

Uvod u Veb i Internet Tehnologije (2021/2022.)
Jun1 – grupa 1

Ispit se radi najviše 180 minuta. Maksimalan broj poena je 65.

Jun 10, 2022

Zadaci

1. **(7 poena:)** U datoteci 1.html napisati validan HTML5 kod koji definiše strukturu dokumenta kao na narednoj slici, a zatim u datoteci 1.css napisati validan CSS3 kod kojim se postiže stilizovanje dokumenta kao na narednoj slici.

Pretraga knjiga:

Zanr:

Cena od: do:

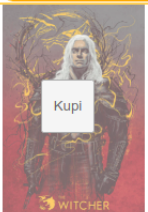


2. **(8 poena:)** U datoteci 2.js dat je niz knjige koji sadrži nazive slika naslovnih strana knjiga. Koristeći programski jezik JavaScript i Web API, omogućiti da se klikom na levo narandžasto dugme (<) prikaže prethodna knjiga iz niza (ako nema prethodne, prikazati poslednju) i klikom na desno narandžasto dugme (>) prikaže sledeća knjiga iz niza (ako nema sledeće prikazati prvu). Kada miš udje u prostor slike, treba da se zamagli slika i da se pojavi dugme Kupi, a kada miš izađe iz prostora slike, sve treba da postane isto kao i ranije.

Pretraga knjiga:

Zanr:

Cena od: do:



< >

Za naredne zadatke je neophodno koristiti podatke koji se nalaze u MongoDB bazi podataka naziva "Knjizara".

3. Dopuniti implementaciju Node.js serverske aplikacije u duhu Model-Pogled-Kontroler arhitekture:
 - (a) **(5 poena:)** Napisati Mongoose shemu koja sadrži narednu strukturu: žanr kome knjiga pripada (žanr, niska), naziv knjige (naziv, niska), ime i prezime autora (autor, niska), dan kada je knjiga stigla u knjižaru (datum, datum), cenu knjige (cena, broj) i naziv slike koja je na naslovnoj strani knjige (slika, niska). Sve informacije su neophodne. Napraviti model sa nazivom Knjiga koji ima datu shemu. Obezbediti da se ovi podaci čuvaju u kolekciji knjige.
 - (b) **(5 poena:)** Napisati Mongoose shemu koja sadrži narednu strukturu: korisničko ime (username, niska), lozinku korisnika (lozinka, niska), id kupljene knjige (id, oid). Svi podaci su neophodni. Napraviti model sa nazivom Kupovina koji ima datu shemu. Obezbediti da se podaci čuvaju u kolekciji kupovine.
 - (c) **(5 poena:)** Slanjem GET zahteva na stranicu **http://localhost:3000/**, serverski deo aplikacije treba da prikaže formular kao na narednoj slici, koji se sastoji od tekstualnog polja za unos korisničkog imena i polja za lozinku.

Korisnicko ime:

Lozinka:

Klikom na dugme "Prijavi se", šalje se POST zahtev stranici **http://localhost:3000/knjige**, pri čemu se prosledjuju podaci o korisniku.

- (d) **(25 poena:)** Slanjem POST zahteva na stranicu **http://localhost:3000/knjige**, samo sa informacijama o korisniku, serverski deo aplikacije, za korisnika koji se prijavio prikazuje sve knjige koje je on kupio, uređene po nazivu. Klikom na dugme "Pretrazi" uzimaju se u obzir žanr i opseg cena i šalje se GET zahtev na **http://localhost:3000/knjige/pretraga**. Slanjem GET zahteva na stranicu **http://localhost:3000/knjige/pretraga**, sa informacijama o korisniku, žanru i ceni, serverski deo aplikacije, za korisnika koji se prijavio prikazuje sve knjige koje je on kupio, a koje pripadaju izabranom žanru i čija je cena u odgovarajućem opsegu.

4. Dopuniti implementaciju Node.js serverske aplikacije u duhu Model-Pogled-Kontroler arhitekture:

- (a) **(10 poena:)** Obezbediti da se klikom na dugme "Kupi" šalje POST zahtev na stranicu **http://localhost:3000/knjige/kupi**. Slanjem POST zahteva na stranicu **http://localhost:3000/knjige/kupi** u kolekciju kupovine unosi se informacija da je dati korisnik kupio traženu knjigu. Potom se preusmerite na početnu stranu.