

# Specifikacija projekta

## Web programiranje 2021/2022

Upotrebom tehnologija i tehnika primenjivanih na predavanjima i vežbama projektovati veb aplikaciju za naručivanje hrane prema sledećoj specifikaciji.

Potrebno je realizovati aplikaciju za sistem koji podržava naručivanje hrane (kao donesi.com ili wolt aplikacije). Aplikaciju koriste 4 grupe (uloge) korisnika: Kupac, Menadžer, Administrator i Dostavljač. Aplikacija rukuje sledećim entitetima:

### - Korisnik

- Korisničko ime (jedinstveno)
- Lozinka
- Ime
- Prezime
- Pol
- Datum rođenja
- Uloga (Admin, Menadžer, Dostavljač, Kupac)

Ako je korisnik Menadžer dodatno ima:

- Restoran za koji je zadužen

Ako je korisnik Dostavljač dodatno ima:

- Porudžbine koje treba da dostavi

Ako je korisnik Kupac dodatno ima:

- Sve porudžbine
- Broj sakupljenih bodova
- Tip kupca

### - Tip kupca

- Ime (npr Zlatni, Srebrni, Bronzani)
- Popust (procenat koji se koristi za obračunavanje cene porudžbine)
- Traženi broj bodova

### - Lokacija

- Geografska dužina
  - Geografska širina
  - Adresa
- Restoran
    - Naziv
    - Tip restorana (italijanski, kineski...)
    - Artikli koje restoran ima u ponudi
    - Lokacija
- Artikal
    - Naziv
    - Cena
    - Tip (jelo, piće)
    - Količina (u gramima ili mililitrima)
    - Opis
- Porudžbina
    - Jedinstveni identifikator narudžbine (UUID)
    - Artikli koji su poručeni
    - Restoran iz kog je poručeno
    - Datum i vreme porudžbine
    - Cena
    - Kupac (korisničko ime)
    - Status (Obrada, u pripremi, čeka dostavljača, u transportu, dostavljena, otkazana)
- Komentar
    - Kupac porudžbine koji je ostavio komentar
    - Restoran na koji se komentar odnosi
    - Tekst komentara
    - Ocena (1-5)

Implementirati sledeće funkcionalnosti:

- Registracija - neregistrovani korisnik se registruje na aplikaciju popunjavajući polja koja su za to predviđena i nakon toga postaje Kupac.
- Administratori se programski učitavaju iz baze i ne mogu se naknadno dodati. **Menadžere i Dostavljače mogu kreirati samo administratori. Kupac ne može da postane Menadžer ni Dostavljač.**

- Prijavljivanje na sistem - neprijavljeni korisnik loguje se na sistem tako što unosi korisničko ime i lozinku korisnika za koji je registrovan. Nakon toga, korisnik je prijavljen i može da izvršava aktivnosti koje su mu predviđene ulogom.
- Ulogovani korisnici mogu da vide svoje profile i da menjaju lične podatke.
- Administratori imaju pregled svih korisnika
- Menadžer ima pregled svog restorana i porudžbina koje su vezane za taj restoran

## Restoran

- Administrator može da kreira nove Menadžere ili Dostavljače
- Administrator može da kreira nove restorane i mogu da unesu informacije o nazivu, tipu, lokaciji. Takođe, admin bira menadžera nadležnog za restoran.
- Prikaz svih restorana u okviru početne strane koja je vidljiva svim tipovima korisnika. Ovaj prikaz treba da sadrži sve restorane i neke osnovne informacije o njima.
- Omogućiti pretragu svih restorana po sledećim kriterijumima:
  - Naziv restorana
  - Tip
  - Lokacija
- Izborom nekog od prikazanih restorana korisnik se prebacuje na prikaz tog konkretnog restorana. Taj prikaz treba da sadrži:
  - Naziv
  - Tip
  - Status (radi/ne radi)
  - Lokacija
  - Ocena restorana (dobijena kao prosek svih komentara ako postoje)
  - Komentare vezane za restoran
  - Lista artikala koje restoran nudi zajedno sa opisom i cenom
- Menadžerima je dostupna funkcionalnost dodavanja novih artikala u restoran za koji je zadužen. Prilikom kreiranja artikala neophodno je uneti naziv, cenu, tip, sliku dok su opis i količina opcioni.
- Pored ovoga, menadžeri mogu da menjaju informacije o postojećim artiklima. Takođe, menadžer može i da obriše artikle.

## Porudžbine

- Kupci imaju opciju pregleda svih svojih porudžbina u okviru svog korisničkog profila.
- Dostavljači imaju opciju pregleda svih porudžbina koje su u statusu "Čeka dostavljača" i one za koje su oni zaduženi u okviru svog korisničkog profila.



- Menadžer ima opciju pregleda svih porudžbina koje su vezane za restoran za koji je on zadužen.
- Porudžbine mogu kreirati samo kupci. Kupac može da poruči iz bilo kog restorana koji je otvoren (Status radi).
  - Koraci prilikom kreiranja porudžbine:
    - Kupac iz liste artikala koje restoran nudi dodaje željene artikle u korpu (u svakom trenutku kupac može da izbaciti artikle iz korpe)
    - Nakon što je završio sa odabirom artikala, kupac ide na pregled korpe u kom mu se izlistavaju informacije o artiklu (naziv, količina, cena) kao i ukupna cena porudžbine. Kupac može da izbaciti artikal iz korpe, kao i da smanji količinu (Količina ne sme biti manja od 1).
    - Kupac klikne na dugme poruči i nakon toga se kreira narudžbina. Ta narudžbina se stavlja u status Obrada i dodaje se u listu porudžbina restorana iz kojeg je kupac poručio.
    - Menadžer tog restorana može da promeni status porudžbine u “u pripremi” i nakon toga u status “čeka dostavljača”.
    - Dostavljač tada može da vidi tu porudžbinu i da je preuzme time što će promeniti njen status u “u transportu”. Nakon toga dostavljač menja status još jednom u “dostavljena” čime se ciklus završava.
  - Kada porudžbina stigne u status “dostavljena”, korisnik koji ju je kreirao dobija određen broj bodova. Ovaj broj bodova se računa po sledećoj formuli:
    - $\text{Broj\_bodova} = \text{ukupna\_cena\_porudžbine} / 1000 * 133$

## Komentari

- Nakon što je porudžbina dostavljena (status narudžbine je “Dostavljena”), korisnik koji je kreirao narudžbinu može da ostavi komentar kao i ocenu. Nakon što je komentar kreiran on se vezuje za restoran iz kog je korisnik poručio hranu i prikazuje se na restoranovoj stranici.

## Administrator

- Administrator ima mogućnost prikaza svih korisnika u sistemu, i može da ih pretražuje po imenu, prezimenu i korisničkom imenu.
- Takođe administratori mogu da brišu restorane.

## Ocenjivanje projektnog zadatka

- Za verzioniranje potrebno je koristiti git. Repozitorijum treba biti dostupan (public) na <https://github.com>
- Nakon formiranja timova, dostavite Vašem asistentu sledeće informacije:

- Spisak članova tima
  - Link ka javnom repozitorijumu
- U git istoriji se mora jasno videti tok razvoja u koji moraju biti uključeni svi članovi tima. Ne prihvata se projekat sa malim brojem ogromnih commitova.
- Dizajn korisničkog interfejsa kao i stilovi su prepušteni studentu. Za olakšanje formiranja layouta stranica studenti mogu koristiti neki od postojećih framework-a (Bootstrap, MaterializeCSS...)
- Serverski deo aplikacije treba da bude realizovan korišćenjem Java programskog jezika i Spring (SpringBoot) radnog okvira.
- Zadatak se sastoji iz tri kontrolne tačke:
  - 1. Predstavlja model (napravljene klase i veze između njih)
  - 2. Predstavlja funkcionalnosti backenda (Java spring deo)
  - 3. Predstavlja uvezivanje sa frontendom (Html, css, js, opciono Vue.js)