

Liepājas Valsts tehnikums

“FABLAB” pārvaldības lietotnes izstrāde “IND”

Kvalifikācijas eksāmena praktiskās daļas dokumentācija

Darba autors:

Endijs Kudums 4PT-1

Darba vadītājs:

Programmēšanas skolotājs, Kristovskis Raimonds

Eksāmena datums 2025. gada \_\_. jūnijs

Liepāja 2025

**Saturs**

[Ievads 5](#_Toc194575799)

[1. Uzdevuma formulējums 7](#_Toc194575800)

[2. Programmatūras prasību specifikācija 8](#_Toc194575801)

[2.1. Produkta perspektīva 8](#_Toc194575802)

[2.2. Sistēmas funkcionālās prasības 8](#_Toc194575803)

[2.2.1. Autorizācijas sistēma 8](#_Toc194575804)

[2.2.2. Reģistrēšanas sistēma 9](#_Toc194575805)

[2.2.3. Paroles atiestatīšanas sistēma 9](#_Toc194575806)

[2.2.4. QR koda skeneris 10](#_Toc194575807)

[2.2.5. Notikuma izveidošana kalendārā 10](#_Toc194575808)

[2.2.6. Sānu josla funkcionalitāte 11](#_Toc194575809)

[2.2.7. Inventāra instrukciju apskate 12](#_Toc194575810)

[2.2.8. Inventāra izmaiņas: pievienošana, atņemšana un rediģēšana 13](#_Toc194575811)

[2.2.9. Staciju saraksta attēlošana 13](#_Toc194575812)

[2.2.10. Inventāra saraksta atsvaidzināšana 14](#_Toc194575813)

[2.2.11. Jauns aprīkojums pārvaldība 14](#_Toc194575814)

[2.2.12. Uzdevumu saraksts 15](#_Toc194575815)

[2.2.13. Uzdot uzdevumu 16](#_Toc194575816)

[2.2.14. Darbības žurnāls 16](#_Toc194575817)

[2.2.15. Ziņojums par problēmu 17](#_Toc194575818)

[2.2.16. Iestatījumi 17](#_Toc194575819)

[2.2.17. Stacijas kārtošana 18](#_Toc194575820)

[2.2.18. Kalendāra notikuma statusa maiņa 18](#_Toc194575821)

[2.2.19. Uznirstošais logs uzdevumiem kalendārā 19](#_Toc194575822)

[2.2.20. Statusa maiņa uzdevumiem 19](#_Toc194575823)

[2.2.21. Lietotāja statusa maiņa 19](#_Toc194575824)

[2.2.22. Lietotāja konta dzēšaņa 20](#_Toc194575825)

[2.3. Sistēmas nefunkcionālās prasības 20](#_Toc194575826)

[2.4. Gala lietotāja raksturiezīmes 21](#_Toc194575827)

[2.4.1. Administratori 21](#_Toc194575828)

[2.4.2. Darbinieki 21](#_Toc194575829)

[3. Izstrādes līdzekļu, rīku apraksts un izvēles pamatojums 22](#_Toc194575830)

[3.1. Izvēlēto risinājuma līdzekļu un valodu apraksts 22](#_Toc194575831)

[3.1.1. Java 22](#_Toc194575832)

[3.1.2. Firebase 22](#_Toc194575833)

[3.1.3. Android Studio 22](#_Toc194575834)

[3.2. Iespējamo (alternatīvo) risinājuma līdzekļu un valodu apraksts 23](#_Toc194575835)

[3.2.1. Kotlin 23](#_Toc194575836)

[3.2.2. Python 23](#_Toc194575837)

[3.2.3. SQLite 23](#_Toc194575838)

[3.2.4. Realm 24](#_Toc194575839)

[3.2.5. IntelliJ IDEA 24](#_Toc194575840)

[3.2.6. Qt Creator 24](#_Toc194575841)

[4. Sistēmas modelēšana un projektēšana 25](#_Toc194575842)

[4.1. Sistēmas struktūras modelis 25](#_Toc194575843)

[4.1.1. Sistēmas struktūra (komponenšu diagramma) 25](#_Toc194575844)

[4.1.2. Klašu diagramma 26](#_Toc194575845)

[4.2. Funkcionālais un dinamiskais sistēmas modelis 29](#_Toc194575846)

[4.2.1. Lietojumgadījumu diagramma 29](#_Toc194575847)

[4.2.2. Aktivitāšu diagramma 30](#_Toc194575848)

[4.2.3. Stāvokļu diagramma 33](#_Toc194575849)

[4.3. Datu struktūru apraksts 35](#_Toc194575850)

[5. Lietotāju ceļvedis 37](#_Toc194575851)

[5.1. Lietotāja konts un autentifikācija 37](#_Toc194575852)

[5.1.1. Reģistrēšanās 37](#_Toc194575853)

[5.1.2. Pierakstīšanās 38](#_Toc194575854)

[5.1.3. Aizmirsi paroli 38](#_Toc194575855)

[5.2. Sākuma ekrāns un navigācija 39](#_Toc194575856)

[5.2.1. Galvenais ekrāns 39](#_Toc194575857)

[5.2.2. Sānu josla 39](#_Toc194575858)

[5.3. Inventāra pārvaldība 40](#_Toc194575859)

[5.3.1. Iekārtas apskatīšana (Parasts lietotājs) 40](#_Toc194575860)

[5.3.2. Iekārtas apskatīšana un tā rediģēšana (Darbinieks vai administrators) 41](#_Toc194575861)

[5.3.3. Iekārtes pievienošana (Darbinieks vai administrators) 42](#_Toc194575862)

[5.4. Kalendāra notikuma statusa rediģēšana 42](#_Toc194575863)

[5.4.1. Notikuma apskatīšana 42](#_Toc194575864)

[5.4.2. Notikuma status maiņa (Darbinieki vai administratori) 43](#_Toc194575865)

[6. Testēšanas dokumentācija 44](#_Toc194575866)

[6.1. Izvēlētās testēšanas metodes, rīku apraksts un pamatojums 44](#_Toc194575867)

[6.2. Testpiemēru kopa 44](#_Toc194575868)

[6.3. Testēšanas žurnāls 46](#_Toc194575869)

[7. Secinājumi 61](#_Toc194575870)

[8. Lietoto terminu un saīsinājumu skaidrojumi 62](#_Toc194575871)

[9. Literatūras un informācijas avotu saraksts 63](#_Toc194575872)

# Ievads

FABLAB ir vieta, kur var izmantot modernas tehnoloģijas, piemēram, 3D printerus un lāzergriezējus, lai veidotu dažādus projektus. Taču šādu telpu pārvaldība var būt sarežģīta. Bieži vien ir grūtības ar inventāra uzskaiti, notikumu plānošanu un dokumentāciju glabāšanu. Lai to atvieglotu, ir nepieciešama mobilā lietotne, kas palīdzētu vienkāršot visu svarīgo informāciju vienuviet. Šī lietotne ļaus viegli pārbaudīt inventāra pieejamību, redzēt nākamos notikumus un piekļūt vajadzīgajiem dokumentiem. Tā padarīs FABLAB pārvaldību efektīvāku un vienkāršāku gan administratoriem, gan lietotājiem. Šī programma palīdzēs ietaupīt laiku, uzlabot darba organizāciju un nodrošinās, ka viss svarīgais ir viegli pieejams. Šajā dokumentācijā tiks aprakstīta izstrādājamās sistēmas specifikācija, tās funkcionalitāte un tehnoloģijas, kas tiks izmantotas sistēmas izveidei. Dokumentācija ir sadalīta vairākās daļās:

* **Uzdevuma formulējums** – tiek definēti galvenie sistēmas mērķi un tās nepieciešamība.
* **Programmatūras prasību specifikācija** – ietver informāciju par sistēmas funkcionālajām un nefunkcionālajām prasībām, kā arī gala lietotāja raksturiezīmēm.
* **Izstrādes līdzekļu un rīku apraksts** – tiek aprakstīti izmantotie rīki, programmēšanas valodas un alternatīvie risinājumi.
* **Sistēmas modelēšana un projektēšana** – ietver sistēmas arhitektūras, klašu un ER diagrammu, lietojum gadījumu diagrammas un aktivitāšu diagrammas izklāstu.
* **Lietotāju ceļvedis** – sniedz norādījumus par sistēmas lietošanu.
* **Testēšanas dokumentācija** – apraksta testēšanas metodoloģiju, izmantotos rīkus un testpiemērus.
* **Secinājumi** – izklāsta secinājumus par sistēmas izstrādi un tās iespējamo attīstību nākotnē.
* **Lietoto terminu un saīsinājumu skaidrojumi** – sniedz skaidrojumus par specifiskiem terminiem, kas izmantoti dokumentācijā.
* **Literatūras un informācijas avotu saraksts** – ietver atsauces uz izmantoto literatūru un resursiem.

Šī dokumentācija kalpos kā ceļvedis izstrādes procesā, nodrošinot skaidru struktūru un prasības, kas jāievēro sistēmas izstrādē.

# Uzdevuma formulējums

Aplikācijas galvenais uzdevums ir palīdzēt pārvaldīt FABLAB telpu un tās iekārtas vienkārši un ērti, nodrošinot efektīvu resursu plānošanu, caurspīdīgu informācijas apriti un ērtu lietošanas pieredzi. Tā būs moderns digitālais rīks, kas ļaus:

**Uzskaitīt inventāru** -

Lietotāji varēs apskatīt pilnu FABLAB telpā pieejamo iekārtu un rīku sarakstu, uzzināt to statusu, pieejamību un tehniskās specifikācijas. Ja lietotājam nepieciešams konkrēts aprīkojums, viņš varēs to rezervēt noteiktam laikam, izvairoties no problēmām un nodrošinot efektīvu resursu izmantošanu. Turklāt sistēmā būs iespējams sekot līdzi inventāra stāvoklim un ziņot par nepieciešamajiem remontiem vai trūkstošajām vienībām.

**Notikumu pārvaldība** -

Aplikācijā būs integrēts pasākumu un meistarklašu kalendārs, kurā lietotāji varēs apskatīt gaidāmos pasākumus, uzzināt sīkāku informāciju par tiem. Šī funkcija palīdzēs sakārtot aktivitātes, veicinās lietotāju iesaisti un nodrošinās, ka neviens nepalaidīs garām sev interesējošus pasākumus.

**Instrukcijas glabāšanu -**

Lai atvieglotu piekļuvi svarīgai informācijai, aplikācijā būs pieejama strukturēta instrukcijas krātuve. Tajā tiks saglabātas visas iekārtu lietošanas instrukcijas, drošības noteikumi, tehniskās specifikācijas un citi būtiski dokumenti. Lietotāji varēs tos ātri atrast un izmantot, tādējādi uzlabojot darba drošību un samazinot laiku, kas tiek tērēts informācijas meklēšanai.

**Komunikācijas uzlabošana -**

Lietotne nodrošinās efektīvu informācijas apmaiņu starp administrāciju un lietotājiem. Piemēram, ja kāda iekārta ir sabojājusies, lietotāji varēs nekavējoties ziņot administratoriem, lai problēma tiktu ātri novērsta. Tāpat administratori varēs nosūtīt paziņojumus par svarīgām izmaiņām, gaidāmiem remontdarbiem. Šī funkcija palīdzēs uzturēt kārtību, samazināt pārpratumus un nodrošināt raitu darbplūsmu.

Aplikācijas mērķis ir padarīt FABLAB pārvaldību efektīvāku un lietotājiem draudzīgāku, samazinot laikietilpīgus procesus un atvieglojot ikdienas darbu. Tā būs uzticams palīgs gan administratoriem, gan lietotājiem, veicinot sakārtotu un produktīvu darba vidi.

# Programmatūras prasību specifikācija

Šajā nodaļa tiks aprakstīts visas FABLAB programatūras funkcionālās un nefunkcionālās prasības.

## Produkta perspektīva

FABLAB mobilā lietotne būs neaizstājams rīks gan administratoriem, gan lietotājiem, ļaujot:

* Ērti piekļūt informācijai par pieejamo inventāru, iekārtām un resursiem, kā arī rezervēt nepieciešamo aprīkojumu savam darbam.
* Sekot līdzi gaidāmajiem notikumiem, piemēram, meistarklasēm un pasākumiem, un viegli reģistrēties tiem.
* Pieejot visu vajadzīgo dokumentāciju, piemēram, instrukcijas un drošības noteikumus, strādājot droši un efektīvi.
* Uzlabot komunikāciju ar administratoriem, piemēram, ziņojot par bojājumiem vai saņemot paziņojumus par inventāra statusu.

## Sistēmas funkcionālās prasības

### Autorizācijas sistēma

Mērķis: Funkcijas nepieciešamība ir lai FABLAB darbinieki var pieslēgties sistēmā.

Ievaddati:

1. **tabula**

**Darbinieku autorizācijas dati**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| E-pasts | Jā |  |
| Parole | Jā | Tiek šifrēts datubāzē |

Apstrāde: Nospiežot “piesakieties” pogu aplikācija pārbauda ievadītos datus ja tie sakrīt tad ļauj pieslēgties sistēma un ja nesakrīt tad izvada paziņojum par to ka ievadītie dati nav pareizi.

Izvaddati:

1. Nepieciešami aizpildīt visus ievadlaukus;
2. Paziņojums par nepareizu lietotājvārdu vai paroli;
3. Veiksmīgi tiek sistēmā.

### Reģistrēšanas sistēma

Mērķis: Funkcijas nepieciešamība ir lai viegli varētu piereģistrēt FABLAB darbiniekus.

Ievaddati:

1. **tabula**

**Darbinieku autorizācijas dati**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| Pilnais vārds un uzvārds | Jā | Tikai ar latviešu burtiem |
| Dzimšanas diena | Jā |  |
| E-pasts | Jā |  |
| Parole | Jā | Tiek šifrēts datubāzē |
| Statuss | Jā | Tiek automātiski izveidots kad reģistrējās. Pēc noklusējuma status ir Lietotājs |

Apstrāde: Nospiežot piesakieties pogu aplikācija pārbauda ievadītos datus ja tie sakrīt tad ļauj pieslēgties sistēma un ja nesakrīt tad izvada paziņojum par to ka ievadītie dati nav pareizi.

Izvaddati:

1. Nepieciešami aizpildīt visus ievadlaukus;
2. Paziņojums par nepareizu lietotājvārdu vai paroli;
3. Veiksmīgi tiek sistēmā.

### Paroles atiestatīšanas sistēma

Mērķis: Nodrošināt, ka lietotājs var atiestatīt paroli, ja tā ir aizmirsta.

Ievaddati:

1. **tabula**

**Paroles atiestatīšanas dati**

|  |  |  |
| --- | --- | --- |
| Nosaukums | Obligāts | Piezīmes |
| Reģistrētais e-pasts | Jā | Tiek pārbaudīts formāts. |
| Jaunā parole | Jā | Tiek šifrēta un validēta (min. 8 rakstzīmes). |

Apstrāde: Sistēma nosūta e-pastā saiti paroles atiestatīšanai. Lietotājs ievada jauno paroli, un tā tiek saglabāta datubāzē.

Izvaddati:

1. Paziņojums par veiksmīgu paroles atiestatīšanu.

### QR koda skeneris

Mērķis: Ātra piekļuve inventāra informācijai, noskenējot QR kodu.

Ievaddati:

1. **tabula**

**QR koda skenera dati**

|  |  |  |
| --- | --- | --- |
| Nosaukums | Obligāts | Piezīmes |
| QR kods | Jā | Tiek noskenēts, izmantojot kameru. |

Apstrāde: Sistēma nolasīs QR kodā iekļauto identifikatoru. Meklēs atbilstošo ierakstu datubāzē un parādīs inventāra detaļas.

Izvaddati: Inventāra informācijas kartīte ar šādiem datiem: Nosaukums, attēls, apraksts, telpas atrašanās vieta.

### Notikuma izveidošana kalendārā

Mērķis: Darbinieki var redzēt pasākumus un pievienot jaunus lai citi darbinieki var tos arī redzēt.

Ievaddati:

1. **tabula**

**Kalendāra notikuma izveidošanas dati**

|  |  |  |
| --- | --- | --- |
| Nosaukums | Obligāts | Piezīmes |
| Stacija | Jā | Pie kuras stacijas |
| Apraksts | Jā |  |
| Cilvēku skaits | Jā |  |
| Datums | Jā |  |
| Sākuma laiks | Jā |  |
| Beigu laiks | Jā |  |

Apstrāde: Atrodoties paredzētā vietā un aizpildot visu vajadzīgos datus ļauj izveidot jaunu ierakstu kalendārā kas pēc izveidošanas tiek attēlots kalendārā.

Izvaddati:

1. Veiksmīgi izveidots kalendārs
2. Netika aizpildīts visi datu lauki

### Sānu josla funkcionalitāte

Mērķis: Nodrošināt lietotājiem ērtu un ātru navigāciju uz lietotnes galvenajām funkcijām, balstoties uz viņu piekļuves līmeni.

Ievaddati:

1. **tabula**

**Sarakstu attēlošana**

|  |  |  |
| --- | --- | --- |
| Nosaukums | Apraksts | Pieejamība (Lietotājs, Administrators, darbinieks) |
| Jauns aprīkojums | Atver sadaļu, kur administrators var pievienot jaunu inventāru, norādot tā detaļas (tips, nosaukums, utt.). | Tikai darbinieki |
| Uzdevumi | Parāda visus uzdevumus, kas piešķirti lietotājam (piemēram, remontdarbi, apkope, inventāra pārbaude). | Tikai darbiniekiem |
| Uzdot uzdevumu | Ļauj darbiniekiem piešķirt uzdevumus konkrētiem lietotājiem, nosakot termiņu un prioritāti. | Tikai darbiniekiem. |
| Darbību žurnāls | Sniedz piekļuvi sarakstam ar darbībām, kas veiktas sistēmā (piem., inventāra izmaiņas, uzdevumu statuss). | Tikai Administrators |
| Ziņojums | Ļauj lietotājam nosūtīt ziņojumu administratoram (piemēram, par bojājumu vai problēmu). | Visi lietotāji |
| Iestatījumi | Piekļuve konta iestatījumiem (parole, e-pasts, paziņojumu iestatījumi). Funkcionalitāte mainās atkarībā no statusa. | Visi lietotāji |

Apstrāde:

Sākotnējā ielādē:

* Kad lietotājs pieslēdzas lietotnei, sistēma nosaka viņa statusu (Lietotājs vai Administrators).
* Sānu josla tiek dinamiski ģenerēts, parādot tikai tās sadaļas, kurām lietotājam ir piekļuve.

Noklikšķinot uz sadaļas:

* Sistēma nosūta pieprasījumu firebase serverim, lai ielādētu nepieciešamos datus konkrētajai funkcijai.
* Apstrādes darbības katrai sadaļai:
  + Jauns aprīkojums: atveras veidlapa inventāra pievienošanai, ļaujot aizpildīt informāciju un augšupielādēt attēlu.
  + Uzdevumi Parāda: piešķirto uzdevumu sarakstu, filtrējot pēc statusa vai termiņa.
  + Uzdot uzdevumu: atveras uzdevuma pievienošanas logs ar lietotāju sarakstu un iespējām definēt svarīgumu, termiņu, aprakstu.
  + Darbību žurnāls: Parāda darbību vēsturi, piemēram, pēdējās izmaiņas inventārā.
  + Ziņojums: Veras logs, kurā lietotājs var nosūtīt ziņojumu administratoram.
  + Iestatījumi: Parāda lietotāja konta iestatījumus, kur var mainīt paroli, e-pastu vai paziņojumu preferences.

Izvaddati: Veiksmīgi atveras izvēlētā sadaļa

### Inventāra instrukciju apskate

Mērķis: Lietotājam tiek nodrošināta iespēja ātri apskatīt izvēlētā inventāra instrukcijas, lai iegūtu detalizētu informāciju par tā lietošanu vai drošības prasībām.

Ievaddati:

1. **tabula**

**Sarakstu attēlošana**

|  |  |  |
| --- | --- | --- |
| Nosaukums | Obligāts | Piezīmes |
| Inventāra ID | Jā | Atlasīts no inventāra saraksta. |
| Nospiežot pogu "Instrukcija" | Jā | Aktivizē instrukcijas attēlošanas funkciju. |

Apstrāde: Kad lietotājs atrodas inventāra detalizācijas skatā un nospiež pogu **"**Instrukcija**"**:

1. Sistēma pārbauda, vai instrukcija ir pieejama attiecīgajam inventāram.
2. Ja instrukcija ir pieejama, tā tiek ielādēta un parādīta lietotājam.
3. Ja instrukcija nav pieejama, tiek rādīts paziņojums, piemēram: **"**Šim inventāram nav pievienota instrukcija.**"**

Izvaddati: Lietotājam tiek parādīta inventāra instrukcija šādā formā:

* Instrukcijas teksts un attēli.
* Norādes uz drošības prasībām.
* Darbību soļi vai pamācības video.

### Inventāra izmaiņas: pievienošana, atņemšana un rediģēšana

Mērķis: Lietotājam ir iespēja vienkārši apskatīt pieejamo staciju sarakstu.

Ievaddati:

1. **tabula**

**Inventāru izmaiņas**

|  |  |  |
| --- | --- | --- |
| Nosaukums | Obligāts | Piezīmes |
| Inventāra ID | Jā | Unikāls identifikators inventāra vienībai. |
| Izmaiņas veids | Jā | “Pievienot,” “Atņemt,” vai “Rediģēt.” |
| Maināmais daudzums | Jā | Skaitliska vērtība, ko pievieno, atņem vai iestata. |

Apstrāde:

1. Lietotājs izvēlas inventāra vienību no saraksta vai meklēšanas.
2. Norāda izmaiņu veidu (piemēram, pievienot daudzumu, atņemt vai pilnībā mainīt).
3. Sistēma veic šādas pārbaudes:

* Maksimuma pārbaude: Nepieļauj pārsniegt iepriekš definēto maksimālo daudzumu.
* Minimuma pārbaude: Nepieļauj inventāra daudzumu nokrist zem nulles vai kritiskā minimuma vērtības.

1. Ja pārbaudes iziet veiksmīgi, inventāra daudzums tiek atjaunināts.
2. Ja pārbaudes netiek izturētas, lietotājs saņem paziņojumu:

* "Inventāra daudzums nevar pārsniegt maksimālo atļauto vērtību."
* "Inventāra daudzums nevar būt negatīvs."

Izvaddati: Atjaunināta inventāra informācija, ieskaitot jaunā daudzuma vērtību. Paziņojums: “Inventāra daudzums veiksmīgi atjaunināts.”

Kļūdas ziņojums ar atbilstošu iemeslu.

### Staciju saraksta attēlošana

Mērķis: Lietotājam ir iespēja vienkārši apskatīt pieejamo staciju sarakstu.

Ievaddati:

1. **tabula**

**Sarakstu attēlošana**

|  |  |  |
| --- | --- | --- |
| Nosaukums | Obligāts | Piezīmes |
| Pogu nospiešana | Jā | Aktivizē saraksta pieprasījumu no datubāzes. |

Apstrāde: Sistēma pieprasa visu staciju datus no datubāzes. Saraksts tiek dinamiski ielādēts un attēlots lietotāja ekrānā.

Izvaddati: Parādīts pilns staciju saraksts ar galvenajiem atribūtiem nosaukums, attēls un stacijas ID.

### Inventāra saraksta atsvaidzināšana

Mērķis: Nodrošināt lietotājiem iespēju manuāli atsvaidzināt inventāra sarakstu, lai iegūtu jaunākos datus no datubāzes bez nepieciešamības atjaunot visu lapu.

Ievaddati:

1. **Tabula**

**Atsvaidzināšana dati**

|  |  |  |
| --- | --- | --- |
| Nosaukums | Obligāts | Piezīmes |
| Nospiežot pogu | Jā | Aktivizē datu ielādē no datubāzes. |

Apstrāde: Lietotājs atrodas inventāra sadaļā un nospiež pogu “Atsvaidzināt”

Sistēma veic šādas darbības:

* Nosūta pieprasījumu uz serveri, lai iegūtu jaunākos inventāra datus no datubāzes.
* Pārbauda, vai pieprasījums ir veiksmīgi izpildīts.
* Atjauno inventāra sarakstu ar iegūtajiem datiem.

Ja pieprasījums neizdodas, lietotājs saņem kļūdas paziņojumu, piemēram, "Neizdevās ielādēt jaunākos datus. Lūdzu, mēģiniet vēlreiz."

Izvaddati:

Veiksmīgas darbības gadījumā:

* Tiek parādīts atjaunināts inventāra saraksts.
* Paziņojums: “Inventāra saraksts veiksmīgi atsvaidzināts.”

Kļūdu gadījumā:

* Paziņojums: “Neizdevās ielādēt jaunākos datus. Pārbaudiet savu savienojumu.”

### Jauns aprīkojuma pārvaldība

Mērķis: Efektīva inventāra pievienošana, rediģēšana un uzskaite.

Ievaddati:

1. **Tabula**

**Sarakstu attēlošana**

|  |  |  |
| --- | --- | --- |
| Nosaukums | Obligāts | Piezīmes |
| Inventāra tips | Jā | Aktīvs vai patēriņa materiāls. |
| Koda numurs | Jā | Unikāls iekšējais kods. |
| Nosaukums | Jā | Inventāra nosaukums. |
| Mērvienība | Jā | Piem., gabali, litri. |
| Skaits | Jā | Pieejamais inventāra daudzums. |
| Minimālais/maksimālais/ kritiskais krājums | Jā | Krājuma uzraudzība |
| Apraksts | Jā | Inventāra papildu informācija. |
| Izglītības kods | Jā | Saistība ar izglītības standartiem. |
| Attēls | Jā | Inventāra vizuālais attēlojums. |

Apstrāde: Darbinieki ievada inventāra detaļas vai veic izmaiņas esošajā ierakstā. Sistēma pārbauda, vai ievadītā informācija atbilst noteiktajiem standartiem.

Izvaddati: Atjaunots inventāra saraksts ar aktuāliem datiem.

### Uzdevumu saraksts

Mērķis: Darbiniekiem ir iespēja vienkārši apskatīt pieejamo staciju sarakstu.

Ievaddati:

1. **Tabula**

**Uzdevumu sarakstu attēlošana**

|  |  |  |
| --- | --- | --- |
| Nosaukums | Obligāts | Piezīmes |
| Apraksts | Jā | Papildus informācija par uzdevumu. |
| Termiņš | Jā | Datums un laiks līdz kuram uzdevums jāizpilda. |
| Prioritāte | Jā | Zema, vidēja vai augsta. |
| Status | Jā | Pabeigts vai nepabeigts |
| Piešķirotājs | Jā | Kas piešķīra uzdevumu |

Apstrāde: Uzdevuma dati tiek ievadīti un glabāti sistēmā. Lietotājs dzēš piešķirtos uzdevumus, kad tie tiek pabeigti

Izvaddati: Uzdevumu saraksts

### Uzdot uzdevumu

Mērķis: Šīs funkcionalitātes mērķis ir ļaut lietotājiem izveidot un piešķirt uzdevumus citiem lietotājiem, nodrošinot, ka uzdevumu var piešķirt tikai tiem lietotājiem, kuri nav administratori, un ka uzdevuma termiņš nav pagātnē.

Ievaddati:

1. **Tabula**

**Sarakstu attēlošana**

|  |  |  |
| --- | --- | --- |
| Nosaukums | Obligāts | Piezīmes |
| Kam | Jā | Lietotājs vai lietotāju grupa, izņemot administratoru. |
| Kalendārs | Jā | Datums un laiks, kad uzdevums jāizpilda. Nav iespējams izvēlēties pagātnes datumu. |
| Steidzamība | Jā | Steidzami vai nav steidzami. |
| Apraksts | Jā | Papildus informācija par uzdevumu. |

Apstrāde: Lietotājs aizpilda nepieciešamos ievaddatus. Sistēma pārbauda datus – apskatās, vai lietotājs nav administrators un vai kalendāra datums nav pagātnes datums. Ja visi dati ir derīgi, uzdevums tiek veiksmīgi saglabāts un piešķirts darbiniekam.

Izvaddati:

1. Paziņojums par nepareiziem datiem
2. Uzdevums veiksmīgi piešķirts

### Darbības žurnāls

Mērķis: Šīs funkcionalitātes mērķis ir ļaut lietotājiem pārskatāmi aplūkot visus veiktos darbus un saņemt informāciju par lietotāja aktivitātēm aplikācijā.

Ievaddati:

1. **Tabula**

**Darbības žurnāla dati**

|  |  |  |
| --- | --- | --- |
| Nosaukums | Obligāts | Piezīmes |
| Sakārtošana pēc | Jā | Kritēriji darbību žurnāla kārtošanai. |

Apstrāde: Sistēma saņem lietotāja izvēlēto kārtošanas kritēriju un sakārto darbības žurnālu atbilstoši tam. Lietotāji var izvēlēties starp dažādiem kārtošanas veidiem.

Izvaddati: Paziņojums par to kā darbības žurnāls tika sakārtots

### Ziņojums par problēmu

Mērķis: Šīs funkcionalitātes mērķis ir ļaut lietotājiem ziņot par problēmām, kas saistītas ar telpām, stacijām vai inventāru, un nosūtīt ziņojumu e-pastā visiem darbiniekiem un administratoriem.

Ievaddati:

1. **Tabula**

**Problēmas dati**

|  |  |  |
| --- | --- | --- |
| Nosaukums | Obligāts | Piezīmes |
| Telpas nr. | Jā | Norāda telpas numuru, kurā konstatēta problēma. |
| Stacija | Jā | Norāda staciju, kurā konstatēta problēma. |
| Inventārs | Jā | Norāda problēmu saistībā ar konkrēto inventāru. |
| Apraksts | Jā | Detalizēts problēmas apraksts. |

Apstrāde: Lietotājs aizpilda nepieciešamos ievaddatus un nosūta ziņojumu. Sistēma nosūta ziņojumu uz e-pastu visiem darbiniekiem un administratoriem.

Izvaddati: Ziņojums veiksmīgi nosūtīts visiem darbiniekiem un administratoriem.

### Iestatījumi

Mērķis: Šīs funkcionalitātes mērķis ir nodrošināt lietotājiem iespēju pielāgot lietotnes iestatījumus un pārvaldīt savu kontu, piemēram, mainīt valodu, motīvu, izrakstīties no konta vai atiestatīt paroli.

Ievaddati:

1. **tabula**

**Iestatījuma dati**

|  |  |  |
| --- | --- | --- |
| Nosaukums | Obligāts | Piezīmes |
| Valoda | Nē | Lietotāji var izvēlēties starp latviešu (LV) un angļu (EN) valodu. |
| Motīvs | Nē | Lietotāji var izvēlēties starp FABLAB krāsām vai dabasmāju motīvu. |
| Izrakstīties | Nē | Lietotāji var izrakstīties no sava konta. |
| Paroles atiestatīšana | Nē | Lietotāji var atiestatīt savu paroli. |
| Dzēst nesen izmantotās stacijas | Nē | Lietotāji var izdzēst savu nesen izmantotas stacijas lai tas sāka skaitīt no jauna |

Apstrāde: Lietotājs veic izvēli starp iestatījumiem, un sistēma saglabā attiecīgos izmaiņu datus. Pēc izmaiņu veikšanas, lietotājs var turpināt lietot lietotni ar jauniem uzstādījumiem.

Izvaddati:

1. Valoda veiksmīgi mainīta
2. Motīvs veiksmīgi mainīts
3. Lietotājs veiksmīgi izrakstījies
4. Parole veiksmīgi atiestatīta
5. Nesen izmantotās stacijas tika izdzēstas

### Stacijas kārtošana

Mērķis: Saglabāt lietotāja izvēlētās stacijas, lai uzlabotu navigācijas pieredzi un rādītu biežāk izmantotās stacijas saraksta augšgalā.

Ievaddati:

1. **Tabula**

**Sarakstu attēlošana**

|  |  |  |
| --- | --- | --- |
| Nosaukums | Obligāts | Piezīmes |
| Stacija | Jā |  |

Apstrāde: Katra reize, kad lietotājs uzspiež uz kādas stacijas, tā tiek automātiski pārvietota uz saraksta augšgalu.

Izvaddati: Stacijas saraksts ar izmainītu kārtojumu

### Kalendāra notikuma statusa maiņa

Mērķis: Parādīt vizuāli uzdevumu statusus kalendārā, ļaujot lietotājiem viegli identificēt notikuma stāvokli.

Ievaddati:

1. **Tabula**

**Sarakstu attēlošana**

|  |  |  |
| --- | --- | --- |
| Nosaukums | Obligāts | Piezīmes |
| Kalendāru nospiešanu | Jā |  |

Apstrāde: Kalendāra dienas tiek automātiski krāsotas atkarībā no uzdevumu statusa.

Izvaddati: Dienas krāsa mainās uz sarkanu, zaļu, oranžu vai violetu, atbilstoši statusam.

* Sarkans - notikums ir atcelts.
* Zaļš - notikums ir pieņemts.
* Oranžs - notikums ir gaidāms.
* Violets - notikums ir pabeigts.

### Uznirstošais logs uzdevumiem kalendārā

Mērķis: Ļaut lietotājiem redzēt visus uzdevumus konkrētajā dienā un ātri mainīt statusus.

Ievaddati:

1. **Tabula**

**Sarakstu attēlošana**

|  |  |  |
| --- | --- | --- |
| Nosaukums | Obligāts | Piezīmes |
| Diena | Jā | Jābūt notikums dienā |

Apstrāde: Nospiežot uz dienas, kurā ir uzdevumi, tiek atvērts uznirstošais logs.

Izvaddati: Logs ar notikumiem un tā datiem, kur zem katra notikuma ir pogas statusa maiņai (pieņemt, noraidīt, pabeigts).

### Statusa maiņa uzdevumiem

Mērķis: Atļaut lietotājiem mainīt uzdevumu statusu tieši no kalendāra, lai atvieglotu uzdevumu pārvaldību.

Ievaddati:

1. **Tabula**

**Sarakstu attēlošana**

|  |  |  |
| --- | --- | --- |
| Nosaukums | Obligāts | Piezīmes |
| Pogu nospiešana | Jā | Pieņemt noraidīt vai pabeigt |

Apstrāde: Lietotājs izvēlas statusu uznirstošajā logā no pieejamajām opcijām (pieņemt, noraidīt, pabeigts).

Izvaddati: Statuss veiksmīgi mainīts uzdevumam, piemēram, pieņemts, noraidīts vai pabeigts.

### Lietotāja statusa maiņa

Mērķis: Atļaut administratoriem mainīt citu cilvēka konta statusu.

Ievaddati:

1. **Tabula**

|  |  |  |
| --- | --- | --- |
| Nosaukums | Obligāts | Piezīmes |
| Konta vārds | Jā | No izvēles saraksta ir atrādīti visi konti kuri nav dzēsti vai nav pašreizējais konts |
| Statuss | Jā | **“**Darbinieks**”, “**Lietotājs**”, “**Administrators**”** |

Apstrāde: Izvēloties kontu tiek parādīts pašreizējais status. To var nomainīt uzspiežot uz nolaižamās izvēlnes, kurā tiek parādīts visas iespējamās izvēles.

Izvaddati: Konta statuss tiek veiksmīgi nomainīts.

### Lietotāja konta dzēšana

Mērķis: Atļaut administratoriem dzēst cita lietotāja kontu.

Ievaddati:

1. **Tabula**

|  |  |  |
| --- | --- | --- |
| Nosaukums | Obligāts | Piezīmes |
| Konta vārds | Jā |  |
| Dzēšanas poga | Jā | Nospiežot dzēst pogu izvelētais konts tiek izdzēsts |

Apstrāde: Izvēlēties to kontu kuru vēlās izdzēst pēc tam tiek nospiest poga “Dzēst” nospiežot uz tās pogas tiek parādīts uznirstošais logs kurā tiek pārjautāts vai tiešām vēlās izdzēst šo kontu.

Izvaddati: Konts ir veiksmīgi izdzēsts.

## Sistēmas nefunkcionālās prasības

1. Sistēmai jāreaģē uz lietotāja pieprasījumiem mazāk nekā 2 sekundēs.
2. Dati tiek šifrēti, un dati tiek aizsargāti pret uzbrucēju piekļuvi.
3. Lietotnei jābūt intuitīvai un ērti lietojamai arī nepieredzējušiem lietotājiem.
4. Lietotnei jādarbojas Android platformās.
5. Sistēmai jāspēj apstrādāt pieaugošu lietotāju un inventāra skaitu.
6. Kods jābūt dokumentētam, lai atvieglotu tālāku attīstību un uzturēšanu.
7. Datu dublējumkopijas jāveido automātiski katru dienu, lai novērstu datu zudumu.
8. Sistēmai jābūt pielāgojamai nākotnes funkcionalitātes paplašināšanai.
9. Sistēmai jābūt spējīgai nodrošināt piekļuvi vairākiem lietotājiem vienlaikus, nezaudējot veiktspēju.
10. Sistēmai jābūt viegli uzturamai un jāatbalsta regulāri atjauninājumi, lai saglabātu augstu drošības līmeni un uzlabotu funkcionalitāti.

## Gala lietotāja raksturiezīmes

### Administratori

Loma: Atbild par FABLAB staciju un resursu pārvaldību, sistēmas uzturēšanu un lietotāju pārraudzību.  
Raksturiezīmes:

* Laba tehniskā izpratne par FABLAB darbību un staciju funkcionalitāti.
* Nepieciešamība pēc efektīviem rīkiem inventāra un notikumu pārvaldībai.
* Vēlme ātri piekļūt svarīgai dokumentācijai un saņemt atskaites par sistēmas darbību.

### Darbinieki

Loma: Ikdienā izmanto stacijas un resursus projektu īstenošanai vai piedalās pasākumos un meistarklasēs.  
Raksturiezīmes:

* Pamata tehniskās zināšanas par FABLAB aprīkojumu un darba procesiem.
* Nepieciešamība rezervēt resursus vai piekļūt informācijai par pieejamajiem instrumentiem un iekārtām.
* Vēlme piedalīties pasākumos un sekot līdzi stacijas aktivitātēm.

# Izstrādes līdzekļu, rīku apraksts un izvēles pamatojums

Šajā sadaļā ir aprakstīti un pamatoti izvēlētie izstrādes līdzekļi. Aprakstītas arī alternatīvas izvēles izvēlētiem līdzekļiem.

## Izvēlēto risinājuma līdzekļu un valodu apraksts

### Java

Apraksts: Java ir programmēšanas valoda, kas ir plaši izmantota mobilo lietotņu izstrādē, īpaši Android vidē, pateicoties tās platformneatkarībai, objektorientētajai struktūrai un lielajai izstrādātāju kopienai.

Pamatojums: Java nodrošina augstu stabilitātes līmeni, plašu dokumentācijas un bibliotēku atbalstu, kā arī tai ir liela un aktīva izstrādātāju kopiena, kas atvieglo problēmu risināšanu un zināšanu apmaiņu. Turklāt Java jau ilgstoši ir bijusi primārā programmēšanas valoda Android lietotņu izstrādē, nodrošinot ciešu integrāciju ar Android SDK un plašu saderību ar esošajām Android ierīcēm, kas padara to par uzticamu un efektīvu izvēli lietotņu izstrādē šajā vidē.

### Firebase

Apraksts: Firebase ir Google mobilo lietojumprogrammu izstrādes platforma, kas palīdz jums izveidot, uzlabot un paplašināt lietotni, nodrošinot plašu rīku un pakalpojumu klāstu, tostarp autentifikāciju, reāllaika datubāzi, mākoņglabātuvi, lietotāju analītiku, paziņojumu nosūtīšanu un daudz ko citu, ļaujot izstrādātājiem koncentrēties uz lietotnes funkcionalitāti.

Pamatojums: Firebase ļauj vienkāršot datu glabāšanu un sinhronizāciju, kas ir būtiski dinamiskām un reāllaika lietotnēm. Tā integrācija ar Android Studio padara to par ērtu un efektīvu risinājumu izstrādātājiem.

### Android Studio

Apraksts: Android Studio ir oficiālā integrētā izstrādes vide (IDE) Google Android operētājsistēmai, kas ir veidota, izmantojot JetBrains IntelliJ IDEA programmatūru, un ir īpaši izstrādāta Android izstrādes vajadzībām.

Pamatojums: Android Studio piedāvā spēcīgus rīkus Android lietotņu izstrādei, tostarp vizuālo interfeisa redaktoru, emulatoru un dažādus testēšanas rīkus. Tā piedāvā dziļu Firebase integrāciju, padarot datu apstrādi un autentifikāciju efektīvāku.

## Iespējamo (alternatīvo) risinājuma līdzekļu un valodu apraksts

### Kotlin

Apraksts: Kotlin ir mūsdienīga programmēšanas valoda, kas ir pilnībā savietojama ar Java un piedāvā modernāku sintaksi. Tā ir viegli lietojama, palīdz rakstīt īsāku un saprotamāku kodu, un bieži tiek izmantota Android lietotņu izstrādē.

Pamatojums: Kotlin ir oficiālā programmēšanas valoda Android izstrādei un piedāvā vairākas priekšrocības salīdzinājumā ar Java, piemēram, īsāku kodu un uzlabotu drošību, taču Java tika izvēlēta, jo tai ir plašāks atbalsts un stabilitāte.

### Python

Apraksts: Python ir daudzpusīga valoda, kas tiek izmantota dažādos programmēšanas pielietojumos, tostarp mākoņdatošanas un mākslīgā intelekta risinājumos. Tā ir viegli saprotama un piemērota gan iesācējiem, gan pieredzējušiem programmētājiem.

Pamatojums: Lai gan Python nav primārā valoda Android izstrādei, to varētu izmantot servera puses loģikas vai analītikas rīku izstrādei, tomēr Firebase nodrošina nepieciešamo servera puses funkcionalitāti.

### SQLite

Apraksts: SQLite ir viegls un zemas prasības datubāzes risinājums, kas tiek izmantots datu glabāšanai vietējās lietotnēs. Tā ir integrēta bibliotēka, kas neprasa ārēju serveri, un tiek bieži izmantota mobilajās lietotnēs.

Pamatojums: SQLite piedāvā ātru un efektīvu datu glabāšanas risinājumu mobilajām lietotnēm, tomēr Firebase tika izvēlēts, jo tas nodrošina sinhronizāciju reāllaikā un mākonī glabātos datus, kas ir būtiski šim projektam gan arī tas ir viegli pievienojams android studio.

### Realm

Apraksts: Realm ir mobilajām lietotnēm paredzēts datubāzes risinājums, kas nodrošina ātru un efektīvu datu apstrādi. Tā atbalsta sinhronizāciju reāllaikā un ir īpaši piemērota dinamiskām, datu intensīvām lietotnēm.

Pamatojums: Realm ir lielisks risinājums, taču Firebase tika izvēlēts, jo tam ir labāka integrācija ar Android Studio, kā arī plašāks atbalsts izstrādātāju kopienā un plašāks funkcionalitātes klāsts, tostarp autentifikācija un mākonī glabāšana.

### IntelliJ IDEA

Apraksts: IntelliJ IDEA ir spēcīga IDE, ko izstrādājusi JetBrains, un kas atbalsta vairākas programmēšanas valodas, tostarp Java. Tā ir vispārēja IDE, kas piedāvā lieliskas funkcijas kodēšanas, testēšanas un uzturēšanas jomā.

Pamatojums: Lai gan IntelliJ IDEA ir spēcīga IDE, Android Studio tika izvēlēts, jo tā ir īpaši izstrādāta Android izstrādei un piedāvā papildu rīkus, piemēram, vizuālo interfeisa redaktoru un Android emulatoru, kas atvieglo izstrādes procesu.

### Qt Creator

Apraksts: Qt Creator ir IDE, kas tiek izmantota C++ valodas izstrādei un atbalsta arī lietotņu izstrādi vairākās platformās, tostarp Android un iOS. Tā piedāvā spēcīgu rīku klāstu lietotņu izstrādei un ir piemērota arī datu bāzu un tīkla risinājumiem.

Pamatojums: Qt Creator ir laba alternatīva daudz platformu lietotņu izstrādei, taču Android Studio tika izvēlēts, jo tā ir oficiālā IDE Android lietotņu izstrādei, piedāvājot dziļu integrāciju ar Android funkcijām un optimizētu darba vidi Android izstrādātājiem.

# Sistēmas modelēšana un projektēšana

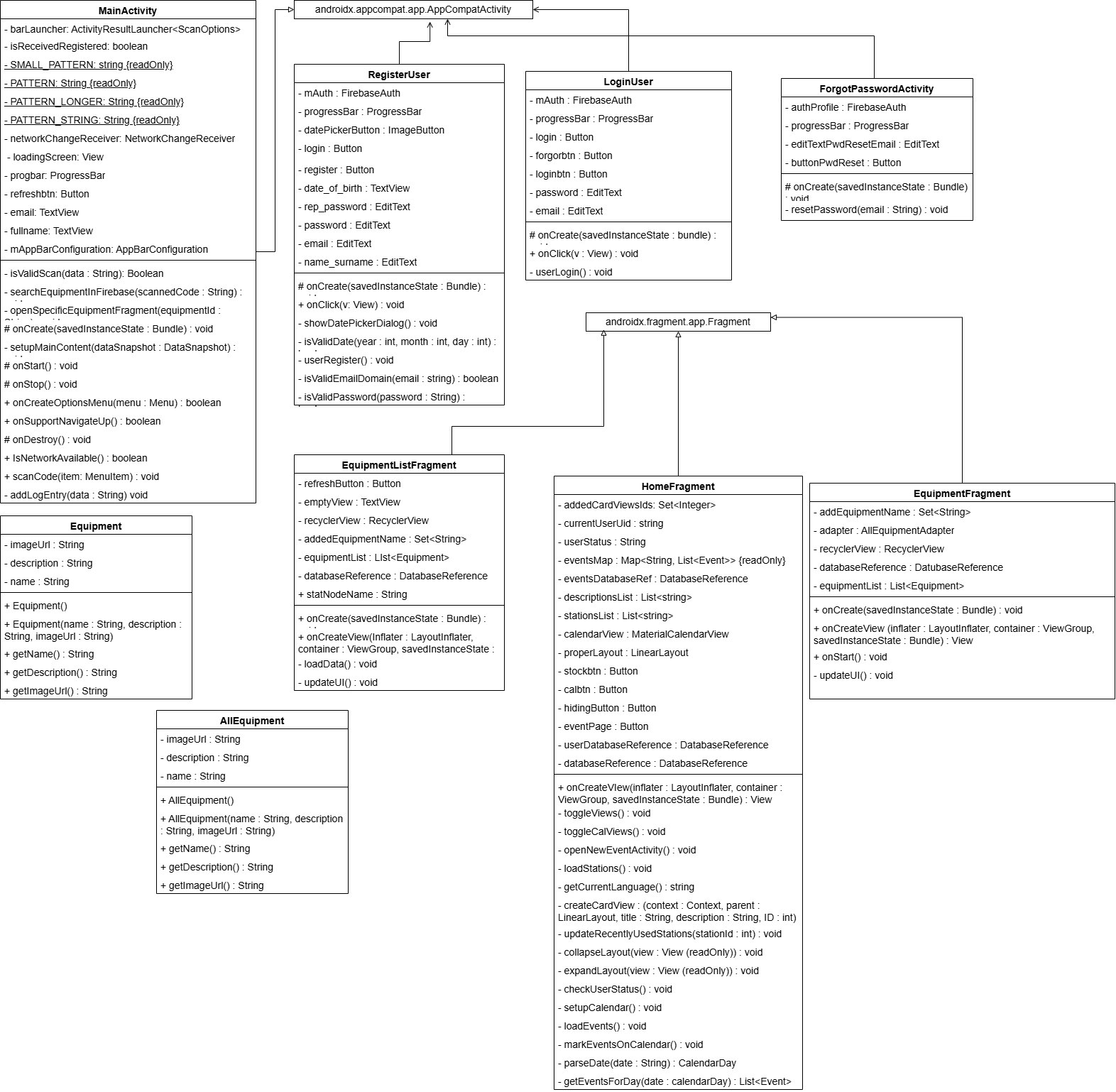
Šajā sadaļā tiek attēlota un aprakstīta sistēmas arhitektūra, iekļaujot tās galvenās sastāvdaļas un to savstarpējo mijiedarbību. Arhitektūra apraksta, kā sistēma ir strukturēta, kādi komponenti to veido un kā šie komponenti sadarbojas, lai nodrošinātu lietotnes funkcionalitāti un veiktspēju.

## Sistēmas struktūras modelis

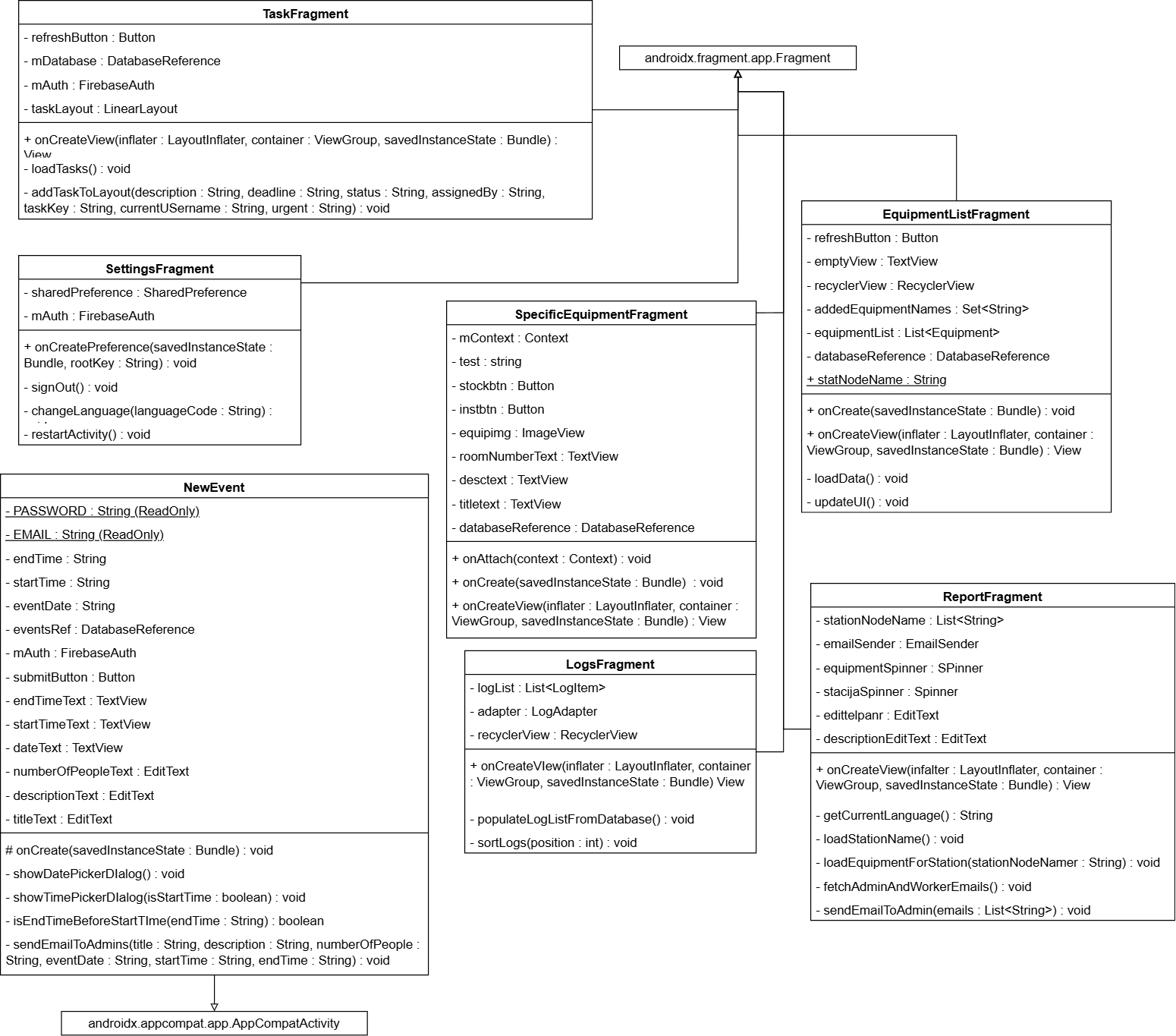
### Sistēmas struktūra (komponenšu diagramma)

**1. attēls komponenšu diagramma**

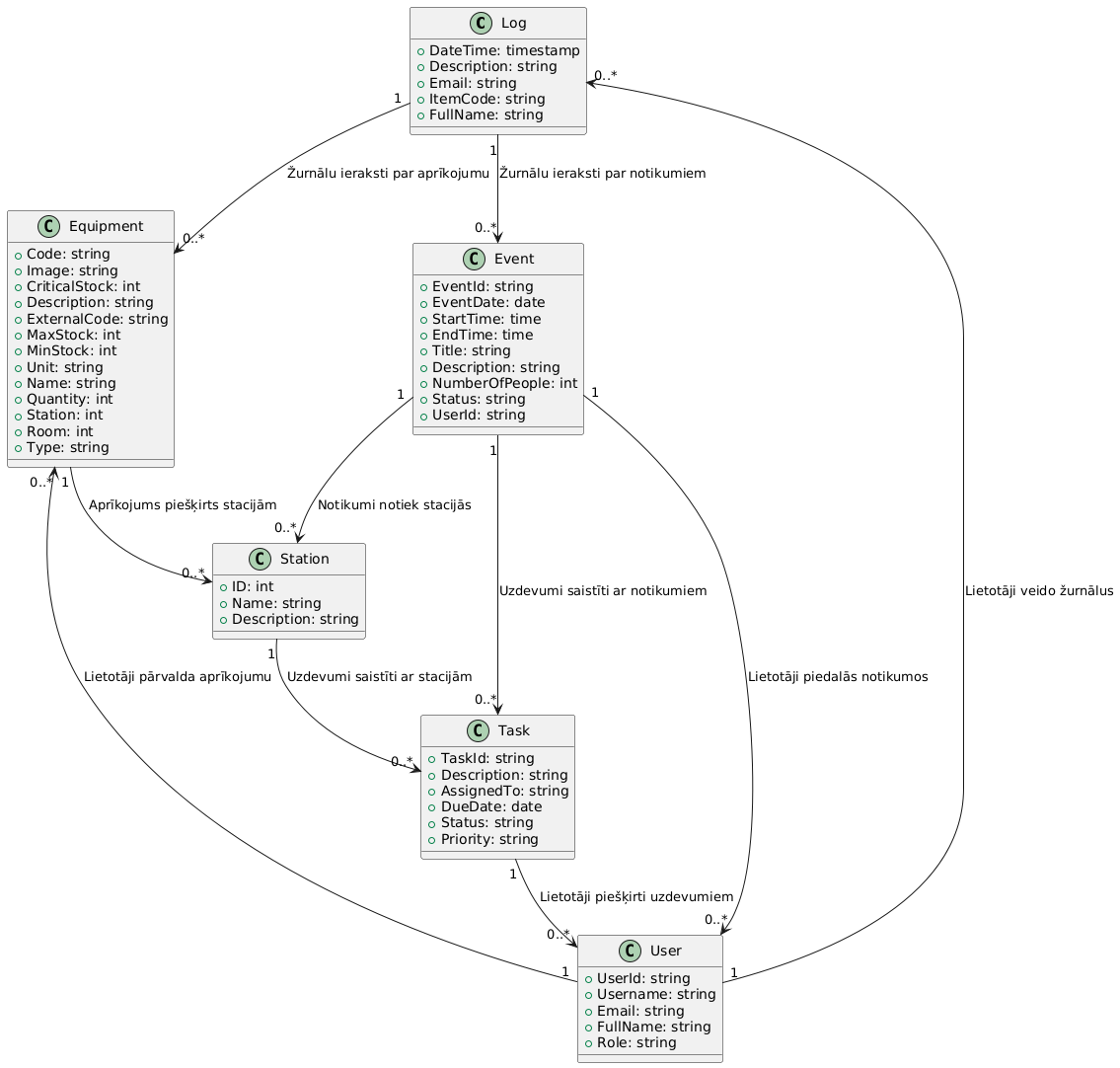
### Klašu diagramma



**2. attēls Autorizācijas sistēma un galvenās lapas klases diagramma**



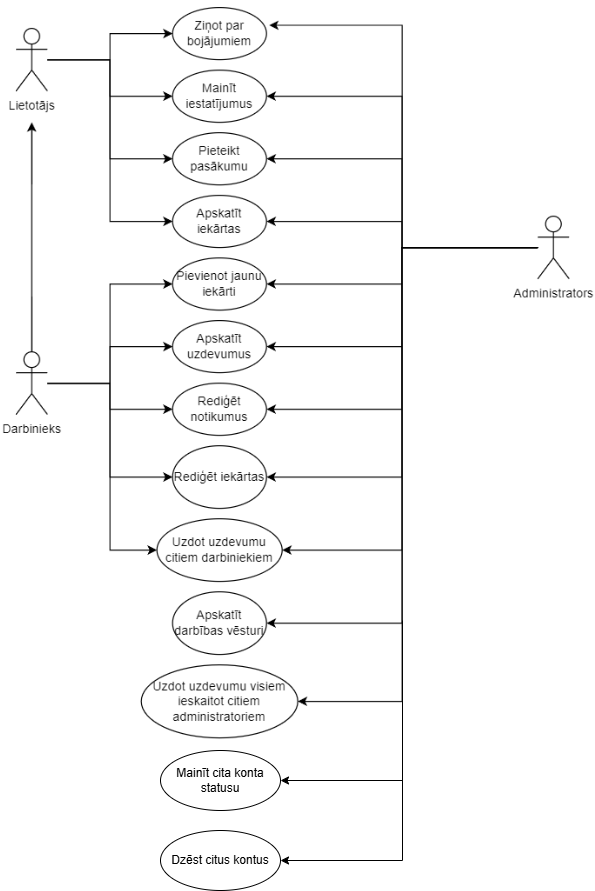
**3. attēls Atlikušās klašu diagrammas**



**4 . attēls. Firebase datubāzes klases diagramma**

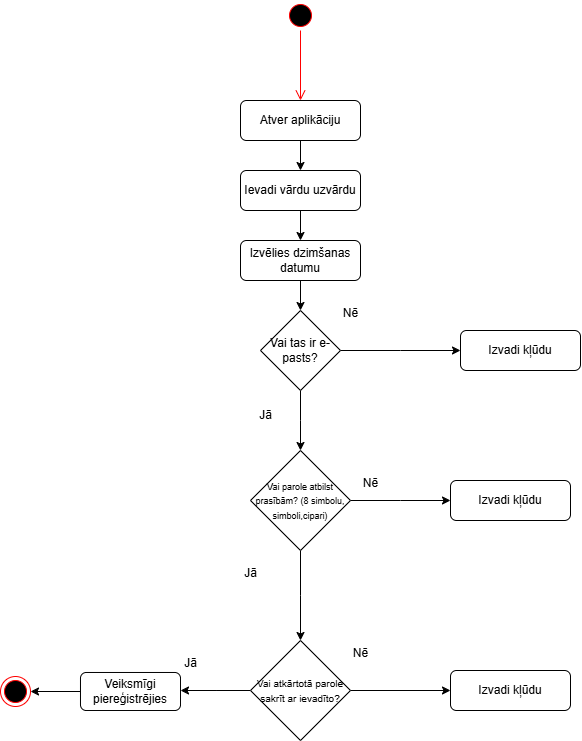
## Funkcionālais un dinamiskais sistēmas modelis

### Lietojumgadījumu diagramma

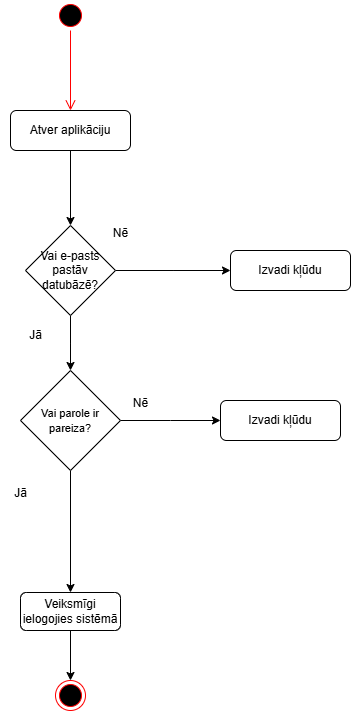


**5. attēls. Lietojumgadījumu diagramma**

### Aktivitāšu diagramma



**6. attēls. Reģistrācijas aktivitāšu diagramma**

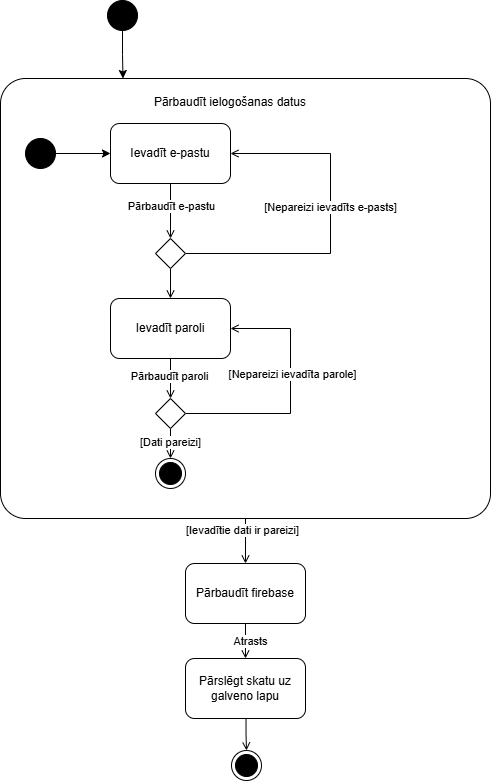


**7. attēls Ielogošanas sistēma aktivitāšu diagramma**

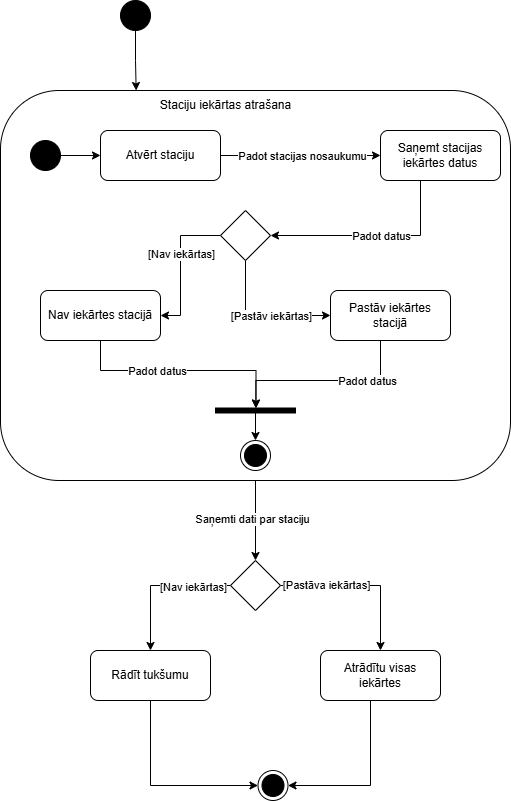


**8. attēls Jaunās iekārtas izveidošana aktivitāšu diagramma**

### Stāvokļu diagramma

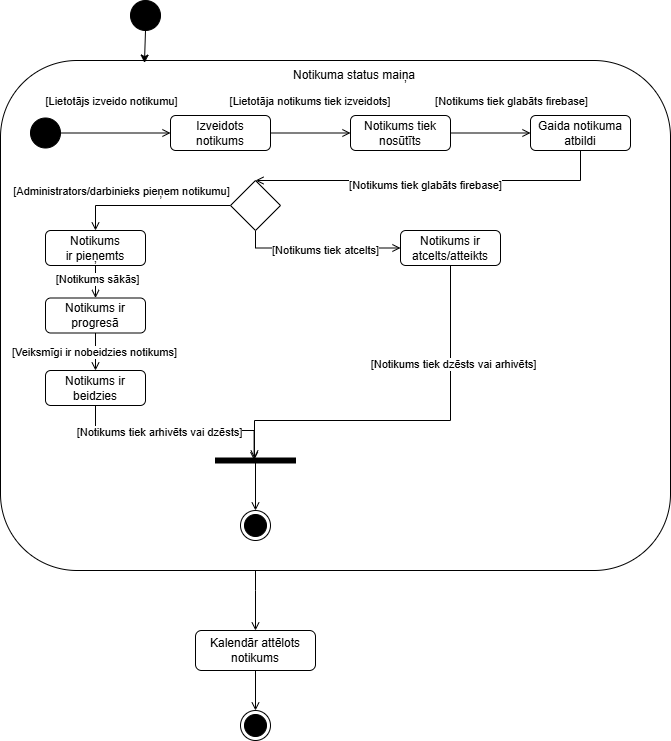


**9. attēls Ielogošanas sistēmas stāvokļu diagramma**



**10. attēls staciju iekārtas meklēšana stāvokļu diagramma**

## Datu struktūru apraksts



**11. attēls notikuma statusa maiņa stāvokļu diagramma**

Datu struktūra tiek veidota, balstoties uz Firebase NoSQL modeli, kur dati tiek kārtoti kolekcijās un dokumentos. Katrā kolekcijā var būt vairāki dokumenti, un katrs dokuments var saturēt strukturētus datus ar atslēgu-vērtību pāriem.

Šajā projektā Firebase tiek izmantots vairākām galvenajām funkcijām:

* **Lietotāju dati –** tiek glabāti kolekcijā *users* (lietotāji), kur katram lietotājam ir unikāls UID kā dokumenta ID. Lietotāja dokumentā tiek saglabāta informācija, piemēram, vārds, e-pasts, statuss (administrators, darbinieks, lietotājs) un citas nepieciešamās detaļas.
* **Notikumu pārvaldība** – kolekcijā *events* (notikumi) tiek glabāti notikumi, kas ir pieejami lietotājiem. Katrs notikums satur informāciju par laiku, vietu, dalībnieku skaitu, statusu (apstiprināts, noraidīts, pabeigts) un citu būtisku informāciju.
* **Stacijas un aprīkojums –** kolekcijā *stations* (stacijas) tiek glabāti staciju dati ar unikāliem ID, kas saistīti ar konkrētu aprīkojumu kolekcijā *equipment* (iekārtas). Aprīkojuma dokumentos tiek saglabāti lauki, piemēram, *Nosaukums*, *Kods*, *Skaits*, *Tips*, un kritiskā krājumu informācija.
* **Kartīšu pozīciju saglabāšana –** lietotāja interfeisā pielāgotās kartīšu pozīcijas tiek glabātas Firebase datubāzē, lai nodrošinātu konsekventu izkārtojumu pēc atkārtotas ielādes.

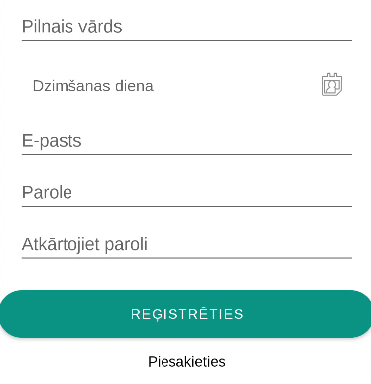
# Lietotāju ceļvedis

Šī lietotne ir paredzēta inventāra pārvaldībai, ļaujot darbiniekiem apskatīt, pievienot un pārvaldīt aprīkojumu, veidot uzdevumus un notikumus, kā arī apskatīt atskaites.

## Lietotāja konts un autentifikācija

### Reģistrēšanās

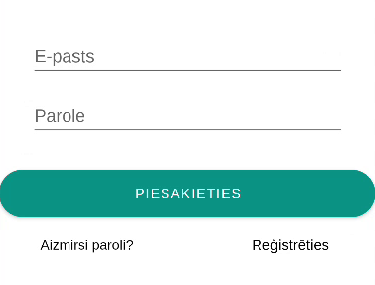
* Atveriet lietotni.
* Noklikšķiniet uz "Reģistrēties".
* Ievadiet savu e-pastu, paroli un citus nepieciešamos datus.
  + Parolei jābūt vismaz 8 rakstzīmēm, jāiekļauj vismaz 1 lielais burts, 1 mazais burts, 1 cipars un 1 simbols.
* Nospiežot pogu “Reģistrēties” tiek reģistrēts konts.



**12. attēls Reģistrēšanas skats**

### Pierakstīšanās

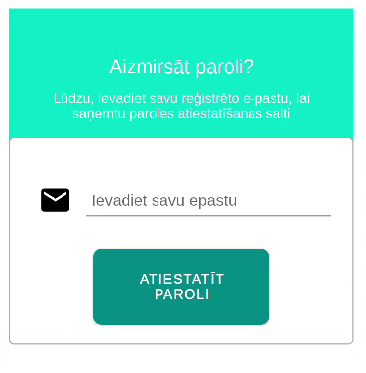
* Ievadiet savu e-pastu un paroli.
* Noklikšķiniet uz "Pieslēgties".



**13. attēls Pieteikšanās skats**

### Aizmirsi paroli

* Noklikšķiniet uz "Aizmirsu paroli".
* Ievadiet savu e-pasta adresi, lai saņemtu paroles atiestatīšanas saiti.
* Atverot paroles atiestatīšanas saiti, tiek pieprasīts ievadīt jaunu paroli.

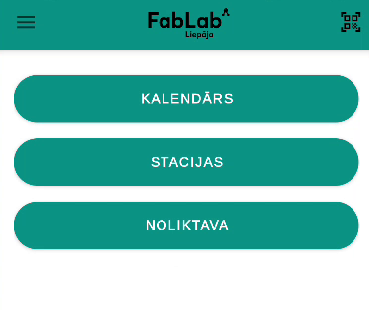


**14. attēls Aizmirsi paroli skats**

## Sākuma ekrāns un navigācija

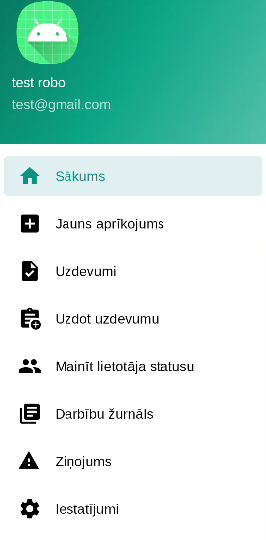
### Galvenais ekrāns

* Kalendārs – rāda plānotos notikumus.
* Stacijas - rāda visas stacijas
* Noliktava - rāda visas iekārtas, kas atrodas datubāzē (tikai redzams darbiniekiem un administratoriem).



**15. attēls Galvenā ekrāna skats**

### Sānu josla



**16. attēls Sānu joslas skats**

* Jauns aprīkojums – rāda jauna aprīkojuma pievienošanu. (darbiniekiem un administratoriem)
* Uzdevumi – rāda uzdotos uzdevumus. (darbiniekiem un administratoriem)
* Uzdot uzdevumu – rāda dot iespēju uzdot uzdevumu. (darbiniekiem un administratoriem)
* Darbības žurnāls – rāda darbības žurnālu. (administratoriem)
* Ziņojums – rāda ziņojuma padošanu.
* Iestatījumi – rāda iespējamos iestatījumus.

## Inventāra pārvaldība



### Iekārtas apskatīšana (Parasts lietotājs)

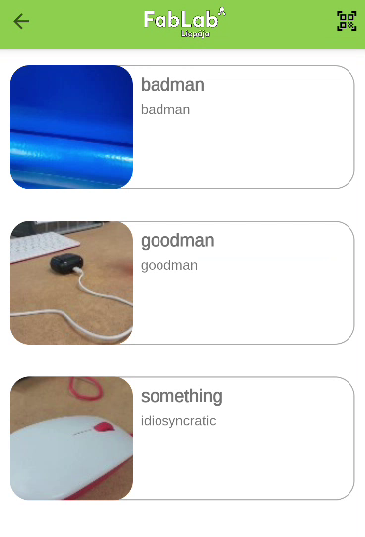
* Uzspiest uz pogu “Stacija” (skat. attēls 17 bultiņa 1)
  + Tiek parādīts visas stacijas.
* Izvēlēties staciju (skat. attēls 17 bultiņa 2)
  + Pēc staciju izvēles tiek rādītas tās iekārtas, kas atrodas izvēlētajā stacijā.
* Izvēlēties iekārtu



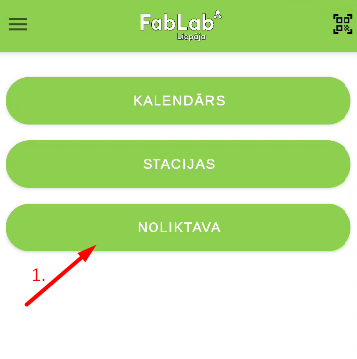
**17. attēls Stacijas apskatīšanas skats**

### Iekārtas apskatīšana un tā rediģēšana (Darbinieks vai administrators)

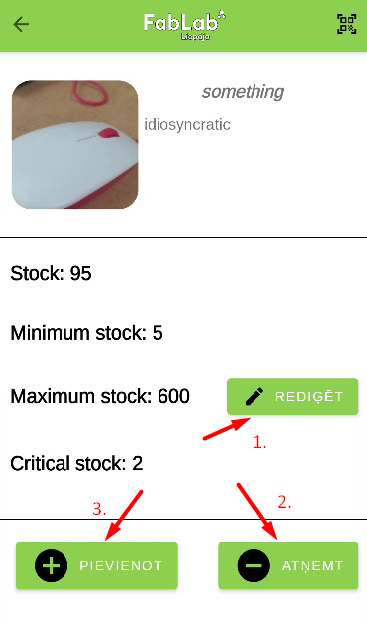
* Uzspiest uz pogu “Noliktava” (skat. 18. attēlu bultiņu ar ciparu 1)
  + Tiek parādītas visas iekārtas, kas pastāv.
* Izvēlēties iekārti (skat. 19. attēlu)
* Nospiest uz pogu “Inventārs”
* Nomainīt tās iekārtu vēlamos datus
  + Rediģēt iekārtu maksimālo krājumu (skat. 21. attēlu 1. bultiņu)
  + Noņemt daudzum no iekārtu (skat. 21.attēlu 2. bultiņu)
  + Pievienot daudzumu pie iekārtu (skat. 21. attēlu 3. bultiņu)



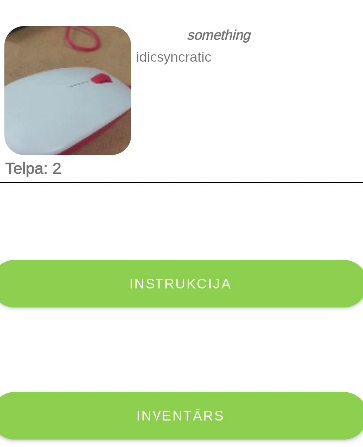
**19. attēls Iekārtu izvēlēšana**



**18. attēls Iekārtu apskatīšana**



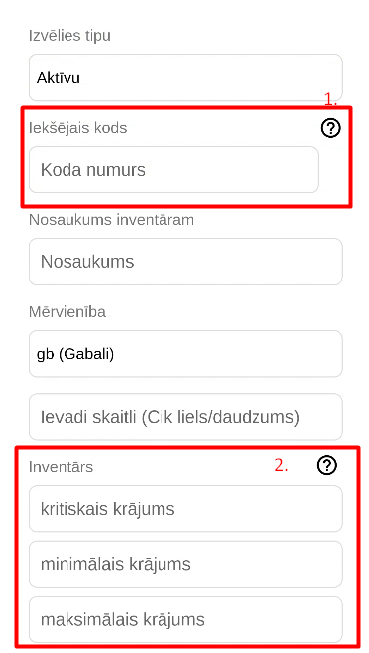
**21. attēls Iekāršu datu rediģēšana**



**20. attēls Iekāršu detalizēts skats**

### Iekārtu pievienošana (Darbinieks vai administrators)

* Atvērt sānu joslu
* Nospiest uz pogu “Jauns aprīkojums”
* Aizpildīt datus
  + Pie iekšējā koda lauciņa jāievada pēc modeļa (skat. 22 attēli skaitlis 1.) Modelī pirmais cipars ir telpas numurs, otrais cipars ir stacijas numurs, trešais cipars ir iekārtas veids: 1 – aktīvs vai 2 – patēriņš, un pēc tam seko preces nosaukums. Zemāk ir piemēri, pēc kādiem principiem tas jāievada:
    - 10\_5\_1\_test1234



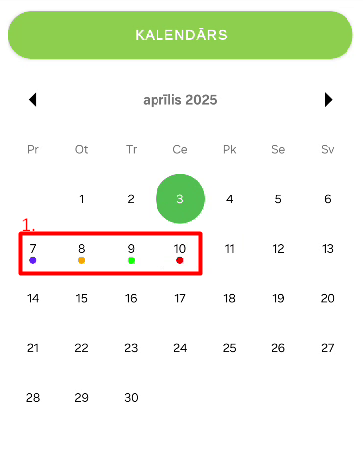
**22. attēls Iekārtu izveidošanas skats**

* + - 10\_15\_2\_test4321
    - 1\_15\_1\_test2135
    - 5\_3\_1\_test4124
  + Pie inventāra krājuma lauciņa jāseko šādam principam (skat. 22. attēlu skaitlis 2.): kritiskais krājums < minimālais krājums < maksimālais krājums.
  + Izglītības kodam jābūt 8 ciparu kodam.
  + Attēlu pievienošanas poga ir aktīva tikai tad, kad visi dati ir aizpildīti pareizi.
* Nospiest “Pievienot”

## Kalendāra notikuma statusa rediģēšana

### Notikuma apskatīšana

* Uzspiest uz pogu “Kalendārs”
* Uzspiest uz dienu kurā notiek notikumu
  + Var redzēt, kurās dienās ir notikums, pēc krāsas, kas atrodas zem datuma (skat 23. attēlu 1. skaitli).
    - Sarkans - notikums ir atcelts.

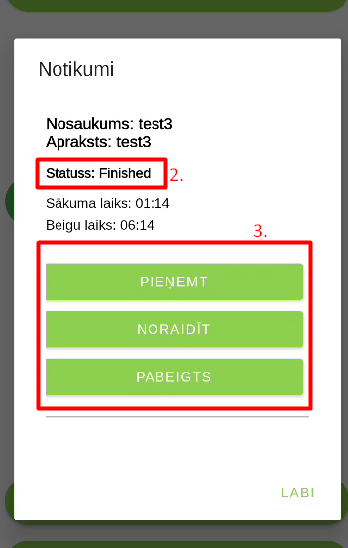


**23. attēls Kalendāra notikuma statusi**

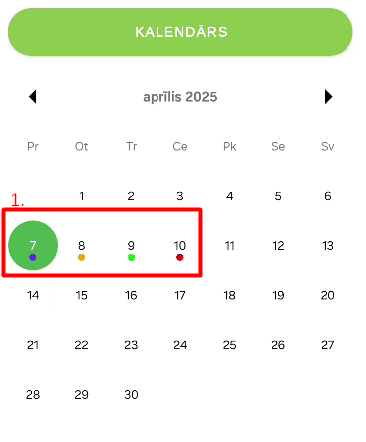
* + - Zaļš - notikums ir pieņemts.
    - Oranžs - notikums ir gaidāms.
    - Violets - notikums ir pabeigts.
* Pēc tās nospiešanas jāparādās uznirstošajam logam, kurā attēloti notikuma dati.

### Notikuma status maiņa (Darbinieki vai administratori)

* Uzspiest uz pogu “Kalendārs”
* Uzspiest uz dienu kurā notiek notikumu (skat. 24. attēlu 1. skaitlis)
  + Var redzēt, kurās dienās ir notikums, pēc krāsas, kas atrodas zem datuma.
* Izvelēties uz kuru statusu mainīt “Pieņemt”, “Noraidīt”, “Pabeigts” (skat. 25. attēlu 3. skaitli)
  + Nospiežot uz jebkuras no minētajām pogām, tiek nomainīts notikuma statuss un tā krāsa. (skat. 25. attēlu 2. skaitli)



**25. attēls Notikuma statusa maiņa**



**24. attēls notikuma izvēle**

# Testēšanas dokumentācija

Šajā nodaļā tiek parādīti un aprakstīti veidi, kā aplikācija tika testēta, un ar kādiem paņēmieniem tika nodrošināta tās funkcionalitāte, veiktspēja un drošība.

## Izvēlētās testēšanas metodes, rīku apraksts un pamatojums

1. **Izmantoti rīki**

**Firebase Test Lab**

*Apraksts:* Google mākoņpakalpojums, kas ļauj testēt Android lietotnes uz dažādām reālām ierīcēm un emulatoriem, lai nodrošinātu to saderību ar citām ierīcēm.

*Pamatojums:* Firebase Test Lab tika izvēlēts, jo tas ļauj automatizēti pārbaudīt lietotni uz dažādām ierīcēm un Android versijām, samazinot manuālās testēšanas nepieciešamību.

**Espresso tests**

*Apraksts:* Espresso ir rīks, kas tiek izmantots Android UI testēšanai, lai simulētu lietotāja mijiedarbību ar lietotni, tostarp pogu nospiešanu, ekrāna ritināšanu un teksta ievadi.

*Pamatojums:* Espresso tika izvēlēts tā ātruma un integrācijas dēļ ar Android testēšanas vidi, kas ļauj veikt precīzus un uzticamus UI testus, samazinot manuālās testēšanas laiku un kļūdu iespējamību.

1. **Metodes**

**UI testēšana**

*Apraksts:* Izmanto lietotāja saskarnes testēšanai, lai modelētu lietotāja mijiedarbību un nodrošinātu, ka visas UI komponentes darbojas kā paredzēts.

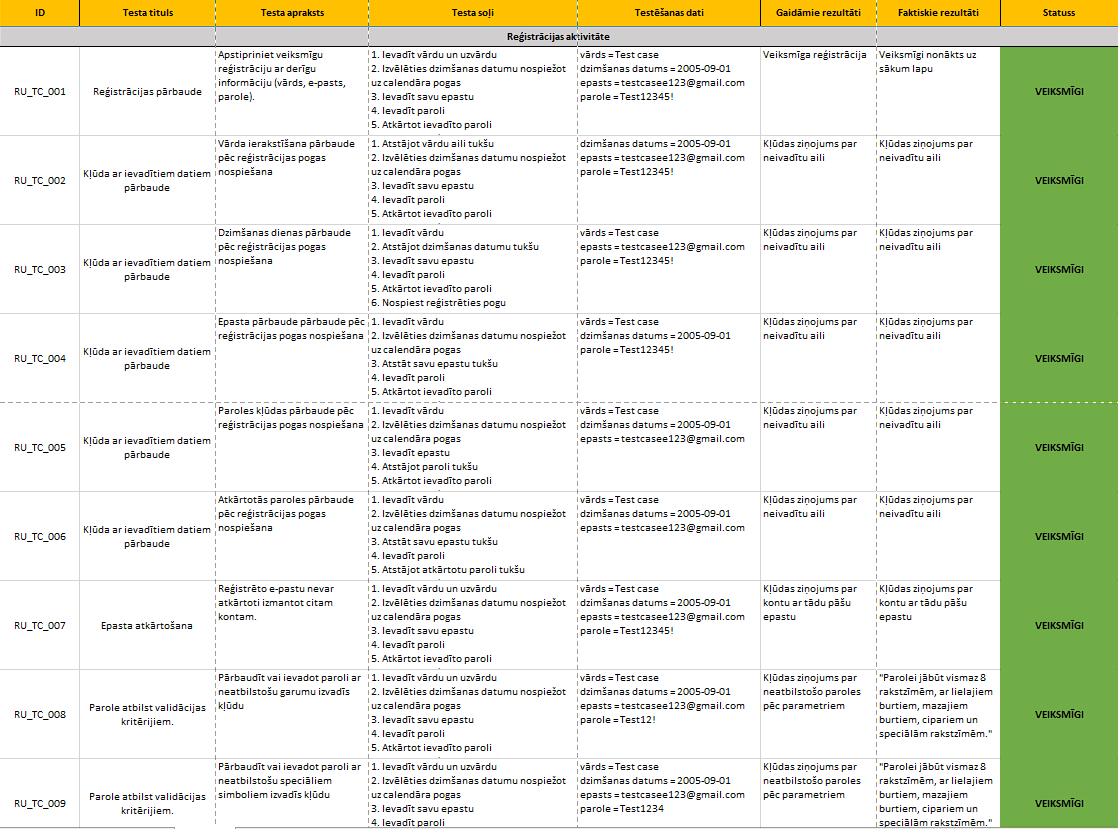
**Melnās kastes (Black box) testēšana**

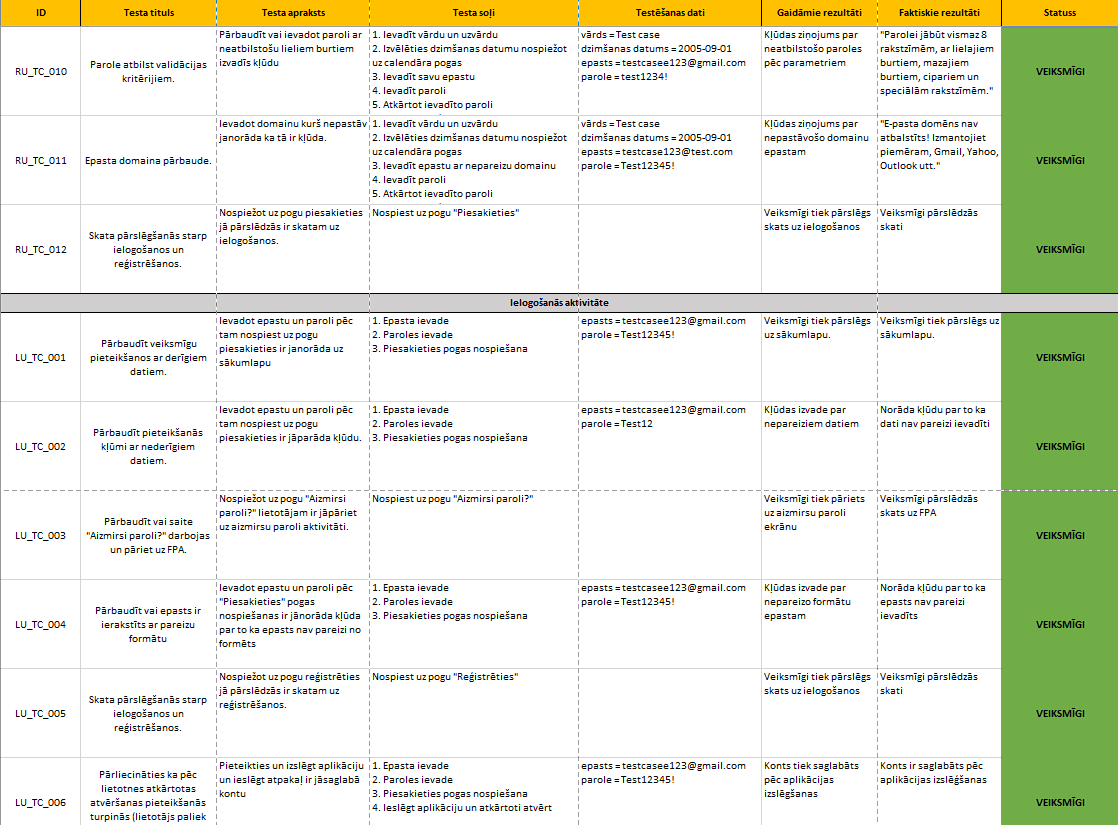
*Apraksts:* Testēšanas metode, kurā tiek pārbaudīta lietotnes funkcionālā darbība, nezinot tās iekšējo kodu vai struktūru. Testētāji koncentrējas uz ievadiem un gaidītajiem izvadiem, lai nodrošinātu, ka lietotne darbojas atbilstoši prasībām.

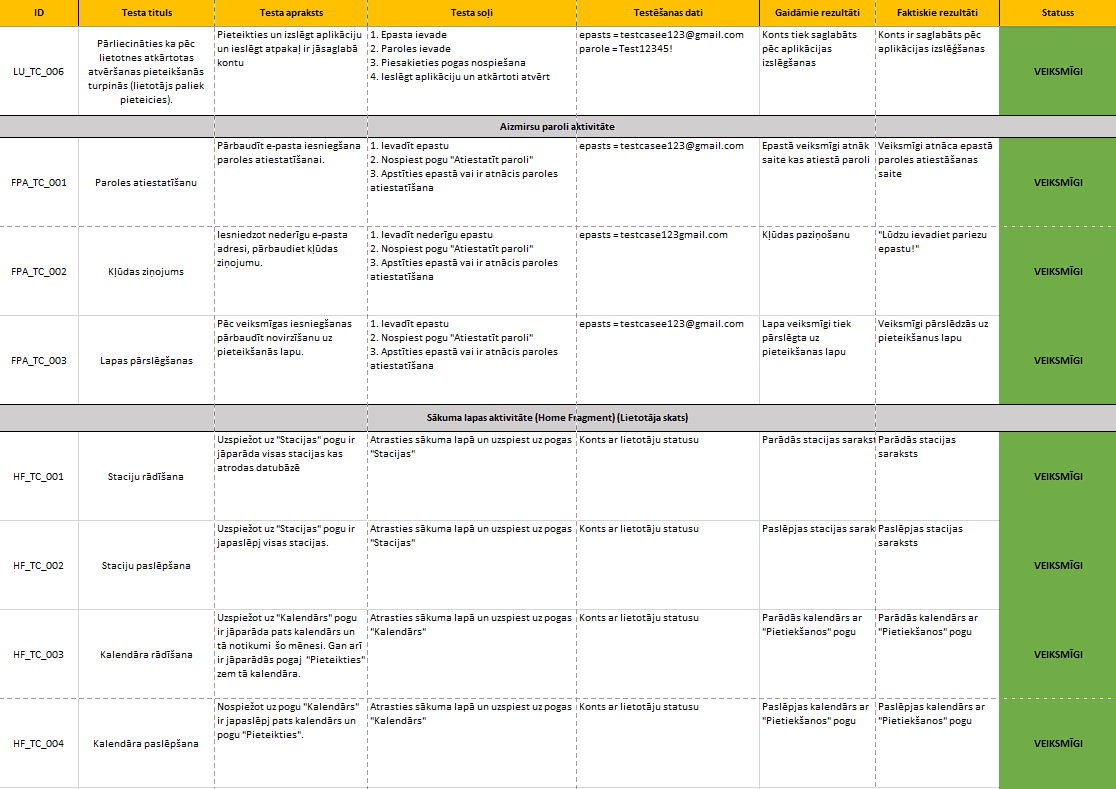
## Testpiemēru kopa

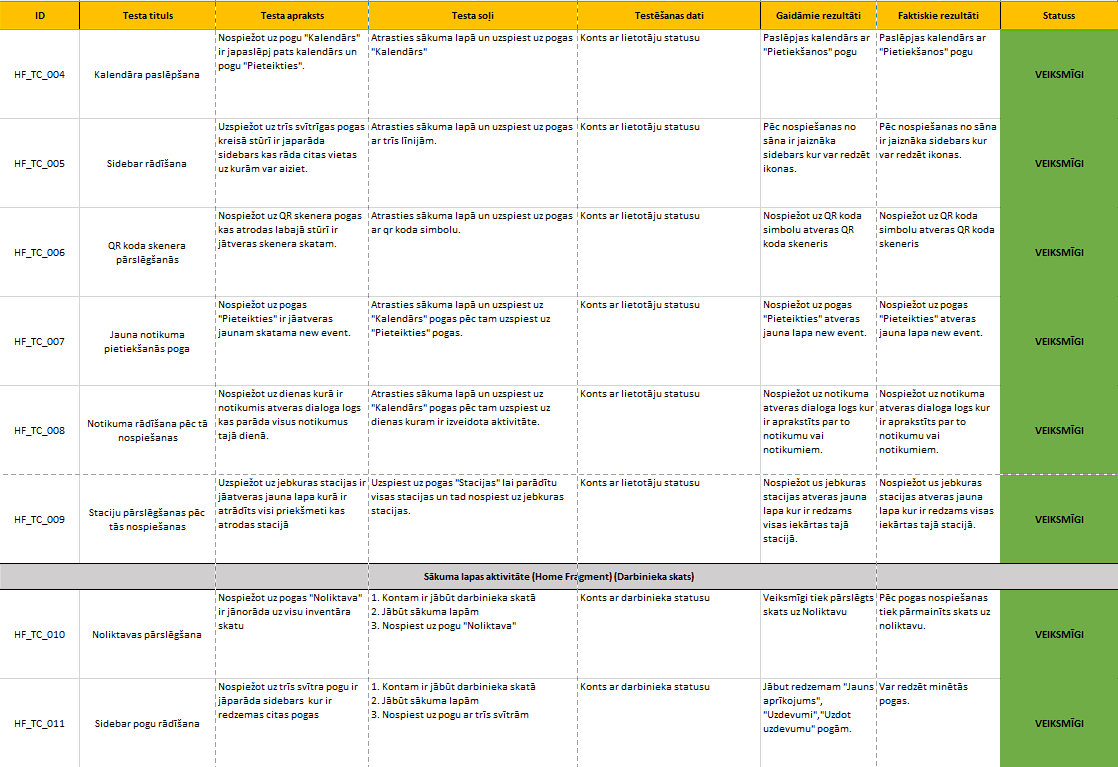
|  |  |
| --- | --- |
| **ID** | **Skaidrojums** |
| LU | Login user |
| RU | Register user |
| FPA | Forgot password activity |
| HF | Home fragment |
| LF | Logs fragment |
| RF | Report fragment |
| SF | Settings fragment |
| EF | Equipment fragment |
| SPECF | Spec stock fragment |
| TF | Task fragment |
| ELF | Equipment list fragment |
| NEF | New equipment fragment |
| MA | Main activity |
| NE | NewEvent |
| TC | Test case |
| TA | Task Assigning fragment |

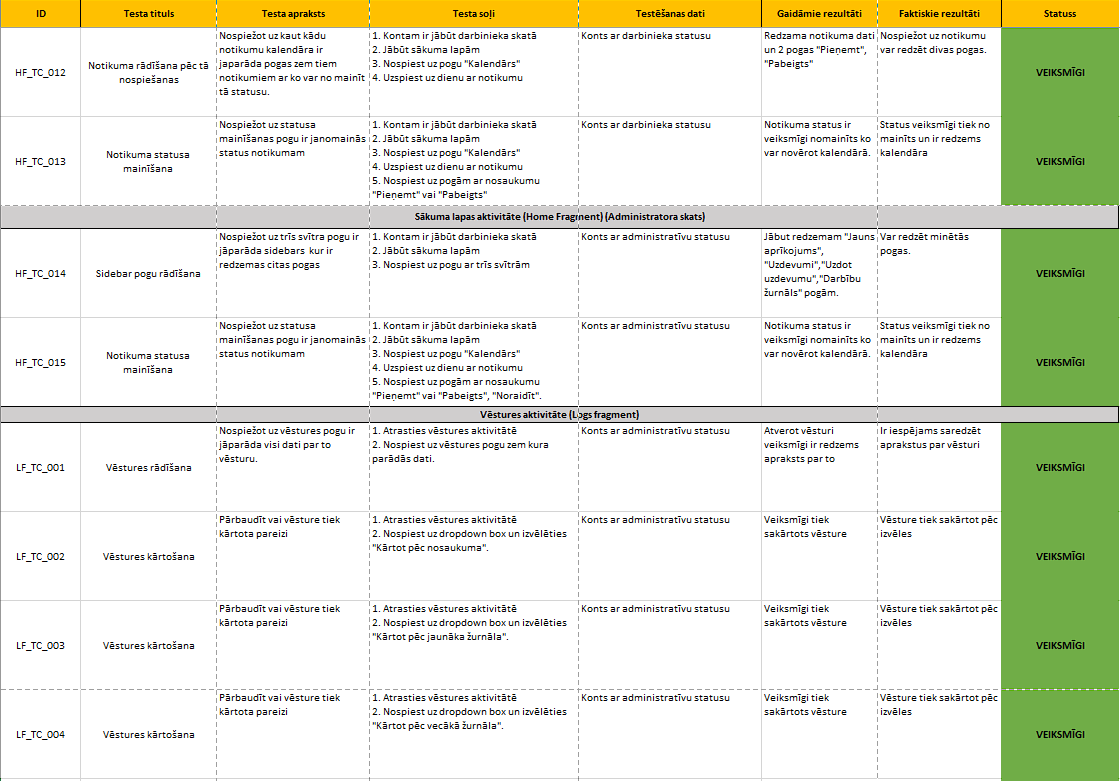
## Testēšanas žurnāls



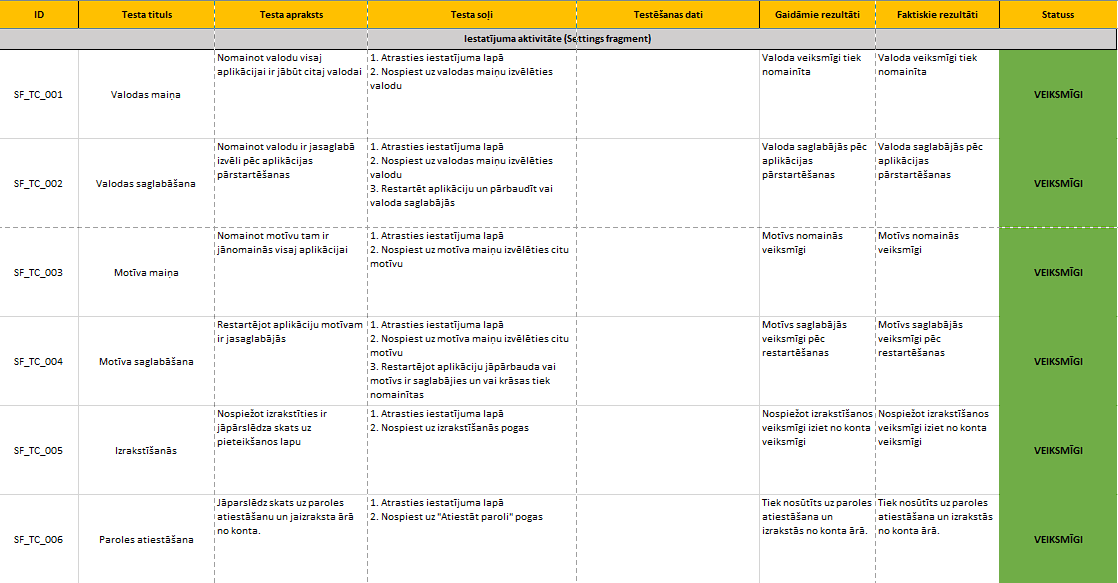


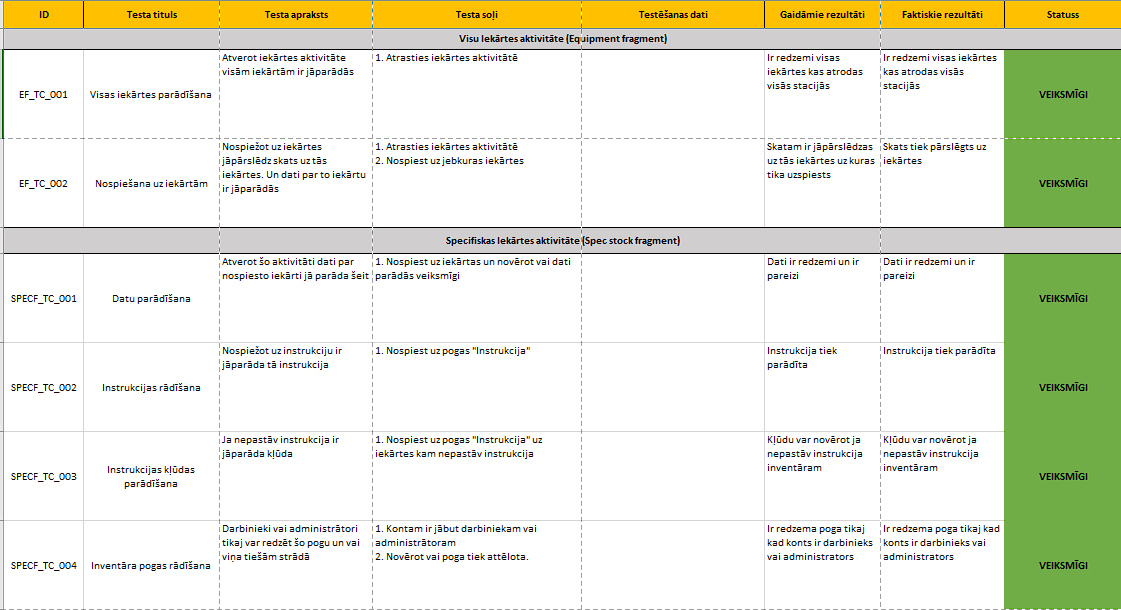


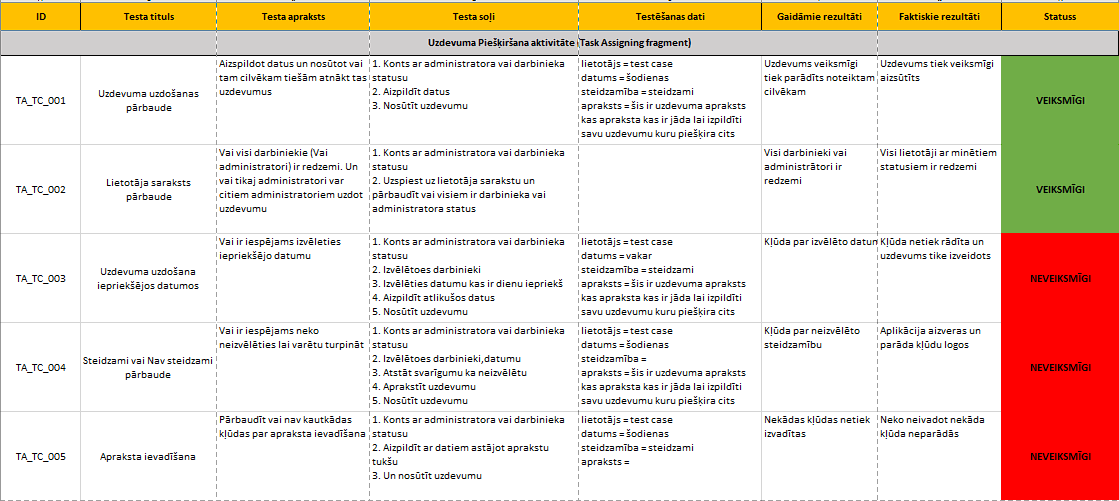


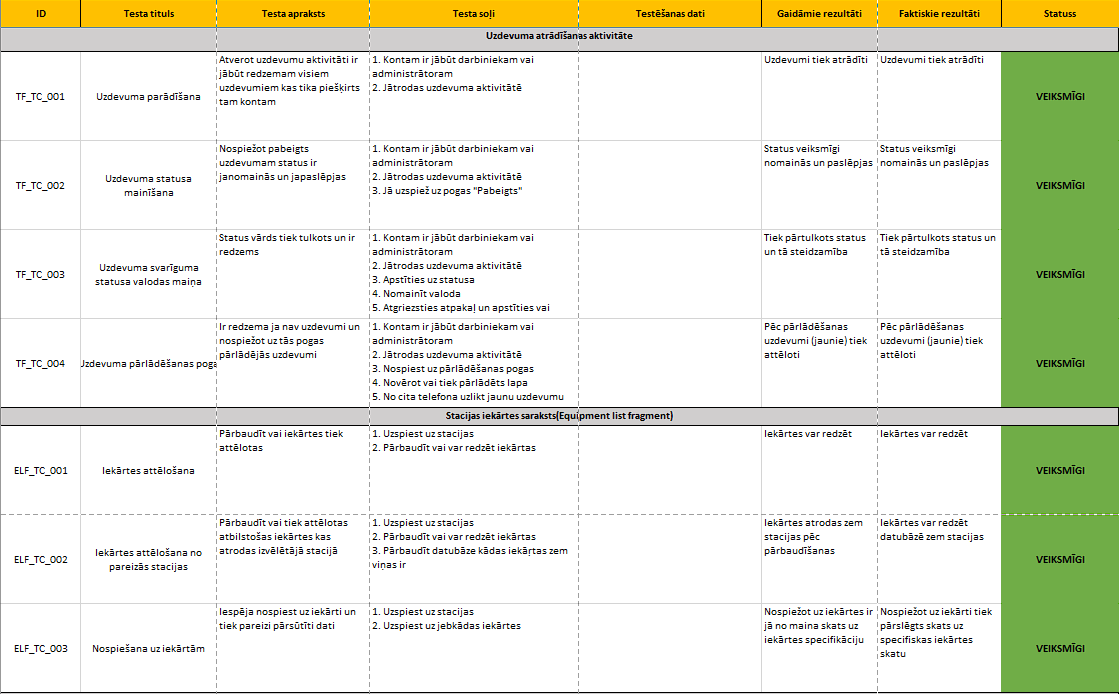




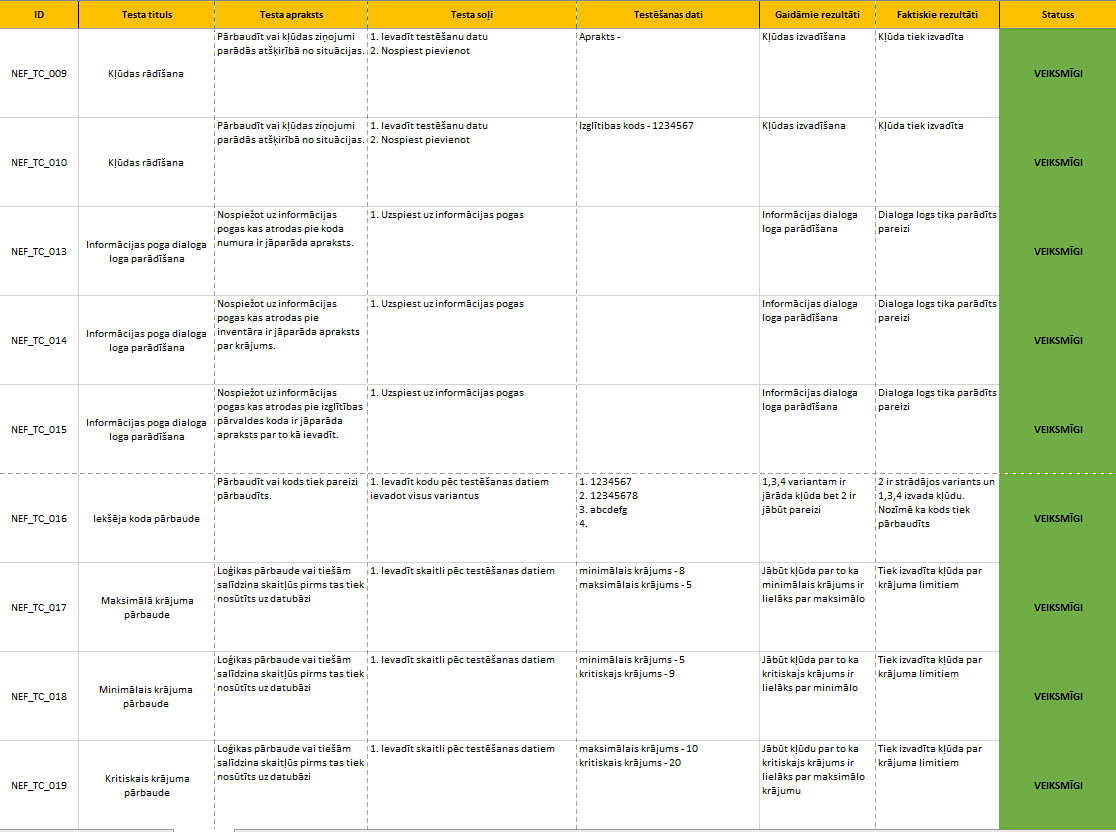










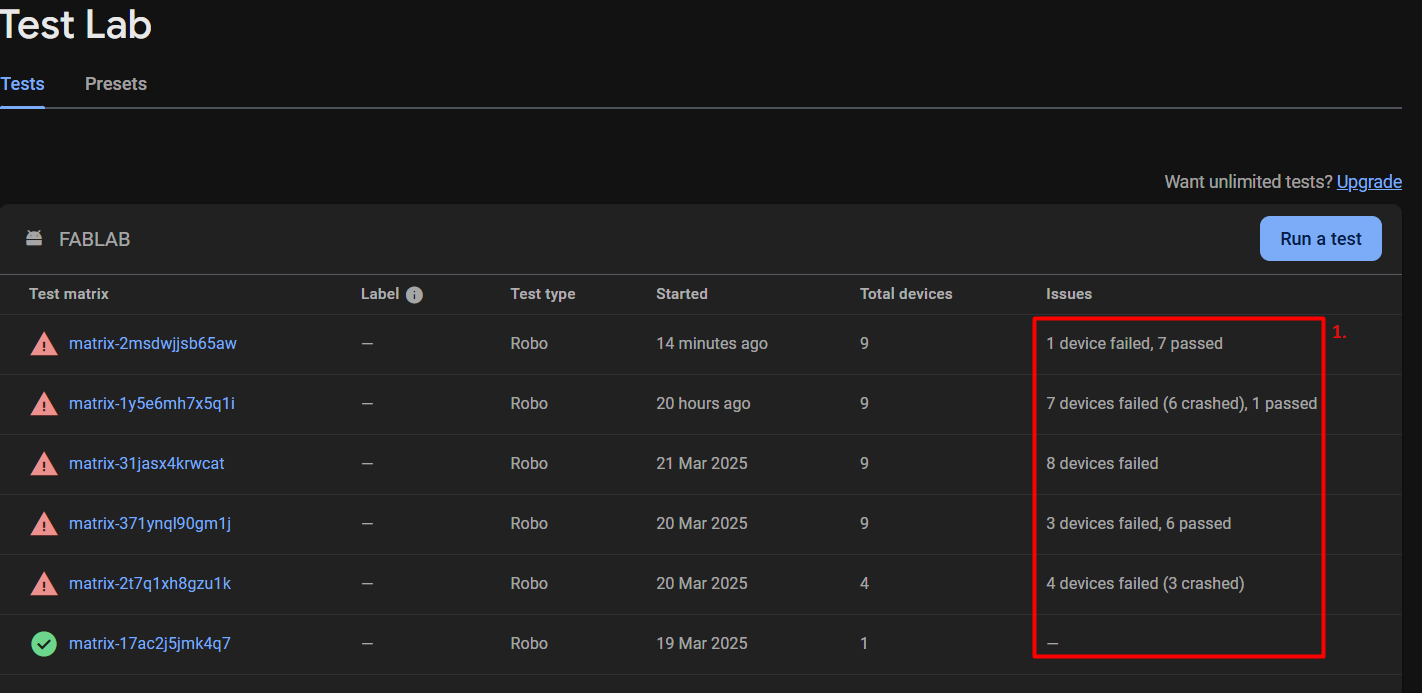






### Firebase test lab žurnāls

Zemāk esošajā attēlā varat redzēt sīkāku informāciju par testiem. Cik ierīču izmantoja, cik no tām izturēja vai neizturēja testu. Firebase test lab ļauj lietotājam izmantot līdz pat 15 ierīcēm 5 fiziskām un 10 virtuālajām ierīcēm. Šis tests ļauj noskaidrot, vai lietojumprogramma darbojas dažādos ekrānos.



**26. attēls Firebase test lab**

# Secinājumi

Šī aplikācija tika izveidota ar mērķi nodrošināt pārvaldību, uzdevumu plānošanu un resursu pārvaldību stacijas vidē. Tas palīdz lietotājiem ātri atrast nepieciešamo informāciju, pārvaldīt krājumus, pievienot vai atjaunināt, kā arī ieplānot notikumus un uzdevumus kalendārā. Tika ņemti vērā dažāda veida lietotāji – gan parastie lietotāji, gan darbinieki, gan administratori. Katram no tiem ir pieejamas funkcijas, kas pielāgotas viņu vajadzībām un pienākumiem. Lietojumprogrammai ir pievienotas arī vairākas dažādas iespējas, piemēram, QR koda skeneris ātrai meklēšanai, automātiska attēlu uzņemšana, brīdinājumi, kad krājumi nokrītas zem paredzētā skaita, vairāku valodu atbalsts un detalizēta notikumu vadība. Tā kā uzsvars tika likts arī uz lietojamību, lietotāja interfeiss ir pielāgots dažādiem ekrāna izmēriem. Visa aplikācija ir izstrādāta, lai vienkāršotu darba procesu stacijās, padarītu informācijas apmaiņu ātrāku un efektīvāku, kā arī novērstu kļūdu un pārpratumu iespējamību ikdienas darbā. Papildus šīm pamatfunkcijām lietojumprogramma piedāvā strukturētu un uzticamu sistēmu būtisku ikdienas darbību pārvaldībai. Integrācija ar Firebase nodrošina reāllaika datu sinhronizāciju, nodrošinot, ka lietotāji vienmēr piekļūst jaunākajai informācijai neatkarīgi no viņu atrašanās vietas. Izmantojot efektīvu krājumu pārvaldību, aprīkojuma izsekošanu un uzdevumu piešķiršanu, lietojumprogramma samazina vajadzību pēc manuālas koordinācijas un samazina aizkavēšanās vai nepareizas saziņas iespējamību. Tādu funkciju kā pasākumu plānošanas un automātisko brīdinājumu par krājumu samazināšanos iekļaušana veicina labus darba ieradumus un palīdz lietotājiem izvairīties no iespējamām problēmām. Tikmēr iespēja piešķirt lomas un ierobežot piekļuvi, pamatojoties uz lietotāja statusu, papildina drošības un atbildības līmeni, nodrošinot, ka katrs lietotājs redz un pārvalda tikai to, kas viņam ir būtisks. Šī lietojumprogramma ne tikai uzlabo individuālo produktivitāti, bet arī uzlabo komandas sadarbību, izmantojot galvenās funkcijas vienā, ērti lietojamā platformā. Tas kalpo kā spēcīgs instruments organizācijas uzlabošanai, resursu pieejamības nodrošināšanai un staciju vides efektīvas darbības atbalstam. Kopumā sistēma atspoguļo pārdomātu funkcionalitātes, elastības un uz lietotāju orientēta dizaina kombināciju. To var paplašināt un pielāgot pēc vajadzības, padarot to par ilgtspējīgu risinājumu nākotnes izaugsmei un mainīgajām prasībām uz stacijām balstītā resursu un aprīkojuma pārvaldībā.

# Lietoto terminu un saīsinājumu skaidrojumi

|  |  |
| --- | --- |
| Termins | Skaidrojums |
| Serveris | datorprogramma vai ierīce, kas sniedz pakalpojumu citai datorprogrammai un tās lietotājam, ko sauc arī par klientu. |
| Mākoņdatošanas | skaitļošanas pakalpojumu, tostarp serveru, krātuvju, datubāzu, tīklu, programmatūras, analītikas un izlūkošanas, piegāde internetā. |
| UID | ir ciparu vai burtu un ciparu virkne, kas ir saistīta ar vienu entītiju noteiktā sistēmā. |
| Mākoņpakalpojums | lietojumprogrammu un infrastruktūras resursi, kas pastāv internetā. |
| Emulators | programmatūra, ar kuras palīdzību mikrodatoru vai personālo datoru var izmantot kā kādas datoru sistēmas termināli. |
| QR kods | mašīnlasāms kods, kas sastāv no melnbaltu kvadrātu masīva, ko parasti izmanto internetu adrešu vai citas informācijas glabāšanai, ko telefona kamera var nolasīt. |
| Firebase | ir mākoņa balstīta platforma, kas paredzēta tīmekļa un mobilo lietojumprogrammu izstrādei, palīdzot vienkāršot izstrādes procesu. |
| Firebase test lab | Ir mākoņa balstīta lietotņu testēšanas infrastruktūra, kas ļauj testēt lietotni dažādās ierīcēs un konfigurācijās. |
| UI | cilvēka un datora mijiedarbības un saziņas punkts ierīcē. Tas var ietvert displejus, tastatūras, peli un darbvirsmas izskatu. |
| ID | identifikācijas saīsinājums. |
| NoSQL | nerelāciju datu bāzes, kurās dati tiek glabāti netabulārā formātā, nevis uz kārtulām balstītās relāciju tabulās, piemēram, relāciju datubāzēs. |
| IDE | lietojumprogramma, kas palīdz programmētājiem efektīvi izstrādāt programmatūras kodu. |
| Android SDK | Android SDK (Programmatūras izstrādes komplekts) ir izstrādes rīku komplekts, ko izmanto, lai izstrādātu lietojumprogrammas Android platformai. |

# Literatūras un informācijas avotu saraksts

<https://developer.android.com/develop/ui/compose/documentation> – aplikācijas UI elementi

<https://www.tutorialspoint.com/how-to-use-multiple-threads-in-android> – multithreading paraugs

<https://firebase.google.com/docs/android/setup> – firebase pievienošana android aplikācijai

<https://firebase.google.com/docs/database/android/start#java> – firebase datubāzes izveidošana

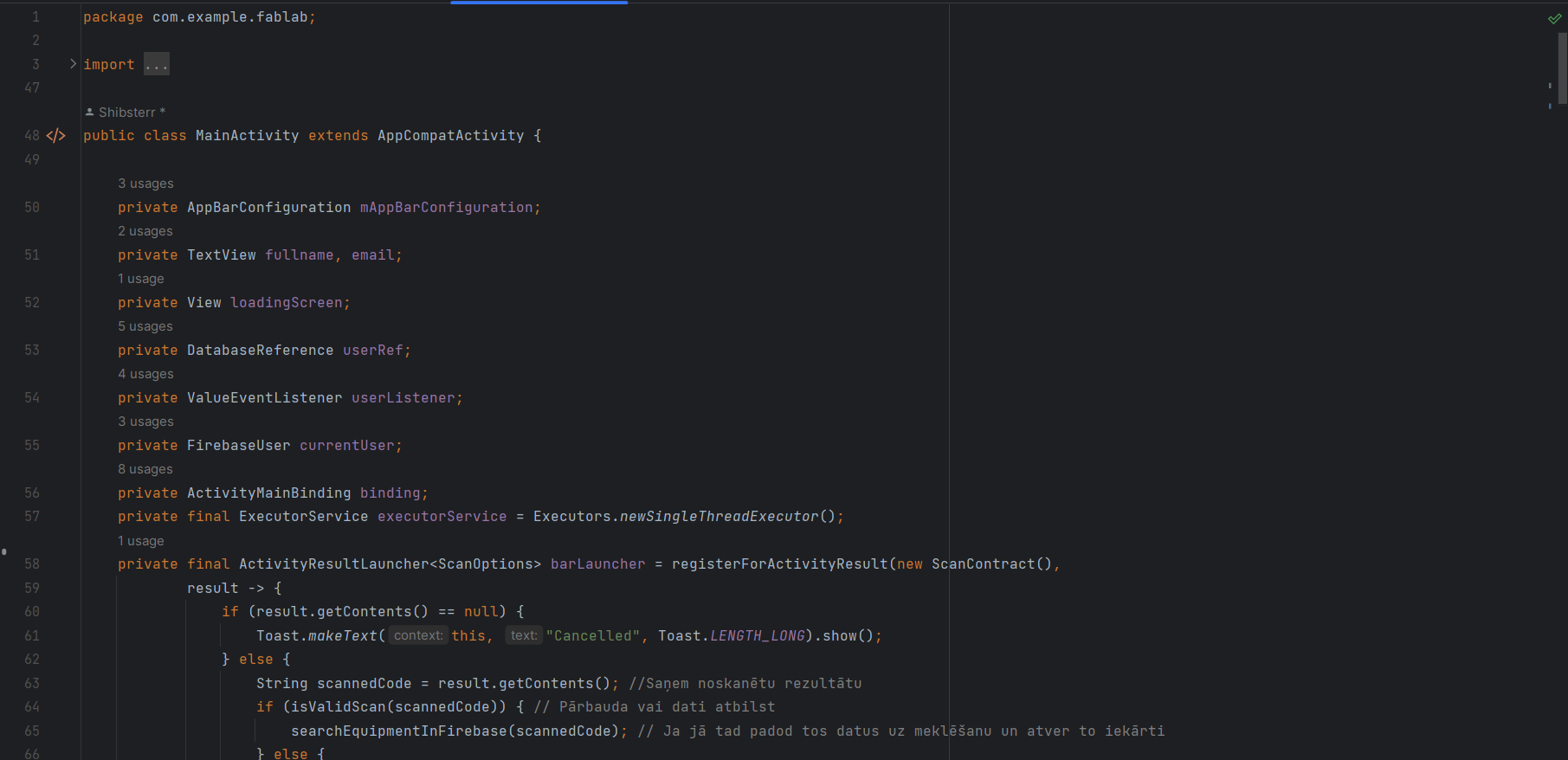
<https://firebase.google.com/docs/storage/android/start> – firebase krātuves izmantošana

<https://firebase.google.com/docs/auth/android/start> – firebase autentifikācija paraugs

<https://www.youtube.com/watch?v=Q0gRqbtFLcw> – splashscreen piemērs ka to izveidot

<https://www.youtube.com/watch?v=roruU4hVwXA> – epastu sūtīšana caur aplikāciju

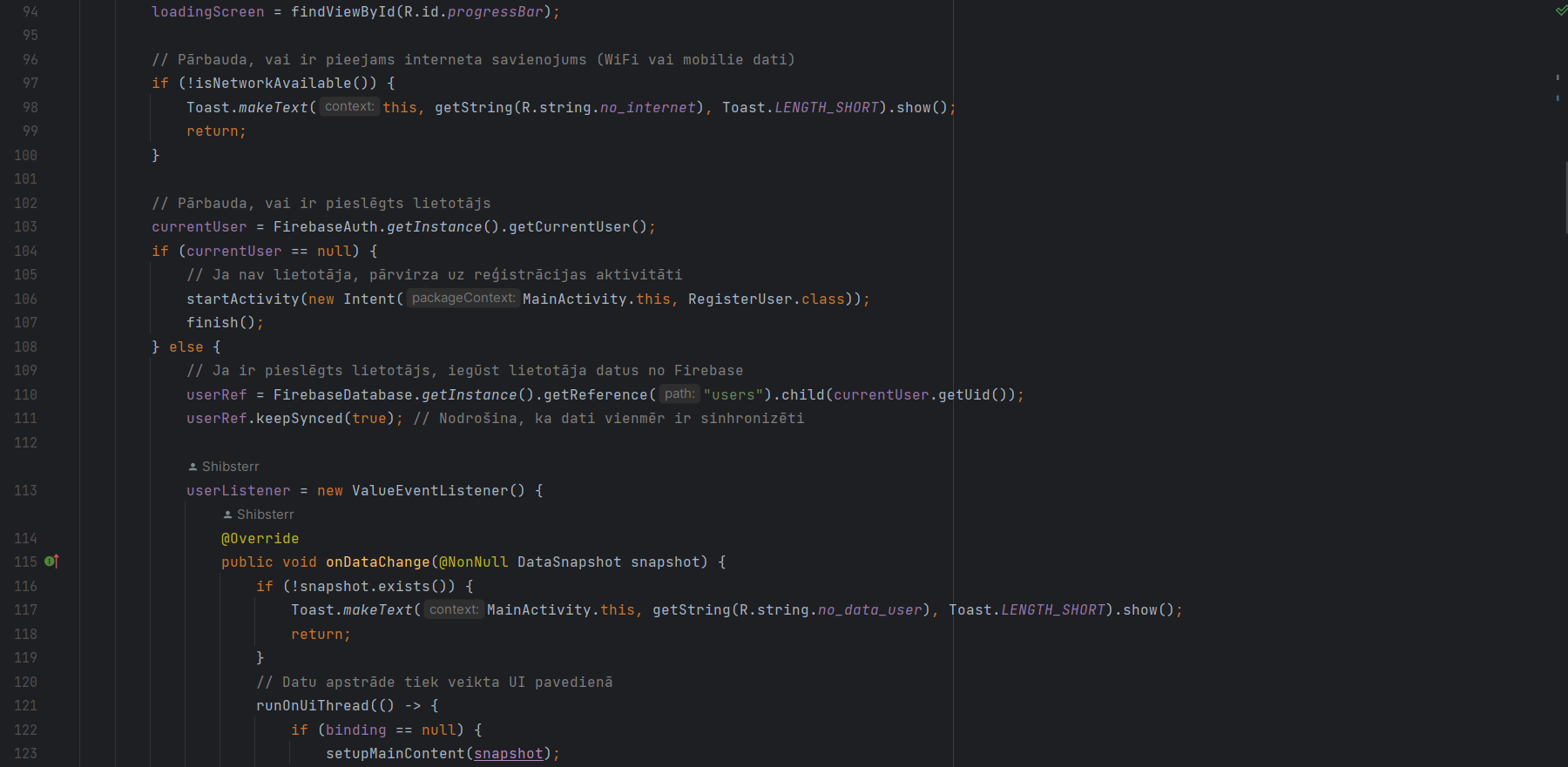
# Pielikums



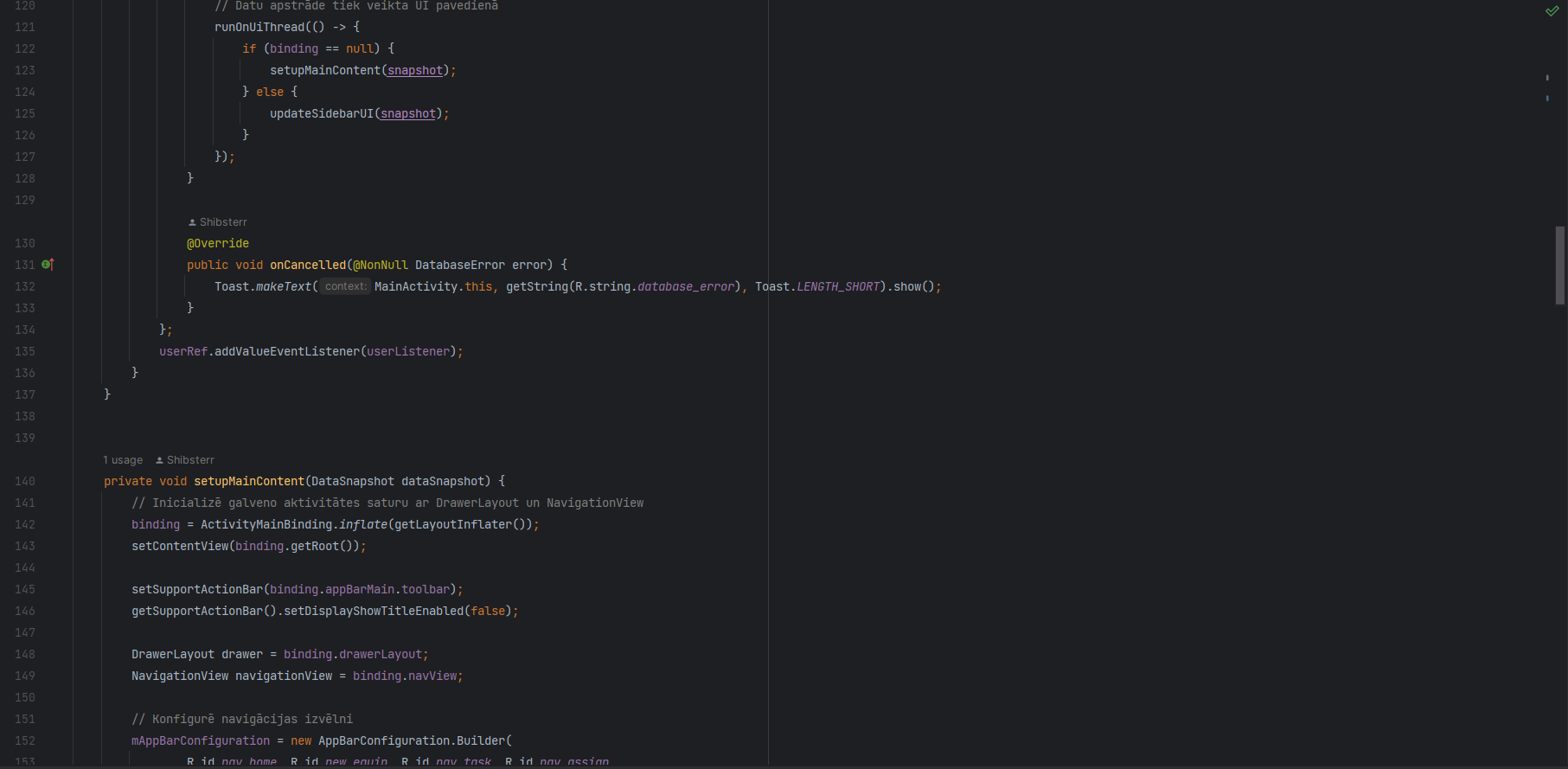
**27. attēls galvenā aktivitātes koda fragments**



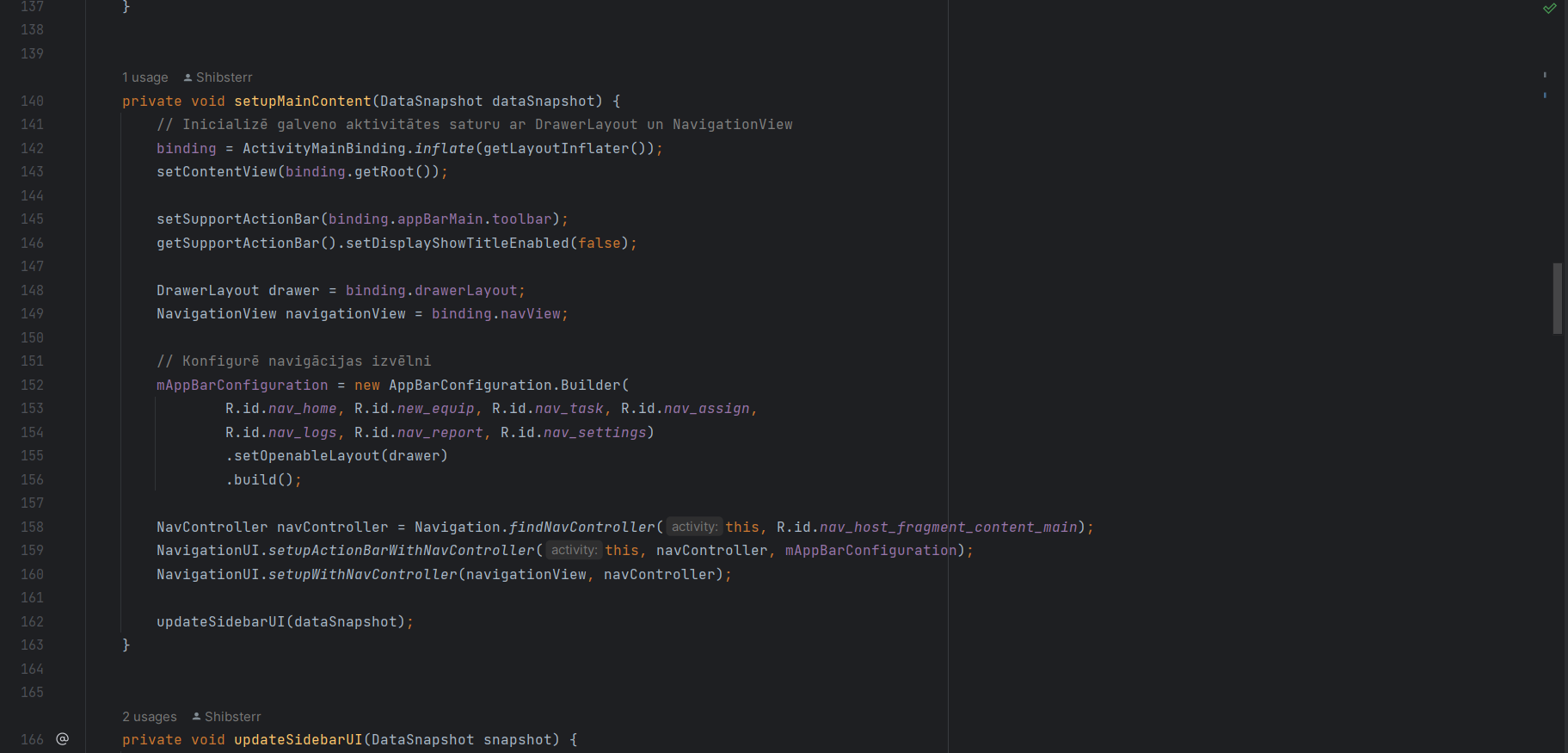
**28. attēls galvenā aktivitātes koda fragments**



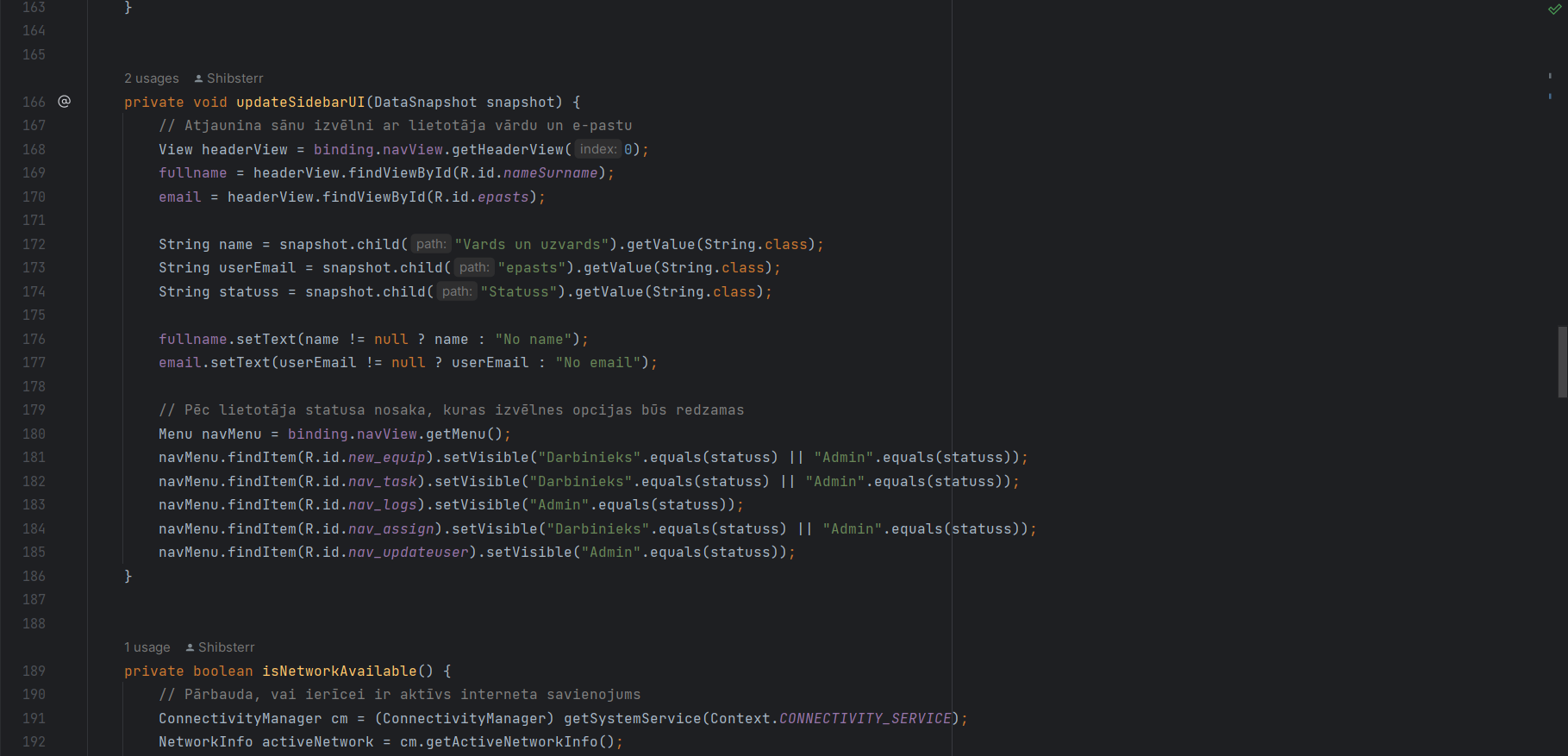
**29. attēls galvenā aktivitātes koda fragments**



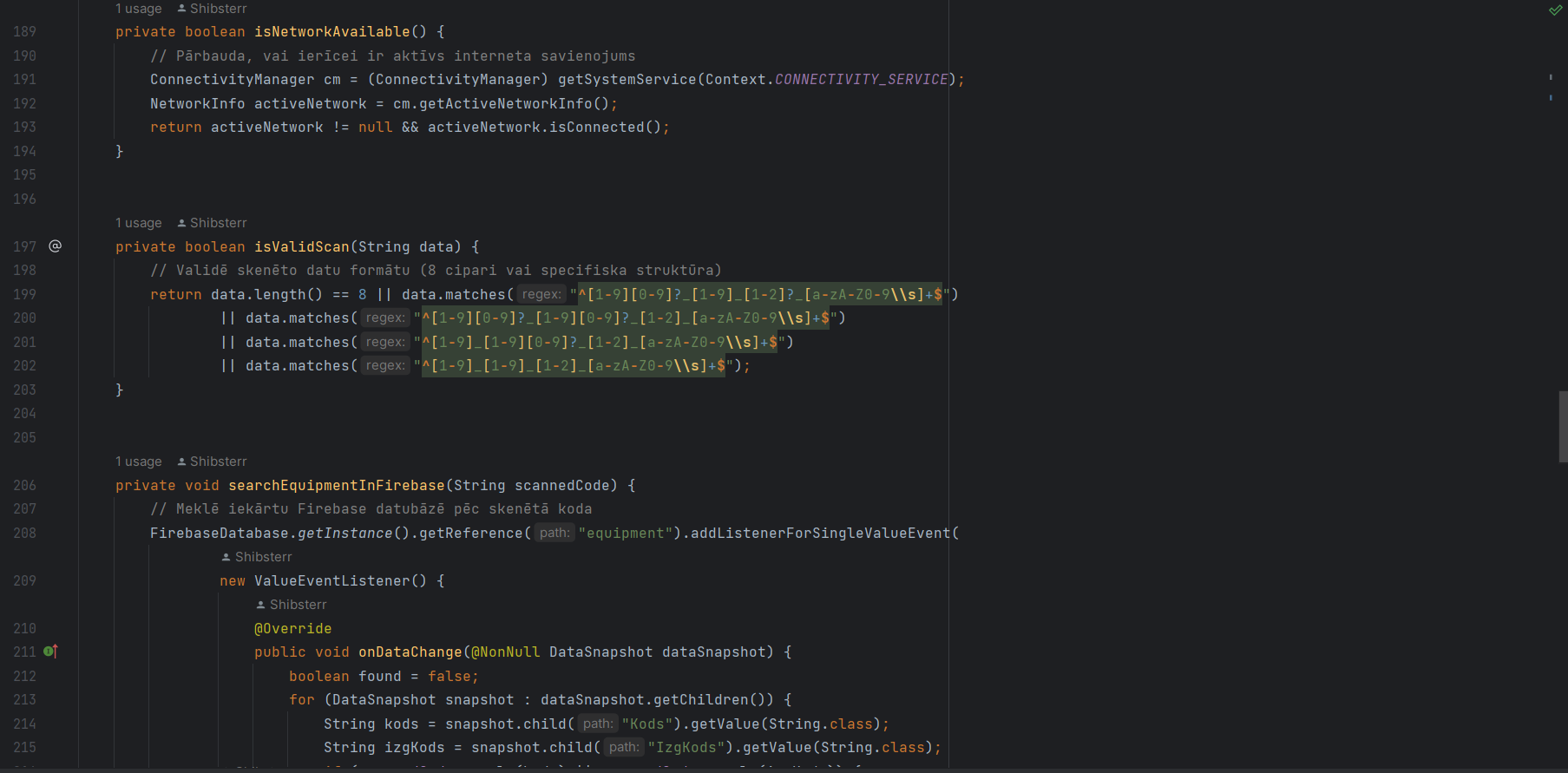
**30. attēls galvenā aktivitātes koda fragments**



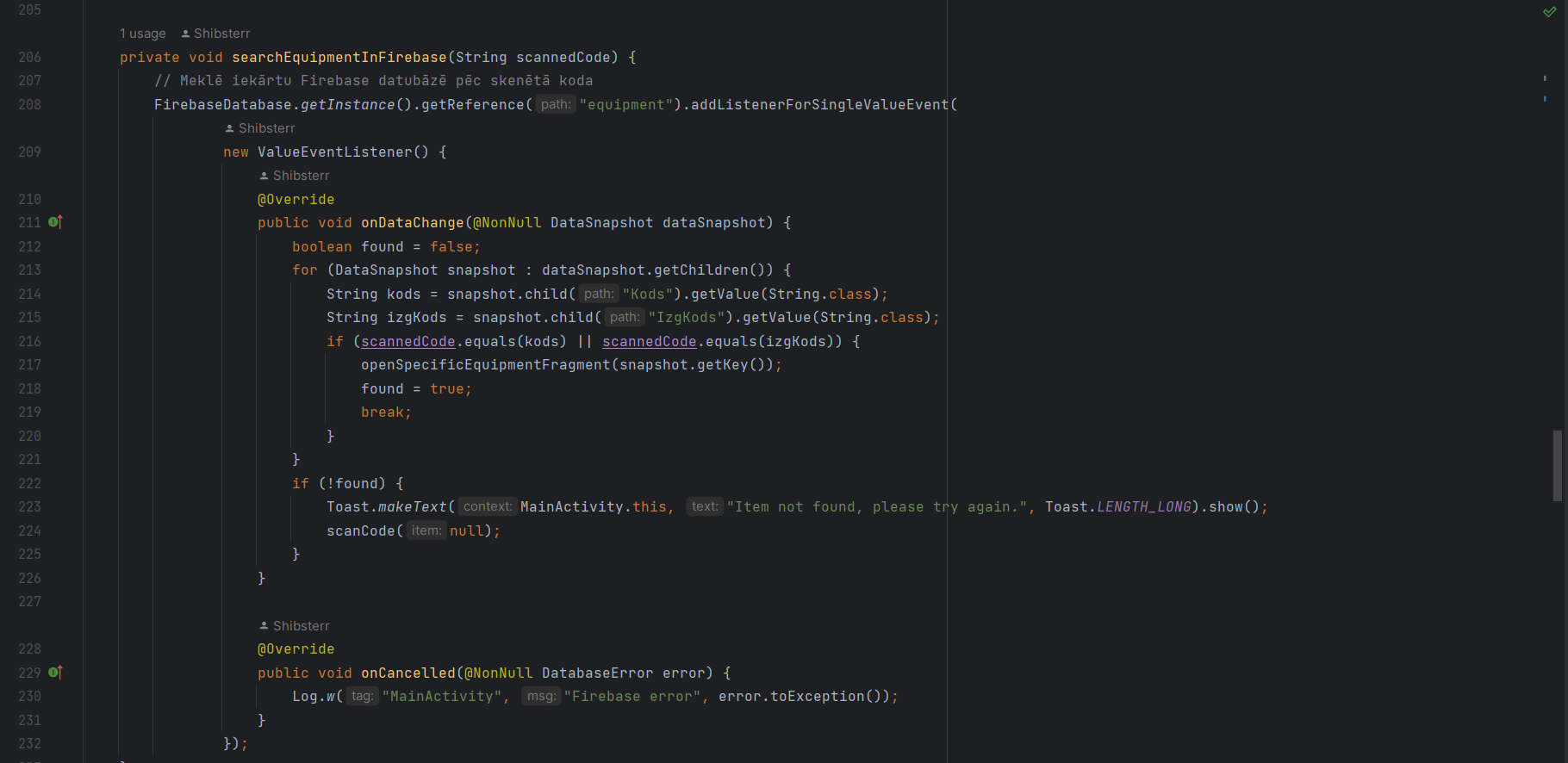
**31. attēls galvenā aktivitātes koda fragments**



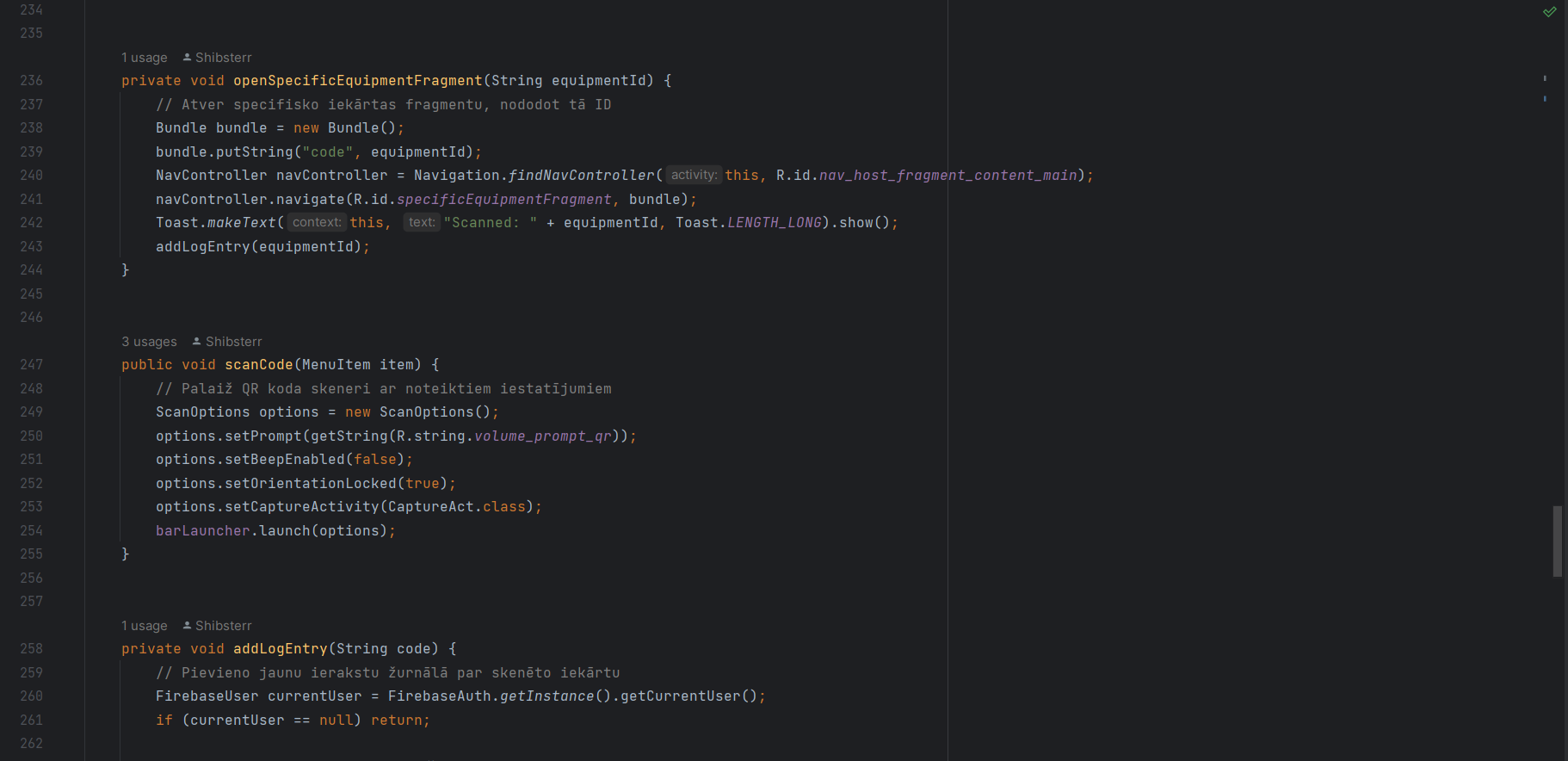
**32. attēls galvenā aktivitātes koda fragments**



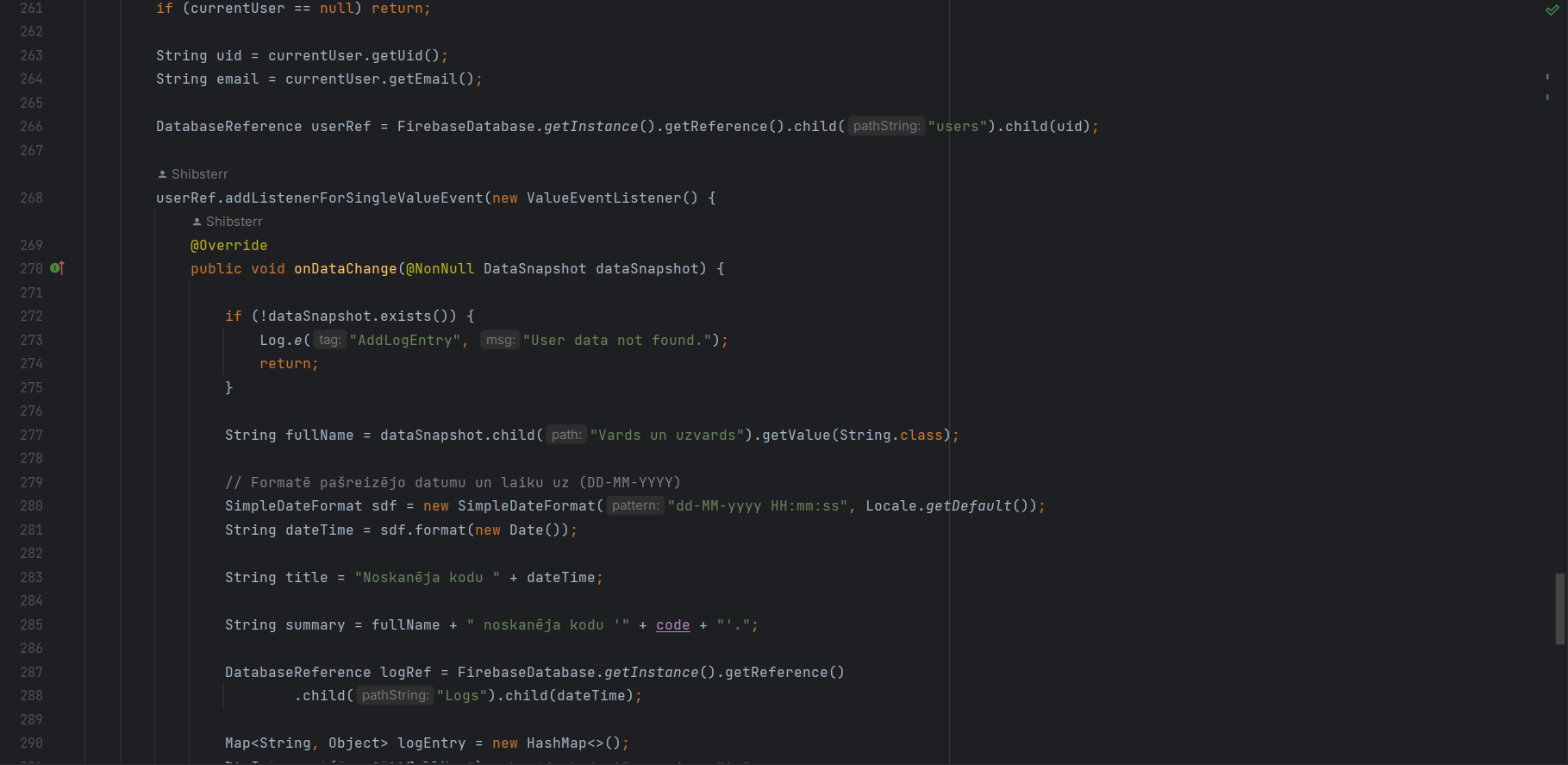
**33. attēls galvenā aktivitātes koda fragments**



**34. attēls galvenā aktivitātes koda fragments**



**35. attēls galvenā aktivitātes koda fragments**



**36. attēls galvenā aktivitātes koda fragments**



**37. attēls galvenā aktivitātes koda fragments**



**38. attēls galvenā aktivitātes koda fragments**