JELGAVAS TEHNIKUMS

**RALFS CVIKLINSKIS  
  
Mājas lapas “NoRokas” prototipa izstrāde   
kvalifikācijas darbs ieguvei kvalifikācijā - programmēšanas tehniķis**

Darba izpildītājs: ………………………..….. 410. gr. Izgl. R. Cviklinskis  
*paraksts, datums*

Jelgava 2025……………

ANOTĀCIJA

Ralfs C. *WEB* lapas izstrāde mantu pārdošanai: kvalifikācijas darbs. Jelgava: JT, 2026….

Šis darbs risina problēmu, ka pērkot mantas un produktus no citiem cilvēkiem, nevar zināt to kvalitāti, piemēram, malku, dārzeņus u.c., tāpēc šajā saitē to problēmu atrisināt varēs ar iespēju komentēt zem produktiem un lasīt citu komentārus.

Mērķis ir izveidot *WEB* lapu kur var ielogoties, veidot rakstus, tos rediģēt, dzēst, redzēt citu lietotāju rakstus un zem tiem komentēt.

Darbs tiek izstrādāts individuāli.

ANNOTATION

Ralfs C. WEB page development for selling goods: qualification work. Jelgava: JT, 2026….

This work solves the problem that, when buying goods and products from other people, you cannot know their quality, for example, firewood, vegetables, etc., so this site will solve this problem with the ability to comment under products and read other people's comments.

The goal is to create a WEB page where you can log in, create posts, edit, delete them, see other users posts and comment under them.

The work is developed individually.

Saturs

[ievads 8](#_Toc214982866)

[1. UZDEVUMA NOSTĀDNE 11](#_Toc214982867)

[1.1. Līdzīgie produkti: 11](#_Toc214982868)

[1.2. Kopīgie secinājumi: 14](#_Toc214982869)

[1.3. Mana projekta unikalitāte: 14](#_Toc214982870)

[1.4. Projekta mērķauditorija 14](#_Toc214982871)

[1.5. Gala rezultāta izskats: 14](#_Toc214982872)

[1.6. Projekta ierobežojumi: 14](#_Toc214982873)

[2. programmatūras izstrādes vide 15](#_Toc214982874)

[2.1. Teksta redaktors 15](#_Toc214982875)

[2.1.1. *Visual Studio Code*: 15](#_Toc214982876)

[2.2. Datubāze un tās rīki 15](#_Toc214982877)

[2.2.1. *MySQL*: 15](#_Toc214982878)

[2.2.2. *HeidiSQL*: 15](#_Toc214982879)

[2.3. Programmēšanas valodas 15](#_Toc214982880)

[2.3.1. *HTML*: 15](#_Toc214982881)

[2.3.2. *JS*: 16](#_Toc214982882)

[2.3.3. CSS: 16](#_Toc214982883)

[2.3.4. *PHP*: 16](#_Toc214982884)

[2.4. Ietvari 16](#_Toc214982885)

[2.4.1. *CodeIgniter* 16](#_Toc214982886)

[3. programmatūras prasību specifikācija 18](#_Toc214982887)

[3.1. Sākotnēja ideja 18](#_Toc214982888)

[3.2. Nefunkcionālās prasības 18](#_Toc214982889)

[3.2.1. Navigācijas josla: 18](#_Toc214982890)

[3.2.2. Mājas lapa: 19](#_Toc214982891)

[3.2.3. Raksta logs: 20](#_Toc214982892)

[3.2.4. Profila Lapa: 20](#_Toc214982893)

[3.2.5. Konta izveidošanas lapa: 21](#_Toc214982894)

[3.2.6. Iestatījumu logs: 21](#_Toc214982895)

[3.2.7. Konta izdzēšanas apstiprināšanas logs: 22](#_Toc214982896)

[3.2.8. Par mums lapa: 23](#_Toc214982897)

[3.2.9. Rakstu veidošana: 23](#_Toc214982898)

[3.2.10. Raksta dzēšanas apstiprinājums: 25](#_Toc214982899)

[3.2.11. Raksta filtrēšanas logs 25](#_Toc214982900)

[3.3. Raksta detalizēta informācija 26](#_Toc214982901)

[3.4. Funkcionālas prasības 26](#_Toc214982902)

[3.4.1. Galvenes pogu funkcijas 26](#_Toc214982903)

[3.4.2. Konta izveidošana 26](#_Toc214982904)

[3.4.3. Lietotāja ielogošana 27](#_Toc214982905)

[3.4.4. Profila bildes mainīšana 28](#_Toc214982906)

[3.4.5. Raksta veidošana 28](#_Toc214982907)

[3.4.6. Raksta rediģēšana 29](#_Toc214982908)

[3.4.7. Raksta dzēšana 29](#_Toc214982909)

[3.4.8. Raksta filtrēšana 29](#_Toc214982910)

[3.4.9. Raksta meklēšana 30](#_Toc214982911)

[3.4.10. Raksta informācija 31](#_Toc214982912)

[3.5. Sistēmas struktūras modelis 31](#_Toc214982913)

[3.5.1. Datubāzes apraksts 31](#_Toc214982914)

[3.5.2. Tabula “account” 32](#_Toc214982915)

[3.5.3. Tabula “user” 32](#_Toc214982916)

[3.5.4. Tabula “posts” 33](#_Toc214982917)

[3.5.5. Tabula “tags” 33](#_Toc214982918)

[Secinājumi 34](#_Toc214982919)

[Izmantoti avoti 35](#_Toc214982920)

[pielikumi 37](#_Toc214982921)

[1. Pielikums. ***index.php*** 38](#_Toc214982922)

ATTĒLU SARAKSTS

[1.attēls. **Utrupe** 11](#_Toc214982923)

[2.attēls. **Ss.com** 11](#_Toc214982924)

[3.attēls. **Zviedrumebeles** 12](#_Toc214982925)

[4.attēls. **Marketplace** 13](#_Toc214982926)

[5.attēls. **Ptmebeles** 13](#_Toc214982927)

[6.attēls. **Sākotnējā struktūrskice** 18](#_Toc214982928)

[7.attēls. **Navigācijas joslas struktūrskice** 19](#_Toc214982929)

[8.attēls. **Profila loga struktūrskice** 19](#_Toc214982930)

[9.attēls. **Mājas-lapas-struktūrskice** 20](#_Toc214982931)

[10.attēls. **Rakstastruktūrskice** 20](#_Toc214982932)

[11.attēls. **Profila lapas struktūrskice** 21](#_Toc214982933)

[12.attēls. **Konta izveidošanas loga struktūrskice** 21](#_Toc214982934)

[13.attēls. **Konta iestatījumu struktūrskice** 22](#_Toc214982935)

[14.attēls. **Konta dzēšanas logs** 22](#_Toc214982936)

[15.attēls. **Konta dzēšanas logs** 23](#_Toc214982937)

[16.attēls. **Par mums lapas struktūrskice** 23](#_Toc214982938)

[17.attēls. **Raksta formas pamata lauki struktūrskice** 24](#_Toc214982939)

[18.attēls. **Raksta formas bildes struktūrskice** 24](#_Toc214982940)

[19.attēls. **Raksta formas *tags* struktūrskice** 25](#_Toc214982941)

[20.attēls. **Raksta dzēšanas apstiprinājums struktūrskice** 25](#_Toc214982942)

[21.attēls. **Raksta filtrēšanas logs struktūrskice** 25](#_Toc214982943)

[22.attēls. **Raksta detaļas struktūrskice** 26](#_Toc214982944)

[23.attēls. **Konta veidošanas blokshēma** 27](#_Toc214982945)

[24.attēls. **Ielogošanās blokshēma** 28](#_Toc214982946)

[25.attēls. **Rakstu veidošanas blokshēma** 29](#_Toc214982947)

[26.attēls. **Rakstu filtrēšanas blokshēma** 30](#_Toc214982948)

[27.attēls. **Savu rakstu parādīšanas blokshēma** 30](#_Toc214982949)

[28.attēls. **Rakstu meklēšanas blokshēma** 31](#_Toc214982950)

[29.attēls. **Datubāzes struktūras diagramma** 32](#_Toc214982951)

tabulu saraksts

[1. tabula **“account”** 32](#_Toc214982952)

[2. tabula **“user”** 32](#_Toc214982953)

[3. tabula **“post”** 33](#_Toc214982954)

[4. tabula **“tags”** 33](#_Toc214982955)

# ievads

**Darba mērķis**:

Izstrādāt prototipa WEB lapu, kas nodrošina platformu lietotājiem pašražotu preču, pakalpojumu, mantu publicēšanai un pārdošanai.

**Darba uzdevumi**:

1. Izveidot iespēju, kur lietotājs var pieteikties, reģistrēties un mainīt profila iestatījumus.
2. Izveidot iespēju lietotājam veidot rakstus, kas saturēs informāciju par to, ko lietotājs vēlas pārdot.
3. Izveidot iespēju lietotājam rediģēt un dzēst savus rakstus.
4. Izveidot mājas lapu, kur var redzēt visus rakstus kas ir veidoti.
5. Izveidot filtrēšanas iespēju.
6. Izveidot iespēju lietotājam komentēt visus rakstus.
7. Izveidot iespēju redzēt zem raksta komentārus.
8. Testēt programmu tā prototipa izstrādes laikā.

SAĪSINĀJUMI UN AKRONĪMI, TO SKAIDROJUMI

|  |  |
| --- | --- |
| *PHP* | atvērtā pirmkoda skriptu valoda, kura sākotnēji bija paredzēta servera puses lietojumos dinamiska tīmekļa lapu ģenerēšanai[6]. |
| *JS* | programmēšanas valoda un tīmekļa galvenā tehnoloģija tīmekļa lapas darbības nodrošināšanai klienta pusē[8]. |
| *HTML* | standarta iezīmēšanas valoda dokumentiem, kas paredzēti rādīšanai tīmekļa pārlūkprogrammā. Tas nosaka tīmekļa satura saturu un struktūru[7]. |
| *CSS* | stila lapas valoda, ko izmanto, lai norādītu dokumenta prezentāciju un stilu, kas rakstīts iezīmēšanas valodā, piemēram, HTML vai XML[9]. |
| *DOM* | Dokumentu objektu modelis (*DOM*) savieno tīmekļa lapas ar skriptiem vai programmēšanas valodām, attēlojot dokumenta struktūru, piemēram, HTML, kas attēlo tīmekļa lapu, atmiņā.[2] |
| *API* | Lietojumprogrammu saskare (*API*) ir savienojums starp datoriem vai datorprogrammām. Tā ir programmatūras saskares veids, kas piedāvā pakalpojumu citām programmatūras daļām.[14] |

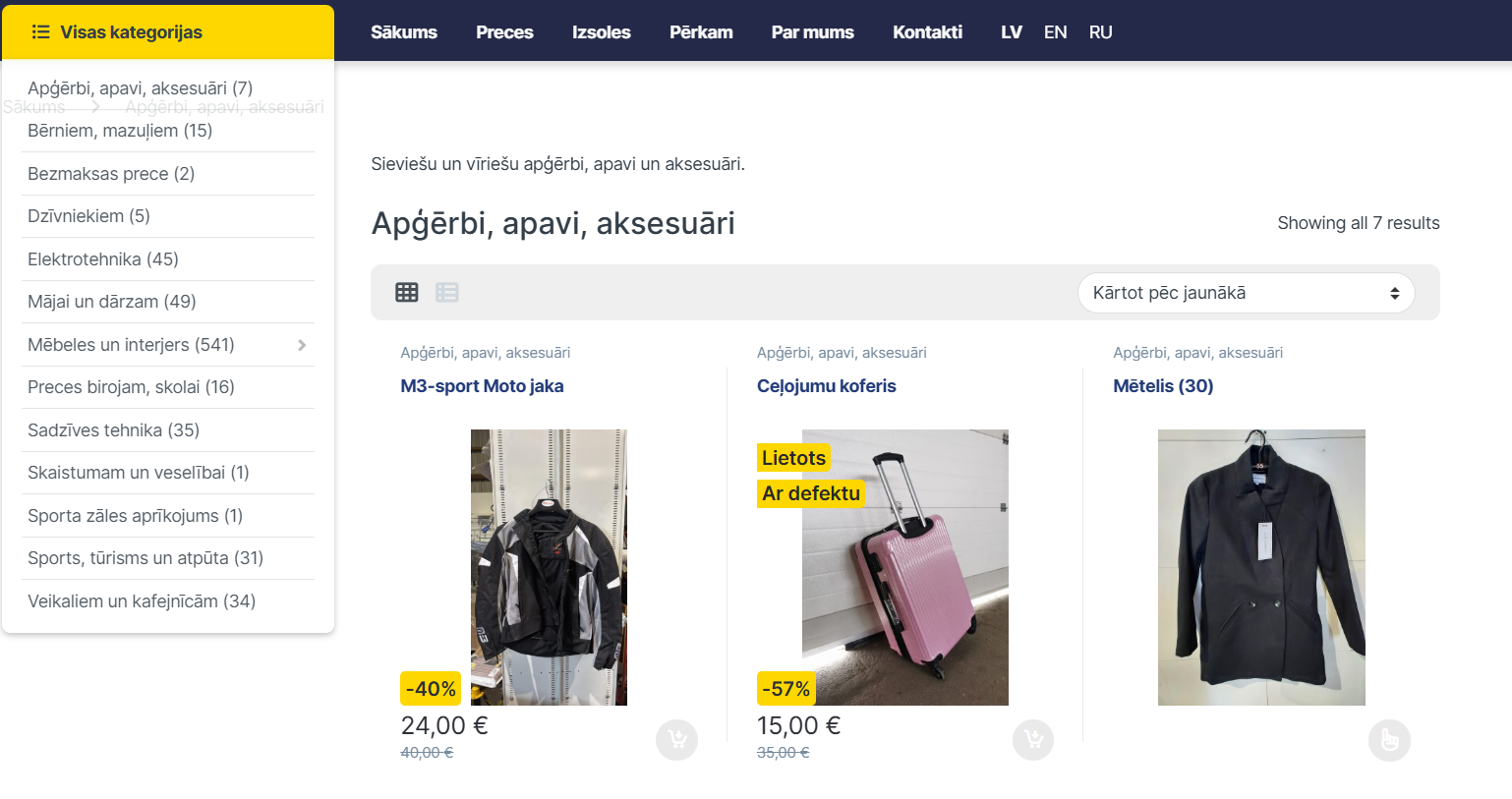
DOKUMENTĀCIJĀ IZMANTOTIE TERMINI UN TO SKAIDROJUMI

|  |  |
| --- | --- |
| *Post* | raksts, attēls vai cits satura vienums, kas publicēts tiešsaistē, parasti emuārā vai sociālajos saziņas līdzekļos[10]. |
| *Tags* | *tags* ir atslēgvārds vai termins, kas piešķirts kādai informācijai (piemēram, interneta grāmatzīmei, multividei, datu bāzes ierakstam vai datora failam)[11]. |

1. UZDEVUMA NOSTĀDNE

Manā projektā būs iespējams izveidot rakstus kur var ielikt bildes, nosaukumu, aprakstu un *tags*, šo rakstu būs iespējams redzēt citiem kas ir šajā saitē. Izmantojot visu *tags* sarakstu būs iespēja filtrēt visus rakstus*.* Būs iespēja ielogoties, kas dos iespēju rakstīt komentārus zem katra raksta, kurus arī būs iespējams redzēt visiem.

* 1. Līdzīgie produkti:
* Utrupe[18]



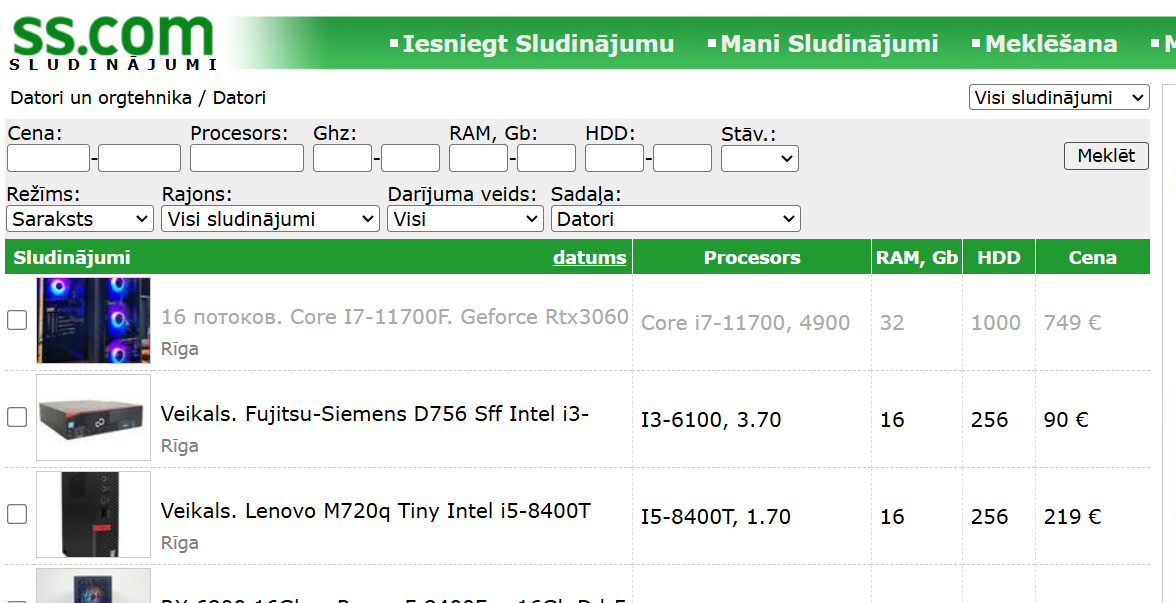
1.attēls. **Utrupe**

(skatīt 1. attēlu)Šajā saitē ir iespējams dot savas mantas, lietotas un jaunas, kuras tiks pārdotas lietotājiem, un daļā no cenas tiks dota tam kurš iedeva savu mantu. Komentēt nav iespējas, kas atņem iespēju sazināties ar citiem par mantas un saites kvalitāti.

Kopīgs: var pārdot savas mantas citiem

Atšķirīgs: nav komentāri

* ss.com[17]



2.attēls. **Ss.com**

(skatīt 2. attēlu)Šajā saitē jebkurš var pārdot ko vēlas, paši izvēlas cenu un mantas aprakstu, un paši arī piegādā mantu, vai arī tas kurš vēlas mantu dabūt pats to paņem. Šajā saitē nav komentāri.

Kopīgs: var pārdot savas mantas citiem ar pašizvēlētu cenu

Atšķirīgs: nav iespēja komentēt

* zviedru mebeles [19]



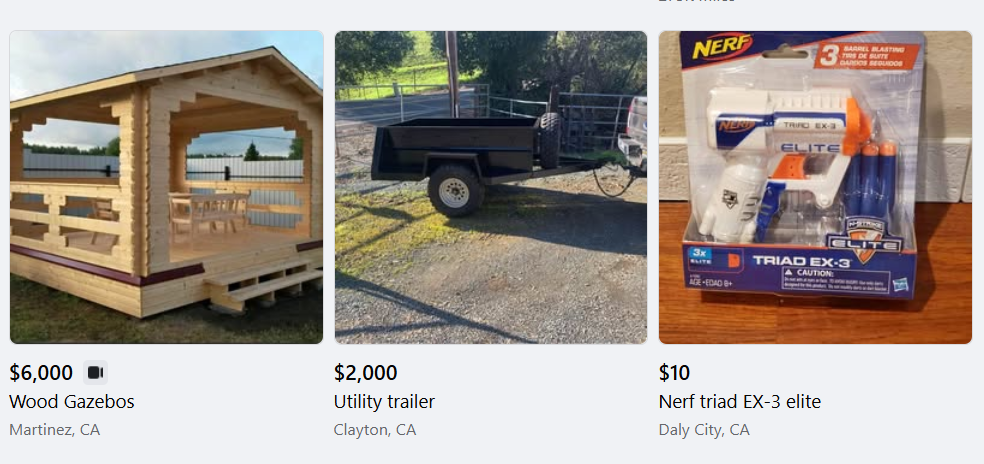
3.attēls. **Zviedrumebeles**

(skatīt 3. attēlu)Šajā saitē var apskatīt lietotas mēbeles kuras var nopirkt viņu veikalā. Mēbeles tiek pārbaudītas uz kvalitāti, un visas problēmas tiek pieminētas.

Kopīgs: var nopirkt lietotas mēbeles

Atšķirīgs: nav komentāri, pārdot var tikai uz vietas

* *Facebook marketplace* [3]



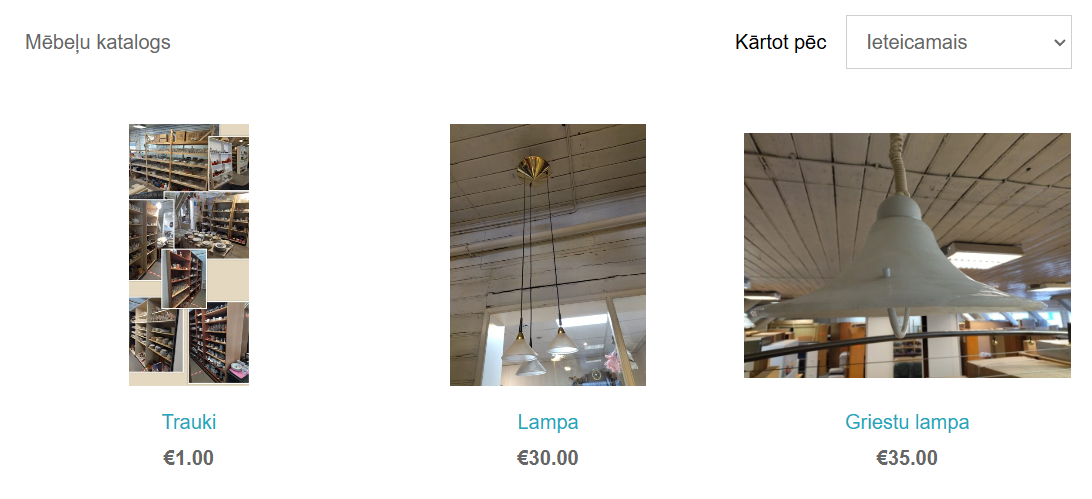
4.attēls. **Marketplace**

(skatīt 4. attēlu)Šajā saitē var apskatīt mantas kas tiek pārdotas, no krēsliem līdz mašīnām. Katrs var pārdot mantas par pašizvēlētu cenu, un jebkurš to var nopirkt. Visas pirkšanas notiek starp lietotājiem.

Kopīgs: var pārdot ko vēlas, izvēlas pats savu cenu

Atšķirīgs: nav iespējas komentēt.

* ptmebeles [16]



5.attēls. **Ptmebeles**

(skatīt 5. attēlu)Šajā saitē ir iespējams nopirkt lietotas mēbeles, pa lielākam mēbeles kuras var izmantot mājā. Jebkurš var pārdot savas mēbeles ja tās nav salauztas.

Kopīgs: var pārdot mājas mēbeles par savu cenu

Atšķirīgs: nav iespējams komentēt

* 1. Kopīgie secinājumi:

Pa lielākam visas saites kas pārdod lietotas mantas ir ļoti līdzīgas, lielāka atšķirība ir tā, ka dažreiz mantas pārdod pats lietotājs, dažreiz kompānija kas uztur saiti. Bet visbiežāk mantas pārdod pats lietotājs, un tās arī piegādā pircējam vai pircējs tās pats paņem. Galvenā problēma šajās saitēs ir tā, ka nevar komentēt, un pārdot pa lielākam var tikai mēbeles, paštaisītus produktus nevar, piemēram, malku, konservus u.c.

* 1. Mana projekta unikalitāte:

Manā projektā lietotājiem būs vairāk iespējas kontaktēties ar citiem lietotājiem, kas dos iespēju labāk izvēlēties mantu, un izvairīties no sliktām vai bojātām mantām, vēl būs iespēja pārdot sevis ražotus produktus, piemēram, malku un rotaslietas, kas dos vairāk cilvēkiem iespēju izmantot šo saiti.

* 1. Projekta mērķauditorija

Šīs projekts būs viss vairāk mērķēts uz tiem, kas paši veido un ražo produktus, piemēram, paštaisītas rotas lietas vai kurināmā malka, un kā ekstra lietotāji varēs pārdot lietotās mantas.

* 1. Gala rezultāta izskats:

Mana projekta galējais izskats varētu būt līdzīgs dažām no pieminētajām saitēm, bet ar vairāk funkcijām , piemēram, komentēšana kas palīdzēs lietotājam.

* 1. Projekta ierobežojumi:

1. Viss lielākais ierobežojums būs laiks, jo ar savām spējām šo projektu ātri nesanāks pabeigt, un visu laiku arī nevar iesaistīt šajā projektā, kas to tikai palēlinās.
2. Budžets nebūs problēma, jo šai saite vajadzētu tikai vienu nelielu datubāzi, kas daudz nemaksā, un pa lielākam būs viena cilvēka projekts.
3. programmatūras izstrādes vide
   1. Teksta redaktors
      1. *Visual Studio Code*:

*Visual Studio Code* ir bezmaksas, viegls, bet jaudīgs pirmkoda redaktors, kas darbojas gan datorā, gan tīmeklī un ir pieejams operētājsistēmām *Windows*, *macOS* un *Linux*. Tam ir iebūvēts *JS*, *TypeScript* un *Node.js* atbalsts, un tam ir bagātīga paplašinājumu ekosistēma citām programmēšanas valodām (piemēram, *C++*, *C#*, *Java*, *Python*, *PHP* un *Go*), izpildlaikiem (piemēram, .*NET* un *Unity*), vidēm (piemēram, *Docker* un *Kubernetes*) un mākoņiem (piemēram, *Amazon Web Services*, *Microsoft Azure* un *Google Cloud Platform*).[5]

Es izvēlējos šo teksta redaktoru, jo tas ir bez maksas, dod plašas iespējas programmēt, ir populārs un ērti izmantojams.

* 1. Datubāze un tās rīki
     1. *MySQL*:

*MySQL* ir relāciju datubāzes pārvaldības sistēma (*RDBMS*), kas ir bezmaksas, atvērtā koda un izmanto dažādas patentētas licences, tostarp *GNU* vispārējo publisko licenci (*GPL*). Kā *RDBMS*, *MySQL* izmanto *SQL*, lai pārvaldītu datus datubāzē. Tā organizē korelētus datus vienā vai vairākās datu tabulās, un šī korelācija palīdz strukturēt datus.[13]

Es izvēlējos *MySQL* savai datubāzei, jo šī sistēma man ir visvairāk pazīstama, tā ir bezmaksas un atbalsta daudzas sistēmas.

* + 1. *HeidiSQL*:

*HeidiSQL* ir bezmaksas un atvērtā koda programmatūra cilvēkiem, kas strādā ar datubāzēm, un tās mērķis ir būt intuitīvi lietojamai. *HeidiSQL* ļauj izveidot savienojumu ar dažādām datubāzes sistēmām, ieskaitot *MySQL*. [4]

Es izvelējos *HeidiSQL*, jo sen jau izmantoju, man pašam labi zināms, un tas strādā ar *MySQL* relāciju datubāzes pārvaldības sistēmu.

* 1. Programmēšanas valodas
     1. *HTML*:

Hiperteksta iezīmēšanas valoda (*HTML*) ir standarta iezīmēšanas valoda dokumentiem, kas paredzēti attēlošanai tīmekļa pārlūkprogrammā. Tā nosaka tīmekļa satura saturu un struktūru. Tīmekļa pārlūkprogrammas saņem *HTML* dokumentus no tīmekļa servera vai lokālās krātuves un atveido šos dokumentus multimediju tīmekļa lapās. *HTML* semantiski apraksta tīmekļa lapas struktūru un sākotnēji ietvēra norādes tās izskatam.[7]

Es izvēlējos *HTML*, jo tā ir standarta tīmekļa lapu iezīmēšanas valoda, kas no nozīmē to, ka to atbalsta gandrīz katra sistēma un programma, ka nodrošina tās strādāšanu uz jebkuras sistēmas.

* + 1. *JS*:

*JS(JavaScript)* ir augsta līmeņa, bieži vien tieši laikā kompilēta valoda, kas atbilst *ECMAScript* standartam. Tai ir dinamiska tipizēšana, uz prototipiem balstīta objekt orientācija un pirmās klases funkcijas. Tā ir daudz paradigma, kas atbalsta notikumu vadītus, funkcionālus un imperatīvus programmēšanas stilus. Tai ir lietojumprogrammu programmēšanas saskarnes (*API*) darbam ar tekstu, datumiem, regulārām izteiksmēm, standarta datu struktūrām un dokumentu objektu modeli (*DOM*). Tīmekļa pārlūkprogrammām ir īpaša *JavaScript* programma, kas izpilda klienta kodu.[12]

*JS* tika izvēlēts, jo tas ir cieši saistīts ar tīmekļa lapām, kas samazina risku par programmas nestrādāšanu un dod daudz kontroli pār lapām. *JS* vēl ir pieejamas daudz bibliotēkas, kas var dod vēl vairāk kontroli pār lapu un iespēju veidot ļoti īpašas darbības kas atšķiras no citām tīmekļa lapām

* + 1. CSS:

Kaskādes stila lapas (*CSS*) ir stila lapu valoda, ko izmanto, lai norādītu dokumenta, kas rakstīts iezīmēšanas valodā, piemēram, *HTML* vai *XML*, noformējumu un stilu. *CSS* ir globālā tīmekļa stūrakmens tehnoloģija līdzās *HTML* un *JavaScript*.[15]

Izmantoju *CSS*, jo tas ir viens no labākajiem veidiem, kā uztaisīt stilu *HTML*, un tas arī ir izmantojams jebkurā sistēmā.

* + 1. *PHP*:

Populāra vispārējas nozīmes skriptvaloda, kas ir īpaši piemērota tīmekļa izstrādei. Ātra, elastīga un pragmatiska. PHP kodu tīmekļa serverī parasti apstrādā PHP interpretētājs, kas ieviests kā modulis vai *Common Gateway Interface* (*CGI*) izpildāmais fails.[6]

Izvēlējos *PHP*, jo tas ļauj vieglāk savienot saiti ar datubāzi, un tas arī satur rīkus lai veidotu saiti.

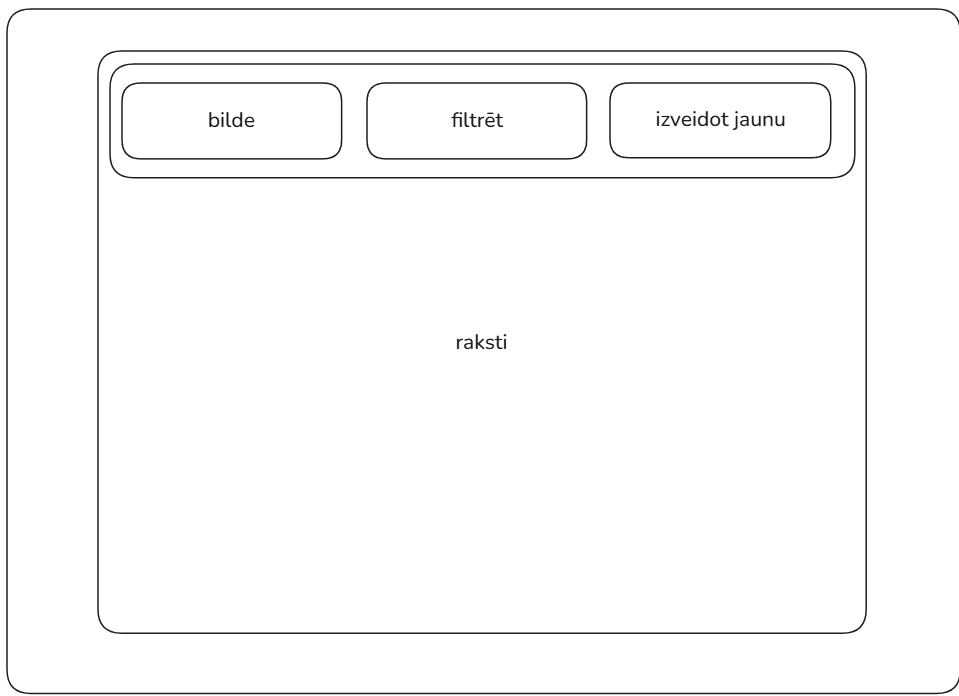
* 1. Ietvari
     1. *CodeIgniter*

Tas ir lietojumprogrammu izstrādes ietvars — rīku komplekts — cilvēkiem, kas veido tīmekļa vietnes. Tā mērķis ir nodrošināt daudz ātrāku projektu izstrādi nekā rakstot kodu no nulles, nodrošinot bagātīgu bibliotēku komplektu bieži nepieciešamajiem uzdevumiem, kā arī vienkāršu saskarni un loģisku struktūru piekļuvei šīm bibliotēkām. [1]

Es izvēlējos *CodeIgniter* šim projektam, jo tas atvieglo tīmekļa lapu veidošanu, samazinot vajadzīgo laiku lai to veidotu, nodrošina lielāku drošību, ātrumu un autentifikāciju.

1. programmatūras prasību specifikācija
   1. Sākotnēja ideja

Pirmajā prototipā man pa lielākam bija viena lapa, kurā var izveidot, redzēt un filtrēt savus rakstus.



6.attēls. **Sākotnējā struktūrskice**

* 1. Nefunkcionālās prasības
     1. Navigācijas josla:

**Logo** – (skatīt 7. attēlu)šeit būs redzama mājas lapas galvenā bilde

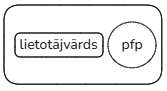
**Vairāk** – (skatīt 7. attēlu) nometams logs kas parādīs citas lapas uz kurām var iet, piemēram, nākotnes plāni, lapas kļūdu reportēšana u.c.

**Kontakti** – (skatīt 7. attēlu) šī poga novedīs uz citu lapu kur būs redzami veidi kā ar mums var kontaktēties



.attēls. **Navigācijas joslas struktūrskice**

**Profils** – (skatīt 8. attēlu) logā būs redzama lietotāja profila bilde un lietotājvārds



.attēls. **Profila loga struktūrskice**

* + 1. Mājas lapa:

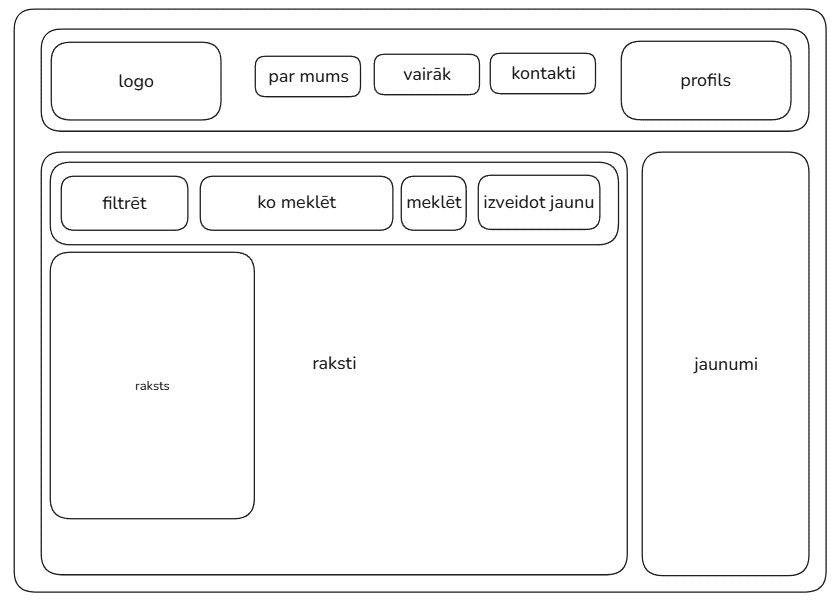
**Logs jaunumi** –atradīsies mājas lapas labajā pusē (skatīt 9. attēlu) un rādīs visu jaunāko, kas tiek mainīts mājas lapā.

**Poga filtrēt** – atradīsies rakstu loga augšējā kreisajā pusē(skatīt 9. attēlu).

**Poga izveidot** **jaunu** – atradīsies rakstu loga augšējā (skatīt 9. attēlu).

**Logs** **ko** **meklēt** – atradīsies rakstu loga augšējā kreisajā pusē(skatīt 9. attēlu).

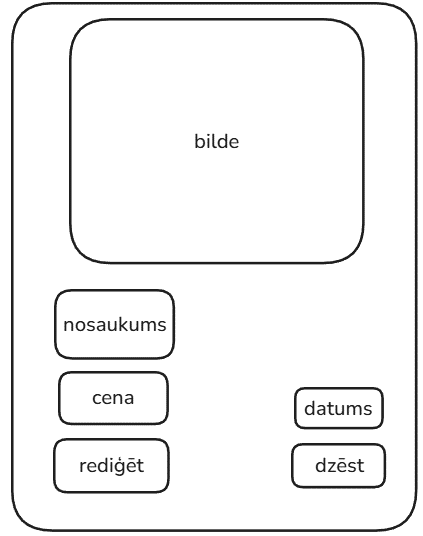
**Logs** **raksti** – atradīsies mājas lapas kreisajā pusē (skatīt 9. attēlu), zem navigācijas joslas un šajā logā būs redzami visi raksti, kurus lietotāji ir izveidojuši



9.attēls. **Mājas-lapas-struktūrskice**

* + 1. Raksta logs:

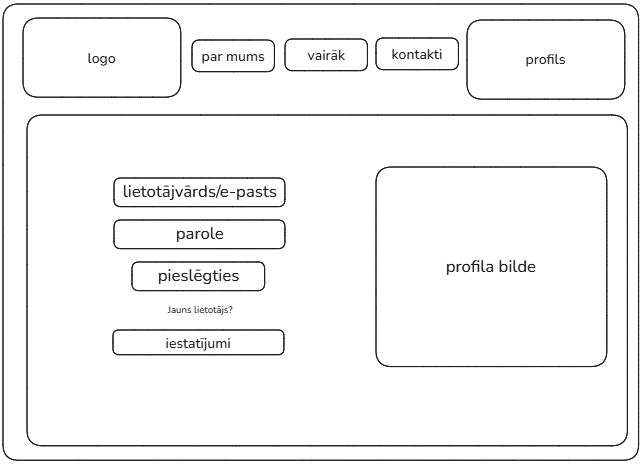
Uz raksta loga (skatīt 10. attēlu) būs redzama bilde, nosaukums, apraksts, datums kad izveidoja, poga “rediģēt”(skatīt 3.4.6. punktu.) un poga “dzēst”. Visi raksti būs sakārtoti viens aiz otra laukā “raksti” (skatīt 9. attēlu).



10.attēls. **Rakstastruktūrskice**

* + 1. Profila Lapa:

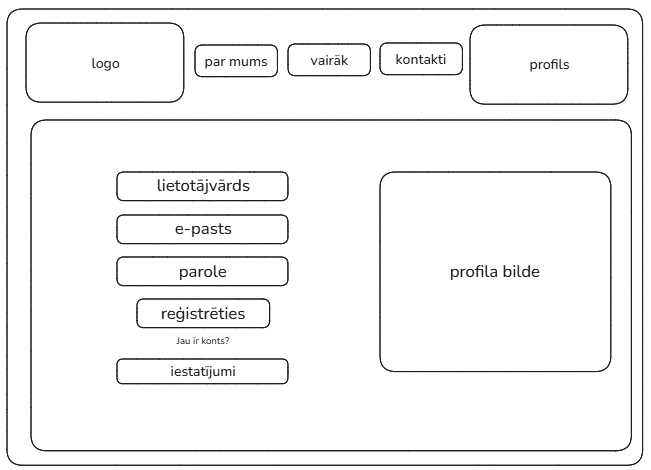
Profila lapā (skatīt 11. attēlu) būs redzams lietotājvārds, e-pasts, parole un profila bilde



11.attēls. **Profila lapas struktūrskice**

* + 1. Konta izveidošanas lapa:

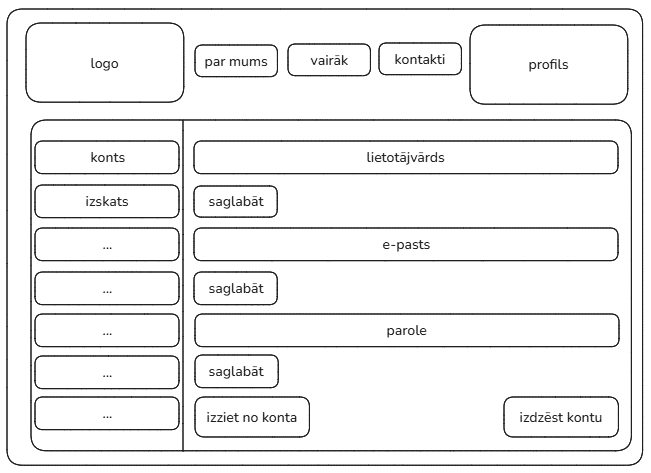
Konta izveidošanas lapā (skatīt 12. attēlu) var izveidot jaunu lietotāja kontu, izmantojot e-pastu, paroli, izveidojot lietotājvārdu un profila bildi.



12.attēls.**Konta izveidošanas loga struktūrskice**

* + 1. Iestatījumu logs:

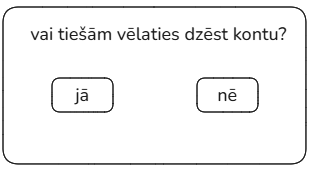
Iestatījumu lapā (skatīt 13. attēlu) būs iespējams mainīt mājas lapas izskatu, savu kontu u.c.



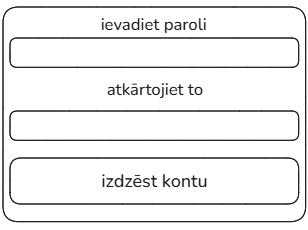
13.attēls. **Konta iestatījumu struktūrskice**

* + 1. Konta izdzēšanas apstiprināšanas logs:

Šajos logos (skatīt 14. attēlu un 15. attēlu) vajadzēs apstiprināt konta izdzēšanu.



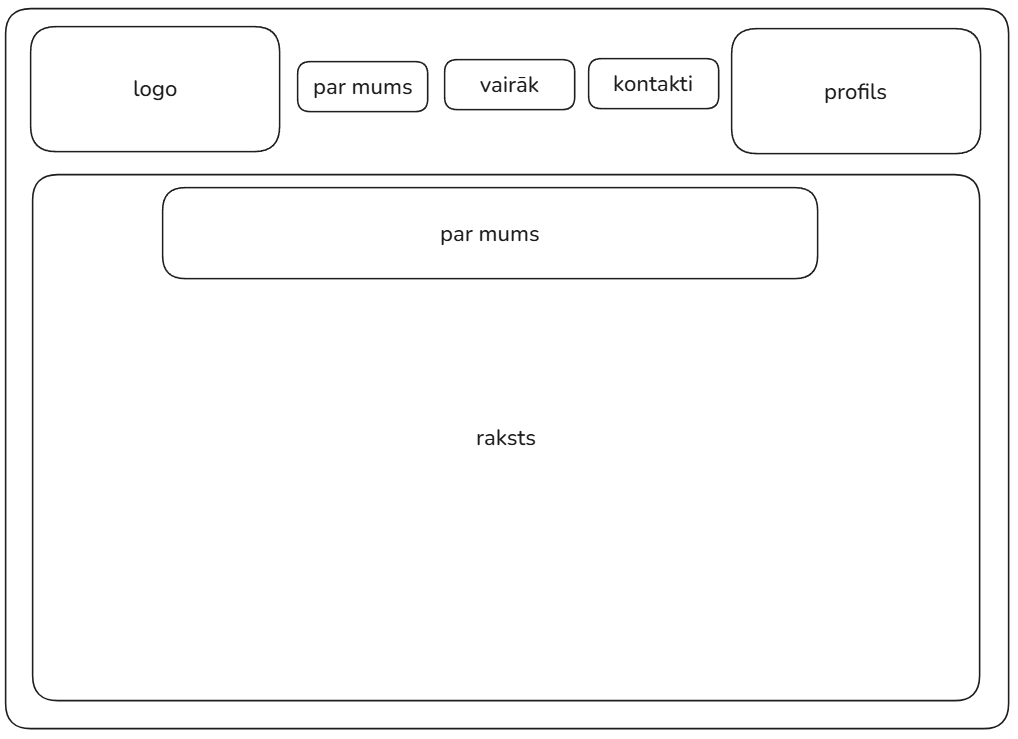
14.attēls. **Konta dzēšanas logs**



15.attēls. **Konta dzēšanas logs**

* + 1. Par mums lapa:

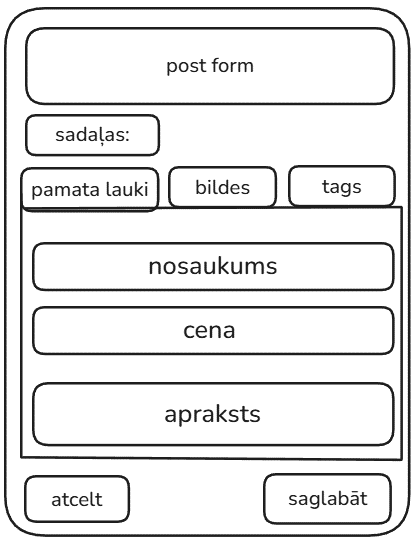
Šajā lapā (skatīt 16. attēlu) būs šī projekta apraksts, kad to sāka veidot, kāds bija progress, kādus rīkus izmantoja, kurš to veidoja u.c.



16.attēls. **Par mums lapas struktūrskice**

* + 1. Rakstu veidošana:
       1. Rakstaformas pamata lauki:

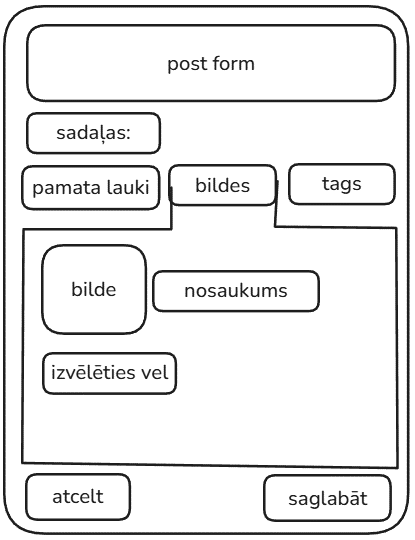
Šajā laukā (skatīt 17. attēlu) lietotājs ievada raksta nosaukumu, kategoriju, aprakstu.



17.attēls. **Raksta formas pamata lauki struktūrskice**

* + - 1. Raksta formas bildes:

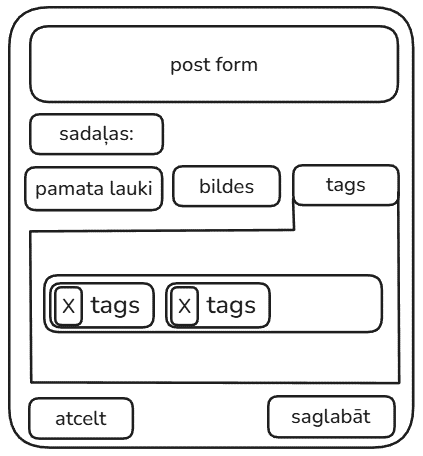
Šajā laukā (skatīt 18. attēlu) lietotājs izvēlas bildes izmantojot failu meklētāju. Pie tām būs redzams bildes nosaukums kuru var mainīt.



18.attēls. **Raksta formas bildes struktūrskice**

* + - 1. Raksta formas *tags*:

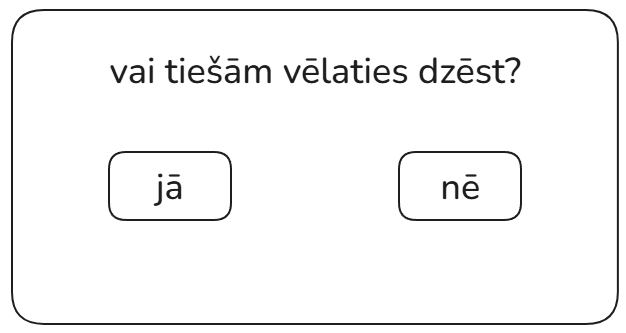
Šajā laukā (skatīt 19. attēlu) lietotājs var pievienot un noņemt *tags* kurus saglabās ar rakstu.



19.attēls. **Raksta formas *tags* struktūrskice**

* + 1. Raksta dzēšanas apstiprinājums:

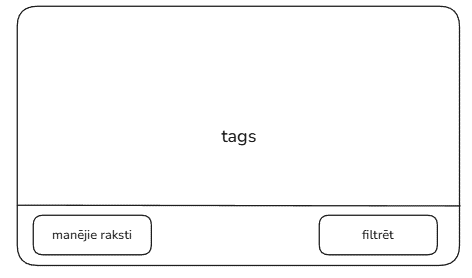
(skatīt 20. attēlu)Lietotājs apstiprina vai atceļ raksta dzēšanu



20.attēls. **Raksta dzēšanas apstiprinājums struktūrskice**

* + 1. Raksta filtrēšanas logs

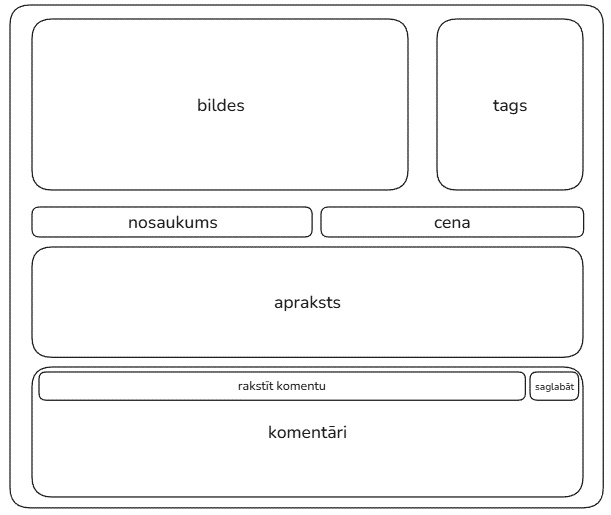
(skatīt 21. attēlu) Šajā logā tiks rādīti visi *tags* kuri ir pieejami, apakšā būs poga “manēji raksti” un poga “filtrēt”.



21.attēls. **Raksta filtrēšanas logs struktūrskice**

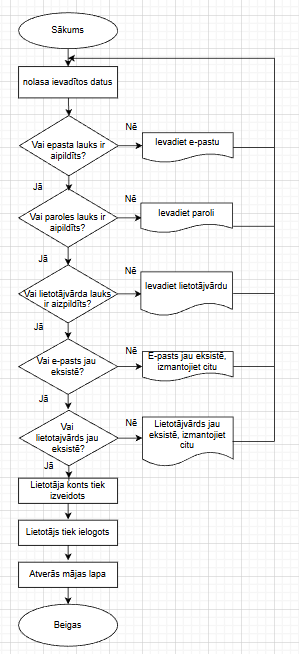
* 1. Raksta detalizēta informācija

(skatīt 22. attēlu) Būs redzama visa informācija par rakst – bildes, *tags*, nosaukums, cena, apraksts un komentāri



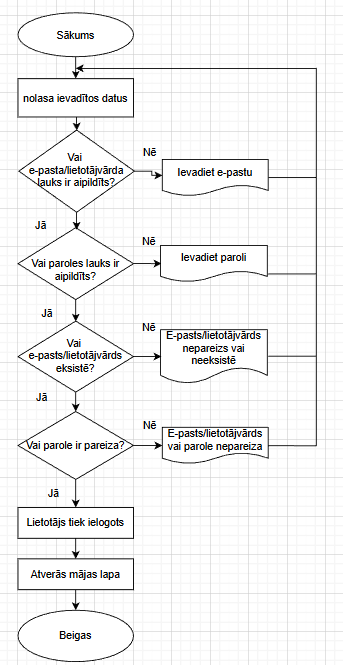
22.attēls. **Raksta detaļas struktūrskice**

* 1. Funkcionālas prasības
     1. Galvenes pogu funkcijas
* (skatīt 9. attēlu) uzspiežot pogu “par mums” tiks atvērta cita mājas lapa (skatīt 16. attēlu). Konts kuru lietotājs izmanto nemainās.
* (skatīt 9. attēlu) Uzspiežot uz lapas logo, atvērsies mājas lapa.
* (skatīt 9. attēlu) Uzspiežot uz pogu “par mums”, atvērsies cita lapa(skatīt 16. attēlu)
  + 1. Konta izveidošana



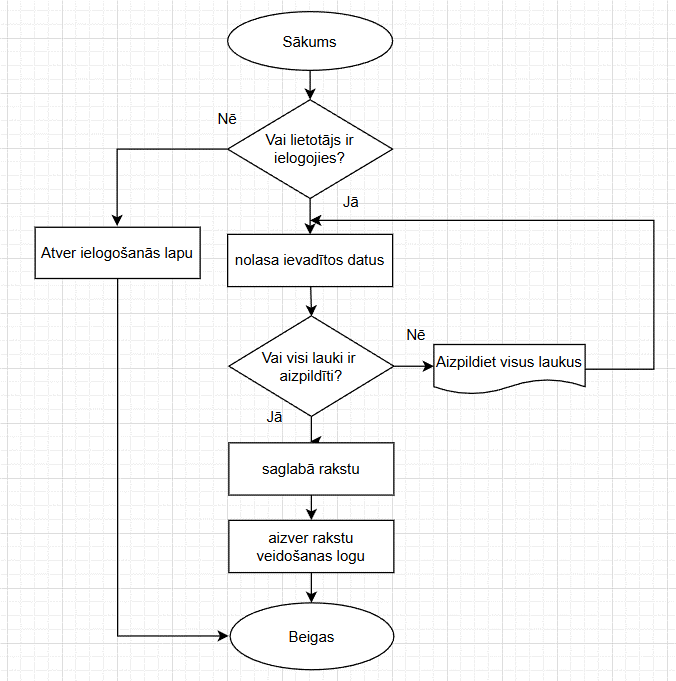
.attēls. **Konta veidošanas blokshēma**

* (skatīt 11. attēlu) uzspiežot pogu “jauns lietotājs?” atvērsies konta izveidošanas logs(skatīt 12. attēlu). Lietotājs ievada lietotājvārdu, e-pastu, paroli(skatīt 23. attēlu). Nospiežot pogu “izveidot”, šie dati tiek šifrēti un sūtīti uz datubāzi, kur tie tiek noformēti un saglabāti.
  + 1. Lietotāja ielogošana



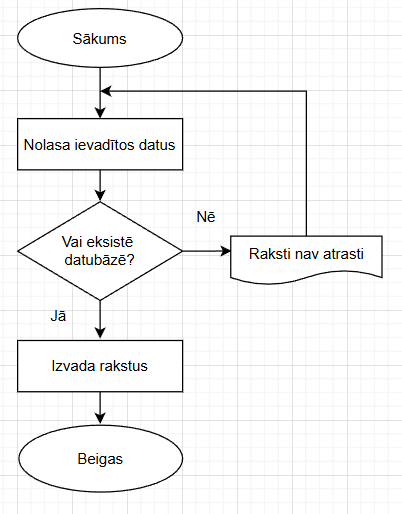
.attēls. **Ielogošanās blokshēma**

* (skatīt 9. attēlu) uzspiežot uz profila atvērsies pieteikšanās logs(skatīt 11. attēlu). šajā logā lietotājs ievada e-pastu/lietotājvārdu, paroli, nospiežot pogu “pieslēgties”, e-pasts\lietotājvārds, parole tiek aizsūtīti uz datubāzi, kur tiek pārbaudīts vai tāds lietotājs eksistē, ja tāds ir, tad lietotājs tiek ielaists savā kontā, ja tāds nav, tad paziņo ka parole vai e-pasts/lietotājvārds ir nepareizs
  + 1. Profila bildes mainīšana
* (skatīt 11. attēlu) uzspiežot uz profila bildes, atvērsies datora failu pārlūks, kur lietotājs var izvēlēties bildi kuru izmantot savā profilā.
  + 1. Raksta veidošana



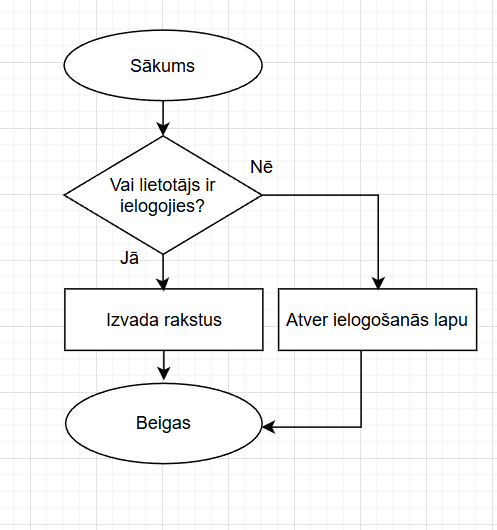
.attēls. **Rakstu veidošanas blokshēma**

* (skatīt 9. attēlu) uzspiežot uz pogas “izveidot jaunu”, atvērsies logs ar vairākām sadaļām, pamata lauki, bildes, *tags* (skatīt 17. attēlu). Pamata lauka nodaļā lietotājs ievada raksta nosaukumu, cenu, aprakstu. Bildes logā var izvēlēties vienu vai vairākas bildes izmantojot failu meklētāju. *Tags* logā var izvedot un izdzēst savus *tags* kurus saglabās ar šo rakstu (skatīt 25. attēlu). Spiežot pogu “atcelt” šis logs aizvērsies, un atverot šo logu atkal neatjauninot lapu, visa ievadītā informācija nepazudīs.
  + 1. Raksta rediģēšana
* (skatīt 10. attēlu) uzspiežot uz pogas “rediģēt”, atvērsies logs ar vairākām formu logiem – nosaukums, bilde, *tags* (skatīt 17. attēlu), katrs lauks būs aizpildīts ar tiem datiem kas šajā rakstā ir saglabāti. Raktu rediģēšanai izmanto kodu lai vajadzīgo informāciju aizsūtītu uz aizmugursistēmu (skatīt 1. pielikumu), kur to apstrādā un saglabā.
  + 1. Raksta dzēšana
* (skatīt 10. attēlu) uzspiežot uz pogas “dzēst”, atvērsies logs(skatīt 20. attēlu). spiežot pogu “nē”, logs aizvērsies un nekas nenotiks, spiežot pogu “jā”, logs aizvērsies un izvēlētais raksts tiks izdzēsts
  + 1. Raksta filtrēšana
* (skatīt 21. attēlu) *tags* logā visi *tags* tiks rādīti kā pogas, uzspiežot (izvēlējoties) to krāsa nomainās uz pelēku, uzspiežot atkal (atceļot izvēli) to krāsa nomainās uz zilu. Uzspiežot pogu “filtrēt”, šis logs aizvērās un visi raksti kas satur visus izvelētos *tags* tiks parādīti logā “raksti” (skatīt 9. un 26. attēlu).



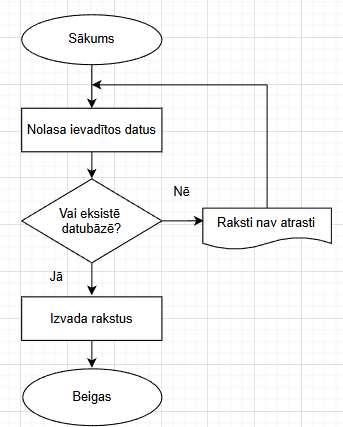
.attēls. **Rakstu filtrēšanas blokshēma**

* (skatīt 21. attēlu) Uzspiežot pogu “mani raksti”, logs aizvērsies un, ja lietotājs ir savā kontā, tad logā “raksti” (skatīt 9. attēlu) parādīs visus rakstus kurus šis lietotājs ir veidojis, ja lietotājs nav savā kontā, tad šis logs aizvērsies un atvērsies profila logs (skatīt 11. un 27. attēlu).



.attēls. **Savu rakstu parādīšanas blokshēma**

* + 1. Raksta meklēšana



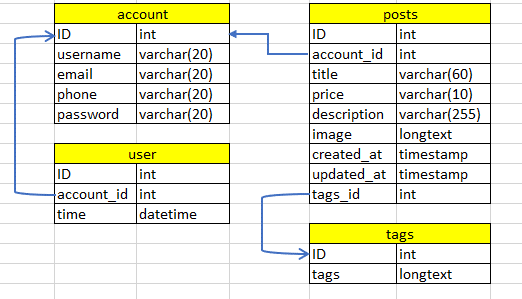
.attēls. **Rakstu meklēšanas blokshēma**

* (skatīt 9. attēlu) Uz loga “ko meklēt” var uzspiest un ierakstīt raksta nosaukumu, un uzspiežot pogu “meklēt”, visi raksti pazudīs no loga “raksti”, un parādīsies tie raksti, kuru nosaukums satur to, kas ir ierakstīts logā “ko meklēt”.
  + 1. Raksta informācija
* Uzspiežot uz pašu rakstu(skatīt 9. attēlu), atvērsies lapa, kas saturēs raksta detalizētu informāciju(skatīt 22. attēlu).
  1. Sistēmas struktūras modelis
     1. Datubāzes apraksts

Datubāzes nosaukums ir “norokas\_db”, to pārvalda ar rīku “*HeidiSQL*” un tās dzinējs ir “*MySQL*”.

Šajā datubāzē atrodas 4 tabulas:

* “account”(skatīt 1. tabula)glabā informāciju par izveidotajiem kontiem
* “user”(skatīt 2. tabula), glabā informāciju par lietotājiem, kas pašlaik ir ielogojušies
* ”posts”(skatīt 3. tabula), glabā informāciju par veidotajiem rakstiem
* “tags”(skatīt 4. tabula), glabā katra raksta *tags*



.attēls. **Datubāzes struktūras diagramma**

Noklusējuma rakstzīmju kopa datubāzē ir *utf8mb4*.

* + 1. Tabula “account”

Šī tabula (skatīt 1. tabula tabulu) glabā informāciju par sistēmas lietotāju kontiem. Tā satur lietotāju unikālo identifikatoru, lietotājvārdu, e-pastu, mobilo numuru un paroli.

1. tabula **“account”**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nosaukums** | **Tips** | **Atribūti** | **Komentāri** |
| ID | int | NOT NULL, PRIMARY KEY, AUTO INCREMENT | Unikāls identifikators kontam |
| username | varchar(20) | NOT NULL | Lietotāja lietotājvārds |
| email | varchar(20) | NOT NULL | Lietotāja e-pasta adrese |
| phone | varchar(20) | NOT NULL | Lietotāja mobilais numurs |
| password | varchar(20) | NOT NULL | Šifrēta lietotāja parole |

* + 1. Tabula “user”

Šī tabula (skatīt 2. tabula tabula) glabā informāciju par lietotājiem, kas pašlaik ir ielogojušies un izmanto saiti. Tā satur lietotāja *ID* un laiku, kad lietotājs ielogojās, kas tiek atjaunināts pēc katras lietotāja darbības(atver citu lapu vai logu, atjaunina lapu u.c.)

2. tabula **“user”**

|  |  |  |  |
| --- | --- | --- | --- |
| Nosaukums | Tips | Atribūti | Komentāri |
| ID | Int | NOT NULL, PRIMARY KEY, AUTO INCREMENT | Unikāls identifikators lietotājam |
| account\_id | Int | NOT NULL | Lietotāja konta ID |
| time | datetime | NOT NULL | Laiks, kad lietotājas ielogojās |

* + 1. Tabula “posts”

Šī tabula (skatīt 3. tabula tabulu) glabā visu informāciju par pašu rakstu. Tā satur veidotāja konta *ID*, nosaukums, cena, apraksts, bilde/es, laiks kad izveidoja un pēdējo reizi izmainīja, un *tags ID*, ar ko var atrast raksta *tags* no citas tabulas (skatīt 3. tabula tabulu).

3. tabula **“post”**

|  |  |  |  |
| --- | --- | --- | --- |
| Nosaukums | Tips | Atribūti | Komentāri |
| ID | int | NOT NULL, PRIMARY KEY, AUTO INCREMENT | Unikāls identifikators rakstam |
| account\_id | int | NOT NULL | Lietotāja konta ID |
| title | varchar(60) | NOT NULL | Raksta nosaukums |
| cena | varchar(10) | NOT NULL | Raksta cena |
| description | varchar(255) | NOT NULL | Raksta apraksts |
| image | longtext | NOT NULL | Saraksts ar raksta bildēm |
| created\_at | timestamp | NOT NULL | Datums un laiks, kad rakstu izveidoja |
| updated\_at | timestamp | NOT NULL | Datums un laiks, kad rakstu pēdējo reizi izmainīja |
| tags\_id | int | NOT NULL | Saraksts ar *tag* ID, kurus šis raksts izmanto |

* + 1. Tabula “tags”

Šī tabula (skatīt 4. tabula tabulu) glabā visus *tags* kas ir veidoti. Katram *tag* ir savs unikāls *ID*.

4. tabula **“tags”**

|  |  |  |  |
| --- | --- | --- | --- |
| Nosaukumi | Tips | Atribūti | Komentāri |
| ID | int | NOT NULL, PRIMARY KEY, AUTO INCREMENT | Unikāls identifikators priekš *tag* |
| tags | longtext | NOT NULL | *Tags* kuru raksts izmantos |

# Secinājumi

1. Mājas lapas daļēji izveidota
2. Var veidot rakstus, tos rediģēt un dzēst
3. Ir iespējams filtrēt visus rakstus izmantojot *tags*
4. Uzspiežot uz rakstu atvērās logs kas rāda detalizētu informāciju par to rakstu

# Izmantoti avoti

1. codeignitor.com [Tiešsaiste] Pieejams: <https://codeigniter.com/user_guide/intro/index.html> [skatīts 18.09.2025]
2. developer.mozilla.org [Tiešsaiste] Pieejams: <https://developer.mozilla.org/en-US/docs/Web/API/Document_Object_Model> [skatīts 18.09.2025]
3. facebook.com [Tiešsaiste] Pieejams: <https://www.facebook.com/marketplace/> [skatīts 06.01.2025]
4. heidisql.com [Tiešsaiste] Pieejams: <https://www.heidisql.com/> [skatīts 13.01.2025
5. infoworld.com [Tiešsaiste] Pieejams: <https://www.infoworld.com/article/2335960/what-is-visual-studio-code-microsofts-extensible-code-editor.html> [skatīts 14.01.2025]
6. ne.wikipedia.org [Tiešsaiste] Pieejams: <https://en.wikipedia.org/wiki/PHP> [skatīts 13.01.2025]
7. ne.wikipedia.org [Tiešsaiste] Pieejams: <https://en.wikipedia.org/wiki/HTML> [skatīts 13.01.2025]
8. ne.wikipedia.org [Tiešsaiste] Pieejams: <https://en.wikipedia.org/wiki/JavaScript> [skatīts 13.01.2025]
9. ne.wikipedia.org [Tiešsaiste] Pieejams: <https://en.wikipedia.org/wiki/CSS> [skatīts 13.01.2025]
10. ne.wikipedia.org [Tiešsaiste] Pieejams: <https://en.wikipedia.org/wiki/Post> [skatīts 14.01.2025]
11. ne.wikipedia.org [Tiešsaiste] Pieejams: <https://en.wikipedia.org/wiki/Tag_(metadata)> [skatīts 14.01.2025]
12. ne.wikipedia.org [Tiešsaiste] Pieejams: <https://en.wikipedia.org/wiki/JavaScript> [skatīts 18.09.2025]
13. ne.wikipedia.org [Tiešsaiste] Pieejams: <https://en.wikipedia.org/wiki/MySQL> [skatīts 18.09.2025]
14. ne.wikipedia.org [Tiešsaiste] Pieejams: <https://en.wikipedia.org/wiki/API> [skatīts 18.09.2025]
15. ne.wikipedia.org [Tiešsaiste] Pieejams: <https://en.wikipedia.org/wiki/CSS> [skatīts 18.09.2025]
16. ptmbeles.lv [Tiešsaiste] Pieejams: <https://www.ptmebeles.lv/katalogs/> [skatīts 06.01.2025]
17. ss.com [Tiešsaiste] Pieejams: [ss.com](https://www.ss.com/) [skatīts 06.01.2025]
18. utrupe.lv [Tiešsaiste] Pieejams: <https://www.utrupe.lv/> [skatīts 06.01.2025]
19. zviedrumebeles.lv [Tiešsaiste] Pieejams: <https://zviedrumebeles.lv/> [skatīts 06.01.2025]

# pielikumi

1. Pielikums. ***index.php***

$(document).delegate('.post\_edit\_btn', 'click', function(e) {

e.preventDefault();

const id = $(this).attr('id');

$.ajax({

url: '<?= base\_url(' post/edit/') ?>/' + id,

method: 'get',

success: function(response) {

imagename = JSON.parse(response.message.image, true);

tagname = JSON.parse(response.message.tags, true);

$("#pid").val(response.message.id);

$("#title").val(response.message.title);

$("#category").val(response.message.category);

$("#body").val(response.message.body);

showOldFiles(imagename);

}

});

});