



Univerzitet u Sarajevu  
Elektrotehnički fakultet u Sarajevu  
Odsjek za računarstvo i informatiku



Izvještaj o projektu iz predmeta Razvoj programskih rješenja  
Aplikacija za kupovinu kino karata

**Student:** Alen Mrdović

**Broj indeksa:** 18629

**Predmetni nastavnik:** Vanr. Prof. Dr. Vedran  
Ljubović

# Sadržaj

<b>1</b>	<b>Organizacija aplikacije</b>	<b>2</b>
<b>2</b>	<b>Funkcionalnosti vezane za prijavu</b>	<b>3</b>
2.1	Registracija . . . . .	3
2.2	Prijava . . . . .	4
<b>3</b>	<b>Funkcionalnosti administratora</b>	<b>6</b>
3.1	Pregled korisnika . . . . .	6
3.2	Brisanje korisnika . . . . .	6
3.3	Pregled filmova . . . . .	6
3.4	Dodavanje filmova . . . . .	7
<b>4</b>	<b>Funkcionalnosti običnih korisnika</b>	<b>8</b>
4.1	Pregled filmova . . . . .	8
4.2	Detalji o filmu . . . . .	8
4.3	Odabir projekcije . . . . .	8
4.4	Finalizacija rezervacije . . . . .	9

# Poglavlje 1

## Organizacija aplikacije

- Baza podataka koja je korištena je SQLite, odnosno koristio se DB Browser for SQLite. Za dohvaćanje podataka iz baze kreirana je DAO klasa CinemaDAO, kako bi se uspio više ispoštovati MVC+DAL pattern.
- Obzirom da baza podataka za ovaj projekat nije read only, .db fajl čuva se u RPRprojekat folderu (umjesto u resources), kao i dump baze podataka.
- U nastojanju da se što više ispoštuje MVC+DAL pattern, projekat je uređen na sljedeći način:
  - U folderu resources nalaze se folderi css, fxml i images. U trenutku slanja projekta korišten je fxml folder. Obzirom na veliki broj .fxml fajlova, isti su grupisani po namjeni, najprije po tome da li su vezani za administratora ili običnog korisnika, a zatim unutar tih foldera fajlovi vezani za rezervaciju karata, pregled filmova itd.
  - U src folderu, odnosno u RPRMovieApp paketu, nalaze se podpaketi:
    - \* beans - sadrži podatkovne klase
    - \* controllers - sadrži kontrolere, koji su dalje podijeljeni u grupe na isti način kao i fxml datoteke
    - \* DAO - sadrži DAO klasu CinemaDAO

U ovom izvještaju su naglašene sve stvari koje su se koristile, osim onih koje su se intenzivno koristile na laboratorijskim vježbama (neke standardne kontrole poput TextField, Label,...). Generalno što se tiče kolekcija, korišten je ArrayList, a samim tim i ObservableList u kontrolerskim klasama. Aplikacija nije potpuno implementirana, te su ovdje obrazložene samo stvari koje su se implementirale.

## Poglavlje 2

# Funkcionalnosti vezane za prijavu

### 2.1 Registracija

Klikom na dugme Sign Up u početnom prozoru korisnik započinje registraciju. Korisnik se registruje na sistem tako što unosi sljedeće podatke:

- Email - tekstualno polje
- Korisničko ime - tekstualno polje
- Spol - **Radio buttons**. Generalna napomena za sve grupe srodnih dugmadi je da je korišten i **Button Bar**
- Datum rođenja - tri padajuća menija preko kojih bira dan, mjesec i godinu rođenja
- Šifru
- Ponovni unos šifre

Sva polja su obavezna.

Postoje i izvjesna ograničenja na podatke:

- Korisnik mora unijeti email koji je validnog formata.
- Pored toga što email mora biti validnog formata, korisnik ne može unijeti email nekog drugog korisnika kao validan podatak. Ovo se provjerava preko odgovarajućih metoda DAO klase, obzirom da se podaci o postojećem korisniku dobavljaju iz baze.

- Slično kao za email, korisnik mora napisati korisničko ime koje nije prazno i koje već neko nije ranije zauzeo
- Korisnik mora odabrati tačno jedan spol, a onemogućeno je da se odaberu dva
- Šifra korisnika mora sadržati minimalno 6 karaktera bez drugih ograničenja (što je nesigurno, jer šifra može biti "aaaaaa", pa bi jedna nadogradnja za ovaj projekat bila da se uvedu mnogo restriktivnija ograničenja)
- Prema zahtjevu iz intervjuja, korisnik mora imati minimalno 15 godina da bi bio registrovan

Funkcionalnosti korištene u ovom dijelu su:

- Provjera validnosti formata emaila omogućena je preko klase **EMailValidator**, koja je downloadovana sa stranice commons.apache.org. Da bi se klasa EMail validator mogla importovati u projekat, bilo je potrebno da se ide na File > ProjectStructure > Modules > Dependencies, a zatim klikom na + ikonu se dodaje dependency, koji je u projekat ovdje stavljen kao **.jar** datoteka. Da bi se dobila ova .jar datoteka, mora se downloadovati **.bin** folder sa stranice.
- Datum rođenja se zadaje preko tri **choice boxa**, koji su po defaultu namješteni na 1.1.1900. Odgovarajućim metodama, koje su realizovane u sklopu **setOnAction** metoda za odabir mjeseca, osigurano je da se lijevi choice box popuni danima koji odgovaraju odabranom mjesecu i godini (korisnik ne može izabrati datum rođenja, na primjer, 31.4.2002.). Slično je urađeno i za godinu iz razloga što se mora provjeriti da li je godina prijestupna. Razlog iz kojeg nije odabran **Date Picker** je zbog opsega godina koje se mogu odabrati, jer bi korisnik morao veliki broj puta klikati za godinu nazad da odabere svoju godinu rođenja. U **setOnAction** metodama za mjesec i godinu korištene su **lambda funkcije**. Kasnije u aplikaciji za sličnu namjenu je korištena i anonimna klasa.

Po uspješnoj registraciji aktivira se Alert prozor tipa Information. Ovo je jedno od mjesta gdje je korišten **Alert** umjesto custom .fxml datoteke iz razloga što se korisniku prikazuje samo poruka o uspješnoj registraciji.

## 2.2 Prijava

Klikom na dugme Log In u početnom prozoru korisnik započinje prijavu. Korisnik unosi u odgovarajuća polja svoje korisničko ime i lozinku. Ako se desi bilo koja od sljedećih situacija:

- Korisnik unese nepostojeći username
- Korisnik ne unese ništa
- Korisnik unese postojeći username, ali pogrešnu lozinku

Aplikacija javlja grešku i traži korisnika da ponovo unese podatke. Provjera lozinke obavlja se koristeći metode iz DAO klase. **admin** je posebni korisnik koji ima drukčije privilegije i izgled aplikacije je generalno drukčiji. Osoba se ne može registrovati kao **admin**. Konkretno u ovom projektu, admin je ručno dodan u bazu.

Ipak, neovisno od toga ko se prijavio, korisnik će dobiti na početnom ekranu poruku "Welcome, <ime korisnika>". Za početni ekran korišten je **Border-Pane**, za razliku od **GridPanea** koji je korišten u većini ostalih dijelova aplikacije.

## Poglavlje 3

# Funkcionalnosti administratora

Admin na svom početnom ekranu može vidjeti, u lijevom dijelu, dugmad View Users i View films.

### 3.1 Pregled korisnika

Klikom na dugme View Users administrator ima pregled svih korisnika. Za pregled korisnika korišten je **List View**, a elementi su korisnička imena istih. Klikom na odgovarajući element admin može vidjeti druge podatke o korisniku (što isključuje lozinku iz sigurnosnih razloga), tako što se otvori novi prozor. Ovo je realizovano preko **setOnMouseClickedMetode**, gdje je korištena **anonimna klasa**. Ispod **TextArea** gdje pišu podaci o korisniku, admin može vidjeti dugmad Go Back i Delete User.

### 3.2 Brisanje korisnika

Ova funkcionalnost postiže se klikom na Delete User. Adminu najprije iskače poruka u kojoj se provjerava da li zaista želi obrisati tog korisnika. Potvrdom o brisanju korisnik se briše.

### 3.3 Pregled filmova

Slično kao i za korisnike, administrator može vidjeti i trenutne filmove. Klikom na View films najprije se pojavljuju opcije Find films i Add new film. Velikim dijelom je implementirana funkcionalnost za Add new film.

## 3.4 Dodavanje filmova

Na prozoru za dodavanje filmova dodaje se sljedeće:

- Naslov filma - nema ograničenja osim da ne bude prazan
- Režiser - pri čemu postoje dvije mogućnosti:
  - Odabir postojećeg režisera iz choice box
  - Dodavanje novog režisera pri čemu se navodi ime
- Žanrovi - pri čemu se u **choiceBox** nude razni žanrovi. Klikom na žanr isti se dodaje ispod choice boxa ali nestaje kao opcija u choice boxu. Žanrovi se redaju abecedno kako se dodaju. Mora se odabrati barem jedan
- Jezici - potpuno ista logika kao za žanrove
- Trajanje - preko **Spinnera** admin može uvećavati trajanje filma s korakom od jedne minute
- Datum izlaska - ovdje je odabran **Date Picker**, jer je razumno pretpostaviti da će film izaći trenutne ili naredne godine
- Sinopsis - kratak opis filma

Sva polja su obavezna, što se indicira porukom u slučaju da se nešto ne unese. Ako je unos ipak uspješan, pojavljuje se **Alert** tipa **Information**, slično kao za uspješnu registraciju.



## Poglavlje 4

# Funkcionalnosti običnih korisnika

Korisnik na pozdravnom ekranu s lijeve strane može vidjeti dugmad: Now in cinemas, Upcoming movies, Check reservations i My CinemaCard. Funkcionalnost Now in Cinemas nije implementirana do nivoa da može dohvatati najnovije filmove, ali se na primjeru filmova iz baze mogu objasniti neke osnovne funkcionalnosti.

### 4.1 Pregled filmova

Klikom na Now in Cinemas korisnik može vidjeti filmove unutar **ListView**. Klikom na odgovarajući element **ListView** (što je realizovano preko `setOnClickListener`) otvaraju se detalji o filmu

### 4.2 Detalji o filmu

Što se tiče detalja o filmu, najprije se prikazuju detalji koje je korisnik vidio na stranici prije. Ispod toga se nalaze podaci o svim projekcijama za taj film. Na ovom mjestu korišten je **TableView**. Klikom na neki red **TableView** korisnik može započeti rezervaciju. Ovo je realizovano koristeći **selectedItemProperty** za tabelu, gdje bi se kao odabrani element posmatrao red koji odgovara projekciji

### 4.3 Odabir projekcije

Funkcionalnosti implementirane za projekciju su odabir sjedišta i završetak rezervacije. Klikom na Check Seats korisnik može vidjeti slobodna mjesta (označena zelenom bojom). Mjesta koja su zauzeta označena su crvenom

bojom, a mjesta koja je korisnik uzeo plavom bojom. Korisnik klikom na "plavo" sjedište oslobađa isto. Ovo je realizovano postavljajući **stil sjedišta** (koje je realizovano preko **Button**) **iz koda**. Također, obzirom na broj i način implementacije sjedišta, **onAction** je stavljen **u kodu**, a ne u .fxml datoteci, preko **setOnAction**. Korisnik ima pravo da bira sjedište koliko god puta želi (uz prethodno zapamćeni izbor), sve dok ne finalizira rezervaciju, što se indicira **Alert (Information)** prozorom kada se napusti prozor za odabir sjedišta.

## 4.4 Finalizacija rezervacije

Korisniku se, kada odabere sjedišta i klikne za nastavak rezervacije, nudi da odabere način plaćanja. Iako konkretne akcije vezane za isti nisu implementirane, korisnik mora odabrati jednu opciju prije završetka rezervacije. Nakon toga korisniku se javlja poruka o uspješnoj rezervaciji (**Alert**).