

*Datum: 05.03.2025.*

## Upute za predaju seminarskog rada

Prije predaje seminarskog rada na pregled potrebno je uraditi sljedeće korake:

- ✓ Postaviti dokument sa opisom sistema preporuke na git repozitorij seminarskog rada.
  - Opis implementacije recommender sistema.
  - Putanju i printscreen source code-a glavne logike recommender sistema.
  - Putanju i printscreen iz pokrenute aplikacije gdje se prikazuju preporuke.
- ✓ Napraviti build mobilne android i windows aplikaciju, te build-ane fajlove postaviti na git repozitorij seminarskog rada (Upute su u nastavku.)
- ✓ Postaviti link git repozitorija na DL sistem (sekcija zadaci). Na gitu treba biti kompletan source code, uključujući FE koji je buildan u prethodnom koraku.
- ✓ Preporuka da se ne koriste SSL tj. https za pristup API-u, jer self signed certifikati mogu isteći ili biti nevalidni.
- ✓ Na DL sistem ne postavljati konfiguracijske fajlove, npr. fajlovi sa ekstenzijom .env, jer oni trebaju biti sastavni dio git repozitorija.
- ✓ Ako GitHub dozvoljava, konfiguracijski fajl (npr. .env) može ostati u izvornom obliku. U suprotnom, potrebno ga je zamijeniti sa zipovaon verzijom (koristiti šifru „fit“). Zamjenski fajl se treba nalaziti u istom folderu kao originalni fajl. \*\*

### Upute za Android aplikaciju

- ✓ Adresa za Web API treba biti **10.0.2.2**, a radi se o standardnoj adresi za Google Android Emulator AVD.
- ✓ Očistiti stare build fajlove

```
flutter clean
```

- ✓ Uradite build mobilne aplikacije

```
flutter build apk --release
```

- ✓ Prethodni korak bi trebao generisati instalacijski .apk fajl na putanji

```
folder-mobilne-app/build/app/outputs/flutter-apk/app-release.apk
```

- ✓ Provjeriti ispravnost rada android aplikacije u AVD prateći sljedeće korake:
  - Obrišite staru verziju aplikacije u AVD
  - Prevucite .apk fajl u AVD kako biste instalirati aplikaciju
  - Pokrenite aplikaciju nakon instalacije i provjerite ispravnost rada.

### Upute za Windows aplikaciju

- ✓ Adresa za Web API treba biti **localhost**
- ✓ Očistiti stare build fajlove

```
flutter clean
```

- ✓ Uradite build windows aplikacije

```
flutter build windows --release
```

- ✓ Prethodni korak bi trebao generisati .exe fajl na putanji

```
folder-desktop-app/build/windows/x64/runner/Release/
```

- ✓ Pokretanjem prethodno generisanog .exe fajla provjeriti ispravnost rada windows aplikacije

### Upute za predaju rada

- ✓ Kreirane foldere

```
folder-mobilne-app/build/app/outputs/flutter-apk
```

```
folder-desktop-app/build/windows/x64/runner/Release/
```

zip-ovati šifrom "fit" u arhivu "fit-build-20gg-mm-dd.zip" pri čemu je:

- dd - dan build-a npr. 12.
- mm - mjesec build-a npr. 06.
- gg - godina build-a npr. 25

Zbog ograničenja GitHub-a na rad sa fajlovima veličine do 100MB, prilikom zip-ovanja je potrebno odabrati opciju `split`, te kreirati arhivu fajlova od npr. 90 MB.

U slučaju `split` arhive, obično se kreiraju dodatni fajlovi sa ekstenzijom ".z01", ".z02" itd., te je i te fajlove također potrebno commit-ovati \*\*.

**\*\*** U slučaju da ste već napravili lokalni `git commit`, a operacija `git push`, zbog github ograničenja (veličine fajlova, osjetljivi konfiguracijski podaci – `env` i sl.) nije uspjela, koristite opciju `git amend` da bi modifikovali zadnji commit, a zatim ponovo opciju `git push`.

### Sistem preporuke

Prilikom predaje projekta, također je potrebno commit-ovati i dokument koji opisuje rad sistema preporuke (dokument imenovati „recommender-dokumentacija.pdf“).

