



**Jesenji semestra, 2017/18**

## PROJEKTNI ZADATAK

**Dnevnik za dijabetičare**

<b>Ime:</b>	Sara Davidović
<b>BrojIndeksa:</b>	1924
<b>Predmet:</b>	Uvod u softversko inženjerstvo
<b>Šifra predmeta:</b>	SE201
<b>Profesor:</b>	Prof. dr Dragan Domazet
<b>Asistent:</b>	Lazar Mrkela
<b>Datum:</b>	30.01.2017

## Sadržaj

<b>1. Uvod</b>	3
Svrha projekta	3
Opseg projekta	3
Opšti pregled	3
<b>2. Generalni opis sistema</b>	4
Funkcije sistema	4
Karakteristike korisnika	4
Ograničenja sistema	4
Pretpostavke i zavisnosti	4
<b>3. Specifični zahtevi</b>	5
Eksterni zahtevi interfejsa	5
Korisnički interfejs	5
Hardverski interfejs	5
Softverski interfejs	5
Komunikacioni interfejs	5
Funkcionalni zahtevi	5
Administrator se prijavljuje na sistem	6
Administrator briše korisnike	6
Korisnik može da kreira svoj korisnički nalog	7
Korisnik može da se uloguje na sistem	7
Korisnik može da se odjavi sa sistema	7
Korisnik može da izmeni svoj nalog	8
Korisnik može da vidi pregled dnevnika	8
Korisnik može da vidi HBA1C	9
Korisnik može da unese insulin	9
Korisnik može da unese glikemiju	9
<b>Klasni dijagram</b>	10
<b>4. Use Case dijagrami</b>	11
Slučajevi korišćenja - Korisnik	12
Slučajevi korišćenja - Administrator	14
<b>5. Ne-funkcionalni zahtevi</b>	15
<b>6. Implementacija dizajn paterna</b>	16

7.	Modeli analize sistema .....	19
	Sekvencijalni dijagrami za administratora .....	19
8.	Dijagram infrastructure .....	31
	Aplikacijski dijagram.....	31
9.	Arhitektura sistema.....	32
10.	Testiranje .....	34
11.	Zaključak.....	34
12.	Literatura.....	34

## 1. Uvod

U ovom projektu je opisan softver koji će pomoći mnogim ljudima sa autoimunom bolešću zvanom Dijabetes. Softver može da se koristi u bolnici, kod kuće i na pametnom uređaju. Ona pamti šta joj korisnik napiše, da li su to glikemije ili insulin nije bitno. On u svojoj bazi podataka pamti podatke i pamti ih kako bi mogao posle da prikaže korisniku ukoliko mu on traži. Ovo bi olakšalo način odlaženja kod doktora, jer bi imao svoj dzepni dnevnik koji mu je dostupan skoro uvek.

### Svrha projekta

Cilj ovog softvera je da jednom dijabetičaru omogući lakše vođenje evidencije o zdravstvenom problemu, jer je kod dijabetičara ključno za drav život da vode redovnu evidenciju (dnevnik) kako bi mogao da uvidi gde greši oko regulisanja insulina i šećera.

### Opseg projekta

- Naziv projekta je Dzepni dnevnik za dijabetičare.
- Softver treba da omogući lakšu evidenciju glukoze u šećeru i količine insulina.
- Beneficije ovog sistema jesu da pomogne korisniku da lakše vodi računa o svom zdravlju.

### Opšti pregled

Ostatak ovog dokumenta sadrži dijagrame koji su neophodni za razvoj softvera i koji olakšavaju razvojnom timu da razume strukturu projekta. Sadrži zahteve, dijagrame slučajeva korišćenja, dijagrame sekvenci i sve ostale dijagrame koji pružaju potrebne informacije za pravilan razvoj softvera.

## 2. Generalni opis sistema

Ovaj program beleži i pamti korisnika od trenutka logovanja, jer nam je neophodno da svaki korisnik ima svoj nalog kako bi se za svakog korisnika pamtili određeni podaci. Kasnije se određeni podaci sabiraju i daju rezultat koji je neophodan u našoj aplikaciji.

### Funkcije sistema

Glavne funkcije softvera su:

- Autentifikacija i autorizacija
- Sinhronizacija podataka sa bazom podataka
- Unos glikemija po datumu i vremenu
- Unos insulina, njegove vrste isto po datumu i vremenu
- Izmena podataka o korisniku
- Pregled dnevnika
- Tromesečni HBA1C

### Karakteristike korisnika

U ovu aplikaciju mogu se ulogovati obični korisnici i administratori. Administratori imaju sve osobine kao i običan korisnik osim što pored toga oni mogu i da brišu obične korisnike.

### Ograničenja sistema

- Korisnik mora biti na internet konekciji
- Po nekom ne rečenom načelu, korisnik ne bi trebao da unosi lažne podatke

### Pretpostavke i zavisnosti

S obzirom da ovaj softver ne zavisi od ni od kakvih operativnih sistema niti od uređaja na kom je instaliran, neće imati problema u daljem funkcionisanju i nikakve promene neće uticati na njegov rad i neće stvarati nikakve greške pri radu.

### 3. Specifični zahtevi

#### Eksterni zahtevi interfejsa

##### Korisnički interfejs

Pošto je aplikacija prilično jednostavna, ni sam interfejs ne bi trebao da bude kompleksan. On je jedinstven svuda i za svakog korisnika.

Administrator:

Kod ovog korisnika otvara se obična forma za log in jer predpostavljamo da je naš korisnik već ulogovan i ima veća prava nego drugi. Ovaj korisnik zatim ide na početnu stranu gde ima meni bar i selektuje ono što želi da učini. Ima mogućnost da selektuje da unese glikemiju, insulin, pregled, hba1c, podesavanja za korisnika, obriše korisnika ili da se odjavi.

Obični korisnik:

ima skoro sve kao i administrator, samo što se njemu umanjuje pravo da obriše korisnika.

Što se vizualnog izgleda tiče, uglavnom su kao i kod administratora sve obične forme za unos i list view za prikaz.

##### Hardverski interfejs

Osim miša, tastature (ili kod touch screen uređaja prsti), nije neophodno ništa drugo.

##### Softverski interfejs

Ne zahteva nikakve dodatne interfejse.

##### Komunikacioni interfejs

Samo vezu sa internetom.

##### Funkcionalni zahtevi

Funkcionalni zahtevi i korisnika i administratora(Slika 1)

4.	<b>Ffunkcionalni zahtevi</b>	REQ_0013	Undefined	0	Undefined	Draft
4.1	<b>Korisnik može da kreira svoj korisnički nalog.</b> Svrha ovog sistema je da omogući korisnicima da kreiraju svoj korisnički nalog. Korisnik popunjava formu za kreiranje naloga. Podaci o kreiranom nalogu se čuvaju u bazi podataka. Korisnik je kreirao nalog sa kojim može pristupiti sistemu.	REQ_0014	Undefined		Undefined	Draft
4.2	<b>Korisnik može da se uloguje na sistem.</b> Svrha ovog sistema je da omogući korisnicima da se prijave na sistem. Korisnik popunjava formu za pristupanje sistemu. Procesom verifikacije se potvrđuju uneti podaci i da li oni postoje u bazi podataka. Korisnik je pristupio sistemu.	REQ_0015	Undefined		Undefined	Draft
4.3	<b>Korisnik može da se odjavi sa sistema.</b> Svrha ovog zahteva je da omogući korisnicima da se odjave sa sistema. Korisnik se klikom na dugme "Odjavi se" odjavljuje sa sistema. Korisnik je odjavljen sa sistema.	REQ_0016	Undefined		Undefined	Draft
4.4	<b>Korisnik može da izmeni svoj nalog</b> Svrha ovog sistema je da omogući korisnicima da izmene svoje naloge (svoje podatke=, ali ne i od drugih korisnika Korisnik klikom na dugme "Podešavanja" dobija formu sa svojim podacima i informacijama. Svaka izmena se čuva u bazi podataka. Korisnik je uspešno editovao svoje podatke.	REQ_0018	Undefined		Undefined	Draft
4.5	<b>Administrator može da obriše korisnike.</b> Svrha ovog sistema je da omogući administratorima da brisu sve korisnike. Administrator klikom na dugme "Delete" brise određene korisnike. Korisnik se brise iz baze podataka i više nije vidljiva na sistemu. Administrator je uspešno obrisao korisnika.	REQ_0019	Undefined		Undefined	Draft
4.6	<b>Korisnik može da vidi pregled dnevnika</b> Svrha ovog sistema je da omogući korisniku da vidi podatke koje je nekad uneo, kako bi ovaj imao uvid u svoje trenutno stanje. View list mu to omogućava. Korisnik je uspešno pregledao svoje podatke.	REQ_0020	Undefined		Undefined	Draft
4.7	<b>Korisnik može da vidi HBA1C</b> Svrha ovog sistema je da izračuna zlatnu sredinu glikemija za zadnja 3 meseca. HBA1C je od velike koristi korisniku i zahvaljujući njemu on može da zna da li vodi računa o svom zdravlju kako treba ili ne. Korisnik je video hba1c.	REQ_0021	Undefined		Undefined	Draft
4.8	<b>Korisnik može da unese insulin</b> Svrha ovog sistema je a omogući korisniku da unese podatke o insulinu koji je dodao. Dobija formu gde bira da li bira insulin ili glikemiju. Bira insulin. Nakon toga bira datum i vreme u kojem je dodao insulin. Bira i tip insulina, a zatim ga dodaje kao stavku i onda čuva. Korisnik je uspešno sačuvao podatke o dodavanju insulina.	REQ_0022	Undefined		Undefined	Draft
4.9	<b>Korisnik može da unese glikemiju</b> Svrha ovog sistema je da omogući korisniku da unese podatke o glikemiji. U istoj formi kao za insulin on bira da unese glikemiju. Upisuje vreme i datum i vrednost glikemije. Dodaje kao stavku, a zatim i je i pamti. Glikemija je sačuvana.	REQ_0023	Undefined		Undefined	Draft

Slika 1-Funkcionalni zahtevi korisnika i administratora

## Administrator se prijavljuje na sistem

Uvod

Administrator traži pristup sistemu.

Ulaz

Administrator dobija formu za login.

Proces

Sistem proverava unete podatke.

Izlaz

Administrator dobija pristup sistemu.

Izuzetak

Ako administrator nije uneo dobre podatke, pojaviće se poruka sa greškom.

## Administrator briše korisnike

Uvod

Administrator traži spisak korisnika.

Ulaz

Administrator dobija spisak korisnika i bira kojeg želi da obriše.

Proces

Sistem pretražuje bazu i briše korisnika.

Izlaz

Administrator dobija poruku o uspešnom brisanju korisnika.

Izuzetak

Ako korisnik ne postoji u bazi, pojaviće se poruka sa greškom.

### **Korisnik može da kreira svoj korisnički nalog**

Uvod

Svrha ovog sistema je da omogući korisnicima da kreiraju svoj korisnički nalog.

Ulaz

Korisnik popunjava formu za kreiranje naloga.

Proces

Podaci o kreiranom nalogu se čuvaju u bazi podataka.

Izlaz

Korisnik je kreirao nalog sa kojim može pristupiti sistemu.

Izuzetak

Ukoliko korisnik postoji pojaviće se poruka sa greškom.

### **Korisnik može da se uloguje na sistem**

Uvod

Svrha ovog sistema je da omogući korisnicima da se prijave na sistem.

Ulaz

Korisnik popunjava formu za pristupanje sistemu.

Proces

Procesom verifikacije se potvrđuju uneti podaci i da li oni postoje u bazi podataka.

Izlaz

Korisnik je pristupio sistemu..

Izuzetak

Ukoliko korisnik pogrešno unese podatke, pojaviće se poruka o tome.

### **Korisnik može da se odjavi sa sistema.**

Uvod

Svrha ovog zahteva je da omogući korisnicima da se odjave sa sistema.

Ulaz

Korisnik se klikom na dugme “Odjavi se” odjavljuje sa sistema

Proces

Procesom odjavljivanja, prekida se sesija korisnika.

Izlaz

Korisnik je odjavljen sa sistema

Izuzetak

Korisnik možda nije na internetu ili je došlo do sistemske greške o kojoj će korisnik biti obavešten.

**Korisnik može da izmeni svoj nalog**

## Uvod

Svrha ovog sistema je da omogući korisnicima da izmene svoje naloge (svoje podatke), ali ne i od drugih korisnika.

## Ulaz

Korisnik klikom na dugme “Podešavanja” dobija formu sa svojim podacima i informacijama..

## Proces

Svaka izmena se cuva u bazi podataka.

## Izlaz

Korisnik je uspesno editovao svoje podatke.

## Izuzetak

Ukoliko unese pogrešno podatke, pojaviće se poruka o tome.

**Korisnik može da vidi pregled dnevnika**

## Uvod

Svrha ovog sistema je da omogući korisniku da vidi podatke koje je nekad uneo, kako bi ovaj imao uvid u svoje trenutno stanje

## Ulaz

View list mu to omogućava.

## Proces

Obrada zahteva.

## Izlaz

Korisnik je uspešno pregledao svoje podatke

## Izuzetak

Ukoliko je došlo do diskonekcije sa bazom.



### Korisnik može da vidi HBA1C

#### Uvod

Svrha ovog sistema je da izračuna zlatnu sredinu glikemija za zadnja 3 meseca. HBA1C je od velike koristi korisniku i zahvaljujući njemu on može da zna da li vodi računa o svom zdravlju kako treba ili ne.

#### Ulaz

Nema nikakvih podataka za unos.

#### Proces

Upit se obrađuje.

#### Izlaz

Korisnik je video hba1c.

#### Izuzetak

Ukoliko ne postoji nikakva vrednost u bazi, korisnik dobija poruku.

### Korisnik može da unese insulin

#### Uvod

Svrha ovog sistema je a omogućiti korisniku da unese podatke o insulinu koji je dodao.

#### Ulaz

Dobija formu gde bira da li bira insulin ili glikemiju. Bira insulin. Nakon toga bira datum i vreme u kojem je dodao insulin. Bira i tip insulina, a zatim ga dodaje kao stavku-

#### Proces

Nakon što je uradio sve navedeno, upit se šalje i podaci se čuvaju.

#### Izlaz

Dobija se poruka o sačuvanim podacima.

#### Izuzetak

Ukoliko su pogrešno uneti podaci, prikazuje se poruka.

### Korisnik može da unese glikemiju

#### Uvod

Svrha ovog sistema je da omogućiti korisniku da unese podatke o glikemiji.

#### Ulaz

U istoj formi kao za insulin on bira da unese glikemiju. Upisuje vreme i datum i vrednost glikemije. . Dodaje kao stavku.

#### Proces

Šalje se upit i podaci se čuvaju u bazi.

#### Izlaz

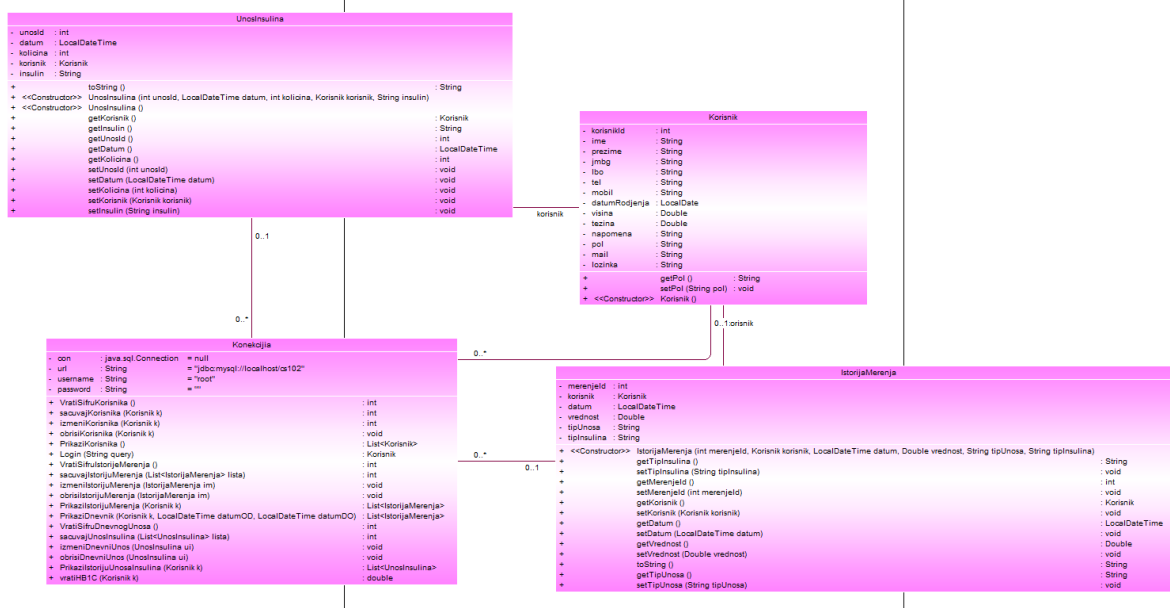
Prikazuje se poruka da li su podaci sačuvani.

#### Izuzetak

Ukoliko su pogrešno uneti podaci, prikazuje se poruka.

## Klasni dijagram

Dijagram klasa (deo UML-a) je vrsta strukturnog dijagrama u softverskom inženjeringu, koji opisuje strukturu sastava objašnjavajući klase unutar sastava, njihove atribute i odnose.

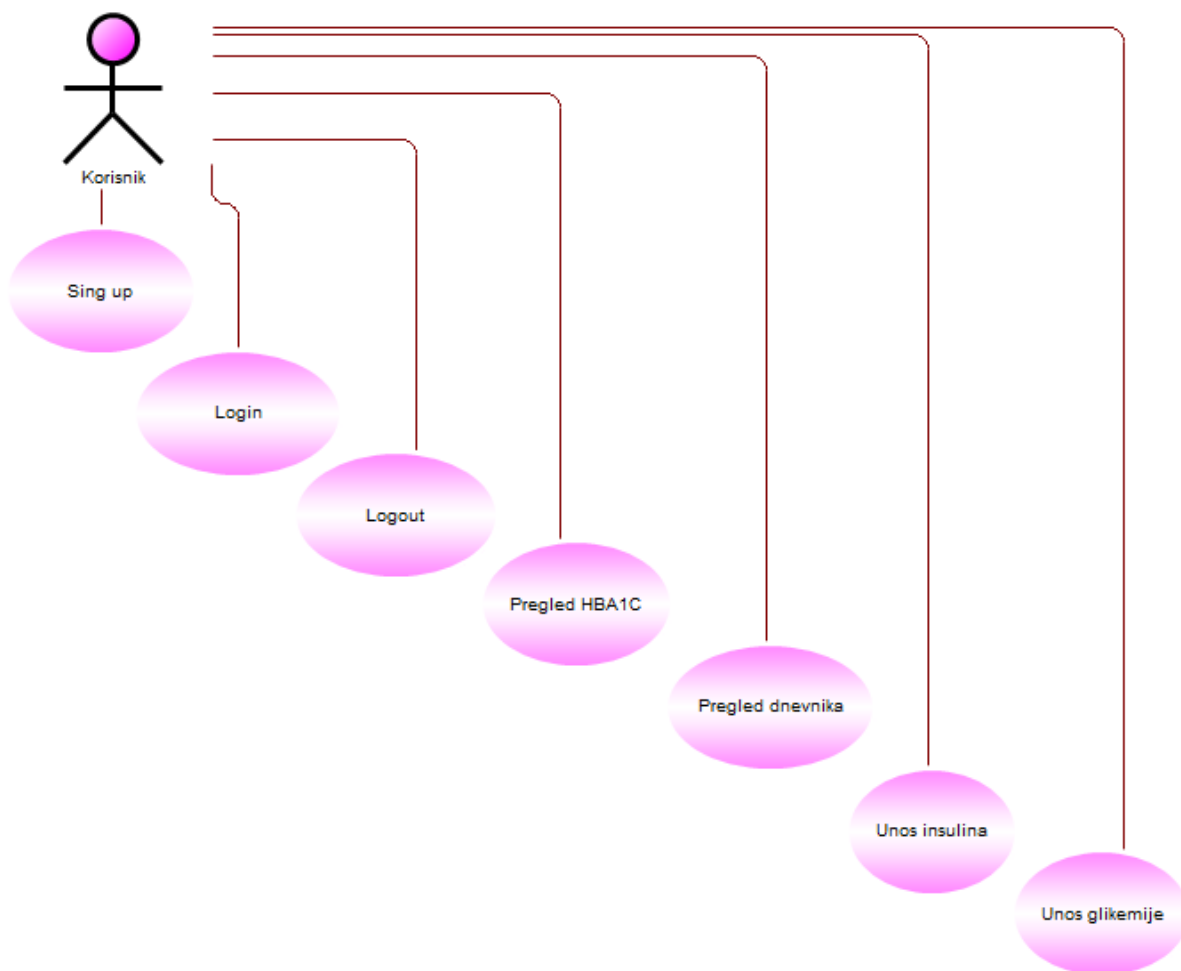


Slika 2-Klasni diagram

## 4. Use Case dijagrami

### Use Case korisnika

Na Slici 2 možemo videti use case diagram koji nam ilustruje šta korisnik može da radi na sistemu.



Slika 3-Use case -Korisnik

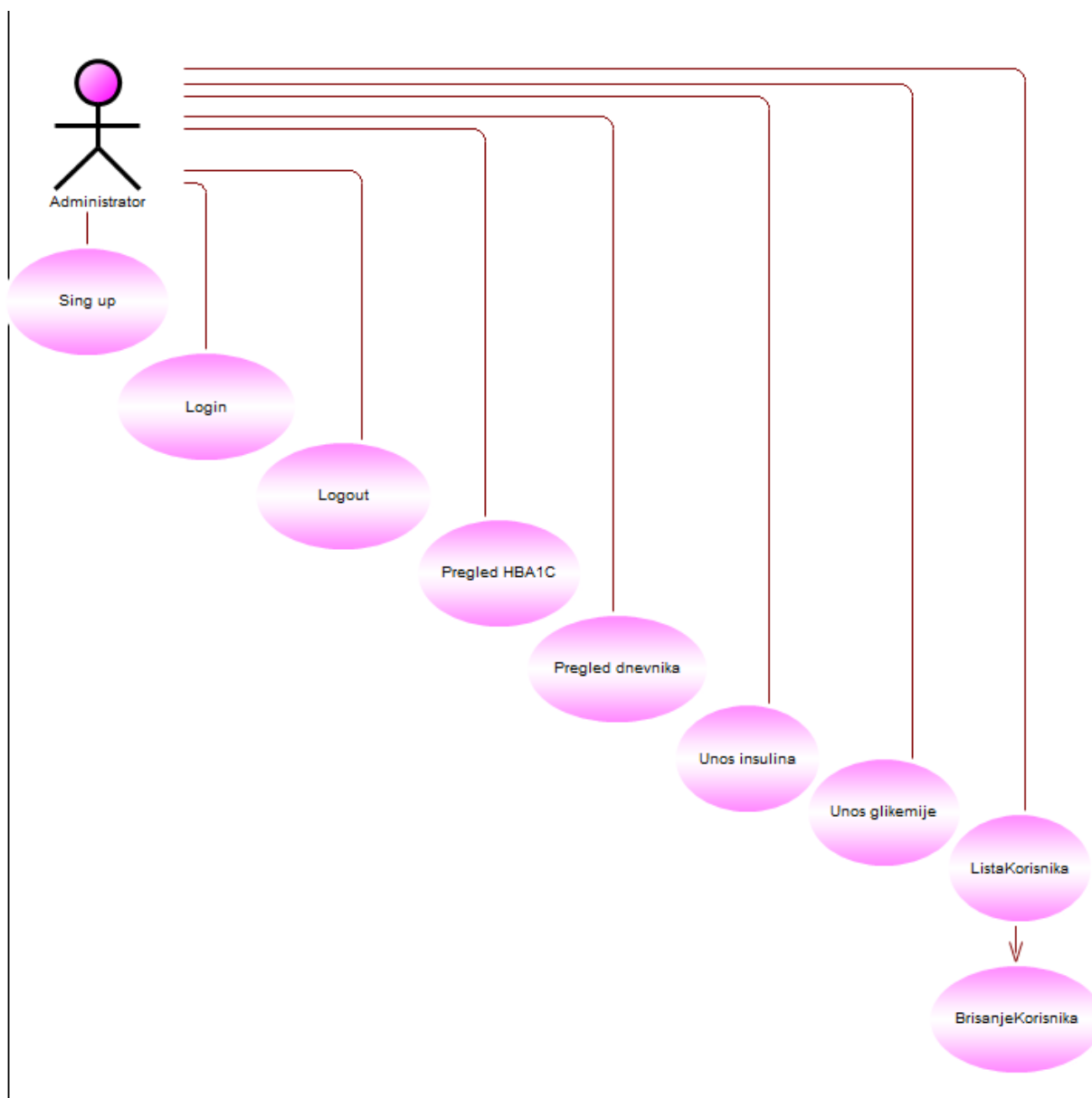
### Slučajevi korišćenja - Korisnik

Use Case	Preduslovi	Koraci	Izuzeci	Postuslovi
Login	Korisnik mora da ima kreiran nalog	Korisnik unosi korisničke podatke	Nisu uneti pravilni podaci	Korisnik ima pristup sistemu
Unos prisustva	Korisnik mora da bude ulogovan na svoj nalog	Korisnik pristupa svom nalogu i unosi prisustva	Sistem trenutno ne može da pruži opciju unosa prisustva	Prisustvo je uneseno
Unos glikemije	Mora postojati forma	Korisnik unosi podatke u formu za glikemiju	Podaci su uneti pogrešno	Podaci su uspešno sačuvani
Unos insulina	Mora postojati forma	Korisnik unosi podatke u formu za insulin	Podaci su uneti pogrešno	Podaci su uspešno sačuvani
Pregled liste	Korisnik mora imati unešenu makar jednu vrednost u bazi	Korisnik pretražuje podatke po datumu i vremenu	Korisnik je izabrao kriterijum pretrage za koji u bazi ne postoje podaci	Korisnik je dobio uvid u listu
Pregled HBA1C	Korisnik mora imati unešene glikemije za predhodna 3 meseca	Korisnik bira opciju klikom u meni baru na HBA1C	Korisnik nema unešene glikemije za određen period	Korisnik jasno vidi rezultat

Tabela 1-Slučajevi korišćenja za korisnika

### Use Case administratora

Na Slici 3 možemo videti use case diagram koji nam ilustruje šta administrator može da radi na sistemu.



Slika 4-Use Case diagram-Administrator

### Slučajevi korišćenja - Administrator

Use Case	Preduslovi	Koraci	Izuzeci	Postuslovi
Login	Admin mora da ima kreiran nalog	Admin unosi korisničke podatke	Nisu uneti pravilni podaci	Admin ima pristup sistemu
Unos prisustva	Admin mora da bude ulogovan na svoj nalog	Admin pristupa svom nalogu i unosi prisustva	Sistem trenutno ne može da pruži opciju unosa prisustva	Prisustvo je uneseno
Unos glikemije	Mora postojati forma	Admin unosi podatke u formu za glikemiju	Podaci su uneti pogrešno	Podaci su uspešno sačuvani
Unos insulina	Mora postojati forma	Admin unosi podatke u formu za insulin	Podaci su uneti pogrešno	Podaci su uspešno sačuvani
Pregled liste	Admin mora imati unešenu makar jednu vrednost u bazi	Admin pretražuje podatke po datumu i vremenu	Admin je izabrao kriterijum pretrage za koji u bazi ne postoje podaci	Admin je dobio uvid u listu
Pregled HBA1C	Admin mora imati unešene glikemije za predhodna 3 meseca	Admin bira opciju klikom u meni baru na HBA1C	Admin nema unešene glikemije za određen period	Admin jasno vidi rezultat
Lista korisnika	Korisnik mora da postoji u bazi	Admin pretražuje korisnika po imenu i prezimenu	Admin je izabrao kriterijum pretrage za koji u bazi ne postoje korisnici	Admin je dobio uvid u listu korisnika
Brisanje korisnika	Korisnik mora da postoji u bazi	Administrator nalazi u bazi korisnika kojeg želi da briše	Korisnik nije pronađen u bazi	Uspešno je obrisao korisnik

Tabela 2-Slučajevi korišćenja za administratora

## 5. Ne-funkcionalni zahtevi

Ne-funkcionalni zahtevi nam opisuju kako i šta sistem treba da radi. Ovi zahtevi su oni koje funkcionalni zahtevi ne mogu da urade.

5	3.	Nefunkcionalni zahtevi	REQ_0005	Undefined	0	Undefined	Draft
→	3.1	Administrator baze podataka upravlja korisničkim podacima	REQ_0006	Undefined		Undefined	Draft
7	3.2	Administrator baze upravlja brisanjem događajima	REQ_0007	Undefined		Undefined	Draft
8	3.3	Sistem mora biti dostupan u svakom trenutku	REQ_0008	Undefined		Undefined	Draft
9	3.4	Sistem mora da prikazuje informacije u realnom vremenu	REQ_0009	Undefined		Undefined	Draft
10	3.5	Veb interfejs mora biti interaktivan i lak za korišćenje.	REQ_0010	Undefined		Undefined	Draft
11	3.6	Veb interfejs mora da ima jednostavnu navigaciju.	REQ_0011	Undefined		Undefined	Draft
12	3.7	Samo sistem administrator ima pravo da promeni parametre sistema.	REQ_0012	Undefined		Undefined	Draft

Slika 5-Ne-funkcionalni zahtevi

### ➤ Performanse

Sistem treba da funkcioniše bez ikakvih kašnjenja.

### ➤ Pouzdanost

Sistem ne sme da otkáže kada je korisniku najpotrebniji.

### ➤ Dostupnost

Sistem mora da funkcioniše u realnom vremenu i da bude dostupan korisniku non-stop

➤ **Sigurnost**

Zaštita korisničkih podataka treba da bude prioritet. obični korisnici ne bi trebali nikad da imaju mogućnost pristupa administratorskim nalogima.

➤ **Održivost**

Baza podataka treba da bude fleksibilna. Ukoliko broj podataka i korisnika poraste, ona mora to izdržati.

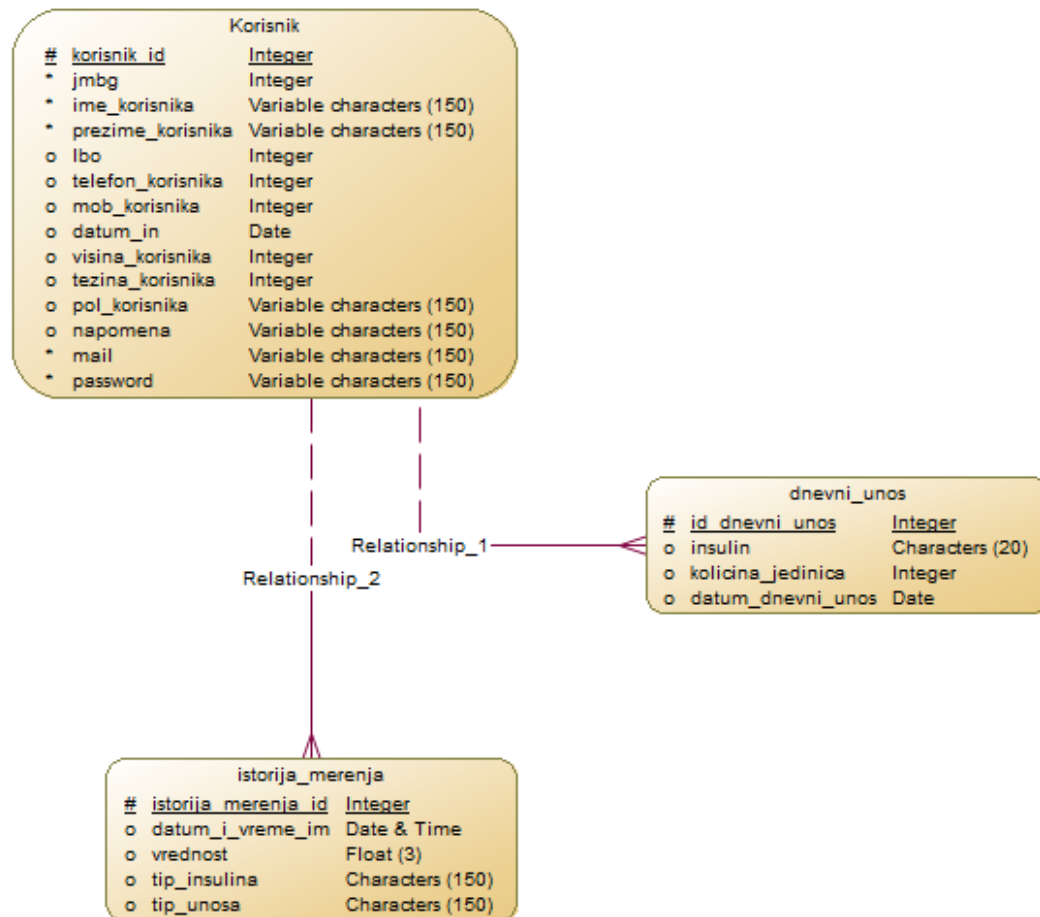
➤ **Portabilnost**

Aplikacija mora da bude prilagođena za sve vrste platformi. Od računara do tablet telefona.

## 6. Implementacija dizajn paterna

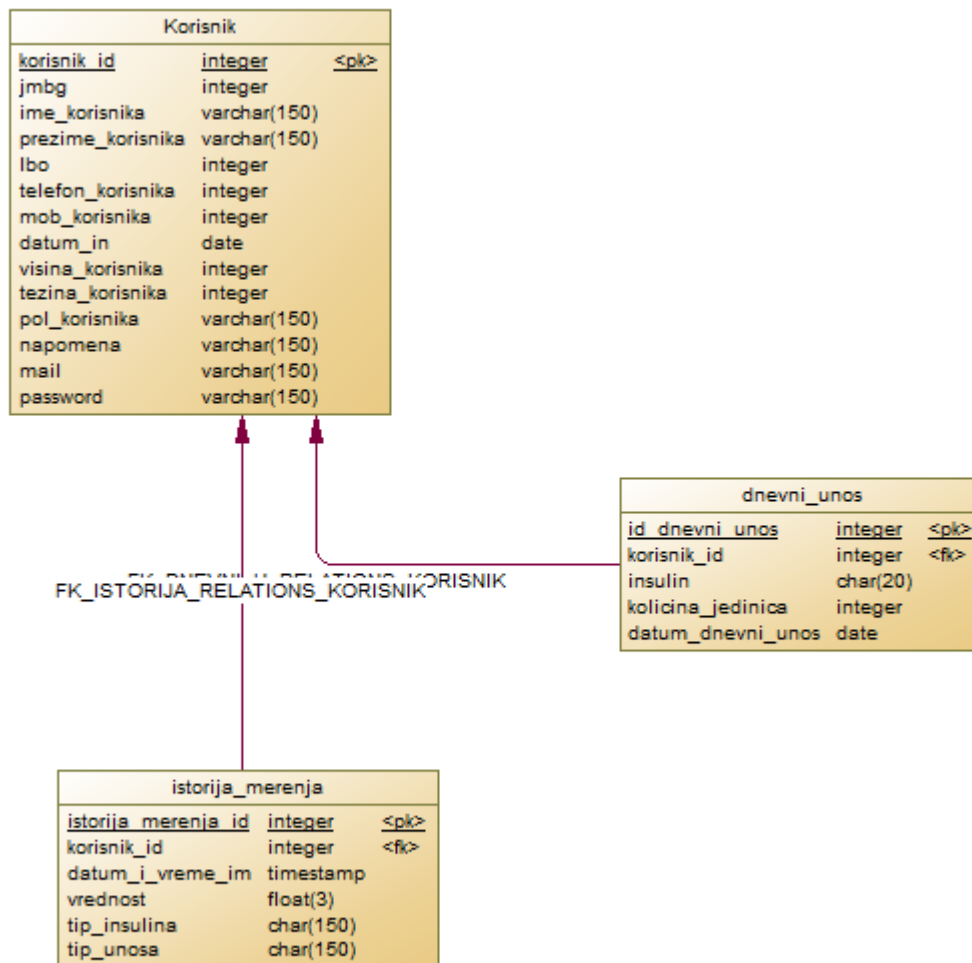
Konceptualni model baze podataka





Slika 6-Konceptualni model baze podataka

## Fizički model baze podataka

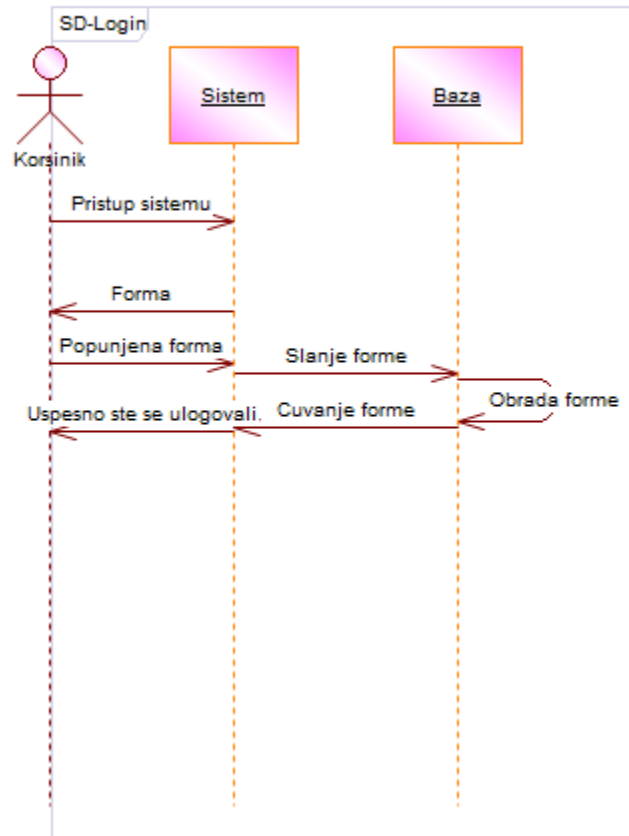


Slika 7-fizicki model baze podataka

## 7. Modeli analize sistema

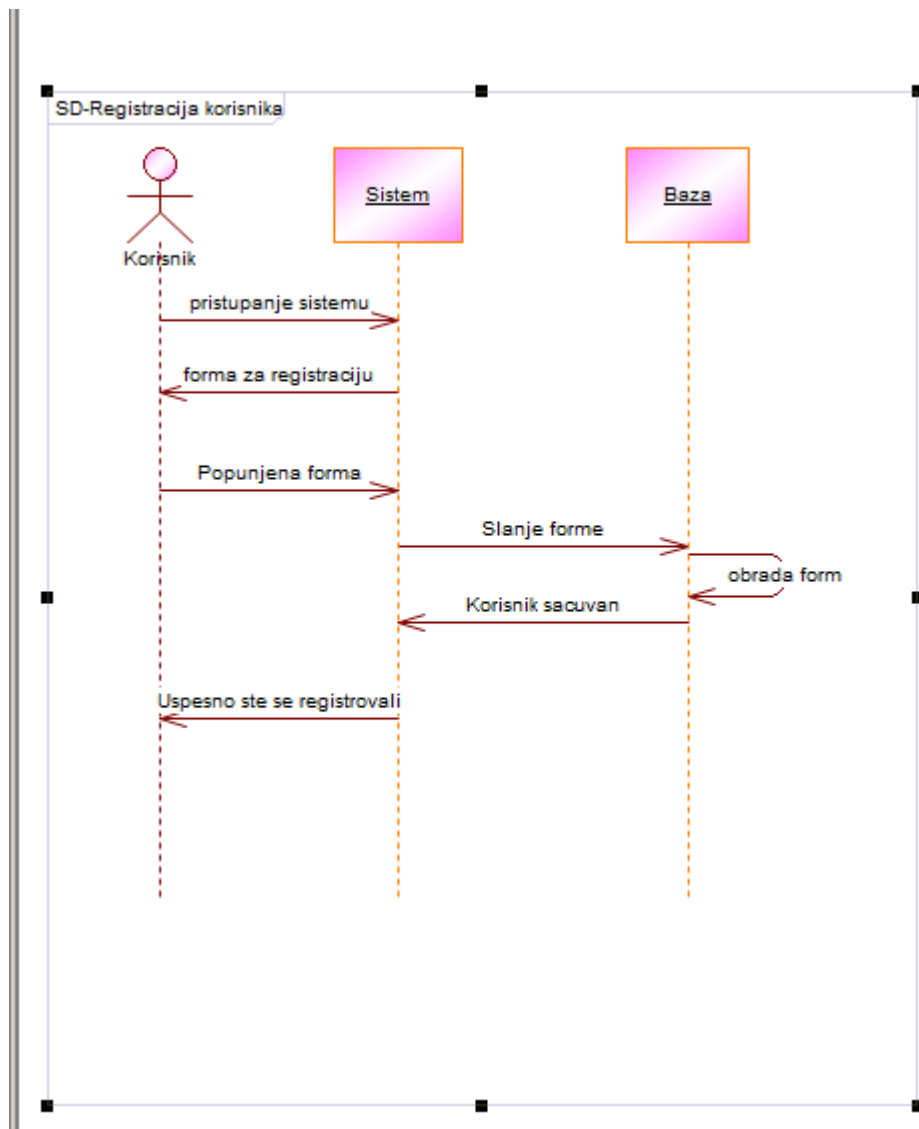
### Sekvencijalni dijagrami za administratora

Korisnik pristupa sistemu kao na Slici 8, dobija formu koju popunjava. Na slici ispod ilustrovan je sekvencijalni dijagram koji prikazuje kako se korisnik loguje na sistem.



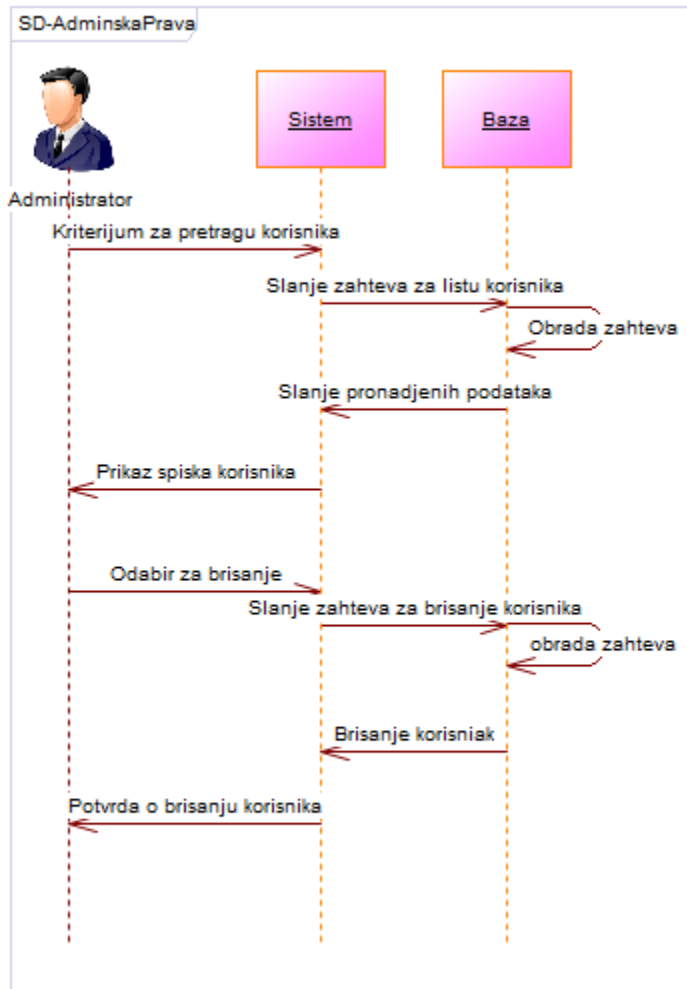
Slika 8-Sekvencijalni dijagram-login

Da bi se korisnik registrova na sistem, mora da popuni formu sličnu log formi. Slika 9



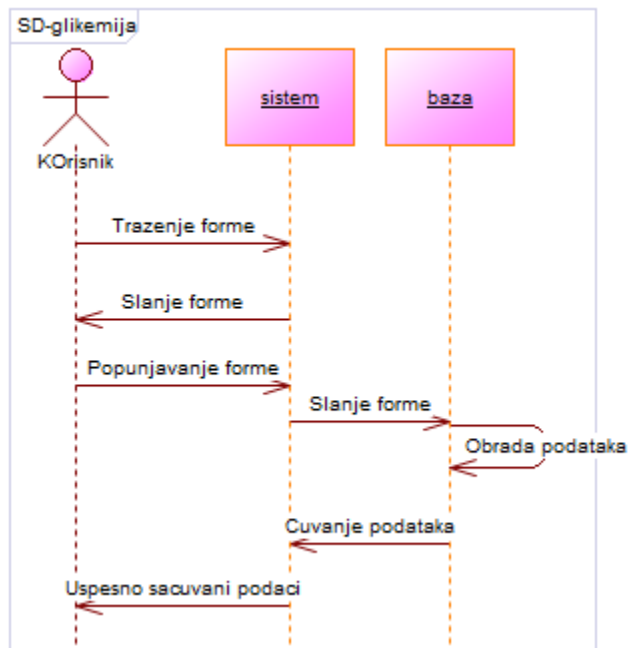
Slika 9-Sekvencijalni dijagram-Registracija

Admin može da vidi spisak korisnika i sa njega da ukloni koga on želi Slika 10.



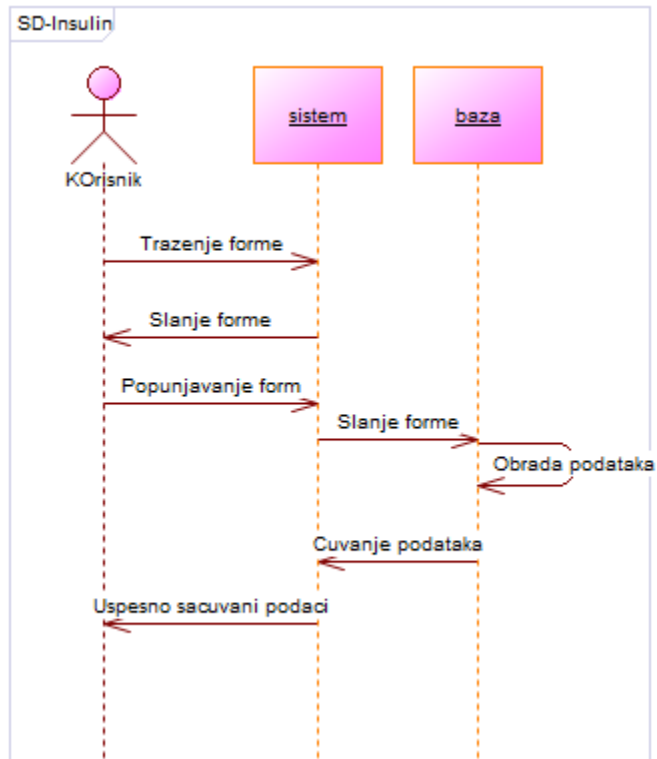
Slika 10-Sekvencijalni dijagram-AdminskaPrava

Sekvencijalni dijagram glikemija prikazuje kako se unosi glikemija Slika 11.



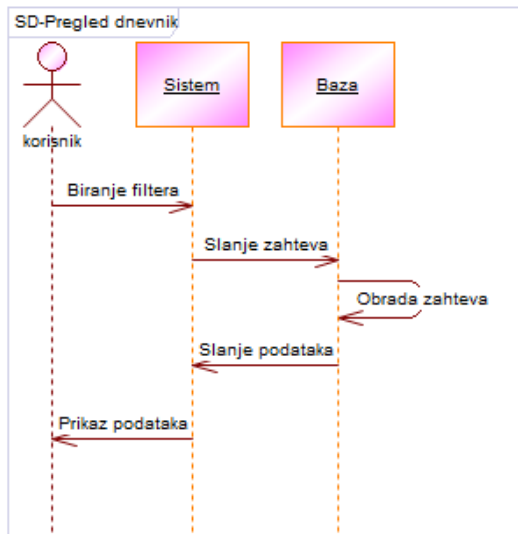
Slika 11-Sekvencijalni dijagram-glikemija

Sekvencijalni dijagram insulin prikazuje kako se unosi glikemija Slika 12.



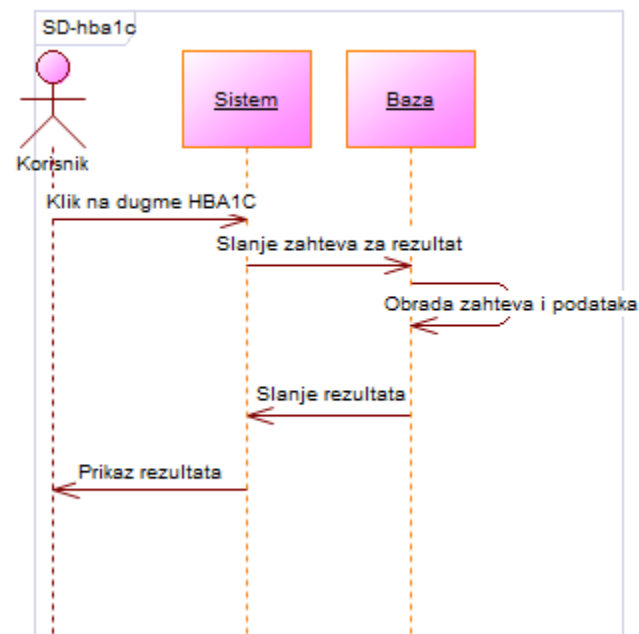
Slika 12-Sekvencijalni dijagram-insulin

Sekvencijalni dijagram dnevnik prikazuje glikemij i insulin u listi Slika 13.



Slika 13-Sekvencijalni dijagram-pregledDnevnika

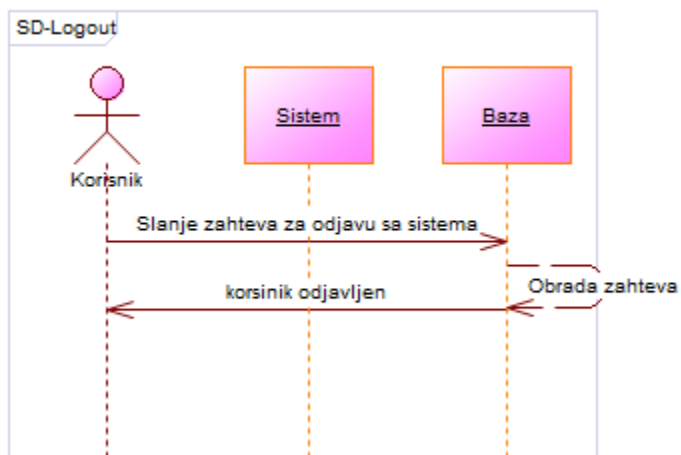
Sekvencijalni dijagram hba1c prikazuje glikemijsku vrednost za poslednja 3 meseca Slika 14.



Slika 14-Sekvencijalni dijagram-HBA1C



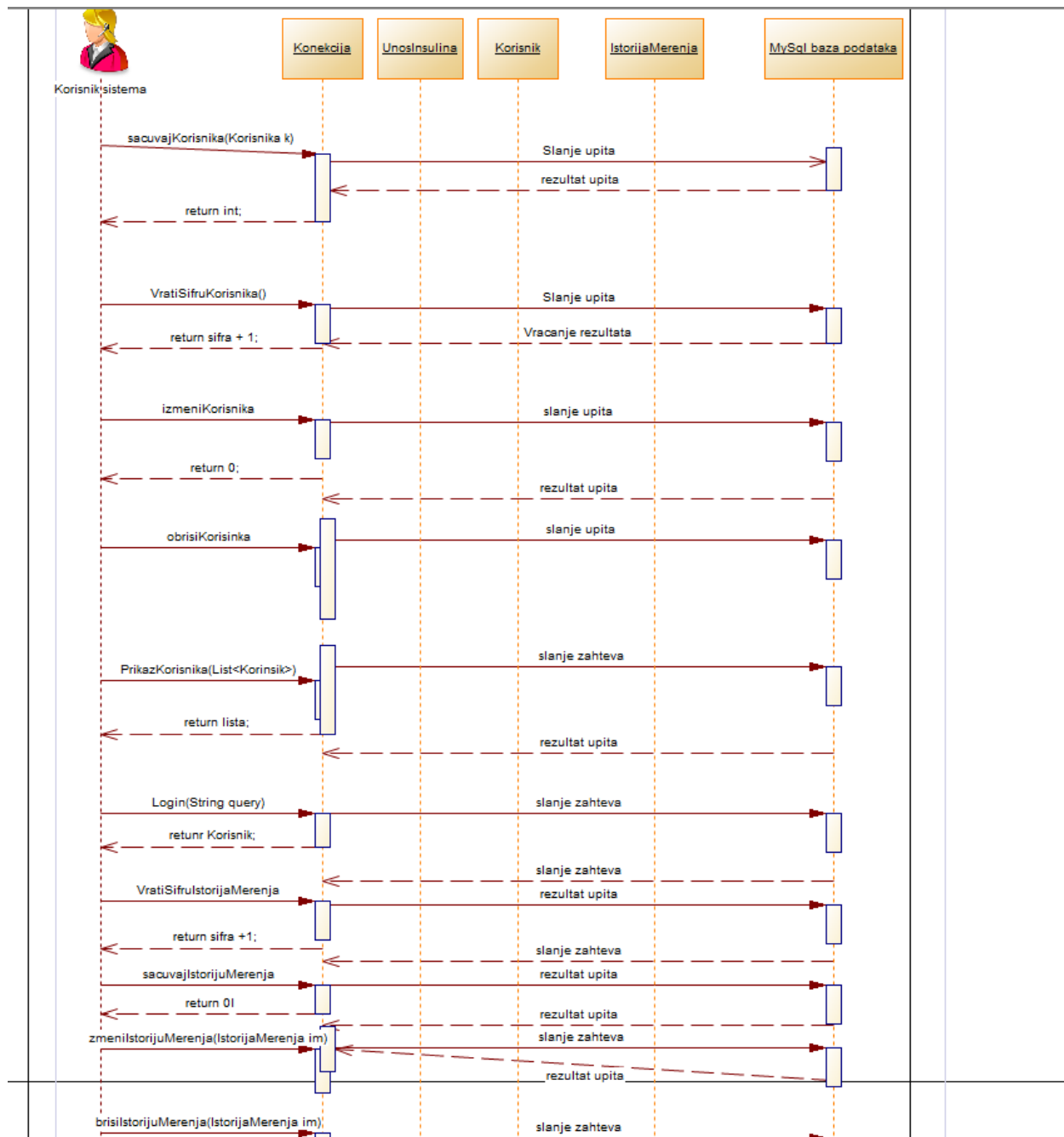
Sekvencijalni dijagram logout isključuje korisnika sa sistema Slika 14.



Slika 15-Sekvencijalni dijagram-LogOut

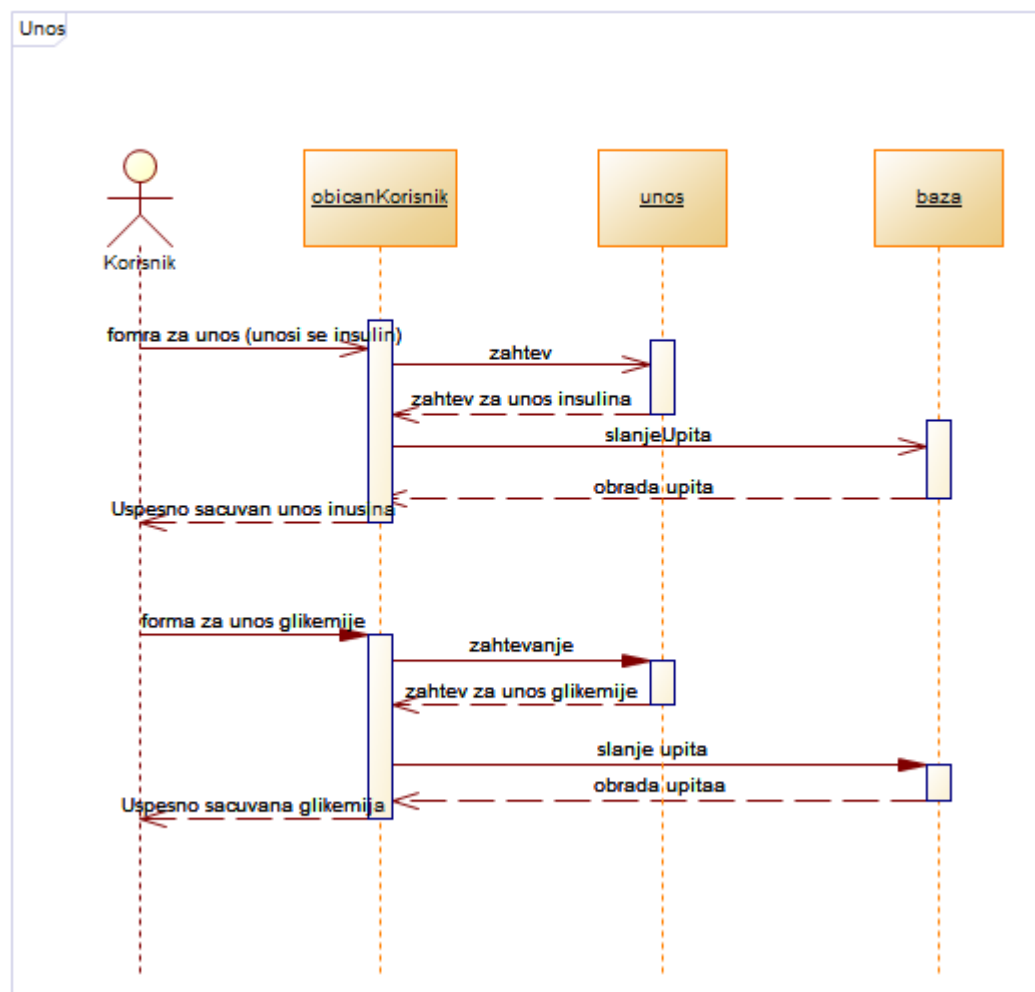
Detaljniji sekvencijalni dijagrami

Prvi sekvencijalni dijagram prikazaće logovanje korisnika – obični korisnici i administratori Slika 16.



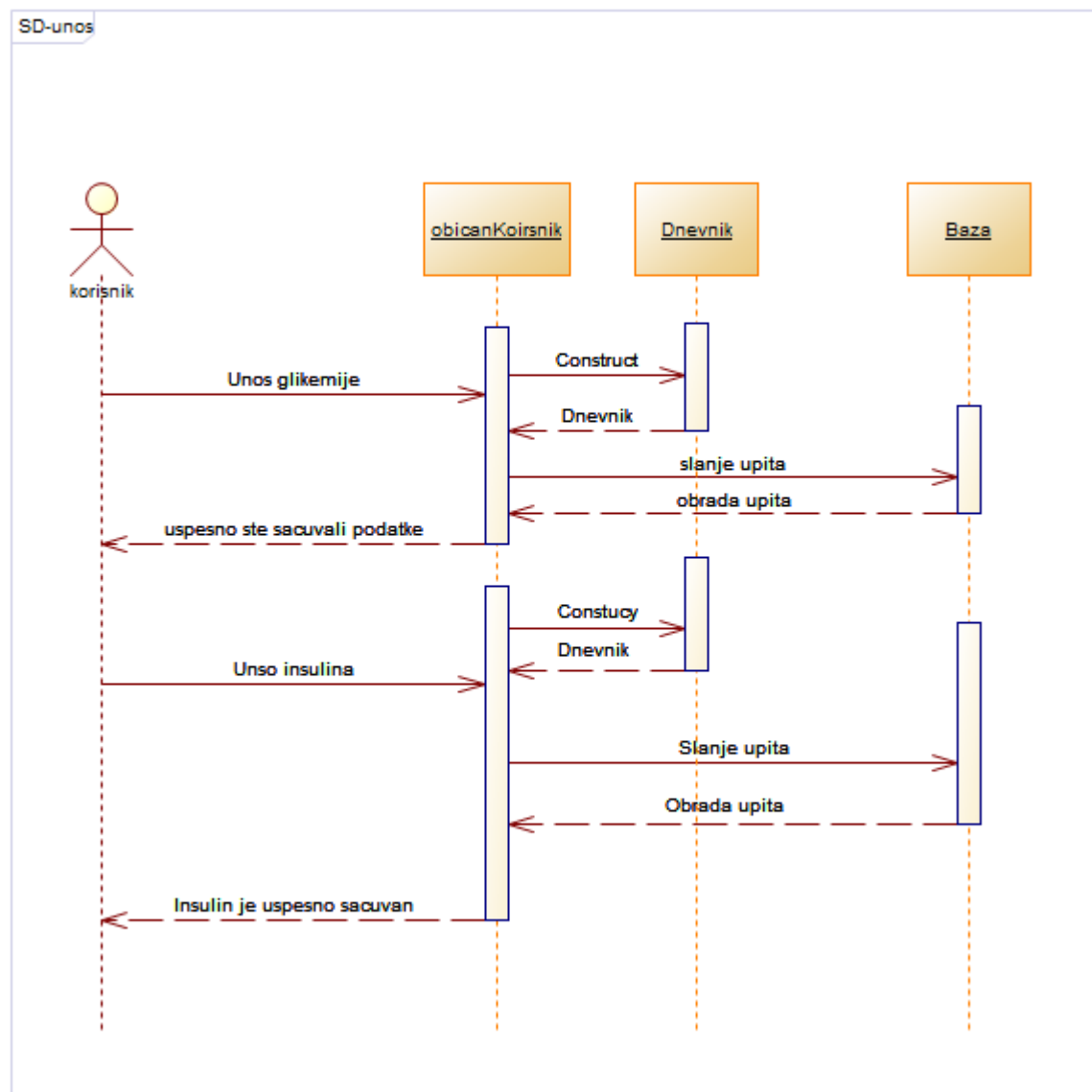
Slika 16-SD detaljniji korisnici

Sledeći detaljniji sekvencijalni dijagram je unos Slika 17,



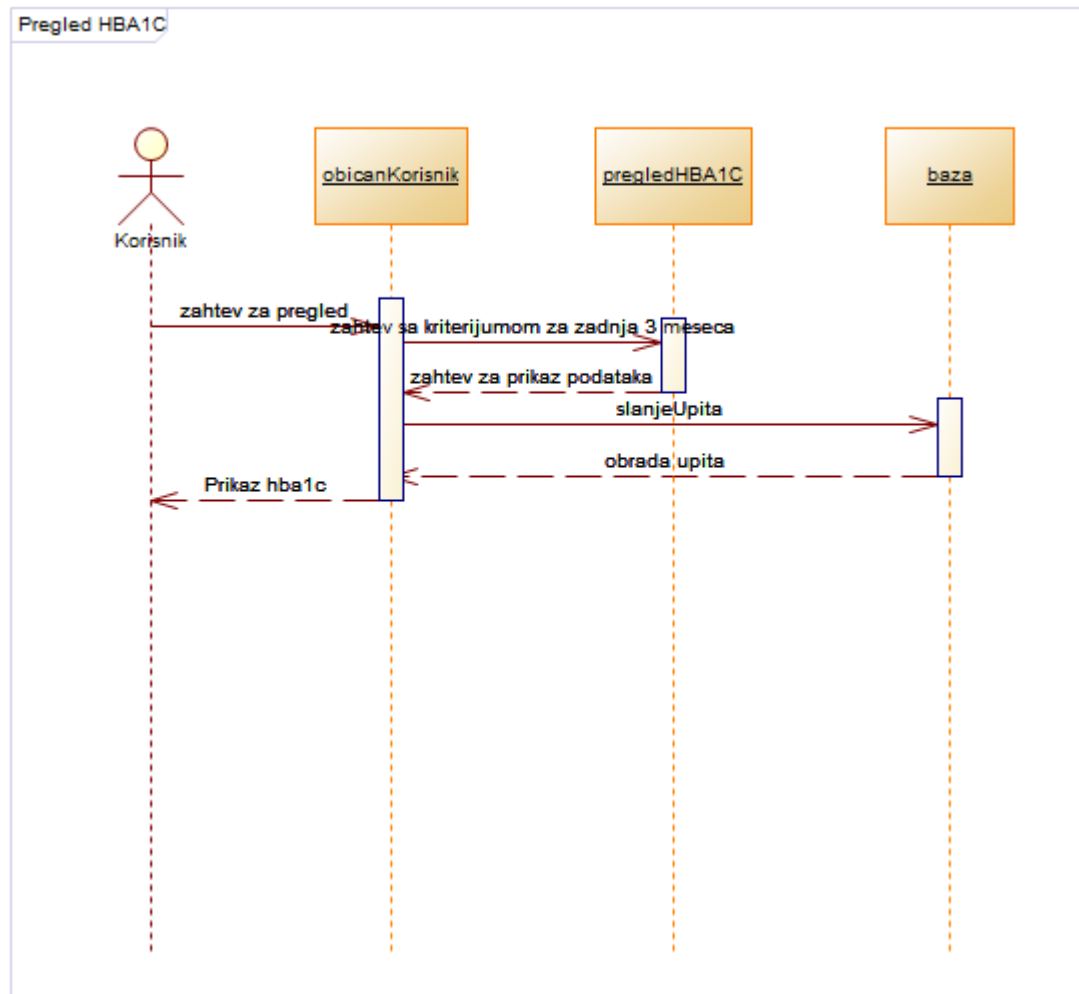
Slika 17-SD-unosi

Nakon unosa, prelazimo na prikaz dnevnika Slika 18,



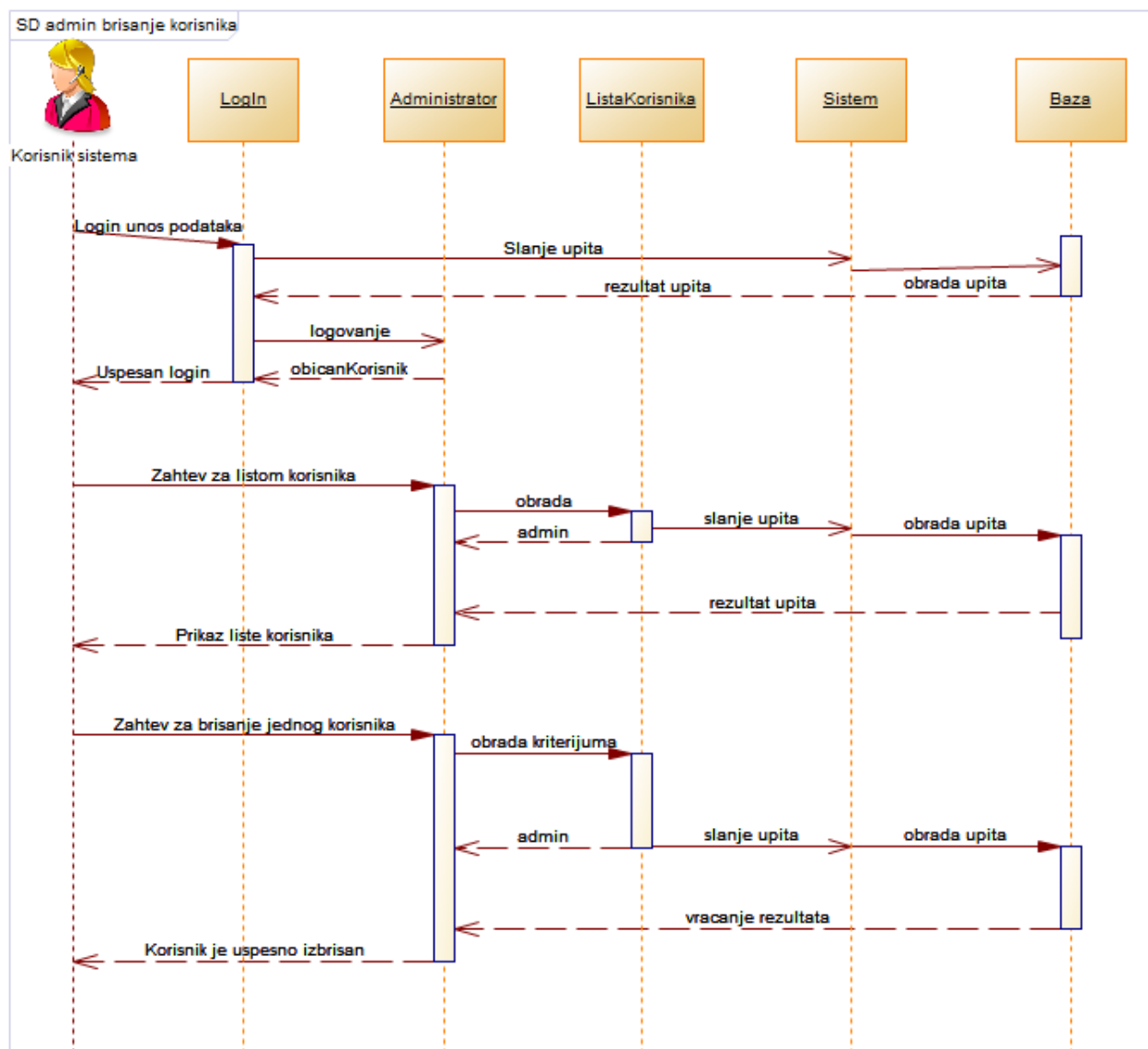
Slika 18-SD-dnevnik

Pošto imamo podatke u bazi, možemo videti i HBA1C Slika 19



Slika 19-SD-HBA1C

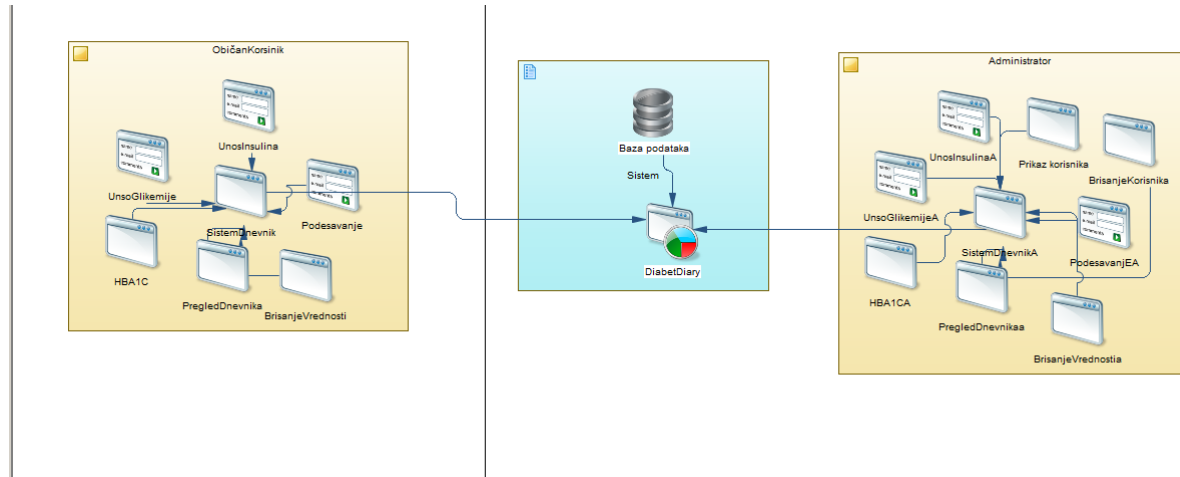
Ukoliko postoje i obični korisnici pored adminisitratora, administrator ih može ukloniti Slika 20



Slika 20-SD-admin brise korisnike

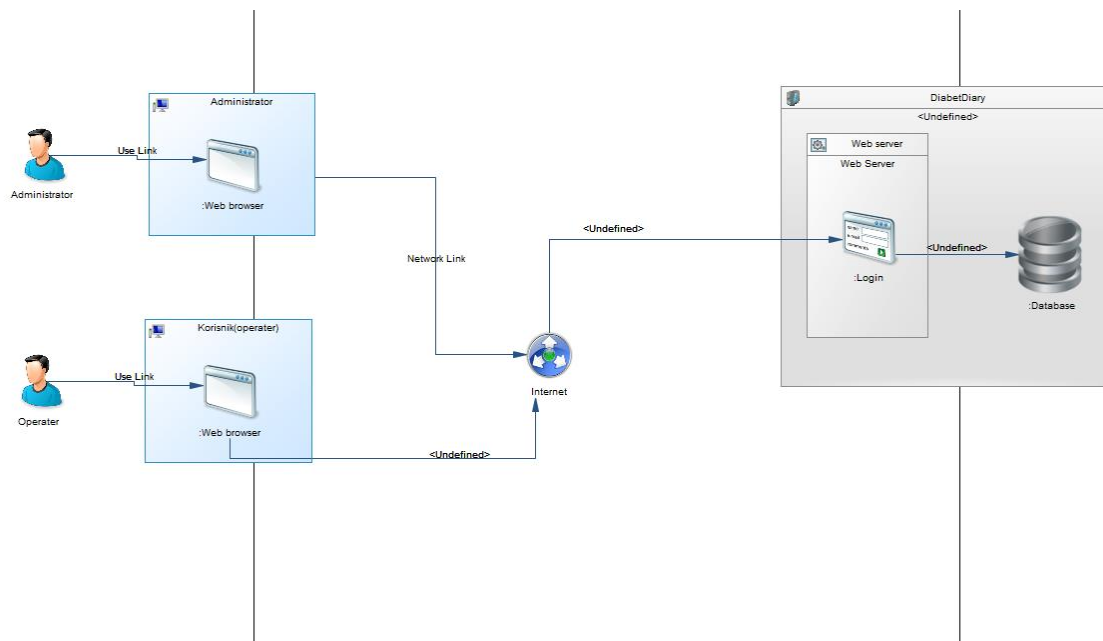
## 8. Dijagram infrastructure

### Aplikacijski dijagram



Slika 21-Aplikacijska dijagram

### Infrastruktura



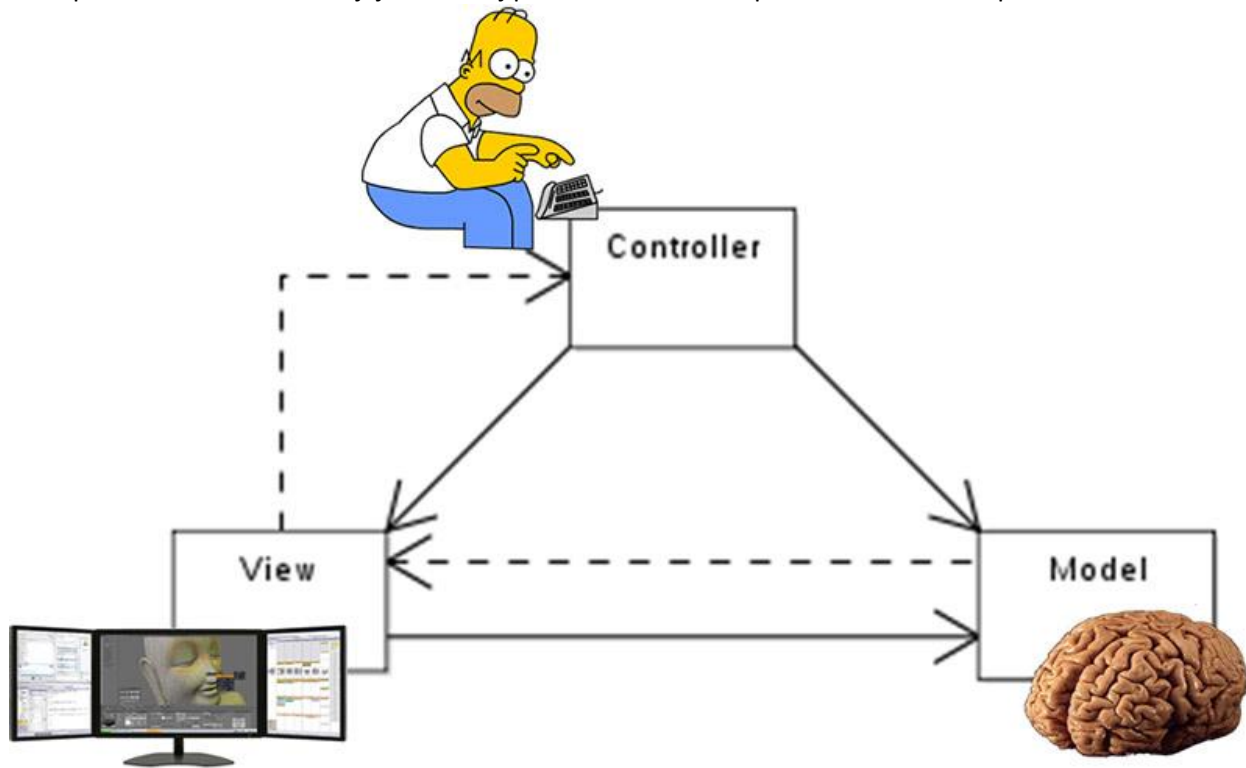
Slika 22-Infrastrukturni dijagram

## 9. Arhitektura sistema

“Diabet diary” koristi dve arhitekture:

- MVC arhitekturu Slika 23

MVC podrazumeva šablon koji je u širokoj primeni u svetu. On podrazumeva 3 komponente:

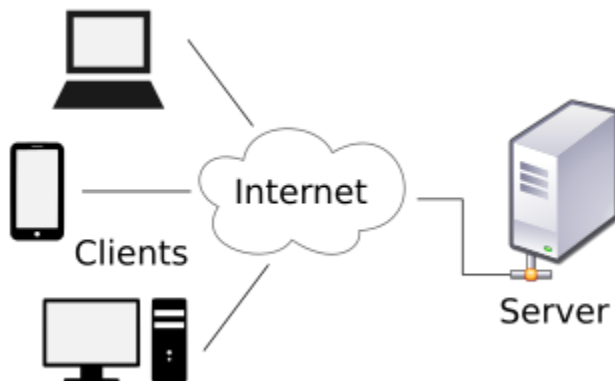


Slika 23-Prikaz MVC šablona

1. **Model** - predstavlja stanje sistema koje mogu promeniti operacije model-a. Model ne mora da zna ko su view i controller.
2. **View** - obezbeđuje korisniku interfejs pomoću koga korisnik unosi podatke i poziva odgovarajuće operacije koje treba da se izvrše nad model-om. View prikazuje korisniku stanje model-a.
3. **Controller** - osluškuje i prihvata zahtev od klijenta za izvršenje operacije. Nakon toga poziva operaciju koja je definisana u model-u, i ukoliko model promeni stanje, obaveštava view o promeni stanja.



➤ Klijent – Server arhitekturu Slika 24



*Slika 24-Klijent-server model*

U ovom sistemu ovo podrazumeva sistem za obradu podataka po klijent/server modelu i mogu se uočiti tri klase komponenti **server, klijent i mreža**. Sistem koristi server gde čuva podatke o korisnicima. Osnova klijent-server arhitekture jeste slanje zahteva i dobijanje odgovora. Server bi trebao da centralizovano obezbedi integritet podataka za aplikaciju, da obezbedi podatke, upravlja bazom podataka... Klijent-aplikacija upravlja interfejsom za korisnika i izvršava deo logike aplikacije. Mreža – računarska mreža prenosi podatke od servera ka klijentu i obrnuto.

## 10. Testiranje

e

```

6 package Baza;
7
8 import Klase.IstorijaMerenja;
9 import Klase.Korisnik;
10 import Klase.UnosInsulina;
11 import java.time.LocalDate;
12 import java.time.LocalDateTime;
13 import java.util.List;
14 import org.junit.After;
15 import org.junit.AfterClass;
16 import org.junit.Before;
17 import org.junit.BeforeClass;
18 import org.junit.Test;
19 import static org.junit.Assert.*;
20
21 /**
22  *
23  * @author Saryca
24  */
25 public class KonekcijaTest {
26
27     /**
28      * Test of vratiHB1C method, of class Konekcija.
29      */
30     @Test
31     public void testVratiHB1C() {
32         System.out.println("vratiHB1C");
33         Korisnik k = new Korisnik(2);
34         //ukoliko se promeni vrednost u bazi podataka i ovde se mora zameniti ocekivani rezultat
35         double expectedResult = 8.5;
36         double result = Konekcija.vratiHB1C(k);
37         assertEquals(expResult, result, 0.1);
38         // TODO review the generated test code and remove the default call to fail.
39         //fail("The test case is a prototype.");
40     }
41
42 }

```

Slika- 1- Test klasa koja testira HBA1C vrednost.

nedostaje mi testiranje.

## 11. Zaključak

Projekcija ove aplikacije sprovedena je uspešno kroz sve faze razvoja jednog softverskog sistema. Detaljne analize ove aplikacije su mi pomogle da razumem kompleksnost razvoja jedne aplikacije. Za izradu ovog projekta bilo je potrebno primeniti različite metodologije planski vođenog razvoja softverskog sistema. Sistem ima mogućnost da se izradi u budućnosti i da se koristi. Autoru ove aplikacije je pomoglo da se usavrši u izradi projekta jedne aplikacije. Zahvaljujući ovom projektu, razumela sam prdmet SE201-Uvod u softversko inženjerstvo.

## 12. Literatura

1. prof. dr Domazet, Dragan, and MSc Lazar Mrkela. Predavanja i Vežbe iz Predmeta SE201 – Uvod u Softversko Inženjerstvo. Izdanje za nastavnu godinu 2017/2018. Beograd: Univerzitet

Metropolitan, 2017. Web. LAMS Univerziteta Metropolitan. Dostupno na:  
<http://lms.metropolitan.ac.rs>. Posećeno: 27 Decembar 2017.

2. Iz Vikipedije, slobodne enciklopedije, "MVC arhitektura," [Online]. Available:  
[https://sr.wikipedia.org/wiki/MVC\\_arhitektura](https://sr.wikipedia.org/wiki/MVC_arhitektura). Accessed: Jan. 31.10.2017