

# Primjena geolokacijskih servisa u razvoju web aplikacija

---

**Turković, Vladimir**

**Undergraduate thesis / Završni rad**

**2021**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **University of Zagreb, Faculty of Organization and Informatics / Sveučilište u Zagrebu, Fakultet organizacije i informatike**

*Permanent link / Trajna poveznica:* <https://urn.nsk.hr/urn:nbn:hr:211:855493>

*Rights / Prava:* [Attribution 3.0 Unported/Imenovanje 3.0](#)

*Download date / Datum preuzimanja:* **2025-03-20**



*Repository / Repozitorij:*

[Faculty of Organization and Informatics - Digital Repository](#)



**SVEUČILIŠTE U ZAGREBU  
FAKULTET ORGANIZACIJE I INFORMATIKE  
VARAŽDIN**

**Vladimir Turković**

**PRIMJENA GEOLOKACIJSKIH SERVISA U RAZVOJU  
WEB APLIKACIJA**

**ZAVRŠNI RAD**

**Varaždin, 2021.**

**SVEUČILIŠTE U ZAGREBU**  
**FAKULTET ORGANIZACIJE I INFORMATIKE**  
**V A R A Ź D I N**

**Vladimir Turković**

**Matični broj: 44260**

**Studij: Poslovni sustavi**

**PRIMJENA GEOLOKACIJSKIH SERVISA U RAZVOJU**  
**WEB APLIKACIJA**

**ZAVRŠNI RAD**

**Mentor:**  
**dr. sc. Matija Novak**

**Varaždin, travanj 2021.**

**Vladimir Turković**

### **Izjava o izvornosti**

Izjavljujem da je moj završni/diplomski rad izvorni rezultat mojeg rada te da se u izradi istoga nisam koristio drugim izvorima osim onima koji su u njemu navedeni. Za izradu rada su korištene etički prikladne i prihvatljive metode i tehnike rada.

*Autor potvrdio prihvaćanjem odredbi u sustavu FOI-radovi*

---

## Sažetak

Tema ovog rada je primjena geolokacijskih servisa u praktične svrhe prilikom izrade web aplikacija. U radu će se usporediti jedni od najpoznatijih geolokacijskih servisa današnjice Mapbox, Leaflet i Google Maps. Opisane su prednosti i nedostaci prilikom njihove implementacije u web aplikacije. Nakon usporedbe svih servisa fokus je na odabranom servisu Google Maps. Nakon toga slijedi detaljna razrada servisa, od njegovog nastanka, razvoja, pa do današnjeg oblika. Opisana je njegova dostupnost, dokumentacija, funkcionalnost i najčešći oblici primjene. Nakon obrade odabranog servisa predstavljena je web aplikacija, čiji je smisao bila demonstracija upotrebe geolokacijskog servisa u praktičnoj primjeni. Tamo je opisana implementacija servisa, asinkrono povezivanje servisa, razni načini upotrebe i mogućnosti koje servis pruža.

**Ključne riječi:** geolokacijski servisi, javascript, node.js, web aplikacija, programiranje

# Sadržaj

1. Uvod.....	1
2. Korištene tehnologije .....	2
3. Geolokacijski servisi .....	3
3.1 Opća upotreba .....	3
3.2 Sigurnost i privatnost .....	4
3.3 Praktična primjena .....	5
3.3.1 Geolokacijski objekt .....	5
3.3.2 Dobivanje trenutne lokacije .....	5
3.3.3 Praćenje trenutne lokacije.....	6
3.3.4 Opis lokacije .....	6
3.3.5 Upravljanje pogreškama .....	6
4. Primjeri geolokacijskih servisa .....	7
4.1 Mapbox.....	7
4.1.1 Prednosti .....	7
4.1.2 Nedostaci.....	8
4.2 Leaflet.....	8
4.2.1 Značajke .....	8
4.2.2 Primjena .....	8
4.3 Google Maps .....	9
4.3.1 Primjena .....	9
5. Usporedba geolokacijskih servisa.....	11
5.1 Usporedba servisa .....	11
5.1.1 Jednostavnost upotrebe.....	11
5.1.2 Cijena i plaćanje .....	12
5.1.3 Značajke .....	13
5.2 Odabir servisa.....	14
6. Izrada web aplikacije .....	15
6.1 Opis aplikacije.....	15
6.2 Model baze podataka.....	15
6.3 Povezivanje i korištenje baze podataka .....	18
6.4 Uloge korisnika u aplikaciji .....	19
6.4.1 Ne registrirani korisnik .....	19
6.4.2 Naručitelj.....	20
6.4.3 Vozač.....	22
6.4.4 Menadžer.....	25
6.4.5 Administrator.....	30

6.5 Upotreba geolokacijskog servisa u aplikaciji .....	32
6.5.1 Trenutna lokacija i vremenska prognoza.....	32
6.5.2 Automatska sugestija lokacija.....	33
6.5.3 Prikaz i personalizacija karte.....	33
6.5.4 Geokodiranje .....	34
6.5.5 Prikaz oznaka na karti.....	35
6.5.6 Kreiranje rute .....	35
6.5.7 Napredna tražilica .....	36
7. Prednosti i nedostaci u radu sa odabranim servisom.....	37
7.1 Prednosti.....	37
7.2 Nedostaci.....	37
8. Zaključak.....	39
Popis literature .....	40
Popis slika .....	41
Popis tablica.....	42

# 1. Uvod

Geolokacijski servisi su danas neophodni u funkcioniranju većine aplikacija. Veliki izbor i jasne dokumentacije su pridonijele njihovoj popularnosti u programerskom svijetu. Iako veću primjenu i ulogu imaju u razvoju mobilnih aplikacija, u ovom radu će fokus biti na razvoju web aplikacija.

Današnje web aplikacije su složene i idu u veliku širinu prilikom pružanja svojih usluga. Često se viđa da aplikacije koje imaju neku sasvim nepovezanu svrhu, na jedan ili na drugi način pronalaze koristan i prirodan način da ukomponiraju geolokacijski servis u svoju uslugu.

Veliku ulogu u popularnosti servisa je pridonio razvoj informatičkih tehnologija, posebno AJAX (eng. Asynchronous JavaScript and XML). AJAX je skup tehnologija koje se koriste na korisničkoj strani kako bi se ostvarilo asinkrono funkcioniranje aplikacije. Asinkrono znači da se dio koda koji je deklariran kao asinkron ne mora čekati, već se kod nastavlja dalje izvoditi kao da je sinkroniziran. Tek kada se asinkrona funkcija izvrši omogućava se pristup podacima koje je ta funkcija izvodila. Time se omogućava korisniku da nastavi koristiti aplikaciju iako ona do kraja nije izvedena. Upravo taj način je najbolji i preporučen način korištenja geolokacijskog servisa.

U radu će se često spominjati riječ servis, a o čemu je zapravo riječ? Servis ili aplikacijsko programsko sučelje (eng. Application Programming Interface, API) je programsko rješenje koje je omogućeno svim programerima na korištenje. Svrha servisa je da se programerima pružaju već gotova programska rješenja široke uporabe. Servisi pružaju razna kompleksna rješenja na koja programerima nije isplativo izdvojiti vrijeme za razvoj. Svima su dostupni isti servisi, a način na koji će se funkcije i podaci tog servisa iskoristiti čini diferencijaciju između aplikacija. Svaki servis ima skup pravila i specifikacija koje su obuhvaćene u dokumentaciji samog servisa. Dokumentacija servisa igra ima veliki značaj u korištenju servisa, jer je upravo ona prvi kontakt i način komunikacije stvaratelja servisa i krajnjeg korisnika. Što je dokumentacija jasnija, to je korištenje servisa lakše. Naravno da korištenje tuđeg rješenja dolazi sa određenom cijenom. Tako da svaki servis u svom opisu sadrži plan članstva odnosno cijenu usluge. Cijene su najčešće skalirane po broju zahtjeva koji se pošalje preko određenog servisnog ključa (engl. API key) koji je dodijeljen prilikom registracije korisničkog profila. Mali broj zahtjeva koji je inače dovoljan prilikom razvoja aplikacije je često besplatan, a kada se aplikacija javno objavi potrebno je odabrati prikladan plan naplate ovisno o broju zahtjeva.



## 2. Korištene tehnologije

U ovom poglavlju će se nabrojati i kratko opisati programski alati i pomoćni alati koji su bili korišteni za demonstraciju upotrebe geolokacijskih servisa u web aplikaciji. Korištene tehnologije i alati su:

- HTML (eng. HyperText Markup Language) je standard za stvaranje osnovne strukture svake web stranice. Hypertext se odnosi na povezivanje web stranica, a markup nam govori da je riječ o opisnom jeziku. Sastoji se od raznih elemenata koji se navode unutar oznaka (eng. Tags). [1]
- CSS (eng. Cascading Style Sheets) je standardni jezik za uređivanje izgleda HTML elemenata na web stranicama. Namijenjen je isključivo za obradu HTML i XML dokumenata. Isto je jedan od najosnovnijih jezika sa kojim se programeri suočavaju kada je u pitanju dizajn web stranica. [2]
- JavaScript je jedan od najpoznatijih programskih jezika za izradu dinamičkih web stranica na korisničkoj strani. Korišten je u ovom radu kao glavni alat za odvijanje dinamičkog dijela aplikacije. Spada u kategoriju interpreterskih, pravovremeno sastavljenih programskih jezika. Najčešće se koristi za izradu web stranica, a koriste ga i neki drugi programski alati kao npr. Node.js, Apache CouchDB, Adobe Acrobat itd. [3]
- Node.js je programski alat baziran na JavaScript okolini koji služi za razvoj i povezivanje baze podataka aplikacija. Njegova mogućnost upravljanja paketa pruža programerima mogućnost implementacije raznih modula koji pomažu prilikom razvoja serverske strane aplikacija. [4]
- Express je najpoznatiji mikrookvir napisan u JavaScriptu, a pomaže u razvoju baze podataka aplikacije u Node.js okolini. Popularan je zbog svoje jednostavnosti i minimalizma. [4]
- MongoDB je tip NoSQL baze podataka koja je orijentirana na dokumente. NoSQL govori da je riječ o bazi koja je horizontalno skalarna, ima dinamične sheme nestrukturiranih podataka bez tradicionalnih struktura tablica. Najveća prednost je kompatibilnost sa JSON (eng. JavaScript Object Notation) formatom zapisa podataka koji je popularan u modernim aplikacijama. [5]
- Mongoose je paket koji služi za jednostavniju upotrebu MongoDB-a u Node.js-u. Nudi rješenja za kreiranje shema prilikom modeliranja baze podataka i razne mogućnosti kao npr. provjere valjanosti, izrada upita, kukice poslovne logike itd. [6]
- JQuery je JavaScript biblioteka poznata pod izrazom „piši manje, čini više“. Glavna svrha jQuery-a je olakšati korištenje JavaScript-a. Više linija koda u JavaScript-u, jQuery zamjeni sa jednom linijom koda koristeći već ugrađene metode. [7]
- Bootstrap je najpoznatiji CSS mikrookvir za razvoj osjetljivih web stranica. Bootstrap je zapravo velika kolekcija HTML, CSS i JavaScript koda koji olakšava posao programeru prilikom dizajn stranice. [8]

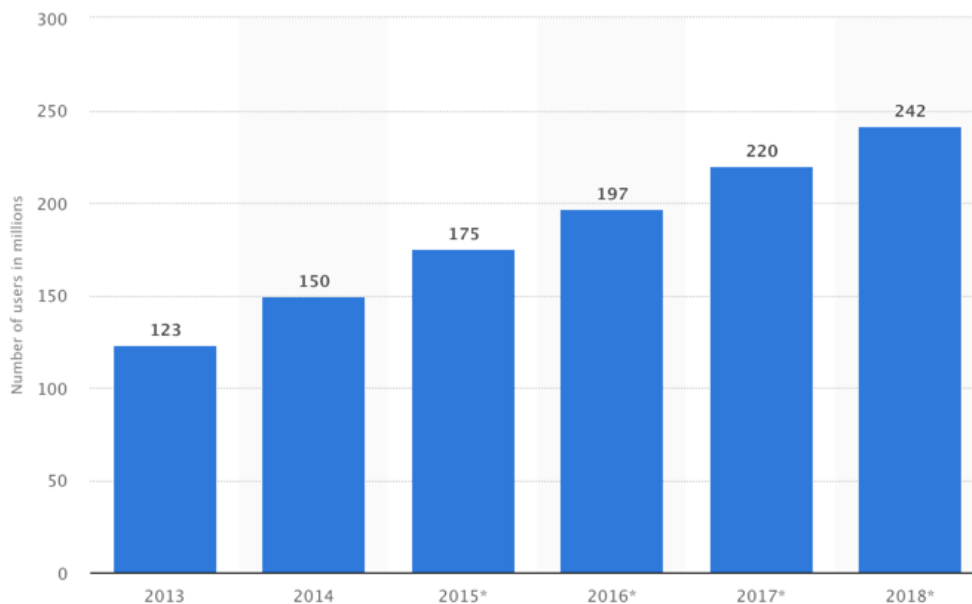
## 3. Geolokacijski servisi

Geolokacijski servis je vrsta servisa koji služi za pružanje geolokacijskih informacija. Riječ servis nam govori da je u pitanju vanjski program bez kojega naš uređaj ili preglednik ne bi mogao sam obavljati funkciju. Servisi su najčešće napravljeni od strane firme koje su specijalizirane u određenom području, a nerijetko imaju monopol nad podacima koje pružaju. Zbog toga su u nekim slučajevima programeri prisiljeni koristiti jedan od ponuđenih servisa. [9]

Geolokacijski servisi rade na temelju GPS-a (eng. Global Positioning System), mobilnih telekomunikacijskih tornjeva, wi-fi pristupnih točaka i interneta. Korištenjem jedne tehnologije ili povezujući više njih pružaju lokaciju elektroničkog uređaja koji se njome koristi. Lokacija uređaja je definirana zemljopisnim koordinatama, tj. zemljopisnom dužinom i širinom. Osim lokacije pružaju razne druge korisne informacije uređaja kao npr. zemlju, regiju, grad, vremensku zonu, poštanski kod itd. [10]

### 3.1 Opća upotreba

Eksplozivni rast i razvoj informatičke tehnologije koju su predvodili divovi poput Google-a, Facebook-a, Amazon-a je prouzročio da se osobni podaci klijenta tj. pojedinca izrazito cijene. U tom kontekstu se savršeno pronašla upotreba geolokacijskog servisa tj. praćenje lokacija i kretanja korisnika. Lokacija korisnika i njegovo dnevno kretanje predstavlja jednu od esencijalnih informacija za takva poduzeća.



Slika 1. Rast upotrebe geolokacijskih servisa od 2013. do 2018. [11]

Podjela glavnih kategorija upotrebe geolokacijskih informacija [12]:

- marketing
- financije
- navigacija
- promet

- **znanstvena i geografska istraživanja**

Kada je u pitanju marketing, radi se o personaliziranim reklamama ovisno o lokacijama korisnika. Podaci se svakodnevno prikupljaju preko satelita ili mobilnih telefonskih tornjeva kako bi se stvorila slika o dnevnim navikama korisnika. Tako se bolje cilja pojedinca, a sama kampanja marketinga je efektivnija. Prednosti geolokacijskog marketinga su [13]:

- Visoko ciljana publika
- Prikupljanje podataka o trenutnim kupcima
- Privlačenje novih kupaca
- Mjerenje rezultata

Upotreba geolokacijskih podataka u financijske svrhe ima izrazitu važnost. Primjerice prilikom korištenja kartice za plaćanje, financijske institucije mogu usporediti lokaciju korisnika sa lokacijom mjesta plaćanja. Ako se lokacije ne poklapaju, kartica će biti deaktivirana. U slučaju da se lokacije poklapaju korisnik izbjegava razna ometanja prilikom korištenja usluge koje se inače javljaju, ako su zabilježene neobične aktivnosti. Isto tako geolokacija omogućava bankovnim klijentima praktičan način da pristupe lokalnim bankomatima nakon radnog vremena putem mobilnih telefona umjesto kartica [13].

Navigacija je najočiglednija upotreba geolokacijskih servisa. Kad god se koriste aplikacije za prikaz rute upotrebljavaju se geolokacijske informacije. Pa se tako koriste na primjer za:

- Promet
- Pronalaženje lokacija
- Putovanja
- Fitness
- Taxi
- Dostava
- Proširena stvarnost

Navigacija je prvotni razlog zbog čega je geolokacija razvijena, ali zanimljivo je kako danas ima tek sporednu ulogu kad se priča o omjeru upotrebe geolokacijskih servisa.

## **3.2 Sigurnost i privatnost**

Iako prednosti geolokacijskih servisa premašuju nedostatke, bitno je razmotriti kako to utječe na korisnike. Ažuriranjem prava privatnosti korisnika poduzeća koja skupljaju geolokacijske podatke moraju jasno naglasiti koje se točno informacije koriste i za što se koriste. Poduzeća isto tako moraju zaštititi te podatke i onemogućiti svojim zaposlenicima neprimjeren pristup. Također korisnici ne žele da se njihovi jednom podijeljeni podaci koriste za više namjena bez njihovog znanja.

Poduzeća koja žele izgraditi vjerne korisnike, moraju iskreno obavijestiti svoje korisnike o tome kako se njihove informacije koriste. Na primjer prilikom instalacije aplikacije korisnici su obaviješteni kako će se njihovi geo podaci koristiti i imaju mogućnost zabrane dijeljenja lokacije. [14]

## 3.3 Praktična primjena

U ovom poglavlju će se opisati najpoznatiji način sa kojim se programeri susreću kada je riječ o implementiranju geolokacijskog servisa u stolne i mobilne aplikacije.

### 3.3.1 Geolokacijski objekt

Jednostavan i besplatan geolokacijski servis dostupan je u gotovo svim preglednicima. Pristupa mu se pomoću „navigator.geolocation“ objekta. Bitno je napomenuti da se njime može jedino služiti u sigurnosnom načinu tj. ako je korisnik to dozvolio. Ovdje je prikazan kod koji to provjerava:

```
if('geolocation' in navigator) {  
    /* geolokacija je dostupna */  
} else {  
    /* geolokacija nije dostupna */  
}
```

### 3.3.2 Dobivanje trenutne lokacije

Za dobivanje trenutne lokacije uređaja koristi se metoda „getCurrentPosition()“. Ona inicira asinkroni zahtjev prema hardveru za pozicioniranje kako bi dobili ažuriranu informaciju o lokaciji uređaja. Kada je lokacija određena, izvršava se definirana funkcija povratnog poziva.

```
navigator.geolocation.getCurrentPosition((position) => {  
    getCoords(position.coords.latitude, position.coords.longitude);  
});
```

Gore prikazana funkcija „getCoords“ će se izvesti kada je lokacija uređaja dobivena. Može se i definirati druga funkcija povratnog poziva koja bi se odvijala u slučaju da dođe do pogreške. Treći parametar ove metode je objekt koji definira odgovor funkcije. On nije obavezan, a u njemu se može definirati maksimalna starost dobavljene lokacije, vrijeme čekanja zahtjeva i razinu točnosti određivanja položaja.

Primjer poziva funkcije:

```
function success(position) {  
    getCoords(position.coords.latitude, position.coords.longitude);  
}  
  
function error() {  
    alert('Position not available!');  
}  
  
const options = {enableHighAccuracy: true, maximumAge: 40000,  
    timeout: 30000};  
  
const watchID = navigator.geolocation.watchPosition(success,  
    error, options);
```

Prema zadanim postavkama namješten je što brži odgovor neovisno o preciznosti i starosti lokacije. To je korisno kada se traži brz odgovor neovisno o preciznosti.

### 3.3.3 Praćenje trenutne lokacije

Ako se trenutna lokacija uređaja promjeni, to se može saznati uz pomoć „watchPosition()“ funkcije. Ona ima iste parametre kao i „getCurrentPosition“ funkcija. Funkcija povratnog poziva je pozvana više puta, kako bi se ažurirala promjena položaja uređaja ili kako bi se pružila preciznija lokacija zato što se koriste razne tehnike pri dobavljanju lokacije. Primjer korištenja funkcije:

```
const watchID = navigator.geolocation.watchPosition((position) => {  
    getCoords(position.coords.latitude, position.coords.longitude);  
});
```

Funkcija „watchPosition()“ vraća jedinstveni id koji se može koristiti za identifikaciju promatrača položaja. Vrijednost tog id-a zajedno se koristi sa funkcijom „clearWatch()“ da bi se zaustavilo gledanje korisnikove lokacije.

```
navigator.geolocation.clearWatch(watchID);
```

### 3.3.4 Opis lokacije

Lokacija uređaja opisana je koristeći se „GeolocationPosition“ objektom. „GeolocationPosition“ objekt sadrži samo dva podatka. Vrijednost koordinata koji opisuju trenutni položaj uređaja i vremensku oznaku koja predstavlja vrijeme u kojemu su podaci u položaju bili dohvaćeni. Instanca „GeolocationCoordinates“ sadrži razna svojstva, a najbitnija od njih su zemljopisna širina i dužina. Pa tako većina uspješnih povratnih poziva započinje slično kao ovaj primjer:

```
function success(position) {  
    const latitude = position.coords.latitude;  
    const longitude = position.coords.longitude;  
    // Učini nešto sa zemljopisnom dužinom i širinom }
```

Druga svojstva koje instanca „GeolocationCoordinates“ nudi su nadmorska visina, brzina, smjer i mjerenje točnosti navedenih podataka.

### 3.3.5 Upravljanje pogreškama

Funkcija povratnog poziva prilikom greške se inicira unutar funkcija „getCurrentPosition()“ i „watchPosition()“. Ona očekuje „GeolocationPositionError“ objekt kao prvi parametar. Taj objekt se sastoji od koda koji opisuje tip greške i poruke koja opisuje grešku, a čitljiva je za korisnika. Jedan od načina korištenja:

```
function errorCallback(error) {  
    alert(`ERROR(${error.code}): ${error.message}`);  
};
```

## 4. Primjeri geolokacijskih servisa

U ovom poglavlju će biti opisani neki od najpoznatijih geolokacijskih servisa koji se ističu po svojim različitim karakteristikama, te tako programerima stvaraju širi spektar opcija prilikom odabira servisa za svoju aplikaciju. Glavni kriterij podijele geolokacijski servisa je vlasništvo, pa ih tako dijelimo na:

- Javni – npr. Leaflet, OpenStreetMaps
- Privatni – npr. Mapbox, Bing maps, Apple maps, Google maps

### 4.1 Mapbox

Mapbox je američka firma koja pruža geolokacijske usluge, a svjetsku popularnost dostigla je 2010. godine. Firma ima više od 420 milijuna mjesečnih korisnika i osigurava stabilnu podršku za preko pet milijardi zahtjeva dnevno. Tajna njihovog uspjeha krije se u ciljanju tržišne niše kreiranja personaliziranih mapa koji je u vrijeme njihovog nastanka bio limitiran od strane Google-a. Mapbox je danas prihvaćen kao standard za izgradnju personaliziranih mapa koju koriste razna poznata poduzeća kao naprimjer DHL, Snapchat, The Weather Channel, Foresquare itd. [15]

Većina podataka koje Mapbox koristi su javno dostupni. Pružaju česta ažuriranja lokacijskih podataka koja se brzo mijenjaju. Njihova platforma se bazira na OpenStreetMap-i koja je otvorenog koda. Programerima omogućavaju da dodaju različite markere, linije, polilinijske, poligone i slojeve iz vanjskih izvora (u GeoJSON, GPX i drugim formatima). Mapbox tehnologija se temelji na Node.js jeziku, Mapniku (paket alata otvorenog koda za prikazivanje karata), GDAL (prevoditeljska biblioteka za rasterske i vektorske formate geoprostornih podataka) i Leaflet (JavaScript biblioteka za interaktivne karte). [15]

#### 4.1.1 Prednosti

Glavne prednosti su:

- Jedinствене mogućnosti prilagodbe - Recenzije pokazuju da je Mapbox prilagodljiviji od Google Mapsa. Iako puno sustava za mapiranje nudi gotovu kartu, Mapbox nudi potpunu slobodu koju programeri mogu iskoristiti prilikom izrade kako god žele. Mapbox omogućuje stvaranju stila karte koji odgovara profilu određene aplikacije. Programeri mogu postaviti fontove i shemu boja te dodati funkcionalnosti poput detaljnih uputa i informacija o terenu. [16]
- SDK-ovi otvorenog koda - Mapbox SDK-ovi (eng. Software development kit) su otvorenog koda. Mapbox dijeli svoj kod na GitHub-u, tako da ga se uvijek može vidjeti, analizirati i poboljšati. Mnogi talentirani programeri aktivno doprinose bazi koda. Mapbox SDK temelje se na Mapbox GL Native-u. Ova biblioteka omogućuje ugradnju interaktivnih, prilagodljivih vektorskih karata u matične aplikacije na više platformi. [16]
- Izvanmrežne karte - Mapbox pruža veću fleksibilnost u pogledu izvan mrežnog načina rada naspram Google-a. Zahvaljujući korištenju vektorskih karata, Mapbox podržava izvan mrežne funkcije. Aplikacije stvorene s Mapbox Mobile SDK-ovima mogu preuzeti karte za odabrana zemljopisna područja za upotrebu kada uređaj nema mrežnu vezu.

Uz to, Mapbox-ovi mobilni SDK-ovi automatski predmemoriraju resurse tražene tijekom normalne upotrebe. [16]

## 4.1.2 Nedostaci

Glavni nedostatak je relativno slaba pokrivenost. Mnogo je mjesta na kojima Google ima bolju pokrivenost od OpenStreetMap usluga. Razlog tome je uglavnom činjenica da Google Maps dobiva konstantna ažuriranja. Primjerice, u Indiji i Izraelu Google Maps bit će bolja opcija. [16]

## 4.2 Leaflet

Leaflet je JavaScript biblioteka otvorenog tipa koja služi za pružanje geolokacijskih informacija i mapiranje aplikacija. Prvi put izdan 2011. godine. Podržava većinu mobilnih i stolnih platformi, uključujući HTML5 i CSS3. Neke od poznatih firmi koje koriste Leaflet su: Foursquare, Pinterest, Flickr. [17]

Leaflet omogućuje programerima bez geografske pozadine da vrlo lako kreiraju web karte na javnom poslužitelju. Omogućuje učitavanje podataka o značajkama iz GeoJSON datoteka, oblikovanje i stvaranje interaktivnih slojeva, poput oznaka s iskačućim prozorima i drugo. Razvio ga je Vladimir Agafonkin, koji se pridružio Mapboxu 2013. godine. [17]

### 4.2.1 Značajke

Leaflet izvorno podržava slojeve usluge web-mapa (WMS), slojeve GeoJSON-a i vektorske slojeve. Putem dodataka podržane su mnoge druge vrste slojeva. Kao i ostale knjižnice web-mapa, osnovni model prikaza koji implementira Leaflet jedna je osnovna karta, plus jedan ili više prozirnih slojeva, s jednim ili više vektorskih objekata prikazanih na vrhu. [18]

Elementi Leaflet-a [17]:

- Rasterske vrste (TileLayer i ImageOverlay)
- Tipovi vektora (Put, poligon i određene vrste kao što je Krug)
- Grupirane vrste (LayerGroup, FeatureGroup i GeoJSON)
- Upravljanje (zumiranje, pregled slojeva, itd.)

Također postoje razne klase korisnih programa poput sučelja za upravljanje projekcijama, transformacijama i interakciju s DOM-om (Document Object Model). Leaflet ima osnovnu podršku za nekoliko GIS (Geographic Information System) standardnih formata, dok su ostali podržani u dodacima. [19]

### 4.2.2 Primjena

Tipična upotreba Leaflet-a uključuje vezivanje elementa "mape" na HTML element kao što je div. Slojevi i oznake zatim se dodaju elementu karte. Jedinствена značajka Leaflet-a je prikazivanje karata u nesferičnim Mercatorovim projekcijama. Sama knjižnica servisa dostupna je putem varijable „L”. [19]

```
// kreiranje nove mape u "map" div-u
// postavljanje pogleda na određenoj poziciji
var map = L.map('map').setView([51.505, -0.09], 13);
```

```
// dodvanja OpenStreetMap sloja  
L.tileLayer('http://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png')  
.addTo(map);
```

## 4.3 Google Maps

Prva asocijacija kada je pokrenuta tema geolokacije je Google Maps, daleko najpoznatiji i najdominantniji geolokacijski servis. Usluga Google Maps razvijena je od strane Google-a. Nudi satelitske snimke, zračne fotografije, mape ulica, 360 stupnjeva interaktivni panoramski pogled na ulice, prometne uvjete u stvarnom vremenu, planiranje ruta raznih vrsta prijevoza itd. Ima milijardu mjesečni korisnika u 2020. godini. [20]

Google Maps započeo je kao C++ program za radne površine tvrtke „Where 2 Technologies“ . U listopadu 2004. tvrtku je kupio Google koji ju je pretvorio u web aplikaciju. Nakon dodatnih akvizicija tvrtke za vizualizaciju geoprostornih podataka Google Maps pokrenut je u veljači 2005. Korisničko sučelje usluge koristi JavaScript, XML i Ajax. Google Maps nudi API koji omogućuje ugrađivanje karata na web stranice i nudi lokator za tvrtke i druge organizacije u brojnim zemljama širom svijeta. Google Map Maker omogućio je korisnicima da zajednički prošire i ažuriraju mapiranje usluge širom svijeta. [20]

Satelitski prikaz Google Maps-a je pogled odozgo prema dolje ili iz ptičje perspektive. Većina slika visoke razlučivosti gradova su zračne snimke snimljene iz zrakoplova koji lete na 240 do 460 m, dok je većina ostalih slika sa satelita. Mnogo dostupnih satelitskih snimaka nije starije od tri godine i redovito se ažurira. Google Maps prethodno je koristio varijantu Mercatorove projekcije i stoga nije mogao točno prikazati područja oko polova. U kolovozu 2018., verzija Google Mapsa za stolna računala ažurirana je tako da prikazuje 3D globus, a u postavkama je još uvijek moguće vratiti se na 2D kartu. [20]

Google Maps za Android i iOS uređaje objavljen je u rujnu 2008. godine i sadrži GPS detaljnu navigaciju, zajedno s namjenskim značajkama pomoći pri parkiranju. U kolovozu 2013. utvrđeno je da je najpopularnija svjetska aplikacija za pametne telefone, a više od 54% svjetskih vlasnika pametnih telefona koristilo ju je barem jednom. [20]

### 4.3.1 Primjena

U ovom poglavlju će se objasniti osnovni način na koji se poziva Google Maps servis i implementacija karte u aplikaciji. Za funkcioniranje Google servisa potrebno je prvo kreirati servisni ključ (eng. API key). Zatim se kreira html „script tag“ kojem je dodijeljena poveznica na Google Maps servis. Poveznica sadrži servisni ključ i ime funkcije povratnog poziva. U funkciji povratnog poziva pomoću naredbe „new google.maps.Map“ se kreira karta koja je vezana na html objekt sa identifikatorom „map“. Pod atribut „center“ se unose koordinate centra karte, a „zoom“ određuje razinu udaljenosti pogleda karte. [21]

JavaScript:

```
// Kreiranje html script tag-a  
let script = document.createElement('script');  
script.src = 'https://maps.googleapis.com/maps/api/js?key=  
API_KEY&callback=initMap';  
script.async = true;
```



```
// Dodavanje funkcije povratnog poziva na „window“ objekt
window.initMap = function() {
    map = new google.maps.Map(document.getElementById("map"), {
        center: { lat: -34.397, lng: 150.644 },
        zoom: 8, });};
// Dodavanje „script“ elementa na html head
document.head.appendChild(script);
```

### Html:

```
<body>
    <div id="map"></div>
</body>
```

## 5. Usporedba geolokacijskih servisa

Kako bi projekt bio što uspješniji, potrebno je odabrati servis koji najbolje odgovara njegovim potrebama. Tržište je puno raznovrsnih geolokacijskih servisa, a u svrhu usporedbe odabrana su tri konkurentna servisa.

### 5.1 Usporedba servisa

U ovom poglavlju će se napraviti usporedba prije navedenih servisa. Servisi će biti uspoređivani u područjima jednostavnosti upotrebe, dokumentacije, cijeni i plaćanju, te značajkama.

#### 5.1.1 Jednostavnost upotrebe

Suvremeni servisi mogu biti izrazito napredni i komplicirani za korisnike koji se po prvi puta susreću sa njima. Zato je jednostavnost upotrebe i implementacije osnovnih funkcionalnosti jedna od najbitnijih značajki servisa. Ono što je potrebno prije korištenja bilo kojeg servisa je kreiranje korisničkog računa na službenoj stranici servisa. To je bitno zbog dodijele servisnog ključa pomoću kojega se prati ostvareni promet. Servisni ključ je potrebno navesti prilikom svakog poziva prema servisu. U nastavku će biti navedeni primjeri koda za prikaz karti pojedinih servisa. [22]

##### Leaflet:

```
var map = L.map('map').setView([51.505, -0.09], 13);
L.tileLayer('https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png', {
  attribution: '&copy; <a
href="https://www.openstreetmap.org/copyright">OpenStreetMap</a>
contributors'
}).addTo(map);
L.marker([51.5, -0.09]).addTo(map)
  .bindPopup('A pretty CSS3 popup.<br> Easily customizable.')
  .openPopup();
```

##### Mapbox:

```
<script src='https://api.mapbox.com/mapbox-gl-js/v2.1.1/mapbox-
gl.js'></script>
<div id='map' style='width: 400px; height: 300px;'></div>
<script>
mapboxgl.accessToken = 'MAPBOX.TOKEN';
var map = new mapboxgl.Map({
  container: 'map',
  style: 'mapbox://styles/mapbox/streets-v11',
```

```
center: [-74.5, 40],
zoom: 9}); </script>
```

### Google Maps:

```
<body><div id="map"></div></body>

<script
src=https://maps.googleapis.com/maps/api/js?key=YOUR_API_KEY&callback=initM
ap&libraries=&v=weekly async></script>

<script>

let map;

function initMap() {

map = new google.maps.Map(document.getElementById("map"), {
center: { lat: -34.397, lng: 150.644 }, zoom: 8, });}</script>
```

## 5.1.2 Cijena i plaćanje

Leaflet pošto je JavaScript biblioteka otvorenog koda ne naplaćuje usluge korištenja. Mapbox i Google su privatne firme i one imaju razne planove naplaćivanja na koje se korisnici mogu odlučiti ovisno o upotrebi.

Tablica 1. Cjenik Mapbox-a [21]

Broj mjesečnih zahtjeva	Cijena po 1000 zahtjeva
0 - 50.000	Besplatno
50.000 - 100.000	5\$
100.000 - 200.000	4\$
200.000 - 1.000.000	3\$
1.000.000 +	Potreban ugovor

Tablica 2. Cjenik Google-a [22]

Tip servisa	Cijena po 1000 zahtjeva
Static Maps	2\$
Dynamic Maps	7\$
Directions	5\$
Geocoding	5\$
Autocomplete	17\$
Geolocation	5\$

Mapbox nudi besplatno korištenje do 50.000 zahtjeva neovisno o tipu servisa, a Google nudi besplatanih 200\$ svaki mjesec zavisno o servisu.

### 5.1.3 Značajke

Značajke ili mogućnosti koje servisi imaju stvara najveću razliku između njih. Svaki servis ima neke prednosti po kojima se ističe od drugih.

Prva značajka po kojoj se Leaflet ističe je veličina biblioteke od samo 39 kilobajta. On je dizajniran sa fokusom na jednostavnost, izvedbu i upotrebljivost. Djeluje učinkovito na svim glavnim radnim površinama i mobilnim platformama. Leaflet je tip otvorenog koda i ima podršku velike zajednice korisnika. Zahvaljujući njima Leaflet pruža na stotine nastavaka koje si korisnici mogu odabrati. Svi nastavci su spremljeni u javno dostupnom GitHub repozitoriju sa imenima održavatelja. [17]

Sa druge strane Mapbox i Google Maps su tipovi servisa sa zatvorenim pristupom kodu. Ne nude mogućnost nadograđivanja koda vlastitog servisa, ali ono što nude je i više nego dovoljno za potrebe korisnika.

Mapbox se ističe po beskonačnim mogućnostima stiliziranja karata. Cilj Mapbox-a je da korisnici postignu zadivljujuću, korisnu, relevantnu i kontekstualnu kartu. To upravo mogu postići uz pomoć komponenata u Mapbox studiju. Dizajneri mogu prilagoditi prikaz karte kako bi istaknuli značajke koje odgovaraju njihovoj publici. Karte bi trebale učiniti više od komuniciranja lokacije. Trebale bi dopuniti ono za što se koriste i olakšati razumijevanje prikazanih podataka. Pomoću prilagođenih karata izrađenih za određeni slučaj korištenja, dizajneri se mogu usredotočiti na čitljivost i prilagodbu estetike usredotočujući se na najrelevantnije informacije. [16]

U Mapbox studiju se nudi šest stilova karata sa zadanim stilom i prilagodbama dizajniranim za specifične slučajeve korištenja. Od navigacije do nekretnina, fitness aplikacija do vizualizacija podataka. Osnovne mape započinju postupak prilagodbe svojim jedinstvenim, zadanim stilom, pa je odabir pravog stila presudan za postavljanje tona dizajna i početak rada. Odabir stila teksta daje karakter karti i jednostavan je prekidač koji kartu brzo čini specifičnom. Prilagođavanje oblikovanja slova i fonta također može dodati više konteksta karti. Što je nešto čitljivije, to je više na vizualnoj hijerarhiji. Mapbox studio ima ugrađen preko 45 stilova teksta. [16]

Google Maps Platform dio je Google Cloud platforme. Karte, rute i mjesta uključuju sve usluge potrebne za razvoj interaktivnih karata i njihovo integriranje u vlastite programe. Google Maps može u web stranice ugraditi statičke i dinamičke karte, slike ulica sa pogledima od 360 stupnjeva. Rute se mogu koristiti za izračun i proračun ruta, uzimajući u obzir trenutne informacije o prometu. [21]

Google Places pruža podatke o lokaciji za više od 150 milijuna lokacija širom svijeta. Više od desetak API-ja i SDK-a pruža jednostavnu integraciju s aplikacijama i web uslugama za Android i iOS. [20]

Google Maps Matrix prikazuje korisniku najbližu rutu između željenih lokacija.

Geolocation servis omogućuje precizno određivanje mjesta putnika. Da bi to učinio, servis obrađuje podatke koje je Google prikupio na Wi-Fi i GSM mrežama. Obrnuto geokodiranje određuje početnu adresu. [20]

Google Places servis pruža detaljne informacije o ustanovama i lokacijama. Automatsko dovršavanje adrese olakšava unos i omogućuje samo adrese koje stvarno postoje. To

sprječava slanje vozača na nepostojeći kućni broj. Nadalje, korisnik ne mora nužno navesti adresu, jer baza podataka Google Places sadrži ne samo adrese već i više od 100 milijuna zanimljivih mjesta poput zračnih luka, kazališta i stadiona. [20]

Google Maps Directions prikazuje stvarno vrijeme putovanja potrebno vozilu da bi stiglo do lokacije, a utvrđuje se uzimajući u obzir trenutnu prometnu situaciju. [20]

## 5.2 Odabir servisa

Nakon detaljne usporedbe puno je lakše doći do zaključka uvezi odabira geolokacijskog servisa. Iako su sva tri vrhunski servisi koji prednjače u svom spektru djelovanja, Google Maps je ipak najbolji odabir kada su u pitanju rute i precizni, ažurni podaci vezani za rute. Za svrhu izrade aplikacije ovog rada je izabran Google Maps servis.

Iako se čini da Google Maps postoji od početka vremena, to nije slučaj kada je riječ o ažuriranju podataka i pružanju novih funkcionalnosti. Pogotovo pri promjeni njegovih cjenovnih politika.

Google Maps Platform, može se pohvaliti sa 99% pokrivenosti svijeta i više od milijardu aktivnih mjesečnih korisnika. Dugo je to industrijski standard za integraciju karata koji koriste Bolt, Uber, Allianz i mnoga druga velika i manja poduzeća. Korištenjem flote svojih satelita, automobila s Street View-om i Android uređaja, kao i angažiranjem lokalnih suradnika, Google pruža ogromne količine detaljnih i točnih podataka uz ažuriranja u stvarnom vremenu. A proizvodi servisi poput Google Maps, Routes i Places namijenjeni su udovoljavanju potrebama bilo koje industrije, ako poduzeće može podnijeti cijenu. [20]

## 6. Izrada web aplikacije

Programski kod „Geo Path“ aplikacije se nalazi na javnom repozitoriju GitHub-a kojemu se može pristupiti pomoću sljedećeg linka: <https://github.com/vturkovic/geo-path>. Na GitHub-u se nalaze detaljne upute kako se može pokrenuti aplikacija i poveznica za online pristup aplikaciji. Aplikacija je zaštićena uz pomoć GNU General Public License v3.0

### 6.1 Opis aplikacije

Za svrhu obrade teme ovog rada napravljena je web aplikacija sa fokusom maksimalnog iskorištenja geolokacijskih servisa. Naziv aplikacije je „Geo Path“ iz čega se lako može naslutiti svrha. „Geo Path“ je aplikacija namijenjena za poduzeća koja u svom poslovanju pružaju mogućnost dostave. Aplikacija služi kao vizualni alat za menadžere poduzeća koji pomaže u kreiranju dostavljачkih ruta. Radi potpunosti aplikacije implementirana je mogućnost kreiranja narudžbi od strane naručitelja.

Uloge korisnika su:

- Neregistrirani korisnik
- Registrirani korisnik : Naručitelj, Vozač, Moderator / Menadžer ,Administrator

Za dinamički dio aplikacije je korišten Javascript i jQuery. Pozivi prema servisu i bazi su ostvareni uz pomoć AJAX tehnologije koji uvelike utječe na tečnost aplikacije. Serverska strana je odrađena uz pomoć Node.js-a i Express-a, a tip baze je NoSql tj. MongoDB koji je ostvaren uz pomoć Mongoose shema. Za vizualni dio aplikacije korišteni su CSS i Bootstrap.

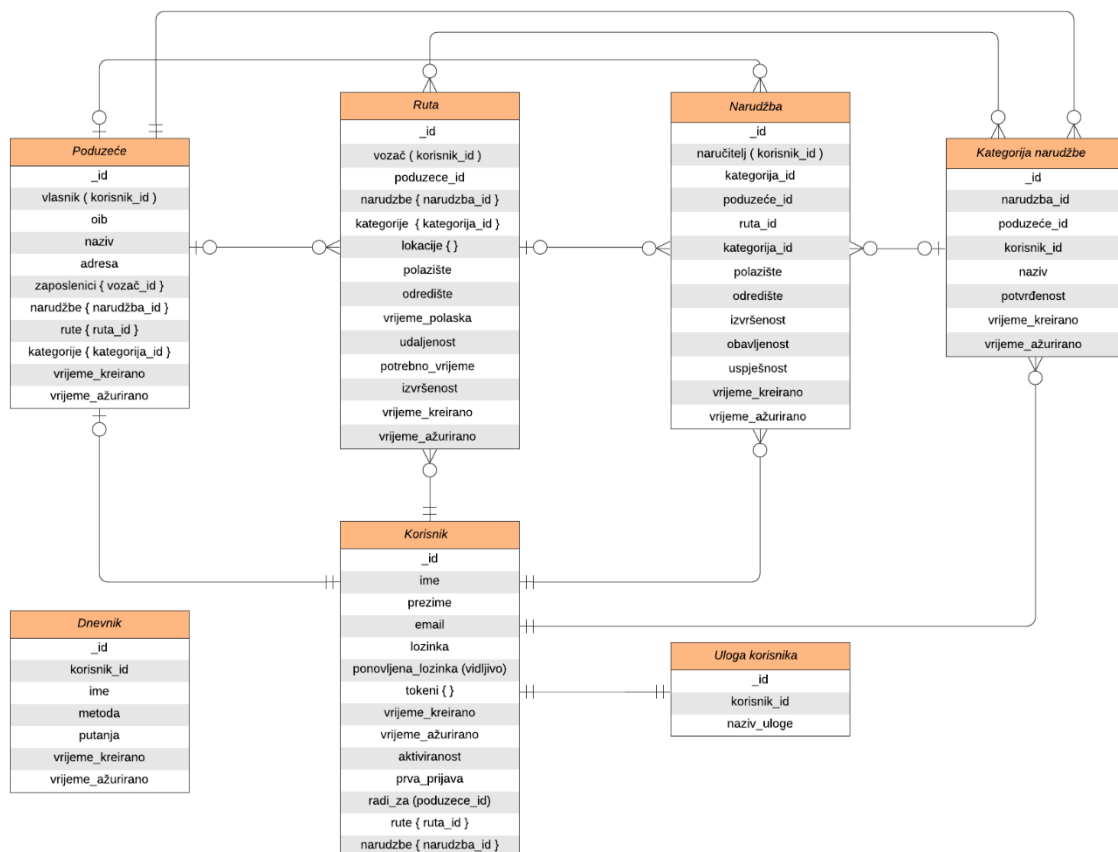
### 6.2 Model baze podataka

Izrada modela baze podataka je prva stvar koja je potrebna prilikom izrade složene web aplikacije. Model pruža orijentir za programera koji mu uvelike olakšava zadatak prilikom programiranja. Isto tako dobro organizirana baza ne stvara prepreke prilikom skaliranja količine podataka koji se unose u bazu, što izravno utječe na brzinu pristupa i jednostavnost upotrebe podataka.

Kako je već rečeno, za izradu baze podataka ove aplikacije korišten je „NoSQL“ mehanizam baze. „NoSQL“ se odnosi na nerelacijski mehanizam koji se razlikuje od standardnog relacijskog ili tabličnog unosa podataka. Prednosti „NoSQL“ pristupa su jednostavnost dizajna, jednostavnije „vodoravno“ skaliranje klastera, lakša kontrola nad dostupnošću i ograničavanje nepodudaranja objektno-relacijske impedancije. Baza je konkretno realizirana uz pomoć MongoDB baze podataka i Mongoose pomoćnog alata za izradu shema.

ER model baze podataka

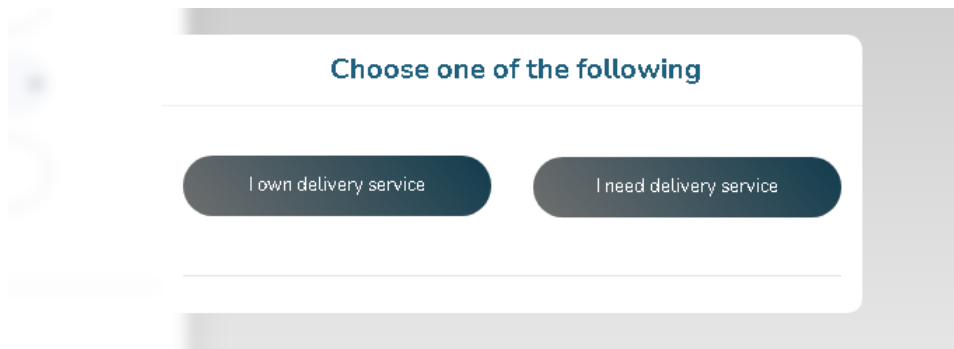
Model entiteta i veza (ili ER model) opisuje međusobno povezane entitete u bazi podataka. Osnovni ER model sastoji se od različitih entiteta, koji klasificiraju interesne stvari i određuje odnose koji mogu postojati između entiteta. Skup atributa koji opisuju određeni entitet ili objekt je smješten unutar jednog objekta. U MongoDB bazi podataka svaki objekt ima automatski generiran primarni ključ ili id.



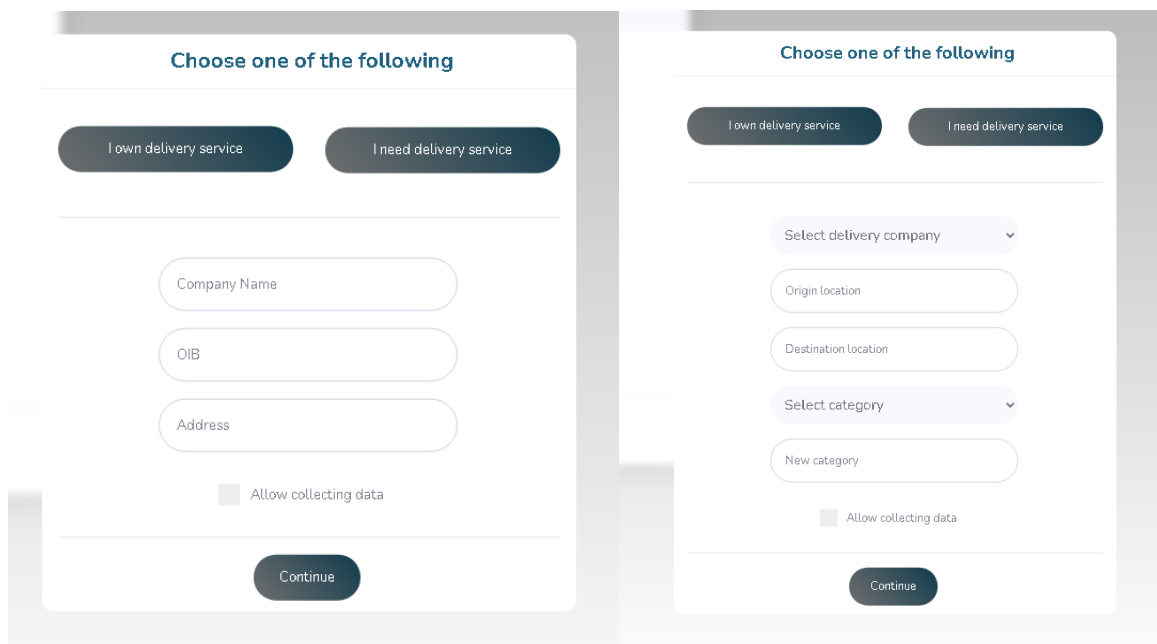
Slika 2. ER model baze podataka

Ovaj ER model je izrađen uz pomoć Lucidchart alata. Model sadrži sedam entiteta: korisnik, uloga korisnika, poduzeće, narudžba, kategorija narudžbe, ruta i dnevnik. Sve kreće od tablice korisnik. Prilikom registracije stvara se novi objekt korisnika kojem je dodijeljen jedinstveni id. Svaki korisnik mora unesti ime, prezime, email, lozinku i ponovljenu lozinku. Kod unosa tih podataka izvršavaju se razne validacije kako na korisničkoj tako i na serverskoj strani. Nakon toga je na email korisnika poslan jedinstveni aktivacijski link sa tokenom, kojeg korisnik mora u određenom vremenu kliknuti tj. aktivirati kako bi atribut „aktivirano“ postao istinit, a korisnikov račun postao aktivan. Čim je kreiran korisnik odmah je kreiran i objekt uloge korisnika koji je vezan na tog korisnika. Zadana uloga koja je dodijeljena svakom korisniku prilikom registracije je „korisnik“.

Prilikom prve prijave korisnike dočeka skočni prozorčić u kojem donose odluku o kreiranju poduzeća ili kreiranju narudžbe. U slučaju kreiranja poduzeća njima je automatski dodijeljena uloga moderatora tj. menadžera, a u slučaju kreiranja narudžbe uloga im ostaje „korisnik“. Korisnikov atribut „prva prijava“ poprima vrijednost datuma i vremena prve prijave. On služi kako bi se skočni prozorčić za diferencijaciju korisnika pojavio samo prilikom prve prijave.



Slika 3. Skočni prozorčić - odabir



Slika 4. Skočni prozorčić - forma menadžer / naručitelj

Po što se radi o prvoj prijavi korisnika potrebno je prihvatiti korištenje podataka. Prilikom ispunjavanja forme o poduzeću popunjavaju se naziv, oib i adresa poduzeća. Dovođenjem forme se kreira poduzeće koje je vanjskim ključem vezano na korisnika. Korisniku je u tom trenutku dodijeljena uloga menadžera, a sa ulogom menadžera on ima pogled moderatora u aplikaciji. Kod kreiranja rute menadžer ispunjava formu u kojoj su polazište, odredište, vrijeme polaska i odabir vozača obavezni atributi. Moguće je unesti do 10 lokacija rute što je limitirano od strane Google-ovog servisa. Udaljenost (izražena u kilometrima) i potrebno vrijeme za obavljanje rute (izraženo u minutama) je automatski generirano uz pomoć servisa i spremljeno u rutu. Nakon izvršenja forme ruta je kreirana i vezana na određenog vozača vanjskim ključem.

Kod otvaranja pogleda vozača dolazi do izražaja atribut vrijeme polaska rute zato što su po njemu sve dodijeljene rute tom vozaču filtrirane kako bi mu se prikazala sljedeća ruta. Ruta na sebe ima vezane narudžbe i kategorije narudžbi koje je prethodno dodijelio menadžer. Vozač kako izvršava određene narudžbe ispunjava formu o izvršenosti rute. Vozač može izvršiti rutu bez da je obavio sve narudžbe. U tom slučaju narudžbe koje nije obavio postaju neuspješno izvršene.



Dnevnik služi za sastavljanje statistika koje su vidljive vlasniku aplikacije tj. administratoru. Unos u bazu dnevnika se događa prilikom bilo kojeg poziva, bilo prema vanjskom servisu ili prema bazi podataka. Zapisuje se tko je podnio zahtjev, kada je podnesen zahtjev, metoda zahtjeva i putanja odnosno ruta zahtjeva.

## 6.3 Povezivanje i korištenje baze podataka

Kako je već prije spomenuto, za povezivanje i korištenje baze podataka je upotrijebljen Node.js sa Express-om. To je postignuto uz pomoć ove naredbe:

```
const express = require("express");  
const app = express();
```

Uz pomoć ove naredbe je podešena lokacija javnog statičnog dijela aplikacije za express i podržani zapis JSON formata:

```
app.use(express.static(publicDirectoryPath));  
app.use(express.json());
```

Nadalje Express omogućava i olakšava korištenje usmjerivača. Pa je tako za svaki entitet modela napravljen poseban usmjerivač.

```
const userRouter = require("./routers/user");  
const orderRouter = require("./routers/order");  
const companyRouter = require("./routers/company");  
const roleRouter = require("./routers/role");  
const logRouter = require("./routers/log");  
const routeRouter = require("./routers/route");  
const orderCategory = require("./routers/order-category");
```

```
app.use(userRouter, orderRouter, companyRouter, roleRouter, logRouter,  
routeRouter, orderCategory);
```

Samo spajanje na bazu je odrađeno uz pomoć Mongoose alata. Njegova svrha je olakšavanje korištenja MongoDB-a, tako što ima pojednostavljene naredbe i kreiranje shema. Spajanje na bazu uz pomoć Mongoose-a:

```
const mongoose = require("mongoose");  
mongoose.connect(process.env.MONGODB_URL, {  
  useNewUrlParser: true,  
  useCreateIndex: true,  
  useFindAndModify: false,  
});
```

Primjer mongoose sheme za korisnika:

```
const userSchema = new mongoose.Schema({
  firstName: {type: String, required: true, trim: true},
  lastName: {type: String, required: true, trim: true},
  email: {type: String, unique: true, required: true, trim: true,
    lowercase: true, validate(value) {if (!validator.isEmail(value)) {throw new
    Error("Email is invalid");}},},},
  password: {type: String, required: true, trim: true},},
  repeatPassword: {type: String, required: true, trim: true},
  confirmed:{type: Boolean, default: false},
  address:{type: String},
  tokens: [{token: {type: String, required: true}},},},},
  firstLogin:{type: String, default: ""},
  worksFor: {type: String}
},{timestamps: true});
```

Primjer rute za kreiranje novog korisnika:

```
router.post("/users", async (req, res) => {
  const user = new User(req.body);
  try {
    await user.save();
    const token = await user.generateAuthTokenRegistration();
    sendConfirmationEmail (user.email, user.firstName, token);
    res.status(201).send({ user, token});
  } catch (e) {
    res.status(400).send(e);}});
```

## 6.4 Uloge korisnika u aplikaciji

Kao što je već spomenuto postoji pet uloga korisnika u „GeoPath“ aplikaciji. Ne registrirani korisnik, naručitelj, vozač, menadžer i administrator. U nastavku će biti prikazani dijagrami korištenja za svaku pojedinu ulogu.

### 6.4.1 Ne registrirani korisnik

Aplikacija je limitirana samo na registrirane korisnike. Pa tako ne registrirani korisnici imaju samo mogućnost registracije.

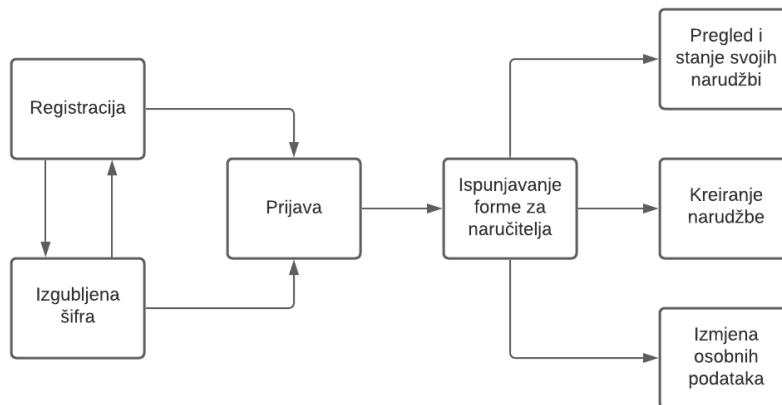


Slika 5. Dijagram korištenja - ne registrirani korisnik

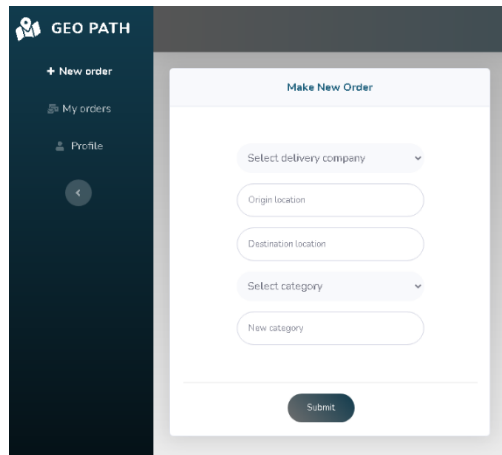
Kod ispunjavanja registracijske forme korisnik mora unesti ime, prezime, email, lozinku, ponovljenu lozinku. Isto tako mora ispuniti Google v2 recaptcha-u koja služi kao sigurnosna obrana od automatski generiranih kreiranja računa. Kod registracije se provjerava svaki unos. Oba unosa lozinki moraju biti identični i email mora biti jedinstven jer on služi za prijavu. Kada je registracija uspješno izvršena, prikazuje se skočni prozorčić koji obavještava o uspješnosti registracije i slanju aktivacijske poveznice na mail. Aktivacijska poveznica se sastoji od tokena koji je jedinstveno kreiran prilikom svake registracije i prijave. Poveznica ima rok trajanja od sat vremena. Klikom na poveznicu atribut aktiviranost postaje istinit tj. korisnik je aktiviran, a sama poveznica korisnika prebacuje na stranicu prijave.

### 6.4.2 Naručitelj

Naručitelj ili obični korisnik se postaje tako što se prilikom ispunjavanja početne forme odabire opcija kreiranja narudžbe. U tom procesu ispunjavaju se podaci o narudžbi što uključuje odabir jednog od dostupnih poduzeća dostave, adresa polazišta i odredišta narudžbe, te se unosi vlastita kategorija ili jedna od dostupnih za specifično poduzeće.



Slika 6. Dijagram korištenja - naručitelj

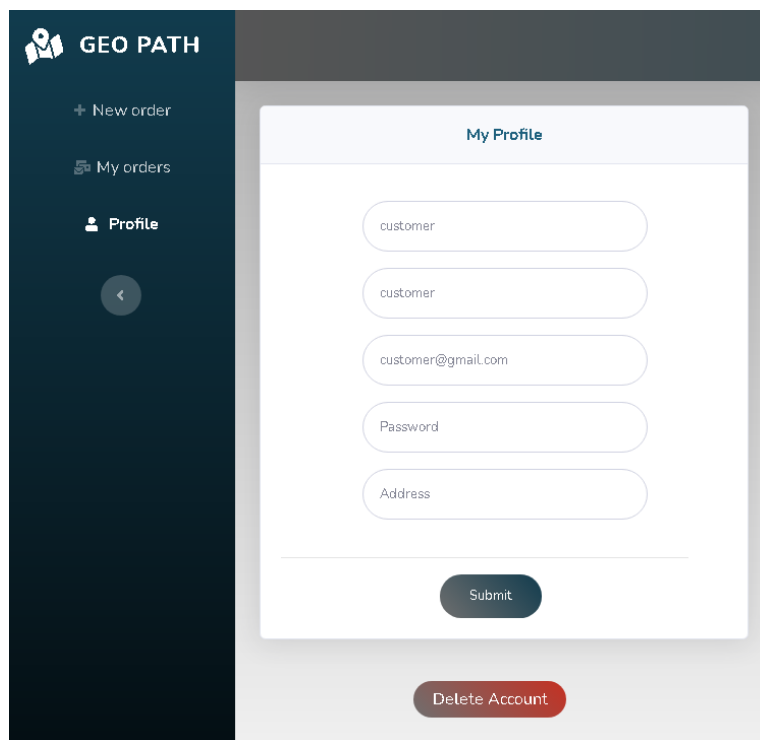


Slika 7. Stranica za kreiranje nove narudžbe

Completed	Processed	Unsuccessful	Company	Origin	Destination	Category	Created	Updated
true	true	true	poduzece1	Zadar, Croatia	Zagreb, Croatia	Izdjelivo	2021-02-12	2021-02-12
true	true	true	poduzece1	Gospić, Croatia	Zagreb, Croatia	Izdjelivo	2021-02-12	2021-02-12
false	false	false	poduzece1	Koronica, Croatia	Zagreb, Croatia	Izdjelivo	2021-02-12	2021-02-13
false	false	true	poduzece1	Knin, Croatia	Šibenik, Croatia	Izdjelivo	2021-02-12	2021-02-12
false	false	true	poduzece1	Gospić, Croatia	Zagreb, Croatia	Izdjelivo	2021-02-13	2021-02-13
false	false	false	poduzece1	Split, Croatia	Zagreb, Croatia	Izdjelivo	2021-02-13	2021-02-13
true	false	true	poduzece1	Crkvenica, Croatia	Zagreb, Croatia	Izdjelivo	2021-02-13	2021-02-13

Slika 8. Stranica za pregled vlastitih narudžbi

Naručitelj ima mogućnost pregleda vlastitih narudžbi koje se prikazuju u tabličnom formatu. Svaki tablični format u aplikaciji ima mogućnost odabira prikazivanja broja rezultata po stranici, odabir stranice rezultata, sortiranja uzlazno ili silazno po svakom stupcu i pretraživanje po ključnoj riječi. U ovoj tablici je fokus na trenutnom stanju narudžbe. Korisnik može vidjeti da li je narudžba provedena, neuspješno ili uspješno izvršena. Ostale informacije su: odabrano poduzeće, polazište, odredište, kategorija narudžbe, vrijeme kreiranja i ažuriranja narudžbe.

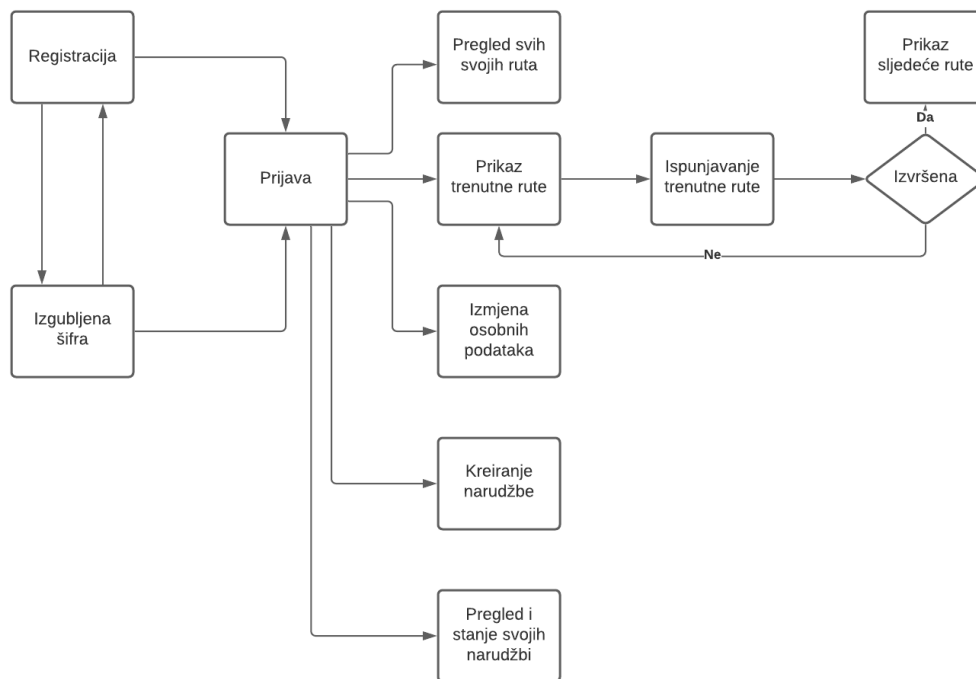


Slika 9. Stranica za uređivanje vlastitog korisničkog računa

Naručitelj kao i svaki korisnik ima mogućnost uređivanja vlastitog korisničkog računa. Pa tako može mijenjati podatke kao što su: ime, prezime, email, lozinka i adresa, a isto tako može i obrisati korisnički račun. Svaki korisnik još može mijenjati temu aplikacije i odjaviti se.

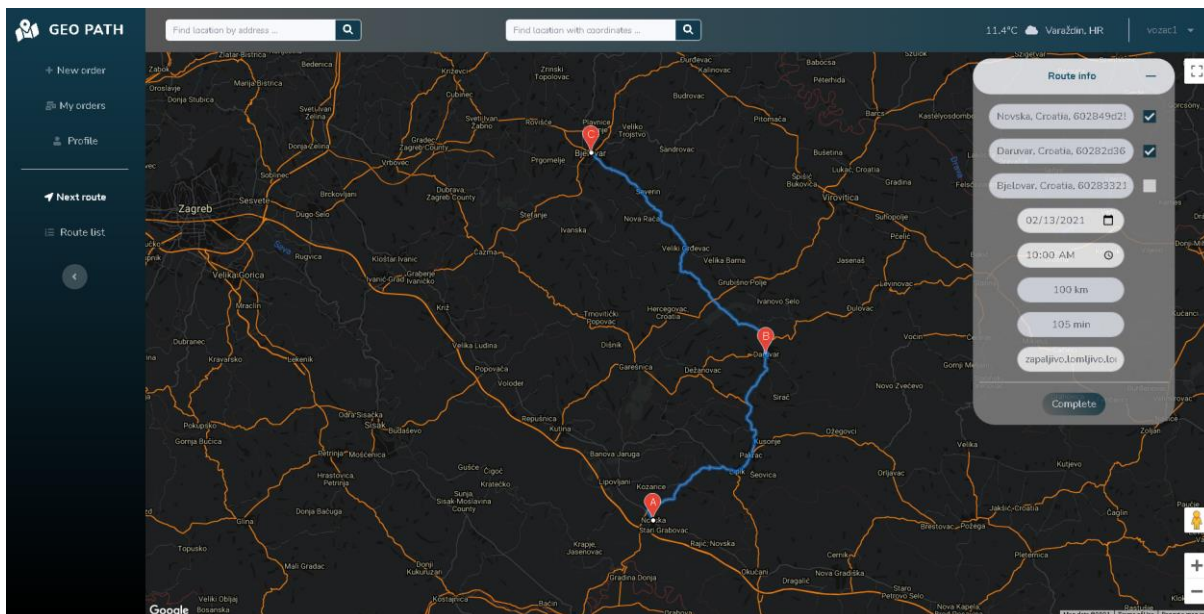
### 6.4.3 Vozač

Vozač je osoba koja se bavi dostavom narudžbi u određenom poduzeću. Vozači ne kreiraju sami svoje korisničke račune, već to isključivo rade menadžeri njihovog poduzeća. Menadžer nekog poduzeća kada kreirana korisnički račun vozača na njega veže id svog poduzeća. Na taj način se ostvaruje veza između vozača, poduzeća i menadžera, a izbjegava se mogućnost grešaka prilikom povezivanja. Vozač kada mu je kreiran korisnički račun mora aktivirati račun pomoću aktivacijskog mail-a ili to može umjesto njega napraviti menadžer.

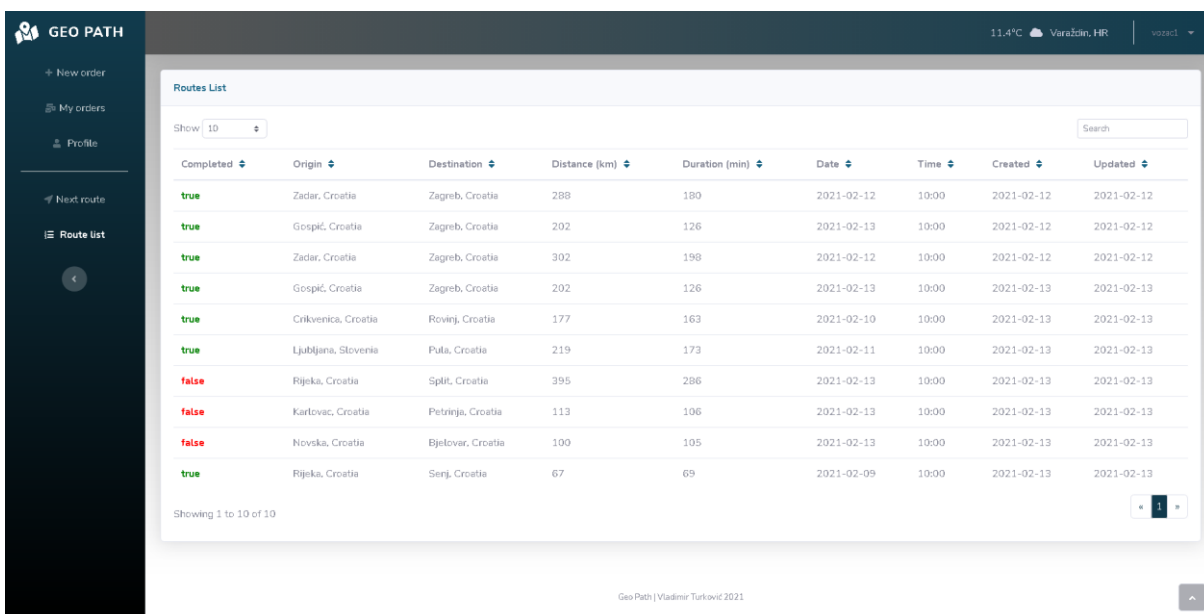


Slika 10. Dijagram korištenja - vozač

Dio ove aplikacije namijenjen za vozače fokusiran je na praktičnost. Vozač čim se prijavi u aplikaciju dočeka ga pogled na kartu sa ispisom trenutne rute. Svaka ruta ima obavezno dodijeljeno vrijeme polaska, pa se po tom vremenu filtriraju sve neizvršene rute i pronalazi se sljedeća ruta. U desnom kutu karte vozač ima jednostavan alat uz pomoć kojega označava izvršenost pojedine narudžbe. Ispunjavanjem kvadratića se pojedinoj narudžbi dodjeljuje vrijednost izvršenosti. Uz informacije o narudžbama, vozač vidi i informacije o ruti kao što su: datum i vrijeme polaska, udaljenost, potrebno vrijeme i kategorije narudžbi. Kada je vozač obavio sve lokacije tj. narudžbe klikom na „Complete“ potvrđuje izvršenost rute, te mu se automatski prikazuje sljedeća ruta. Vozač može obaviti rutu bez da je obavio pojedine narudžbe. U tom slučaju je ruta i dalje izvršena, ali narudžbe koje nisu bile označene tj. obavljene poprimaju vrijednost neuspješne izvršenosti. Takve narudžbe se menadžeru pokazuju žutom bojom markera na mapi o čemu će kasnije biti više govora.



Slika 11. Stranica trenutne rute vozača

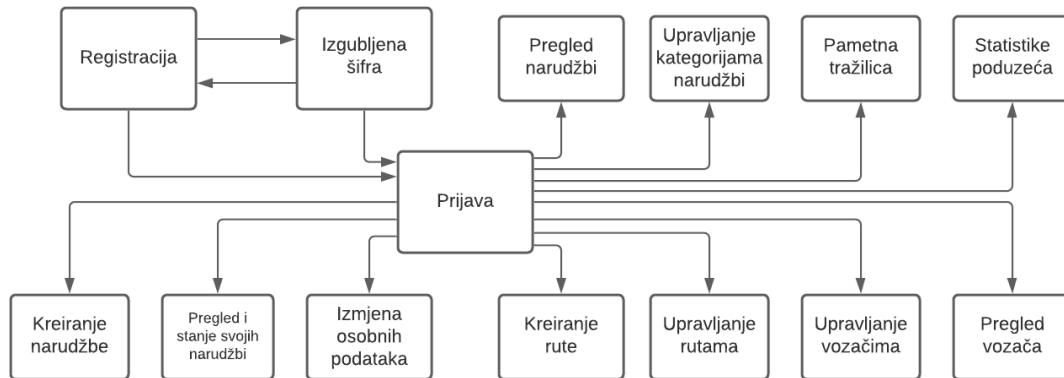


Slika 12. Stranica pregleda svih ruta pojedinog vozača

Vozač ima još mogućnost pregleda povijesti svih svojih ruta u tabličnom zapisu. Informacije o ruti su: izvršenost, polazište, odredište, udaljenost, potrebno vrijeme, datum i vrijeme polaska, datum i vrijeme kreiranja i ažuriranja rute.

## 6.4.4 Menadžer

Glavni fokus „GeoPath“ aplikacije je uloga menadžera. Menadžer je osoba u dostavljačkom poduzeću zadužena za organizaciju prijevoza. Ulogu menadžera može dobiti svaki korisnik koja prilikom prve prijave ispuni formu o kreiranju poduzeća. Kako bi ta forma bila uspješna potrebno je unesti ime, oib i adresu poduzeća. Nakon uspješnog ispunjenja forme kreirano poduzeće se veže na korisnika i korisnik poprima ulogu menadžera.



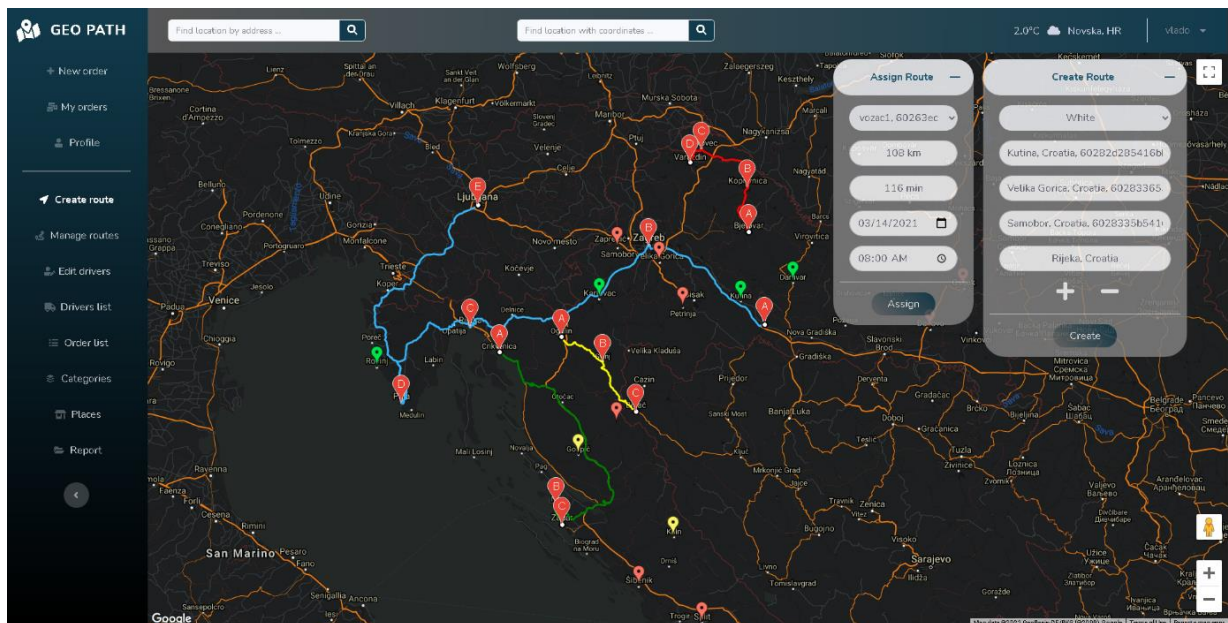
Slika 13. Dijagram korištenja - menadžer

Menadžer je zadužen za kreiranje dostavljačkih ruta, kreiranje korisničkih računa vozača, pregled statusa i statistika poduzeća. Menadžer kada se prijavi preusmjeren je na stranicu na kojoj je prikaza karta. Na toj karti su uz pomoć oznaka označene sve neizvršene narudžbe. Narudžbe u ovoj aplikaciji su podijeljene na:

- Izvršene (ne vide se na karti)
- Neizvršene:
  - Dodijeljene ruti (zelene boje)
  - Nedodijeljene ruti (crvene boje)
  - Neuspješno izvršene (žute boje)

Čim je nova narudžba kreirana ona je odmah vidljiva na menadžerovoj karti narudžbi. Kartata je centrirana po najranije kreiranoj i neobrađenoj narudžbi. U gornjem desnom kutu on ima dostupan alat uz pomoć kojega kreira rute. Potrebno je unesti najmanje dvije lokacije (polazište, odredište), a najviše deset lokacija, što je limitirano od strane Google Maps servisa. Unos lokacije odrađuje se tako da se klikne na oznaku narudžbe na karti. Klikom na oznaku se učitava adresa lokacije i id narudžbe koja je vezana za tu lokaciju. Moguće je i ručno unesti adresu lokacije, ali u tom slučaju niti jedna narudžba nije vezana za tu lokaciju. Ovdje je bitno napomenuti da je svaki unos adrese u aplikaciji standardiziran uz pomoć „Google Auto Complete Places „ servisa koja automatski nakon par slova nudi sugestiju adrese. Time su izbjegnuti razni problemi prilikom validacije unosa adresa i postignuto je da su sve adrese jednakog tipa unosa.

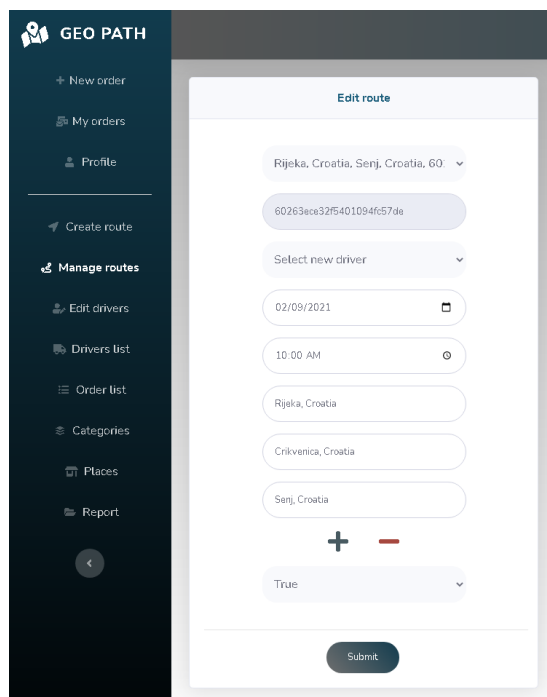




Slika 14. Stranica kreiranja rute

Kada je menadžer popunio sve podatke o ruti, klikom na „Create“ stvara se ruta. Tada se zadana ruta prikazuje na karti sa označenim lokacijama i orijentacijom smjera rute. Isto tako se pojavljuje novi izbornik koji služi za dodjelu rute vozaču. U tom izborniku menadžer odabire jednog od dostupnih vozača svojeg poduzeća. Unosi datum i vrijeme izvođenja rute, a udaljenost, potrebno vrijeme i kategorije narudžbi su već dodijeljeni ruti, te mu to služi kao korisna informacija.

Menadžer ima mogućnost upravljanja svim kreiranim rutama. Pojedini rutama može mijenjati dodijeljenog vozača, lokacije, vrijeme i datum izvedbe, a može umjesto vozača ispuniti izvršenost rute. Zatim ima mogućnost kreiranja i upravljanja korisničkim računima svojih vozača. Može mijenjati osobne podatke vozača, aktivirati njihove račune ili ih obrisati.



Slika 15. Upravljanje rutama

Zatim menadžer ima pregled svih vozača u tabličnom zapisu i informacije o vozačima. Također vidi da li su vozači aktivirali svoje korisničke račune. Dalje ima pregled svih narudžbi svojeg poduzeća u tabličnom zapisu i sve informacije vezane uz njih. Ovdje također može pratiti statuse narudžbi.

Order ID	Completed	Processed	Company	Origin	Destination	Created	Updated	Category	Unsuccessful
6026e9a55b31261e1c72280a	true	true	poduzece1	Zadar, Croatia	Zagreb, Croatia	2021-02-12	2021-02-12	lonjivo	true
6026e9b15b31261e1c72280d	true	true	poduzece1	Gospić, Croatia	Zagreb, Croatia	2021-02-12	2021-02-12	lonjivo	true
6026e9bc5b31261e1c722810	false	false	poduzece1	Korenica, Croatia	Zagreb, Croatia	2021-02-12	2021-02-13	lonjivo	false
6026e94b5b31261e1c722818	false	false	poduzece1	Krin, Croatia	Šibenik, Croatia	2021-02-12	2021-02-12	lonjivo	true
60282d1b5416bb73e9340734	false	false	poduzece1	Novska, Croatia	Zagreb, Croatia	2021-02-13	2021-02-13	brza dostava	false
60282d285416bb73e9340737	false	true	poduzece1	Kutina, Croatia	Zagreb, Croatia	2021-02-13	2021-02-13	lonjivo	false
60282d365416bb73e934073a	false	true	poduzece1	Daruvar, Croatia	Zagreb, Croatia	2021-02-13	2021-02-13	lonjivo	false
60282d5e5416bb73e934073d	false	true	poduzece1	Karlovac, Croatia	Čakovec, Croatia	2021-02-13	2021-02-13	brza dostava	false
60282e75416bb73e9340740	false	false	poduzece1	Zavojna ulica, Vrežbin, Croatia	Zagreb, Croatia	2021-02-13	2021-02-13	spora dostava	false
60282eb55416bb73e9340743	true	true	poduzece1	Rijeka, Croatia	Zagreb, Croatia	2021-02-13	2021-02-13	zapaljivo	false

Slika 16. Pregled narudžbi

Menadžer upravlja kategorijama narudžbi svojeg poduzeća. Za vrijeme izrade narudžbe naručitelj može izabrati postojeću kategoriju ili napraviti novu kategoriju u određenom poduzeću. Ta kategorija će biti dodijeljena njegovoj narudžbi, ali neće biti vidljiva

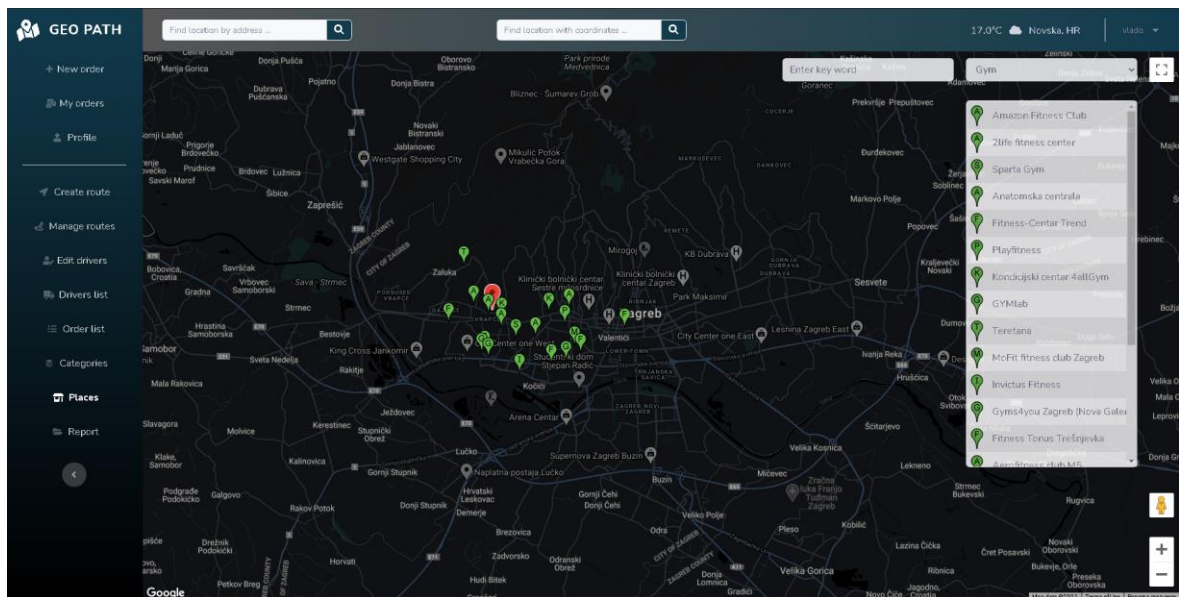
ostalim korisnicima prilikom kreiranja narudžbe sve dok ona ne bude prihvaćena. Menadžer može kreirati nove kategorije ili prihvatiti kategorije kreirane od strane korisnika. Tek kad ih je prihvatio te kategorije postaju vidljive ostalim korisnicima prilikom kreiranja narudžbe. U pregledu kategorija u tabličnom zapisu su ispisane sve informacije o kategorijama sa istaknutim statusom prihvaćenosti.

The screenshot displays the 'GEO PATH' web application interface. On the left is a dark sidebar with navigation links: 'New order', 'My orders', 'Profile', 'Create route', 'Manage routes', 'Edit drivers', 'Drivers list', 'Order list', 'Categories', 'Places', and 'Report'. The main content area is divided into three sections: 'Create New Category' (with a form for 'Category Name' and a 'Create' button), 'Confirm Category' (with a 'Select category' dropdown and a 'Confirm' button), and 'Order Category List'. The 'Order Category List' section features a table with columns for 'Category ID', 'Name', 'Created by', 'Created Date', and 'Status'. Below the table, it indicates 'Showing 1 to 10 of 9' items.

Category ID	Name	Created by	Created Date	Status
60263f3e32f5401094fc57d3	lomljivo	60263e4232f5401094fc57d3	2021-02-12	true
60263f4232f5401094fc5777	zapaljivo	60263e4232f5401094fc57d3	2021-02-12	true
60263f4832f5401094fc57f9	brza dostava	60263e4232f5401094fc57d3	2021-02-12	true
60263f5032f5401094fc57fb	spora dostava	60263e4232f5401094fc57d3	2021-02-12	true
60263f9232f5401094fc57fe	istra	60263e4232f5401094fc57d3	2021-02-12	false
60266defe06c0241806fb8c2	spora dostava 1	60266ce9e06c0241806fb8b8	2021-02-12	true
602848225416bb73e83407af	zapaljivo	602844695416bb73e83407a1	2021-02-13	false
6028648d5416bb73e8340896	nova kategorija	60263e4232f5401094fc57d3	2021-02-13	true
602864ad5416bb73e8340899	nova kategorija 1	60263e4232f5401094fc57d3	2021-02-13	true

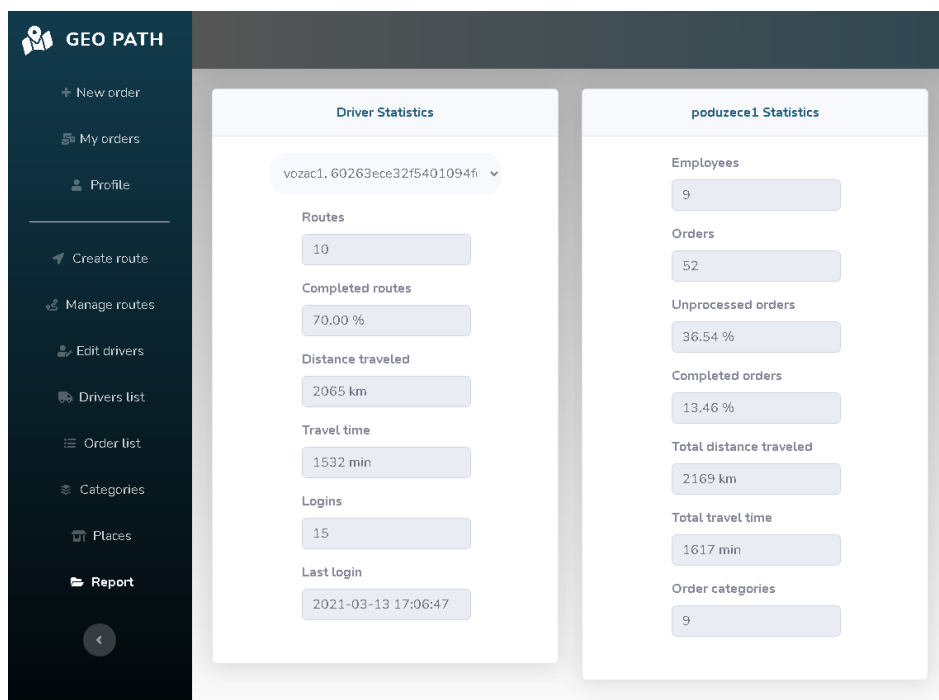
Slika 17. Pregled kategorija

Na stranici napredne tražilice menadžer ima mogućnost pretraživanja objekata na karti po ključnoj riječi i tipu. Nakon unesene ključne riječi i tipa objekta na karti se prikazuju oznake koji označavaju lokacije svih tih objekata. U desnom kutu se otvara tablica sa ispisom svih tih objekata, a klikom na pojedini objekt se karta centrirana na taj objekt i otvara se prozorčić sa osnovnim informacijama objekta. Karata je početno centrirana po menadžerovoj trenutnoj lokaciji.



Slika 18. Napredna tražilica

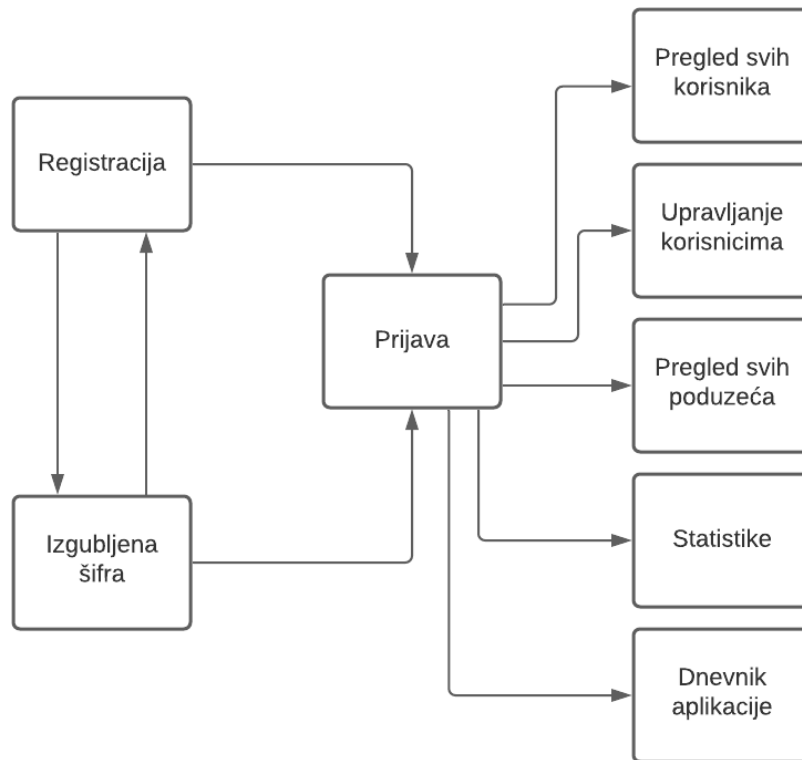
Naposlijetku menadžer ima pregled statistika vozača i poduzeća. Odabirom pojedinog vozača menadžer ima uvid u informacije o broju dodijeljenih ruta, postotak završenih ruta, ukupnu udaljenost i provedeno vrijeme u vožnji, te broj prijava i vrijeme zadnje prijave. U cjelokupnim statistikama vlastitog poduzeća menadžer vidi informacije o broju zaposlenika, broju narudžbi, postotak neprovedenih narudžbi, postotak izvršenih narudžbi, ukupnu udaljenost i provedeno vrijeme u vožnji svih vozača, te broj kategorija narudžbi.



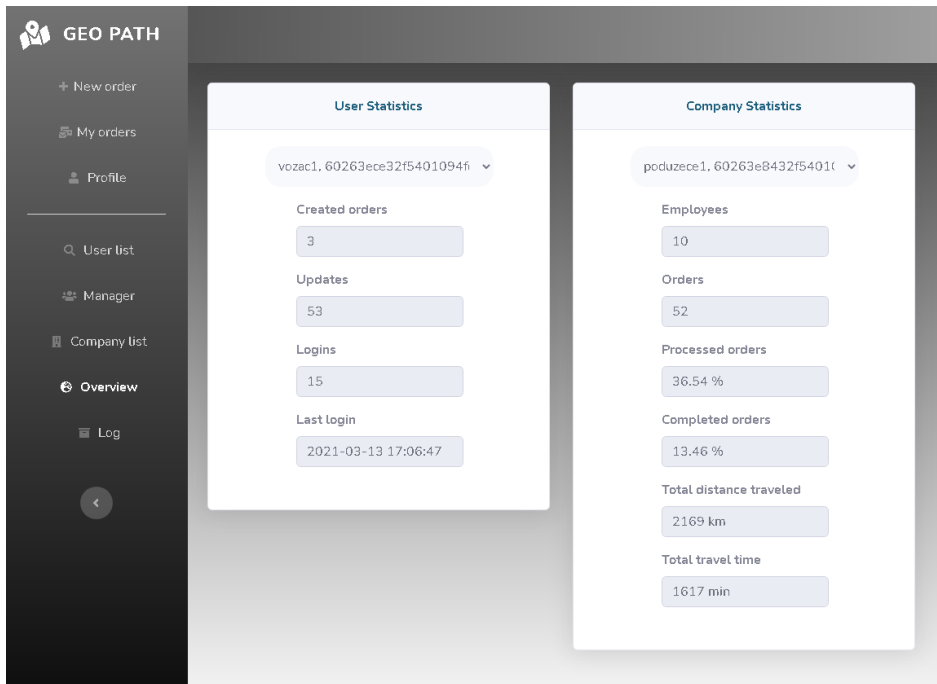
Slika 19. Statistike menadžera

## 6.4.5 Administrator

Administrator je samo jedan, on je vlasnik aplikacije. Ima uvid u sve korisnike i može upravljati svim korisnicima. Može mijenjati osobne podatke korisnika, aktivirati ili deaktivirati korisnike ili po potrebi i brisati korisnike. Također ima uvid u sva poduzeća i njihove informacije, te vidi statistike pojedinih korisnika i poduzeća.



Slika 20. Dijagram korištenja - administrator



Slika 21. Statistike korisnika i poduzeća

The screenshot shows an 'Activity Log' table with the following data:

Method	Path	Created At	User ID	First Name
GET	/orders/60211d1f618bd5734975028	2021-02-08	60211d1f618bd5734975028	antonio
GET	/orderCategory/602115c7aa9fe58ee0caea19	2021-02-08	60211d1f618bd5734975028	antonio
POST	/orderCategory/	2021-02-08	60211d1f618bd5734975028	antonio
POST	/orders	2021-02-08	60211d1f618bd5734975028	antonio
GET	/orders/60211d1f618bd5734975028	2021-02-08	60211d1f618bd5734975028	antonio
GET	/orders/60211d1f618bd5734975028	2021-02-08	60211d1f618bd5734975028	antonio
GET	/orderCategory/602115c7aa9fe58ee0caea19	2021-02-08	60211d1f618bd5734975028	antonio
GET	/orderCategory/602115c7aa9fe58ee0caea19	2021-02-08	60212dbab88b906808535355	frane
PATCH	/assess/	2021-02-08	60212dbab88b906808535355	frane
POST	/orderCategory/	2021-02-08	60212dbab88b906808535355	frane
POST	/orders	2021-02-08	60212dbab88b906808535355	frane
GET	/orders/60212dbab88b906808535355	2021-02-08	60212dbab88b906808535355	frane
GET	/orderCategory/602115c7aa9fe58ee0caea19	2021-02-08	60212dbab88b906808535355	frane
GET	/orderCategory/60211d1f618bd5734975028	2021-02-08	60212dbab88b906808535355	frane

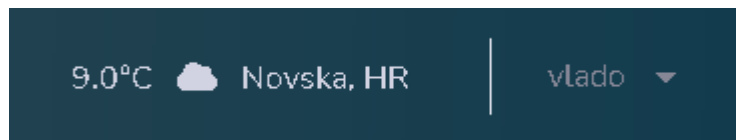
Slika 22. Dnevnik aplikacije

## 6.5 Upotreba geolokacijskog servisa u aplikaciji

### 6.5.1 Trenutna lokacija i vremenska prognoza

Prva primjena koja je jednaka za sve korisnike neovisno o ulozi je pronalaženje trenutne lokacije korisnika. To je jedna od prvih funkcija koja se izvodi nakon što se korisnik prijavi u aplikaciju. Pronalaženje lokacije se odvija uz pomoć funkcije „navigator.geolocation“ koja je dostupna u svim pretraživačima. Za djelovanje te funkcije potrebno je odobriti dijeljenje lokacije u pretraživaču. Djelovanje te funkcije je objašnjeno u poglavlju „Geolokacijski servis - Praktična primjena“. Dobivena lokacija je izražena u koordinatama kao geografska širina i dužina, te je spremljena u kolačić.

Prvo korištenje korisnikove lokacije odvija se za prikaze trenutne adrese u gornjem desnom kutu izbornika. Uz prikaz trenutne adrese prikazuje se i trenutna vremenska prognoza na tom području. To je ostvareno uz pomoć „OpenWeatherMap“ servisa. Za upotrebu tog servisa potrebno je otvoriti račun na službenoj stranici „OpenWeatherMap“ servisa i kreirati API ključ. Taj servis funkcionira na način da se u zahtjevu pošalju koordinate lokacije, a on zauzvrat vraća informacije o vremenu za tu lokaciju.



Slika 23. Prikaz trenutne lokacije i vremenske prognoze

Kod:

```
fetch(`http://api.openweathermap.org/data/2.5/weather?lat=${lat}&lon=${lng}
&appid=${api_key}&units=metric`)
  .then(response => response.text())
  .then(result => {
    let data = JSON.parse(result)
    let city = data.name
    let country = data.sys.country
    let temp = data.main.temp.toFixed(1)
    let weatherId = data.weather[0].id
  })
  .catch((error) => console.log("error", error));
```

Servis vraća odgovor u JSON formatu iz kojega se onda mogu upotrijebiti razne informacije. U aplikaciji je napravljena mogućnost ako korisnik ima unesenu svoju adresu pod osobnim podacima, trenutna lokacija i vrijeme će se prikazivati po toj lokaciji.

## 6.5.2 Automatska sugestija lokacija

Diljem ove aplikacije je potreban unos adresa. Kako bi se izbjegle razne komplikacije i greške prilikom unosa adresa, savršeno se pronašao Google-ov servis za automatsku sugestiju lokacija. „Places Autocomplete“ je jedna od mogućnosti koje nudi lokacijska biblioteka Google Maps servisa. Funkcionira na način da se prilikom unosa teksta adrese pružaju sugestije o traženim mjestima, ulicama, adresama i kućnim brojevima. Time se korisnika usmjerava na odabir ponuđene adrese koja će uvijek biti u pravilnom formatu.

Učitavanje biblioteke u html script tag-u:

```
<script async  
src="https://maps.googleapis.com/maps/api/js?key=YOUR_API_KEY&libraries=places&callback=initMap">  
</script>
```

Primjena:

```
function initMap(){  
const input = document.getElementById("search-address-input");  
const autocomplete = new google.maps.places.Autocomplete(input);  
}
```

## 6.5.3 Prikaz i personalizacija karte

Najosnovnija upotreba Google Maps servisa omogućuje prilagođavanje karata vlastitim sadržajem i slikama za prikaz na web stranicama i mobilnim uređajima. Servis JavaScript Maps sadrži četiri osnovne vrste karata (putokaz, satelit, hibrid i teren) kojim se može izmijeniti pomoću slojeva, stilova, kontrola te raznih drugih usluga.

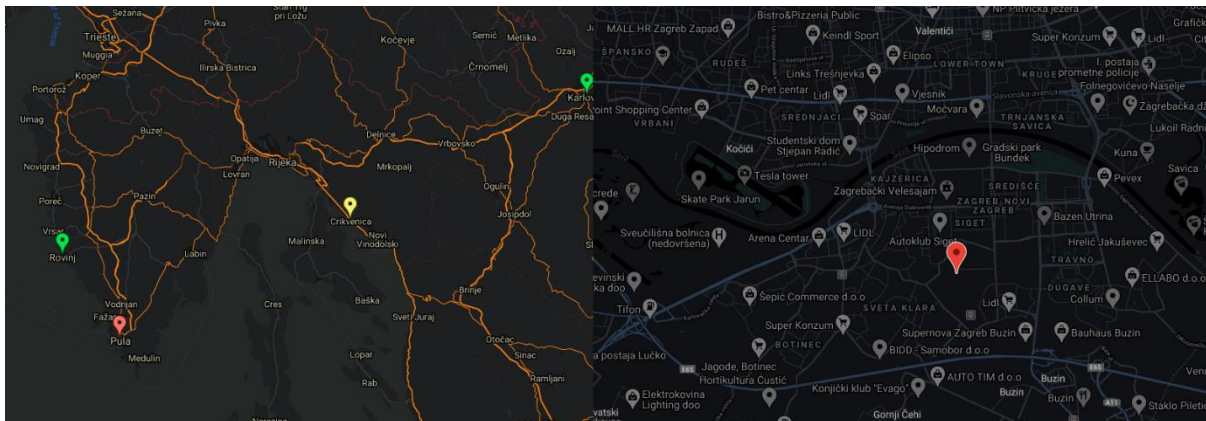
Html:

```
<div id="map"></div>  
  
<script  
src="https://maps.googleapis.com/maps/api/js?key=YOUR_API_KEY&callback=initMap&libraries=&v=weekly" async >  
</script>
```

Javascript:

```
let map;  
function initMap() {  
  map = new google.maps.Map(document.getElementById("map"), {  
    center: { lat: -34.397, lng: 150.644 }, zoom: 8,  
  });  
}
```





Slika 24. Tipovi karata

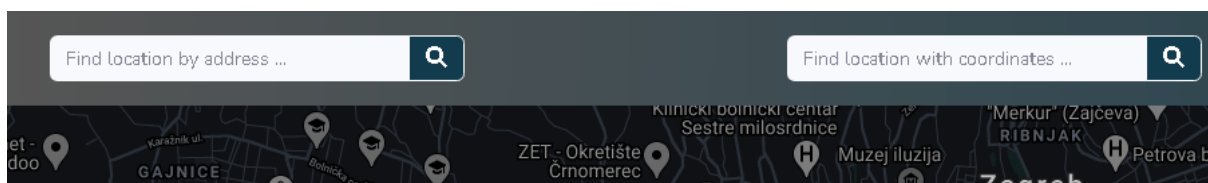
Google-ov Maps servis omogućava stiliziranje karata prema osobnim preferencijama. Dana je potpuna sloboda korisnicima u izboru boja i načinu isticanja objekata. Na slici 33. vidimo primjer personalizacije karte. Na lijevoj slici su istaknute ceste, a na desnoj slici je fokus na lokacijama i njihovim osnovnim informacijama.

### 6.5.4 Geokodiranje

Geokodiranje je proces prebacivanja adrese u geografske koordinate ili prebacivanje koordinata u adresu. To se u ovoj aplikaciji koristi kako bi se pronašla određena lokacija putem tražilice na karti i kako bi se postavila oznaka.

Programski kod zahtjeva:

```
fetch(`https://maps.googleapis.com/maps/api/geocode/json?address=${address}&key=API_KEY`, requestOptions)
  .then(response => response.text())
  .then(result => {
    const res = JSON.parse(result);
    const lat = res.results[0].geometry.location.lat;
    const lng = res.results[0].geometry.location.lng;
  })
  .catch(error => console.log('error', error));
```



Slika 25. Traženje lokacije po adresi ili koordinatama

Sve informacije o lokaciji adrese su zapisane u JSON formatu. Nakon što su pronađene koordinate lokacije karta se centrirala po toj lokaciji i postavlja se oznaka.

## 6.5.5 Prikaz oznaka na karti

Pin ili marker je oznaka koja identificira mjesto na karti. Oznaka prema zadanim postavkama koristi standardnu sliku. Markeri mogu prikazivati prilagođene slike, u tom slučaju se obično nazivaju "ikonama". Mogu se postaviti prilagođene ikone u konstruktoru markera ili pozivom „setIcon()“ na markeru.

Markeri su dizajnirani da budu interaktivni. Na primjer, prema zadanim postavkama primaju događaje "klik", tako da se mogu dodati slušatelj događaja kako bi se otvorio prozor s informacijama koji prikazuje prilagođene informacije. Korisnicima se mogu dopustiti radnje poput pomicanje markera na karti postavljanjem svojstva povlačenja markera na „true“.

Primjer programskog koda sa funkcijom slušanja na događaj „klik“:

```
marker = new google.maps.Marker({
  position: new google.maps.LatLng(markerOrders[i].lat, markerOrders[i].lng),
  map: map,
  animation: google.maps.Animation.DROP,
  icon: image});
google.maps.event.addListener(marker, 'click', (function(marker, i) {
  return function() {
    infowindow.setContent(markerOrders[i].address, markerOrders[i].destination,
    markerOrders[i].id)
    infowindow.open(map, marker);
    map.setCenter(data[i].originCoords);
    addLocationInput(markerOrders[i].id, markerOrders[i].address);
  }
})(marker, i));
```

## 6.5.6 Kreiranje rute

Pristup usluzi „Directions“ je asinkron, jer Google Maps servis mora uputiti poziv vanjskom poslužitelju. Iz tog razloga treba proslijediti metodu povratnog poziva za izvršenje nakon dovršetka zahtjeva. Ova metoda povratnog poziva trebala bi obraditi rezultate. Isto tako treba imati na umu da usluga „Directions“ može vratiti više od jednog mogućeg plana puta kao niz zasebnih ruta.

Da bi se kreirala ruta pomoću Google Maps servisa, potrebno je stvoriti objekt tipa „DirectionsService“ i pozvati „DirectionsService.route()“ funkciju da se pokrene zahtjev usluge „Directions“, prosljeđujući joj „DirectionsRequest“ objekt koji sadrži ulazne uvjete i način povratnog poziva koji će se izvršiti po primitku odgovora.

Primjer programskog koda:

```
let directionsService = new google.maps.DirectionsService();
let renderOptions = { draggable: true } };
let directionDisplay = new google.maps.DirectionsRenderer(renderOptions);
```

```

directionDisplay.setMap(map);

let request = {
  origin: originAddress,
  destination: destinationAddress,
  waypoints: waypoints, //niz lokacija
  optimizeWaypoints: true,
  travelMode: google.maps.DirectionsTravelMode.DRIVING};

directionsService.route(request, function (response, status) {
  if (status == google.maps.DirectionsStatus.OK) {
    directionDisplay.setDirections(response);}
  else { //rukovanje pogreškama }});

```

### 6.5.7 Napredna tražilica

Google Maps Places servis omogućuje pretraživanje podataka o mjestima pomoću različitih kategorija, uključujući ustanove, istaknute točke interesa i zemljopisne lokacije. Mjesta se mogu tražiti po blizini ili tekstnom nizu. Pretraživanje mjesta vraća popis mjesta zajedno sa sažetim informacijama o svakom mjestu.

Zahtjev za pronalaženje mjesta uzima tekst i vraća mjesto. Ulaz može biti bilo koja vrsta tekstualnih podataka mjesta, kao što su ime, adresa ili kućni broj. Zahtjev mora biti niz. Zahtjev za pronalaženje mjesta pomoću podataka koji nije niz, poput koordinate generira pogrešku.

Primjer zahtjeva:

```

service = new google.maps.places.PlacesService(map);

let request = {
  location: event.latLng,
  radius: 7000,
  types: „restaurant“,
  name: keywordFormatted };

service.nearbySearch(request, callback)

```

## 7. Prednosti i nedostaci u radu sa odabranim servisom

Niti jedan servis nije savršen, pa čak ako je i u pitanju Google. Unutar ovog poglavlja će se analizirati sve prednosti i nedostaci odabranog servisa.

### 7.1 Prednosti

Većina korisnika pametnih telefona i računala upoznata je sa sučeljem Google Maps-a, što će vjerojatno stvoriti osjećaj urođenog povjerenja prilikom interakcije s verzijom ugrađenom u aplikaciju. Tijekom godina svog postojanja, Google Maps osvojio je područje bez presedana svojim uslugama mapiranja i prikupio nedokučive količine informacija na lokalnoj razini. S Google-ovim satelitima, automobilima sa Street View-om i prilagođenim popravcima, Google-ova geografska pokrivenost smatra se najboljom. Sintaksa slična JSON-u koju koristi Google Maps učitava se odmah zajedno s mapom. Može se kontrolirati vidljivost, boja i prozirnost svih elemenata karte. [20]

Street View je dodan u mobilno izdanje Google Maps u rujnu 2019. Ova je značajka jedinstvena ponuda koju nema ni jedna druga usluga mapiranja. Značajka pruža interaktivne panorame iz različitih kutova na različitim ulicama širom svijeta. Integriranjem Street View-a u mobilnu ili web verziju svojih aplikacija, restorani, trgovine i drugi pružatelji usluga mogu omogućiti kupcima da gotovo posjete njihova mjesta. [20]

Google Maps trenutno podržava preko 80 jezika, a popis raste. Google je 2018. u softver Google Maps dodao 39 novih jezika, uključujući švedski i armenski, dopuštajući dodatnih 1,25 milijardi ljudi da koriste uslugu na svom jeziku. [20]

Najveća prednost proizlazi iz kombiniranja svih Google Maps servisa u jednu smislenu cjelinu.

### 7.2 Nedostaci

Google Maps ograničava broj zahtjeva po sekundi koji se mogu slati korisnicima. Tablicu s ograničenjima može se pronaći na web stranici Google Maps-a. Na primjer, servisu za upute o rutama može se poslati 50 zahtjeva u sekundi. Također se mora neprestano nadgledati upotreba servisa kako se ne bi premašila potrošnja, a ako se premaši ograničenje, račun će biti blokiran.

Iako se novi prilagođeni stilovi objavljuju u beta verziji, Google Maps servis trenutno ima ograničenu podršku za stvaranje jedinstvenog izgleda i stila kada su karte integrirane u aplikaciji.

Google Maps servis je zatvorenog tipa programskog koda. Bilo da je to zato što Google ne zahtijeva pomoć zajednice programera ili zato što se korporacija drži prava da implementira određene značajke bez vanjskog pristanka, stvari jednostavno stoje tako. Google Maps koristi više od milijardu ljudi svakog mjeseca, a ti ljudi svakodnevno šalju podatke. Bolje upute za korisnike znači više podataka za Google. Ti podaci se prikupljaju putem aplikacije Google Maps, ali i GPS postavki lokacije mobilnog uređaja. Opseg onoga što Google zna može se

vidjeti na vremenskoj traci kretanja u Google Maps aplikaciji. Sve male crvene točkice i plave rute prikazane na karti otkrivaju lokacije koje posjećujemo i rute kojima se krećemo. Jedan od načina ublažavanja prikupljanja podataka je putem opcija limitiranja dijeljenja podataka na mobilnom uređaju ili koristeći neku od alternativnih opcija poput OpenStreetMap koja je otvorenog tipa koda.

Model cijena Google Maps-a nije lako razumjeti. Kako bi privukao korisnike, Google svakog mjeseca izdvaja 200 dolara po korisniku za pozive prema servisu. Kada se taj iznos potroši, Google počinje naplaćivati. Povećanje cijena Google Maps servisa u 2018. izazvalo je burnu reakciju među vjernim korisnicima. Izvrsna usluga poput ove zaslužuje unovčavanje i trebala bi biti profitabilna, ali povjerenje kupaca još je jedna valuta na kojoj ne vrijedi štedjeti.

## 8. Zaključak

Razvoj informacijskih tehnologija od strane IT divova kao što su Google, Facebook, Amazon i drugi, uvelike je utjecao na dodjelu važnosti osobnih informacija o pojedincu. U toj priči se kao jedna od temeljnih informacija pronašla geolokacija. Izrazito lako dostupna, a jako cijenjena informacija.

Praćenje podataka o lokaciji je sveprisutno. Napetost između privatnosti i inovacija u ovom području pogoršava se brzim razvojem tehnologija praćenja i metodologija analitike podataka, kao i ogroman broj dostupnih potrošačkih podataka. Rizici gubitka privatnosti usko su povezani sa tim. Zakoni o privatnosti štite podatke o lokaciji, ali takva je zaštita ograničena na podatke o lokaciji koji su pojedincu identificirani. Zakoni se općenito odnose samo na početno sakupljanje podataka, međutim, nedavni medijski računi i radnje provedbe opisuju sekundarno tržište na kojem se identificirani podaci o lokaciji redovito prikupljaju i koriste ih treće strane sa kojima pojedinac nema izravne veze. Anonimni podaci o lokaciji se redovito kombiniraju s identificiranim osobnim podacima i koriste ih treće strane s kojima pojedinac nema izravan odnos za sastavljanje sveobuhvatnih profila pojedinca. Trenutni zakoni se ne bave tim praksama na sekundarnom tržištu. Rizici koje prati praćenje lokacije i profiliranje su dovoljni da zajamče razmatranje regulatorne intervencije raznim točkama.

Informacije o potrošačima su poprimile jednu od ključnih uloga u poslovanju poduzeća. Upravo to je prouzrokovalo razvoj geolokacijskih tehnologija i servisa koji su danas pristupačniji više nego ikad. Tržište servisa je prepuno izvrsnih servisa na koje se pojedinac može odlučiti ovisno o zahtjevima svoje aplikacije. Također je potrebno uzeti u obzir zahtjeve budućih korisnika aplikacije posebice uvezi osjetljivog područja privatnosti. Za svrhu ovog rada je korišten Google Maps servis pošto nije planirana komercijalna upotreba, tako da podaci bez mogućnosti reguliranja ipak završavaju u rukama Google-a. Korisnici „Geo Path“ aplikacije su upozoreni o dijeljenju njihove lokacije i podataka.

Učenje o Google Maps servisu otvara oči i nagrađuje, posebno razumijevanje servisnog okvira i implementacije unutar vlastite aplikacije. Za ovaj projekt su korištene gotovo sve usluge u sklopu Google Maps Javascript servisa kako bi se što više iskoristio potencijal servisa. Do servisnog ključa nije teško doći, ali potreba za unošenjem kreditne kartice se ističe kao nedostatak nad ostalim konkurentima. Jasna dokumentacija ispunjena raznim primjerima je mnogo olakšala posao. Sama upotreba servisa je izravna, jasno definirana i rigidna, samim time teško ga je upotrijebiti na način na koji nije predviđen da se koristi. Direktan sustav rukovanja greškama jasno navodi korisnika na probleme na koje nailazi prilikom implementacije servisa. Izuzevši rigidnost strukture servisa, on ipak pruža razne postavke koje će za prosječnog korisnika biti i više nego zadovoljavajuće. Online alat za stiliziranje karti koji je nedavno uveden je veliki korak unaprijed što se tiče personalizacije servisa i dolazi kao odgovor konkurencijama. Ovaj rad je dokaz da svaki programer koji se po prvi put susreće sa geolokacijskim servisom može uz pomoć dobre dokumentacije i primjera uspješno upotrijebiti servis u svrhu izrade vlastite aplikacije.

## Popis literature

- [1] A. C. Schwickert, "HTML - Hypertext Markup Language," *Informatik-Spektrum*, 1997. <https://developer.mozilla.org/en-US/docs/Web/HTML> (pristupljeno 09.03.2021).
- [2] J. Krause and J. Krause, "CSS: Cascading Style Sheets," *Introducing Web Development*, 2016. <https://developer.mozilla.org/en-US/docs/Web/CSS> (pristupljeno 09.03.2021).
- [3] Mozilla, "JavaScript." 2021. <https://developer.mozilla.org/en-US/docs/Web/JavaScript> (pristupljeno 09.03.2021).
- [4] M. Azat, *Express Web Framework*. Apress, 2014.
- [5] MongoDB, "What is MongoDB?," 2015. <https://www.mongodb.com/what-is-mongodb> (pristupljeno 09.03.2021).
- [6] Mongoosejs, "mongoose.," 2018. <https://mongoosejs.com/docs/guide.html> (pristupljeno 09.03.2021).
- [7] Mozilla, "jQuery", 2020. <https://developer.mozilla.org/en-US/docs/Glossary/jQuery> (pristupljeno 09.03.2021).
- [8] Mozilla, "Bootstrap.", 2020. <https://developer.mozilla.org/en-US/docs/Glossary/Bootstrap> (pristupljeno 09.03.2021).
- [9] N. Uvarov, "Geolocation API," 2016. [https://developer.mozilla.org/en-US/docs/Web/API/Geolocation\\_API](https://developer.mozilla.org/en-US/docs/Web/API/Geolocation_API) (pristupljeno 11.03.2021).
- [10] J. R. Sklar, *Geolocation Techniques*. Springer, 2018.
- [11] Statista Research Department, "U.S. location-based service users 2013-2018 Published by Statista Research Department, Jun 11, 2015 This statistic presents the number of location-based service users in the United States from 2013 to 2018. In 2014, 150 million users were accessing locati.", 2015. <https://www.statista.com/statistics/436071/location-based-service-users-usa/> (pristupljeno 11.03.2021).
- [12] P. M. Boshell, "The power of place: Geolocation tracking and privacy," *Business Law Today*, 2019. <https://businesslawtoday.org/2019/03/power-place-geolocation-tracking-privacy/> (pristupljeno 29.03.2021).
- [13] Investopedia, "Investopedia - Geolocation." 2021. <https://www.investopedia.com/terms/g/geolocation.asp> (pristupljeno 12.03.2021).
- [14] Betsie Estes, "Geolocation—The Risk and Benefits of a Trending Technology." , 2016. <https://www.isaca.org/resources/isaca-journal/issues/2016/volume-5/geolocation-the-risk-and-benefits-of-a-trending-technology> (pristupljeno 11.03.2021).
- [15] Mapbox, "Mapbox Company Introduction." , 2020. <https://www.mapbox.com/about/company/> (pristupljeno 13.03.2021).
- [16] Mapbox, "Mapbox Docs.", 2020. <https://www.mapbox.com/help/how-mapbox-works-overview/> (pristupljeno 13.03.2021).
- [17] Leaflet, "Leaflet docs.", 2016. <https://leafletjs.com/reference-1.7.1.html> (pristupljeno 14.03.2021).
- [18] Wiki Openstreetmap, "Leaflet | © OpenStreetMap.", 2016.

<https://wiki.openstreetmap.org/wiki/Leaflet> (pristupljeno 14.03.2021).

- [19] Leaflet, "Leaflet Github." 2016. <https://rstudio.github.io/leaflet/> (pristupljeno 14.03.2021).
- [20] Google "Google Maps About." 2016. [google.com/maps/about](https://google.com/maps/about) (pristupljeno 15.03.2021).
- [21] Google, "Javascript Google Maps API Documentation," 2016, 2016. <https://developers.google.com/maps/documentation/javascript/?hl=es> (pristupljeno 15.03.2021).
- [22] W. Xu, Y. Tao, and X. Guan, *Experimental Comparison of Free IP Geolocation Services*, vol. 895. Springer, 2020.

## Popis slika

Slika 1. Rast upotrebe geolokacijskih servisa od 2013. do 2018. [11].....	3
Slika 2. ER model baze podataka.....	16
Slika 3. Skočni prozorčić - odabir .....	17
Slika 4. Skočni prozorčić - forma menadžer / naručitelj.....	17
Slika 5. Dijagram korištenja - ne registrirani korisnik.....	20
Slika 6. Dijagram korištenja - naručitelj.....	20
Slika 7. Stranica za kreiranje nove narudžbe.....	21
Slika 8. Stranica za pregled vlastitih narudžbi.....	21
Slika 9. Stranica za uređivanje vlastitog korisničkog računa .....	22
Slika 10. Dijagram korištenja - vozač.....	23
Slika 11. Stranica trenutne rute vozača .....	24
Slika 12. Stranica pregleda svih ruta pojedinog vozača .....	24
Slika 13. Dijagram korištenja - menadžer .....	25
Slika 14. Stranica kreiranje rute.....	26
Slika 15. Upravljanje rutama.....	27
Slika 16. Pregled narudžbi.....	27
Slika 17. Pregled kategorija .....	28
Slika 18. Napredna tražilica .....	29
Slika 19. Statistike menadžera .....	29
Slika 20. Dijagram korištenja - administrator.....	30
Slika 21. Statistike korisnika i poduzeća .....	31



Slika 22. Dnevnik aplikacije .....	31
Slika 23. Prikaz trenutne lokacije i vremenske prognoze .....	32
Slika 24. Tipovi karata .....	34
Slika 25. Traženje lokacije po adresi ili koordinatama.....	34

## **Popis tablica**

Tablica 1. Cjenik Mapbox-a [21] .....	12
Tablica 2. Cjenik Google-a [22] .....	12