

# **RĪGAS VALSTS TEHNIKUMS**

## **DATORIKAS NODAĻA**

Izglītības programma: Programmēšana

### **KVALIFIKĀCIJAS DARBS**

**“Tiešsaistes un klātienes daudzpētāju diskusiju  
rīks izklaidēm un situāciju modelēšanai.”**

Paskaidrojošais raksts    lpp.

Audzēknis:

Edgars Polis

Prakses vadītājs:

Ilona Demčenko

Nodaļas vadītājs:

Normunds Barbāns

**Rīga 2023**

## ANOTĀCIJA

Kvalifikācijas darba tēma ir "Tiešsaistes un klātienēs daudzpēlētāju diskusiju rīks izklaidēm un situāciju modelēšanai". Izstrādājamā programmsistēma veido interaktīvas kartīšu spēles, kuras var izmantot gan izklaides, gan izglītības nolūkos, lai stiprinātu darba kolektīvu sapratni un ģimenes kopā pavadīto laiku. Sistēma ļauj lietotājiem izveidot, pārvaldīt un spēlēt dažādas kartīšu spēles ar jautājumiem un atbildēm. Izstrādes procesā izmantotie rīki ir TypeScript, Svelte, SQLite, SvelteKit, Vite, Prettier, PocketBase, Tailwind CSS, PostCSS, SkeletonUI, Nanoid, Node.js, Fish, Concurrently un Visual Studio Code.

Kvalifikācijas darbs ietver ievadu, uzdevumu nostādni, prasību specifikāciju, uzdevuma risināšanas līdzekļu izvēles pamatojumu, programmatūras produkta modelēšanas un projektēšanas aprakstu, datu struktūru aprakstu, lietotāja ceļvedi, testēšanu un nobeigumu. Ievadā ir aprakstīta aktuālā problēma un tās risinājums. Uzdevumu nostādnē ir norādīti uzdevumi, kurus sistēmai būs nepieciešams veikt. Prasību specifikācija sastāv no ieejas un izejas informācijas, kā arī no sistēmas funkcionālajām un nefunkcionālajām prasībām. Uzdevuma risināšanas līdzekļu izvēles pamatojumā ir norādīti, kādi līdzekļi tiks izmantoti. Programmatūras produkta modelēšanas un projektēšanas apraksts sastāv no sistēmas struktūras modeļa, kas ietver sistēmas arhitektūru un ER modeli, un funkcionālās sistēmas struktūras modeļa, kas satur datu plūsmas modeli. Datu struktūru aprakstā tiek parādīta datu bāzes relāciju shēma, kā arī tabulu struktūra ar aprakstu, kas ietver datu tipu un datu garumu norādīšanu. Lietotāja ceļvedī ir norādītas nepieciešamās sistēmas prasības aparatūrai un programmatūrai, sistēmas instalācija un palaišana, kā arī programmas apraksts, kas paskaidro, kā pareizi lietot sistēmu. Testa piemērā ir dots detalizēts rezervācijas pievienošanas apraksts ar vizuāliem attēliem.

Kvalifikācijas darbs sastāv no \_\_ lappusēm, kurā ietilpst \_\_ attēli, \_\_ tabulas un \_\_ pielikumi. Pielikumi satur pārskatu piemērus, ekrānattēlu piemērus un tabulu relāciju shēmu.

## ANNOTATION

The topic of the thesis is "Online and face-to-face multiplayer discussion tool for entertainment and situation modeling". The system creates interactive card games that can be used for both entertainment and educational purposes, to strengthen the understanding of work teams and for families to spend time together. The system allows users to create, manage and play a variety of question and answer card games. The tools used in the development process are TypeScript, Svelte, SQLite, SvelteKit, Vite, Prettier, PocketBase, Tailwind CSS, PostCSS, SkeletonUI, Nanoid, Node.js, Fish, Concurrently and Visual Studio Code.

The qualification work includes an introduction, statement of tasks, specification of requirements, rationale for choosing tools for solving the task, description of software product modelling and design, description of data structures, user guide, testing, and conclusion. The introduction describes the current problem and it's solution. The task statement specifies the tasks that the system will need to perform. The requirements specification consists of input and output information, as well as the functional and non-functional requirements of the system. The justification for choosing means of solving the task indicates which means will be used. The modelling and design description of a software product consists of a system structure model, which includes the system architecture and an ER model, and a functional system structure model, which contains a data flow model. The description of data structures shows the relational schema of the database, as well as the table structure with a description that includes specifying data types and data lengths. The user guide contains the necessary system requirements for hardware and software, system installation and startup, and a program description that explains how to use the system properly. The test example provides a detailed description of adding a reservation with visual images.

The thesis consists of \_\_ pages, which includes \_\_ figures, \_\_ tables and \_\_ appendices. Appendices contain example reports, example screenshots, and table relational schema.

# SATURS

<b>IEVADS.....</b>	<b>5</b>
<b>1. UZDEVUMA NOSTĀDNE.....</b>	<b>6</b>
<b>2. PRASĪBU SPECIFIKĀCIJA.....</b>	<b>7</b>
2.1. Ieejas un izejas informācijas apraksts.....	7
2.1.1. Ieejas informācijas apraksts.....	7
2.1.2. Izejas informācijas apraksts.....	8
2.2. Funkcionālās prasības.....	8
2.3. Nefunkcionālās prasības.....	9
<b>3. UZDEVUMA RISINĀŠANAS LĪDZEKĻU IZVĒLES PAMATOJUMS.....</b>	<b>12</b>
3.1. Programmēšanas valodas.....	12
3.2. Palīgprogrammas.....	12
3.2. Čaulas.....	13
3.3. Izstrādes vide.....	13
<b>4. PROGRAMMATŪRAS PRODUKTA MODELĒŠANA UN PROJEKTĒŠANA.....</b>	<b>14</b>
4.1. Sistēmas struktūras modelis.....	14
4.1.1. Sistēmas arhitektūra.....	14
4.1.2. Sistēmas ER modelis.....	15
4.2. Funkcionālais sistēmas modelis.....	16
4.2.1. Datu plūsmu modelis.....	16
<b>5. DATU STRUKTŪRU APRAKSTS.....</b>	<b>19</b>
<b>6. LIETOTĀJA CEĻVEDIS.....</b>	<b>22</b>
6.1. Sistēmas prasības.....	22
6.2. Sistēmas instalācija un palaišana.....	23
6.2.1. Izstrādes vides sagatavošana.....	23
6.2.2. Koda iegūšana un uzstādīšana.....	23
6.2.3. Servera palaišana.....	23
6.3. Programmas apraksts.....	24
6.4. Testa piemērs.....	25
<b>NOBEIGUMS.....</b>	<b>26</b>
<b>INFORMĀCIJAS AVOTI.....</b>	<b>27</b>
<b>PIELIKUMI.....</b>	<b>28</b>
1. pielikums. Pārskatu piemēri.....	28
2. pielikums. Ekrānattēlu piemēri.....	28
3. pielikums. Tabulu relāciju shēma.....	30

## IEVADS

Kāršu spēles ir iecienītas visur - gan darbā, gan mājās, gan uz ballītēm, gan vienkārši, kad ir klusa vakara stunda. Taču nav vienmēr iespējams tās spēlēt klātienē. Vai dēļ COVID vīrusa, vai liela dalībnieku skaita, vai fiziska attāluma, tiešsaistes grupas spēles ir atradušas savu vietu pasaulē.

Šādu spēļu sistēmu var izmantot ne tikai izklaidēm, bet arī kā mācību līdzekli, kas spēj padarīt procesu aizraujošāku. Tā var palīdzēt darba kolektīviem stiprināt savstarpējo sapratni un mazināt ikdienas spriedzi, un ģimenēm kopā pavadīt laiku, kas nes prieku un tuvina paaudzes.

Mūsdienu dzīve, kurā darba biedri un draugi bieži ir visās pasaules malās, liekot meklēt alternatīvas klasiskajām spēlēm. Bez tiešsaistes spēlēm mēs zaudējam ne tikai izklaides iespējas, bet arī iespēju uzturēt draudzīgas attiecības. Ir jau vairākas esošas interaktīvas spēles, kuras var apvienot izglītību ar izklaides elementiem, bet šī kartīšu spēle piedāvā vēl vairāk - spēlētāji var radīt savas spēles un tās pielāgot jebkurai situācijai vai notikumam.

Šī spēle ir paredzēta vidēs kur pagaidām izmanto citas tiešsaistes spēles, kā piemēram Kahoot. Tas iekļauj virtuālus darba un ģimenes pasākumus, klātienes pasākumus ar projektoru vai lielu ekrānu, ballītes, utml. Lietotāji var izvēlēties jau vairākus gatavus kartīšu komplektus, vai arī izveidot savu ar intuitīvu kartīšu izveides rīku.

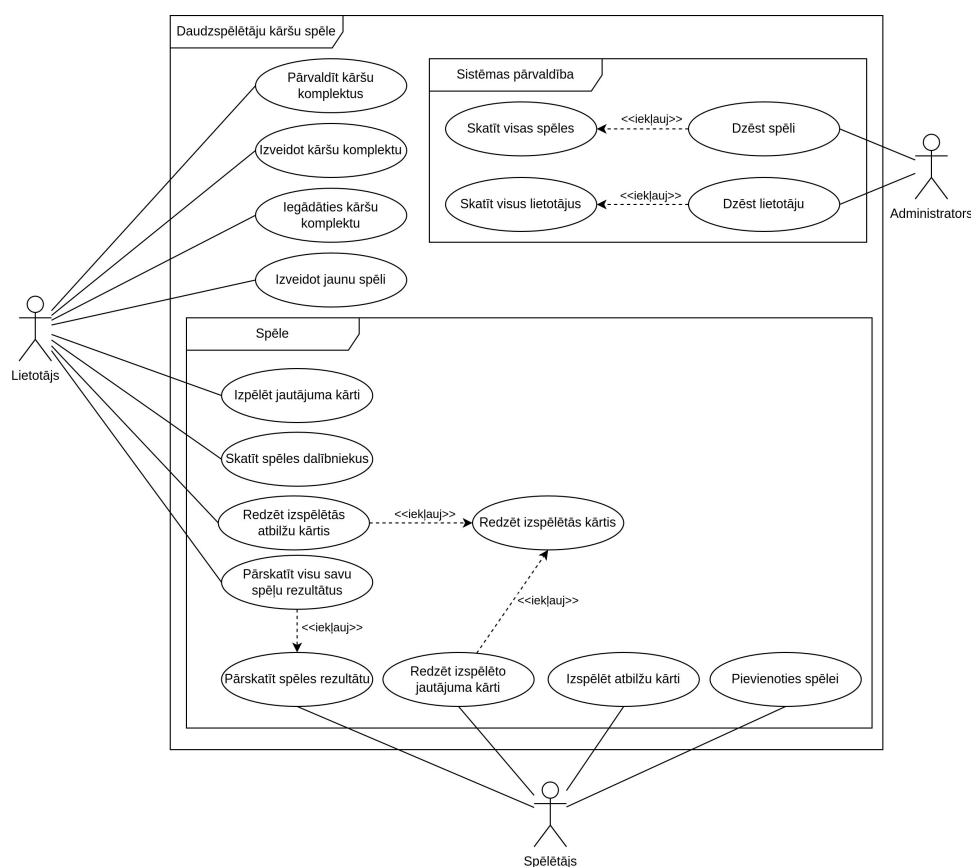
Spēlētāji pievienojas spēlei noskenējot QR kodu savā telefonā, vai ievadot unikālu saiti datorā. Pašas spēles pamats ir vienkāršs: spēles vadītājs izspēle jautājuma kārti, un spēlētājiem tiek piedāvātas vairākas atbilžu kārtis. Spēlētājs izvēlas savu atbildi, un tā tiek iesniegta spēles vadītājam. Pēc tam atkarībā no izvēlētajiem noteikumiem var notikt balsošana par labāko atbildi, vai visi kuri iesniedz saņem punktus, vai arī punkti spēlē nav.

# 1. UZDEVUMA NOSTĀDNE

Kvalifikācijas darba uzdevums ir izveidot viena jautājuma, vairāku atbilžu tiešsaistes kartīšu spēli ar iekļautiem un lietotāja veidotiem komplektiem. Sistēmā nepieciešams realizēt iespēju lietotājam izveidot spēli, izveidot kartīšu komplektu un kārtis tajos, kā arī uzaicināt citus spēlētājus piedalīties spēlē ar unikālu saiti QR koda formātā.

Ir plānotas vairākas funkcijas (skat. 1. att.):

- lietotājs var izveidot jaunu spēli, izvēlēties tai komplektus, un pievienoties kā spēles vadītājs;
- spēlei spēlētāji var viegli pievienoties ar QR kodu;
- vadītājs iesniedz jautājumu, tas parādās uz ekrāna, spēlētājiem jāatrod un jāiesniedz atbilde telefonā;
- spēlētājam izspēlējot kārti tā uzreiz parādās uz galvenā (spēles vadītāja) ekrāna;
- lietotājs var izveidot kontu ar e-pastu un paroli;
- lietotājs var nopirktu kartīšu komplektus;
- lietotājs var izveidot jaunus kartīšu komplektus.



1.att. Lietojumgadījuma diagramma

## 2. PRASĪBU SPECIFIKĀCIJA

### 2.1. Ieejas un izejas informācijas apraksts

#### 2.1.1. Ieejas informācijas apraksts

Sistēmā tiks nodrošināta šādas ieejas informācijas apstrāde.

1. Informācija par **lietotāju** sastāvēs no šādiem datiem.

- Vārds – burtu teksts ar izmēru līdz 30 rakstzīmēm. (Piemēram “Jānis”)
- E-pasts – burtu teksts ar izmēru līdz 30 rakstzīmēm. (Piemēram “[janis@gmail.com](mailto:janis@gmail.com)”)
- Parole - burtu teksts ar izmēru no 8 līdz 30 rakstzīmēm. (Piemēram “Parole12345!”)

2. Informācija par **spēlētāju** sastāvēs no šādiem datiem.

- Vārds - burtu teksts ar izmēru līdz 30 rakstzīmēm. (Piemēram “Alise”)

3. Informācija par **kārti** sastāvēs no šādiem datiem.

- Tips – Izvēle starp “Jautājuma” un “Atbildes”
- Nosaukums - Kārts nosaukums pēc kuras tās var tikt grupētas iekš komplekta (Piemēram “Smieklīgs”)
- Teksts - Kārts teksts (Piemēram “Es jau sen vēlējos uzzināt kā \_\_\_\_ darbojas, es to izmantošu visur!”)

4. Informācija par **kartīšu komplektu** sastāv no šādiem datiem.

- Nosaukums - kartīšu komplekta nosaukums (Piemēram “Cards against humanity (base game)”)
- Apraksts - kartīšu komplekta apraksts (Piemēram “Cards against humanity is a party game for horrible people. Each round, one player asks a question from a black card, and everyone else answers with their funniest white card.”)
- Oficiāls - Boolean izvēle vai kartīšu komplekts ir oficiāls (Piemēram “TRUE”)

### *2.1.2. Izejas informācijas apraksts*

Sistēmā tiks nodrošināta šādas izejas informācijas apstrāde.

1. **Drukājama spēles gaitas atskaite** ieskaitot spēlētāju vārdus, atbildes uzdotajiem jautājumiem, spēles ilgumu, utt.
2. **Drukājama visu spēļu atskaite** ieskaitot spēlētāju vārdus, atbildes uzdotajiem jautājumiem, spēles ilgumu, utt. (skat. 13. att.)
3. **Drukājama visu lietotāju atskaite** (administratoram) ieskaitot spēlētāju vārdus, radišanas datumu un e-pastu.
4. **Drukājama visu kartīšu komplektu atskaite** (administratoram) ieskatot vārdu, aprakstu un iekļautās kārtis.

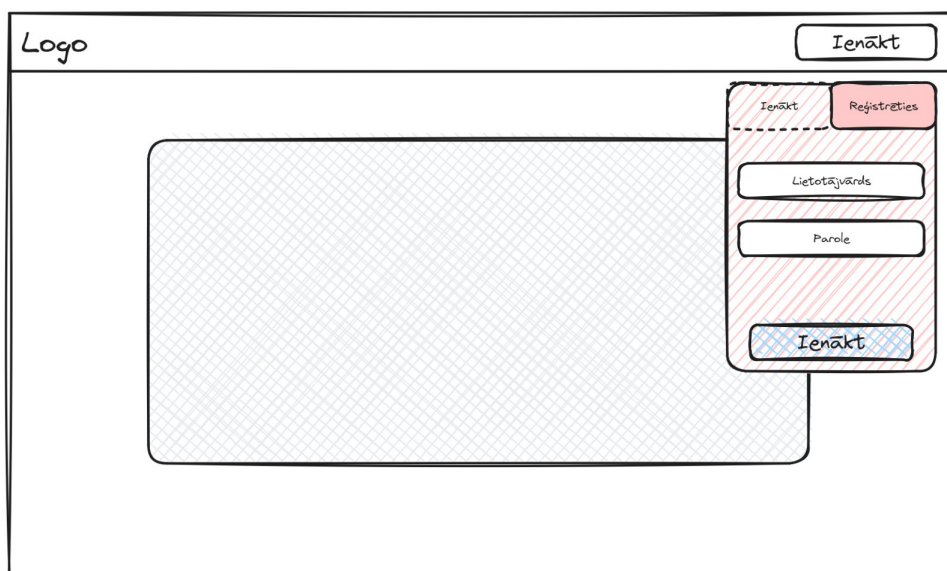


## 2.2 Funkcionālās prasības

1. Lietotāju reģistrācija
  - 1.1. Sistēmai jānodrošina jauna lietotāja reģistrācijas iespēja, pieņemot ievadītos datus par lietotāju (vārds, e-pasts, parole) un pārbaudot to formāta atbilstību norādītajām prasībām.
  - 1.2. Ja reģistrācijas laikā kāds no obligātiem laukiem nav aizpildīts, sistēmai jāizvada attiecīgs kļūdas paziņojums.
  - 1.3. Sistēmai jāpārbauda, vai ievadītais lietotāja vārds jau eksistē sistēmā, un, ja tas ir pieejams, jāizvada kļūdas paziņojums.
  - 1.4. Sistēmai jāveic ievadītās paroles drošības pārbaude un, ja tā neatbilst noteiktajiem kritērijiem, jāizvada brīdinājums par nepietiekamu drošības līmeni.
2. Kāršu komplektu izveide un pārvaldība
  - 2.1. Sistēmai jāpiedāvā funkcionalitāte kartīšu komplektu izveidei, ļaujot lietotājam definēt kartīšu tipus (“Jautājuma” vai “Atbildes”), nosaukumus un tekstus.
  - 2.2. Sistēmai jānodrošina iespēja lietotājam rediģēt un dzēst savus izveidotos kartīšu komplektus.
  - 2.3. Klientiem jābūt iespējai iegādāties gatavus kartīšu komplektus caur iekļautu e-veikalu.
3. Spēles izveide un vadīšana
  - 3.1. Sistēmai jānodrošina iespēja izveidot jaunu spēli, pievienojot tai izveidotos vai iegādātos kartīšu komplektus.
  - 3.2. Spēles izveides laikā sistēmai jāģenerē unikāls QR kods, lai viesi varētu pievienoties spēlei.
  - 3.3. Spēles laikā vadītājam jābūt iespējai iesniegt jautājumus, un spēlētājiem - atbildēt uz tiem, izspēlējot atbilžu kārtis.
  - 3.4. Atbilžu kartīšu izspēlēšana ir jāsinhronizē ar spēles vadītāja ekrānu.
4. Lietotāju un spēļu administrēšana
  - 4.1. Administratoram jābūt sistēmai, kura ļauj pārvaldīt spēļu un lietotāju sarakstu.
  - 4.2. Administratoram jābūt iespējai izdzēst jebkuru spēli vai lietotāju kontu pēc nepieciešamības.

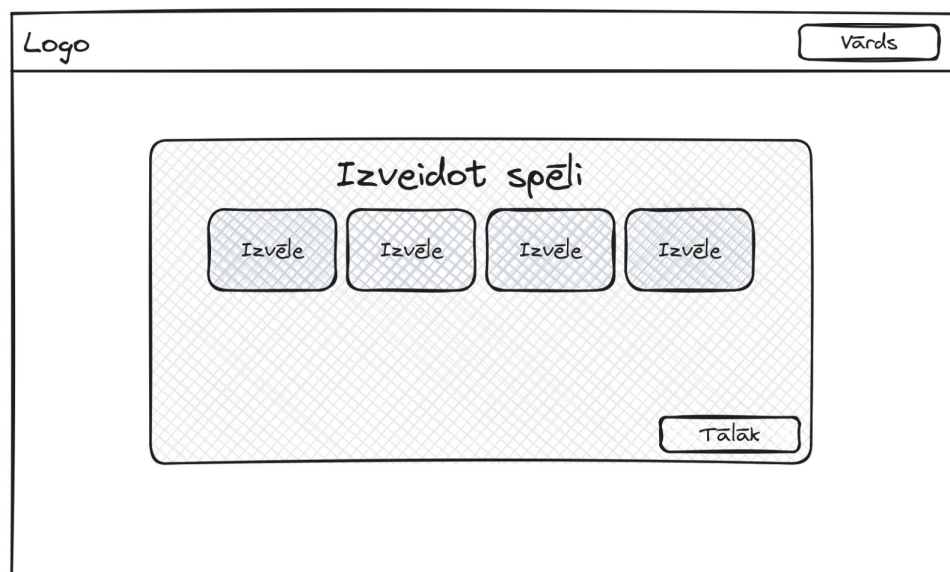
## **2.3. Nefunkcionālās prasības**

1. Veiktspēja
  - 1.1. Sistēmai jāspēj atbalstīt līdz 10 cilvēkiem spēlē un 100 cilvēkiem vienlaikus bez novērojamām spēles ātruma izmaiņām.
  - 1.2. Atbilžu kartīšu sinhronizācija ar vadītāja ekrānu jānotiek reāllaikā, ar ne vairāk kā 2 sekunžu aizturi.
2. Uzticamība un pieejamība
  - 2.1. Sistēmai jānodrošina vismaz 90% laika pieejamību.
  - 2.2. Sistēmā jābūt mehānismam datu dublēšanai un ātrai atjaunošanai pēc avārijas.
  - 2.3. Jābūt iekļautai automātiskai kļūdu atklāšanai un ziņošanai.
3. Drošība
  - 3.1. Sistēmas lietotāji nevar piekļūt datiem kas nav nepieciešami.
  - 3.2. Visas paroles tiek glabātas šifrētā veidā
4. Lietotāja pieredze
  - 4.1. Lietotāja un administratora interfeisam jābūt optimizētam standarta 16:9 1080p monitoram.
  - 4.2. Spēlētāja interfeisam jābūt optimizētam populāriem tālruņa modeļiem.
  - 4.3. Interfeisam jābūt intuitīvam un viegli saprotamam.
  - 4.4. Spēlētājam jāspēj viegli pievienoties spēlei noskenējot QR kodu (skat. 4. att.)
  - 4.5. Lietotājam jāspēj pierakstīties vai reģistrēties sākuma lapā (skat. 2. att.)
  - 4.6. Lietotājam jāspēj dažos vienkāršos soļos izveidot jaunu spēli (skat. 3. att.)
  - 4.7. Lietotājam un spēlētājiem jāredz viegli pārskatāms spēles lauks ar izspēlētajām kārtīm. (skat. 4. att.)



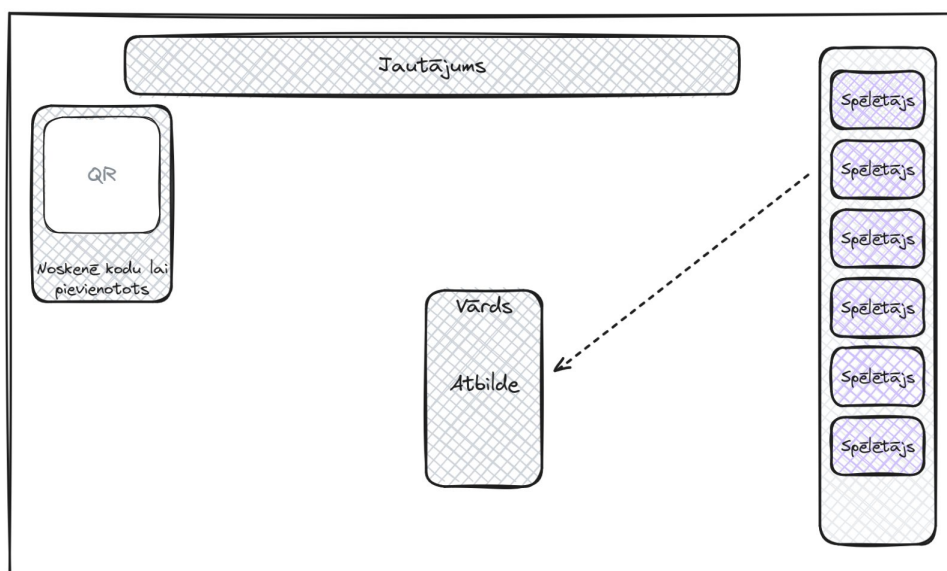
2. att. Glavenās mājaslapas skice

Skice (skat. 2. att.) attēlo galveno mājaslapas lapu, ar iespēju lietotājam ienākt vai reģistrēties.



3. att. Spēles izvēlnes skice

Skice (skat. 3. att.) attēlo spēles izveides izvēlni. Spēlētājs var izvēlēties spēles kārtis, noteikumus, utt. Spēles radīšana notiek vairākos soļos.



4. att. Spēles laukums skice

Skice (skat. 4. att.) attēlo spēles laukumu ar spēlētāju vārdiem, jautājuma kārti, atbilžu kārtīm un QR kodu ar kuru var pievienoties.

### 3. UZDEVUMA RISINĀŠANAS LĪDZEKĻU IZVĒLES PAMATOJUMS

#### 3.1. Programmēšanas valodas

1. **TypeScript** 5.3.3 - Ļauj rakstīt JavaScript kodu ar definētiem mainīgo tiem, atvieglojot programmēšanu un novēršot biežas kļūdas.
2. **Svelte** 4.2.8 - Ļauj veidot interaktīvas lietotāja saskarnes ar daudz mazāku koda apjomu.
3. **SQLite** 3.44.2 – Uztur servera datubāzi tajā pašā procesā kā pats serveris, atvieglojot gan lokālu gan produkcijas uzstādīšanu.

#### 3.2. Palīgprogrammas

1. **SvelteKit** 1.27.7 - Nodrošina pilnu Web serveri ap Svelte programmēšanas valodu ieskaitot maršrutēšanu, HTML ģenerēšanu, utt. Izmantojot SvelteKit nevis vienkāršu Svelte risinājumu, lietotājam ir daudz labāka pieredze mājaslapā.
2. **Vite** 5.0.7 - Izstrādes serveris un būvēšanas rīks. Sniedz ātru lapas atjaunošanu pēc koda izmaiņām, integrējas ar SvelteKit lai optimizētu gala mājaslapas skriptus un stilus, kā arī pārveido TypeScript pārlūkam saprotamā JavaScript valodā.
3. **Prettier** 3.1.1 - Koda formatēšanas rīks. Nodrošina vienotu koda stila ievērošanu visās izmantotajās valodās.
4. **PocketBase** 0.20.0 - Servera risinājums, kas atvieglo bieži izmantotas datubāzes funkcijas, kā piem. ORM, autentifikāciju un reāllaika saziņu izmantojot REST API un JavaScript SDK.
5. **Tailwind CSS** 3.3.6 - Stilizācijas ietvars, kas ļauj izveidot CSS dizainu izmantojot utilītu klases. Ļauj izveidot dizainu daudz ātrāk bez nepieciešamības rakstīt atsevišķu CSS kodu.
6. **PostCSS** 8.4.32 - Integrējas ar TailWind CSS nodrošinot papildus CSS optimizācijas.
7. **SkeletonUI** 2.6.0 - UI komponentu bibliotēka, kas nodrošina gatavus dizaina elementus kā lapas izklājumu, pogas, tabulas, utt. Bibliotēka dziļi integrējas ar Tailwind nodrošinot vienmērīgu dizainu, un Svelte vieglai interaktivitātei.
8. **Nanoid** 5.0.4 - Ģenerē unikālus ID datubāzes sistēmai un testa datiem. Aizņem mazāk atmiņu nekā pārlūkos iekļautais UUID 4 ģenerators.

## 3.2. Čaulas

1. **Node.js** 20.10.0 - Uztur visus izstrādes vides rīkus, kā arī nodrošina produkcijas serveri. Ļauj izmantot JavaScript gan mājaslapai, gan serverim.
2. **Fish** 3.6.4 - Vienkārša Linux čaula ar Python līdzīgu sintaksi. Atvieglo operācijas ar daudziem failiem.
3. **Concurrently** 8.2.2 - Rīks, lai vienlaicīgi palaistu vairākas komandas un apvienotu to izvades. Atvieglo lokālu kodēšanu un atklūdošanu.

## 3.3. Izstrādes vide

1. **Visual studio code** 1.84.2 - IDE ar fokusu uz pārlūkiem un Web tehnoloģijām. Satur nepieciešamus spraudņus priekš Svelte, Tailwind, TypeScript, utt.
2. **Nix** 2.18.1 - Ļauj norādīt nepieciešamās programmas un bibliotēkas konfigurācijas failā, no kura tiek izveidota izstrādes vide. Atvieglo kodēšanas vides uzstādīšanu, pat ja izmanto citu datoru.

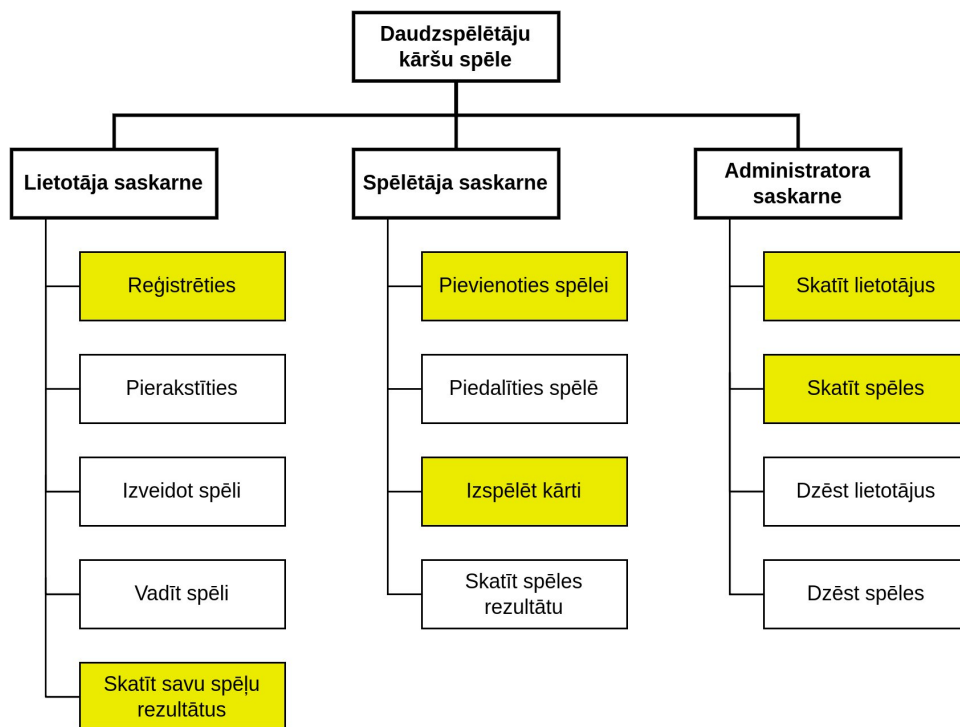
## 4. PROGRAMMATŪRAS PRODUKTA MODELĒŠANA UN PROJEKTĒŠANA

### 4.1. Sistēmas struktūras modelis

#### 4.1.1. Sistēmas arhitektūra

Sistēma (skat. 5. att.) iekļauj trīs galvenās lietotāju grupas: Klienti, Viesi un Administratori. Katrai no šīm grupām ir noteikts funkciju kopums.

1. **Lietotājs** var pierakstīties, izveidot kartīšu komplektus, izveidot jaunas spēles un piedalīties tajās.
2. **Spēlētājs** var pievienoties spēlei un tajā izspēlēt atbilstošu kārti.
3. **Administrators** pārvalda spēles un lietotājus. Tas var skatīt visas spēles, lietotājus un veikt administratīvās darbības, piemēram, dzēst lietotājus vai spēles.

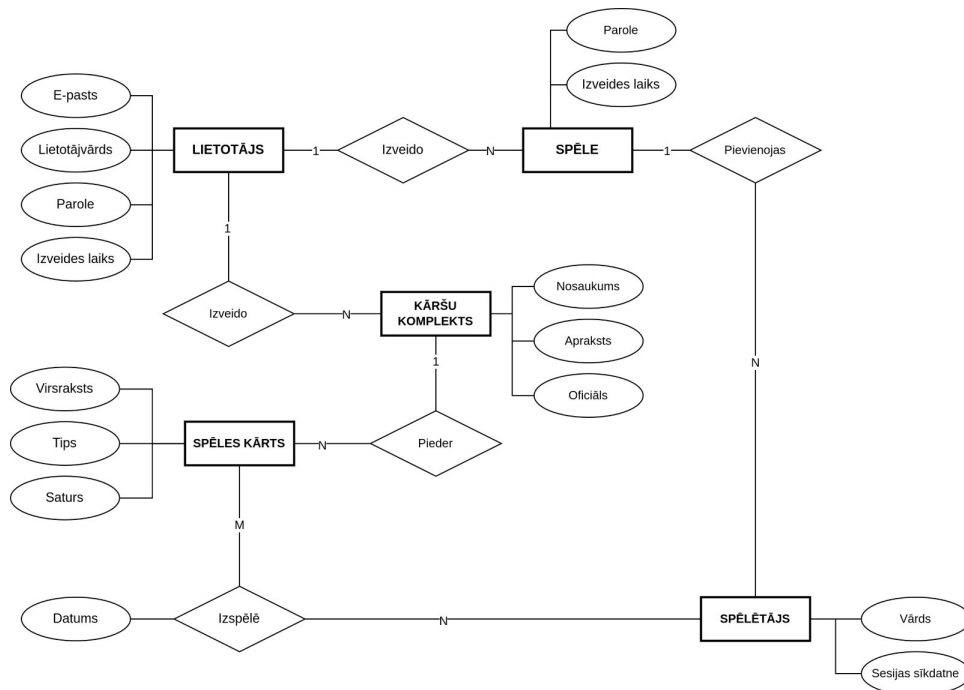


5. att. Funkcionālās dekompozīcijas diagramma

#### 4.1.2. Sistēmas ER modelis

Sistēmas ER-modelis sastāv no 6 entītijām (skat. 6. att.), kas nodrošina sistēmas funkcionalitāti. Modelis demonstrē entītijū savstarpējās saistības:

- starp entītijām „Lietotājs” un „Spēle” ir attiecība viens pret daudziem, jo viens lietotājs var izveidot vairākas spēles;
- starp entītijām „Spēle” un „Spēlētājs” ir attiecība viens pret daudziem, jo vairāki spēlētāji var pievienoties vienai spēlei;
- starp entītijām „Spēlētājs” un „Spēles kārts” ir attiecība daudzi pret daudziem, jo katrs spēlētājs spēles gaitā var izspēlēt vairākas kārtis, un, atkarībā no noteikumiem, vairāki spēlētāji var izspēlēt vienu un to pašu kārti;
- starp entītijām „Kāršu komplekts” un „Spēles kārts” ir attiecība viens pret daudziem, jo vienā kartīšu komplektā var būt vairākas spēles kārtis;
- starp entītijām „Lietotājs” un „Kāršu komplekts” ir attiecība viens pret daudziem, jo viens lietotājs var izveidot vairākus kartīšu komplektus;



6.att. Sistēmas ER-diagramma



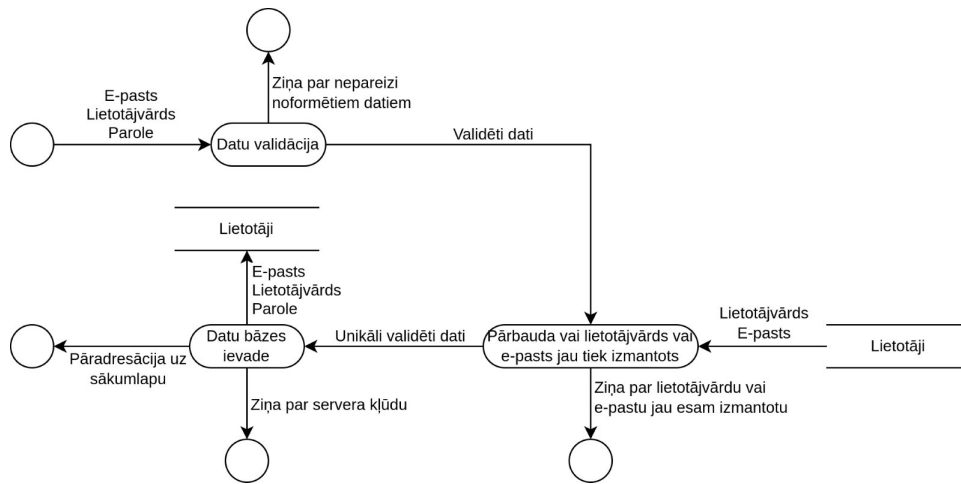
## 4.2. Funkcionālais sistēmas modelis

### 4.2.1. Datu plūsmu modelis

Datu plūsmas diagrammas parāda sistēmas darbību ar ievadītiem datiem un tās darbībām, ieskaitot izmantotās datu bāzes. Dažas no datu plūsmām ieskaita.

#### 1. Lietotāja reģistrācija (skat. 7. att.)

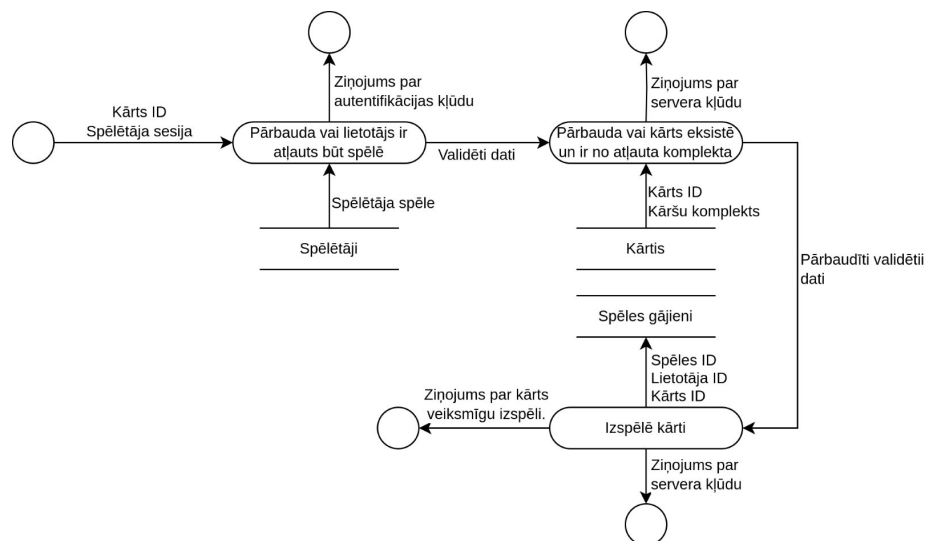
Lai reģistrētos, lietotājs ievada savu e-pastu, lietotājvārdu un paroli. Serveris validē datus, pārbauda vai lietotājs jau eksistē salīdzinot tos ar “Lietotāji” datubāzi, un, ja viss sakrīt, izveido jaunu lietotāju ar dotajiem datiem. Kļūdas tiks parādītas ar “Toast” stila ziņu lietotājam.



7. att. Lietotāja reģistrācijas datu plūsmas diagramma

#### 2. Spēlētāja kārts izspēle (skat. 8. att.)

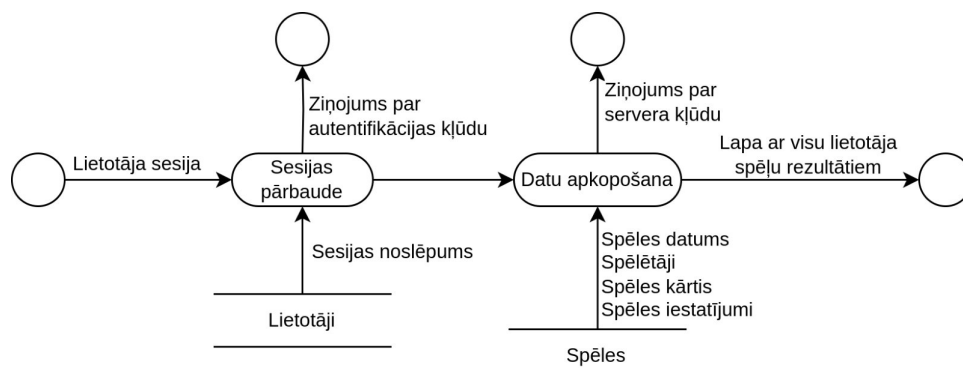
Spēlētājam izspēlējot kārti tiek pārbaudīts vai spēlētājs eksistē, vai spēlētājs to veic pareizajā spēlē un vai kārts eksistē, tad dati tiek ievadīti “Spēles gājieni” tabulā. Kļūdas tiks parādītas ar “Toast” stila ziņu lietotājam.



8. att. Spēlētāja kārts izspēles datu plūsmas diagramma

### 3. Skatīt lietotāja spēles rezultātus (skat. 9. att.)

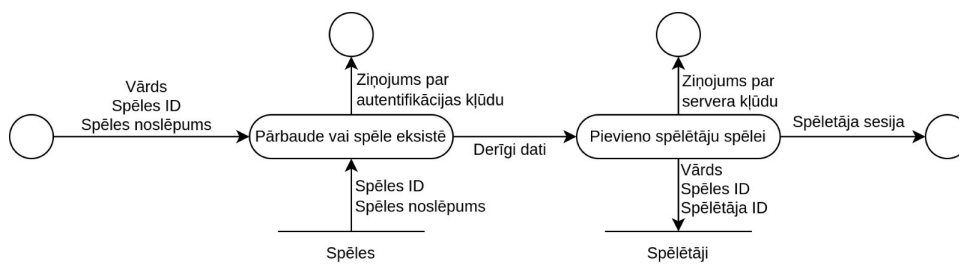
Lietotājam veicot pieprasījumu redzēt savu spēļu rezultātu vēsturi tas nosūta pieprasījumu ar savu sesijas noslēpumu. Serveris dekodē sesiju un atrod lietotāja ID. Serveris tad izmanto ID lai no “Spēles” datubāzes atrastu visas lietotāja spēles, un atgriež tās kā lapu.



9. att. Lietotāja spēles rezultātu datu plūsmas diagramma

### 4. Spēlētāja pievienošanās spēlei (skat. 10. att.)

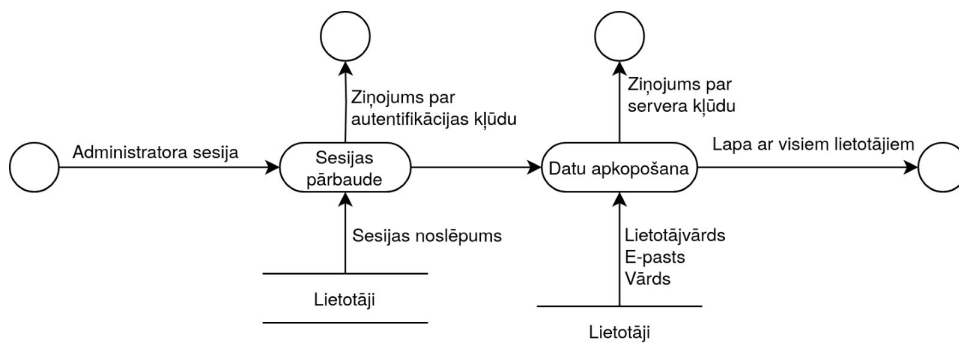
Spēlētājs var pievienoties spēlei iesniedzot savu vārdu, spēles ID un Noslēpumu. Gan spēles ID, gan noslēpums ir pieejami spēles QR koda URL. Sistēma pārbauda šos datus un izveido spēlētāja sesiju. Serveris tad atgriež spēles sesiju spēlētājam kurš to var tālāk izmatot lai veiktu spēles darbības. Kļūdas tiks parādītas kā pilnas lapas teksts ar lūgumu mēģināt pievienoties vēlreiz.



10. att. Spēlētāja pievienošanās datu plūsmas diagramma

## 5. Administratora lietotāju skatīšana (skat. 11. att.)

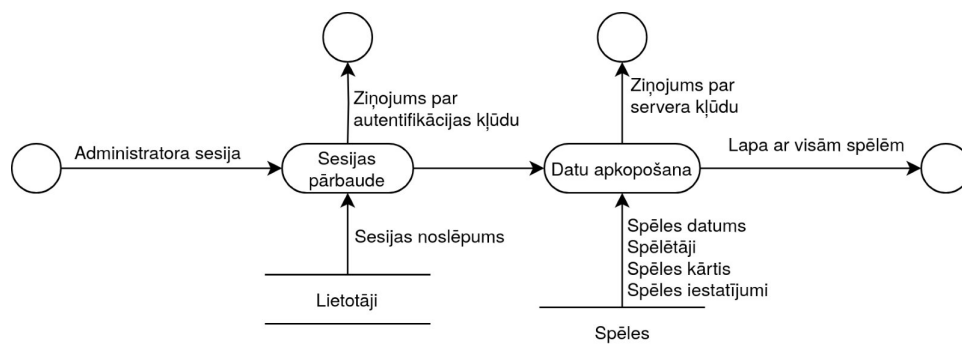
Administrators var redzēt visus lietotājus veicot pieprasījumu serverim ar savu sesijas noslēpumu. Serveris pārbauda vai sesija ir derīga un nāk no administratora, tad apkopo lietotāju datus un atgriež tos kā lapu ar visiem lietotājiem.



11. att. Administratora lietotāju skatīšanas datu plūsmas diagramma

## 6. Administratora spēļu skatīšana (skat. 12. att.)

Administrators var redzēt visus lietotājus veicot pieprasījumu serverim ar savu sesijas noslēpumu. Serveris pārbauda vai sesija ir derīga un nāk no administratora, tad apkopo lietotāju datus un atgriež tos kā lapu ar visām spēlēm.



12. att. Administratora spēļu skatīšanas datu plūsmas diagramma

## 5. DATU STRUKTŪRU APRAKSTS

Datu bāze sastāv no 6 tabulām, kas glabā lietotājus, spēles, kārtis, spēlētājus un kartīšu komplektus. Katra tabula arī iekļauj radīšanas un pēdējās izmaiņas datumus. 3. pielikumā ir dota tabulu relāciju shēma.

Datu bāzes ID lauks ir automātiski ģenerēts kā unikāls 15 rakstzīmju teksts. ID strādā kā primārā atslēga un relāciju mērķis. Šī datu bāzes struktūra atvieglo veidošanu, ļaujot lielāku brīvību pārējiem laukiem.

1. Tabula "**lietotaji**" glabā datus par mājaslapas lietotājiem.
2. Tabula "**karsuKomplekti**" glabā datus par kartīšu komplektiem.
3. Tabula "**spelesGajieni**" glabā datus par katru spēlētāja gājienu spēlē.
4. Tabula "**speletaji**" glabā datus par spēlētājiem.
5. Tabula "**spelesKartis**" glabā datus par spēļu kārtīm.
6. Tabula "**speles**" glabā datus par spēlēm.

1. tabula

Tabulas "**lietotaji**" struktūra.

Nr.	Nosaukums	Tips	Izmērs	Apraksts
1	ID	text	15	Unikāls identifikators.
2	lietotajvards	text	30	Lietotāja vārds
3	vards	text	30	Lietotāja "ekrāna" vārds
4	epasts	text	30	Lietotāja e-pasts
5	created	date	-	Radīšanas datums
6	updated	date	-	Atjaunošanas datums

Tabula (skat. 1. tabula) glabā lietotāja datus kā e-pastu, paroli, un izvēlēto vārdu. Tā arī iekļauj radīšanas un atjaunošanas datumus.

2. tabula

Tabulas "**karsuKomplekti**" struktūra.

Nr.	Nosaukums	Tips	Izmērs	Apraksts
1	ID	text	15	Unikāls identifikators.
2	vards	text	30	Kāršu komplekta nosaukums
3	official	bool	1	Vai komplekts ir oficiāls
4	Radītāja ID	text	15	Saite uz lietotāju, kas izveidojis
5	apraksts	text	200	Kāršu komplekta apraksts
5	created	date	-	Radīšanas datums
6	updated	date	-	Atjaunošanas datums

Tabula (skat. 2. tabula) glabā datus par kartīšu komplektu, tā aprakstu un radītāju. Tā arī iekļauj radīšanas un atjaunošanas datumus.

Tabulas “spelesGajieni” struktūra.

Nr.	Nosaukums	Tips	Izmērs	Apraksts
1	ID	text	15	Unikāls identifikators.
2	Spēlētāja ID	text	15	Saite uz spēlētāju, kas veicis gājienu.
4	Kārts ID	text	15	Saite uz kārti, kas tika izmantota gājienā.
5	created	date	-	Radišanas datums

Tabula (skat. 3. tabula) Glabā datus par spēles gājieniem. Šī ir starptabula starp spēlētāju un spēles kārtīm. (skat. 2. att.)

Tabulas “speletaji” struktūra.

Nr.	Nosaukums	Tips	Izmērs	Apraksts
1	ID	text	15	Unikāls identifikators.
2	Spēles ID	text	15	Saite uz spēli, kurā spēlētājs piedalās.
3	vards	text	30	Spēlētāja vārds.
4	secret	text	32	Spēles parole.
5	created	date	-	Radišanas datums

Tabula (skat. 4. tabula) Glabā datus par spēlētāju. Spēlētājs ir unikāls spēlei. Tā arī iekļauj radišanas un atjaunošanas datumus.

Tabulas “spelesKartis” struktūra.

Nr.	Nosaukums	Tips	Izmērs	Apraksts
1	ID	text	15	Unikāls identifikators.
2	Komplekta ID	text	15	Saite uz kartīšu komplektu
3	virsraksts	text	30	Kārts virsraksts
4	saturs	text	100	Kārts saturs
5	custom	json	100	Pielāgoti dati JSON formātā
6	tips	select	-	Kārts tips (jautājuma, atbilžu)
5	created	date	-	Radišanas datums
6	updated	date	-	Atjaunošanas datums

Tabula (skat. 5. tabula) glabā datus par spēles kārti, ieskaitot tās virsrakstu un saturu, kā arī papildus informāciju par tās radišanu.

Tabulas “**speles**” struktūra.

Nr.	Nosaukums	Tips	Izmērs	Apraksts
1	ID	text	15	Unikāls identifikators, 15 simboli garš
2	Radītāja ID	text	15	Saite uz spēles izveidotāju
3	secret	text	32	Slepenais kods, kas saistīts ar spēli
4	noteikumi	json	100	Spēles noteikumi JSON formātā
5	Komplektu ID	relation[]	15+	Saite uz vairākiem kartīšu komplektiem, kas lietoti spēlē.
6	Kārts ID	text	15	Saite uz jautājuma kārti
5	created	date	-	Radīšanas datums
6	updated	date	-	Atjaunošanas datums

Tabula (skat. 6. tabula) glabā datus par spēli, ieskaitot tās radītāju un paroli.

## **6. LIETOTĀJA CEĻVEDIS**

### **6.1. Sistēmas prasības**

Lai lietotājs varētu izmantot izstrādāto tiešsaistes un klātienēs daudzpēlētāju diskusiju rīku izklaidēm un situāciju modelēšanai, ir nepieciešams:

Vadītājam ir nepieciešams:

- Moderns pārlūks:
  - Safari – 14. versija vai jaunāka;
  - Windows Edge – 98.0.1108.43 vai jaunāka;
  - Google Chrome – 98.0.4758.107 vai jaunāka;
  - Opera – 84.0.4316.31 vai jaunāka;
  - Mozilla Firefox – 96.0 vai jaunāka;
- Stabils interneta savienojums

Spēlētājam ir nepieciešams:

- Moderna mobilā ierīce;
- Iebūvēts QR koda lasītājs vai aplikācija ar šo funkciju;
- Moderns mobilais pārlūks;
- Stabils interneta savienojums;

## 6.2. Sistēmas instalācija un palaišana

Šī rīka uzstādīšanai ir nepieciešami vairāki soļi, lai izveidotu programmēšanas vidi un uzbūvētu sistēmu.

### 6.2.1. Izstrādes vides sagatavošana

Vispirms ir nepieciešams nodrošināt, ka uz datora ir uzstādīta izstrādes vide ar nepieciešamajiem rīkiem. Tas ietver Linux operētājsistēmu vai WSL, Node.js, Visual Studio Code, Nix, u.c. Kas ir norādīti 3. sadaļā. Node.js ir nepieciešams, lai uzturētu izstrādes vides rīkus un produkcijas serveri, savukārt Visual Studio Code kalpos kā integrētā izstrādes vide (IDE), kurā tiks veikta koda izstrāde un pārvaldība. Nix nodrošinās nepieciešamās programmas un bibliotēkas konfigurācijas failā, no kura tiks izveidota izstrādes vide.

### 6.2.2. Koda iegūšana un uzstādīšana

Nākamais solis ir iegūt sistēmas kodu no GitHub. Pēc koda iegūšanas, izpildiet komandu *pnpm install*, lai instalētu visas bibliotēkas, kas nepieciešamas projekta izstrādei.

### 6.2.3. Servera palaišana

Pēc visu nepieciešamo bibliotēku instalēšanas, izpildiet komandu *pnpm dev*, lai palaistu izstrādes serveri, izmantojot Vite un SvelteKit. Pārliecinieties, ka sistēma darbojas pareizi, piekļūstot tai caur pārlūkprogrammu saitē *http://localhost:3000*.



### **6.3. Programmas apraksts**

Ir jāapraksta visas sistēmas katru lietotāja saskarnes (GUI) elementu ar visām iespējamiem tajā darbībām un servisu. Katram GUI elementam jādod ekrāna paraugu ar ievadinformācijas piemēriem. Tas ir garākais punkts dokumentā. Ekrāna uzņēmumi nevar sekot viens aiz otra, starp tiem obligāti jābūt tekstam. Uz katru attēlu jābūt atsaucei tekstā. Jābeidz šo sadaļu ar norādījumiem, kā pabeigt darbību ar jūsu programmu.

Tiks izdarīts vēlāk.

## 6.4. Testa piemērs

Tālāk ir aprakstīti funkcionālo prasību testēšanas procesi, kas tika veikti, lai pārbaudītu sistēmas darbību.

Nr.	Prasības numurs	Prasības nosaukums	Ievaddati/Situācijas apraksts	Sagaidāmais rezultāts	Statuss
1	1	Lietotāju reģistrācija	Ievada lietotājvārdu, e-pastu un paroli atbilstoši prasībām	Lietotājs veiksmīgi reģistrēts	Pareizi
2			Ievada tikai lietotājvārdu	Kļūdas paziņojums par trūkstošu paroli	Pareizi
3			Ievada esošu lietotājvārdu	Kļūdas paziņojums par lietotājvārda aizņemtību	Pareizi
5	3	Spēles izveide un vadīšana	Izveido jaunu spēli un pievieno izveidotu kartīšu komplektu	Spēle veiksmīgi izveidota, ģenerēts unikāls QR kods	Pareizi
7			Izveido spēli un spēlētājs mēģina pievienoties skenējot ģenerēto QR kodu	Spēlētājs veiksmīgi pievienojas spēlei	Pareizi
8			Izveido spēli un spēlētāji mēģina pievienoties ar nepareizu QR kodu	Spēlētājs nepievienojas spēlei, kļūdas ziņojums par nepareizu kodu	Pareizi
9	2	Kāršu komplektu izveide un pārvaldība	Lietotājs izveido jaunu kartīšu komplektu ar nosaukumu un tekstu	Kāršu komplekts veiksmīgi izveidots	Pareizi
10			Lietotājs pievieno jaunu karti komplektam	Karte veiksmīgi pievienota komplektam	Pareizi
11			Lietotājs mēģina pievienot karti bez teksta	„Izveidot” poga nav aktīva, neļaujot lietotājam izveidot tukšu kārti	Pareizi

## NOBEIGUMS

Kvalifikācijas darba izstrādes laikā ir izveidots tiešsaistes un klātienes daudzpēlētāju diskusiju rīks, kas paredzēts izklaidēm un situāciju modelēšanai. Projekta ietvaros veiksmīgi realizētas vairākas būtiskas funkcijas, ieskaitot spēles izveidi un vadīšanu, lietotāju reģistrāciju un administrēšanu, kā arī kartīšu komplektu izveidi, rediģēšanu un dzēšanu. Sistēma tika izstrādāta, izmantojot mūsdienīgas tehnoloģijas un programmēšanas valodas, piemēram, TypeScript un Svelte, kā arī tādus rīkus kā SvelteKit, PocketBase un Tailwind CSS. Darba izstrādes laikā uzstādītie uzdevumi tika izpildīti veiksmīgi.

Pašlaik sistēma produkcijā netiek izmantota, jo dēļ lielā daudzuma elementu un API, kas ir nepieciešami, tās drošība nav pārbaudīta. Sistēma ir veiksmīgi testēta un demonstrēta, taču tās drošības aspekti vēl nav pilnībā nodrošināti, lai to varētu izmantot reālos apstākļos ar plašu lietotāju loku.

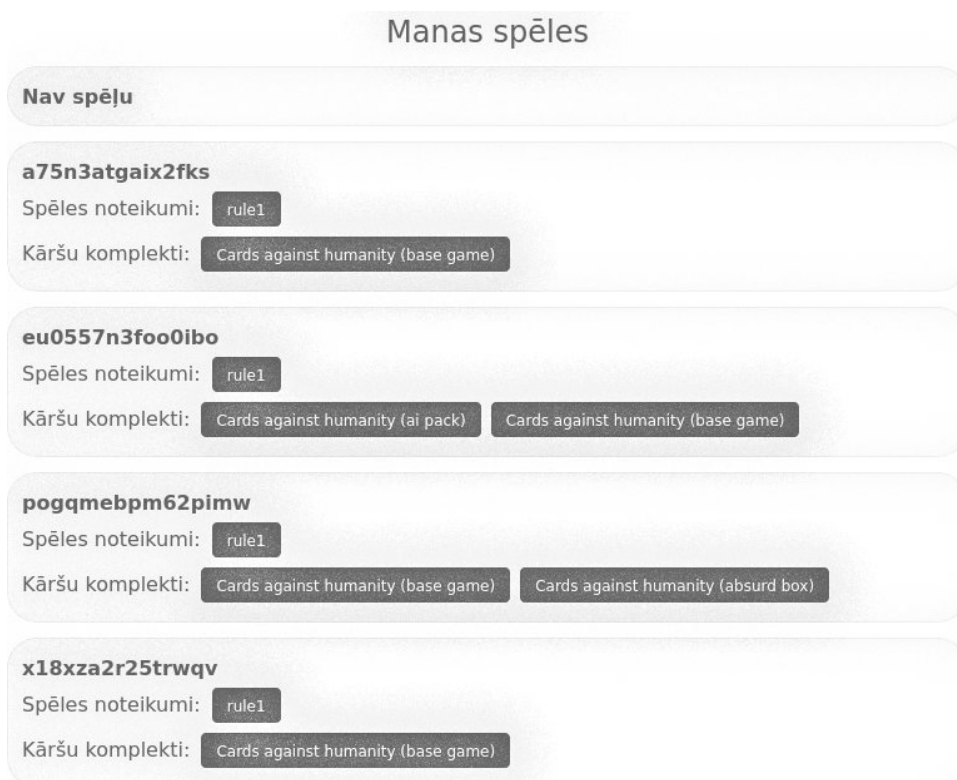
Nākotnē šo sistēmu varēs izmantot vēl plašāk, papildinot tās funkcionalitāti un pielietojuma iespējas. Iespējams, to varēs attīstīt par lielāku platformu, kas piedāvā vēl vairāk spēles scenāriju un pielāgošanas iespēju. Turpmākā attīstība varētu ietvert papildus izglītojošo spēļu komplektus, e-veikalu papildu komplektu iegādei, kā arī integrāciju ar citām izklaides un mācību platformām, tādējādi paplašinot lietotāju loku un uzlabojot sistēmas funkcionalitāti.

## INFORMĀCIJAS AVOTI

- [ANG] SvelteKit dokumentācija - <https://kit.svelte.dev/docs/introduction>
- [ANG] PocketBase dokumentācija - <https://pocketbase.io/docs>
- [ANG] Web Development Best Practices - <https://www.thinkful.com/blog/web-development-best-practices/>
- [ANG] SvelteKit Tutorial: Build a Website From Scratch - <https://prismic.io/blog/svelte-sveltekit-tutorial>
- [ANG] Creating Responsive views in Svelte.js and SvelteKit - <https://adamgreen.tech/blog/responsive-views-in-svelte-js>
- [ANG] ER diagrammas definīcija un piemēri - [https://en.wikipedia.org/wiki/Entity%E2%80%93relationship\\_model](https://en.wikipedia.org/wiki/Entity%E2%80%93relationship_model)
- [ANG] Diagrammu veidošanas pamācība - <https://plantuml.com/guide>
- [ANG] SkeletonUI dokumentācija - <https://www.skeleton.dev/docs/get-started>
- [ANG] TailWind dokumentācija - <https://tailwindcss.com/docs/installation>
- RVT dokumentācijas noformēšanas noteikumi - <https://e.rvt.lv/mod/resource/view.php?id=18424>
- Lietojumgadījumu diagrammas definīcija un piemēri - <https://e.rvt.lv/mod/resource/view.php?id=19115>
- [ANG] How To Use Pocketbase Auth For A Login Screen (step-by-step) - <https://www.programonaut.com/how-to-use-pocketbase-auth-for-a-login-screen-step-by-step/>
- [ANG] How to Build a Simple API in SvelteKit 2023 - <https://codevoweb.com/how-to-build-a-simple-api-in-sveltekit/>
- [ANG] Application state management with Svelte - <https://blog.logrocket.com/application-state-management-svelte/>
- [ANG] SvelteKit Tutorial: Build a Website From Scratch - <https://prismic.io/blog/svelte-sveltekit-tutorial>

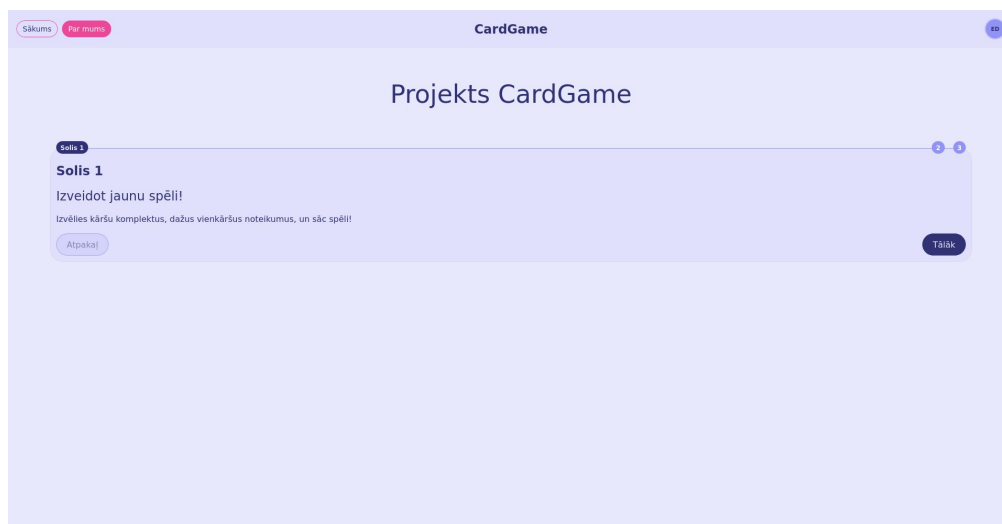
# PIELIKUMI

## 1. pielikums. Pārskatu piemēri



13. att. Lietotāja spēļu pārskats

## 2. pielikums. Ekrānattēlu piemēri



14. att. Mājaslapas sākumlapa ar spēles izveides rīku

CardGame

Sākums Par mums

Iestatījumi Drošība Kāršu komplekti Spēļu vēsture

Vārds  
Edgars

E-pasts  
edgars@example.com

Saglabāt

15. att. Lietotāja iestatījumi

CardGame

Sākums Par mums

Iestatījumi Drošība Kāršu komplekti Spēļu vēsture

## Mani kāršu komplekti

**Mans kāršu komplekts** Kāršu komplekta apraksts  
Rediģēt Dzēst

**Jāna balīte** Jāna dzimšanas dienas balītes jautājumi  
Rediģēt Dzēst

**Darba sapulce** Tēmas kuras jādiskutē  
Rediģēt Dzēst

**Salidojums** Skolas atmiņas un stāsti  
Rediģēt Dzēst

**Izveidot jaunu kāršu komplektu**

Nosaukums

Apraksts

Izveidot

16. att. Lietotāja kartīšu komplektu pārvaldīšana

CardGame

Sākums Par mums

Kāršu komplekts: Mans kāršu komplekts

Nosaukums  
Mans kāršu komplekts

Apraksts  
Kāršu komplekta apraksts

Saglabāt

**Kārts 1**  
Kārts 1 saturs  
Dzēst

**Kārts 2**  
Kārts 2 saturs  
Dzēst

**Kārts 3**  
Kārts 3 saturs  
Dzēst

**Izveidot jaunu kārti**

Virsraksts

Saturs

Tips  
Jautājums

Izveidot

17. att. Kāršu pārvaldīšana

### 3. pielikums. Tabulu relāciju shēma

