

# **RĪGAS VALSTS TEHNIKUMS**

## **DATORIKAS NODAĻA**

Izglītības programma: Programmēšana

### **KVALIFIKĀCIJAS DARBS**

#### **“Programmetāju tiešsaistes komunikēšanas platforma”**

Paskaidrojošais raksts 40 lpp.

Audzēknis:

Mets Latuskins

Prakses vadītājs:

Igors Litvjakovs

Nodaļas vadītājs:

Normunds Barbāns

**Rīga 2023**

## **ANOTĀCIJA**

Šajā kvalifikācijas darbā tiek prezentēts un analizēts tīmeklī balstītas sistēmas izstrādes process, kura mērķis ir pilnveidot izglītības un komunikācijas procesus, īpašu uzmanību pievēršot “komunicēšanas platformai”, kas ir sistēmas pasūtītājs. sistēma ir izstrādāta, izmantojot php, javascript un html/css valodas un mysql datubāzi.

Darba struktūra ietver ievaddaļu, problēmas izklāstu, programmatūras risinājuma izvēles pamatojumu, modelēšanu un projektēšanu, datu struktūras aprakstu, lietotāja rokasgrāmatu, secinājumu, informācijas avotus un lietojumus.

Ievadā aprakstīta problēma un sistēmas atbilstība, un problēmas izklāsts sniedz īsu sistēmas galveno mērķu un funkcionalitātes aprakstu. Prasību specifikācija definē ievades un izvades informācijas aprakstus un iedala prasības funkcionālās un nefunkcionālās kategorijās.

Datu struktūras aprakstā ir detalizēti aprakstīti datu bāzes tabulu lauki un to attiecības. Lietotāja rokasgrāmatā ir ietverta informācija par sistēmas prasībām, instalēšana, programmas apraksts un testēšanas piemēri.

Kopumā šis darbs sniedz priekšstatu par tīmekļa sistēmas izstrādes procesu, koncentrējoties uz “Online Platformas” specifiskajām vajadzībām un prasībām, un ietver plašu informāciju par sistēmas izveidi un darbību, kā arī sniedz skaidra izpratne par izvēlētajām tehnoloģijām un to pielietojumu.

Kvalifikācijas darbs sastāv 40 lappusēm, 28 attēliem, 13 tabulām un 3 pielikumiem.

## ANNOTATION

This qualification thesis presents and analyzes the process of developing a web-based system, which aims to improve education and communication processes, paying special attention to the "communication platform", which is the customer of the system. the system is developed using php, javascript and html/css languages and mysql database.

The structure of the paper includes introduction, problem statement, rationale for software solution selection, modeling and design, data structure description, user manual, conclusion, information sources and applications.

The introduction describes the problem and the suitability of the system, and the problem statement provides a brief description of the main goals and functionality of the system. A requirements specification defines descriptions of input and output information and divides requirements into functional and non-functional categories.

The data structure description describes in detail the fields of the database tables and their relationships. The user guide contains information about system requirements, installation, program description and testing examples.

In general, this work provides an overview of the web system development process, focusing on the specific needs and requirements of the "Online Platform", and includes extensive information on the system's creation and operation, as well as a clear understanding of the selected technologies and their application. The qualification work consists of 40 pages, 28 images, 13 tables and 3 appendices.

# SATURS

## IEVADS

Šobrīd ārkārtīgi svarīgi ir izmantot modernās tehnoloģijas, lai uzlabotu izglītības procesus un uzlabotu darba efektivitāti. Arvien pieaugoša tehnoloģiskā attīstība, kas notiek katru dienu, sniedz iespējas uzlabot mācību pieredzi, padarot to pieejamāku visiem dalībniekiem – gan studentiem, gan pieredzējušiem programmētājiem.

Tāpēc mūsdienu IT sistēmās esošo problēmu risināšanai tiek veidota “Programmētāju komunikācijas platforma”.

Salīdzinot ar esošajiem konkurentiem, tādiem kā "stack-overflow" un "CodeForum", mūsu izstrādātā sistēma piedāvās būtiskus uzlabojumus un jaunas funkcijas, kas atbilst mūsdienu IT sfēras prasībām, tai skaitā iespēju viegli atrast sev vēlamo tēmu un tādējādi atrisināt problēmas. problēma ātrāk.

Biznesa analīze skaidri parāda, ka lietotāji (gan studenti, gan parastie programmētāji) gūs labumu no sistēmas, izmantojot daudzpusīgu un lietotājam draudzīgu rīku komplektu, kas uzlabos kodēšanas procesu, veicinot efektīvāku sadarbību un komunikāciju. Tādējādi piedāvāju ne tikai risinājumu, bet arī ieguvumu sabiedrībai kopumā - uzlabojumu strauji augošajā un ārkārtīgi pieprasītajā IT nozarē.

## • UZDEVUMA NOSTĀDNE

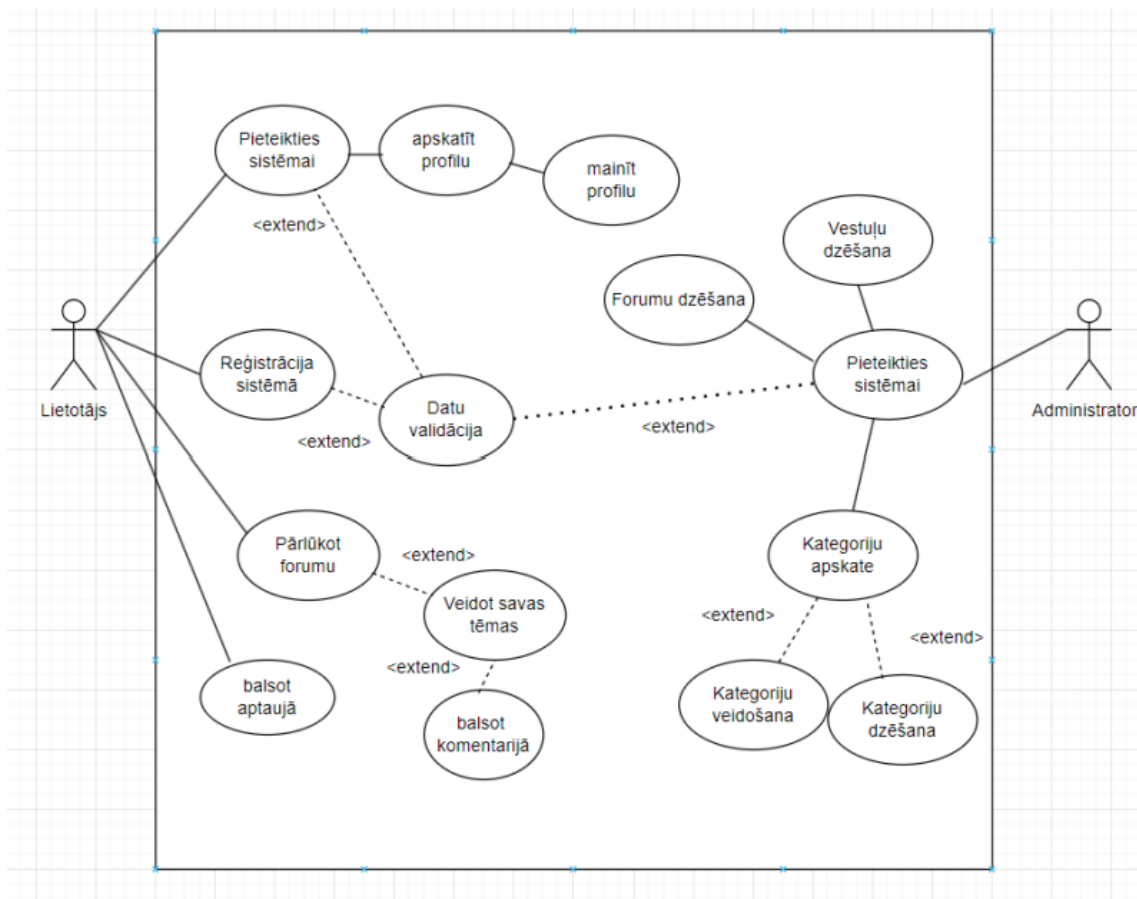
Kvalifikācijas darba uzdevums ir radīt "Tiešsaistes un komunikēšanas platformu", kas apvieno mūsdienīgas tehnoloģijas ar efektīvu problēmu risināšanu un komunikācijas procesu organizēšanu.

Platforma būs nepieciešama, lai nodrošinātu nepārtrauktu komunikēšanas procesu, kas būs aktuāls tik ilgi, kamēr pastāvēs mācību vajadzība. Tās galvenā mērķauditorija būs studenti un IT sfēras darbnieki.

Izglītības platforma iekļaus šādas jauninājumus un funkcijas:

- Reģistrācija un pierakstīšana lai nodrošinātu personalizētu piekļuvi uz vienti.
- Dažādas piekļuves tiesības, ko nodrošina administratori, atkarībā no lietotāja lomas un nepieciešamības.
- Failu attēlošanas sistēma serverī, kas ļauj apskatīt ielādētus attēlus.
- Forumā ar iespējām uzdot jautājumus, komentēt, kā arī administrācijai dzēst vai rediģēt lietotāju ziņojumus.
- Saturu kategorizēšana, lai lietotājiem būtu ērtāk atrast nepieciešamo informāciju.

Paredzēts, ka sistēmu lietos trīs lietotāju klases: Lietotājs(Lietotājs), ) un Administrators(skat.1.1.att).



1.1.att. Lietojumgadījuma diagramma

## • PRASĪBU SPECIFIKĀCIJA

### • Ieejas un izejas informācijas apraksts

#### • *Ieejas informācijas apraksts*

Sistēmā tiks nodrošināta šādas ieejas informācijas apstrāde.

• **Lietotāja dati** tiks ievadīti sistēmā, kad lietotājs reģistrēsies vai pieteiksies sistēmai. Šie dati ietver:

- Vārds –lietotāja vārds– burtu teksts ar izmēru līdz 20 rakstzīmēm. (Piem. “Mets”)
- E-pasts –lietotāja e-pasts– burtu teksts ar izmēru līdz 20 rakstzīmēm. (Piem. “Mets@gmail.com”)
- Lietotājvārds –lietotāja lietotājvārds– burtu teksts ar izmēru līdz 40 rakstzīmēm. (Piem. “MetsMets”)
- Parole –lietotāja parole– burtu teksts ar izmēru līdz 255 rakstzīmēm. (Piem. “ManaParole123”)

Lietotājvārds, vārds, e-pasts un parole tiks ievadīti no klaviatūras.

• **Jautājuma dati** tiks ievadīti sistēmā, kad lietotājs izveido jauno jautājumu. Šie dati ietver:

- Lietotājvārds – lietotāja lietotājvārds – burtu teksts ar izmēru līdz 40 rakstzīmēm. (Piem. “Mets”)
- Nosaukums – jautājuma nosaukums - burtu teksts ar izmēru līdz 50 rakstzīmēm. (Piem. “Kāpēc php un Javascript??”)
- Teksts – jautājumu teksts – burtu teksts ar izmēru līdz 200 rakstzīmēm. (Piem. “teksts”)

Nosaukums un teksts tiks ievadīti no klaviatūras. Lietotājvārds un izveides datums ievadīti automātiski, kad lietotājs iesniegs jautājumu

• **Komentāra pie jautājuma dati** tiks ievadīti sistēmā, kad lietotājs izveido jauno komentāru zem kādā jautājuma. Šie dati ietver:

- Lietotājvārds – lietotāja lietotājvārds – burtu teksts ar izmēru līdz 40 rakstzīmēm. (Piem. “Metss”)

- Teksts – jautājumu teksts – burtu teksts ar izmēru līdz 200 rakstzīmēm. (Piem. “komentāra teksts”)

Teksts tiks ievadīts no klaviatūras. Lietotājevārds ievadīts automātiski, kad lietotājs iesniegs komentāru.

- Nosaukums – tēmas nosaukums – burtu teksts ar izmēru līdz 50 rakstzīmēm. (Piem. “topic nosaukums”)
- Apraksts - tematas apraksts - burtu teksts ar izmēru līdz 255 rakstzīmēm. (Piem. “Java apraksts”)

Nosaukums, Apraksts un autors tiks ievadīts no klaviatūras. Lietotājevārds un Faila nosaukums ievadītas automātiski, kad lietotājs iesniegs failu.

- **Kategorijas dati** tiks ievadīti sistēmā, kad administrators pievieno jaunu kategoriju. Šie dati ietver:

- Nosaukums – kategorijas nosaukums – burtu teksts ar izmēru līdz 40 rakstzīmēm. (Piem. “Java”)
- Apraksts – kategorijas apraksts – burtu teksts ar izmēru līdz 255 rakstzīmēm. (Piem. “programmēšanas valoda”)

Teksts un novērtējums tiks ievadīti no klaviatūras.



- **Funkcionālās prasības**

1. Reģistrēšana un atļauja:

- 1.1. Nodrošiniet lietotājam iespēju reģistrēties, ievadot nepieciešamo informāciju, piemēram, vārdu, e-pastu, paroli u. c. 1.2.
- 1.2. Ja obligātie lauki nav aizpildīti, ģenerējiet kļūdas ziņojumu.
- 1.3. Pārbaudiet ievadītā lietotājvārda vai e-pasta unikalitāti starp jau esošajiem lietotājiem un atbilstības gadījumā izsniedziet attiecīgu paziņojumu.
- 1.4. Pēc veiksmīgas reģistrācijas saglabāt lietotāja datus datubāzē.

2. Lietotāja profils:

- 2.1. Nodrošināt iespēju autorizēties, izmantojot lietotājvārdu vai e-pastu kopā ar paroli.
- 2.2. Izvadiet kļūdas ziņojumu, ja autorizācijas laikā lauki ir tukši.

3. Forums:

- 3.1. Nodrošināt iespēju uzdot jautājumus, atstāt komentārus.
- 3.2. Sniegt lietotājiem tiesības veidot blogus un aptāujus.
- 3.3. Administrātoram nodrošināt iespēju veidot kategorijas, dzēst vestuļus.

4. Profils un rediģēšana:

- 4.1. Ļaujiet lietotājiem apskatīt un rediģēt savu profilu, tostarp mainīt paroli un atjaunināt informāciju.
- 4.2. Ļaujiet lietotājiem apskatīt citu lietotāju profilu.

5. Datu drošība:

- 5.1. Nodrošiniet lietotāju datus, tos šifrējot, lai novērstu nesankcionētu piekļuvi.
- 5.2. Nodrošiniet lietotājiem iespēju nepieciešamības gadījumā mainīt paroli.

6. Kategorijas:

- 6.1. Sniedziet lietotājiem iespēju apskatīt kategorijas.
- 6.2. Administrators var veidot jaunus kategorijas.

- **Nefunkcionālās prasības**

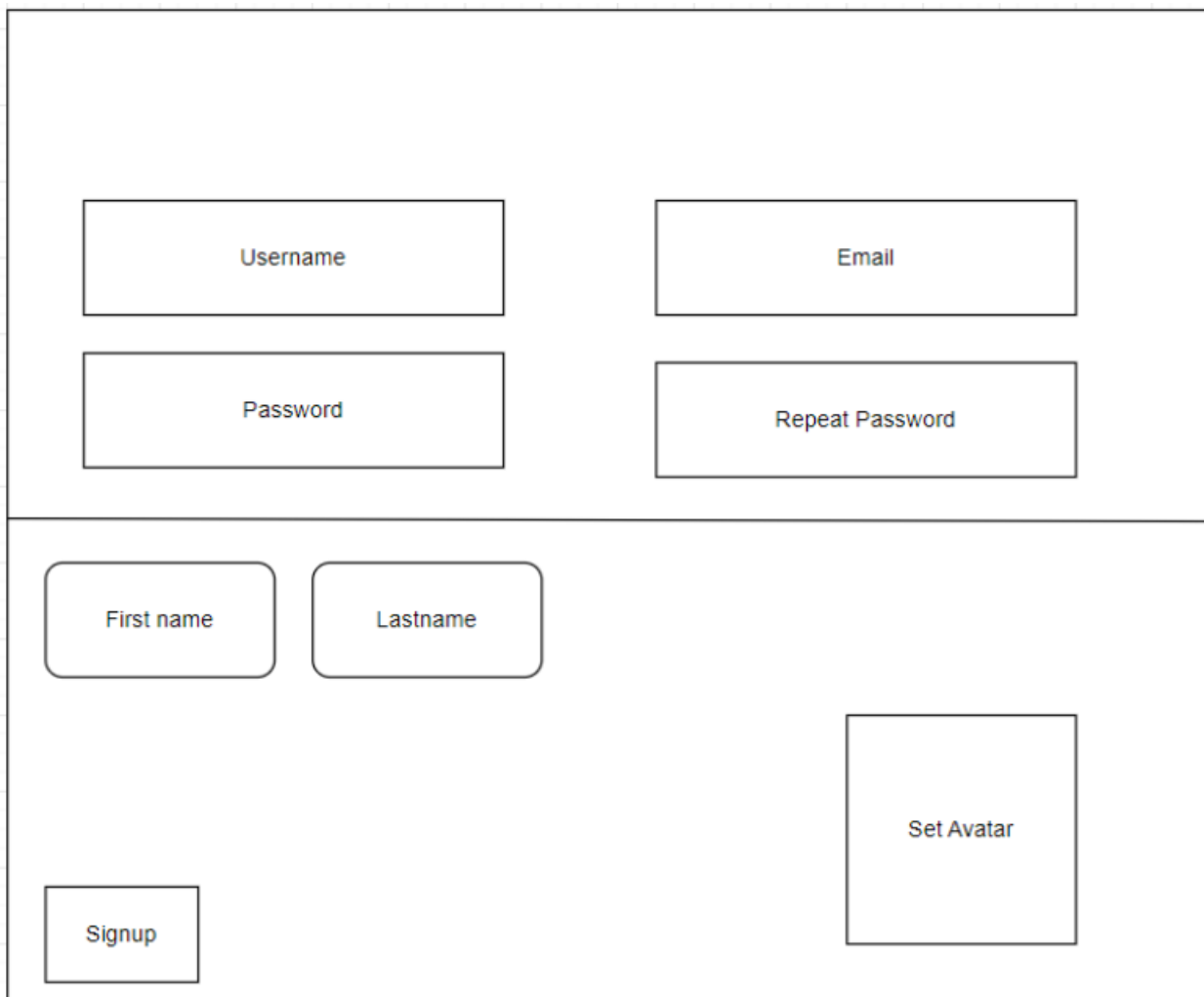
1. sistēmas saskanēs valodai jābūt pieejamai 1 valodā: angļu valodā.
2. tīmekļa lietojumprogrammai ir jāspēj pielāgoties mūsdienās izmantotajiem ekrāna izmēriem, lai to varētu izmantot uz dažāda izmēra monitoriem.
3. dizainam jābūt ieturētam organizācijas logotipa krāsās.
4. Sistēmai jānodrošina lietotāju datu drošība, izmantojot šifrēšanas tehnoloģijas.
5. Reģistrācijas procesam nevajadzētu aizņemt vairāk par 5 minūtēm (neskaitot administratora apstiprinājumu).
6. Sistēmas saskarei jābūt viegli saprotamai un lietojamai.
7. Sistēmas kļūdu paziņojumiem jābūt skaidriem un saprotamiem.
8. Sistēmas reakcijas laikam nevajadzētu pārsniegt 1 sekundi.
9. Sistēmai jābūt pieejamai 24 stundas diennaktī 7 dienas nedēļā.
10. Sistēmai jābūt noturīgai pret pārslodzēm un kļūdām.
11. Sistēmai jābūt atjauninātai un aprīkotai ar jaunākajām drošības funkcijām un uzlabojumiem.

Sistēmas ekrānu skices:

- Sistēmas reģistrācijas skice (skat. 2.3.1. att. un 2.3.2. att)

The diagram illustrates the layout of the registration form. It consists of four input fields arranged in a grid. The top row contains three fields: 'Username', 'Password', and 'Login'. The bottom row contains one field: 'Sign up'. All fields are rectangular with rounded corners and a light gray border. The text 'Username', 'Password', and 'Login' is centered within their respective fields. The text 'Sign up' is also centered within its field.

2.3.1.att. Sistēmas reģistrācijas pieprasījuma nosūtīšana



A registration form sketch divided into two horizontal sections. The top section contains four rectangular input fields arranged in a 2x2 grid: 'Username' and 'Email' in the top row, and 'Password' and 'Repeat Password' in the bottom row. The bottom section contains three rounded rectangular input fields: 'First name' and 'Lastname' side-by-side on the left, and a larger 'Set Avatar' field on the right. A 'Signup' button is located in the bottom-left corner of the bottom section.

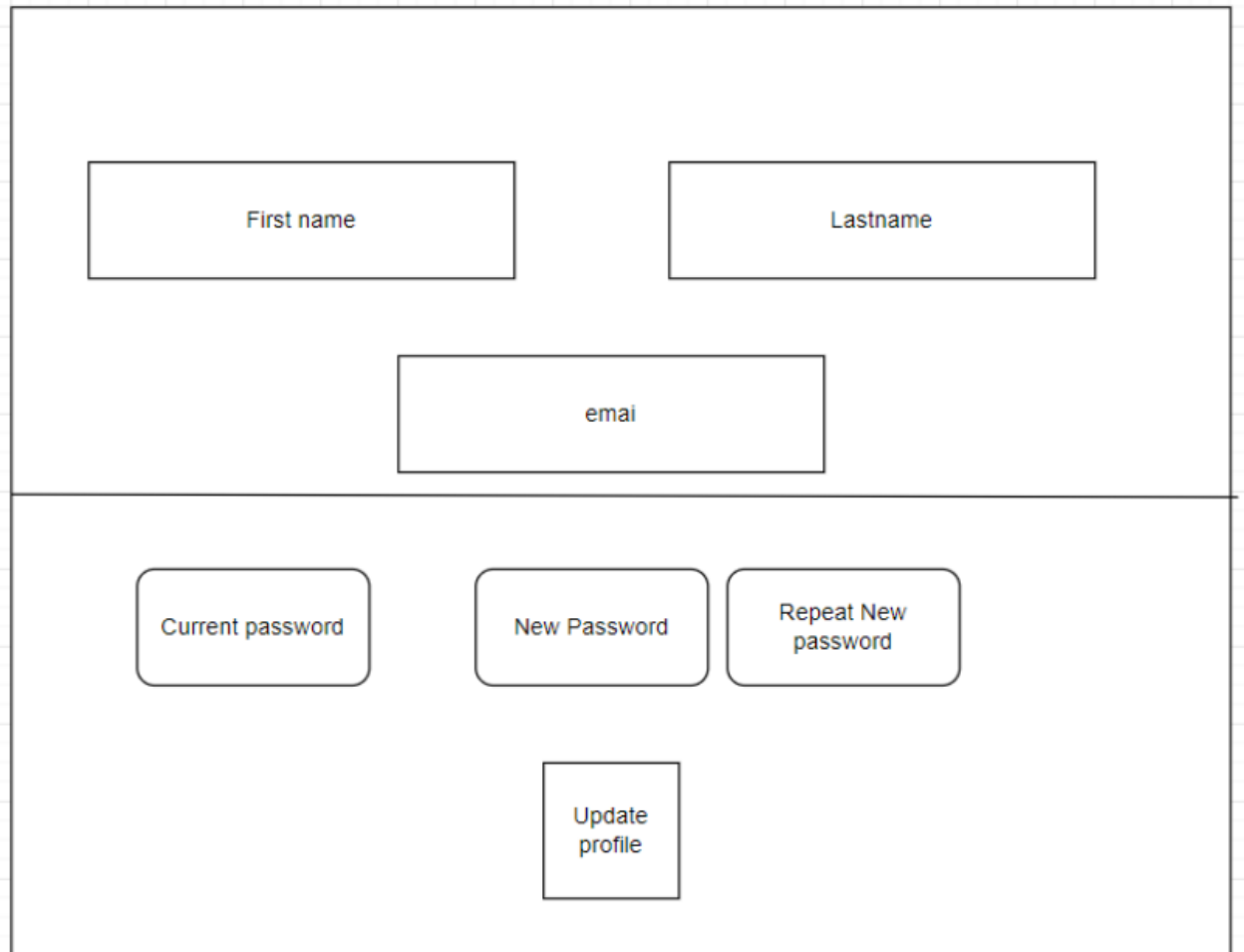
Username	Email
Password	Repeat Password

First name	Lastname	Set Avatar
Signup		

2.3.2.att. Sistēmas reģistrācijas pabeigšanas skice

Šī skice attēlo sistēmas interfeisu, kurā neregistrēti lietotāji var reģistrēties.



A user profile update form divided into two sections by a horizontal line. The top section contains three input fields: 'First name' and 'Lastname' side-by-side, and 'email' centered below them. The bottom section contains three rounded input fields: 'Current password', 'New Password', and 'Repeat New password' side-by-side, and an 'Update profile' button centered below them.

First name	Lastname
email	

---

Current password	New Password	Repeat New password
Update profile		

2.3.3att. Lietotāju porfīla datus mainīšana.

Šī skice attēlo sistēmas interfeisu, kurā tiek parādīti lietotāji dati un to iespēja mainīt(parole , vārds,uzvārds, pasts , u.t.t)

## • UZDEVUMA RISINĀŠANAS LĪDZEKĻU IZVĒLES PAMATOJUMS

Sistēma ir tiešsaistes platforma, kas radīta efektīvai darbībai pārlūkprogrammu vidē. Tā ir izveidota, sastāvot no divām pamata komponentēm: lietotāja interfeisa daļas (frontend) un servera puses (backend).

Izstrādē tika izmantotas šādas programmēšanas vides:

- **Visual Studio 2019** ir moderna izstrādes vide, kas piedāvā izstrādātājiem ērtu un produktīvu darba vidi. Tā ietver integrētu kodu redaktoru, atklūdošanas rīkus, projektu un failu pārvaldību, kā arī daudzas citās funkcijas, kas palīdz veiksmīgi īstenot programmatūras projektus.
- **WAMPP:** WAMPP ir izstrādes vide, kas piedāvā Apache serveri, MySQL datubāzi, PHP interpretatorus. Tas nodrošina pilnvērtīgu izstrādes vidi web aplikāciju veidošanai. Lietotāja daļa tiek veidota, izmantojot dažas valodas, kas aprakstītas turpmāk:
- **HTML (versijā 5.3):** HTML ir hiperteksta marķēšanas valoda, ko izmanto tīmekļa lapu front-end daļas projektēšanai, izmantojot iezīmēšanas valodu. Tās funkcionalitāte ideāli piemērota sistēmas prasībām, piedāvājot iespēju strukturēt un formatēt saturu.
- **CSS (versijā 3):** CSS ir vienkārša izstrādāta valoda, kas paredzēta, lai vienkāršotu tīmekļa lapu izskatu veidošanas procesu. Tā ļauj piemērot stilus tīmekļa lapām, piedāvājot iespēju pielāgot izskatu un izvairīties no sarežģītas stilu iestatīšanas. Tās lietošanas ērtums ir galvenais faktors izvēlē.
- **Bootstrap:** Bootstrap ir tīmekļa izstrādes rīks, kas nodrošina gatavus stila šablonus un komponentus, lai vienkāršotu tīmekļa lapu izveidi. Tas ļauj izstrādātājiem viegli pielāgot tīmekļa lapas izskatu, izmantojot iebūvētos CSS klases un komponentus. Bootstrap piedāvā plašu funkciju klāstu, kas padara tīmekļa lapu izstrādi efektīvāku un ērtāku, īpaši jaunajiem izstrādātājiem. Šīs platformas lietošana nodrošina vienkāršu un konsistentu dizainu, kas ir būtisks faktors izvēlē, īpaši tiem, kuriem nav plašas pieredzes tīmekļa izstrādē.
- **JavaScript (versijā 1.8.5):** JavaScript ir slavena skriptu valoda, ko izmanto, lai padarītu vietnes interaktīvas un pievienotu dinamiskumu. To plaši izmanto, lai uzlabotu vietņu funkcionalitāti un piedāvātu lietotājiem dinamiskas un responsīvas pieredzes.

Servera daļa tiek veidota, izmantojot dažas valodas un rīkus, kas aprakstītas turpmāk:

- **MySQL (versijā 8.0):** MySQL ir relāciju datubāzes pārvaldības sistēma, kas tiek izmantota datu glabāšanai un iegūšanai no citām programmatūras lietojumprogrammām. Tā ir izvēlēta kā datu bāzes risinājums, jo piedāvā efektīvu datu organizāciju, labu atbalstu transakcijām un ir labi pielāgojama Laravel lietojumprogrammu izstrādei.
- **phpMyAdmin:** phpMyAdmin ir PHP rakstīta programma datubāžu pārvaldībai, kas piedāvā grafisku lietotāja saskarni MySQL datubāzēm. Tas atvieglo datubāžu pārvaldību un manipulāciju ar datiem.

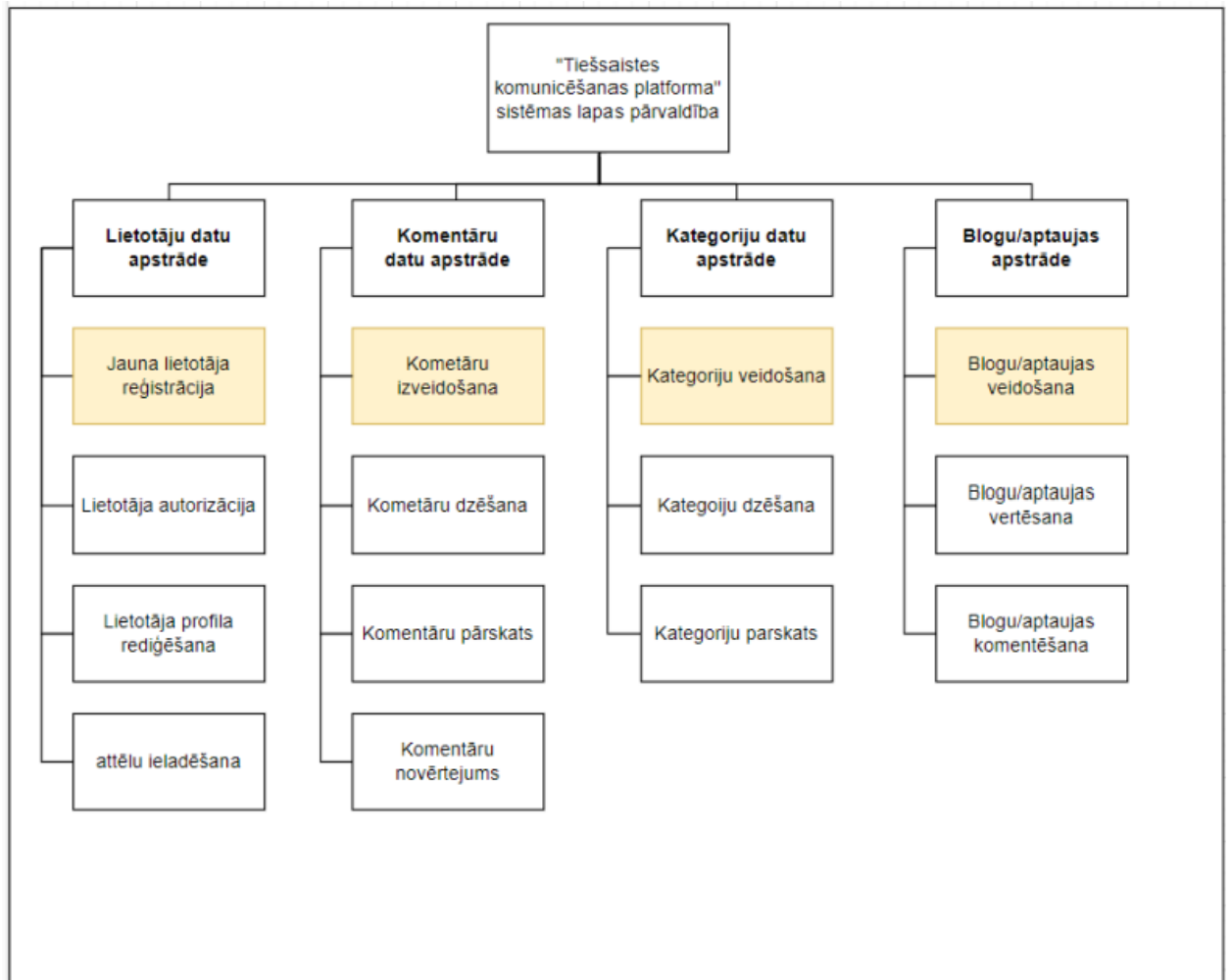
## • PROGRAMMATŪRAS PRODUKTA MODELĒŠANA UN PROJEKTĒŠANA

### • Sistēmas struktūras modelis

#### • *Sistēmas arhitektūra*

Sistēmai (skat. 3.1. att.) sastāvēs no četrām apakšsistēmām: lietotāju datu apstrādes,

komentāru datu apstrādes, kategoriju datu apstrādes un blogu/aptaujas datu apstrādes apakšsistēmām.



3.1. att. Sistēmas funkcionālā dekompozīcijas diagramma

- **Lietotāju datu apstrādes apakšsistēma** ir atbildīga par lietotāju reģistrāciju, autorizāciju, rediģēšanu un attēlu ielādēšanu. Šī apakšsistēma saglabā informāciju par lietotājiem, to datiem un citiem datiem.
- **Komentāru datu apstrādes apakšsistēma** ir atbildīga par lietotāju komentāru izveidošanu un novērtēšanu. Šī apakšsistēma saglabā informāciju par lietotāju komentāriem, to datiem un citiem datiem.

- **Blogu/aptaujū datu apstrādes apakšsistēma** ir atbildīga par blogu/aptaujū veidošanu un mainīšanu. Šī apakšsistēma saglabā informāciju par blogu/aptaujū, to datiem un citiem datiem.
- **Kategoriju datu apstrādes apakšsistēma** ir atbildīga par kategoriju izveidošanu un dzēšanu. Šī apakšsistēma saglabā informāciju par kategoriju, to datiem.

### *Sistēmas ER modelis*

Sistēmas ER-modelis sastāv no 4 entītijām (skat. 1. piel.), kas nodrošina pamat informācijas uzglabāšanu un apstrādi. Sistēmas ER-modeļa entītijas ir:

- **Lietotājs** – šī entītija reprezentē sistēmas lietotājus. Katram lietotājam ir unikāls lietotājvārds, e-pasts un parole. Lietotājs var būt gan administrators, , gan parasts lietotājs(lietotājs).
- **Bilde** – šī entītija parada ielādētu bildi, kas tiek lejup lādēti nolietotāja vai administratora puses.
- **Jautājums** – šī entītija reprezentē foruma jautājumus. Katram jautājumam ir unikāls reģistrācijas numurs, lietotāja vārds, nosaukums un teksts.
- **Jautājuma komentārs** – šī entītija reprezentē foruma komentārus. Katram komentāram ir unikāls reģistrācijas numurs, lietotāja vārds un teksts.
- **Kategorija** - šī entītija reprezentē kategorijas. Katram kategorijas ir nosaukums, apraksts un unikāls reģistrācijas numurs.
- **Jautājuma kategorija** - šī entītija reprezentē jautājuma kategorijas. Katram kategorijas ir nosaukums, apraksts un unikāls reģistrācijas numurs.
- **Lietotājs** ir saistīts ar **Jautājumiem**, jo viens lietotājs var veikt vairākas jautājumus.
- **Jautājums** ir saistīts ar **Komentāriem** un **Lietotājiem**, jo katrs komentārs ir saistīts ar konkrētu jautājumu un lietotāju.
- **Lietotājs** ir saistīts ar **Failiem**, jo viens lietotājs var lejup lādēt ne tikai bildi profilā, bet arī ka fonu aptaujā / blogā.

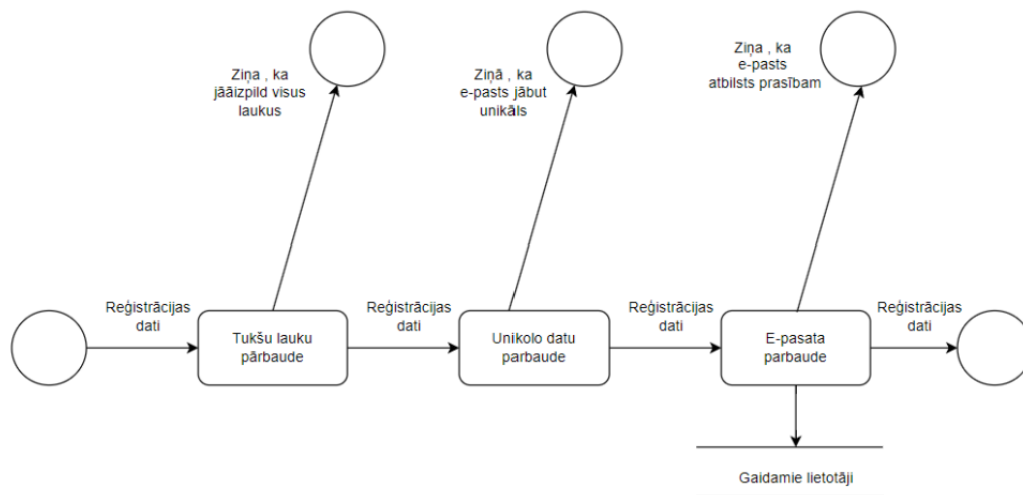


- **Funkcionālais sistēmas modelis**

- *Datu plūsmu modelis*

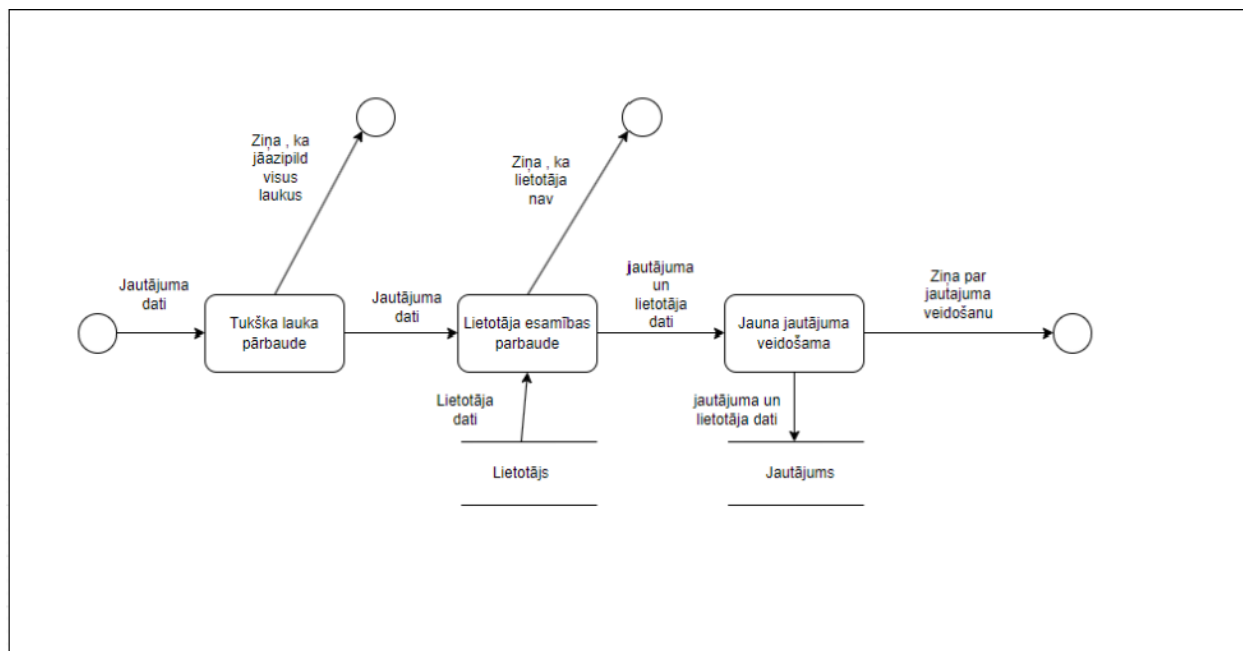
Datu apstrādes procesi:

- **Jauna lietotāja reģistrācija** – šīs operācijas procesā (skat. 3.2.1.) lietotājam ir jāievada dati, kuras, pēc veiksmīgu validāciju tiek ievadītas datubāzē. Kad reģistrācija ir pabeigta veiksmīgi lietotājs var ielogoties un aiziet vietnē.



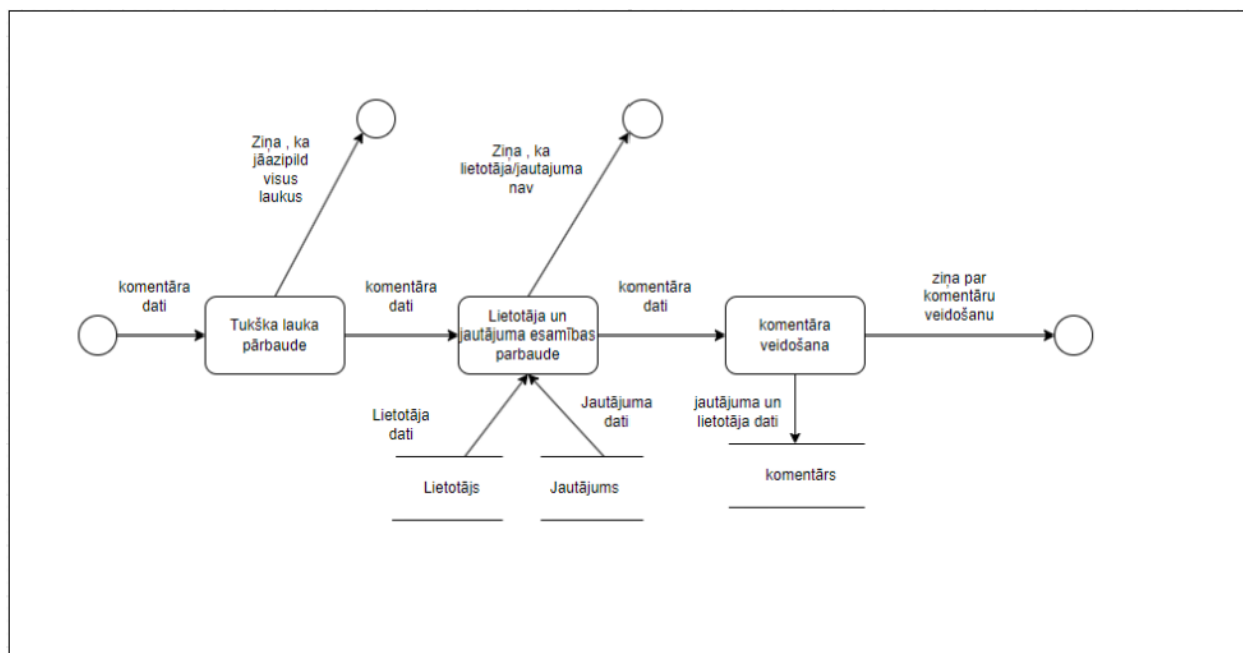
3.2.1. att. Jauna lietotāja reģistrācija plūsmu diagramma.

- **Jautājuma izveidošana** – šīs operācijas procesā (skat. 3.2.3 . att.) lietotājam ir jāievada dati, kuras, pēc veiksmīgu validāciju, tiek ievadītas datubāzē un tiks parādīts teksts ar veiksmīgu jautājuma veidošanu.



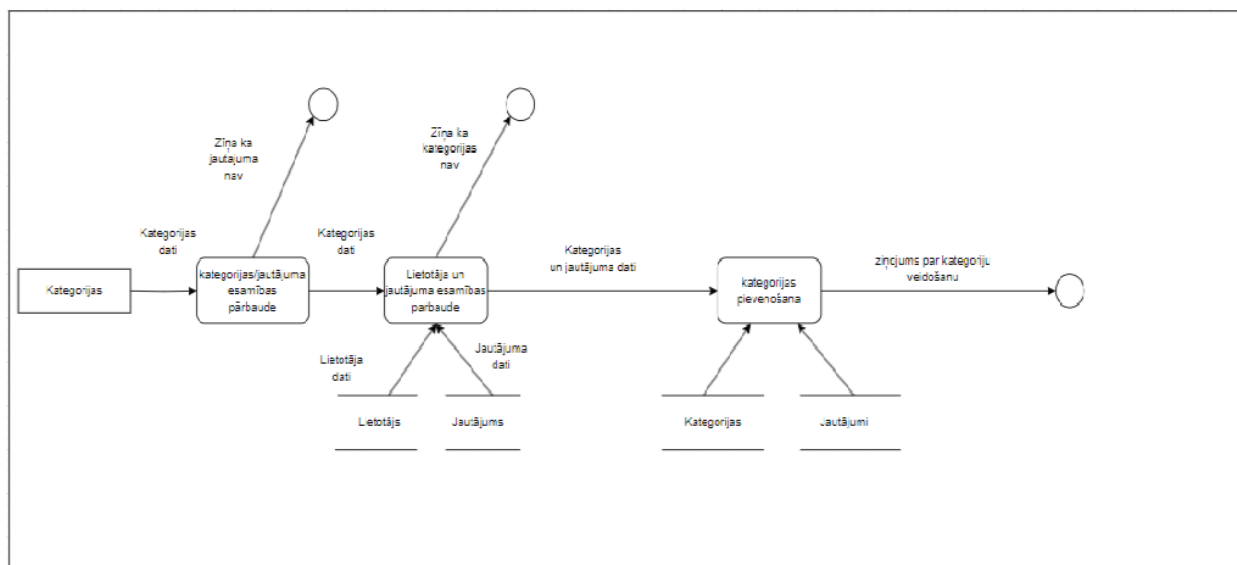
3.2.3.att. Jautājuma izveidošana plūsmu diagramma

- **Komentāru izveidošana** - šīs operācijas procesā (skat. 3.2.4. att.) lietotājam ir jāievada dati, kuras, pēc veiksmīgu validāciju, tiek ievadītas datubāzē un tiks parādīts teksts ar veiksmīgu komentāra veidošanu.



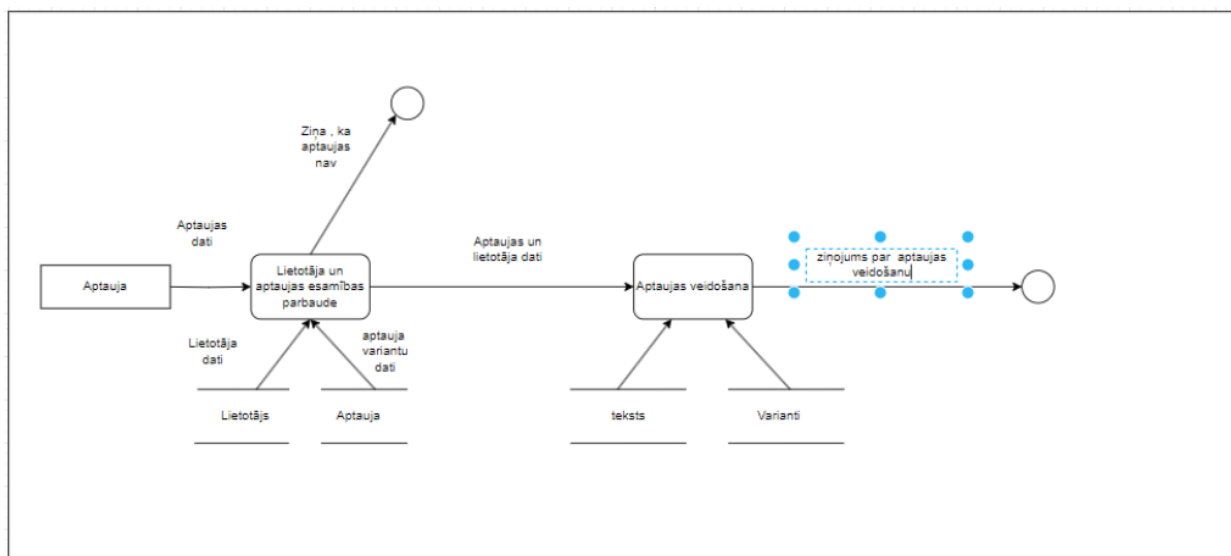
3.2.4.att. Jautājuma izveidošana plūsmu diagramma.

- **Kategoriju izveidošana** - šīs operācijas procesā (skat. 3.2.5. att.) administratoram ir jāievada dati, kuras, pēc veiksmīgu validāciju, tiek ievadītas datubāzē un tiks parādīts teksts ar veiksmīgu kategoriju izveidošanu.



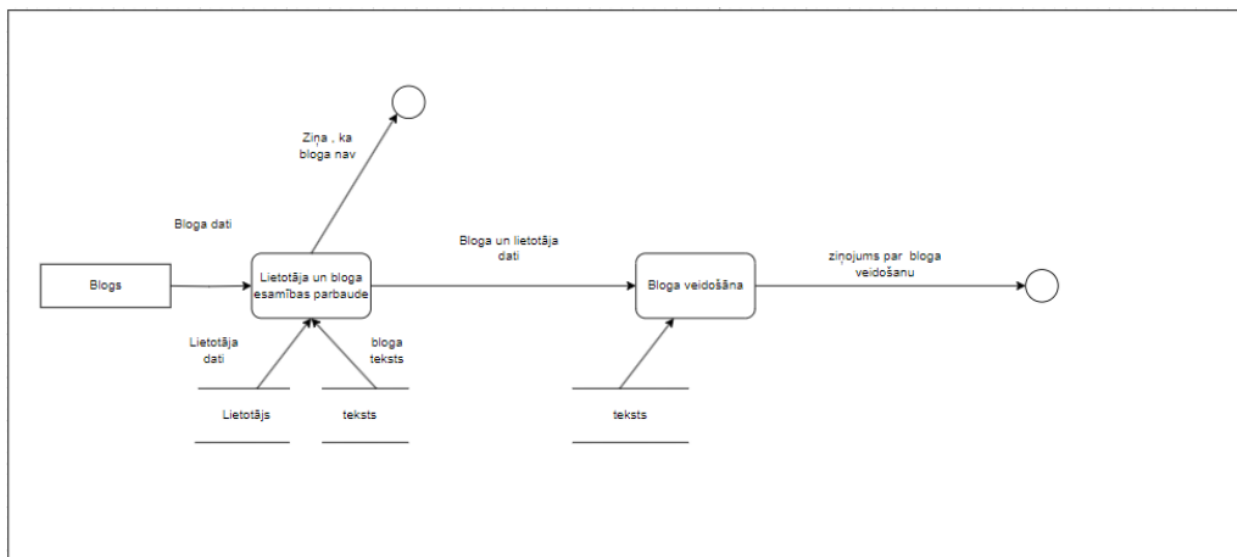
3.2.5.att. Kategorijas izveidošana plūsmu diagramma.

**Aptāuju veidošana** - šīs operācijas procesā (skat. 3.2.6. att.) lietotājam ir jāievada dati, kuras, pēc veiksmīgu validāciju, tiek ievadītas datubāzē un tiks parādīts teksts ar veiksmīgu aptāuju izveidošanu.



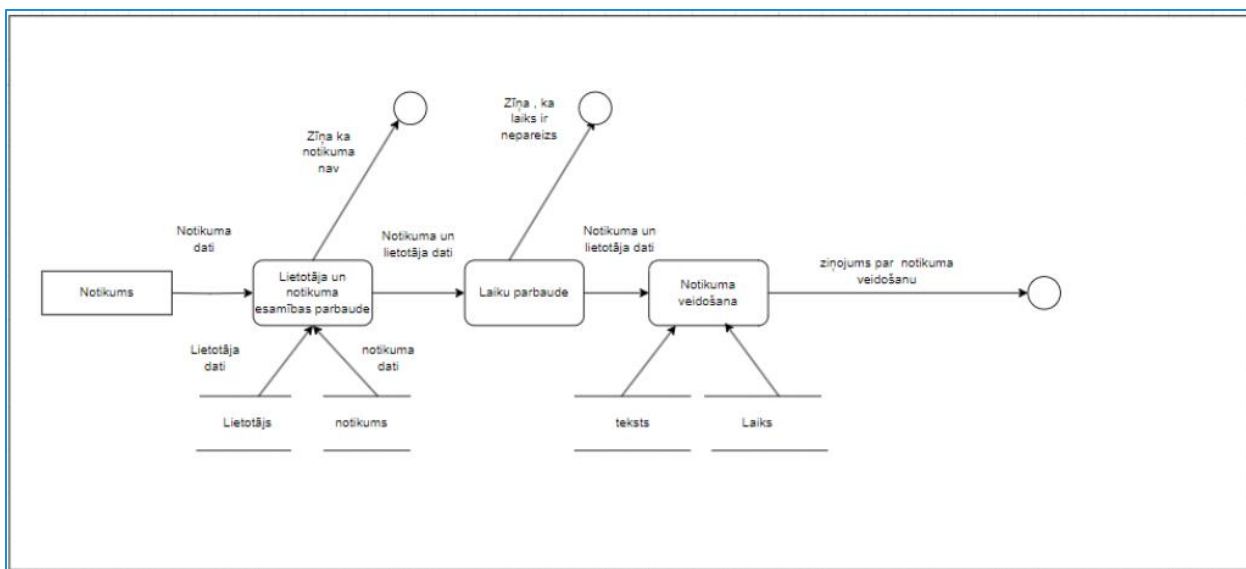
3.2.6.att. Aptaujas izveidošana plūsmu diagramma.

**Bloga veidošana** - šīs operācijas procesā (skat. 3.2.7. att.) lietotājam ir jāievada dati, kuras, pēc veiksmīgu validāciju, tiek ievadītas datubāzē un tiks parādīts teksts ar veiksmīgu aptāuju izveidošanu. (Blogs tik izveidots lai lietotājs var dalīties ar kadiem ziņiem bez atbildem).



3.2.7.att. Bloga izveidošana plūsmu diagramma.

**Notikuma veidošana** - šīs operācijas procesā (skat. 3.2.8. att.) lietotājam ir jāievada dati, kuras, pēc veiksmīgu validāciju, tiek ievadītas datubāzē un tiks parādīts teksts ar veiksmīgu aptauju izveidošanu. (Notikums izveidots lai parādītu kādu specifisku dati un notikumu kas tik atbilst tam datumam).



3.2.8.att. Notikuma izveidošana plūsmu diagramma.

# DATU STRUKTŪRU APRAKSTS

Veidojot datubāzi, liela uzmanība tika pievērsta tabulām, lai nodrošinātu efektīvas attiecības un vislabāko iespējamo datu glabāšanas vidi. Datu bāze sastāv no - tabulām, kas satur sevī informāciju par gaidāmiem lietotājiem, lietotājiem, , kategorijām, lietotāju jautājumiem un komentāriem(forums).

- Tabula **“users”** ietver 11 lauki un glabā datus par lietotājus.
- Tabula **Topics** ietver 5 laukus un glabā datus par izveidotam tēmatam.
- Tabule **Posts** ietver 6 lauki un glabā datus par pašiem jautājumiem iekš postā.
- Tabula **Post Votes** ietver 5 lauki un glabā datus par komentaru reitingu.
- Tabula **“Blogs”** ietver 6 lauki un glabā datus par blogiem.
- Tabula **“Blog votes”** ietver 5 lauki un glabā datus balsiem blogā.
- Tabula **“Events”** ietver 6 lauki un glabā datus par notikumiem.
- Tabula **“Event\_info”** ietver 5 lauki un glabā tekstu par notikumiem.
- Tabula **“Polls”** ietver 8 lauki un glabā datus par aptauju.
- Tabula **“Poll\_options”** ietver 5 lauki un glabā aptaujas konfigurāciju.
- Tabula **“Poll\_votes”** ietver 4 lauki un glabā datus kurš tik balso.
- Tabula **“Categories”** ietver 3 lauki un glabā datus par kategorijām un datiem.
- Tabula **“Pwdreset”** ietver 5 lauki un glabā datus par parola mainīšanu.

1. Tabula **“users”** satur informāciju par lietotājiem, kuri jau tiek izveidoti sistēmā. Tie tiek automātiski izveidoti, kad lietotājs tiek veiksmīgi reģistrējies. Tajā ietilpst dati, piemēram, lietotāja id, vārds, uzvārds, e-pasts, dzimusi, avatāru, lietotājvārds, parole, apraksts ,lauks ar pārbaidīšanu vai lietotājs ir administrators.

Tabulas “**users**” struktūra

Nr	Nosaukums	Tips	Izmērs	Apraksts
1.	IdUsers	int	11	Unikālais kods
2.	UserLevel	int	5	Vai lietotājs ir admins
3.	f_name	varchar	50	lietotāja vārds
4.	l_name	varchar	50	lietotāja uzvārds
5.	uidUsers	tinytext	20	Lietotāja segvārds
6.	emailUsers	tinytext	20	Lietotāja pasts
7.	pwdUsers	longtext	50	Lietotāja parole
8.	gender	char	1	Lietotāja dzimums
9.	Headline	varchar	500	apraksts zem profila
10	bio	varchar	4000	lietotāja apraksts
11	IserImg	varchar	500	lietotāja bilde

2. Tabula “**Topics**” satur informāciju par tematam kurie uzraksta lietotāji , parada , kad tik



izveidots un kurš lietotājs izveido. Tajā ietilpst dati , piemēram , datums , tema rūnāšanai , kategorija.

2.tabula

Tabulas “**Topics**” struktūra

Nr.	Nosaukums	Tips	Izmērs	Apraksts
1.	topic_id	int	11	Unikālais kods
2.	topic_subject	varchar	255	Vai lietotājs ir admins
3.	topic_date	datetime	10	lietotāja vārds
4.	topic_cat	int	255	lietotāja uzvārds
5.	topic_by	int	255	kurš lietotājs uzrakstīja

3. Tabula “**Posts**” satur informāciju par komentāriem kurie raksta lietotāji iekš topikā.

Tajā ietilpst dati, piemēram, komentarija balsi , komentara datums ,komentara teksts, autors.

3.tabula

Tabulas “**Posts**” struktūra

Nr.	Nosaukums	Tips	Izmērs	Apraksts
1.	post_id	int	11	Unikālais kods
2.	post_content	text	255	komentars
3.	post_date	datetime	10	kad lietotājs uzrakstīja
4.	post_topic	int	255	kura topikā ir komentars
5.	post_by	int	255	Lietotājs kurš uzraksta
6.	post_votes	int	-	komentarija reitings

4. Tabula “**Post\_votes**” satur informāciju par balsojumiem , kurie atbilst kads reitings būs

komentaram. Tajā ietilpst dati, piemēram, balsa datums , balsa autors.

4.tabula

Tabulas “**Post\_votes**” struktūra

Nr.	Nosaukums	Tips	Izmērs	Apraksts
1.	vote_id	int	11	Unikālais kods
2.	votePost	int	255	kurā komentārā bija balsojums
3.	voteBy	int	255	kurš lietotājs balso
4.	voteDate	date	10	datums balsojumam
5.	vote	int	255	+1/-1 balsojums

5. Tabula “**Blogs**” satur informāciju par blogu kuru veido lietotājs , un iekšdatus.

Tajā ietilpst dati, piemēram, bloga nosaukums , bilde , datums , balsu daudzums , iekšteksts.

Tabulas “**Blogs**” struktūra

Nr.	Nosaukums	Tips	Izmērs	Apraksts
1.	blog_id	int	11	Unikālais kods
2.	blog_title	varchar	100	bloga nosaukums
3.	blog_img	varchar	1000	bloga bilde
4.	blog_by	int	50	lietotājs kurš veido blogu
5.	blog_date	date	10	bloga datums
6.	blog_votes	int	255	bloga balsojumi
7.	blog_content	longtext	5000	teksts iekš blogā

6. Tabula “**Blog votes**” satur cik balsojumus ir blogam.

Tajā ietilpst dati, piemērām, balsa datums , kurš balso.

6.tabula

Tabulas “**Blog votes**” struktūra

Nr.	Nosaukums	Tips	Izmērs	Apraksts
1.	vote_id	int	11	Unikālais kods
2.	voteBlog	int	255	kurā bloga tik veikts balsojums
3.	voteBy	int	255	kurš lietotājs balsoja
4.	voteDate	date	10	balsojuma datums
5.	vote	int	255	balsojums

7. Tabula “**Events**” satur informāciju par notikumu , to iekšinformāciju un datumu , kad būtu notikums. Tajā ietilpst dati, piemēram, notikuma datums , nosaukums , bilde , autors.

7.tabula

Tabulas “**Events**” struktūra

Nr.	Nosaukums	Tips	Izmērs	Apraksts
1.	event_id	int	11	Unikālais kods
2.	event_by	int	50	notikuma

				autors
3.	title	varchar	1000	nosaukums
4.	date_created	date	10	kada laika izveidots
5.	event_date	varchar	10	notikuma datums
6.	event_image	varchar	255	notikuma bilde

8. Tabula “**Event\_info**” satur informāciju par notikumu , to iekšinformāciju un datumu , kad būtu notikums. Tajā ietilpst dati, piemēram, apraksts , nosaukums.

8.tabula

Tabulas “**Event\_info**” struktūra

Nr.	Nosaukums	Tips	Izmērs	Apraksts
1.	event_id	int	11	Unikālais kods
2.	event	int	50	notikuma id
3.	title	varchar	100	nosaukums
4.	headline	varchar	100	apgalvojum

				s
5.	description	varchar	6000	teksts

9. Tabula “**polls**” satur informāciju par aptauju un to izvēli.

Tajā ietilpst dati, piemēram, aptaujas variants ,varianti , autors.

9.tabula

Tabulas “**polls**” struktūra

Nr.	Nosaukums	Tips	Izmērs	Apraksts
1.	Id	int	10	unikāls identifikators
2.	subject	varchar	255	aptaujas nosaukums
3.	created	datetime	10	laiks kad izveidots
4.	modified	datetime	10	laiks kad

				mainīts
5.	status	enum	1	vai ir izveidots
6.	created_by	int	50	kurš izveidoja
7.	poll_desc	varchar	5000	apraksts
8.	locked	int	50	atvērts/aizv ērts

10. Tabula “**poll\_options**” satur informāciju par aptauju , un visus izveidotus variantus.  
(tas nosaukums un unikālais id).

10.tabula

Tabulas “**poll\_options**” struktūra

Nr.	Nosaukums	Tips	Izmērs	Apraksts
1.	Id	int	10	unikāls identifikators
2.	poll_id	int	50	aptauja identifikators( kur ir izvēle)
3.	name	varchar	255	variantu nosaukums
4.	created	datetime	10	kad



				izveidots
5.	modified	datetime	10	kad izmainits
6.	status	enum	1	status vai ir izveidots

11. Tabula “**poll\_votes**” satur informāciju par balsošanu aptaujā.

11.tabula

Tabulas “**poll\_votes**” struktūra

Nr.	Nosaukums	Tips	Izmērs	Apraksts
1.	Id	int	10	unikāls identifikators
2.	poll_id	int	50	izvēlētie varianti
3.	poll_option_id	int	50	unikāls identifikators
4.	vote_by	int	50	kurš lietotājs balso

12. Tabula “**Categories**” satur informāciju par kategorijām.  
Tajā ietilpst dati, piemēram, kategorijas nosaukums un apraksts.

12.tabula

Tabulas “**Categories**” struktūra

Nr.	Nosaukums	Tips	Izmērs	Apraksts
1.	cat_id	int	10	unikāls identifikators
2.	cat_name	varchar	255	kategorijas nosaukums
3.	cat_description	varchar	255	apraksts

13. Tabula “**pwdreset**” satur informāciju par paroli un to mainīšanu

13.tabula

Tabulas “**pwdreset**” struktūra

Nr.	Nosaukums	Tips	Izmērs	Apraksts
1.	pwdResetID	int	10	unikāls identifikators
2.	pwdResetEmail	text	255	pastas mainīšana

3.	pwdREsetSelecto r	text	255	paroles mainišana
4.	pwdResetToken	longtext	255	paroles mainišana
5.	pwdResetExpires	text	255	paroles mainišana kļūda

## LIETOTĀJA CEĻVEDIS

### • Sistēmas prasības aparatūrai un programmatūrai

Neskatoties uz to, ka sistēma ir realizētā ar WEB-tehnoģiju palīdzību un neprasa specifiskas ierīces, ir daži prasījumi stabilai darbībai:

- Windows operētājsistēma.
- Koda redaktors(piemēram: Visual Studio Code)
- PHP bāzēts rīks Laravel
- Stiprs un stabils interneta signāls.
- RAM datora atmiņa vismaz 2 Gb.
- Web-aplikācijai ir jāstrādā no visiem interneta pārlūkiem, bet labāk ņemt kādu no pazīstamākajiem, piemēram: Opera, Edge, Firefox utt.

### • Sistēmas instalācija un palaišana

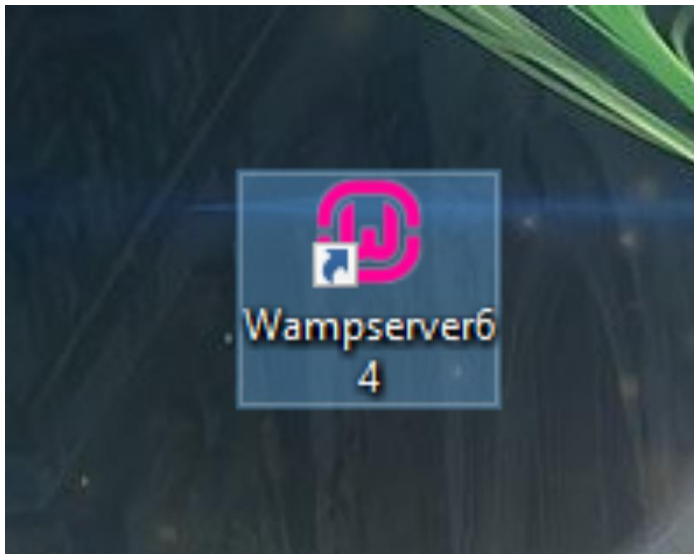
Sakarā ar to, ka programma ir realizēta ar WEB-tehnoloģijas palīdzību, lai to palaist ir jāveic sekojošus soļus:

- Uzinstalēt uz datoru serveri, ar datubāzi un iespēju uztaisīt virtuālo hostu(piemēram:

Wampp)

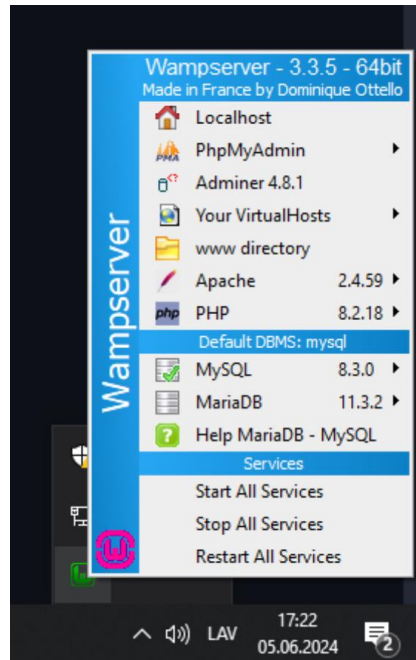
- Uzinstalēt uz datoru PHP programmēšanas valodu(nāc iekšā ar Wampp)
- Uzinstalēt Composer - lietojumprogrammas līmeņa atkarību pārvaldnieks PHP programmēšanas valodai.
- Lejupielādēt koda mapi

([https://github.com/rvt-prog-kval-24/programm-taju\\_soci-lais\\_tikls](https://github.com/rvt-prog-kval-24/programm-taju_soci-lais_tikls)) un ielikt zem servera mapes C:/: diskā htdocs (ceļš C:\wampp\www).



4.1. attēls - wamp icon.

- Pirmokārt , jāatvert Wamp programmu( PhpMyadmin un Apache , tik atvēras automatiski.) 4.1. att.



4.2. attēls - wamp menu.

- Pēc tam aiziem uz windows menu un tur nepieciešams nospiegt uz **wamp** icon un pēc tam uz **PhpMyAdmin**.(4.3. att)



**Language**

English

**Log In**

Username: root

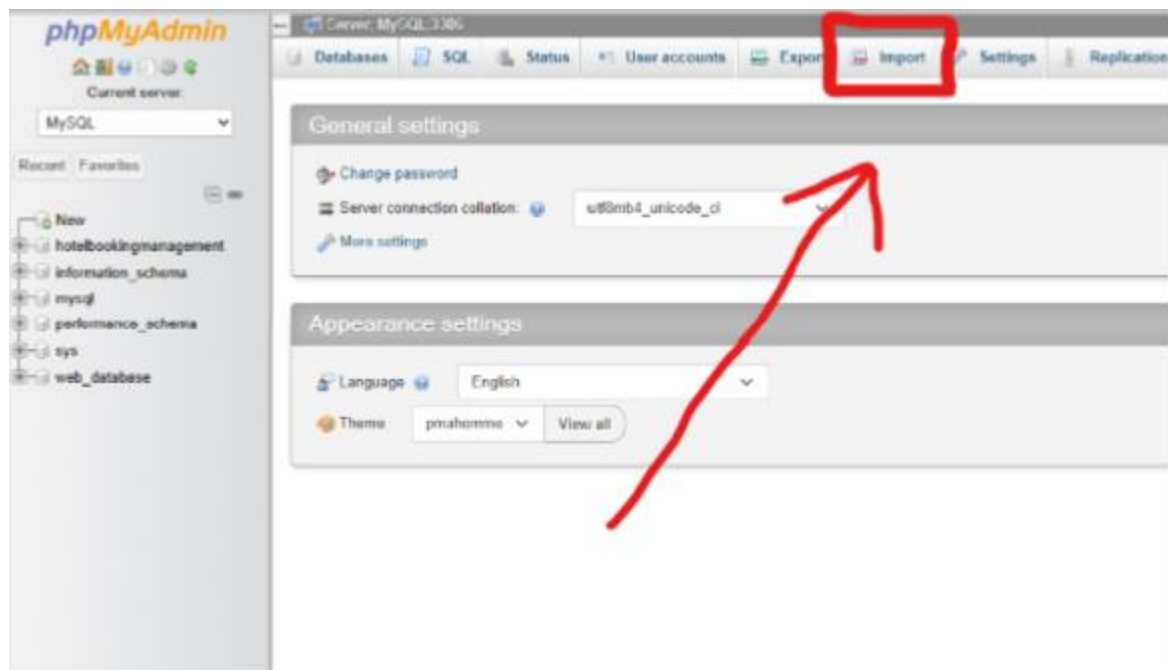
Password:

Server choice: MySQL

**Log in**

4.3. attēls - phpMyAdmin menu.

Iekš PhpMyAdmin laukiem rakstām - username: root , paroli nerakstām. (4.3. att.)



4.4. attēls - importēšana.

Jānospiež pogu **Import** (4.4. att.)

File to import:

File may be compressed (gzip, zip) or uncompressed.  
A compressed file's name must end in [format].[compression]. Example: .sql.zip

Browse your computer: (Max: 128MiB)

Избран файл: web\_database.sql

You may also drag and drop a file on any page.

Character set of the file:

utf-8

4.5 attēls - datu bāzes importēšana.

Jāimporte failu - **web\_database.sql** (4.5. att)

**Format**

SQL

**Format-specific options:**

SQL compatibility mode:

NONE

☒ Do not use AUTO\_INCREMENT for zero values

**Import**

4.6 attēls - Import Poga.

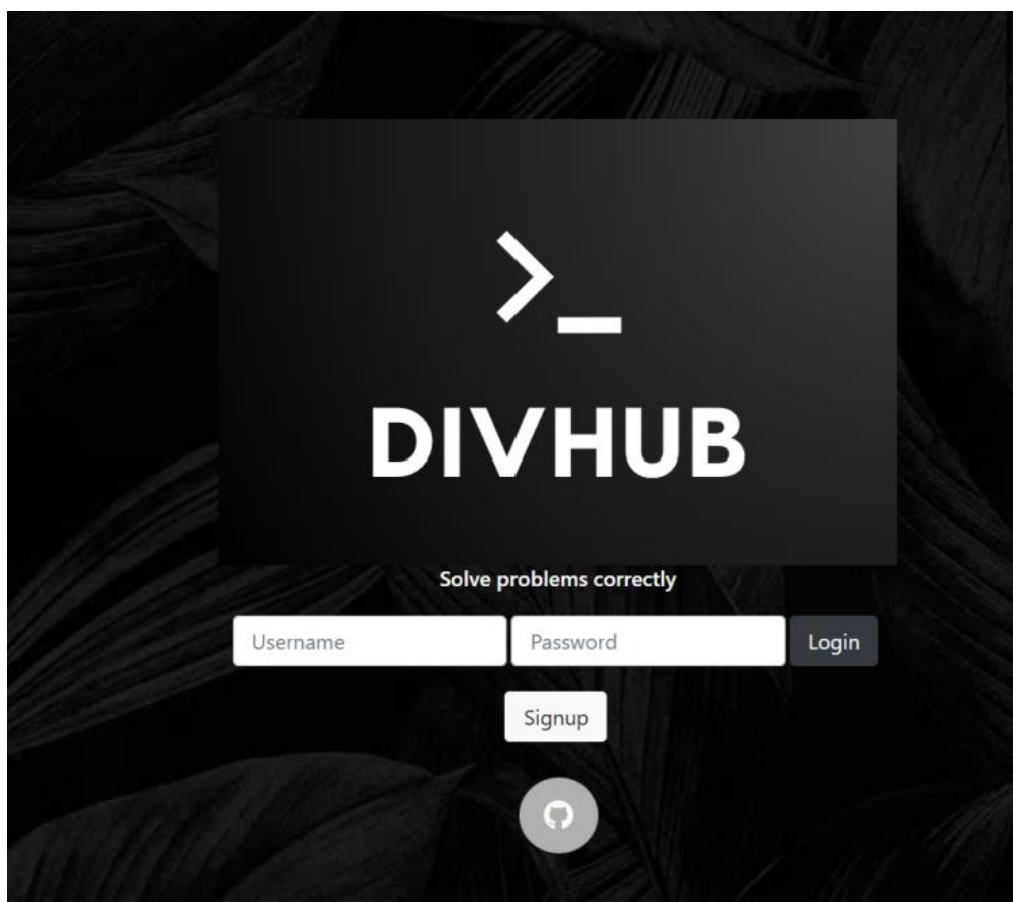
Jaiesniedz pogu - **Import** un aizliagt phpMyAdmin (4.6. att)

- Pieslēgties jebkuram interneta tīklam vai izmantot mobilos datus.
- Izvēlēties pārlūku, ar kuru palīdzību atvērt pašu sistēmu.
- Un ierakstīt linku: localhost/website



# • Programmas apraksts

- *Autorizācija reģistrētiem lietotājiem*
- Atvēram pārlūkā saitī: <http://localhost/website/login.php>
- Pirmokart paradas login lappuse , ir divi lauki - sēgvardām un parolei.Arī divas pogas – login/signup. Registrācijai nospiežam pogu - **Signup** (5.1 att.)



5.1 attēls - login lappaspuse.

Reģistrācijas lappa - Jāievada datus lai pieteikties. Obligatie dati - slēgvards , pasts , parole.(5.2. att)

Signup and Begin!

Username: Abcd Email: pasts123@gmail.com

Password: parole1234 Confirm Password: \*\*\*\*\*

Optional

First Name: Vards Last Name: Uzvars

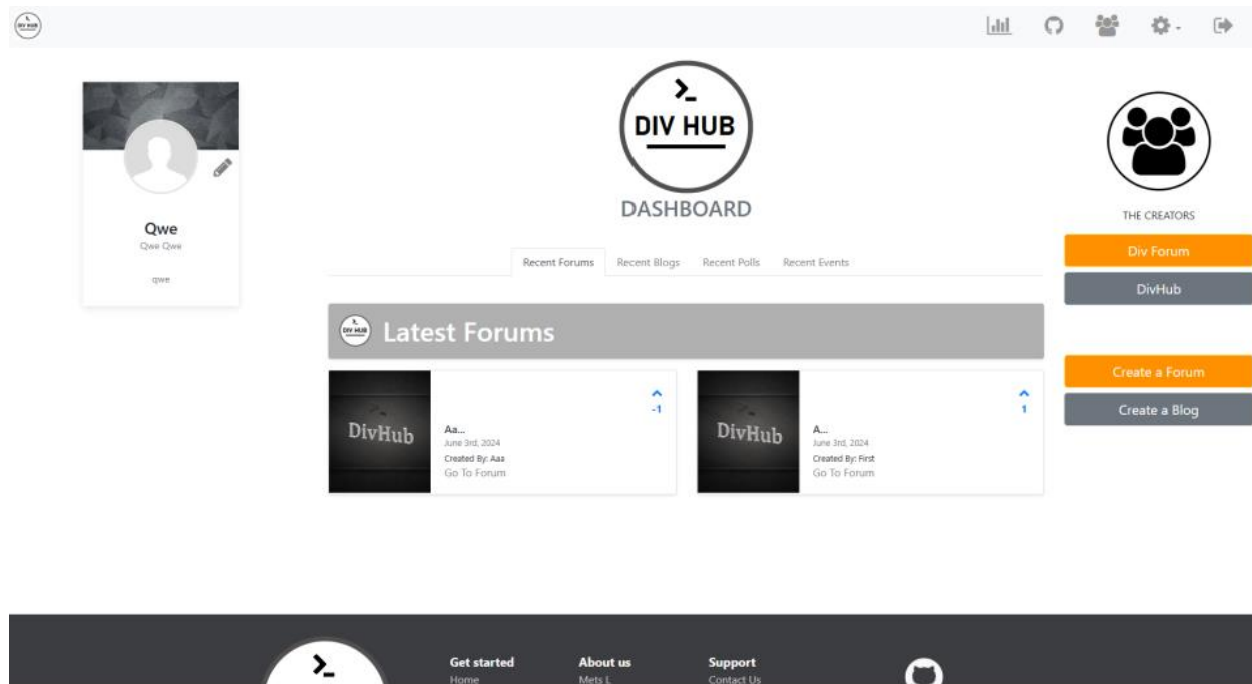
Profile Picture: [Placeholder Image] Set Avatar

Headline: never give up!

Bio: ruby programmer

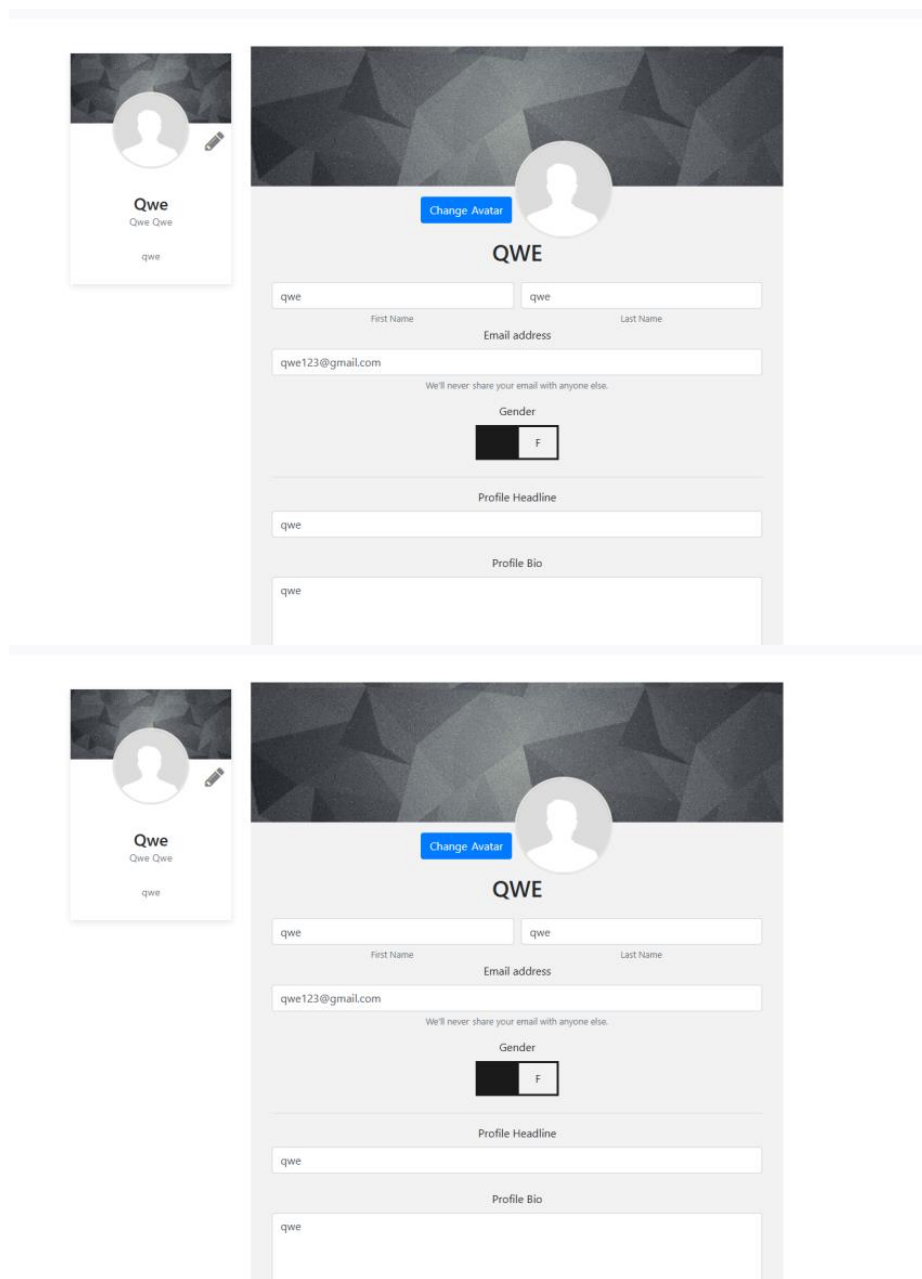
5.2. attēls - reģistracija.

Galvena sociāla tīkla lappa kur ir viss vajadzīgs , profila mainīšana , pedējie jautājumi un pogas orientācijai vietnē- (5.3. att)



5.3. attēls - galvena lappa.

Šeit ir iespēja rediģēt profilu un to datus, vārda un uzvārda un pasta mainīšana ir pieejama.- (5.4. att)



5.4. attēls- datu redīgēšana.

Ir iespēja mainīt paroli, datus par sevi. - (5.5. att)

We'll never share your email with anyone else.

Gender

☐ ☒ F

---

Profile Headline

qwe

Profile Bio

qwe

---

Change Password

Current Password

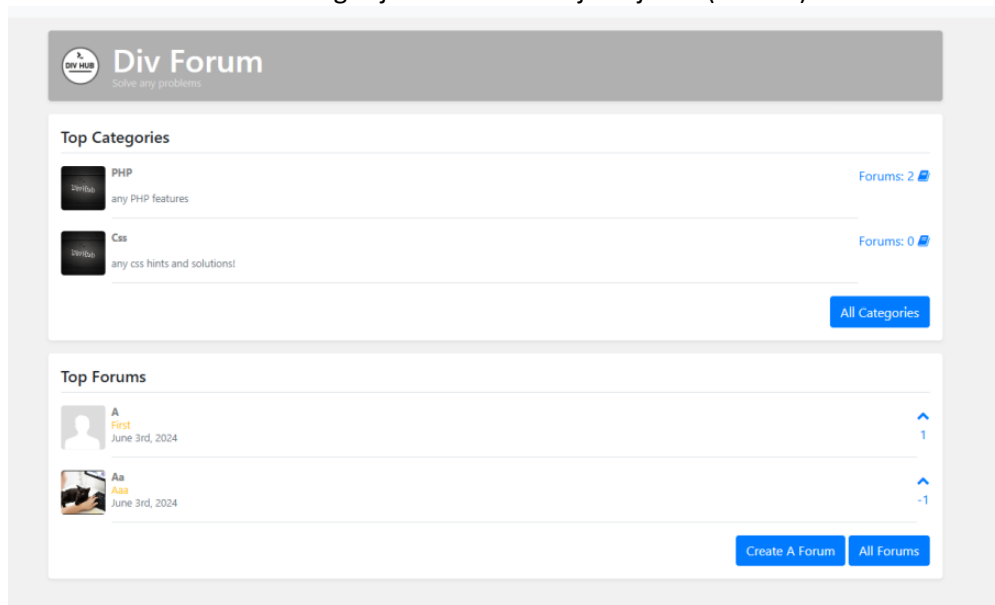
New Password

Repeat New Password

Update Profile

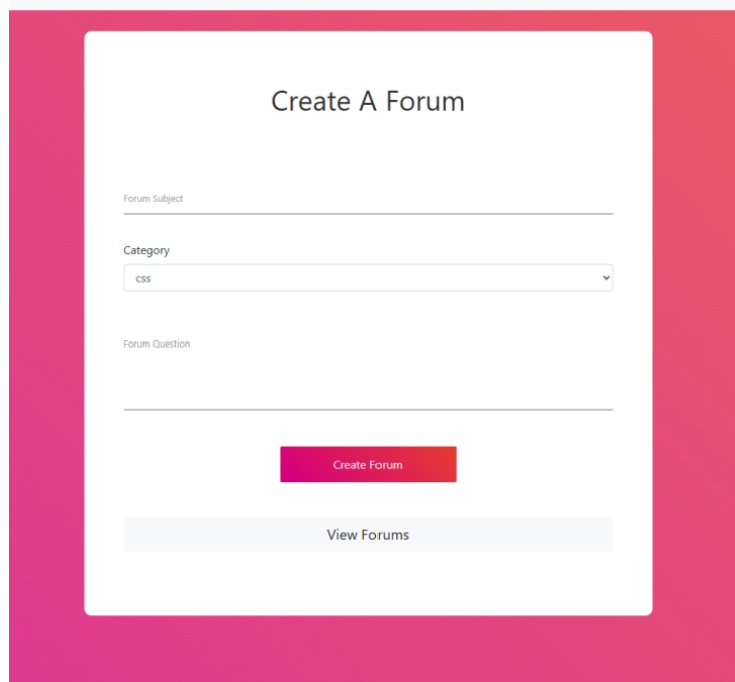
5.5. attēls - paroles maiņa.

## Kategorijas un izveidotie jautājumi -(5.6. att)



5.6.attēls - Kategorijas un jautājumi.

Ir iespēja uzdot jautājumu un izvēlēties vajadzīgo kategoriju. - (5.7. att)



The image shows a web form titled "Create A Forum" set against a pink background. The form is white and contains the following elements: a text input field for "Forum Subject", a dropdown menu for "Category" with "css" selected, a text input field for "Forum Question", a red "Create Forum" button, and a light gray "View Forums" button.

Create A Forum

Forum Subject

Category

css

Forum Question

Create Forum


View Forums

5.7. attēls - jautājumu veidošana.

Ir iespēja veidot jaunus komentarus ,dzēst tos un balsot.-(5.8.att)

June 5th, 2024

#5



first

aa

Joined:


Sep 27, 2018

Messages:


3


Likes:

0




:3





0



DIV Forum

Submit reply

5.8. attēls - komentaru veidošana.

## NOBEIGUMS

Sistēmas izstrādes gaitā tika izpildīti vairāki uzdevumi un atrisinātas problēmas, un projekts šobrīd ir gandrīz pabeigts. Lietotājam draudzīgums, vienkāršs interfeiss un optimizācija pat zemas klases ierīcēm bija galvenais projektēšanas un izstrādes procesā. Tehnoloģijas strauji attīstās, un ir



nepieciešami pastāvīgi uzlabojumi un jauninājumi. Darbā tika izmantotas atbilstošas tehnoloģijas un rīki datu bāzes operāciju veikšanai, izmantojot SQL Server, nodrošinot elastību turpmākajām platformas izmaiņām un uzlabojumiem.

Projekta izstrādes gaitā tika izmantots Visual Studio Code, kas piedāvā efektīvu programmatūras izstrādes vidi. Lietotāja saskarne ir izstrādāta, izmantojot HTML5, CSS3 un JavaScript, lai nodrošinātu optimālu lietotāja pieredzi tīmeklī. Šīs valodas un tehnoloģijas ļauj izveidot interaktīvas un bagātīgas lietotāja saskarnes. Katrs no šiem rīkiem un tehnoloģijām tika pielietots konkrētiem uzdevumiem vai moduļiem, nodrošinot daudzpusīgu pieeju projektam.

Sākotnēji bija problēmas ar atsevišķu funkciju izstrādi un ieviešanu. bet jo vairāk iedziļinājos projektā, jo labāk tas sanāca!

## INFORMĀCIJAS AVOTI

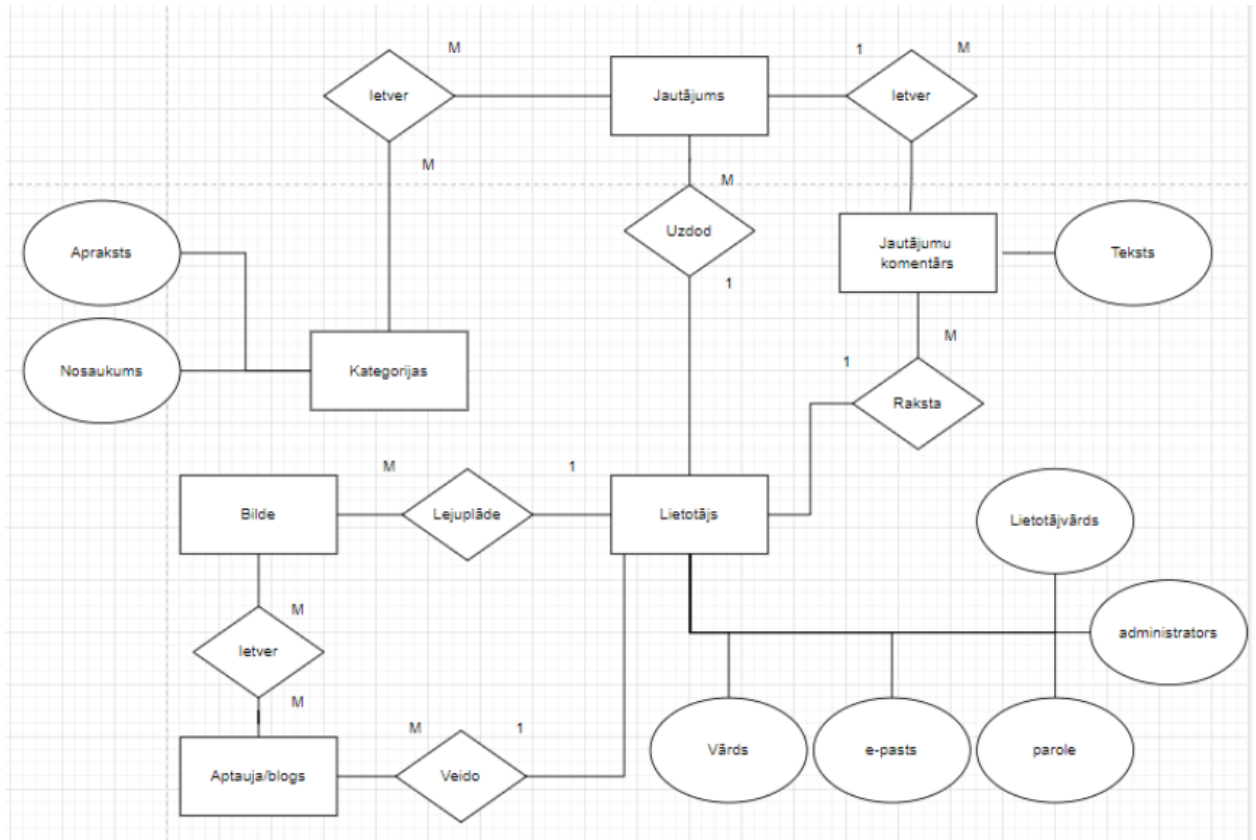
- JavaScript (JS) dokumentācija - <https://developer.mozilla.org/en-US/docs/Web/JavaScript> - (Resurss apskatīts 20.04.2024)
- PHP dokumentācija - <https://www.php.net/manual/en/> - (Resurss apskatīts 20.05.2024)
- CSS dokumentācija - <https://developer.mozilla.org/en-US/docs/Web/CSS> - (Resurss apskatīts 15.5.2024)
- MySQL (SQL) dokumentācija - <https://dev.mysql.com/doc/> - (Resurss apskatīts 01.05.2024)
- Github - [https://github.com/rvt-prog-kval-24/programm-taju\\_soci-lais\\_tikls-](https://github.com/rvt-prog-kval-24/programm-taju_soci-lais_tikls-) (Resurss

apskatīts 1.06.2024)

- Bootstrap - <https://getbootstrap.com/docs/5.0/getting-started/introduction/> - (Resurss  
apskatīts 22.05.2024)

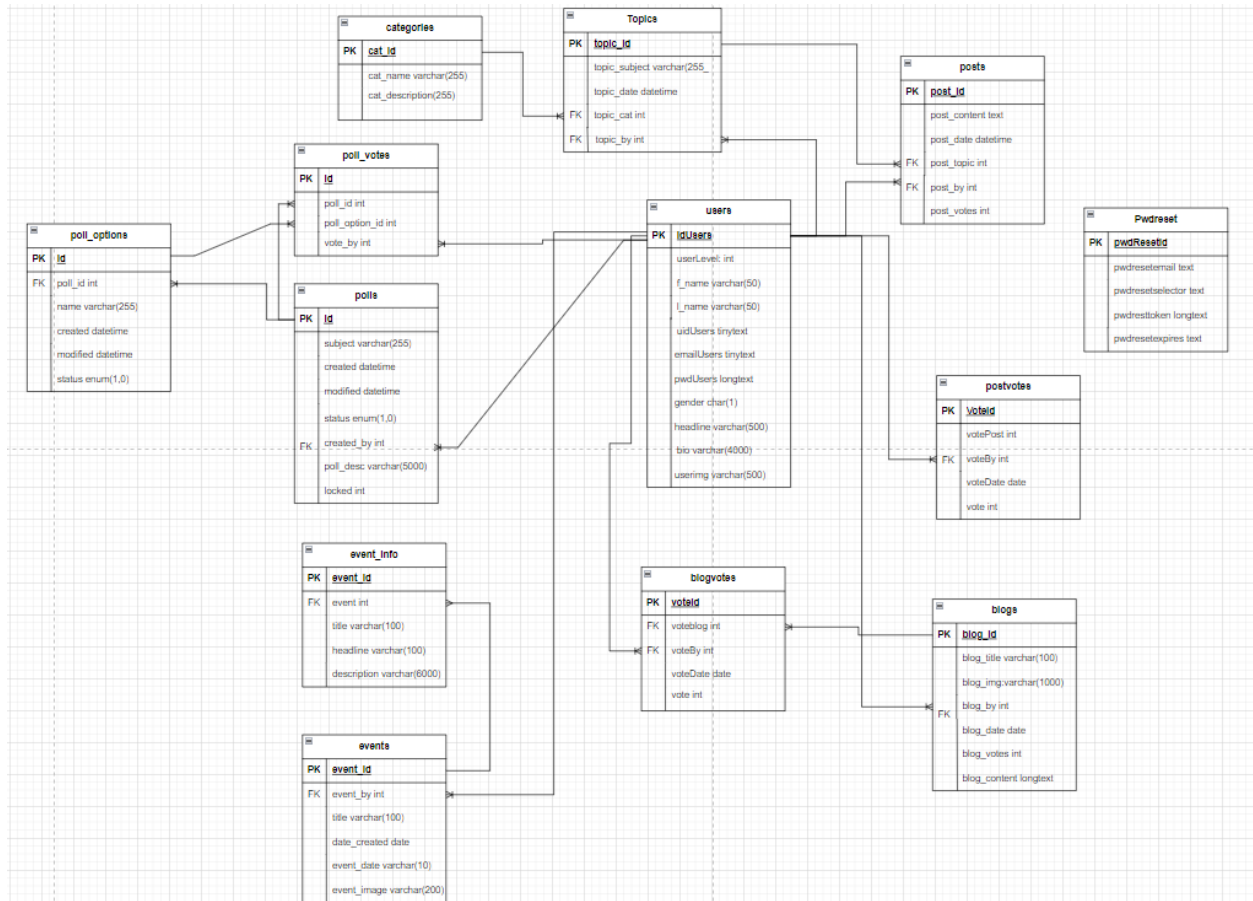
**PIELIKUMI**

- pielikums



ER-modelis

- pielikums



Tabulu relāciju shēma

Reģistrācijas

kods:

```
<?php
```

```
if (isset($_POST['signup-submit']))  
{
```

```
    require 'dbh.inc.php';
```

```
    $userName = $_POST['uid'];  
    $email = $_POST['mail'];  
    $password = $_POST['pwd'];  
    $passwordRepeat = $_POST['pwd-repeat'];  
    $gender = $_POST['gender'];  
    $headline = $_POST['headline'];  
    $bio = $_POST['bio'];  
    $f_name = $_POST['f-name'];  
    $l_name = $_POST['l-name'];
```

```
    if (empty($userName) || empty($email) || empty($password) || empty($passwordRepeat))  
    {
```

```
        header("Location: ../signup.php?error=emptyfields&uid=".$userName."&mail=".$email);  
        exit();  
    }
```

```
    else if (!filter_var($email, FILTER_VALIDATE_EMAIL) && !preg_match("/^[a-zA-Z0-9]*$/", $userName))
```

```

{
    header("Location: ../signup.php?error=invalidmailuid");
    exit();
}
else if (!filter_var($email, FILTER_VALIDATE_EMAIL))
{
    header("Location: ../signup.php?error=invalidmail&uid=".$userName);
    exit();
}
else if (!preg_match("/^[a-zA-Z0-9]*$/", $userName))
{
    header("Location: ../signup.php?error=invaliduid&mail=".$email);
    exit();
}
else if ($password !== $passwordRepeat)
{
    header("Location: ../signup.php?error=passwordcheck&uid=".$userName."&mail=".$email);
    exit();
}
else
{
    // checking if a user already exists with the given username
    $sql = "select uidUsers from users where uidUsers=?;";
    $stmt = mysqli_stmt_init($conn);
    if (!mysqli_stmt_prepare($stmt, $sql))
    {
        header("Location: ../signup.php?error=sqlerror");
        exit();
    }
    else

```

```

{
    mysqli_stmt_bind_param($stmt, "s", $userName);
    mysqli_stmt_execute($stmt);
    mysqli_stmt_store_result($stmt);

    $resultCheck = mysqli_stmt_num_rows($stmt);

    if ($resultCheck > 0)
    {
        header("Location: ../signup.php?error=usertaken&mail=".$email);
        exit();
    }
    else
    {
        $FileNameNew = 'default.png';
        require 'upload.inc.php';

        $sql = "insert into users(uidUsers, emailUsers, f_name, l_name, pwdUsers, gender,
"
            . "headline, bio, userImg) "
            . "values (?, ?, ?, ?, ?, ?, ?, ?)";
        $stmt = mysqli_stmt_init($conn);
        if (!mysqli_stmt_prepare($stmt, $sql))
        {
            header("Location: ../signup.php?error=sqlerror");
            exit();
        }
        else
        {
            $hashedPwd = password_hash($password, PASSWORD_DEFAULT);

```

```

        mysqli_stmt_bind_param($stmt, "ssssssss", $userName, $email, $f_name,
    $l_name,

        $hashedPwd, $gender,
        $headline, $bio, $FileNameNew);
    mysqli_stmt_execute($stmt);
    mysqli_stmt_store_result($stmt);

    header("Location: ../signup.php?signup=success");
    exit();
    }
    }
    }
}

mysqli_stmt_close($stmt);
mysqli_close($conn);

}

else
{
    header("Location: ../signup.php");
    exit();
}3

```



