

Programsko inženjerstvo

Ak. god. 2023./2024.

DentAll

Dokumentacija, Rev. 1

Grupa: *ZuBit*

Voditelj: *Danko Delimar*

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

Nastavnik: *Goran Rajić*

Sadržaj

1 Dnevnik promjena dokumentacije	3
2 Opis projektnog zadatka	5
3 Specifikacija programske potpore	9
3.1 Funkcionalni zahtjevi	9
3.1.1 Obrasci uporabe	11
3.1.2 Sekvencijski dijagrami	18
3.2 Ostali zahtjevi	19
4 Arhitektura i dizajn sustava	20
4.1 Baza podataka	20
4.1.1 Opis tablica	21
4.1.2 Dijagram baze podataka	23
4.2 Dijagram razreda	24
4.3 Dijagram stanja	25
4.4 Dijagram aktivnosti	26
4.5 Dijagram komponenti	27
5 Implementacija i korisničko sučelje	28
5.1 Korištene tehnologije i alati	28
5.2 Ispitivanje programskog rješenja	29
5.2.1 Ispitivanje komponenti	29
5.2.2 Ispitivanje sustava	29
5.3 Dijagram razmještaja	30
5.4 Upute za puštanje u pogon	31
6 Zaključak i budući rad	32
Popis literature	33
Indeks slika i dijagrama	34

Dodatak: Prikaz aktivnosti grupe

35

1. Dnevnik promjena dokumentacije

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Napravljen predložak.	*	20.10.2023.
0.2	Dodan opis projekta, dionici, funkcionalni zahtjevi i neki use casevi	*	25.10.2023.
0.3	Dodan ostatak <i>Use Caseva</i>	*	29.10.2023.
0.6		*	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.
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.

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

Rev.	Opis promjene/dodatka	Autori	Datum
2.0	Konačni tekst predložka 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

Ideja ovog projekta je napraviti aplikaciju koja olakšava dentalnim klinikama proces ponude potpunog plana liječenja koji uključuje medicinske podatke, smještaj u kojem će se pacijent nalaziti i prijevoz od klinike do smještaja i obratno. Ovo je važna aplikacija za klinike koje žele svoje usluge ponuditi pacijentima iz drugih gradova ili država tzv. "dentalni turizam" zato što ih mogućnost takve ponude čini puno privlačnijima potencijalnim klijentima koji nisu upoznati s lokacijom na kojoj se klinika nalazi. Time se klijenti, a i klinike mogu fokusirati samo na posao, a naša aplikacija obavlja svu logistiku.

Aplikaciju mogu koristiti sve dentalne klinike kojima je potrebna organizacija smještaja i prijevoza oko tog smještaja. Posebice ciljamo na klinike koje žele proširiti posao oglašavanjem u inozemstvu i žele ponuditi potpuni paket potencijalnim klijentima kako bi se istaknule naspram konkurencije. Takvim klinikama je ovakva aplikacija jako potrebna zato što je jedan od glavnih razloga zašto dentalni turizam nije toliko popularan koliko može biti je činjenica da je potencijalnim klijentima težak zadatak organizirati si cijeli put u neku stranu državu o kojoj možda ništa ne znaju.

U trenutnom obliku aplikacija cilja dentalne klinike jer su zahvati koje dentalne klinike nude relativno jednostavni za definirati u smislu termina i taj sektor medicine ima veliki potencijal za privući ljude iz stranih država jer je usluga puno jeftinija. No aplikacija se jednostavno može promijeniti i nadograditi da bude za cijeli medicinski sektor ili bilo kakvu djelatnost koja želi ponuditi takav potpuni paket smještaja, prijevoza i usluge.

Aplikacija se može koristiti iz perspektive tri administratora koji predstavljaju tri dijela paketa koje aplikacija spaja, a to su smještajni administrator, prijevozni administrator i korisnički administrator. Krajnji korisnik tj. pacijent klinike nema interakciju s aplikacijom nego samo na račun elektroničke pošte koji mu je definiran od strane korisničkog administratora prima informaciju o cijelom paketu smještaja, prijevoza, kliničkih usluga i datuma kada treba stići u državu.

Smještajni administrator ima najveće ovlasti u sustavu i može dodavati nove korisnike i dodjeljivati im uloge. Glavna uloga smještajnog administratora je dodavanje u sustav novih smještaja i popunjavanje osnovnih podataka o tim smještajima. Smještajnim administratorima je omogućen pregled lokacije stana na karti uz pomoć Google Maps API-a. Svi osnovni podatci se mogu mijenjati, a cijeli smještaj se može izbrisati. Osnovni podatci uključuju:

- Tip stana (u vlasništvu klinike ili ne)
- Veličina stana
- Ocjena stana
- Adresa
- Vremenska dostupnost

Prijevozni administrator ima glavnu ulogu u unosu osnovnih podataka o prijevoznicima koji su dostupni u aplikaciji. Osnovni osobni podatci prijevoznika se ne mogu mijenjati, svi ostali se mogu, a cijeli prijevoznik se može izbrisati. Osnovni podatci uključuju:

- Ime i prezime prijevoznika
- Email
- Broj telefona
- Tip prijevoznog sredstva
- Kapacitet prijevoznog sredstva
- Radno vrijeme

Korisnički administrator ima glavnu ulogu u unosu osnovnih podataka o pacijentima klinike. Detalje tretmana ne unose korisnički administratori već se oni dohvaćaju iz postojećeg sustava klinike. Moguće je mijenjati podatke o pacijentima i brisati pacijente. Osnovi podatci o pacijentima su:

- Ime i prezime pacijenta
- Email

- Broj telefona
- Vrijeme dolaska u državu
- Mjesto dolaska u državu
- Preferencija o veličini i ocjeni stana

Svaki korisnički račun može imati više administratorskih uloga. Svaki administrator ima web sučelje unutar kojeg može obavljati potrebne operacije dodavanja, izmjenjivanja i brisanja podataka. Takvo jednostavno web sučelje olakšava posao administratora kada moraju raditi potrebne izmjene.

Kada korisnički administrator unese novog korisnika aplikacija mora provjeriti postoji li dostupan smještaj u danom terminu. Kada se pronade termin u kojem postoji slobodan smještaj čeka se potvrda od interne aplikacije klinike da je zaključan plan tretmana. Kada je zaključan plan tretmana, aplikacija automatski šalje poruku elektroničke pošte klijentu sa cijelim paketom i šalje dostupnim prijevoznicima informacije o klijentu i informacije o smještaju.

Aplikacija ne prati podatke o tretmanu pacijenta nego te podatke mora pratiti interna aplikacija klinike i ti podatci se uključuju u kompletan plan iz te interne aplikacije. Također aplikacija ne dozvoljava nikakav pristup ili pregled ne registriranim korisnicima s obzirom da je zamišljena za internu uporabu unutra klinike, a ne za vanjsku uporabu od strane korisnika. Sva komunikacija s vanjskim korisnicima kao što su pacijenti ili prijevoznici odvija se automatski preko elektroničke pošte.

S obzirom na to da je glavna funkcionalnost aplikacije smanjenje mentalnog napora na logistiku, moguća proširenja aplikacije većinom su vezana za dodatno olakšanje logistike. Moguća je implementacija povezanosti s internacionalnim prijevoznicima kako bi klijenti klinika morali učiniti što manje. Također je moguća integracija sa sustavima vlakova ili avioprijevoznika kako bi se klijentu pružio potpuno organizirani plan puta do države destinacije. Druga vrsta mogućeg proširenja je automatsko izračunavanje troška ovakvog plana i automatsko uračunavanje tog troška u cjelokupni trošak tretmana. U taj trošak bi bili uključeni troškovi poput troškova prijevoza i troška čišćenja stanova u kojima klijenti borave za vrijeme ostanka u državi. Povezano s tim mogu se u aplikaciju uključiti i usluge čistača kojima se automatski šalje poruka da treba očistiti određeni stan.

U našem kratkom istraživanju sličnih aplikacija nismo pronašli aplikaciju koja rješava isti problem. Većina se aplikacija u sektoru dentalnog turizma bavi prikazom tretmana pacijentima, ali ne diraju logistiku kako doći do neke klinike ili gdje stanovati u vremenu kada im se obavlja posao na zubima. Mislimo da zato ova aplikacija ima dobru priliku iskoristiti nišu u tržištu koja trenutno nije jako zastupljena. Također je moguće da trenutno ovakvu funkcionalnost klinike rješavaju internim rješenjima no naša aplikacija će omogućiti uz jednostavan oblik pretplate da svaka zainteresirana klinika ima ovakve mogućnosti.

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. Dentalne klinike
2. Vlasnici smještaja
3. Prijevoznici
4. Administratori koji koriste aplikaciju
5. Klijenti tj. pacijenti
6. Razvojni tim

Aktori i njihovi funkcionalni zahtjevi:

1. Smještajni administrator (inicijator) može:
 - (a) Pregledavati podatke o smještajima (tip, ocjena, adresa, period dostupnosti)
 - (b) Dodavati nove korisnike (ne klijente nego korisnike sustava)
 - (c) Korisnicima dodavati nove uloge
 - (d) Dodati, izmjenjivati i brisati podatke o smještaju
 - (e) Vidjeti prikaz smještaja na karti

2. Prijevoznički administrator (inicijator) može:

- (a) Pregledavati podatke o prijevoznicima
- (b) Dodati osnovne osobne podatke prijevoznika
- (c) Dodati kontaktne podatke i podatke o vrsti i kapacitetu vozila
- (d) Izmjenjivati neosnovne podatke (kontakt, vrsta i kapacitet vozila)
- (e) Izbrisati prijevoznika

3. Korisnički administrator (inicijator) može:

- (a) Dodati podatke o klijentima (osobni podatci, kontakt, preferencije o smještaju)

4. Baza podataka (sudionik) može:

- (a) Pohranjuje sve podatke o prijevoznicima
- (b) Pohranjuje sve podatke o smještaju
- (c) Pohranjuje ne medicinske podatke o klijentima

5. Aplikacija (sudionik) može:

- (a) Dohvatiti medicinske podatke s vanjskog sučelja
- (b) Poslati poruku elektroničke pošte klijentima i
- (c) Pohranjuje ne medicinske podatke o klijentima

3.1.1 Obrasci uporabe

dio 1. revizije

Opis obrazaca uporabe

Funkcionalne zahtjeve razraditi u obliku obrazaca uporabe. Svaki obrazac je potrebno razraditi prema donjem predlošku. Ukoliko u nekom koraku može doći do odstupanja, potrebno je to odstupanje opisati i po mogućnosti ponuditi rješenje kojim bi se tijekom obrasca vratio na osnovni tijek.

UC<broj obrasca> -<ime obrasca>

- **Glavni sudionik:** <sudionik>
- **Cilj:** <cilj>
- **Sudionici:** <sudionici>
- **Preduvjet:** <preduvjet>
- **Opis osnovnog tijeka:**
 1. <opis korak jedan>
 2. <opis korak dva>
 3. <opis korak tri>
 4. <opis korak četiri>
 5. <opis korak pet>
- **Opis mogućih odstupanja:**
 - 2.a <opis mogućeg scenarija odstupanja u koraku 2>
 1. <opis rješenja mogućeg scenarija korak 1>
 2. <opis rješenja mogućeg scenarija korak 2>
 - 2.b <opis mogućeg scenarija odstupanja u koraku 2>
 - 3.a <opis mogućeg scenarija odstupanja u koraku 3>

UC1 - Prijava smještajnog administratora

- **Glavni sudionik:** Smještajni administrator
- **Cilj:** Prijaviti se u sustav kao smještajni administrator
- **Sudionici:** Baza podataka
- **Preduvjet:** Registracija?
- **Opis osnovnog tijeka:**
 1. Unos korisničkog imena i lozinke te odabir uloge smještajnog administratora

2. Potvrda o postojanju računa i ispravnosti podataka
3. Pristup funkcijama smještajnog administratora
- **Opis mogućih odstupanja:**
 - 2.a Uneseni podatci nisu u točnom formatu
 1. Obavijestiti korisnika koji podatci nisu u točnom formatu
 2. Korisnik mijenja potrebne podatke i pokušava opet
 - 2.b Korisnički račun ne postoji, lozinka nije točna ili korisničko ime nije točno
 1. Obavijestiti korisnika da su korisničko ime ili lozinka netočni
 2. Korisnik mijenja potrebne podatke i pokušava opet

UC2 - Prijava prijevoznog administratora

- **Glavni sudionik:** Prijevozni administrator
- **Cilj:** Prijaviti se u sustav kao prijevozni administrator
- **Sudionici:** Baza podataka
- **Preduvjet:** Registracija?
- **Opis osnovnog tijeka:**
 1. Unos korisničkog imena i lozinke te odabir uloge prijevoznog administratora
 2. Potvrda o postojanju računa i ispravnosti podataka
 3. Pristup funkcijama prijevoznog administratora
- **Opis mogućih odstupanja:**
 - 2.a Uneseni podatci nisu u točnom formatu
 1. Obavijestiti korisnika koji podatci nisu u točnom formatu
 2. Korisnik mijenja potrebne podatke i pokušava opet
 - 2.b Korisnički račun ne postoji, lozinka nije točna ili korisničko ime nije točno
 1. Obavijestiti korisnika da su korisničko ime ili lozinka netočni
 2. Korisnik mijenja potrebne podatke i pokušava opet

UC3 - Prijava korisničkog administratora

- **Glavni sudionik:** Korisnički administrator
- **Cilj:** Prijaviti se u sustav kao korisnički administrator
- **Sudionici:** Baza podataka
- **Preduvjet:** Registracija?
- **Opis osnovnog tijeka:**

1. Unos korisničkog imena i lozinke te odabir uloge korisničkog administratora
2. Potvrda o postojanju računa i ispravnosti podataka
3. Pristup funkcijama korisničkog administratora
- **Opis mogućih odstupanja:**
 - 2.a Uneseni podatci nisu u točnom formatu
 1. Obavijestiti korisnika koji podatci nisu u točnom formatu
 2. Korisnik mijenja potrebne podatke i pokušava opet
 - 2.b Korisnički račun ne postoji, lozinka nije točna ili korisničko ime nije točno
 1. Obavijestiti korisnika da su korisničko ime ili lozinka netočni
 2. Korisnik mijenja potrebne podatke i pokušava opet

UC4 - Dodavanje novog korisnika

- **Glavni sudionik:** Smještajni administrator
- **Cilj:** Dodati novog korisnika sustava
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava smještajnog administratora
- **Opis osnovnog tijeka:**
 1. Unos korisničkog imena i lozinke novog korisnika te odabir svih uloga koje će korisnik imati
 2. Provjera postoji li već korisnik s takvim imenom
 3. Upis novog korisnika u bazu podataka
- **Opis mogućih odstupanja:**
 - 2.a Polja korisničko ime i/ili lozinka su prazni
 1. Obavijestiti korisnika koji podatci su prazni
 2. Korisnik mijenja potrebne podatke i pokušava opet
 - 2.b Postoji korisnički račun s tim imenom
 1. Obavijestiti korisnika da postoji račun s tim imenom
 2. Korisnik mijenja korisničko ime i pokušava opet
 - 2.c Nije odabrana niti jedna uloga
 1. Obavijestiti korisnika da račun mora imati neku ulogu
 2. Korisnik dodaje jednu ili više uloga i pokušava opet

UC5 - Pregled postojećih smještaja

- **Glavni sudionik:** Smještajni administrator

- **Cilj:** Pregledati postojeće smještaje
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava smještajnog administratora
- **Opis osnovnog tijeka:**
 1. Smještajni administrator odabire opcije "Postojeći smještaji"
 2. Aplikacija prikazuje kao listu sve postojeće smještaje

UC6 - Dodavanje novog smještaja

- **Glavni sudionik:** Smještajni administrator
- **Cilj:** Dodati novi smještaj
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava smještajnog administratora
- **Opis osnovnog tijeka:**
 1. Unos potrebnih podataka za dodavanje novog smještaja
 2. Provjera jesu li uneseni svi podatci
 3. Upis novog smještaja u bazu podataka
- **Opis mogućih odstupanja:**
 - 2.a Neki od podataka nije unesen
 1. Obavijestiti korisnika koji podatci su prazni
 2. Korisnik mijenja potrebne podatke i pokušava opet

UC7 - Promjena podataka smještaja

- **Glavni sudionik:** Smještajni administrator
- **Cilj:** Promijeniti podatke smještaja
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava smještajnog administratora
- **Opis osnovnog tijeka:**
 1. Smještajni administrator odabire "Promijeni" opciju na smještaju
 2. Smještajni administrator mijenja podatke o smještaju
 3. Smještajni administrator sprema promijene
 4. Pregledava se ispravnost podataka
 5. Mijenjaju se podatci u bazi podataka
- **Opis mogućih odstupanja:**
 - 3.a Smještajni administrator izađe bez spremanja
 1. Podatci se ne spremaju u bazu podataka
 - 4.a Neki od podataka nije unesen

1. Obavijestiti smještajnog administratora koji podatci su prazni
2. Smještajni administrator mijenja potrebne podatke i pokušava opet

UC8 - Brisanje smještaja

- **Glavni sudionik:** Smještajni administrator
- **Cilj:** Promijeniti podatke smještaja
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava smještajnog administratora
- **Opis osnovnog tijeka:**
 1. Smještajni administrator odabire "Izbriši" opciju na smještaju
 2. Aplikacija pokazuje upit smještajnom administratoru je li siguran
 3. Ako je smještaj se briše iz baze u protivnom se ništa ne desi

UC9 - Pregled postojećih prijevoznika

- **Glavni sudionik:** Prijevozni administrator
- **Cilj:** Pregledati postojeće prijevoznike
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava prijevoznog administratora
- **Opis osnovnog tijeka:**
 1. Prijevozni administrator odabire opcije "Postojeći prijevoznici"
 2. Aplikacija prikazuje kao listu sve postojeće prijevoznike

UC10 - Dodavanje novog prijevoznika

- **Glavni sudionik:** Prijevozni administrator
- **Cilj:** Dodati novog prijevoznika
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava prijevoznog administratora
- **Opis osnovnog tijeka:**
 1. Unos potrebnih podataka za dodavanje novog prijevoznika
 2. Provjera jesu li uneseni svi podatci
 3. Upis novog prijevoznika u bazu podataka
- **Opis mogućih odstupanja:**
 - 2.a Neki od podataka nije unesen
 1. Obavijestiti prijevoznog administratora koji podatci su prazni
 2. Prijevozni administrator mijenja potrebne podatke i pokušava opet

UC11 - Promjena podataka smještaja

- **Glavni sudionik:** Prijevozni administrator
- **Cilj:** Promijeniti podatke o prijevozniku
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava prijevoznog administratora
- **Opis osnovnog tijeka:**
 1. Prijevozni administrator odabire "Promijeni" opciju na postojećem prijevozniku
 2. Prijevozni administrator mijenja podatke o prijevozniku
 3. Prijevozni administrator sprema promijene
 4. Pregledava se ispravnost podataka
 5. Mijenjaju se podatci u bazi podataka
- **Opis mogućih odstupanja:**
 - 3.a Prijevozni administrator izađe bez spremanja
 1. Podatci se ne spremaju u bazu podataka
 - 4.a Neki od podataka nije unesen
 1. Obavijestiti prijevoznog administratora koji podatci su prazni
 2. Prijevozni administrator mijenja potrebne podatke i pokušava opet

UC12 - Brisanje prijevoznika

- **Glavni sudionik:** Prijevozni administrator
- **Cilj:** Promijeniti podatke smještaja
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava prijevoznog administratora
- **Opis osnovnog tijeka:**
 1. Prijevozni administrator odabire "Izbriši" opciju na prijevozniku
 2. Aplikacija pokazuje upit prijevoznom administratoru je li siguran
 3. Ako je, prijevoznik se briše iz baze u protivnom se ništa ne desi

UC13 - Dodavanje novog korisnika

- **Glavni sudionik:** Korisnički administrator
- **Cilj:** Dodati novog korisnika
- **Sudionici:** Baza podataka
- **Preduvjet:** Prijava korisničkog administratora
- **Opis osnovnog tijeka:**
 1. Unos potrebnih podataka za dodavanje novog korisnika
 2. Dohvat medicinskih podataka o korisniku

3. Provjera jesu li uneseni svi podatci
4. Upis novog prijevoznika u bazu podataka
- **Opis mogućih odstupanja:**
 - 2.b Ne mogu se dohvatiti medicinski podatci
 1. Aplikacija pokušava dohvatiti podatke opet
 2. Ako je podatke nemoguće dohvatiti, nemoguće je unijeti novog korisnika
 - 3.a Neki od podataka nije unesen
 1. Obavijestiti korisničkog administratora koji podatci su prazni
 2. Korisnički administrator mijenja potrebne podatke i pokušava opet

Dijagrami obrazaca uporabe

Prikazati odnos aktora i obrazaca uporabe odgovarajućim UML dijagramom. Nije nužno nacrtati sve na jednom dijagramu. Modelirati po razinama apstrakcije i skupovima srodnih funkcionalnosti.

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

U našem projektu odabrali smo PostgreSQL kao susatav za upravljanje bazama podataka. Implementacija naše PostgreSQL baze podataka obuhvaća nekoliko ključnih elemenata, uključujući organizaciju podataka u tablicama i uspostavljanje veza između tablica radi složenih upita. Baza podatka sastoji se od slijedećih entiteta:

- KLINIKA
- SMJEŠTAJ
- PRIJEVOZNIK
- VOZILO
- KORISNIK
- PUTOVANJE

4.1.1 Opis tablica

KLINIKA Entitet KLINIKA sadrži informacije o ID-u klinike, nazivu i adresi. Prema tome, entitet KLINIKA posjeduje sljedeće atribute: IDKlinika, naziv i adresa. Entitet KLINIKA u vezi je *One-to-Many* s entitetom SMJEŠTAJ preko atributa IDKlinika i u vezi *One-to-Many* s entitetom PUTOVANJE preko atributa IDKlinika.

KLINIKA		
IDKlinika	INT	Identifikacijski ključ klinike
naziv	VARCHAR	Naziv klinike
adresa	VARCHAR	Adresa klinike

SMJEŠTAJ Entitet SMJEŠTAJ sadrži podatke o ID-u smještaja, tipu stana, kategoriji opremljenosti, adresi kao i vremenskom periodu dostupnosti za korištenje. Sukladno tome, entitet SMJEŠTAJ posjeduje sljedeće atribute: IDSmještaj, tip, kategorija, adresa i dostupnost. Entitet SMJEŠTAJ u vezi je *Many-to-One* s entitetom KLINIKA preko atributa IDKlinika i u vezi *One-to-Many* s entitetom PUTOVANJE preko atributa IDPutovanje.

SMJEŠTAJ		
IDSmještaj	INT	Identifikacijski ključ smještaja
tip	VARCHAR	Tip stana
kategorija	VARCHAR	Kategorija opremljenosti
adresa	VARCHAR	Adresa smještaja
dostupnost	TIMESTAMP	Vremenski period dostupnosti za korištenje
IDKlinika	INT	Identifikacijski ključ klinike

PRIJEVOZNIK Entitet PRIJEVOZNIK sadrži informacije o ID-u prijevoznika, kontaktnim podacima i o radnom vremenu u kojem su raspoloživi. Prema tome, entitet PRIJEVOZNIK posjeduje sljedeće atribute: IDKPrijevoznik, kontakt i radnoVrijeme. Entitet PRIJEVOZNIK u vezi je *One-to-Many* s entitetom VOZILO preko atributa IDPrijevoznik i u vezi *One-to-Many* s entitetom PUTOVANJE preko atributa IDPrijevoznik.

PRIJEVOZNIK		
IDPrijevoznik	INT	Identifikacijski ključ prijevoznika
kontakt	VARCHAR	Kontaktne podatke prijevoznika
radnoVrijeme	TIME	Radno vrijeme u kojem su prijevoznici raspoloživi

VOZILO Entitet VOZILO sadrži informacije o ID-u vozila, vrsti i kapacitet prijevoznog sredstva. Prema tome, entitet VOZILO posjeduje sljedeće atribute: IDVozilo, vrsta i kapacitet. Entitet VOZILO u vezi je *Many-to-One* s entitetom PRIJEVOZNIK preko atributa IDPrijevoznik.

VOZILO		
IDVozilo	INT	Identifikacijski ključ vozila
vrsta	VARCHAR	Vrsta vozila
kapacitet	VARCHAR	Kapacitet vozila
IDPrijevoznik	INT	Identifikacijski ključ prijevoznika

KORISNIK Entitet KORISNIK sadrži informacije o ID-u korisnika, imenu, prezimenu, kontaktnim podacima i preferencijama vezanim uz veličinu i kvalitetu smještaja. Prema tome, entitet KORISNIK posjeduje sljedeće atribute: IDKorisnik, ime, prezime, kontakt i preferencije. Entitet KORISNIK u vezi je *One-to-Many* s entitetom PUTOVANJE preko atributa IDKorisnik.

KORISNIK		
IDKorisnik	INT	Identifikacijski ključ korisnika
ime	VARCHAR	Ime korisnika
prezime	VARCHAR	Prezime korisnika
kontakt	VARCHAR	Kontakt korisnika
preferencije	VARCHAR	Preferencije vezane uz veličinu i kvalitetu smještaja

PUTOVANJE Entitet PUTOVANJE sadrži informacije o ID-u putovanja, vremenu i smjeru putovanja. Prema tome, entitet PUTOVANJE posjeduje sljedeće

atribute: IDPutovanje, vrijeme i smjer. Entitet PUTOVANJE u vezi je *Many-to-One* s entitetom KLINIKA preko atributa IDKorisnik, u vezi *Many-to-One* s entitetom SMJEŠTAJ preko atributa IDSmještaj, u vezi *Many-to-One* s entitetom KORISNIK preko atributa IDKorisnik, u vezi *Many-to-One* s entitetom PRIJEVOZNIK preko atributa IDPrijevoznik. Atributi adresa1 i adresa2 su atributi iz kojih saznajemo adresu polaska ili dolaska u ovisnosti o smjeru koji može biti 1 ili 0. Entitet PUTOVANJE u vezi je *One-to-One* s entitetom KLINIKA preko atributa adresa, u vezi *One-to-One* s entitetom SMJEŠTAJ preko atributa adresa.

PUTOVANJE		
IDPutovanje	INT	Identifikacijski ključ putovanja
vrijeme	TIME	Vrijeme putovanja
smjer	INT	Smjer u kojem se putovanje izvodi
adresa1	VARCHAR	Adresa klinike
adresa2	VARCHAR	Adresa smještaja
IDKorisnik	INT	Identifikacijski ključ korisnika
IDKlinika	INT	Identifikacijski ključ klinike
IDPrijevoznik	INT	Identifikacijski ključ prijevoznika
IDSmještaj	INT	Identifikacijski ključ smještaja

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 Selenium¹. 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.

¹<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: 9. studenoga 2023.
- Prisustvovali: I.Prezime, I.Prezime
- Teme sastanka:
 - opis prve teme
 - opis druge teme

2. sastanak

- Datum: u ovom formatu: 9. studenoga 2023.
- 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							
Dijagram razmještaja							

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

	Ime Prezime voditelja	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime
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 stranice, u izborniku Repository, pritiskom na stavku Contributors.