

Aplikacija za on-lajn prodaju nakita

Aplikacija treba da omogući administratorima dodavanje nakita za prodaju. Svaki proizvod koji se dodaje u bazu treba da sadrži dodatne informacije, kao što su opis, materijal, fizičke karakteristike i fotografije. Samo prijavljeni korisnici, koji se na portal prijave sa ispravnih parametrima naloga administratora mogu da pristupe panelu portala namenjenom za administraciju. U ovom panelu mogu da dodaju nove proizvođače, kategorije proizvoda i same proizvode. Svaki komad nakita se sastoji od naslova, jedne ili više fotografija, kratkog i detaljnog opisa proizvoda, kao i cene. Proizvodi treba da pripadaju određenoj kategoriji. Administrator može kroz panel da dodaje i menja kategorije nakita. Korisniku treba omogućiti prikaz svih vrsta nakita poređanih po odabranom podatku, npr. po ceni, po imenu, po datumu dodavanja u bazu itd. Kategorije prikazati u vidu matrice sa slikama za svaku od kategorija na početnoj stranici veb sajta, gde svaka kategorija ima opis i sliku koja predstavlja generalizovani prikaz vrste nakita. Kada korisnik otvori stranicu nekog proizvoda, treba da vidi sve detalje o njemu, kao što su naslov, mala galerija slika tog komada nakita, kratak i detaljan opis, kao i cenu. Treba da dobije mogućnost da izvrši naručivanje nakita on-lajn. Izabrani komad nakita treba čuvati u korpi proizvoda dokle god korisnik ne odabere opciju za kraj kupovine. Prilikom dodavanja u korpu, korisnik treba da unese koju količinu proizvoda želi da poruči, tj. koliko komada. Podrazumevano je jedan komad odabranog nakita. Na kraju kupovine, prikazati korisniku spisak proizvoda koje je dodao u korpu i ponuditi da neki proizvod obriše, da se vrati nazad na kupovinu ili da potvrdi kupovinu. Ako korisnik potvrdi kupovinu, uzeti podatke o korisniku, među kojima su ime, prezime, adresa stanovanja i adresa elektronske pošte. Poslati korisniku na adresu elektronske pošte koju je upisao u formularu prilikom narudžbine spisak stavki koje je kupio, a kopiju mejla dostaviti na određenu adresu komercijaliste koji će nastaviti proces realizacije porudžbine. Grafički interfejs veb sajta treba da bude realizovan sa responsive dizajnom.

Ograničenja:

Aplikacija mora da bude realizovana na Node.js platformi korišćenjem Nest.js razvojnog okvira i sav kôd aplikacije treba da bude organizovan prema pravilima MVC arhitekture. Baza podataka mora da bude relaciona i treba koristiti MySQL/MariaDB RDBMS. Sav generisani HTML kôd koji proizvodi aplikacija mora da bude 100% validan, tj. da generisani kôd prođe proveru W3C Validatorom (dopuštena su upozorenja, ali ne i greške). Aplikacija može grafički korisnički interfejs da generiše na strani servera, korišćenjem šablona za generisanje HTML koda (proizvoljan templating engine, integrisan sa Nest.js aplikacijom) ili da bude serviran statički deo stranice koji pomoću JavaScript-a dinamički formira komponente na front-end-u, a podatke doprema asinhrono kroz veb servis (API) metode obezbeđene u okviru same aplikacije.

Potrebno je obezbediti određeni stepen provere podataka koji se od korisnika upućuju aplikaciji. Moguća su četiri sloja zaštite i to: (1) HTML pattern u poljima za unos podataka u formularima; (2) JavaScript validacija vrednosti unetih u polja za unos podataka u formularima na front-end-u; (3) Provera korišćenjem adekvatnih testova ili korišćenjem regularnih izraza na strani servera u Node.js aplikaciji (moguće je i korišćenjem izričitih šema - Schema) i (4) provera na nivou baze podataka korišćenjem okidača nad samim tabelama baze podataka.

Neophodno je napisati prateću projektnu dokumentaciju o izradi aplikacije koja sadrži (1) model baze podataka sa detaljnim opisom svih tabela, njihovih polja i relacija; (2) dijagram

organizacije elemenata aplikacije sa akcentom na isticanje MVC arhitekture na konkretnom primeru jednog odabranog zahteva, tj. rute/putanje koju Vaša aplikacija obrađuje; (3) popis svih aktivnosti koje su podržane kroz aplikaciju za sve uloge korisnika aplikacije prikazane u obliku Use-Case dijagrama; (4) popis svih kontrolera i njihovih metoda koji obavljaju potrebnu programsku i poslovnu logiku sa ciljem izvršavanja svih predviđenih aktivnosti aplikacije, kao i (5) sve ostale elemente dokumentacije predviđene uputstvom za izradu dokumentacije objavljenom na stranici predmeta u sekciji sa predavanja.

Izrada projekta mora da bude sprovedena korišćenjem alata za verziranje koda Git, a kompletan kôd aplikacije bude dostupan na javnom Git repozitorijumu, npr. na besplatnim GitHub ili Bitbucket servisima. Ne može ceo projekat da bude otpremljen u samo nekoliko masovnih Git commit-a, već mora da bude pokazano da je projekat realizovan u kontinuitetu, da su korišćene grane (branching), da je bilo paralelnog rada u više grana koje su spojene (merging) sa ili bez konflikata (conflict resolution).

Autor teme za projektni zadatak: Milan Tair ([linkedin.com/in/milantex](https://www.linkedin.com/in/milantex))

Ova tema je rezervisana od strane studenata sa brojevima indeksa:

- 2016201116 (primarno odabrao/la)

M. Tair, "Teme projektnih zadataka za predmet Praktikum Internet i veb tehnologije", 2016-2020. [Online]. Available at: <http://zadatak.singidunum.ac.rs/app/piivt-biranje-tema/> [Accessed: 2020-03-30]