**RĪGAS VALSTS TEHNIKUMS**

DATORIKAS NODAĻA

Izglītības programma: Programmēšana

**KVALIFIKĀCIJAS DARBS**

**“Tīmekļa lietotne preču pārdošanas sludinājumu portāla sistēma”**

Paskaidrojošais raksts \_\_\_ lpp.

Audzēknis: Iļja Fatkuļins

Prakses vadītājs: Ilona Demčenko

Nodaļas vadītājs: Normunds Barbāns

**Rīga 2023**

**ANOTĀCIJA**

Kvalifikācijas darbā ir aprakstīta tīmekļa lietotne preču pārdošanas sludinājumu portāla sistēmas izstrāde. Sistēma dod iespēju lietotājiem apkopot un apstrādāt datus par preču pārdošanas sludinājumiem. Sistēma tika izstrādāta un programmēta, izmantojot Java programmēšanas valodu, izmantojot Spring Framework ietvaru, un PostgreSQL relāciju datubāzu vadības sistēmu.

Kvalifikācijas darbā ietver ievadu, uzdevumu nosākdīt, prasību specifikāciju, uzdevumu risināšanas līdzekļus, izvēles pamatojumu, programmatūras produkta modelēšanas un projektēšanas aprakstu, lietotāju ceļvežus, tabulu un diagrammu veidā vizualizētus datus un funkcionalitātes pārbaudes rezultātus. Sistēmas izstrādāšana sākās ar prasību analīzi un modelēšanu, pēc tam sekoja struktūras un datu plūsmas modeļa izstrāde, kā arī sistēmas arhitektūras un ER modeļa veidošana.

Kvalifikācijas darbs sastāv no \_ lappusēm, kurā ietilpst \_ attēli, 10 tabulas un pielikumi.

**ANNOTATION**

The qualification work describes the development of a web application for a product sales advertisement portal system. The system allows users to collect and process data on product sales advertisements. The system was developed and programmed using the Java programming language, utilizing the Spring Framework, and the PostgreSQL relational database management system.

The qualification work includes an introduction, task definition, requirement specification, problem-solving tools, choice justification, description of product modeling and design, user guides, visualized data in the form of tables and diagrams, and functionality test results. The development of the system began with requirements analysis and modeling, followed by the design of the structure and data flow model, as well as the creation of the system architecture and ER model.

The qualification work consists of \_ pages, including \_ figures, 10 tables, and appendices.

**SATURS**

[IEVADS 5](#_desx4xllndkb)

[1. UZDEVUMA NOSTĀDNE 6](#_30j0zll)

[2. PRASĪBU SPECIFIKĀCIJA 8](#_1t3h5sf)

[2.1. Ieejas un izejas informācijas apraksts 8](#_4d34og8)

[2.1.1. Ieejas informācijas apraksts 8](#_2s8eyo1)

[2.1.2. Izejas informācijas apraksts 8](#_17dp8vu)

[2.2. Funkcionālās prasības 9](#_3rdcrjn)

[2.3. Nefunkcionālās prasības 10](#_26in1rg)

[3. UZDEVUMA RISINĀŠANAS LĪDZEKĻU IZVĒLES PAMATOJUMS 12](#_4xf8h2reoleu)

[4. PROGRAMMATŪRAS PRODUKTA MODELĒŠANA UN PROJEKTĒŠANA 14](#_2lu02zm85ag6)

[4.1. Sistēmas struktūras modelis 14](#_vh5rayia89xp)

[4.1.1. Sistēmas arhitektūra 14](#_tl5mbbd6yrz8)

[4.1.2. Sistēmas ER modelis 15](#_qy4nxhgq7980)

[4.2. Funkcionālais sistēmas modelis 15](#_i7a10axtedf7)

[4.2.1. Datu plūsmu modelis 15](#_q38j6ut4e4qp)

[5. DATU STRUKTŪRU APRAKSTS 21](#_4jypi2ckdzhn)

[6. LIETOTĀJA CEĻVEDIS 25](#_qkyf3cju9e3l)

[6.1. Sistēmas prasības 25](#_d326hroiawmj)

[6.2. Sistēmas instalācija un palaišana 25](#_23km354ke0az)

[6.3. Programmas apraksts 26](#_ijspixh82ulw)

[6.4. Testa piemērs 26](#_p4tbwmnbvj3f)

[NOBEIGUMS 28](#_4jt8belbxgj1)

[INFORMĀCIJAS AVOTI 29](#_lnxbz9)

# IEVADS

Digitālās tehnoloģijas intensīvā izaugsme un interneta nozīmības pieaugums ir radījis neatsveramu iespēju dažādu pakalpojumu digitalizācijai, vienlaikus piedāvājot gandrīz neierobežotas iespējas uzņēmējdarbības attīstībai. Sludinājumu portāla izveide ir tieši laikā, jo modernā sabiedrība arvien vairāk tiecas pēc efektivitātes un ērtības - gan pircējiem, gan pārdevējiem. Šāda veida sistēmas izveide atbilst patērētāju mainīgajām prasībām un digitālo pakalpojumu pieprasījumam, kā arī nodrošina cilvēkiem jaunu veidu, kā viegli un ātri piekļūt vajadzīgai informācijai, precēm un pakalpojumiem.

Pašlaik, bez kvalitatīvas sludinājumu platformas, lietotājiem nākas sastapties ar dažādām grūtībām: informācijas izkliedētību, piekļuves sarežģītību un drošības riskiem. Sludinājumu portāls atbildēs uz šīm problēmām, nodrošinot centralizētu, vienkārši lietojamu un drošu platformu gan ikdienas lietotājiem, gan biznesa vajadzībām. Tas ne tikai uzlabos esošo situāciju, bet arī radīs jaunas iespējas, piemēram, mazāk populāru produktu un pakalpojumu tirgus segmentu attīstību.

Veicot tirgus izpēti, tika konstatēts, ka Latvijā šobrīd ir viens šāds strādājošs portāls. Sakarā ar to, ka tas ir vienīgais pie mums, tas tagad ir paaugstinājis cenas un ilgu laiku nav atjaunināts, šim portālam ir ļoti novecojis izskats un funkcionalitāte. Piedāvātais risinājums nodrošinās lietotājiem modernu un saprotamo funkcionalitāte, kur sludinajumi ir bezmaksas. Administratīvās funkcijas, kas piešķirtas sistēmas administratoriem, kalpos kā garantija portāla drošībai un kvalitatīvai satura pārvaldībai.

Izstrādātā platforma sniegs būtisku ieguldījumu ne tikai atsevišķu lietotāju ikdienas dzīvē, bet arī veicinās uzņēmējdarbības izaugsmi, jo biznesa klientiem tiks nodrošināta iespēja efektīvi sasniegt savu mērķauditoriju. Tādējādi šis sludinājumu portāls kļūs par tiltu starp produkta vai pakalpojuma sniedzējiem un to patērētājiem, radot vērtību abām pusēm.

Apkopojot, sludinājumu portāls ir radīts kā atbilde uz mūsdienu tirgus vajadzībām un tendencēm, apvienojot lietotājam draudzīgu saskarni, augstu drošības līmeni un plašas funkcionalitātes. Tāds inovatīvs piegājiens sludinājumu publicēšanai internetā ne tikai atvieglos ikdienas darījumu veikšanu, bet arī uzlabos kopējo digitālo ekonomiku un veicinās digitālo prasmju attīstību.

# UZDEVUMA NOSTĀDNE

Kvalifikācijas darba uzdevums ir izveidot **"Tīmekļa aplīkacija - preču pārdošanas sludinājumu tīmekļa portāla sistēma"**. Sistēmā nepieciešams realizēt iespēju efektīvi pārvaldīt sludinājumus, kā arī nodrošināt lietotājiem vienkāršu un intuitīvu veidu, kā izveidot un pārvaldīt savus sludinājumus.

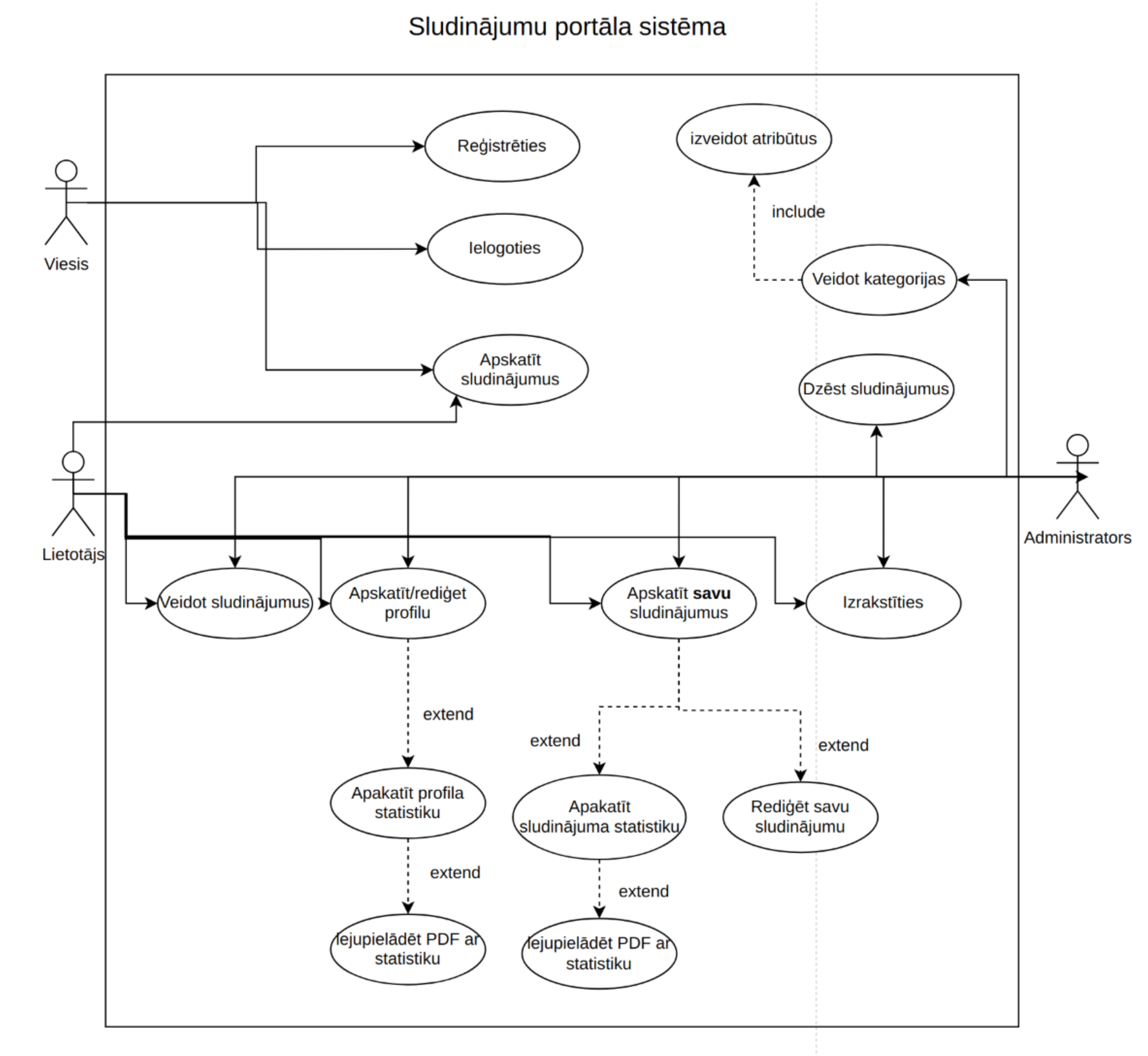
Pašreizējā tirgus situācijā novērojams pieprasījums pēc moderna sludinājumu portāla, kas piedāvā augstākas personalizācijas un lietotāja pieredzes iespējas nekā pašreiz pieejamie risinājumi, ka arī sludinājumu veidošana būs bezmaksas. Šāda sistēma būs īpaši noderīga privātpersonām un uzņēmumiem, kas meklē efektīvu un mērķtiecīgu veidu, kā sasniegt potenciālos klientus.

Ir plānotas vairākas funkcijas:

* Sludinājumu apskatīšana pēc kategorijām un meklēšanas kritērijiem.
* Sludinājumu veidošana ar pielāgojamām opcijām.
* Reģistrēšanās un lietotāja profila pārvaldība.
* Dažāda statistika
* Lietotāju lomu un piekļuves tiesību pārvaldība.
* Administrēšanas funkcijas portāla uzturēšanai.

Funkcijas ir grupētas pēc lietotāju lomām, kur, piemēram, viesi varēs apskatīt sludinājumus, bet reģistrēti lietotāji - izveidot un pārvaldīt tos, savukārt administratori varēs uzturēt sistēmas kārtību un drošību. Lietojumgadījuma diagrammu skat.1.att.

# 



1.att. Lietojumgadījuma diagramma

# 

# PRASĪBU SPECIFIKĀCIJA

## Ieejas un izejas informācijas apraksts

### **Ieejas informācijas apraksts**

Sistēmā tiks nodrošināta šādas ieejas informācijas apstrāde.

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

* Email – lietotajā e-pasts - burtu teksts ar izmēru no 7 līdz 100 rakstzīmēm (piem. “ilja@gmail.com”).
* Parole – lietotajā parolē - burtu teksts ar izmēru līdz 100 rakstzīmēm (piem. “1234”).

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

* Nosaukums – lomas nosaukums - burtu teksts ar izmēru līdz 100 rakstzīmēm (piem. “admin”).

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

* Nosaukums – sludinājuma nosaukums - burtu teksts ar izmēru līdz 100 rakstzīmēm (piem. “Macbook Air 2020”).
* Cena – pārdodamās preces cena - Skaitlis kas ir vairāk par 0 (piem. “800.50”).
* Apraksts – pārdodamās preces apraksts - burtu teksts ar izmēru līdz 10000 rakstzīmēm (piem. “Dators 2020 gada...”).
* Atributi - pārdodamās preces atribūts (preces īpašībā) - burtu teskt, atribūts ir pārdodamās preces parametrs vai īpašība (Ievadīta no klaviatūras) (piem. “Operatīva atmiņa: 16GB”).
* Attēlas - preces attēls - jpg/jpeg/png formāta faili (ielādēts no formas “Veidot jauno produktu”).

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

* Nosaukums - kategorijas nosaukums – burtu teksts ar izmēru līdz 100 rakstzīmēm (piem. “datori”).

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

* Nosaukums – atribūta nosaukums - burtu teksts ar izmēru līdz 100 rakstzīmēm (piem. “Motora apjoms”).

### **Izejas informācijas apraksts**

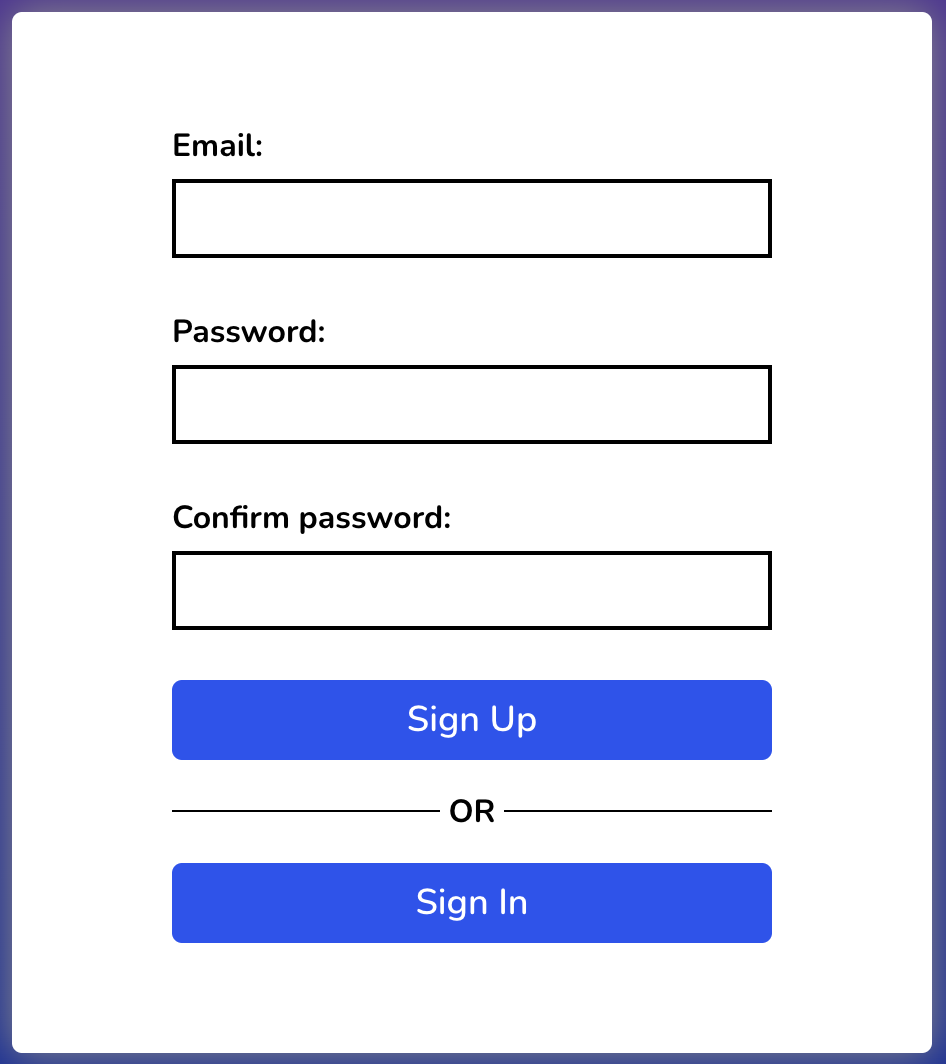
1. **E-pasta paziņojums** lietotājiem pēc reģistrācijas - paldies par reģistrāciju.
2. **E-pasta paziņojums** ar sešciparu kodu priekš parole atveseļošanās vai mainīšanai.
3. **PDF fails** ar lietotāja profila statistiku
4. **PDF fails** ar sludinājuma statistiku
5. **PDF fails** ar kategorijas/sadaļas statistiku

## Funkcionālās prasības

1. Jānodrošina iespēja reģistrēt jaunu lietotāju.
   1. Jāparedz ieejas informācijas par lietotāju (skat. 2.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.
   4. Ja lietotājs ir bloķēts, izvadīt par to paziņojumu.
3. Jānodrošina lietotājam veidot jaunus sludinājumus.
   1. Ja kāds no laukiem nav ievadīts, izvadīt par to paziņojumu.
   2. Ja cena ir mazāk par 0, izvadīt paziņojumu
4. Jānodrošina lietotājam veidot jaunus atribūtus un to nozīmes, laikā kad viņš veido jauno sludinājumu
   1. Ja kāds no laukiem nav ievadīts, izvadīt par to paziņojumu.
5. Jānodrošina lietotājam dzēst/rediģēt savus sludinājumus
6. Jānodrošina lietotājam pārskatīt kategorijas
7. Jānodrošina lietotājam pārskatīt sludinājumus pec kategorijas
   1. Ja kategorija neeksistē, izvadīt par to paziņojumu
8. Jānodrošina lietotājam pārskatīt detalizēto informāciju par sludinājumu
   1. Ja sludinājums neeksistē, izvadīt par to paziņojumu
9. Jānodrošina lietotajiem pārskatīt citus lietotājus profilus un viņa sludinājumus
   1. Ja lietotājs neeksistē, izvadīt par to paziņojumu
10. Jānodrošina lietotājiem apskatīt savu profilu un mainīt datus
11. Jānodrošina administratoram bloķēt lietotājus un sludinājums, ja tas pārkāp noteikumus
12. Jānodrošina administratoram mainīt lietotāja lomas
13. Jānodrošina administratoram veidot jaunas kategorijas
    1. Ja kāds no obligātiem laukiem nav ievadīts, tad izvadīt par to kļūdas paziņojumu.
    2. Ja kategorija eksistē, izvadīt par to paziņojumu

## 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/melna krāsā.



2.att. Sistēmas reģistrācijas formas skice

Šī skice attēlo sistēmas interfeisu, kurā lietotāji var veikt reģistrāciju. Lietotājam ir jāievada e-pastu, paroli, paroli otro reizi, vai nospiežt pogu “Sign In” un veikt autorizāciju, ja lietotājs jau ir reģistrēts.

# 

3.att. Sistēmas autorizācijas formas skice

Šī skice attēlo sistēmas interfeisu, kurā lietotāji var veikt autorizāciju. Lietotājam ir jāievada e-pastu un paroli, vai nospiežt pogu “Sign Up” un veikt reģistrāciju, ja lietotājs vel nav reģistrēts.

# UZDEVUMA RISINĀŠANAS LĪDZEKĻU IZVĒLES PAMATOJUMS

Aplīkācija sludinājumu portāla sistēma būs web aplikācija, tāpēc bija izveletie rīki priekš web aplikācijas izstradei.

**Java**

* Versija: 21
* Raksturojums: Java ir izplatīta, objektorientēta programmēšanas valoda, kas ir populāra lietotņu serveru izstrādē.
* Izmantošanas iemesls: Java tika izvēlēta sava stabilitātes, drošības un platformas neatkarības dēļ, kas ir būtiski servera puses lietotņu izstrādē, kā arī tās plašās ekosistēmas un atbalsta dēļ.

**Spring Framework**

* Versija: 6.0
* Raksturojums: Spring ir viegli paplašināms, moduļu Java EE ietvars.
* Izmantošanas iemesls: Spring tika izmantots, lai atvieglotu injicēšanas mehānismu, samazinātu koda atkarības un nodrošinātu elastību, izstrādājot biznesa loģikas slāni.

**Hibernate**

* Versija: 6.4
* Raksturojums: Hibernate ir ORM (Object-Relational Mapping) rīks Java vidē.
* Izmantošanas iemesls: Hibernate nodrošina vienkāršu un efektīvu veidu, kā pārvaldīt datubāzes operācijas, minimizējot nepieciešamību rakstīt atkārtotu SQL kodu un piedāvājot labu integrāciju ar Spring.

**Spring Data JPA**

* Versija: 3.2.0
* Raksturojums: Spring Data JPA ir Spring ietvara modulis.
* Izmantošanas iemesls: Tā izmantošana atvieglo datu piekļuvi, samazina nepieciešamību pēc boilerplate koda, un nodrošina vienkāršu integrāciju starp Java objektiem un datubāzi.

**PostgreSQL**

* Versija: 16
* Raksturojums: PostgreSQL ir spēcīga, atvērtā koda objektu-relationālā datu bāze.
* Izmantošanas iemesls: PostgreSQL tika izvēlēta tās augstās veiktspējas, atbalsta sarežģītām vaicājumu operācijām un lielu datu kopu apstrādes dēļ, kas ir būtiski jūsu projekta prasībām.

**HTML, CSS, JavaScript**

* Versija: HTML5, CSS3, ECMAScript 6+
* Raksturojums: Šīs tehnoloģijas ir standarts tīmekļa lietotņu front-end izstrādē, nodrošinot struktūru, stilu un interaktivitāti.
* Izmantošanas iemesls: Tās tika izvēlētas, lai nodrošinātu lietotāja saskarnes struktūru, stilu un interaktivitāti, kas ir nepieciešama dinamiskai un reaģējošai tīmekļa lietojumprogrammai.

**React**

* Versija: 18.2
* Raksturojums: React ir deklaratīva, efektīva un elastīga JavaScript bibliotēka lietotāja saskarnes izstrādei.
* Izmantošanas iemesls: React tika izvēlēts, lai izstrādātu interaktīvu un dinamisku lietotāja saskarni, izmantojot komponentu bāzētu pieeju, kas atvieglo uzturēšanu un paplašināšanu.

**Axios**

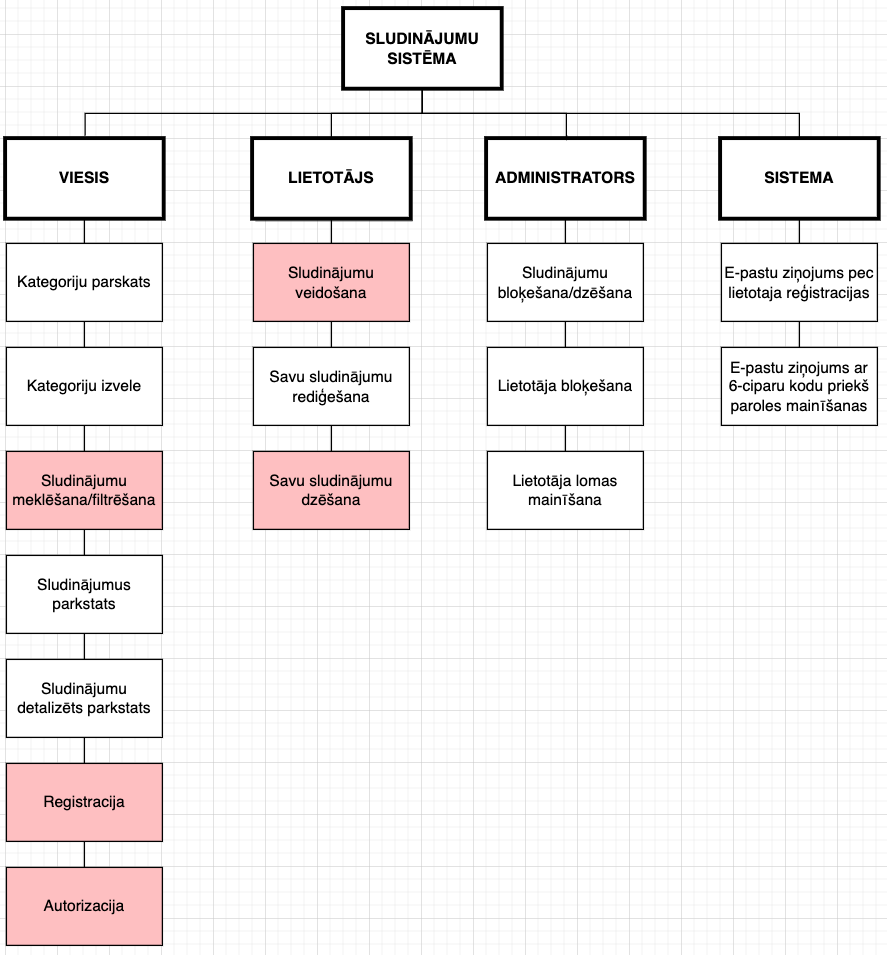
* Versija: 1.6.2
* Raksturojums: Axios ir solījumus atbalstošs HTTP klienta ietvars, kas izmantots datu ieguvei no servera.
* Izmantošanas iemesls: Tā izmantošana veicina efektīvu un vienkāršu asinhronu datu apmaiņu starp front-end un back-end, nodrošinot ērtu API izsaukumu veikšanu.

# PROGRAMMATŪRAS PRODUKTA MODELĒŠANA UN PROJEKTĒŠANA

## Sistēmas struktūras modelis

### **Sistēmas arhitektūra**

Sistēma ir sadalīta uz 4 daļam: Viesis, Lietotājs, Administrators un Sistēma. Katrā daļa izplidā savas darbības un atbild par savu jomu. Funkcionālās dekompozicijas diagramma ir 4.attēlā.

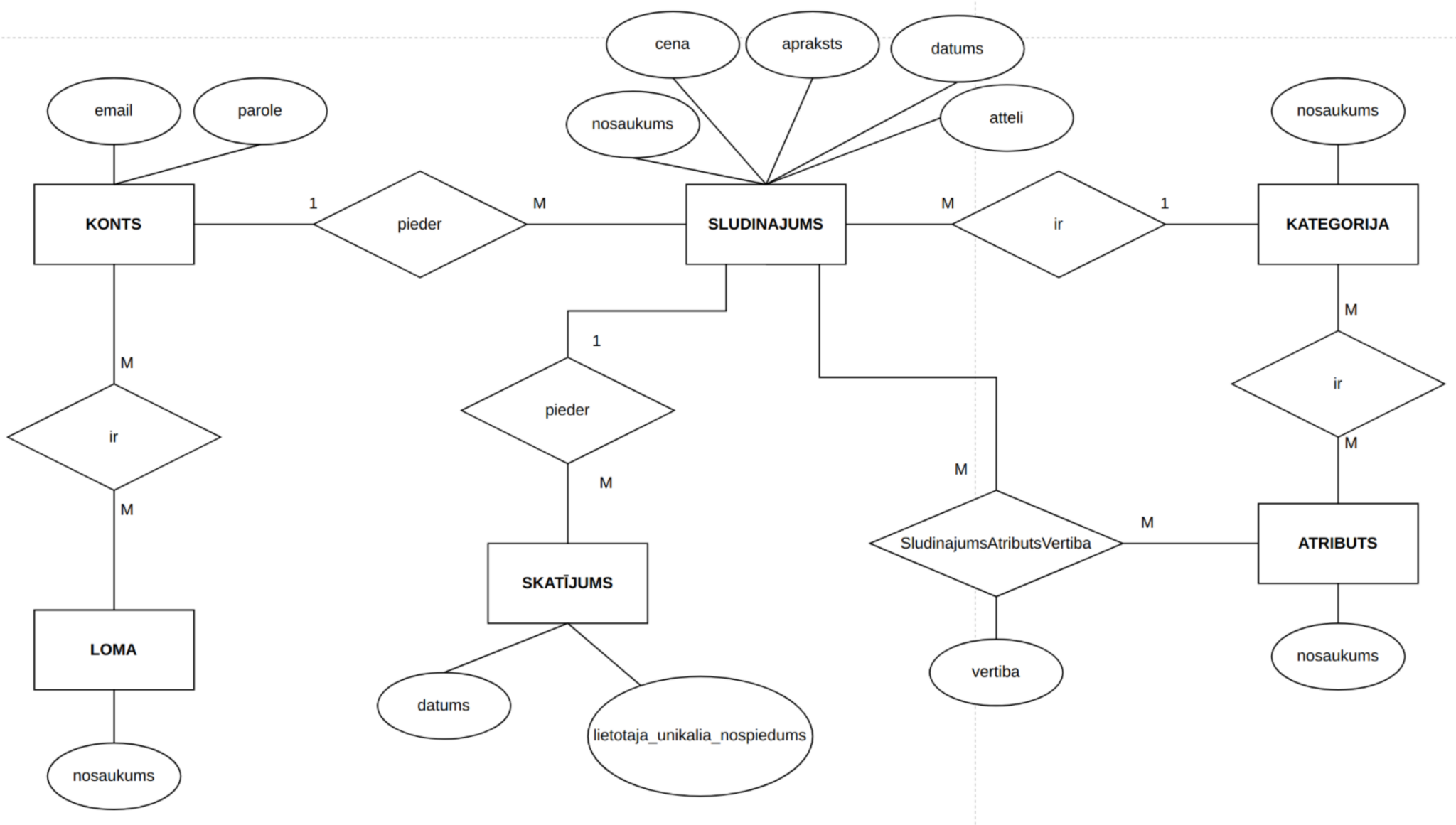


4.att. Funkcionālās dekompozīcijas diagrammu

### **Sistēmas ER modelis**

Sistēmas ER-modelis sastāv no 5 entitijām (skat. 5. att.), kas nodrošina pamat informācijas uzglabāšanu un apstrādi. Tie ir Loma, Konts, Sludīnājums, Kategorija, Atrībuts.

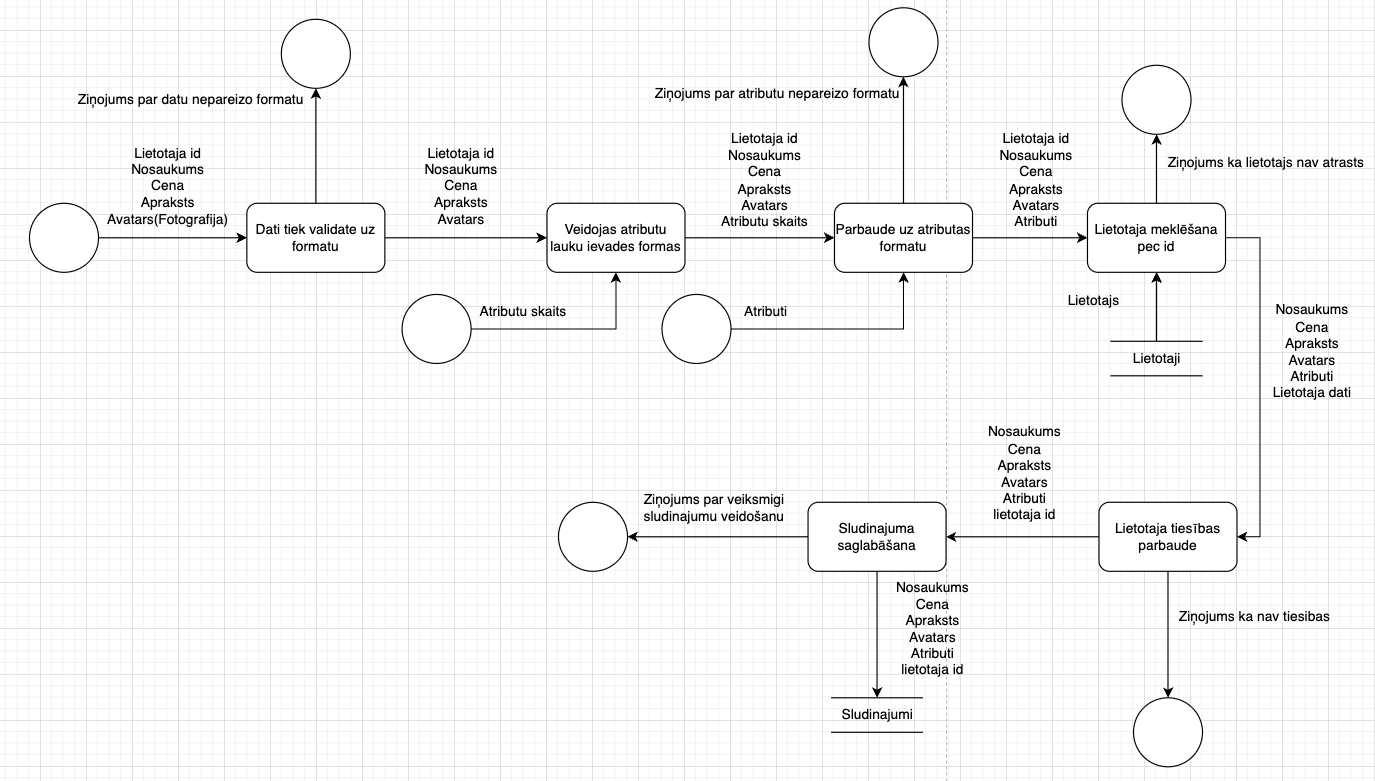
Entitijas Konts un Loma ir saišu “Many To Many”, ir 2 iemesli, kāpēc tas būtu jādara: Izstrades ir izmantots Frameworks – "Spring” un priekš drošibas ir izmantota “Spring Security” un deļ tas ir vieglāk un labāk izmantot tādu saišu, lai nākotnē nevajāg nekas mainīt. Otrs iemesls ir lai nav “Administratoram” iedot tādas pašas funkcijas kā “Lietotajs”, un kontam būtu divas lomas.

5.att. Sistēmas ER-diagramma

## Funkcionālais sistēmas modelis

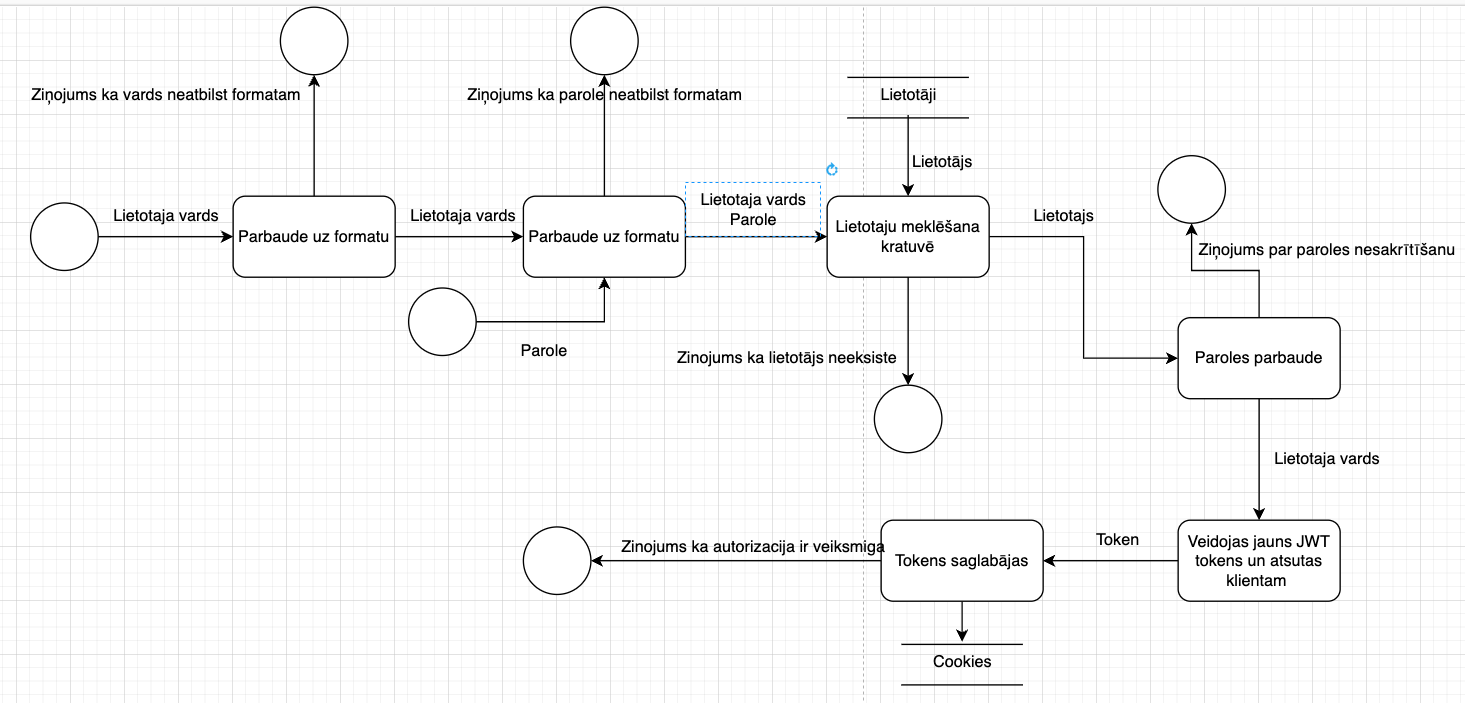
### **Datu plūsmu modelis**

“**Jauna sludinajuma veidošanas process**” diagramma (skat.6.att.). Lietotājs ievada datus: lietotaja id, sludinajumu nosaukums, cena, apraksts un avataru(fotografiju). Visi šie dati tiek parbaudīti uz formatu, ja formats neatbilst tiek izvadīts ziņojums. Lietotājs ievada atributu skaitu(cik preču paramētrus viņš grīb pievienot), pec skaita veidojas formas lauki. Lietotājs ievada visus atributus(Atributs piem. var būt “Motora tilpums” vai “Operatīvas atmiņas apjoms”), atributi tiek parbaudīti uz formātu, ja formats neatbilst tiek izvadīts ziņojums.. Notiek lietotāja datu meklēšana no datu bāzes pec ID, ja lietotājs nav atrasts tiek izvadits ziņojums ka lietotājs nav atrasts. Pēc tam parbauda lietotāja lomu, vai viņam ir tiesības veidot sludinājumus, ja ne tad tiek izvadīts ziņojums ka nav tiesibas. Sludinājumu dati tiek saglabāti datu bazē un tiek izvadīts ziņojums par veiksmīgo sludinājuma veidošanu.



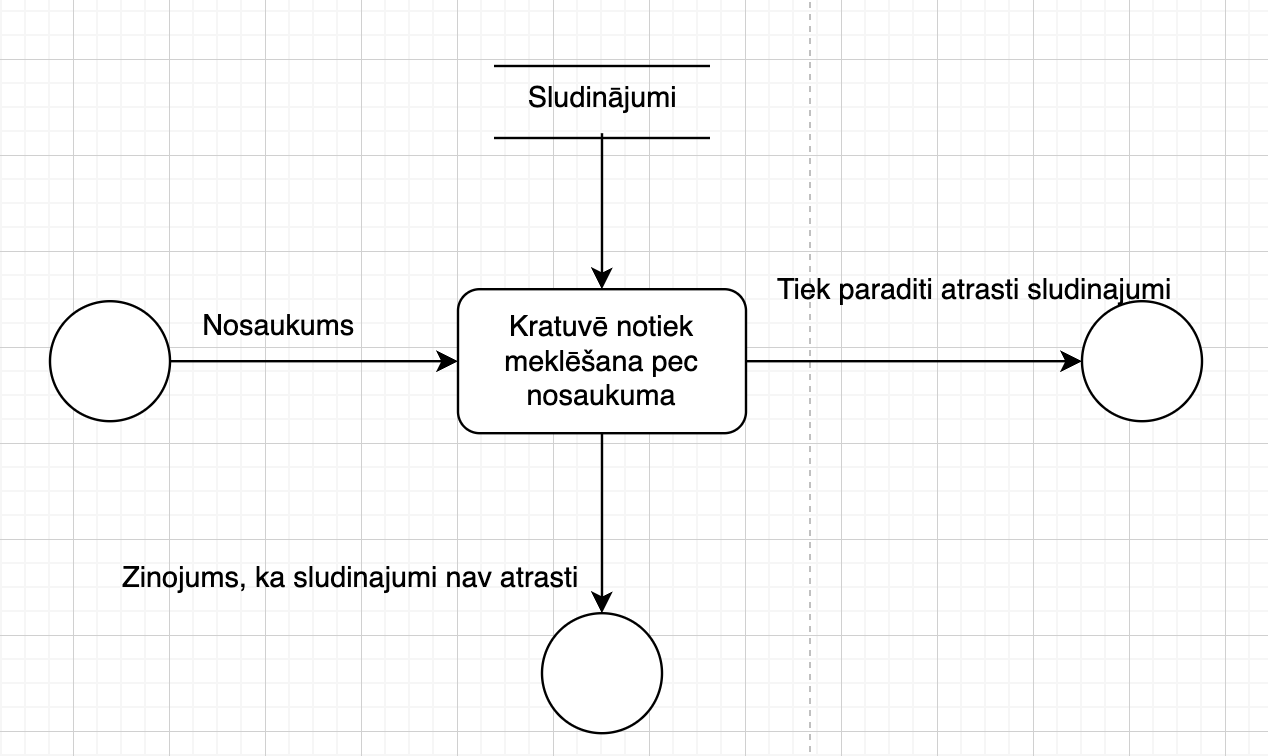
6.att. Sludinajuma veidošanas datu plūsmu diagramma

“**Autorizacijas process**” diagramma (skat. 7.att.). Lietotājs ievadata liet.vardu, tas tiek parbaudīts uz formatu, ja neatbilst tad izvadas ziņojums. Lietotājs ievada parole, tas tiek parbaudīts uz formatu, ja neatbilst tad izvadas ziņojums. Notiek lietotāju meklēšana datu bāzē pec liet.varda, ja lietotājs nav atrasts ir izvadīts ziņojums ka lietotājs neeksiste. Savieno lietotāja ievadīto parole un parole no datu bāzes, ja nesakrīt, tad ir izvadīts ziņojums ka paroles nesakrīt. Tiek veidot JWT tokens(unikalais šifrētas tokens priekš lietāja autorizacijas, tokena derīgums ir 1 stunda). Token saglābājas lietāja datorā kukī un ir izvadīt ziņojums par veiksmigo autorizāciju.



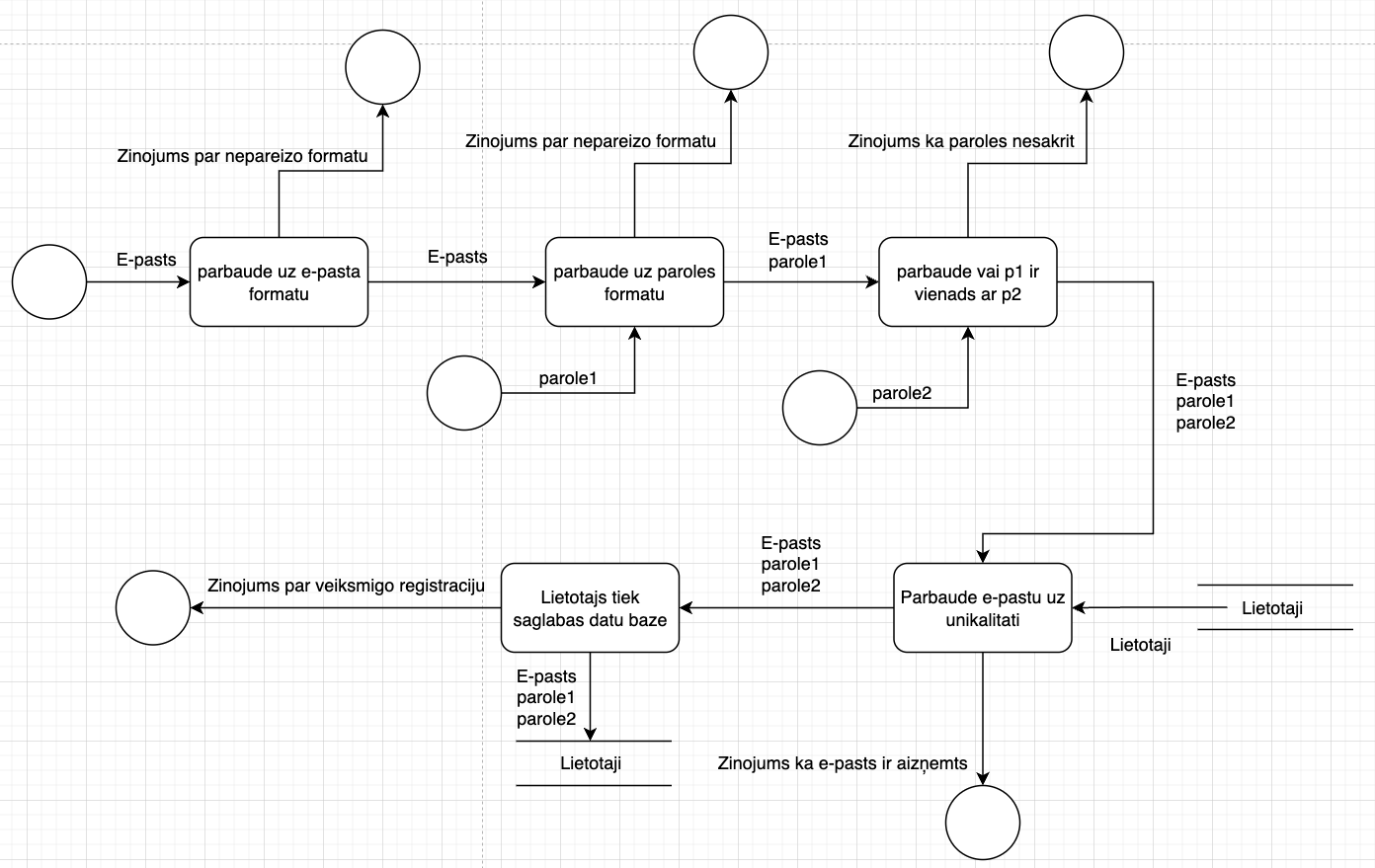
7.att. Autorizacijas datu plūsmu diagramma

“**Sludinājumu meklēšanas process**” diagramma (skat. 8.att). Lietotājs ievadata sludinājumu nosaukumu. Pec nosaukuma notiek meklēšana datu bāzē, ja nav atrasti tad ir izvadīts ziņojums ka nav atrasti. TIek paraditi atrasti sludinājumi.



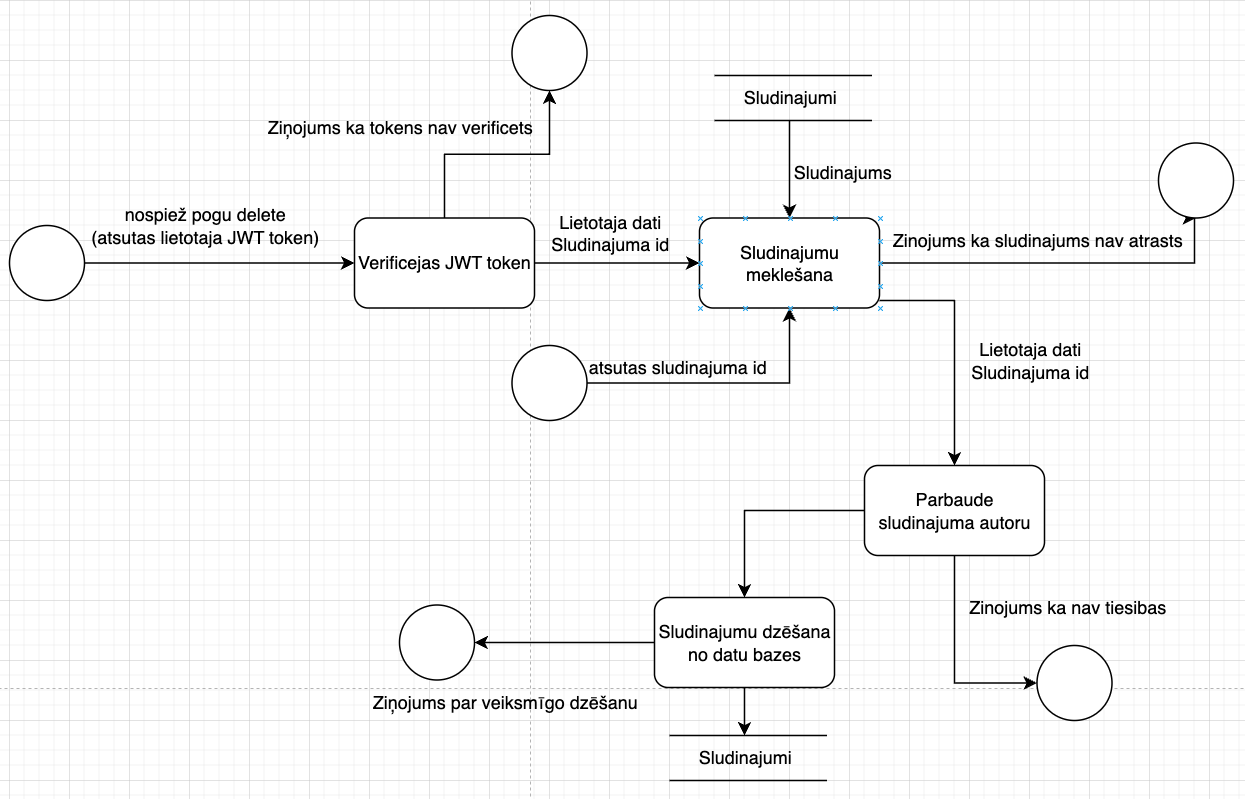
8.att. Meklēšanas datu plūsmu diagramma

“**Reģistrācijas process**” diagramma (skat. 9.att.). Lietotājs ievada e-pastu, tas tiek parbaudīts uz formatu, ja neatbilst ir izvadīts ziņojums par nepareizo formatu. Lietotājs ievada parole1, tas tiek parbaudīts uz formatu, ja neatbilst ir izvadīts ziņojums par nepareizo formatu. Lietotājs ievada parole2, savieno parole1 un parole2, ja tie nav vienadi, tad ir izvadīts ziņojums ka paroles nasakrīt. Notiek lietotāju meklēšana pec e-pasta, ja ir atrasts, tad ir izvadīts ziņojums ka lietotajs jau eksiste. Lietotaja dati tiek saglabāti datu bāzē.



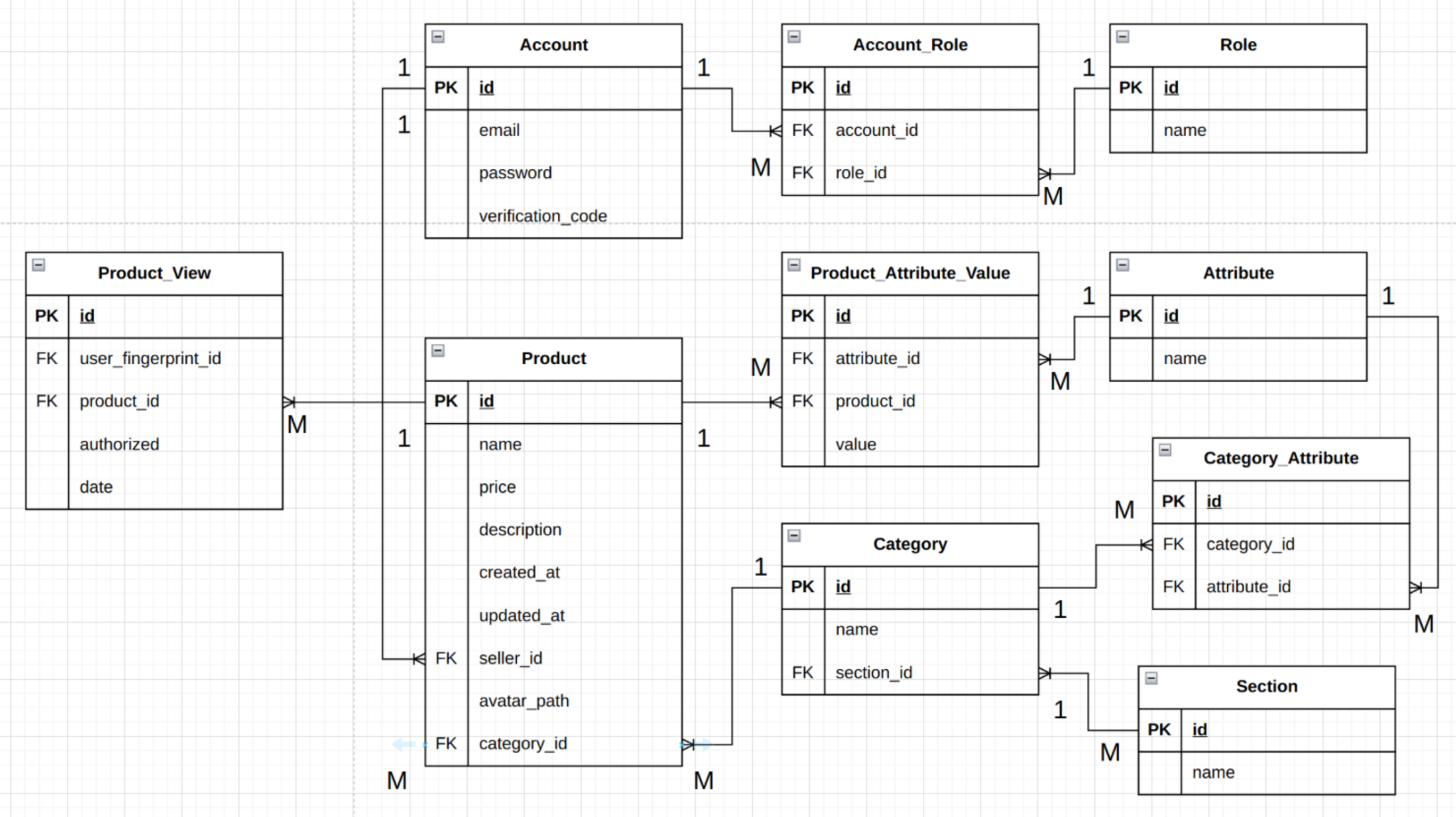
9.att. Registracijas datu plūsmu diagramma

“**Sludinajumu dzēšanas process**” diagramma (skat. 10.att.). Lietotājs nospiež pogu delete un atsutas lietotāja JWT tokens. JWT tokens verificejas, ja nav verificets tad ir izvadīts ziņojums. Atsutas sludinājumi id un notiek sludinājumu meklēšana datu bazē, ja nav atrasts tad ir izvadīts ziņojums ka sludinājums nav atrasts. JWT verificēšanas procesa ir ieguti lietotāja dati pec atšifrēšanas un notiek parbaude vai sludinājuma autors ir lietotājs, ja ne tad izvadas ziņojums ka nav tiesības. Notiek sludinājumu dzēšana no datu bāzes un izvadās ziņojums par veiksmīgo dzēšanu



10.att. Sludinajumu dzēšanas datu plūsmu diagramma

# DATU STRUKTŪRU APRAKSTS



11.att. Tabulu saišu shēma

Visi dati tiks saglabāti servera datu bāzē. Tajā tiks glabāti dati par lietotājiem, sludinājumiem, lomām un citi dati. Datu bāzē sastāv no 9 tabulām. Tabulu saišu shēma ir 11.attēla.

Tabulas datu bāzē:

* Tabula "**Account**” glabā lietotāja kontus.
* Tabula “**Role**” glabā datus par lomam.
* Tabula “**Account\_Role**” - tabula, kas realizē Many To Many attiecības starp tabulām Tabula "Account” un “Role”.
* Tabula “**Product**” glaba datus par sludinajumiem.
* Tabula “**Attribute**” - tabula ar datiem par attributiem priekš sludinajumiem, tas ir attributs/parametrs priekš sludinajumiem, piemēram mašinai atribūts “motora apjoms”.
* Tabula “**Product\_attribute\_value**” - tabula, kas realizē Many To Many attiecības starp tabulām “Product” un “Attribute” ar papildu lauku “value”.
* Tabula “**Category**” glabā datus par kategorijām.
* Tabula “**Section”** glabā datus par sekcijām.
* Tabula “**Category\_Attribute**” - tabula, kas realizē Many To Many attiecības starp tabulām “Category” un “Attribute”.
* Tabula “**Product\_View**” glaba informaciju par unikaliem produkta apsktatiem.

1.tabula

Tabulas **“Product”** sturktūra

| **Nr.** | **Nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| --- | --- | --- | --- | --- |
| 1. | id | int | 11 | Unikālāis kods |
| 2. | name | varchar | 100 | Sludinājuma nosaukums |
| 3. | price | float | 20 | Cena |
| 4. | description | varchar | 10000 | Sludinājuma apraksts |
| 5. | created\_at | date | - | Sludinājuma izveidošanas datums |
| 6. | updated\_at | date | - | Sludinājuma izmaiņas datums |
| 7. | seller\_id | int | 11 | Lietotāja unikālāis kods |
| 8. | avatar\_path | varchar | 100 | Ceļl kur serverā glabās sludinājuma avatars |

Tabula ”**Product**”, kas ir saistīta ar “Account” primāra atslēga un ar seller\_id.

2.tabula

Tabulas **“Attribute”** struktūra

| **Nr.** | **Nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| --- | --- | --- | --- | --- |
| 1. | id | int | 11 | Unikālāis kods |
| 2. | name | varchar | 100 | Atritūta nosaukums |

Tabula ”**Attribute**”

3.tabula

Tabulas **“Product\_Attribute\_Value”** struktūra

| **Nr.** | **Nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| --- | --- | --- | --- | --- |
| 1. | id | int | 11 | Unikālāis kods |
| 2. | attribute\_id | int | 11 | Atribūta unikālāis kods |
| 4. | product\_id | int | 11 | Sludinājuma unikālais kods |
| 5. | value | varchar | 100 | Atribūta vertība |

Saišu tabula “**Product\_Attribute\_Value**”, kas ir saistīta ar “Attribute” primāra atslēga un attribute\_id, un saistīta ar “roduct” primāra atslēga un product\_id.

4.tabula

Tabulas “**Section**” struktūra

| **Nr.** | **Nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| --- | --- | --- | --- | --- |
| 1. | id | int | 11 | Unikālāis kods |
| 2. | name | varchar | 100 | Sekcijās nosaukums |

Tabula “**Section**”

5.tabula

Tabulas “**Category**” struktūra

| **Nr.** | **Nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| --- | --- | --- | --- | --- |
| 1. | id | int | 11 | Unikālāis kods |
| 2. | name | varchar | 100 | Kategorijas nosaukums |
| 3. | section\_id | int | 11 | Sekcijās unikālāis kods |

Tabula “**Category**”, kas ir saistīta ar “Section” primāra atslēga un section\_id.

6.tabula

Tabulas “**Account**” struktūra

| **Nr.** | **Nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| --- | --- | --- | --- | --- |
| 1. | id | int | 11 | Unikālāis kods |
| 2. | email | varchar | 100 | Lietotāja e-pasts |
| 3. | password | varchar | 100 | Lietotāja parolē |
| f4. | verification\_code | varchar | 100 | Kods, kas izsūtas lietotājam uz e-pastu priekš parolēs mainīšanas |

Tabula “**Account**”

7.tabula

Tabulas “**Role**” struktūra

| **Nr.** | **Nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| --- | --- | --- | --- | --- |
| 1. | id | int | 11 | Unikālāis kods |
| 2. | name | varchar | 100 | Nosaukums |

Tabula “**Role**”

8.tabula

Tabulas “**Account\_Role**” struktūra

| **Nr.** | **Nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| --- | --- | --- | --- | --- |
| 1. | id | int | 11 | Unikālāis kods |
| 2. | account\_id | int | 11 | Lietotāja unikālāis kods |
| 3. | role\_id | int | 11 | Lomas unikālāis kods |

Tabula “**Account\_Role”,** kas saista saistīta ar “Account” primāra atslēga un account\_id, un saistīta ar “Role” primāra atslēga un role\_id.

9.tabula

Tabulas “**Category\_Attribute**” struktūra

| **Nr.** | **Nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| --- | --- | --- | --- | --- |
| 1. | id | int | 11 | Unikālāis kods |
| 2. | category\_id | int | 11 | Kategorijas unikālāis kods |
| 3. | attribute\_id | int | 11 | Atribūta unikālāis kods |

Tabula **“Category\_Attribute”,** kas saista saistīta ar “Category” primāra atslēga un category\_id, un saistīta ar “Attribute” primāra atslēga un attribute\_id.

10.tabula

Tabulas “**Product\_View**” struktūra

| **Nr.** | **Nosaukums** | **Tips** | **Izmērs** | **Apraksts** |
| --- | --- | --- | --- | --- |
| 1. | id | int | 11 | Unikālāis kods |
| 2. | authorized | bool | 1 | Vai lietotaj ir autorizēts |
| 3. | user\_fingerprint\_id | int | 11 | Lietotaja unikālāispirkstu nospiedums |
| 4. | product\_id | int | 11 | Produkta unikālāis kods |
| 5. | date | date | - | Parskāta datums |

Tabula “**Product\_View”,** kas saista saistīta ar “Product” primāra atslēga un product\_id.

# LIETOTĀJA CEĻVEDIS

### **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, kas spēj darbināt iepriekš 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.

* Prasības sistēmas instalācijai

Lai instalētu un palaistu sistēmu, nepieciešams:

1. Node.js:
   * Minimālā versija: 14.x
   * Instalēšanas instrukcija: https://nodejs.org/en/download/
2. Docker:
   * Minimālā versija: 20.x
   * Instalēšanas instrukcija: https://docs.docker.com/engine/install/

* Instalācijas un palaišanas soļi

1. Node.js instalācija:
   * Lejupielādējiet un instalējiet Node.js no Node.js oficiālās lapas.
   * Pārbaudiet instalāciju, ievadot terminālā: node -v un npm -v, lai pārliecinātos, ka Node.js un npm ir pareizi uzstādīti.
2. Docker instalācija:
   * Lejupielādējiet un instalējiet Docker no Docker oficiālās lapas.
   * Pārbaudiet instalāciju, ievadot terminālā: docker -v un docker-compose -v, lai pārliecinātos, ka Docker un Docker Compose ir pareizi uzstādīti.
3. Sistēmas avota koda iegūšana:
   * Klonējiet repozitoriju no GitHub:
   * git clone <https://github.com/IljaFatkulin/AdvertisementPortal/>
   * cd AdvertisementPortal
4. Atkarību instalēšana:
   * Ievadiet repozitorija direktorijā un instalējiet nepieciešamās atkarības, izmantojot npm:
   * npm install
5. Docker konteineru palaišana:
   * Palaidiet Docker konteinerus, izmantojot Docker Compose:
   * docker-compose up
6. Sistēmas palaišana:
   * Sistēma būs pieejama pārlūkprogrammā, ievadot lokālo adresi: http://localhost:3000

* Piekļuves informācija

Piekļuves dati no visām sistēmā izmantojamām lomām:

* + Administrators:
    - Lietotājvārds: admin@gmail.com
    - Parole: 123
  + Lietotājs
    - Lietotājvārds: user@gmail.com
    - Parole: 123

### **Programmas apraksts**

### **Testa piemērs**

| **Nr.** | **Prasības numurs** | **Prasības nosaukums** | **Ievaddati/situācijas apraksts** | **Sagaidāmais rezultāts** | **Statuss** |
| --- | --- | --- | --- | --- | --- |
| 1. | 2.4 | Ja lietotājs ir bloķēts, izvadīt par to paziņojumu. | Pareizi dati: lietāja e-pasts, parole(Lietājs ir bloķets) | Autorizācija būs neveiksmīga un būs izvadīts paziņojusm - “Konts ir slēgts” | Pareizi |
| 2. | 1.1 | Jāparedz ieejas informācijas par lietotāju (skat. 2.att.) ievadīšana un pārbaude un formāta pareizību. | E-pasts: “ilja”  Parole1: “123”  Parole2: “123” | Reģistrācija būs neveiksmīga un būs izvadits paziņojusm - “Nepareizs e-pasta formāts” | Pareizi |
| 3. | E-pasts: “[ilja@gmail.com](mailto:ilja@gmail.com)”  Parole1: “123”  Pareol2: “1234” | Reģistrācija būs neveiksmīga un būs izvadits paziņojusm - “Paroles nesakrīt” | Pareizi |
| 4. | E-pasts: “[ilja@gmail.com](mailto:ilja@gmail.com)”  Parole1: “123”  Pareol2: “123” | Reģistrācija būs veiksmīga un lietājs un automatisku autorizēts ar jauno lietotāja datiem un novirzīts uz kategorijas lapu | Pareizi |
| 5. | 2.1 | Ja lietotāja eksiste un parole ir pareiza aktīvs sistēmai ir jānodrošina autorizācija, pieslēdzoties ar e-pastu un paroli. | E-pasts: “[ilja@gmail.com](mailto:ilja@gmail.com)”  Parole: “123” | Autorizācija būs veiksmīga un lietotājs būs novirzīts uz kategorijas lapu | Pareizi |
| 6. | E-pasts: “maksims@gmail.com”  Parole: “123”  (Lietotājs ar tadiem datiem neeksiste) | Autorizācija būs neveiksmīga un būs izvadīts paziņojusm - “Konts nav atrasts” | Pareizi |
| 7. | E-pasts: “[ilja@gmail.com](mailto:ilja@gmail.com)”  Parole: “1234”  (Pareiza parole ir “123”) | Autorizācija būs neveiksmīga un būs izvadīts paziņojusm - “Nederīga parole” | Pareizi |
| 8. | 9. | Jānodrošina lietotajiem pārskatīt citus lietotājus profilus un viņa sludinājumus | Links: “<http://localhost:3000/profile/ilja@gmail.com>” | Lietotājs redzēs lietotāja - [ilja@gmail.com](mailto:ilja@gmail.com) profilu un viņa sludinājumus | Pareizi |
| 9. | 9.1. | Ja lietotājs neeksistē, izvadīt par to paziņojumu | Links: “[http://localhost:3000/profile/123oais@gmail.com](http://localhost:3000/profile/ilja@gmail.com)” | Būs izvadīts tukšs profils un pazīņojums - “Sludinājumi nav atrasti” | Pareizi |

# 

# NOBEIGUMS

Šobrīd ir paveikti aptuveni 80 procenti no visas plānotās funkcionalitātes un uzrakstītas aptuveni vairāk nekā 5000 koda rindiņas. Šobrīd realizētās funkcijas: reģistrācija, autorizācija, mijiedarbība ar profilu, kategoriju un sludinājumu apskate, jaunu kategoriju veidošana, administrators var izveidot jaunus filtrus konkrētas kategorijas sludinājumu meklēšanai, jaunu sludinājumu un to atribūtu veidošanai, dažādi filtri un meklēšana.

Lai projekts tiktu pabeigts 100 procentos, nepieciešams pabeigt dizainu un pievienot dažas nerealizētas funkcijas

# INFORMĀCIJAS AVOTI

1. [ANG] Java dokumentācija - <https://docs.oracle.com/en/java/> (Resurss apskatīts 02.11.2023)
2. [ANG] Spring Framework dokumentācija - <https://docs.spring.io/spring-framework/reference/index.html> (Resurss apskatīts 01.11.2023)
3. [ANG] Javascript dokumentācija - <https://developer.mozilla.org/en-US/docs/Web/javascript> (Resurss apskatīts 04.11.2023)
4. [ANG] React Framework dokumentācija - <https://legacy.reactjs.org/docs/getting-started.html> (Resurss apskatīts 03.11.2023)
5. [ANG] PosgreSQL dokumentācija - <https://www.postgresql.org/docs/> (Resurss apskatīts 05.11.2023)
6. [ANG] Hibernate ORM dokemntācija - <https://hibernate.org/orm/documentation/6.4/> (Resurss apskatīts 01.11.2023)