Web portal za organizaciju putovanja uz savjete lokalnog stanovnistva

Romić, Valentin

Undergraduate thesis / Završni rad

2023

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:200:592072

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-05-15**

Repository / Repozitorij:

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA OSIJEK

Stručni studij

WEB PORTAL ZA ORGANIZACIJU PUTOVANJA UZ SAVIJETE LOKALNOG STANOVNIŠTVA

Završni rad

Valentin Romić

Osijek, 2023.

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA **OSIJEK**

Obrazac Z1S: Obrazac za imenovanje Povjerenstva za završni ispit na preddiplomskom stručnom studiju

Osijek, 18.09.2023.

Odboru za završne i diplomske ispite

Imenovanje Povjerenstva za završni ispit na preddiplomskom stručnom studiju

Ime i prezime Pristupnika:	Valentin Romić
Studij, smjer:	Stručni prijediplomski studij Računarstvo
Mat. br. Pristupnika, godina upisa:	AR 4688, 26.07.2018.
OIB Pristupnika:	23522973610
Mentor:	Marina Peko, dipl. ing.
Sumentor:	,
Sumentor iz tvrtke:	
Predsjednik Povjerenstva:	Robert Šojo, mag. ing. comp.
Član Povjerenstva 1:	Marina Peko, dipl. ing.
Član Povjerenstva 2:	dr. sc. Ivana Hartmann-Tolić
Naslov završnog rada:	Web portal za organizaciju putovanja uz savjete lokalnog stanovnistva
Znanstvena grana završnog rada:	**Procesno računarstvo (zn. polje računarstvo)**
Zadatak završnog rada	Portal nudi mogućnost registracije putnika ili recenzenta. Putnik bira destinaciju te ima mogućnost planiranja putovanja u smislu lokacija koje planira posjetiti (nema opciju iznajmljivanja smještaja) pri čemu koristi savjete lokalnog stanovništva te im, također, može postavljati upite. Recenzent je osoba koja je lokalni stanovnik mjesta koje se može izabrati kao destinacija za putovanje. Unosi razne znamenitosti koje bi bile zanimljive putnicima, a nema ih u raznim brošurama (lokalni savjeti).
Prijedlog ocjene pismenog dijela ispita (završnog rada):	Izvrstan (5)
Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:	Primjena znanja stečenih na fakultetu: 3 bod/boda Postignuti rezultati u odnosu na složenost zadatka: 3 bod/boda Jasnoća pismenog izražavanja: 3 bod/boda Razina samostalnosti: 3 razina
Datum prijedloga ocjene od strane mentora:	18.09.2023.

Potvrda mentora o predaji konačne verzije rada:	Mentor elektronički potpisao predaju konačne verzije.
	Datum:

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA
I INFORMACIJSKIH TEHNOLOGIJA **OSIJEK**

IZJAVA O ORIGINALNOSTI RADA

Osijek, 28.09.2023.

Ime i prezime studenta:	Valentin Romić
Studij:	Stručni prijediplomski studij Računarstvo
Mat. br. studenta, godina upisa:	AR 4688, 26.07.2018.
Turnitin podudaranje [%]:	13

Ovom izjavom izjavljujem da je rad pod nazivom: **Web portal za organizaciju putovanja uz savjete lokalnog stanovnistva**

izrađen pod vodstvom mentora Marina Peko, dipl. ing.

i sumentora ,

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija.
Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

SADRŽAJ

1. UVOD 1

1.1. Zadatak završnog rada 1

2. SLIČNE WEB APLIKACIJE 2

2.1. Lonley Planet 2

2.2. Tripadvisor 3

2.3. Facebook 3

3. KORIŠTENE TEHNOLOGIJE (MERN) 4

3.1. JavaScript 5

3.2. MongoDB 5

3.3. ReactJs 6

3.4. NodeJs 6

3.4.1. Node Package Manager 7

3.5. ExpressJs 8

4. PROCES IZRADE 9

4.1. Konfiguracija 9

4.2. Baza podataka 10

4.3. Struktura baze podataka 10

4.4. Autorizacija i autentifikacija 11

4.5. Struktura aplikacije 14

5. FUNKCIONALNOSTI 16

5.1. Početna stranica 17

5.2. Registracija i prijava 18

5.3. Glavna stranica 19

5.4. Profilna stranica 20

5.5. Responzivnost 20

6. Zaključak 22

LITERATURA **23**

SAŽETAK **24**

ABSTRACT **25**

ŽIVOTOPIS **26**

1. UVOD

U ovom završnom radu izrađena je web aplikacija za organizaciju putovanja uz savjete lokalnog stanovništva. Cijela aplikacija je izrađena koristeći MERN, od baze podataka do korisničkog sučelja. MERN je kratica za MongoDB, ExpressJs, ReactJs i NodeJs skup tehnologija za izradu web aplikacija uz korištenje JavaScripta i JSON. MongoDB služi kao baza podataka, pomoću koje su pohranjeni svi podatci korisnika i objava ove aplikacije. ExpressJs upravlja rutama, zahtjevima i odgovorima na serveru, a NodeJs obrađuje zahtjeve s klijentske strane. ReactJs omogućava izradu složenih sučelja s jednostavnim komponentama.

Napravljena je web aplikacija koja korisniku nudi mogućnost registracije kao putnik ili recenzent. Nakon registracije i prijave korisnik ima uvid u sve objave koje može sortirati prema gradovima. Ako se korisnik prijavi kao recenzent ima mogućnost objave sadržaja, putnici tu mogućnost nemaju. Putnik može reagiranjem na objave spremiti objave koje ga zanimaju i uz pomoć njih bolje isplanirati svoje putovanje.

U drugom poglavlju uspoređujemo popularne web aplikacije sa zadanim zadatkom. Tehnologije korištene prilikom izrade projekta su opisane u trećem poglavlju. Četvrto poglavlje se odnosi na konfiguraciju raznih modula i postavljanje servera. U zadnjem poglavlju navedene su sve funkcionalnosti web aplikacije i njezin izgled.

1.1. Zadatak završnog rada

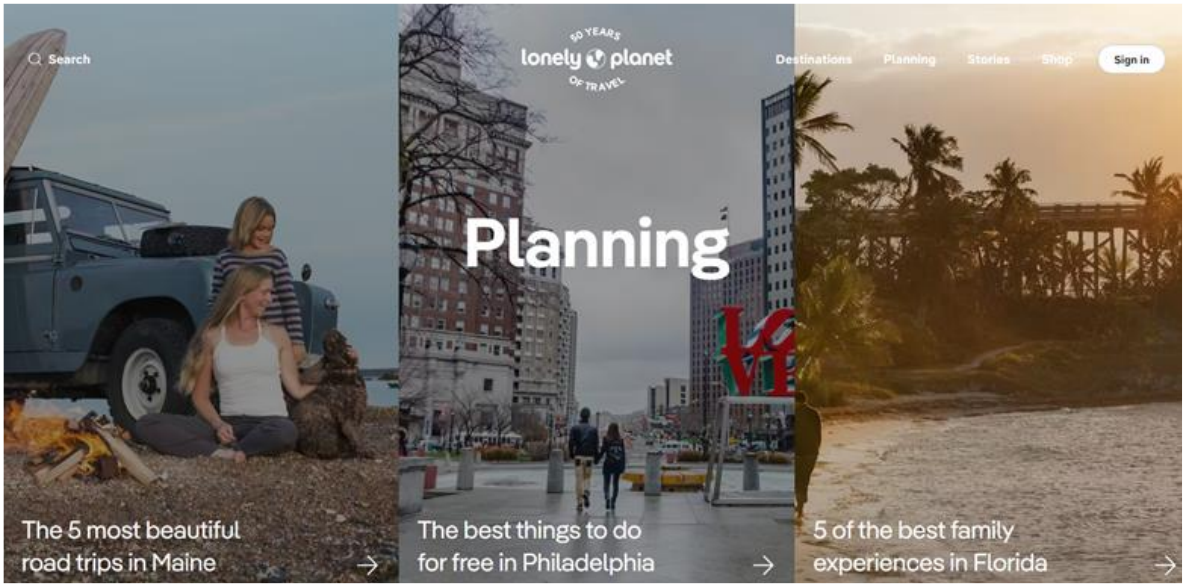
Kreirati web aplikaciju koja nudi mogućnost registracije korisnika kao putnik ili recenzent. Recenzent je osoba koja za svoje mjesto stanovanja unosi znamenitosti koje bi bile zanimljive putnicima za posjetiti, a ne nalaze se na raznim brošurama i kartama. Putnici se prijavljuju u aplikaciju, te na uvid imaju razne objave (preporuke) recenzenata i prema njihovim savjetima prave plan putovanja.

2. SLIČNE WEB APLIKACIJE

Trenutno postoji velik broj web aplikacija za planiranje putovanja. Većina tih stranica radi na principu da osobe koje su posjetile neku lokaciju pišu članke ili prave objave koje korisnici mogu vidjeti i prema tome napraviti svoj plan putovanja. Slijedi nekoliko web aplikacija za planiranje putovanja i na koji način su to izveli.

2.1. Lonley Planet

Lonely Planet je web aplikacija pomoću koje je moguće planirati putovanje na nekoliko različitih načina. Jedan od načina je preporuka prema destinaciji putovanja, korisnik unosi grad koji bi želio posjetiti i na uvid dobiva razne znamenitosti i događaje u tom gradu. Još neki od načina su planiranje prema budžetu putnika i planiranje prema interesima korisnika kao što su koncerti, znamenitosti, hrana, priroda. Korisnici aplikacije nemaju mogućnost objave preporuka, to obavljaju zaposlenici Lonely Planet-a. Slika 2.1. prikazuje naslovnu stranicu Lonely Planet web aplikacije [1].



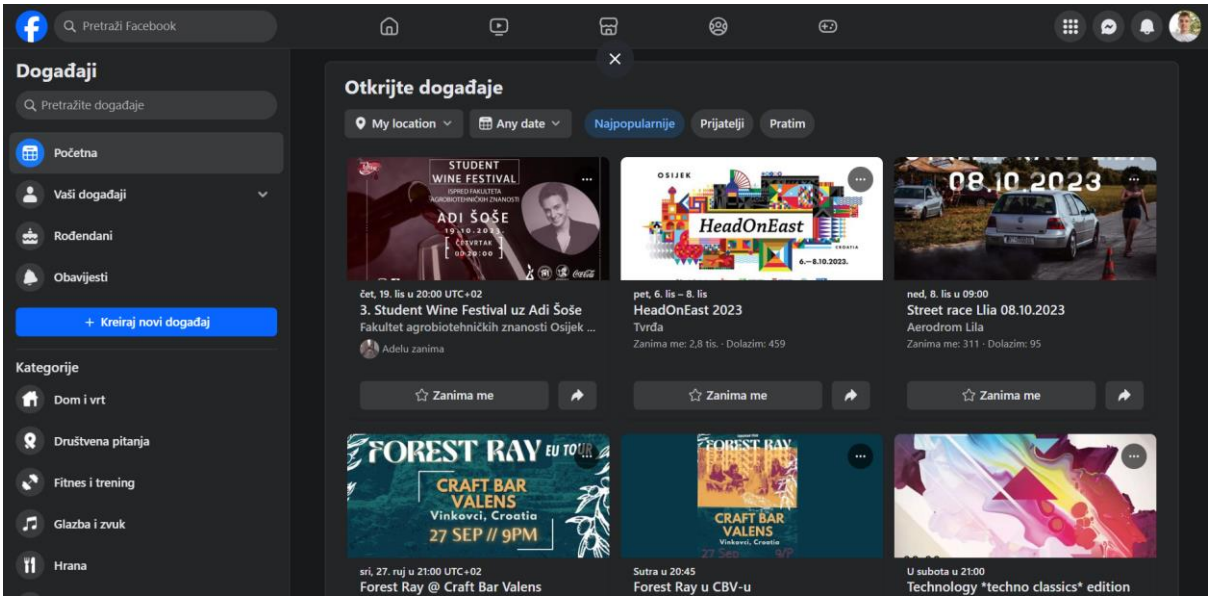
Sl. 2.1. Naslovna stranica Lonly Planet

2.2. Tripadvisor

Putnici diljem svijeta koriste web aplikaciju Tripadvisor kako bi otkrili gdje odsjesti, što raditi i gdje jesti [2]. Korisnici uzimaju u obzir savijete koje postavljaju osobe koje su već tu lokaciju posjetile. Tripadvisor također nudi planiranje putovanja pomoću umjetne inteligencije koja postavlja niz pitanja. Korisnik odgovara na pitanja i prema njima dobiva plan putovanja.

2.3. Facebook

Facebook je popularna web aplikacija koja omogućuje korisnicima izradu računa preko kojeg korisnici mogu komunicirati i razmjenjivati osobna iskustva [3]. Iako primarni zadatak Facebooka nije vezan za planiranje putovanja, na slici 2.2. je prikazan dio aplikacije posvećen događajima. Korisnici imaju mogućnost izrade raznih događaja na koje drugi korisnici mogu postavljati upite i označiti ukoliko bi voljeli posjetiti.



Sl. 2.2. Facebook događaji

3. KORIŠTENE TEHNOLOGIJE (MERN)

Za praktični dio web aplikacije korišten je HTML, CSS i JavaScript tehnologije koje su temelj svake moderne web aplikacije. HTML i CSS su zaduženi za izgled/dizajn stranice, a JavaScript za funkcionalnost. Također su korištene tehnologije MERN stoga, koje su ključne za izradu ove web aplikacije.

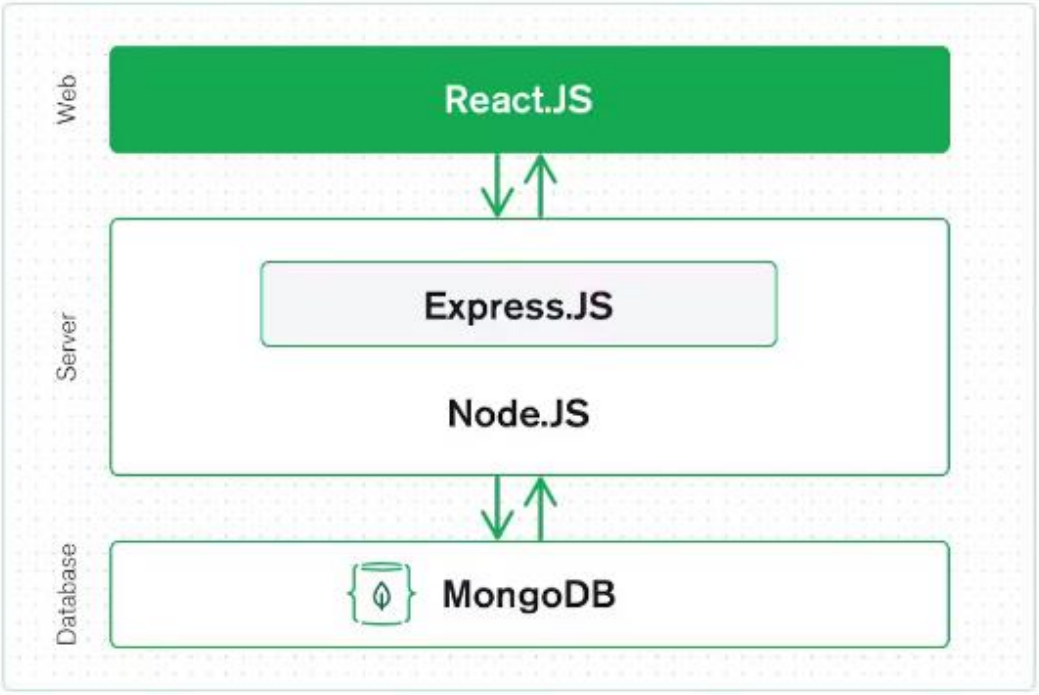
MERN stog je skup tehnologija koji se koriste za razvoj web aplikacija. MERN čine četiri tehnologije koje međusobno jako dobro funkcioniraju. Te četiri tehnologije su :

- MongoDB
- ExpressJs
- NodeJs
- ReactJs

Svaka od ove četiri tehnologije ima važnu ulogu u procesu razvoja web aplikacija. Pomoću MERN stoga moguće je izgraditi potpune aplikacije koje se sastoje od pozadinskih, prednjih i komponenti baze podataka. Glavna svrha MERN stoga je da razvija web aplikacije koristeći samo JavaScript.

Slika 3.1. prikazuje tri razine MERN stoga:

- Web – Gornjom razinom MERN stoga upravlja ReactJs. ReactJs je jedna od najkorištenijih frontend JavaScript biblioteka koji se koriste za razvoj i izradu web aplikacija. Pomoću ReactJs konstruiramo složena web sučelja koristeći jednostavne komponente.
- Server – Srednjom razinom MERN stoga upravljaju dvije komponente, ExpressJs i NodeJs. Ove dvije komponente rade istovremeno jer ExpressJs. održava okvir na strani poslužitelja, koji radi unutar NodeJs poslužitelja. ExpressJs je jedan od najkorištenijih razvojnih JavaScript okvira. NodeJs je poslužiteljsko okruženje otvorenog izvornog koda i izvršava JavaScript kod izvan preglednika. Zbog toga je koristan korisniku računala da brzo stvori bilo koju mrežnu uslugu.
- Baza podataka – Zadnja razina MERN stoga je jedna od najvažnijih i njom upravlja MongoDB. Zadatak baze podataka je pohraniti sve podatke koje su potrebni za aplikaciju. Generira dvije ili više kopija podataka tako da kad god sustav zakaže, može dohvatiti točne informacije [4].



Sl. 3.1. MERN stog

3.1. JavaScript

JavaScript je najkorišteniji programski jezik za izradu web stranica. Izvorno je razvijen od strane Netscapea za dodavanje dinamičkih i interaktivnih elemenata na web stranice. Na JavaScript je utjecala Java, ali sintaksa je puno sličnija programskom jeziku C. [5]

JavaScript je skriptni jezik na strani klijenta koji izvorni kod izvršava na računalu korisnika web aplikacije, a ne na web poslužitelju. Drugim riječima, JavaScript funkcije se izvode tek nakon što je web stranica učitana, bez komuniciranja s web poslužiteljem.

Neke od prednosti JavaScripta: brzina, jednostavnost, svestranost, dostupnost.

Neke od nedostataka JavaScripta: sigurnost i oslanjanje na krajnjeg korisnika [6].

3.2. MongoDB

MongoDB je baza podataka koja se koristi za izgradnju vrlo dostupnih i skalabilnih internet aplikacija. Nudi upravljačke programe za sve glavne programske jezike. MongoDB omogućuje izradu svoju aplikaciju bez trošenja vremena na konfiguriranje baze podataka.

Za razliku od pohranjivanja podataka u tablice redaka i stupaca kao što radi SQL baza podataka, u MongoDB svaki zapis u bazi podataka je dokument opisan u BSON-u, binarnom prikazu podataka. Aplikacije zatim te iste podatke mogu dohvatiti u JSON formatu.

MongoDB je jedna od najkorištenijih baza podataka, zbog olakšane pohrane, upravljanja i pristupa podatcima prilikom izrade web aplikacija. MongoDB se može koristiti s većinom programskih jezik.

3.3. ReactJs

ReactJs je open-source JavaScript biblioteka i okvir koji je razvio Facebook. ReactJs se koristi za učinkovitu, brzu i jednostavnu izgradnju interaktivnih korisničkih sučelja i web-aplikacija. ReactJs je puno efikasnija opcija nego obični JavaScript jer značajno smanjuje količinu koda.

U ReactJs razvijate svoje aplikacije stvaranjem komponenti za višekratnu upotrebu. Umjesto da se bavi cijelim korisničkim sučeljem kao jednom jedinicom, ReactJs potiče programere da razdvoje ova složena sučelja u pojedinačne komponente za višekratnu upotrebu koje čine građevne blokove cijelog sučelja. Radeći to, okvir ReactJS kombinira brzinu i učinkovitost JavaScripta s učinkovitijom metodom manipuliranja DOM-om za brže renderiranje web stranica i stvaranje vrlo dinamičnih i responzivnih web aplikacija.

ReactJs je stekao stabilnost i popularnost zbog svoje sposobnosti stvaranja brzih, učinkovitih i skalabilnih web aplikacija. Danas ga koriste tisuće web aplikacija. Nekoliko značajnih web aplikacija koje koriste ReactJs su: Facebook, Instagram, Netflix, Reddit, Uber, SoundCloud, WhatsApp [7].

3.4. NodeJs

NodeJs je višeplatformsko okruženje otvorenog izvornog koda i biblioteka koja se koristi za pokretanje web aplikacija izvan preglednika klijenta. Stvorio ga je Ryan Dahl 2009. zato što je bio nezadovoljan postojećim rješenjima za izradu web aplikacija.

Slijedi popis značajki zbog kojih je NodeJs najizabraniji od strane softverskih arhitekata.

- **Iznimno brz:** Njegova biblioteka je jako brza u izvršavanju koda jer je NodeJs izrađen na Google Chrome V8 JavaScript Engineu.
- **I/O je asinkroni i vođen događajima:** Svi API-ji biblioteke NodeJs su asinkroni, tj. ne blokiraju. Dakle, poslužitelj temeljen na NodeJs nikada ne čeka da API vrati podatke. Poslužitelj prelazi na sljedeći API nakon što ga pozove, a mehanizam obavijesti o događajima NodeJs pomaže poslužitelju da dobije odgovor od prethodnog API poziva. To je također razlog što je vrlo brz.
- **Jedna niti:** NodeJs slijedi model jedne niti s petljom događaja.
- **Visoko skalabilan:** NodeJs je visoko skalabilan jer mehanizam događaja pomaže poslužitelju da odgovori na način koji ne blokira.
- **Nema međuspremnika:** NodeJs skraćuje ukupno vrijeme obrade prilikom učitavanja audio i video datoteka. NodeJs aplikacije nikad ne spremaju nikakve podatke. Ove aplikacije jednostavno vraćaju podatke u komadima.
- **Otvoreni kod:** NodeJs ima zajednicu otvorenog koda koja je proizvela mnoge izvrsne module za dodavanje dodatnih mogućnosti NodeJs aplikacijama.
- **Licenca:** NodeJs je objavljen pod MIT licencom.

Neke od aplikacija koje koriste NodeJs su Netflix, Uber, Twitter i Slack [8].

3.4.1. Node Package Manager

Node Package Manager (NPM) jedna je od najvećih prednosti NodeJs. NPM omogućuje preuzimanje i korištenje kodnih paketa drugih programera u vašim projektima. Kao rezultat toga, nije potrebno razvijati kod on nule.

NPM je zadani upravitelj paketa i najčešće se koristi. Ima izvrsnu podršku i odličan je za početnike. A u kombinaciji s GitHubom, najvećim svjetskim spremištem koda, imate pristup ogromnoj količini koda koji se može koristiti za rješavanje raznih problema [9].

3.5. ExpressJs

ExpressJs je web okvir kojem je svrha rad na funkcionalnosti web poslužitelja NodeJs. ExpressJs je sloj izrađen povrh NodeJs koji ubrzava i olakšava upravljanje poslužiteljem i rutama. ExpressJs ima veliku brzinu izvršenja zbog petlje događaja, koja eliminira svaku neučinkovitost [10].

Neke od značajki ExpressJs frameworka:

- Može se koristiti za dizajniranje jednostranih, višestraničnih i hibridnih web aplikacija.
- Mogućnost postavljanja međuprograma koji odgovara na HTTP zahtjeve.
- Povećava učinkovitost mobilnih aplikacija
- Omogućuje dinamičko prikazivanje HTML stranica na temelju prosljeđivanja argumenata predlošcima.
- Napisan u JavaScriptu, što znači jednostavnost učenja i pisanja koda.

4. PROCES IZRADE

4.1. Konfiguracija

Prije početka samog programiranja potrebno je instalirati određene pakete i module koji su potrebni za povezivanje stražnjeg, prednjeg dijela aplikacije i baze podataka, zaštitu podataka, te autentifikaciju i autorizaciju korisnika aplikacije.

Neki od instalirani modula i paketa su:

- Mongodb – omogućava povezivanje s bazom podataka i rad s podatcima
- ExpressJs – web okvir za NodeJs
- Morgan – omogućava rukovanje s HTTP zahtjevima
- Helmet – za sigurnost HTTP zahtjeva
- Cors – komunikacija između domena
- Dotenv – odvaja konfiguracijske datoteke od koda
- Nodemon – prilikom promjene izvora koda automatski resetira server

Instaliranje ovih i drugih modula i paketa nam olakšava komunikaciju između različitih komponenata aplikacije.

Nakon instalacije potrebno ih je deklarirati. Slika 4.1. prikazuje način na koji se vrši deklaracija. Pomoću funkcije app.use() postavljamo navedene funkcije međuprograma. Svaka app.use() funkcija se poziva prilikom slanja zahtjeva na server.

Linija ***Kod***

```
1:      /*KONFIGURACIJA*/
2:      import express from "express";
3:      import helmet from "helmet";
4:      import cors from "cors";
5:      import morgan from "morgan";
6:      import dotenv from "dotenv";
7:      import mongoose from "mongoose";
...                ...
20:     dotenv.config();
21:     const app = express();
22:     app.use(express.json());
23:     app.use(helmet());
24:     app.use(morgan("common"));
25:     app.use(cors(options));
```

Sl. 4.1. Kod konfiguracije stražnjeg dijela aplikacije

Sve instalirane ovisnosti i njihove verzije moguće je provjeriti u package.json [11].

4.2. Baza podataka

Za bazu podataka odabran je MongoDB zbog jednostavnosti rukovanja podatcima te njihovoj sigurnosti. Nakon instalacije mongoose paketa potrebno je postaviti konekciju između MongoDB servera i naše aplikacije na način prikazan na slici 4.2.

Linija ***Kod***

```
1:      /*MONGOOSE SETUP*/
2:      Const PORT = process.env.PORT || 6001;
3:      mongoose.connect(process.env.MONGO_URL, {
4:          useNewUrlParser: true,
5:          useUnifiedTopology: true,
6:      )}
7:      .then(() => {
8:          app.listen(PORT, () => console.log(´Server Port: ${PORT}´));
9:      })
10:     .catch((error) => console.log(´${error} did not connect´));
```

Sl. 4.2. Postavljanje konekcije između servera i aplikacije

Pomoću mongoose.connect() funkcije spajamo se na bazu podataka od NodeJs servera pomoću „MONGO_URL“ koji je dohvaćen na web stranici mongodb.com gdje je kreirana baza podataka. Ako je povezivanje uspješno u konzoli će se ispisati PORT koji se koristi, u protivnom će konzola prikazati grešku.

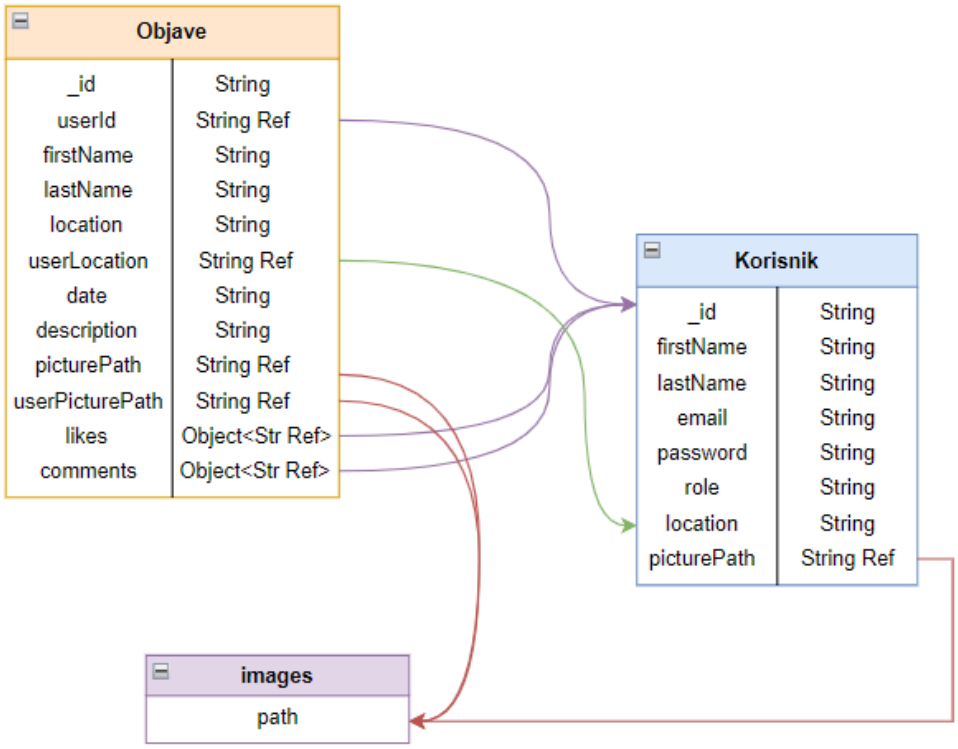
4.3. Struktura baze podataka

ER dijagram je vrsta dijagrama toka koji prikazuje kako se entiteti kao što su korisnici, objekti ili koncepti međusobno povezuju unutar nekog sustava. Također ER dijagram predstavlja podatke u grafičkom obliku, što ih čini lakšim za razumijevanje. Izrada sheme baze podataka na početku izrade aplikacije uvelike pomaže daljnjoj izradi web aplikacije. Podatci iz web aplikacije su podijeljeni na sljedeće entitete :

Podatci iz web aplikacije su podijeljeni na sljedeće entitete :

- korisnici
- objave
- slike

Na slici 4.3. su prikazane vrste podataka i njihove međusobne veze. Tablica Korisnik je popis svih prijavljenih korisnika web aplikacije i u njoj se nalaze osobni podatci svakog korisnika. „_id“ je identifikacijski broj koji se automatski dodjeljuje nakon svake registracije korisnika i za svakog korisnika je drugačiji. Pomoću njega je lako doći do potrebnog korisnika, te tako pristupiti njegovim podatcima. Jedan korisnik može imati više objava, svaka objava sadrži „_id“ objave koja je jedinstvena za svaku objavu, te „userId“, što predstavlja jedinstvenu oznaku korisnika koji je autor objave. Sve profilne slike korisnika i slike korištene za objave su spremljene u posebnu bazu nazvanu „images“.



Sl. 4.3. Struktura baze podataka

Izradom ovakve sheme prije pisanja samog koda, olakšavate i ubrzavate izradu web aplikacije.

4.4. Autorizacija i autentifikacija

Autentifikacija i autorizacija su jedni od najbitnijih dijelova web aplikacije. Autentifikacija je sama registracija i prijava korisnika na web aplikaciju, koja se obično izvršava ispunjavanjem polja za korisničko ime, email i lozinku. Nakon autentifikacije, pomoću autorizacije određujemo koju razinu pristupa određeni korisnici imaju, kao što su pristup određenim funkcionalnostima i

podatcima. Svakim danom slušamo o sve većem broju kompanija koje ugrožavaju informacije svojih korisnika i samim time uništavaju svoj ugled, što uvećava samu važnost autentifikacije i autorizacije.

Slika 4.4. prikazuje isječak koda koji prikazuje način registracije korisnika u bazu podataka. Asinkronim pozivom funkcija „register" se odvaja, te tako postaje nezavisna od ostatka sinkronih funkcija. To znači da ostatak koda se nastavlja izvršavati, što smanjuje vrijeme učitavanja same stranice. Pomoću „req.body" zahtijevamo podatke od prednjeg djela aplikacije, tj. Nakon što korisnik ispuni potrebna polja, klikom na gumb „Registracija" šalje podatke na stražnji dio aplikacije.

Linija ***Kod***

```
1:      /*REGISTRACIJA*/
2:      export const register = async (req, res) => {
3:        try {
4:          const {
5:             firstName,
...               ...
10:            picturePath,
11:         } = req.body;
12:     const salt = await bcrypt.genSalt();
13:     const passwordHash = await bcrypt.hash(password, salt);
14:     const newUser = new User({
15:            firstName,
...               ...
21:            picturePath,
22:     });
23:     const savedUser = await newUser.save();
24:     res.status(201).json({savedUser});
25:     } catch (err) {
26:     res.status(500).json({ error: err.message });
27:     }};
```

Sl. 4.4. Primjer koda registracije korisnika

Lozinka korisnika se pomoću Bcrypt modula pretvara u mnoštvo slova i brojeva, korištenjem određenih algoritama. Ako dođe do ugrožavanja podataka, lozinka korisnika je enkriptirana te beskorisna napadaču. Prije nego što se korisnikovi podatci spreme, unesena lozinka se zamijeni s enkriptiranom lozinkom. Nakon što se svi podatci uspješno spreme šalju se na bazu podataka na koju će biti pohranjeni.

Slika 4.5. prikazuje isječak koda za prijavu korisnika. Nakon provjere podudaranja emaila i lozinke korisnika s podatcima spremljenim u bazi podataka, korisniku se dodjeljuje JWT - „JSON Web

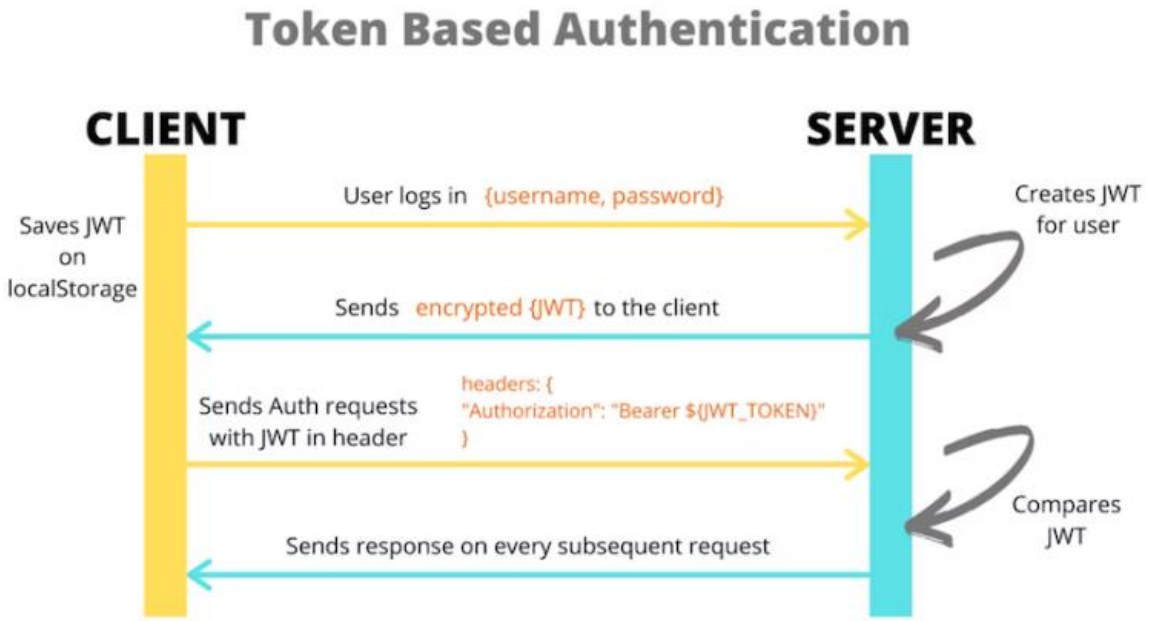
Token“. JSON Web Token (JWT) je otvoreni standard (RFC 7519) koji definira kompaktan i samostalan način za siguran prijenos informacija između strana kao JSON objekt [12].

Linija Kod

```
1:      /*PRIJAVA*/
2:      export const login = async (req, res) => {
3:        try {
4:          const { email, password } = req.body;
5:          const user = await User.findOne({ email: email });
6:          if (!user) return res.status(400).json({msg:"Incorrect mail."});
7:          const isMatch = await bcrypt.compare(password, user.password);
            if (!isMatch) return res.status(400).json({ msg: "Incorrect
8:      password." });
9:      const token = jwt.sign({ id: user._id }, process.env.JWT_SECRET);
10:     delete user.password;
11:     res.status(200).json({ token, user });
12:     } catch (err) {
13:        res.status(500).json({ error: err.message });
14:     }};
```

Sl. 4.5. Primjer koda prijave korisnika

Slika 4.6. pokazuje što se događa kada se korisnik prijavi na web aplikaciju. Nakon prijave, poslužitelj stvara token korisniku (lokalno pohranjen), kada korisnik zatraži bilo kakve podatke od poslužitelja.



Sl. 4.6. Autentifikacija koristeći web token

Za svaki zahtjev provjerava se valjanost tokena, tj. provjerava se je li korisnik i dalje prijavljen i zatim šalje željeni odgovor nazad klijentu . Prilikom odjave korisnika s web aplikacije, token se briše sa strane klijenta [13].

4.5. Struktura aplikacije

Web aplikacija je sastavljena od četiri stranice koje se mogu pristupiti preko ruta ili putanja koje su postavljene na način prikazan na slici 4.7.

Linija Kod

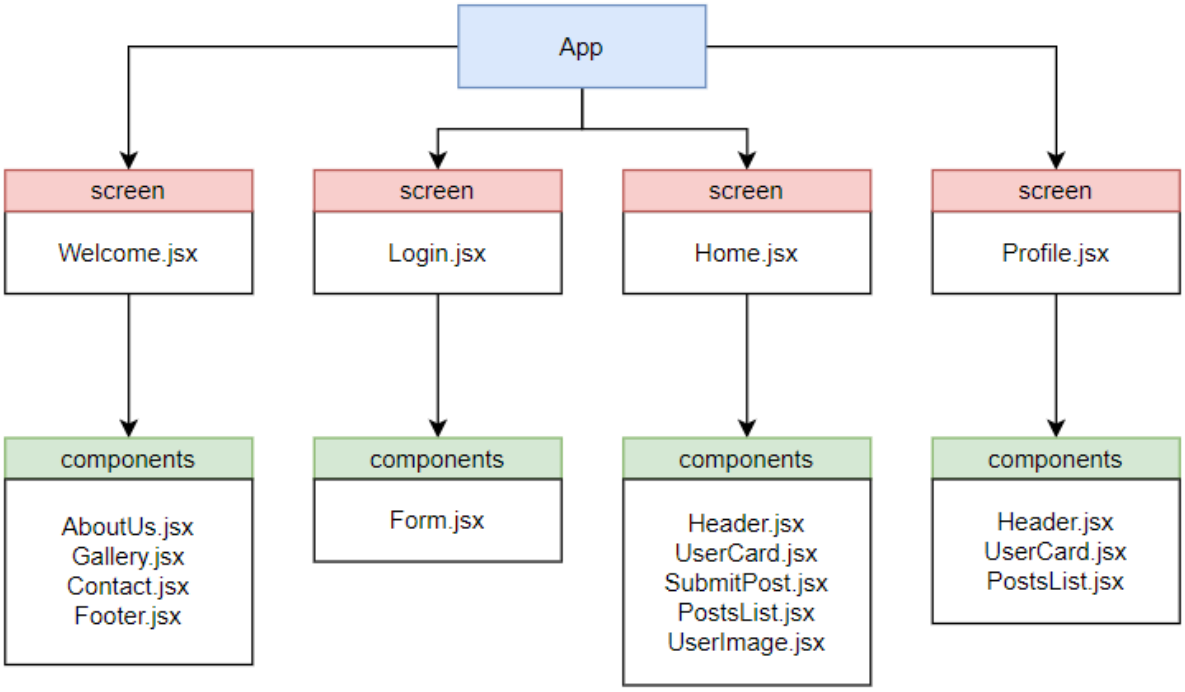
```
1:      /*ROUTES*/
2:      <Routes>
3:         <Route path="/" element={<Welcome/>} />
           <Route path="/home" element={isAuth ? <Home /> : <Navigate
4:      to="/"/>} />
5:         <Route path="/login" element={<Login />} />
           <Route path="/profile/:userId" element={isAuth ? <Profile /> :
6:      <Navigate to="/"/>} />
7:      </Routes>
```

Sl. 4.7. Postavljanje ruta preko kojih pristupano određenim stranicama

Ruta „/“ predstavlja prvo što korisnik vidi kada pristupi web aplikaciji. Na rutu „/home“ koja vodi na glavnu stranicu s objavama i rutu „/profile/:userId“ koja vodi na profil korisnika, je moguće pristupiti samo ako se korisnik aplikacije prijavi. Tu prijavu može izvršiti preko „/login“ rute.

Velika prednost ReactJs je ta što nam omogućuje izradu korisničkog sučelja od zasebnih dijelova koda koji se nazivaju komponente. Umjesto izrade cijelog korisničkog sučelja pod jednom datotekom, naš kod možemo podijeliti na komponente. Jednu komponentu možemo koristiti na više dijelova aplikacije.

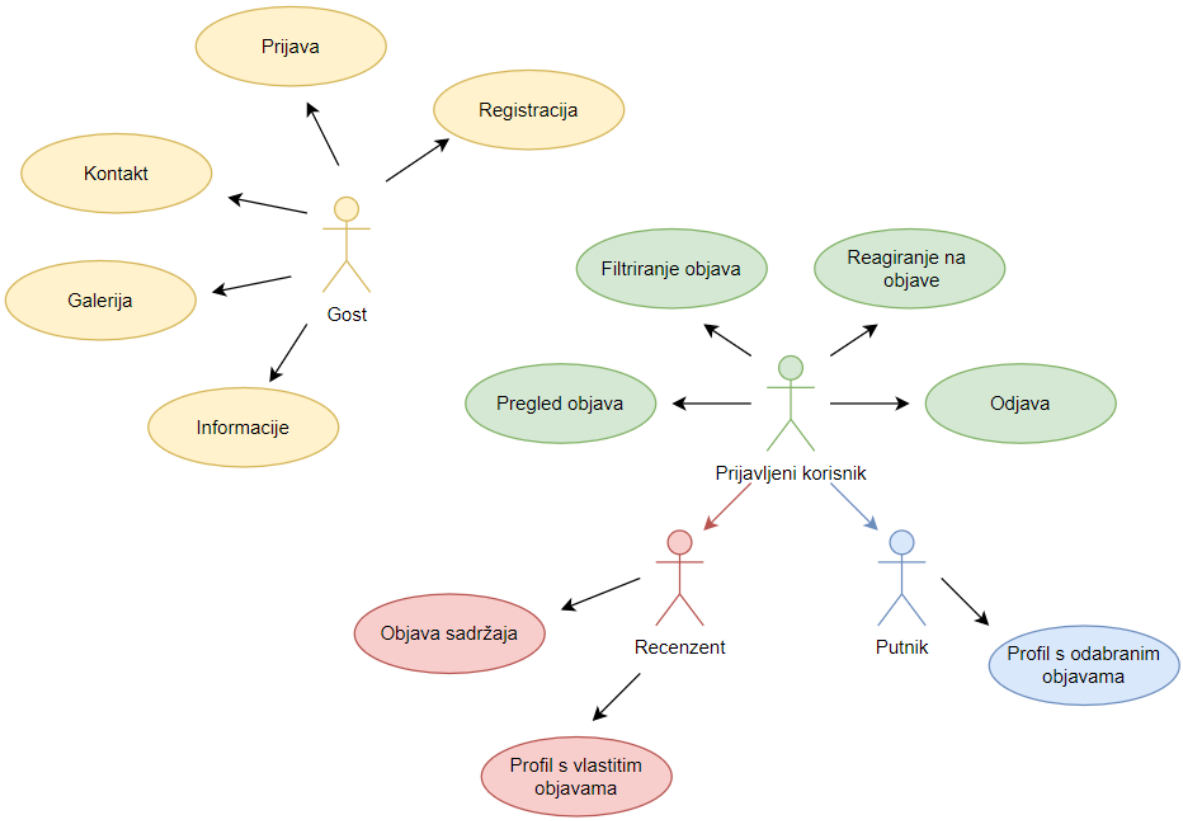
Slika 4.8. predstavlja strukturu prednjeg dijela aplikacije. Primjer kako ReactJs dijeli kod na komponente vidimo na „Home“ i „Profile“ stranici koje dijele komponentu „Header.jsx“ što predstavlja zaglavlje stranice. Tako smanjujemo veličinu koda i povećavamo brzinu izrade aplikacije.



Sl. 4.8. Struktura aplikacije

5. FUNKCIONALNOSTI

Izrađena web aplikacija sastoji se od više stranica na kojima se nalaze različite funkcionalnosti. Pristup stranicama je ovisan o vrsti korisnika aplikacije. Postoje dvije vrste korisnika aplikacije, neprijavljeni korisnik tj. gost i prijavljeni korisnik koji ima dosta veći pristup funkcionalnostima stranice. Slika 5.1. prikazuje skup funkcionalnosti kojima korisnici imaju pristup, obzirom na vrstu korisnika.

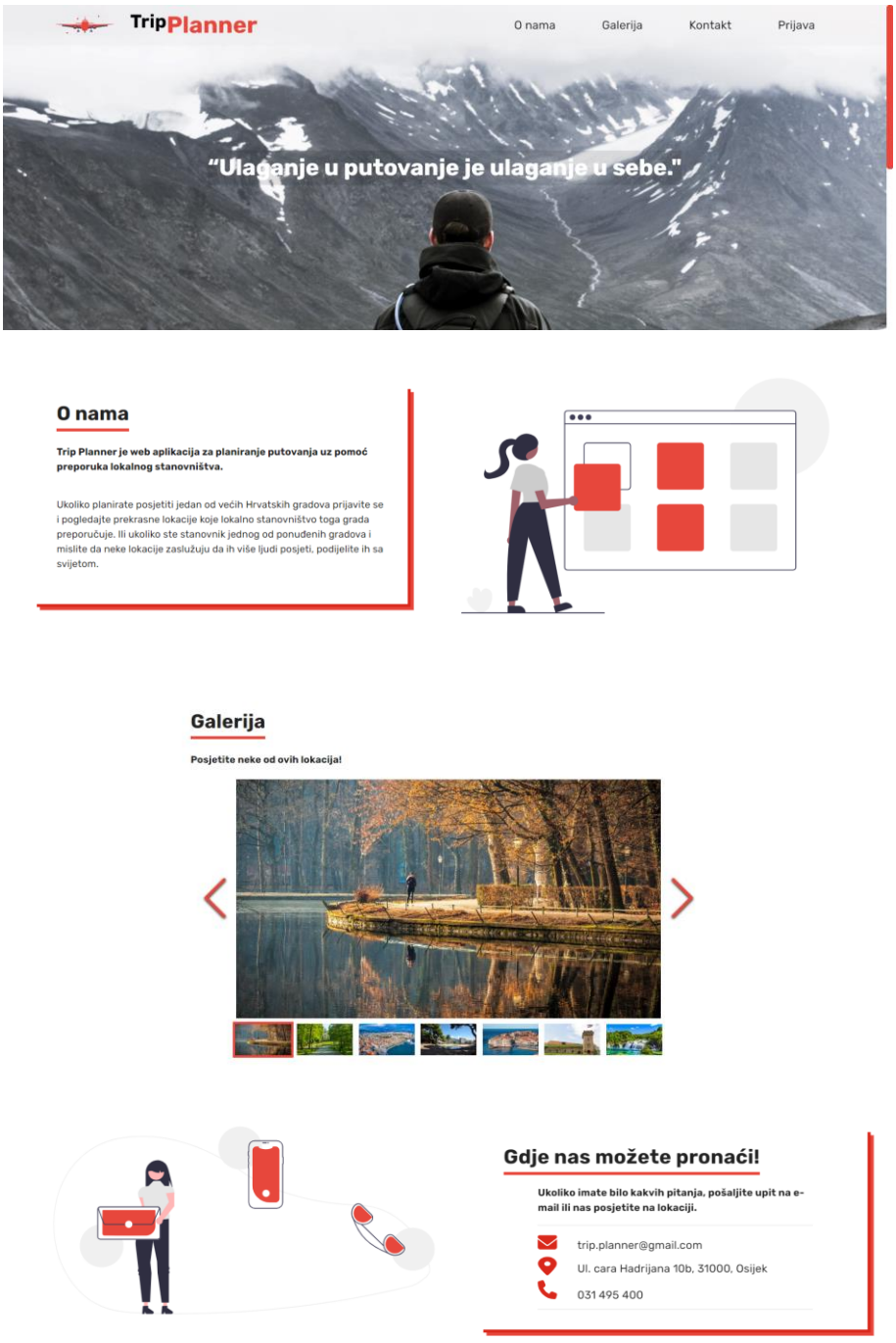


Sl. 5.1. Skup funkcionalnosti

Gost ili neprijavljeni korisnik ima mogućnost prijave, registracije, te pristup ograničenim informacijama i galeriji. Prijavljenim korisnicima se omogućuje pristup pregledu objava, filtriranju objava, reagiranju, komentiranju objave, te odjavi. Prijavljeni korisnici se dijele na recenzente i putnike, koje odabiru prilikom registracije. Prijavljeni korisnici također imaju mogućnost promjene vrste računa.

5.1. Početna stranica

Početna stranica ima veliku važnost jer ostavlja prvi dojam korisniku, ne bitno o vrsti web aplikacije. Upravo to prikazuje značaj dobrog stila stranice. Iako je glavni dio svake stranice njen sadržaj, a ne njen izgled, pomoću kvalitetnog odabira fonta, boje i rasporeda elemenata potičemo korisnike da se zadrže i malo više istraže web aplikaciju. Slika 5.2. predstavlja početnu stranicu.

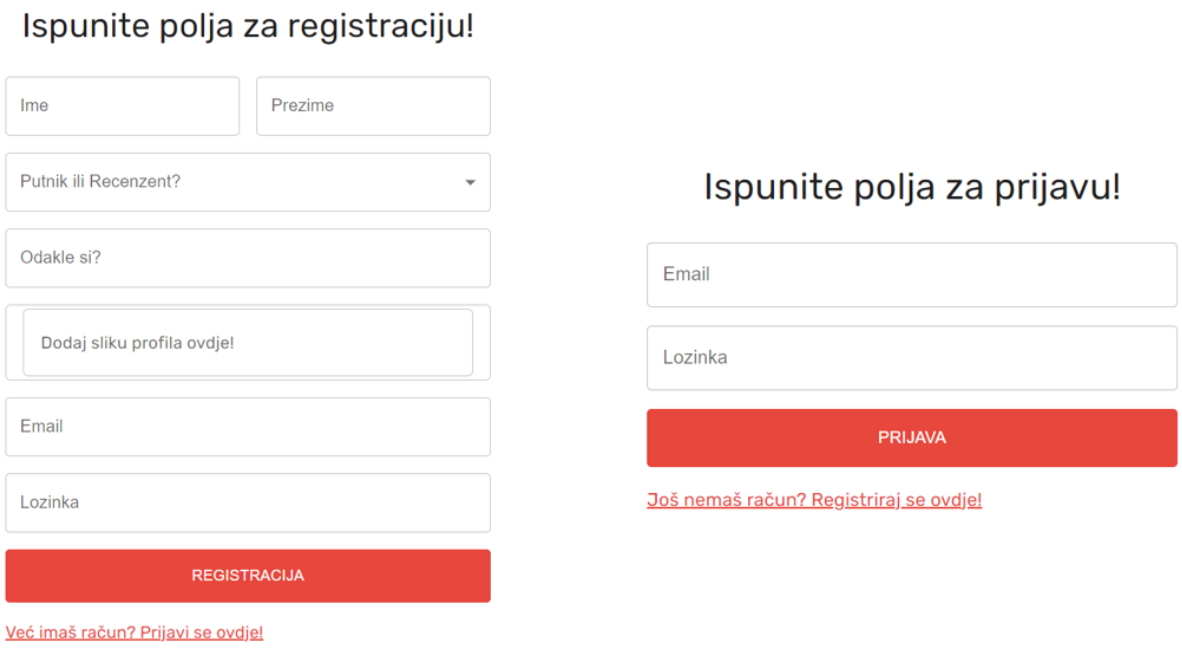


Sl. 5.2. Prikaz početne stranice

Početna stranice se sastoji od zaglavlja i komponenata „O nama“, „Galerija“, „Kontakt“, „Prijava“. Na zaglavlju web aplikacije se nalazi logo, ime stranice, te traka za navigaciju komponenata „O nama“, „Galerija“, „Kontakt“ koji daju informacije i bolji opis web aplikacije. Gost ili neprijavljeni korisnik ima i mogućnost prijave koju može pronaći u navigaciji zaglavlja [14].

5.2. Registracija i prijava

Gost ili neprijavljeni korisnik nemaju pristup glavnom dijelu stranice, tj. pregledu i izradi objava. Kako bi dobio pristup gost se mora registrirati pomoću formulara prikazanog na slici 5.3. Nakon ispune polja za ime, prezime i lokaciju, korisnik odabire vrstu profila. Recenzent je osoba koja je lokalni stanovnik mjesta koje se može izabrati kao destinacija za putovanje. Putnik bira destinaciju te ima mogućnost planiranja putovanja u smislu lokacija koje planira posjetiti. Glavna razlika recenzenta i putnika je mogućnost postavljanja objava. Zatim korisnik dovršava registraciju odabirom slike, emaila koji se ne podudara s već prijavljenim korisnicima te lozinkom. Ako korisnik ostavi polje prazno ili ga pogrešno ispuni dobiva upozorenje koje polje je u pitanju.

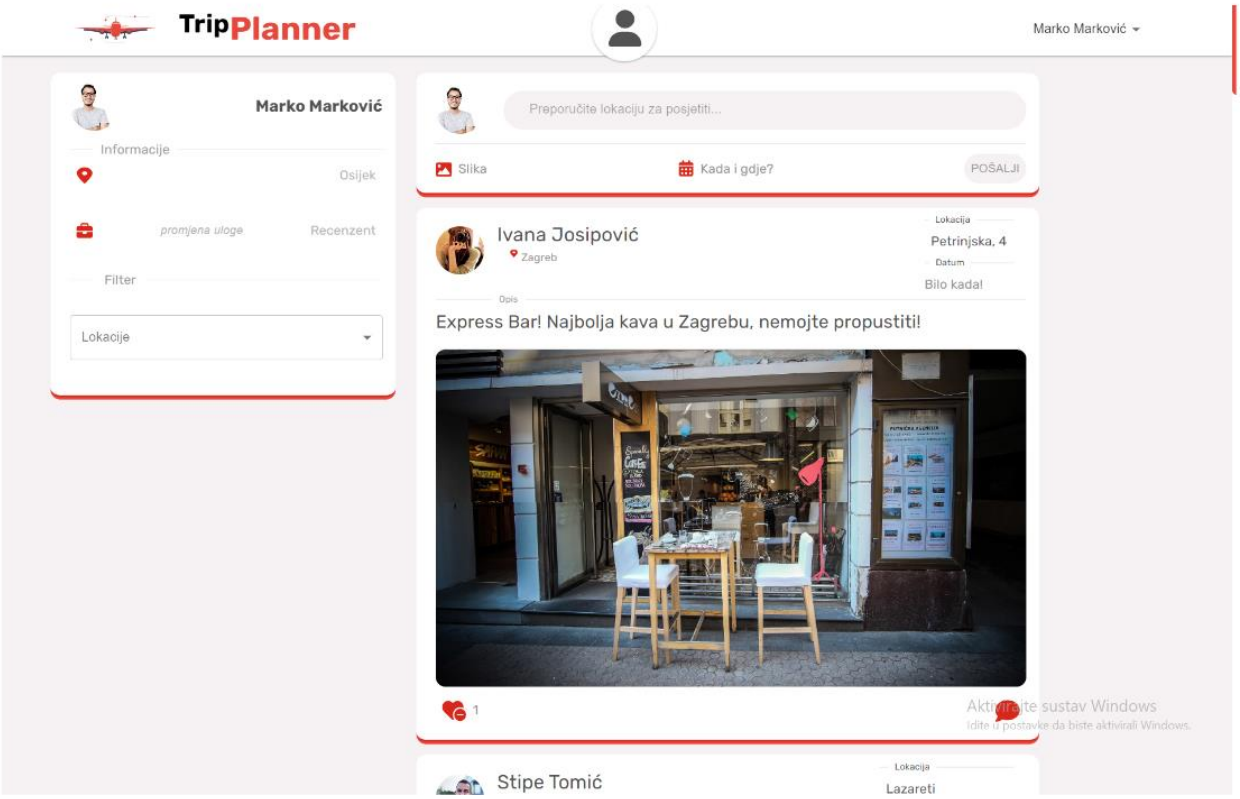


Sl. 5.3. Forme za registraciju i prijavu

Nakon registracije gost se prijavljuje na stranicu pomoću valjane email adrese i lozinke. Te dolazi na glavnu stranicu sa sadržajem koji ga zanima.

5.3. Glavna stranica

Korisnici ispravnom prijavom, ovisno o njihovoj vrsti (putnik ili recenzent) dobivaju pristup glavnom dijelu stranice. Glavni dio stranice su objave koje recenzenti na osnovu svog životnog iskustva dijele s ostatkom korisnika. Glavni dio stranice je prikazan na slici 5.4.

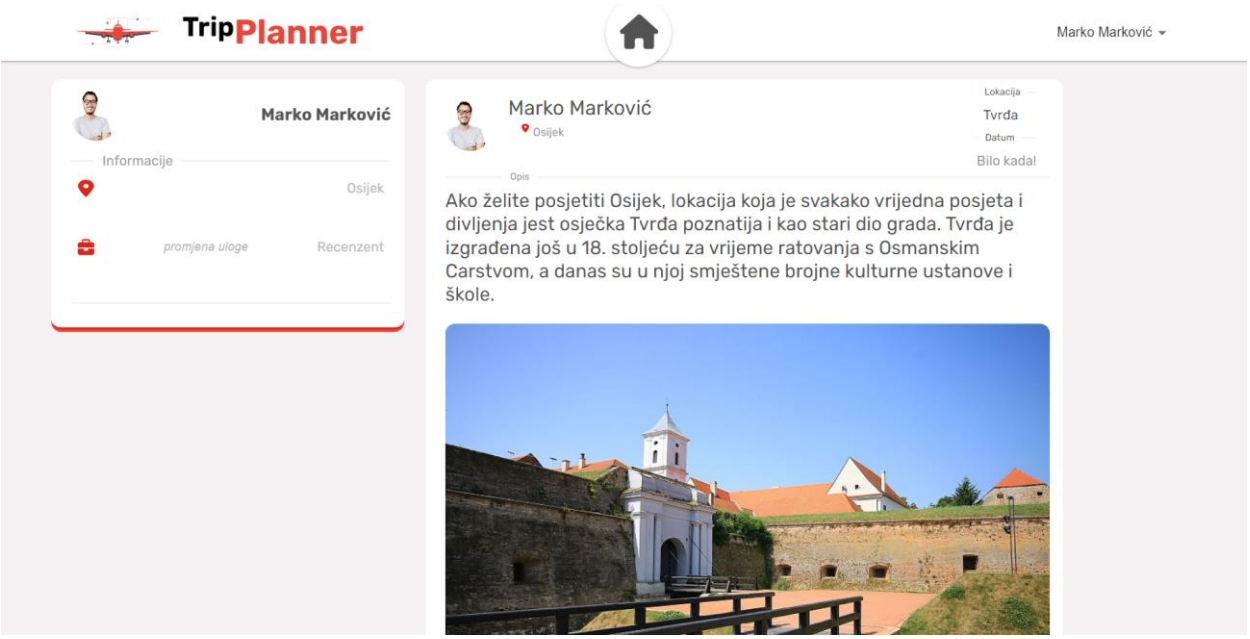


Sl. 5.4. Glavna stranica

Glavna stranica se sastoji od zaglavlja, koje se sastoji od loga, imena web stranice, ikone profila koja klikom nas vodi na našu osobnu stranicu, te imena korisnika. Korisnik klikom na svoje ime u gornjem desnom kutu ima mogućnost odjave. Ispod zaglavlja nalaze se dvije komponente: kartica profila i lista objava. Kartica ima mogućnost promijene vrste korisnika iz recenzenta u putnika i obratno. Promjena uloge zahtjeva ponovnu prijavu korisnika. Također na toj kartici postoji mogućnost filtriranja objava po lokaciji. Desno od kartice profila nalazi se glavni dio s objavama korisnika. Svaka objava ima informacije o autoru objave, opis toga što recenzent preporučuje, lokaciju, sliku, te ako postoji, vrijeme događanja. Pri dnu objave se nalaze dvije ikone, lijeva ikona za reagiranje na objavu i desna ikona za komentiranje ako korisnik ima nekakvih upita za autora objave ili druge korisnike stranice. Ako se korisnik prijavio kao recenzent iznad objava će dobiti komponentu za izradu objave. Putnici tu mogućnost nemaju.

5.4. Profilna stranica

Profilna stranica prikazana na slici 5.5. je slična glavnoj stranici, također se sastoji od zaglavlja, kartice profila i listom objava. Lista objava se mijenja ovisno o tome prijavite li se kao recenzent ili putnik.



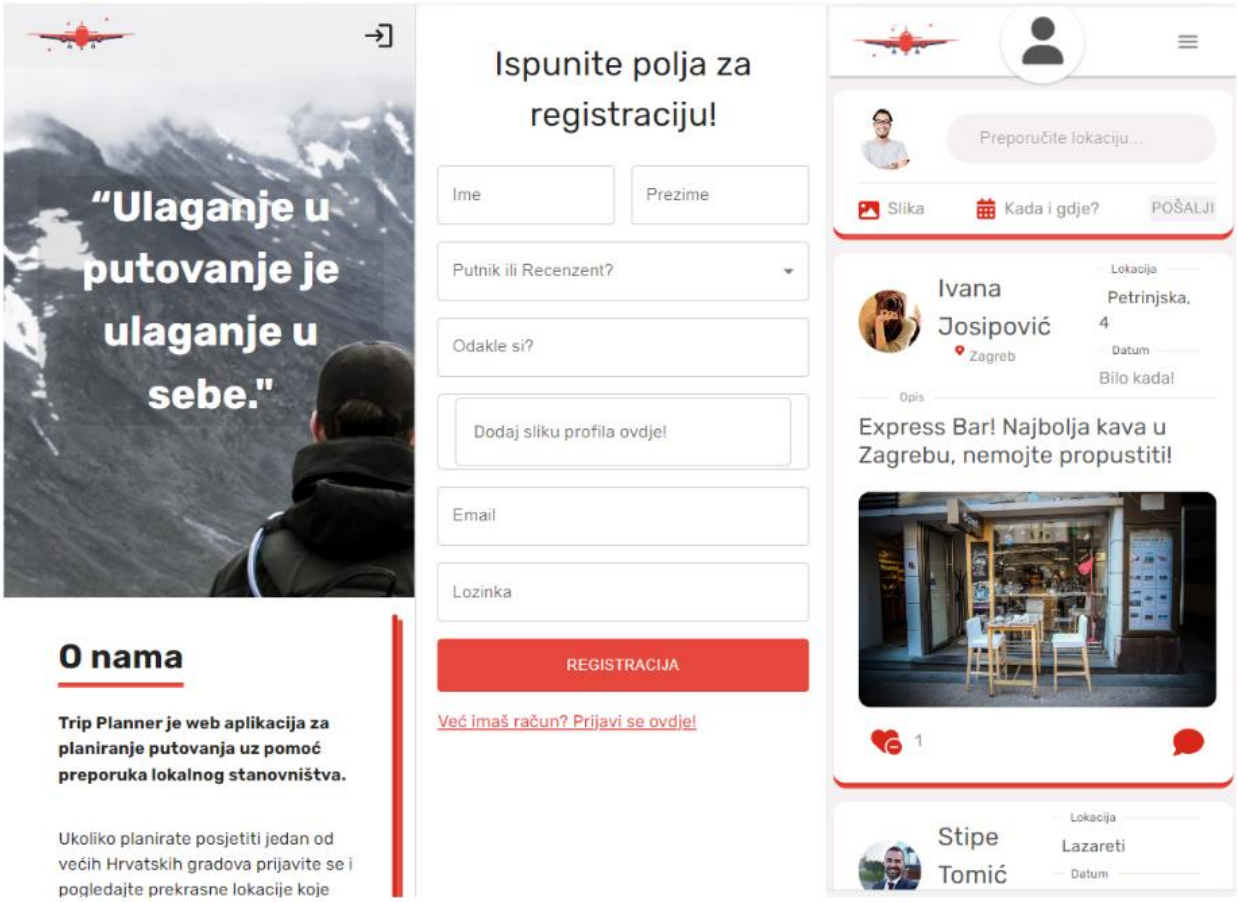
Sl. 5.5. Profilna stranica

Recenzenti na profilnoj stranici dobiju uvid u sve objave koje su oni kreirali, dok putnici imaju popis svih objava na koje su pozitivno reagirali. Prilikom posjeta jednog od odabranih gradova imaju brz i jednostavan pristup svim objavama s lokacijama koje su željeli posjetiti.

5.5. Responzivnost

Cijela web aplikacija je responzivna, što znači da se sadržaj prilagođava veličini ekrana korisnika. Kako ne bi morali raditi više zasebnih web stranica za različite rezolucije, Tako uz pomoć responzivnog dizajna možemo pristupiti na web aplikaciju koristeći mobitel, tablet ili laptop. Ako ne postoji responzivnost, pristupom preko mobilnog uređaja web aplikaciju bi konstantno morali pomicati lijevo, desno te povećavati da bi pročitali sadržaj i tako dobivamo frustriranog korisnika koji će vjerojatno napustiti stranicu.

Na slici 5.6. je prikazana web aplikacija na zaslonu mobilnog uređaja iPHONE 12. Određeni elementi su se morali skratiti/ukloniti kako bi sve stalo na manji zaslon, ali glavni dio sadržaja je ostao pregledan.



Sl. 5.6. Mobilni prikaz web aplikacije

Sa sve većim posjetama preko pametnih telefona, responzivnost web aplikacije više nije samo potreba nego i standard [15].

6. Zaključak

Putovanje je jedinstveno iskustvo jer je to najbolji način da se odvojite od pritisaka i napetosti svakodnevnog života. Cilj ovog rada je izrada aplikacije koja olakšava korisniku odabir lokacije za posjetiti, koje drugi korisnici predlažu uzimajući u obzir vlastita iskustva.

Web aplikacija je napravljena kao društveni medij na kojem korisnici koji planiraju putovanje komuniciraju preko objava. Objave postavlja lokalno stanovništvo kako bi obasjalo svijetlo na lokacije koje zaslužuju veću posjećenost u njihovom gradu. Korisnici objave mogu filtrirati po gradovima. Sve objave na koje korisnici pozitivno reagiraju nalaze se na profilnoj stranici i tako tijekom putovanja, prijavom na aplikaciju i odlaskom na profilnu stranicu imaju brz uvid u lokacije koje žele posjetiti. Web aplikacija je responzivna, što znači da se sadržaj prilagođava veličini ekrana.

Sve funkcionalnosti web stranice od registriranja i prijave korisnika do postavljanja objava su napravljene koristeći MERN tehnologije. MERN stogu raste popularnost među programerima zbog svoje skalabilnosti i jednostavnosti korištenja.

LITERATURA

[1] Red Ventures company , »Lonely Planet,« [Mrežno]. Available: https://www.lonelyplanet.com/about. [26.09.2023.]

[2] »Tripadvisor,« [Mrežno]. Available: https://tripadvisor.mediaroom.com/us-about-us. [26.09.2023.]

[3] Meta, »Facebook,« [Mrežno]. Available: https://www.facebook.com/. [26.09.2023.]

[4] MongoDB, Inc., »MongoDB,« 2023. [Mrežno]. Available: https://www.mongodb.com/mern-stack. [05.09.2023.]

[5] M. Pereša, »JavaScript,« 2015.. [Mrežno]. Available: https://repozitorij.unipu.hr/islandora/object/unipu:495/datastream/PDF. [18.09.2023.]

[6] MDN contributors, »Mozzila.org,« 8 July 2023. [Mrežno]. Available: https://developer.mozilla.org/en-US/docs/Web/JavaScript. [05.09.2023.]

[7] Meta Platforms, Inc., »React Legacy,« 2023. [Mrežno]. Available: https://legacy.reactjs.org/docs/getting-started.html. [05.09.2023.]

[8] OpenJS Foundation, »NodeJs,« 2023. [Mrežno]. Available: https://nodejs.org/en/about. [12.09.2023.]

[9] »Npm docs,« 2023.. [Mrežno]. Available: https://docs.npmjs.com/about-npm. [12.09.2023.]

[10] JavaTpoint, »JavatPoint,« 2021. [Mrežno]. Available: https://www.javatpoint.com/expressjs-tutorial. [07.09.2023.]

[11] K. Ukey, »Dev.to,« 19 March 2022. [Mrežno]. Available: https://dev.to/kunalukey/how-to-setup-and-deploy-a-mern-stack-project-for-free-5acl. [07.09.2023.]

[12] auth0, »JWT,« 2023. [Mrežno]. Available: https://jwt.io/introduction. [12.09.2023.]

[13] OneLogin, »OneLogin,« 2023. [Mrežno]. Available: https://www.onelogin.com/learn/authentication-vs-authorization . [12.09.2023.]

[14] P. Vaughan, »Hubspot,« 10 June 2021. [Mrežno]. Available: https://blog.hubspot.com/blog/tabid/6307/bid/32566/why-landing-pages-are-an-indispensable-part-of-marketing.aspx. [12.09.2023.]

[15] MDN contributors, »Mozilla,« 2023 5 July. [Mrežno]. Available: https://developer.mozilla.org/en-US/docs/Learn/CSS/CSS_layout/Responsive_Design. [12.09.2023.]

SAŽETAK

Naslov : Web portal za organizaciju putovanja uz savijete lokalnog stanovništva

Glavni zadatak ovog rada bio je izraditi web aplikaciju na koju se mogu prijaviti dvije vrste korisnika: recenzenti i putnici. Putnik bira destinaciju te ima mogućnost planiranja putovanja u smislu lokacija koje planira posjetiti. Recenzent je osoba koja je lokalni stanovnik mjesta koje se može izabrati kao destinacija za putovanje. Web aplikacija se sastoji od tri stranice: početne, glavne i profilne stranice. Početna stranica je prvo što korisnici ugledaju prilikom dolaska na stranicu. Sve objave korisnika se nalaze na glavnoj stranici kojoj korisnici imaju pristup samo ako se pravilno registriraju i prijave. Profilna stranica, uz informacije korisnika sadrži i sve objave na koje je putnik pozitivno reagirao ili u slučaju recenzenta, sve objave koje je kreirao. Cijela aplikacija je responzivna. Web aplikacija je napravljena primjenom MongoDB, ExpressJs, ReactJs i NodeJs tehnologija.

Ključne riječi: Baza podataka, MERN, Putovanje, Recenzija, Web aplikacija.

ABSTRACT

Title: Web portal for travel organization with advice from local population

Main task of this paper was to create a web application for two types of users: reviewers and travelers. The traveler chooses the destination and has the option of planning the trip in terms of the locations he plans to visit. The reviewer is a person who is a local resident of the place that can be chosen as a travel destination. The web application consists of three pages: home page, main page and profile page. The home page is the first thing that users see when they visit the site. All user posts are on the main page, which users only have access to if they register and log in properly. The profile page, in addition to the user's information, also contains all the posts traveler liked, or in the case of the reviewer, all the posts he created. The entire application is responsive. The web application was created using MongoDB, ExpressJs, ReactJs and NodeJs technologies.

Keywords: Database, MERN, Review, Travel, Web application.

ŽIVOTOPIS

Autor ovog završnog rada je Valentin Romić, student 3. godine prijediplomskog stručnog studija Računarstvo na fakultetu elektrotehnike, računarstva i informacijskih tehnologija u Osijeku. Rođen 16.02.1999. u Vinkovcima. Osnovnu školu je završio u školi „Josip Kozarac“ Soljani. Nakon osnovne škole, nastavlja školovanje u srednjoj strukovnoj tehničkoj školi u Županji. Tijekom studiranja u Osijeku obavlja stručnu praksu u Reroot agency. Reroot se bavi izradom web aplikacija koristeći ReactJs kojim je napravljena i ova aplikacija za organizaciju putovanja uz savijete lokalnog stanovništva.

_____

Potpis autora