**Projekts - Filmu arhīvs**

**Problēmas apskats:** Bieži vien, vēloties uzzināt pilnvērtīgu informāciju par kādu televīzijas seriālu vai filmu ir jāveic laikietilpīgi meklējumi dažādās saitēs, kurās informācija mēdz būt neprecīza un nesakrīt ar citiem avotiem, tāpēc ir nepieciešamība vienai saitei, kas spēj nolasīt visu vajadzīgo informāciju un izsniegt to klientam bez liekas laika tērēšanas.

**Projekta mērķis un specifikācijas:**

Izveidot funkcionējošu tīmekļa vietni ar viegli izmantojamu un saprotamu dizainu visiem vecuma cilvēkiem, kas:

* Ļauj lietotājam ielogoties savā personīgajā kontā.
* Iegūt viegli salasāmu informāciju par savām mīļākajām filmam un televīzijas šoviem.
* Dod iespēju saglabāt meklēto informāciju savā kontā un viegli aplūkot to vēlāk.
* Salīdzināt savu mīļāko filmu statistikas.
* Administrātoram ir iespēja pārlūkot lietotāju daudzumu un datus.

Izmantošu Python un HTML, lai izveidotu funkcionējošu tīmekļa saskarni, kas ir viegli savienojama izmantojot Python un vizuāli pielāgojama izmantojot HTML un CSS.

**Lietotājstāsti:**

Identifikatora struktūra:

FA\_V1, kur:

* FA norāda uz projektu – filmu arhīvs
* \_ atdala projektu no prasības
* V norāda uz lietotāja lomu (V – viesis, L – lietotājs, A – administrators)
* 1 ir prasības kārtas numurs, kurš neatkārtojas

1. FA\_V1: Kā viesies, es vēlos ātri un vienkārši iegūt informāciju par kādu filmu.
2. FA\_L1: Kā lietotājs, es vēlos iegūt informāciju par filmām ērtā veidā un saglabāt savas mīļākās filmas savā kontā.
3. FA\_A1: Kā administrators, es vēlos īpašus login datus un zināt cik daudz lietotāju izmanto manu sistēmu.

1.

* Meklēšanas logs redzams lapas vidū.
* Zem meklēšanas loga ir iespēja reģistrēties vai pieslēgties esošam kontam.
* Ievadot filmas vai šova nosaukumu, ekrānā parādās visa iespējamā informācija par to.
* Pieejama opcija doties atpakaļ uz sākuma lapu pēc informācijas apskatīšanas.

2.

* Meklēšanas logs redzams lapas vidū
* Ielogojoties, sākuma lapā ir pieejama opcija izlogoties un pārlūkot profilu
* Ievadot filmas vai šova nosaukumu, ekrānā parādās visa iespējamā informācija par to.
* Pieejama opcija meklēto šovu saglabāt savā kontā ar pogu vai arī doties atpakaļ uz sākuma lapu.
* Profilā pārskatā redzamas saglabātās filmas un to informācija, kā arī poga, kas ved uz filmu salīdzināšanu.
* Filmu salīdzināšanā ir divi logi, kuros var izvēlēties savas saglabātās filmas un pa vidu tām ir poga ‘’salīdzināt’’
* Salīdzinot filmas, tiek iegūti abu filmu statistikas dati, kas ir salīdzināti vieni pret otru.

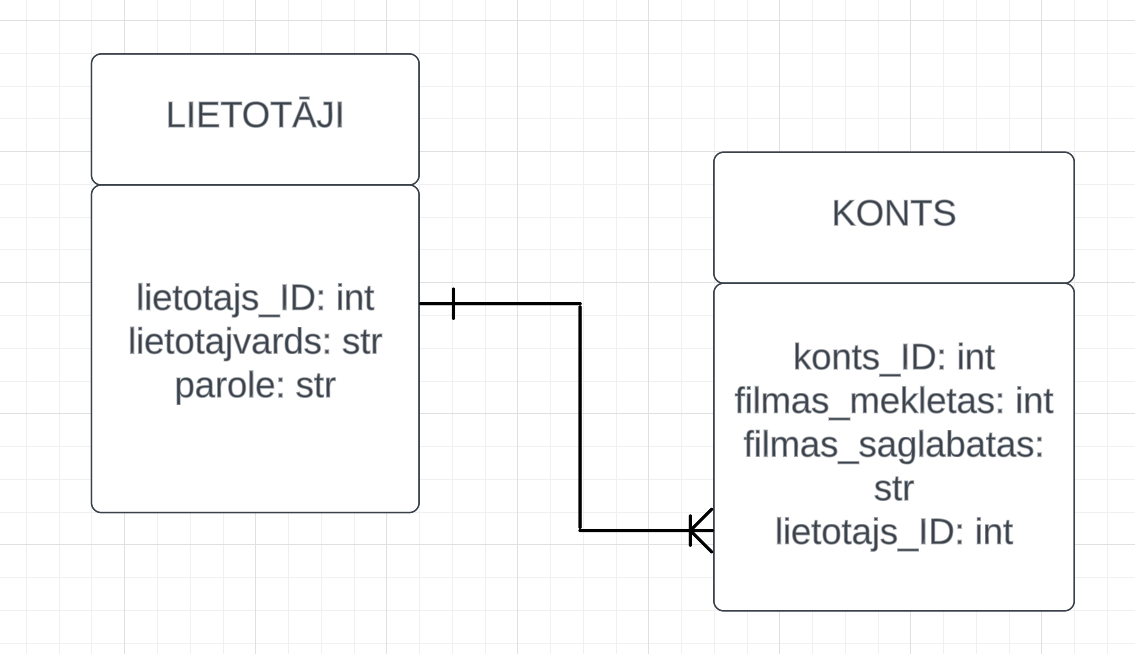
3.

* Meklēšanas logs redzams lapas vidū.
* Ievadot filmas vai šova nosaukumu, ekrānā parādās visa iespējamā informācija par to.
* Sākuma lapā ir poga, kas atver sarakstu ar visiem rēgistrētajiem profiliem (lietotājvārds, meklēto filmu skaits, saglabāto filmu skaits)
* Ir opcija izlogoties no administrātora konta

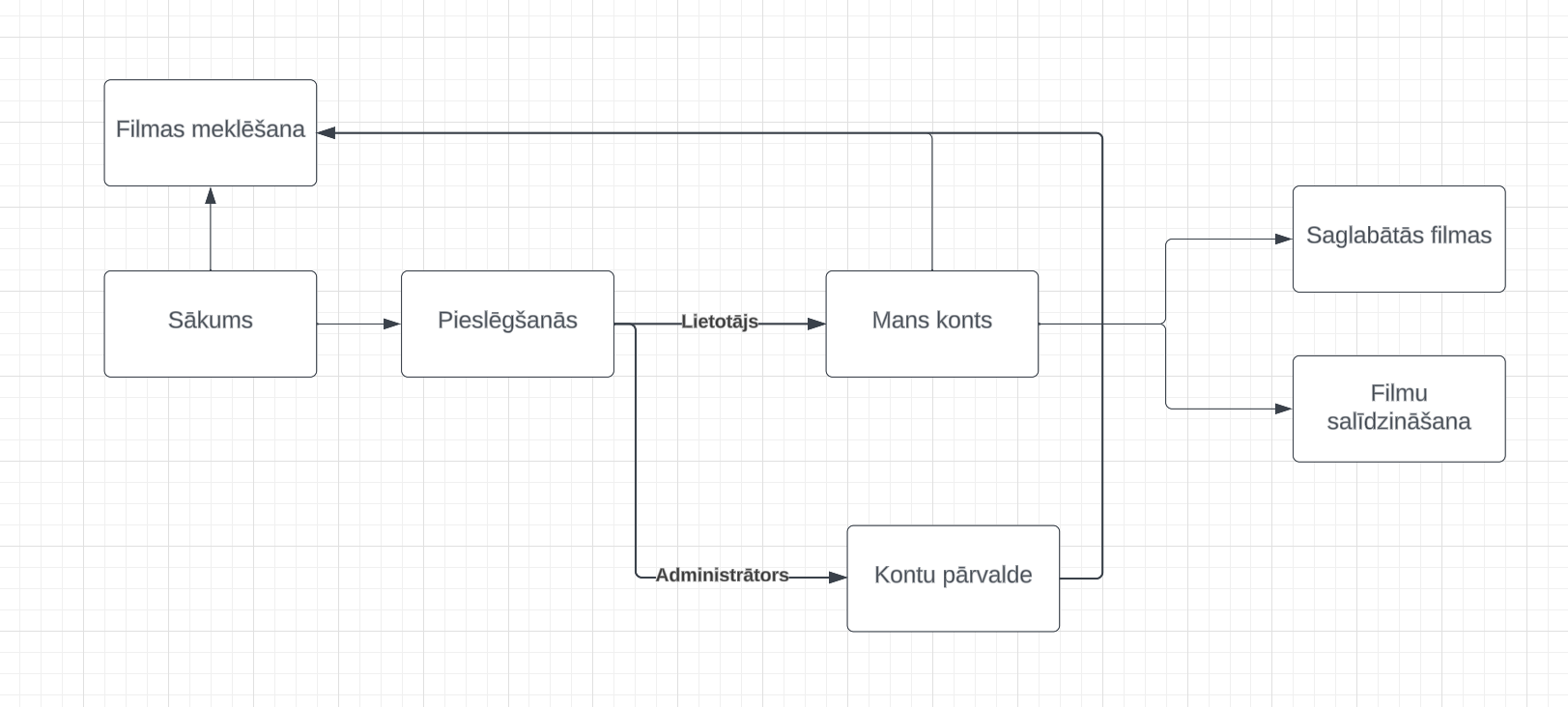
**Prioritātes:**

* Lai saglabātu meklētās filmas ir nepieciešama lietotāja reģistrācija, tai ir augstāka prioritāte nekā saglabāto filmu attēlošana.
* Filmas informācijas attēlošana profilā atvieglo procesu, bet nav obligāta. Augstāka prioritāte ir filmu nosaukumu attēlošana, jo lietotājs spēj sameklēt informāciju atkartoti, ja ir zināms filmas nosaukums.
* Statistiku salīdzināšana ir atkarīga no lietotāju konta saglabātajām filmām, tā kā prioritāte filmu saglabāšanai nav visaugstākā, tad to salīdzināšanai arī nav ļoti augsta.
* Administrātora iespēja dzēst profilus nav obligāta, tapēc prioritāte tai ir zema.
* Iespēja izziet no profila ir nepieciešama, lai aizsargātu savus datus, tai ir augstākā prioritāte.

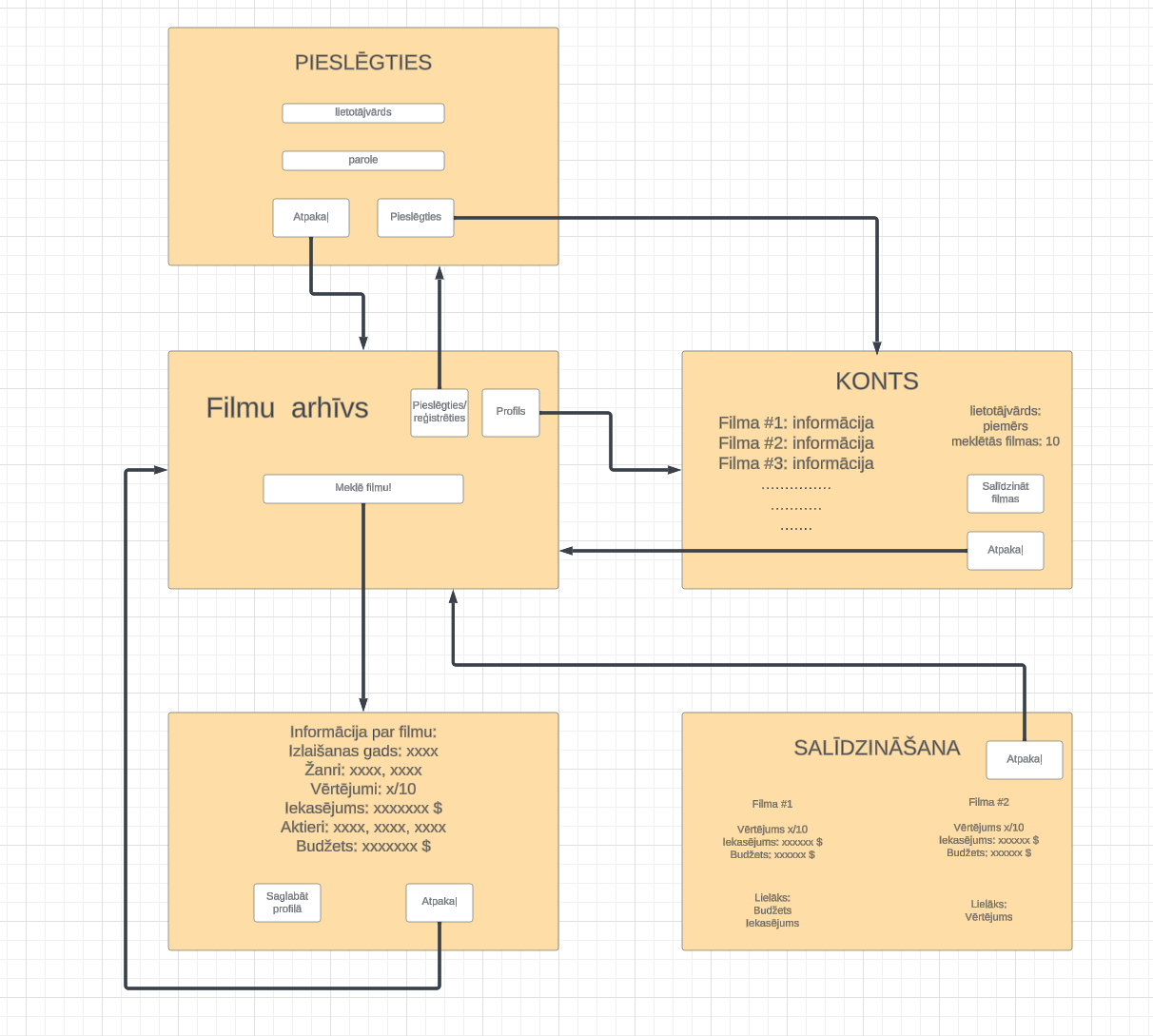
**Datu bāzes projektējums:**



**Saskarņu atkarības:**

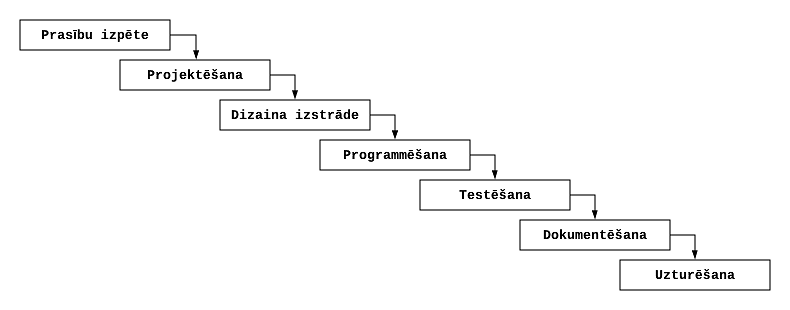


**Dizaina izstrāde:**



**Programmatūras izstrādes plans:**

Savā projektā izmantošu ūdenskrituma izstrādes modeli:



Mans plāns ir tiešs un, manuprāt, tajā neredzu ko vel pievienot, tas nozīmē, ka sekojot ūdenskrituma izstrādes modelim, es varu perfekti izpildīt katru izplānoto soli un nonākt pie pilnveidīga un pabeigta beigu produkta.

**Prasību izpēte, projektēšana un dizaina izstrāde ir norādīta dokumenta sākumā.**

**Programmēšana:**

Zinot savu uzdevumu, izmantošu savu mīļāko un, manuprāt, piemērotāko valodu šajam projektam – python. Keršos izveidot nelielu prototipu, kurā izveidošu pašus pamatus kā: datubāzes un tabulu izveidošanu, programmatūras URL ceļu izveidi caur @app.route, pogas ar kurām pārvietoties pa URL ceļiem pēc dizainā izstrādātās sistēmas. Kad būs ievietoti pamati, atradīšu piemērotu API ar kuru var iegūt precīzus un pareizus datus par filmām un televīzijas šoviem, tad izmantošu to, lai izveidotu funkciju, kas ļaus lietotājam ievadīt vēlēto filmu un ievadīto nosaukumu ievietos URL no API adreses, rezultātā iegūstot informāciju par vēlēto filmu. Tad pievienošu login sistēmu un veidu kā izlogoties no konta, kā arī izveidošu speciālu login sistēmu tikai administrātoram/iem. Pēdējais galvenais solis ir pievienot profila pārskatu, kas ļaus lietotājam redzēt dažādas statistikas par savu profilu un veidu kā salīdzināt divas dažādas filmas, tad vizuāli noformēt visu programmu izmantojot CSS. Pārējais ir ar mazāku prioritāti, bet centīšos paveikt visus iespējamos prasījumus!

**Testēšana:**

Alfa testēšanai izmantošu savu draugu, kurš pārzin programmēšanu un spēs atrast kļūdas un nodot tās man analīzei.

Beta testēšanai lūgšu palīdzību vecākiem, kas spēs izmantot programmu un, iespējams, pamanīs retāk sastopamas kļūdas par kurām ziņos man un es tās analizēšu.

**Dokumentēšana:**

Izveidošu lietotāja rokasgrāmatu ar gatavo programmu un sniegšu viegli saprotamas vizuālās instrukcijas un aprakstus katrai pieejamai funkcijai.

**Uzturēšana:**

Caur replit.com būs iespējams piekļūt uz izmantot programmu, kur tā tiks uzturēta vieglai pieejai. Kā arī github būs pieejams publicētais projekts, dokumentācija un lietotāja dienasgrāmata, kas ļaus klientiem apgūt programmu.

**Atkļūdošanas un akcepttetēšanas pārskats:**

Projekta programmēšanas izstrādes laikā biju saķēries ar daudzām dažādām kļūdām, kas sākotnēji padarīja produktu kļudainu un pat pilnībā nefunkcionējošu, taču spēju izlabot visas kļudas ar saviem spēkiem, kā arī alfa un beta testētājiem.

Izveidojot programmas login sistēmu, nonācu pie situācijas, kur jebkurš lietotājs spēja ielogoties administratora lapā un pārlūkot visus kontus, kas ir milzīga problēma, jo tas padarīja lietotāju datus pilnībā neaizsargātus. Apgāju šo problēmu izveidojot atsevišķu tabulu, kurā ir tikai un vienīgi administratora konta dati un pievienoju atsevišķu slēptu login sistēmu tikai administratoriem. Tādā veidā lietotāji var izmantot savu login sistēmu un ienākt tikai un vienīgi lietotāja lapā.

Pēc veiksmīgas API savienošanas ar lietotāja vēlētajiem filmu nosaukumiem, radās kļūda, kas rezultātu lapā nerādija filmu plakātus un visus datus attēlotajā tabulā rādīja par 1 vienību uz priekšu (Nosaukuma vietā Gads, Gada vietā Žanrs utt.), sākumā konstatēju, ka problēma ir ar pašu API, kas, iespējams, izdrukā kļūdainu JSON failu, bet patiesībā problēma bija Python kodā. Kad veidu tabulu filmas datiem, netīšām nepievienoju vienu vienību un kad tabula tika izdrukāta, aizmirstā vienība tika izlaista. Filmas plakāts netika attēlots dēļ nepareiza HTML koda, kurā netika specificēts, ka vienību ir jāatēlo kā bildi. Abas kļūdas tika salabotas un ar rezultātu attēlošanu vairs nebija problēmu.

**Alfa testēšana:**

Lietotāja profila lapā nebija pogas ar ko iziet uz sākuma lapu, kas tika ātri pamanīts un atkļūdots, pievienojot “atpakaļ’’ pogu, kas aizveda lietotāju uz sākuma lapu.

Profila logā nebija redzama mīļāko filmu informācija, tikai to nosaukumi. Tika veikts kompromiss priekš laika ietilpīguma un šī funkcija tika vienkāršota.

Ja lietotājs devās uz sākuma lapu un ielogojās citā kontā, tad visa informācija saglabājās iepriekšējā kontā. Problēma tika atkļūdota izdzēšot sessijas tabulu un izveidojot to no jauna kad lietotājs vēlas ienākt savā kontā.

**Beta testēšana:**

Tika konstatēta kļūda, kur pilnībā jauni konti pēc programmas izlaišanas beta versijā, savā profila lapā pie mīļākajām filmām redz komatu (,) pirms pirmās mīļākās filmas ieraksta. Tā ir maza drukas kļūda, kas tika atkļūdota pievienojot if konstrukciju, kas nepievieno komatu pirms pirmās mīļākās filmas.

Izmantojot administratora kontu, aplūkojot visus kontus, kas izveidoti, nav iespējams tos dzēst. Šī funkcija nav nepieciešama priekš programmas galvenās funkcionalitātes tāpēc tika nepievienota.

**Pēc atkļūdošanas tika veikti testpiemēri:**

1. FA\_V1\_T1

Scenārijs: Viesis ieraksta meklēšanas logā savu vēlēto filmas nosaukumu.

Sasniedzamais rezultāts: Tiek atvērta rezultātu lapa ar visu pieejamo informāciju un iespēju doties atpakaļ uz sākumu.

1. FA\_V1\_T2

Scenārijs: Viesis reģistrē savu kontu

Sasniedzamais rezultāts: Viesis nonāk sākuma lapā un viņa konts tiek reģistrēts datubāzē.

1. FA\_V1\_T3

Scenārijs: Viesis ienāk savā kontā

Sasniedzamais rezultāts: Viesis kļūst par lietotāju un nonāk lietotāja sākuma lapā, kur var aplūkot savu profilu, meklēt filmas un tās saglabāt.

1. FA\_L1\_T1

Scenārijs: Lietotājs ieraksta meklēšanas logā savu izvēlēto filmas nosaukumu

Sasniedzamais rezultāts: Tiek atvērta rezultātu lapa ar visu pieejamo informāciju un iespēju doties atpakaļ uz sākumu vai saglabāt filmu profilā.

1. FA\_L1\_T2

Scenārijs: Lietotājs aplūko savu profila lapu

Sasniedzamais rezultāts: Tiek parādīts lietotāja lietotājvārds, meklēto filmu skaits un mīļākās filmas, blakus stastikai ir iespēja salīdzināt filmas vai doties atpakaļ uz sākuma lapu.

1. FA\_L1\_T3

Scenārijs: Lietotājs izziet no sava konta

Sasniedzamais rezultāts: Lietotājs tiek nogādāts uz sākumlapu kā viesis, kur ir iespēja atkārtoti ielogoties, reģistrēties vai meklēt filmu kā viesis.

1. FA\_L1\_T4

Scenārijs: Lietotājs nepareizi ievada sava konta pieslēgšanās datus

Sasniedzamais rezultāts: Lietotājs tiek nogādāts atpakaļ pieslēgšanas logā ar prasību ievadīt datus velreiz.

1. FA\_A1\_T1

Scenārijs: Administrators ievada URL adresi /konts\_pieslegties\_admin un pievienojās administratora kontam (lietotajvards: Galvenais, parole: 2468)

Sasniedzamais rezultāts: Administrators tiek nogādāts administratora sākuma lapā

1. FA\_A1\_T2

Scenārijs: Administrators iereksta meklēšanas logā filmas nosaukumu

Sasniedzamais rezultāts: Tiek atvērta rezultātu lapa ar visu pieejamo informāciju par filmu (tāpat kā viesu lapa 1. Testā)

1. FA\_A1\_T3

Scenārijs: Administrators ieiet logā, kur spēj apskatīt visus lietotāju kontus.

Sasniedzamais rezultāts: Administrators redz tabulu ar visu lietotāju kontu informāciju

1. FA\_L1\_T3

Scenārijs: Administrators iziet no administratora konta.

Sasniedzamais rezultāts: Administrators tiek nogādāts uz sākumlapu kā viesis, kur ir iespēja atkārtoti ielogoties, reģistrēties vai meklēt filmu kā viesis.

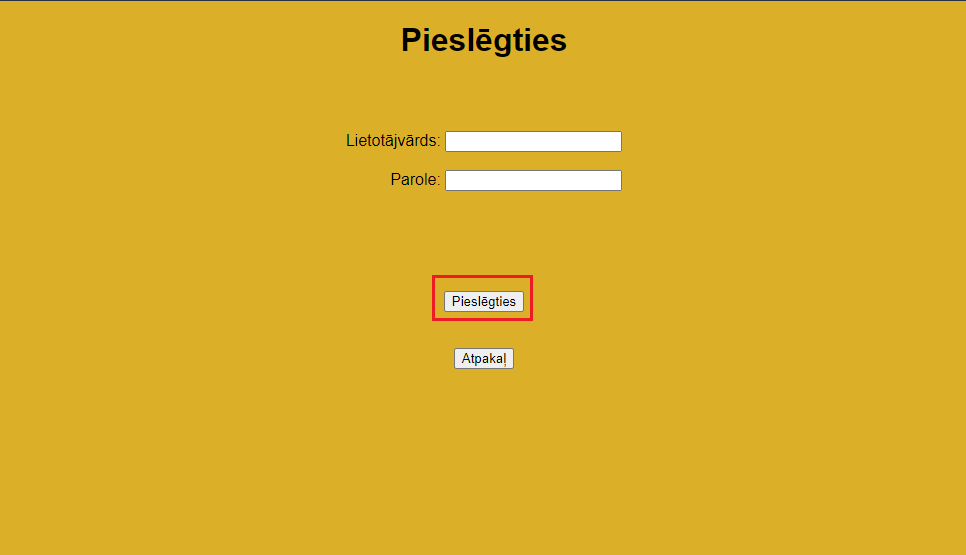
Pēc atkļūdošanas un akcepttestēšanas es varu secināt pēc savas un citu pieredzes, ka produkts ir funckionējošs. Tiek pilnībā vai daļēji izpildītas visas solītās funkcijas bez kļūdām.

**Lietotāja rokasgrāmata/ceļvedis:**

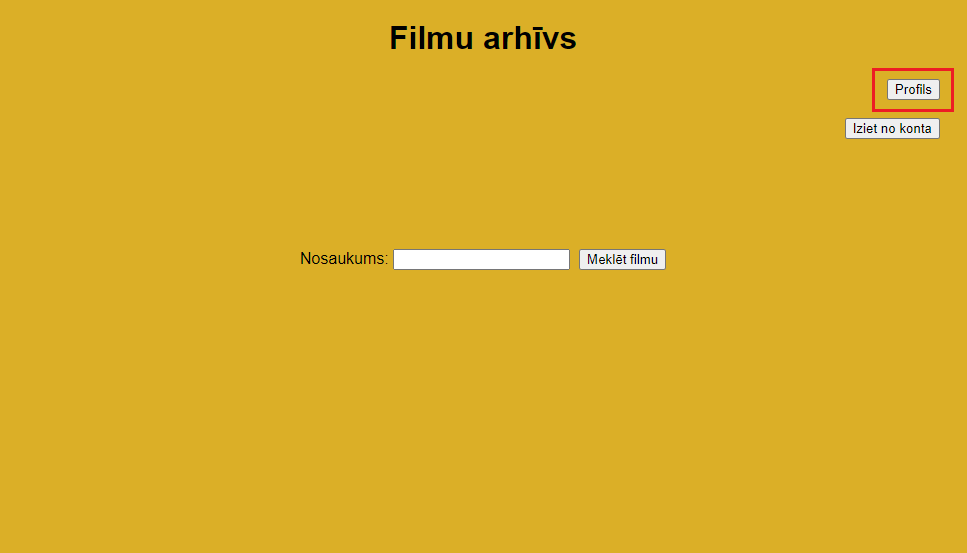
* + 1. Filmu meklēšana – Lapas galvenā funkcija, kas pieejama jebkuram tipa lietotājam. Lai atrastu informāciju par kādu filmu vai televīzijas šovu, ir jāuzklikšķina meklēšanas logs, kas apvilkts zaļā krāsa uz redzamās sākuma lapas parauga, tad jāievada savs vēlētais filmas nosaukums un jāklikšķina uz pogas “meklēt filmu”, kas apvilkta sarkanā krāsā. Pēc pogas uzspiešanas ir redzama rezultātu lapa ar visu pieejamo informāciju.

1.attels

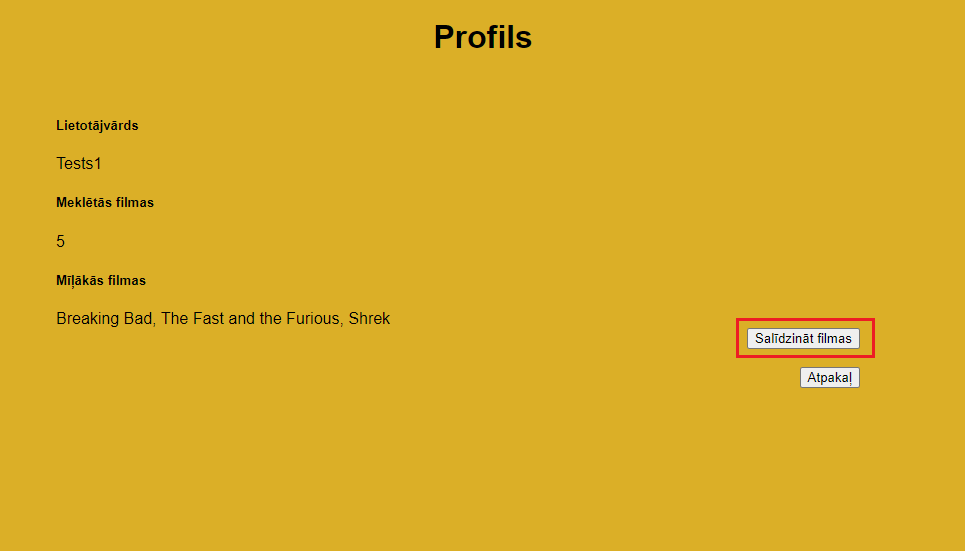
* + 1. Reģistrēšanās kontam – Lai piekļūtu visām programmas funkcijām, ir nepieciešams konts. Kontu var izveidot uzklikšķinot uz pogas “reģistrēties” (apvilkts zilā 1. attēls). Piespiežot pogu, klients nonāk reģistrēšanās logā. Šajā logā ir redzami divi logi, kur jāievada lietotājvārds un parole. Lietotājvārdam un parolei nav nekādu ierobežojumu. Pēc datu ievades, klientam jānospiež poga “reģistrēties”. Reģistrējoties, klients tiks aizsūtīts atpakaļ uz sākuma lapu.

2.attels

* + 1. Pieslēgšanās kontam – Kad konts ir reģistrēts, klients tam var pieslēgties spiežot pogu ”pieslēgties” (apvilkts brūnā 1. Attēls). Piespiežot pogu, klients nonāk pieslēgšanās logā un tāpat kā 2. Solī, ievada datus, kurus izmantoja reģistrējot savu kontu. Pēc veiksmīgas datus ievades, klients nonāk lietotāja sākuma lapā.
    2. Profila pārskatīšana – Klients var pārskatīt savus konta datus, kad uzspiež uz pogas “profils” lietotāja sākuma lapā. Uzspiežot pogu, lietotājs tiek novirzīts uz profila pārskatu, kur var apskatīt savus profila datus.

3.attels

* + 1. Filmu salīdzināšana – Atrodoties profila pārskata lapā, spiežot pogu “salīdzināt filmas”, klients tiks novirzīts uz filmu salīdzināšanas datu ievades lapu. Pēc brīvas izvēles, ievadot pirmās filmas un otrās filmas nosaukumu un piespiežot pogu “salīdzināt”, klients tiek novirzīts uz salīdzināšanas rezultātu logu, kur var aplūkot abu filmu informāciju blakus vienu otrai.

4.attels

5.attēls

* + 1. Iziešana no konta – Lai izzietu no konta, klientam ir jābut ielogotam savā kontā. To var izdarīt dodoties uz lietotāja sākuma lapu, jā šobrīd klients neatrodās sākuma lapā, tad viņš var izmantot jebkuru no pieejamajām pogām “atpakaļ” visos programmas logos, lai nokļūtu atpakaļ lietotāja sākuma lapā. Kad klients ir sākuma lapā, jānospiež poga “Iziet no konta”. Nospiežot pogu, klients tiks izmests no sava konta un novirzīts uz viesu sākuma lapu.

6.attēls

**Licence:**

**Free-software license**

Programmas veidošanā tika izmantots bezmaksas public API, kas nedod man tiesības izmantot šo programmu komerciāliem nolūkiem, kā arī pats personīgi vēlos šo programmu izplatīt brīvā, bezmaksas veidā.

**Programmas kods:**

Programmas kods atrodams: <https://replit.com/@jogurclv/ProjektsFeb>

Github repozitorija: https://github.com/jogurc/ProjektsFeb