotājiem. Produkta perspektīvu savai sistēmai es varu paņemt pat no vairākiem skata punktiem.

Sāksim ar pirmo kas būtu no potenciālo sistēmas īpašnieku puses. No viņu puses skatoties uz šo sistēmu viņiem vajadzētu redzēt to kā biznesa paplašināšanas iespēju. Kā jau redzams mūsdienās viss pāriet uz tehnoloģisko pusi, uz internetu tāpēc mūsdienās, visiem uzņēmumiem ir jābūt savai mājaslapai, aplikācijai vai kaut kam līdzīgam un visi uzņēmumi tiecas uz to, jo internets ir vieglākais veids kā iegūt atbalstu, potenciālos klientus, ja to dara pareizi. Tāpēc es veidoju šo sistēmu, kura no potenciālo īpašnieku puses skatoties vajadzētu izskatīties minimalistisks un mūsdienīgs veids, kā dabūt savu veikalu interneta vidē un tādā veidā modernizētu savu uzņēmumu.

No uzņēmuma darbinieku puses skatoties mana izveidotā administrēšanas sistēma ir viegls veids kā pārvaldīt un pārskatīt uzņēmuma datubāzes datus, jo viss notiktu ar mājaslapas interfeisu, nebūtu jāraksta nekādas koda komandrindas vai jāveic datu manipulācija tieša saskarsmē ar datubāzē, piemēram, ar MySQLWorkbench interfeisu vai ar SQL vaicājumiem. Viņu darbs būtu atvieglināts un produktivitāte uzlabota.

Un visbeidzot produkta perspektīva no pašu interneta veikala lietotāju puses. Ja potenciālajiem lietotājiem interesētu preces, kas tiktu liktas šajā interneta veikalā, lai gan es veidoju to priekš ģitāru pārdošanas, teorētiski varētu to pārmainīt priekš jebkāda citu preču klāsta, tad no viņu skatu punkta viņi redzēs, viegli izmantojamu, minimalistisku ar patīkamu interfeisu interneta veikalu, kuram, būs pieejams motīva pārmainīšanas funkcija un unikāla kategoriju izvēlne. Tāpēc kopumā interneta veikala lietotājiem būs patīkama pieredze izmantojot šo aplikāciju.

* 1. Sistēmas funkcionālās prasības

P.1. Reģistrēšanās

Mērķis:

Funkcija “Reģistrēšanās” nodrošina iespēju lietotājam reģistrēties interneta veikalā, tādā veidā izveidojot savu kontu un saglabājot to datubāzē.

Ievaddati:

Nosacījumi:

1. laukā “Vārds” iespējams ievadīt ne vairāk kā 45 simbolus;
2. laukā “Uzvārds” iespējams ievadīt ne vairāk kā 45 simbolus;
3. laukā “Tālrunis” iespējams ievadīt ne vairāk kā 12 simbolus;
4. laukā “E-pasts” iespējams ievadīt ne vairāk kā 100 simbolus un starp
5. ievadītajiem simboliem obligāti jābūt “@” simbolam;
6. laukā “Pasta indekss” iespējams ievadīt ne vairāk kā 7 simbolus;
7. laukā “Pilsēta” iespējams ievadīt ne vairāk kā 60 simbolus;
8. laukā “Novads” iespējams ievadīt ne vairāk kā 60 simbolus;
9. laukā “Pagasts” iespējams ievadīt ne vairāk kā 60 simbolus;
10. laukā “Iela” iespējams ievadīt ne vairāk kā 60 simbolus;
11. laukā “Mājas nosaukums” iespējams ievadīt ne vairāk kā 60 simbolus;
12. laukā “Dzīvokļa numurs” iespējams ievadīt ne vairāk kā 60 simbolus;
13. laukā “Parole” iespējams ievadīt ne vairāk kā 100 simbolus;
14. laukā “Parole atkārtoti” iespējams ievadīt ne vairāk kā 100 simbolus.

Apstrāde:

Notiek pārbaude, vai vajadzīgie lauki ir aizpildīti un vai tie ir aizpildīti korekti.

Izvaddati:

1. interneta veikala autorizēšanās lapa;
2. paziņojums vai vizuāls attēlojums par nekorektu datu ievadi;
3. paziņojums vai vizuāls attēlojums par neaizpildītiem laukiem.

P.2. Autorizēšanās

Mērķis:

Funkcija “Autorizēšanās” nodrošina iespēju lietotājam autorizēties sistēmā un veikt

nepieciešamās darbības, kuras atšķiras un ir limitētas pēc autorizētā lietotāja – administratora vai interneta veikala lietotāja.

Ievaddati:

Jāievada sava e-pasta adrese un sava parole.

Apstrāde:

1. pārbauda, vai visi obligātie lauki ir aizpildīti ar vērtībām;
2. pārbauda, vai ievadītie dati sakrīt ar datu bāzē esošajiem datiem.

Izvaddati:

1. paziņojums vai vizuāls attēlojums par to, ka obligātajiem laukiem ir jābūt aizpildītiem;
2. paziņojums vai vizuāls attēlojums par to, ka e-pastā nav starp ievadītajiem simboliem ievadīts “@” simbols
3. paziņojums vai vizuāls attēlojums par to, ka kaut kas ir ievadīts nepareizi;
4. interneta veikala sākumlapa vai administratora paneļa pārskata lapa.

P.3. Izlogošanās

Mērķis:

Funkcija “Izlogošanās” nodrošina iespēju lietotājam vai administratoram izlogoties no sava konta.

Ievaddati:

Datorpeles kreisā taustiņa klikšķis uz izlogošanās pogas.

Apstrāde:

Izlogo lietotāju vai administratoru no sistēmas un pāradresē viņu uz attiecīgo lapu.

Izvaddati:

1. interneta veikala autorizēšanās lapa;
2. administratora paneļa autorizēšanās lapa.

P.4. Preces ielikšana grozā

Mērķis:

Funkcija “Ielikt grozā” nodrošina iespēju lietotājam attiecīgi izvēlēto preci saglabāt virtuālā grozā, priekš vēlākas apstrādes.

Ievaddati:

Datorpeles kreisā taustiņa klikšķis uz ielikt grozā pogu.

Apstrāde:

Izvēlētā prece tiek saglabāta virtuālajā grozā ar izvēlēto daudzumu, ja tā ir pieejama.

Izvaddati:

Virtuālais grozs ir ticis papildināts ar izvēlēto preci, izceļas groza ikona.

P.5. Preces skaita nomainīšana

Mērķis:

Funkcija “Preces skaita nomainīšana” nodrošina iespēju lietotājam nomainīt preces skaitu, ar ko attiecīgi papildināt grozu.

Ievaddati:

Nosacījumi:

1. poga pa kreisi no skaita laukuma, ja skaits nav viens;
2. poga pa labi no skaita laukuma, ja skaits nav pieejamības daudzums;
3. ievadīt skaitu uzreiz ievades laukumā, ar nosacījumu, kur nevar ievadīt mazāk par viens vai vairāk par pieejamo daudzumu.

Apstrāde:

Uzspiežot uz kādu no pogām preces skaits ievades laukā nomainīsies.

Izvaddati:

Preces skaits tiks nomainīts, kas būs atspoguļots ar skaita lauka palīdzību.

P.6. Pasūtījuma veikšana

Mērķis:

Funkcija “Pasūtījuma veikšana” nodrošina iespēju lietotājam pasūtījuma veikšanas formā ievadīt visus vajadzīgos datus, veikt pasūtījuma pārskatu un izpildīt savu pasūtījumu ar izvēlētajām precēm.

Ievaddati:

Aizpildīt sekojošos laukumus datu formā:

1. laukā “Vārds” iespējams ievadīt ne vairāk kā 45 simbolus;
2. laukā “Uzvārds” iespējams ievadīt ne vairāk kā 45 simbolus;
3. laukā “Tālrunis” iespējams ievadīt ne vairāk kā 12 simbolus;
4. laukā “E-pasts” iespējams ievadīt ne vairāk kā 100 simbolus un starp
5. ievadītajiem simboliem obligāti jābūt “@” simbolam;
6. laukā “Pasta indekss” iespējams ievadīt ne vairāk kā 7 simbolus;
7. laukā “Pilsēta” iespējams ievadīt ne vairāk kā 60 simbolus;
8. laukā “Novads” iespējams ievadīt ne vairāk kā 60 simbolus;
9. laukā “Pagasts” iespējams ievadīt ne vairāk kā 60 simbolus;
10. laukā “Iela” iespējams ievadīt ne vairāk kā 60 simbolus;
11. laukā “Mājas nosaukums” iespējams ievadīt ne vairāk kā 60 simbolus;
12. laukā “Dzīvokļa numurs” iespējams ievadīt ne vairāk kā 60 simbolus;

Aizpildīt sekojošos laukumus apmaksas formā:

1. laukā “Vārds, uzvārds uz kartes” iespējams ievadīt ne vairāk kā 100 simbolus;
2. laukā “Kartes numurs” iespējams ievadīt ne vairāk kā 16 ciparus;
3. laukā “Derīga līdz” iespējams ievadīt tikai mēnesi un gadu;
4. laukā “CVV” iespējams ievadīt ne vairāk kā 3 ciparus;

Un pēc veiksmīgām formu izpildēm veikt kreisā taustiņa klikšķi uz pirkšanas pogu.

Apstrāde:

Notiek pārbaude, vai vajadzīgie lauki ir aizpildīti un vai tie ir aizpildīti korekti. Pozitīvas aizpildīšanas gadījumā saglabā datus datubāzē.

Izvaddati:

1. paziņojums par veiksmīga pasūtījuma veikšanu;
2. paziņojums vai vizuāls attēlojums par nekorektu datu ievadi;
3. paziņojums vai vizuāls attēlojums par neaizpildītiem laukiem.

P.7. Sistēmas administratoru puses ierakstu pievienošana

Mērķis:

Funkcija “Ierakstu pievienošana” nodrošina iespēju administratoram pievienot jaunus ierakstus datubāzes tabulās.

Ievaddati:

1. peles kreisā taustiņa klikšķis uz pievienošanas pogu, kur tā ir pieejama;
2. aizpildīt korekti visus jaunā ieraksta laukus;
3. peles kreisā taustiņa klikšķis uz jaunā ieraksta saglabāšanas pogu.

Apstrāde:

Tiek mēģināts ievietot jauno ierakstu datubāzē.

Izvaddati:

1. paziņojums vai vizuāls attēlojums par veiksmīgu ieraksta pievienošanu;
2. paziņojums vai vizuāls attēlojums par neveiksmīgu ieraksta pievienošanu.

P.8. Sistēmas administratoru puses ierakstu dzēšana

Mērķis:

Funkcija “Ierakstu dzēšana” nodrošina iespēju administratoram izdzēst ierakstus no datubāzes tabulām.

Ievaddati:

1. datorpeles kreisā taustiņa klikšķis uz izvēlēto dzēšamo ierakstu;
2. datorpeles kreisā taustiņa klikšķis uz attiecīgā ieraksta dzēšanas pogu.

Apstrāde:

Tiek veikta attiecīgā ieraksta dzēšana no datubāzes.

Izvaddati:

1. veiksmīga ieraksta dzēšanas gadījumā tabula bez dzēstā ieraksta;
2. paziņojums vai vizuāls attēlojums par neveiksmīgu ieraksta dzēšanu.

P.9. Sistēmas administratoru puses ierakstu rediģēšana

Mērķis:

Funkcija “Ierakstu rediģēšana” nodrošina iespēju administratoram rediģēt datus datubāzes tabulās.

Ievaddati:

1. veikt datorpeles kreisā taustiņa dubultklikšķi uz tabulas ieraksta šūnu, atļauts tikai konkrētām šūnām;
2. veikt datu rediģēšanu;
3. peles kreisā taustiņa klikšķis uz attiecīgā ieraksta saglabāšanas pogu.

Apstrāde:

Tiek mēģināts atjaunināt attiecīgo ierakstu datubāzē.

Izvaddati:

1. paziņojums vai vizuāls attēlojums par veiksmīgu ieraksta rediģēšanu;
2. paziņojums vai vizuāls attēlojums par neveiksmīgu ieraksta rediģēšanu.
   1. Sistēmas nefunkcionālās prasības
3. Sistēmas saskarnei ir jābūt Latvijas Republikas valsts valodā.
4. Sistēmas saskarnei ir jābūt ērtai un ergonomiskai (tādai, kas minimizē lietotāja slodzi, piemēram, viegli uztveramai).
5. Parolēm datubāzē ir jābūt šifrētām.
6. Sistēmai ir jābūt atbalstītai kā minimums uz populārākajām Chromium pārlūkprogrammām.
7. Sistēmas saskarnei ir jābūt adaptīvai uz visādu ekrānu izmēriem.
   1. Gala lietotāja raksturiezīmes

Zinot to ka es veidoju ģitāru tirdzniecības interneta veikalu un tā administrēšanas sistēmu, praktiski būs 2 veida gala lietotāji – administratori, kuri izmantos administrēšanas sistēmu un interneta veikala lietotāji.

Administratori būs uzņēmuma darbinieki, kuri izmantos administrēšanas sistēmas pusi. Šiem darbiniekiem nebūs jābūt īpašas zināšanas par IT jomu vai programmēšanu, ja viņi izprot kā izmantot ikdienā sastopamas mājaslapu un aplikāciju funkcijas, tad viņi spēs izmantot administrēšanas sistēmu, viņiem būs pieejams datubāzes pārskats un datu manipulēšanas iespējas, pār šo datubāzi. Administratori būs atbildīgie par sistēmas datu uzturēšanu un atjaunošanu. Viņi spēs veikt visas CRUD operācijas administratoru mājaslapas pusē ar tās interfeisa palīdzību.

Un runājot par interneta veikala lietotājiem, tie var būt jebkuri cilvēki, kuri prot izmantot jau ikdienā sastopamas mājaslapas un aplikācijas, tāpat kā administratori. Interneta veikala lietotājiem nav obligāti jābūt klientiem, cilvēki, kuri vēlētos, piemērams, uzzināt vairāk informāciju par uzņēmumu, kurai piederētu šis interneta veikals arī būs iespēja to izdarīt, bet ja kāda no šīm personām arī vēlētos veikt pasūtījumu, tad šai personai būtu jābūt savai bankas kartei, lai veiktu apmaksu par veikto pirkumu. Un kontu interneta veikalā varēs uztaisīt arī jebkurš, pēc uzstādītajiem uzņēmuma noteikumiem, kurai piederētu šis interneta veikals, bet tas nebūs jāizveido obligāti, lai veiktu pirkumu.

1. Izstrādes līdzekļu, rīku apraksts un izvēles pamatojums

Šajā nodaļā varēs izlasīt, kāpēc es izvēlējos izmantot tieši tos līdzekļus un valodas, kurus es izmantoju šajā projektā un alternatīvos variantus, kurus varēju izmantot šī paša projekta veidošanai, bet neizmantoju.

* 1. Izvēlēto risinājuma līdzekļu un valodu apraksts

Es šajā projektā izvēlējos izmantot sekojošās valodas:

* JavaScript – Tā ir skriptu valoda, kas balstīta uz prototipu koncepta. Izvēlējos šo valodu, jo ar to programmai var uztaisīt gan servera un klienta pusi. Precīzāk man būtu jāsaka ka izmantoju React un Node.js un tas ir pateicoties tam ka man šīs lietas bija jāmācās prakses laikā, tāpēc šīs zināšanas izlēmu pielietot arī eksāmena veidošanai.
  + React – Tā ir bezmaksas un atvērtā koda klienta puses JavaScript bibliotēka lietotāja interfeisa veidošanai, pamatojoties uz komponentiem.
  + Node.js – Tā ir atvērtā pirmkoda, starpplatformu JavaScript reāllaika vide, kas ļauj izpildīt JavaScript kodu servera pusē.
* HTML – Tā ir iezīmēšanas valoda, kas ir izstrādāta tīmekļa lappušu un citas pārlūkprogrammā attēlojamas informācijas glabāšanai.
* CSS – Tā ir īpaša stila lapas valoda, ko lieto, lai aprakstītu izskatu iezīmēšanas valodā veidotiem dokumentiem. Bet specifiskāk lielākoties visur es izmantoju MUI styled() rīku, ko es arī prakses laikā esmu uzzinājis un iemācījies.
  + MUI styled() – Tas ir rīks stilizētu komponentu veidošanai. Atļauj rakstīt CSS sintaksi JavaScript valodā.

Un sekojošos līdzekļus:

* Visual Studio Code – Tas ir koda redaktors. Mūsdienās ļoti populārs un plaši izmantots. Es izvēlējos izmantot šo koda redaktoru, dēļ tā paplašinājuma veikala un klāsta, nevienam citam koda redaktoram nav pieejams tik plašs klāsts ar paplašinājumiem, kā arī izmantoju to dēļ tā plašajām funkcijām. Tur var pielāgot pat mazākos redaktora iestatījumus priekš savām vajadzībām, ko es esmu veicis, citus koda redaktorus tik detalizēti nav iespējams modificēt. Tur arī ir uzreiz pieejams logs ar termināļu atvēršanu pašā redaktorā, kā, piemēram, tev var būt atvērts terminālis priekš komandrindu izpildīšanas un terminālis priekš Git darbībām.

Manis izmantotie Visual Studio Code paplašinājumi:

* + Auto Rename Tag – Automātiski pārdēvē pārī savienotos HTML/XML tagus.
  + Code Spell Checker – Koda pareizrakstības pārbaudītājs.
  + Color Highlight – Koda redaktorā iezīmē CSS/tīmekļa krāsas.
  + ESLint – Integrē ESLint koda redaktorā (Tas atrod problēmas kodā).
  + Image preview – Rāda attēla priekšskatījumu koda redaktorā.
  + Prettier - Code formatter – Koda formatētājs.
  + REST Client – Ļauj nosūtīt HTTP pieprasījumus un skatīt atbildes koda redaktorā.
  + Simple React Snippets – Saīsināti koda fragmenti.
  + vscode-styled-components – Sintakses izcelšana priekš styled-components (Tai skaitā MUI styled()).
* Word 2016 – Tas ir teksta procesors, redaktors.
* MySQL Workbench – Tas ir vizuāls datubāzes dizaina rīks balstīts uz MySQL datubāzes sistēmas. Izmantoju to, jo skolā mums šo rīku un datubāzes veidu ļoti padziļināti mācīja, tāpēc es esmu pārliecināts ar savām spēja par to. Un neskaitot to, ka izprotu MySQL datubāzes sistēmu un saprotu, kā strādāt ar MySQL Workbench, šie rīki arī ir attiecīgi labi sistēmai ko es izstrādāju, kas ir interneta veikals un tur relāciju datubāze ir piemērota un ar MySQL Workbench palīdzību viegli pārvaldāma un izveidojama tās izstrādē.
* Git – Tā ir versiju kontroles sistēma.
* GitHub – Tas ir tīmeklī balstīts Git repozitoriju mitināšanas pakalpojums.
* Trello – Tā ir kanban stila sarakstu veidošanas lietojumprogramma. Lai gan priekš šī projekta izstrādes, sāku izmantot šo aplikāciju, tikai tad kad biju jau pusē ticis projekta izstrādē, manuprāt to ir tāpat vērts pieminēt. Trello es izmantoju, lai, piemēram, viss kas jāizdara nebūtu jāatceras no galvas un lai sadalītu uzdevumus veicamos soļos. Trello esmu pats apguvis, veicot spēļu izstrādi.
  1. Alternatīvo risinājuma līdzekļu un valodu apraksts

Alternatīvās valodas, kuras es varēju izmantot:

* PHP – Tā ir servera skriptu valoda un rīks dinamisku un interaktīvu mājaslapu izveidei. Lai gan esmu izmantojis un mācījies šo valodu skolā, manuprāt, JavaScript ir daudz universālāka valoda. PHP man likās neorganizēta un limitēta valoda. Lai gan es varu izmantot PHP ar JavaScript, bet skatoties uz to ka es varu JavaScript izmantot pilnīgi visam gan servera un klienta pusei, pat priekš CSS, tad es labāk izvēlos to.
* Python – Tā ir augsta līmeņa programmēšanas valoda. Tā ir interpretējama objektorientētā skriptu valoda. Esmu tikai pats minimāli mācījies šo valodu un pat uztaisījis ļoti vienkāršotu internetveikala mājaslapu, bet tas bija ļoti sen, tāpēc zināšanas pār šo valodu vairs, praktiski, man nav.

Alternatīvie līdzekļi, kurus es varēju izmantot:

* Notepad++ vai Sublime Text 4 – Tie ir koda redaktori. Lai gan esmu izmantojis abus no šiem koda redaktoriem, manuprāt VS Code ir labākā opcija. Notepad++ interfeisa dizains liekas novecojis un lai gan Sublime Text 4 interfeiss ir moderns, pēc informācijas sameklēšanas esmu uzzinājis, ka šajos redaktoros es nevarētu atrast un izmantot visus paplašinājumus kādi man ir VS Code.
* MariaDB, phpMyAdmin, MongoDB un HeidiSQL – Pirmos trīs risinājumus es esmu izmantojis un tās varēja būt alternatīvas pret MySQL Workbench, bet man ar tām ir ļoti maz pieredze salīdzinājumā ar MySQL Workbench, tāpēc tas ir lielākais iemesls pret visiem šiem risinājumiem. Lai gan MariaDB skaitās kā labāka MySQL versija, man būtu jāizmanto tā ar komandrindu palīdzību vai ar rīku kā HeidiSQL, kurš arī sader ar MySQL, bet kā jau esmu teicis man ir ļoti lielas zināšanas par MySQL Workbench, tāpēc neredzu iemeslu izmantot un iemācīties kaut ko citu, specifiski priekš šī projekta, ja jau esošais risinājums ir ļoti labs.
* ClickUp – Tas ir sadarbības un projektu pārvaldības rīks. Šis ir vienīgais alternatīvs, ko es varu nozīmīgi minēt, priekš Trello, jo šo aplikāciju es izmantoju savas prakses laikā, priekš projekta izstrādes. Lai gan abas aplikācijas ir līdzīgas es izvēlējos Trello, jo ClickUp no manas pieredzes liekas lēnāks un arī es iepriekš citiem saviem projektiem esmu izmantojis Trello, kā pārvaldības aplikāciju, tāpēc man ar to jau ir daudz pieredze un kopumā jūtos ērtāk strādāt ar to.

1. Sistēmas modelēšana un projektēšana

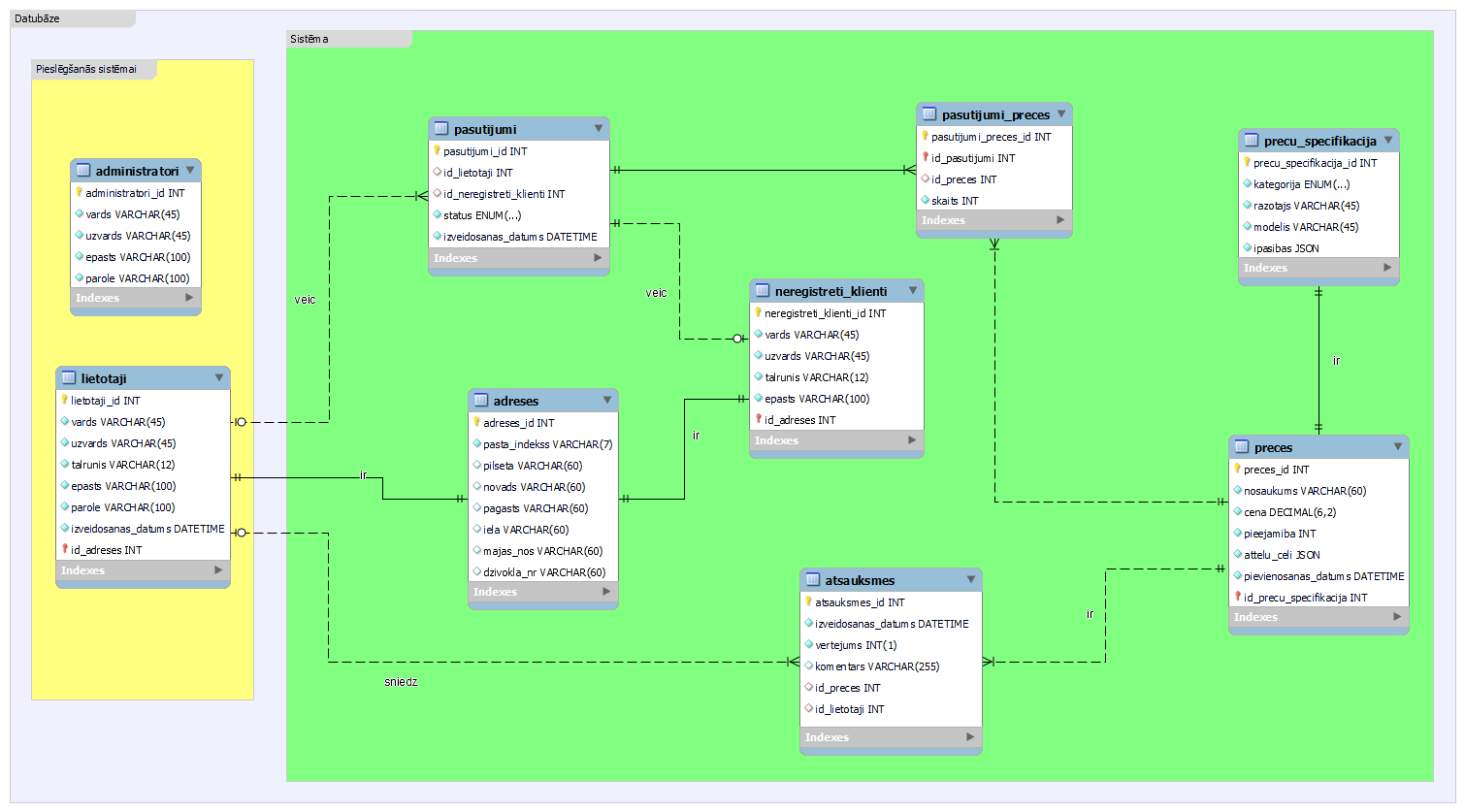
Datubāzes modelēšana bija patiesībā ļoti grūta, vajadzēja veikt daudz informācijas meklēšanu un izpēti, kā arī palīdzības prasīšanu, kam varēju, bet kad viss par modelēšanu bija izdomāts un skaidrs, tad tās projektēšana bija vienkārša, pateicoties MySQL Workbench. Un sistēmas modelēšana bija izaicinoša dažās vietās, kā, piemēram, administrācijas sistēma un tās datu tabulas, bet kad šiem dažiem šķēršļiem tiku pāri tālākā projektēšana norisinājās bez problēmām. Nākamajās apakšnodaļās varēs apskatīties vairākas diagrammas par manu sistēmu un tās datubāzi, kā arī izlasīt aprakstus par katru apakšnodaļu.

* 1. Sistēmas struktūras modelis

Šajā apakšnodaļā no klašu diagrammas un vai ER diagrammas es attēlošu ER diagrammu, jo skatoties uz to ka es veidoju mājaslapu, tur nav klases, lai gan nenoliedzami es izmantoju JavaScript valodu, kas atbalsta klases un React bibliotēku, kura arī atbalsta klases, bet tas nav ieteicams un labākā prakse, tas tikai padarītu kodu sarežģītāku neko neiegūstot.

* + 1. Klašu diagramma / ER diagramma

Šeit ir redzama entītiju attiecību diagramma, kur kā nosaukumā jau saprotams var redzēt attiecības (saites) starp datubāzes tabulām, palīgvārdus uz saitēm un pašas tabulas un to laukus. Es, lai vieglāk pārskatītu, esmu atdalījis sistēmas pieslēgšanās tabulas (kontu tabulas) no pārejās sistēmas, tās ir ar dzeltenu krāsu aizmugurē un pārējā sistēma ir ar zaļu.



1.attēls. ER diagramma

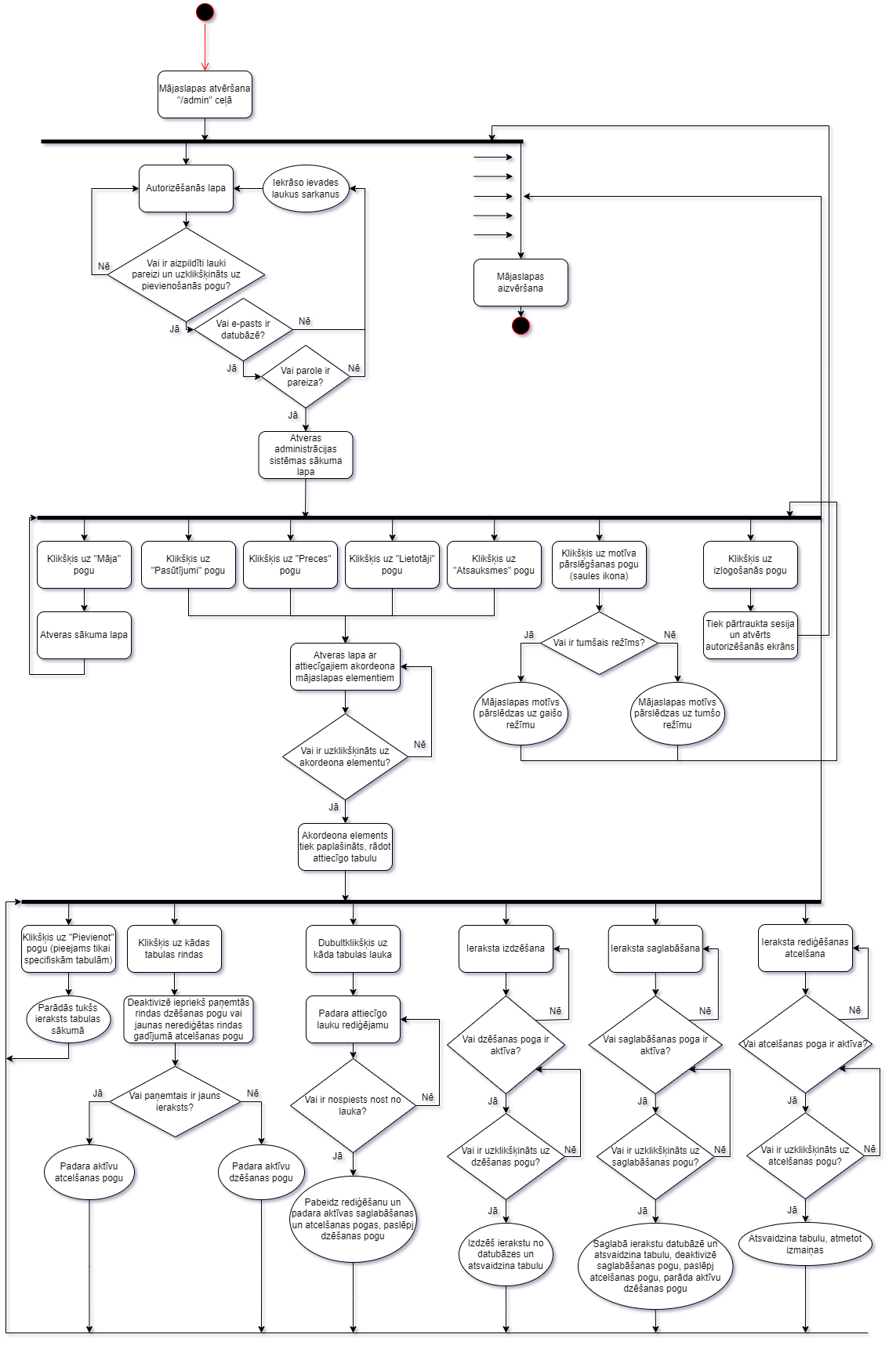
* 1. Funkcionālais un dinamiskais sistēmas modelis

Šeit var apskatīties vairāk par pašu sistēmu, tas ir, par interneta veikalu un tā administrēšanas pusi. Var apskatīties kā tā darbojas sīki detaļās aktivitāšu diagrammās un plašākā skatā lietojumgadījumu diagrammās. Kā arī sistēmas moduļu aprakstā un algoritmu shēmas var iepazīties vairāk ar manis veidotajiem sistēmas moduļiem un tie ir no kā šī mājaslapa sastāv un tiek veidota.

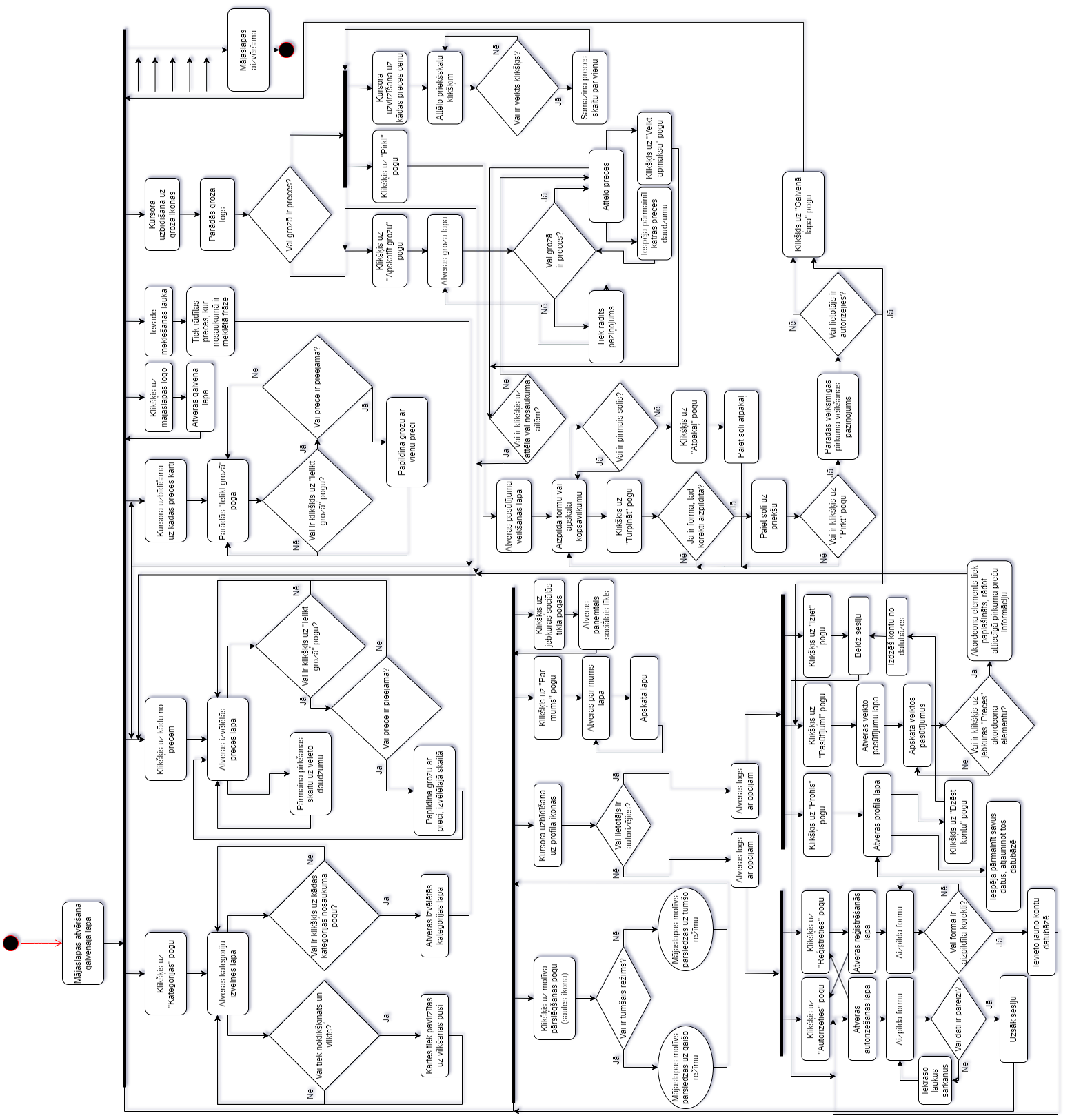
* + 1. Aktivitāšu diagrammas

Šeit var apskatīties aktivitāšu diagrammas manai sistēmai, kurās, praktiski, var redzēt visu sistēmas darbību scenārijus un iznākumus, balstoties uz veiktajām darbībām. Skatoties uz to ka veidoju interneta veikalu un administratora sistēmu, šis nav mazs projekts, tāpēc dēļ sekojošajiem iemesliem veidošu administratora un interneta veikala lietotāja pusēm atsevišķas aktivitāšu diagrammas. Pirmkārt, pārskatāmības dēļ, jo no abām sistēmas pusēm sanāk izveidot ļoti lielas diagrammas. Otrkārt, mājaslapas pušu atšķirību dēļ, jo abas sistēmas puses, kas ir interneta veikala un administratora, var pat uzskatīt par pavisam dažādām sistēmām, dēļ to atšķirīgajām funkcijām un lietotājiem.

2.attēls. Aktivitāšu diagramma administratora pusei



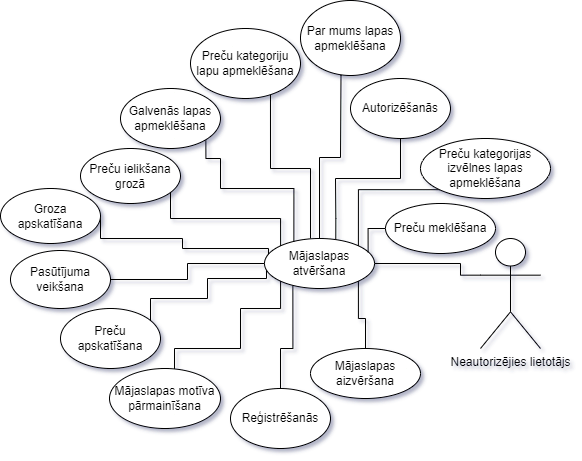
3.attēls. Aktivitāšu diagramma lietotāja pusei



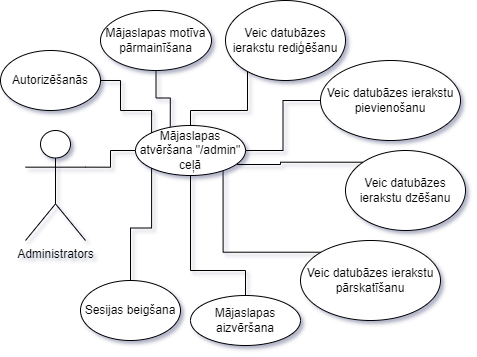
* + 1. Lietojumgadījumu diagrammas

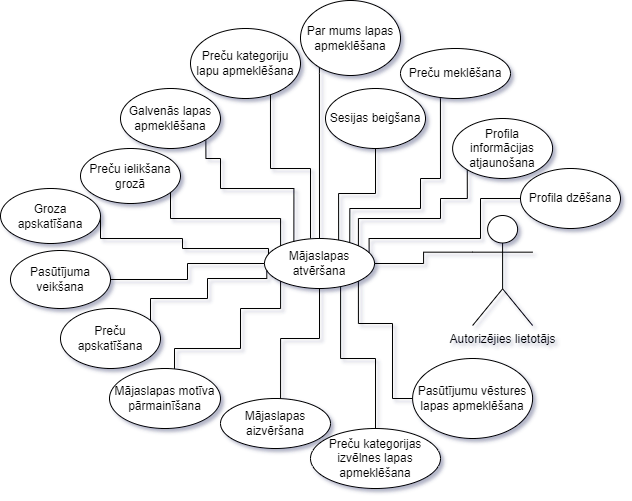
Šeit var apskatīties priekš manas sistēmas izveidotas lietojumgadījuma diagrammas, tās, praktiski, ir vienkāršotas aktivitāšu diagrammas un domātas priekš vispārējas iespējamās funkcionalitātes apskata, skatoties no sistēmas lietotāja skatupunkta. Manā gadījumā ir trīs lietotāju veidi: autorizējies un neautorizējies interneta veikala lietotājs, kā arī administrators.

4.attēls. Neautorizēta lietotāja lietojumgadījuma diagramma



5.attēls. Administratora lietojumgadījuma diagramma

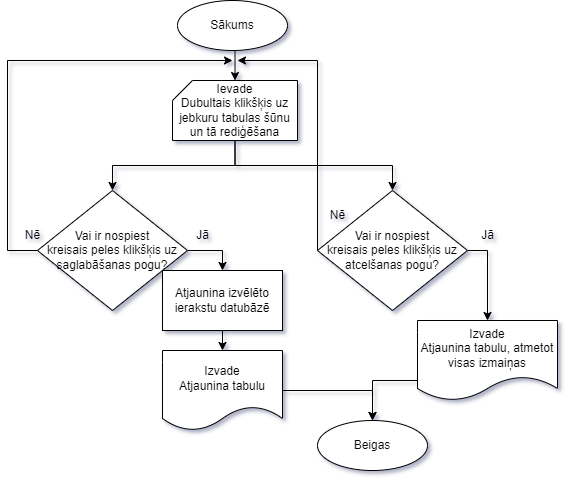


6.attēls. Autorizēta lietotāja lietojumgadījuma diagramma

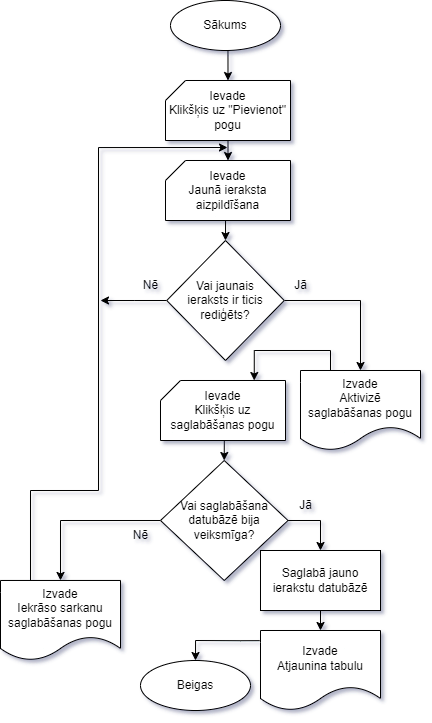
* + 1. Sistēmas moduļu apraksts un algoritmu shēmas

Kopumā man mājaslapai ir daudzi sistēmas moduļi, kas veido visu sistēmu kopā, bet jāsaka ka vairāki moduļi ir mazi un pat loģika ir sadalīta, kā, piemēram, grozs, kurš nav viens liels modulis, bet gan ir vairākās vietās pamanāms. Tāpēc es esmu izveidojis algoritmu shēmas lielākajiem sistēmas moduļiem un pēc šo diagrammu apskatīšanas jūs varēsiet izprast vairāk to loģiku, kā es esmu veidojis tos un manu domāšanu aiz tiem.

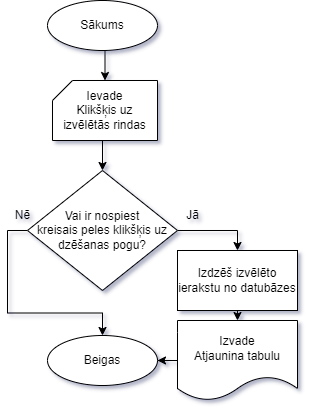
7.attēls. Administratora datu atjaunošanas modulis



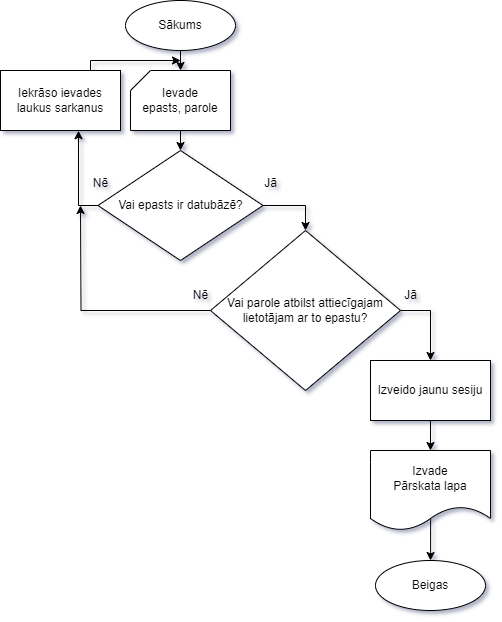
8.attēls. Administratora datu pievienošanas modulis



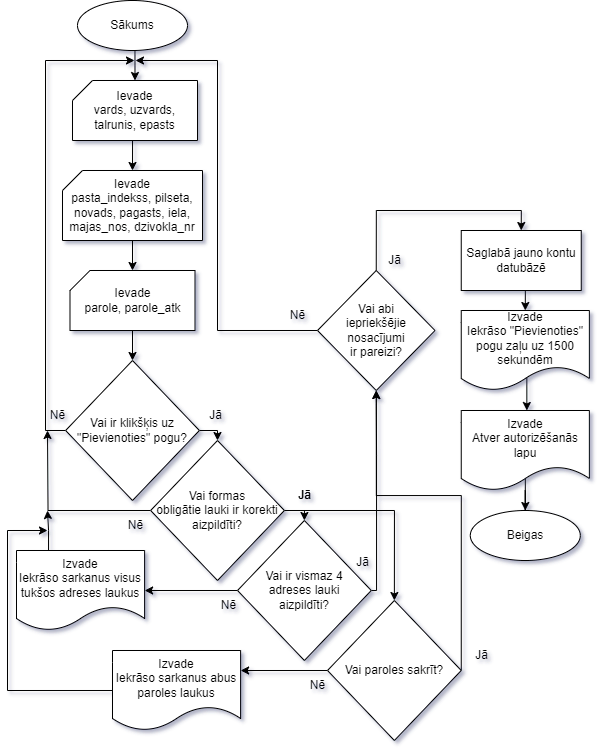
9.attēls. Administratora datu dzēšanas modulis



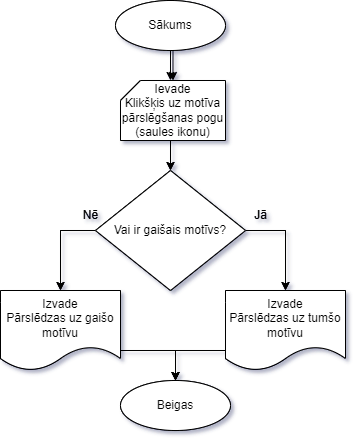
10.attēls. Autorizēšanās modulis



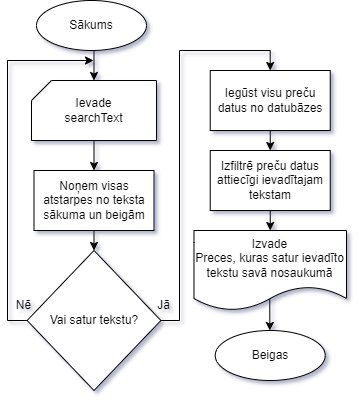
11.attēls. Reģistrēšanās modulis



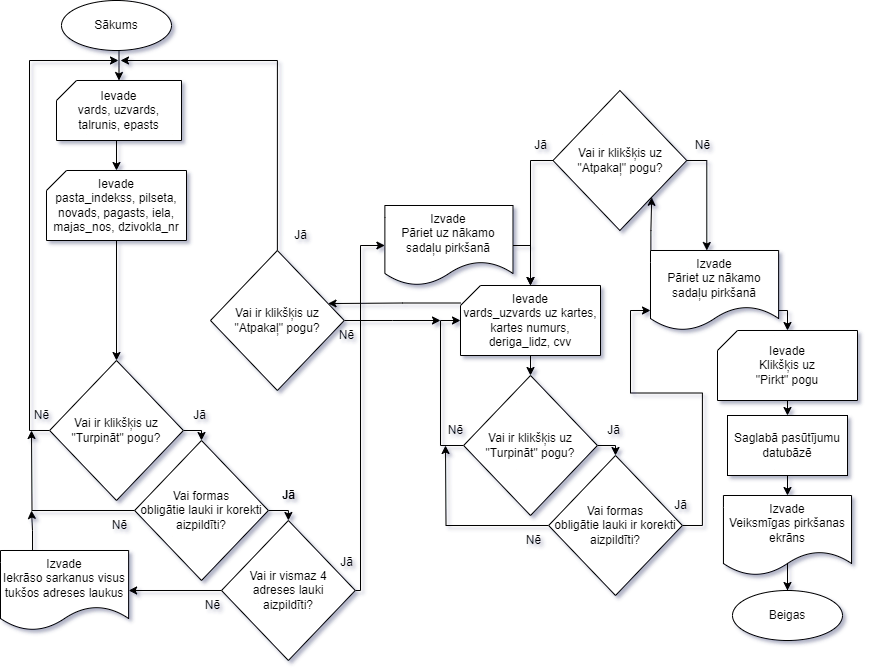
12.attēls. Motīva pārslēgšanas modulis



13.attēls. Preču meklēšanas modulis



14.attēls. Pirkšanas modulis



1. Lietotāju ceļvedis
2. Testēšanas dokumentācija
   1. Izvēlētās testēšanas metodes, rīku apraksts un pamatojums
   2. Alternatīvās testēšanas metodes un rīki
   3. Testpiemēru kopa
   4. Testēšanas žurnāls

Secinājumi

Projekta izveidi es uzsāku 04.02.2023, tātad kopā es veltīju apmēram 4 mēnešus, priekš tās izveides. Šis ir ilgākais un lielākais projekts, ko es savā programmēšanas karjerā esmu izveidojis. Un tas nav tikai salīdzinājumā ar kaut kādiem skolas projektiem, es, piemēram, esmu pats veicis vairāku spēļu izstrādi, kuri lai gan arī bija lieli projekti, bet šis projekts pārspēj tos visus. Projekta uzsākšana bija ļoti lēna, jo šis ir ļoti liels projekts un tam ir jāvelta ļoti daudz darbs, kā arī uzsākot projektu, nebija tā ka es zināju visu, ko vajadzēja, no galvas. Man, šī projekta izpildes laikā, vajadzēja apgūt ļoti daudz jaunas zināšanas gan pašam meklējot un apgūstot tās, gan ar skolas skolotāja un prakses mentora palīdzību.

Kā jau es iepriekš dokumentā esmu minējis, daudzas iemaņas šī projekta izstrādē man nāk no prakses, jo tur es apguvu tehnoloģijas, kuras šeit es pielietoju, bet es arī pats esmu iemācījies šīs tehnoloģijas vairāk un apguvis ko jaunu. Es esmu paplašinājis savas dizainera prasmes, servera puses kodēšanas zināšanas un vispārīgus mājaslapu izstrādes principus. Protams, šis projekts arī sastāv no datubāzes, kuras izstrādi es veicu ļoti cītīgi un detalizēti, tas man, vienkārši, paplašina pieredzi datubāžu izstrādē. Un veidojot dokumentāciju, es atkārtoju skolā apgūtās prasmes, kā, piemēram, visādu diagrammu izstrādi un strukturizētu teikumu veidošanu.

Projekta izstrāde noteikti bija interesanta un aizraujoša, jo man bija pilnīga brīva izvēle un kontrole par to, ko es taisu. Lai gan no vienas puses liekas ka es taisu ierastu lietu, kas ir interneta veikals, patiesībā, es mēģināju katrā brīdī apstāties un padomāt, kā es varētu kaut ko uztaisīt unikālāk, lietderīgāk un vai, vienkārši, interesantāk, katru šo raksturojošo vārdu es arī, manuprāt, esmu atspoguļojis ļoti labi savā sistēmā.

Pabeidzot šo projektu, pat ja es prezentēju to kā pabeigtu sistēmu, īstenībā, es jau esmu izplānojis ļoti daudz jaunus uzdevumus un idejas uz priekšu, jo projekta izstrādes laikā es varēju saskatīt daudzas funkcijas, ko es tā arī beigās neieviesu un daudzas uzlabojamas vietas. Lai gan es neesmu drošs par projekta turpināšanu, es zinu ka es esmu izlaidis pilnīgi funkcionālu un bez problēmām lietojamu pirmo versiju, kura, manuprāt, ir sanākusi labi.

Lietoto saīsinājumu un terminu skaidrojums

* API (no angļu val. *Application programming interface*) – Lietojumprogrammas saskarne ir iepriekš definētu klašu, procedūru, funkciju, struktūru un konstanšu kopums, kas tiek pasniegts kā interfeisa tips, kuru iespējams izmantot ārējiem programmatūras produktiem.
* Bibliotēka (angļu val. *Library*) – Bibliotēkas programmas ir paredzētas atkārtotai lietošanai un var tikt izmantotas jaunu programmu veidošanai.
* CRUD (no angļu val. *Create, read, update and delete*) – Datorprogrammēšanā izveide, lasīšana, atjaunināšana un dzēšana ir četras pastāvīgās pamatdarbības, kuras moduļiem būtu jāveic kā minimums.
* Chromium – Chromium ir bezmaksas atvērtā pirmkoda tīmekļa pārlūkprogrammas projekts, ko galvenokārt izstrādā un uztur Google.
* Datubāze (angļu val. *Database*) – Datubāze ir informācijas kopums ar noteiktu struktūru.
* IT (no angļu val. *Information technology*) – Informācijas tehnoloģija ir zināšanu, metožu, paņēmienu un tehniskā aprīkojuma kopums, kas ar datoru un sakaru līdzekļu starpniecību nodrošina jebkuras informācijas iegūšanu, glabāšanu un izplatīšanu.
* Klase (angļu val. *Class*) – Klase ir apraksts, pēc kura tiek izveidoti objekti. Šajā aprakstā tiek aprakstīti jeb deklarēti objekta iekšējie mainīgie un funkcijas.
* SQL (no angļu val. *Structured query language*) – SQL ir vaicājumu valoda, kas paredzēta datu manipulēšanai relāciju datubāžu pārvaldības sistēmās.
* Serveris (angļu val. *Server*) – Serveris ir datorsistēma klienta servera arhitektūrā, kas nodrošina klientu pieprasījumu apstrādi un nosūta tiem atbildes.

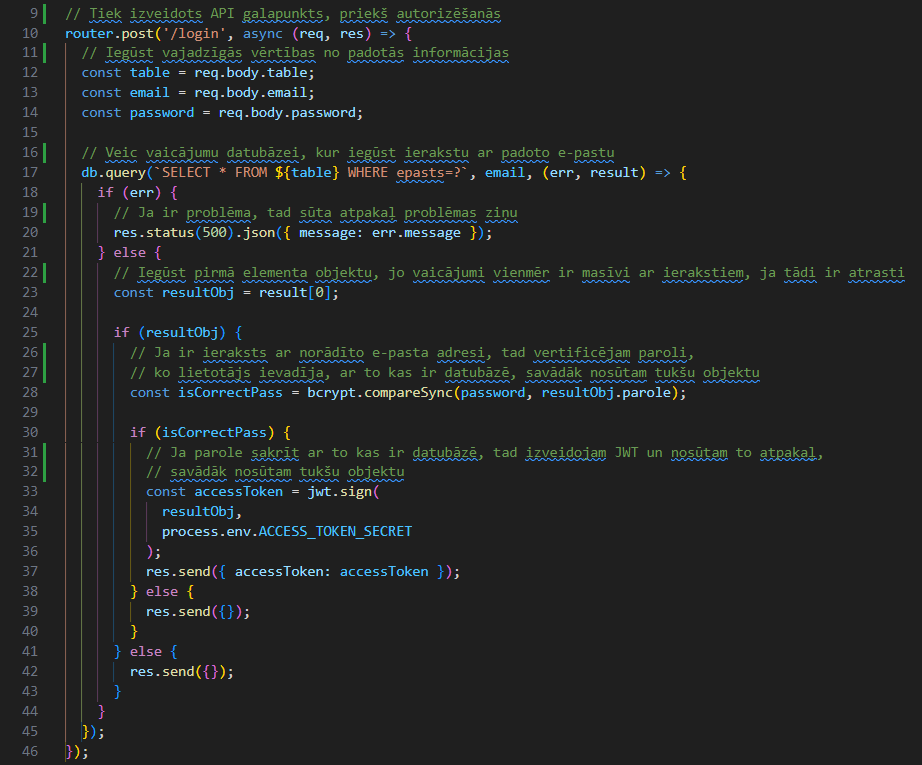
Literatūras un informācijas avotu saraksts

* docs.railway.app/develop/variables#reference-variables
* github.com/mui/material-ui/issues/9427
* makingofsoftware.com/2020/02/product-requirements-specification-product-perspective/
* mui.com
  + mui.com/material-ui/react-app-bar/
  + mui.com/material-ui/react-grid/
  + mui.com/x/api/data-grid/data-grid/
* stackoverflow.com
  + stackoverflow.com/questions/68347965/how-to-customise-datagrid-footer-by-adding-a-new-field-next-to-the-pagination
  + stackoverflow.com/questions/67970440/how-to-make-material-data-grid-width-to-fill-the-parent-component-in-react-js
  + stackoverflow.com/questions/23510627/where-to-store-orders-for-non-registered-users-on-e-commerce-site/23512500
* taimoorsattar.com/blogs/format-input-text-while-typing-javascript
* youtube.com
  + youtube.com/watch?v=1HamqOuv2Cw
  + youtube.com/watch?v=re3OIOr9dJI&t=2256s
  + youtube.com/watch?v=htGfnF1zN4g

Pielikumi

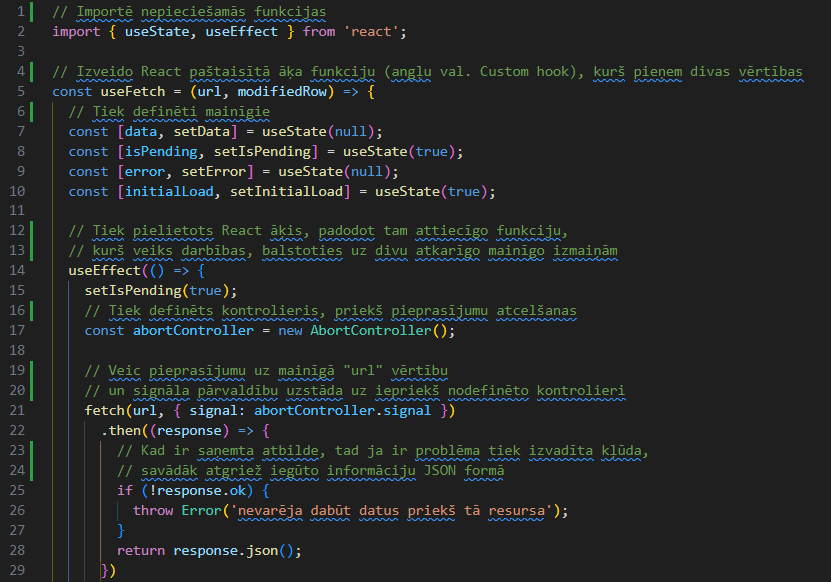
1.pielikums

Servera puses autorizēšanās API



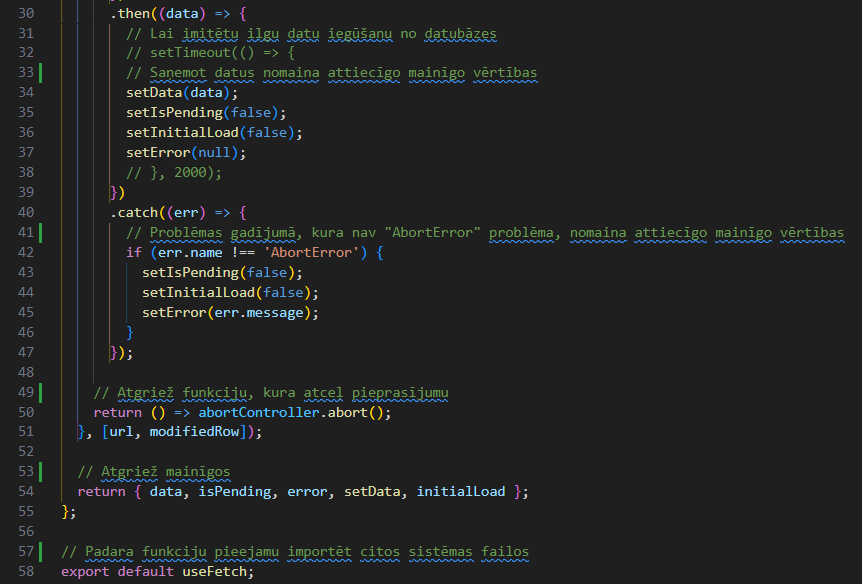
2.pielikums

Paša izveidotā React āķa funkcijas (Custom hook) sākums



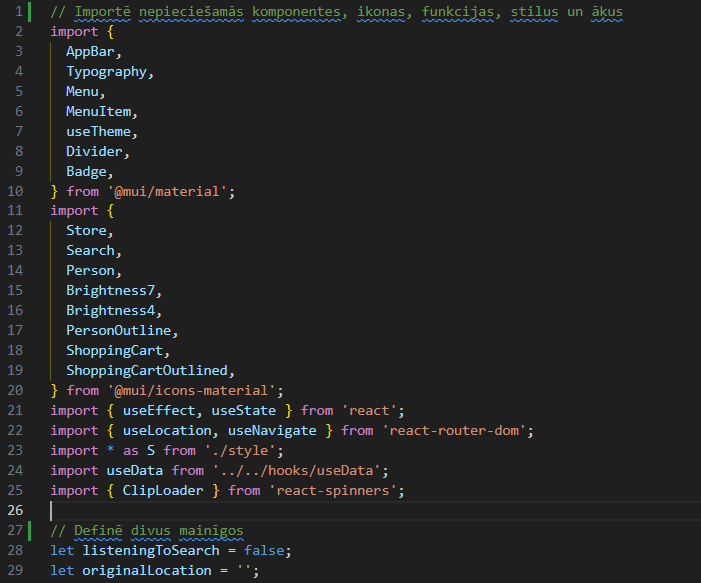
3.pielikums

Paša izveidotā React āķa funkcijas (Custom hook) beigas



4.pielikums

Lietotāju puses navigācijas joslas funkcionālā puse 1



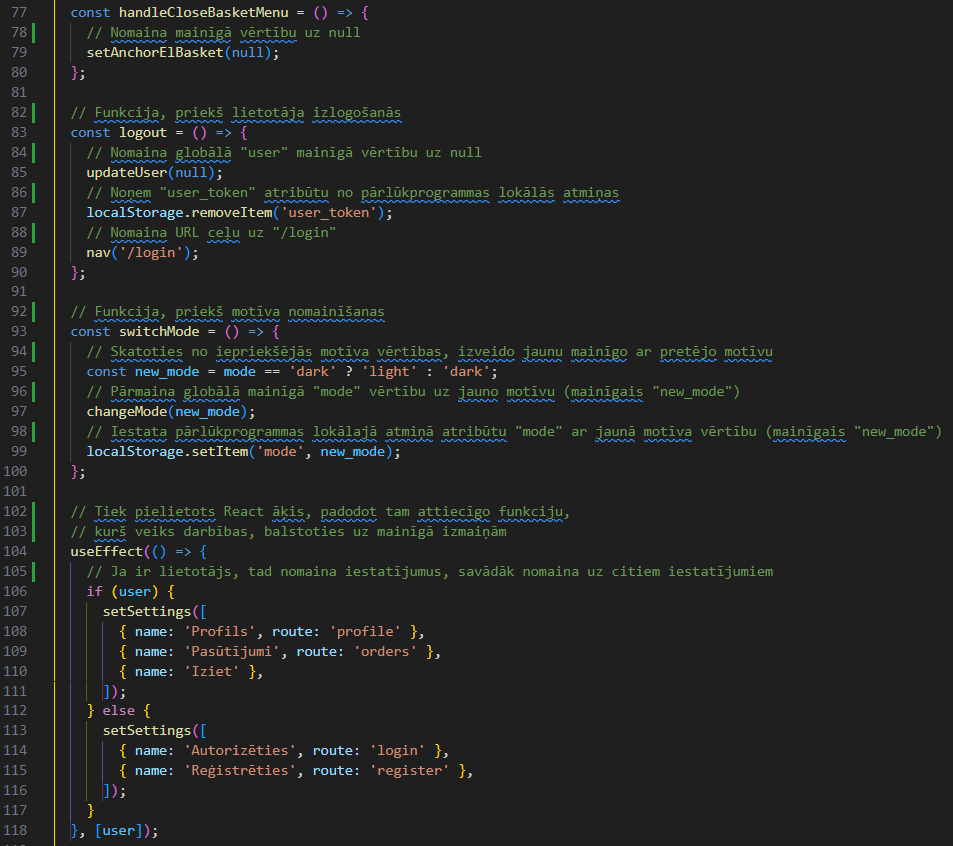
5.pielikums

Lietotāju puses navigācijas joslas funkcionālā puse 2



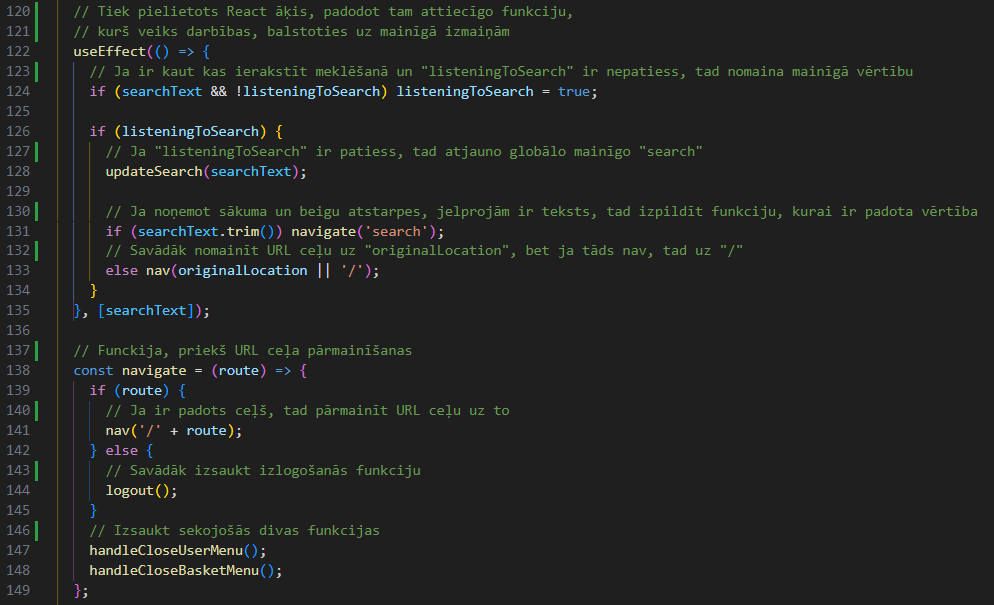
6.pielikums

Lietotāju puses navigācijas joslas funkcionālā puse 3



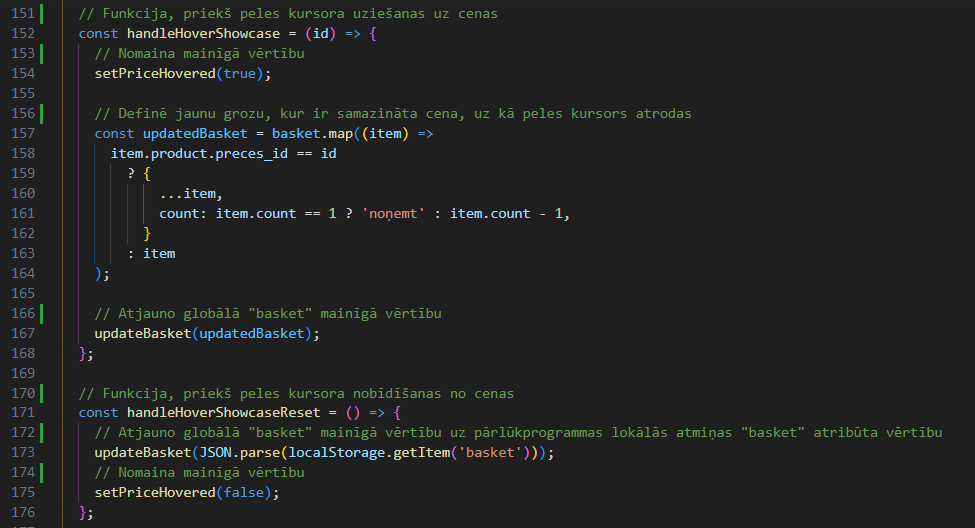
7.pielikums

Lietotāju puses navigācijas joslas funkcionālā puse 4



8.pielikums

Lietotāju puses navigācijas joslas funkcionālā puse 5



9.pielikums

Lietotāju puses navigācijas joslas funkcionālā puse 6

