

Četvrta laboratorijska vežba iz predmeta Principi softverskog inženjerstva

Primer ispita

Podrazumevano se učitava stranica za prijavljivanje korisnika, osim ako korisnik nije već prijavljen. Omogućeno mu je pritiskom linka za registraciju da pređe na stranicu za registraciju, gde može da kreira svoj nalog. Kada se prijavi, korisnik prelazi na stranicu za prikaz njegovih porudžbina. Pritiskom na dugme za dodavanje porudžbine na veb stranici za prikaz porudžbina se prelazi na stranicu za dodavanje porudžbine, gde se nakon uspešne porudžbine korisnik vraća na stranicu za prikaz njegovih porudžbina. Porudžbina je uspešno uneta ukoliko su sva polja uneta i ukoliko je količina proizvoda pozitivna. Porudžbina se unosi odabirom proizvoda iz padajućeg menija i unosom količine proizvoda za narudžbinu. U nastavku se nalazi spisak stavki koji je potrebno realizovati (a pored svake stavke je naznačeno koliko poena nosi):

- a) [15] Implementirati aplikaciju po opisanoj specifikaciji u *Django* radnom okviru. U nastavku se nalazi raspodela funkcionalnosti po poenima.
- [1] Podignut je server, koji na korenoj putanji aplikacije se podrazumevano otvara stranica za prijavu. Ukoliko je korisnik već prijavljen prelazi se na stranicu za prikaz porudžbina.
 - [1] Sistem je implementiran tako da nije moguće pristupiti funkcionalnostima unosa porudžbine i prikaza porudžbina, ukoliko korisnik nije prijavljen na sistem.
 - [2] Korisnik uvek ima prikazan navigacioni meni, koji izlistava odgovarajuće stavke u skladu sa time da li je korisnik prijavljen na sistem ili ne.
 - [1] Prijavljeni korisnik može da se odjavi sa sistema pritiskom na odgovarajući link.
 - [2] Funkcionalnost prijave je implementirana sa svim neuspešnim scenarijima. Za ovu funkcionalnost (kao i registraciju) je potrebno koristiti ugrađenu korisničku klasu omogućenu od strane *Django* radnog okvira. Forma za prijavu treba da radi u skladu sa ugrađenim podrazumevanim ponašanjem formi koje nudi *Django* radni okvir za ovu funkcionalnost.
 - [2] Funkcionalnost registracije, na koju se prelazi preko linka sa stranice sa prijave je sa svim neuspešnim scenarijima. Forma za registraciju treba da radi u skladu sa ugrađenim podrazumevanim ponašanjem formi koje nudi *Django* radni okvir za ovu funkcionalnost.
 - [2] Nakon uspešne prijave se ispravno izlistavaju narudžbine prijavljenog korisnika.
 - [2] Unos porudžbine je ispravno implementiran i nova porudžbina se unosi ispravno ukoliko su svi kriterijumi za unos zadovoljeni.
 - [1] Ukoliko nisu uneta sva polja za porudžbinu, potrebno je ispisati poruku: "All input fields are required!"
 - [1] Ukoliko uneta količina proizvoda nije pozitivan broj, potrebno je ispisati poruku: "The amount should be a positive number!".

- b) [3] Korišćenjem alata *Selenium IDE*, napisati test primere kojim ćete testirati rad aplikacije, koja je opisana u tekstu zadatka i implementirana u tački a). Dovoljno je testirati sve scenarije vezane za registraciju korisnika.
- c) [2] Dokumentovati metode i klase aplikacije smisleno u skladu sa konvencijama *Django* radnog okvira. Omogućiti prikaz dokumentacije u administracionom panelu *Django* radnog okvira.
- d) [2] Izgenerisati dijagram klasa modela u formatu png fajla.

Napomene:

- Poruke i putanje (ukoliko nisu navedene u postavci, uzeti razumnu pretpostavku o sadržaju poruka) treba da budu iste kao u postavci zadatka. Svaka poruka/putanja koja nije u skladu sa postavkom oduzima po 1 poen.
- Ukoliko rešenje nije predato na disk za predaju u skladu sa zahtevanim formatom, oduzima se 2 poena.
- U slučaju da se zahteva dokumentovanje aplikacije, metode i klase moraju biti smisleno dokumentovane, u suprotnom se dobija maksimalno 50% osvojenih poena za tu stavku.