

Prikaz modova kretanja korisnika GSM mreže

Tomljenović, Andrija

Undergraduate thesis / Završni rad

2018

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Transport and Traffic Sciences / Sveučilište u Zagrebu, Fakultet prometnih znanosti**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:119:807628>

Rights / Prava: [In copyright](#) / [Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2024-12-25**



Repository / Repozitorij:

[Faculty of Transport and Traffic Sciences - Institutional Repository](#)



SVEUČILIŠTE U ZAGREBU
FAKULTET PROMETNIH ZNANOSTI

Andrija Tomljenović

Prikaz modova kretanja korisnika GSM mreže

ZAVRŠNI RAD

Zagreb,2018

Zagreb, 5. travnja 2018.

Zavod: **Zavod za inteligentne transportne sustave**
Predmet: **Baze podataka**

ZAVRŠNI ZADATAK br. 4945

Pristupnik: **Andrija Tomljenović (0135243327)**
Studij: **Promet**
Smjer: **Informacijsko-komunikacijski promet**

Zadatak: **Prikaz modova kretanja korisnika GSM mreže**

Opis zadatka:

Izraditi korisničko sučelje za prikaz trajektorija gibanja korisnika uključujući informaciju o korištenom modu prijevoza. Podatke je potrebno filtrirati i pohraniti u relacijsku bazu. U studiju slučaja omogućiti izvršavanje upita nad GPS podacima i dostupnim podacima GSM korisnika kroz korisničko sučelje.

Mentor:

Predsjednik povjerenstva za
završni ispit:

prof. dr. sc. Tonči Carić



prof. dr. sc. Hrvoje Gold

SVEUČILIŠTE U ZAGREBU
FAKULTET PROMETNIH ZNANOSTI

ZAVRŠNI RAD

**PRIKAZ MODOVA KRETANJA KORISNIKA GSM
MREŽE**

**VISUALIZATION OF GSM USERS TRANSPORT MODES
TRAJECTORIES**

Mentor: prof. dr. sc. Tonči Carić

Student: Andrija Tomljenović

JMBAG: 0135243327

Zagreb, 2018

Sažetak

Zadatak ovog završnog rada je izraditi korisničko sučelje pomoću kojeg će se prikazivati putanja kretanja korisnika s obzirom na mod prijevoza. Prije izrade samog sučelja dobiveni podaci su filtrirani i prilagođeni formatom za pohranu u relacijsku bazu podataka. Prilikom prikaza kretanja korisnika korištena je klasifikacija prema modu prijevoza, klasifikaciju je moguće napraviti na nekoliko načina. Prikazivanjem trajektorija korisnika i analizom podataka mogu se primijetiti različiti uzorci u kretanjima i navikama korisnika. Takve informacije mogu se koristiti u različite svrhe.

KLJUČNE RIJEČI: baza podataka, GPS, GSM, trajektorija, mod prijevoza

Summary

The task of this final thesis is to create user interface that will show user's path of traveling regarding the mode of transport. Before creating the user interface, given data are filtered and adapted so it can be stored in the relational database. When visualizing user's movement, a classification according to the mode of transport has been used, classification can be done in several ways. By displaying user trajectories and data analysis, different patterns can be observed in user's movements and habits. Such information can be used for various purposes.

KEYWORDS: database, GPS, GSM, trajectory, mode of transport

Sadržaj:

1.	Uvod	1
2.	Metode prikupljanja i filtriranja podataka o kretanju GSM korisnika	2
2.1.	Prikupljanje GPS podataka o kretanju korisnika	2
2.2.	Prikupljanje podataka o lokaciji GSM korisnika temeljem baznih postaja.....	3
3.	Pohrana podataka o kretanju GSM korisnika u relacijsku bazu	7
3.1.	Baza podataka i DBMS	7
3.2.	Priprema podataka za unos u bazu	8
3.3.	Unos podataka u relacijsku bazu	9
3.3.1.	Kreiranje baze podataka.....	10
3.3.2.	Unos podataka u bazu i dodavanje u tablice.....	12
4.	Vizualizacija i klasifikacija trajektorija prema modovima	15
4.1.	Vizualizacija trajektorija.....	15
4.2.	Klasifikacija trajektorija prema modu	18
4.3.	Primjena vizualizacije i klasifikacije trajektorija	19
5.	Studij slučaja za javno dostupne skupove podataka	21
5.1.	Razvojno okruženje	21
5.2.	Izrada sučelja	22
5.2.1.	Instalacija NuGet paketa	23
5.2.2.	Povezivanje aplikacije s bazom podataka.....	23
5.2.3.	Programiranje sučelja	24
5.2.3.1.	Kontrole karte	25
5.2.3.2.	Kontrole baze sa GPS podacima.....	28
5.2.3.3.	Kontrole baze sa podacima baznih postaja	31
6.	Zaključak	34
	Popis literature	35
	Popis kratica	37

1. Uvod

U ovome završnom radu biti će obrađen postupak izrade sučelja za prikaz trajektorija gibanja korisnika s obzirom na mod prijevoza.

Cilj ovog završnog rada je izraditi korisničko sučelje koje će upitima nad bazom filtriranih podataka prikazivati trajektorije gibanja korisnika te ih klasificirati prema modu prijevoza. Pojasniti će se kako su korišteni podaci prikupljeni te će se opisati postupak obrade podataka, pohrane podataka u relacijsku bazu podataka te izrade grafičkog sučelja. Pojasniti će se i funkcionalnosti izrađene aplikacije.

Rad je podijeljen u šest cjelina:

1. Uvod
2. Metode prikupljanja i filtriranja podataka o kretanju GSM korisnika
3. Pohrana podataka o kretanju GSM korisnika u relacijsku bazu
4. Vizualizacija i klasifikacija trajektorija prema modovima
5. Studij slučaja za javno dostupne skupove podataka
6. Zaključak

U drugom poglavlju opisane su metode prikupljanja korištenih podataka te je opisan korišteni skup podataka koji je korišten tijekom izrade ovoga rada.

U trećem poglavlju obrađuje se postupak pripreme podataka i pohrane podataka u relacijsku bazu podataka.

U četvrtom poglavlju obrađuju se načini i metode prikaza i klasifikacije kretanja korisnika prema modu prijevoza koji korisnik koristi.

U petom poglavlju opisan je postupak izrade grafičkog sučelja, te je su opisane funkcije i način rada izrađenog sučelja. U posljednjem poglavlju dani su zaključci istraživanja.

2. Metode prikupljanja i filtriranja podataka o kretanju GSM korisnika

Pri izradi ovoga rada korištena su dva različita skupa podataka. Prvi skup podataka predstavlja zapise o rutama kretanja korisnika prikupljene u sklopu Geolife projekta kojeg je provodio Microsoft Research Asia, [6]. Drugi skup podataka predstavlja zapise korisnika koji sadrže vrijeme prijave i lokaciju bazne stanice, u obliku koordinata, na koju se uređaj prijavio. Ovaj skup podataka prikupljen je u sklopu UrbanCPS projekta, [8].

2.1. Prikupljanje GPS podataka o kretanju korisnika

Globalni pozicijski sustav (*Global Positioning System - GPS*) predstavlja sustav satelita koji omogućuju točno lociranje uređaja sa GPS prijemnikom. GPS sustav satelita se sastoji od minimalno 24 satelita od kojih je 21 aktivno, a 3 služe kao rezerva. GPS sustav se sastoji od tri dijela: svemirski dio, kontrolni dio i zemaljski dio. Pod svemirski dio pripadaju sateliti koji kruže po orbiti na visini od oko 20 000 km te tako omogućavaju pokrivanje cijele Zemlje GPS signalom. Kontrolni dio se sastoji od četiri autonomne kontrolne stanice i jedne glavne kontrolne stanice koja prima podatke preko autonomnih stanica te nadzire kretanje i putanje satelita. Posljednji, korisnički, segment predstavlja sve korisničke uređaje koji se koriste GPS sustavom, imaju GPS prijemnike te obrađuju GPS podatke kako bi dobili željene informacije poput lokacije, brzine, ubrzanja i slično. Sateliti odašilju signale na nekoliko frekvencija, postoje L1 i L2 signali. L1 signali su za civilne prijammike te odašilju na frekvenciji od 1575,42 MHz sa vrlo malom snagom od 20 do 50 W. Obzirom na frekvenciju i snagu odašiljanja za kvalitetan signal i maksimalnu točnost potrebno je biti na što preglednijem mjestu u smislu da oko prijemnika ima što manje fizičkih prepreka kroz koje signal ne prolazi jer mogu uzrokovati različite degradacije i greške u signalu što drastično utječe na točnost određivanja pozicije, [5].

Skup podataka korišten u ovom radu koji prikazuje kretanje korisnika prikupljen je praćenjem kretanja korisnika pomoću GPS-a. Prikupljeni podaci su rezultat kretanja 178 korisnika kroz period od četiri godine od travnja 2007. godine do listopada 2011. godine. Podaci su prikupljeni preko korisničkih uređaja kao što su mobilni telefoni sa GPS modulom i GPS *loggeri*. Učestalost odnosno uzorak zapisivanja je u 91% zapisa bio ili vremenski: svakih 1 do 5 sekundi ili baziran na prijeđenom putu: svakih 5 do 10 metara. Kretanja korisnika koja su se

zapisivala predstavljaju svakodnevne aktivnosti korisnika što daje raznovrsnost podataka za daljnju analizu i obradu. Podaci sadrže informacije o kojemu se korisniku radi (u obliku brojanog identifikatora), koordinate položaja, geografska visinu, brzinu, ubrzanje, trenutku zapisa te informaciju o korištenom modu prijevoza. Slika 1. prikazuje sirove podatke GPS uređaja. Primjerice jedan korisnik može u jednom putovanju koristiti nekoliko različitih modova prijevoza, može putovati biciklom do željezničke postaje, zatim vlakom do neke udaljene željezničke postaje te od tamo ponovo biciklom do konačnog odredišta. Izvorni podaci su filtrirani i očišćeni od neispravnih zapisa. Neispravni zapisi se mogu pojaviti zbog različitih faktora koji utječu na zapisivanje podataka, očitavanje senzora, točnost mjerenja i slično. Cilj filtriranja je dobiti što uporabljiviji skup podataka u vidu da će svaki zapis sadržavati sve potrebne podatke te da neće biti drastičnih fluktuacija u odnosu na slične zapise, [6].

```

index,lat,long,altitude,trajectory_id,subfolder,labels,datetime,distance,timedelta,velocity,acceleration
180198,39.894178,116.3182,-777.0,20080328144824,10,train,2008-03-28 14:54:40,253.3751236730825,0 days 00:00:34.000000000,7.452209519796544,0.00898445935714632
180199,39.894505,116.321132,-777.0,20080328144824,10,train,2008-03-28 14:55:14,457.7031871384317,0 days 00:00:59.000000000,7.757681137939518,0.018562310808961625
180200,39.894953,116.326452,-777.0,20080328144824,10,train,2008-03-28 14:56:13,522.3185910644271,0 days 00:00:59.000000000,8.852857475668257,0.043408160762301466
180201,39.8946,116.332542,-777.0,20080328144824,10,train,2008-03-28 14:57:12,673.4223986779987,0 days 00:00:59.000000000,11.413938960644042,0.04894609625462979
180202,39.889622,116.33704,-777.0,20080328144824,10,train,2008-03-28 14:58:11,843.803759740365,0 days 00:00:59.000000000,14.301758639667199,0.015335740720686159
180203,39.88209000000001,116.338353,-777.0,20080328144824,10,train,2008-03-28 14:59:10,912.3940405312613,0 days 00:01:00.000000000,15.206567342187686,0.01931930029211536
180204,39.873873,116.338455,-777.0,20080328144824,10,train,2008-03-28 15:00:10,965.577796223162,0 days 00:00:59.000000000,16.365725359714606,0.03464862983866235
180205,39.865182,116.338058,-777.0,20080328144824,10,train,2008-03-28 15:01:09,1086.1896766915456,0 days 00:00:59.000000000,18.409994520195685,0.0443487076912709
180206,39.855403,116.33771000000002,-777.0,20080328144824,10,train,2008-03-28 15:02:08,1240.5675281648594,0 days 00:00:59.000000000,21.026568273980672,0.08510435674674048

```

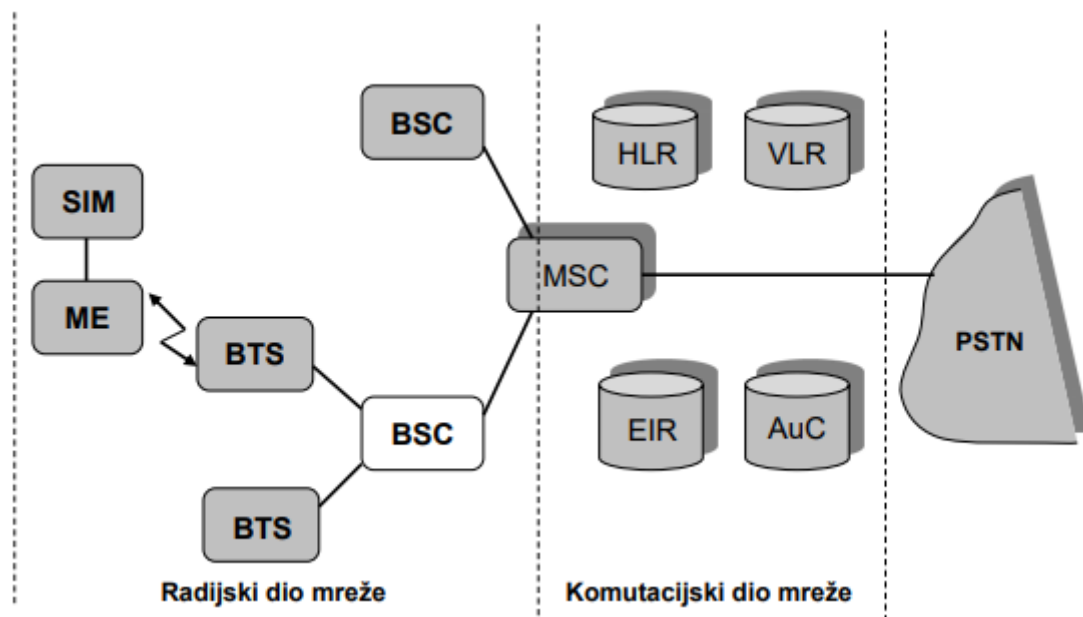
Slika 1. Prikaz preuzetih podataka o kretanju GPS korisnika

Izvor: izradio autor

2.2. Prikupljanje podataka o lokaciji GSM korisnika temeljem baznih postaja

Globalni sustav mobilne telefonije (*Global System for Mobile Communications – GSM*) je digitalna pokretna ćelijska mreža druge generacije (2G) koja omogućuje prijenos glasovnog i podatkovnog prometa. Prva komercijalna upotreba započela je 1991. godine, a koristi se još i danas. Glavna usluga GSM mreže je telefonija, a glavni napredak u odnosu na analogne sustave je mogućnost slanja SMS (engl. *Short Messaging Service*) poruka. Omogućavao je i pristup internetu, ali sa vrlo malom brzinom prijenosa od samo 9,6 kbps. Kako je potreba za internetom sve više bujala kroz godine razvijene su i dvije nadogradnje na GSM koje su omogućavale komutaciju paketa i veće brzine prijenosa. Prva nadogradnja je bila GPRS (engl. *General Packet Radio Services*) koja je omogućavala prijenos komutacijom paketa brzine od 56 do 114 kbps i smatrala se 2.5G, a druga je bila EDGE (engl. *Enhanced Data Rates for GSM Evolution*) koja je omogućavala brzine prijenosa do 384 kbps i smatrala se 2.75G. Širina frekvencijskog pojasa je 25 MHz, a frekvencijsko područje 890-915 MHz za smjer uređaj-bazna stanica i 935-

960 MHz za smjer bazna stanica-uređaj. Frekvencijski pojas od 25 MHz podijeljen je na 125 kanala od kojih svaki ima frekvencijski opseg od 200 kHz. Svaki kanal podijeljen je na 26 okvira, okviri su podijeljeni na 8 vremenskih odsječaka i ti odsječci su podijeljeni na 8 polja. Svaki od 8 vremenskih odsječaka ima trajanje 0,57 sekundi, a jedan se koristi za slanje, jedan za primanje i ostalih 6 za prijenos kontrolnih signala. Višestruki pristup riješen je TDMA (engl. *Time Division Multiple Access*) tehnikom dok je višestruki prijenos riješen pomoću FDD (engl. *Frequency Division Duplex*) tehnike, [7] [13].



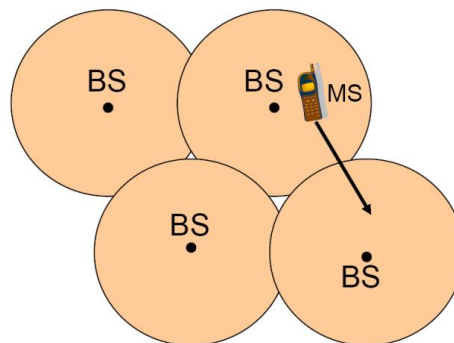
Slika 2. Prikaz arhitekture GSM mreže
Izvor: [13]

Kao što je ranije spomenuto GSM je ćelijska mreža što znači da su bazne postaje postavljene da tvore mrežu ćelija i tako pokrivaju određeno područje signalom, jedna ćelija obično pokriva područje od nekoliko kilometara. Na slici 2. moguće je vidjeti arhitekturu GSM sustava koja se sastoji od:

- **SIM** (engl. *Subscriber Identity Module*) – SIM kartica koja se nalazi u korisničkom uređaju te omogućuje sustavu identifikaciju i autentifikaciju korisnika.
- **ME** (engl. *Mobile Equipment*) – mobilna oprema podrazumijeva korisnikov terminalni uređaj pomoću kojeg korisnik šalje, prima i pregledava podatke. Također posjeduje sklopovsku i programsku opremu koja je potrebna za prijenos podataka (antene, procesori, memorije i sl.).
- **BTS** (engl. *Base Transciever Station*) – bazna postaja je krajnji dio GSM mreže koji komunicira sa ME, sadrži antenski sustav i primopredajni uređaj. Bazne stanice čine

ćelijsku strukturu, iako bazne stanice zrače u svim smjerovima jednako, pa čine krug zbog *handovera* dometi se do određene količine preklapaju, pa se mogu aproksimirati ćelijama. Na slici 3. je prikazana struktura ćelija.

- **BSC** (engl. *Base Station Controler*) – upravitelj baznih postaja koji upravlja radom, *handover*(prijelaz ME iz područja jedne u područje druge bazne postaje za vrijeme trajanja sesije) poziva između baznih postaja, pozivima prema baznim postajama te lokacijama korisnika.
- **MSC** (engl. *Mobile Switching Centre*) – Komutacijski centar, brine se o prosljeđivanju, uspostavljanju, prekidanju, nadziranju poziva unutar GSM mreže i poziva usmjerenih van GSM mreže prema ostalim mobilnim i fiksnim mrežama i obrnuto.
- Komutacijski dio mreže – sastoji se od baza podataka koje sadrže informacije o korisnicima. Služe da bi se prvo prilikom zahtjeva za komunikacijom korisnika i njegovu mobilnu opremu moglo identificirati i na kraju sesije pravilno naplaćivati ovisno o tarifi za pojedinog korisnika, [13].



Slika 3. Prikaz ćelijske strukture GSM sustava
Izvor: [13]

Drugi skup podataka kao što je ranije pojašnjeno predstavlja zapise pokušaja uspostave poziva pojedinih korisničkih uređaja (ME) na bazne stanice (BTS). Radi zaštite privatnosti korisnika, čiji su se podaci prikupljali, stvarni identifikacijski brojevi SIM kartica su zamijenjeni serijskim brojevima, a datumi su potpuno uklonjeni iz zapisa. Na slici 4. se može vidjeti prikaz podataka baznih stanica. Podaci su prikupljeni u kineskom gradu Shenzhenu, te su javno dostupni za korištenje u svrhe znanstvenih istraživanja, [8]. Podaci su prikupljeni na bazi od preko 10 milijuna korisnika dva velika operatera u gradu Shenzhenu. Podaci su prikupljeni u sklopu projekta UrbanCPS to jest Cyber-Physical System čija je svrha bila istražiti kako optimalno integrirati modele bazirane na različitim oblicima skupova podataka na temelju tri praktična

problema: heterogenosti podataka, oskudnosti ulaznih podataka i nepoznavanja temeljnog podrijetla podataka, [2].

```
OznakaUredaja,VrijemeSpajanja,GeoSirina,GeoDuzina  
55561,23:14:46,22.55541,114.22666  
55565,23:21:56,22.530884,114.113062  
55565,23:02:06,22.530884,114.113062  
55565,23:09:12,22.530884,114.113062
```

Slika 4. Prikaz preuzetih podataka o baznim postajama
Izvor: izradio autor

3. Pohrana podataka o kretanju GSM korisnika u relacijsku bazu

Podatci koji su dobiveni za manipulaciju odnosno izradu sučelja, koje će te podatke koristiti kao izvor informacija za prikaz kretanja korisnika te moda kojim se korisnici kreću, su zapisani u tekstualnoj datoteci. Podatke u takvome formatu teško je filtrirati i koristiti kao dobar izvor informacija za aplikaciju. Stoga su podaci pohranjeni u relacijsku bazu podataka kako bi se olakšalo manipuliranje i njihovo korištenje.

3.1. Baza podataka i DBMS

Baza podataka predstavlja skup strukturiranih i međusobno povezanih podataka. Njena svrha je pohraniti podatke te ih tako organizirati da se nad njima s lakoćom mogu obavljati operacije pretrage, izmjene, dodavanja i brisanja podataka. Današnji računalni svijet je nezamisliv bez baza podataka, u svakodnevnoj interakciji sa računalima, pametnim telefonima i sličnim uređajima vrlo često se u pozadini koriste baze podataka. Baze podataka mogu se podijeliti po različitim modelima podataka:

- Hijerarhijski model podataka
- Mrežni model podataka
- Objektni model podataka
- Relacijski model podataka
- Objektno-relacijski model podataka
- Dimenzijski model podataka

Model podataka predstavlja strukturu baze podataka te pomoću tipa podataka, odnosa između podataka i ograničenja omogućuje ispunjavanje njene svrhe. Model podataka također opisuje i skup operacija koje se koriste za manipulaciju, brisanje, dodavanje i sl. Hijerarhijski model podataka kako mu i ime govori strukturira podatke u hijerarhiju čiju strukturu se može opisati stablastom topologijom. Postoje dvije vrste čvora: čvor dijete i čvor roditelj svaki čvor dijete može imati samo jednog roditelja dok svaki roditelj može imati više djece. Nedostatak ovih baza je ako se obriše čvor roditelj i svi čvorovi dijete će se obrisati, te je postoji velika redundancija unutar baze. Mrežni model predstavljao je unaprjeđenje hijerarhijskog modela jer je omogućavao da jedan čvor dijete ima više čvorova roditelja te je zbog toga bio znatno

fleksibilniji. Objektni model podataka se služi objektima kako bi definirao podatke, svaki od objekata ima u sebi opisano ponašanje. Objektima se pristupa pomoću metoda, servisa ili operacija koje su posebno definirane za svaki objekt. Relacijski model baze podataka je vjerojatno danas najrašireniji model. Strukturiran je tako da među podacima postoje relacije. Sastoji se od entiteta, atributa, ključeva, relacija i ovisnostima među atributima. Za relacije se može reći da predstavljaju tablice zato što svaki redak predstavlja skup povezanih podataka. Primjerice ako neki teleoperater ima bazu podataka o korisnicima u koju pohranjuje sve podatke o korisniku njegov interes za podacima će biti ime, prezime, adresa, OIB, datum rođenja i slični podaci o korisniku. Prema tome svaki redak tablice ove baze bi sadržavao niz podataka koji su vezani za pojedinog korisnika, a svi redci imaju relaciju odnosno povezani su u smislu da predstavljaju iste informacije za različite korisnike. Relacije odnosno tablice još imaju i vlastiti naziv kao u ovome slučaju Korisnik jer predstavljaju podatke o korisniku. Stupci pojedine relacije nazivaju se atributima. Ključevi se koriste kako bi se jedinstveno definirao svaki redak i kako bi se povezale tablice međusobno, tj. odnosi među tablicama. Objektno-relacijski model podataka predstavlja proširenje relacijskog modela, glavna promjena mu je da korisnici sami mogu definirati tip podataka (klasu). Dimenzijski model predstavlja specijalnu preradu relacijskog modela. Dimenzijsku bazu predstavlja jedna velika tablica činjenica, a te činjenice su opisane uporabom dimenzija i veličina, [1] [11].

Sama baza podataka nema funkciju koja omogućuje izvođenje operacija poput dodavanja, brisanja i dohvaćanja podataka već je za to potrebna odvojena programska podrška. Takva programska podrška naziva se sustav za upravljanje bazama podataka (*Database Management System – DBMS*). Za izradu i upravljanje baze u ovome radu korišten je SQL Server DBMS. SQL (engl. *Search Query Language*) predstavlja računalni jezik koji služi upravljanju i obavljanju operacija nad relacijskom bazom podataka. SQL Server je pak Microsoftov programski paket koji omogućuje kreiranje, upravljanje i ostale operacije nad bazom, a koristi se SQL računalnim jezikom za obavljanje tih radnji, [11].

3.2. Priprema podataka za unos u bazu

U prvom skupu podataka se nalaze zapisi s koordinatama i modovima kretanja GSM korisnika. Prije samoga dodavanja u bazu, podatke je bilo potrebno obraditi kako bi podaci odgovarali dostupnim tipovima podataka u bazi. Skup podataka se sastojao od stupaca index,

lat, long, altitude, trajectory_id, subfolder, labels, datetime, distance, timedelta, velocity i acceleration. Svi podatci u svom izvornom obliku su bili pogodni za dodavanje u bazu s obzirom na dostupne tipove podataka u SQL Serveru osim stupca trajectory_id koji je bio oblika: „0 days 00:00:34.000000000“ kao što je vidljivo na gornjem dijelu slike 5. Obzirom da je potrebno ovaj stupac dodati u bazu kao atribut tipa podataka DateTime ili Time, a ni jedan od tih tipova nije bio odgovarajućeg formata, zapis je prilagođen datumskom tipu podatka.

Drugi skup podataka bio je znatno jednostavniji za pripremu, sastojao se od stupaca : SIM Card ID, Time, Latitude i Longitude. U ovome skupu podataka dodani su naslovi atributa prilikom dodavanja podataka u bazu jer podaci u svom izvornom obliku nemaju definirano zaglavlje. Za potrebu ovog završnog rada obrađen je samo dio podataka dovoljan za analizu ovog tipa. Podaci su odbačeni vodeći računa da se sačuvaju izvorni omjeri količine podataka iz pojedinog razdoblja jer su podaci bili spremljeni u 9 datoteka i grupirani u razdoblja duljine 2 sata počevši od 6 sati prijepodne pa sve do ponoći.

Prilikom unosa podataka u bazu potrebno je voditi računa i o normalizaciji odnosno poštivanju tri normalne forme. Normalizacija predstavlja postupak dijeljenja i promjene strukture relacijske sheme kako bi se smanjila zalihost i nepravilnosti u podacima. Prva normalna forma nam nalaže kako jedan atribut ne smije sadržavati skup podataka, domena mora sadržavati samo jednostavne vrijednosti te ne smiju postojati ponavljajući stupci. Korišteni skupovi podataka su već u izvornom obliku zadovoljavali prvu normalnu formu, pa nije bilo potrebe raditi promjene vezane za normalnu formu. Druga normalna forma nam nalaže kako prvo tablica mora zadovoljiti prvu normalnu formu, te da svaki atribut mora isključivo i potpuno biti ovisan o primarnom ključu. Kroz drugu normalnu formu mod prijevoza izbačen je u zasebnu tablicu zato što ne ovisi potpuno o primarnom ključu. Te treća normalna forma nalaže da svaki atribut ne smije biti ovisan niti o jednom drugom atributu osim primarnog ključa, [12].

3.3. Unos podataka u relacijsku bazu

Prije nego se podaci unesu u SQL Server potrebno je kreirati odvojenu bazu podataka za svaki od dva korištena skupa. Bazu je potrebno kreirati tako da tip podataka odgovara stvarnom podatku koji se nalazi u pojedinom polju. Na slici 5. su vidljive skripte koje su pokretanjem u SQL Serveru kreirale baze podataka i sve attribute sa odgovarajućim tipovima podataka. Tipovi podataka koji su korišteni pri izradi baza podataka jesu:

- *Int* – egzaktni cjelobrojni zapis.

- *Varchar(x)* – predstavlja tekst varijabilne veličine i ne podržava dijakritičke znakove, x označava maksimalni broj znakova koji se mogu unijeti.
- *Float* – aproksimirani brojevi zapisani sa plivajućim zarezom.
- *Decimal* – tip brojevanog zapisa u kojemu se navodi točan broj decimalnih mjesta koja se žele prikazati.
- *Datetime* – format koji reprezentira datum i vrijeme, jedan od mogućih formata je „YYYY-MM-DD hh:mm:ss“
- *Time* – predstavlja 24 satni zapis vremena formata „hh:mm:ss“

3.3.1. Kreiranje baze podataka

Kao što je vidljivo na slici 5. (lijevo) prva baza podataka kreirana je pod imenom PodaciUredaja i sadrži dvije tablice. Prva tablica pod nazivom ModPrijevoza sastoji se od dva stupca IDModPrijevoza i Naziv. IDModPrijevoza je univerzalni identifikator to jest primarni ključ ove tablice. Zadatak primarnog ključa je jednoznačno odrediti svaki zapis unutar tablice tako da ni jedan drugi zapis nema identičan primarni ključ. Ovaj stupac je tipa podatka *integer* (u kodu ključna riječ „int“) zato što će svi zapisi imati jednoznačnu cjelobrojnu vrijednost. Drugi stupac Naziv sadrži sve različite modove prijevoza koji se pojavljuju u podacima i tipa je *varchar(20)*. Druga tablica sadrži jedanaest stupaca:

- IDZapis – primarni ključ ove tablice, u kodu označeno ključnom riječi „primary key“ te je dodana opcija „identity(1,1)“ čime je omogućeno da se primarni ključ automatski generira počevši od 1 te svaki sljedeći zapis s inkrementom 1
- GeoSirina – geografska širina pojedinog zapisa, tipa je *float*
- GeoDuzina – geografska dužina pojedinog zapisa, tipa je *float*
- Visina – predstavlja nadmorsku visinu uređaja, tipa je *decimal*
- TrajectoryID – predstavlja oznaku pojedine putanje, to jest korisnika, tipa je *varchar(15)*
- Datum – egzaktno vrijeme i datum zapisa, tipa je *datetime*
- DuljinaPuti – predstavlja vrijednost prijeđenog puta od prethodnog zapisa, tipa je *float*
- TimeDelata – vremenska razlika između trenutnog i prethodnog zapisa, tipa je *time*
- Brzina – predstavlja trenutnu brzinu kretanja korisnika, tipa je *float*
- Ubrzanje – predstavlja trenutnu akceleraciju korisnika, tipa je *float*

- ModPrijevoza – strani ključ koji povezuje ovu tablicu sa tablicom ModPrijevoza, u ovome stupcu biti će prikazan odgovarajući IDModPrijevoza obzirom na korišteni mod prijevoza.

Važno je prvo kreirati tablicu ModPrijevoza jer inače je nemoguće postaviti strani ključ u tablici Zapis, te će nam SQL javiti pogrešku da izvršavanje koda nije moguće. S obzirom da su podaci prethodno filtrirani tako da svaki zapis sadrži sve podatke prilikom kreiranja tablice pomoću ključnih riječi *not null* određeno je da u bazi ne smije postojati prazno polje to jest da vrijednost polja ne smije biti *null*.

The image shows two side-by-side windows of a SQL script editor. The left window, titled 'PodaciUredaja.sql...CR93A\Andrija (52)*', contains the following SQL code:

```

create database PodaciUredaja

create table ModPrijevoza(
  IDModPrijevoza int not null primary key identity(1,1),
  Naziv varchar(20) not null
)

create table Zapis(
  IDZapis int not null primary key identity(1,1),
  GeoSirina float not null,
  GeoDuzina float not null,
  Visina decimal not null,
  TrajectoryId varchar(15) not null,
  Datum datetime not null,
  DuljinaPuti float not null,
  TimeDelta time not null,
  Brzina float not null,
  Ubrzanje float not null,
  ModPrijevozaID int not null foreign key references ModPrijevoza(IDModPrijevoza),
)

```

The right window, titled 'PodaciStanice.sql...CR93A\Andrija (53)*', contains the following SQL code:

```

create database PodaciStanice

create table Zapis(
  IDZapis int not null primary key identity(1,1),
  OznakaUredaja int not null,
  VrijemeSpajanja time not null,
  GeoSirina float not null,
  GeoDuzina float not null
)

```

Slika 5. Skripte za kreiranje baza podataka, GPS podaci korisnika (lijevo) i podaci baznih stanica korisnika (desno)
Izvor: izradio autor

Druge baze podataka kreirane su pod imenom PodaciStanice. Ova baza će imati samo jednu tablicu koja se sastoji od stupaca:

- IDZapis – primarni ključ tablice, jednoznačno opisuje svaki redak
- OznakaUredaja – predstavlja serijski broj korisničke SIM kartice tako to jest označava pojedinog korisnika
- VrijemeSpajanja – trenutak u kojem je uređaj pokušao uspostaviti poziv
- GeoSirina i GeoDuzina – predstavljaju koordinate bazne postaje na koju se korisnik spajao.

Kada su tablice kreirane SQL nudi mogućnost pregleda relacijske sheme koja grafički prikazuje povezanost između pojedinih tablica u smislu odnosa među atributima kao stranim ključevima. Na slici 6. se mogu vidjeti prikazi relacijskih shema prethodno izrađenih baza. Obzirom da su

kreirane baze poprilično jednostavne i shematski prikazi će biti vrlo jednostavni. Za kompleksnije baze shematski prikazi su izvrstan alat za lakše razumijevanje i opis baze.



Slika 6. Prikaz relacijskih shema baza podataka, GPS podaci (lijevo) i podaci baznih stanica (desno)

Izvor: izradio autor

3.3.2. Unos podataka u bazu i dodavanje u tablice

Prije samoga popunjavanja kreiranih tablica podacima prvo je potrebno te podatke u obliku kakvome jesu uvesti u bazu podataka, kada se podaci uvezu potrebno je upitom selektirati željene podatke i dodati ih u već kreirane tablice. Izvorni izgled preuzetih podataka prikazan je ranije u radu.

Podaci su dodani u bazu pomoću funkcije Import data. Kada SQL Server uveze podatke kreira novu tablicu u bazi u koju se uvoze podaci. Ta tablica za sve stupce ima tip podataka *nvarchar* tako da se svi podaci uvezu bez obzira na format. Kada se podaci nalaze u bazi potrebno je pomoću SQL upita izabrati podatke koje želimo i ubaciti ih u odgovarajuće tablice.

Prvo je potrebno popuniti tablicu ModPrijevoza kako bi se prilikom popunjavanja tablice Zapis mogle pravilno pridružiti vrijednosti stranih ključeva. Za dodavanje podataka koristi se naredba „*insert into*“, nakon naredbe upisuje se naziv tablice u koju se dodaju podaci. Potom pomoću „*select*“ naredbe selektiramo podatke koji se žele unijeti u tablicu. S obzirom da se u tablici ModPrijevoza trebaju nalaziti svi različiti modovi prijevoza i to za svaki mod isključivo po jedan zapis kako bi se zadovoljila normalizacija baze podataka, podaci se dodaju pomoću sljedeće naredbe:

`insert into ModPrijevoza select distinct labels from GPSwithMode`

Ovim upitom izvršava se sljedeće: u tablicu ModPrijevoza dodaj sve različite podatke (ključna riječ „*distinct*“) iz stupca *labels* koji se nalazi u tablici GPSwithMode. GPSwithMode je naziv datoteke iz koje su uvezeni podaci za ovu bazu stoga je i SQL prilikom uvoza tablici nadjenio isto ime. Na slici 7. moguće je vidjeti kako su selektirani samo različiti modovi i dodani u ciljane tablicu te je svakom dodijeljen različit identifikator.

altitude	trajectory_id	subfolder	labels	date_time	distance
-777.0	20080328144824	10	train	2008-03-28 14:54:40	253.3751236730825
-777.0	20080328144824	10	train	2008-03-28 14:55:14	457.7031871384317
-777.0	20080328144824	10	train	2008-03-28 14:56:13	522.3185910644271
-777.0	20080328144824	10	train	2008-03-28 14:57:12	673.4223986779987
-777.0	20080328144824	10	train	2008-03-28 14:58:11	843.803759740365
-777.0	20080328144824	10	train	2008-03-28 14:59:10	912.3940405312613
-777.0	20080328144824	10	train	2008-03-28 15:00:10	965.577796223162
-777.0	20080328144824	10	train	2008-03-28 15:01:09	1086.1896766915456

IDModPrijevoza	Naziv
1	train
2	bike
3	walk
4	motorcycle
5	boat
6	taxi
7	subway
8	bus
9	car
10	airplane
11	run

Slika 7. Rezultat izvršavanja upita kojim se dodaju podaci u tablicu Mod prijevoza
Izvor: izradio autor

Nakon uspješnog popunjavanja tablice ModPrijevoza može se popuniti tablica Zapis. Kod dodavanja podataka u ovu tablicu potrebno je kreirati takav upit da se u stupcu ModPrijevozaID dodaju odgovarajući identifikatori iz tablice ModPrijevoza. To se postiže sljedećim upitom:

```
insert into
Zapis(GeoSirina, GeoDuzina, Visina, TrajectoryId, Datum, DuljinaPuti,
TimeDelta, Brzina, Ubrzanje, ModPrijevozaID)
select lat, long, altitude, trajectory_id, date_time, distance, timedelta,
velocity, acceleration,
(select IDModPrijevoza from ModPrijevoza where Naziv=labels)
from GPSwithMode2
```

Pri upitu za dodavanje podataka u ModPrijevoza nije bilo potrebno specificirati koje sve stupce iz tablice se ubacuju podaci jer je postojao samo jedan stupac Naziv (drugi stupac s primarnim ključem se automatski popunjavao). Prilikom dodavanja podataka u tablicu Zapis potrebno je specificirati u koje stupce se dodaju podaci jer se neće dodati svi stupci iz tablice s uvezenim

podacima. Stupci koji se dodaju moraju se navesti redoslijedom tako da odgovaraju redoslijedu stupaca u koje se dodaju podaci. U posljednjem stupcu ModPrijevozaID potrebno je koristiti podupit kojim će se iz tablice ModPrijevoza dodati IDModPrijevoza prema odgovarajućem modu.

Kada je prva baza popunjena potrebno je popuniti i drugu bazu. Druga baza je znatno jednostavnija za popuniti jer je samo jedna tablica te nema potrebe za podupitima jer nema stranih ključeva. Dolje navedenim upitom se popunjava baza PodaciStanice:

```
insert into Zapis(OznakaUredaja, VrijemeSpajanja,GeoSirina,GeoDuzina)
select OznakaUredaja, VrijemeSpajanja, GeoSirina, GeoDuzina
from [08d10]
```

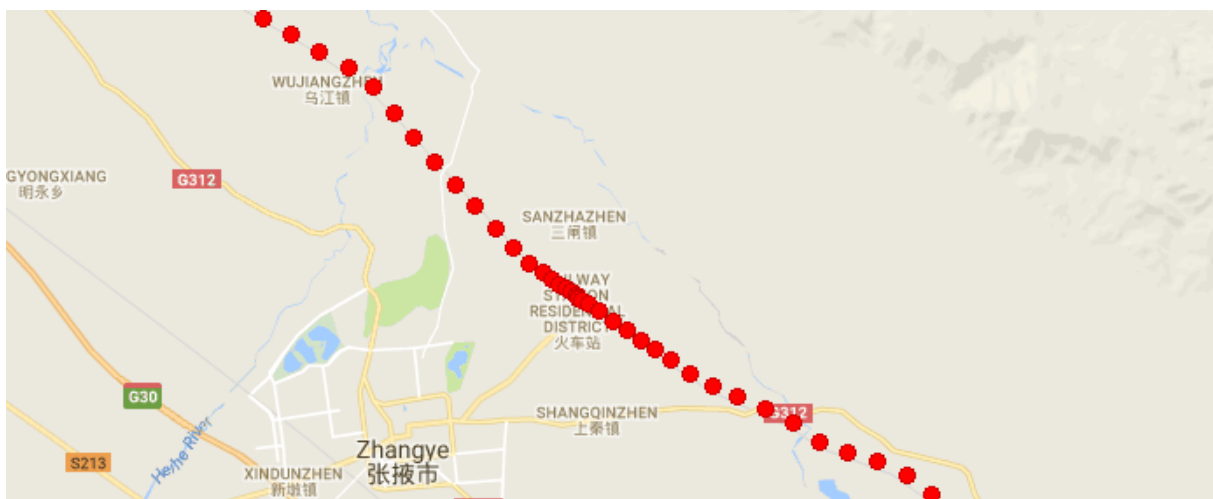
Potrebno je kreirati 8 ovakvih upita iz razloga što za razliku od prethodne baze gdje su svi podaci bili u jednoj datoteci podaci za ovu bazu nalaze se u 8 različitih datoteka. Tablice bi sada trebale biti popunjene podacima, uređene i spremne za korištenje.

4. Vizualizacija i klasifikacija trajektorija prema modovima

Obradom velikih skupova podataka može se doći do mnoštva korisnih informacija. Jedan od načina analize podataka je i prikazivanje podataka grafički. Grafički prikaz podataka uvelike ovisi o vrsti podataka koji se obrađuje, podaci se mogu prikazati na različite načine poput grafikona, aproksimiranim matematičkim funkcijama, ili kao kod izrade ovoga rada prikazati podatke na geografskoj karti.

4.1. Vizualizacija trajektorija

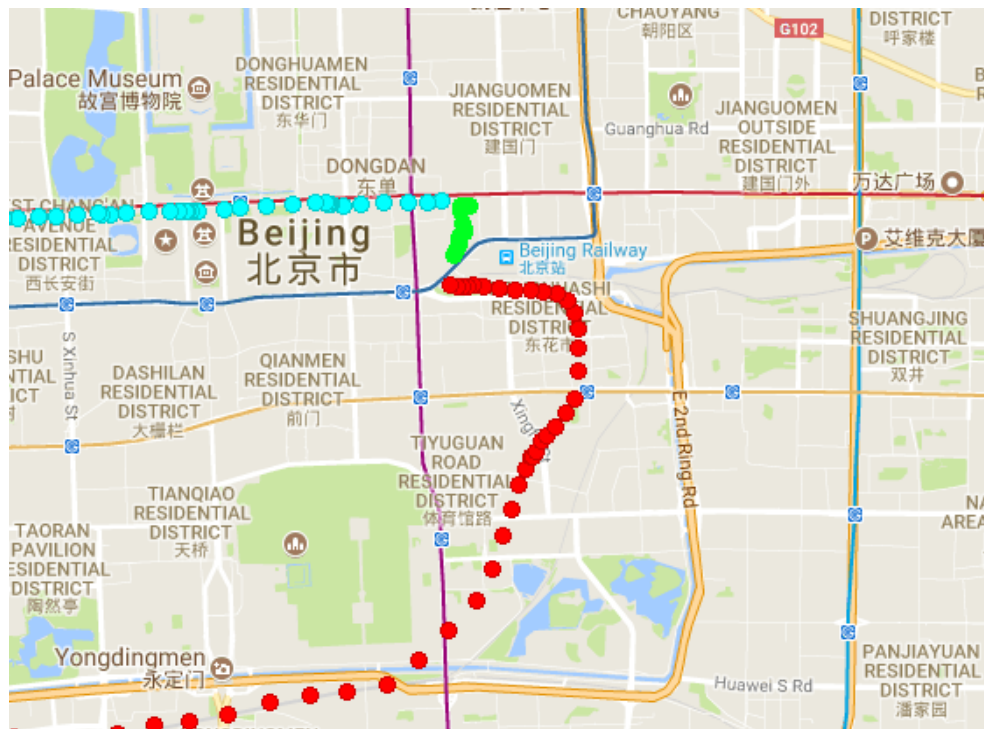
Skup podataka s GPS točkama kretanja korisnika sadrži niz kontinuiranih točaka za svaku pojedinu trajektoriju. Iscrtavanjem tih točaka na karti se može vidjeti točna putanja kojom se korisnik kretao. Promatrajući tu putanju mogu se primijetiti uzorci kretanja u smislu moda prijevoza. U korištenom skupu podataka postoji 11 različitih modova prijevoza: vlak, bicikl, hod, motocikl, brod, taxi, podzemna željeznica, autobus, automobil, zrakoplov i trčanje. S obzirom da neki od ovih modova predstavljaju isto prijevozno sredstvo također će i prilikom prikaza imati slične, pa čak i identične uzorke kretanja. Iako svaki zapis sadrži i podatak o modu prijevoza prema uzorku kretanja može se pretpostaviti o kojemu modu je riječ, pa je tako na slici 8. prikazan dio putanje kretanja korisnika koji se vozi vlakom. Iz prikaza se može pretpostaviti lokacija željezničke postaje.



Slika 8. Prikaz dijela putanje kretanja korisnika modom prijevoza vlak
Izvor: izradio autor

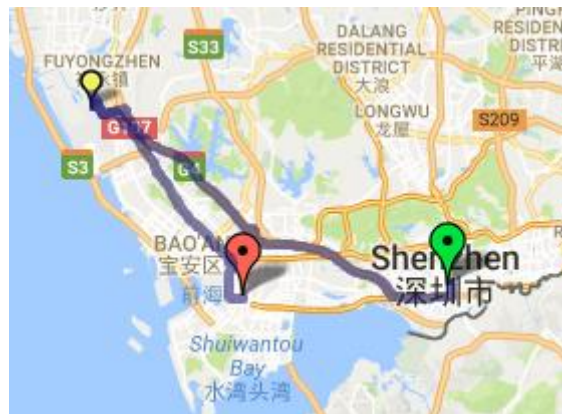
Na prikazanom dijelu putanje vlak se kreće s lijeva na desno. Sve točke pojedinih trajektorija su bilježene istim uzorkom (5-10 m ili u rasponu od 1-5 sekundi). Stoga se može zaključiti da se prema prikazanoj gustoći zapisa vlak kreće stalnom brzinom sve do otprilike sredine prikazane putanje gdje se zapisi počinju postepeno zgušnjavati u prostoru, te zatim u jednoj točki postanu najgušći. Za tu točku se može pretpostaviti da je željeznička postaja. Nakon željezničke postaje se može zamijetiti kako vlak počinje ponovo ubrzavati do neke stalne brzine te dalje nastavlja nepromijenjenom brzinom. Ako se dohvate podaci za prikazanu trajektoriju iz baze podataka vidljivo je da pretpostavljene koordinate željezničke postaje odgovaraju koordinatama zapisa gdje je brzina kretanja jednaka nuli. Pomoću ove metode može se pretpostaviti cijela mreža željezničkih postaja za dostupne podatke. Ovo je samo jedan od primjera primjene vizualizacije i analize promatranog skupa podataka.

Pojedina trajektorija može se sastojati i od više modova prijevoza, pa je potrebno razlikovati različite modove prilikom prikaza. Najjednostavniji način klasificiranja modova bi bio prikazati različite modove različitim bojama. Na slici 9. prikazan je jedan takav primjer gdje je korisnik došao vlakom na željezničku postaju (crvena trajektorija), zatim je pješice (zeleno trajektorija) otišao do autobusne stanice i putovanje nastavio autobusom (plava trajektorija).



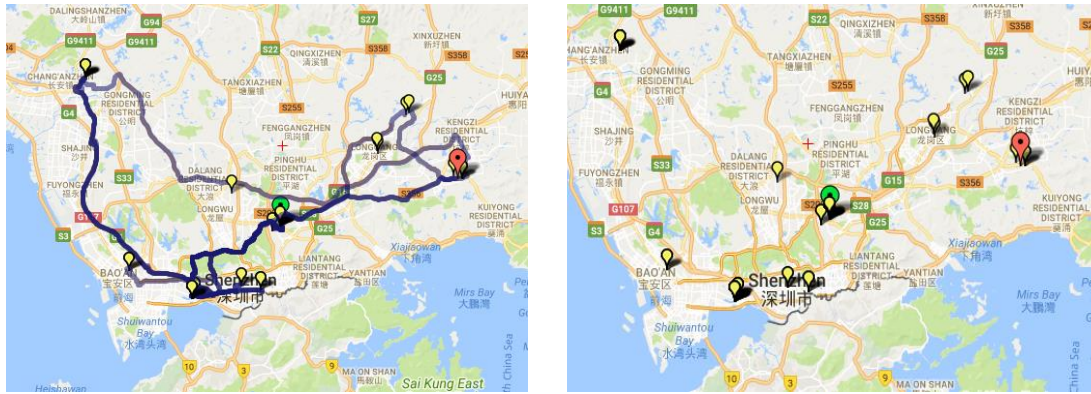
Slika 9. Prikaz putanje jednog korisnika s više modova prijevoza
Izvor: izradio autor

Skup podataka o baznim stanicama prikazuje različite informacije u odnosu na prethodno opisani skup jer je karakteristika podataka različita. Primjerice kod ovog skupa podataka moguće je prikazati aproksimativnu putanju korisnika na temelju posjećenih BSova. Aproksimativnu zato što postoje zapisi samo kada je korisnik pokušao uspostaviti poziv, što znači da se korisnik nije nužno morao kretati prikazanom putanjom, također točna lokacija korisnika nije poznata već samo lokacija bazne postaje unutar koje je korisnik u danom trenutku pokušao uspostaviti poziv. Na ovaj način prikupljena lokacija daje informaciju da se korisnik nalazi unutar dometa pojedine bazne postaje koji može iznositi do 35 km, ali u urbanim mjestima i ovisno o području iznosi od 1 do 3 km, [3]. Na slici 10. prikazana je pretpostavljena ruta korisnika između 10 i 11 sati na temelju posjećenih baznih postaja. Zeleni marker označava početak odnosno početnu baznu stanicu, žuti markeri (u ovome slučaju marker) označavaju posjećene bazne postaje, a crveni marker označava posljednju baznu postaju odnosno kraj rute.



Slika 10. Prikaz pretpostavljene rute kretanja korisnika na temelju posjećenih baznih postaja
Izvor: izradio autor

Ovakav prikaz povoljan je samo za manji broj zapisa jer putanja postaje nepregledna iscrtavanjem veće količine ruta kao što je moguće vidjeti na lijevom dijelu slike 11. koja prikazuje sve rute i posjećene bazne stanice(lijevo) i samo posjećene stanice bez iscrtane rute(desno). Na slici 11. se može primijetiti da zbog karakteristika algoritma koji iscrtava rute prikaz je izrazito nepregledan jer se veći broj ruta preklapa te se ne može na ovaj način rekonstruirati stvarna ruta kretanja korisnika. Prikaz cijelog kretanja korisnika moguće je prikazati i iscrtavanjem samo određenog broja posjećenih baznih postaja na način da ruta bude pregledna te da bude jasno kuda se korisnik kretao, ali ova metoda unosi dodatnu pogrešku u pretpostavljenu rutu koja je ionako već podosta neprecizna zbog karakteristika samih podataka.



Slika 11. Prikaz rute prema posjećenim baznim stanicama jednog korisnika
Izvor: izradio autor

4.2. Klasifikacija trajektorija prema modu

Modove prijevoza moguće je klasificirati kao što je ranije u radu navedeno pomoću grafičkog prikaza pod pretpostavkom da su svi zapisi bilježenim istim ili vrlo sličnim uzorkom. Prilikom iscrtavanja trajektorije pojedinog korisnika moguće je uočiti povećanje i smanjenje brzine kretanja pomoću gustoće prikazanih točaka kao što je opisano kod određivanja lokacije željeznički postaja. Ovo je samo interpretiranje podataka koji postoje u bazi pomoću koordinata. Obično se klasifikacija modova prijevoza ne analizira grafički već se na temelju izmjerenih parametara pojedine rute, poput brzine i akceleracije, određuje mod kretanja korisnika. Tako u skupu podataka svaki zapis sadrži podatke o trenutnoj brzini i ubrzanju. Stoga je mod kretanja moguće odrediti i pomoću analize odnosno određivanja prosječne brzine kretanja i akceleracije za svaki mod. U tablici 1. prikazane su vrijednosti prosječnih brzina za pojedini mod prijevoza. Iz tablice je vidljivo kako je svaki mod okarakteriziran zasebnom prosječnom brzinom. Postoje vrijednosti poput trčanja i hodanja koje imaju približno iste vrijednosti, pa je njihova klasifikacije uz ove parametre neprecizna te se takvi modovi prijevoza grupiraju u jedan mod prijevoza, u ovom slučaju pješak.

Tablica 1. Prosječna brzina različitih modova prijevoza, vrijednosti prosječne brzine izračunate su obradom opisanog skupa podataka

Mod Prijevoza	Prosječna brzina
Vlak	25,10865297
Bicikl	3,224108991
Pješak	1,554683683
Motocikl	6,381468829
Brod	7,025786409
Taxi	9,177375149
Podzemna željeznica	11,16467986
Autobus	5,907737335
Automobil	10,17296342
Zrakoplov	123,2478351
Trčanje	1,694703709

Izvor: izradio autor

Iz drugog skupa podataka bilo bi izrazito teško i neprecizno određivati modove kretanja korisnika na isti način. Ovakav skup podataka izvrstan je za analizu količine prometa u pokretnoj mreži u svrhu optimizacije, planiranja i nadogradnje mreže. Primjerice analizom ovakvog skupa podataka mogu se utvrditi navike kretanja korisnika, koncentracija korisnika na pojedinom području u određeno vrijeme, što omogućuje planiranje kapaciteta mreže na tome području.

4.3. Primjena vizualizacije i klasifikacije trajektorija

Osim ranije spomenutih primjena prilikom planiranja pokretnih mreža jedna od mogućih primjena jesu usluge bazirane na lokaciji (*Location Based Service* - LBS). LBS se mogu opisati kao usluge koje se koriste podacima o lokaciji korisnika kako bi pružile potrebne informacije. Ovakav tip usluge posebno je interesantan u današnje doba pametnih telefona, kada većina mobilnih terminalnih uređaja ima GPS modul. Primjer LBS usluge bi bio informiranje korisnika

o turističkim atrakcijama na nekome području. Kada korisnik dođe na neko područje i bazna postaja registrira njegov uređaj odmah bi mu stigla poruka o lokalnim atrakcijama, [4].

Također danas su vrlo popularne OTT (engl. *Over the Top*) usluge. OTT usluge predstavljaju pružanje telekomunikacijskih usluga mimo teleoperatera putem Interneta. Postoji mnoštvo OTT usluga koje se mogu svrstati pod LBS usluge. Kao primjer takvih usluga moguće je navesti Snapchat koji nudi slanje tekstualnih, slikovnih i video poruka te omogućuje korisnicima praćenje lokacije drugih korisnika. Također postoji mnoštvo usluga koje bazirano na korisnikovoj trenutnoj lokaciji nude restorane, hotele i slično.

5. Studij slučaja za javno dostupne skupove podataka

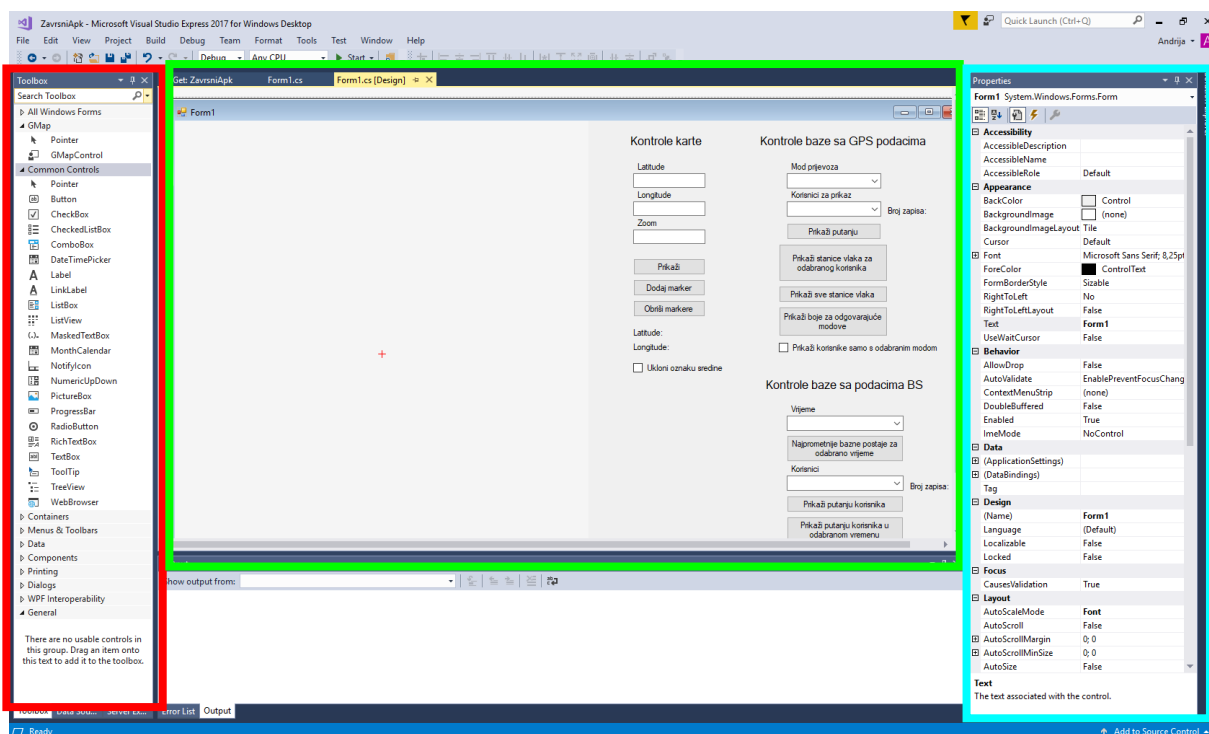
Kao dio ovoga rada bilo je potrebno izraditi aplikaciju to jest korisničko sučelje kroz koje će se vršiti upiti nad opisanim skupovima podataka, te grafički na karti prikazivati dostupne podatke s obzirom na korisnika i njegov mod kretanja. Grafičko korisničko sučelje izrađeno je pomoću Microsoftovog alata Visual Studio Express, a podaci su pohranjeni u SQL (engl. *Structured Query Language*) bazi podataka.

5.1. Razvojno okruženje

Kao razvojno okruženje pri izradi sučelja korištena je besplatna inačica Microsoftovog Visual Studia. Visual Studio Express predstavlja razvojno okruženje i moćan alat za programiranje i izradu aplikacija. Primarni programski jezik koji se koristi u VS je C#, a podržava i druge programske jezike poput C/C++ i Phytona. Prilikom kreiranja novog projekta VS nudi niz predložaka od kojih su značajniji:

- Aplikacija s tekstualnim sučeljem– omogućuje izradu programa koji koriste konzolu kao sučelje za unos i ispis podataka. Najčešće se koristi za izradu programa koji nemaju potrebe za kompleksnijim sučeljima kako bi obavljali svoju funkciju.
- Aplikacija s grafičkim sučeljem – predložak koji stvara početni kod za izradu aplikacija sa kompleksnijim sučeljima u obliku grafičkog sučelja. Ovakav oblik predloška omogućuje korisniku da sam dodaje kontrole poput gumba i padajućih izbornika te da ih sam oblikuje i programira.

Za izradu sučelja korišten je predložak aplikacije s grafičkim sučeljem, te je za programiranje korišten C# programski jezik. C# je objektno orijentirani programski jezik. Za razliku od ne objektnog programiranja gdje je kod kontinuiran i u jednom komadu kod objektno orijentiranog programiranja kod je podijeljen na objekte, zasebne grupe koje funkcioniraju neovisno o drugim objektima, a zajedno čine neki program. Svaki objekt ima opisan sadržaj i definirano ponašanje kroz deklaraciju klasa i metoda. Prednosti objektno orijentiranog programiranja jesu lakše otklanjanje grešaka, jednostavnost održavanja, mogućnosti ponovnog korištenja koda (pojedinih objekta) i učinkovitost, [9].



Slika 12. Prikaz sučelja Microsoft Visual Studio razvojnog okruženja
Izvor: izradio autor

Slika 12. prikazuje osnovno sučelje Visual Studio-a. Crvenim pravokutnikom označeno je područje palete alata koji su dostupni i mogu se dodavati u prozor aplikacije. Zeleni pravokutnik označava područje gdje je moguće dizajnirati izgled aplikacije, veličinu prozora same aplikacije, ali i veličine svih komponenti. U plavo označenom pravokutniku mogu se uređivati svojstva pojedinih komponenata i kontrola poput teksta koji će pisati na gumbu ili imena pomoću kojeg se referencira na pojedinu kontrolu unutar samog koda prilikom programiranja. Također moguće je odabrati događaje poput klika na gumb prilikom kojih će se izvršiti dio koda.

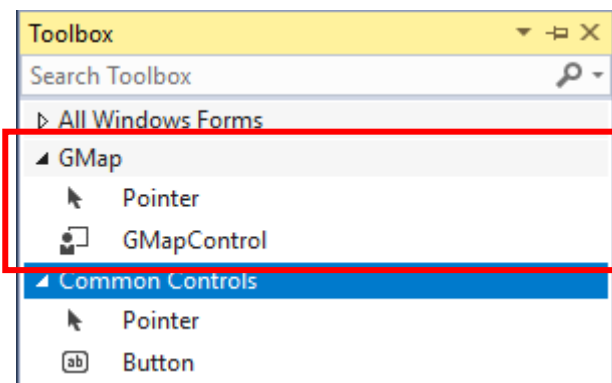
5.2. Izrada sučelja

Kao osnovu za prikaz dostupnih podataka korištena je Google karta. Prije ikakve obrade i prikaza podataka bilo je potrebno implementirati Google kartu u sučelje. S obzirom da standardno VS nema nikakvih opcija za implementaciju karte potrebno je vlastoručno napisati

kod koji će omogućiti korištenje karte ili preuzeti već napisani kod podijeljen od strane drugih korisnika koji je javno dostupan.

5.2.1. Instalacija NuGet paketa

NuGet Package predstavlja funkciju VSa gdje programeri mogu podijeliti svoj kod s drugim korisnicima. To je jednostavno sažet kod koji je moguće implementirati u vlastiti projekt kako bi se proširile funkcionalnosti programa. Pomoću NuGet Package manager pronađen je GMap.NET.Windows NuGet paket nakon čijeg preuzimanja je potrebno dodati kontrole koje dolaze u paketu u alate kako bi se mogle koristiti. Nakon uspješne instalacije paketa unutar prozora *Toolbox* trebale bi se pojaviti kontrole kao što je crvenim pravokutnikom označeno na slici 13, [10].



Slika 13. Prikaz prozora Toolbox GMap kontrola
Izvor: izradio autor

5.2.2. Povezivanje aplikacije s bazom podataka

Kako bi aplikacija mogla iscertavati rute i obavljati predviđene funkcije potrebni su joj podaci. Aplikacija sadrži samo podatke koje kroz kod inicijaliziramo i postavimo im vrijednosti, a za velike količine podataka to je izrazito nepraktično. Stoga je podatke koji se žele obrađivati najbolje pohraniti u bazu podataka, kao što je za korištene skupove podataka već napravljeno. VS nudi mogućnost jednostavnog spajanja na bilo koju bazu podataka, izvršavanja upita nad podacima u spojenoj bazi i čitanje podataka dobivenih upitom.

Prvo je potrebno definirati na koji server i na koju bazu na tome serveru se želi povezati, a to se definira pomoću *connection stringa*. Zatim je potrebno napisati upit kojim će se dobiti željeni podaci, te na kraju otvarati redak po redak i čitati podatke koji su potrebni, cijeli ovaj postupak je detaljnije pojašnjen dalje u tekstu.

5.2.3. Programiranje sučelja

Nakon što su uvezeni potrebni NuGet paketi može se prijeći na izradu samoga sučelja. Prvo je potrebno za svaki skup kontrola postaviti željene kontrole u prozor aplikacije koja se izrađuje i definirati njihovu funkciju. Potom je potrebno programirati kontrole kako bi obavljale željene funkcije.

Za potrebe ovoga rada izraditi će se tri skupine kontrola:

1. Kontrole karte – osnovne kontrole koje omogućuju prikazivanje upisanih koordinata, dodavanje markera na željenu točku, prikaz koordinata neke točke te čišćenje karte, tj. uklanjanje svih markera i ruta koje se nalaze na karti.
2. Kontrole baze sa GPS podacima – pomoću ovih kontrola se želi postići filtriranje korisnika prema modu prijevoza (one koji putuju samo s odabranim modom ili sve korisnike koji putuju odabranim modom), prikazati putanju kretanja korisnika, ako korisnik putuje vlakom prikazati sve pretpostavljene željezničke postaje prema dostupnim podacima, prikazati sve moguće željezničke postaje prema dostupnim podacima.
3. Kontrole baze sa podacima baznih postaja – pomoću ovih kontrola se želi omogućiti odabir vremenskog razdoblja koje je sadržano u podacima te prikazati najprometnije bazne postaje za odabrano vrijeme, prikazati pretpostavljenu rutu odabranog korisnika u odabranom vremenu i u cijelom dostupnom vremenskom razdoblju.

5.2.3.1. Kontrole karte

Kao osnovne mogućnosti karte dostupne su sljedeće funkcije:

- Mogućnost upisivanja geografske širine i geografske dužine te prikazivanje istih na karti.
- Prilikom prikazivanja upisane točke na karti moguće je odabrati *zoom* karte, a ukoliko se ne odabere postavljena je zadana vrijednost.
- Postavljanje markera na označeno područje (sredina prikazanog područja na karti).
- Uklanjanje svih markera koji se trenutno nalaze na karti(uključujući i markere kreirane radi prikaza trajektorije).
- Klikom na kartu ispisuju se koordinate točke.
- Mogućnost uklanjanja zadane oznake sredine karte.



Slika 14. Prikaz osnovnih kontrola grafičkog sučelja upravljanje geografskom kartom
Izvor: izradio autor

Pod brojem 1 na slici 14. nalaze se kontrole tipa *Textbox* u koje je moguće upisati vrijednosti geografske širine (engl. *latitude*) i dužine (engl. *longitude*) te željenu razinu zumiranja. Za ove kontrole nema posebnog koda već se koriste unutar drugih kontrola za unos informacija.

Pod brojem 2 se nalazi gumb „Prikaži“. Klikom na ovaj gumb izvršava se metoda koji sadrži sljedeći kod:

```

29 private void button_PriKoor_Click(object sender, EventArgs e)
30 {
31
32     if (textBox_Lat.Text == string.Empty || textBox_Longt.Text == string.Empty)
33         MessageBox.Show("Molim unesite koordinate!", "Koordinate nisu unesene!");
34     else
35     {
36
37         double lat = Convert.ToDouble(textBox_Lat.Text);
38         double longt = Convert.ToDouble(textBox_Longt.Text);
39         gMap.MapProvider = GMapProviders.GoogleMap;
40         gMap.Position = new PointLatLng(lat, longt);
41
42
43         gMap.MinZoom = 0;
44         gMap.MaxZoom = 18;
45         if (textBox_Zoom.Text != string.Empty)
46         {
47             gMap.Zoom = Convert.ToDouble(textBox_Zoom.Text);
48         }
49         else gMap.Zoom = 13;
50     }
51 }

```

Slika 15. Isječak koda koji prikazuje metodu koja se pokreće nakon pritiska gumba „Prikaži“
Izvor: izradio autor

1. Pomoću *if* (redak 32) grananja provjerava se jesu li upisane koordinate. Ukoliko nisu korisniku se prikazuje *MessageBox*, tj. prozor s porukom da koordinate nisu unesene
2. Ukoliko su koordinate unesene tekst upisan u tekstualna polja konvertira se u tip podatka *double* (redak 37 i 38), poziva se željena karta (redak 39) i pozicija karte se postavlja na upisane koordinate (redak 40)
3. Kartu je moguće zumirati, ali je potrebno definirati maksimalnu i minimalnu razinu zumiranja te se provjerava ima li podataka od korisnika o željenoj razini, ako nije onda se postavlja zadana razina (redci 43 do 49).

Pod brojem 3 nalazi se gumb čijim se klikom na kartu dodaje marker na centar prikazanog područja (na karti označeno crvenim križićem). Klikom na gumb izvršava se sljedeći kod:

```

53 private void button_DodMar_Click(object sender, EventArgs e)
54 {
55     PointLatLng point = gMap.Position;
56     GMapMarker marker = new GMarkerGoogle(point, GMarkerGoogleType.red_pushpin);
57     GMapOverlay markers = new GMapOverlay("markers");
58     markers.Markers.Add(marker);
59     gMap.Overlays.Add(markers);
60     gMap.Position = point;
61 }

```

Slika 16. Isječak koda koji prikazuje metodu koja se pokreće nakon pritiska gumba „Dodaj Marker“
Izvor: izradio autor

1. Inicijalizira se točka tipa *PointLatLng* te joj se odmah dodjeljuju koordinate sredine prikazanog područja. (redak 55)
2. Inicijalizira se novi marker te se definiraju koordinate na koje će se postaviti i vrsta markera. (redak 56)
3. Inicijalizacija novog *overlaya* pod nazivom „markers“. (redak 57)
4. U *overlay* se dodaje kreirani marker. (redak 58)
5. Te se u *gMap* kontrolu dodaje kreirani *overlay* sa svim dodanim markerima.

Pod brojem 4 nalazi se gumb pomoću kojega se brišu svi markeri i rute koje se nalaze na karti, klikom na gumb izvršava se sljedeći kod:

```

63  private void button_ObrMark_Click(object sender, EventArgs e)
64  {
65      gMap.Overlays.Clear();
66      gMap.Refresh();
67  }

```

Slika 17. Isječak koda koji prikazuje metodu koja se pokreće nakon pritiska gumba „Obriši markere“

Izvor: izradio autor

1. Brišu se svi dodani *overlayi*
2. Potrebno je osvježiti *gMap* kontrolu kako bi se markeri uistinu maknuli.

Pod brojem 5 nalaze se dvije kontrole tipa *label* koje klikom miša negdje na kartu ispisuju koordinate odabrane točke, klikom na gumb izvršava se sljedeći kod:

```

643  private void gMap_MouseClick(object sender, MouseEventArgs e)
644  {
645      if (e.Button == MouseButtons.Left)
646      {
647          var point = gMap.FromLocalToLatLng(e.X, e.Y);
648          label_Lat.Text = "Latitude: " + Convert.ToString(point.Lat);
649          label_Lng.Text = "Longitude: " + Convert.ToString(point.Lng);
650      }
651  }
652

```

Slika 18. Isječak koda koji prikazuje metodu koja se pokreće nakon pritiska lijeve tipke miša na kartu

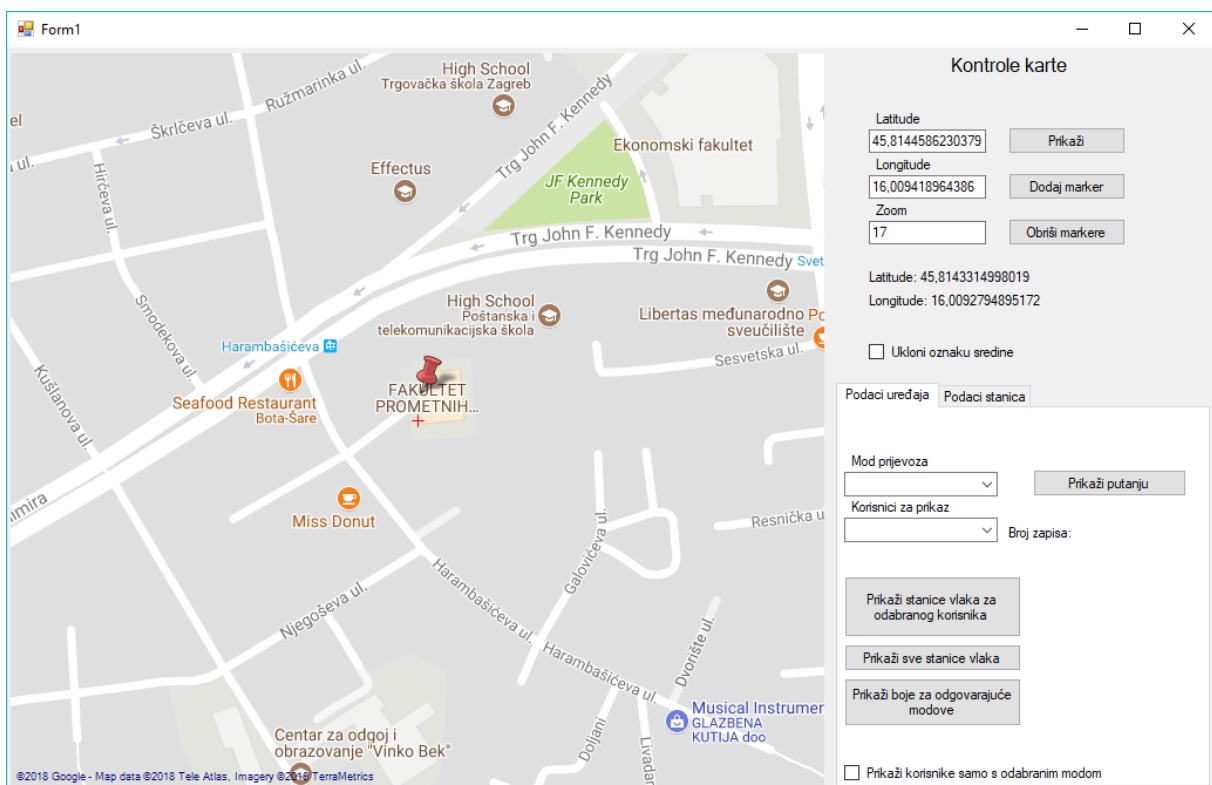
Izvor: izradio autor

1. Uvjetom *if* se ispituje koja tipka miša je odabrana, ukoliko je lijeva tipka odabrana kod se izvršava

2. Inicijalizira se točka, te se pretvaraju koordinate klika miša u geografske koordinate i dodjeljuju se inicijaliziranoj točki
3. Odgovarajuće tekstualne oznake ispisuju pojedinu koordinatu.

Pod brojem 6 nalazi se *CheckBox* odnosno kućica koju se može označiti ukoliko korisnik želi ukloniti oznaku sredine karte (crveni križić) koja je po zadanim postavkama uvijek prikazana.

Slika 19. prikazuje primjer uporabe kontrola, upisane su željene koordinate i postavljen je marker. Na ovoj slici moguće je vidjeti oznaku sredine, tj. crveni križić jer kućica za uklanjanje oznake nije označena.



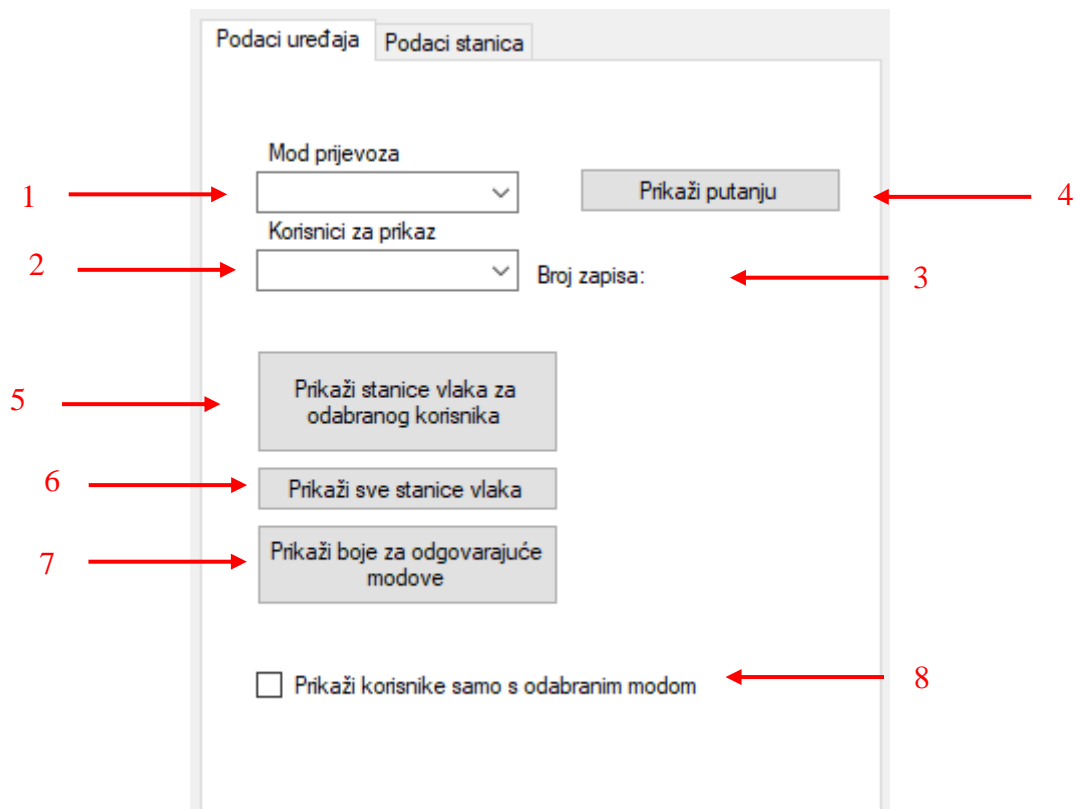
Slika 19. Grafički prikaz korištenja kontrola karte
Izvor: izradio autor

5.2.3.2. Kontrole baze sa GPS podacima

Osnovne mogućnosti izvršavanja upita i prikazivanja podataka iz baze PodaciUredaja:

- Filtriranje korisnika prema modu prijevoza, svi korisnici koji koriste odabrani mod ili samo korisnici koji koriste isključivo odabrani mod
- Prikaz trajektorije odabranog korisnika
- Ukoliko je korisnik putovao vlakom, prikaz svih pretpostavljenih željezničkih postaja

- Prikaz svih pretpostavljenih željezničkih postaja iz dostupnih podataka.



Slika 20. Prikaz kontrola koje omogućuju izvršavanje upita nad bazom GPS podataka
Izvor: izradio autor

Slika 20. prikazuje kontrole za rukovanje bazom PodaciUredaja. Na slici brojem 1 označen je padajući izbornik na kojemu se odabire mod prema kojemu se želi filtrirati korisnike. Modovi koji su dostupni za odabir se prilikom pokretanja aplikacije učitavaju iz baze podataka.

Broj 2 označava padajući izbornik pomoću kojega se filtriraju korisnici prema modu odabranom na padajućem izborniku označenim brojem 1. Dodatno je moguće označiti i kućicu pod brojem 8 ako se žele korisnici koji koriste isključivo odabrani mod, inače se prikazuju svi korisnici koji koriste odabrani mod bez obzira koriste li se u pojedinom trajektoriju kojim drugim modom.

Kontrola *label* označena brojem 3 ispisuje broj zapisa za odabranog korisnika, primarna svrha ove kontrole je predati informaciju o broju zapisa prilikom klika na gumb pod brojem 4 radi optimizacije prikaza.

Brojem 4 označen je gumb koji ako je odabran korisnik iscrtava putanju korisnika u obliku niza markera kada se klikne na njega. Na slici 21. moguće je vidjeti kod koji se izvršava klikom na gumb (radi bolje kvalitete prikazan je samo bitniji dio bloka koda):

```
private void button_PrikazPutanje_Click(object sender, EventArgs e)
{
    gMap.Overlays.Clear();
    gMap.MapProvider = GMapProviders.GoogleMap;
    SqlConnection sql = new SqlConnection
        ("Data Source= DESKTOP-15CR93A\\SQL EXPRESS;Initial Catalog=PodaciUredaja;Integrated Security=True;MultipleActiveResultSets=True");
    sql.Open();
    SqlDataReader reader;
    SqlCommand cmd = new SqlCommand("select GeoSirina, GeoDuzina, ModPrijevozaID from Zapis where TrajectoryId = " + comboBox_Kor.Text, sql);
    reader = cmd.ExecuteReader();
    double GeoSirina, GeoDuzina;
    int razmak = OptimizacijaPrikaza(Convert.ToInt16(label_brZap.Text.Substring(13)));
    int brojac = 0;
    PointLatLng tocka;
    while (reader.Read())
    {
        if (razmak == 0 || brojac % razmak == 0)
        {
            GeoSirina = reader.GetDouble(0); //Latitude
            GeoDuzina = reader.GetDouble(1); //Longitude
            tocka = new PointLatLng(GeoSirina, GeoDuzina);
            GMapMarker marker = new GMarkerGoogle(tocka, OdabirMarkera(Convert.ToInt16(reader.GetValue(2))));
            GMapOverlay markeri;
            markeri = new GMapOverlay("markeri");
            markeri.Markers.Add(marker);
            gMap.Overlays.Add(markeri);
            gMap.Position = tocka;
        }
        brojac++;
    }
    reader.Close();
}
```

Slika 21. Isječak koda koji prikazuje metodu koja se pokreće nakon pritiska na gumb „Prikaži putanju“
Izvor: izradio autor

Sljedeći postupak ide skladno linijama koda na slici 21.:

1. Čisti se *overlay* kako bi se maknuli svi markeri od prethodno iscrtanih putanja.
2. Poziva se karta koju želimo koristiti.
3. Inicijalizira se nova veza na SQL Server i upisuje se *connection string* pomoću kojeg se VS spaja na SQL Server odnosno definira se na koji server i koju bazu na tome serveru se treba povezati.
4. Stvorena veza se otvara.
5. Inicijalizira se nova SQL naredba te se upisuje upit pomoću kojeg se filtriraju željeni podaci.
6. Pokreće se čitač i ovdje se izvršava prethodno definirani upit te su u ovom trenutku podaci spremni za čitanje.
7. Inicijaliziraju se varijable u koje će se pohranjivati koordinate.

8. Inicijaliziraju se varijable pomoću kojih se vrši optimizacija prikaza u smislu količine markera koju je potrebno iscrtati. Količina markera koju aplikacija može u jednoj iteraciji iscrtati na Google karti je oko 2000 stoga je za trajektorije koje imaju veći broj točaka potrebno iscrtati dovoljnu količinu da aplikacija može raditi bez poteškoća i da se zadrži prepoznatljivost trajektorije na karti. Za tu svrhu kreirana je posebna metoda koja prema broju zapisa pojedinog korisnika određuje koje će se točke ispisati.
9. *While* petlja koja otvara redak po redak iz rezultata prethodno predanog upita SQL Serveru i omogućava čitanje svih podataka u trenutno otvorenom retku.
10. Unutar petlje se provjerava uvjet prema kojem će se marker prikazati ili neće s obzirom na varijable brojac i razmak.
11. Unutar *if* grane se ponavlja već opisana procedura postavljanja markera.

Gumb označen brojem 5 pretražuje i iscrtava zapise odabranog korisnika koji imaju brzinu jednaku 0. Tako se dobivaju pretpostavljene lokacije željezničkih postaja za pojedinog korisnika.

Gumb označen brojem 6 pretražuje sve korisnike koji imaju mod prijevoza vlak i iscrtava zapise koji imaju brzinu jednaku 0. Tako se dobivaju pretpostavljene lokacije željezničkih postaja.

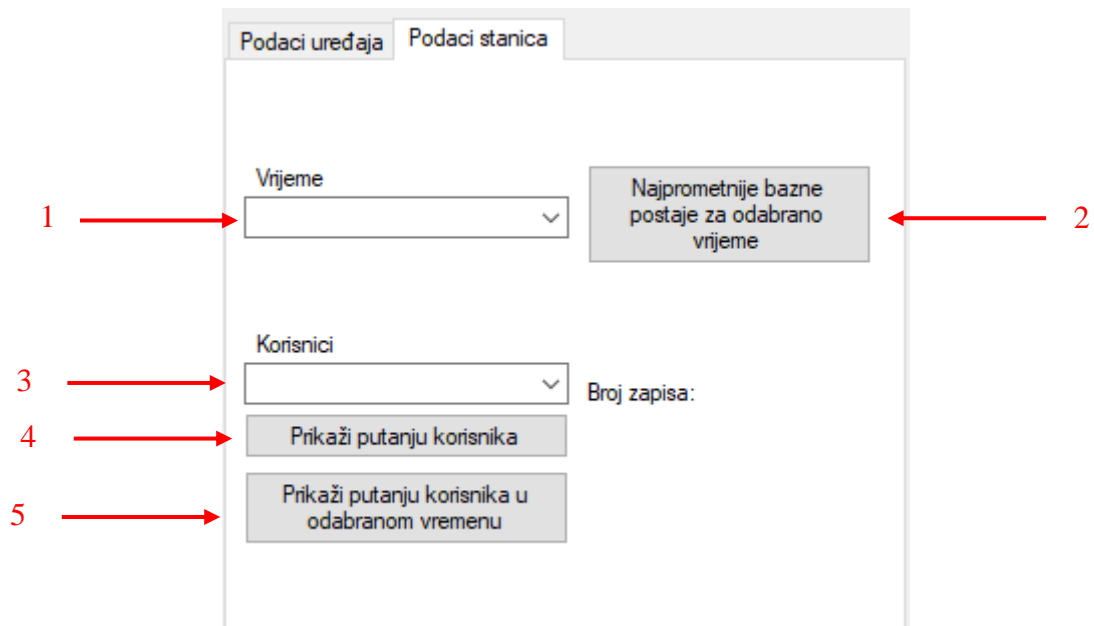
Naposljetku gumb pod brojem 7 prikazuje legendu u tekstualnom obliku, odnosno prikazuje odgovarajuće boje markera ovisno o modu prijevoza.

Na slici 8. moguće je vidjeti rezultat iscrtavanja rute korisnika, koji je filtriran prema modu vlak, ali se osim vlaka kretao autobusom i pješice.

5.2.3.3. Kontrole baze sa podacima baznih postaja

Mogućnosti koje pružaju kontrole vezane za bazu podataka PodaciStanice:

- Odabir vremenskog razdoblja prema kojem se žele filtrirati podaci
- Prikaz najprometnijih baznih postaja za odabrano vrijeme
- Odabir korisnika te prikaz pretpostavljene putanje korisnika kroz cijeli dan i kroz odabrano vremensko razdoblje.



Slika 22. Prikaz kontrola koje omogućuju izvršavanje upita nad bazom podataka sa baznih stanica

Izvor: izradio autor

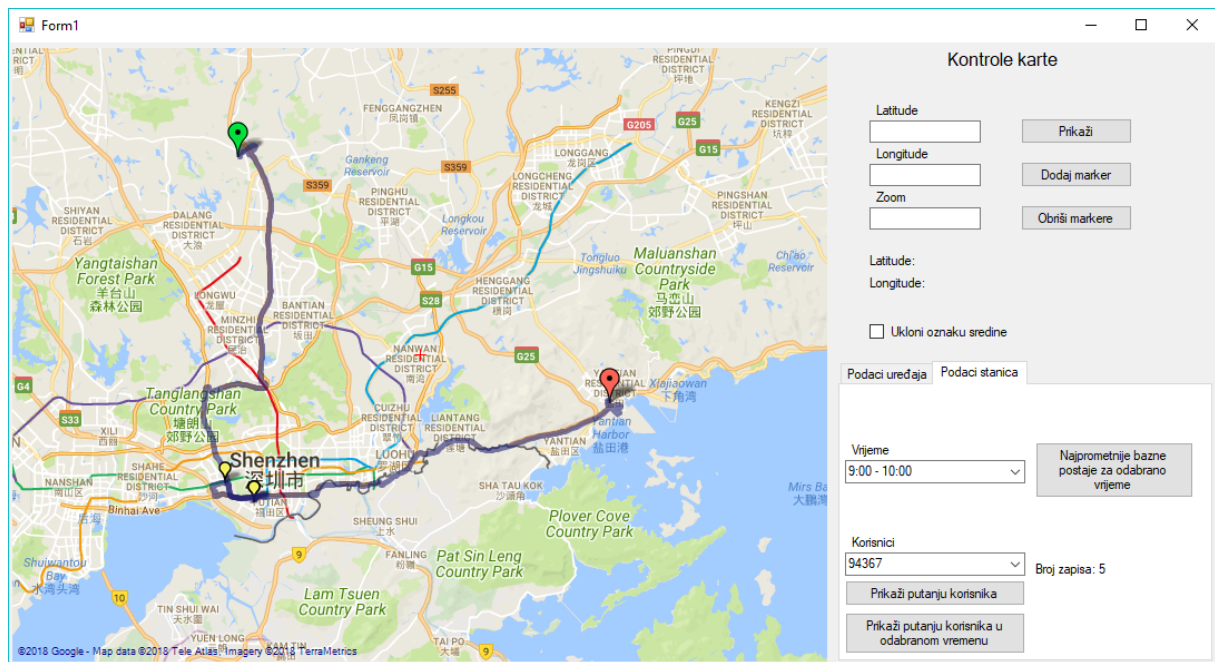
Brojem 1 označen je padajući izbornik u kojemu se odabire vremensko razdoblje za koje se želi prikazati bazne postaje ili putanju korisnika

Brojem 2 označen je gumb čijim se pritiskom iscrtava 10 najprometnijih baznih postaja u odabranom razdoblju

Brojem 3 označen je padajući izbornik u kojemu je moguće odabrati pojedinog korisnika za kojega želimo prikazati putanju, te s desne strane *label* kontrola koja prikazuje broj zapisa za odabranog korisnika.

Gumb označen brojem 4 iscrtava putanju korisnika kroz cijeli dan

Gumb označen brojem 5 iscrtava putanju korisnika samo u odabranom vremenu.



Slika 23. Prikaz grafičkog sučelja sa svim kontrolama
 Izvor: izradio autor

Na slici 23. prikazan je primjer iscrtavanja pretpostavljene putanje odabranog korisnika u vremenskom razdoblju od 9:00 do 10:00 sati.

6. Zaključak

U završnom radu obrađena su dva različita javno dostupna skupa podataka. Jedan skup podataka predstava GPS zapise o kretanju korisnika te je sadrži i podatke poput brzine, akceleracije i mod prijevoza. Drugi skup sadrži zapise korisnika po baznim postajama GSM mreže. Za oba skupa podataka izrađene su relacijske baze podataka sa pripadajućim tablicama u koje su dodani podaci. Podatke je prije pohrane u bazu podataka trebalo prilagoditi dostupnim tipovima podataka kako bi se uspješno pohranili. Pri izradi baze podataka provjerena je usklađenost sa normalnim formama, pa su podaci i u tom smislu obrađeni kako bi se minimalizirala zalihost i eliminirala dvosmislenost podataka.

Kreirana baza podataka povezana je sa izrađenim grafičkim korisničkim sučeljem čija je osnovna funkcionalnost prikaz pohranjenih podataka i dostupnih parametara. Za oba skupa podataka prilagođene su kontrole kako bi se prikazale specifičnosti unutar skupa podataka, pa je tako podatke iz prvog skupa moguće filtrirati po modu prijevoza. Različiti modovi prijevoza prikazuju se markerima različitih boja. Modove prijevoza moguće je klasificirati prema značajkama poput prosječne brzine ili akceleracije iz dostupnih podataka. Na primjeru detekcije željezničkih postaja prikazano je kako je moguće detektirati usporenja čak i na vizualnom prikazu. Drugi analizirani skup podataka sadrži podatke koji, iako su isto lokacijski, su bitno različiti. Tako je i grafički prikaz tih podataka sa baznih postaja temeljen na davanju različitih informacija od onih za prvi skup podataka. Za ovaj skup podataka grafičkim sučeljem omogućen je prikaz najprometnijih baznih postaja u odabranom vremenskom intervalu tijekom dana. Osim toga, prikazuje se aproksimativna putanja kretanja korisnika na temelju prijave na pojedinu baznu stanicu.

Vizualizacija i klasifikacija trajektorija omogućuje uočavanje uzoraka i navika u kretanju korisnika. Informacije tog tipa imaju široku primjenu. Primjerice analizom podataka i prikazom najprometnijih baznih postaja može se zaključiti gdje su najveće koncentracije korisnika i pomoću toga planirati unaprjeđenje i daljnji razvoj pokretne mreže.

Popis literature

Tiskani članak u časopisu:

1. Elmasri, R., Navathe, S., B.: Fundamentals of Database Systems 6th Edition, Addison-Wesley, 2010.
2. Desheng Zhang, Juanjuan Zhao, Fan Zhang, Tian He UrbanCPS: a Cyber-Physical System based on Multi-source Big Infrastructure Data for Heterogeneous Model Integration. ICCPS '15, April 14 - 16, 2015, Seattle, WA, USA
3. Ameen, Jalal. Cell planning in GSM mobile. WSEAS TRANSACTIONS on COMMUNICATIONS
4. M. Aal-Nouman, H. Takruri-Rizk and M. Hope, "Efficient Communications for Location-Based Services Using Spare Extensions of Control Channels in Mobile Networks," 2016 8th IFIP International Conference on New Technologies, Mobility and Security (NTMS), Larnaca, 2016, pp. 1-6.

Internetski izvori:

5. <http://docbook.rasip.fer.hr/ddb/res/45/Ch2.html> (pristupljeno 29.08.2018.)
6. <https://www.microsoft.com/en-us/research/publication/geolife-gps-trajectory-dataset-user-guide/> (pristupljeno 29.08.2018.)
7. <http://www.fpz.unizg.hr/ztos/PRSUS/GSM.pdf> (pristupljeno 29.08.2018.)
8. <https://www-users.cs.umn.edu/~tianhe/BIGDATA/> (pristupljeno 02.09.2018.)
9. <https://www.techopedia.com/definition/3235/object-oriented-programming-oop> (pristupljeno 02.09.2018.)
10. <https://www.nuget.org/> (pristupljeno 07.2018.)

Ostali izvori:

11. Autorizirana predavanja iz kolegija Baze podataka: https://moodle.srce.hr/2017-2018/pluginfile.php/1321190/mod_resource/content/6/2.tjedan.pdf (pristupljeno 20.08.2018)
12. Autorizirana predavanja iz kolegija Baze podataka: https://moodle.srce.hr/2017-2018/pluginfile.php/1321250/mod_resource/content/3/11.tjedan.pdf (pristupljeno 29.08.2018.)
13. Autorizirana predavanja iz kolegija Mobilne komunikacije: https://moodle.srce.hr/2017-2018/pluginfile.php/1622992/mod_resource/content/1/04-Mobilne%20komunikacije.pdf (pristupljeno 29.08.2018.)

Popis slika

Slika 1. Prikaz preuzetih podataka o kretanju GPS korisnika	3
Slika 2. Prikaz arhitekture GSM mreže	4
Slika 3. Prikaz ćelijske strukture GSM sustava	5
Slika 4. Prikaz preuzetih podataka o baznim postajama.....	6
Slika 5. Skripte za kreiranja baza podataka, GPS podaci korisnika (lijevo) i podaci baznih stanica korisnika (desno).....	11
Slika 6. Prikaz relacijskih shema baza podataka, GPS podaci (lijevo) i podaci baznih stanica (desno).....	12
Slika 7. Rezultat izvršavanja upita kojim se dodaju podaci u tablicu Mod prijevoza	13
Slika 8. Prikaz dijela putanje kretanja korisnika modom prijevoza vlak.....	15
Slika 9. Prikaz putanje jednog korisnika s više modova prijevoza.....	16
Slika 10. Prikaz pretpostavljene rute kretanja korisnika na temelju posjećenih baznih postaja	17
Slika 11. Prikaz rute prema posjećenim baznim stanicama jednog korisnika	18
Slika 12. Prikaz sučelja Microsoft Visual Studio razvojnog okruženja	22
Slika 13. Prikaz prozora Toolbox GMap kontrola.....	23
Slika 14. Prikaz osnovnih kontrola grafičkog sučelja upravljanje geografskom kartom	25
Slika 15. Isječak koda koji prikazuje metodu koja se pokreće nakon pritiska gumba „Prikaži“	26
Slika 16. Isječak koda koji prikazuje metodu koja se pokreće nakon pritiska gumba „Dodaj Marker“	26
Slika 17. Isječak koda koji prikazuje metodu koja se pokreće nakon pritiska gumba „Obriši markere“	27
Slika 18. Isječak koda koji prikazuje metodu koja se pokreće nakon pritiska lijeve tipke miša na kartu	27
Slika 19. Grafički prikaz korištenja kontrola karte.....	28
Slika 20. Prikaz kontrola koje omogućuju izvršavanje upita nad bazom GPS podataka	29
Slika 21. Isječak koda koji prikazuje metodu koja se pokreće nakon pritiska na gumb „Prikaži putanju“	30
Slika 22. Prikaz kontrola koje omogućuju izvršavanje upita nad bazom podataka sa baznih stanica.....	32
Slika 23. Prikaz grafičkog sučelja sa svim kontrolama	33

Popis kratica

GPS – Global Positioning System

GSM – Global System for Mobile Communications / Groupe Special Mobile

GPRS – General Packet Radio Services

TDMA – Time Division Multiple Access

FDD – Frequency Division Duplex

EDGE – Enhanced Data Rates for GSM Evolution

SIM – Subscriber Identity Module

ME – Mobile Equipment

BTS – Base Transciever Station

BSC – Base Station Controler

MSC – Mobile Switching Centre

DBMS – Database Management System

SQL – Search Query Language

LBS – Location Based Service

OTT – Over the Top usluge

VS – Visual Studio



Sveučilište u Zagrebu
Fakultet prometnih znanosti
10000 Zagreb
Vukelićeva 4

IZJAVA O AKADEMSKOJ ČESTITOSTI I SUGLASNOST

Izjavljujem i svojim potpisom potvrđujem kako je ovaj završni rad

isključivo rezultat mog vlastitog rada koji se temelji na mojim istraživanjima i oslanja se na objavljenu literaturu što pokazuju korištene bilješke i bibliografija.

Izjavljujem kako nijedan dio rada nije napisan na nedozvoljen način, niti je prepisan iz necitiranog rada, te nijedan dio rada ne krši bilo čija autorska prava.

Izjavljujem također, kako nijedan dio rada nije iskorišten za bilo koji drugi rad u bilo kojoj drugoj visokoškolskoj, znanstvenoj ili obrazovnoj ustanovi.

Svojim potpisom potvrđujem i dajem suglasnost za javnu objavu završnog rada
pod naslovom **Prikaz modova kretanja korisnika GSM mreže**

Visualization of GSM Users Transport Modes Trajectories

na internetskim stranicama i repozitoriju Fakulteta prometnih znanosti, Digitalnom akademskom repozitoriju (DAR) pri Nacionalnoj i sveučilišnoj knjižnici u Zagrebu.

U Zagrebu, 10.9.2018

Student/ica:

Anđelija Tomljanović
(potpis)