

# Programsko inženjerstvo

Ak. god. 2020./2021.

<Naziv projekta>

Dokumentacija, Rev. <1 ili 2>

Grupa: <Naziv grupe>

Voditelj: <Ime i prezime voditelja>

Datum predaje: <dan>. <mjesec>. <godina>.

Nastavnik: <Ime i prezime nastavnika zaduženog za vašu grupu>

# Sadržaj

# 1. Dnevnik promjena dokumentacije

## *Kontinuirano osvježavanje*

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Napravljen predložak.	*	22.08.2013.
0.2	Dopisane upute za povijest dokumentacije. Dodane reference.	*	24.08.2013.
0.5	Dodan <i>Use Case</i> dijagram i jedan sekvencijski dijagram, funkcionalni i nefunkcionalni zahtjevi i dodatak A	*	25.08.2013.
0.6	Arhitektura i dizajn sustava, algoritmi i strukture podataka	*	26.08.2013.
0.8	Povijest rada i trenutni status implementacije, Zaključci i plan daljnjeg rada	*	28.08.2013.
0.9	Opisi obrazaca uporabe	*	07.09.2013.
0.10	Preveden uvod	*	08.09.2013.
0.11	Sekvencijski dijagrami	*	09.09.2013.
0.12.1	Započeo dijagrame razreda	*	10.09.2013.
0.12.2	Nastavak dijagrama razreda	*	11.09.2013.
1.0	Verzija samo s bitnim dijelovima za 1. ciklus	*	11.09.2013.
1.1	Uređivanje teksta – funkcionalni i nefunkcionalni zahtjevi	* *	14.09.2013.

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

Rev.	Opis promjene/dodatka	Autori	Datum
1.2	Manje izmjene:Timer - Brojilo vremena	*	15.09.2013.
1.3	Popravljeni dijagrami obrazaca uporabe	*	15.09.2013.
1.5	Generalna revizija strukture dokumenta	*	19.09.2013.
1.5.1	Manja revizija (dijagram razmještaja)	*	20.09.2013.
2.0	Konačni tekst predloška dokumentacije	*	28.09.2013.

*Moraju postojati glavne revizije dokumenata 1.0 i 2.0 na kraju prvog i drugog ciklusa. Između tih revizija mogu postojati manje revizije već prema tome kako se dokument bude nadopunjavao. Očekuje se da nakon svake značajnije promjene (dodatka, izmjene, uklanjanja dijelova teksta i popratnih grafičkih sadržaja) dokumenta se to zabilježi kao revizija. Npr., revizije unutar prvog ciklusa će imati oznake 0.1, 0.2, ..., 0.9, 0.10, 0.11.. sve do konačne revizije prvog ciklusa 1.0. U drugom ciklusu se nastavlja s revizijama 1.1, 1.2, itd.*

## 2. Opis projektnog zadatka

### *dio 1. revizije*

*Cilj projekta je razviti programsku podršku za stvaranje aplikacije „ParkShare“ koja će korisniku omogućiti da rezervira i plati parking. Korisnik će moći također pregledati sva slobodna parkirališna mjesta za automobile i bicikle. Prilikom pokretanja sustava korisnik će na mapi vidjeti obližnja parkirališta i dobit će rutu do najbližeg te će moći vidjeti ima li slobodnih mjesta kako ne bi došao na popunjeno parkiralište. Neregistrirani korisnik šalje zahtjev za registraciju s odabranom ulogom koju želi, dakle vođitelj parkinga ili klijent. Neregistriranom korisniku pokazuju se dostupna parkirališta i dostupna mjesta, a klijentima se prikazuje i zauzetost mjesta u stvarnom vremenu. Za registraciju korisniku su potrebni sljedeći podaci:*

- *Korisničko ime*
- *Lozinka*
- *Ime*
- *Prezime*
- *Slika osobne*
- *IBAN račun*
- *Email adresa*

*Registracija završava potvrdom putem emaila korisnika, a vođitelje mora potvrditi dodatno i administrator. Svaki korisnik aplikacije može vidjeti svoje privatne podatke u aplikaciji i može ih mijenjati. Također korisnik može izbrisati svoj korisnički račun. Postoje 3 vrste korisnika:*

- *Klijent*
- *Administrator*
- *Voditelj parkirališta*

*Administrator može vidjeti popis svih klijenata i vođitelja i njihove osobne podatke i može ih mijenjati po volji ako je potrebno. Administrator može također pregledati sva parkirališta i njihova mjesta u aplikaciji i može mijenjati njihove podatke. Admi-*

nistrator može ukloniti ili promijeniti korisnički račun bilo kojeg ne-administrativnog korisnika. Voditelj zapisuje specifikacije svog parkirališta kao što su:

- Naziv parkirališta
- Opis
- Fotografija
- Cjenik

Voditelj ucrtava na karti svako parkirališno mjesto svog parkirališta i može za po želji odabrano parkirališno mjesto odrediti može li se rezervirati i postavlja senzore na parkirališna mjesta koji govore je li mjesto zauzeto. Nakon ucrtavanja parkirališnih mjesta zapisuje ukupan kapacitet parkirališta. Klijent na karti pregledava parkirališta u svojoj blizini i klikom na njih može vidjeti dostupne informacije o tom parkiralištu ujedno i je li ima slobodnih mjesta na parkiralištu. Ukoliko klijent želi biti siguran, on može rezervirati parkirališno mjesto tako da upiše termin u kojem želi parkirališno mjesto i pokazat će mu sva dostupna parkirališna mjesta na karti koja se mogu rezervirati. Drugi način za rezervaciju je da klijent unaprijed odabere parkirališna mjesta koja želi i onda će mu se prikazati kalendar s dostupnim terminima rezervacije. Klijent može rezervirati mjesto na proizvoljno dugo vremena i može rezervacija biti definirana kao ponavljajuća. Voditelj za svoje parkiralište određuje cijenu ovisno o trajanju rezervacije. Ukoliko se klijent odluči rezervirati parkirališno mjesto, parking mu se odmah naplaćuje putem aplikacije, ako pak klijent dođe bez rezervacije i odabere slobodno parkirališno mjesto, parking se naplaćuje na licu mjesta. U aplikaciji svaki klijent posjeduje novčanik kojim plaća parking, a u svakom trenu može uplatiti određenu svotu novca u novčanik kako bi mogao plaćati parking. Voditelji mogu vidjeti statistiku zauzetosti parkirališnih mjesta kroz vrijeme na svom parkiralištu. Prikupljaju se povijesne informacije kako je kroz vrijeme parkiralište bilo zauzeto i statistički se izračunava zauzetost u tom periodu vremena te se voditelju to prikazuje u obliku grafa. Voditelj može ucrtati najviše jedno parkiralište u aplikaciji, a parkirališna mjesta može ucrtavati i brisati kad god poželi, isto tako može izbrisati cijelo parkiralište. Voditelj parkinga može izmijeniti mogućnost rezervacije nekog parkirališnog mjesta te promijeniti cijenu parkinga. Napomenuli smo da će na svakom parkiralištu biti moguće ponuditi parking i biciklima. Voditelj parkirališta ne ucrtava parkirališna mjesta kao za automobile već samo određuje ukupan kapacitet parkirališta za bicikle. Korisnici bicikla neće morati brinuti o naplati parking jer se on neće naplaćivati takvim korisnicima samim tim neće imati ni opciju za rezervaciju parkinga. Svaki voditelj mora na karti naznačiti je li parkiralište dostupno i za bicikle kako bi korisnici znali na karti gdje mogu parkirati svoje bicikle. Parkirališna mjesta za bicikle neće

*imati senzor za očitavanje dostupnosti mjesta. U bazu podataka spremamo sve podatke o korisnicima i njihovim ovlastima. Pohranjujemo sve podatke o parkiralištima(ime, cijena, opis), također sve informacije o parkirališnim mjestima(zauzetost, tip). Svaki klijent u aplikaciji nakon korištenja određenog parkirališta dobiva „Like“ opciju, ikonu palčića gore i dolje te odabire klikom je li mu se parkiralište sviđalo ili ne. Također može i zanemariti danu opciju nije obvezno odgovoriti. Aplikacija mora podržavati sustav u kojem može biti više korisnika odjednom.*

## 3. Specifikacija programske potpore

### 3.1 Funkcionalni zahtjevi

#### *dio 1. revizije*

Navesti **dionike** koji imaju **interes u ovom sustavu** ili **su nositelji odgovornosti**. To su prije svega korisnici, ali i administratori sustava, naručitelji, razvojni tim.

Navesti **aktore** koji izravno **koriste** ili **komuniciraju sa sustavom**. Oni mogu imati inicijatorsku ulogu, tj. započinju određene procese u sustavu ili samo sudioničku ulogu, tj. obavljaju određeni posao. Za svakog aktora navesti funkcionalne zahtjeve koji se na njega odnose.

#### **Dionici:**

1. Asistent (Naručitelj)
2. Korisnici
3. Vlasnici parkinga
4. Administrator(i)
5. Razvojni tim

#### **Aktori i njihovi funkcionalni zahtjevi:**

1. Vlasnik parkinga (inicijator) može:
  - (a) Napraviti (max 1) parkiralište, te ga izbrisati, ili promijeniti ime i cjenik
  - (b) Ucrtati nova i izbrisati postojeća parking mjesta za svoj parking
  - (c) Definirati i promijeniti mogućnost rezervacije pojedinog parkirališnog mjesta
  - (d) Pregledati svoje osobne podatke i promijeniti ih
  - (e) Izbrisati svoj korisnički račun
  - (f) Pregledati statistiku zauzetosti svog parkirališta i parkirališnih mjesta kroz vrijeme u obliku grafa



2. Neregistrirani/neprijavljeni korisnik (inicijator) može:
  - (a) Registrirati se kao klijent ili kao vlasnik parkinga
  - (b) Prijaviti se
  - (c) Pregledati sva parkirališta i parkirališna mjesta dostupna u aplikaciji
3. Prijavljeni klijent (inicijator) može:
  - (a) Odjaviti se
  - (b) Pregledati svoje osobne podatke i promijeniti ih
  - (c) izbrisati korisnički račun
  - (d) Pregledati sva parkirališta i parkirališna mjesta dostupna u aplikaciji te njihovu zauzetost u realnom vremenu i rezervati ih
  - (e) Upisati destinaciju, tip prijevoza i procjenu trajanja parkinga te dobiti mogućnost rezervacije najbližeg parkirališnog mjesta
  - (f) Uplatiti novac u račun
  - (g) Pregledati povijest transakcija i rezervacija
4. Administrator (inicijator) može:
  - (a) Odjaviti se
  - (b) Izbrisati ili promijeniti korisnički račun bilo kojeg ne-administratorskog korisnika
  - (c) Pregledati sva parkirališta i parkirališna mjesta dostupna u aplikaciji, te promijeniti njihove podatke
  - (d) Potvrditi novi račun vlasnika parkirališta
5. Baza podataka (sudionik):
  - (a) Pohranjuje sve podatke o korisnicima i njihovim ovlastima
  - (b) Pohranjuje sve podatke o parkiralištima (ime, cjenik)
  - (c) Pohranjuje sve podatke o parkirališnim mjestima (zauzetost, tip)
6. Senzor zauzetosti (inicijator):
  - (a) Šalje signal kad se parkirališno mjesto zauzme ili oslobodi

### 3.1.1 Obrasci uporabe

#### Opis obrazaca uporabe

##### UC1 - Dodavanje parkirališta

- **Glavni sudionik:** Voditelj parkirališta
- **Cilj:** Dodavanje parkirališta u bazu podataka
- **Sudionici:** Voditelj parkirališta, baza podataka
- **Preduvjet:** Korisnik mora imati ulogu voditelja parkinga
- **Opis osnovnog tijeka:**
  1. Voditelj parkirališta odabire opciju za dodavanje parkinga
  2. Voditelj parkirališta unosi informacije o svom parkiralištu
  3. Voditelj parkirališta na karti unosi dostupna parkirališna mjesta
  4. Voditelj parkirališta postavlja senzor koji osvježava informaciju zauzetosti parkirališnog
- **Opis mogućih odstupanja:**
  - 2.a Unešeno već postojeće ime parkinga
    1. Aplikacija ispisuje grešku
  - 3.a Označavanje mjesta na karti na kojem već postoji parking
    1. Aplikacija ispisuje grešku
  - 4.a Aplikacija se ne može povezati sa senzorom
    1. Aplikacija ispisuje grešku

##### UC2 - Brisanje parkirališta

- **Glavni sudionik:** Voditelj parkirališta
- **Cilj:** Brisanje parkirališta
- **Sudionici:** Voditelj parkirališta, baza podataka
- **Preduvjet:** Korisnik mora imati ulogu voditelja parkinga, voditelj mora upravljati tim parkingom, parkiralište već postoji u bazi podataka
- **Opis osnovnog tijeka:**
  1. Voditelj zatraži brisanje parkirališta
  2. Parkiralište se briše iz baze podataka
  3. Voditelju se javlja da je parkiralište uspješno izbrisano
- **Opis mogućih odstupanja:**
  - 2.a Neuspješno brisanje podataka iz baze podataka
    1. Aplikacija ispisuje grešku

##### UC3 - Izmjena podataka o parkiralištu

- **Glavni sudionik:** Voditelj parkirališta
- **Cilj:** Izmjena podataka o parkiralištu
- **Sudionici:** Voditelj parkirališta, baza podataka
- **Preduvjet:** Korisnik mora imati ulogu voditelja parkinga, voditelj mora upravljati tim parkingom, parkiralište već postoji u bazi podataka
- **Opis osnovnog tijeka:**
  1. Voditelj zatraži izmjenu podataka
  2. Voditelj unosi nove izmjenjene podatke o parkiralištu
  3. Novi podatci se spremaju u bazu podataka
  4. Voditelju se javlja da su podatci uspješno izmjenjeni
- **Opis mogućih odstupanja:**
  - 3.a Neuspješno spremanje podataka u bazu podataka
    1. Aplikacija ispisuje grešku

#### UC4 - Izmjena podataka o parkirališnom mjestu

- **Glavni sudionik:** Voditelj parkirališta
- **Cilj:** Izmjena podataka o parkirališnom mjestu
- **Sudionici:** Voditelj parkirališta, baza podataka
- **Preduvjet:** Korisnik mora imati ulogu voditelja parkinga, voditelj mora upravljati tim parkingom, parkiralište i parkirno mjesto već postoji u bazi podataka
- **Opis osnovnog tijeka:**
  1. Voditelj zatraži izmjenu podataka
  2. Voditelj unosi nove izmjenjene podatke o parkirališnom mjestu
  3. Novi podatci se spremaju u bazu podataka
  4. Voditelju se javlja da su podatci uspješno izmjenjeni
- **Opis mogućih odstupanja:**
  - 3.a Neuspješno spremanje podataka u bazu podataka
    1. Aplikacija ispisuje grešku

#### UC5 - Dodavanje parkirališnog mjesta

- **Glavni sudionik:** Voditelj parkirališta
- **Cilj:** Dodavanje parkirališnog mjesta
- **Sudionici:** Voditelj parkirališta, baza podataka
- **Preduvjet:** Korisnik mora imati ulogu voditelja parkinga, voditelj mora upravljati tim parkingom, parkiralište i parkirno mjesto već postoji u bazi podataka
- **Opis osnovnog tijeka:**

1. Voditelj zatraži dodavanje novog parkirališnog mjesta na parkiralištu
  2. Voditelj na karti ucrtaje parkirališno mjesto
  3. Voditelj parkirališta definira je li moguće rezervirati parkirališno mjesto
  4. Podatci se spremaju u bazu podataka
  5. Voditelju se javlja da su podatci uspješno spremljeni u bazi podataka
- **Opis mogućih odstupanja:**
- 2.a Voditelj dodaje parkirališno mjesto na lokaciji na kojoj već postoji parkirališno mjesto
    1. Voditelja se upozorava o grešci
  - 4.a Neuspješno spremanje podataka u bazu podataka
    1. Aplikacija ispisuje grešku

#### **UC6 - Brisanje parking mjesta**

- **Glavni sudionik:** Voditelj parkirališta
- **Cilj:** Brisanje parking mjesta
- **Sudionici:** Voditelj parkirališta, baza podataka
- **Preduvjet:** Korisnik mora imati ulogu voditelja parkinga, voditelj mora upravljati tim parkingom, parkiralište već postoji u bazi podataka
- **Opis osnovnog tijeka:**
  1. Voditelj zatraži brisanje parkirališnog mjesta
  2. Parkirališno mjesto se briše iz baze podataka
  3. Voditelju se javlja da je parkiralište uspješno izbrisano
- **Opis mogućih odstupanja:**
  - 2.a Neuspješno brisanje podataka iz baze podataka
    1. Aplikacija ispisuje grešku

#### **UC7 - Definiranje mogućnosti rezervacije pojedinog parkirališnog mjesta**

- **Glavni sudionik:** Voditelj parkirališta
- **Cilj:** Definiranje mogućnosti rezervacije pojedinog parkirališnog mjesta
- **Sudionici:** Voditelj parkirališta, baza podataka
- **Preduvjet:** Korisnik mora imati ulogu voditelja parkinga, voditelj mora upravljati tim parkingom, parkiralište već postoji u bazi podataka
- **Opis osnovnog tijeka:**
  1. Voditelj odabire parkirališno mjesto
  2. Voditelj definira mogućnost rezervacije parkirališnog mjesta
  3. Informacije se spremaju u bazi podataka

4. Voditelju se javlja da su podatci uspješno spremljeni

– **Opis mogućih odstupanja:**

3.a Neuspješno spremanje podataka u bazu podataka

1. Aplikacija ispisuje grešku

**UC8 - Promjena mogućnosti rezervacije pojedinog parkirališnog mjesta**

– **Glavni sudionik:** Senzor zauzetosti

– **Cilj:** Omogućuje ponovnu rezervaciju novog tek oslobođenog mjesta

– **Sudionici:** Baza podataka

– **Preduvjet:**

– **Opis osnovnog tijeka:**

1. Vozilo dođe ili ode sa parkirališnog mjesta

2. Senzor registrira promjenu

3. Promjena se pohranjuje u bazu podataka

– **Opis mogućih odstupanja:**

2.a Kvar na senzoru zauzetosti

1. Hitan popravak ili zamjena senzora od strane vlasnika parkinga

**UC9 - Pregled osobnih podataka**

– **Glavni sudionik:** Administrator

– **Cilj:** Dati administratoru posebnu mogućnost pregleda osobnih podataka svih registriranih korisnika

– **Sudionici:** Baza podataka

– **Preduvjet:** Glavni sudionik nužno mora biti prijavljen kao administrator

– **Opis osnovnog tijeka:**

1. Administrator zatraži pregled osobnih podataka određenog registriranog korisnika

2. Određeni podaci se dohvaćaju iz baze podataka

3. Podaci se formatirano prikazuju administratoru

– **Opis mogućih odstupanja:**

2.a Neuspjelo dohvaćanje podataka iz baze podataka

1. Aplikacija ispisuje grešku

**UC10 - Promjena osobnih podataka**

– **Glavni sudionik:** Administrator

– **Cilj:** Dati administratoru posebnu mogućnost izmjene osobnih poda-

taka svih registriranih korisnika

- **Sudionici:** Baza podataka
- **Preduvjet:** Glavni sudionik nužno mora biti prijavljen kao administrator
- **Opis osnovnog tijeka:**
  1. Administrator zatraži pregled osobnih podataka određenog registriranog korisnika
  2. Određeni podaci se dohvaćaju iz baze podataka
  3. Administrator odluči mijenjati osobne podatke odabranog korisnika
  4. Administrator potvrdi promjene
  5. Promjene se pohranjuju u bazu podataka
- **Opis mogućih odstupanja:**
  - 2.a Neuspjelo dohvaćanje podataka iz baze podataka
    1. Aplikacija ispisuje grešku
  - 3.a Administrator je unio krivi format podataka
    1. Aplikacija upozorava na krivi format i ne dopušta potvrdu
  - 5.a Neuspjela pohrana u bazu podataka
    1. Aplikacija ispisuje grešku

#### UC11 - Registracija u sustav

- **Glavni sudionik:** Neregistrirani korisnik
- **Cilj:** Omogućuje neregistriranom korisniku registraciju u sustav
- **Sudionici:** Baza podataka
- **Preduvjet:** Glavni sudionik nije registriran od prije
- **Opis osnovnog tijeka:**
  1. Neregistrirani korisnik šalje zahtjev za registraciju sa određenom ulogom za koju se hoće prijaviti (voditelj parkinga ili klijent)
  2. Neregistrirani korisnik upisuje potrebne podatke (ime, prezime email,...)
  3. Neregistrirani korisnik potvrđuje svoje podatke
  4. Ako se korisnik prijavio kao voditelj parkinga administrator ga mora potvrditi
  5. Podaci o novoregistriranom korisniku se pohranjuju u bazu podataka
  6. Registracija se završava potvrdom preko email adrese
- **Opis mogućih odstupanja:**
  - 2.a Upisani podaci nisu u pravilnom formatu

1. Aplikacija upozorava na krivi format i ne dopušta nastavak

5.a Neuspjela pohrana u bazu podataka

1. Aplikacija dojavljuje grešku i ne dopušta nastavak

#### **UC12 - Prijava u sustav**

- **Glavni sudionik:** Bilo koji registrirani korisnik (klijent, vlasnik, admin)
- **Cilj:** Dobiti pristup korisničkom sučelju
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je već registriran u sustav
- **Opis osnovnog tijeka:**
  1. Korisnik unosi korisničko ime i lozinku
  2. Provjerava se postoje li upisani podaci u bazi podataka
  3. Korisnik dobiva pristup korisničkim funkcijama
- **Opis mogućih odstupanja:**
  - 2.a Upisano korisničko i lozinka su netočni
    1. Sustav obavještava korisnika o neispravnom upisu

#### **UC13 - Odjava iz sustava**

- **Glavni sudionik:** Bilo koji registrirani korisnik (klijent, vlasnik, admin)
- **Cilj:** Odjavljivanje iz korisničkog sučelja
- **Sudionici:**
- **Preduvjet:** Korisnik prijavljen u sustav
- **Opis osnovnog tijeka:**
  1. Korisnik zatraži odjavu iz sustava
  2. Korisnik izlazi iz korisničkog sučelja

#### **UC14 - Brisanje korisničkog računa**

- **Glavni sudionik:** Bilo koji registrirani korisnik (klijent, vlasnik, admin)
- **Cilj:** Omogućuje brisanje korisničkog računa
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav
- **Opis osnovnog tijeka:**
  1. Korisnik zatraži brisanje korisničkog računa
  2. Korisničko ime i lozinka korisnika se brišu iz baze podataka
  3. Izlazak iz korisničkog sučelja

#### **UC15 - Pregled statistike zauzetosti**

- **Glavni sudionik:** Voditelj

- **Cilj:** Voditelj može voditi evidenciju o zauzetosti svojih parkirališnih mjesta
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen kao voditelj
- **Opis osnovnog tijeka:**
  1. Voditelj odluči vidjeti zauzetost svih svojih parkirališnim mjesta
  2. Dohvaćanje podataka o zauzetosti iz baze podataka
  3. Voditelj pregledava podatke te donosi daljnje odluke
- **Opis mogućih odstupanja:**
  - 2.a Nema nijedno parkirno mjesto
    1. Ispis pogreške
    2. Voditelju omogućeno dodavanje parkirnih mjesta

#### **UC16 - Pregled dostupnih parkirališnih mjesta**

- **Glavni sudionik:** Registrirani i neregistrirani korisnici
- **Cilj:** Uvid u dostupna parkirališna mjesta
- **Sudionici:** Baza podataka
- **Preduvjet:** Nema
- **Opis osnovnog tijeka:**
  1. Odabir parkinga te parkirališnog mjesta
  2. Dohvaćanje podataka iz baze podataka
  3. Daljnje registriranje u slučaju da korisnik želi izvršiti rezervaciju
- **Opis mogućih odstupanja:**
  - 2.a Nema slobodnih parkirališnih mjesta
    1. Poruka o popunjenosti parkinga

#### **UC17 - Pregled zauzetosti parkirališnih mjesta**

- **Glavni sudionik:** Klijent
- **Cilj:** Svaki klijent može evidentirati zauzetost o svim parkirnim mjestima
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik prijavljen kao klijent te postoje parkirna mjesta
- **Opis osnovnog tijeka:**
  1. Klijent odluči vidjeti podatke o zauzetosti
  2. Dohvat svih parkirnih mjesta
  3. Odabir pojedinog parkirnog mjesta
  4. Dohvat podataka iz baze podataka



5. Uvid u dostavljene podatke

– **Opis mogućih odstupanja:**

2.a Nema nijedno parkirno mjesto

1. Poruka

**UC18 - Rezervacija parkirališnog mjesta**

– **Glavni sudionik:** Klijent

– **Cilj:** Klijent rezervira parkirališno mjesto

– **Sudionici:** Baza podataka

– **Preduvjet:** Korisnik rezerviran kao klijent

– **Opis osnovnog tijeka:**

1. Korisnik odluči rezervirati parkirno mjesto

2. Otvaranje karte te odabir tipa prijevoza te trajanje parkinga

3. Iscrtavanje rute do najbliže slobodnog parkirališnog mjesta

4. Klijent odabire jedan od 2 načina rezerviranja

5. Nastavak na plaćanje

– **Opis mogućih odstupanja:**

3.a Nema slobodnih parkirališnih mjesta

1. Ispis pogreške

2. Preusmjerenje nazad na ispunjavanje obrasca o traženju slobodnih parkirnih mjesta

**UC19 - Upis destinacije, tipa prijevoza i procjene trajanja parkinga**

– **Glavni sudionik:** Klijent

– **Cilj:** Unos podataka za rezervaciju parkinga

– **Sudionici:** Baza podataka

– **Preduvjet:** Korisnik registriran kao klijent

– **Opis osnovnog tijeka:**

1. Upis destinacije, tipa prijevoza i procjene trajanja parkinga

2. Filtriranje slobodnih parkirnih mjesta

3. Rezervacija parkirnog mjesta

– **Opis mogućih odstupanja:**

2.a Unos krivih podataka

1. Poruka te ponovna mogućnost unosa podataka

**UC20 - Uplata novca na račun**

– **Glavni sudionik:** Klijent

- **Cilj:** Uplata klijenta na račun vlasnika parkinga
- **Sudionici:** Baza podataka, Vlasnik parkinga
- **Preduvjet:** Korisnik registriran kao klijent te klijent izvršio rezervaciju
- **Opis osnovnog tijeka:**
  1. Uvid u cijenu željenog parkirališnog mjesta
  2. Klijent dodaje novac u novčanik
  3. Klijent plaća novac na vođiteljeov račun
  4. Ažuriranje novčanika u bazi podataka
  5. Potvrda o uspješnoj transakciji
- **Opis mogućih odstupanja:**
  - 2.a Klijent nema dovoljno novca u novčaniku
    1. Ispis poruke
    2. Preusmjeravanje na dodavanje novca u novčanik
  - 3.a Neuspjelo ažuriranje podataka
    1. Ispis poruke
    2. Preusmjeravanje na ponovno izvršavanje transakcije

#### UC21 -Pregled povijesti transakcija i rezervacija

- **Glavni sudionik:** Vođitelj parkirališta
- **Cilj:**Pregled povijesti transakcija i rezervacija
- **Sudionici:** Vođitelj parkirališta i baza podataka
- **Preduvjet:** Korisnik mora imati ulogu vođitelja parkinga, vođitelj mora upravljati tim parkingom, parkiralište već postoji u bazi podataka
- **Opis osnovnog tijeka:**
  1. Vođitelj zatraži pregled povijesti transakcija i rezervacija
  2. Podatci se dohvaćaju iz baze podataka
  3. Podatci se ispisuju vođitelju
- **Opis mogućih odstupanja:**
  - 2.a Neuspješno spremanje podataka u bazu podataka
    1. Aplikacija ispisuje grešku

#### UC22 - Potvrda računa vlasnika

- **Glavni sudionik:** Administrator
- **Cilj:** Potvrditi račun vođitelja parkinga
- **Sudionici:** Baza podataka
- **Preduvjet:** Vlasnik je napravio račun potvrdom email adrese te korisnik je prijavljen kao administrator

– Opis osnovnog tijeka:

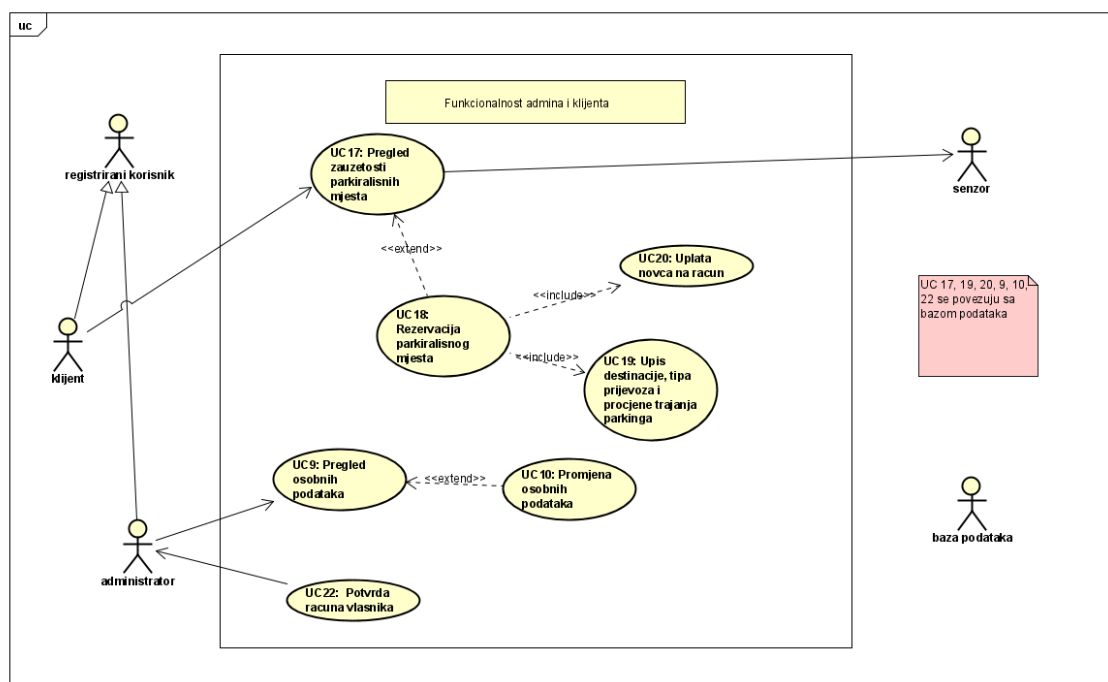
1. Administrator pregledava vođiteljev račun
2. Odabire funkcionalnost potvrđivanja u slučaju valjanosti računa su-  
protno odbacuje račun

– Opis mogućih odstupanja:

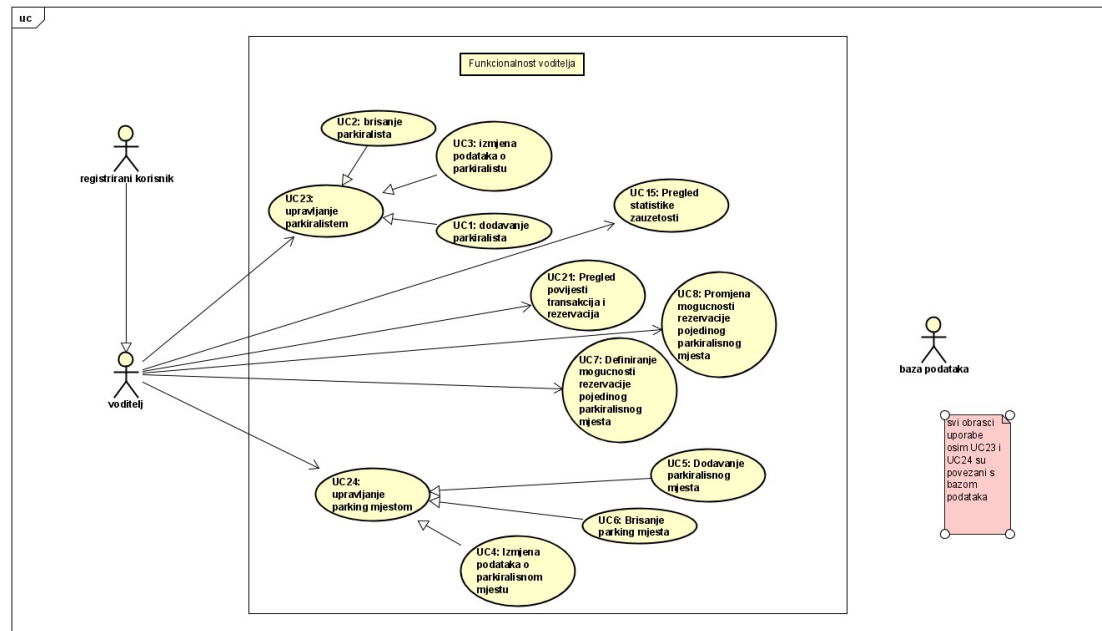
Dijagrami obrazaca uporabe



Slika 3.1: UC registrirani i neregistrirani korisnik



Slika 3.2: UC funkcionalnost klijenta i administratora



Slika 3.3: UC funkcionalnost voditelja

### 3.1.2 Sekvencijski dijagrami

#### *dio 1. revizije*

*Nacrtati sekvencijske dijagrame koji modeliraju najvažnije dijelove sustava (max. 4 dijagrama). Ukoliko postoji nedoumica oko odabira, razjasniti s asistentom. Uz svaki dijagram napisati detaljni opis dijagrama.*

## 3.2 Ostali zahtjevi

### *dio 1. revizije*

Nefunkcionalni zahtjevi i zahtjevi domene primjene dopunjuju funkcionalne zahtjeve. Oni opisuju **kako se sustav treba ponašati** i koja **ograničenja** treba poštivati (performanse, korisničko iskustvo, pouzdanost, standardi kvalitete, sigurnost...). Primjeri takvih zahtjeva u Vašem projektu mogu biti: podržani jezici korisničkog sučelja, vrijeme odziva, najveći mogući podržani broj korisnika, podržane web/mobilne platforme, razina zaštite (protokoli komunikacije, kriptiranje...)... Svaki takav zahtjev potrebno je navesti u jednoj ili dvije rečenice.

## 4. Arhitektura i dizajn sustava

### *dio 1. revizije*

*Potrebno je opisati stil arhitekture te identificirati: podsustave, preslikavanje na radnu platformu, spremišta podataka, mrežne protokole, globalni upravljački tok i sklopovsko-programске zahtjeve. Po točkama razraditi i popratiti odgovarajućim skicama:*

- izbor arhitekture temeljem principa oblikovanja pokazanih na predavanjima (objasniti zašto ste baš odabrali takvu arhitekturu)*
- organizaciju sustava s najviše razine apstrakcije (npr. klijent-poslužitelj, baza podataka, datotečni sustav, grafičko sučelje)*
- organizaciju aplikacije (npr. slojevi frontend i backend, MVC arhitektura)*

### 4.1 Baza podataka

#### *dio 1. revizije*

*Potrebno je opisati koju vrstu i implementaciju baze podataka ste odabrali, glavne komponente od kojih se sastoji i slično.*

#### 4.1.1 Opis tablica

*Svaku tablicu je potrebno opisati po zadanom predlošku. Lijevo se nalazi točno ime varijable u bazi podataka, u sredini se nalazi tip podataka, a desno se nalazi opis varijable. Svjetlozelenom bojom označite primarni ključ. Svjetlo plavom označite strani ključ*

korisnik - ime tablice		
IDKorisnik	INT	Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod
korisnickoIme	VARCHAR	
email	VARCHAR	
ime	VARCHAR	
primjer	VARCHAR	

#### 4.1.2 Dijagram baze podataka

*U ovom potpoglavlju potrebno je umetnuti dijagram baze podataka. Primarni i strani ključevi moraju biti označeni, a tablice povezane. Bazu podataka je potrebno normalizirati. Podsjetite se kolegija "Baze podataka".*



## 4.2 Dijagram razreda

*Potrebno je priložiti dijagram razreda s pripadajućim opisom. Zbog preglednosti je moguće dijagram razlomiti na više njih, ali moraju biti grupirani prema sličnim razinama apstrakcije i srodnim funkcionalnostima.*

### **dio 1. revizije**

*Prilikom prve predaje projekta, potrebno je priložiti potpuno razrađen dijagram razreda vezan uz **generičku funkcionalnost** sustava. Ostale funkcionalnosti trebaju biti idejno razrađene u dijagramu sa sljedećim komponentama: nazivi razreda, nazivi metoda i vrste pristupa metodama (npr. javni, zaštićeni), nazivi atributa razreda, veze i odnosi između razreda.*

### **dio 2. revizije**

*Prilikom druge predaje projekta dijagram razreda i opisi moraju odgovarati stvarnom stanju implementacije*

## 4.3 Dijagram stanja

### *dio 2. revizije*

*Potrebno je priložiti dijagram stanja i opisati ga. Dovoljan je jedan dijagram stanja koji prikazuje **značajan dio funkcionalnosti** sustava. Na primjer, stanja korisničkog sučelja i tijekom korištenja neke ključne funkcionalnosti jesu značajan dio sustava, a registracija i prijava nisu.*

## 4.4 Dijagram aktivnosti

*dio 2. revizije*

*Potrebno je priložiti dijagram aktivnosti s pripadajućim opisom. Dijagram aktivnosti treba prikazivati značajan dio sustava.*

## 4.5 Dijagram komponenti

### *dio 2. revizije*

*Potrebno je priložiti dijagram komponenti s pripadajućim opisom. Dijagram komponenti treba prikazivati strukturu cijele aplikacije.*

## 5. Implementacija i korisničko sučelje

### 5.1 Korištene tehnologije i alati

#### *dio 2. revizije*

*Detaljno navesti sve tehnologije i alate koji su primijenjeni pri izradi dokumentacije i aplikacije. Ukratko ih opisati, te navesti njihovo značenje i mjesto primjene. Za svaki navedeni alat i tehnologiju je potrebno **navesti internet poveznicu** gdje se mogu preuzeti ili više saznati o njima.*

## 5.2 Ispitivanje programskog rješenja

### *dio 2. revizije*

*U ovom poglavlju je potrebno opisati provedbu ispitivanja implementiranih funkcionalnosti na razini komponenti i na razini cijelog sustava s prikazom odabranih ispitnih slučajeva. Studenti trebaju ispitati temeljnu funkcionalnost i rubne uvjete.*

### 5.2.1 Ispitivanje komponenti

*Potrebno je provesti ispitivanje jedinica (engl. unit testing) nad razredima koji implementiraju temeljne funkcionalnosti. Razraditi **minimalno 6 ispitnih slučajeva** u kojima će se ispitati redovni slučajevi, rubni uvjeti te izazivanje pogreške (engl. exception throwing). Poželjno je stvoriti i ispitni slučaj koji koristi funkcionalnosti koje nisu implementirane. Potrebno je priložiti izvorni kôd svih ispitnih slučajeva te prikaz rezultata izvođenja ispita u razvojnom okruženju (prolaz/pad ispita).*

### 5.2.2 Ispitivanje sustava

*Potrebno je provesti i opisati ispitivanje sustava koristeći radni okvir Selenium<sup>1</sup>. Razraditi **minimalno 4 ispitna slučaja** u kojima će se ispitati redovni slučajevi, rubni uvjeti te poziv funkcionalnosti koja nije implementirana/izaziva pogrešku kako bi se vidjelo na koji način sustav reagira kada nešto nije u potpunosti ostvareno. Ispitni slučaj se treba sastojati od ulaza (npr. korisničko ime i lozinka), očekivanog izlaza ili rezultata, koraka ispitivanja i dobivenog izlaza ili rezultata.*

*Izradu ispitnih slučajeva pomoću radnog okvira Selenium moguće je provesti pomoću jednog od sljedeća dva alata:*

- dodatak za preglednik **Selenium IDE** - snimanje korisnikovih akcija radi automatskog ponavljanja ispita*
- **Selenium WebDriver** - podrška za pisanje ispita u jezicima Java, C#, PHP koristeći posebno programsko sučelje.*

*Detalji o korištenju alata Selenium bit će prikazani na posebnom predavanju tijekom semestra.*

---

<sup>1</sup><https://www.seleniumhq.org/>

## 5.3 Dijagram razmještaja

### *dio 2. revizije*

*Potrebno je umetnuti **specifikacijski** dijagram razmještaja i opisati ga. Moguće je umjesto specifikacijskog dijagrama razmještaja umetnuti dijagram razmještaja instanci, pod uvjetom da taj dijagram bolje opisuje neki važniji dio sustava.*

## 5.4 Upute za puštanje u pogon

### *dio 2. revizije*

*U ovom poglavlju potrebno je dati upute za puštanje u pogon (engl. deployment) ostvarene aplikacije. Na primjer, za web aplikacije, opisati postupak kojim se od izvornog kôda dolazi do potpuno postavljene baze podataka i poslužitelja koji odgovara na upite korisnika. Za mobilnu aplikaciju, postupak kojim se aplikacija izgradi, te postavi na neku od trgovina. Za stolnu (engl. desktop) aplikaciju, postupak kojim se aplikacija instalira na računalo. Ukoliko mobilne i stolne aplikacije komuniciraju s poslužiteljem i/ili bazom podataka, opisati i postupak njihovog postavljanja. Pri izradi uputa preporučuje se **naglasiti korake instalacije uporabom natuknica** te koristiti što je više moguće **slike ekrana** (engl. screenshots) kako bi upute bile jasne i jednostavne za slijediti.*

*Dovršenu aplikaciju potrebno je pokrenuti na javno dostupnom poslužitelju. Studentima se preporuča korištenje neke od sljedećih besplatnih usluga: Amazon AWS, Microsoft Azure ili Heroku. Mobilne aplikacije trebaju biti objavljene na F-Droid, Google Play ili Amazon App trgovini.*



## 6. Zaključak i budući rad

### *dio 2. revizije*

*U ovom poglavlju potrebno je napisati osvrt na vrijeme izrade projektnog zadatka, koji su tehnički izazovi prepoznati, jesu li riješeni ili kako bi mogli biti riješeni, koja su znanja stečena pri izradi projekta, koja bi znanja bila posebno potrebna za brže i kvalitetnije ostvarenje projekta i koje bi bile perspektive za nastavak rada u projektnoj grupi.*

*Potrebno je točno popisati funkcionalnosti koje nisu implementirane u ostvarenoj aplikaciji.*

# Popis literature

## *Kontinuirano osvježavanje*

*Popisati sve reference i literaturu koja je pomogla pri ostvarivanju projekta.*

1. Programsko inženjerstvo, FER ZEMRIS, <http://www.fer.hr/predmet/proinz>
2. I. Sommerville, "Software engineering", 8th ed, Addison Wesley, 2007.
3. T.C.Lethbridge, R.Langaniere, "Object-Oriented Software Engineering", 2nd ed. McGraw-Hill, 2005.
4. I. Marsic, "Software engineering book", Department of Electrical and Computer Engineering, Rutgers University, <http://www.ece.rutgers.edu/~marsic/books/SE>
5. The Unified Modeling Language, <https://www.uml-diagrams.org/>
6. Astah Community, <http://astah.net/editions/uml-new>

# Indeks slika i dijagrama

# Dodatak: Prikaz aktivnosti grupe

## Dnevnik sastajanja

### *Kontinuirano osvježavanje*

*U ovom dijelu potrebno je redovito osvježavati dnevnik sastajanja prema predlošku.*

#### 1. sastanak

- Datum: u ovom formatu: 4. studenoga 2021.
- Prisustvovali: I.Prezime, I.Prezime
- Teme sastanka:
  - \* opis prve teme
  - \* opis druge teme

#### 2. sastanak

- Datum: u ovom formatu: 4. studenoga 2021.
- Prisustvovali: I.Prezime, I.Prezime
- Teme sastanka:
  - \* opis prve teme
  - \* opis druge teme

## Tablica aktivnosti

### *Kontinuirano osvježavanje*

*Napomena: Doprinosi u aktivnostima treba navesti u satima po članovima grupe po aktivnosti.*

	Ime Prezime voditelja	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime
Upravljanje projektom							
Opis projektnog zadatka							
Funkcionalni zahtjevi							
Opis pojedinih obrazaca							
Dijagram obrazaca							
Sekvencijski dijagrami							
Opis ostalih zahtjeva							
Arhitektura i dizajn sustava							
Baza podataka							
Dijagram razreda							
Dijagram stanja							
Dijagram aktivnosti							
Dijagram komponenti							
Korištene tehnologije i alati							
Ispitivanje programskog rješenja							

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

	Ime Prezime voditelja	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime
Dijagram razmještaja							
Upute za puštanje u pogon							
Dnevnik sastajanja							
Zaključak i budući rad							
Popis literature							
<i>Dodatne stavke kako ste podijelili izradu aplikacije</i>							
<i>npr. izrada početne stranice</i>							
<i>izrada baze podataka</i>							
<i>spajanje s bazom podataka</i>							
<i>back end</i>							

## Dijagrami pregleda promjena

### *dio 2. revizije*

*Prenijeti dijagram pregleda promjena nad datotekama projekta. Potrebno je na kraju projekta generirane grafove s gitlaba prenijeti u ovo poglavlje dokumentacije. Dijagrami za vlastiti projekt se mogu preuzeti s [gitlab.com](https://gitlab.com) stranice, u izborniku Repository, pritiskom na stavku Contributors.*