

Projektni zadatak

decembar 2022

Napraviti jednostavan sistem koji koristi izmišljena *web* prodavnica ETFBL_IP. Sistem se sastoji od nekoliko dijelova koji su opisani u nastavku:

Web shop aplikacija

Ovo je centralna aplikacija sistema koja se koristi za kupovinu i prodaju proizvoda. Aplikacija prikazuje proizvode koje nude korisnici. Svi proizvodi imaju kratak naslov, detaljan opis i kategorije (npr. vozila, nekretnine, računari, telefoni, usluge...), osnovne zajedničke osobine (cijena, nov/polovan proizvod), lokaciju gdje se prodaje proizvod/usluga, slike, informacije o prodavcu i kontakt. Za svaku kategoriju dodaju se specifični atributi (npr. za nekretnine vrsta stan ili kuća, sprat, broj kvadrata...). Sve moguće kategorije i osobine/atributi definišu se u administratorskoj aplikaciji. Potrebno je omogućiti filtriranje i pretraživanje po ovim kriterijumima. Studenti mogu pogledati kako su implementirani slični sistemi za više detalja. Nije dozvoljeno da se istovremeno prikazuju svi proizvodi već je potrebno podržati neku vrstu straničenja (klasično straničenje - *pagination*, *virtual scroll*, ...). Ponude su prikazane kao kartice, sa obaveznom prikazom naziva, slike i cijene. Klikom na jednu karticu otvaraju se detalji na novoj stranici gdje su prikazane sve informacije. Moguće je postavljati pitanja, a sva konverzacija za određenu ponudu se prikazuje svim korisnicima kao komentar.

Korisnici koji nemaju nalog mogu pretraživati ponude, otvarati detalje, ali ne mogu postavljati pitanja ili kupovati. Nalog se kreira na formi za registraciju gdje se unose ime, prezime, grad, korisničko ime, lozinka, avatar (opciono) i mail. Nakon popunjavanja forme, na mail se šalje slučajno generisan PIN (4 broja), koji se unosi na stranici za aktivaciju naloga. Kada je nalog uspješno kreiran i aktiviran, korisnik pristupa osnovnoj stranici ima dostupne sve funkcionalnosti. Ako je nalog kreiran i nije aktiviran, korisnik prilikom prijave na sistem dobija formu za aktivaciju (ako unese tačno korisničko ime i lozinku) i tada se ponovo generiše PIN i šalje na mail. Prijavljeni korisnici mogu promijeniti svoje podatke (osim korisničkog imena) na posebnoj stranici. Osim toga mogu pregledati prethodne kupovine.

Kupovina se obavlja tako što korisnici biraju način plaćanja (kartično, prilikom preuzimanja, ...), a ponuda se evidentira kao kupljena. Nije potrebno detaljno obraditi način plaćanja, već samo osnovnu upotrebu (npr. unos broja kartice).

Svaki korisnik može kreirati novu ponudu koja će biti dostupna u pretrazi drugim korisnicima. Osim toga, korisnik može pregledati svoje ponude (aktivne i završene) i obrisati bilo koju svoju ponudu.

Korisnici koji imaju nalog mogu poslati poruku korisničkoj podršci sa forme koja se nalazi negdje u aplikaciji. Prilikom kontaktiranja čuva se informacija o korisniku i sadržaj poruke.

Aplikacija mora imati uniforman izgled svih stranica i mora biti responsivna. Za izradu koristiti Angular i SpringBoot i bazu podataka po izboru. Dozvoljeno je koristiti gotove biblioteke kao što je Bootstrap ili Material. Sve funkcionalnosti SpringBoot aplikacije dostupne su preko RESTful servisa.

Administratorska aplikacija

Za pristup administratorskoj aplikaciji potrebno je imati nalog koji se otvara direktno u bazi podataka (ime, prezime, korisničko ime, lozinka). Na početnoj stranici nalazi se forma za prijavu. Ako je prijava uspješna otvara se stranica na kojoj se nalazi meni sa opcijama:

- Kategorije: omogućava upravljanje podacima (CRUD) o kategorijama i specifičnim kategorijama ponuda.
- Korisnici: omogućava upravljanje podacima (CRUD) o korisnicima *web shop* aplikacije.
- Statistika: omogućava pregled logova *web shop backend* aplikacije.

Implementacija administratorske aplikacije mora biti pomoću JSP M2. Dozvoljeno je korištenje gotovih biblioteka za dizajn korisničkog interfejsa.

Aplikacija za korisničku podršku

Za pristup administratorskoj aplikaciji potrebno je imati nalog koji se otvara direktno u bazi podataka (ime, prezime, korisničko ime, lozinka). Ovaj nalog nije isti kao nalog za administratorsku ili *web shop* aplikaciju. Ako je prijava uspješna otvara se stranica na kojoj operater može pregledati sve primljene poruke. U tabeli se prikazuju sve nepročitane poruke, a otvaranjem jedne status se mijenja u pročitana. Na poruke se odgovara slanjem *mail-a*. Operater može pretražiti sve poruke po sadržaju.

Aplikaciju za korisničku podršku implementirati upotrebom JSP-a.

Napomene:

- Projektni zadatak se radi pojedinačno.
- Prilikom predaje projektnog zadatka potrebno je predati kompletan izvorni kod, model podataka, kao i model aplikacija. Poželjno je definisati testne podatke.
- Pri realizaciji projektnog zadatka zabranjeno je korištenje generatora koda, osim onih koji su korišteni na laboratorijskim vježbama ili predavanjima.
- Interfejs prema korisniku treba da bude organizovan na jednoobrazan način tj., sve stranice treba da imaju sličan izgled i intuitivne kontrole i prikaz.
- Svaki vid serverske validacije potrebno je da bude što efikasnije realizovan. Obavezno napraviti klijentsku validaciju.

- Nije dozvoljeno koristiti stored procedure, funkcije i okidače (triggere). Dakle, baza podataka ne smije imati bilo kakvu logiku osim definisanih tabela i ograničenja koja važe među njima.
- Ovaj projektni zadatak vrijedi do objave novog projektnog zadatka.