

# .NET Web Development 2 - Finalni test

Korišćenjem radnih okvira ASP.NET Web API, Entity Framework i Bootstrap, jQuery biblioteke i Partial View-a realizovati Web aplikaciju za vođenje evidencije o sportskim ligama i sportskim klubovima. Aplikacija treba da obezbedi rad sa sledećim entitetima:

## Sportska Liga:

- Id - identifikator
- Naziv - tekstualna vrednost
- Skraćenica - tekstualna vrednost sa najviše 4 karaktera
- Godina osnivanja - numerička vrednost manja od 2017, a veća od 0.

## Sportski Klub:

- Id - identifikator
- Naziv - tekstualna vrednost
- Mesto - tekstualna vrednost
- Godina osnivanja - numerička vrednost manja od 2016, a veća od 0
- Cena karte - decimalna vrednost veća od 0
- Liga - veza sa instancom klase Sportska Liga (jedan klub može pripadati samo jednoj ligi).

1) Pomoću radnih okvira ASP.NET Web API i Entity Framework implementirati sledeći REST API:

- GET api/lige** - preuzimanje svih liga - **OBAVEZNA FUNKCIONALNOST!!!**
- GET api/lige/{id}** - preuzimanje liga po zadatom id-u
- GET api/lige?godina={vrednost}** - preuzimanje svih liga čija je godina osnivanja nakon prosleđene vrednosti, sortirano po godini opadajuće
- GET api/najstarije** - preuzimanje dve najstarije lige, sortirane po godini rastuće
- GET api/klubovi** - preuzimanje svih klubova, sortiranih po godini osnivanja rastuće - **OBAVEZNA FUNKCIONALNOST!!!**
- GET api/klubovi/{id}** - preuzimanje kluba po zadatom id-u
- POST api/klubovi** - dodavanje novog kluba
- PUT api/klubovi/{id}** - izmena postojećeg kluba
- DELETE api/klubovi/{id}** - brisanje postojećeg kluba
- POST api/pretraga** - preuzimanje svih klubova čija je godina osnivanja između dva unete vrednosti (**start** i **kraj**), sortiranih po godini osnivanja rastuće
- Registracija i prijava korisnika.

2) Jedinično testirati sledeće funkcionalnosti:

- a) Funkcionalnost **1)f)** sa jediničnim testom kada akcija vraća status kod 200 i objekat
- b) Funkcionalnost **1)f)** sa jediničnim testom kada akcija vraća status kod 404
- c) Funkcionalnost **1)g)** sa jediničnim testom kada akcija vraća status kod 201 i objekat
- d) Funkcionalnost **1)i)** sa jediničnim testom kada akcija vraća status kod 200 bez objekta.

3) Pomoću radnog okvira Bootstrap, biblioteke jQuery i Partial View-a realizovati sledeću Single Page Application:

- a) Prilikom prvog pokretanja, stranica sadrži formu za prijavu korisnika i tabelarni prikaz sportskih klubova po funkcionalnosti **1)e)**, kao što je prikazano na slici 1. Za Ligu se prikazuje njena skraćenica. - **OBAVEZNA FUNKCIONALNOST!!!**

[Klubovi](#) [Home](#) [Api](#)

## Prijava

Korisničko ime:

Lozinka:

Prijava

Registracija

## Sportski klubovi

Naziv	Mesto	Liga	Godina
Arsenal	London	EL	1886
CSKA	Moskva	UCL	1911
Javor	Ivanjica	JSL	1912

Slika 1.

- b) Ukoliko korisnik pritisne dugme **Registracija**, umesto forme za prijavu mu se prikazuje forma za registraciju, kao što je prikazano na slici 2. Ukoliko se korisnik uspešno registrovao, umesto forme za registraciju mu se prikazuje

forma za prijavu (slika 1). Ukoliko je došlo do greške prilikom registracije, korisnik će biti obavešten pomoću **alert-a**.

Klubovi Home Api

Registracija

Korisničko ime:

Lozinka:

Ponovite lozinku:

Registruj se

Sportski klubovi

Naziv	Mesto	Liga	Godina
Arsenal	London	EL	1886
CSKA	Moskva	UCL	1911
Javor	Ivanjica	JSL	1912

Slika 2.

- c) Kada se korisnik prijavljuje pritiskom dugmeta **Prijava**, ukoliko je došlo do greške, potrebno je obavestiti korisnika pomoću **alert-a**. Ukoliko se korisnik uspešno prijavi na sistem, prikazaće mu se sadržaj sa slike 3.
- d) Nakon labele **Prijavljeni korisnik** se prikazuje korisničko ime prijavljenog korisnika. Klikom na dugme **Odjava** korisnik se odjavljuje sa sistema i nakon toga mu se prikazuje sadržaj opisan u **3)a)**.
- e) Tabela klubova se proširuje još jednom kolonom sa desne strane, koja pored svakog festivala dodaje dugme **Obriši**, na čiji klik se briše odabrani klub i osvežava prikaz klubova u tabeli. Ovu akciju može da vrši samo prijavljeni korisnik.
- f) Iznad tabele se nalazi forma za pretragu klubova koja prati funkcionalnost **1)j)**. Ovu akciju može da vrši samo prijavljeni korisnik. Klubovi koji ispunjavaju kriterijum pretrage se prikazuju u tabeli klubova i moguće je njihovo brisanje kao što je opisano u funkcionalnosti **3)e)**. Ukoliko je došlo do greške prilikom unosa kriterijuma za pretragu, ili samog dobavljanja klubova sa REST API-ja, korisnika obavestiti putem **alert-a**.
- g) Ispod tabele se nalazi forma za dodavanje novog kluba, koja prati funkcionalnost **1)g)**. Ovu akciju može da vrši samo prijavljeni korisnik. Liga se

bira iz padajućeg menija koji se popunjava pomoću funkcionalnosti **1)a)**. Ovu akciju može da vrši samo prijavljeni korisnik. U padajućem meniju se prikazuje naziv lige. Klikom na dugme **Dodavanje** vrši se dodavanje novog kluba. Ovu akciju može da vrši samo prijavljeni korisnik. Ukoliko je dodavanje bilo uspešno, polja u formi za dodavanje će se očistiti, a u tabeli klubova će se osvežiti prikaz. Ukoliko je došlo do greške prilikom unosa podataka, ili prilikom komunikacije sa REST API-jem, korisnik će se obavestiti putem **alert-a**. Klikom na dugme odustajanje čiste se polja u formi za dodavanje.

[Klubovi](#) [Home](#) [Api](#)

Prijavljeni korisnik: **pera@mail.com** [Odjava](#)

## Sportski klubovi

Pretraga po godini osnivanja

Od  Do  [Pretraga](#)

Naziv	Mesto	Liga	Godina	Akcija
Arsenal	London	EL	1886	[ Obriši ]
CSKA	Moskva	UCL	1911	[ Obriši ]
Javor	Ivanjica	JSL	1912	[ Obriši ]

## Dodavanje novog Sportskog Kluba

Naziv:

Mesto:

Godina osnivanja:

Cena karte:

Liga:

▼

[Dodavanje](#) [Odustajanje](#)

Slika 3.

**Napomene:**

- **Test nosi 100 bodova, za polaganje testa je neophodno osvojiti 50 bodova uz poštovanje uslova za pregled testa.**
- **Uslov za pregled testa je implementiranje funkcionalnosti 1)a), 1)e) i 3)a).**
- **Popuniti bazu podataka sa minimum 3 test podatka za Sportsku Ligu i minimum 3 test podatka za Sportski klub koristeći Code First migracije.**
- **Zabranjena upotreba Scaffolding-a. Sve funkcionalnosti koje su scaffold-ovane, ili na neki način koriste scaffold-ovanu funkcionalnost bodovaće se sa 0 bodova.**
- **Ako ima potrebe, a u skladu sa dobrom programerskom praksom, koristiti AutoMapper, Dependency Injection koncept, OAuth.**
- **Svaka komunikacija sa kolegama je najstrožije zabranjena. Saobraćaj u učionici se snima.**