

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

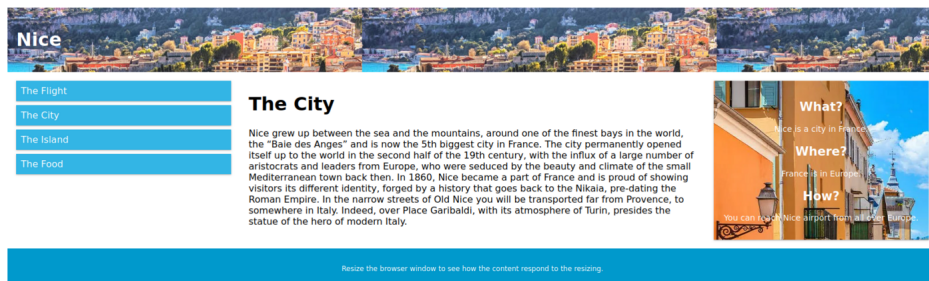
## Sep

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

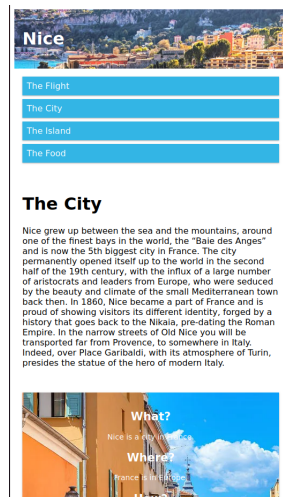
Sep 11, 2024

### Zadaci

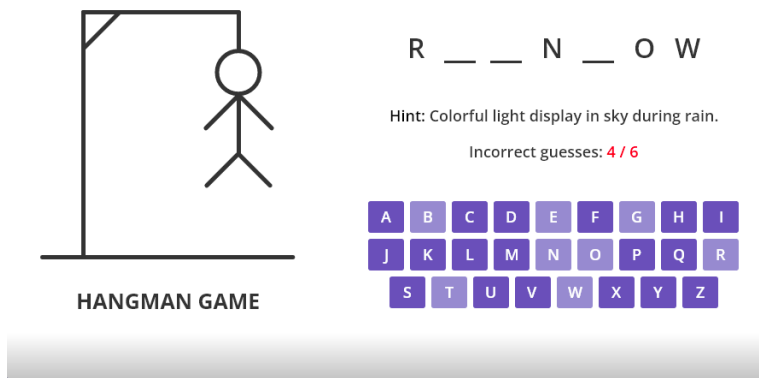
1. **(10 poena:)** U datoteci 1.html napisati validan HTML5 kod koji definiše strukturu dokumenta kao na narednoj slici, a zatim u datoteci 1.css napisati validan CSS3 kod kojim se postiže stilizovanje dokumenta kao na narednoj slici. Za stilizovanje iskoristiti prednosti biblioteke Bootstrap.



Omogućiti da na manjim uredjajima prikaz bude izmenjen tako da elementi budu jedan ispod drugog kao na slici.



2. **(10 poena:)** U 2.js nalazi se spisak reči koje korisnik može da pogadja. Na početku igre (kada korisnik klikne na HANGMAN GAME) se slučajno bira reč. Nakon toga korisnik bira slova klikom na dugme koje sadrži željeno slovo. Ukoliko se slova nalaze u reči, odgovarajuće crtice će se zameniti slovima. Ukoliko se slova ne nalaze u izabranoj reči, povećava se broj neuspešnih pokušaja. Koristeći programski jezik JavaScript i Web API, omogućiti simulaciju igre pogadjanja reči.



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

3. Dopuniti implementaciju Node.js serverske aplikacije u duhu Model-

Pogled-Kontroler arhitekture:

- (a) **(5 poena:)** Napisati Mongoose shemu koja ima narednu strukturu: mesto polaska, mesto dolaska, trajanje leta, datum polaska, oznaku da li je let direktan ili ne, cenu, broj slobodnih mesta. Napraviti model sa nazivom Letovi koji ima datu shemu. Obezbediti da se ovi podaci čuvaju u kolekciji letovi.
- (b) **(5 poena:)** Napisati Mongoose shemu koja ima narednu strukturu: grad, opis, spisak pitanja sa odgovorima. Napraviti model sa nazivom Gradovi koji ima datu shemu. Obezbediti da se ovi podaci čuvaju u kolekciji gradovi.
- (c) **(5 poena:)** Slanjem GET zahteva na stranicu **http://localhost:3000/**, serverski deo aplikacije treba da prikaže stranicu sa formom gde korisnik bira letove. Klikom na dugme "Search flights" šalje se POST zahtev stranici **http://localhost:3000/letovi**.

The screenshot shows a web application interface. At the top is a purple 'Welcome' banner. Below it, on the left, is a vertical menu with links: 'The Flight', 'The City', 'The Island', and 'The Food'. The main content area is titled 'The Flight' and contains a flight search form. The form has fields for 'From' (City or Airport), 'To' (City or Airport), 'Depart' (mm / dd / yyyy), and 'Return' (mm / dd / yyyy). Below these are dropdowns for 'Adults 12+' (set to 1), 'Children 2-11' (set to 0), and 'Infants less than 2' (set to 0). There is a 'Cabin' dropdown set to 'Economy' and a checkbox for 'Direct flights'. A 'Search flights' button is at the bottom of the form. On the right side of the form is a blue box with the text: 'What? Plan your flight.', 'Where? Wherever you want.', and 'How? Using our page.' At the bottom of the page is a blue footer with the text: 'Resize the browser window to see how the content respond to the resizing.'

- (d) **(20 poena:)** Slanjem POST zahteva na stranicu **http://localhost:3000/letovi**, prikazuju se svi dostupni letovi kao na slici.

The screenshot shows the same web application interface as before, but the 'The Flight' section now displays two flight options instead of a search form. The options are: '• BEG direct NCE 2h 05min \$423 din' and '• BEG-MUN-NCE 3h 20min 6000din'. Below the list is a 'See Nice' button. The rest of the interface, including the 'Welcome' banner, the left menu, the right sidebar, and the footer, remains the same.

Naziv dugmeta ispod letova treba da bude "See" plus naziv grada

gde korisnik želi da putuje. Klikom na to dugme šalje se GET zahteva na stranicu **http://localhost:3000/krajnjadestinacija** i prikazuju se informacije o gradu pri čemu se na slučajan način biraju tri pitanja sa odgovorima.

- (e) Dopuniti implementaciju Node.js serverske aplikacije u duhu Model-Pogled-Kontroler arhitekture:

- i. **(10 poena:)** Slanjem GET zahteva na stranicu **http://localhost:3000/krajnjadestinacija**, ažurira se baza tako što se rezerviša ranije odabrani broj mesta na prvom slobodnom letu na željenu destinaciju. Ako je u pitanju let sa presedanjem, na svakom letu rezervisati taj broj mesta.