

## Vega IT / KONTEH Hackathon

### Kvalifikacioni zadatak

## <Zauzmi svoje mesto>

#### Opis:

Koliko često ti se dešavalo da u čitaonici ne možeš da pronađeš slobodno mesto za učenje?

Ovogodišnjom temom hakatona želeli bismo da sistem učinimo malo više fer tako da studenti mogu da zauzmu svoje mesto i vreme u okviru kojeg mogu neometano da uče u čitaonici.

Kako bismo uspeli da za sam početak pripremimo jedan deo implementacije za hakaton, spremili smo ti prvi zadatak.

Ono što treba da uradiš je da kreiraš API koji ćeš koristiti tokom hakatona i koji će ti biti izvor informacija za samu aplikaciju.

#### Domen zadatka:

- **Room:** U aplikaciji će postojati domenski entitet same čitaonice (na eng. Reading room, ali ćemo mi koristiti samo reč room) i ona će predstavljati entitet koji grupiše mesta za čitanje (eng. Desk) u samoj čitaonici. Jedna čitaonica mora da ima barem jedno mesto za čitanje.
- **Faculty:** Predstavlja entitet koji grupiše sve čitaonice koje postoje u okviru jednog fakulteta i on može sadržati nula, jednu ili više čitaonica. Fakultet ima radno vreme čitaonica, tj vreme u kojem studenti mogu da koriste čitaonice u okviru fakulteta.
- **Desk:** Samo mesto za čitanje predstavlja zapravo skup glavne funkcionalnosti ove naše aplikacije i baš ta mesta će se bukirati preko aplikacije. Mesto za čitanje u svakom trenutku može biti prazno ili okupirano. Nije moguće da dva ili više studenata okupiraju isto mesto u istom trenutku. Isto tako, mesto ne može da bude okupirano van radnog vremena fakulteta.
- **\*Booking:** Kako bismo imali način da zauzmemo jedno mesto u čitaonici, potrebna nam je biznis logika rezervisanja tog mesta, ali će ovaj deo biti glavni zadatak na hakatonu i nije očekivano da se bilo šta implementira za ovaj kvalifikacioni zadatak.



### Specifikacija:

- **Faculty, Room, Desk:** U okviru ovog zadatka je neophodno servirati podatke koji se nalaze u ovom dokumentu. To znači da je neophodno implementirati GET metode za sva tri domenska entiteta. Pored ovih osnovnih podataka, potrebno je implementirati i operacije za kreiranje i brisanje Desk entiteta.
  - **Faculty**
    - GET /faculties
  - **Room**
    - GET /rooms
    - GET /rooms?faculty=XYZ
  - **Desk**
    - GET /desks
    - GET /desks?room=XYZ
    - POST /desks
    - DELETE /desks/:id
- **Booking** funkcionalnosti nisu deo ovog kvalifikacionog zadatka i ne treba ih implementirati.

### Ostale informacije:

1. U okviru ovog zadatka je potrebno napraviti REST API u tehnologiji po izboru (.NET, Java, NodeJS, Python i druge).
2. U okviru ovog zadatka je potrebno uspostaviti relacionu bazu podataka i dodati podatke, koji dolaze uz ovaj zadatak, u istu.
3. U okviru ovog zadatka nije potrebno pravljenje klijentske aplikacije (HTML + CSS + JS, niti bilo šta drugo).
4. Kriterijumi za odabir učesnika na hakatonu će biti:
  - a. Pokrivenost specifikacije iz ovog zadatka
  - b. Domensko razumevanje problema kroz detalje implementacije
  - c. Kvalitet napisanog koda
  - d. Brzina predavanja zadatka se neće uzimati u obzir
5. Podaci koji treba da budu dostupni iz aplikacije se nalaze na sledećem linku:

[Reading-Rooms-Data](#)



Vega IT  
**Novosadskog sajma 2,**  
21000 Novi Sad, Serbia  
[vegait.rs](http://vegait.rs)

**Slanje zadatka:**

- Zadatak je potrebno postaviti na javni github repozitorijum
- Link do tog repozitorijuma treba proslediti na **hakaton@konteh.org** i uz to poslati sledeće podatke:
  - Ime i prezime
  - Programski jezici koje si radio
- Pregledanje zadatka će se vršiti nakon što se prijave zatvore 10. marta u ponoć.  
*\*U obzir se uzima samo stanje na repozitorijumu tačno u ponoć 10. marta, što znači da sve naknadne izmene, na repozitorijumu, neće biti uzete u razmatranje.*
- Repozitorijumi koji budu imali više od jednog autora će automatski biti diskvalifikovani

