

Uvod u Veb i Internet Tehnologije (2023/2024.)

Rok

Ispit se radi najviše 180 minuta. Maksimalan broj poena je 60.

Maj 25, 2024

Zadaci

1. (10 poena:) U datoteci 1.html napisati validan HTML5 kod koji definiše strukturu dokumenta kao na narednoj slici. Za stilizovanje je potrebno koristiti biblioteku *bootstrap* (bootstrap.css), dodatno stilizovanje dodati u datoteci 1.css.

Napomena: Dovoljno je navesti dve knjige po izboru, dok će ostatak biti generisan prilikom izrade poslednjeg zadatka.

The screenshot shows a web browser displaying a page titled "Dnevnik čitanja". The page has a pink header bar. Below the header, there is a table of books and a form to add a new book.

Naslov	Pisac	Godina izdanja	Broj strana	Procitano	Link
Pride and Prejudice	Jane Austen	1813	226	da	informacije
Molloy, Malone Dies, The Unnamable, the trilogy	Samuel Beckett	1952	256	da	informacije
Le Père Goriot	Honoré de Balzac	1835	443	da	informacije
Ficciones	Jorge Luis Borges	1965	224	da	informacije

Unesi novu knjigu

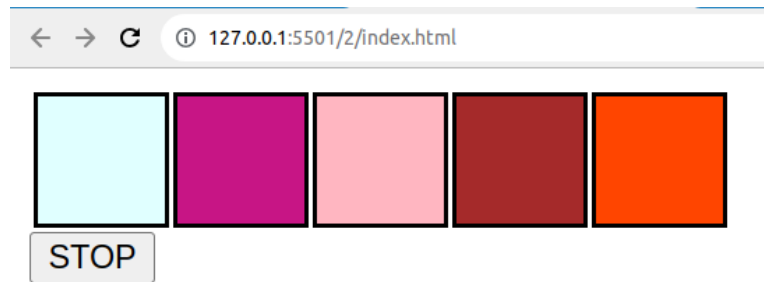
Naslov:

Pisac:

Godina izdanja:

Broj strana:

2. (10 poena:) Koristeći JavaScript i Web API generisati prikaz kao na slici ispod. Boje kvadrata na prikazu treba da se menjaju na svake 2 sekunde. Boja svakog kvadrata treba da se izabere nasumično iz niza boja datih u fajlu 2.js. Klikom na dugme *stop* boje treba da prestanu da se menjaju, odnosno ostaju one koje su zadnje postavljene. Boje na slici su nasumično izabrane.



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

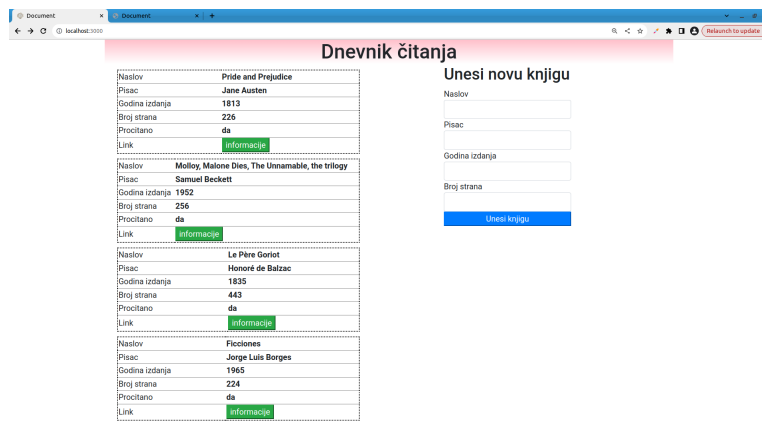
3. Dopuniti implementaciju Node.js serverske aplikacije u duhu Model-Pogled-Kontroler arhitekture:

(a) **(5 poena:)** Napisati Mongoose shemu koja sadrži narednu strukturu:

- id - identifikator u bazi,
- title - niska, obavezno polje,
- author - niska,
- pages - broj,
- year - broj,
- read - istinitosna vrednost, podrazumevano *false*.

Napraviti model sa nazivom *Knjige* koji ima datu shemu. Obezbediti da se ovi podaci čuvaju u kolekciji *knjige*.

(b) **(15 poena:)** Slanjem GET zahteva na stranicu **http://localhost:3000**, serverski deo aplikacije treba da prikaže početnu stranicu:



Klikom na dugme "informacije" na odgovarajućoj knjizi šalje se GET zahtev stranici

http://localhost:3000/knjiga. Pri slanju je potrebno proslediti identifikator knjige za kojoj je kliknuto dugme.

Klikom na dugme "Unesi knjigu" treba poslati POST zahteva ka stranici **http://localhost:3000/novaKnjiga**.

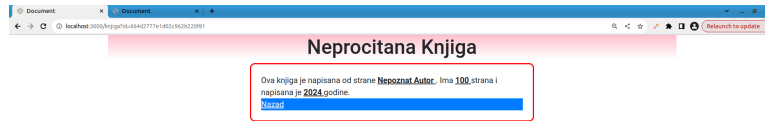
- (c) (10 poena:) Slanjem GET zahteva na stranicu **http://localhost:3000/knjiga** treba prikazati stranicu sa informacijama o knjizi.

Potrebno je iz baze dohvatiti sve podatke o knjizi čiji je identifikator prosleđen. Obratiti pažnju da na stranici podvučene reči odgovaraju knjizi koja se prikazuje. Ukoliko je knjiga pročitana okvir treba da bude zelen, dok ukoliko knjiga nije pročitana okvir treba da bude crven. Prelaskom preko teksta *nazad* potrebno je obojiti pozadinu dugmeta kao na slici sa nepročitanim knjigom. Klikom na dugme *nazad* je potrebno vratiti se na prethodnu stranicu.

Primer pročitane knjige:

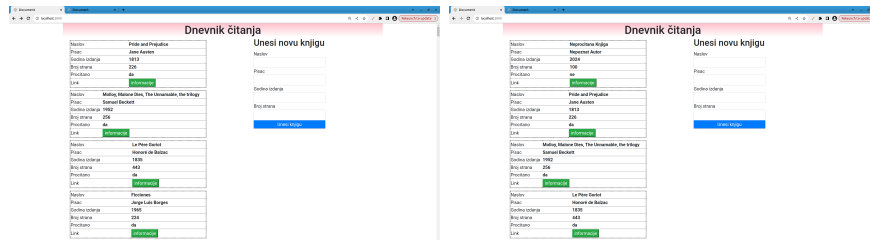


Primer nepročitane knjige:



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

- (a) **(10 poena:)** Slanjem POST zahteva na stranicu **<http://localhost:3000/unesiKnjigu>** se unosi nova knjiga u bazu. Nakon toga nastavlja da se prikazuje ista stranica, ali sa dodatom novom knjigom:



Nakon unosa nova knjiga treba da se nađe pri vrhu stranice jer je nepročitana, dok će njen redosled među nepročitanim knjigama odrediti godina izdanja po zahtevu dela pod b).