

Vidzemes Tehnoloģiju un dizaina tehnikums

Profesionālās vidējās izglītības programma – Programmēšana

Kvalifikācija – Programmēšanas tehnīkis

Izglītības ieguves forma – Klātiene

Kvalifikācijas darba praktiskās daļas dokumentācija

Tīmekļa vietne *BlogIt*

Autors: IP20 grupas audzēknis

Matīss Bālinš

(paraksts)

ANOTĀCIJA

Bloga tīmekļa vietne *BlogIt*, Bālinš, Matīss. Kvalifikācijas darbs, 89 lapaspuses, 130 attēli, 10 tabulas, 22 literatūras avoti, 2 pielikumi.

Kvalifikācijas darba mērķis ir izstrādāt modernu, viegli lietojamu un parocīgu bloga tīmekļa vietni, kurā lietotājs spēs dalīties ar savu radošo darbību, dzīves notikumiem un viņam interesējošam darbībām, kā arī apskatīt un iedvesmoties no citiem tīmekļa vietnes ievietotiem blogiem. Vietne veicina lasītāju iesaistīšanos un mijiedarbību, nodrošinot dažādas iespējas, dalīties ar domām, publicēt, komentēt rakstus, kā arī saglabāt un pārvaldīt tos. Darbā ir detalizēti aprakstītas prasības, kuras tika uzstādītas, lai nodrošinātu vietnes veiksmīgu izstrādi. Literatūras apskatā un variantu salīdzinājumā tiek analizēti citi populāri blogi, lai identificētu veiksmīgus elementus un iegūtu iedvesmu izstrādei.

Prasību specifikācijā tiek aprakstītas prasības mājaslapas darbībai gan funkcionālās, gan nefunkcionālās. Funkcionālais un dinamiskais sistēmu modelis attēlo vietnes lietotāja pieredzi, kā arī dati, kas tiek pārnesti vietnes ietvaros. Lietotāja ceļvedis sniedz skaidru priekšstatu par to, kā izmantot tīmekļa vietni, tā lai to spētu ērti lietot jebkura vecuma cilvēki, veicot pieredzi intuitīvu un baudāmu.

Darba mērķis tika sasniegts, un rezultātā radās *BlogIt* - tīmekļa vietne, kas piedāvā lasītājiem ievietot un apskatīt iedvesmojošu un daudzveidīgu saturu. Vietnes interaktīvās funkcijas un iespējas komentēt un dalīties ar rakstiem veicina aktīvu lasītāju kopienas veidošanos, piedāvājot viņiem pilnvērtīgu tīmekļa vietnes izmantošanas pieredzi.

Saite uz tīmekļa vietni : <https://into.id.lv/>

Saite uz priekšgalsistēmas *GitHub repository* : <https://github.com/Matiss1B/blog-react>

Saite uz aizmugursistēmas *GitHub repository* : <https://github.com/Matiss1B/blog>

ANNOTATION

Blog website *BlogIt*. Balins, Matiss. Qualification thesis, 89 pages, 130 images, 10 tables, 22 literature sources, 2 appendices.

The Qualification job aims to develop a modern, easy-to-use and handy blog website where the user will be able to share their creative activity, life events and activities of interest to them, and to view and inspire from other blogs posted on the website. The site encourages readers to engage and interact by providing a variety of options, sharing thoughts, publishing, commenting on articles, and saving and managing them. The work details the requirements that were set up to ensure the successful development of the site. The literary Review and variations comparison analyzes other popular blogs to identify successful elements and get inspiration for development.

The requirements specification describes the requirements for the operation of the website, both functional and non-functional. The functional and dynamic system model represents the user experience of the site, as well as the data that is transferred within the site. The user's guide gives you a clear idea of how to use the website so that people of any age can easily use it, making experiences intuitive and enjoyable.

The work's goal was achieved and the result was *BlogIt* - a website that offers readers to insert and view inspirational and diverse content. The site's interactive features and opportunities to comment and share articles contribute to building an active reader Community, offering them a full experience of using the website.

Link to website : <https://into.id.lv/>

Link to frontend *GitHub repository* : <https://github.com/Matiss1B/blog-react>

Link to backend *GitHub repository* : <https://github.com/Matiss1B/blog>

SATURA RĀDĪTĀJS

| | |
|---|----|
| Anotācija..... | 2 |
| Annotation | 3 |
| Satura rādītājs | 4 |
| Definīcijas un akronīmi | 6 |
| Ievads..... | 8 |
| 1. Literatūras apskats un variantu salīdzinājums | 9 |
| 1.1. Tīmekļa vietne <i>Mashable</i> | 9 |
| 1.2. Tīmekļa vietne <i>Web Dev Simplified Blog</i> | 11 |
| 1.3. Tīmekļa vietņu funkciju salīdzinājums..... | 13 |
| 2. Prasību specifikācija | 14 |
| 2.1. Funkcionālās prasības..... | 14 |
| 2.2. Nefunkcionālās prasības | 28 |
| 2.2.1. Reaģējoša tīmekļa vietne | 28 |
| 2.2.2. Datu drošība un paroles | 29 |
| 2.2.3. Pieejamība | 30 |
| 2.2.4. Veikspēja..... | 30 |
| 2.2.5. Uzturamība | 30 |
| 2.2.6. Dizains un izkārtojums | 31 |
| 3. Mērķa sasniegšanas līdzekļu izvēle un pamatojums | 32 |
| 4. Sistēmas un datu struktūras modelis..... | 34 |
| 4.1. ER diagramma | 34 |
| 4.2. Komponenšu diagrammas | 35 |
| 4.2.1. Datubāzes komponenšu diagramma | 35 |
| 4.2.2. Priekšgalsistēmas komponenšu diagramma | 36 |
| 5. Funkcionālais un dinamiskais sistēmas modelis | 37 |
| 5.1. Lietošanas gadījumu diagramma | 37 |
| 5.2. Sekvences diagrammas..... | 38 |
| 5.2.1. Tīmekļa vietnes atvēršana | 38 |
| 5.2.2. Pielēgšanās tīmekļa vietnei..... | 39 |
| 5.2.3. Reģistrācija tīmekļa vietnē | 39 |
| 5.3. Rekomendāciju algoritms | 40 |
| 6. Testēšana..... | 41 |
| 6.1. Manuālā testēšana..... | 41 |
| 6.1.1. Pieslēgšanās | 41 |

| | | |
|--------|--|----|
| 6.1.2. | Reģistrācija | 42 |
| 6.1.3. | Bloga pievienošana..... | 43 |
| 6.1.4. | Bloga Redīgēšana | 44 |
| 6.1.5. | Profila Redīgēšana..... | 46 |
| 6.2. | Automātiskā testēšana | 47 |
| 6.2.1. | Pieslēgšanās testi | 48 |
| 6.2.2. | Reģistrācijas testi..... | 48 |
| 6.2.3. | Bloga funkciju testi..... | 49 |
| 6.2.4. | Lietotājkonta funkciju testi..... | 50 |
| 6.2.5. | Komentāru funkciju testi | 50 |
| 6.2.6. | Sekotāju funkciju testi | 51 |
| 6.3. | Darbības testēšana pārlūkprogrammās | 51 |
| 7. | Lietotāja ceļvedis..... | 52 |
| 7.1. | Tīmekļa vietnes atvēršana | 52 |
| 7.2. | Pieslēgšanās ar lietotāja kontu..... | 53 |
| 7.3. | Lietotāja konta reģistrācija | 54 |
| 7.4. | Navigācijas joslas izmantošana | 57 |
| 7.5. | Blogu saraksta apskate..... | 61 |
| 7.6. | Komentāru ievade..... | 63 |
| 7.7. | Manu blogu saraksta apskate | 64 |
| 7.8. | Blogu meklēšanas iespēja..... | 65 |
| 7.9. | Bloga pievienošana..... | 67 |
| 7.10. | Blogu redīgēšanas iespēja..... | 70 |
| 7.11. | Lietotāja Profila apskate | 74 |
| 7.12. | Piesekošana un atsekošana lietotājam | 76 |
| 7.13. | Sekotāju un profilu kam seko lietotājs sarakstu pārvaldība | 77 |
| 7.14. | Profila redīgēšanas iespēja..... | 79 |
| 7.15. | Akcenta krāsas maiņa | 81 |
| 7.16. | Paroles maiņas iespēja..... | 81 |
| | Secinājumi un priekšlikumi | 85 |
| | Izmantotā literatūra..... | 87 |
| | Pielikumi..... | 89 |

DEFINĪCIJAS UN AKRONĪMI

| Jēdziens | Skaidrojums |
|---|--|
| <i>API</i> | API ir mehānismi, kas ļauj diviem programmatūras komponentiem sazināties vienam ar otru, izmantojot definīciju un protokolu kopu (aws, 2024). |
| Aizmugursistēma (angļu v. <i>back-end</i>) | Datoru pasaule <i>back-end</i> attiecas uz jebkuru vietnes vai programmatūras programmas daļu, ko lietotāji nerēdz. Tas atšķiras no priekšgala , kas attiecas uz programmas vai vietnes lietotāja interfeisu. Programmēšanas terminoloģijā aizmugursistēma ir datu piekļuves slānis (TechTerms, 2024a). |
| Priekšgalsistēma (angļu v. <i>front-end</i>) | Programmatūras programmas vai vietnes priekšgals ir viss, ar ko lietotājs mijiedarbojas. No lietotāja viedokļa priekšgals ir lietotāja interfeisa sinonīms . No izstrādātāja viedokļa saskarnes funkcijas nodrošina interfeisa dizains un programmēšana (TechTerms, 2024b). |
| Starpvietņu skriptēšana (angļu v. <i>Cross-Site Scripting</i>) | Cross-Site Scripting (XSS) uzbrukumi ir injekcijas veids, kurā ļaunprātīgi skripti tiek ievadīti citādi labdabīgās un uzticamās vietnēs. XSS uzbrukumi notiek, kad uzbrucējs izmanto tīmekļa lietojumprogrammu, lai nosūtītu ļaunprātīgu kodu, parasti pārlūkprogrammas puses skripta veidā, citam galalietotājam (KirstenS, 2024). |
| SQL injekcija (angļu v. <i>SQL injection</i>) | SQL injekcija, kas dažkārt tiek saīsināta uz SQLi, ir ievainojamības veids, kurā uzbrucējs izmanto SQL (strukturētās vaicājumu valodas) koda daļu, lai manipulētu ar datu bāzi un piekļūtu potenciāli vērtīgai informācijai. Tas ir viens no visizplatītākajiem un draudīgākajiem uzbrukumu veidiem, jo to potenciāli var izmantot pret jebkuru tīmekļa lietojumprogrammu vai vietni, kas izmanto uz SQL balstītu datu bāzi (kas ir lielākā daļa no tiem) (Kaspersky, 2024). |
| Rekomendāciju algoritms (angļu v. <i>Recommendation Algorithm</i>) | Ieteikuma algoritms ir datu kopas un mašīnmācīšanās kombinācija, kas sniedz personalizētus ieteikumus, pamatojoties uz lietotāju vēsturiskajiem datiem (Jamie Beatty, 2024). |
| Mākslīgais intelekts (angļu v. <i>Artificial</i> | Mākslīgais intelekts jeb AI ir tehnoloģija, kas ļauj datoriem un mašīnām simulēt cilvēka intelektu un problēmu risināšanas spējas. Kā datorzinātnu joma mākslīgais intelekts ietver un bieži tiek minēts kopā ar |

| | |
|--|--|
| <i>Intelligence</i> jeb <i>AI</i>) | mašīnmācīšanos un dziļo mācīšanos . Šīs disciplīnas ietver mākslīgā intelekta algoritmu izstrādi, kas modelēti pēc cilvēka smadzeņu lēmumu pieņemšanas procesiem, kas var “mācīties” no pieejamajiem datiem un laika gaitā veikt arvien precīzākas klasifikācijas vai prognozes (IBM, 2024). |
| Authentifikācijas žetons (angļu v. <i>Authentication token</i>) | Autentifikācijas žetons ļauj interneta lietotājiem piekļūt lietojumprogrammām, pakalpojumiem, vietnēm un lietojumprogrammu saskarnēm (API), neievadot savus pieteikšanās akreditācijas datus katru reizi, kad viņi apmeklē. (Fortinet, 2024). |

IEVADS

Emuārs (turpmāk **Blogs**) ir populārs izpausmes veids internetā. Bloga tīmekļa vietnes šobrīd ir viens no izplatītākajiem savas individuālas darbības izpausmes veidiem, kurā lietotāji var ievietot ierakstus un lasīt citu cilvēku ierakstus. Vadoties pēc 2023. gada statistikas, pasaulei ir aptuveni 1,9 miljardi tīmekļa vietnes un vairāk kā 600 miljoni no tām ir tieši bloga tīmekļa vietnes, kā arī aptuveni 73% interneta lietotāji lasa blogus regulāri (Ong, Si Quan, 2023).

Izveidojot bloga mājaslapu, atklājas bezgalīgas iespējas dalīties ar saviem viedokļiem, zināšanām un pieredzi tiešsaistē. Šāda veida platforma nodrošina iespēju piesaistīt auditoriju, radīt kopienas un veicināt dialogu. Blogs sniedz iespēju ne tikai izpausties, bet arī dalīties ar citiem savā ekspertīzē un dzīves ceļojumos.

Šodien liela daļa bloga tīmekļa vietnes ir vienmuļas un vienveidīgas, bieži tīmekļa vietnēs izkārtojums un krāsu salikums ir nepievilcīgs, kas iespējams atraida lietotāju, jo tīmekļa vietnē ir nepievilcīgi krāsu toni, tādēļ es savu bloga tīmekļa vietni vēlos padarīt unikālu un ar iespēju lietotājam personalizēt to pēc savas gaumes. Lietotājam būs iespēja mainīt akcenta krāsu, tādejādi tiek izveidota saikne ar lietotāju, kura uzlabo lietošanas pieredzi un padara to komfortablu.

Mērķis

Kvalifikācijas darba mērķis ir izstrādāt modernu, viegli lietojamu un parocīgu bloga tīmekļa vietni, kurā lietotājs spēs dalīties ar savu radošo darbību, dzīves notikumiem un viņam interesējošam darbībām, kā arī apskatīt un iedvesmoties no citiem tīmekļa vietnē ievietotiem blogiem.

Uzdevumi

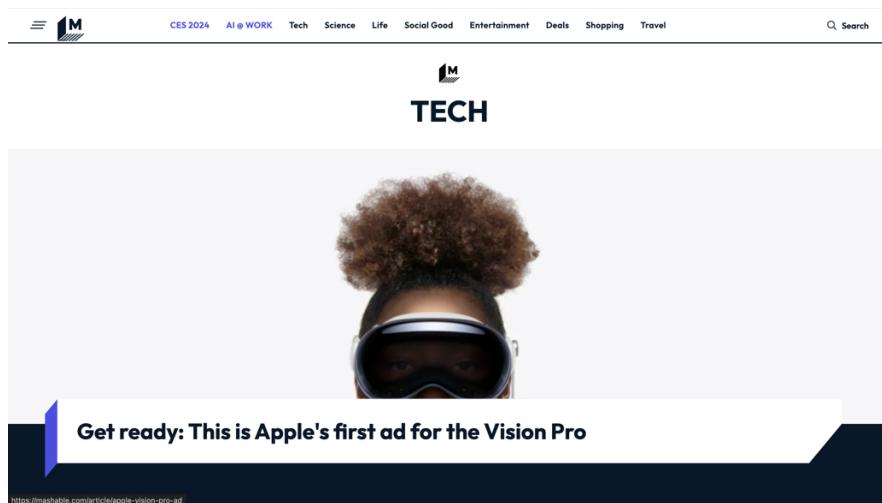
1. Izveidot plānu uzdevumu veikšanai.
2. Izstrādāt intuitīvu, gaumīgu un viegli saprotamu priekšgalsistēmu (angļu v. *front-end*).
3. Izstrādāt datubāzi un tās arhitektūru.
4. Izstrādāt aizmugursistēmu (angļu v. *back-end*), kas nodrošina visu funkcionalitāti attiecīgi pēc izvirzītajām prasībām.
5. Izstrādāt testus, veikt aptauju un gūt atsauces no lietotājiem, veikt tīmekļa vietnes testēšanu.
6. Novērst testa laikā radušās nepilnības un pilnveidot tīmekļa vietni.

1. LITERATŪRAS APSKATS UN VARIANTU SALĪDZINĀJUMS

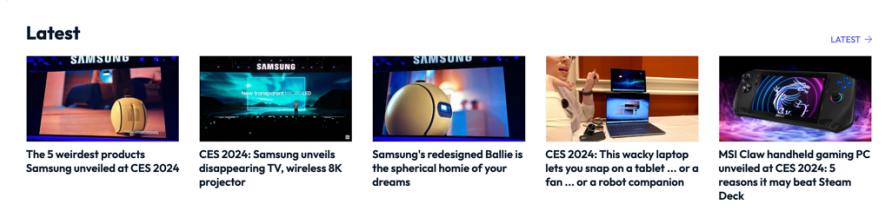
Literatūras apskats un variantu salīdzinājums ir nepieciešams, lai identificētu alternatīvas tīmekļa vietnes un ar analīzes palīdzību atrastu nepilnības un priekšrocības. Šajā nodaļā tiek pētītas esošas tīmekļa vietnes un veikti secinājumi, lai atrastu labāko mājaslapas veidošanas risinājumu.

1.1. TĪMEKLĀ VIETNE *MASHABLE*

Pirmā alternatīva ir tīmekļa vietne *Mashable* apskatīt šeit: <https://mashable.com/> Šajā vietnē ir iespējams apskatīt blogu kategorijas (skatīt 1. attēlā), kā arī apskatīt blogu sarakstu pēc attiecīgās kategorijas (skatīt 2. attēlā), protams, ir iespēja arī lasīt un apskatīt izvēlēto blogu (skatīt 3. attēlā).



1. attēls. *Mashable* tīmekļa vietnes sākumlapa



2. attēls. *Mashable* tīmekļa vietnes blogu saraksts

1. Ballie: Your 'true AI companion'



Credit: Kimberly Gedeon / Mashable

Ballie was first introduced at CES 2020, but now, Samsung has upgraded the rolling AI bot with advanced intelligence. It's much larger now with a sleeker design. Here is, in part, what it can do.

- Be your robot AI companion
- Come to you if you beckon it

3. attēls. *Mashable* tīmekļa vietnes bloga apskate

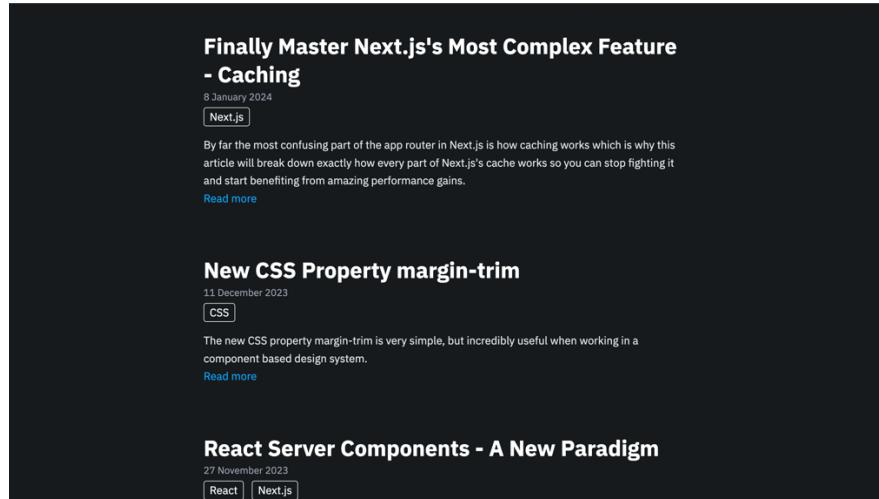
Tīmekļa vietnes *Mashable* izpētes apkopojums

Tīmekļa vietne *Mashable* ir viegli lietojama un moderna ar iespējām apskatīt visdažādāko kategoriju blogus. Tomēr tika konstatēti vairāki trūkumi, ko nepieciešams uzlabot:

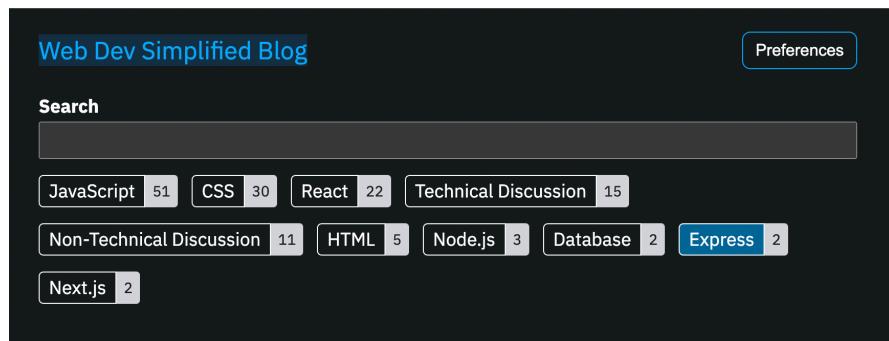
1. autorizācijas trūkums;
2. pievienošanas funkcijas trūkums;
3. komentēšanas funkcijas trūkums;
4. šķirstot tīmekļa vietni uz leju, pazīd navigācijas josla, kas ir ļoti neērti;
5. neizprotama izkārtojuma maiņa blogu saraksta sekcijā.

1.2. TĪMEKĻA VIETNE *WEB DEV SIMPLIFIED BLOG*

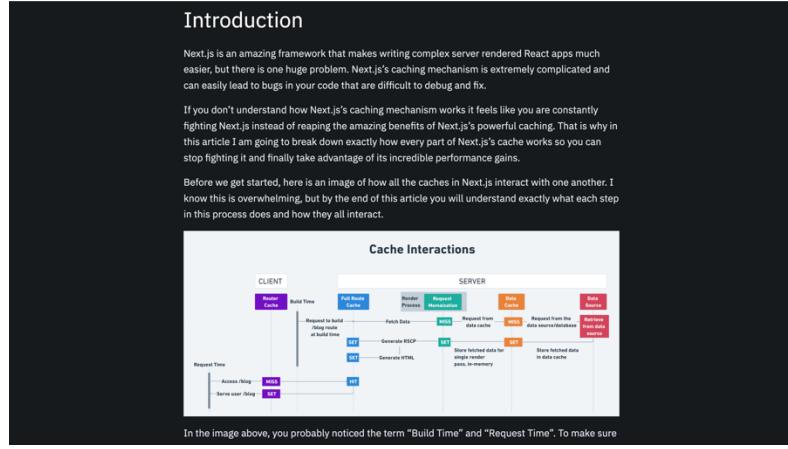
Otrā alternatīva ir tīmekļa vietne *Web Dev Simplified Blog*, apskatīt šeit: <https://blog.webdevsimplified.com/>. Šī vietne balstās tieši uz programmēšanas nozari, tīmekļa vietnē ir iespēja apskatīt kopējo blogu sarakstu (skatīt 4. attēlā), iespēja meklēt blogus un filtrētos pēc kategorijām (skatīt 5. attēlā), kā arī tīmekļa vietne sniedz iespēju izlasīt un apskatīt izvēlēto blogu (skatīt 6. attēlā).



4. attēls. *Web Dev Simplified Blog* tīmekļa vietnes blogu saraksts



5. attēls. *Web Dev Simplified Blog* tīmekļa vietnes filtrācijas sadaļa



6. attēls. *Web Dev Simplified Blog* tīmekļa bloga sadaļa

Tīmekļa vietnes *Web Dev Simplified Blog* izpētes apkopojums

Tīmekļa vietne *Web Dev Simplified Blog* ir moderna, intuitīva tīmekļa ar vairākām dizaina priekšrocībām. Vietnes izkārtojums ir ļoti ērts uz izprotams, patīkami krāsu toni, liela priekšrocība ir tumša un gaiša motīva maiņa, kas mūsdienās kļūst par standartu, bet ne visām tīmekļa vietnēm ir šāda funkcija. Šī vietne ir piemērota individuāla satura veidošanai, bet ne kā blogs, kurā saturu var ievietot ikviens, līdz ar to tīmekļa vietne saskaras ar šādiem trūkumiem:

1. autorizācijas trūkums;
2. pievienošanas funkcijas trūkums;
3. komentēšanas funkcijas trūkums;
4. šķirstot tīmekļa vietni uz leju, pazūd navigācijas josla, kas ir ļoti neērti;
5. nepārskatāma blogu saraksta sadaļa;
6. galvenajā sadaļā, sākotnēji tiek atspoguļots bloga nosaukums un apraksts, bet netiek atspoguļots neviens attēls.

Apskatot šo tīmekļa vietni tika konstatētas vairākas priekšrocības:

1. tumšā motīva iestatīšanas iespēja;
2. filtrācijas iespēja (piemēram, spēja apskatīt vairākas kategorijas vienlaicīgi);
3. moderns dizains.

1.3. TĪMEKĻA VIETŅU FUNKCIJU SALĪDZINĀJUMS

Salīdzinājums tiek veikts, lai apskatītu kuras no konkrētajām funkcijām ir pieejamas katrā tīmekļa vietnē. Ar (+) tiek atzīmēts, ja minētā funkcija šajā tīmekļa vietnē ir pieejama, ar (-) tiek atzīmēts, ja minētā funkcija šajā tīmekļa vietnē nav pieejama (skatīt 1. tabulā).

1. tabula

Tīmekļa vietņu funkcionalitātes salīdzinājums

| Funkcijas | <i>Mashable</i> | <i>Web Dev Simplified</i> | <i>BlogIt</i> |
|---|-----------------|---------------------------|---------------|
| Autorizēšanās | - | - | + |
| Blogu meklēšana | + | + | + |
| Blogu sīkāka apskate | + | + | + |
| Komentēšana | - | - | + |
| Blogu saglabāšana | - | - | + |
| Blogu ievietošana | - | - | + |
| Konfigurēt tīmekļa vietnes krāsu paleti | - | - | + |

2. PRASĪBU SPECIFIKĀCIJA

Viena no galvenajām projekta izstrādes sadaļām ir prasību specifikācija, tā sastāv no funkcionālajām un nefunkcionālajām prasībām. Ar tās palīdzību sistēmas izstrādātājs, balstoties uz ieinteresētās puses definētajām prasībām, spēj izveidot sistēmas izstrādes plānu. Prasību specifikācija palīdz skaidri noteikt nepieciešamās sistēmas funkcijas. Ja specifikācija netiek izstrādāta precīzi un pārdomāti, sistēmas izstrādes procesā var rasties problēmas. Problēmu rezultātā palielinās risks saskarties ar nepilnībām, izveidotajā sistēmā.

2.1. FUNKCIONĀLĀS PRASĪBAS

Funkcionālās prasības norāda uz funkcijām, kuras ir jāiekļauj sistēmā, lai lietotājs spētu veikt konkrētu darbību. Šīs prasības definē ieinteresētā puse, sistēmas izstrādātājs tās specifiski apraksta un veido konkrētus nosacījumus funkcijas darbībā. Turpmāk dotās prasības tiks aprakstītas izmantojot *Gherkin* pierakstu sistēmu (Gherkin, 2024).

Funkcionālā prasība: Lietotājam tiek dota iespēja izveidot personīgu kontu tīmekļa vietnē.

User Story / Izstrādājamā vienība: Man kā tīmekļa vietnes lietotājam ir nepieciešama iespēja izveidot savu personīgu kontu, lai spētu sevi identificēt citiem lietotājiem, un iegūtu iespēju izmantot pārējās tīmekļa vietnes piedāvātās funkcijas.

Konteksts:

Kad lietotājs, tīmeklī ir atvēris vietni

Un kad lietotājs ir atvēris reģistrācijas sadaļu

Scenārijs: Veiksmīgā iznākuma scenārijs jeb tīmekļa vietnes lietotājs veiksmīgi izveidojis personīgu kontu

Ja lietotājs ir aizpildījis ievadlauku (“Name”), izpildot visus nepieciešamos nosacījumus

| Nepieciešamais nosacījums |

| Minimālais rakstzīmju skaits: 2 |

| Maksimālais rakstzīmju skaits: 15 |

Ja lietotājs ir aizpildījis ievadlauku (“Surname”), izpildot visus nepieciešamos nosacījumus

| Nepieciešamais nosacījums |

| Minimālais rakstzīmju skaits: 9 |

| Maksimālais rakstzīmju skaits: 20 |

Ja lietotājs ir aizpildījis ievadlauku (“Email”), izpildot visus nepieciešamos nosacījumus

| Nepieciešamais nosacījums |

| Minimālais rakstzīmju skaits: 5 |

| Maksimālais rakstzīmju skaits: 20 |

| Atbilstība e-pasta formātam (iekļaujot: “.”, “@”) |

| Unikalitāte (Nedrīkst būt eksistējošs kunds ar ievadīto e-pastu) |

Ja lietotājs ir aizpildījis ievadlauku (“Password”), izpildot visus nepieciešamos nosacījumus

| Nepieciešamais nosacījums |

| Minimālais rakstzīmju skaits: 9 |

| Maksimālais rakstzīmju skaits: 20 |

Tad lietotājam tiek izveidots personīgs kunds

Scenārijs: (“Name”) ievadlauks nav aizpildīts atbilstoši nosacījumiem

Ja nav izpildīts kāds no nosacījumiem

Tad zem ievadlauka tiek izvadīts kļūdu ziņojums

Scenārijs: (“Surname”) ievadlauks nav aizpildīts atbilstoši nosacījumiem

Ja nav izpildīts kāds no nosacījumiem

Tad zem ievadlauka tiek izvadīts kļūdu ziņojums

Scenārijs: (“Email”) ievadlauks nav aizpildīts atbilstoši nosacījumiem

Ja nav izpildīts kāds no nosacījumiem

Tad zem ievadlauka tiek izvadīts kļūdu ziņojums

Scenārijs: (“Password”) ievadlauks nav aizpildīts atbilstoši nosacījumiem

Ja nav izpildīts kāds no nosacījumiem

Tad zem ievadlauka tiek izvadīts kļūdu ziņojums

Funkcionālā prasība: Lietotājam tiek dota iespēja pieslēgties tīmekļa vietnē ar izveidotu personīgu kontu

User Story / Izstrādājamā vienība: Man kā tīmekļa vietnes lietotājam ir nepieciešama iespēja pieslēgties tīmekļa vietnē ar sevis personīgu kontu, lai iegūtu iespēju izmantot pārējās tīmekļa vietnes piedāvātās funkcijas.

Konteksts:

Kad lietotājs, tīmeklī ir atvēris vietni

Un kad lietotājs ir atvēris pieslēgšanās sadaļu

Scenārijs: Veiksmīgā iznākuma scenārijs jeb tīmekļa vietnes lietotājs veiksmīgi izveidojis personīgu kontu

Ja lietotājs ir aizpildījis ievadlauku (“Email”), izpildot visus nepieciešamos nosacījumus

| Nepieciešamais nosacījums |

| Minimālais rakstzīmju skaits: 5 |

| Maksimālais rakstzīmju skaits: 20 |

| Atbilstība e-pasta formātam (iekļaujot “.”, “@”) |

| Unikalitāte (Nedrīkst būt eksistējošs kunds ar ievadīto e-pastu) |

| Jāievada piereģistrēts/eksistējošs e-pasts |

Ja lietotājs ir aizpildījis ievadlauku (“Password”), izpildot visus nepieciešamos nosacījumus

| Nepieciešamais nosacījums |

| Minimālais rakstzīmju skaits: 9 |

| Maksimālais rakstzīmju skaits: 20 |

Tad lietotājam tiek izveidots personīgs kunds

Scenārijs: (“Email”) ievadlauks nav aizpildīts atbilstoši nosacījumiem

Ja nav izpildīts kāds no nosacījumiem

Tad zem ievadlauka tiek izvadīts kļūdu ziņojums

Scenārijs: (“Password”) ievadlauks nav aizpildīts atbilstoši nosacījumiem

Ja nav izpildīts kāds no nosacījumiem

Tad zem ievadlauka tiek izvadīts kļūdu ziņojums

Funkcionālā prasība: Lietotājam tiek dota iespēja izrakstīties no tīmekļa vietnes ar iepriekš izveidotu lietotāja kontu

User Story / Izstrādājamā vienība: Man kā tīmekļa vietnes lietotājam ir nepieciešama iespēja izrakstīties no tīmekļa vietnes..

Konteksts:

Kad lietotājs, tīmeklī ir atvēris vietni

Scenārijs: Veiksmīgā iznākuma scenārijs jeb lietotājam veiksmīgi izdevies izrakstīties no tīmekļa vietnes.

Ja lietotājs ir navigācijas joslā atvēris *My profile* sadaļu

Un ja lietotājs nospiedis pogu *Logout*

Tad lietotājas tiek izrakstīts no tīmekļa vietnes un pārvirzīts uz pieslēgšanās skatu.

Funkcionālā prasība: Lietotājam tiek dota iespēja pievienot blogu

User Story / Izstrādājamā vienība: Man kā tīmekļa vietnes lietotājam ir nepieciešama iespēja pieveinot bloga rakstu, lai spētu izpausties un parādīt to citiem tīmekļa vietnes lietotājiem

Piezīmes: Lietotājam, bloga pievienošanas laikā, nav nepieciešams ievadīt katru no ievadlaukiem, bet nepieciešams lai visi prasītie ievadlauki ir aizpildīti

Konteksts:

Kad lietotājs, pieslēdzies savam kontam

Un kad lietotājs ir atvēris bloga pievienošanas sadaļu (Add);

Scenārijs: Veiksmīgā iznākuma scenārijs jeb tīmekļa vietnes lietotājs veiksmīgi pievienojis blogu

Ja lietotājs ir aizpildījis ievadlauku (“Title”), izpildot visus nepieciešamos nosacījumus

| Nepieciešamais nosacījums |

| Minimālais rakstzīmju skaits: 4 |

| Maksimālais rakstzīmju skaits: 50 |

Ja lietotājs ir aizpildījis ievadlauku (“Description”), izpildot visus nepieciešamos nosacījumus

| Nepieciešamais nosacījums |

| Minimālais rakstzīmju skaits: 4 |

| Maksimālais rakstzīmju skaits: 4000 |

Ja lietotājs ir izvēlējies kategoriju, nolaižamajā izvēlnē (“Category”).

Ja lietotājs ir izvēlējies foto attēlu, augšpielādējot to no savas sistēmas lokālās atmiņas

Tad ekrāna labajā, augšējā stūrī parādās uznirstošais logs, ar paziņojumu par veiksmīgu bloga izveidošanu

Scenārijs: Ievadlauks (“Title”) nav aizpildīts atbilstoši uzdevuma nosacījumiem

Ja nav izpildīts kāds no nosacījumiem

Tad zem ievadlauka tiek izvadīts klūdu ziņojums

Scenārijs: Ievadlauks (“Description”) nav aizpildīts atbilstoši uzdevuma nosacījumiem

Ja nav izpildīts kāds no nosacījumiem

Tad zem ievadlauka tiek izvadīts klūdu ziņojums

Scenārijs: Nav izvēlēta kategorija, nolaižamajā izvēlnē (“Category”)

Tad zem izvēlnes. tiek izvadīts klūdu ziņojums

Scenārijs: Nav izvēlēts un augšpielādēts foto attēls

Tad zem augšpielādes sadaļas tiek izvadīts klūdu ziņojums

Funkcionālā prasība: Lietotājam tiek dota iespēja redīģēt informāciju, kādam no saviem blogiem

User Story / Izstrādājamā vienība: Man kā tīmekļa vietnes lietotājam ir nepieciešama iespēja rediģēt kādu no saviem bloga rakstiem, lai gadījumā, kad tiek ievadīta kļūdaina informācija, vai arī tā ir mainījusies, spētu to labot.

Piezīmes: Lietotājam, bloga rediģēšanas laikā, nav nepieciešams rediģēt katru no ievadlaukiem, bet nepieciešams lai visi prasītie ievadlauki ir aizpildīti

Konteksts:

Kad lietotājs, pieslēdzies savam kontam

Un kad lietotājs ir atvēris savu profila sadaļu (My profile->Profile);

Un kad lietotājs ir izvēlējies blogu, kuru rediģēt

Scenārijs: Veiksmīgā iznākuma scenārijs jeb tīmekļa vietnes lietotājs veiksmīgi spējis rediģēt bloga informāciju

Ja lietotājs ir aizpildījis ievadlauku (“Title”), izpildot visus nepieciešamos nosacījumus

| Nepieciešamais nosacījums |

| Minimālais rakstzīmju skaits: 4 |

| Maksimālais rakstzīmju skaits: 50 |

Ja lietotājs ir aizpildījis ievadlauku (“Description”), izpildot visus nepieciešamos nosacījumus

| Nepieciešamais nosacījums |

| Minimālais rakstzīmju skaits: 4 |

| Maksimālais rakstzīmju skaits: 4000 |

Ja lietotājs ir izvēlējies kategoriju, nolaižamajā izvēlnē (“Category”).

Ja lietotājs ir izvēlējies foto attēlu, augšpielādējot to no savas sistēmas lokālās atmiņas

Tad ekrāna labajā, augšējā stūrī parādās uznirstošais logs, ar paziņojumu par veiksmīgu bloga rediģēšanu

Scenārijs: Ievadlauks (“Title”) nav aizpildīts atbilstoši uzdevuma nosacījumiem

Ja nav izpildīts kāds no nosacījumiem

Tad zem ievadlauka tiek izvadīts kļūdu ziņojums

Scenārijs: Ievadlauks (“Description”) nav aizpildīts atbilstoši uzdevuma nosacījumiem

Ja nav izpildīts kāds no nosacījumiem

Tad zem ievadlauka tiek izvadīts kļūdu ziņojums

Scenārijs: Nav izvēlēta kategorija, nolaižamajā izvēlnē (“Category”)

Tad zem izvēlnes. tiek izvadīts kļūdu ziņojums

Scenārijs: Nav izvēlēts un augšpielādēts foto attēls

Tad zem augšpielādes sadaļas tiek izvadīts kļūdu ziņojums

Funkcionālā prasība: Lietotājam tiek dota iespēja apskatīt viņa izvēlēto blogu un tā saturu

User Story / Izstrādājamā vienība: Man kā tīmekļa vietnes lietotājam ir nepieciešama iespēja apskatīt kādu no tīmekļa vietnē esošajiem blogiem jeb apskatīt tā saturu

Konteksts:

Kad lietotājs, pieslēdzies savam kontam

Un kad lietotājs ir atvēris kādu no blogu saraksta sadaļām:

| Sadaļa |

| Visi blogi (Blogs-> All blogs) |

| Līdzīgie manām interesēm (Blogs-> For you) |

| Lietotājiem kuriem es sekoju (Bogs-> Followers) |

| Saglabātie blogi (My Profile->Saved) |

| Mani blogi (My Profile->Profile) |

Vai kad lietotājs ir izvēlējies apskatīt kādu no kategorijām

Scenārijs: Veiksmīgā iznākuma scenārijs jeb tīmekļa vietnes lietotājs veiksmīgi spējis apskatīt izvēlēto blogu un tā informāciju

Ja lietotājs izvēlējies blogu un ar kurSORU uzvIRZĪJIS uz tā attēla

Un ja lietotājs ar peles kreiso taustiņu vai skārienjūtīgo paliktni ir uzklikšķinājis uz izvēlētā bloga

Tad lietotājs tiek pārvirzīts uz bloga apskates sadaļu, kurā tas spēj apskatīt bloga informāciju

Scenārijs: Lietotājs nav spējis uzklikšķināt uz kāda konkrēta bloga

Tad lietotājs paliek blogu saraksta sadaļā, un netiek pārvirzīts uz blogu apskates sadaļu

Funkcionālā prasība: Lietotājam tiek dota iespēja apskatīt blogu sarakstu kas paredzēts priekš viņa, un veidots vadoties pēc rekomendāciju algoritma, ka ari apskatīt konkrētu blogu (“For you”)

User Story / Izstrādājamā vienība: Man kā tīmekļa vietnes lietotājam ir nepieciešama iespēja apskatīt līdzīgos blogus kas ir manām interesēm

Piezīmes Šajā sadaļā netiks rādīti nekādi rezultāti, ja lietotājs iepriekš nebūs apskatījis kādu blogu

Konteksts:

Kad lietotājs, pieslēdzies savam kontam

Un kad lietotājs ir atvēris kādu no blogu saraksta sadaļu (“For you”)

Scenārijs: Veiksmīgā iznākuma scenārijs jeb tīmekļa vietnes lietotājs veiksmīgi spējis apskatīt izvēlēto blogu sarakstu kas līdzīgs tā interesēm un apskatīt konkrēto blogu.

Ja lietotājs atvēris (“For you”) blogu sarakstu

Ja lietotājs ar kursoru ir izvēlējies kādu blogu apskatei

Un ja lietotājs ir uzklikšķinājis uz izvēlētā bloga

Tad lietotājs tiek pārvirzīts uz bloga apskates sadaļu, kurā tas spēj apskatīt bloga informāciju

Scenārijs: Lietotājs nav spējis atvērt (“For you”) skatu

Tad lietotājs netiek pārvirzīts uz (“For you”) blogu saraksta skatu

Funkcionālā prasība: Lietotājam tiek dota iespēja apskatīt blogu sarakstu kurus ievietojuši lietotāji kam tas seko (“Followers”)

User Story / Izstrādājamā vienība: Man kā tīmekļa vietnes lietotājam ir nepieciešama iespēja apskatīt blogus, kurus ievietojuši lietotāji kam es sekoju

Konteksts:

Kad lietotājs, pieslēdzies savam kontam

Un kad lietotājs ir atvēris kādu no blogu saraksta sadaļu (“Followers”)

Scenārijs: Veiksmīgā iznākuma scenārijs jeb tīmekļa vietnes lietotājs veiksmīgi spējis apskatīt izvēlēto blogu sarakstu, kurā ir blogi, kurus ievietojuši lietotāji kam tas seko, un apskatīt konkrēto blogu.

Ja lietotājs atvēris (“Followers”) blogu sarakstu

Ja lietotājs ar kursoru ir izvēlējies kādu blogu apskatei

Un ja lietotājs ir uzklikšķinājis uz izvēlētā bloga

Tad lietotājs tiek pārvirzīts uz bloga apskates sadaļu, kurā tas spēj apskatīt bloga informāciju

Scenārijs: Lietotājs nav spējis atvērt blogu saraksta sadaļu (“Followers”)

Tad netiek pārvirzīts uz blogu apskates sadaļu

Scenārijs: Lietotājs nav spējis uzklikšķināt uz kāda konkrēta bloga

Tad lietotājs paliek blogu saraksta sadaļā, un netiek pārvirzīts uz blogu apskates sadaļu

Funkcionālā prasība: Lietotājam tiek dota iespēja meklēt kādu konkrētu blogu pēc tā nosaukuma, vai mirkļa birkas

User Story / Izstrādājamā vienība: Man kā tīmekļa vietnes lietotājam ir nepieciešama iespēja meklēt blogu pēc tā nosaukuma vai mirkļa birkas

Konteksts:

Kad lietotājs, pieslēdzies savam kontam

Un kad lietotājs ir atvēris kādu no blogu saraksta sadaļām:

- | Sadaļa |
- | Visi blogi (Blogs-> All blogs) |
- | Līdzīgie manām interesēm (Blogs-> For you) |
- | Lietotājiem kuriem es sekoju (Bogs-> Followers) |
- | Saglabātie blogi (My Profile->Saved) |
- | Mani blogi (My Profile->Profile) |

Vai kad lietotājs ir izvēlējies apskatīt kādu no kategorijām

Scenārijs: Veiksmīgā iznākuma scenārijs jeb tīmekļa vietnes lietotājs veiksmīgi izmantojis meklētāju, un ir izdevies atrast meklēto blogu

Ja lietotājs, meklēšanas ievadlaukā ir ievadījis rakstzīmju salikumu vai vārdu

Ja lietotājs, meklēšanas izvēlējies meklēšanas veidu :

- | Meklēšanas veidi|
- | Pēc nosaukuma |
- | Pēc mirkļa birkas |

Un ja datubāzē eksistē blogs, kura nosaukumā ir šis ievadītais rakstzīmju salikums, vai arī šis rakstzīmju salikums sakrīt ar kādu no mirkļa birkām

Tad tīmekļa vietnes blogu sarakstā parādās blogi, kuru nosaukums vai mirkļa birka satur ievadlaukā ievadīto informāciju.

Scenārijs: Meklēšanas ievadlauks nav aizpildīts

Ja lietotājs nav aizpildījis ievadlauku

Ja lietotājs nav izvēlējies meklēšanas tipu

Un ja lietotājs ir notīrījis ievadlaukā ievadīto informāciju

Tad tiek atgriezti visi blogi atkarībā no izvēlētā blogu saraksta vai kategorijas

Scenārijs: Lietotājs ievadījis neatbilstošu informāciju meklēšanas ievadlaukā

Ja lietotājs ievadījis rakstzīmju salikumu vai vārdu, kuru nesatur neviens bloga nosaukums un mirkļa birka.

Tad blogu saraksta sadaļa tiek parādīts kļūdu ziņojums, ka netika atrasti rezultāti

Funkcionālā prasība: Lietotājam tiek dota iespēja saglabāt kādu no blogiem

User Story / Izstrādājamā vienība: Man kā tīmekļa vietnes lietotājam ir nepieciešama iespēja saglabāt kādu no blogiem, lai vajadzības gadījuma, to spētu ātrāk atrast.

Konteksts:

Kad lietotājs, pieslēdzies savam kontam

Un kad lietotājs ir izvēlējies kādu kategoriju

Vai kad lietotājs izvēlējies citu blogu sadaļu

Un kad lietotājs ir izvēlējies un atvēris konkrētu blogu

Scenārijs: Veiksmīgā iznākuma scenārijs jeb tīmekļa vietnes lietotājs veiksmīgi saglabājis sevis izvēlētu blogu

Ja lietotājs ir uzklikšķinājis uz saglabāšanas ikonas

Tad pa kreisi no saglabāšanas tiek parādīts teksts saglabāts (angļu v. *Saved*), kā arī blogs tiek saglabāts.

Scenārijs: Lietotājs nav spējis uzklikšķināt uz saglabāšanas ikonas

Tad netiek saglabāts šis bloga ieraksts

Funkcionālā prasība: Lietotājam tiek dota iespēja rediģēt personīga konta informāciju

User Story / Izstrādājamā vienība: Man kā tīmekļa vietnes lietotājam ir nepieciešama iespēja rediģēt savu konta informāciju, lai es spētu identificēties ar jauno informāciju

Piezīmes: Lietotājam, personīga konta rediģēšanas laikā, nav nepieciešams rediģēt katru no ievadlaukiem, bet lietotājs izvēlās ko vēlās rediģēt.

Konteksts:

Kad lietotājs, pieslēdzies savam kontam

Kad lietotājs ir atvēris iestatījumu sadaļu (My Profile->Profile)

Un kad lietotājs ir noklikšķinājis uz zobraza ikonas, atverot iestatījumu skatu

Scenārijs: Veiksmīgā iznākuma scenārijs jeb tīmekļa vietnes lietotājs veiksmīgi spējis rediģēt savu konta informāciju.

Ja lietotājs ir aizpildījis ievadlauku (“Email”), izpildot visus nepieciešamos nosacījumus

| Nepieciešamais nosacījums |

| Minimālais rakstzīmju skaits: 5 |

| Maksimālais rakstzīmju skaits: 20 |

| Atbilstība e-pasta formātam (iekļaujot: “.” , “@”) |

| Unikalitātē (Nedrīkst būt eksistējošs kunds ar ievadīto e-pastu) |

Ja lietotājs ir aizpildījis ievadlauku (“Name”), izpildot visus nepieciešamos nosacījumus

| Nepieciešamais nosacījums |

| Minimālais rakstzīmju skaits: 2 |

| Maksimālais rakstzīmju skaits: 15 |

Ja lietotājs ir aizpildījis ievadlauku (“Surname”), izpildot visus nepieciešamos nosacījumus

| Nepieciešamais nosacījums |

| Minimālais rakstzīmju skaits: 9 |

| Maksimālais rakstzīmju skaits: 20 |

Un ja lietotājs ir uzklikšķinājis uz pogas (“Save”)

Tad ekrāna labajā, augšējā stūrī parādās uznirstošais logs, ar paziņojumu par profila informācijas rediģēšanu

Scenārijs: (“Name”) ievadlauks nav aizpildīts atbilstoši nosacījumiem

Ja nav izpildīts kāds no nosacījumiem

Tad zem ievadlauka tiek izvadīts klūdu ziņojums

Scenārijs: (“Surname”) ievadlauks nav aizpildīts atbilstoši nosacījumiem

Ja nav izpildīts kāds no nosacījumiem

Tad zem ievadlauka tiek izvadīts klūdu ziņojums

Scenārijs: (“Email”) ievadlauks nav aizpildīts atbilstoši nosacījumiem

Ja nav izpildīts kāds no nosacījumiem

Tad zem ievadlauka tiek izvadīts klūdu ziņojums

Funkcionālā prasība: Lietotājam tiek dota iespēja rediģēt augšupielādēt profila attēlu

User Story / Izstrādājamā vienība: Man kā tīmekļa vietnes lietotājam ir nepieciešama iespēja nomainīt noklusēju profila attēlu, lai identificētu sevi

Konteksts:

Kad lietotājs ir atvēris iestatījumu sadaļu (My Profile->Profile)

Un kad lietotājs ir noklikšķinājis uz zobraza ikonas, atverot iestatījumu skatu

Scenārijs: Veiksmīgā iznākuma scenārijs jeb tīmekļa vietnes lietotājs veiksmīgi spējis augšupielādēt profila attēlu

Ja lietotājs ir izvēlējies datni no lokālas sistēmas

Ja visi pārējie konta informācijas lauki būs aizpildīti

Ja datnes tips ir

| datnes tips |

| jpg |

| png |

| jpeg |

Un ja lietotājs ir uzklikšķinājis uz pogas (“Save”)

Tad ekrāna labajā, augšējā stūrī parādās uznirstošais logs, ar paziņojumu par profila informācijas rediģēšanu

Scenārijs: Ja kāds no profila informācijas ievadlaukiem nav aizpildīts

Ja kāds no profila ievadlaukiem nav aizpildīts

Tad zem neaizpildītā ievadlauka parādīsies klūdu ziņojums

Scenārijs: Ja augšupielādētajam profila attēlam ir neatbilstošs datu tips

Ja profila attēlam ir neatbilstošs datu tips vai tas nav augšupielādēts

Tad zem noklusējuma vai jau iepriekš augšupielādētā profila attēla parādīsies klūdu ziņojums

Funkcionālā prasība: Lietotājam tiek dota mainīt paroli

User Story / Izstrādājamā vienība: Man kā tīmekļa vietnes lietotājam ir nepieciešama iespēja mainīt paroli, lai spētu pieslēgties kontam ar to

Konteksts:

Kad lietotājs, pieslēdzies savam kontam

Un kad lietotājs ir atvēris paroles maiņas sadaļu

Scenārijs: Veiksmīgā iznākuma scenārijs jeb tīmekļa vietnes lietotājs veiksmīgi spējis veikt paroles nomaiņu

Ja lietotājs ir aizpildījis ievadlauku (“Email”), izpildot visus nepieciešamos nosacījumus

| Nepieciešamais nosacījums |

| Minimālais rakstzīmju skaits: 5 |

| Maksimālais rakstzīmju skaits: 20 |

| Atbilstība e-pasta formātam (iekļaujot: “.” , “@”) |

Ja lietotājs ar datorpeles kreiso taustiņu vai skārienjūtīgo paliktni ir uzklikšķinājis uz pogas (“Send”)

Ja lietotājs ir atvēris iepriekš norādīto e-pastu un atvēris sūtīto vēstuli

Ja lietotājs ir atvēris vēstulē sūtīto vietrādi

Ja lietotājs ir aizpildījis ievadlauku (“Password”), izpildot visus nepieciešamos nosacījumus

| Nepieciešamais nosacījums |

| Minimālais rakstzīmju skaits: 9 |

| Maksimālais rakstzīmju skaits: 20 |

Ja lietotājs ir aizpildījis ievadlauku (“Confirm password”), izpildot visus nepieciešamos nosacījumus

| Nepieciešamais nosacījums |

| Minimālais rakstzīmju skaits: 9 |

| Maksimālais rakstzīmju skaits: 20 |

Ja ievadlaukos (“Password”) un (“Confirm password”), ievadītās vērtības ir identiskas

Un ja lietotājs ar datorpeles kreiso taustiņu vai skārienjūtīgo paliktni ir uzklikšķinājis uz pogas (“Reset”)

Tad tiek parādīts ziņojums par veiksmīgu paroles maiņu, un lietotājs tiek aizvirzīts uz pierakstīšanās skatu

Scenārijs: (“Email”) ievadlauks nav aizpildīts atbilstoši nosacījumiem

Ja nav izpildīts kāds no nosacījumiem

Tad zem ievadlauka tiek izvadīts kļūdu ziņojums

Scenārijs: lietotājs ievadījis kļūdainu e-pasta adresi

Ja lietotājs ievadījis kļūdainu e-pasta adresi

Tad lietotājs nesaņem vēstuli ar vietrādi priekš paroles nomaiņas, jāveic vēl viens mēģinājums

Scenārijs: lietotājs nokavējis vēstulē atsūtītā vietrāža darbības laiku

Ja lietotājs nokavējis vēstulē atsūtītā vietrāža darbības laiku

Tad lietotājs netiks pārvirzīts uz paroles maiņas skatu, jāveic vēl viens mēģinājums

Scenārijs: (“Password”) ievadlauks nav aizpildīts atbilstoši nosacījumiem

Ja nav izpildīts kāds no nosacījumiem

Tad zem ievadlauka tiek izvadīts kļūdu ziņojums

Scenārijs: (“Confirm password”) ievadlauks nav aizpildīts atbilstoši nosacījumiem

Ja nav izpildīts kāds no nosacījumiem

Tad zem ievadlauka tiek izvadīts kļūdu ziņojums

Funkcionālā prasība: Lietotājam tiek dota iespēja ierakstīt komentāru zem kāda izvēlēta bloga

User Story / Izstrādājamā vienība: Man kā tīmekļa vietnes lietotājam ir nepieciešama iespēja komentēt kādu no blogiem, lai izpaustu savu viedokli

Konteksts:

Kad lietotājs, pieslēdzies savam kontam

Un kad lietotājs ir izvēlējies kādu kategoriju

Vai kad lietotājs ir atvēris kādu citu no blogu kategorijām

Un kad lietotājs ir izvēlējies un atvēris konkrētu blogu

Scenārijs: Veiksmīgā iznākuma scenārijs jeb tīmekļa vietnes lietotājs veiksmīgi spējis ievietot komentāru

Ja lietotājs ir aizpildījis ievadlauku (“Comment”)

Un ja lietotājs ir uzklikšķinājis uz sūtīšanas ikonas

Tad komentārs tiek pievienots, un tas parādās komentāru saraksta augšgalā

Scenārijs: Lietotājs nav aizpildījis ievadlauku (“Comment”)

Ja lietotājs nav aizpildījis ievadlauku (“Comment”)

Tad uzklikšķinot uz sūtīšanas ikonas, netiek veikta darbība

Funkcionālā prasība: Lietotājam tiek dota iespēja sekot kādam lietotājam

Ietekmes karte:

Dizaina paraugs:

User Story / Izstrādājamā vienība: Man kā tīmekļa vietnes lietotājam ir nepieciešama iespēja sekot kādam lietotājam

Konteksts:

Kad lietotājs, pieslēdzies savam kontam

Un kad lietotājs ir atvēris kāda lietotāja profilu

Scenārijs: Veiksmīgā iznākuma scenārijs jeb tīmekļa vietnes lietotājs veiksmīgi spējis piesekot kādam citam lietotājam

Ja lietotājs spējis atvērt lietotāja profilu un noklikšķinājis uz piesekošanas pogas (“Follow”)

Tad šis lietotāja profils parādīsies profila sadaļā

Scenārijs: Lietotājs nav spējis uzklikšķināt uz piesekošanas pogas (“Follow”)

Tad lietotājs nesāks sekot šim lietotāja profilam

Funkcionālā prasība: Lietotājam tiek dota iespēja pārstāt sekot kādam lietotājam apskatot lietotāja kontu

User Story / Izstrādājamā vienība: Man kā tīmekļa vietnes lietotājam ir nepieciešama iespēja pārstāt sekot kādam no lietotājiem

Konteksts:

Kad lietotājs, pieslēdzies savam kontam

Un kad lietotājs ir atvēris kāda lietotāja profilu

Scenārijs: Veiksmīgā iznākuma scenārijs jeb tīmekļa vietnes lietotājs veiksmīgi spējis pārstāt sekot lietotājam

Ja lietotājs spējis atvērt lietotāja profilu un noklikšķinājis uz pogas (“Following”)

Tad šis lietotāja profils tiks izņemts no lietotāja saraksta

Scenārijs: Lietotājs nav spējis uzklikšķināt uz pogas (“Following”)

Tad lietotājs nepārstās sekot šim lietotāja profilam

Funkcionālā prasība: Lietotājam tiek dota iespēja pārstāt sekot kādam lietotājam apskatot to savā profilā

User Story / Izstrādājamā vienība: Man kā tīmekļa vietnes lietotājam ir nepieciešama iespēja pārstāt sekot kādam no lietotājiem

Konteksts:

Kad lietotājs, pieslēdzies savam kontam

Un kad lietotājs ir atvēris kāda savu profilu (“My profile ->Profile”)

Scenārijs: Veiksmīgā iznākuma scenārijs jeb tīmekļa vietnes lietotājs veiksmīgi spējis pārstāt sekot lietotājam

Ja lietotājs spējis atvērt savu lietotāja profilu

Ja lietotājs spējis uzklikšķināt uz lietotāju skaita kam viņš seko (“Following”)

Un ja lietotājs spējis uzklikšķināt uz atsekošana pogas (“Unfollow”), kas atrodas pretī attiecīgajam lietotājam

Tad šis lietotāja profils tiks izņemts no lietotāja saraksta (“Following”)

Scenārijs: Lietotājs nav spējis uzklikšķināt uz pogas (“Unfollow”)

Tad lietotājs nepārstās sekot šim lietotāja profilam

2.2. NEFUNKCIONĀLĀS PRASĪBAS

Galvenokārt, nefunkcionālās prasības nosaka nepieciešamos līdzekļus, lai sasniegtu nostādītos mērķus. Tās nodrošina pilnvērtīgu sistēmas darbību, bet nav tieši saistītas ar tās funkcionalitāti. Šīs funkcijas apraksta sistēmas vispārējās īpašības, lietotāja pieredzi un citus aspektus (skatīt 2. tabulā)

2. tabula

Prasību uzskaitījums

| Kārtas nr. | Prasība | Apraksts |
|------------|-----------------------------|--|
| 1. | Reaģējoša tīmekļa vietne | Tīmekļa vietnei nepieciešamība pilnvērtīgi darboties uz dažāda izmēra ekrāniem. Vietnes saturam jābūt saprotamam un saredzamam. |
| 2. | Datu aizsardzība un drošība | Nepieciešamība šifrēt paroles un ieviest unikālus žetonus, kuri tiek izveidoti pieslēdzoties, tādejādi neautorizēti lietotāji nespēj piekļūt datiem. |
| 3. | Pieejamība | Tīmekļa vietnē tiek izvēlēta krāsu palete, lai krāsu akcentus spētu saskatīt arī cilvēki ar redzes traucējumiem, kā arī ir iespēja iestatīt personalizētu krāsu akcentu. |
| 5. | Veikspēja | Tīmekļa vietnei jāspēj izpildīt visas funkcijas, ātri un efektīvi, lai nerastu neērtības lietotājiem |
| 6. | Uzturamība | Lietotnei jābūt pieejamai 24 stundas diennaktī, 7 dienas nedēļā, kā arī jābūt iespējai bez problēmām pārlikt, tīmekļa vietni uz kādu serveri. |
| 7. | Dizains un izkārtojums | Dizainam un izkārtojumam jābūt intuitīvam un modernam, kas sniegs lietotājam patīkamu pieredzi, lietojot tīmekļa vietni. |

2.2.1. REAĢĒJOŠA TĪMEKĻA VIETNE

Pielāgojamība ierīcēm un ekrāna izmēriem

Reaģējoša tīmekļa vietne pielāgojas dažādu ierīču un ekrāna izmēru prasībām, nodrošinot optimālu lietojamību gan datoros, gan mobilajās ierīcēs. Tas ir īpaši svarīgi, ņemot vērā dažādo ierīču klāstu, ko cilvēki izmanto ikdienā.

Lietotāja pieredze

Reagējošas tīmekļa vietnes piedāvā lietotājiem gludu un kvalitatīvu pieredzi, neatkarīgi no tā, kādā ierīcē tie to izmanto. Laba lietotāja pieredze ir būtiska, lai saglabātu lietotāju interesi un uzticību.

2.2.2. DATU DROŠĪBA UN PAROLES

Aizmugursistēmas (angļu v. *back-end*) izstrādē izmantoju programmēšanas valodas *PHP* ietvaru *Laravel*. Šis ietvars sevī ietver vairākus principus un funkcijas, kas palīdz nodrošināt datu aizsardzību:

Eloquent ORM

Eloquent ORM, palīdz izveidot drošus datu pieprasījumus, lai novērstu SQL injekcijas. Eloquent ļauj izveidot datu modeļus, kas nodrošina vieglu un drošu piekļuvi datiem. Tas automātiski nolasa un neļauj ievadīt SQL injekcijas (Otwell, 2024a).

Middleware

Palīdz veikt dažādas darbības pirms vai pēc pieprasījuma apstrādes. *Middleware* var tikt pielāgota datu pārbaudēm, autentifikācijai un citām drošības funkcijām (Otwell, 2024b).

Datu drošība ir viena no svarīgākajām prasībām tādēļ papildus izmantoju vēl šādus principus:

Žetonu balstīta autentifikācija (angļu v. *Token-based Authentication*)

Žetonu balstītā autentifikācija ir autentifikācijas metode, kas izmanto žetonus (angļu v. *token*). Datoru drošības un tīmekļa izstrādes kontekstā, žetons (angļu v. *token*) sastāv no nejauši izvēlētām rakstu zīmēm un simboliem, kas kalpo kā lietotāja identitātes pierādījums. Šī metode tiek bieži izmantota, lai nodrošinātu piekļuvi resursiem un API (lietotāja programmēšanas saskarne) internetā, šī metode ir droša, jo dati tiek aizsargāti ar žetonu, kuru ir sarežģīti uzminēt. (Fortinet, 2024).

Manis veidotajā sistēmā šis žetons (angļu v. *token*), tiek izveidots veiksmīgas pieslēgšanās brīdī un tiek izdzēsts izrakstīšanās gadījumā. Ikreiz, kad tiek veikts pieprasījums uz aizmugursistēmu ar *API*, sākotnēji tiek izpildīts iepriekš minētais *Middleware* kurā tiek pārbaudīts vai konkrētais žetons (angļu v. *token*) eksistē servera datubāzē un vai sesijas laiks nav beidzies, ja žetons (angļu v. *token*), eksistē, tad lietotājam tiek dota pieeja datiem un funkcijām, ja tas neeksistē, lietotājs tiek aizvirzīts uz pieslēgšanās skatu, izdzēšot žetonu (angļu v. *token*).

Paroles un to šifrēšana

Savā sistēmā izmantoju *Laravel* iebūvēto šifrēšanas metodi, kurā tiek izmantota *Bcrypt* kā noklusējuma kriptogrāfisko funkciju. Paroles šifrēšana nepieciešama lai aizsargātu lietotāju kontus no trešo personu kiberuzbrukumiem.

2.2.3. PIEEJAMĪBA

Krāsu paletes konfigurēšana

Tīmekļa vietnes lietotājam būs iespēja iestatījumos konfigurēt krāsu paleti, mainot akcenta krāsu. Lietotājiem ar redzes traucējumiem būs iespēja atrast sev ērtāko akcenta krāsu, padarot tīmekļa vietnes izmantošanu ērtāku.

Attēlu alternatīvie tagi

Šie tagi palīdz aprakstīt attēlu saturu, it īpaši tiem lietotājiem, kuri nevar redzēt attēlus vai izmanto ekrāna lasītājus. Attēlu alternatīvais teksts tiek rādīts, ja attēls nevar tikt ielādēts, ja lietotājs atspējo attēlu rādīšanu savā pārlūkprogrammā vai, ja lietotājs ir neredzīgs un izmanto ekrāna lasītāju.

Savā sistēmā attēlu alternatīvajā tagā ievietošu bloga nosaukumu, lai neredzīgie lietotāji spētu saprast, kas tiek attēlots attēlā

2.2.4. VEIKTSPĒJA

Veiktspēja ir viena no galvenajām prasībām jebkuras sistēmas izveidē. Ikvienai sistēmai ir svarīgi, lai tā spētu lietotāja pieprasītās funkcijas izpildīt bez aizķeršanās un datus atgriezt ātri.

Ko ietekmē sistēmas veiktspēja:

- Lietotāja pieredzi
 - Lietotājam veiktspēja ir ļoti svarīgs aspekts tīmekļa vietnes lietošanā. Tas sagaida ātru un efektīvu datu un pieprasījumu apstrādi, ja tas notiek lēni, lietotājs var zaudēt interesi un atstāt tīmekļa vietni.
- SEO un sasniedzamību
 - Google un citi tīmekļa meklētāji pārskata un ņem vērā tīmekļa vietnes veiktspēju, tādejādi noskaidrojot to, cik ātri lietotāji spēs atrast to. Jo labāka veiktspēja jo labāka pozīcija tīmekļa meklētājos

2.2.5. UZTURAMĪBA

Uzturamība ir ļoti svarīga vairāk ieinteresētai pusei jeb persona, kurai tiek veidota konkrētā sistēma vai tīmekļa vietne. Sistēmas izstrādātājam jāparedz vairāki scenāriji un jācenšas atvieglot šo scenāriju izpilde, ko veiks ieinteresētā puse. Personai kurai tiek veidota tīmekļa vietnei ir svarīgi uzturamības aspekti.

Viegla sistēmas/tīmekļa vietnes pārvietošana no kāda servera un citu

Jābūt konfigurācijas failiem, attiecīgi, kuros glabājas servera informācija, lai pārvietošanas gadījumā, tos var mainīt. Ja nebūs šāda konfigurācijas faila, tad nāksies informāciju mainīt katrā projekta failā kur tas nepieciešams.

2.2.6. DIZAINS UN IZKĀRTOJUMS

Dizains un izkārtojums ir svarīgs aspekts veiksmīgas tīmekļa vietnes izveidošanā. Pieslēdzoties tīmekļa vietnē, lietotājs gūst pirmos iespaidus tieši no dizaina aspekta, un tieši šis pirmais iespaids var ietekmēt lietotāja pieredzi izmantojot to. Ja lietotājam dizains nešķitīs pievilcīgs un izkārtojums nebūs intuitīvs, tad lietotājs neizmantos šo tīmekļa vietni, visdrīzāk neatgriezīsies tajā, jo būs sabojāts pirmais iespaids.

Manis veidotajā tīmekļa vietnē nepieciešamās izkārtojuma, dizaina un sadaļu prasības:

- pieslēgšanās un pierakstīšanās skats;
- navigācijas josla;
- titullapa;
- bloga pievienošanas skats;
- bloga redīģēšanas skats;
- blogu saraksta skats;
- bloga apskates skats;
- lietotāja profila skats;
- profila un iestatījumu skats;
- pievilcīga krāsu palete;
- intuitīvs izkārtojums;
- u.c.

3. MĒRĶA SASNIEGŠANAS LĪDZEKĻU IZVĒLE UN PAMATOJUMS

PHP - Servera puses skriptu valoda, izmantojama dinamisku tīmekļa lapu izstrādei. Tā ļauj integrēt HTML kodu ar iekšējiem skriptiem, piedāvājot dinamiskas funkcijas un datu apstrādi. PHP ir platformas neatkarīga, to izmanto daudzi tīmekļa attīstītāji, jo tā ir viegli uztverama, atbalsta dažādas datu bāzes un nodrošina elastīgumu un izturību tīmekļa vietnēm (PHP, 2024).

JavaScript - Tīmekļa lapu skriptu valoda, ko lieto pārlūki, lai piešķirtu dinamiskas un interaktīvas funkcijas tīmekļa vietnēm. Tas tiek izmantots klienta puses programmēšanai, ļaujot manipulēt HTML un CSS, veicot darbības pārlūkprogrammā. JavaScript atbalsta dažādas bibliotēkas un ietver dažādas tehnoloģijas, piemēram, AJAX, kas ļauj asinhroni mainīt datus bez lapas atkārtotas ielādes. JavaScript ir būtisks tīmekļa attīstības elements un nodrošina bagātīgas lietotāja pieredzes izveidi (mdn web docs, 2024a).

CSS - (Cascading Style Sheets) ir tīmekļa lapu stila valoda, kas nosaka izskatu un izvietojumu HTML dokumentos. Tas ļauj attīstītājiem precīzi kontrolēt elementu izmērus, krāsas, fontus un izkārtojumu, nodrošinot vienotu vizuālo izteiksmi tīmekļa vietnēs. CSS piedāvā iespēju ieviest stila norādes gan tieši HTML failā, gan arī atsevišķos CSS failos, kas padara kodu uzturamu un pārvaldāmu. Tas ir būtisks līdzeklis tīmekļa attīstītājiem, lai sasniegtu profesionālu un pievilcīgu dizainu (mdn web docs, 2024b).

Laravel - PHP pamatota, moderna un eleganta tīmekļa lietotņu izstrādes karkass, kas piedāvā ērtu un efektīvu veidu veidot drošas tīmekļa aplikācijas. Tas iekļauj daudzas funkcijas, piemēram, MVC arhitektūru, datu migrāciju, iebūvētu autentifikāciju un spēcīgu ORM (Eloquent). Laravel piedāvā ērtu sintaksi, kas paātrina izstrādi, kā arī plašu kopienas atbalstu un dokumentāciju. Tas ir iecienīts rīks tīmekļa attīstītājiem, lai ātri un efektīvi radītu modernas lietotnes (Otwell, 2024c).

React - JavaScript bibliotēka, ko izmanto lietotāja saskarnes izveidei tīmekļa lietotnēs. To radīja Facebook, un tas piedāvā efektīvu un deklaratīvu veidu, kā veidot UI komponentes. React izmanto virtual DOM, optimizējot lapas atjaunināšanu un nodrošinot lietotājiem ātru un efektīvu lietošanas pieredzi. Tā atbalsta komponentu atkārtotu izmantošanu un vienkāršo datu plūsmu, padarot kodu uztveramu un uzturamu. React ir populārs tīmekļa attīstības rīks un bieži tiek kombinēts ar citām tehnoloģijām, piemēram, Redux, lai veidotu pilnvērtīgas lietotnes (React, 2024).

SCSS - CSS priekšprocesors, kas piedāvā paplašinātas iespējas stilu definēšanai. Tas ievieš mainīgos un citus programmēšanas elementus, padarot stila definīcijas elastīgākas un vieglāk uztveramas. SCSS izmanto saīsinājumus un nodrošina hierarhiju, funkcijas un citas paplašinātas iespējas, kas atvieglo kodu pārvaldīšanu un atkārtotu izmantošanu. Pēc tam tas tiek kompilēts par

parasto CSS, lai pārlūki varētu to interpretēt. SCSS ir plaši izmantots tīmekļa attīstībā, it īpaši, izmantojot rīkus kā Sass (SCSS, 2024).

MySQL - Atvērtā koda relāciju datu bāzu pārvaldības sistēma, ko izmanto, lai organizētu un uzglabātu strukturētus datus. Tas piedāvā efektīvu un drošu veidu, kā apstrādāt lielu datu apjomu, un to bieži izmanto tīmekļa lietotņu, uzņēmumu un citu projektu datu glabāšanai un pārvaldībai. MySQL atbalsta SQL valodu, kas nodrošina vienkāršu un standartizētu veidu, kā valdīt pār datiem. Tas ir viens no populārākajiem datu bāzu pārvaldības sistēmu risinājumiem, ko lieto daudzi attīstītāji visā pasaule (MySQL, 2024).

Docker - Konteinerizācijas platforma, kas ļauj izstrādātājiem izveidot, testēt un izpildīt aplikācijas kā neatkarīgus, pārnēsājamus konteinerus. Konteineri ietver visu nepieciešamo vides un atkarību informāciju, padarot aplikācijas viegli pārnēsājamas starp dažādām vidēm. Docker piedāvā vienkāršu saskarni un automatizāciju procesiem, piemēram, lietotņu implementācijai. Tas palīdz samazināt atkarību konfliktus un nodrošina konsekvenči starp izstrādes un izpildes vidi, padarot aplikāciju pārvietošanu starp dažādām infrastruktūrām ātrāku un efektīvāku (Doker, 2024).

4. SISTĒMAS UN DATU STRUKTŪRAS MODELIS

Sistēmas struktūras modelis apraksta dažādu sistēmas sadaļu darbību. Tas vizuāli parāda darbību struktūru un principus. Šajā sadaļā tiks aprakstīti vairāki sistēmas struktūras modeļu paveidi:

- ER diagramma,
- komponenšu diagrammas:
 - datubāžu komponenšu diagramma,
 - priekšgalsistēmas komponenšu diagramma.

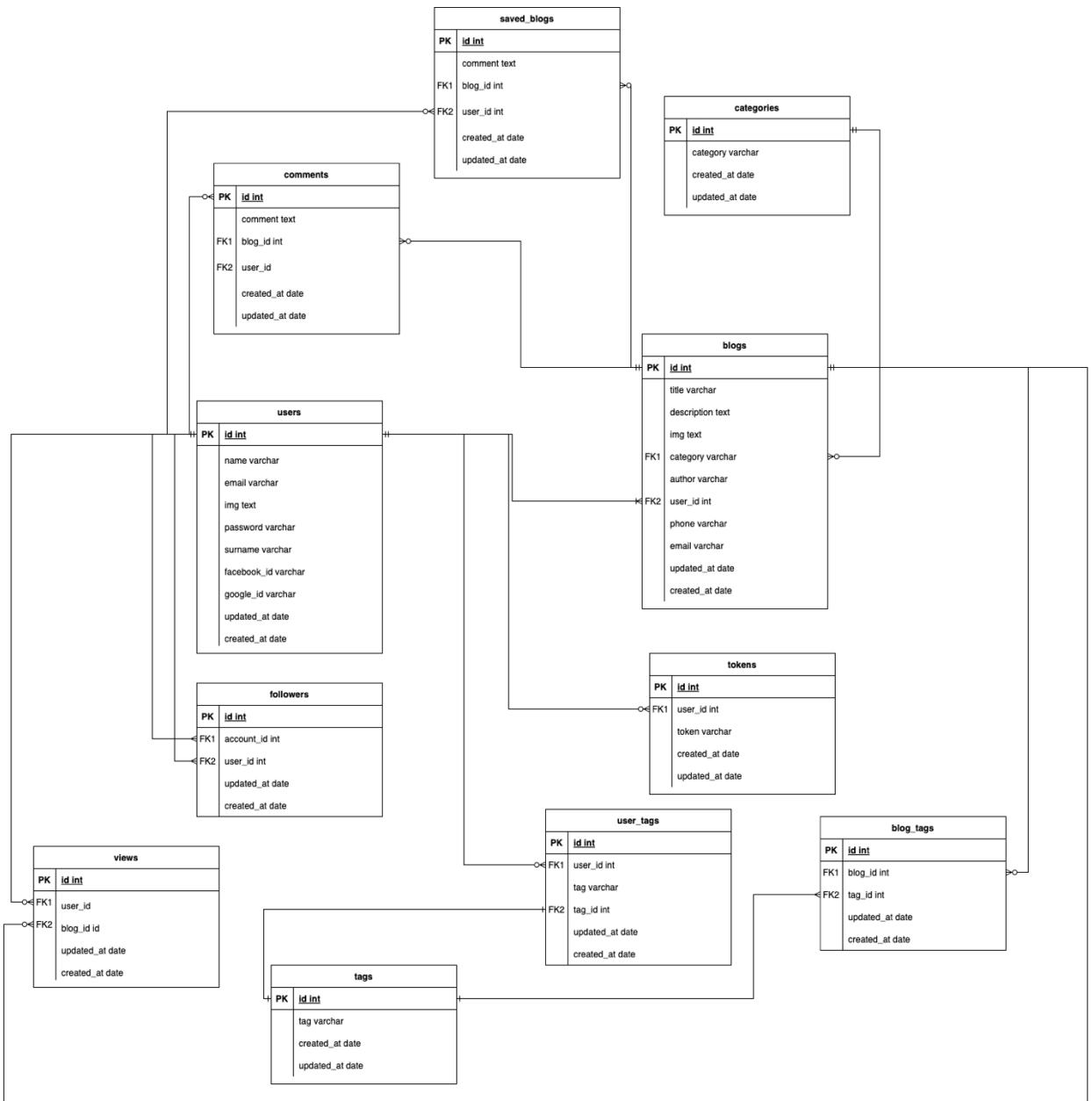
4.1. ER DIAGRAMMA

ER diagrammas (angļu v. *Entity–relationship model*) ir datu bāzu modelēšanas metode, kura tiek izmantota, lai izveidotu datu bāzes shēmu vai datu modeli un izprastu veidojamās datu bāzes uzbūvi un pamatprincipus. Diagrammas parasti izmanto relāciju datu bāzes izstrādē (Peter Pin-shan Chen, 1976).

Savas tīmekļa vietnes darbībā, datu glabāšanā izmantoju 6 datubāzes tabulas:

- *Users* – glabā lietotāju konta un pieslēgšanās informāciju;
- *Blogs* – glabā bloga informāciju;
- *Category* – glabā kategoriju informāciju;
- *Saved blogs* – glabā informāciju par saglabātiem blogiem;
- *Tokens* – glabā autorizēšanās žetonus;
- *Comments* – glabā blogu komentārus;
- *Followers* – glabā sekotājus un profilus kam seko lietotāji;
- *Tags* – glabā mirkļa birkas;
- *User tags* – glabā mirkļa birkas kuras apskatījis lietotājs;
- *Blog tags* – glabā bloga mirkļa birkas.
- *Views* – glabā lietotāja apskatītos blogus.

ER diagrammā tiek attēlotas datubāzes un relācijas, kā arī vizuāli attēloti savienojumi starp datubāzēm (skatīt 7. attēlā).



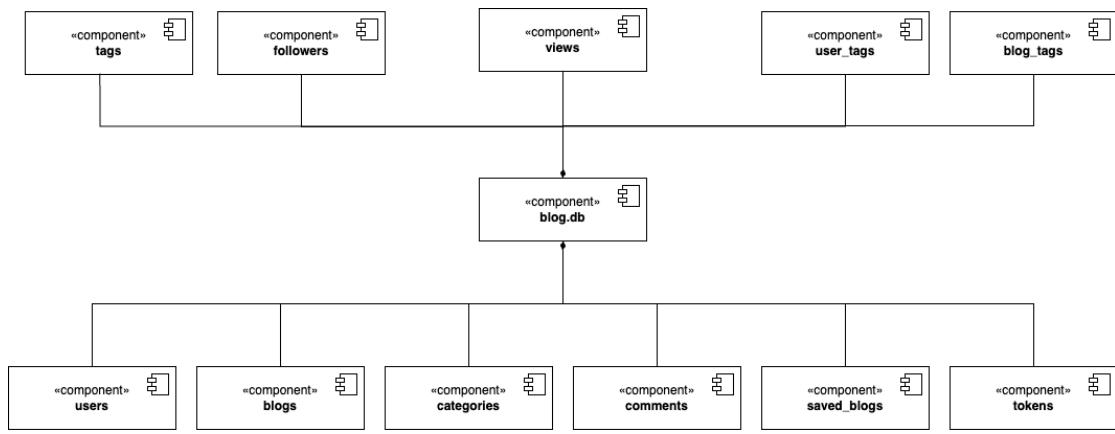
7. attēls. *BlogIt* ER diagramma

4.2. KOMPONENŠU DIAGRAMMAS

Komponenšu diagramma atspoguļo sistēmas struktūru, komponentes. Šīs diagrammas izmanto programmatūras projektēšanas posmā, lai spētu izveidotu loģisku un stabili struktūru, kas atviegloju sistēmas izstrādi. Komponenšu diagrammā tiek atspoguļotas komponenšu savstarpējās attiecības.

4.2.1. DATUBĀZES KOMPONENŠU DIAGRAMMA

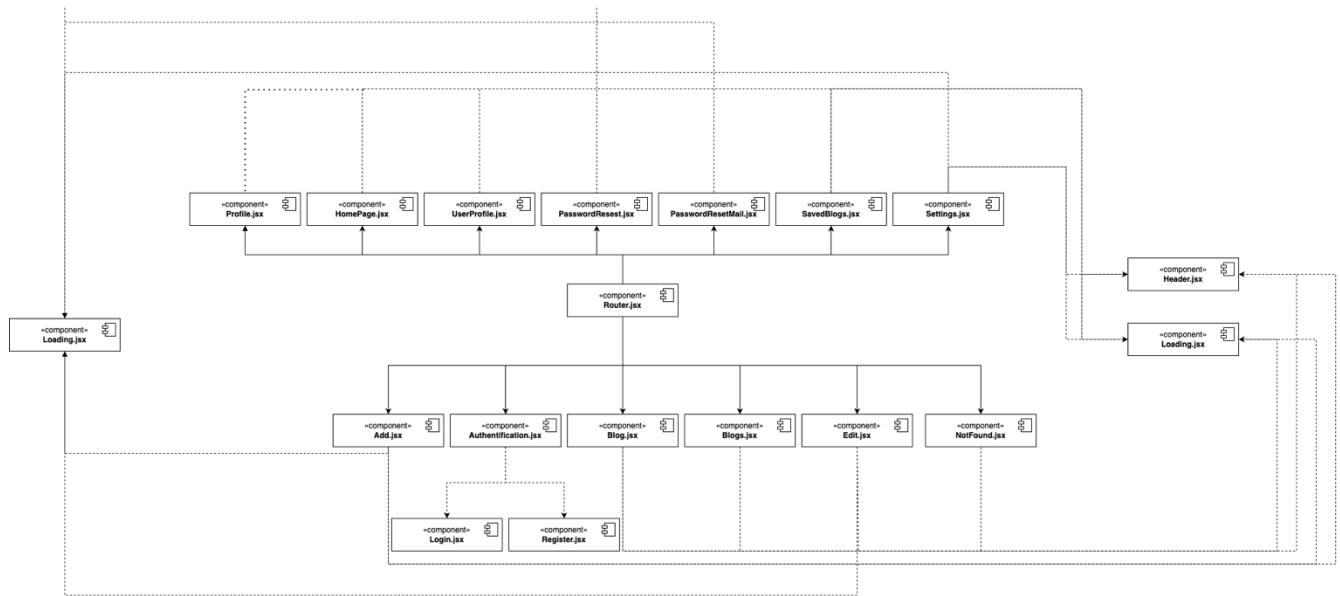
Šī diagramma attēlo datubāzes struktūru un tabulu nosaukumus, un savstarpējās attiecības (skatīt 8. attēlā).



8. attēls. *BlogIt* datubāzes komponenšu diagramma

4.2.2. PRIEKŠGALSISTĒMAS KOMPONENŠU DIAGRAMMA

Šī diagramma attēlo priekšgalsistēmas komponenšu savstarpējo mijiedarbību un atkarību (skatīt 9. attēlā).



9. attēls. *BlogIt* priekšgalsistēmas komponenšu diagramma

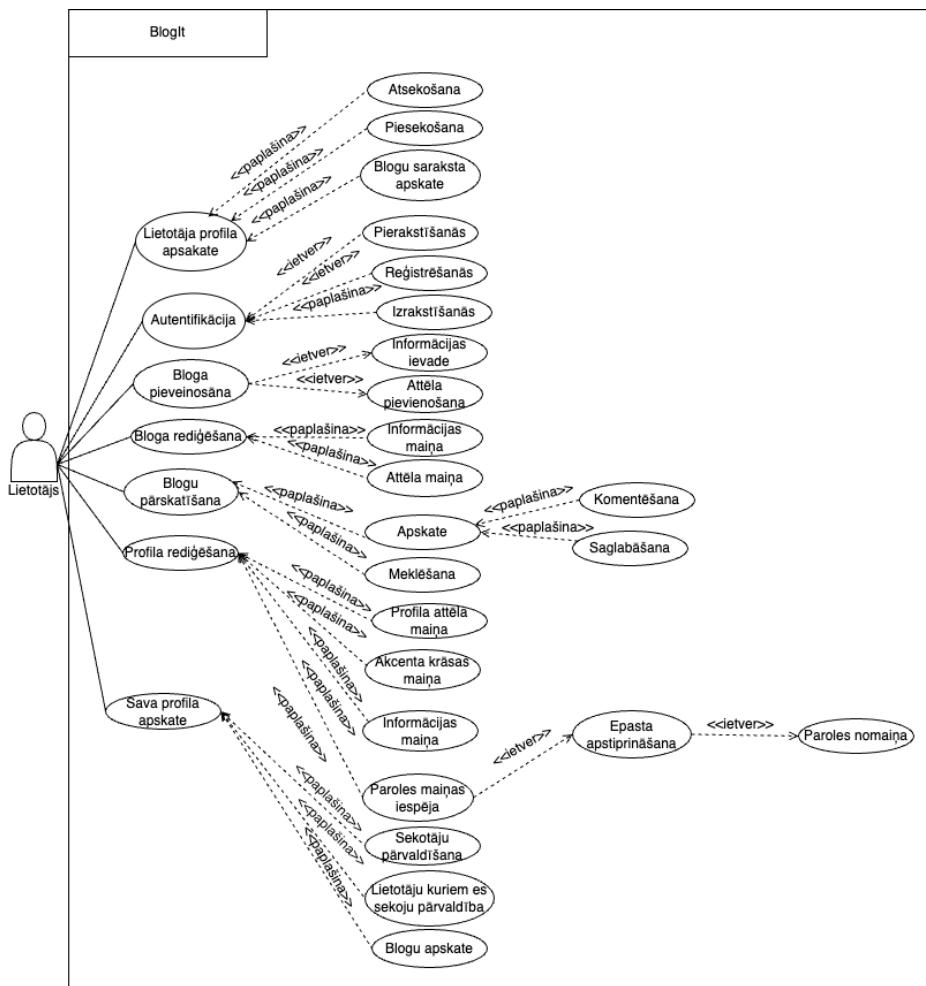
5. FUNKCIONĀLAIS UN DINAMISKAIS SISTĒMAS MODELIS

Funkcionālais un dinamisks sistēmas modelis apraksta datu plūsmas un lietotāja virzību tīmekļa vietnē. Šajā sadaļā tiks aprakstīti vairāki dinamiskie sistēmas modeļi:

- lietošanas gadījumu diagramma,
- sekvences diagrammas:
 - tīmekļa vietnes atvēršana,
 - pieslēgšanās tīmekļa vietnei,
 - reģistrācija tīmekļa vietnē,

5.1. LIETOŠANAS GADĪJUMU DIAGRAMMA

Lietošanas gadījumu diagramma (angļu v. *user case diagram*) ir UML diagrammas veids, kas palīdz attēlo sistēmas struktūru. Diagramma spēj atspoguļot lietotāja ceļu līdz kādai konkrētai funkcijai. Lietošanas gadījumu diagramma, palīdz sistēmas plānošanu pirms izstrādes posmā, kā piemēram, plānošanā, testēšanā un, protams, sniedz izpratni par tīmekļa vietnes izkārtojumu un iespējamajiem skatiem (skatīt 10. attēlā).



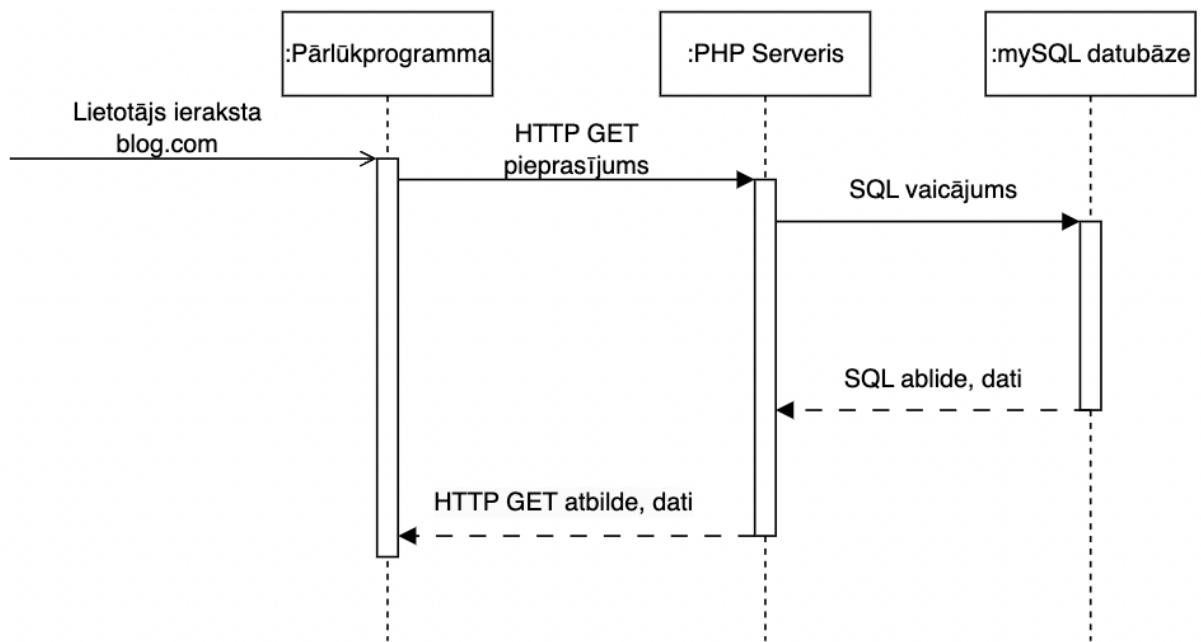
10. attēls. *BlogIt* lietošanas gadījumu diagramma

5.2. SEKVENCES DIAGRAMMAS

Sekvences diagrammas ir viens no UML diagrammas veidiem, kas dod ieskatu sistēmas un lietotāja savstarpējā mijiedarbībā. Tās tiek izmantotās programmatūras un sistēmas izstrādē lai parādītu šīs mijiedarbības, nosūtot ziņojumus un veicot informācijas apmaiņu.

5.2.1. TĪMEKĻA VIETNES ATVĒRŠANA

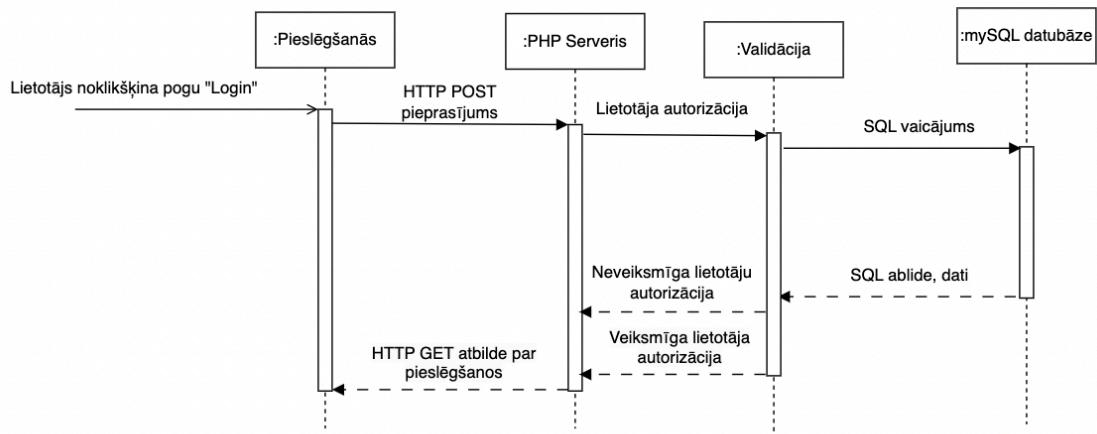
Šajā sekvences diagrammā tiek atspoguļots process jeb darbība, kad lietotājs pārlūkprogrammā sameklē un atver konkrēto tīmekļa vietni (skatīt 11. attēlā).



11.attēls *BlogIt* sekvences diagramma – tīmekļa vietnes atvēršana

5.2.2. PIELĒGŠANĀS TĪMEKĻA VIETNEI

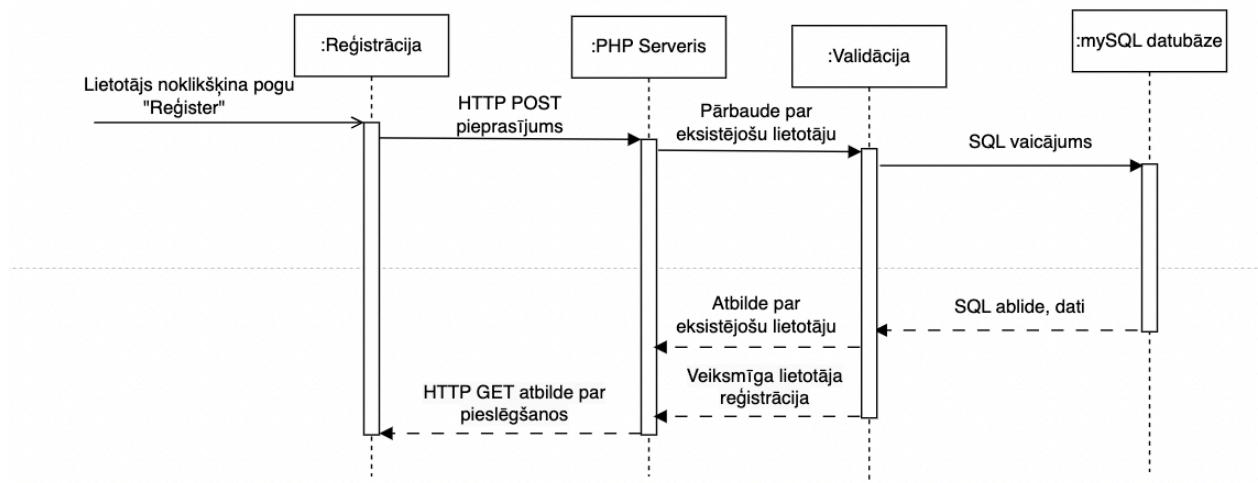
Šajā sekvoences diagrammā tiek atspoguļots process jeb darbība, kad lietotājs cenšas pieslēgties tīmekļa vietnei ar iepriekš izveidotu lietotāja kontu (skatīt 12. attēlā).



12. attēls. *BlogIt* sekvoences diagramma – pieslēgšanās tīmekļa vietnei

5.2.3. REĢISTRĀCIJA TĪMEKĻA VIETNĒ

Šajā sekvoences diagrammā tiek atspoguļots process jeb darbība, kad lietotājs vēlas reģistrēties tīmekļa vietnē, līdz ar to izveidojot lietotāja kontu (skatīt 13. attēlā).



13. attēls. *BlogIt* sekvoences diagramma – reģistrācija tīmekļa vietnē

5.3. REKOMENDĀCIJU ALGORITMS

Rekomendāciju algoritms ir algoritms, kas veido ieteikumu sarakstus lietotājam, vadoties pēc lietotāja vēstures. Tas uzlabo tīmekļa vietnes lietošanas pieredzi sniedzot lietotājam iespēju skatīt blogus, kas varētu interesēt tieši šim lietotājam. Izvēlējos izveidot šādu algoritmu, jo *For you* (latviešu v. *Priekš tevis*) ir izplatīts algoritms vairākās modernās un populārās tīmekļa vietnēs un lietotnēs, piemēram, *TikTok* un *YouTube*. Šajā sadaļā tiks aprakstīts kā strādā manis izveidotais algoritms, tas tiks paskaidrots ar piemēriem.

Piemēra konteksts:

Datubāzē atrodas šādas mirkļa birkas:

- car,
- supercar,
- home,
- racing,
- life,
- race,
- mylife,
- food.

Datubāzē atrodas blogs, kuram ir šādas mirkļa birkas :

- car,
- race,
- life.

Algoritma izpildes princips

- Lietotājs atver iepriekš minēto blogu tad, datubāzē tabulā *user_tags* tiek pievienotas šīs mirkļa birkas, kā tās ko lietotājs ir apskatījis, kā arī tabulā *views* tiek reģistrēts, ka lietotājs ir apskatījis konkrēto blogu.
- Lietotājs atverot skatu *For you* (latviešu v. *Priekš tevis*), var redzēt blogus kuri ir asociēti mirkļa birkām – **car, race, life**. Kā arī tiek paņemtas citas līdzīgās mirkļa birkas, minētajā gadījumā tās būs – **car, supercar, racing, race, life, mylife**.

Piezīmes

Rekomendāciju algoritms neatgriež iepriekš skatītos blogus, kā arī tas neatgriež blogus, kurus ievietojis pats lietotājs.

6. TESTĒŠANA

Testēšana ir viens no svarīgākajiem jebkuras sistēmas vai programmatūras izstrādes procesiem. Tā palīdz izprast sistēmas darbību un uzvedību atsevišķas situācijās, kā arī atrast un noteikt sistēmas darbības klūdas, saprast vājos punktus sistēmā. Galvenais mērķis testēšanā ir konstatēt klūdas un, kā arī veikt validāciju un verifikāciju. Testēšanas rezultātā sistēmas izstrādātājs spēj konstatēt funkcijas kurās nepieciešams uzlabot. Šajā sadaļā veikšu manuālo un automātisko testēšanu, lai pilnvērtīgi izprastu sistēmas darbību.

6.1. MANUĀLĀ TESTĒŠANA

Manuālajā testēšanā, sistēmas izstrādātājs pārskata un pārbauda tās sistēmas funkcijas, kurām nepieciešama informācijas ievade. Tādejādi tiek pārbaudīta validāciju un ziņojumu atgriešanas darbība. Veicot manuālo testēšanu tiek simulēta lietotāja darbība, veicot noteiktas funkcijas, kas palīdz noteikt iespējamās nepilnības.

6.1.1. PIESLĒGŠANĀS

Funkcija:

Lietotājam nepieciešama pieslēgšanās tīmekļa vietnei ar iepriekš izveidotu lietotāja kontu.

Testa apraksts:

Testēšanas procesā tiks testēti dažādi funkcijas izpildes scenāriji, tie tiks piefiksēti tabulā (skatīt 3. tabulā)

3. tabula

Pieslēgšanās testu tabula

| Kārtas nr. | Scenārijs | Sagaidāmais rezultāts | Iegūtais rezultāts |
|------------|---|--|---|
| 1. | Lietotājs aizpilda ievadlaukus atbilstoši validācijas prasībām. | Lietotājs veiksmīgi tiek pieslēgts konkrētajam kontam un pārvirzīts uz tīmekļa vietni. | Sagaidāmais rezultāts apstiprinājās, lietotājs tika pārvirzīts uz tīmekļa vietni. |
| 2. | Lietotājs ievada nepareizu pieslēgšanās informāciju. | Tiek izvadīts klūdu ziņojums par nepareizu e-pasta adresi vai paroli. | Sagaidāmais rezultāts apstiprinājās, tika izvadīts klūdu ziņojums “Email or password is incorrect”. |
| 3. | Lietotājs atstāj tukšumu kādā no ievadlaukiem. | Zem ievadlauka tiek izvadīts klūdu ziņojums | Sagaidāmais rezultāts apstiprinājās, tika izvadīts |

| | | | |
|----|--|--|---|
| | | par nepieciešamību aizpildīt konkrēto ievadlauku, piemēram ievadlauku “Email”. | kļūdu ziņojums “Email is required!”. |
| 4. | Lietotājs ievada informāciju kas ir mazāka par minimālo pieļaujamo rakstzīmju skaitu, piemēram, informāciju ar rakstzīmju skaitu: 1. | Zem ievadlauka tiek izvadīts kļūdu ziņojums par minimālo rakstzīmju skaitu konkrētajam ievadlaukam. | Sagaidāmais rezultāts apstiprinājās, tika izvadīts kļūdu ziņojums “This field must be atleast 5 characters long!”. |
| 5. | Lietotājs ievada informāciju kas ir lielāka par maksimaļo pieļaujamo rakstzīmju skaitu, piemēram, informāciju ar rakstzīmju skaitu: 100. | Zem ievadlauka tiek izvadīts kļūdu ziņojums par maksimālo rakstzīmju skaitu konkrētajam ievadlaukam. | Sagaidāmais rezultāts apstiprinājās, tika izvadīts kļūdu ziņojums “This field must not be greater than 20 characters!”. |

6.1.2. REĢISTRĀCIJA

Funkcija:

Lietotājam nepieciešama iespēja reģistrēties tīmekļa vietnei.

Testa apraksts:

Testēšanas procesā tiks testēti dažādi funkcijas izpildes scenāriji, tie tiks piefiksēti tabulā (skatīt 4. tabulā).

4. tabula

Reģistrācijas testu tabula

| Kārtas nr. | Scenārijs | Sagaidāmais rezultāts | Iegūtais rezultāts |
|------------|---|--|--|
| 1. | Lietotājs aizpilda ievadlaukus atbilstoši validācijas prasībām. | Lietotāja korts tiek veiksmīgi izveidots un tiek atgriezts veiksmes ziņojums par veiksmīgu reģistrāciju. | Sagaidāmais rezultāts apstiprinājās, tika atgriezts ziņojums “Your account is created!”. |
| 2. | Lietotājs ievada datubāzē eksistējošu e-pasta adresi. | Tiek izvadīts kļūdu ziņojums par to, ka e-pasts ir aizņemts. | Sagaidāmais rezultāts apstiprinājās, tika izvadīts |

| | | | |
|----|--|---|---|
| | | | kļūdu ziņojums “This email is taken”. |
| 3. | Lietotājs atstāj tukšumu kādā no ievadlaukiem. | Zem ievadlauka tiek izvadīts kļūdu ziņojums par nepieciešamību aizpildīt konkrēto ievadlauku, piemēram, ievadlauku “Email”. | Sagaidāmais rezultāts apstiprinājās, tika izvadīts kļūdu ziņojums “Email is required!”. |
| 4. | Lietotājs ievada informāciju kas ir mazāka par minimālo pieļaujamo rakstzīmju skaitu, piemēram, informāciju ar rakstzīmju skaitu: 1. | Zem ievadlauka tiek izvadīts kļūdu ziņojums par minimālo rakstzīmju skaitu konkrētajam ievadlaukam. | Sagaidāmais rezultāts apstiprinājās, tika izvadīts kļūdu ziņojums “This field must be atleast 5 characters long!”. |
| 5. | Lietotājs ievada informāciju kas ir lielāka par maksimālo pieļaujamo rakstzīmju skaitu, piemēram, informāciju ar rakstzīmju skaitu: 100. | Zem ievadlauka tiek izvadīts kļūdu ziņojums par maksimālo rakstzīmju skaitu konkrētajam ievadlaukam. | Sagaidāmais rezultāts apstiprinājās, tika izvadīts kļūdu ziņojums “This field must not be greater than 20 characters!”. |

6.1.3. BLOGA PIEVIENOŠANA

Funkcija:

Lietotājam nepieciešama iespēja pievienot blogu

Testa apraksts:

Testēšanas procesā tiks testēti dažādi funkcijas izpildes scenāriji, tie tiks piefiksēti tabulā (skatīt 5. tabulā).

5. tabula

Bloga pievienošanas testu tabula

| Kārtas nr. | Scenārijs | Sagaidāmais rezultāts | Iegūtais rezultāts |
|------------|---|---|---|
| 1. | Lietotājs aizpilda ievadlaukus atbilstoši validācijas prasībām. | Lietotāja korts tiek veiksmīgi izveidots un tiek atgriezts veiksmes | Sagaidāmais rezultāts apstiprinājās, tika atgriezts ziņojums “Blog created successfully”. |

| | | | |
|----|---|---|---|
| | | ziņojums par bloga pievienošanu. | |
| 2. | Lietotājs atstāj tukšumu kādā no ievadlaukiem. | Zem ievadlauka tiek izvadīts klūdu ziņojums par nepieciešamību aizpildīt konkrēto ievadlauku, piemēram, ievadlauku "Title". | Sagaidāmais rezultāts apstiprinājās, tika izvadīts klūdu ziņojums "The title field is required!". |
| 3. | Lietotājs ievada informāciju kas ir mazāka par minimālo pieļaujamo rakstzīmju skaitu, piemēram, informāciju ar rakstzīmju skaitu: 1. | Zem ievadlauka tiek izvadīts klūdu ziņojums par minimālo rakstzīmju skaitu konkrētajam ievadlaukam. | Sagaidāmais rezultāts apstiprinājās, tika izvadīts klūdu ziņojums "This field must be atleast 5 characters long!". |
| 4. | Lietotājs ievada informāciju kas ir lielāka par maksimālo pieļaujamo rakstzīmju skaitu, piemēram, informāciju ar rakstzīmju skaitu: 100 | Zem ievadlauka tiek izvadīts klūdu ziņojums par maksimālo rakstzīmju skaitu konkrētajam ievadlaukam. | Sagaidāmais rezultāts apstiprinājās, tika izvadīts klūdu ziņojums "This field must not be greater than 20 characters!". |
| 5. | Lietotājs nav izvēlējies bloga attēlu. | Zem attēla augšpielādes sekcijas tiek izvadīts par nepieciešamību izvēlēties attēlu. | Sagaidāmais rezultāts apstiprinājās, tika izvadīts klūdu ziņojums "The img field is required!". |

6.1.4. BLOGA REDIGĒŠANA

Funkcija:

Lietotājam nepieciešama iespēja rediģēt blogu.

Testa apraksts:

Testēšanas procesā tiks testēti dažādi funkcijas izpildes scenāriji, tie tiks piefiksēti tabulā (skatīt 6. tabulā).

Bloga redīgēšanas testu tabula

| Kārtas nr. | Scenārijs | Sagaidāmais rezultāts | Iegūtais rezultāts |
|------------|--|---|---|
| 1. | Lietotājs aizpilda ievadlaukus atbilstoši validācijas prasībām. | Lietotāja korts tiek veiksmīgi izveidots un tiek atgriezts veiksmes ziņojums par bloga pievienošanu. | Sagaidāmais rezultāts apstiprinājās, tika atgriezts ziņojums "Blog updated successfully!". |
| 2. | Lietotājs atstāj tukšumu kādā no ievadlaukiem. | Zem ievadlauka tiek izvadīts klūdu ziņojums par nepieciešamību aizpildīt konkrēto ievadlauku, piemēram, ievadlauku "Title". | Sagaidāmais rezultāts apstiprinājās, tika izvadīts klūdu ziņojums "The title field is required!". |
| 3. | Lietotājs ievada informāciju kas ir mazāka par minimālo pieļaujamo rakstzīmju skaitu, piemēram, informāciju ar rakstzīmju skaitu: 1. | Zem ievadlauka tiek izvadīts klūdu ziņojums par minimālo rakstzīmju skaitu konkrētajam ievadlaukam. | Sagaidāmais rezultāts apstiprinājās, tika izvadīts klūdu ziņojums "This field must be atleast 5 characters long!". |
| 4. | Lietotājs ievada informāciju kas ir lielāka par maksimālo pieļaujamo rakstzīmju skaitu, piemēram, informāciju ar rakstzīmju skaitu: 100. | Zem ievadlauka tiek izvadīts klūdu ziņojums par maksimālo rakstzīmju skaitu konkrētajam ievadlaukam. | Sagaidāmais rezultāts apstiprinājās, tika izvadīts klūdu ziņojums "This field must not be greater than 20 characters!". |
| 5. | Lietotājs nav izvēlējies bloga attēlu. | Zem attēla augšpielādes sekcijas tiek izvadīts par nepieciešamību izvēlēties attēlu. | Sagaidāmais rezultāts apstiprinājās, tika izvadīts klūdu ziņojums "The img field is required!". |

6.1.5. PROFILA REDIGĒŠANA

Funkcija:

Lietotājam nepieciešama iespēja pievienot redīgēt savu profila informāciju

Testa apraksts:

Testēšanas procesā tiks testēti dažādi funkcijas izpildes scenāriji, tie tiks piefiksēti tabulā (skatīt 7. tabulā).

7. tabula

Profila redīgēšanas testu tabula

| Kārtas nr. | Scenārijs | Sagaidāmais rezultāts | Iegūtais rezultāts |
|------------|--|--|---|
| 1. | Lietotājs aizpilda ievadlaukus atbilstoši validācijas prasībām. | Lietotāja korts tiek veiksmīgi izveidots un tiek atgriezts veiksmes ziņojums par bloga pievienošanu. | Sagaidāmais rezultāts apstiprinājās, tika atgriezts ziņojums “Profile updated successfully!”. |
| 2. | Lietotājs atstāj tukšumu kādā no ievadlaukiem. | Zem ievadlauka tiek izvadīts kļūdu ziņojums par nepieciešamību aizpildīt konkrēto ievadlauku, piemēram, ievadlauku “Name”. | Sagaidāmais rezultāts apstiprinājās, tika izvadīts kļūdu ziņojums “The title field is required!”. |
| 3. | Lietotājs ievada informāciju kas ir mazāka par minimālo pieļaujamo rakstzīmju skaitu, piemēram, informāciju ar rakstzīmju skaitu: 1. | Zem ievadlauka tiek izvadīts kļūdu ziņojums par minimālo rakstzīmju skaitu konkrētajam ievadlaukam. | Sagaidāmais rezultāts apstiprinājās, tika izvadīts kļūdu ziņojums “This field must be atleast 5 characters long!”. |
| 4. | Lietotājs ievada informāciju kas ir lielāka par maksimālo pieļaujamo rakstzīmju skaitu, piemēram, informāciju ar rakstzīmju skaitu: 100. | Zem ievadlauka tiek izvadīts kļūdu ziņojums par maksimālo rakstzīmju skaitu konkrētajam ievadlaukam. | Sagaidāmais rezultāts apstiprinājās, tika izvadīts kļūdu ziņojums “This field must not be greater than 20 characters!”. |

| | | | |
|----|---|--|--|
| 5. | Lietotājs ievada datubāzē eksistējošu e-pasta adresi, kas nav saistīta ar viņa lietotāja kontu. | Tiek izvadīts kļūdu ziņojums par to, ka e-pasts ir aizņemts. | Sagaidāmais rezultāts apstiprinājās, tika izvadīts kļūdu ziņojums “This email is taken”. |
|----|---|--|--|

6.2. AUTOMĀTISKĀ TESTĒŠANA

Automātiskajā testēšanā, sistēmas izstrādātājs pārskata un pārbauda visas galvenās sistēmas funkcijas un to darbību. Tādejādi tiek pārbaudīta validāciju, ziņojumu un datu atgriešanas darbība, kā arī tiek pārbaudīts vai tiek izpildīta pieprasītā darbība.

Automātiskajā testēšanā izmantošu *Laravel* iebūvēto testēšanu *Feature* testi var pārbaudīt lielāku koda daļu, tostarp to, kā vairāki objekti mijiedarbojas viens ar otru, vai pat pilnu HTTP

pieprasījumu JSON galapunktam. Šāda veida testi nodrošina vislielāko pārliecību, ka kopumā darbojas, kā paredzēts (Otwell, 2024a).

Testa piemērs (skatīt 16. attēlā)

Tiek testēts vai lietotājs ievadot pareizo pieslēgšanās informāciju spēj pieslēgties savam lietotāja kontam, ja lietotājs pieslēdzies veiksmīgi, tad tiek atgriezts status 200, līdz ar to tests ir veiksmīgs (skatīt 14. attēlā), ja status nav vienāds ar 200 tas nozīmē ka tests nav izpildījies veiksmīgi (skatīt 15. attēlā).

```
PASS Tests\Feature\LoginTest
✓ it logs in user
```

14. attēls. Veiksmīgi izpildīta testa piemērs

```
FAIL Tests\Feature\LoginTest
✗ it logs in user
```

15. attēls. Neveiksmīgi izpildīta testa piemērs

```

/** @test */
new *
public function it_logs_in_user()
{
    $str = Str::random( length: 5 );
    $password = "password123";
    $create = [
        'email' => $str.'@example.com',
        'password' => Hash::make($password),
        'name' => 'John',
        'surname' => 'Doe',
    ];
    User::factory()->create($create);

    $loginData = [
        'email' => $str.'@example.com',
        'password' => $password,
    ];

    $response = $this->postJson( uri: 'api/v1/login' , $loginData);

    $response->assertStatus( status: 200);
}

```

16. attēls. Testa piemērs kodā

6.2.1. PIESLĒGŠANĀS TESTI

Apraksts

Tiks pārbaudītas *AuthenticationController.php* pieslēgšanās funkcijas un to darbība, kā arī tiks atspoguļots ekrānšāviņš no testu palaišanas terminālī (skatīt 17. attēlā).

Testējamās funkcijas un validācijas

- Vai lietotājs spēs pieslēgties tīmekļa vietnei ar derīgiem pieslēgšanās datiem.
- Vai tiek nolasīts tas, ka lietotājs mēģina pieslēgties ar nederīgiem datiem.
- Vai tiek nolasīts tas, ka ievadlauki ir tukši, atgriežot kļūdu ziņojumu.
- Vai tiek pārbaudīts e-pasta adreses formāts.

Testa rezultāti

```

PASS Tests\Feature\LoginTest
✓ it logs in user
✓ it returns message if entered incorrect login info
✓ it requires all fields to be filled
✓ it validates email format

```

17. attēls. Pieslēgšanās testu rezultāti

6.2.2. REĢISTRĀCIJAS TESTI

Apraksts

Tiks pārbaudītas *AuthenticationController.php* reģistrācijas funkcijas un to darbība, kā arī tiks atspoguļots ekrānšāviņš no testu palaišanas terminālī (skatīt 18. attēlā).

Testējamās funkcijas un validācijas

- Vai lietotājs spēs izveidot lietotāja kontu tīmekļa vietnē ar derīgiem datiem.
- Vai tiek nolasīts tas, ka lietotājs mēģina reģistrēt jau esošu e-pasta adresi.
- Vai tiek nolasīts tas, ka ievadlauki ir tukši, atgriežot kļūdu ziņojumu.
- Vai tiek pārbaudīts e-pasta adreses formāts.

Testa rezultāti

```
PASS Tests\Feature\RegisterTest
✓ it registers a new user with valid data
✓ it returns error if email is already taken
✓ it requires all fields to be filled
✓ it validates email format
```

18. attēls. Reģistrācijas testu rezultāti

6.2.3. BLOGA FUNKCIJU TESTI

Apraksts

Tiks pārbaudītas *BlogController.php* funkcijas un to darbība, kā arī tiks atspoguļots ekrānšāviņš no testu palaišanas terminālī (skatīt 19. attēlā).

Testējamās funkcijas un validācijas

- Vai tiek atgriezti visi ieraksti.
- Vai tiek atgriezti ieraksti, meklējot pēc konkrēta *id*.
- Vai tiek atgriezti ieraksti, meklējot pēc konkrētas kategorijas.
- Vai tiek atgriezti ieraksti, kas atbilst personām kurām seko lietotājs.
- Vai tiek atgriezti ieraksti, ko sniedz rekomendācijas *For you* (latviešu val. *Priekš tevis*).
- Vai tiek izdzēsts ieraksts.
- Vai ir iespēja saglābāt ierakstu.
- Vai ir iespēja rediģēt ierakstu.
- Vai ir iespēja izveidot ierakstu.

Testa rezultāti

```
PASS Tests\Feature\BlogTest
✓ it return all blogs
✓ it return blogs with certain id
✓ it return blogs with certain category
✓ it return blogs for following profiles
✓ it return blogs for you page
✓ it deletes blog
✓ it saves blog
✓ it edit blog
✓ it craete blog
```

19. attēls. Bloga kontroliera funkcijas

6.2.4. LIETOTĀJKONTA FUNKCIJU TESTI

Apraksts

Tiks pārbaudītas *AutheticationController.php* profila funkcijas un to darbība, kā arī tiks atspoguļots ekrānšāviņš no testu palaišanas terminālī (skatīt 20. attēlā).

Testējamās funkcijas un validācijas

- Vai lietotājs spēj rediģēt konta informāciju.
- Vai tiek pārbaudīts e-pasta adreses formāts.
- Vai tiek aizsūtīts paroles maiņas vēstule uz norādīto e-pastu pēc pieprasījuma.
- Vai lietotājs spēj izrakstīties no konta.

Testa rezultāti

```
PASS Tests\Feature\ProfileTest
✓ it updates profile info
✓ it validates profile update email format
✓ it validates password email sending email format
✓ it sends password reset email
✓ it logs out user
```

20. attēls. Profila funkciju testu rezultāti

6.2.5. KOMENTĀRU FUNKCIJU TESTI

Apraksts

Tiks pārbaudītas *CommentsController.php* funkcijas un to darbība, kā arī tiks atspoguļots ekrānšāviņš no testu palaišanas terminālī (skatīt 21. attēlā).

Testējamās funkcijas un validācijas

- Vai lietotājs spēj pievienot komentāru.
- Vai ir iespēja iegūt visus ieraksta komentārus.

Testa rezultāti

```
PASS Tests\Feature\CommentTest
✓ it inserts comment
✓ it return all comments
```

21. attēls. Komentāru kontroliera funkciju testi

6.2.6. SEKOTĀJU FUNKCIJU TESTI

Apraksts

Tiks pārbaudītas *FollowersController.php* funkcijas un to darbība, kā arī tiks atspoguļots ekrānšāviņš no testu palaišanas terminālī (skatīt 22. attēlā).

Testējamās funkcijas un validācijas

- Vai lietotājs spēj sākt sekot un pārstāt sekot kādam lietotājam.
- Vai lietotājs spēj noņemt kādu no sava sekotāju saraksta.

Testa rezultāti

```
PASS Tests\Feature\FollowerTest
✓ it follows and unfollows user
✓ it removes follower
```

22.attēls Sekotāju testa rezultāti

6.3. DARBĪBAS TESTĒŠANA PĀRLŪKPROGRAMMĀS

Darbības testēšana dažādās pārlūkprogrammās ir nepieciešama, lai uzlabotu tīmekļa vietnes izmantošanas pieredzi ikvienam lietotājam. Statistika liecina ka no 2023. gada augustā, visvairāk izmantotā pārlūkprogramma bija *Google Chrome*, kuru izmantoja 63,56% lietotāju, nākamā bija *Safari* ar 19,84%, un kā trešā bija *Microsoft Edge*, kuru izmantoja 5,43% lietotāju. Tātad testa rezultāti tiks apkopoti par *Google Chrome* (skatīt 8. tabulā), *Safari* (skatīt 9. tabulā) un *Microsoft Edge* (skatīt 10. tabulā) (Statista, 2024).

8. tabula

Google Chrome testu tabula

| Problēmas apraksts | Iespējamais risinājums |
|---|------------------------|
| Netika konstatētas lietotnes darbības problēmas | Nav nepieciešams |

Safari testu tabula

| Problēmas apraksts | Iespējamais risinājums |
|---|------------------------|
| Netika konstatētas lietotnes darbības problēmas | Nav nepieciešams |

Microsoft Edge testu tabula

| Problēmas apraksts | Iespējamais risinājums |
|---|------------------------|
| Netika konstatētas lietotnes darbības problēmas | Nav nepieciešams |

Secinājumi

Testu rezultātā secināju ka tīmekļa vietnes darbībā netika konstatētas problēmas, izmantojot jebkuru no iepriekš minētajām pārlūkprogrammām. Tas palielina lietotāju klāstu, ja tīmekļa vietne strādās nevainojami, lietotāja iemīļotajā pārlūkprogrammā.

7. LIETOTĀJA CEĻVEDIS

Lietotāja ceļveža sadaļa sistēmas izstrādātājs apraksta kā lietot tīmekļa vietni un tajā piedāvātās funkcijas. Tādejādi gadījumā ja lietotājam rodas kādi sarežģījumi vai lietotājs nesaproš kā pareizi lietot kādu piedāvāto funkciju, tas varētu apskatīt kā to izdarīt. Lietotāja ceļvedim jābūt viegli saprotama, intuitīvam un ar skaidrām, secīgām norādēm, kā paveikt katru darbību.

7.1. TĪMEKĻA VIETNES ATVĒRŠANA

Apraksts

Šajā sadaļa tiks paskaidrots, kas nepieciešams, lai atvērtu tīmekļa vietni, izskaidrojot to ar izpildes soliem un vizuālu piemēru.

Izpildes soli

Lai atvērt tīmekļa vietni, lietotājam nepieciešams:

- atvērt kādu pārlūkprogrammu (skatīt 23. attēlā),



23. attēls. Pārlūkprogrammu piemēri

- mēklēšanas joslā ievadīt saiti: <https://into.id.lv/> (skatīt 24. attēlā)



24. attēls. Meklēšanas josla pārlūkprogrammā *Google Chrome*

- izvēlēties pogu “Enter”, lai atvērtu tīmekļa vietni.

7.2. PIESLĒGŠANĀS AR LIETOTĀJA KONTU

Apraksts

Šajā sadaļā tiks paskaidrots kas nepieciešams lai atvērtu, lai lietotājs spētu pieslēgties tīmekļa izskaidrojot to ar izpildes soļiem un vizuālu piemēru (skatīt 29. attēlā).

Atverot tīmekļa vietni kreisajā pusē atrodas ievades lauki un poga *Login* (latviešu val. *Pieslēgties*) ar kuras palīdzību, ja ir aizpildīti ievadlauki, var pieslēgties tīmekļa vietnei ar iepriekš izveidoto kontu. Virs ievadlaukiem atrodas poga *Register* (latviešu v. *Reģistrēties*), kuru nospiežot var pārslēgties uz reģistrācijas skatu.

Izpildes soļi

Lai pieslēgtos tīmekļa vietnei ar izveidoto kontu, lietotājam nepieciešams:

- ievadīt ievadlaukā *Email* (latviešu val. *E-pasts*), e-pasta adresi (skatīt 25. attēlā),



25. attēls. E-pasta ievades paraugs

- ievadīt ievadlaukā *Password* (latviešu val. *Parole*), konkrētajam kontam attiecīgo paroli (skatīt 26. attēlā),



26. attēls. Paroles ievades paraugs

- uzklikšķināt uz pogas *Login* (latviešu val. *Pieslēgties*) (skatīt 27. attēlā), ja ievadītā informācija būs nepareiza, tiks izvadīts kļūdu ziņojums (skatīt 28. attēlā), ja konta informācija būs pareiza, lietotājs tiks pārvirzīts uz galveno skatu.

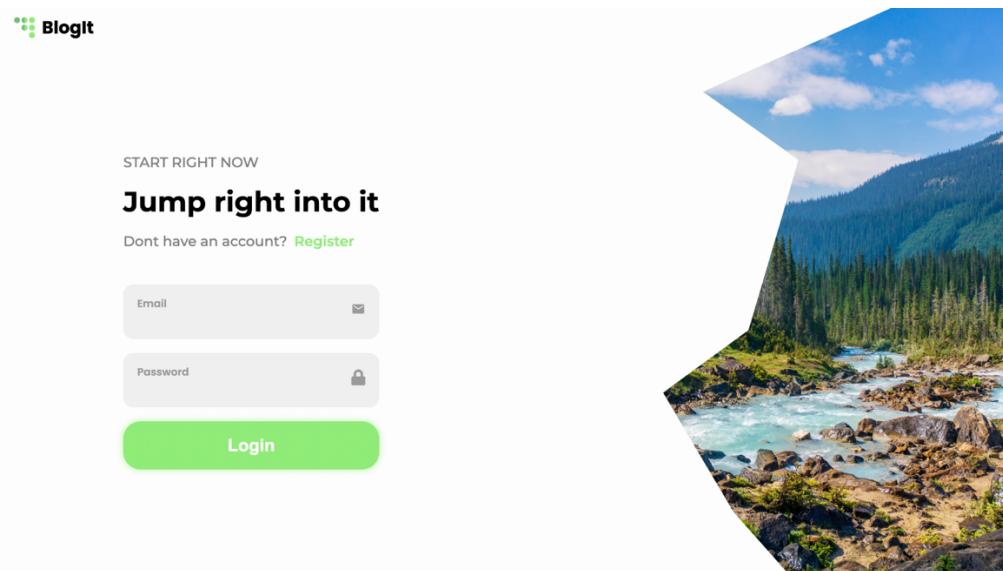
Login

27. attēls. Pieslēgšanās poga

Email or password is incorrect

28. attēls. Kļūdu ziņojums par nepareizi ievadītu konta informāciju

Vizuālais piemērs



29. attēls. Pieslēgšanās skats

7.3. LIETOTĀJA KONTA REGISTRĀCIJA

Apraksts

Šajā sadaļā tiks paskaidrots kas nepieciešams lai atvērtu, lai lietotājs spētu reģistrēties tīmekļa vietnē, izskaidrojot to ar izpildes soļiem un vizuālu piemēru.

Atverot reģistrācijas skatu, kreisajā pusē atrodas ievades lauki un poga *Register* (latviešu val. *Reģistrēties*), ar kuras palīdzību, ja ir aizpildīti ievadlauki, var pieslēgties tīmekļa vietnei ar iepriekš izveidotu kontu. Virs ievadlaukiem atrodas poga *Login* (latviešu val. *Pieslēgties*) kuru nospiežot var pārslēgties uz reģistrācijas skatu (skatīt 37. attēlā un 38. attēlā).

Izpildes soļi

Lai pieslēgtos tīmekļa vietnei ar izveidoto kontu, lietotājam nepieciešams:

- ievadīt ievadlaukā *Name* (latviešu val. *Vārds*), savu vārdu (skatīt 30. attēlā),

A screenshot of a user interface element showing an input field for 'Name'. The placeholder text 'Name' is in green. A user has entered 'Jānis'. This entry is highlighted with a green rectangular border, while the rest of the field is grey.

30. attēls. Vārda ievades paraugs

- ievadīt ievadlaukā *Surname* (latviešu val. *Uzvārds*), savu uzvārdu (skatīt 31. attēlā),

A screenshot of a user interface element showing an input field for 'Surname'. The placeholder text 'Surname' is in green. A user has entered 'Krūmiņš'. This entry is highlighted with a green rectangular border, while the rest of the field is grey.

31. attēls. Uzvārda ievades paraugs

- ievadīt ievadlaukā *Email* (latviešu val. *E-pasts*), e-pasta adresi (skatīt 32. attēlā),

A screenshot of a registration form. On the left, there is a light gray input field with the placeholder 'Email' and a small envelope icon. To its right, a green-bordered rounded rectangle contains the text 'Email' above the address 'janis@gmail.com'.

32. attēls. E-pasta ievades paraugs (Reģistrācija)

- ievadīt ievadlaukā *Password* (latviešu val. *Parole*), sevis izvēlēto paroli ar vismaz 8 rakstzīmēm (skatīt 33. attēlā),

A screenshot of a registration form. On the left, there is a light gray input field with the placeholder 'Password' and a lock icon. To its right, a green-bordered rounded rectangle contains the text 'Password' above a redacted password value consisting of five dots '.....'.

33. attēls. Paroles ievades paraugs

- noklikšķināt pogu *Register* (latviešu val. *Reģistrēties*) (skatīt 34. attēlā), tiks izvadīts veiksmes ziņojums par konta izveidi (skatīt 35. attēlā), ja lietotājs vēlēsies, būs iespēja uzreiz pieslēgties, nospiežot pogu *Login* (latviešu val. *Pieslēgties*) (skatīt 36. attēlā).

Register

34. attēls. Reģistrācijas poga

Your account is created

35. attēls. Veiksmes ziņojums par konta izveidi

Login

36. attēls. Pieslēgšanās poga

FEW STEPS

Create an account

Have an account? [Login](#)

Name

Surname

Email

Password

[Register](#)



37. attēls. Reģistrācijas skats

Your account is created 

Create another one? [Register](#)

Name Jānis

Surname Krūmiņš

Email janis@gmail.com

Password 

[Login](#)



38. attēls. Reģistrācijas skats (veiksmīgas konta izveides gadījumā)

7.4. NAVIGĀCIJAS JOSLAS IZMANTOŠANA

Apraksts

Šajā sadaļā tiks paskaidrots, kā izmantot navigācijas joslu, izskaidrojot to ar izpildes soļiem un vizuāliem piemēriem. Pieslēdzoties tīmekļa vietnei, kreisajā pusē ir redzama navigācijas josla ar navigācijas pogām, ar kuru palīdzību iespējams pārslēgties starp skatiem.

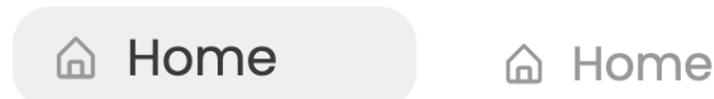
Aktīva skata navigācijas joslas poga tiek iekrāsota ar pelēku fonu un neaktīvā skata navigācijas joslas poga ir bez iekrāsotā fona (skatīt 40. attēlā).

Piezīmes

Ja lietotājs tīmekļa vietnei pieslēdzies izmantojot mobilo ierīci, tad navigācijas joslu tas var atvērt nospiežot uz izvēlnes ikonas (skatīt 39.attēlā)



39.attēls Izvelnes ikona



40. attēls Navigācijas joslas pogas

Izpildes soļi

Lai atvērtu *Home* (latviešu val. *Mājas*) skatu, jānoklikšķina uz pogas *Home*, tad lietotājs tiks pārvirzīts uz šo skatu (skatīt 41. attēlā).

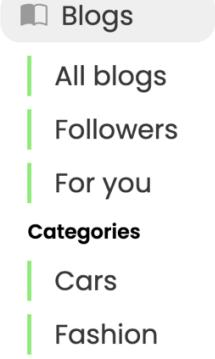


41. attēls. Navigācijas joslas poga *Home*

Lai atvērtu *Blogs* (latviešu val. *Emuāru*) skatu, jānoklikšķina uz pogas, *Blogs* (skatīt 42. attēlā), tad atvēries papildu sadaļa, kurā jāizvēlas kādu no attēlotajām kategorijām (skatīt 43. attēlā), tad lietotājs tiks pārvirzīts uz *Blog* skatu kurā attēloti šīs kategorijas blogi.



42. attēls. Navigācijas joslas poga *Blogs*



Blogs

All blogs

Followers

For you

Categories

Cars

Fashion

43. attēls. Blogu kategoriju pogas

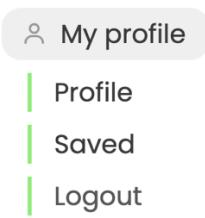
Lai atvērtu *New Blog* (latviešu val. *Jauns emuārs*) skatu, un jānoklikšķina uz pogas *New blog*, tad lietotājs tiks pārvirzīts uz šo skatu (skatīt 44. attēlā).



+ New blog

44. attēls. Navigācijas joslas *New blog*

Lai atvērtu *Profile* (latviešu val. *Profils*) skatu, kurā var apskatīt sevis pievienotus blogus, sekotājus un profilus kam seko lietotājs, jānoklikšķina uz pogas *My profile* (skatīt 45. attēlā), tad tiks parādīta papildsekcija (skatīt 46. attēlā), kurā jānoklikšķina uz pogas *Profile*, tad lietotājs tiks pārvirzīts uz profila skatu (skatīt 47. attēlā).



👤 My profile

45. attēls. Navigācijas joslas *My profile* poga

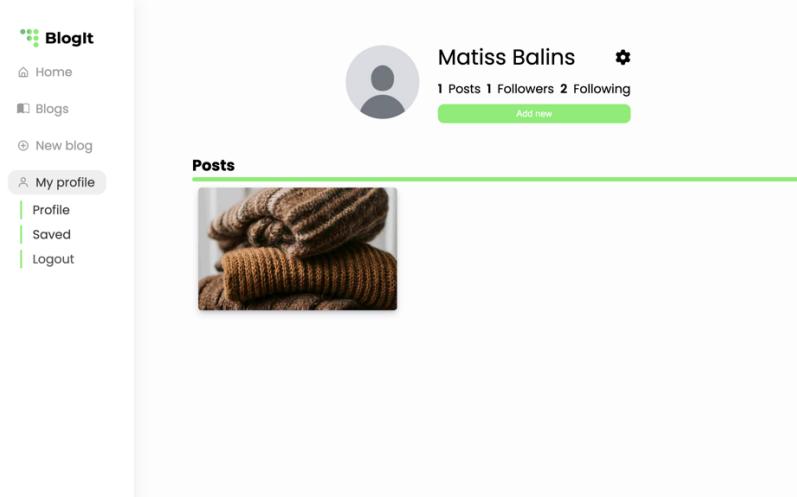
👤 My profile

Profile

Saved

Logout

46. attēls. *My profile* papildus sekcija

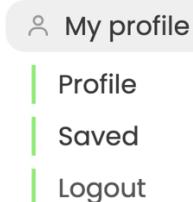


47. attēls. Profila skats

Lai atvērtu *Saved* (latviešu val. *Saglabātie*) skatu, kurā var apskatīt sevis saglabātos blogu ierakstus, jānoklikšķina uz pogas *My profile* (skatīt 48. attēlā), tad atvērsies papildus sekcija ar apakšnodaļām (skatīt 49. attēlā), kurā jānoklikšķina uz pogas *Saved* (latviešu val. *Saglabātie*) (skatīt 50. attēlā), tad lietotājs tiks pārvirzīts uz skatu ar viņa pievienotiem blogiem.

👤 *My profile*

48. attēls. Navigācijas joslas *My profile* poga

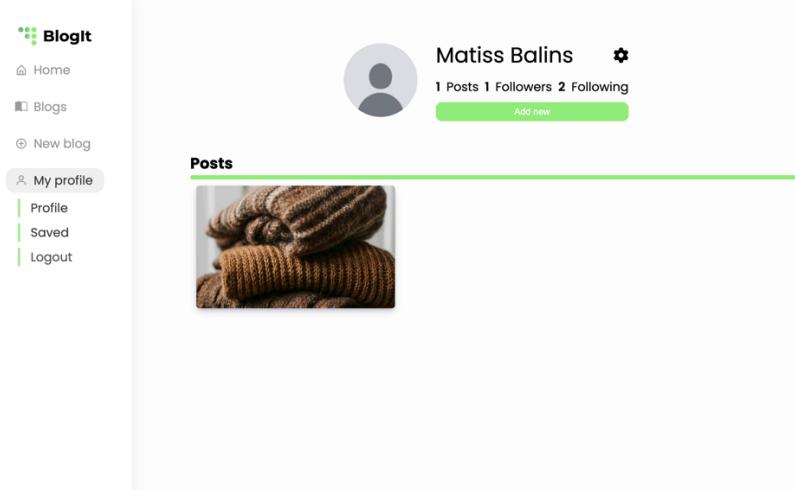


49. attēls. *My profile* papildus sekcija

| *Saved*

50. attēls. Poga *Saved*

Lai atvērtu *Settings* (latviešu val. *Iestatījumi*) skatu, kurā var apskatīt profila informāciju un iestatījumus, Jāatver profila skats (skatīt 51. attēlā), tad jānoklikšķina uz zobraza ikonas (skatīt 52. attēlā), tad lietotājs tiks pārvirzīts uz *Settings* (latviešu val. *Iestatījumi*) skatu.



51. attēls. Profila skats

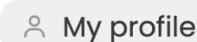


52. attēls. Zobrata ikona

Lai izrakstītos no lietotāja kontu un pārietu uz pieslēgšanās skatu, jānoklikšķina uz pogas *My profile* (skatīt 53. attēlā), tad atvērsies papildus sekcija ar apakšnodaļām (skatīt 54. attēlā), kurā jānoklikšķina uz pogas *Logout* (latviešu val. *Izrakstīties*) (skatīt 55.attēlā), tad lietotājs tiks izrakstīts un pārvirzīts uz pieslēgšanās skatu.



53. attēls. Navigācijas joslas *My profile* poga



- Profile
- Saved
- Logout

54. attēls. *My profile* papildus sekcija

Logout

55. attēls. poga *Logout*

7.5. BLOGU SARAKSTA APSKATE

Apraksts

Šajā sadaļā tiks paskaidrots kā lietotājs var apskatīt blogu sarakstu, *Saved* (latviešu val. *Saglabātie*) un *Blogs* (latviešu val. *Emuāru*), *For you* (latviešu val. *Tev*), *Followers* (latviešu val. *Sekotāji*), skatos, meklēt blogu un nonākt pie bloga sīkākas informācijas, izskaidrojot to ar izpildes soļiem un vizuāliem piemēriem (skatīt 60. attēlā un 61. attēlā).

Atverot blogu saraksta skatu, augšējā daļā ir redzama meklēšanas josla, un zem tās ir izvietoti visi atbilstošie blogi.

Piezīmes

Skatā *For you* (latviešu val. *Priekš tevis*), tiks attēloti blogi tikai tad, ja lietotājs iepriekš būs apskatījis kādu blogu, kam ir mirkļa birka, jo šajā skatā informācija tiek attēlota izmantojot rekomendāciju algoritmu, kas palīdz atrast lietotājam interesējošos blogus.

Skatā *Followers* (latviešu val. *Sekotāji*), tiks attēloti blogi, tikai tad, ja lietotājs būs sācis sekot kādam citam lietotāja profilam, un ja šajā lietotāja profilā būs ievietoti blogi.

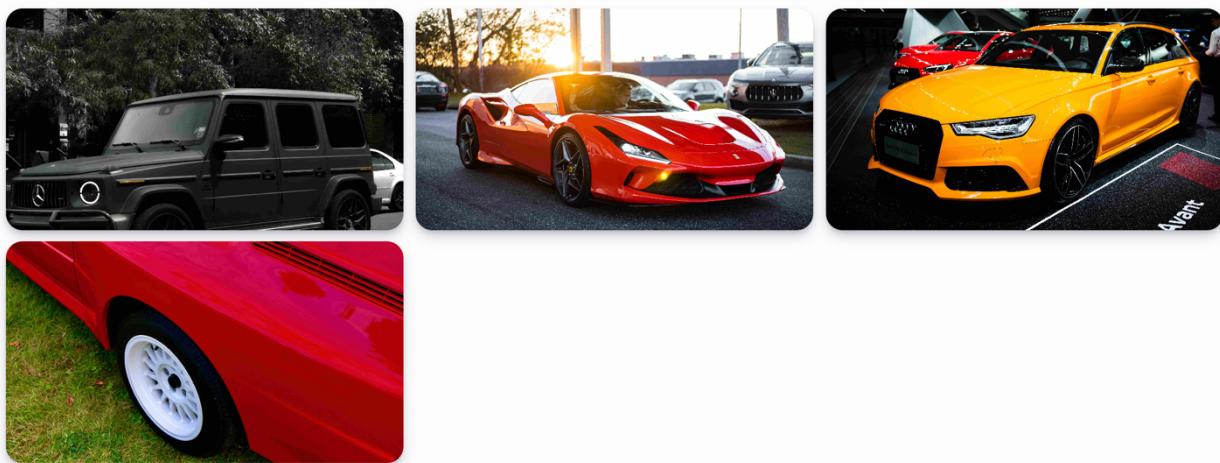
Ja netiek atrasts neviens ieraksts pēc dotajiem kritērijiem, tiek atgriezts ziņojums par to, ka netika atrast neviens blogs (skatīt 56. attēlā)

 No blog posts found

56. attēls. Ziņojums par neatrastiem ierakstiem

Izpildes soļi

Lai apskatītu blogu sarakstu nepieciešams atvērt blogu skatu, tur tiks attēloti blogi kā taisnstūrveida lauki ar attēlu (skatīt 57. attēlā).



57. attēls. Blogu saraksts

Lai apskātu bloga nosaukumu un autoru, ar kurSORU nepieciešams uziet uz vēlamā bloga, tad šajā kvadrātveida laukā parādīsies bloga autora vārds un profila attēls kā arī tiks parādīts bloga nosaukums (skatīt 58. attēlā).



58. attēls. Bloga sīkākas informācijas apskate

Lai apskatītu bloga saturu, ar kurSORU nepieciešams uziet uz vēlamā bloga, tad šajā kvadrātveida laukā parādīsies bloga autora vārds un profila attēls kā arī tiks parādīts bloga nosaukums, nepieciešams uzklikšķināt uz šī bloga (skatīt 59. attēlā), tad lietotājs tiks pārvirzīts uz bloga skatu.

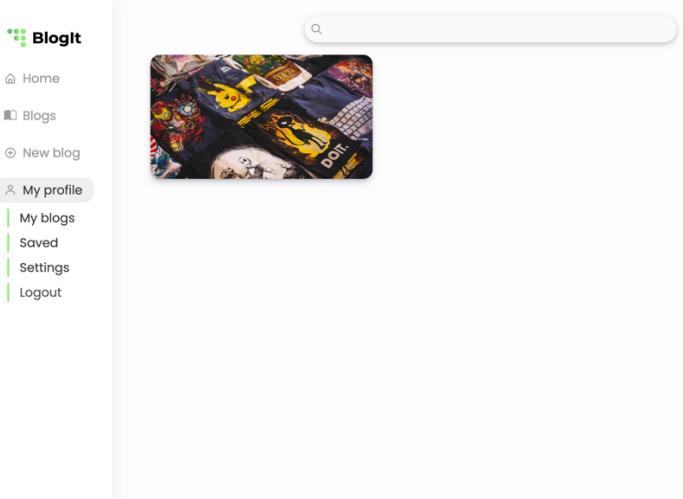


59. attēls. Bloga saturs apskate

Vizuāls piemērs

Screenshot of a blog platform interface showing a sidebar with categories like Cars and Fashion, and a main area displaying car images.

60. attēls. Blogu saraksta skats



61.attēls. Saglabāto blogu saraksta skats

7.6. KOMENTĀRU IEVADE

Apraksts

Šajā sadaļā tiks paskaidrots kā lietotājs var komentēt kādu attiecīgo blogu, to paskaidrojot ar izpildes soļiem un vizuāliem piemēriem.

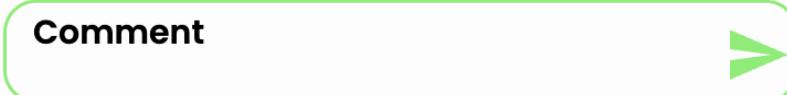
Izpildes soļi

Lai ievadītu komentētu, lietotājam nepieciešams :

- atvērt blogu apskates skatu,
- ierakstīt komentāru ievades laukā (skatīt 62. attēlā),
- noklikšķināt uz sūtīšanas ikonas (skatīt 62. attēlā),
- tad komentārs tiks parādīts saraksta augšpusē (skatīt 63. attēlā)



62.attēls Komentāru ievades lauks



Matisss
New comment

63.attēls Komentāru ievade

7.7. MANU BLOGU SARAKSTA APSKATE

Apraksts

Šajā sadaļā tiks paskaidrots kā lietotājs var apskatīt sevis ievietoto blogu un nonākt bloga redīģēšanas skatā to paskaidrojot ar izpildes soliem un vizuāliem piemēriem (skatīt 66. attēlā).

Izpildes soli

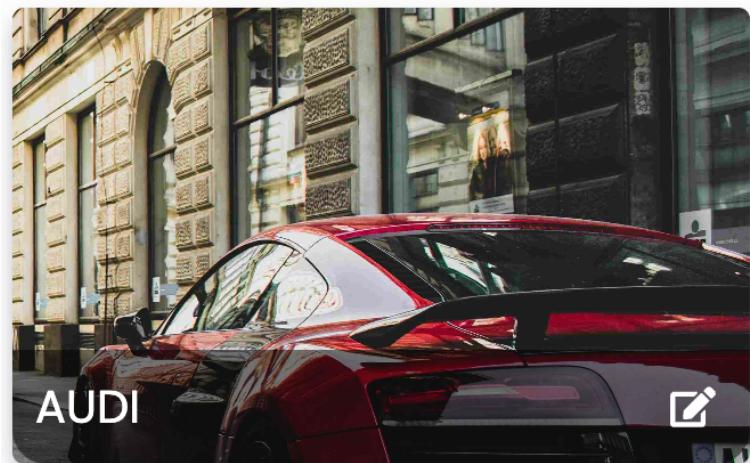
Lai apskatītu savu blogu sarakstu nepieciešams atvērt manu profila skatu, tur tiks attēloti lietotāja ievietotie blogi kā taisnstūrveida lauki ar attēlu (skatīt 64. attēlā).

Posts



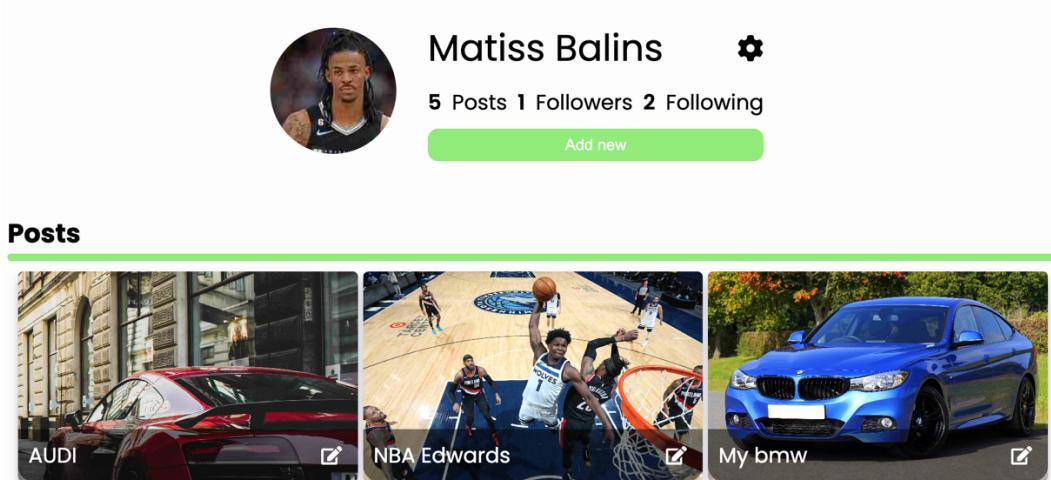
64. attēls. Manu blogu saraksts

Lai apskatītu bloga nosaukumu, kā arī ja vēlas to redīģēt, ar kurSORU nepieciešams uziet uz vēlamā bloga, tad šajā kvadrātveida laukā parādīsies bloga nosaukums kā arī tiks parādīta redīģēšanas ikona, ja lietotājs vēlas redīģēt šī bloga informāciju, tad lietotājam jāuzklikšķina uz šīs taisnstūrveida, tad tas tiks pārvirzīts uz bloga redīģēšanas skatu (skatīt 65. attēlā).



65. attēls. Bloga informācijas apskate un redīģēšanā

Vizuāls piemērs



66. attēls. Manu blogu saraksts

7.8. BLOGU MEKLĒŠANAS IESPĒJA

Apraksts

Šajā sadaļā tiks aprakstīts kā meklēt blogus šajos skatos: *Saved* (latviešu Tīme. *Saglabātie*), *Blogs* (latviešu val. *Emuāru*), *For you* (latviešu val. *Tev*), *Followers* (latviešu val. *Sekotāji*), vai arī kategoriju skatos, ko izvēlās lietotājs. Tas tiks aprakstīts ar izpildes soliem un vizuāliem piemēriem.

Piezīmes

Katrā no iepriekš pieminētajiem skatiem tā augšpusē atrodas meklēšanas josla ar kuras palīdzību tiks veikta meklēšana.

Izpildes soli

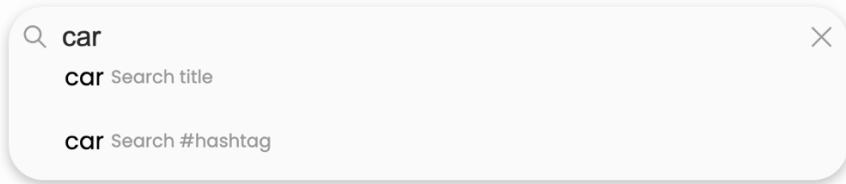
Lai meklētu blogu pēc tā nosaukuma nepieciešams:

1. ar kurSORU uziet uz meklēšanas joslas un izvēlēties (skatīt 67. attēlā),



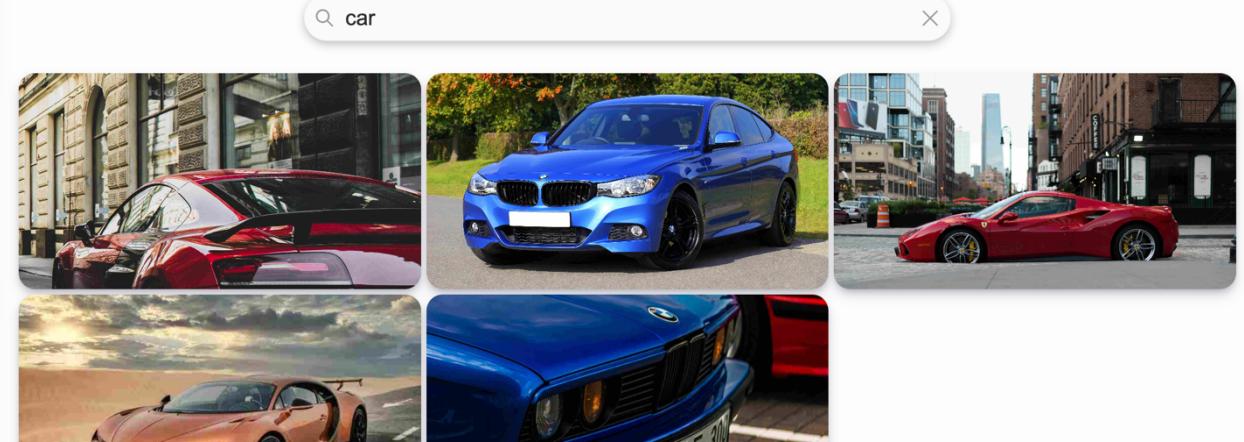
67. attēls. Blogu meklēšanas josla

2. ievadīt nepieciešamo atslēgas vārdu pēc kura vēlaties meklēt blogus, tad automātiski tiks parādītas iespējas meklēt pēc mirkļa birkas vai nosaukuma (skatīt 68. attēlā), tad izvēlās iespēju, tiek parādīti visi attiecīgie rezultāti (skatīt 69. attēlā).



68.attēls Meklēšanas izvēle

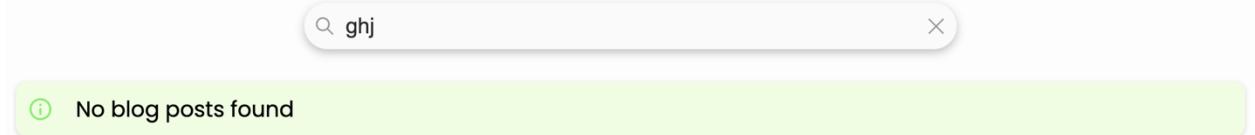
List



69. attēls. Meklēšana ar rezultātiem

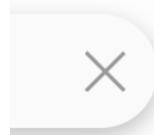
Ja lietotājs ievada atslēgas vārdu, kas nav saistīts ar nevienu datubāzē esošo blogu, vai nav neviens bloga ar attiecīgo mirkļa birku, netiek atgriezts neviens blogs un tiek parādīts ziņojums (skatīt 70. attēlā).

List



70. attēls. Nederīga atslēgas vārda ievadīšana

Ja lietotājs vēlas redzēt visus blogus, tam ar kurSORU jāuziet uz krustiņa kurš atrodas meklēšanas joslas labajā pusē (skatīt 71. attēlā) un jānoklikšķina tas vai arī manuāli jānodzēš atslēgas vārds, tad automātiski zem meklēšanas joslas tiek atgriezti visi blogi.



71. attēls. Meklēšanas joslas notīrīšana

7.9. BLOGA PIEVIENOŠANA

Apraksts

Šajā sadaļā tiks paskaidrots un aprakstīts kā lietotājs var pievienot blogu, tas tiks skaidrots ar izpildes soļu un vizuālu piemēru palīdzību (skatīt 81. attēlā). Atverot *New Blog* (latviešu val. *Jauns emuārs*) skatu, ir vairāki ievades lauku, kurus aizpildot var pievienot blogu.

Piezīmes

Izpildes soļos tiek paskaidrots kā ievietot blogu un paskaidro tikai obligāti nepieciešamos laukus.

Ja lietotājs vēlas pievienot e-pastu, nepieciešams:

- ievadīt ievadlaukā *Email* (latviešu val. *E-pasts*), e-pasta adresi (skatīt 72. attēlā),



72. attēls. E-pasta ievades paraugs (Bloga pievienošana)

Ja lietotājs vēlas pievienot Telefona nummuru, nepieciešams:

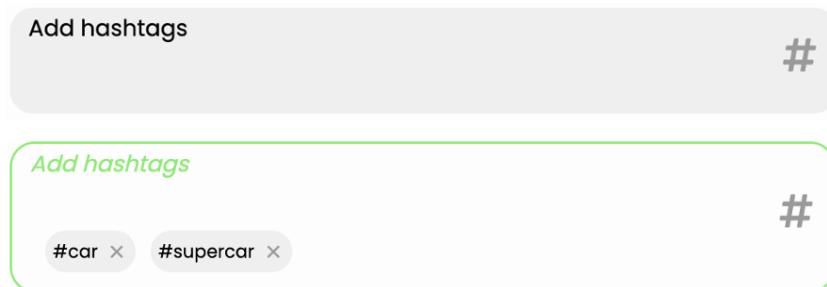
- ievadīt ievadlaukā *Phone* (latviešu val. *Telefons*), telefona numurs (skatīt 73. attēlā),



73. attēls. Telefona numura ievades paraugs (Bloga pievienošana)

Ja lietotājs vēlas pievienot mirkļa birkas, nepieciešams:

- ievadīt ievadlaukā *Hashtag* (latviešu val. *Mirkļa birka*), atslēgas vārdu, tad uzspiežot taustiņu *Enter* vai *Space* tiks izvadīta šī mirkļa birka lai to noņemtu, nepieciešams uzklikšķināt uz blakus esošā krustiņa (skatīt 74. attēlā).



74. attēls. Mirkļa birkas ievades paraugs (Bloga pievienošana)

Izpildes soļi

Lai izveidotu, lietotājam nepieciešams:

- ievadīt ievadlaukā *Title* (latviešu val. *Virsraksts*), virsrakstu kas neeksistē datubāzē (skatīt 75. attēlā),

The screenshot shows a user interface element for entering a title. It consists of a light gray rectangular box containing the text "Title*" in red. Below it is a larger input field with the placeholder "Virsraksts". A green rounded rectangle highlights the input field, indicating it is the active or selected field.

75. attēls. Virsraksta ievades lauks

- izvēlēties kategoriju izvēlēs laukā *Category* (latviešu val. *Kategorija*) (skatīt 76. attēlā),

The screenshot shows a user interface element for selecting a category. It consists of a light gray rectangular box containing the text "Category*" in green. Below it is a dropdown menu with the option "Cars" selected. A green rounded rectangle highlights the dropdown menu, indicating it is the active or selected field.

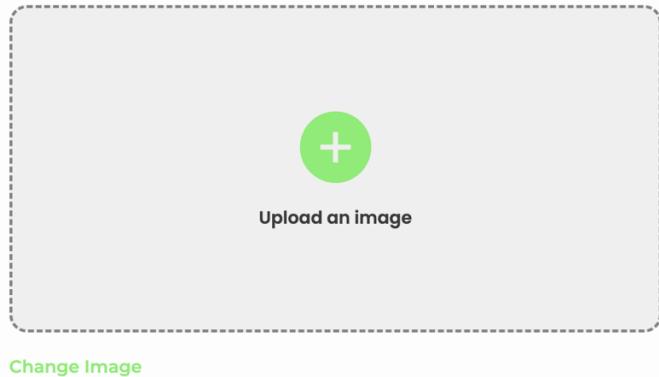
76. attēls. Kategoriju izvēles lauks

- ievadīt ievadlaukā *Description* (latviešu val. *Apraksts*), bloga aprakstu (skatīt 77. attēlā),

The screenshot shows a user interface element for entering a description. It consists of a light gray rectangular box containing the text "Description*" in green. Below it is a larger input field with the text "Apraksts" entered. A green rounded rectangle highlights the input field, indicating it is the active or selected field.

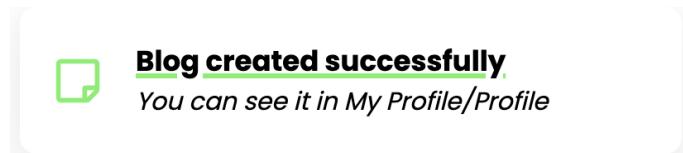
77. attēls. Apraksta ievades lauks

- izvēlēties titulattēlu izvēlēs laukā *Upload an image* (latviešu val. *Augšupielādēt attēlu*) (skatīt 78. attēls),



78. attēls. Attēla augšupielādes lauks

- jāizvēlas poga *Submit* (latviešu val. *Iesniegt*), ja lietotājs ievades laukus ir aizvadījis atbilstoši prasībām, tiks izveidots jauns blogs un tiks parādīts veiksmes ziņojums par bloga veiksmīgu izveidi (skatīt 79. attēlā). Ja lietotājs nebūs aizpildījis ievades laukus atbilstoši prasībām, zem katra ievades lauka tiks izvadīts kļūdu ziņojums (skatīt 80. attēlā).



79. attēls. Bloga izveides veiksmes ziņojums

A screenshot of a form with two fields. The first field is labeled "Title*" and has a red error message below it: "The title field is required.". The second field is labeled "Phone" and has a red error message below it: "This field must be at least 8 characters.". There are icons of a clipboard and a telephone handset next to their respective labels.

80. attēls. Ievades lauku kļūdu ziņojums

Vizuāls piemērs



Home
Blogs
New blog
My profile

Create your own content

Pen your journey, inspire hearts, and leave a mark forever

Blog info

All required fields will be marked with *

| | | | |
|----------------|----------------------|-----------|----------------------|
| Title* | <input type="text"/> | Category* | Select a category |
| Phone | <input type="text"/> | Email | <input type="text"/> |
| Add hashtags # | | | |
| Description* | | | |

Upload as file

Submit

Image upload

Upload to see image



Change Image

81.attēls. Blogu pievienošanas skats

7.10. BLOGU REDIĢĒŠANAS IESPĒJA

Apraksts

Šajā sadaļā tiks paskaidrots kā lietotājs var rediģēt sevis ievietotos blogus, tas tiks paskaidrots ar izpildes soliem un vizuāliem piemēriem. Lai rediģētu blogu, lietotājam nepieciešams atvērt skatu *Profile* (latviešu val. *Profils*) un jāizvēlas kāds no blogiem, un jānoklikšķina uz tā rediģēšanas ikonas (skatīt 82. attēlā un 83. attēlā), tad lietotājs tiks pārvirzīts uz rediģēšanas skatu, kurā ir redzama bloga informācija kas ievadīta ievadlaukos.



82.attēls Bloga izvēle



83.Bloga rediģēšanas ikona

Izpildes soli (Bloga informācijas rediģēšana)

Lai veiktu rediģēšanu ir jāmaina kādi no ievadītajiem datiem:

- ja lietotājs vēlās rediģēt virsrakstu, jāmaina informācija, ievades laukā *Title* (latviešu val. *Virsraksts*) (skatīt 84. attēlā);

| | |
|------------|--|
| Title* | |
| Virsraksts | |

84. attēls. Virsraksta redīgēšana

- ja lietotājs vēlas redīgēt kategoriju, jāmaina izvēle, izvēles laukā *Category* (latviešu val. *Kategorija*) (skatīt 85. attēlā);

| | |
|-------------------|--|
| Category* | |
| Select a category | |
| Cars | |

85. attēls. Kategorijas redīgēšana

- ja lietotājs vēlas redīgēt telefona numuru, jāmaina informācija, ievades laukā *Phone* (latviešu val. *Telefons*) (skatīts 86. attēlā);

| | |
|----------|--|
| Phone | |
| 22245671 | |

86. attēls. Telefona numura redīgēšana

- ja lietotājs vēlas redīgēt bloga e-pastu, jāmaina informācija, ievades laukā *Email* (latviešu val. *E-pasts*) (skatīt 87. attēlā);

| | |
|-----------------|--|
| Email | |
| janis@gmail.com | |

87. attēls. E-pasta redīgēšana

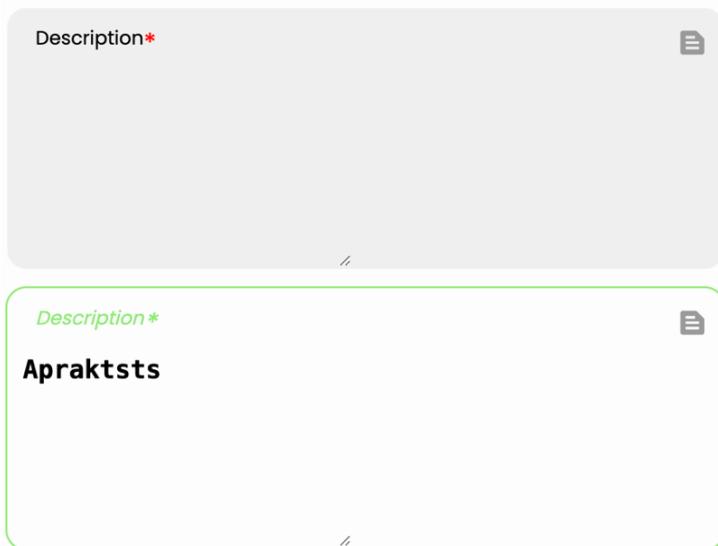
- ja lietotājs vēlas redīgēt blogam piesaistītās mirkļa birkas, lai pievienotu jaunas mirkļa birkas, ievadlaukā *Hashtag* (latviešu val. *Mirkļa birka*), jāievada atslēgas vārdi, tad uzspiežot taustiņu *Enter* vai *Space* tiks izvadīta šī mirkļa birka lai noņemtu to, vai jau esošās, nepieciešams uzklikšķināt uz blakus esošā krustiņa (skatīt 88. attēlā)

| | |
|--------------|---|
| Add hashtags | # |
|--------------|---|

| | |
|--------------|-------------|
| Add hashtags | # |
| #car x | #supercar x |

88. attēls. Mirkļa birkas ievades paraugs (Bloga redīgēšana)

- ja lietotājs vēlas redīģēt bloga aprakstu, jāmaina informācija, ievades laukā *Description* (latviešu val. *Apraksts*) (skatīt 89. attēlā);



89. attēls. Apraksta redīģēšana

- ja lietotājs vēlas redīģēt titulattēlu, jānospiež uz pogas *Change image* un tad jāizvēlas titulattēlu izvēlēs laukā *Upload an image* (latviešu val. *Augšupielādēt attēlu*) (skatīt 90. attēlā un 91. attēlā);

Change Image

90. attēls. Attēla redīģēšana



91. attēls. Attēla redīģēšanas augšupielāde

- jānospiež poga *Submit* (latviešu val. *Iesniegt*), ja lietotājs ievades laukus ir aizvadījis atbilstoši prasībām, tiks redīģēts blogs un tiks parādīts veiksmes ziņojums par bloga veiksmīgu redīģēšanu (skatīt 92. attēlā). Ja lietotājs nebūs aizpildījis ievades laukus atbilstoši prasībām, zem katra ievades lauka tiks izprintēts kļūdu ziņojums (skatīt 93. attēlā).



Blog updated successfully!

92. attēls. Bloga redīgēšana veiksmes ziņojums

Title*



The title field is required.

Phone



This field must be at least 8 characters.

93. attēls. Ievades lauku kļūdu ziņojums

Izpildes soļi (Bloga dzēšana)

Lai izdzēstu blogu, nepieciešams:

- atvērt bloga redīgēšanas skatu (skatīt 96. attēlā un 97. attēlā),
- uzklikšķināt uz pogas *Delete* (latviešu val. *Dzēst*) (skatīt 94. attēlā), tad tiks parādīts modulārais logs (skatīt 95.attēlā),
- tad šajā modulārajā logā nepieciešams uzklikšķināt uz pogas *Delete* (latviešu val. *Dzēst*) (skatīt 95.attēlā), tad blogs tiks dzēsts un lietotājs tiks pārvirzīts uz savu lietotāja profilu.

Delete

94.attēls Dzēšanas poga

Are you sure you want do delete blog?

If you agree, it will be deleted permanently

Cancel

Delete

95.attēls Dzēšanas modulārais logs

Vizuāls piemērs

Pen your journey, inspire hearts, and leave a mark forever

BlogIt

- Home
- Blogs
- New blog
- My profile

Blog info

All required fields will be marked with *

| | | | |
|-------------------------------|-------------|-----------|------|
| Title* | Car bugatti | Category* | Cars |
| Phone | | Email | |
| Add hashtags # | | | |
| #car × #supercar × #bugatti × | | | |
| Description* | | | |
| My car | | | |

Upload as file

Delete Submit

Image upload

Upload to see image



Change Image

96. attēls. Bloga redīģēšanas skats

Pen your journey, inspire hearts, and leave a mark forever

BlogIt

- Home
- Blogs
- New blog
- My profile

Blog info

All required fields will be marked with *

| | | | |
|-----------------------|---------|-----------|------|
| Title* | Bugatti | Category* | Cars |
| Phone | | Email | |
| Add hashtags # | | | |
| #car × #supercar × | | | |
| Description* | | | |
| My car bugatti | | | |

Upload as file

Delete Submit

Image upload

Upload to see image



Change Image

97. attēls. Bloga redīģēšanas skats (Redīģēts blogs)

7.11. LIETOTĀJA PROFILA APSKATE

Apraksts

Šajā sadaļa tiks paskaidrots kā apskatīt kāda lietotāja profilu, tas tiks paskaidrots ar izpildes soliem un vizuāliem piemēriem. Tiks paskaidroti 3 varianti, kā atvērt un apskatīt lietotāja profilu.

Izpildes soli

Lai apskatītu lietotāja profilu, nepieciešams:

- atvērt kāda konkrēta bloga skatu,
- noklikšķināt uz bloga autora (skatīt 98. attēlā), tad lietotājs tiks pārvirzīts uz autora profila.

Info



Author

Niks Kļaviņš

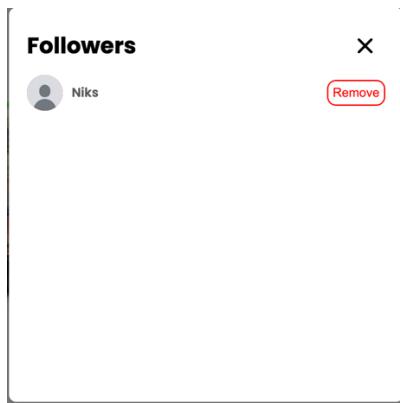
98. attēls. Bloga autors

Lai apskatītu lietotāja profilu, savā sekotāju sarakstā, nepieciešams:

- atvērt savu profilu;
- izvēlēties sekotāju skaitu *Followers* (latviešu val. *Sekotāji*) (skatīt 99. attēlā) autora profilā, tad tiks atvērts sekotāju saraksts (skatīt 100. attēlā);
- sekotāju sarakstā uzklikšķināt uz vārda vai profila attēlu, tad lietotājs tiks pārvirzīts uz lietotāja profila skatu.

1 Followers

99. attēls. Sekotāju skaits



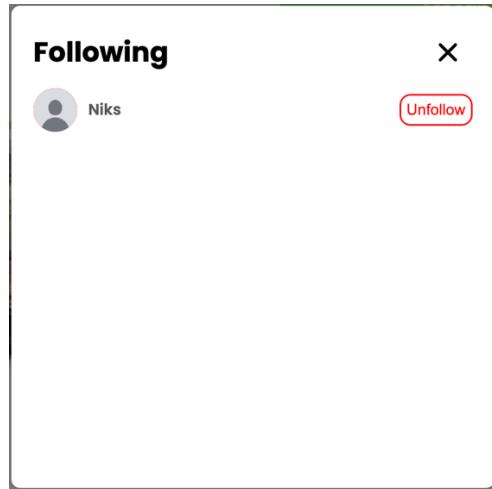
100. attēls. Sekotāju saraksts

Lai apskatītu lietotāja profilā, kam pats seko, nepieciešams:

- atvērt savu profilu;
- izvēlēties sekotāju skaitu *Following* (latviešu val. *Seko*) (skatīt 101. attēlā) autora profilā, tad tiks atvērts saraksts (skatīt 102. attēlā);
- sarakstā uzklikšķināt uz vārda vai profila attēlu, tad lietotājs tiks pārvirzīts uz lietotāja profila skatu.

1 Following

101. attēls. Sekojošo personu skaits



102. attēls. Sekojošo personu saraksts

7.12. PIESEKOŠANA UN ATSEKOŠANA LIETOTĀJAM

Apraksts

Šajā sadaļā tiks paskaidrots, kā piesekot, un kā pārstāt sekot kāda lietotāja profilam, tas tiks paskaidrots ar izpildes soļiem un vizuāliem piemēriem.

Izpildes soļi

Lai piesekotu lietotājam, nepieciešams:

- atvērt kāda lietotāja profilu,
- noklikšķināt uz piesekošanas pogas *Follow* (latviešu val. *Sekot*) (skatīt 103. attēlā),
- tad tiek parādīts teksts *Following* (latviešu val. *Sekojošs*) (skatīt 104. attēlā) un šis profils tiek pievienots sarakstā, kurā atrodas profili, kam seko.

Lai pārstātu sekot lietotājam, nepieciešams:

- atvērt kāda lietotāja profilu,
- noklikšķināt uz pogas *Following* (latviešu val. *Sekot*) (skatīt 104.attēlā),
- tad tiek parādīts teksts *Follow* (latviešu val. *Sekojošs*) (skatīt 103. attēlā) un šis profils tiek izņemts no saraksta, kurā atrodas profili kam jūs sekojat.



Niks Kļaviņš

1 Posts 0 Followers 2 Following

Follow

103. attēls. Poga *Follow* – sekošanas sākšana



Niks Kļaviņš

1 Posts 1 Followers 2 Following

Following

104. attēls. Poga *Following* – sekošanas pārtraukšana

7.13. SEKOTĀJU UN PROFILU KAM SEKO LIETOTĀJS SARAKSTU PĀRVALDĪBA

Apraksts

Šajā sadalā tiks paskaidrots, kā apskatīt savus sekotājus, profilus kam seko, kā izņemt šos profilus no sarakstiem, kā arī kā apskatīt šos profilus. Tas tiks paskaidrots ar izpildes soliem un vizuāliem piemēriem.

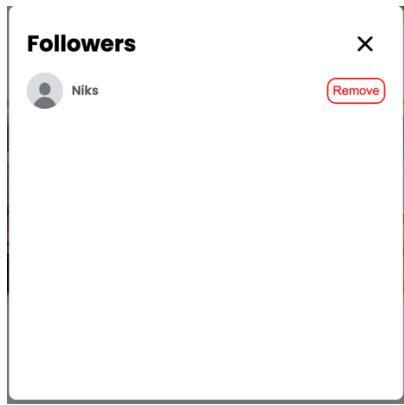
Izpildes soli lai pārvaldītu sekotāju sarakstu

Lai pārvaldītu sekotāju sarakstu, nepieciešams:

- atvērt savu profila skatu (*My profile->Profile*)
- noklikšķināt sekotāju skaita *Followers* (latviešu val. *Sekotāji*) (skatīt 105. attēlā), tad tiks atvērts sekotāju saraksts (skatīt 106. attēlā).
- ja vēlās dzēst lietotāju, no saraksta, nepieciešams noklikšķināt uz pogas *Remove* (latviešu val. *Noņemt*) (skatīt 107. attēlā),
- ja vēlas apskatīt šo profili, nepieciešams sarakstā uzklikšķināt uz profila attēla vai vārda, tad lietotājs tiks pārvirzīts uz profila skatu (skatīt 106. attēlā).

1 Followers

105.attēls Sekotāju skaits (Profila pārvaldība)



106. attēls. Sekotāju saraksts (Profila pārvaldība)

Remove

107. attēls. *Remove* pogas

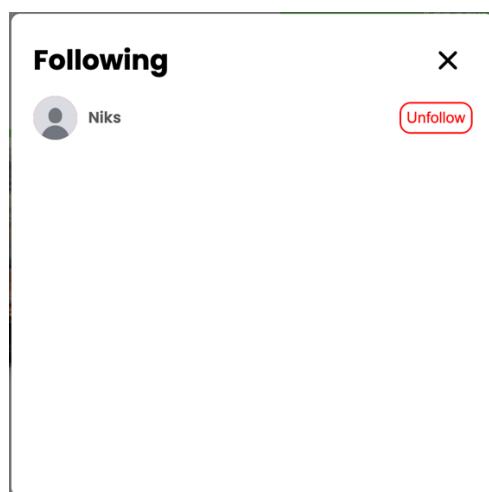
Izpildes soļi lai pārvaldītu profilu sarakstu kam seko lietotājs

Lai pārvaldītu profilu sarakstu kam seko lietotājs, nepieciešams:

- atvērt savu profila skatu (*My profile->Profile*),
- noklikšķināt sekotāju skaita *Following* (latviešu val. *Sekojošs*) (skatīt 108. attēlā), tad tiks atvērts sekotāju saraksts (skatīt 109. attēlā),
- ja vēlās pārstāt sekot lietotāja profilam un izņemt no saraksta, nepieciešams noklikšķināt uz pogas *Unfollow* (latviešu val. *Atsekat*) (skatīt 110. attēlā),
- ja vēlas apskatīt šo profili, nepieciešams sarakstā uzklikšķināt uz profila attēla vai vārda, tad lietotājs tiks pārvirzīts uz profila skatu (skatīt 109. attēlā)

1 Following

108. attēls. Sekojošo skaits (Profila pārvaldība)



109. attēls. Sekojošo saraksts (Profila pārvaldība)

Unfollow

110. attēls. *Unfollow* poga

7.14. PROFILA REDĪGĒŠANAS IESPĒJA

Apraksts

Šajā sadaļa tiks paskaidrots kā rediģēt profila informāciju un kā mainīt profila attēlu, tas tiks paskaidrots ar izpildes soļiem un vizuāliem piemēriem. Lai rediģētu profila informāciju, nepieciešams atvērt *Settings* (latviešu val. *iestatījumi*) un aizpildīt nepieciešamos ievades laukus.

Izpildes soļi

Lai veiktu profila informācijas rediģēšanu, lietotājam jāizvēlas ievadlauks ko rediģēt:

- lai rediģētu vārdu, lietotājam jāmaina informācija ievadlaukā *Name* (latviešu val. *Vārds*) (skatīt 111. attēlā),



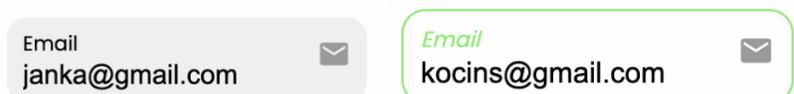
111. attēls. Vārda rediģēšana

- lai rediģētu uzvārdu, lietotājam jāmaina informācija ievadlaukā *Surname* (latviešu val. *Uzvārds*) (skatīt 112. attēlā),



112. attēls. Uzvārda rediģēšana

- lai rediģētu e-pastu, lietotājam jāmaina informācija ievadlaukā *Email* (latviešu val. *E-pasts*) (skatīt 113. attēlā),



113. attēls. E-pasta rediģēšana

- lai rediģētu profila attēlu, lietotājam jāuzspiež poga *Change profile image* (latviešu val. *Mainīt profila attēlu*), un tad jāizvēlas attēls (skatīt 114. attēlā un 115. attēlā),

Change profile image

114. attēls. Profila attēla maiņas poga



115. attēls. Profila attēla maiņa

- jānospiež poga *Save Changes* (latviešu val. *Saglabāt izmaiņas*) (skatīt 116. attēlā), ja lietotājs ievades laukus ir aizvadījis atbilstoši prasībām, tiks rediģēta profila informācija un tiks parādīts veiksmes ziņojums par profila rediģēšanu (skatīt 117. attēlā). Ja lietotājs nebūs aizpildījis ievades laukus atbilstoši prasībām, zem katra ievades lauka tiks izprinēts kļūdu ziņojums (skatīt 118. attēlā).

Save Changes

A large, rounded rectangular button with a bright green gradient background. The text "Save Changes" is centered in a white, bold, sans-serif font.

116. attēls. Profila rediģēšanas poga



117.attēls Veiksmes ziņojums profila rediģēšanā

Name This field must be at least 2 characters.

Surname This field must be at least 2 characters.

Two input fields side-by-side. The first is labeled "Name" and the second "Surname". Each field has a small red validation message below it: "This field must be at least 2 characters.". There are also small icons of people next to each input field.

118. attēls. Neveiksmīga ievades lauku aizpildīšana (Profilā rediģēšana)

7.15. AKCENTA KRĀSAS MAINĀ

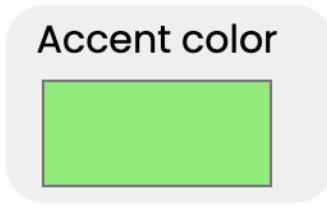
Apraksts

Šajā sadaļā tiks paskaidrots kā nomainīt akcenta krāsu pēc savas izvēles, bloga tīmekļu vietnei, to paskaidrojot ar izpildes soliem un vizuāliem piemēriem. Lai mainītu akcenta krāsu nepieciešams atvērt *Settings* (latviešu val. *iestatījumi*) un mainīt to krāsas laukā.

Izpildes soli

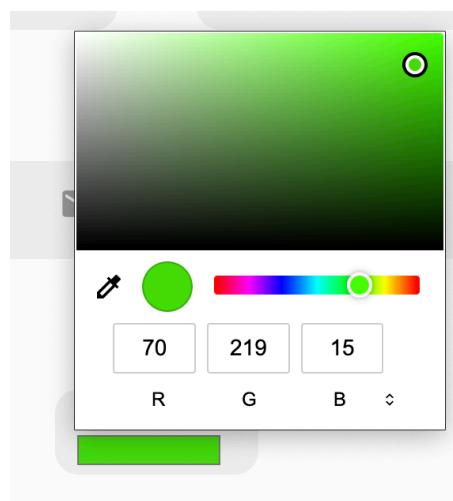
Lai lietotājs spētu mainīt akcenta krāsu:

- ar kurSORU jāuziet uz krāsas lauku *Accent color* (latviešu val. *Akcenta krāsai*), un jāuzklikšķina uz krāsas (skatīt 119. attēlā),



119. attēls. Akcenta krāsas lauks

- tad ar kurSORU vai rakstot krāsas kodu, jānomaina krāsa (skatīt 120. attēlā).



120. attēls. Akcenta krāsas izvēle

7.16. PAROLES MAINĀS IESPĒJA

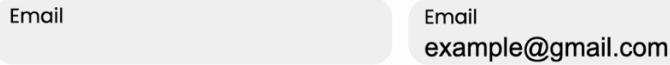
Apraksts

Šajā sadaļā tiks paskaidrots kā veikt paroles nomaiņu, tas tiks izskaidrots ar izpildes soliem un vizuāliem piemēriem (skatīt 129. attēlā un 130. attēlā).

Izpildes soli

Lai lietotājs veiktu paroles nomaiņu, tam ir jāveic šādi soli:

- jāievada e-pasts ievadlaukā *Email* (latviešu val. *E-pasts*), uz kuru vēlaties saņemt saiti paroles nomaiņai (skatīt 121. attēlā),

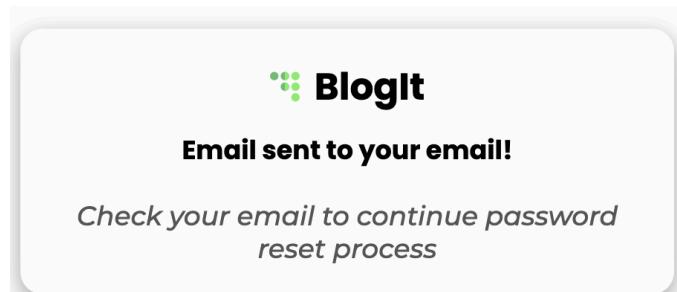


121. attēls. E-pasta ievades lauks (Paroles maiņa)

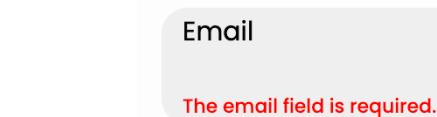
- jānospiež poga *Send* (latviešu val. *Sūtīt*) (skatīt 122. attēlā), ja lietotājs ir aizpildījis ievadlauku pēc nepieciešamajām prasībām, tiks parādīts veiksmes ziņojums par veiksmīgu e-pasta nosūtīšanu (skatīt 123. attēlā), bet ja ievadlauks nav aizpildīts atbilstoši prasībām, zem ievadlauka tiks parādīts klūdu ziņojums (skatīt 124. attēlā).

Send

122. attēls. E-pasta sūtīšanas poga



123. attēls. Veiksmes ziņojums par e-pasta nosūtīšanu



124. attēls. Ievades lauka klūdu ziņojums (E-pasta nosūtīšana)

- jāatver norādītais e-pasts, jāatver vēstule, un vēstulē jānospiež uz norādītās saites, tad lietotājs tiks pārvirzīts uz paroles maiņas skatu
- jāievada parole ievadlaukā *Password* (latviešu val. *Parole*) (skatīt 125. attēlā),



125. attēls. Paroles maiņas ievadlauks

- jāievada atkārtota parole ievadlaukā *Confirm password* (latviešu val. *Apsstiprināt paroli*), šim parolēm jābūt vienādām (skatīt 126. attēlā),

Confirm password
.....

Confirm password

126. attēls Paroles apstiprināšanas ievadlauks

- jānospiež poga *Reset* (latviešu val. *Atiestatīt*) (skatīt 127. attēlā), ja lietotājs ir aizpildījis ievadlaukus pēc nepieciešamajām prasībām, tiks parādīts veiksmes ziņojums par paroles maiņu, bet ja ievadlauki nav aizpildīti atbilstoši prasībām, zem ievadlauka tiks parādīts kļūdu ziņojums (skatīt 128. attēlā).



127. attēls. Paroles maiņas poga

Password

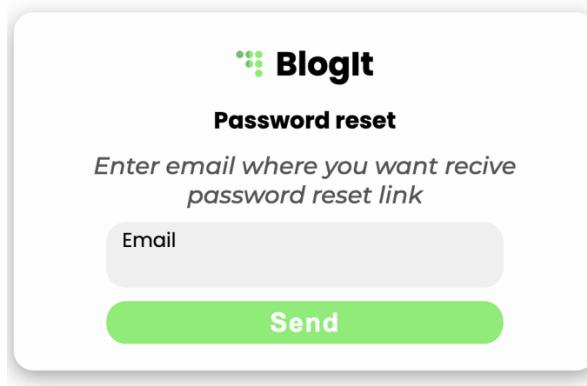
The password field is required.

Confirm password

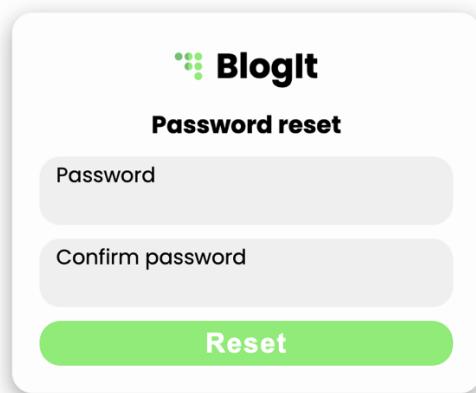
The confirm password field is required.

128. attēls. Paroles maiņas ievadlauku kļūdu paziņojumi

Vizuāli piemēri



129. attēls. E-pasta nosūtīšanas skats



130. attēls. Paroles maiņas skats

SECINĀJUMI UN PRIEKŠLIKUMI

1. Darba mērķis tika sasniegts, tika izveidota moderna, viegli lietojama un parocīga bloga tīmekļa vietni, kurā lietotājs spēs dalīties ar savu radošo darbību, dzīves notikumiem un viņam interesējošam darbībām, kā arī apskatīt un iedvesmoties no citiem tīmekļa vietnē ievietotiem blogiem.
2. Priekšgala sistēma (angļu v. *front-end*) veiksmīgi tika izstrādāta, izmantojot jaunākās tehnoloģijas, tika izveidoti visi nepieciešamie skati, lai lietotājs pilnvērtīgi varētu lietot visas piedāvātās funkcijas. Secināju to, ka turpmāk priekšgala sistēmas (angļu v. *front-end*), dizaina izstrādē turpmāk izmantošu *Tailwind CSS*, nevis *SCSS*, tas būtu bijis daudz ātrāk un efektīvāk.
3. Tīmekļa vietne tika izveidota pēc jaunākajiem tīmekļa vietņu dizaina principiem, kas balstās uz minimālismu. Tai tika izveidots pievilcīgs dizains, kas ļauj ikvienam lietotājam saprast, saredzēt un lietot tīmekļa vietnē pieejamās funkcijas.
4. Aizmugures sistēma (angļu v. *back-end*) veiksmīgi tika izstrādāta, izmantojot *PHP* ietvaru *Laravel*, kas atviegloja novērst daudzus drošības aspektus, kā, piemēram, *SQL* injekcijas, *Cross site scripting*, u.c. – *Laravel* spēj noteikt un apturēt *SQL* injekcijas, kā arī konfigurācijā ir iespēja noteikt, kadas tīmekļa vietnes drīkst piekļūt sistēmai, tādejādi apturot *Cross site scripting*. Funkcionalitāte tika veiksmīgi īstenota izmantojot *API*, ar kuru palīdzību notika komunikācija starp aizmugures sistēmu (angļu v. *back-end*) un priekšgala sistēmu (angļu v. *front-end*).
5. Tīmekļa vietnē tika izveidots rekomendāciju algoritms, respektīvi, kad lietotājs apskata kādus citus blogu ierakstus, tiek nolasītas tiem pievienotās mirkļa birkas un rekomendāciju algoritms atrod līdzīgās birkas un tiem pievienotos blogu ierakstus.
6. Tīmekļa vietnē veiksmīgi tika izstrādāta motīvu izvēle, precīzāk, lietotājam tiek dota iespēja mainīt akcenta krāsu tīmekļa vietnē, tas palīdz personalizēt šo vietni, un rada labāku lietošanas pieredzi.
7. Tīmekļa vietnē tika izveidota funkcionalitāte, lai lietotājs varētu sākt sekot un pārstāt sekot kādam lietotājam, tādejādi veidot sadāļu, kurā tas var redzēt sev interesējošo lietotāju ievietotos ierakstus.
8. Tīmekļa vietne tika izveidota funkcionalitāte, lai lietotājs spētu veidot komentāru, tas palīdz piesaistīt vairāk lietotājus, jo tie var vieglāk izpaust savu viedokli, gan par saviem, gan citu lietotāju ievietotajiem blogu ierakstiem.
9. Tīmekļa vietnē veiksmīgi tika veikti manuāli un automatizēti testi, kas ļāva konstatēt kādas nepilnības. Tika pārbaudītas visas lietotājam pieejamās funkcijas, kā arī tika veikta ievades lauku pārbaude funkciju izpildes laikā. Katra funkcija un ievades lauks tika veiksmīgi

pārbaudīts. Testēšana palīdzēja konstatēt radušās kļūmes funkcijas izpildes procesā, tādejādi atvieglojot problēmu risināšanu.

10. Turpmāk tīmekļa vietnē tiek rekomendēts veikt uzlabojumus lietotāja autorizācijas sistēmā, sniegt iespēju pieslēgties tīmekļa vietnei ar:
 - *Google* kontu,
 - *Facebook* kontu,
 - *Github* kontu.
11. Tieki rekomendēts ieviest iespēju pārslēgt galvenos motīvus, piemēram, tumšo motīvu vai gaišo motīvu.
12. Tieki rekomendēts papildināt un uzlabot rekomendāciju algoritmu, papildinot to ar mākslīgo intelektu jeb *AI*, kas palīdzētu precīzāk un labāk sniegt rekomendācijas ne tikai vadoties pēc mirkļa birkām, bet pēc saturu un pat attēliem.
13. Tieki rekomendēts ieviest iespēju lietotājam pievienot bloga aprakstu faila formātā, ja ir tāda nepieciešamība.
14. Tieki rekomendēts ieviest vairākas valodas, precīzāk: angļu un latviešu.
15. Tieki rekomendēts ieviest iespēju lietotājiem sarakstīties vienam ar otru privātajā sarakstē

IZMANTOTĀ LITERATŪRA

aws What is an API (Application Programming Interface) ? [Tiešsaiste]. aws, 2024. [skatīts: 03.06.24] Pieejams: <https://aws.amazon.com/what-is/api/>

Docker Docker overview [Tiešsaiste]. Docker, 2024. [skatīts: 03.06.24] Pieejams: <https://docs.docker.com/get-started/overview/>

Fortinet What Is An Authentication Token? [Tiešsaiste]. Fortinet, 2024. [skatīts: 04.06.24] Pieejams: <https://www.fortinet.com/resources/cyberglossary/authentication-token>

Gherkin Gherkin Syntax [Tiešsaiste]. Gherkin, 2024. [skatīts: 04.06.24] Pieejams: <https://cucumber.io/docs/gherkin/>

php A popular general-purpose scripting language [Tiešsaiste]. php, 2024. [skatīts: 04.06.24] Pieejams: <https://www.php.net/>

mdn web docs What is Javascript? [Tiešsaiste]. Mdn web docs, 2024. [skatīts: 04.06.24] Pieejams: https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First_steps/What_is_JavaScript

IBM What is artificial intelligence (AI)? [Tiešsaiste]. IBM, 2024. [skatīts: 03.06.24] Pieejams: <https://www.ibm.com/topics/artificial-intelligence>

Jamie Beatty What are Recommendation Algorithms? [Tiešsaiste]. embryo, 2024. [skatīts: 03.06.24] Pieejams: <https://embryo.com/blog/recommendation-algorithms/>

Kaspersky What is SQL injection? Definition and explanation [Tiešsaiste]. Kaspersky, 2024. [skatīts: 03.06.24] Pieejams: <https://www.kaspersky.com/resource-center/definitions/sql-injection>

KirstenS Cross Site Scripting (XSS) [Tiešsaiste]. Owasp, 2024. [skatīts: 03.06.24] Pieejams: <https://owasp.org/www-community/attacks/xss/>

mdn web docs What is CSS? [Tiešsaiste]. Mdn web docs, 2024. [skatīts: 04.06.24] Pieejams: https://developer.mozilla.org/en-US/docs/Learn/CSS/First_steps/What_is_CSS

MySQL What is MySQL [Tiešsaiste]. MySQL, 2024. [skatīts: 03.06.24] Pieejams: <https://dev.mysql.com/doc/refman/8.4/en/what-is-mysql.html>

Ong, Si Quan 59 Blogging Stats for 2023 [Tiešsaiste]. ahrefsblog, 2023. [skatīts: 08.01.24] Pieejams: <https://ahrefs.com/blog/blogging-statistics/>

Peter Pin-shan Chen The Entity-Relationship Model: Toward a Unified View of Data.

[Tiešsaiste]. CiteSeerX, 1976. [skatīts: 08.04.24] Pieejams:

<https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.123.1085>

React A JavaScript library for building user interfaces [Tiešsaiste]. React, 2024. [skatīts:

25.03.24] Pieejams: <https://legacy.reactjs.org/>

SCSS CSS with superpowers [Tiešsaiste]. Sass, 2024. [skatīts: 03.06.24] Pieejams: [https://sass-](https://sass-lang.com/)

[lang.com/](https://sass-lang.com/)

Statista Vadošo tīmekļa pārlūkprogrammu versiju globālā tirgus daļa 2023. gada augustā

[Tiešsaiste]. **Statista**, 2024. [skatīts: 03.05.24] Pieejams:

<https://www.statista.com/statistics/268299/most-popular-internet-browsers/>

Taylor Otwell Eloquent: Getting Started [Tiešsaiste]. Laravel, 2024. [skatīts: 01.06.24]

Pieejams: <https://laravel.com/docs/10.x/eloquent>

Taylor Otwell Middleware [Tiešsaiste]. Laravel, 2024. [skatīts: 17.01.24] Pieejams:

<https://laravel.com/docs/10.x/middleware#main-content>

Taylor Otwell The php framework [Tiešsaiste]. Laravel, 2024. [skatīts: 03.06.24] Pieejams:

<https://laravel.com/>

TechTerms Backend Definition [Tiešsaiste]. TechTerms, 2024. [skatīts: 10.05.24] Pieejams:

<https://techterms.com/definition/backend>

TechTerms Frontend Definition [Tiešsaiste]. TechTerms, 2024. [skatīts: 27.02.24] Pieejams:

<https://techterms.com/definition/frontend>

PIELIKUMI

1. pielikums. Lietotāja pieslēgšanās programmas kods

Lai lietotājs spētu izmantot tīmekļa vietnē pieejamās funkcijas, tam sākotnēji nepieciešams pieslēgties ar iepriekš izveidotu lietotāja kontu.

Secīgs koda paskaidrojums

- Sākotnēji tiek veikta ievades lauku informācijas validācija, ja tā bijusi neveiksmīga tiek atgriezti klūdu ziņojumi, ja veiksmīga, programma turpina pildīt uzdevumu.
- Tad tiek izveidots unikāls rakstzīmju savārstījums jeb autorizācijas žetons (angļu v. *token*).
- Tieki veikts mēģinājums autorizēt lietotāju ar datiem, kas tika saņemti no ievades laukiem, ja lietotājs tiek autorizēts, lietotājs tiek pārvirzīts uz galveno tīmekļa vietnes skatu un no aizmugursistēmas (angļu v. *back-end*) uz priekšgalsistēmu (angļu v. *front-end*) tiek nosūtīts iepriekš izveidotais žetons, kas tiek saglabāts un darbojas kā atslēga datu apmaiņai starp aizmugursistēmu (angļu v. *back-end*) un priekšgalsistēmu (angļu v. *front-end*), ja autorizācija bijusi neveiksmīga, tiek atgriezts klūdu ziņojums.

Koda piemērs (pieslēgšanās)

```
public function login(Request $request)
{
    //Tiek veikta validācija, kas pārbauda ievadlaukos ievadīto informāciju
    //Ja informācija ir kludaina, tiek atgriezti klūdu ziņojumi
    $request->validate([
        "email"=> "required|max:20|min:5|email",
        "password"=> "required|max:20|min:5",
    ]);
    //Tiek izveidots unikāls rakstzīnju savārstījums jeb žetons
    $newToken = Str::random(60);
    $credentials = $request->only("email", "password");
    $userCheck = User::where('email', $request->only('email'))->first();
    //Tiek veikta pārbaude vai lietotājs ir autorizēts
    if(Auth::attempt($credentials)) {
        //Tiek atjaunota lietotāja informācija
        $userCheck->update([
            'updated_at' => date('Y-m-d H:i:s'),
        ]);
        //Tiek ievietots datubāzē iepriekš izveidotais žetons
        $token = [
            "user_id"=> Auth::id(),
            "token"=>$newToken,
            "created_at"=>date('Y-m-d H:i:s'),
            "updated_at"=>date('Y-m-d H:i:s'),
        ];
        Tokens::create($token);
        //Tiek atgriezti dati, veiksmīgas pieslēgšanās gadījumā
        //Tiek atgriezts arī žetons, ko priekšgalsistēmā saglabā, tas strādā kā atslēga datu apmaiņai
        return response()->json(["success"=>"OK", "link"=>"home",
        "user"=>$newToken], 200 );
    }
}
```

```

    } else{
        //Ja pieslēgšanās notikusi neveiksmīgi, tiek atgriezts kļudu
        zīņojums
        $errors = [
            "invalid" =>'Username or password is incorrect'
        ];
        return response()->json(["success"=>"ERR", 'errors'=>$errors],
422);
    }
}

```

2. pielikums. Bloga ieraksta pievienošanas programmas kods

Lietotājam nepieciešams pielsēgties mīmekļa vietnē, lai pievienotu bloga ierakstu

Secīgs koda paskaidrojums

- Sākotnēji tiek veikta ievades lauku informācijas validācija, ja tā bijusi neveiksmīga tiek atgriezti kļūdu zīņojumi, ja veiksmīga, programma turpina pildīt uzdevumu.
- Tad tiek sagatavots datu māsīvs ar bloga informāciju.
- Tieka veikts mēģinājums izveidot blogu.
- Ja bloga izveide bijusi veiksmīga, tiek pārbaudīts vai lietotājs ir pievienojis mirkļa birkas, tad, ja tās neeksistē datubāzē tiek saglabātas, tad tiek paņemta unikālā atslēga *id*, piesaistīta blogam un saglabāta datubāzē un tiek atgriezts veiksmes zīņojums.
- Ja bloga izveide bijusi neveiksmīga, tiek atgriezts kļūdu zīņojums.

Koda piemērs (pieslēgšanās)

```

public function create(Request $request)
{
    //Tiek veikta ievades lauku datu validācija
    //Ja kāds no ievades laukiem aizpildīts kļudaini, tiek atgriezti kļudu
    zīņojumi
    $data = $request->input();
    $request->validate([
        "title"=> "required|max:50|min:4",
        "description"=> "required|max:4000|min:4",
        "category"=> "required|max:20|min:4",
        "phone"=> "max:13",
        "email"=> "max:20",
        "img"=> "required"
    ]);
    $img = $request->file("img");
    //Tiek sagatavoti bloga informācijas dati
    $blog = [
        "title" => preg_replace($this->patterns, " ", $data["title"]),
        "description" => preg_replace($this->patterns, " ",
data["description"]),
        "category" => preg_replace($this->patterns, " ",
$data["category"]),

```

```

    "author"=>Session::get("user_id"),
    "user_id"=>Session::get("user_id"),
    "img" => $this->imagesFunctions->compress($img, 15),
];
//Veic pārbaudi, ja lietotājs ir ievadīji e-pasta adresi, tad to pievieno datu masīvam
if($request->input("email")){
    $data["email"] = preg_replace($this->patterns, " ", $request->input("email"));
}
//Veic pārbaudi, ja lietotājs ir ievadīji telefona numuru adresi, tad to pievieno datu masīvam
if($request->input("phone")){
    $data["phone"] = preg_replace($this->patterns, " ", $request->input("phone"));
}
//Tiek izveidots blogs
$create = Blog::create($blog);
if($create){
    //Ja bloga izveide ir bijusi veiksmīga
    //Veic pārbaudi, ja lietotājs ir ievadījis mirkļa birkas
    if($request->input("tags")){
        foreach (explode(", ", preg_replace($this->patterns, " ", $request->input("tags")))) as $tag) {
            //Tiek sadalītas mirkļa birkas, un ja tā neeksistē, datubāzē, to pievieno datubāzei
            $exist = Tag::query()->where("tag", "=", $tag)->first();

            if (!$exist) {
                $newTag = Tag::query()->create([
                    "tag" => $tag
                ]);
                $tagId = $newTag->id;
            } else {
                $tagId = $exist->id;
            }
            //Tiek pievienota mirkļa birkas unikālā atslēga tabulā, kurā tiek glabātas blogu mirkļa birkas
            BlogTag::query()->create([
                "tag_id" => $tagId,
                "blog_id" => $create->id
            ]);
        }
    }
    //Ja viss noritējis veiksmīgi, tiek atgriezts veiksmes ziņojums
    return response()->json(["message"=> "Blog created successfully"], 200);
} else{
    //Ja notikusi kļūme darbības izpildes gaitā, tiek atgriezts kļūdu ziņojums
    return response()->json(["message"=> "Something gone wrong", "error"=>Blog::create($blog)], 300);
}
}

```