

Projektni zadatak 2

-januar 2022-

U sklopu stranice predmeta date su serverska (Spring Boot aplikacija koja je dostupna na portu 8080) i klijentska aplikacija (Angular aplikacija koja je dostupna na portu 4200). U osnovi, čitav sistem tj. obe aplikacije su modifikacija vježbe 7. Glavne razlike su u tom što serverskoj aplikaciji nije potrebna baza podataka već ista dostavlja "mock" podatke. Takođe, serverska i klijentska aplikacija imaju određeni broj grešaka koji je potrebno evidentirati. REST endpointi serverske aplikacije nisu mijenjani ali su dodani statusni kodovi tj. 404 NOT FOUND u slučajevima kada određeni student nije pronađen. Bolje rečeno, specifikacija endpointa je ista kao u vježbi 8, osim što se u ovom slučaju vrši CRUD (*eng. Create, Read, Update, Delete*) nad studentom.

(13 bodova) Koristeći Selenium skup tehnologija napisati testove koji će provjeriti prikazane podatke u klijentskoj aplikaciji prilikom osnovnih CRUD operacija. Provjeravaju se samo prikazani podaci o studentima i glavni elementi vezani za akcije oko istih, tako da nije potrebno provjeravati neke druge komponente tj. sadržaj labela, pretragu, sadržaj menija itd... Fokus zadatka je na provjeri sadržaja u formi za dodavanje i izmjenu studenta, tabeli za prikaz svih studenata i dijalogu za brisanje studenta kao i akcijama na dugmad koja su vezana za isto. Dozvoljeno je koristiti Selenium IDE, ali je konačni projekat sa testovima potrebno dostaviti kao Junit testove u sklopu Java projekta. Naziv Java projekta mora biti u formi ime_prezime (ime i prezime studenta koji predaje).

(6 bodova) Koristeći Postman kreirati set testova za provjeru CRUD operacija za endpointe serverske aplikacije. Specifikacija REST endpointa je ista kao u vježbi 8. Potrebno je provjeriti podatke koje dostavlja svaki endpoint kao i statusne kodove. Naziv projekta koji se dostavlja mora biti u formi ime_prezime (ime i prezime studenta koji predaje).

(6 bodova) Napisati izvještaj u sklopu kojeg su opisane pronađene greške u serverskoj i klijentskoj aplikaciji.

Prvi rok za dostavljanje projektnog zadatka je sedam dana prije prvog ispitnog termina za ovaj predmet.

Napomene:

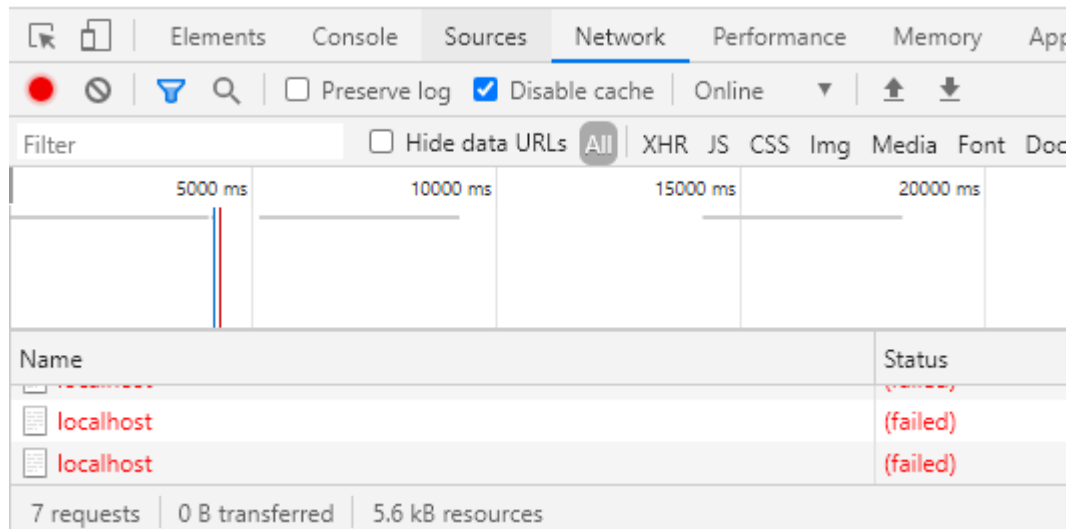
Serversku aplikaciju pokrećemo sa: `java -jar aplikacija.jar`

Klijentsku aplikaciju potrebno je raspakovati u sklopu ROOT foldera veb servera koji je korišten u vježbi 7 i nakon toga pokrenuti server. Prije toga je potrebno obrisati sve što se nalazilo u tom folderu.

Izvještaj sa greškama treba da za svaku grešku sadrži kratak opis od najviše dve rečenice. Primjer je dat u nastavku.

Za praćenje saobraćaja između klijentske i serverske aplikacije može se posmatrati stanje Network taba u sklopu razvojnih alata veb preglednika. Za pristup alatima može se koristiti skraćena na tastaturi

F12. Obavezno čekirati opciju Disable cache ili na neki drugi način očistiti keš (CTRL+F5...), kako biste koristili novu verziju klijentske aplikacije, a ne onu koja je korištena na vježbama i vjerovatno keširana. Primjer prozora (Chrome):



Prilikom pokretanja testova smatrati da prilikom prvog pokretanja na serverskoj strani u evidenciji nema studenata. Za isto je dovoljno ponovno pokretanje serverske aplikacije svaki put kada je potrebno pokrenuti testiranje.

Primjer izvještaja:

Greške u serverskoj aplikaciji

- Opisana prva pronađena greška
- .
- .
- .
- .

Greške u klijentskoj aplikaciji

- Opisana prva pronađena greška
- .
- .
- .
- .

Greške koje postoje u primjeru sa laboratorijskih vježbi nije potrebno evidentirati. Preporuka: Ukrstiti serversku aplikaciju sa laboratorijskih vježbi sa klijentom kojeg ste dobili i obrnuto.

Broj grešaka koje trebate pronaći je nepoznat :D

Atributi klase Student su:

```
private Long studentId;  
private String studentName;  
private String studentEmail;  
private String studentBranch;
```