

## Zadaća 2 - Razvoj web aplikacije TV serije

**Poslužitelj:** spider.foi.hr

**Korijenska mapa zadatka:** /var/www/RWA/2023/zadaca\_02/{LDAP\_korisnicko\_ime}/

**Pristup preko web preglednika:** http://spider.foi.hr:{vaš\_port}

**Portovi:** /var/www/RWA/2023/portovi.js

**Vanjski Servis “The movie data base” (TMDB):** <https://developers.themoviedb.org/>

**Način predaje:** Rješenje postaviti na moodle u radionicu u **zip** obliku do dogovorenog roka. Radi jednostavnosti ova zadata se testira lokalno. Zip sadrži kada se raspakira sljedeću osnovnu strukturu:

- {LDAP\_korisnicko\_ime}
  - server
    - **aplikacija** - direktorij sadrži REST servis prema opisu zadatka 1 i ostale pozadinske stvari koje poziva angular aplikacija
    - **angular** - direktorij sadrži kompajlirane angular datoteke koje poslužuje web aplikacija odmah na korijenskoj putanji /
    - **dokumentacija** - direktorij sadrži html stranice dokumentacije koje poslužuje web aplikacija na putanji /**dokumentacija** - moguće je dokumentaciju ugraditi u samu angular aplikaciju.
    - **konfiguracija.csv** - konfiguracijska datoteka koju koriste servis i aplikacija kod pokretanja
    - **RWA2023{LDAP\_korime}.sqlite** - sama SQLite baza podataka sa podacima
    - **package.json** - npm konfiguracija - koristi se za pokretanje servera sa npm start
  - **angular** - sadrži originalni kod klijentske aplikacije - koji se koristi za ručno kompajliranje i bodovanje te se može koristiti za pokretanje live servera sa *ng serve*
    - **.angular** - ne treba ga predavati, ali ako postoji mora biti prazan
    - **.vscode** - ne treba ga predavati, ali ako postoji mora biti prazan
    - **node\_modules** - ne treba ga predavati, ali ako postoji mora biti prazan
    - **projekti** - sadrži isključivo angular projekt zadatka 2, a ne druge angular projekte
    - **angular.json** - konfiguracija mora sadržavati samo jedan projekt zadatka 2
    - **tsconfig.json** - konfiguracija za TypeScript
    - **package.json** - konfiguracija za npm
    - **package-lock.json** - obrisati

**Struktura zadatka:** U direktoriju smiju biti **samo datoteke i poddirektoriji vezane uz zadatak**, sve ostalo treba ukloniti (npr. skripte s vježbi). **Svugdje u zadatku treba koristiti relativne putanje.** Ako u tekstu zadatka piše naziv/struktura neke datoteke ili direktorija tada se treba držati tog naziva/strukture u ostalim slučajevima naziv/struktura je proizvoljna. **Multimedijalne datoteke postavljene na server moraju poštivati maksimalne veličine 500KB slike i 1MB video!** Ako **se ne pridržava zadane strukture direktorija, nazivlja datoteka i veličina datoteka** student se kažnjava s **oduzimanjem do 50% od**

**ostvarenih bodova.** Poslužitelj **spider.foi.hr** prepoznaje velika i mala slova (eng. case sensitive) kod naziva direktorija, naziva datoteka i ekstenzija.

**Anketa:** Potrebno je ispuniti anketu vezanu uz zadaću na Moodle sustavu. Ako **se ne ispuni anketa ili se ne ispuni do kraja ili sadrži pogrešne informacije** oduzima se **do 50% od ostvarenih bodova**.

**Napomena:** Svi negativni bodovi se akumuliraju na ostale negativne bodove. Na primjer ne predana anketa, pogrešno definirana prava i pogrešna struktura mogu rezultirati s više od 50% negativnih bodova.

**Obrane zadaće:** Na kratkoj obrani na vježbama otvara se kod zadaće i objašnjava se što radi neki dio programskog koda koji nastavnih odredi. **Treba poznavati svaku liniju napisanog koda u zadaći u kontekstu programskog jezika u kojem je napisan, ali i logički koji je smisao koda u kontekstu zadaće.** **Ako student/ica ne da odgovor na većinu postavljenih pitanja smatra se da nije autor zadaće što znači 0 bodova iz zadaće, a time i potpis prema modelu praćenja.** Ako postoji sumnja o autorstvu zadaće, nastavnik zadržava pravo pozvati studenta/icu na konzultacije na dulju obranu zadaće u posebnom terminu konzultacija.

**Pomoć i pitanja:** Sva pitanja u vezi zadaće postavljaju se u **poseban forum za zadaću na sustavu Moodle**. Nema privatnih konzultacija u vezi zadaće osim ako nastavnik na vaše pitanje u forumu ne kaže da dodete na konzultacije zbog specifičnosti problem. Preporučujemo da ne šaljete kod drugima, ne pokazujete kod drugima i ne radite zadaću zajedno iako ste cimeri ili se poznajete već 10 godina. **Zabranjeno je prepisivanje** zadaća; ako se utvrdi da je zadaća prepisana od drugog/ih studenta **svako dobiva 0 bodova**.

**Bodovanje:** Točna skala bodova nije unaprijed poznata i određuje se tek nakon što završi predaja zadaće. Boduju se dijelovi koji su rađeni samostalno nakon vježbi. U bodovanje, uz činjenicu radi li aplikacija prema specifikaciji ulazi: kvaliteta ispunjenosti ankete, kvaliteta programskog koda, model baze podataka, dizajn aplikacije, HTML/CSS validacija, količina grešaka u implementaciji, kvaliteta provjere unosa podataka od strane korisnika (sigurnost) i sl. **Dijelovi označeni ovakvom plavom bojom su za dodatne bodove što znači ako se ne naprave, može se ostvariti maksimalni broj bodova iz zadaće.**

**Validacija:** Sve stranice kada se učitaju moraju ispravno prikazivati hrvatske znakove.

**Dizajn:** Dizajn stranice je za sada proizvoljan, važno je da se stranica može normalno koristiti u preglednicima Firefox i Chrome/Chromium. U ovoj zadaći nije potrebno raditi prilagodbe za mobilne uređaje i tablete, već isključivo za pregled na računalu.

**Okviri/moduli/biblioteke:** U zadaći na strani klijenta i strani poslužitelja se smiju koristiti samo okviri/moduli/biblioteke obrađeni na nastavi. Konkretno za NodeJs **kod izvršavanja** mogu se koristiti oni moduli koji su instalirani na poslužitelju globalno i **ne smiju se instalirati dodatni moduli na razini klijenta**. **Iznimka modul za kreiranje QR koda ako postoji i ako se radi dio za dodatne bodove.** Popis dostupnih modula dobiva se sa: **npm -g list**

**JavaScript klijentska strana:** Koristi se isključivo Angular okvir.

**Baza podataka:** Koristi se SQLite baza koja mora biti pripremljena i predana za jedno sa projektom. Baza je ista kao u zadaći jedan i dozvoljena je modifikacija baze u odnosu na zadaću 1 ako treba.

**Čisti kod:** U zadaći se treba voditi principima čistog koda što ukratko znači sljedeće. Kod organizirati u funkcije, klase i module. Moduli sadrže klase, a klase sadrže funkcije koje čine smislenu cjelinu. Funkcije ne bi trebale imati više od 35 linija programskog koda programskog koda, u što se ne broji definiranje metode, njenih argumenata i lokalnih varijabli, prazna linija, linija samo s { ili }. U jednoj liniji nalazi se jedna instrukcija. Linija nema više od 120 znaka. Radi se uvlačenje koda s 4 znaka space ili 1 tab. Nazivi klasa/funkcija/varijabli su smisleni i oslikavaju svrhu klase/funkcije/varijable. **Nazivi datoteka, direktorija, klasa, funkcija, varijabli i sl. pišu se na hrvatskom jeziku.** Naravno principi čistog koda koji su spomenuti na OWT za HTML i CSS također treba koristiti.

**Komentari:** U zadaći ne smiju biti komentari koji objašnjavaju kod. Kod koji je zakomentiran jer se ne koristi preporuka je da se obriše, ali nije problem ako negdje slučajno ostane. Zakomentiran kod također treba znati objasniti.

**Datum i vrijeme:** Svi datumski podaci moraju se na ekranu prikazivati i unositi u obliku dan.mjesec.godina za datum i sati:minute:sekunde u 24 satnom obliku za vrijeme. Izbjegavati input element tipa date i tipa time jer oni koriste američki model datuma i vremena.

**Provjere unosa korisnika:** Preporuka je da se sav korisnički unos kontrolira kako ne bi bilo moguće srušiti aplikaciju krivim unosima ili napraviti napad. Preporuka je korištenje regularnih izraza jer nosi više bodova nego varijanta s if-ovima.

**Testiranje:** aplikaciju/servis treba testirati i s krivim podacima, provjeriti rubne slučajeve, a ne samo idealni scenarij korištenja. Kod testiranja servisa predlažemo curl naredbu jer možete lako više puta izvršiti. Također radi lakšeg testiranja bolje je napraviti prvo servisni dio, a tek onda aplikativni dio koji koristi REST.

#### Korisne TMDb adrese:

- Registracija: <https://www.themoviedb.org/signup>
- API ključ: <https://www.themoviedb.org/settings/api>
- Dokumentacija: <https://www.themoviedb.org/settings/api>
- API bazični URL: <https://api.themoviedb.org/3/>
- Traženje serija: <https://developer.themoviedb.org/reference/search-tv>
  - [https://api.themoviedb.org/3/search/tv?query=game&include\\_adult=false&language=en-US&page=1](https://api.themoviedb.org/3/search/tv?query=game&include_adult=false&language=en-US&page=1)
  - [https://api.themoviedb.org/3/search/tv?query=game%20of%20thrones&include\\_adult=false&language=en-US&page=1](https://api.themoviedb.org/3/search/tv?query=game%20of%20thrones&include_adult=false&language=en-US&page=1)
- Svi podaci o seriji: <https://developer.themoviedb.org/reference/tv-series-details>
  - <https://api.themoviedb.org/3/tv/1399>
- Putanje do slika: [https://image.tmdb.org/t/p/original/{poster\\_path}](https://image.tmdb.org/t/p/original/{poster_path})  
<https://image.tmdb.org/t/p/original/jXIMScXE4J4EVHUb1JgxZnWbo4.jpg>

## Zadaci

1. Na stranicu dokumentacija ažurirati tablice iz prve zadaće oko informacija koji dijelovi su napravljeni za servis odnosno aplikaciju. Ako je nešto napravljeno djelomično treba to jasno napisati koji dio je napravljen. **Ako ovog dijela nema bodovi vezani uz web aplikaciju i/ili web servis nose 0 bodova.**
2. Na stranicu dokumentacija kreirati tablicu po koracima (počevši od zadaci općenito) ovog dokumenta u kojoj jasno pišu elementi opisa 2 zadaće koji su napravljeni, koji nisu. **Ako ovog dijela nema zadaća nosi 0 bodova.** Može se napraviti još jedna tablica sa dodatnim opisima ako se želi nešto naglasiti. Nije obavezno.
3. Složite package.json tako da se sa **npm start** može pokrenuti server i isto tako jedan package.json, tsconfig.json i angular.json da se može pokrenuti angular projekt.
4. Kod predavanja zadaće izbrisati svoje ključeve za TMDB servis iz konfiguracijske datoteke svaki student ubacuje svoje ključeve u konfiguracijsku datoteku kod testiranja.
5. Klijentski dio Web Aplikacije HTML, CSS i JS na strani klijenta prebaciti u Angular sa TypeScript-om.
6. Pozadinski dio web aplikacije posluhuje kompajliranu verziju Angular aplikacije. Moguće je pokretanje Angular aplikacije preko live servera sa ng serve, ali ta varijanta nosi manje bodova.
7. Pozadinski dio web aplikacije posluhuje dokumentaciju - moguće je dokumentaciju ugraditi u samu angular aplikaciju.
8. Pozadinski dio web aplikacije nudi podršku u obliku REST servisa. REST servis sadrži sve metode kako je to bilo u prvoj zadaći za rad s bazom podataka. Dodatno svi ostali pozadinski dijelovi prve zadaće (npr. komunikacija sa TMDB servisom) koji nemaju direktne veze s bazom se sada tretiraju kao proširenje REST servisa jer klijentski dio aplikacije sada mora slati sve preko fetch zahtjeva. Angular aplikacija nikad ne komunicira direktno sa TMDB servisom već preko pozadinskog dijela kao u prvoj zadaći.
9. Sada svi pozadinski dijelovi osim inicialnog posluživanja Angular aplikacije i prijave traže validan JWT.
10. Informacije o bazičnoj adresi i portu za pozadinski dio koji koristi Angular aplikacija kod slanja fetch zahtjeva čitaju se iz environment datoteke. Hard kodirana varijanta nosi manje bodova.
11. Potrebno je doraditi dizajn stranice ako u prvoj zadaći niste vodili o tome previše računa. Dizajn treba prebaciti korištenjem Sass preprocesora u Sass ili Scss sintaksi. **Čisti CSS ili bootstrap varijante nose 0 bodova.**
12. Dodati reCaptcha v3 sustav na minimalno 3 obrazaca od kojih su obavezni: prijava, registracija i ažuriranje profila.
13. Potrebno je doraditi ažuriranje profila sa opcijom da se uključi dvorazinska autentifikacija. Prilikom prvog uključivanja dvorazinske autentifikacije potrebno je kreirati tajni ključ koji se sprema u bazu i prikazuje korisniku utekstualnom obliku ili u obliku **QR koda (za dodatne bodove)**. Korisnik može isključiti dvorazinsku autentifikaciju ili ponovo uključiti, ali više nema prikaza tajnog ključa.
14. Potrebno je doraditi prijavu da se autentifikacija radi u dvije razine. Prvo korisničko ime i lozinka te nakon toga TOTP ako je uključena dvorazinska autentifikacija za tog korisnika.

15. Potrebno je doraditi prijavu da omogućuje mogućnost prijave sa OAuth putem GitHub računa. Takav korisnik nema TOTP.
16. Korisnik koji se prijavi preko GitHub računa ne radi ništa s bazom podataka. Kod takvog korisnika spremanje favorita se radi u preglednikov local storage. Jednako tako pregled favorita i favoritDetalji se čitaju iz local storage u pregledniku. Stranica profila nije dostupna.
17. Inicijalno mora postojati korisnik iz svake uloge koji nema TOTP i treba navesti podatke na stranici prijava radi lakšeg testiranja.
18. Za dodatne bodove može se prebaciti cijeli pozadinski dio u TypeScript. Kod pokretanja sa npm start obavezno se mora napraviti recompile TypeScript-a i kopiranje svih bitnih statičnih sadržaja da aplikacija normalno radi.