Dokumentacija

# Aplikacija e-trgovine za prodaju usluga

**Profesor**: prof. dr Vladislav Miškovic

**Asistent**: Milan Tair

**Studenti**: Branko Rakonjac 2016202495

# Uvod

  Aplikacija treba da omogući administratorima da dodaju nove artikle u bazu podataka aplikacije za elektronsku trgovinu.

## 

## Cilj razvoja

  Aplikacija predstavlja proizvod koji za cilj ima kupovinu usluga putem Interneta od strane različitih korisnika, kao i pregled ponuđenih artikala.

## Obim sistema

  Korisnicima je od strane sistema dozvoljeno da vrše pregled ponuđenih proizvoda, kao i dodavanje istih u korpu, pravljenje porudžbina i uređivanje istih. Porudžbine sastavljene od strane korisnika bice odobravane od strane administratora, koji takođe mogu da uređuju same proizvode i kategorije aplikacije.

## Prikaz proizvoda

### Perspektiva proizvida

  Aplikacija pomoću koje će korisnici moći da, bez previše utrošenog vremena, vrše pregled ponuđenih proizvoda, da iste kupe i dobiju tako što ce im biti dostavljeno na kućnu adresu.

### Funkcije proizvoda

  Korisnik može vršiti pregled ponude proizvoda aplikacije kao i sortiranje po ceni. Klikom na određeni proizvod on će biti preusmeren na stranicu s dodatnim informacijama o samom proizvodu odakle ga može dodati u korpu. Iz korpe može izbaciti odabrane proizvode ili povecati broj proizvoda u korpi.

Pristup administratorskom delu aplikacije dobija se prijavom, koristeci odgovarajuće parametre korisničkog

imena i lozinke.

### Karakteristike proizvoda

  Aplikacija podrazumeva dva tipa korisnika: korisnik (posetilac) i administrator. Aplikacija nije ograničena po pitanju korisnika koji joj mogu pristupiti. Svim korisnicima dozvoljen je pregled ponuđenih proizvoda, sortiranje po ceni, dodavanje u korpu i naručivanje, sve ovo bez ikakve potrebe za prijavom na aplikaciju. Uzima se u obzir da će je koristiti korisnici sa vrlo malo iskustva u polju e-kupovine, ali tako i oni koji već imaju prethodnog iskustva.

Posedovanjem korektnih parametara za prijavu pristupa se administrativnim pravima pomoću kojih se može uređivati sadržaj.

### Ograničenja

* Upotrebljivost

Aplikacija treba biti takva da korisnici mogu efikasno i bez ikakve pomoći izvršiti željene akcije. Zahtevi korisnika biće obrađivani bez grešaka i predugačkog vremenskog intervala čekanja.

* Performanse

Aplikacija treba biti u sposobnosti da u što kraćem vremenskom intervalu odgovori na zahteve korisnika.

* Bezbednost

Samo administrator mogu da pristupaju bazi podataka i uređuju je, stoga su preuzete određene mere kako bi se sprečio neautorizovan pristup osetljivim podacima.

* Pouzdanost

Aplikacija treba funkcionisati bez grešaka i prekida u radu.

# Reference

# Reference projektnog zahteva Aplikacija e-trgovine za prodaju usluga

# Specifikacija zahteva

 Spoljašnji interfejs

### Korisnički interfejs

Prilikom uspešnog pristupa aplikaciji korisniku se prikazuje pregled proizvoda. Levo od prikaza proizvoda nalaze se kategorije proizvoda koje korisnik može dodavati ili uklanjati iz pregleda, kao i dugme za sortiranje proizvoda po ceni. Desno od pregleda proizvoda nalazi se dugme za prijavljivanje na administratorski nalog, koje vodi ka stranici namenjenoj upravo tome. Ispod dugmeta za prijavu nalazi se dugme za pregled korpe, koje vodi na stranicu u okviru koje se proizvodi mogu uklanjati i dodavati u korpu.

### Softverski interfejs

  Aplikacija se može koristiti dokle god se koristi uređaj čiji operativni sistem podržava neki od savremenih pretraživača kao što su Google Chrome i Mozilla Firefox.

Hardverski interfejs

  Pristup aplikaciji je moguć ukoliko je na korišćenom uređaju omogućena veza sa Internetom.

### Komunikacioni interfejs

  Aplikacija koristi HTTP protocol.

Funkcije koje može da obavlja administrator:

* Uređivanje kategorija

Prilikom prijavljivanja na aplikaciju, administrator može uređivati kategorije.

* Uređivanje osobina proizovda

Administrator ima pravo uređivanja osobina proizvoda kao što su naslov, slika, opis, kategorija pripadnosti, spiska sastojaka i cene po jedinici određenoj za tu kategoriju.

* Dodavanje novih proizvoda

Administrator može već postojećim proizvodima dodavati nove.

* Dodavanje kategorija

Administrator može već postojećim kategorijama dodavati nove.

* Uklanjanje kategorija

Administrator može ukloniti kategoriju sa aplikacije.

* Uklanjanje proizvoda

Administrator može ukloniti proizvod sa aplikacije.

* Pregled porudžbina

Administrator može vršiti pregled porudžbina napravljenih od strane korisnika.

* Uređivanje porudžbina

Administrator može korisnikovu porudžbinu obeležiti statusom prihvaćene, odbijene ili realizovane. Porudžbina će pre navedenih statusa biti stavljena na čekanje.

Funkcije koje može da obavlja korisnik:

* Pregled kategorija

Korisnik može vrsiti pregled ponuđenih kategorija proizvoda, prilikom čega može u pregled dodati kategorije po želji, ili ukloniti iste iz pregleda.

* Pregled proizvoda

Korisnik može vršiti pregled ponuđenih proizvoda u okviru izabrane kategorije. Takođe može vršiti pregled dodatnih informacija o svakom proizvodu zasebno, klikom na izabrani proizvod.

* Dodavanje u korpu

Korisnik može izabrani proizvod, iz pregleda proizvoda, proizvoljan broj puta, dodati u korpu.

* Uređivanje korpe

Korisnik može, prilikom pregleda korpe, da ukloni ili doda izabrani proizvod tako što će kliknuti na odgovarajuće dugme koje se nalazi pored samo proizvoda.

* Poručivanje

Nakon što je korisnik završio s uređivanjem svoje korpe i spreman je da izvrši kupovinu proizvoda, to može učiniti klikom na odgovarajuće dugme, nakon čega će se od korisnika zahtevati podaci kao što su ime, prezime, adresa stanovanja i adresa elektronske pošte.

## Projektna 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.

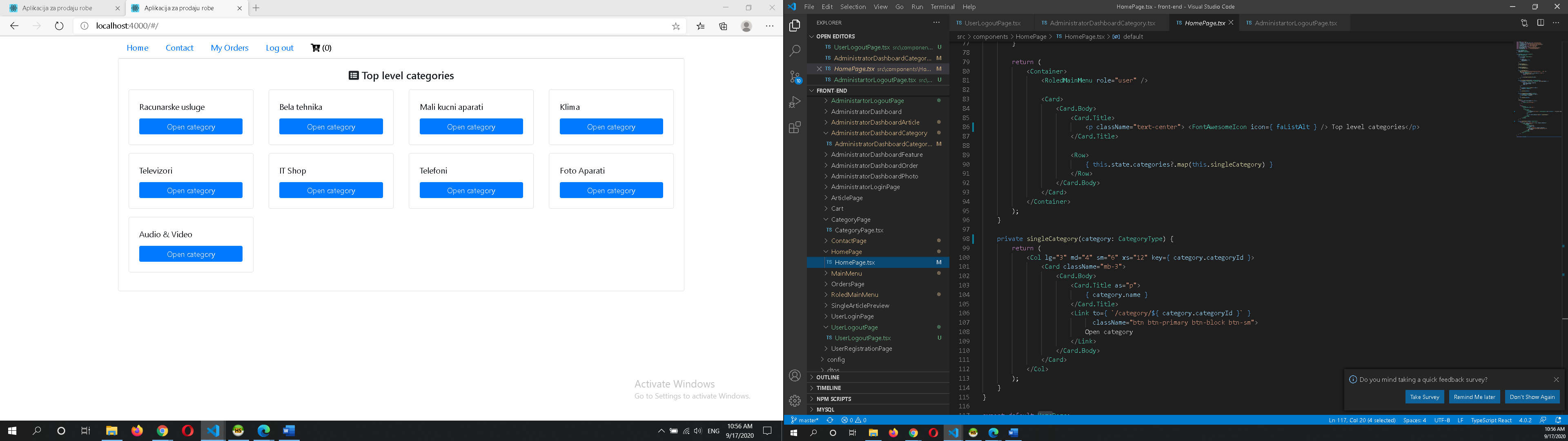
Potrebno je obezbediti određeni stepen provere podataka koji se od korisnika upućuju aplikaciji. Moguća su četiri sloja zaštite i to: (1) HTML pattern u poljima za unos podataka u formularima; (2) JavaScript validacija vrednosti unetih u polja za unos podataka u formularima na front-end-u; (3) Provera 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.

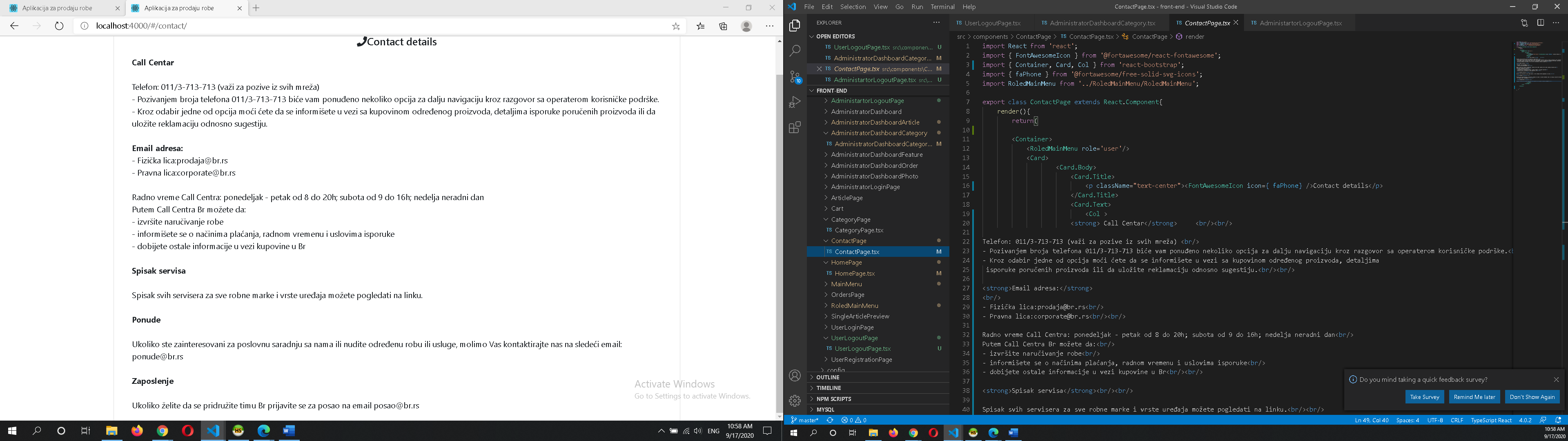
Izrada projekta mora da bude sprovođena korišćenjem alata za verziranje koda Git, a kompletan kôd aplikacije bude dostupan na javnom Git repozitorijumu, npr. na besplatnim GitHub ili Bitbucket servisima. Ne može ceo projekat da bude otpremljen u samo nekoliko masovnih Git commit-a, već mora da bude pokazano 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).

Organizaciona sema:

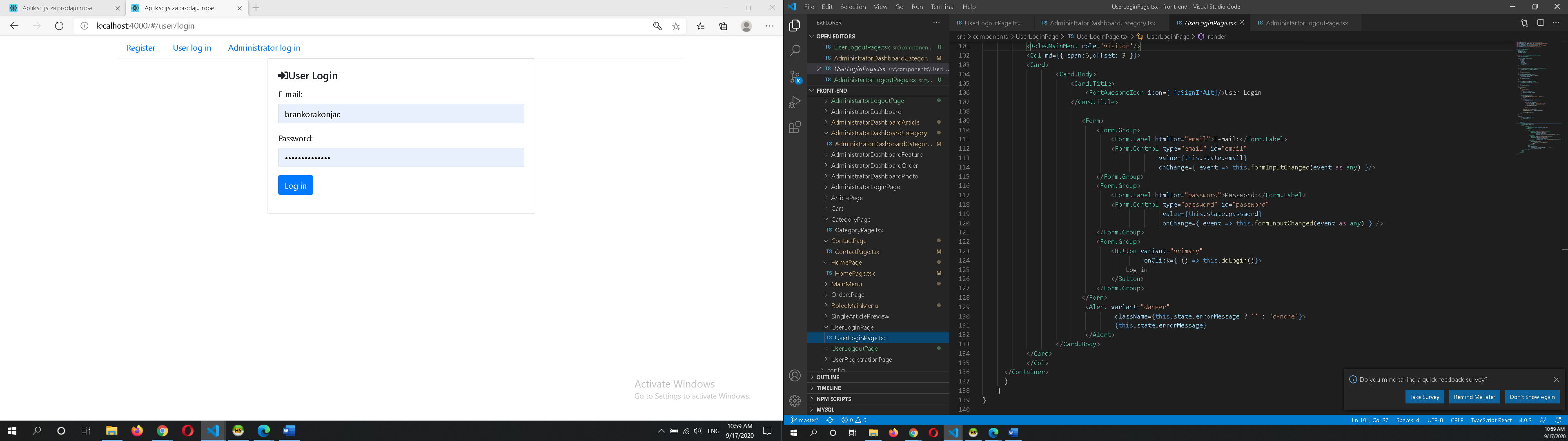
Home:



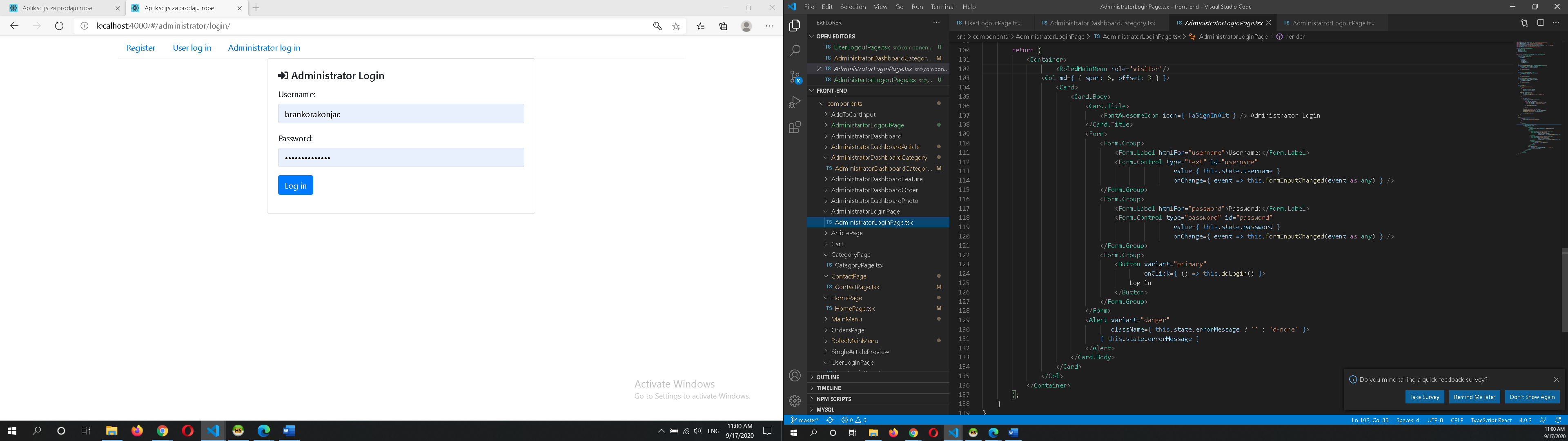
Contact:



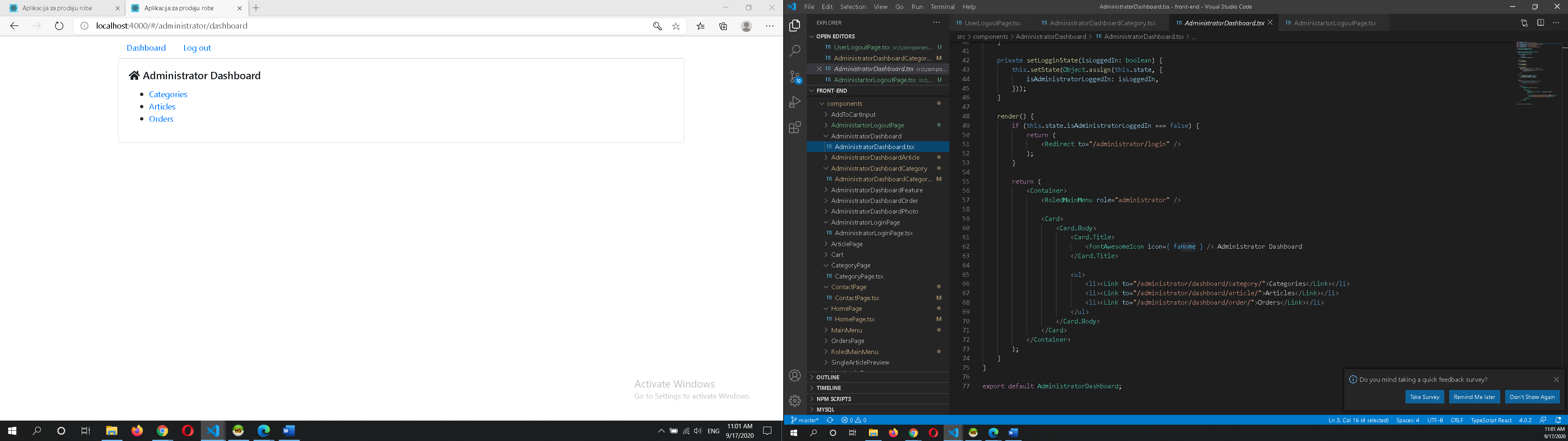
User Login:



Administrator Login:



Administrator page:



Administratoc category:

