

Uvod u Veb i Internet Tehnologije (2020/2021.)

Primer praktičnog dela završnog ispita

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

Zadaci

1. (10 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.

Prodavnica

Naziv	Cena	Broj artikala
Plazma keks 100g	120	7
Naruci proizvod		
Naziv	Cena	Broj artikala
Dugotrajno mleko bez laktoze 1l	126.85	9
Naruci proizvod		
Naziv	Cena	Broj artikala
Prolom voda 1l	40	32
Naruci proizvod		

2. (10 poena) U datoteci `2.js` dat je niz `proizvodi` koji sadrži informacije o odabranim proizvodima. Koristeći programski jezik JavaScript i Web API, omogućiti da se klikom na dugme „Prikaži proizvode”, u elementu čiji je identifikator „`katalogProizvoda`”, prikažu informacije iz niza `proizvodi`, kao na narednoj slici, nakon 1,5 sekunde.

Prikaži proizvode			
Broj	Naziv	Cena	Broj artikala
1	Plazma keks 100g	120	7
2	Dugotrajno mleko bez laktoze 1l	126.85	9
3	Prolom voda 1l	40	32
4	Nescafe 3u1	15.99	25
5	Sok od pomorandže 1.5l	144.47	14
6	Bebi spanać 500g	180	6

Za naredne zadatke je neophodno koristiti podatke koji se nalaze u MongoDB bazi podataka naziva „*Prodavnica*”.

3. (15 poena) Dopuniti implementaciju Node.js serverske aplikacije u duhu *Model-Pogled-Kontroler* arhitekture:
 - (a) (5 poena) Napisati Mongoose shemu koja sadrži narednu strukturu: naziv proizvoda (`naziv`, niska), cena u dinarima (`cena`, broj) i broj dostupnih artikala u prodavnici (`broj_artikala`, broj). Sve informacije su neophodne. Broj dostupnih artikala podrazumevano ima vrednost 0, čak i ako se ne navede pri kreiranju. Napraviti model sa nazivom `Artikal` koji ima datu shemu. Obezbediti da se ovi podaci čuvaju u kolekciji `artikli`.
 - (b) (10 poena) Slanjem GET zahteva na stranicu `http://localhost:3000/artikal`, serverski deo aplikacije dohvata informacije o svim artiklima koji imaju više od 0 dostupnih artikala za kupovinu. Ove informacije se prikazuju kao na narednoj slici:

Prodavnica

Naziv	Cena	Broj artikala
Plazma keks 100g	120	7
Naruci proizvod		
Naziv	Cena	Broj artikala
Dugotrajno mleko bez laktoze 1l	126.85	9
Naruci proizvod		
Naziv	Cena	Broj artikala
Prolom voda 1l	40	32
Naruci proizvod		

Trenutna porudzbina

Identifikator proizvoda:

Ime i prezime:

Broj artikala:

Datum isporuke:

Naruci proizvod

4. (30 poena) Dopuniti implementaciju Node.js serverske aplikacije u duhu *Model-Pogled-Kontroler* arhitekture:

(a) (15 poena) Klikom na dugme „Naruči proizvod” na stranici iz prethodnog zadatka, upisati identifikator odgovarajućeg proizvoda u polje „Identifikator proizvoda” u formularu desno. Obezbediti u validaciji formulara da za polja u formularu važi da:

- Polje „Identifikator proizvoda” ne sme biti prazno (navođenje samo belina se smatra praznim poljem).
- Polje „Ime i prezime” ne sme biti prazno. Njegova vrednost se moze sastojati samo od malih slova, velikih slova i karaktera minus. Takođe, ime i prezime moraju biti odvojeni tačno jednim karakterom razmaka (pretpostaviti da, ukoliko korisnik ima više imena ili više prezimena, ona su razdvojena karakterom minus, na primer Ana-Marija ili Ilić-Petrović).
- Polje „Broj artikala” ne sme biti prazno. Njegova vrednost je strogo pozitivan celi broj.
- Polje „Datum isporuke” ne sme biti prazno. Njegova vrednost predstavlja datum koji mora biti 2 dana nakon tekućeg datuma (tekući datum je datum kada korisnik popunjava taj formular) ili kasnije.

Ukoliko u bilo kom trenutku prilikom validacije formulara dođe do greške, prikazati korisniku poruku o grešci u informacionom prozoru (`window.alert`) i fokusirati polje koje je pogrešno. Takođe, sprečiti da bilo kakav zahtev bude poslat serveru u slučaju greške.

Klikom na dugme „Naruči proizvod”, podaci iz formulara se validiraju na opisani način i šalju na stranicu `http://localhost:3000/porudzbina/` POST metodom.

(b) (5 poena) Napisati shemu koja sadrži narednu strukturu: ime i prezime kupca (`ime_prezime` , niska), broj naručenih artikala (`broj_artikala` , broj), datum naručivanja (`datum_nar` , datum), datum isporuke (`datum_isp` , datum) i identifikator artikla na koji se porudžbina odnosi (`id_artikla` , identifikator). Sve informacije su neophodne. Napraviti model sa nazivom `Porudzbina` koji ima datu shemu.

(c) (10 poena) Omogućiti da se otvaranjem stranice `http://localhost:3000/porudzbina/` dohvataju informacije koje su ovoj stranici prosleđene POST metodom. Na osnovu dohvaćenih informacija, aplikacija redom:

- Proverava da li za artikal čiji je identifikator prosleđen postoji dovoljno količine (broja artikala) da bi naručivanje bilo uspešno. Ukoliko ne postoji, prikazuje se stranica sa informacijama o grešci. Na toj stranici takođe postoji veza ka početnoj stranici. Prikaz je dat na narednoj slici:

Status porudzbine

Postoji problem sa Vasom narudžbinom: Nema dovoljno artikala

[Početna stranica](#)

- Ukoliko ima dovoljno artikala, ažurira informaciju o datom artiklu tako što se količina umanjuje za onoliko koliko je korisnik naručio u formularu. Nakon toga, dodaje novu porudžbinu na osnovu podataka koje je korisnik uneo na prethodnoj stranici. Prikazuje stranicu sa informacijama o uspešnosti naručivanja. Prikaz je dat na narednoj slici:

Status porudzbine

Uspešno ste naručili proizvod!

Obezbediti da se nakon 10 sekundi prikaže početna stranica.