

Uvod u objektno programiranje – Algoritmi i strukture podataka

Specifikacija projektnog zadatka

Projektni zadatak: Kroz jedan integrisan projekat, studenti polažu dva predmeta, što uključuje Uvod u objektno programiranje i Algoritme i strukture podataka. Upotrebom tehnologija i tehnika primenjivanih na predavanjima i vežbama prethodno pomenuta dva predmeta isprojektovati programsku podršku za informacioni sistem **taksi službe** prema datoj specifikaciji.

Opšte napomene: Projektni zadatak treba da bude realizovan u Java programskom jeziku i treba da podrži osnovne funkcionalnosti zamišljene taksi službe. U daljem tekstu je data specifikacija poslovanja taksi službe pri čemu je na studentima da osmisle i implementiraju arhitekturu projekta u skladu sa pokazanim principima objektno-orijentisanog programiranja. Projekat se radi u timu od dva člana (može samostalno) i od svakog tima se očekuje da u konsultaciji sa predmetnim nastavnicima, razvije **jedinstvenu** implementaciju datog zadatka.

Dozvoljeno je korišćenje dodatnih biblioteka i tehnologija koje nisu sastavni deo kursa uz prethodnu konsultaciju sa nastavnicima.

Rok za izradu projekta: 25.6.2021.

Specifikacija zahteva

1. Model podataka

- Taksi služba sadrži sledeće podatke: PIB, naziv, adresa, cena starta vožnje i cena po kilometru.
- U sistemu taksi službe vodi se evidencija o tri tipa korisnika: Mušterije, Vozači i Dispečeri. Svim korisnicima se evidentiraju sledeći podaci: korisničko ime, lozinka, ime, prezime, JMBG, adresa, pol, broj telefona.
 - Dispečeri su zaposleni koji rade u telefonskoj službi i o njima je potrebno, pored podataka zajedničkih za sve korisnike, evidentirati platu, broj telefonske linije i odeljenje na kojem rade. Od telefonskih odeljenja, služba poseduje Odeljenje za prijem vožnji i Odeljenje za reklamacije.
 - Za vozače se dodatno evidentiraju sledeći podaci: plata, broj članske karte udruženja taksista i automobil(vozač u jednom trenutku raspolaže sa jednim vozilom).
- Automobili predstavljaju poseban entitet u sistemu i za svaki se evidentira sledeće: model, proizvođač, godina proizvodnje, broj registarske oznake, broj taksi vozila (jedinствена oznaka koju svako vozilo ima u okviru svoje taksi službe), vrsta automobila (taksi služba raspolaže sa putničkim automobilima i kombi vozilima).
- Osnovni entitet u sklopu poslovanja taksi službe je vožnja. Za svaku vožnju se evidentiraju sledeći podaci: Datum i vreme porudžbine, adresa polaska, adresa destinacije, mušterija koja je naručila vožnju, vozač koji je preuzeo porudžbinu, broj km pređenih u vožnji, trajanje vožnje i status vožnje. Status vožnje može biti KREIRANA, KREIRANA-NA ČEKANJU, DODELJENA, PRIHVAĆENA, ZAVRŠENA, ODBIJENA. Vožnje je moguće poručiti na dva načina:
 - o Telefonskim pozivom: Kada se kreira ova vožnja njen inicijalni status je KREIRANA. Nakon što dispečer vožnju dodeli određenom vozaču njen status postaje DODELJENA.
 - o Putem aplikacije: Za sve porudžbine preko aplikacije potrebno je evidentirati tekstualnu napomenu koju unose korisnici. Kada se kreira ova vožnja njen inicijalni status je KREIRANA-NA ČEKANJU.

2. Funkcionalnosti

2.1. Prijava na sistem

2.1.1. Prijava na sistem – neprijavljeni korisnik mora da se prijavi na sistem tako što unosi korisničko ime i lozinku korisnika za koju je registrovan. Nakon toga, korisnik je prijavljen i može da izvršava aktivnosti predviđene njegovom ulogom. Neophodno je obezbediti prijavu na sistem za sve tipove korisnika pri čemu svako od njih ima pristup određenim funkcionalnostima sistema.

**napomena – svi korisnici se učitavaju iz tekstualnog fajla, nije potrebno implementirati registraciju ni za jedan tip korisnika

2.2. Funkcionalnosti dispečera

Dispečeri imaju pristup svim entitetima i svim akcijama nad njima (smatraju se administratorima sistema):

2.2.1. Prikaz i izmena podataka o taksi službi

2.2.2. Prikaz, dodavanje, izmena i brisanje svih vozača (prilikom dodavanja vozača vozaču je moguće dodeliti jedan od slobodnih automobila, ako postoji);

2.2.3. Prikaz, dodavanje, izmena i brisanje svih automobila (prilikom dodavanja automobila, automobil je moguće dodeliti nekom od vozača koji nemaju dodeljen automobil; brisanje je moguće samo za automobile koji nisu dodeljeni ni jednom vozaču);

2.2.4. Prikaz svih vožnji, i kreiranih putem telefona i kreiranih putem aplikacije;

2.2.5. Dodeljivanje vožnji kreiranih putem telefona vozačima (pri ovom koraku dispečer postavlja vozača i menja status vožnje sa KREIRANA-NA ČEKANJU na DODELJENA);

2.2.6. Kombinovana pretraga svih vozača po imenu, prezimenu, plati i automobilu;

2.2.7. Kombinovana pretraga svih automobila po modelu, proizvođaču, godini proizvodnje, broju registarske oznake i broju taksi vozila;

2.2.8. Prikaz sumiranog izveštaja na dnevnom, nedeljnom, mesečnom i godišnjem nivou. Izveštaj treba da sadrži podatke o ukupnom broju vožnji, broju vožnji poručenim putem aplikacije, broju vožnji poručenim putem telefonskog poziva, broju aktivnih vozača, prosečnom trajanju vožnje, prosečnom broju pređenih kilometara, ukupnoj zaradi za sve vožnje, i prosečnoj zaradi po vožnji. Zaradu vozača po vožnji računati po sledećoj formuli: $\text{start} + \text{brojPredjenihKilometara} * \text{cenaPoKilometru}$.

2.2.9. Prikaz izveštaja o vozačima na dnevnom, nedeljnom, mesečnom i godišnjem nivou. Izveštaj treba o svakom vozaču da sadrži ukupan broj vožnji, ukupan broj pređenih kilometara, ukupno trajanje vožnji, prosečan broj pređenih kilometara po vožnji, prosečno trajanje vožnje, ukupnoj zaradi, prosečnoj zaradi po vožnji i koliko vremena prosečno provede bez vožnje (radno vreme – ukupno trajanje vožnji za taj period). Izveštaj treba prikazati u vidu tabele koja sadrži sve vozače u redovima i sve podatke u zasebnim kolonama.

2.3. Funkcionalnosti vozača

2.3.1. Prikaz istorije sopstvenih vožnji.

2.3.2. Prikaz svih vožnji zakazanih putem aplikacije kao i mogućnost da se “prihvati” ili “odbije” određena vožnja. Prihvatanjem vožnje menja se status iz KREIRANA u PRIHVAĆENA, a odbijanjem vožnje menja se status iz KREIRANA u ODBIJENA.

2.3.3. Prikaz dodeljenih vožnji (vožnje kreirane putem telefonskog poziva, koje je dispečer dodelio vozaču), vozač ima mogućnost da prihvati ili odbije vožnju. Prihvatanjem vožnja menja status iz DODELJENA u PRIHVAĆENA, dok odbijanjem vožnja menja status iz DODELJENA u ODBIJENA.

2.3.4. Za sve prihvaćene vožnje neophodno je obezbediti da se vožnja može završiti, što podrazumeva promenu statusa vožnje iz PRIHVAĆENA u ZAVRŠENA, unos broja km pređenih u vožnji i trajanje vožnje.

2.3.5. Vozač treba da vidi sumiranu statistiku svojih vožnji po danu, nedelji, mesecu i godini. Izveštaj treba da sadrži ukupan broj vožnji, ukupan broj pređenih kilometara, ukupno trajanje vožnji, prosečan broj pređenih kilometara po vožnji, prosečno trajanje vožnje, koliko vremena prosečno provede bez vožnje (radno vreme – ukupno trajanje vožnji za taj period) i kolika je prosečna zarada. Takođe, treba omogućiti pregled ukupne zarade.

2.3.6. Omogućiti vozačima da učestvuju u aukciji za novu vožnju. Vozač unosi koliko minuta mu je potrebno da stigne do date adrese. Dizajnirati algoritam koji će dodeljivati vozačima vožnje. Na dodeljivanje vožnje može uticati potrebno vreme dolaska, ocena vozača, zarada vozača u nekom

intervalu, starost vozila (mušterija može da traži novija vozila), pet friendly itd. Kreirati simulaciju ponuda više vozača i dodelu vožnje. Potrebno je čuvati istoriju aukcija. Za veći broj bodova potrebno je proširiti podatke o vozilima ili vozačima, takođe, potrebno je omogućiti mušterijama da ocene vozača nakon vožnje (ocena od 1 do 5).

2.4. Funkcionalnosti mušterije

2.4.1. Naručivanje vožnje putem telefona

2.4.2. Naručivanje vožnje putem aplikacije

2.4.3. Prikaz istorije sopstvenih vožnji

3. Dodatne napomene

3.1.1. Neophodno je obezbediti odgovarajući način navigacije kroz aplikaciju

3.1.2. Svi entiteti, osim korisnika, treba da imaju identifikator koji će biti tipa int ili long.

3.1.3. Podatke treba čuvati sortirano po ključu.

3.1.4. Pretragu po identifikatoru potrebno je implementirati putem binarne pretrage.

3.1.5. Omogućiti sortiranje svih tabelarnih prikaza u aplikaciji.

3.1.6. Za ocene 7 pa na dalje potrebo je implementirati svoje strukture podataka, za liste i mape.

3.1.7. U zavisnosti od konkretne implementacije, studenti mogu proizvoljno organizovati date entitete ili dodavati druge. Podaci o svim entitetima se čuvaju u tekstualnim datotekama koje mogu biti u proizvoljnom formatu. Za zadate datoteke, potrebno je ručno implementirati metode za čitanje i pisanje podataka. Za podatke koji unapred imaju predefinisan skup mogućih vrednosti (pol korisnika, vrsta automobila, status vožnje, vrsta telefonskog odeljenja, ...) potrebno je implementirati odgovarajuće enumeracije.

3.1.8. Prilikom brisanja, svi entiteti se brišu logički što znači da podaci o njima treba da ostanu u datotekama ali obrisani entiteti (niti njihovi povezani entiteti) ne treba da se vide u sistemu, osim u situacijama kada to nema smisla. (Primer: brisanje vozača ne mora da podrazumeva brisanje automobila)

3.1.9. Svi podaci, osim onih koji su navedeni da mogu da ostanu nedefinisani, su obavezni tako da je, prilikom unosa ili izmene entiteta,

potrebno implementirati proveru koja će onemogućiti akciju ukoliko nisu uneseni svi obavezni podaci.

4. RASPODELA ZADATAKA ZA STUDENTE

4.1. Studenti koji rade u timu

4.1.1. Student 1:

- 2.1.1.
- 2.2.2. , 2.2.4. , 2.2.5. , 2.2.6. , 2.2.8.
- 2.3.3.
- 2.4.1.

4.1.2. Student 2:

- 2.2.1. , 2.2.3. , 2.2.7. , 2.2.9.
- 2.3.1. , 2.3.2. , 2.3.5.
- 2.4.2. , 2.4.3.

4.1.3. Oba:

- 2.3.4. , 2.3.6.
- 3.1.1. , 3.1.2. , 3.1.3. , 3.1.4. , 3.1.5. , 3.1.6. , 3.1.7. , 3.1.8. , 3.1.9.

****napomena** – završavanje vožnje (stavka 2.3.4.) podrazumeva da student 1 implementira ovu funkcionalnost za vožnje kreirane putem telefona, a student 2 implementira ovu funkcionalnost za vožnje kreirane putem aplikacije.

4.2. Studenti koji rade samostalno

Studenti koji se opredele za samostalnu izradu projektnog zadatka implementiraju samo funkcionalnosti navedene u delu Student 1 (sekcija 4.1.1.) i u delu Oba (sekcija 4.1.3.). Obratiti pažnju na napomene.

4.3. Stari studenti koji polažu samo UOP

Stari studenti koji polažu samo Uvod u objektno programiranje implementiraju sledeće funkcionalnosti:

- 2.1.1.
- 2.2.2. , 2.2.4. , 2.2.5.

- 2.3.3. , 2.3.4.
- 2.4.1.
- 3.1.1. , 3.1.2. , 3.1.7. , 3.1.8. , 3.1.9.

****napomena – završavanje vožnje (stavka 2.3.4.) podrazumeva da se implementira ova funkcionalnost za vožnje kreirane putem telefona**

5. BODOVANJE

<i>Skala bodovanja za UOP</i>	
Kreiranje svih klasa i konstruktora sa odgovarajućim atributima za sve potrebne entitete.	15
Kreiranje osnovnih odgovarajućih metoda za rad sa datim entitetima bez GUI-a.	15
Navigacija kroz aplikaciju (3.1.1.)	10
Prijava na sistem svih korisnika (2.1.1.)	10
Prikaz i izmena podataka o taxi službi (2.2.1.)	10
Prikaz, izmena, brisanje i dodavanje vozača (2.2.2.)	20
Prikaz, izmena, brisanje i dodavanje vozila (2.2.3.)	20
Prikaz svih vožnji - za dispečere (2.2.4.)	5
Dodeljivanje vožnji kreiranih putem telefona vozaču (2.2.5.)	5
Prikaz istorije sopstvenih vožnji – za vozače (2.3.1.)	5
Pregled vožnji na čekanju (vožnje kreirane putem aplikacije) uz mogućnost prihvatanja/odbijanja vožnje (2.3.2.)	10
Pregled dodeljenih vožnji (vožnje kreirane putem telefona) uz mogućnost prihvatanja/odbijanja vožnje (2.3.3.)	10
Završavanje vožnje (2.3.4.)	5
Naručivanje vožnji putem telefona (2.4.1.)	5
Naručivanje vožnji putem aplikacije (2.4.2.)	5
Prikaz istorije sopstvenih vožnji - za mušterije (2.4.3.)	5

<i>Skala bodovanja za AISP</i>	
Aukcija	20
Binarna pretraga svih entiteta po ključu (3.1.4)	5
Sortiranje svih tabelarnih prikaza u aplikaciji (3.1.5)	5
Sumirane statistike za vozača (2.3.5)	10
Sumirane statistike za dispečera (2.2.8)	10
Sumirane statistike po vozaču (2.2.9)	15
Kombinovane pretrage entiteta (2.2.6)	7.5
Kombinovane pretrage entiteta (2.2.7)	7.5
Kreiranje, izmena i pregled svih potrebnih entiteta (UOP)	20

6. PRIMER IZGLEDA POJEDINIH DIJALOGA

Napomena:Mockupi su informativne prirode i ne sadrže sve podatke navedene u specifikaciji! Takođe se ne očekuje da aplikacija bude 1-1 sa mockup-om! Mockupi su okvirni kako bi se stekao utisak koji nivo detalja se očekuje!

Prijava na sistem

Korisničko ime:

Lozinka:

Potvrdi

Slika 1. Prijava na sistem

Pregled vozača									
<div> <div>+</div> <div></div> <div>×</div> </div>									
Korisničko ime	Ime	Prezime	JMBG	Adresa	Pol	Br. telefona	Plata (RSD)	Br. članske karte	ID automobila
pera1	Pera	Peric	1111111111111	Bulevar Oslobođenja 3a	MUSKI	06400000000	30000	1212	1
mika1	Mika	Mikic	2222222222222	Lasla Gala 31	MUSKI	06300000000	44000	1213	2
zika123	Zika	Zikic	3333333333333	Jovana Cvijica 34	MUSKI	06100000000	37000	1214	3

Slika 2. Pregled vozača

OSS-Softverske i informacione tehnologije

Dodavanje vozača

Ime:

JMBG:

Prezime:

Broj telefona:

Korisničko ime:

Plata:

Lozinka:

Br. članske karte:

Adresa:

Automobil:

1

Pol:

MUSKI

Potvrdi

Slika 3. Dijalog za dodavanje vozača

Pregled automobila						
<div><div>+</div><div></div><div>×</div></div>						
ID	Proizvođač	Proizvođač	Godina proizvodnje	Br. registarske oznake	Br. taxi vozila	Vrsta automobila
1	Opel	Zafira	2015	NS 222 - MN	0301	PUTNICKI
2	Renault	Kadjar	2017	BG 111 - TU	0302	PUTNICKI
3	Peugeot	Boxer	2018	NS 333 - GB	0303	KOMBI

Slika 4. Pregled automobila



Dodaj automobil

Proizvođač:

Model:

Godina proizvodnje:

Registarska oznaka:

Vrsta:

Dodaj

Slika 5. Dijalog za dodavanje automobila



Prihvatanje vožnje:

Adresa polaska:

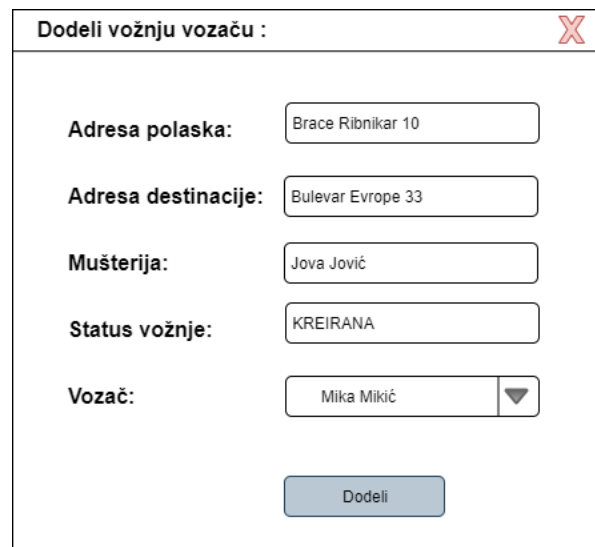
Adresa destinacije:

Mušterija:

Status vožnje:

Potvrdi

Slika 6. Dijalog za prihvatanje vožnje



Dodeli vožnju vozaču :

Adresa polaska:

Adresa destinacije:

Mušterija:

Status vožnje:

Vozač:

Dodeli

Slika 7. Dodeljivanje vožnje vozaču

**napomena: Sva polja osim polja gde se menja status vožnje su readonly

**napomena: Sva polja osim polja gde se menja status vožnje su readonly

Naruči vožnju preko aplikacije

X

Adresa polaska:

Adresa destinacije:

Napomena:

Naruči

Slika 8. Dijalog za naručivanje vožnje preko aplikacije

Naruči vožnju telefonom

X

Adresa polaska:

Adresa destinacije:

Naruči

Slika 9. Dijalog za naručivanje vožnje preko telefona