

## Završni test - JWD 51/52

Korišćenjem radnih okvira Spring Boot, React i Bootstrap razviti Web aplikaciju za prijavu učesnika na olimpijske igre (OI), koju koriste administratori i prijavljeni korisnici.

1.1 Aplikacija bi trebalo da obezbedi rad sa sledećim entitetima, uz čije attribute su data i ograničenja na nivou baze podataka:

Država:

Id - identifikator

Naziv - tekstualna, obavezna, **jedinstvena vrednost** vrednost

Oznaka - tekstualna, **jedinstvena**, troslovna oznaka države (npr. SRB, USA i sl.)

Takmičar:

Id - identifikator

Ime i prezime - tekstualna, obavezna vrednost

Broj osvojenih medalja - numerička vrednost

Datum rođenja - obavezna vrednost

Država - veza sa instancom klase Država (svaki takmičar se takmiči za tačno jednu državu, dok se za jednu državu takmiči više od jednog takmičara)

Prijava:

Id - identifikator

Datum prijave - obavezna vrednost

Disciplina - ime discipline u kojoj se takmiči (npr. košarka, skok u dalj, tenis i sl.)

Takmičar - veza sa instancom klase Takmičar (jedna prijava se vezuje za jednog takmičara, dok se jedan takmičar može prijaviti više puta npr. atletičar se prijavljuje za sprint na 100 i 200m ili teniser za nastupanje u singl i dubl mečevima)

1.2 Pomoću radnog okvira Spring Boot implementirati sledeći REST API:

|   |                    |
|---|--------------------|
| GET /api/drzave - preuzimanje svih država                     | <b>admin, user</b> |
| GET /api/takmicari - preuzimanje svih takmičara (paginirano)  | <b>admin, user</b> |
| GET /api/takmicari/{id} - preuzimanje takmičara po zadatom id | <b>admin, user</b> |
| POST /api/takmicari - dodavanje novog takmičara               | <b>admin</b>       |
| PUT /api/takmicari/{id} - izmena postojećeg takmičara         | <b>admin</b>       |
| DELETE /api/takmicari/{id} - brisanje postojećeg takmičara    | <b>admin</b>       |

1.3 Na nivou API-ja validirati sledeće stavke:


Broj osvojenih medalja - numerička vrednost koja ne sme biti negativna

Ime i prezime takmičara - tekstualna vrednost koja ne sme imati preko 60 karaktera

2.1 Obezbediti unos novog takmičara na **zasebnoj stranici** (slika 1). Na stranici za prikaz i pretragu takmičara postoji dugme za prelazak na stranicu za unos (slika 2). Nakon unosa vrednosti u polja prikazana na slici 1 i klika na dugme 'Dodaj takmičara', on se preko API-ja dodaje u aplikaciju i potom se korisnik vraća na stranicu za prikaz i pretragu. **Unos datuma možete implementirati i tako da se isti unese u vidu teksta.**


Ime i prezime takmicara

Datum rođenja

Broj osvojenih medalja

Država takmicara

**Dodaj takmicara**

**Slika 1.**

2.2 Obezbediti brisanje takmičara. **Nakon klika na dugme obriši u tabeli za prikaz svih takmičara, on se briše iz aplikacije i uklanja iz tabele.**

2.3 Obezbediti paginirani prikaz podataka. Dugme 'Prethodna' i dugme 'Sledeća' iznad tabele za prikaz takmičara (slika 2) omogućuju promenu stranice. Ukoliko se korisnik nalazi na prvoj stranici onemogućiti dugme prethodno, a ukoliko se nalazi na poslednjoj onemogućiti dugme sledeće. *Paginacija se vrši na back-end delu aplikacije.*

2.4 Obezbediti pretragu takmičara, preko forme za filtriranje (slika 2). Prilikom pretrage korisnik može da odabere *državu takmičara* (preko opcije za selekciju, *dropdown*) ili unese broj medalja. Broj medalja se može zadati kroz dva polja gde bi korisnik vršio pretragu npr. svih takmičara iz Srbije koji su osvojili na OI između 1 i 10 medalja. Pronalaze se i prikazuju takmičari koji zadovoljavaju sve kriterijume. Ukoliko korisnik neko polje ne unese, vrednost tog polja se ignoriše u pretrazi. Pretraga se vrši na *back-end* delu aplikacije.

Drzava takmicara

Serbia, SRB

Broj medalja od

1

Broj medalja do

10

Pretraga

Dodaj takmicara Statistika Prethodna Sledeca

| Ime i prezime takmicara | Drzava      | Datum rođenja | Broj osvojenih medalja |        |         |
|-------------------------|-------------|---------------|------------------------|--------|---------|
| Novak Djokovic          | Serbia, SRB | 1987 May 22   | 1                      | Obrisi | Prijava |
| Ivana Španović          | Serbia, SRB | 1990 May 10   | 1                      | Obrisi | Prijava |

**Slika 2.**

2.5 Kompletnu izmenu podataka o takmičaru **nije potrebno** realizovati na *front-end* strani, putem popunjene forme. Izmenu implementirati na *back-end* delu aplikacije.

3. Implementirati prijavu takmičara na OI:

→ Klikom na dugme “Prijava” (slika 2.), inicira se prijava takmičara na OI, gde jedino korisnik može istu inicirati (ne i admin). Prijava se vrši na *back-end* strani i kreira se na sledeći način:

- ◆ Korisnik unosi naziv discipline u kojoj se takmiči
- ◆ Datum prijave se generiše na *back-end* delu aplikacije

4. Implementirati prikaz statistike - na klik dugmeta “Statistika” na početnom ekranu, korisniku se prikazuje sumarni izveštaj o broju medalja. Potrebno je za svaku državu prikazati broj osvojenih medalja na olimpijskim igrama sortirano opadajuće po broju osvojenih medalja. Prikazati podatke tabelarno kao što je prikazano u okviru Slike 3.

| Država                            | Broj osvojenih medalja |
|-----------------------------------|------------------------|
| The United States of America, USA | 2523                   |
| Jamaica, JME                      | 78                     |
| Serbia, SRB                       | 15                     |

**Slika 3.**

5. Aplikacija bi trebalo da bude obezbeđena i sa *back-end*, i sa *front-end* strane što znači da samo prijavljeni korisnik može da joj pristupi. Korisnik se prijavljuje na sistem preko *login* stranice. Za ovaj deo, dozvoljeno je koristiti postojeće materijale kao takve. **HTTPS nije potrebno iskonfigurisati. Role koje mogu pristupiti određenim endpoint-ima aplikacije su date kod definicije API-a.**

**Obavezno uz rešenje priložiti i test podatke za bazu podataka.**