Primjer sustava za potporu odlučivanju u hoteljerstvu

Grbac, Ivica

Undergraduate thesis / Završni rad

2019

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Pula / Sveučilište Jurja Dobrile u Puli**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:137:314930

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-11-29**

Repository / Repozitorij:

Digital Repository Juraj Dobrila University of Pula

Sveučilište Jurja Dobrile u Puli
Fakultet informatike

IVICA GRBAC

PRIMJER SUSTAVA ZA POTPORU ODLUČIVANJU U HOTELIJERSTVU

Završni rad

Pula, 2019.

Sveučilište Jurja Dobrile u Puli
Fakultet informatike

IVICA GRBAC

PRIMJER SUSTAVA ZA POTPORU ODLUČIVANJU U HOTELIJERSTVU

Završni rad

JMBAG: 0145032169, izvanredni student

Studijski smjer: Informatika

Predmet: Upravljački informacijski sustavi

Znanstveno područje: Društvene znanosti
Znanstveno polje: Informacijsko-komunikacijske znanosti
Znanstvena grana: Informacijski sustavi

Mentorica: prof. dr. sc. Vanja Bevanda

Pula, rujan 2019.

IZJAVA O AKADEMSKOJ ČESTITOSTI

Ja, dolje potpisani _____, kandidat za prvostupnika ekonomije/poslovne ekonomije, smjera _____ ovime izjavljujem da je ovaj Završni rad rezultat isključivo mojega vlastitog rada, da se temelji na mojim istraživanjima te da se oslanja na objavljenu literaturu kao što to pokazuju korištene bilješke i bibliografija. Izjavljujem da niti jedan dio Završnog rada nije napisan na nedozvoljen način, odnosno da je prepisan iz kojega necitiranog rada, te da ikoji dio rada krši bilo čija autorska prava. Izjavljujem, također, da nijedan dio rada nije iskorišten za koji drugi rad pri bilo kojoj drugoj visokoškolskoj, znanstvenoj ili radnoj ustanovi.

Student

_____

U Puli, _____, _____ godine

IZJAVA

o korištenju autorskog djela

Ja, _____ dajem odobrenje Sveučilištu Jurja Dobrile u Puli, kao nositelju prava iskorištavanja, da moj završni rad pod nazivom _____ _____ koristi na način da gore navedeno autorsko djelo, kao cjeloviti tekst trajno objavi u javnoj internetskoj bazi Sveučilišne knjižnice Sveučilišta Jurja Dobrile u Puli te kopira u javnu internetsku bazu završnih radova Nacionalne i sveučilišne knjižnice (stavljanje na raspolaganje javnosti), sve u skladu s Zakonom o autorskom pravu i drugim srodnim pravima i dobrom akademskom praksom, a radi promicanja otvorenoga, slobodnoga pristupa znanstvenim informacijama.

Za korištenje autorskog djela na gore navedeni način ne potražujem naknadu.

U Puli, _____ (datum)

Potpis

_____

SADRŽAJ

UVOD 1

1. PRIKAZ PARKIRALIŠTA SA SUSTAVOM PRIJEVOZA 2

2. KORISNIČKI ZAHTJEVI 4

3. SUSTAVNI ZAHTJEVI 5

3.1. Funkcionalni zahtjevi 5

3.2. Ne-funkcionalni zahtjevi 7

4. PROGRAMSKI ALATI I TEHNOLOGIJE KORIŠTENE PRILIKOM IZRADE APLIKACIJE 7

4.1. XAMPP 8

4.1.1. MariaDB 9

4.1.2. phpMyAdmin 10

4.2. NetBeans 12

5. KORIŠTENJE APLIKACIJE 17

6. ANALIZA PODATAKA 25

ZAKLJUČAK 28

POPIS LITERATURE 30

POPIS SLIKA 31

SAŽETAK 32

SUMMARY 33

UVOD

Svaki stanovnik turističkog mjesta ili naselja jako dobro zna kakve izazove i probleme sa sobom donosi turistička sezona. Jedan od tih je svakako i promet unutar urbanih sredina. Beskonačne kolone vozila koje se svakodnevno slijevaju u gradske prostore stvaraju velike gužve i zastoje na prometnicama, a kronični nedostatak parkirnih mjesta mnogo je izraženiji.

Nije rijetkost da hoteli smješteni na atraktivnim lokacijama uz samo more, u gradskim jezgrama i prostorima pod zaštitom UNESCO-a, nemaju dovoljno dostupnih parkirnih mjesta u odnosu na kapacitet. Nemaju adekvatne infrastrukture, promet je, gotovo, onemogućen ili zabranjen.

Jedno od mogućih rješenja koje bi donekle riješilo problem, uredilo gradski promet, rasteretilo prometnice i nedostatna parkirna mjesta je izgradnja parkiralištu s organiziranim sustavom prijevoza za goste hotela.

Iako bi se koristeći ovaj koncept mogla urediti i gradska parkirališta, za potrebe ovog rada govorit će se isključivo o parkiralištu hotela, te se samim time koncentrirati na problematiku vezanu uz takav način rada.

Samo parkiralište nalazilo bi se na lako pristupačnoj lokaciji na rubu ili u okolici grada koja nije predaleko od samih hotela i gradske jezgre, a dovoljno velika da omogući izgradnju optimalnog broja parkirnih mjesta za svaku smještajnu jedinicu po hotelu.

Na taj način gosti ne bi morali prometovati gradskim ulicama, rasteretile bi se gradske prometnice, a ostalo bi i više slobodnih parkirnih mjesta za izletnike i lokalno stanovništvo.

Uz prijevoz gostiju od parkirališta do hotela te od hotela do parkirališta, usluge se mogu proširiti na prijevoz do bolnice, autobusne stanice, ljekarne i ostalih lokacija čime bi se značajno podigla razina kvalitete i broj ponuđenih usluga.

Kako bi se lakše donosilo poslovne odluke na tjednoj bazi, te dugoročni planovi vezani uz pružanje ovakve usluge, moguće je istu poduprijeti aplikacijom koja će olakšati donošenje istih.

Nakon što je opisana problematika i rješenje kojim se ista želi riješiti, u sljedećem poglavlju se prikazuje rješenje koje nije podržano aplikacijom.

U drugom i trećem poglavlju se opisuju korisnički i sistemski zahtjevi koji moraju biti zadovoljeni kako bi aplikacija što bolje podržala korisnike pri izvođenju radnih obaveza.

Četvrto poglavlje se bavi alatima i tehnologijama koje su korištene prilikom izrade praktičnog djela ovog rada, odnosno opisom XAMPP korisničkog sučelja pomoću kojega se pristupa Apache serverskoj aplikaciji te MySQL-u. Kao razvojna okolina korištena je NetBeans IDE u kojoj se pomoću aplikacije JavaFX Scene builder kreiralo grafičko sučelje, dok se za kodiranje koristio JavaFX skriptni jezik.

U petom poglavlju se opisuje rad aplikacije, funkcionalnost elemenata, mogućnosti koje aplikacija pruža te ograničenja koja su implementirana u istu.

Nakon toga slijedi šesto poglavlje posvećeno analizi podataka i donošenju odluka te je za sam kraj iznesen zaključak. U nastavku su priloženi popis literature i popis slika.

1.PRIKAZ PARKIRALIŠTA SA SUSTAVOM PRIJEVOZA

Sustav prijevoza putnika funkcionira na način da je u svakome trenutku na dvije glavne lokacije, parkiralištu i hotelu, uvijek dežurno bar jedno putničko vozilo s vozačem. Također na obje lokacije je najmanje jedan djelatnik u svakom trenutku koji pomaže oko dogovora termina te animira goste dok čekaju prijevoz. Ovisno o broju dolazaka i odlazaka na dnevnoj bazi odlučuje se koliko je vozača i djelatnika parkirališta u smjeni. Također, vodi se računa da je dostupna potrebna količina vozila u periodima velike smjene gostiju i za vrijeme posebnih manifestacija.

Kako bi se postigla opuštena atmosfera od gostiju se ne traži točno vrijeme polaska, nego se gostu nudi usluga prijevoza u najkraćem vremenskom periodu od

zahtjeva za istom. Iznimka su prijevoz do autobusnog kolodvora, pristaništa i ostalih lokacija gdje gosti nastavljaju putovanje nekim drugim prijevoznim sredstvom te su uvjetovani vremenom polaska istog. U tim slučajevima gosti mogu prethodno rezervirati termin u točno određeno vrijeme te imaju prednost nad ostalim korisnicima usluge.

Kada gost dolazi na parkiralište, djelatnik parkirališta mu pomaže oko prtljage te objašnjava kako funkcionira sustav prijevoza. Pokazuje gostu gdje da parkira svoje vozilo. Nakon toga se gosti i njihova prtljaga smještaju u vozilo i kreću prema hotelu. Također, nakon odjave gosta iz hotela, osoblje pomaže gostu s prtljagom te nakon ukrcaja kreću prema parkiralištu.

Za vrijeme boravka u hotelu gost može, u bilo koje vrijeme, zatražiti vožnju do svojeg vozila te mu se isto omogućuje u najkraćem vremenskom roku. Dok s djelatnicima parkirališta dogovara vožnju do hotela nakon što se vrati s izleta, večere i slično.

Iznimka su gosti koji na odmor stižu autobusom, vlakom ili brodom. Njih vozač dočekuje na njihovom stajalištu i vozi direktno u hotel. Na odlasku vraća ih na istu lokaciju na kojoj ih je dočekao.

Vožnje do bolnice i hitne pomoći imaju prioritet nad ostalim uslugama pa prvi slobodni vozač mora goste u što kraćem vremenskom periodu prevesti do potrebne lokacije.

U slučaju dolaska velike grupe ili više gostiju u kratkom vremenskom periodu djelatnici preuzimaju i sortiraju prtljagu gostiju, nakon čega se ista šalje teretnim vozilom u hotel, kako bi se ubrzao prijevoz putnika na njihovo odredište. Na isti se način prtljaga vraća na odredište prilikom odlaska.

Pošto se ovakav način rada svodi većinom na zapisivanje dogovorenih termina na komadu papira ili usmenim dogovorom, lako može doći do komplikacija u pružanju usluge. Također vlasnika parkinga zanima koliko učestalo korisnici koriste uslugu prijevoza ne vezano uz dolazak na odmor ili odlazak s odmora te koje relacije u tim slučajevima koriste.

Kako bi korisnik dobio bolji uvid o potrebnoj količini vozila, djelatnika, troškovima pružanja usluga, te došao do novih saznanja kojima može poboljšati ovu uslugu odlučeno je da se ovaj sustav podrži aplikacijom.

2.KORISNIČKI ZAHTJEVI

Prije početka izrade samoga sustava potrebno je proučiti rad sustava kako bi se utvrdilo na koji način, te koji tip aplikacije je potreban za podržavanje rada istoga. Također potrebno je utvrditi koji su podaci potrebni budućim korisnicima aplikacije, te koje funkcionalnosti kako bi im se olakšao svakodnevni rad. To će se izvesti pomoću korisničkih te sustavnih zahtjeva koji su navedeni u nastavku rada.

Korisnički zahtjevi su izjave, pisane prirodnim jezikom uz priložene dijagrame, koje opisuju što se očekuje od usluga koje aplikacija pruža korisnicima te ograničenja pod kojima se odvijaju.[1]

Kroz razgovor s pružateljem usluga, te djelatnicima samog sustav i analizom istog definirani su sljedeći korisnički zahtjevi :

1. Korisnik kako bi pristupio aplikaciji mora posjedovati vlastito korisničko ime i lozinku.
2. Korisniku treba biti omogućen pregled podataka, unos podataka, izmjena te brisanje podataka.
3. Ograničenja pri unosu te izmjeni podataka će se integrirati u aplikaciju kako bi se izbjegle greške prilikom izvođenja istih.
4. Aplikacija će omogućiti filtriranje prikaza podataka po vremenskom periodu i tipu vozila kako bi se olakšalo snalaženje među istima.
5. Aplikacija će pružati korisniku mogućnost izrade grafičkih prikaza podataka, konkretnije prikaze pomoću grafikona, po parametrima odabranima od strane korisnika.
6. Pošto će aplikaciju koristiti samo radnici odlučeno je da se izradi desktop aplikacija s lokalnom bazom podataka.

[1] I.Sommerville (2010.) Software engineering, 9th edition, str.82

3.SUSTAVNI ZAHTJEVI

Sustavni zahtjevi su detaljniji opisi sistemskih funkcija, usluga, te operacijskih ograničenja. Dokument sustavnih zahtjeva morao bi točno definirati što će se implementirati. Može se koristiti i kao dio ugovora između prodavača i kupca aplikacije.[2]

Softverski sustavni zahtjevi se često dijele na funkcionalne zahtjeve ili ne-funkcionalne zahtjeve. Funkcionalne zahtjevi su izjave o uslugama koje sistem pruža, kako sistem reagira na određene unose, te kako se sistem ponaša u određenim situacijama. Također sustavni zahtjevi opisuju što sustav ne bi trebao raditi.[3]

Ne-funkcionalni zahtjevi su ograničenja vezana uz usluge ili funkcije koje sustav pruža. Uključuju vremenska ograničenja, ograničenja vezana uz razvojne procese, te ograničenja definirana u standardima. Ne-funkcionalni zahtjevi se češće odnose na sustav kao cjelinu nego na individualnu funkciju ili uslugu sustava.[4]

3.1 Funkcionalni zahtjevi

Na temelju zahtjeva korisnika definirani su sljedeći funkcionalni zahtjevi :

1. Korisnik može aplikaciji pristupiti isključivo korištenjem svoga korisničkog imena te svojom lozinkom.
2. Korisnik u svakome trenutku mora imati mogućnost unosa, pregleda, izmjene i brisanja termina na bazi podataka.
3. Kod unosa termina korisnik mora ispuniti sve podatke kako bi unos bio izvršen. Konkretnije korisnik mora odabrati tip vozila, količinu, vozila, datum, vrijeme, polazište,odredište, broj putnika, količinu prtljage.
4. Aplikacija limitira korisnike pri unosu sljedećih podataka:
   4.1 Pri odabiru putničkog vozila korisnik može odabrati definiran broj putnika (1-8).

[2] loc. cit.
[3] ibidem, str. 84
[4] ibidem, str. 85

4.2 Pri odabiru putničkog vozila korisnik može unijeti maksimalno definiranu količinu prtljage (0-10).

4.3 Pri odabiru teretnog vozila korisnik može unijeti količinu u definiranom intervalu vrijednosti (10-50).

4.4 Korisnik nakon odabira lokacije kao polazišta nema mogućnost odabira iste kod odabira odredišta.

4.5 Kod odabira teretnog vozila korisnik nema mogućnost unosa podataka za broj putnika.

5. Ovisno o odabranoj količini vozila prilagođavaju se ograničenja vezana uz unos broja putnika te količine prtljage.
6. Korisnik nema ograničenja pri odabiru datuma pošto termini mogu biti uneseni naknadno.
7. Kod izmjene podataka, korisnik mora odabrati redak koji će izmijeniti, te odabrati opciju izmjena, nakon čega se otvara pogled za unos s već unesenim podacima koje može izmijeniti.
8. Kod brisanja podataka korisnik odabire redak koji želi izbrisati, te nakon odabira opcije izbriši se pojavljuje skočni prozor gdje korisnik potvrđuje brisanje. Na ovaj način se izbjegava pogreška pri brisanju.
9. Pri korištenju pregleda korisniku će biti omogućeno filtriranje podataka po određenim parametrima radi lakšeg snalaženja i boljeg pregleda.
10. Aplikacija će korisniku omogućiti analizu podataka pomoću zaslona za analizu.
11. Zaslon za analizu će pružati korisniku mogućnost izrade grafičkih prikaza podataka po tipu vozila te broju vožnji. Također korisnik bira vremenski raspon, odnosno početni te završni datum, u kojem želi promatrati podatke.
12. Korisnik ima mogućnost pregleda podataka na razini godine, mjeseca ili dana.
13. Korisniku se omogućava grafički prikaz odabranih podataka korištenjem linijskog ili stupčastog grafikona.
14. Korisniku mora biti omogućen jednostavan prijelaz sa zaslona na zaslon, kako bi što jednostavnije koristio aplikaciju.

3.2 Ne-funkcionalni zahtjevi

1. Aplikacija će biti razvijena kao desktop aplikacija s lokalnom bazom podataka te će biti dostupna samo na računalu na kojem je instalirana.
2. Teži se jednostavnosti i preglednosti u izgledu svih ekrana, kako bi aplikacija bila što jednostavnija za korištenje samim korisnicima.
3. Administrator sustava je zadužen za unos korisničkog imena i lozinke na bazu kako bi omogućio rad novim korisnicima.
4. Administrator je zadužen za unos i brisanje dostupnih vozila na bazi podataka ovisno o potrebama poslovanja, te dostupnosti istih.
5. Baza podataka bit će rađena u XAMPP Control panelu.
6. Aplikacija će se razvijati u razvojnoj okolini NetBeans korištenjem skriptnog jezika Java FX, dok će se za izradu grafičkog sučelja koristiti JavaFX Scene Builder.

4.PROGRAMSKI ALATI I TEHNOLOGIJE KORIŠTENE PRILIKOM IZRADE APLIKACIJE

U ovom poglavlju predstavit će se svi alati te tehnologije koje su korištene prilikom izrade aplikacije. Korisničko sučelje XAMPP koje se sastoji od skupine alata za rad sa serverom te izradu i upravljanje bazom podataka. Razvojna okolina NetBeans u kojoj će biti pisan programski kod pomoću JavaFX skriptnog jezika te JavaFX Scene Builder pomoću kojeg će biti izrađeno grafičko sučelje.

4.1 XAMPP

XAMPP je besplatna više-platformska, web server, otvorenog koda skupina alata razvijena od Apache Friends-a. Kratica XAMPP stoji za više-platformski (X), Apache (A), MySQL (M), PHP (P) i Perl (P). To je jednostavan, lagani Apache server koji ekstremno olakšava developerima kreiranje lokalnog http (eng. hypertext transfer protocol) servera u nekoliko klikova.[5]

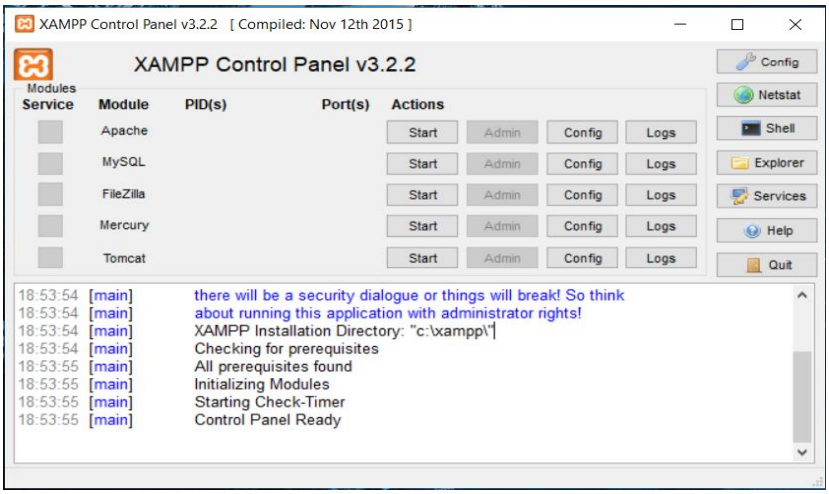
U XAMPP-u Apache je serverska aplikacija, MySQL (eng. structured query language) je baza podataka, dok je PHP (eng. hypertext preprocessor) skriptni jezik na poslužitelju i sve to skupa je uključeno kao izvršni/ekstrahirani dokument. XAMPP radi jednako na Windowsima, Mac-u te Linux-u zbog svoje više-platformske prirode.[6]

Pošto XAMPP sadrži sve Apache serversku aplikaciju te MySQL te je za pokretanje istih dovoljan jedan klik mišem odabran je kao idealno rješenje pri radu sa serverom, te kreiranje baze podataka u ovome radu.

XAMPP je dizajniran pretežito za developere i pruža jako jednostavan način developerima za postavljanje vlastitih, lokalnih LAMP instalacija za web bazirane aplikacije koje razvijaju. Početna XAMPP instalacija također sadrži više PHP aplikacija poput OpenSSL koji se koristi za protokole za prijenos zaštitno kodiranih podataka, grafičkih biblioteka poput GD, te paketa baza podataka poput SQLite, phpMyAdmin i drugih.[7]

XAMPP pruža korisniku jednostavnu upravljačku ploču, koja omogućuje korisniku pokretanje i prekidanje modula poput Apache, MySQL, FileZilla te pristupnih točaka, konfiguriranje datoteka, itd. Upravljačka ploča lagana za korištenje te ima sučelje prilagođeno korisniku.[8]

[5]TechnoSIP.com, Dostupno na: https://www.technosip.com/toolsreviews/technosip-review-xampp/ [Pristupljeno:1. kolovoza 2019.]
[6] loc. cit.
[7] loc. cit.
[8] loc. cit.



Slika 1. -Korisničko sučelje XAMPP

Izvor: Screenshot autora

Za potrebe aplikacije iz ovog seta aplikacija bit će korišteni Apache serverska aplikacija te MySQL odnosno MariaDB za relacijsku bazu podataka. O MariaDB bit će više riječi u nastavku.

4.1.1 MariaDB

MariaDB server je jedan od najpopularnijih servera baze podataka na svijetu. Razvijen je od originalnih developera koji su razvili MySQL s garancijom da će ostati otvorenog koda. Kao najpoznatiji korisnici navode se Wikipedia, WordPress.com i Google.[9]

MariaDB pretvara podatke u strukturirane informacije u širokom nizu aplikacija, od aplikacija za bankarsko poslovanje do Internet stranica. Smatra se poboljšanom zamjenom za MySQL. MariaDB se koristi jer je brza, skalabilna i robusna, s bogatim ekosistemom mehanizama za skladištenje, pluginova te mnoštvom ostalih alata koji je čine pogodnom za uporabu u širokom području.[10]

[9]MariaDB.org, Dostupno na: https://mariadb.org/about/ [Pristupljeno:2. kolovoza 2019.]
[10] loc. cit

4.1.2 phpMyAdmin

phpMyAdmin je besplatan softverski alat pisan u PHP-u, namijenjen za rukovanje te administraciju nad MySql-om. phpMyAdmin podržava širok spektar operacija (upravljanje bazom podataka, tablicama, stupcima, relacijama, indeksima, korisnicima, dozvolama, itd.) može se koristiti na korisničkom sučelju, dok i dalje dozvoljava direktno izvođenje bilo koje SQL naredbe.[11]

Pomoću phpMyAdmin alata izrađena je i baza podataka pod nazivom „prijevoz“ potrebna za rad aplikacije. Baza podataka se sastoji od tri tablice : „Termin“,“Vozila“ te „Polaziste_Odrediste“, koje će uz pomoć slika biti objašnjene u nastavku rada.

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
☐	2	tip_vozila	int(2)			No	None			Change Drop More
☐	3	kolicina_vozila	int(2)			No	None			Change Drop More
☐	4	br_putnika	int(2)			No	None			Change Drop More
☐	5	kolicina_prtljage	int(2)			No	None			Change Drop More
☐	6	polaziste	int(2)			No	None			Change Drop More
☐	7	odrediste	int(2)			No	None			Change Drop More
☐	8	datum	datetime			No	None			Change Drop More

Slika 2. -Prikaz tablice „termin“

Izvor: Screenshot autora

Na slici 2. možemo vidjeti da se tablica „termin“ sastoji od osam stupaca. Prvi stupac je „id“ koji određuje jedinstvenost tablice te je ujedno i primarni ključ, drugi stupac je „tip_vozila“ koji je ujedno vanjski ključ tablice „vozila“, odnosno poveznica na spomenutu tablicu. Treći stupac je „kolicina_ vozila“, dok je četvrti stupac „br_putnika“. Peti stupac je „kolicina_prtljage“. Dok su šesti stupac i sedmi stupac, odnosno „polazište“ i „odredište“ vanjski ključevi na tablicu „polaziste_odrediste“. Dok je osmi stupac „datum“. Na slici su također prikazani svi tipovi podataka po pripadajućim stupcima.

[11]PhpMyAdmin.net, Dostupno na: https://www.phpmyadmin.net/ [Pristupljeno: 4. kolovoza 2019.]

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(2)			No	None		AUTO_INCREMENT	Change Drop More
2	tip_vozila	varchar(50)	utf8_croatian_mysql561_ci		No	None			Change Drop More
3	br_putnika	int(2)			No	None			Change Drop More
4	kolicina_prtljage	int(2)			No	None			Change Drop More
5	kolicina	int(2)			No	None			Change Drop More

Slika 3. -Prikaz tablice „vozila“

Izvor: Screenshot autora

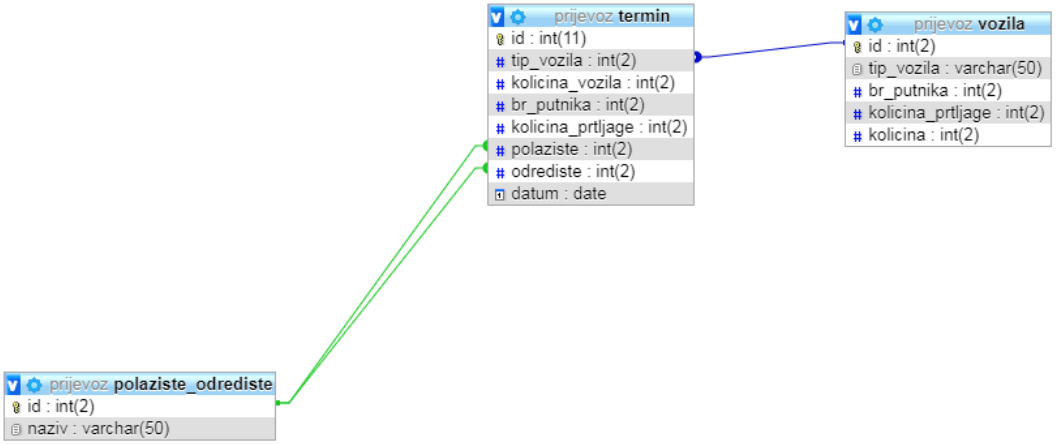
Na slici 3. možemo vidjeti tablicu „vozila“ te njezinu strukturu. Tablica se sastoji od pet stupaca, od kojih je prvi „id“ koji je ujedno i primarni ključ. Ostali stupci su „tip_vozila“, „br_putnika“, „kolicina_prtljage“ te „kolicina“. Također sa slike je vidljivo da stupac „tip_vozila“ uz tip podatka je i dodatak koji omogućuje upis hrvatskih znakova.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(2)			No	None		AUTO_INCREMENT	Change Drop More
2	naziv	varchar(50)	utf8_croatian_mysql561_ci		No	None			Change Drop More

Slika 4. -Prikaz tablice „polaziste_odrediste“

Izvor: Screenshot autora

Sa slike 4. moguće je vidjeti da se tablica „polaziste_odrediste“ sastoji od dva stupca. Prvi je „id“ koji je ujedno i primarni ključ, dok je drugi „naziv“. Također je vidljivo da stupac „naziv“ podržava korištenje hrvatskih znakova.



Slika 5. -Prikaz baze podataka

Izvor: Screenshot autora

Na slici 5. prikazana je baza podataka „prijevoz“ sa prethodno opisanim tablicama te njihovim relacijama.

4.2 NetBeans

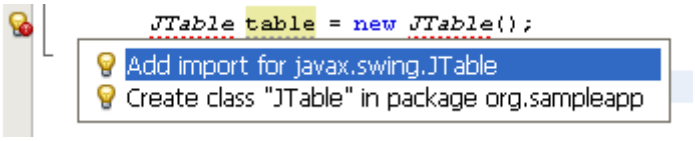
NetBeans razvojna okolina je projekt otvorenog koda posvećen pružanju solidnih produkata za razvoj softvera koji ispunjavaju potrebe developera, korisnika i poslovanja koji se oslanjaju na NetBeans kao bazu njihovih proizvoda. Specifično, da omogući brz razvoj proizvoda kroz efikasno i jednostavno korištenje Java platforme te ostalih relevantnih industrijskih standarda.[12]

Svrha svake razvojne okoline je maksimiziranje produktivnosti i podrška pri pisanju koda uz jedan alat. U nastavku će biti opisane neke pomoćne mogućnosti, opcije prilagodbe, te navigacijske mogućnosti NetBeans razvojne okoline.[13]

[12] NetBeans.org, Dostupno na: https://netbeans.org/about/ [Pristupljeno: 2. kolovoza.2019.]
[13]NetBeans.org, Dostupno na: https://netbeans.org/kb/73/java/editor-codereference.html#hints [Pristupljeno: 2. kolovoza. 2019.]

NetBeans Java Editor pomaže pri brzom kompletiranju i generiranju koda pomoću opcije „pametno“ kompletiranje koda. U globalu, kompletiranje koda je korisno pri brzom nadomještanju nedostajućeg koda, te generiranju blokova koda kada je to potrebno.[14]

Također je korisna opcija upravljanje unosima. Postoji nekoliko načina kako se može raditi s unosima. Java Editor u razvojnoj okolini stalno provjerava kod te dali se pravilno koriste importirane izjave, te odmah obavještava ako potrebna klasa ili izjava nisu unesene.[15]

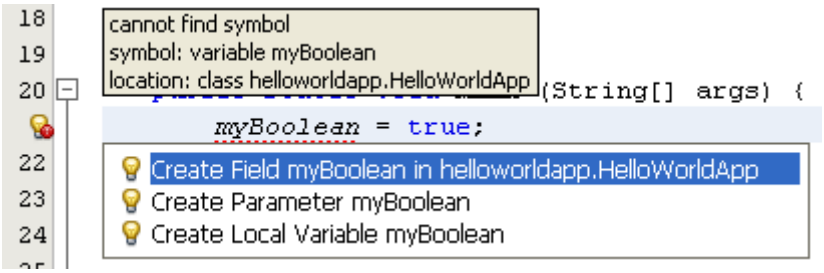


Slika 6. -Prikaz upozorenja o unosu

Izvor: NetBeans

(https://netbeans.org/kb/73/java/editor-codereference.html#hints)

Opcija korištenje savjeta funkcionira na način da dok se piše kod, Java Editor provjerava već napisani dio koda te predlaže sugestije kako ispraviti greške te poboljšati navigaciju kroz kod.[16]



Slika 7. Prikaz sugestije za ispravak greške

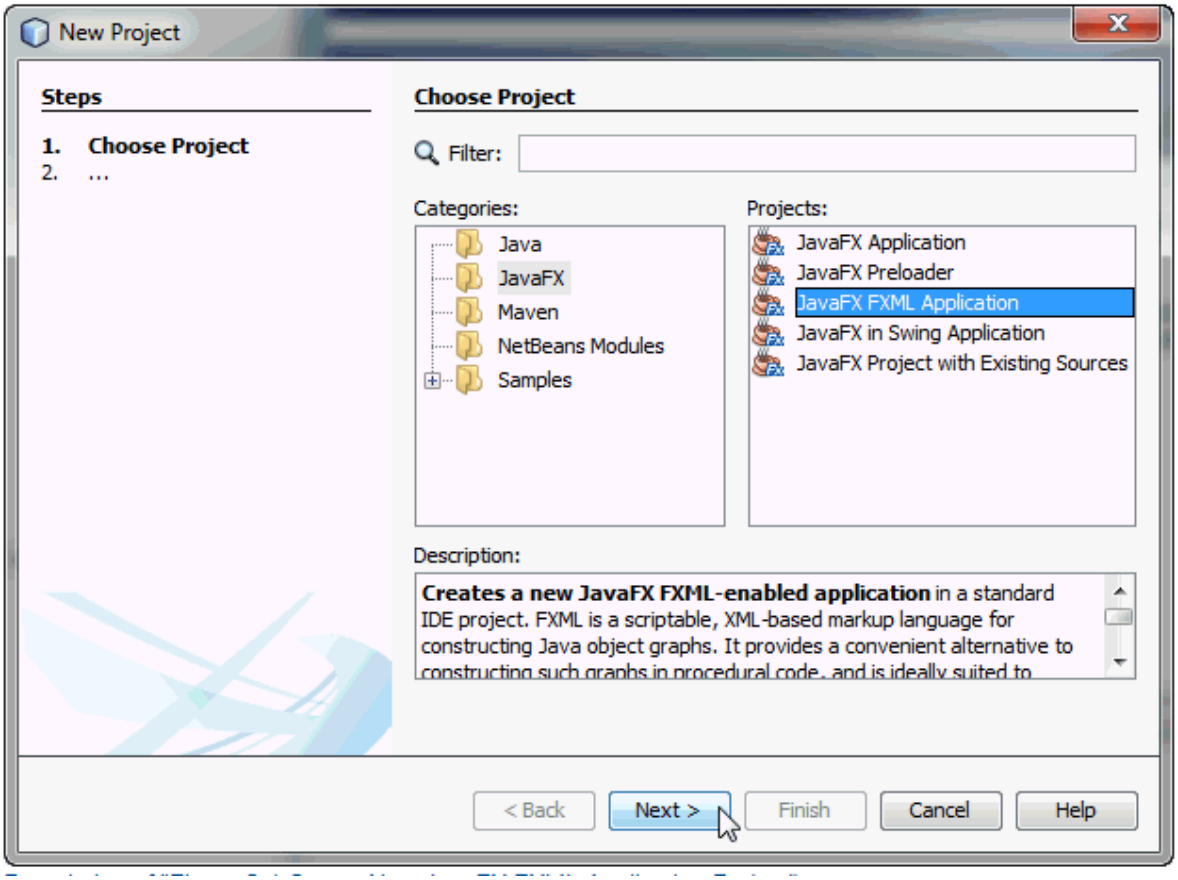
Izvor: NetBeans

(https://netbeans.org/kb/73/java/editor-codereference.html#hints)

[14] loc. cit.
[15] loc. cit.
[16] loc. cit.

Za potrebe ovog rada, to jest izrade praktičnog dijela koristit će se NetBeans razvojna okolina u kombinaciji s JavaFX Scene Builder aplikacijom. JavaFX Scene Builder je odabran pošto pruža mogućnost izrade JavaFX FXML aplikacije, odnosno JavaFX projekta koji je baziran na FXML (eng. xml-based user interface markup language) izgledu.

U NetBeans glavnom izborniku korisnik mora odabrati opciju „Novi projekt“ pri čemu se otvara dijaloški okvir. Nakon čega korisnik bira JavaFX kategoriju te JavaFX FXML Aplikacija pod projekt. Kao što je prikazano na slici 8.



Slika 8. -Novi projekt u NetBeans-u

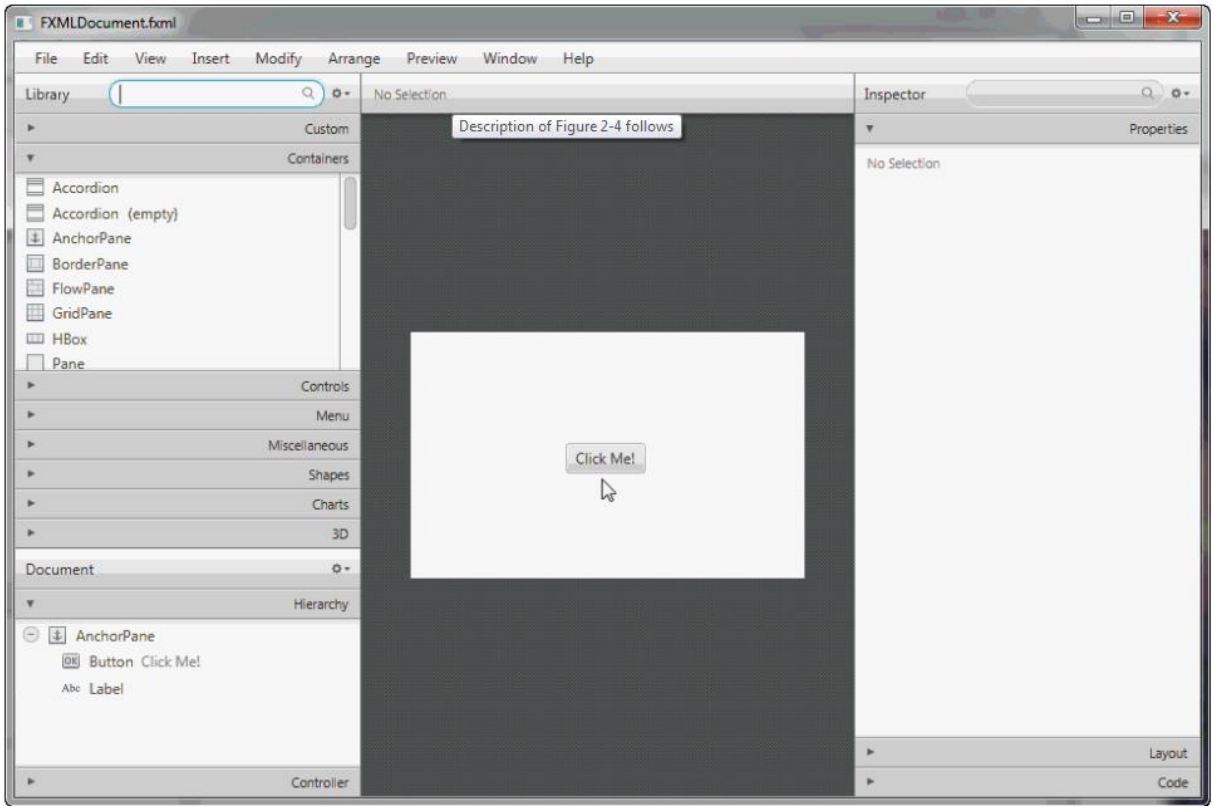
Izvor: Oracle Java documentation

(https://docs.oracle.com/javase/8/scene-builder-2/work-with-java-ides/sb-with-nb.htm?fbclid=IwAR1B7XIlzW_tLjlBZ4Z6tElQbXTAW0C-5ydHPubDR4E5LLH8k3DMTnF0KVU#JSBID108)

Nakon što korisnik kreira novi projekt može modificirati FXML dokument korištenjem Scene Buildera.

JavaFX Scene Builder je alat za izradu vizualnog izgleda aplikacije koji omogućuje korisniku da brzo dizajnira korisničko sučelje JavaFX aplikacije bez kodiranja. Korisnici mogu metodom povlačenja i ispuštanja (eng. drag and drop) komponenti korisničkog sučelja na radnu površinu, modificirati njihova svojstva, aplicirati liste stila (eng. style sheets), dok se FXML kod za izgled koji se gradi automatski generira u pozadini. Kao rezultat se dobiva FXML dokument koji se može kombinirati s Java projektom na način da se korisničko sučelje uveze u aplikacijsku logiku.[17]

[17]Oracle.com, Dostupno na: https://www.oracle.com/technetwork/java/javase/downloads/javafxscenebuilder-info-2157684.html [Pristupljeno: 5.kolovoza 2019.]



Slika 9. -Primjer FXML datoteke u Scene Builderu

Izvor**:** Oracle Java documentation

(https://docs.oracle.com/javase/8/scene-builder-2/work-with-java-ides/sb-with-nb.htm?fbclid=IwAR1B7XIlzW_tLjlBZ4Z6tElQbXTAW0C-5ydHPubDR4E5LLH8k3DMTnF0KVU#JSBID108)

Uz NetBeans kao razvojnu okolinu te JavaFX Scene Builder za izradu korisničkog sučelja pri izradi aplikacije još se koristi i JavaFX koja je API (eng. application programming interface) za Javu. API odnosno aplikacijsko programsko sučelje možemo definirati kao skup pravila i specifikacija koje omogućuju developerima da se koriste resursima i uslugama nekog drugog složenog programa, odnosno funkcijama, metodama i procedurama.

JavaFX jezik je deklarativni skriptni jezik statičkog tipa. Sadrži klasne funkcije (eng. class functions), deklarativnu sintaksu, razumijevanje lista (eng. list-comprehension), te inkrementiranu ovisno baziranu evaluaciju (eng. dependency-based evaluation). JavaFX intenzivno koristi Java2D Swing GUI (eng. graphical user interface) komponente te omogućuje jednostavno kreiranje grafičkih korisničkih sučelja.[18]

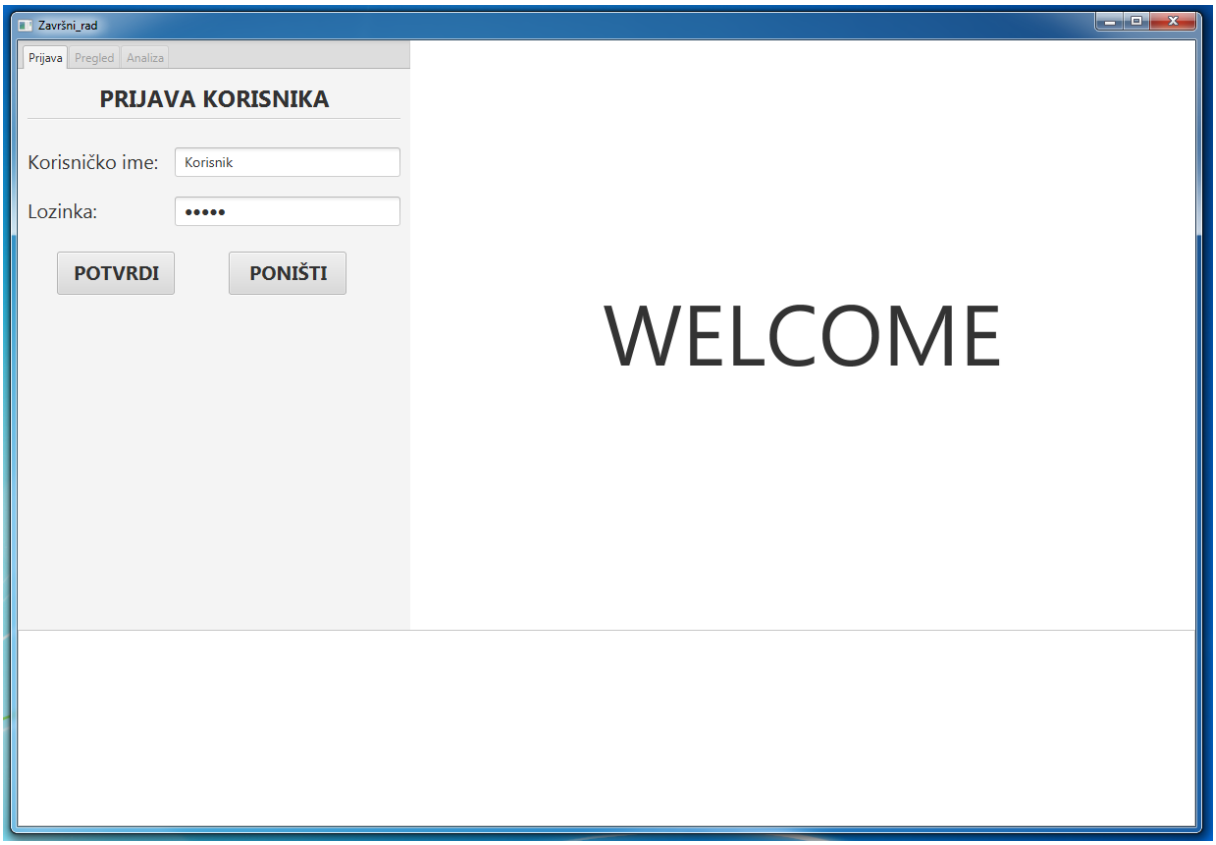
5.KORIŠTENJE APLIKACIJE

Aplikacija se sastoji od tri prozora, prozor za prijavu, prozor za pregled te prozor za analizu. U daljnjem tekstu bit će opisana struktura istih, njihovi elementi, te način na koji se koriste. Nakon ulaska u aplikaciju korisnik pomoću tabova može mijenjati prozore ovisno o potrebama u tom trenutku.

Prilikom pokretanja aplikacije, otvara se prozor za prijavu na kojem se od korisnika zahtijeva da upiše svoje korisničko ime i lozinku. Kako bi korisnik započeo rad s aplikacijom, od njega se zahtjeva unos ispravnog korisničkog imena te ispravne lozinke. Nakon unosa istih korisnik klikom na gumb „POTVRDI“ pristupa aplikaciji. Ukoliko je korisnik upisao netočne podatke, ili iz nekog drugog razloga u tom trenutku ne može pristupiti aplikaciji, može klikom na gumb „PONIŠTI“ izbrisati upisane podatke.

U podnožju zaslona nalazi se lista koja služi za prikaz mogućih grešaka prilikom korištenja aplikacije te prikaz grešaka ako se pojavljuju. Primjerice ako se korisnik uspješno prijavi na aplikaciju, pojavljuje se poruka „Uspješno ste spojeni s bazom!“, dok primjerice u slučaju da korisnik pokušava prijavu na aplikaciju s točnim podacima, te je ista neuspješna zbog problema na vezi sa bazom podataka pojavljuje se poruka „Pogreška u vezi s bazom!“, na taj način korisnik ima uvid o kakvoj se grešci radi te može prijaviti kvar administratoru.

[18] Oracle.com, Dostupno na: https://www.oracle.com/technetwork/java/getting-started-with-javafx-156435.html [Pristupljeno:5. kolovoza 2019.]



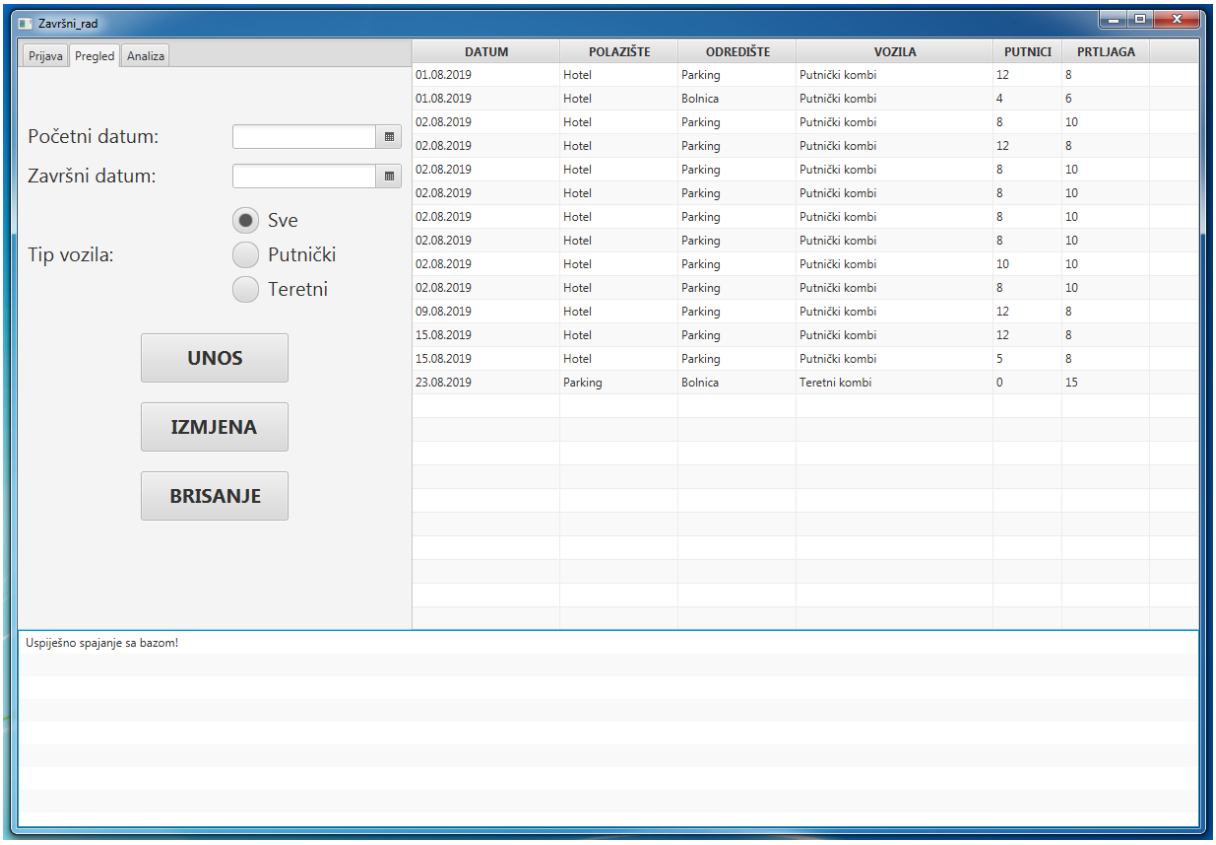
Slika 10. -Prikaz prozora „Prijava“

Izvor: Screenshot autora

Nakon prijave u aplikaciju, korisnik se automatski prebacuje na prozor „Pregled“. Na ovome prozoru korisniku se prikazuje tablica s postojećim podacima o unesenim terminima. Tablica se sastoji od šest stupaca koji su : datum, polazište, odredište, vozila, putnici, prtljaga. Stupac datum prikazuje kada se uneseni termin mora izvesti, ili je bio izveden. Stupac polazište prikazuje lokaciju s koje gosti/prtljaga kreću, dok stupac odredište prikazuje gdje gosti/prtljaga izlaze. Stupac putnici prikazuje koliko gostiju koristi uslugu u jednom terminu, dok stupac prtljaga prikazuje koliko prtljage imaju sa sobom.

Korisnik može promatrati sve unesene podatke, podatke u odabranom vremenskom periodu. Također omogućen je prikaz podataka samo za putnička, odnosno samo za teretna vozila.

Korisniku je omogućeno filtriranje u određenom vremenskom periodu korištenjem dva polja za odabir datuma (eng. date picker) pomoću kojih korisnik odabire početni datum te krajnji datum. Također korisnik može pregledati podatke po tipu vozila koristeći tri radio gumba pomoću kojih sortira pregled za sva vozila, samo putnička vozila ili samo teretna vozila. Ovisno o odabranim parametrima mijenja se izgled tablice.



Slika 11. -Prikaz prozora „Pregled“

Izvor: Screenshot autora

Klikom na gumb „UNOS“ korisniku se prikazuje prozor s unosom, koji na vrhu nudi dva radio gumba kojima korisnik određuje tip vozila da li će se unijeti transakcija za putničko vozilo ili teretno vozilo.

Nakon odabira tipa vozila korisnik bira količinu vozila. Količina vozila bira se iz opadajućeg izbornika, ponuđena je početna vrijednost jedan pošto je to minimalan broj vozila potrebnih za izvršavanje termina. Ovdje je korisniku ponuđen unos

podataka pomoću opadajućeg izbornika pošto se isti puni iz baze te prikazuje trenutačnu količinu dostupnih vozila. Primjerice ako je vozilo u kvaru ili iz nekog razloga nedostupno korisniku, neće biti prikazano u bazi podataka, te korisnik neće moći odabrati isto. Također prilikom posebnih manifestacija ili događaja, korisnik može imati na raspolaganju više vozila za poslovne potrebe, koja će nakon unosa u bazu biti dostupna na opadajućem izborniku.

Sljedeći podatak koji se zahtjeva od korisnika je datum izvođenja termina koji korisnik odabire korištenjem polja za unos datuma. Korisniku je dozvoljeno upisivanje već odrađenih termina, ako iz određenih razloga nije bio u mogućnosti upisati iste prije njihove izvedbe. Primjerice hitna vožnja do bolnice ili hitne pomoći. Također korisnik upisuje buduće termine nakon što prikupi sve potrebne podatke za unošenje istih, te iz tih razloga nema ograničenja pri unosu datuma.

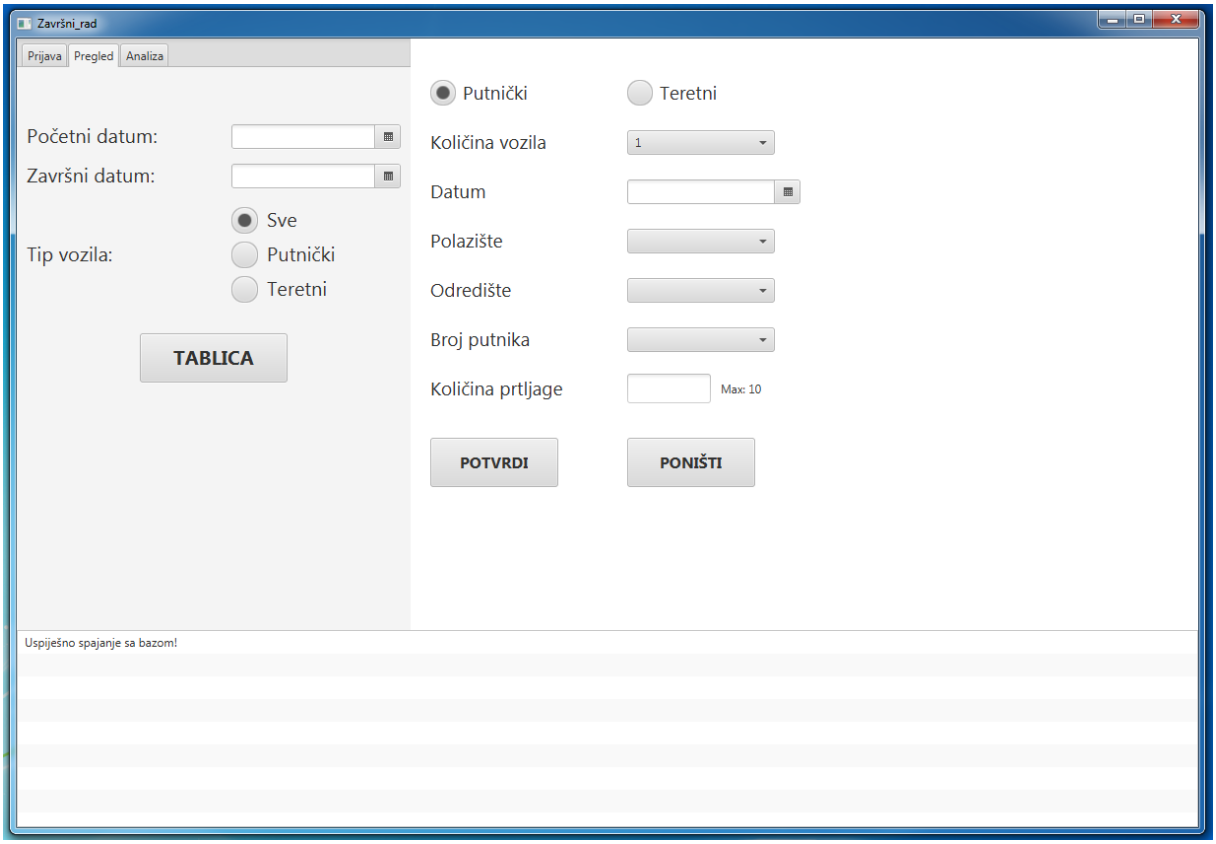
Nakon čega slijede dva opadajuća izbornika u kojima se nalaze već definirana polazišta i odredišta. Trenutačno kao ponuđene lokacije korisnik ima parkiralište, hotel, bolnicu i autobusni kolodvor. Pošto se informacije o polazištu i odredištu izvlače direktno iz baze, unosom nove destinacije na bazu, ista postaje dostupna korisniku za korištenje. Korisniku nije dozvoljen odabir iste lokacije pod polazištem i odredištem, to jest korisnik neće moći potvrditi unos ako navedene nisu različite.

Nakon toga korisnik unosi broj putnika pomoću opadajućeg izbornika. Prilikom odabira jednog vozila korisniku su dostupne vrijednosti između jednog i osam putnika. Također ako je korisnik odabrao više vozila, dostupan mu je unos onoliko putnika koliko ograničenja dopuštaju. Ako je odabrano teretno vozilo korisnik nema mogućnost unosa broja putnika, pošto se isto koristi isključivo za teret.

Nakon unosa broja putnika korisnik unosi količinu prtljage koristeći tekstualno polje, svako putničko vozilo može primiti maksimalno deset komada prtljage. Dok količina kofera može biti nula u slučaju hitnog prijevoza do hitne pomoći ili bolnice, prijevoza gosta do automobila za vrijeme boravka i slično. Ako je odabrano teretno vozilo, korisnik mora unijeti minimalno deset komada prtljage, dok je maksimum pedeset komada prtljage.

Kako bi se termin upisao na bazu potrebno je popuniti sve navedene podatke te kliknuti na gumb „POTVRDI“. Ako sva polja nisu ispunjena, ili je odabrana ista

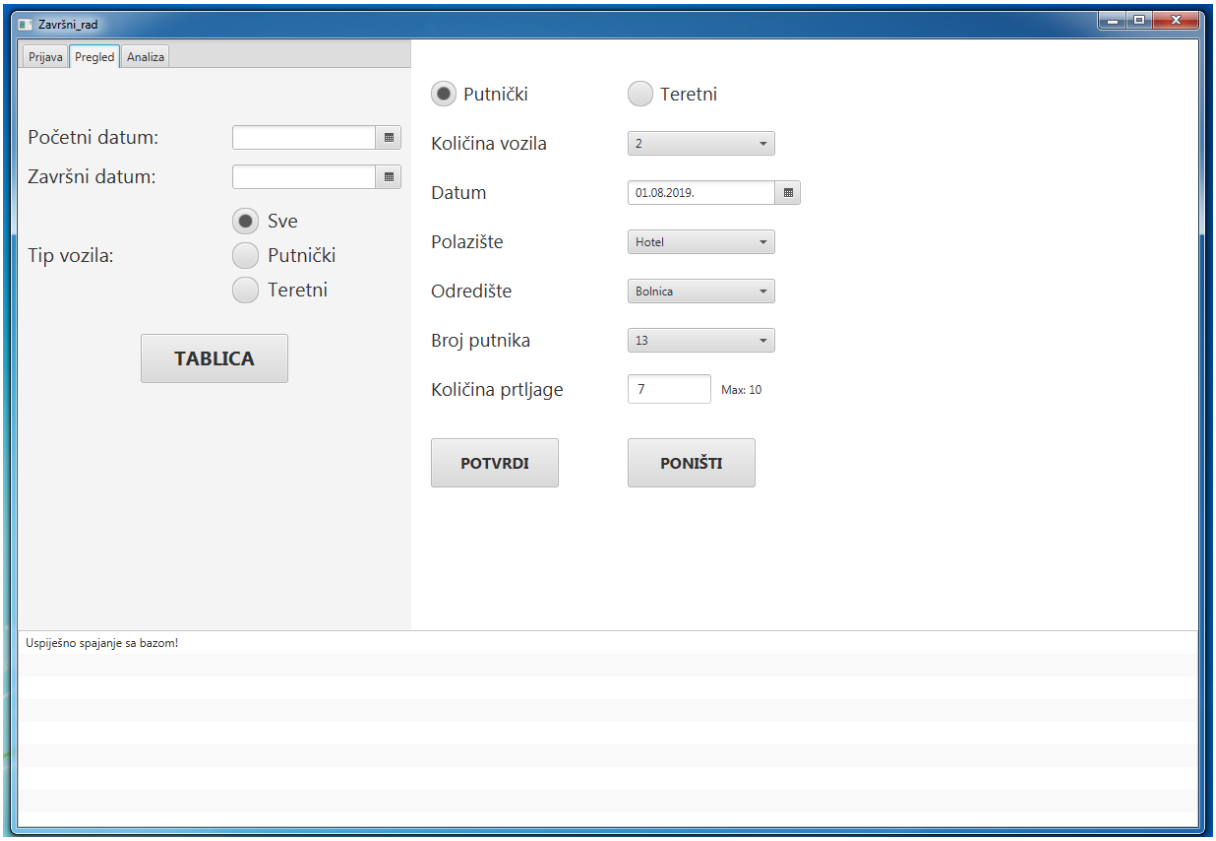
lokacija za polazište i odredište, aplikacija neće dozvoliti korisniku da unese novi termin na bazu. Ako iz bilo kojeg razloga korisnik prilikom unosa odustane od upisa termina, ima na raspolaganju gumb „PONIŠTI“ koji automatski prazni sve elemente. Nakon čega korisnik može započeti s unosom nove transakcije ili klikom na gumb „TABLICA“ vratiti se na pregled podataka u tablici.



Slika 12. -Prikaz zaslona za unos

Izvor:Screenshot autora

Kako bi korisnik unio izmjene u postojećem terminu mora odabrati redak u kojem se transakcija nalazi te kliknuti na gumb „IZMJENA“ nakon čega se otvara zaslon za izmjenu. Zaslon je identičan zaslonu za unos podataka te sadrži ista ograničenja, uz razliku da su već postavljeni svi početni podaci, koji su uneseni pri kreiranju odabranog termina. Korisnik ima mogućnost izmjene svih podataka ili samo jednog podatka, nakon što završi s izmjenama klikom na gumb „POTVRDI“ unosi izmjene u terminu, te ima na raspolaganju gumb „PONIŠTI“ u slučaju da se otkažu izmjene.

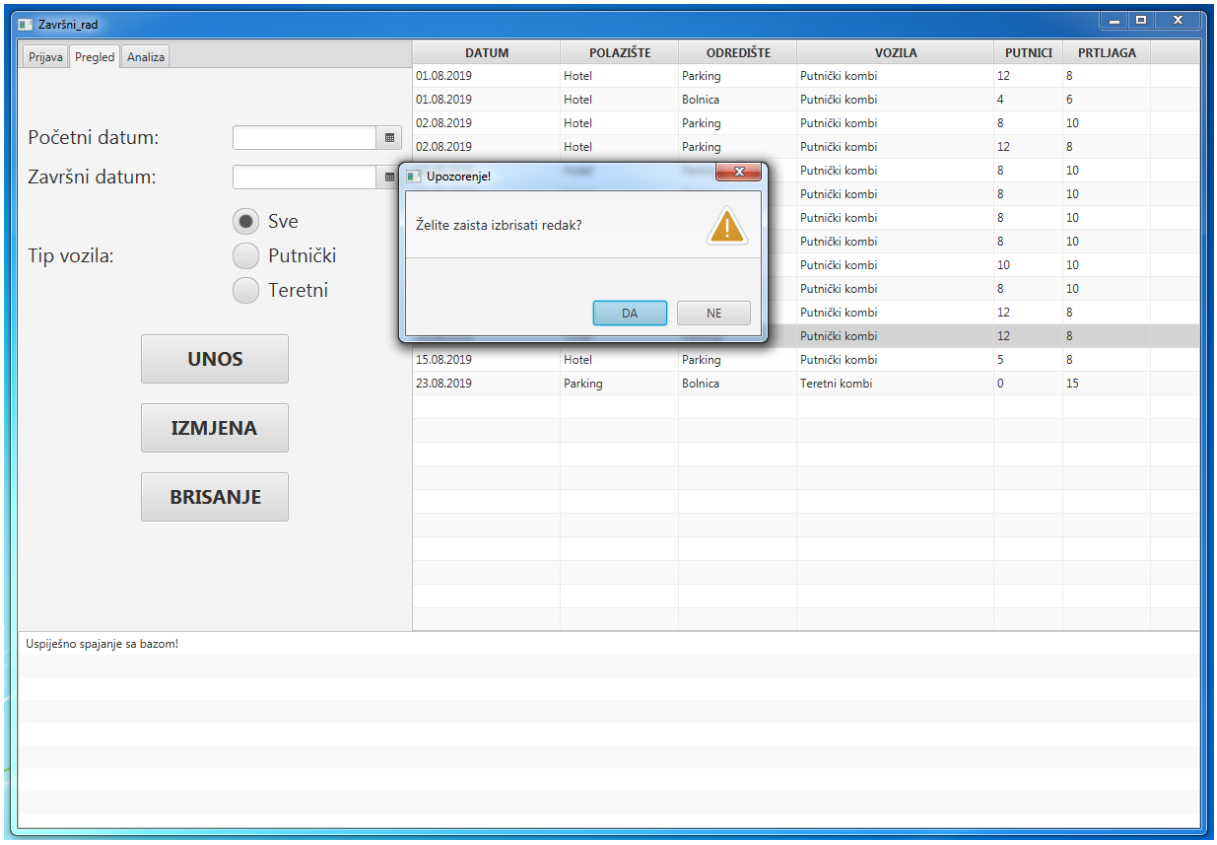


Slika 13. -Prikaz zaslona za izmjenu

Izvor: Screenshot autora

Također korisnik ima i opciju brisanja „Termina“ iz tablice ako zbog određenog razloga nije odrađen ili dođe do otkazivanja prije izvedbe istoga. Kako bi izbrisao određeni termin korisnik mora najprije u tablici odabrati željeni termin, te nakon toka kliknuti na gumb „IZBRIŠI“.

Kako ne bi došlo do greške prilikom brisanja, pojavljuje se skočni prozor gdje korisnik mora potvrditi brisanje, te mu se također daje mogućnost odustajanja od istoga. Ako korisnik klikne gumb „BRISANJE“ bez da je prethodno odabrao termin pojavljuje se skočni prozor koji obavještava korisnika da nema odabranog termina.



Slika 14. -Prikaz opcije brisanje

Izvor: Screenshot autora

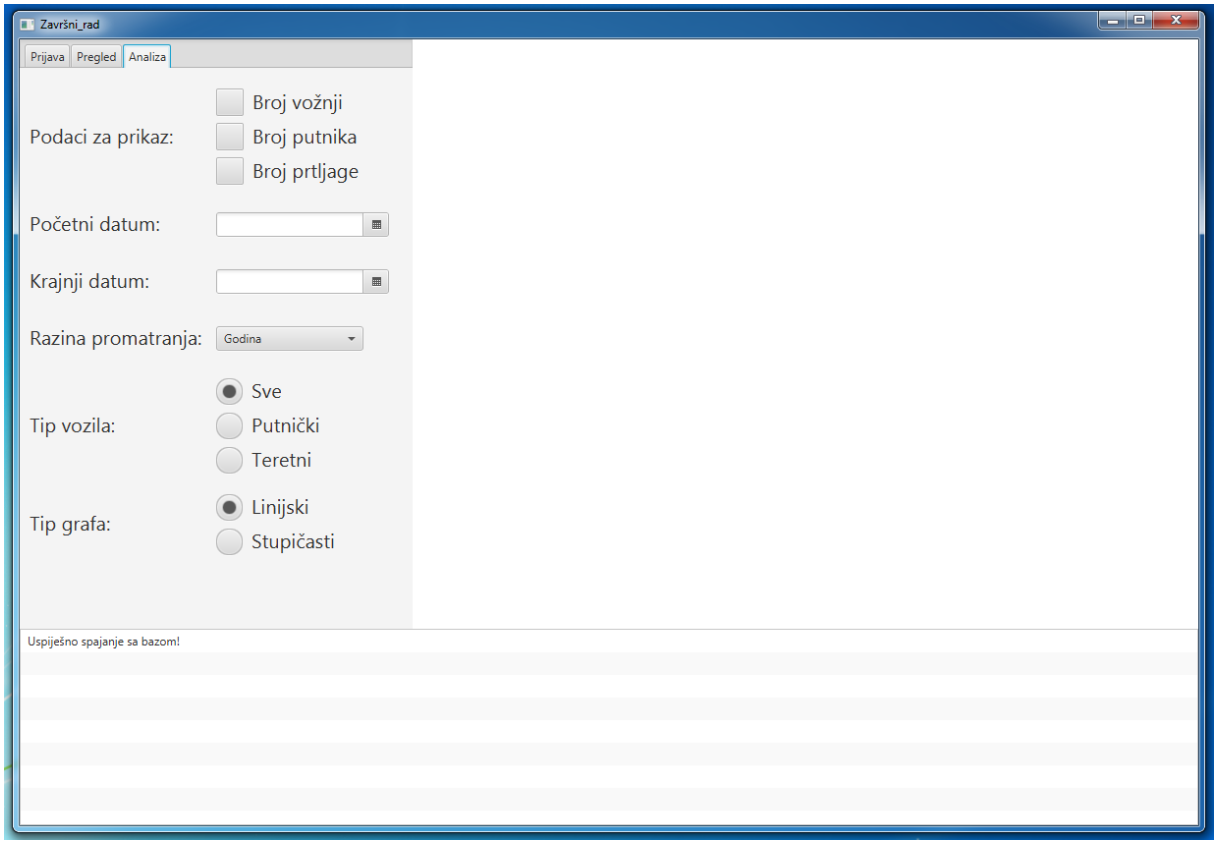
Korisniku je također dostupan prozor za analizu, kojemu pristupa odabirom tab-a „Analiza“.

Korisniku su na raspolaganju tri vrste podataka koje može prikazati. Odabirom okvira za izbor (eng. check box) korisnik može prikazati podatke o broju vožnji, odnosno broju termina, broju putnika te broju prtljage. Podaci mogu biti prikazani samostalno, te se mogu prikazati svi te u različitim kombinacijama radi potrebne usporedbe.

Korisnik također bira u kojem vremenskom rasponu će podaci biti promatrani, što čini unosom početnog te krajnjeg datuma u polja za odabir datuma.

Također korisnik može prikazati podatke po tipu vozila, odnosno može birati prikaz podataka za sva vozila, samo putnička vozila ili samo teretna vozila.

Nakon čega korisnik bira na koji način će podaci biti grafički prikazani. Korisnik podatke može prikazati stupčastim ili linijskim grafom ovisno o potrebama.



Slika 15. -Prikaz zaslona za analizu

Izvor: Screenshot autora

6.ANALIZA PODATAKA

Sedamdesetih godina prošlog stoljeća znanstvenici su počeli prepoznavati važnu ulogu kompjuterski podržanog informacijskog sustava, koji podržava menadžere u njihovom donošenju odluka u rješavanju strukturiranih, polustrukturiranih i nestrukturiranih problema. Od tada sustavi za potporu odlučivanju postaju bitan dio računalno podržanog informacijskog sustava. Razvijaju se s ciljem smanjivanja sveprisutne nesigurnosti kod donošenja poslovnih odluka.[19]

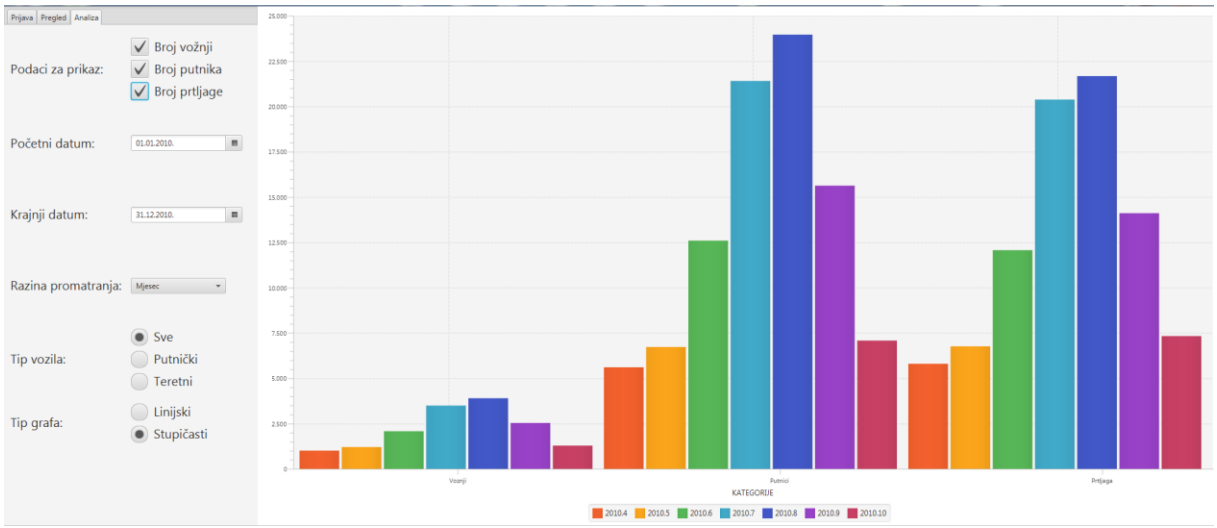
Postoji jako puno definicija sustava za potporu odlučivanju. Jedna od definicija je da je to specifičan kompjuteriziran informacijski sustav koji podupire poslovanje i poslovno odlučivanje u poduzeću. Kvalitetno izrađen sustav za potporu odlučivanju je interaktivan kompjuterski sustav koji ima za cilj pomoći donositeljima odluke da prikupe informacije iz neobrađenih podataka, dokumenata i poslovnih modela da bi identificirali i riješili probleme te donijeli odluke. Sustavi za potporu odlučivanja su dio cjelokupnog informacijskog sustava poduzeća.[20]

U praktičnom djelu ovog rada izrađena je aplikacija koja ima dva cilja. Pogled „Pregled“ namijenjen je za svakodnevno poslovanje, odnosno namijenjen je voditelju odjela kao pomoć pri razmještanju dostupnih vozila po lokacijama, ovisno o intenzitetu prijevoza, pravljenje dnevnog rasporeda, to jest pri odlučivanju koliko je vozača te djelatnika parkirališta potrebno, te na kojim lokacijama. Također može predvidjeti hoće li biti potrebe za teretnim vozilima te u kojoj količini.

Dok je pogled „Analiza“ namijenjen menadžeru te služi kao pomoć pri donošenju dugoročnih odluka. Za tu namjenu odabrani su podaci o broju vožnji, broju putnika te broju prtljage. Ti podaci mogu biti filtrirani odabirom putničkog vozila, teretnog vozila ili odabirom oba vozila. Podaci se promatraju u određenom vremenskom periodu te na razini godine, mjeseca ili dana.

Za grafički prikaz podataka odabrani su linijski te stupčasti grafikoni koji će biti pojašnjeni u nastavku rada pomoću primjera.

[19] Skladištenje.com, Dostupno na: http://www.skladistenje.com/poslovno-odlucivanje-i-sustavi-za-potporu-odlucivanju/ [Pristupljeno: 18. kolovoza 2019.]
[20] loc. cit

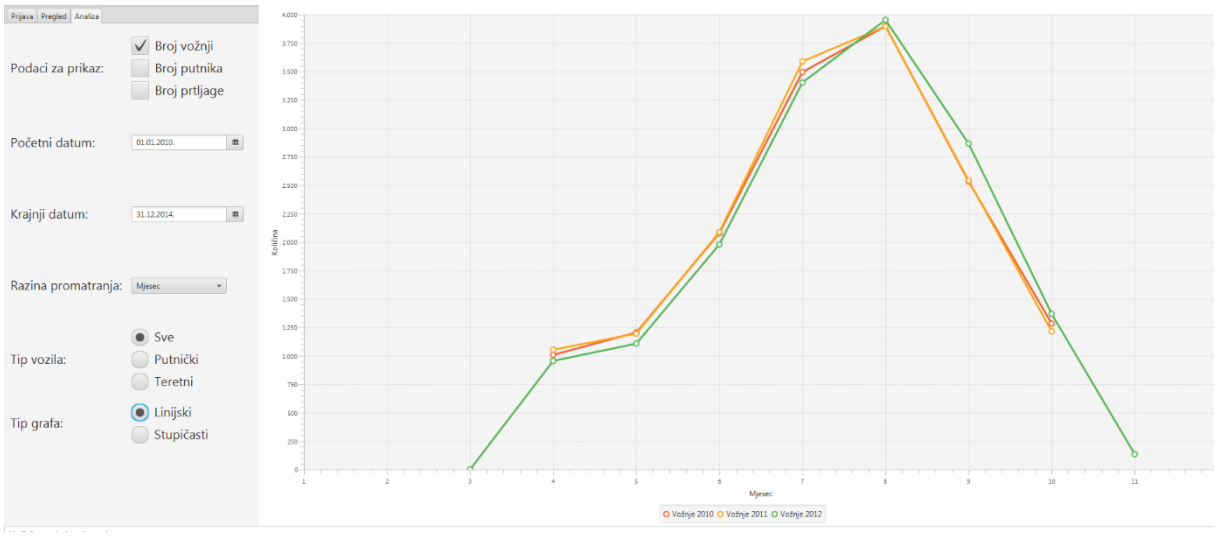


Slika 16. -Podatci o broju vožnji,putnika i prtljage

po mjesecima u godini

Izvor: Creenshot autora

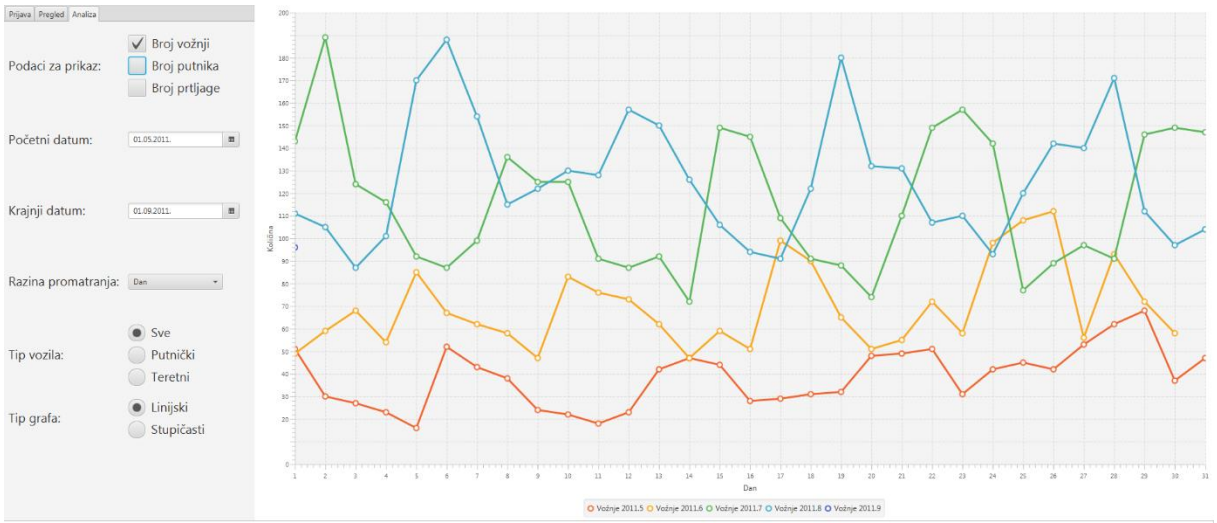
Na prethodno prikazanom grafikonu vidimo omjer između broja vožnji po pojedinom mjesecu, koliko je putnika i prtljage prevezeno u tom periodu. Pomoću ovih podataka mogu se donijeti odluke o potrebnoj količini vozila po mjesecu, odnosno treba li se povećati ili smanjiti vozni park. Također mogu li se vozila u pojedinim periodima koristiti i za ponudu dodatnih usluga prijevoza. Usporedbom ovih podataka s podacima o dolascima i odlascima gostiju u hotel može se vidjeti koliko korisnici često koriste uslugu prijevoza za ostale svrhe, te se pomoću toga formirati cijena prijevoza.



Slika 17. -Podatci o broju vožnji po mjesecima kroz godine

Izvor: Screenshot autora

Pomoću ovog grafikona možemo vidjeti dali postoje promjene u intenzitetu vožnji po pojedinim mjesecima kroz nekoliko godina. Pregledom istih te uz promatranje dodatnih podataka iz ostalih izvora može se utvrditi zbog čega je došlo do promjena intenziteta, ili je konzistentan. Pomoću ovih podataka korisnik može donijeti odluke vezane uz pružanje usluge.



Slika 18. -Podatci o broju vožnji po danima u mjesecu kroz godinu

Izvor: Screenshot autora

Pomoću prethodno prikazanih podataka dobiva se uvid o broju vožnji na dnevnoj bazi po pojedinom mjesecu. Pomoću njih možemo donositi zaključke o potrebnom broju radnika i vozila po pojedinom danu. Također možemo vidjeti u kojim dijelovima tjedna je intenzitet pojačan te sukladno tome prilagoditi način poslovanja.

ZAKLJUČAK

U današnje vrijeme teško je biti konkurentan u poslovanju, pogotovo u sektoru turizma gdje traje borba za svakoga gosta. Ponuda dodatnih usluga i visoka kvaliteta su uvjet za dobro poslovanje, što je teško postići ako se pravovremeno ne donose točne i kvalitetne odluke. Kako bi menadžeri donosili iste potrebni su im pravovremeni i točni podatci koji bi poduprli njihove odluke.

Prikupljanje i prikaz istih gotovo je nemoguće u današnje vrijeme bez potpore aplikativnog rješenja ili više njih, ovisno o strukturi poslovanja. Pomoću aplikacije izrađene u praktičnom djelu ovog rada, korisnik ima na raspolaganju alat za prikupljanje i rad s podacima koji su potrebni za uspješno izvođenje poslovnih zadataka vezanih uz sustav prijevoza.

Dostupan je grafički prikaz prikupljenih podataka pomoću kojih korisnik može donositi kvalitetne odluke kako na dnevnoj bazi tako i pri donošenju istih vezano uz dugoročno planiranje poslovanja.

Jako je važno detaljno specificirati zahtjeve prije početka izrade same aplikacije te donijeti odluke o alatima i tehnologijama koje će biti korištene prilikom izrade iste, kako ne bi došlo do nepotrebnih problema u fazi razvoja.

Također treba dobro definirati koje podatke će se prikupljati te kako će se isti prikazivati, kako bi aplikacija mogla izvršiti zadaće za koje je namijenjena.

Može se zaključiti da je u današnje vrijeme više nego potrebno koristiti aplikativna rješenja u poslovanju, što zbog velike količine podataka koje treba obraditi, što zbog raspoloživog vremena za izvedbu istoga.

POPIS LITERATURE

KNJIGE:

1) I.Sommerville (2011.) *Software engineering*, 9th edition, Pearson 2010.

INTERNET:

1) TechnoSIP.com, Dostupno na: https://www.technosip.com/toolsreviews/technosip-review-xampp/ [Pristupljeno:1. kolovoza 2019.]

2) MariaDB.org, Dostupno na: https://mariadb.org/about/ [Pristupljeno:2. kolovoza 2019.]

3) PhpMyAdmin.net, Dostupno na: https://www.phpmyadmin.net/ [Pristupljeno: 4. kolovoza 2019.]

4) NetBeans.org, Dostupno na: https://netbeans.org/about/ [Pristupljeno: 2. kolovoza.2019.]

5) NetBeans.org, Dostupno na: https://netbeans.org/kb/73/java/editor-codereference.html#hints [Pristupljeno: 2. kolovoza. 2019.]

6) Oracle.com, Dostupno na: https://www.oracle.com/technetwork/java/javase/downloads/javafxscenebuilder-info-2157684.html [Pristupljeno: 5.kolovoza 2019.]

7) Oracle.com, Dostupno na: https://www.oracle.com/technetwork/java/getting-started-with-javafx-156435.html [Pristupljeno:5. kolovoza 2019.]

8) Skladištenje.com, Dostupno na: http://www.skladistenje.com/poslovno-odlucivanje-i-sustavi-za-potporu-odlucivanju/ [Pristupljeno: 18. kolovoza 2019.]

POPIS SLIKA

Slika 1. -Korisničko sučelje XAMPP 9

Slika 2. -Prikaz tablice „termin“ 10

Slika 3. -Prikaz tablice „vozila“ 11

Slika 4. -Prikaz tablice „polaziste_odrediste“ 11

Slika 5. -Prikaz baze podataka 12

Slika 6. -Prikaz upozorenja o unosu 13

Slika 7. -Prikaz sugestije za ispravak greške 13

Slika 8. -Novi projekt u NetBeans-u 14

Slika 9. -Prmjer FXML datoteke u Scene Builderu 16

Slika 10. -Prikaz prozora „Prijava“ 18

Slika 11. -Prikaz prozora „Pregled“ 19

Slika 12. -Prikaz zaslona za unos 21

Slika 13. -Prikaz zaslona za izmjenu 22

Slika 14. -Prikaz opcije brisanje 23

Slika 15. -Prikaz zaslona za analizu 24

Slika 16. -Podatci o broju vožnji,putnika i prtljage 26

Slika 17. -Podatci o broju vožnji po

mjesecima kroz godine 27

Slika 18. -Podatci o broju vožnji po

danima u mjesecu kroz godinu 28

SAŽETAK

Tema završnog rada je izrada aplikacije koja će podržavati rad parkirališta sa sustavom prijevoza. Aplikacija je zamišljena na način da pruža potporu zaposlenicima u obavljanju svakodnevnih zadataka, te ima mogućnost prikaza prikupljenih podataka pomoću grafikona, koji pomažu menadžerima pri donošenju dugoročnih odluka. Za potrebe iste definirani su korisnički i sistemski zahtjevi, te odabrani programski alati i tehnologije koji su korišteni prilikom izrade. Izrađena je desktop aplikacija pomoću alata XAMPP Control Panel, NetBeans IDE, JavaFX Scene Builder dok je kod pisan u JavaFX skriptnom jeziku. Aplikacija koristi Apache server, MariaDB te je baza izrađena pomoću phpMyAdmin softverskog alata. Aplikacija se sastoji od tri pogleda. Prvi pogled je namijenjen za prijavu korisnika u aplikaciju. Drugi pogled je namijenjen za pregled podataka, te omogućuje korisniku unos, izmjenu i brisanje podataka. Dok treći pogled omogućuje korisniku izradu grafikona po odabranim podacima i parametrima.

Ključne riječi: XAMPP Contorl Panel, NetBeans IDE, JavaFX Scene Builder, JavaFX, desktop aplikacija, potpora odlučivanju.

SUMMARY

The teme of the bachelor thesis is to creat an application which will support operation of the parking with the transportation system. The application is designed in a way to support employees in their daily tasks, and has the ability to display the collected data by using charts, that help managers make long-term decisions. For the needs of crating the application user and system requirements are defined, and selected software tools and technologies that were used in the process. A desktop application is created usin tools XAMPP Control Panel, NetBeans IDE, JavaFX Scene Builder while the code is written in JavaFX scripting language. The application uses an Apache server, MariaDB and the database was created usin the phpMyAdmin software tool. The application consists of three views. The first view is intended to log users into the application. The second view is for viewing data, and allows the user to enter, edit and delete data. While the third view allows the user to create a chart according to the selected dana and parameters.

Keywords: XAMPP Control Panel, NetBeans IDE, JavaFX Scene Builder, JavaFX, desktop application, decision support.