A Ilica 242, HR-10000 Zagreb
T (01) 2222 182
E student@algebra.hr
www.algebra.hr

ALGEBRA

PROJEKTNI ZADATAK (Java 1)

Kreirajte aplikaciju koja omogućava ažuriranje podataka o povezanim entitetima, prema želji. U nastavku će za entitete biti odabran primjer **Film** (u nastavku **Movie)** entiteta i pripadajućih entiteta **Glumac (**u nastavku **Actor)**, **Redatelj (**u nastavku **Director)** i slično, što slobodno možete uzeti kao vlastiti.

Prilikom izgradnje aplikacije važno je poštovati najbolje principe objektno orijentirane paradigme i koristeći biblioteke (**Class Library**), prema sljedećem naputku:

- Aplikacija podatke sprema u **Microsoft SQLServer** bazu podataka
- Potrebno je napraviti inicijalizacijsku **DDL** skriptu za kreiranje svih tablica koje će biti korištene u aplikaciji
- Potrebno je napraviti skriptu za brisanje svih podataka iz tablica
- Sav rad sa bazom podataka odvija se pozivanjem procedura iz Java koda, korištenjem **Repository** obrasca
- Aplikacija sadrži više vrsta korisnika, pa je nužno osigurati 2 role - **Administrator** i **Korisnik**(u nastavku **User**)
- **Administrator** mora biti najmanje jedan korisnik sa pripadajućim **korisničko ime/lozinka** (u nastavku **username/password**) parom koji će biti kreiran pomoću procedure i *okinut* nakon inicijalizacijske skripte
- Ulaz u aplikaciju je restriktiran **Login** formom na kojoj je moguća i jednostavna registracija korisnika role **User** (**username/password**)
- Aplikaciju u ispravno stanje za korištenje od strane **User**-a postavlja **Administrator**
    - Prilikom ulaska u aplikaciju, administrator ima mogućnost brisanja svih podataka iz baze (čime se brišu i sve slike sa podatkovnog sustava) i uploadanja novih podataka u bazu, pozivom **RSS čitača (u nastavku RSS Parser)** - spomenuto predstavlja jednostavno i cjelokupno administratorsko sučelje

A Ilica 242, HR-10000 Zagreb
T (01) 2222 182
E student@algebra.hr
www.algebra.hr

ALGEBRA

- **RSS parser** je komponenta aplikacije koja sa određenog **URL**-a parsira sve podatke iz **XML**-a i sprema ih u bazu podataka za naknadno korištenje
    - Primjer: **https://www.blitz-cinestar.hr/rss.aspx?najava=1**
    - Slike je potrebno *download*-ati u lokalni direktorij (primjer: **assets**) te u bazu podataka spremiti njihove relativne putanje

- Nakon ulaska u aplikaciju, **User**-u je predstavljena forma za pregled i promjenu entiteta (**CRUD** operacije), u ovome primjeru **Movie**

- Forme unutar aplikacije moraju biti kvalitetno organizirane (primjer: **JTabbedPane**, prilikom čega svaki **JPanel** predstavlja vlastita klasa prema *principu jedne odgovornosti* - **Single-responsibility principle**)

- Potrebno je kreirati dodatne entitete (primjer: **Actor**, **Director**) koji također imaju **CRUD** forme za njihovo ažuriranje

- Postojeći entitet potrebno je ažurirati na način da se može povezati sa novim entitetima (primjer: **Movie** može biti povezan sa više **Actor**, **Director** entiteta)

- Prikom promjene entiteta ili njegovog brisanja, potrebno je osigurati dosljednu promjenu / brisanje pripadajućih slika iz direktorija sa slikama (primjer: **assets**)

- Za prikaz liste entiteta potrebno je koristiti u najvećoj mjeri **JTable** i pripadajuće **AbstractTableModel** modele

- Za navigaciju u aplikaciji potrebno je koristiti **JMenu**

- Potrebno je implementirati **Drag and drop** funkcionalnost na vlastito odabranom primjeru u aplikaciji (primjer: dodavanje **Actora** na **Movie**)

- Potrebno je implementirati **XML download** entiteta korištenjem **JAXB** biblioteke