

README

Nume: Nămoianu Petre

Grupa: 325CC

Timp de implementare: ~ 100 de ore(o medie de 5-6 ore pe zi cu pauze de câteva zile în vacanță)

Flow-ul aplicației

Am implementat doar terminalul CLI. După login, utilizatorul va fi prezentat cu 3 variante ale aplicației, în funcție de rolul acestuia pe platformă.

```
Welcome back! Enter your Credentials!
email: susan.smith@example.com
password: R8F!b&e9m3U6
Welcome back user susan_smith_123!
Username: susan_smith_123
User experience: 3
Choose action:
    1) View production details
    2) View actor details
    3) View notifications
    4) Search for actor/movie/series
    5) Add/Delete actor/movie/series to/from favorites
    6) Create/Delete request
    7) Add/Remove a review from a production
    8) Logout
Enter a number: |
```

```
Welcome back! Enter your Credentials!
email: emily.wilson@example.com
password: P@ssw0rd!23
Welcome back user emily_wilson_789!
Username: emily_wilson_789
User experience: 8
Choose action:
    1) View production details
    2) View actor details
    3) View notifications
    4) Search for actor/movie/series
    5) Add/Delete actor/movie/series to/from favorites
    6) Create/Delete request
    7) Add/Remove production/actor to/from system
    8) View and solve requests
    9) Update information about actor/production
    10) Logout
Enter a number:
```

```
Welcome back! Enter your Credentials!
email: steven.jackson@example.com
password: St3v3n!#42J@cks0n
Welcome back user steven_jackson_1819!
Username: steven_jackson_1819
User experience: 0
Choose action:
    1) View production details
    2) View actor details
    3) View notifications
    4) Search for actor/movie/series
    5) Add/Delete actor/movie/series to/from favorites
    6) Add/Delete user
    7) Add/Remove production/actor to/from system
    8) View and solve requests
    9) Update information about actor/production
    10) Logout
Enter a number:
```

Pentru “View production details” și “View actor details” sunt afișate toate componentele acestor liste, punând ulterior fi filtrate pentru producții și ordonați alfabetic pentru actori.

La crearea unui request sunt notificați responsabili respectivi fiecărei cerințe și utilizatorul va primi o notificare când requestul a fost rezolvat. Am folosit Observer pattern pentru aceasta.

La crearea unui nou user se folosește un Factory pattern cu ajutorul căruia programul creează automat tipul de utilizator corespunzător, iar câmpul userInformation al fiecărui utilizator este construit cu ajutorul unui Builder pattern. Parola și username-ul sunt generate automat după introducerea numelui și prenumelui noii persoane.

Requesturile pot fi create doar de userii Regular și Contributor, iar Adminii le pot doar rezolva. Clasa internă RequestHolder din Admin ajută la obținerea unei singure liste comune de requesturi pentru toți Adminii din sistem.

Am folosit un Singleton pattern pentru a mă asigura că există doar o singură instanță a aplicației.

Pentru actualizarea experienței am folosit un Strategy pattern cu ajutorul căruia:

- cresc experiența cu 1 pentru un request
- cresc experiența cu 2 pentru un review
- cresc experiența cu 3 pentru o contribuție

Biblioteci suplimentare:

Json-simple pentru citirea mai ușoară din fișierele .json.

Odată ce utilizatorul alege ultima opțiune, și anume logout, va fi prezentat cu următoarea alegere:

```
1) Start app  
2) Exit  
Enter a number:
```

Aceasta poate porni din nou pentru a se loga alt utilizator sau, dacă este ales Exit, ea se va închide.

Pentru fiecare input care necesită un număr m-am asigurat că nu se pot introduce caractere sau numere nepotrivite în situația respectivă, iar la cuvinte am verificat ca ele să respecte condiția cerută(să fie filme, actori, seriale).