#### LOGO_bela_pozadina

**Jesenji semestra, 2017/18**

##### PROJEKTNI ZADATAK

Dnevnik za dijabetičare

**Ime:** Sara Davidović

**BrojIndeksa:** 1924

**Predmet:** Uvod u softversko inženjerstvo

**Šifra predmeta:** SE201

**Profesor:** Prof. dr Dragan Domazet

**Asistent:** Lazar Mrkela

**Datum:** 30.01.2017

Sadržaj

**[1.](#_Toc510619119)****[Uvod](#_Toc510619119)** [3](#_Toc510619119)

[Svrha projekta 3](#_Toc510619120)

[Opseg projekta 3](#_Toc510619121)

[Opšti pregled 3](#_Toc510619122)

[**2.** **Generalni opis sistema** 4](#_Toc510619123)

[Funkcije sistema 4](#_Toc510619124)

[Karakteristike korisnika 4](#_Toc510619125)

[Ograničenja sistema 4](#_Toc510619126)

[Pretpostavke i zavisnosti 4](#_Toc510619127)

[**3.** **Specifični zahtevi** 5](#_Toc510619128)

[Eksterni zahtevi interfejsa 5](#_Toc510619129)

[Korisnički interfejs 5](#_Toc510619130)

[Hardverski interfejs 5](#_Toc510619131)

[Softverski interfejs 5](#_Toc510619132)

[Komunikacioni interfejs 5](#_Toc510619133)

[Funkcionalni zahtevi 5](#_Toc510619134)

[Administrator se prijavljuje na sistem 6](#_Toc510619135)

[Administrator briše korsinike 6](#_Toc510619136)

[Korisnik može da kreira svoj korisnički nalog 7](#_Toc510619137)

[Korisnik može da se uloguje na sistem 7](#_Toc510619138)

[Korisnik može da se odjavi sa sistema. 7](#_Toc510619139)

[Korisnik može da izmeni svoj nalog 8](#_Toc510619140)

[Korisnik može da vidi pregled dnevnika 8](#_Toc510619141)

[Korisnik može da vidi HBA1C 9](#_Toc510619142)

[Korisnik može da unese insulin 9](#_Toc510619143)

[Korisnik može da unese glikemiju 9](#_Toc510619144)

[**Klasni dijagram** 10](#_Toc510619145)

[**4.** **Use Case dijagrami** 11](#_Toc510619146)

[Slučajevi korišćenja - Korisnik 12](#_Toc510619147)

[Slučajevi korišćenja - Administrator 14](#_Toc510619148)

[**5.** **Ne-funkcionalni zahtevi** 15](#_Toc510619149)

[**6.** **Implementacija dizajn paterna** 16](#_Toc510619150)

[**7.** **Modeli analize sistema** 19](#_Toc510619151)

[Sekvencijalni dijagrami za administratora 19](#_Toc510619152)

[**8.** **Dijagram infrastructure** 31](#_Toc510619153)

[Aplikacijski dijagram 31](#_Toc510619154)

[**9.** **Arhitektura sistema** 32](#_Toc510619155)

[**10.** **Testiranje** 34](#_Toc510619156)

[**11.** **Zaključak** 34](#_Toc510619157)

[**12.** **Literatura** 34](#_Toc510619158)

# **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.

# 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.

# **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 skoros 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

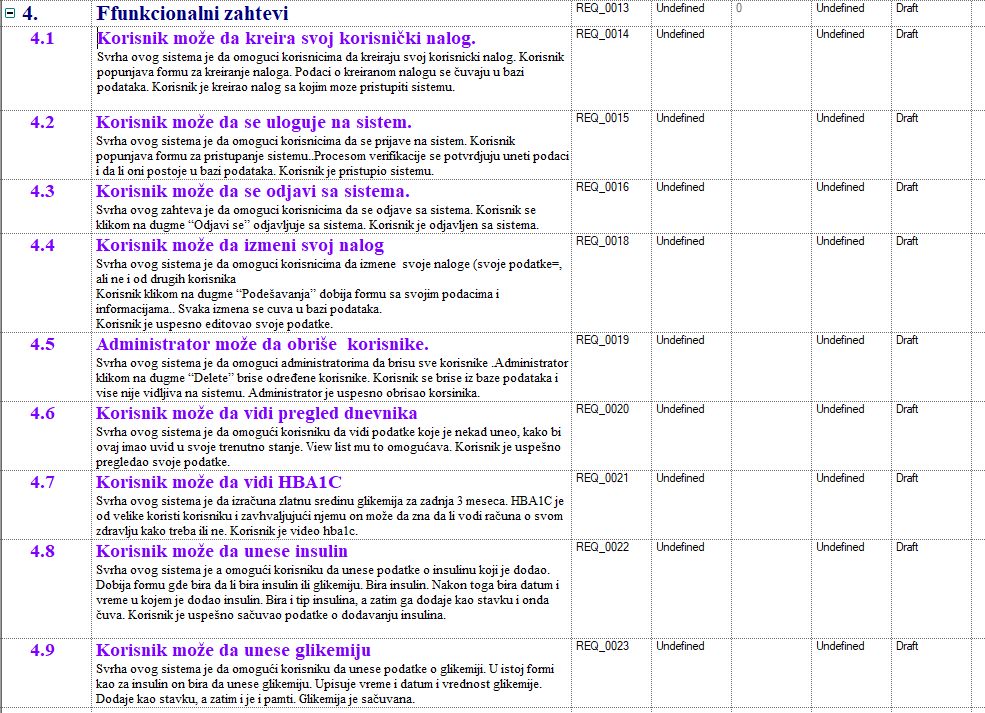
Ne zahteva nikakve dodatne interfejse.

### Komunikacioni interfejs

Samo vezu sa internetom.

### Funkcionalni zahtevi

Funkcionalni zahtevi i korisnika i administratora(Slika 1)



Slika -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 korsinike

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 usprešnom 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 omoguci korisnicima da kreiraju svoj korisnicki 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 porukia sa greškom.

## Korisnik može da se uloguje na sistem

Uvod

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

Ulaz

Korisnik popunjava formu za pristupanje sistemu.

Proces

Procesom verifikacije se potvrdjuju 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 omoguci 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 omoguci 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 zavhvaljujući 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 omogući 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 omogući 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.

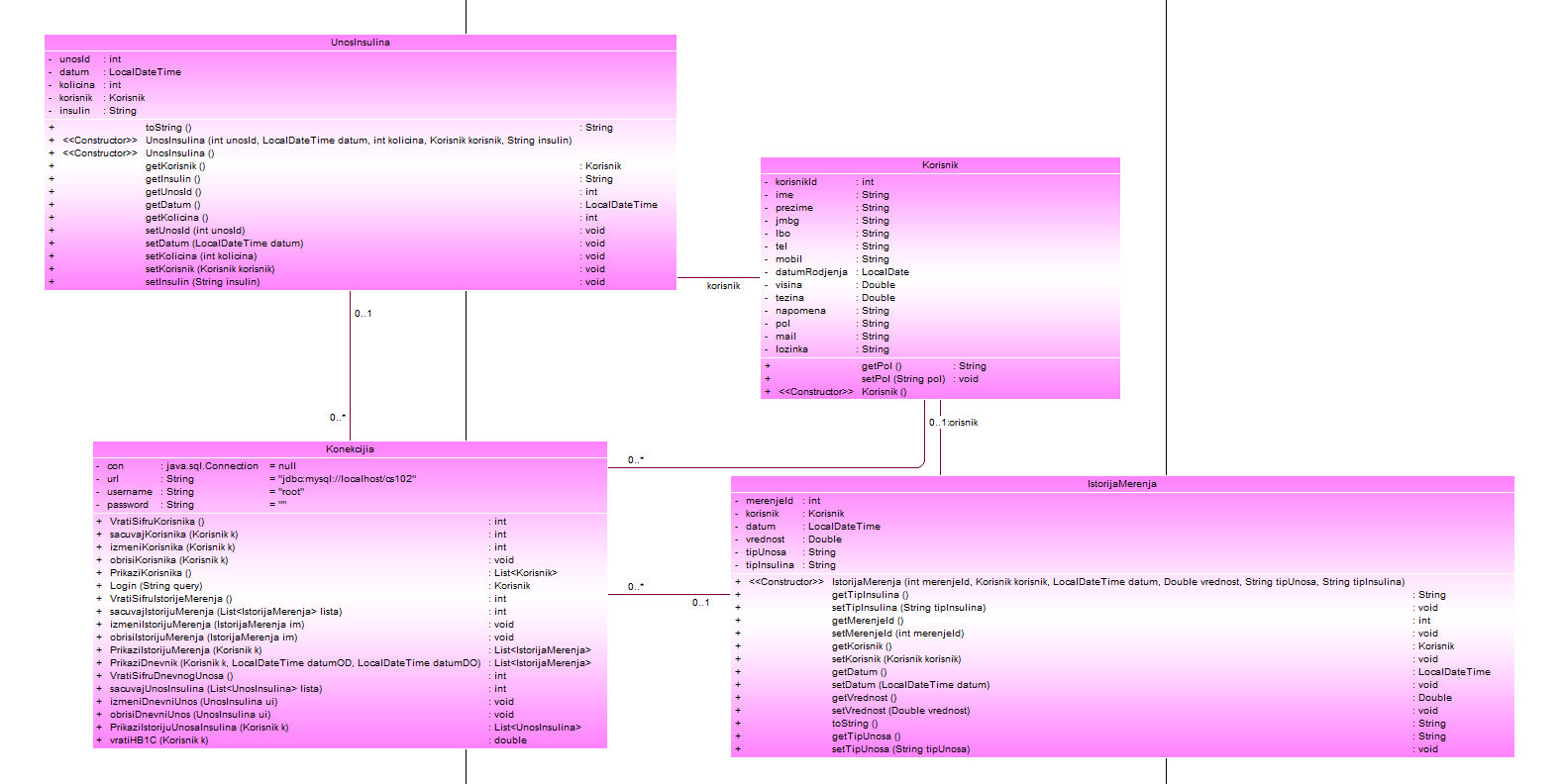
Izuzetak

Ukoliko su pogrešno uneti podaci, prikazuje se poruka.

# **Klasni dijagram**

# 

Dijagram klasa (deo [UML](https://hr.wikipedia.org/w/index.php?title=UML&action=edit&redlink=1)-a) je vrsta strukturnog dijagrama u [softverskom](https://hr.wikipedia.org/wiki/Softver) inžinjeringu, koji opisuje strukturu sastava objašnjavajući klase unutar sastava, njihove atribute i odnose.

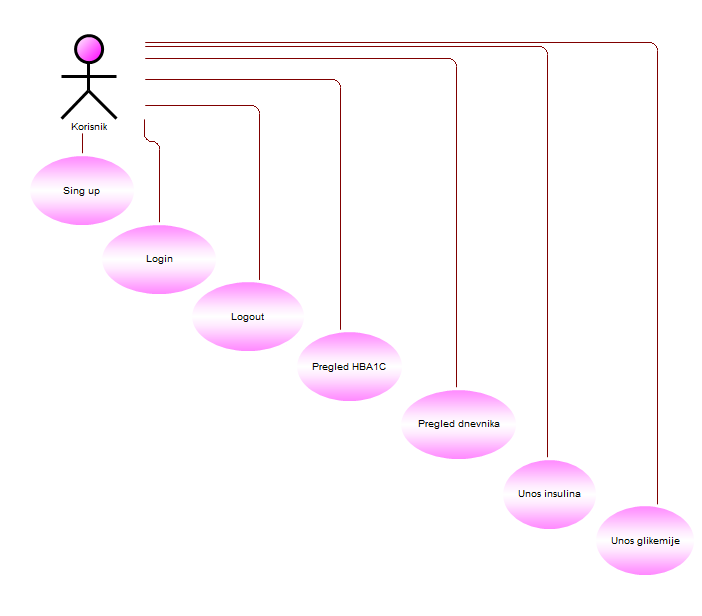


Slika -Klasni diagram

# **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 -Use case -Korisnik

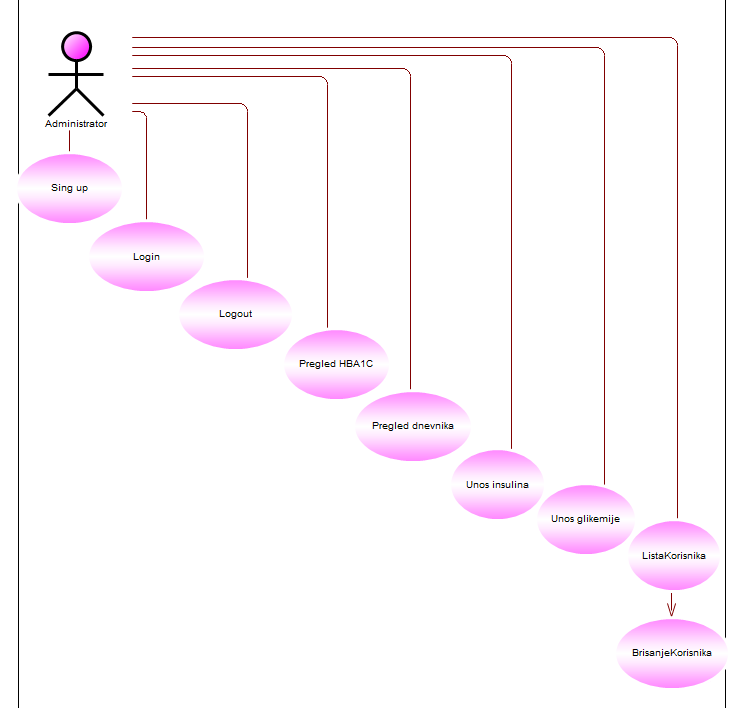
## Slučajevi korišćenja - Korisnik

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **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 -Use Case diagram-Administrator

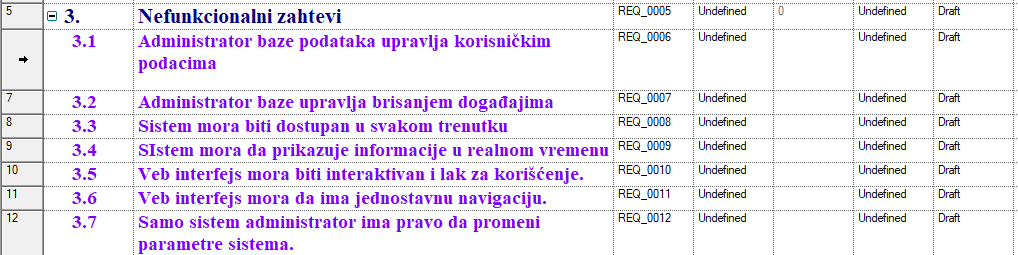
## Slučajevi korišćenja - Administrator

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **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 obrisan korisnik |

Tabela 2-Slučajevi korišćenja za administratora

# **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.



Slika -Ne-funkcionalni zahtevi

* **Performanse**

Sistem treba da funkcioniše bez ikakvih kašnjenja.

* **Pouzdanost**

Sistem ne sme da otkaže 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 nalozima.

* **Održivost**

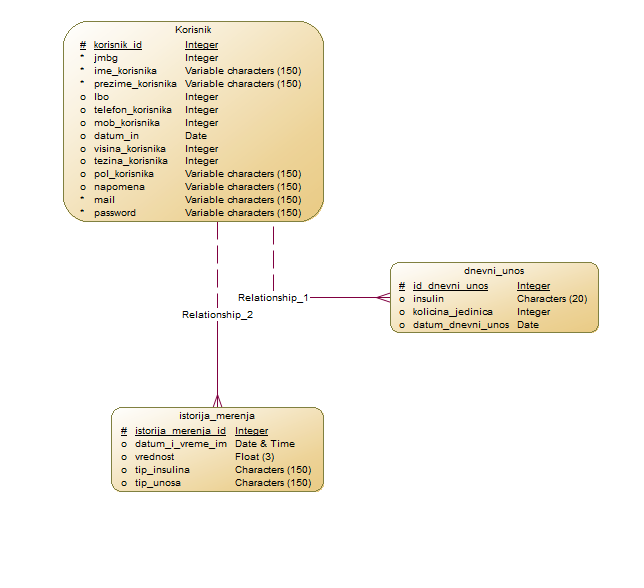
Baza podataka treba da bude fleksibilna. Ukoliko broj podataka i korisnika poraste, ona mora to izdržati.

* **Portabilnost**

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

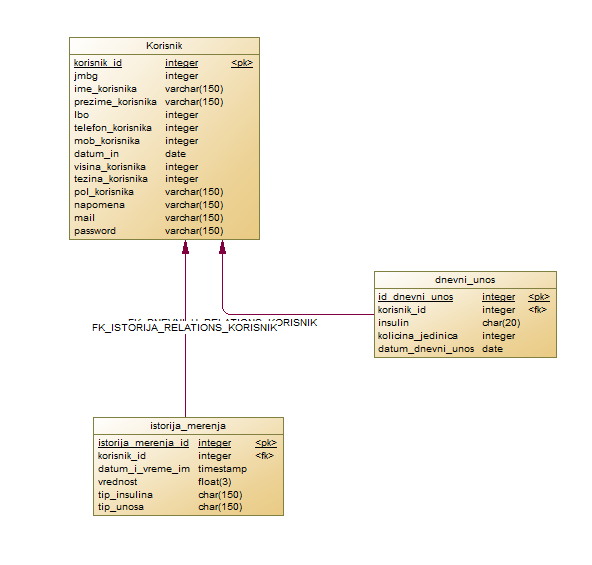
# **Implementacija dizajn paterna**

**Konceptualni model baze podataka**

****

Slika -Konceptualni model baze podataka

**Fizički model baze podataka**

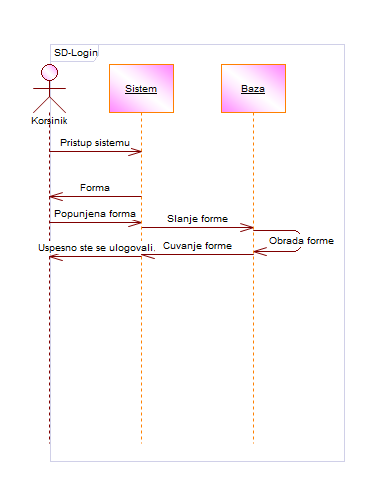
****

Slika -fizicki model baze podataka

# **Modeli analize sistema**

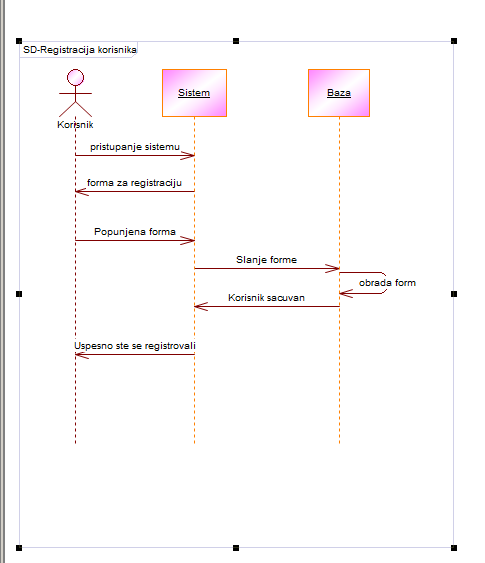
### Sekvencijalni dijagrami za administratora

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.



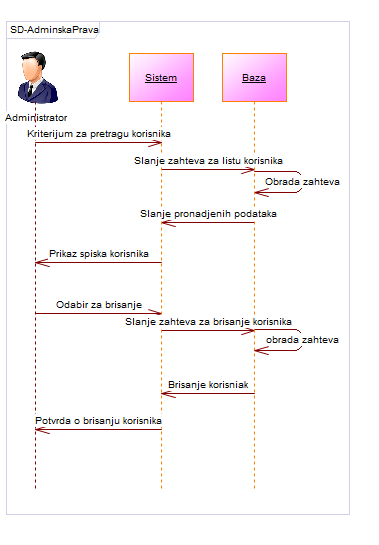
Slika -Sekvencijalni dijagram-login

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



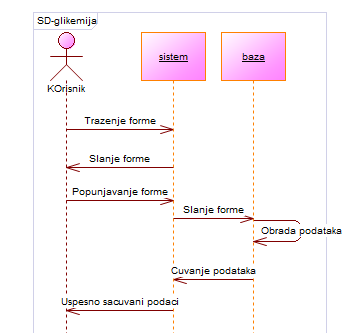
Slika -Sekvencijalni dijagram-Registracija

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



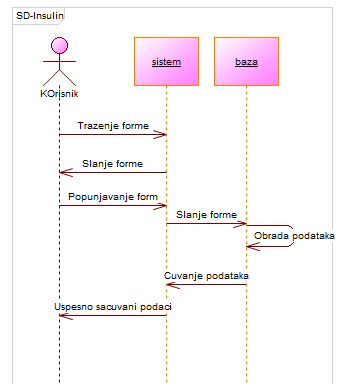
Slika -Sekvencijalni dijagram-AdminskaPrava

Sekvencijalni dijagram glikemija prikazuje kako se unosi glikemija Slika 11.



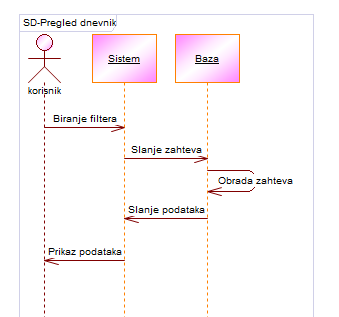
Slika -Sekvencijalni dijagram-glikemija

Sekvencijalni dijagram insulin prikazuje kako se unosi glikemija Slika 12.



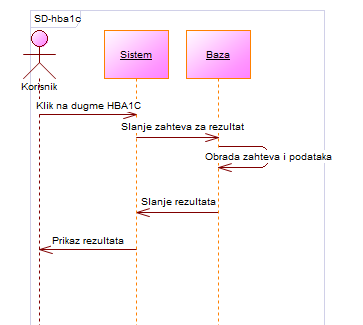
Slika -Sekvencijalni dijagram-insulin

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



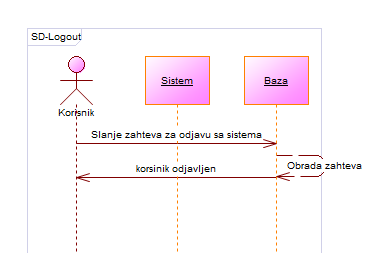
Slika -Sekvencijalni dijagram-pregledDnevnika

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



Slika -Sekvencijalni dijagram-HBA1C

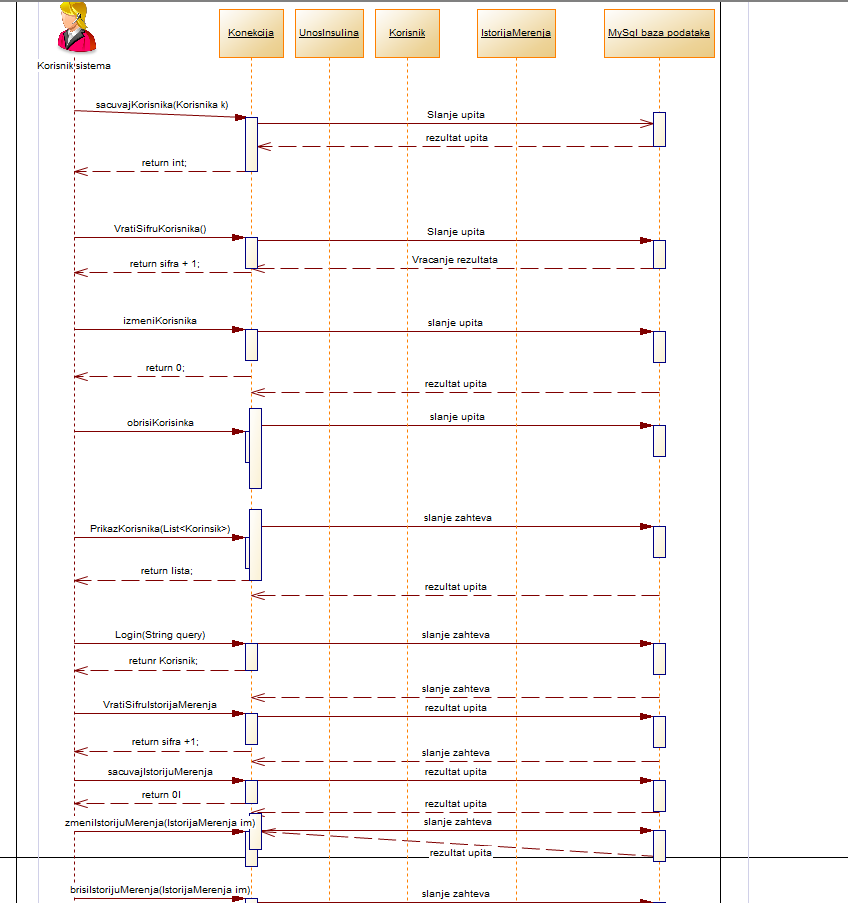
Sekvencijalni dijagram logout isključuje korisnika sa sistema Slika 14.



Slika -Sekvencijalni dijagram-LogOut

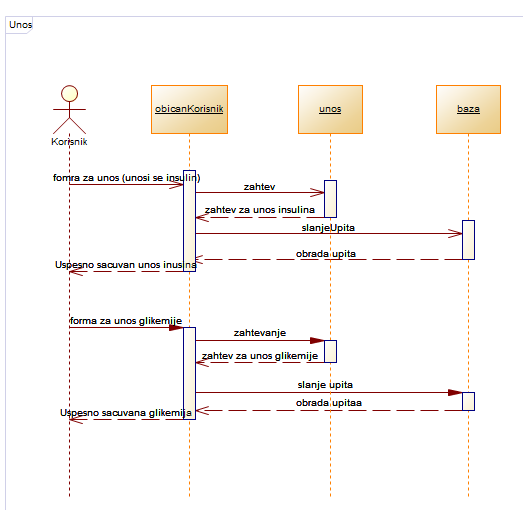
Detaljniji sekvencijalni dijagrami

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



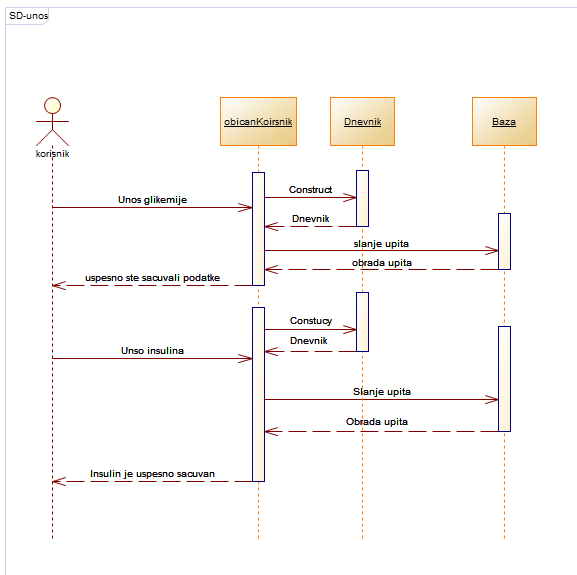
Slika -SD detaljniji korisnici

Sledeći detaljniji sekvencijalni dijagram je unos Slika 17,



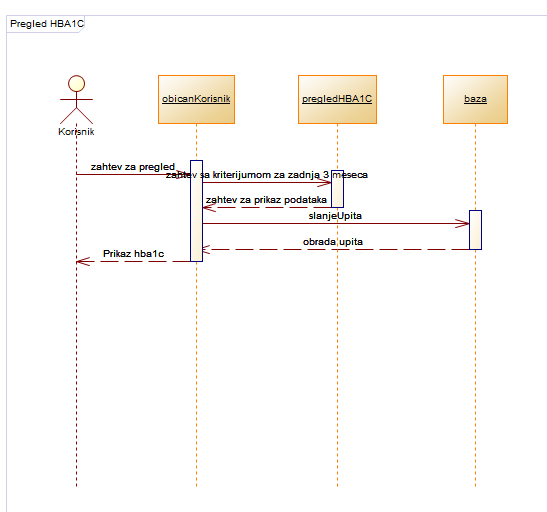
Slika -SD-unosi

Nakon unosa, prelazimo na prikaz dnevnika Slika 18,



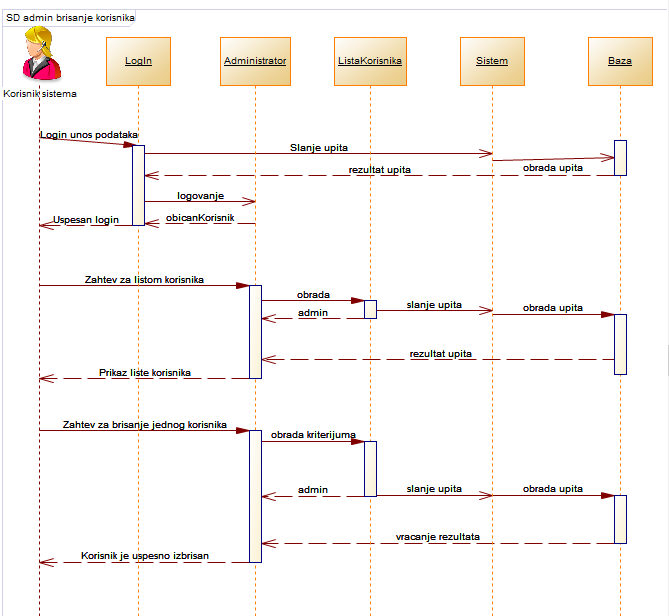
Slika -SD-dnevnik

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



Slika -SD-HBA1C

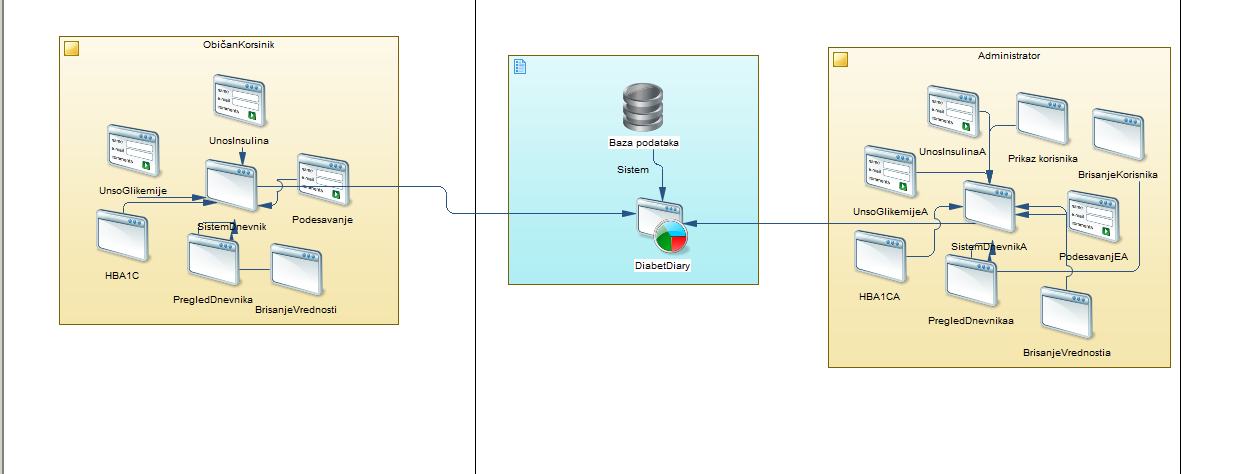
Ukoliko postoje i obični korisnici pored adminsitratora, administrator ih može ukloniti Slika 20



Slika -SD-admin brise korisnike

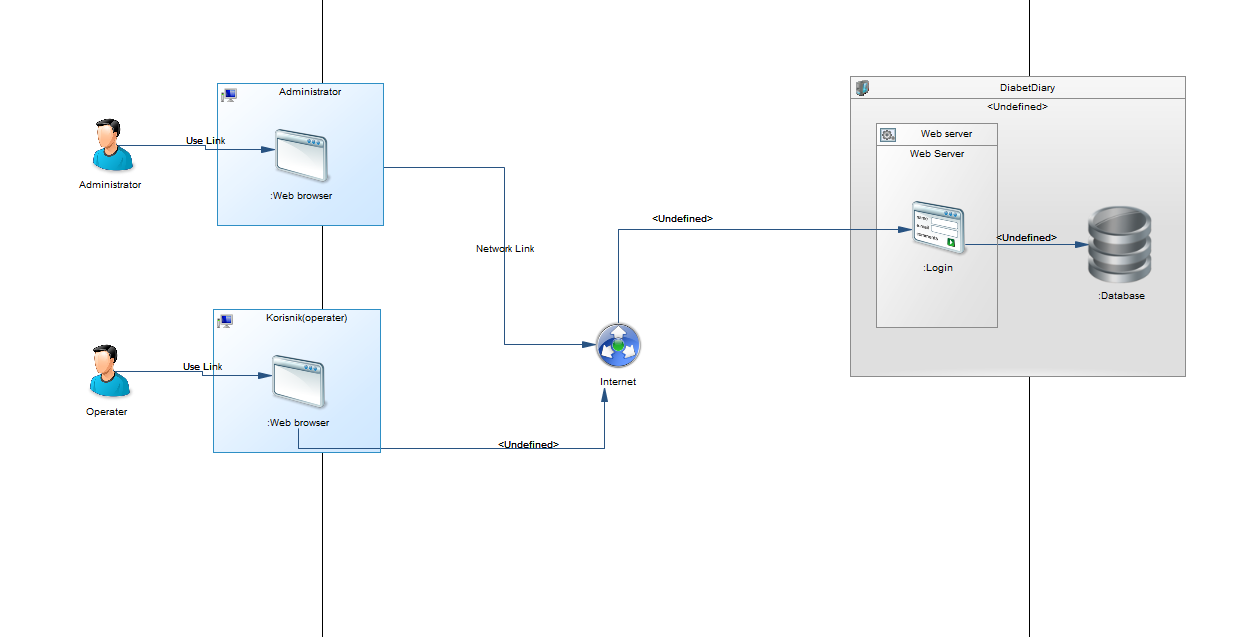
# **Dijagram infrastructure**

### Aplikacijski dijagram



Slika -Aplikacijska dijagram

Infrastruktura



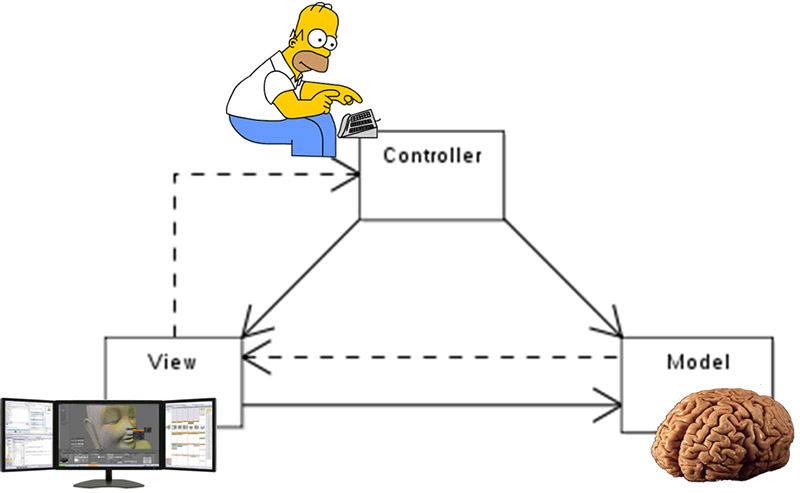
Slika -Infrastrukturni dijagram

# **Arhitektura sistema**

“Diabet diary” koristi dve arhitekture:

* MVC arhitekturu Slika 23

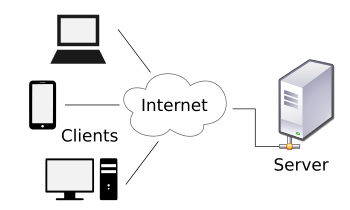
MVC podrazumeva šablon koji je u širokoj primeni u svetu. On podrazumeva 3 komponente:



Slika -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.

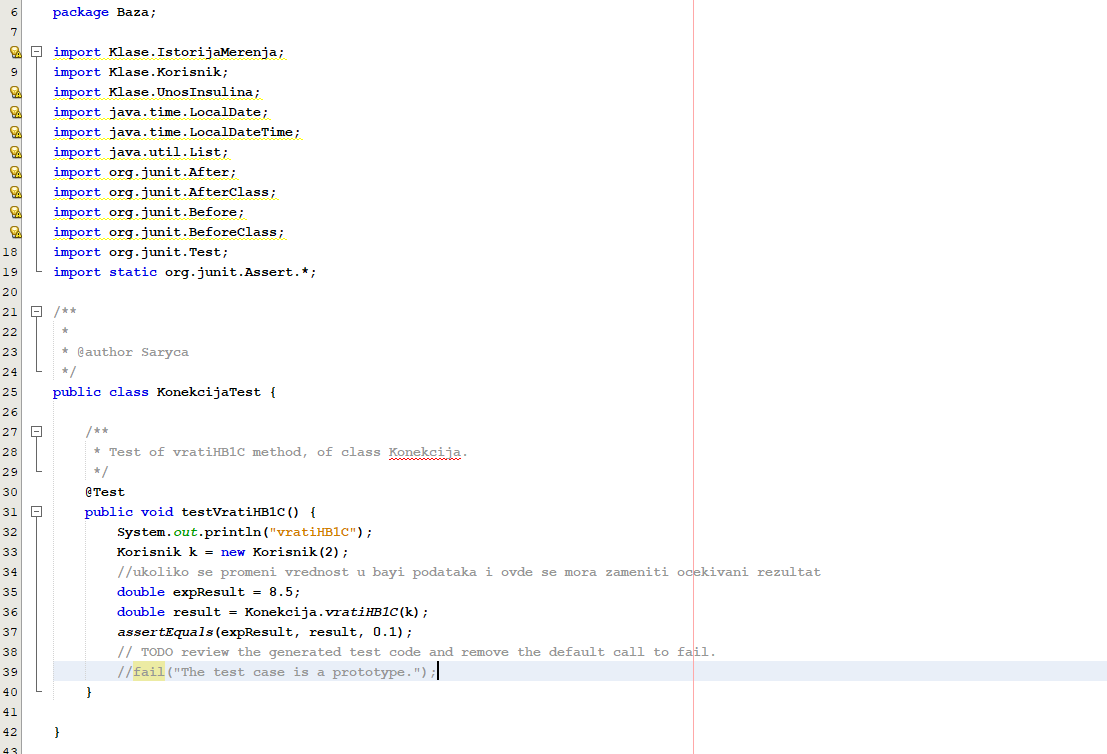
* Klijent – Server arhitekturuSlika 24

****

Slika -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.

# **Testiranje**

e

Slika- - Test klasa koja testira HBA1C vrednost.

nedostaje mi testiranje.

# **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.

# **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>. Accessed: Jan. 31.10.2017