**UNIVERZITET U BEOGRADU**

FAKULTET ORGANIZACIONIH NAUKA

Katedra za softversko inženjerstvo

Laboratorija za softversko inženjerstvo (SILAB)

Predmet: Projektovanje softvera

SEMINARSKI RAD

Tema:

***Softverski sistem za prijavljivanje pacijenata u dom zdravlja u Java okruženju***

|  |  |  |
| --- | --- | --- |
| Profesor: | Student: | Broj indeksa: |
| **Prof. dr Siniša Vlajić** | **Ivana Borisavljević** | **235/14** |

Beograd, 2020

# Sadržaj:

Contents

[Sadržaj: 1](#_Toc42110095)

[1. Korisnički zahtevi 3](#_Toc42110096)

[1.1 Verbalni opis sistema 3](#_Toc42110097)

[1.2 Slučajevi korišćenja 3](#_Toc42110098)

[1.2.1 SK1: Slučaj korišćenja- Unos novog pacijenta 4](#_Toc42110099)

[1.2.2 SK2: Slučaj korišćenja- Izmena pacijenta 5](#_Toc42110100)

[1.2.3 SK3: Slučaj korišćenja- Brisanje pacijenta 6](#_Toc42110101)

[1.2.4 SK4: Slučaj korišćenja: Zakazivanje pregleda 6](#_Toc42110102)

[1.2.5 SK 5: Slučaj korišćenja - Izmena pregleda (Složen) 7](#_Toc42110103)

[1.2.6 SK6: Slučaj korišćenja: Pretraga pregleda 9](#_Toc42110104)

[1.2.7 SK 7: Slučaj korišćenja – Unos novog doktora 9](#_Toc42110105)

[1.2.8 SK 8: Slučaj korišćenja- Izmena doktora 10](#_Toc42110106)

[1.2.9 SK9: Slučaj korišćenja- Brisanje doktora 11](#_Toc42110107)

[1.2.10 SK 10: Slučaj korišćenja – Logovanje službenika 12](#_Toc42110108)

[2.Analiza 12](#_Toc42110109)

[2.1 Ponašanje softverskog sistema-Sistemski dijagrami sekvenci 13](#_Toc42110110)

[2.1.1 DS1: Dijagram sekvenci slučaja korišćenja- Kreiranje novog pacijenta 13](#_Toc42110111)

[2.1.2 DS2: Dijagram sekvenci slučaja koriščenja- Izmena pacijenta 14](#_Toc42110112)

[2.1.3 DS3: Dijagram sekvenci slučaja korišćenja- Brisanje pacijenta 16](#_Toc42110113)

[2.1.4 DS4: Dijagram sekvenci slučaja korišćenja- Zakazivanje pregleda 18](#_Toc42110114)

[2.1.5 DS5: Dijagram sekvenci slučaja korišćenja- Izmena pregleda 20](#_Toc42110115)

[2.1.6 DS6: Dijagram sekvenci slučaja korišćenja- Pretraga pregleda 24](#_Toc42110116)

[2.1.7 DS7: Dijagram sekvenci slučaja korišćenja- Unos novog doktora 26](#_Toc42110117)

[2.1.8 DS8: Dijagram sekvenci slučaja korišćenja- Izmena doktora 27](#_Toc42110118)

[2.1.9 DS9: Dijagram sekvenci slučaja korišćenja- Brisanje doktora 30](#_Toc42110119)

[2.1.10 DS10: Dijagram sekvenci slučaja korišćenja- Logovanje službenika 31](#_Toc42110120)

[2.2 Sistemske operacije 32](#_Toc42110121)

[2.3 Ponašanje softverskog sistema – Definisanje ugovora o sistemskim operacijama 33](#_Toc42110122)

[2.3.1 Ugovor UG1: ZapamtiPacijenta 33](#_Toc42110123)

[2.3.2 Ugovor UG2: VratiListuPacijenata 33](#_Toc42110124)

[2.3.3 Ugovor UG3: IzmeniPacijenta 33](#_Toc42110125)

[2.3.4 Ugovor UG4: ObrišiPacijenta 34](#_Toc42110126)

[2.3.5 Ugovor UG5: ZapamtiDoktora 34](#_Toc42110127)

[2.3.6 Ugovor UG6: VratiListuDoktora 34](#_Toc42110128)

[2.3.7 Ugovor UG7: VratiListuSlobodnihDoktora 34](#_Toc42110129)

[2.3.8 Ugovor UG8: IzmeniDoktora 34](#_Toc42110130)

[2.3.9 Ugovor UG9: ObrišiDoktora 35](#_Toc42110131)

[2.3.10 Ugovor UG10: ZapamtiPregled 35](#_Toc42110132)

[2.3.11 Ugovor UG11: VratiListuPregleda 35](#_Toc42110133)

[2.3.12 Ugovor UG12: IzmeniPregled 35](#_Toc42110134)

[2.3.13 Ugovor UG13: VratiListuSmena 35](#_Toc42110135)

[2.3.14 Ugovor UG14: VratiListuSpecijalizacija 36](#_Toc42110136)

[2.3.15 Ugovor UG15: VratiListuVrstaPregleda 36](#_Toc42110137)

[2.3.16 Ugovor UG16: VratiListuLekova 36](#_Toc42110138)

[2.3.17 Ugovor UG17: NađiSlužbenika 36](#_Toc42110139)

[2.3.18 Ugovor UG18: UčitajPacijenta 36](#_Toc42110140)

[2.3.19 Ugovor UG19: UčitajDoktora 37](#_Toc42110141)

[2.3.20 Ugovor UG20: UčitajPregled 37](#_Toc42110142)

[2.4 Struktura softverskog sistema-Konceptualni model (Domenski model) 38](#_Toc42110143)

[2.5 Struktura softverskog sistema – Relacioni model 39](#_Toc42110144)

[3. Projektovanje 44](#_Toc42110145)

[3.1 Arhitektura softverskog Sistema 44](#_Toc42110146)

[3.2 Projektovanje korisničkog interfejsa 45](#_Toc42110147)

[3.2.1 Projektovanje korisničkog interfejsa 45](#_Toc42110148)

[3.2.2 Projektovanje kontrolera korisničkog interfejsa 82](#_Toc42110149)

[3.3 Projektovanje aplikacione logike 82](#_Toc42110150)

[3.3.1 Komunikacija sa klijentom 83](#_Toc42110151)

[3.3.2. Kontroler 85](#_Toc42110152)

[3.3.3. Ponašanje softverskog sistema 86](#_Toc42110153)

[3.3.4. Broker baze podataka i domenske klase 88](#_Toc42110154)

[3.4. Projektovanje baze podataka 90](#_Toc42110155)

[4. Implementacija 93](#_Toc42110156)

[5. Testiranje 95](#_Toc42110157)

[6. Literatura 95](#_Toc42110158)

# 1. Korisnički zahtevi

## 1.1 Verbalni opis sistema

Potrebno je napraviti sistem koji omogućava prijavljivanje pacijenata u dom zdravlja, kao i vođenje evidencije o prijavljenim pacijentima, njihovim zakazanim i obavljenim pregledima. Omogućiti unos novih, izmenu i brisanje podataka o postojećim pacijentima.

Takođe, potrebno je voditi evidenciju o doktorima koji rade u domu zdravlja, unos novog, izmenu i brisanje postojećih doktora.

Treba omogućiti i zakazivanje pregleda, koje uključuje unos podataka o pacijentu i doktoru koji obavlja pregled. Omogućiti pretragu pregleda, izmenu pregleda i unos liste lekova (terapije) nakon obavljenog pregleda.

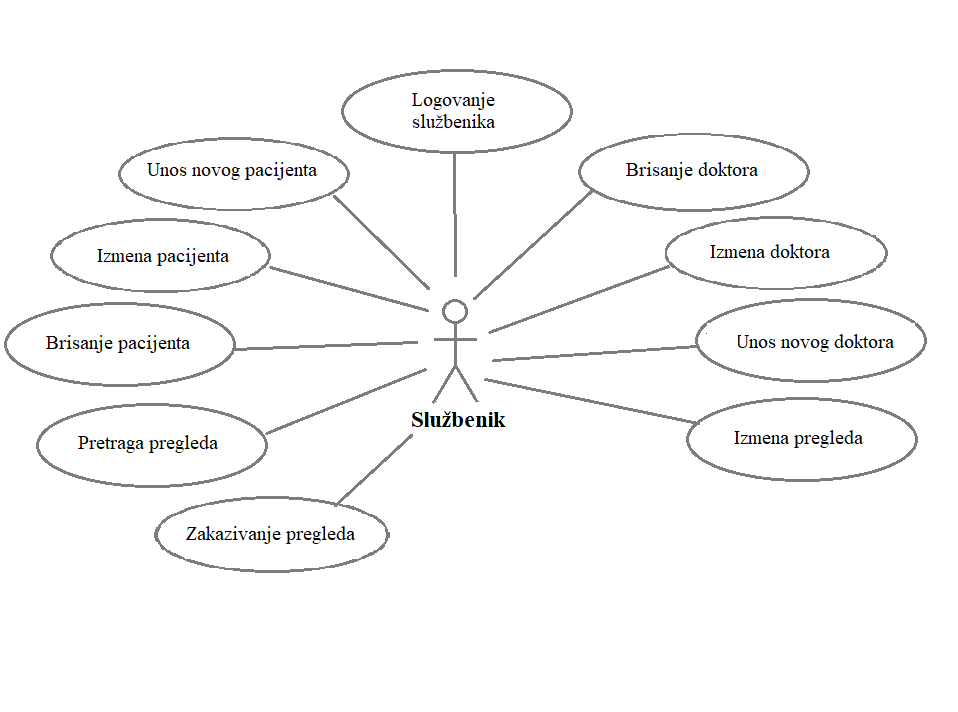
Korisnik sistema je službenik.

Prilikom pristupanja sistemu vrši se provera da li je dati službenik ovlašćen. Da bi pravio izmene, mora prvo biti ulogovan.

## 1.2 Slučajevi korišćenja

Za izradu ovog rada, opisano je 10 slučajeva korišćenja, datih ispod:

1. Unos novog pacijenta
2. Izmena pacijenta
3. Brisanje pacijenata
4. Zakazivanje pregleda
5. Izmena pregleda (Složen)
6. Pretraga pregleda
7. Unos novog doktora
8. Izmena doktora
9. Brisanje doktora
10. Logovanje službenika



### 1.2.1 SK1: Slučaj korišćenja- Unos novog pacijenta

**Naziv SK:**

Unošenje novog pacijenta

**Aktori SK:**

Službenik

**Učesnici SK:**

Službenik i sistem (program)

**Preduslov:** Sistem je uključen i službenik je ulogovan pod svojom šifrom. Sistem prikazuje formu za rad sa pacijentom. Učitana je lista pacijenata.

**Osnovni scenario SK**

1. Službenik unosi podatke o pacijentu. (APUSO)
2. Službenik kontroliše da li je korektno uneo podatke u pacijenta. (ANSO)
3. Službenik poziva sistem da zapamti podatke o pacijentu. (APSO).
4. Sistem pamti podatke o pacijentu. (SO)
5. Sistem prikazuje službeniku poruku: „Sistem je zapamtio pacijenta “(IA)

**Alternativni scenariji**

5.1 Ukoliko sistem ne može da zapamti podatke o pacijentu, on prikazuje službeniku poruku: “Sistem ne može da zapamti pacijenta”. (IA)

### 1.2.2 SK2: Slučaj korišćenja- Izmena pacijenta

**Naziv SK**

Izmena pacijenta

**Autori SK**

Službenik

**Učesnici SK**

Službenik i sistem (program)

**Preduslov:** Sistem je uključen i službenik je ulogovan pod svojom šifrom. Sistem prikazuje formu za rad sa pacijentima. Učitana je lista pacijenata.

**Osnovni scenario SK**

1. Službenik bira pacijenta. (APUSO)
2. Službenik poziva sistem da prikaže sve podatke o izabranom pacijentu. (APSO)
3. Sistem traži podatke o izabranom pacijentu. (SO)
4. Sistem prikazuje službeniku podatke o odabranom pacijentu. (IA)
5. Službenik unosi (menja) podatke o pacijentu. (APUSO)
6. Službenik kontroliše da li je korektno uneo podatke o pacijentu. (ANSO)
7. Službenik poziva sistem da izmeni podatke o pacijentu. (APSO)
8. Sistem menja podatke o pacijentu. (SO)
9. Sistem prikazuje službeniku i poruku: “Sistem je izmenio pacijenta “ . (IA)

**Alternativni scenariji**

4.1 Ukoliko sistem ne može da nađe podatke o pacijentu, on službeniku pikazuje poruku: “Sistem ne može da nađe podatke o izabranom pacijentu! ”. Prekida se izvršenje scenarija. (IA)

9.1 Ukoliko sistem ne može da izmeni podatke o pacijentu, on prikazuje službeniku poruku: „Sistem ne može da izmeni pacijenta! “. (IA)

### 1.2.3 SK3: Slučaj korišćenja- Brisanje pacijenta

**Naziv SK**

Brisanje pacijenta

**Autori SK**

Službenik

**Učesnici SK**

Službenik i sistem (program)

**Preduslov:** Sistem je uključen i službenik je ulogovan pod svojom šifrom. Sistem prikazuje formu za rad sa pacijentima. Učitana je lista pacijenata.

**Osnovni scenario SK**

1. Službenik bira pacijenta kojeg želi da obriše. (APUSO)
2. Službenik poziva sistem da obriše podatke o pacijentu. (APSO)
3. Sistem briše pacijenta i pamti izmene. (SO)
4. Sistem prikazuje službeniku poruku: “Sistem je obrisao pacijenta “. (IA)

**Alternativni scenariji**

4.1. Ukoliko sistem ne može da obriše pacijenta, on prikazuje službeniku poruku: „Sistem ne može da obriše pacijenta! “. (IA)

### 1.2.4 SK4: Slučaj korišćenja: Zakazivanje pregleda

**Naziv SK**

Zakazivanje pregleda

**Aktori SK**

Službenik

**Učesnici SK**

Službenik i sistem (program)

Preduslov: Sistem je uključen i službenik je ulogovan pod svojom šifrom. Sistem prikazuje formu za rad sa pregledima. Učitana je lista pacijenata, izabran je pacijent i učitana je lista vrsta pregleda.

**Osnovni scenario SK**

1. Službenik unosi podatke o pregledu. (APUSO)
2. Službenik kontroliše da li je korektno uneo podatke o pregledu. (ANSO)
3. Službenik poziva sistem da nađe podatke o slobodnim doktorima. (APSO)
4. Sistem traži slobodne doktore. (SO)
5. Sistem prikazuje službeniku pronađene slobodne doktore. (IA)
6. Službenik bira slobodnog doktora. (APUSO)
7. Službenik poziva sistem da zapamti podatke o pregledu. (APSO)
8. Sistem pamti podatke o pregledu. (SO)
9. Sistem prikazuje službeniku poruku: “Sistem je zapamtio pregled! “. (IA)

**Alternativni scenariji**

5.1 Ukoliko sistem ne može da pronađe slobodne doktore, on prikazuje službeniku poruku: Sistem ne može da nađe slobodne doktore!”. Prekida se izvršenje scenarija. (IA)

9.1 Ukoliko sistem ne može da zapamti pregled on prikazuje službeniku poruku: “Sistem ne može da zapamti pregled”. (IA)

### 1.2.5 SK 5: Slučaj korišćenja - Izmena pregleda (Složen)

**Naziv SK**

Izmena pregleda

**Aktori SK**

Službenik

**Učesnici SK**

Službenik i sistem (program)

**Preduslov**: Sistem je uključen i službenik je ulogovan pod svojom šifrom. Sistem prikazuje formu za rad sa pacijentima. Učitana je lista pacijenata, izabran je pacijent i učitana je lista lekova.

**Osnovni scenario SK**

1. Službenik bira pacijenta. (APUSO)
2. Službenik poziva sistem da nađe sve podatke (preglede) o izabranom pacijentu. (APSO)
3. Sistem traži podatke o izabranom pacijentu. (SO)
4. Sistem prikazuje službeniku podatke o izabranom pacijentu (preglede). (IA)
5. Službenik bira pregled. (APUSO)
6. Službenik poziva sistem da prikaže sve podatke o izabranom pregledu. (APSO)
7. Sistem traži podatke o izabranom pregledu. (SO)
8. Sistem prikazuje službeniku podatke o odabranom pregledu. (IA)
9. Službenik menja podatke o pregledu (unosi terapije). (APUSO)
10. Službenik kontroliše da li je korektno uneo podatke o obavljenom pregledu. (ANSO)
11. Službenik poziva sistem da izmeni podatke o pregledu pacijenta. (APSO)
12. Sistem menja podatke o pregledu. (SO)
13. Sistem prikazuje službeniku poruku: “Sistem je izmenio pregled “. (IA)

**Alternativni scenariji**

4.1 Ukoliko sistem ne može da nađe podatke o izabranom pacijentu on prikazuje službeniku poruku: “Sistem ne može da nađe podatke o izabranom pacijentu!” Prekida se izvršenje scenarija. (IA)

8.1 Ukoliko sistem ne može da nađe podatke o izabranom pregledu on prikazuje službeniku poruku: “Sistem ne može da nađe podatke o izabranom pregledu” Prekida se izvršenje scenarija. (IA)

13.1 Ukoliko sistem ne može da izmeni podatke o obavljenom pregledu, on službeniku prikazuje poruku: “Sistem ne može da izmeni pregled “. (IA)

### 

### 1.2.6 SK6: Slučaj korišćenja: Pretraga pregleda

**Naziv SK**

Pretraga pacijenta

**Akteri SK**

Službenik

**Učesnici SK**

Službenik i sistem (program)

**Preduslov:** Sistem je uključen i službenik je ulogovan pod svojom šifrom. Sistem prikazuje formu za rad sa pacijentima. Učitana je lista pacijenata.

**Osnovni scenario SK**

1. Službenik bira pacijenta. (APUSO)
2. Službenik poziva sistem da prikaže sve podatke (istoriju pregleda) o izabranom pacijentu. (APSO)
3. Sistem traži podatke o izabranom pacijentu. (SO)
4. Sistem prikazuje službeniku sve podatke o odabranom pacijentu. (IA)
5. Službenik bira pregled koji želi da prikaže. (APUSO)
6. Službenik poziva sistem da učita podatke o izabranom pregledu. (APSO)
7. Sistem traži podatke o izabranom pregledu. (SO)
8. Sistem prikazuje službeniku sve podatke o odabranom pregledu. (IA)

**Alternativni scenariji**

* 1. Ukoliko sistem ne može da nađe podatke o pacijentu, on službeniku prikazuje poruku: “Sistem ne može da nađe podatke o izabranom pacijentu! “. (IA)
  2. Ukoliko sistem ne može da nađe podatke o pregledu, on službeniku prikazuje poruku: “Sistem ne može da nađe podatke o izabranom pregledu! “. (IA)

### 1.2.7 SK 7: Slučaj korišćenja – Unos novog doktora

**Naziv SK**

Unošenje novog doktora

**Aktori SK**

Službenik

**Učesnici SK**

Službenik i sistem (program)

**Preduslov:** Sistem je uključen i službenik je ulogovan pod svojom šifrom. Sistem prikazuje formu za rad sa doktorom. Učitana je lista doktora, specijalizacija i smena.

**Osnovni scenario SK**

* 1. Službenik unosi podatke o doktoru. (APUSO)
  2. Službenik kontroliše da li je korektno uneo podatke u doktora. (ANSO)
  3. Službenik poziva sistem da zapamti podatke o doktoru. (APSO)
  4. Sistem pamti podatke o doktoru. (SO)
  5. Sistem prikazuje službeniku zapamćenog doktora i poruku: “Sistem je zapamtio doktora”. (IA)

**Alternativni scenariji**

5.1 Ukoliko sistem ne može da zapamti doktora on prikazuje službeniku poruku: “Sistem ne može da zapamti doktora”. (IA)

### 1.2.8 SK 8: Slučaj korišćenja- Izmena doktora

**Naziv SK**

Izmena doktora

**Autori SK**

Službenik

**Učesnici SK**

Službenik i sistem (program)

**Preduslov:** Sistem je uključen i službenik je ulogovan pod svojom šifrom. Sistem prikazuje formu za rad sa doktorima. Učitana je lista doktora i izabran je doktor. Učitana je lista specijalizacija i smena.

**Osnovni scenario SK**

1. Službenik bira doktora. (APUSO)
2. Službenik poziva sistem da prikaže sve podatke o izabranom doktoru. (APSO)
3. Sistem traži podatke o izabranom doktoru. (SO)
4. Sistem prikazuje službeniku podatke o odabranom doktoru. (IA)
5. Službenik unosi (menja) podatke o doktoru. (APUSO)
6. Službenik kontroliše da li je korektno uneo podatke o doktoru. (ANSO)
7. Službenik poziva sistem da izmeni podatke o doktoru. (APSO)
8. Sistem menja podatke o doktoru. (SO)
9. Sistem prikazuje službeniku i poruku: “Sistem je izmenio doktora “ . (IA)

**Alternativni scenariji**

4.1 Ukoliko sistem ne može da nađe podatke o doktoru, on službeniku pikazuje poruku: “Sistem ne može da nađe podatke o izabranom doktoru”. Prekida se izvršenje scenarija. (IA)

9.1 Ukoliko sistem ne može da izmeni podatke o doktoru, on prikazuje službeniku poruku: „Sistem ne može da izmeni doktora “. (IA)

### 1.2.9 SK9: Slučaj korišćenja- Brisanje doktora

**Naziv SK**

Brisanje doktora

**Autori SK**

Službenik

**Učesnici SK**

Službenik i sistem (program)

**Preduslov:** Sistem je uključen i službenik je ulogovan pod svojom šifrom. Sistem prikazuje formu za rad sa doktorima. Učitana je lista doktora.

**Osnovni scenario SK**

1. Službenik bira doktora kojeg želi da obriše. (APUSO)
2. Službenik poziva sistem da obriše podatke o doktoru. (APSO)
3. Sistem briše doktora i pamti izmene. (SO)
4. Sistem prikazuje službeniku poruku: “Sistem je obrisao doktora “. (IA)

**Alternativni scenariji**

4.1. Ukoliko sistem ne može da obriše doktora, on prikazuje službeniku poruku: „Sistem ne može da obriše doktora! “. (IA)

### 1.2.10 SK 10: Slučaj korišćenja – Logovanje službenika

**Naziv SK**

Logovanje službenika

**Aktori SK**

Službenik

**Učesnici SK**

Službenik i sistem (program)

**Preduslov:** Sistem je uključen i prikazuje formu za logovanje službenika.

**Osnovni scenario SK**

1. Službenik unosi podatke za autentifikaciju službenika. (APUSO)
2. Službenik poziva sistem da pronađe službenika sa zadatim podacima. (APSO)
3. Sistem pretražuje službenike. (SO)
4. Sistem prikazuje poruku “Uspešna prijava na sistem” i omogućava pristup sistemu. (IA)

**Alternativni scenariji**

4.1. Ukoliko sistem ne može da nađe službenika prikazuje poruku “Sistem ne može da nađe službenika”. (IA)

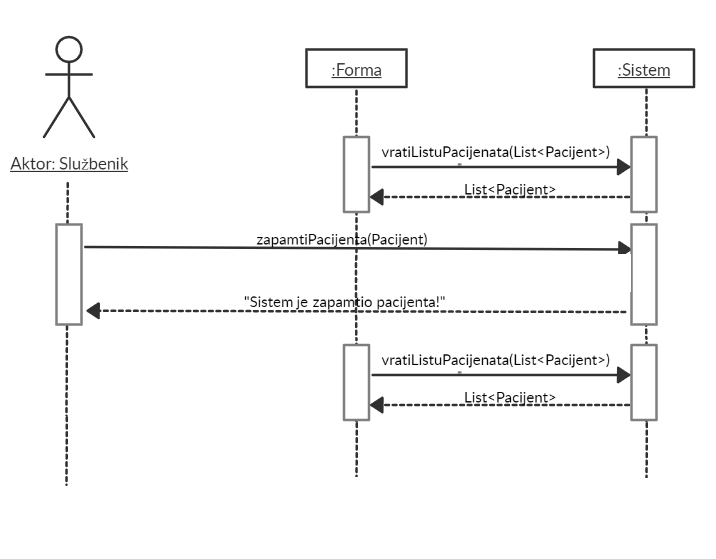
# 2.Analiza

Faza analize prikazuje ponašanje softverskog sistema kao I njegovu logičku strukturu. Ponašanje sistema predstavljeno je pomoću dijagrama sevenci, ugovora o sistemskim operacijama, dok je struktura prikazana pomoću sistemskih dijagrama sekvenci, ugovora o sistemskim operacijama (koji se dobijaju na osnovu sistemskih dijagrama sekvenci), dok je struktura softverskog sistema pikazana pomoću konceptualnog I relacionog modela.

## 2.1 Ponašanje softverskog sistema-Sistemski dijagrami sekvenci

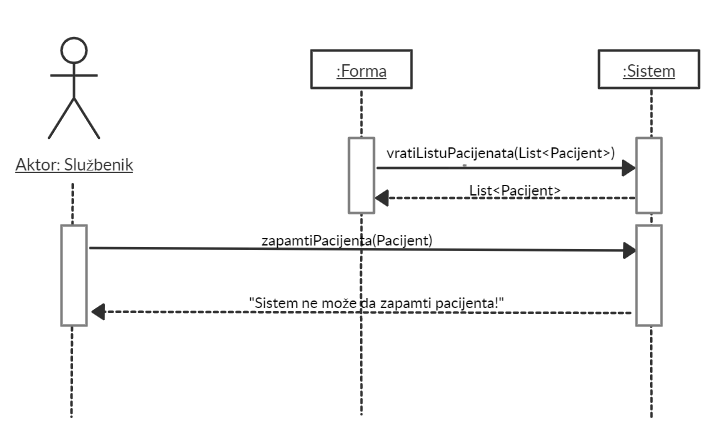
### 2.1.1 DS1: Dijagram sekvenci slučaja korišćenja- Kreiranje novog pacijenta

1. Forma poziva sistem da vrati listu pacijenata. (APSO)
2. Sistem vraća formi listu pacijenata. (IA)
3. Službenik poziva sistem da zapamti podatke o pacijentu. (APSO)
4. Sistem prikazuje službeniku poruku: „Sistem je zapamtio pacijenta “(IA)
5. Forma poziva sistem da vrati listu pacijenata. (APSO)
6. Sistem vraća formi listu pacijenata. (IA)



**Alternativni scenariji**

* 1. Ukoliko sistem ne može da zapamti podatke o pacijentu, on prikazuje službeniku poruku: „Sistem ne može da zapamti pacijenta “. (IA)



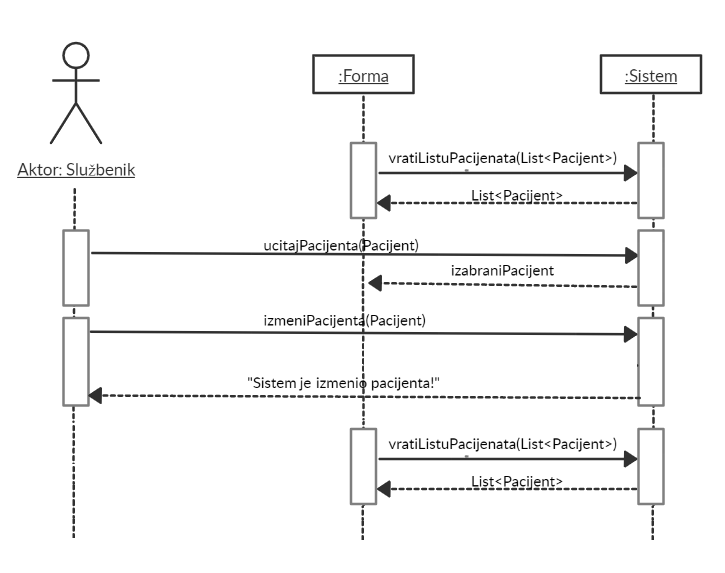
Sa navedenih sekvencnih dijagrama uočavaju se sistemske operacije koje treba projektovati:

1. signal vratiListuPacijenata (List<Pacijent>)

2. signal zapamtiPacijenta (Pacijent)

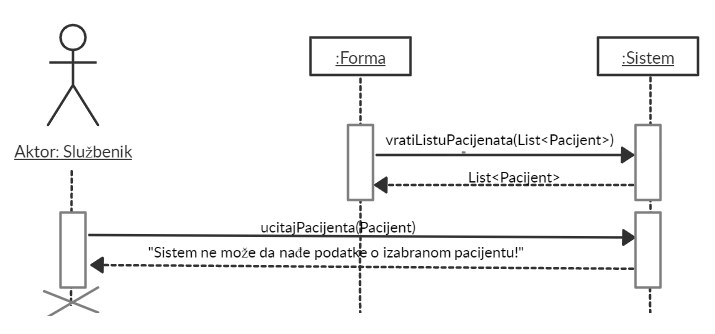
### DS2: Dijagram sekvenci slučaja koriščenja- Izmena pacijenta

* 1. Forma poziva sistem da vrati listu pacijenata. (APSO)
  2. Sistem vraća formi listu pacijenata. (IA)
  3. Službenik poziva sistem da prikaže sve podatke o izabranom pacijentu. (APSO)
  4. Sistem prikazuje formi podatke o odabranom pacijentu. (IA)
  5. Službenik poziva sistem da izmeni podatke o pacijentu. (APSO)
  6. Sistem prikazuje službeniku poruku: „Sistem je izmenio pacijenta “(IA)
  7. Forma poziva sistem da vrati podatke o pacijentima. (APSO)
  8. Sistem vraća formi podatke o pacijentima. (IA)

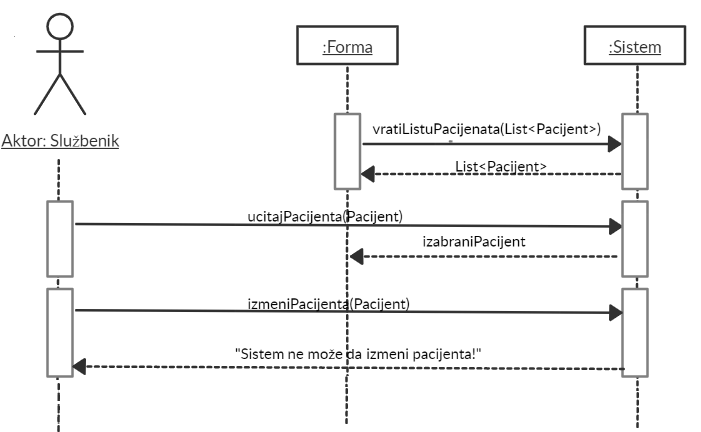


**Alternativni scenariji**

4.1 Ukoliko sistem ne može da nađe podatke o pacijentu, on službeniku prikazuje poruku: „Sistem ne može da nađe podatke o izabranom pacijentu “ Prekida se izvršenje scenarija. (IA)

****

6.1 Ukoliko sistem ne može da izmeni podatke o pacijentu, on službeniku pikazuje poruku: “Sistem ne može da izmeni pacijenta”. (IA)



Sa navedenih sekvencnih dijagrama uočavaju se sistemske operacije koje treba projektovati:

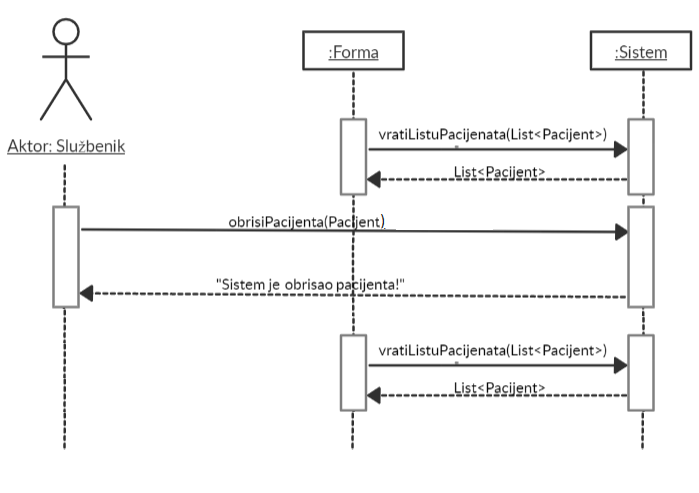
1. signal vratiListuPacijenata (List<Pacijent>)

2. signal ucitajPacijenta (Pacijent)

3. signal izmeniPacijenta (Pacijent)

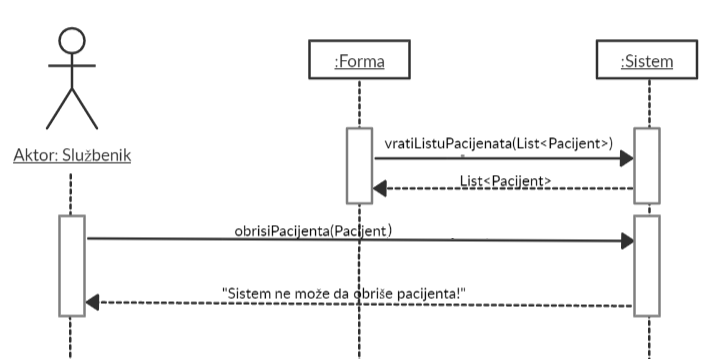
### 2.1.3 DS3: Dijagram sekvenci slučaja korišćenja- Brisanje pacijenta

1. Forma poziva sistem da vrati listu pacijenata. (APSO)
2. Sistem vraća formi listu pacijenata. (IA)
3. Službenik poziva sistem da obriše podatke o pacijentu. (APSO)
4. Sistem prikazuje službeniku poruku: “Sistem je obrisao pacijenta “. (IA)
5. Forma poziva sistem da vrati listu pacijenata. (APSO)
6. Sistem vraća formi listu pacijenata. (IA)



**Alternativni scenariji**

4.1 Ukoliko sistem ne može da obriše pacijenta, on prikazuje službeniku poruku: „Sistem ne može da obriše pacijenta “ (IA)

****

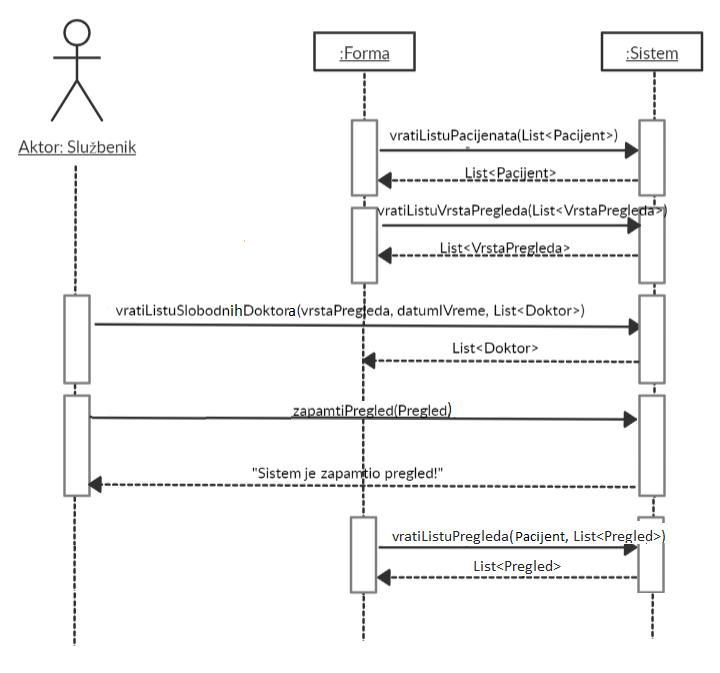
Sa navedenih sekvencnih dijagrama uočavaju se sistemske operacije koje treba projektovati:

1. signal vratiListuPacijenata (List<Pacijent>)

2. signal obrisiPacijenta (Pacijent, List<Pacijent>)

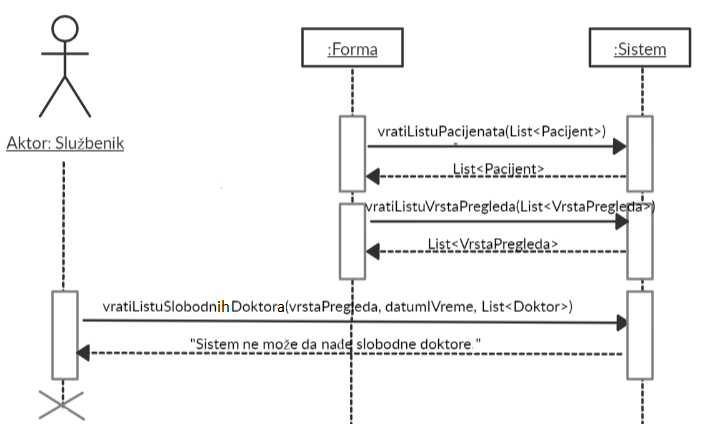
### 2.1.4 DS4: Dijagram sekvenci slučaja korišćenja- Zakazivanje pregleda

1. Forma poziva sistem da vrati listu pacijenata. (APSO)
2. Sistem vraća formi listu pacijenata. (IA)
3. Forma poziva sistem da vrati listu vrsta pregleda. (APSO)
4. Sistem vraća formi listu vrsta pregleda. (IA)
5. Službenik poziva sistem da nađe podatke o slobodnim doktorima. (APSO)
6. Sistem prikazuje službeniku pronađene slobodne doktore. (IA)
7. Službenik poziva sistem da zapamti podatke o pregledu. (APSO)
8. Sistem prikazuje službeniku i poruku “Sistem je zapamtio pregled”.
9. Forma poziva sistem da vrati listu pregleda. (APSO)
10. Sistem vraća formi listu pregelda. (IA)

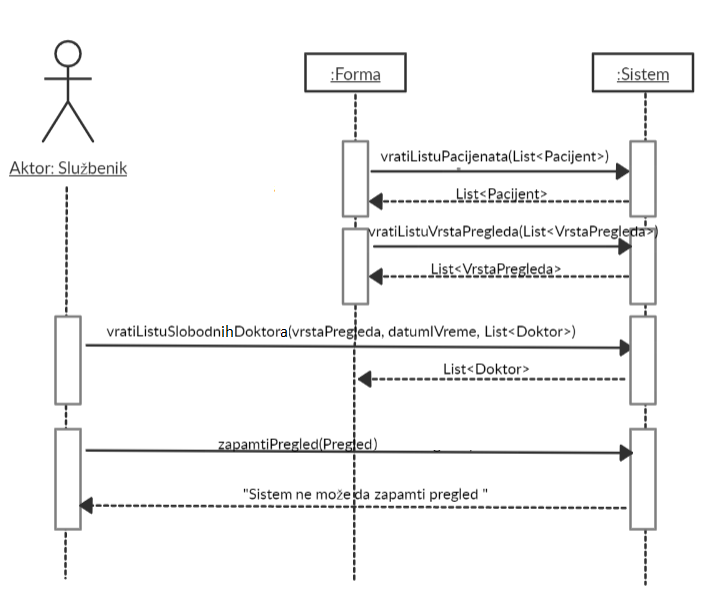


**Alternativna scenarija**

6.1 Ukoliko sistem ne može da pronađe slobodne doktore, on prikazuje službeniku poruku: “Sistem ne može da nađe slobodne doktore”. Prekida se izvršenje scenarija. (IA)



8.1 Ukoliko sistem ne može da zapamti pregled, on prikazuje službeniku poruku “Sistem ne može da zapamti pregled”. (IA)

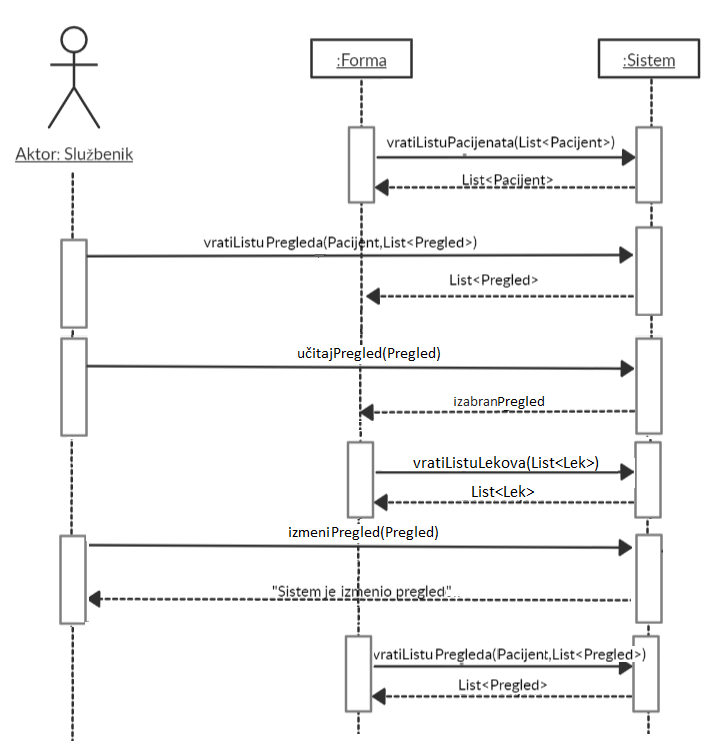


Sa navedenih sekvencnih dijagrama uočavaju se sistemske operacije koje treba projektovati:

1. signal VratiListuPacijenata (List<Pacijent>)
2. signal vratiListuVrstaPregleda (List<VrstaPregleda>)
3. signal VratiListuSlobodnihDoktora (vrstaPregleda, datumIVreme, List<Doktor>)
4. signal ZapamtiPregled (Pregled)
5. signal VratiListuPregleda (Pacijent, List<Pregled>)

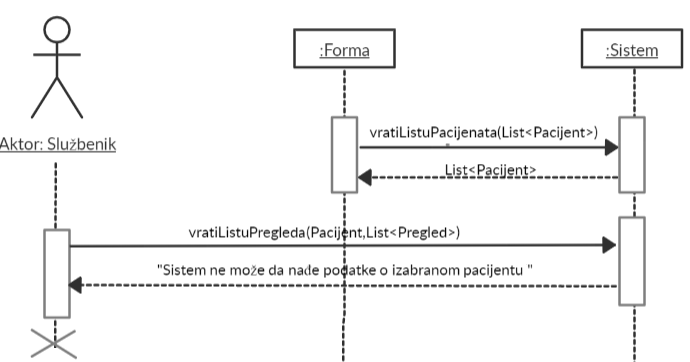
### 2.1.5 DS5: Dijagram sekvenci slučaja korišćenja- Izmena pregleda

1. Forma poziva sistem da vrati pacijente. (APSO)
2. Sistem vraća formi pacijente. (IA)
3. Službenik poziva sistem da nađe sve podatke (preglede) o izabranom pacijentu. (APSO)
4. Sistem prikazuje službeniku sve podatke o izabranom pacijentu (preglede). (IA)
5. Službenik poziva sistem da prikaže sve podatke o izabranom pregledu. (APSO)
6. Sistem prikazuje službeniku podatke o izabranom pregledu. (IA)
7. Forma poziva sistem da vrati listu lekova. (APSO)
8. Sistem vraća formi listu lekova. (IA)
9. Službenik poziva sistem da izmeni podatke o pregledu pacijenta. (APSO)
10. Sistem prikazuje službeniku poruku: “Sistem je izmenio pregled “. (IA)
11. Forma poziva sistem da vrati listu pregleda. (APSO)
12. Sistem vraća formi listu pregleda. (IA)

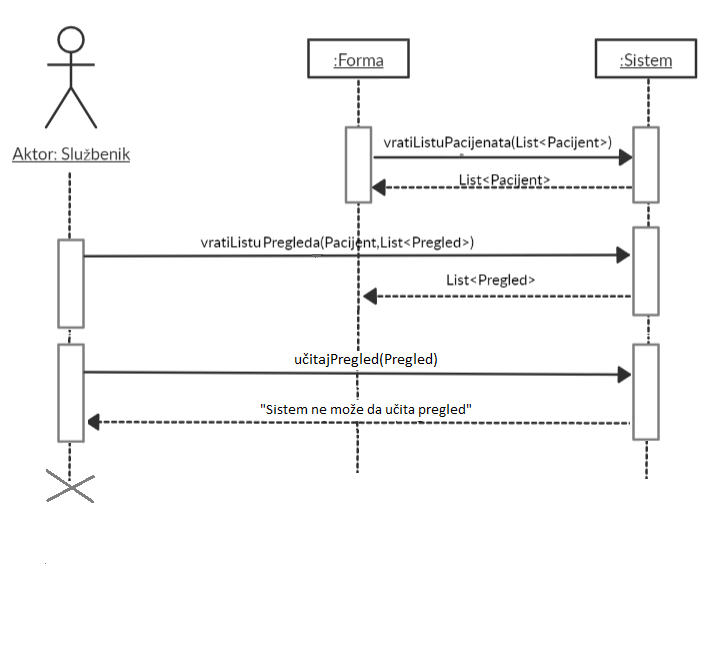


**Alternativna scenarija**

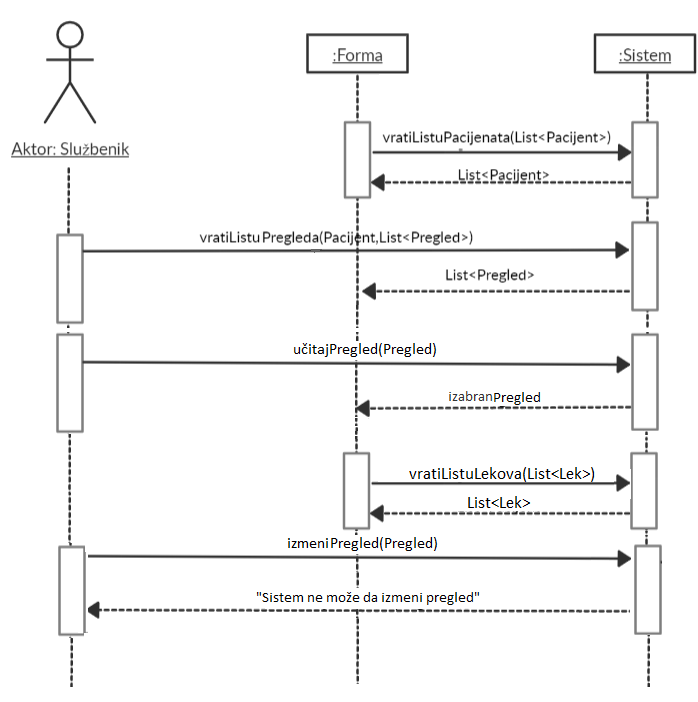
4.1 Ukoliko sistem ne može da nađe podatke o izabranom pacijentu on prikazuje službeniku poruku: “Sistem ne može da nađe podatke o izabranom pacijentu”. Prekida se izvršenje scenarija. (IA)



6.1 Ukoliko sistem ne može da učita podatke o izabranom pregledu, on korisniku prikazuje poruku: “Sistem ne može da učita pregled “. (IA)



10.1 Ukoliko sistem ne može da zapamti podatke o izmenjenom pregledu, on korisniku prikazuje poruku: “ Sistem ne može da izmeni pregled “. (IA)



Sa navedenih sekvencnih dijagrama uočavaju se sistemske operacije koje treba projektovati:

1. signal vratiListuPacijenata (List<Pacijent>)

2. signal vratiListuPregleda (Pacijent, List<Pregled>)

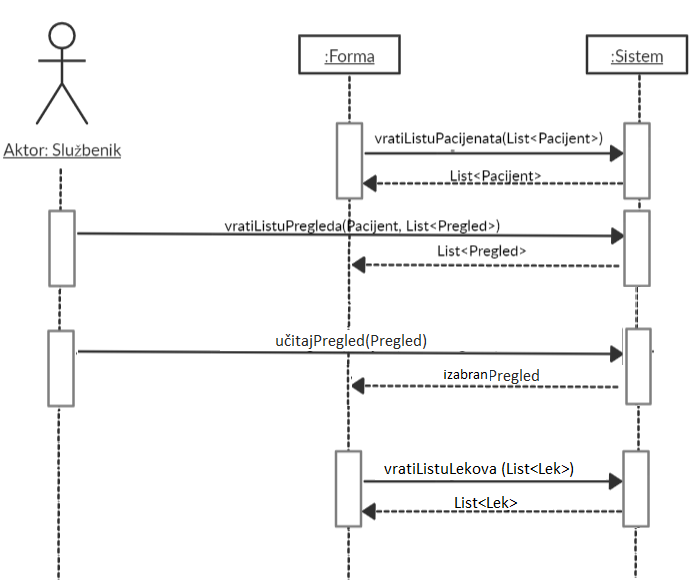
3. signal vratiListuLekova (List<Lek>)

4. signal učitajPregled (Pregled)

5. signal izmeniPregled (Pregled)

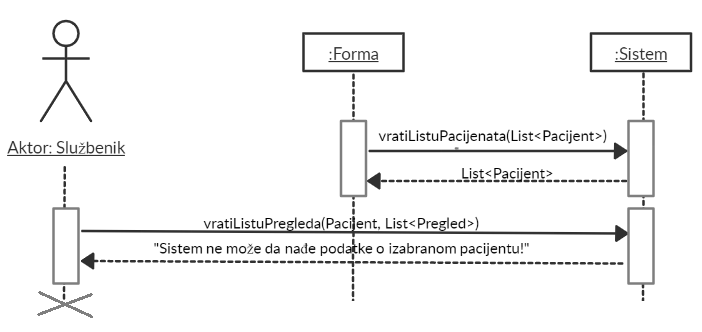
### 2.1.6 DS6: Dijagram sekvenci slučaja korišćenja- Pretraga pregleda

1. Forma poziva sistem da vrati listu pacijenata. (APSO)
2. Sistem vraća formi listu pacijenata. (IA)
3. Službenik poziva sistem da prikaže sve podatke (preglede) o izabranom pacijentu. (APSO)
4. Sistem prikazuje formi sve podatke o odabranom pacijentu (preglede). (IA)
5. Službenik poziva sistem da učita podatke o izabranom pregledu. (APSO)
6. Sistem prikazuje formi sve podatke o izabranom pregledu. (IA)
7. Forma poziva sistem da vrati listu lekova. (APSO)
8. Sistem vraća formi listu lekova. (IA)

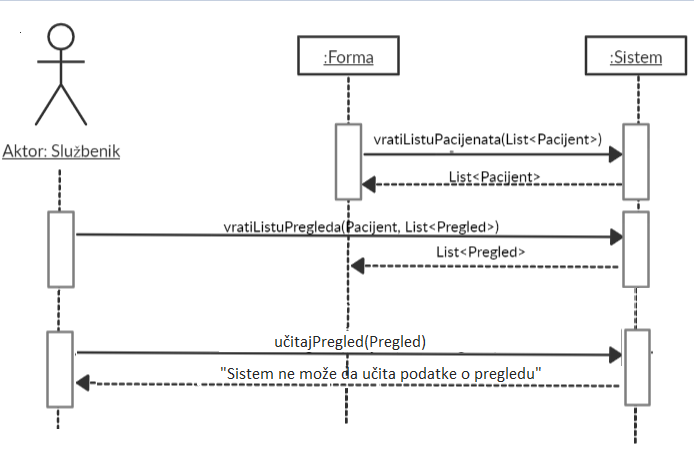


**Alternativni scenariji**

* 1. Ukoliko sistem ne može da nađe podatke o izabranom pacijentu, on službeniku prikazuje poruku: “Sistem ne može da nađe podatke o izabranom pacijentu “. (IA)



* 1. Ukoliko sistem ne može da nađe podatke o izabranom pregledu, on službeniku prikazuje poruku: “Sistem ne može da učita podatke o pregledu “. (IA)



Sa navedenih sekvencnih dijagrama uočavaju se sistemske operacije koje treba projektovati:

1. signal vratiListuPacijenata (List<Pacijent>)

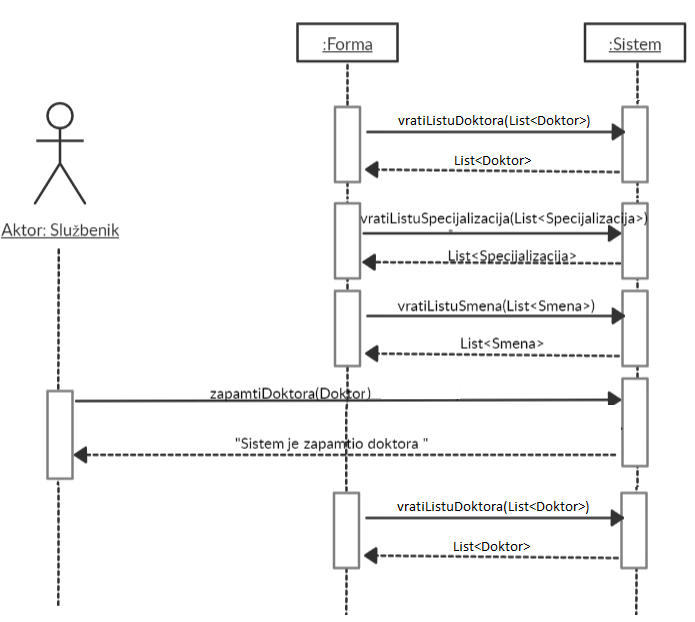
2. signal vratiListuPregleda (Pacijent, List<Pregled>)

3. signal učitajPregled (Pregled)

4. signal vratiListuLekova (List<Lek>)

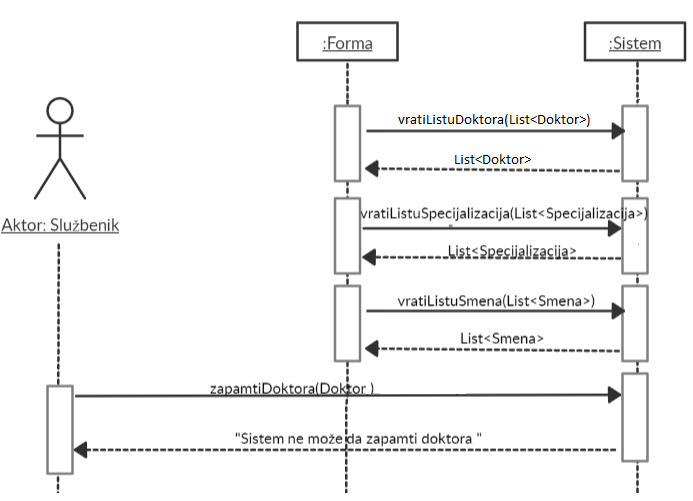
### 2.1.7 DS7: Dijagram sekvenci slučaja korišćenja- Unos novog doktora

1. Forma poziva sistem da vrati listu doktora. (APSO)
2. Sistem vraća formi listu doktora. (IA)
3. Forma poziva sistem da vrati listu specijalizacija. (APSO)
4. Sistem vraća formi listu specijalizacija. (IA)
5. Forma poziva sistem da vrati listu smena. (APSO)
6. Sistem vraća formi listu smena. (IA)
7. Službenik poziva sistem da zapamti podatke o doktoru. (APSO)
8. Sistem prikazuje službeniku poruku “Sistem je zapamtio doktora”. (IA)
9. Forma poziva sistem da vrati podatke o doktorima. (APSO)
10. Sistem vraća formi listu doktora. (IA)

****

**Alternativna scenarija**

8.1 Ukoliko sistem ne može zapamti doktora on prikazuje službeniku poruku: “Sistem ne može da zapamti doktora”. (IA)



Sa navedenih sekvencnih dijagrama uočavaju se sistemske operacije koje treba projektovati:

1. signal vratiListuSpecijalizacija (List<Specijalizacija>)

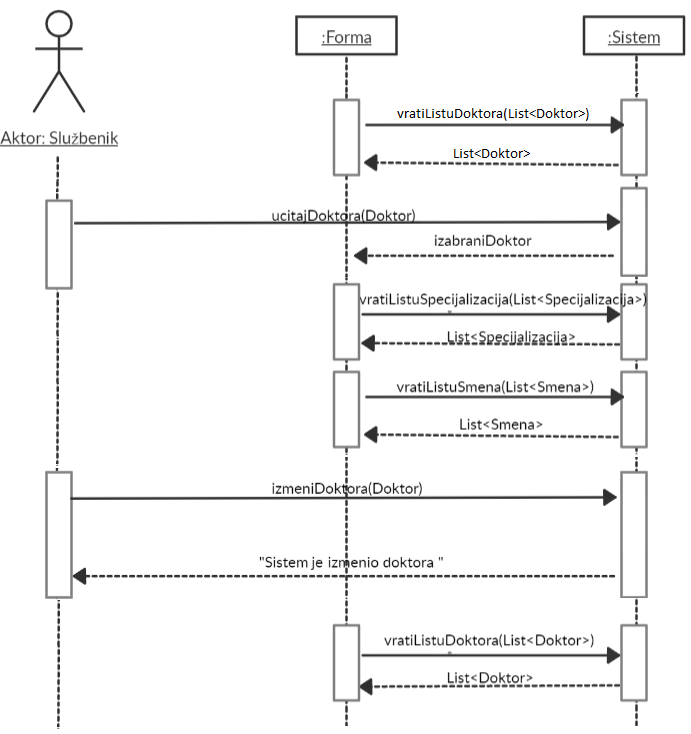
2. signal vratiListuSmena (List<Smena>)

3. signal zapamtiDoktora (Doktor)

4. signal vratiListuDoktora (List<Doktor>)

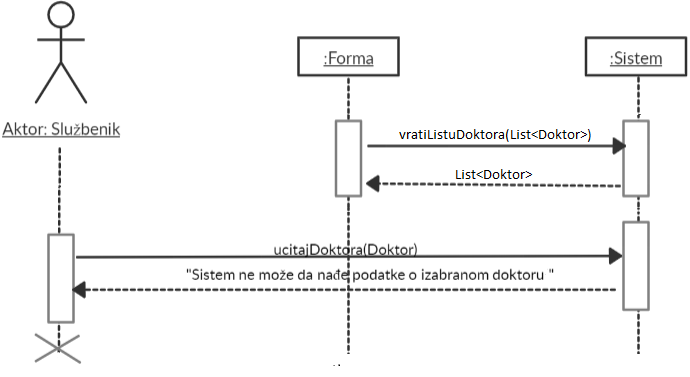
### 2.1.8 DS8: Dijagram sekvenci slučaja korišćenja- Izmena doktora

1. Forma poziva sistem da vrati listu doktora. (APSO)
2. Sistem vraća formi listu doktora. (IA)
3. Službenik poziva sistem da prikaže sve podatke o izabranom doktoru. (APSO)
4. Sistem prikazuje službeniku podatke o izabranom doktoru. (IA)
5. Forma poziva sistem da vrati listu specijalizacija. (APSO)
6. Sistem vraća formi listu specijalizacija. (IA)
7. Forma poziva sistem da vrati listu smena. (APSO)
8. Sistem vraća formi listu smena. (IA)
9. Službenik poziva sistem da izmeni podatke o doktoru. (APSO)
10. Sistem prikazuje službeniku i poruku: “Sistem je izmenio doktora “. (IA)
11. Forma poziva sistem da vrati listu doktora. (APSO)
12. Sistem vraća formi listu doktora. (IA)

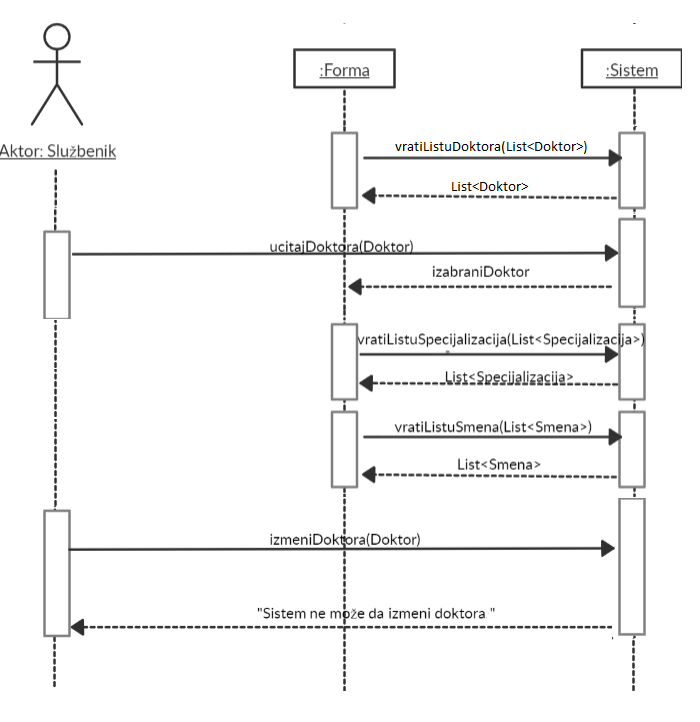


**Alternativna scenarija**

4.1 Ukoliko sistem ne može da nađe podatke o doktoru on službeniku prikazuje poruku: “Sistem ne može da nađe podatke o izabranom doktoru”. Prekida se izvršenje scenarija. (IA)



10.1 Ukoliko sistem ne može da izmeni podatke o doktoru on službeniku prikazuje poruku: “Sistem ne može da izmeni doktora” (IA)

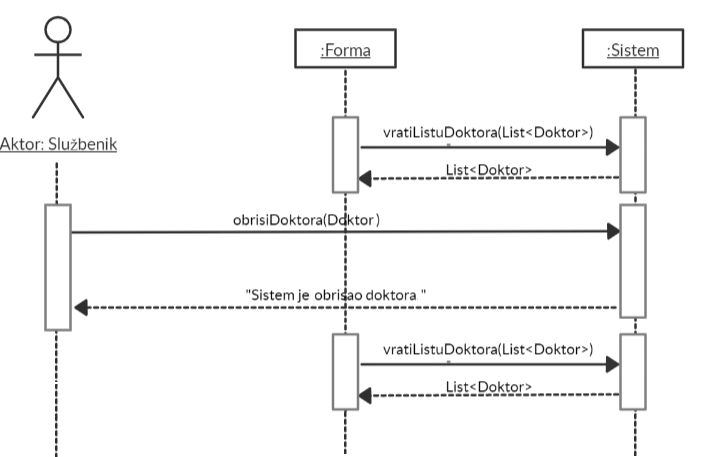


Sa navedenih sekvencnih dijagrama uočavaju se sistemske operacije koje treba projektovati:

1. signal vratiListuDoktora (List<Doktor>)
2. signal vratiListuSpecijalizacija (List<Specijalizacija>)
3. signal vratiListuSmena (List<Smena>)
4. signal ucitajDoktora (Doktor)
5. signal izmeniDoktora (Doktor)

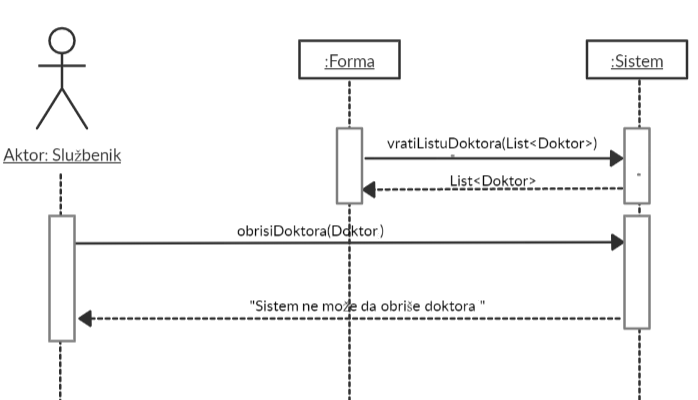
### 2.1.9 DS9: Dijagram sekvenci slučaja korišćenja- Brisanje doktora

1. Forma poziva sistem da vrati listu doktora. (APSO)
2. Sistem prikazuje formi listu doktora. (IA)
3. Službenik poziva sistem da obriše podatke o doktoru. (APSO)
4. Sistem prikazuje službeniku poruku: “Sistem je obrisao doktora”. (IA)
5. Forma poziva sistem da vrati listu doktora. (APSO)
6. Sistem vraća formi listu doktora. (IA)



**Alternativni scenariji**

* 1. Sistem prikazuje službeniku poruku: “Sistem ne može da obriše doktora”. (IA)

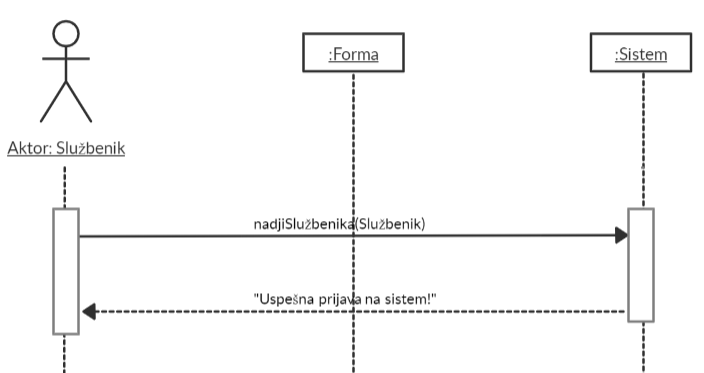


Sa navedenih sekvencnih dijagrama uočavaju se sistemske operacije koje treba projektovati:

1. signal vratiListuDoktora (List<Doktor>)
2. signal obrisiDoktora (Doktor)

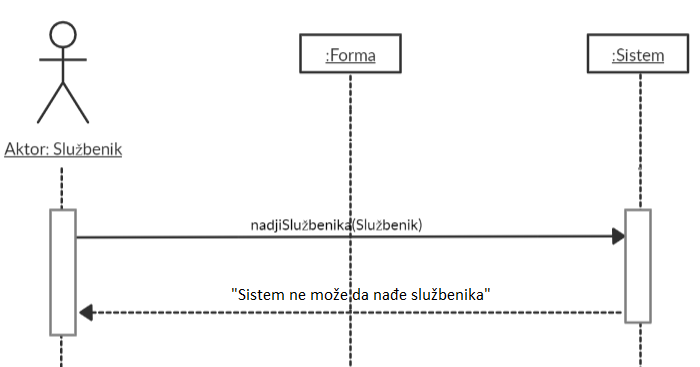
### 2.1.10 DS10: Dijagram sekvenci slučaja korišćenja- Logovanje službenika

1. Službenik poziva sistem da pronađe službenika sa zadatim podacima. (APSO)
2. Sistem prikazuje službeniku poruku: “Uspešna prijava na sistem!” i omogućava pristup sistemu. (IA)



**Alternativni scenariji**

* 1. Sistem prikazuje službeniku poruku: “Sistem ne može da nađe službenika”. (IA)



Sa navedenih sekvencnih dijagrama uočava se jedna sistemska operacija koju treba projektovati:

1. signal NađiSlužbenika (Službenik)

## 2.2 Sistemske operacije

Kao rezultat analize scenarija dobijeno je ukupno 20 sistemskih operacija koje treba projektovati:

1. signal zapamtiPacijenta (Pacijent)

2. signal vratiListuPacijenata (List<Pacijent>)

3. signal izmeniPacijenta (Pacijent)

4. signal obrisiPacijenta (Pacijent)

5. signal zapamtiDoktora (Doktor)

6. signal vratiListuDoktora (List<Doktor>)

7. signal vratiListuSlobodnihDoktora (VrstaPregleda, datumIVreme, List<Doktor>)

8. signal izmeniDoktora (Doktor)

9. signal obrisiDoktora (Doktor)

10. signal zapamtiPregled (Pregled)

11. signal vratiListuPregleda ((Pacijent/Doktor), List<Pregled>)

12. signal izmeniPregled (Pregled)

13. signal vratiListuSmena (List<Smena>)

14. signal vratiListuSpecijalizacija (List<Specijalizacija>)

15. signal vratiListuVrstaPregleda (List<VrstaPregleda>)

16. signal vratiListuLekova (List<Lek>)

17. signal nađiSlužbenika (Službenik)

18. signal učitajPacijenta (Pacijent)

19. signal učitajDoktora (Doktor)

20. signal učitajPregled (Pregled)

## 2.3 Ponašanje softverskog sistema – Definisanje ugovora o sistemskim operacijama

### 2.3.1 Ugovor UG1: ZapamtiPacijenta

Operacija: ZapamtiPacijenta (Pacijent): signal;

Veza sa SK: SK1

Preduslovi: Vrednosna i strukturna ograničenja nad objektom Pacijent moraju biti zadovoljena.

Postuslovi: Pacijent je zapamćen.

### 2.3.2 Ugovor UG2: VratiListuPacijenata

Operacija: VratiListuPacijenata (List<Pacijent>): signal;

Veza sa SK: SK1, SK2, SK3, SK4, SK5, SK6

Preduslovi: -

Postuslovi: -

### 2.3.3 Ugovor UG3: IzmeniPacijenta

Operacija: IzmeniPacijenta (Pacijent): signal;

Veza sa SK: SK2

Preduslovi: Vrednosna i strukturna ograničenja nad objektom Pacijent moraju biti zadovoljena.

Postuslovi: Pacijent je izmenjen.

### 2.3.4 Ugovor UG4: ObrišiPacijenta

Operacija: ObrišiPacijenta (Pacijent, List<Pregled>): signal;

Veza sa SK: SK3

Preduslovi: Strukturna ograničenja nad objektom Pacijent moraju biti zadovoljena.

Postuslovi: Pacijent je obrisan.

### 2.3.5 Ugovor UG5: ZapamtiDoktora

Operacija ZapamtiDoktora (Doktor): signal;

Veza sa SK: SK7

Preduslovi: Strukturna I vrednosna ograničenja nad objektom Doktor moraju biti zadovoljena.

Postuslovi: Doktor je zapamćen.

### 2.3.6 Ugovor UG6: VratiListuDoktora

Operacija: signal VratiListuDoktora (List<Doktor>): signal;

Veza sa SK: SK7, SK8, SK9

Preduslovi: -

Postuslovi: -

### 2.3.7 Ugovor UG7: VratiListuSlobodnihDoktora

Operacija: VratiListuSlobodnihDoktora (VrstaPregleda, DatumIVreme, List<Doktor>): signal;

Veza sa SK: SK4

Preduslovi: -

Postuslovi: -

### 2.3.8 Ugovor UG8: IzmeniDoktora

Operacija: IzmeniDoktora (Doktor): signal;

Veza sa SK: SK8

Preduslovi: Vrednosna I strukturna ograničenja nad objentom Doktor moraju biti zadovoljena.

Postuslovi: Doktor je izmenjen.

### 2.3.9 Ugovor UG9: ObrišiDoktora

Operacija: ObrišiDoktora (Doktor, List<Pregled>): signal;

Veza sa SK: SK9

Preduslovi: Strukturna ograničenja nad objektom Doktor moraju biti zadovoljena.

Postuslovi: Doktor je obrisan.

### 2.3.10 Ugovor UG10: ZapamtiPregled

Operacija: ZapamtiPregled (Pregled): signal;

Veza sa SK: SK4

Preduslovi: Vrednosna i strukturna ograničenja nad objektom Pregled moraju biti zadovoljena.

Postuslovi: Kreiran je novi Pregled.

### 2.3.11 Ugovor UG11: VratiListuPregleda

Operacija: VratiListuPregleda (Objekat, List<Pregled>): signal;

Veza sa SK: SK4, SK5, SK6

Preduslovi: -

Postuslovi: -

### 2.3.12 Ugovor UG12: IzmeniPregled

IzmeniPregled (Pregled)

Operacija IzmeniPregled (Pregled): signal;

Veza sa SK: SK5

Preduslovi: Strukturna I vrednosna ograničenja nad objektom Pregled su zadovoljena.

Postuslovi: Pregled je izmenjen.

### 2.3.13 Ugovor UG13: VratiListuSmena

Operacija: VratiListuSmena (List<Smena>): signal;

Veza sa SK: SK7, SK8

Preduslovi: -

Postuslovi: -

### 2.3.14 Ugovor UG14: VratiListuSpecijalizacija

Operacija: VratiListuSpecijalizacija (List<Specijalizacija>): signal;

Veza sa SK: SK7, SK8

Preduslovi: -

Postuslovi: -

### 2.3.15 Ugovor UG15: VratiListuVrstaPregleda

Operacija: VratiListuVrstaPregleda (List<VrstaPregleda>): signal;

Veza sa SK: SK4

Preduslovi: -

Postuslovi: -

### 2.3.16 Ugovor UG16: VratiListuLekova

Operacija: VratiListuLekova (List<Lek>): signal;

Veza sa SK: SK5, SK6

Preduslovi: -

Postuslovi: -

### 2.3.17 Ugovor UG17: NađiSlužbenika

Operacija: NađiSlužbenika (Službenik): signal;

Veza sa SK: SK10

Preduslovi: -

Postuslovi: -

### 2.3.18 Ugovor UG18: UčitajPacijenta

Operacija: UčitajPacijenta (Pacijent): signal;

Veza sa SK: SK2

Preduslovi: -

Postuslovi: -

### 2.3.19 Ugovor UG19: UčitajDoktora

Operacija: UčitajDoktora (Doktor): signal;

Veza sa SK: SK8

Preduslovi: -

Postuslovi: -

### 2.3.20 Ugovor UG20: UčitajPregled

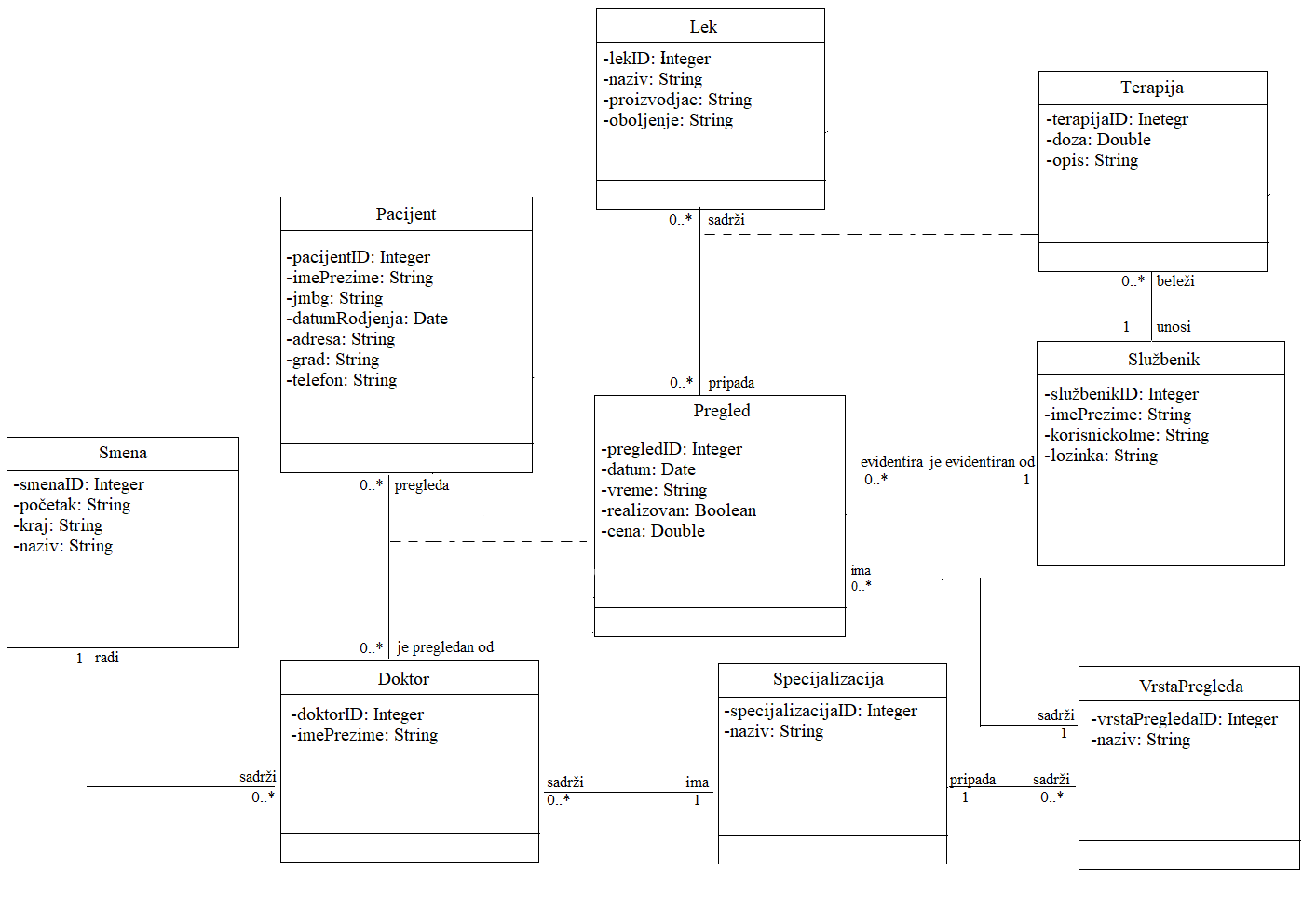
Operacija: UčitajPregled (Pregled): signal;

Veza sa SK: SK5, SK6

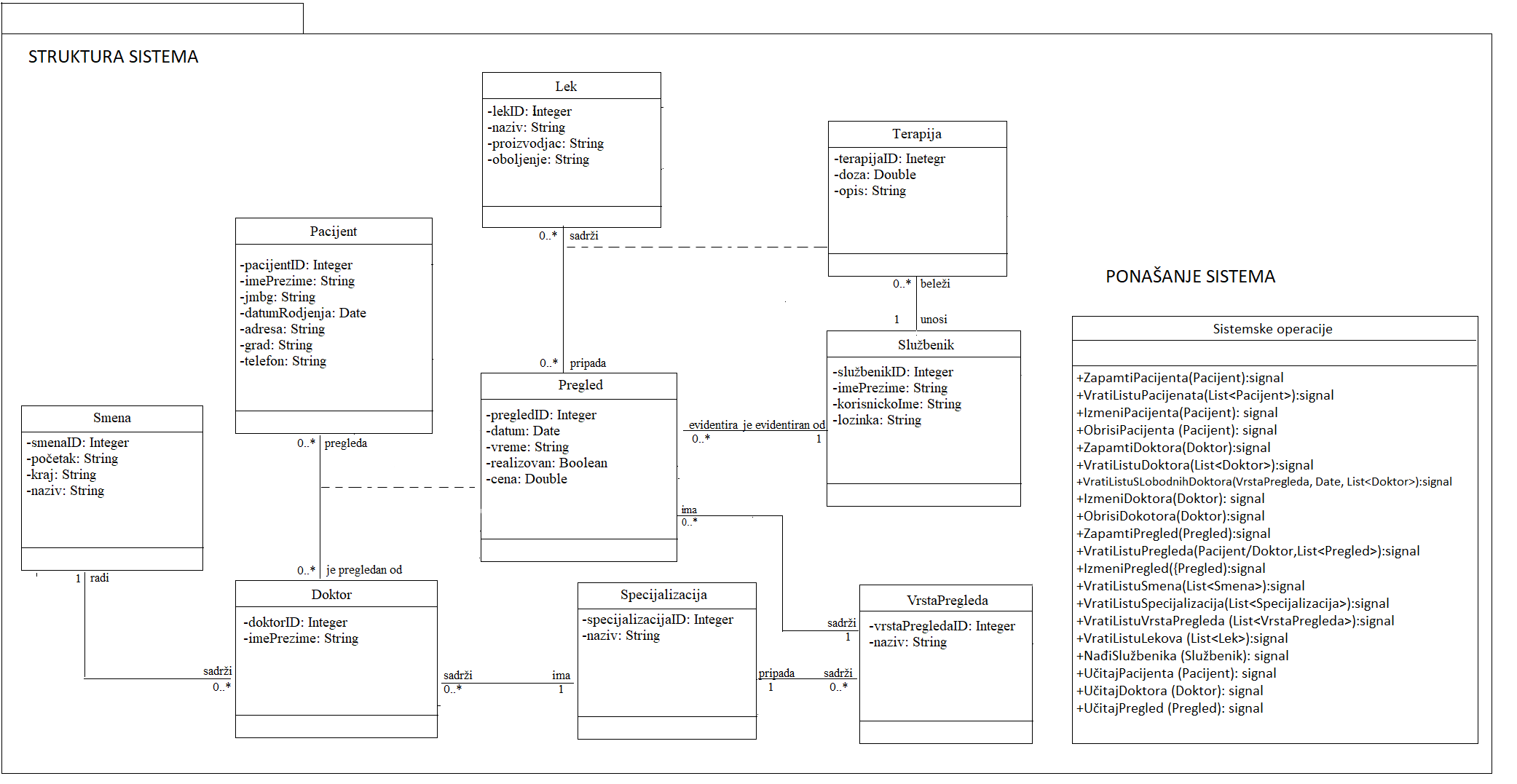
Preduslovi: -

Postuslovi: -

## 2.4 Struktura softverskog sistema-Konceptualni model (Domenski model)

****

Kao rezultat analize SK i pravljenja konceptualnog modela, dobija se logička struktura i ponašanje softverskog sistema:



## 2.5 Struktura softverskog sistema – Relacioni model

Pacijent (pacijentID, imePrezime, jmbg, datumRodjenja, adresa, grad, telefon)

Doktor (doktorID, imePrezime, *specijalizacijaID, smenaID*)

Službenik (službenikID, imePrezime, korisnickoIme, lozinka)

Pregled (pregledID, datum, vreme, realizovan, cena, *vrstaPregledaID, doktorID, pacijentID*)

VrstaPregleda (vrstaPregledaID, naziv, cena, *specijalizacijaID*)

Specijalizacija (specijalizacijaID, naziv)

Smena (smenaID, početak, kraj, naziv)

Lek (lekID, naziv, proizvodjac, oboljenje)

Terapija (terapijaID, doza, opis, *lekID, pregledID, sluzbenikID*)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tabela Pacijent** | | **Prosto vrednosno ograničenje** | | **Složeno vrednosno ograničenje** | | **Strukturno ograničenje** |
| **Atributi** | **Ime**  **atributa** | **Tip atributa** | **Vrednost atributa** | **Međuzav. atributa jedne tabele** | **Međuzav. atributa više tabela** | INSERT/  UPDATE  CASCADES  Pregled  DELETE RESTRICTED  Pregled |
| pacijentID | Integer | not null and >0 |  |  |
| imePrezime | String | not null |  |  |
| jmbg | String | not null |  |  |
| datumRodjenja | Date | not null |  |  |
| Adresa | String | not null |  |  |
| grad | String | not null |  |  |
| telefon | String | not null |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tabela Doktor** | | **Prosto vrednosno ograničenje** | | **Složeno vrednosno ograničenje** | | **Strukturno ograničenje** |
| **Atributi** | **Ime**  **atributa** | **Tip atributa** | **Vrednost atributa** | **Međuzav. atributa jedne tabele** | **Međuzav. atributa više tabela** | INSERT RESTRICTED  Specijalizacija, Smena  UPDATE RESTRICTED  Specijalizacija, Smena  UPDATE  CASCADES  Pregled  DELETE RESTRICTED  Pregled |
| doktorID | Integer | not null and >0 |  |  |
| imePrezime | String | not null |  |  |
| specijalzacijaID | Integer | not null and >0 |  |  |
| smenaID | Integer | not null and >0 |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tabela Službenik** | | **Prosto vrednosno ograničenje** | | **Složeno vrednosno ograničenje** | | **Strukturno ograničenje** |
| **Atributi** | **Ime**  **atributa** | **Tip atributa** | **Vrednost atributa** | **Međuzav. atributa jedne tabele** | **Međuzav. atributa više tabela** | INSERT /  UPDATE  CASCADES  Pregled  DELETE  RESTRICTED  Pregled |
| službenikID | Integer | not null and >0 |  |  |
| imePrezime | String | not null |  |  |
| korisnickoIme | String | not null |  |  |
| lozinka | String | not null |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tabela Pregled** | | **Prosto vrednosno ograničenje** | | **Složeno vrednosno ograničenje** | | **Strukturno ograničenje** |
| **Atributi** | **Ime**  **atributa** | **Tip atributa** | **Vrednost atributa** | **Međuzav. atributa jedne tabele** | **Međuzav. atributa više tabela** | INSERT RESTRICTED  Pacijent, Doktor, VrstaPregleda,  Službenik  UPDATE RESTRICTED  Pacijent, Doktor VrstaPregleda,  Službenik  DELETE / |
| pregledID | Integer | not null and >0 |  |  |
| datum | Date | not null |  |  |
| vreme | String | not null |  |  |
| realizovan | TinyInteger | not null |  |  |
| cena | Double | not null and >=0 |  |  |
| vrstaPregledaID | Integer | not null and >0 |  |  |
| doktorID | Integer | not null and >0 |  |  |
| pacijentID | Integer | not null and >0 |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tabela VrstaPregleda** | | **Prosto vrednosno ograničenje** | | **Složeno vrednosno ograničenje** | | **Strukturno ograničenje** |
| **Atributi** | **Ime**  **atributa** | **Tip atributa** | **Vrednost atributa** | **Međuzav. atributa jedne tabele** | **Međuzav. atributa više tabela** | INSERT RESTRICTED  Specijalizacija  UPDATE RESTRICTED  Specijalizacija  UPDATE CASCADES  Pregled  DELETE  RESTRICTED  Pregled |
| vrstaPregledaID | Integer | not null and >0 |  |  |
| naziv | String | not null |  |  |
| cena | Double | not null and >=0 |  |  |
| specijalizacijaID | Integer | not null and >0 |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tabela Specijalizacija** | | **Prosto vrednosno ograničenje** | | **Složeno vrednosno ograničenje** | | **Strukturno ograničenje** |
|  | specijalizacijaID | Integer | not null and >0 |  |  |  |
| naziv | String | not null |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tabela Smena** | | **Prosto vrednosno ograničenje** | | **Složeno vrednosno ograničenje** | | **Strukturno ograničenje** |
| **Atributi** | **Ime**  **atributa** | **Tip atributa** | **Vrednost atributa** | **Međuzav. atributa jedne tabele** | **Međuzav. atributa više tabela** | INSERT /  UPDATE CASCADES  Doktor  DELETE  RESTRICTED  Doktor |
| smenaID | Integer | not null and >0 |  |  |
| pocetak | String | not null |  |  |
| kraj | String | not null |  |  |
| naziv | String | not null |  |  |
|  | | | | | | |
| **Tabela Lek** | | **Prosto vrednosno ograničenje** | | **Složeno vrednosno ograničenje** | | **Strukturno ograničenje** |
| **Atributi** | **Ime**  **atributa** | **Tip atributa** | **Vrednost atributa** | **Međuzav. atributa jedne tabele** | **Međuzav. atributa više tabela** | INSERT /  UPDATE CASCADES  Terapija  DELETE  RESTRICTED  Terapija |
| lekID | Integer | not null and >0 |  |  |
| naziv | String | not null |  |  |
| proizvodjac | String | not null |  |  |
| oboljenje | String | not null |  |  |

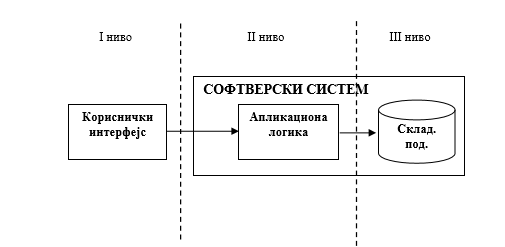
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tabela Terapija** | | **Prosto vrednosno ograničenje** | | **Složeno vrednosno ograničenje** | | **Strukturno ograničenje** |
| **Atributi** | **Ime**  **atributa** | **Tip atributa** | **Vrednost atributa** | **Međuzav. atributa jedne tabele** | **Međuzav. atributa više tabela** | INSERT RESTRICTED  Pregled, Lek,  Službenik  UPDATE RESTRICTED  Pregled, Lek,  Službenik  DELETE / |
| terapijaID | Integer | not null and >0 |  |  |
| doza | Double | not null and >0 |  |  |
| opis | String | not null |  |  |
| pregledID | Integer | not null |  |  |
| lekID | Integer | not null |  |  |
| sluzbenikID | Integer | not null |  |  |

# 3. Projektovanje

## 3.1 Arhitektura softverskog Sistema

Tronivojska arhitektura softverskog Sistema satoji se iz:

1. Korisničkog interfejsa
2. Aplikacione logike
3. Skladišta podataka



## 3.2 Projektovanje korisničkog interfejsa

Korisnički interfejs, shodno Def. Laz1, predstavlja realizaciju ulaza i/ili izlaza softverskog sistema. [1]

Korisnički interfejs se sastoji od:

1. Ekranske forme koja je odgovorna da:

a) prihvata podatke koje unosi aktor

b) prihvata događaje koje pravi aktor

c) poziva kontrolera grafičkog интерфејса, prosleđujući mu prihvaćene podatke

d) prikazuje podatke kje je dobio od kontrolera grafičkog interfejsa

### 3.2.1 Projektovanje korisničkog interfejsa

**SK1: Slučaj korišćenja- Unos novog pacijenta**

**Naziv SK:**

Unošenje novog pacijenta

**Aktori SK:**

Službenik

**Učesnici SK:**

Službenik i sistem (program)

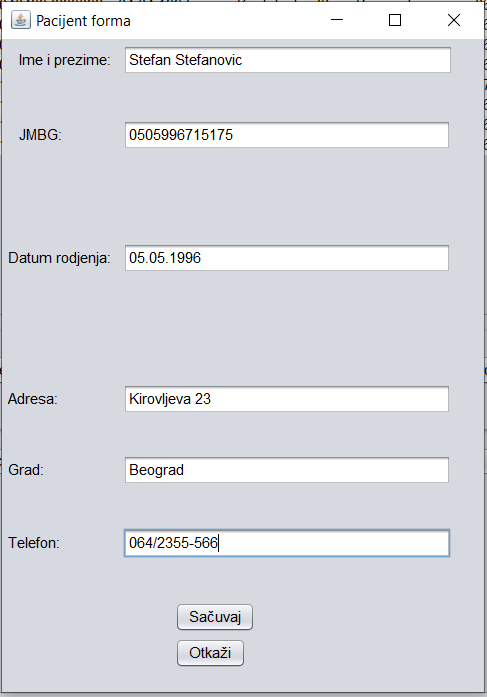
**Preduslov:** Sistem je uključen i službenik je ulogovan pod svojom šifrom. Sistem prikazuje formu za rad sa pacijentom.

Opis: Klikom na dugme Dodaj pacijenta otvara se forma za rad sa pacijentom



**Osnovni scenario SK**

1.Službenik unosi podatke o pacijentu. (APUSO)



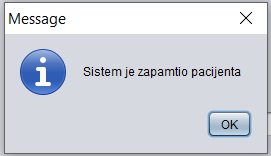
2.Službenik kontroliše da li je korektno uneo podatke u pacijenta. (ANSO)

3.Službenik poziva sistem da zapamti podatke o pacijentu. (APSO).

Opis: Klikom na dugme Sačuvaj poziva se SO ZapamtiPacijenta (Pacijent) i forma za rad sa pacijentima poziva SO VratiListuPacijenata (List<Pacijent>)

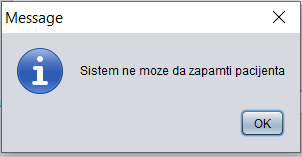
4.Sistem pamti podatke o pacijentu. (SO)

5.Sistem prikazuje službeniku poruku: „Sistem je zapamtio pacijenta “(IA)



**Alternativni scenariji**

* 1. Ukoliko sistem ne može da zapamti podatke o pacijentu, on prikazuje službeniku poruku: “Sistem ne može da zapamti pacijenta”. (IA)



**SK2: Slučaj korišćenja- Izmena pacijenta**

**Naziv SK**

Izmena pacijenta

**Autori SK**

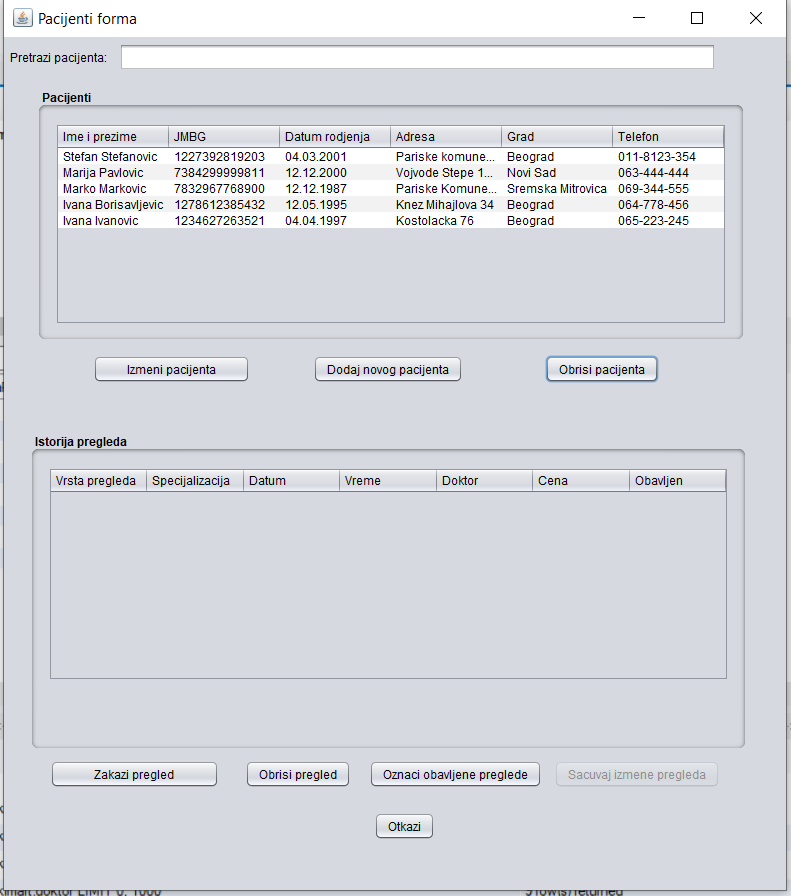
Službenik

**Učesnici SK**

Službenik i sistem (program)

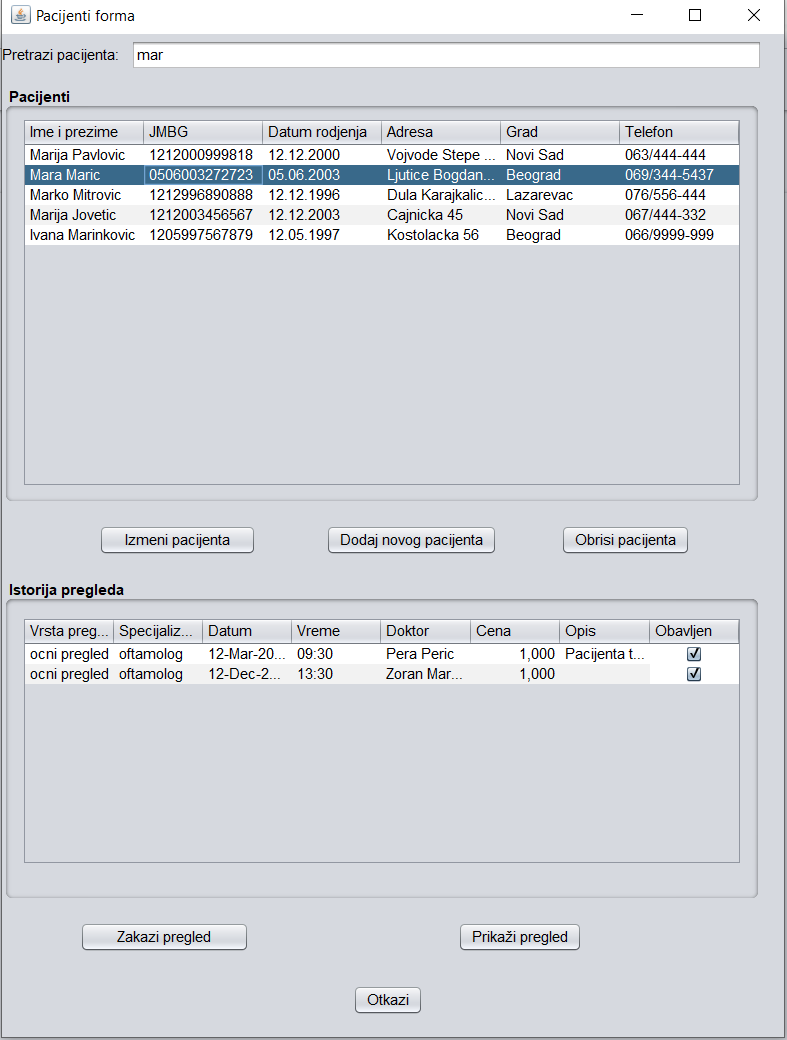
**Preduslov:** Sistem je uključen i službenik je ulogovan pod svojom šifrom. Učitana je lista pacijenata. Sistem prikazuje formu za rad sa pacijentima.

Opis: Klikom na stavku menija Pacijent->Pretraži pacijente poziva se SO VratiListuPacijenata (List<Pacijent>) i otvara se forma za rad sa pacijentima.



**Osnovni scenario SK**

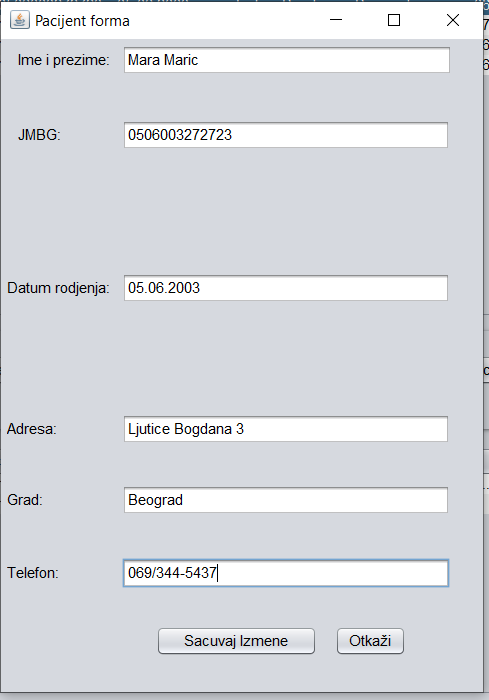
1. Službenik bira pacijenta. (APUSO)



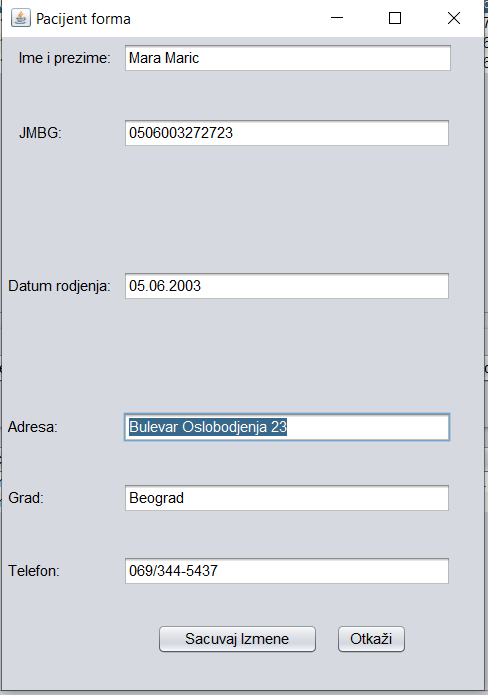
1. Službenik poziva sistem da prikaže sve podatke o izabranom pacijentu. (APSO)

Opis: Klikom na dugme Izmeni pacijenta poziva se SO UčitajPacijenta (Pacijent) i otvara se forma za izmenu pacijenta.

1. Sistem traži podatke o izabranom pacijentu. (SO)
2. Sistem prikazuje službeniku podatke o odabranom pacijentu. (IA)



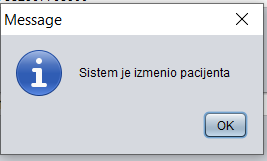
1. Službenik unosi (menja) podatke o pacijentu. (APUSO)



1. Službenik kontroliše da li je korektno uneo podatke o pacijentu. (ANSO)
2. Službenik poziva sistem da izmeni podatke o pacijentu. (APSO)

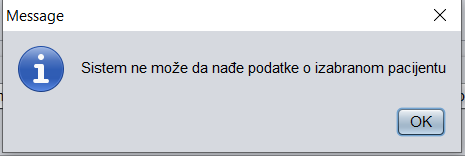
Opis: Klikom na dugme Sačuvaj izmene poziva se SO IzmeniPacijenta (Pacijent) I forma za rad sa pacijentima poziva SO VratiListuPacijenata (List<Pacijent>)

1. Sistem menja podatke o pacijentu. (SO)
2. Sistem prikazuje službeniku i poruku: “Sistem je izmenio pacijenta “ . (IA)

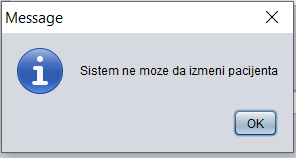


**Alternativni scenariji**

4.1 Ukoliko sistem ne može da nađe podatke o pacijentu, on službeniku pikazuje poruku: “Sistem ne može da nađe podatke o izabranom pacijentu”. Prekida se izvršenje scenarija. (IA)



9.1 Ukoliko sistem ne može da izmeni podatke o pacijentu, on prikazuje službeniku poruku: „Sistem ne može da izmeni pacijenta “. (IA)



**SK3: Slučaj korišćenja- Brisanje pacijenta**

**Naziv SK**

Brisanje pacijenta

**Autori SK**

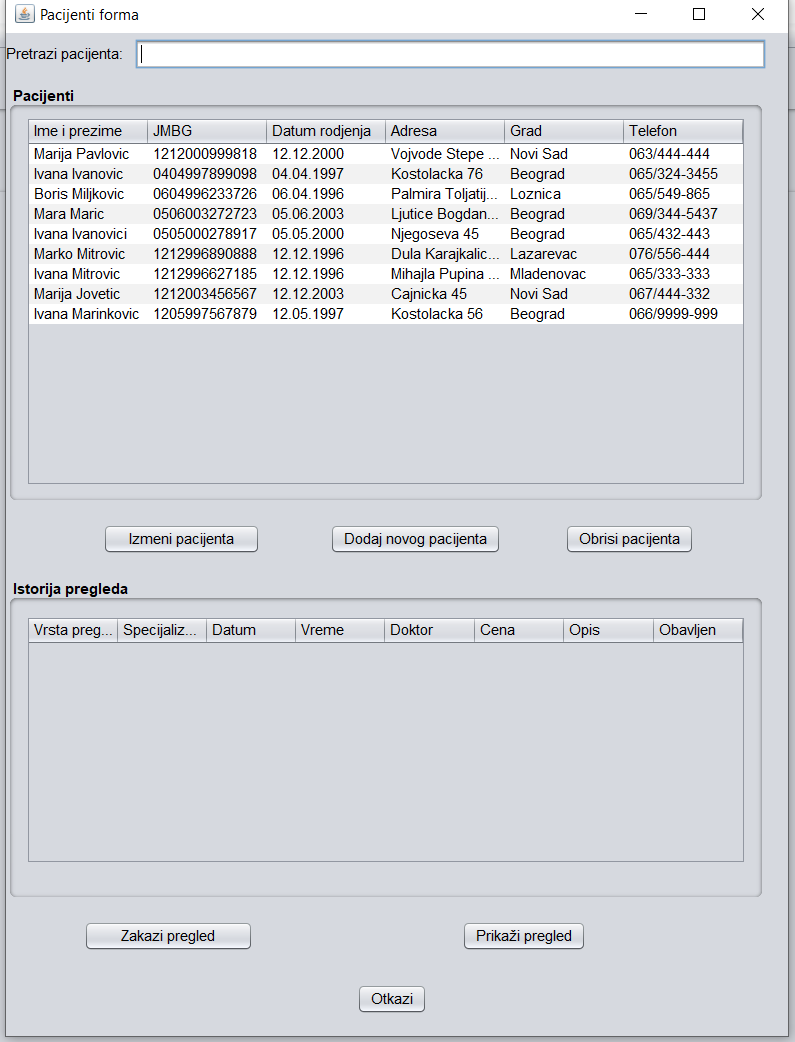
Službenik

**Učesnici SK**

Službenik i sistem (program)

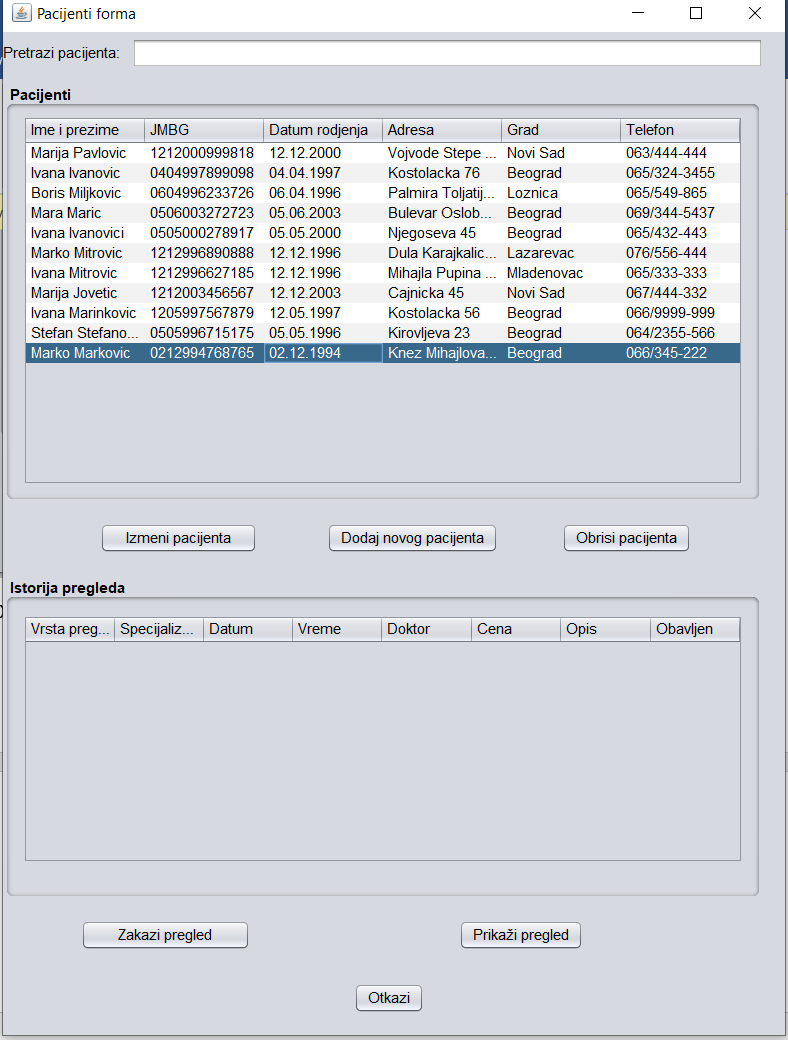
**Preduslov:** Sistem je uključen i službenik je ulogovan pod svojom šifrom. Učitana je lista pacijenata. Sistem prikazuje formu za rad sa pacijentima.

Opis: Klikom na stavku menija Pacijenti->Pretraži pacijente poziva se SO VratiListuPacijenata (List<Pacijent>) i otvara se forma za rad sa pacijentima.



**Osnovni scenario SK**

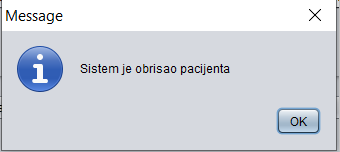
1. Službenik bira pacijenta kojeg želi da obriše. (APUSO)



1. Službenik poziva sistem da obriše podatke o pacijentu. (APSO)

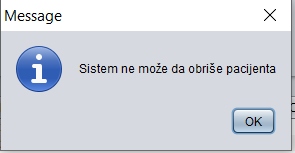
Opis: Klikom na dugme Obrisi pacijenta poziva se SO ObrisiPacijenta (Pacijent) I forma za rad sa pacijentima poziva SO VratiListuPacijenata (List<Pacijent>)

1. Sistem briše pacijenta i pamti izmene. (SO)
2. Sistem prikazuje službeniku poruku: “Sistem je obrisao pacijenta “. (IA)



**Alternativni scenariji**

* 1. Ukoliko sistem ne može da obriše pacijenta, on prikazuje službeniku poruku: „Sistem ne može da obriše pacijenta “. (IA)



**SK4: Slučaj korišćenja: Zakazivanje pregleda**

**Naziv SK**

Zakazivanje pregleda

**Aktori SK**

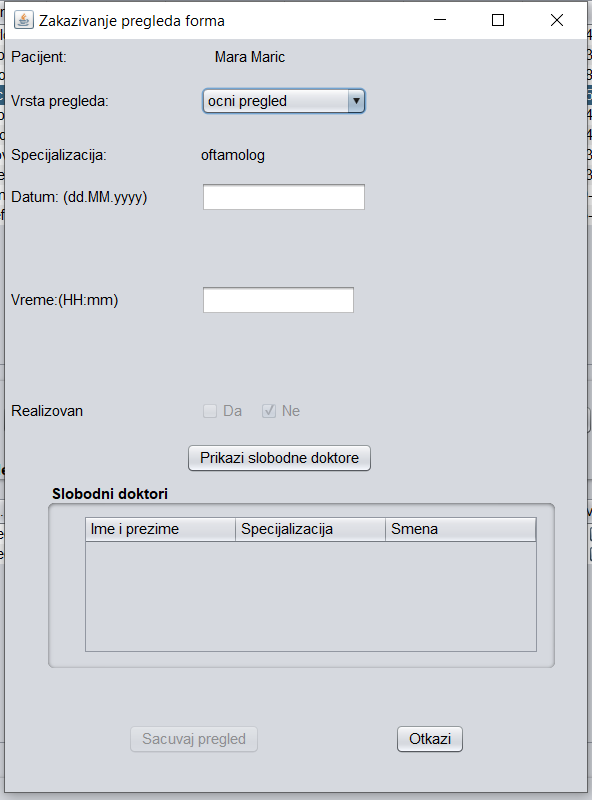
Službenik

**Učesnici SK**

Službenik i sistem (program)

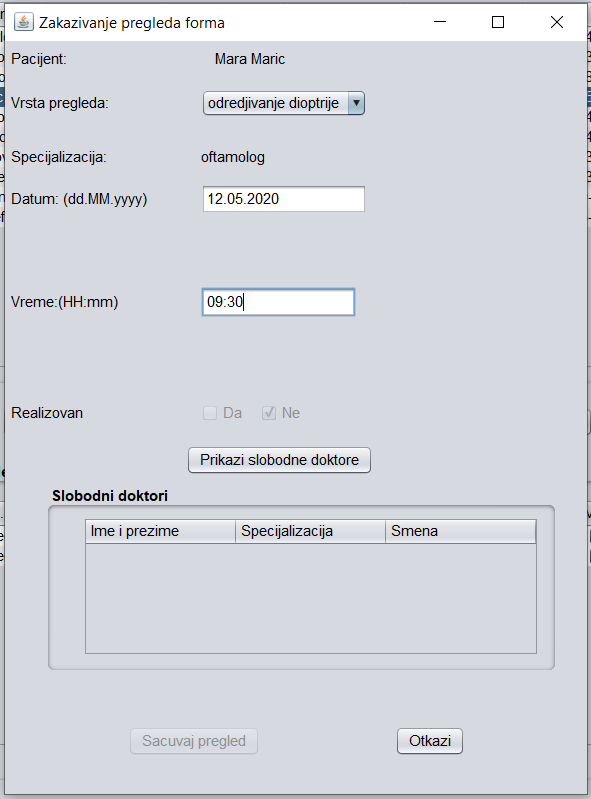
Preduslov: Sistem je uključen i službenik je ulogovan pod svojom šifrom. Odabran je pacijent kome želimo da zakažemo pregled i sistem prikazuje formu za rad sa pregledom.

Opis: Klikom na dugme Zakazi pregled otvara se forma za rad sa pregledom i ona poziva SO VratiListuVrstaPregleda(List<VrstaPregleda>)



**Osnovni scenario SK**

1. Službenik unosi podatke o pregledu. (APUSO)



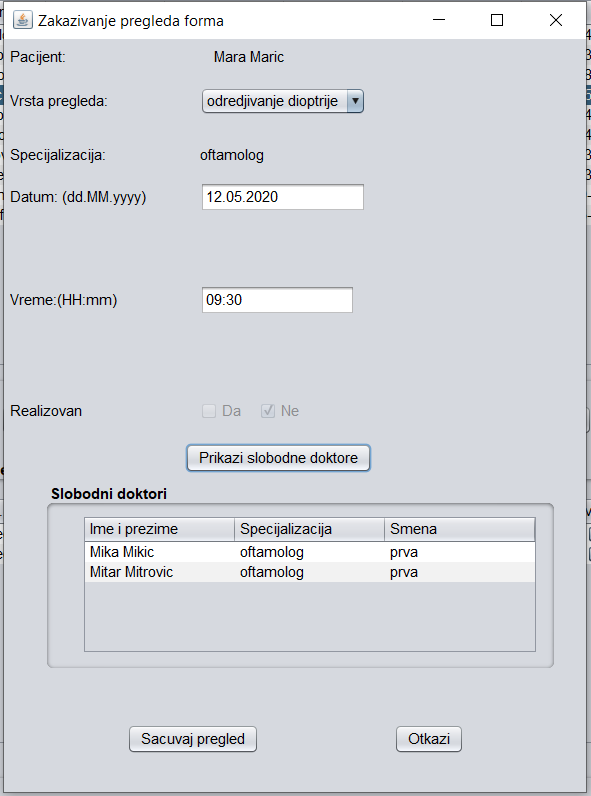
2. Službenik kontroliše da li je korektno uneo podatke o pregledu. (ANSO)

3. Službenik poziva sistem da nađe podatke o slobodnim doktorima. (APSO)

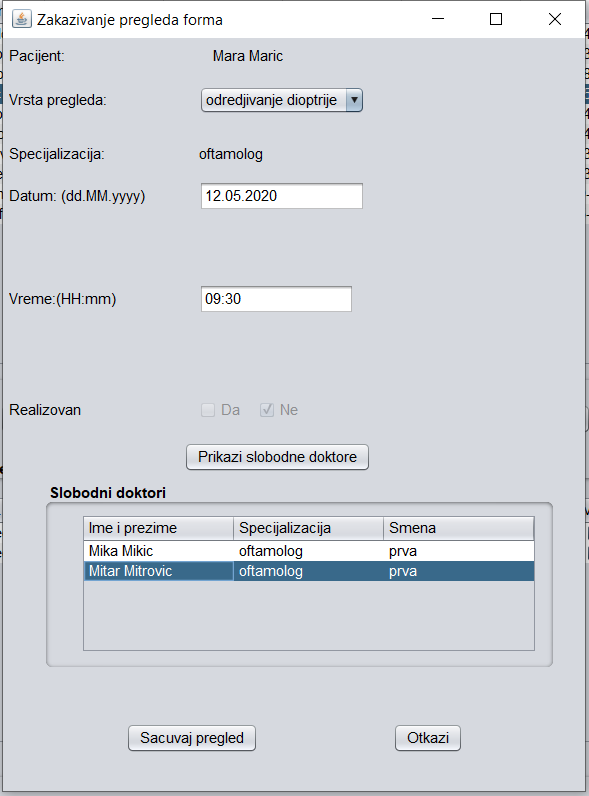
Opis: Klikom na dugme Prikaži slobodne doktore poziva se SO VratiListuSlobodnihDoktora (VrstaPregleda, DatumIVreme, List<Doktor>)

4. Sistem traži slobodne doktore. (SO)

5. Sistem prikazuje službeniku pronađene slobodne doktore. (IA)



6. Službenik bira slobodnog doktora. (APUSO)

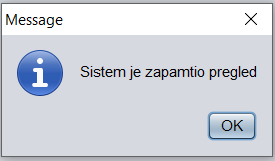


7. Službenik poziva sistem da zapamti podatke o pregledu. (APSO)

Opis: Klikom na dugme Sačuvaj pregled poziva se SO ZapamtiPregled (Pregled) i forma poziva SO VratiListuPregleda (Pacijent)

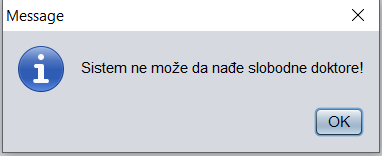
8. Sistem pamti podatke o pregledu. (SO)

9. Sistem prikazuje službeniku poruku: “Sistem je zapamtio pregled “. (IA)

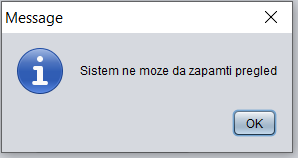


**Alternativna scenarija**

* 1. Ukoliko sistem ne može da pronađe slobodne doktore, on prikazuje službeniku poruku: Sistem ne može da nađe slobodne doktore!”. Prekida se izvršenje scenarija. (IA)



9.1 Ukoliko sistem ne može da zapamti pregled on prikazuje službeniku poruku: “Sistem ne može da zapamti pregled”. (IA)



**SK 5: Slučaj korišćenja - Izmena pregleda (Složen)**

**Naziv SK**

Izmena pregleda

**Aktori SK**

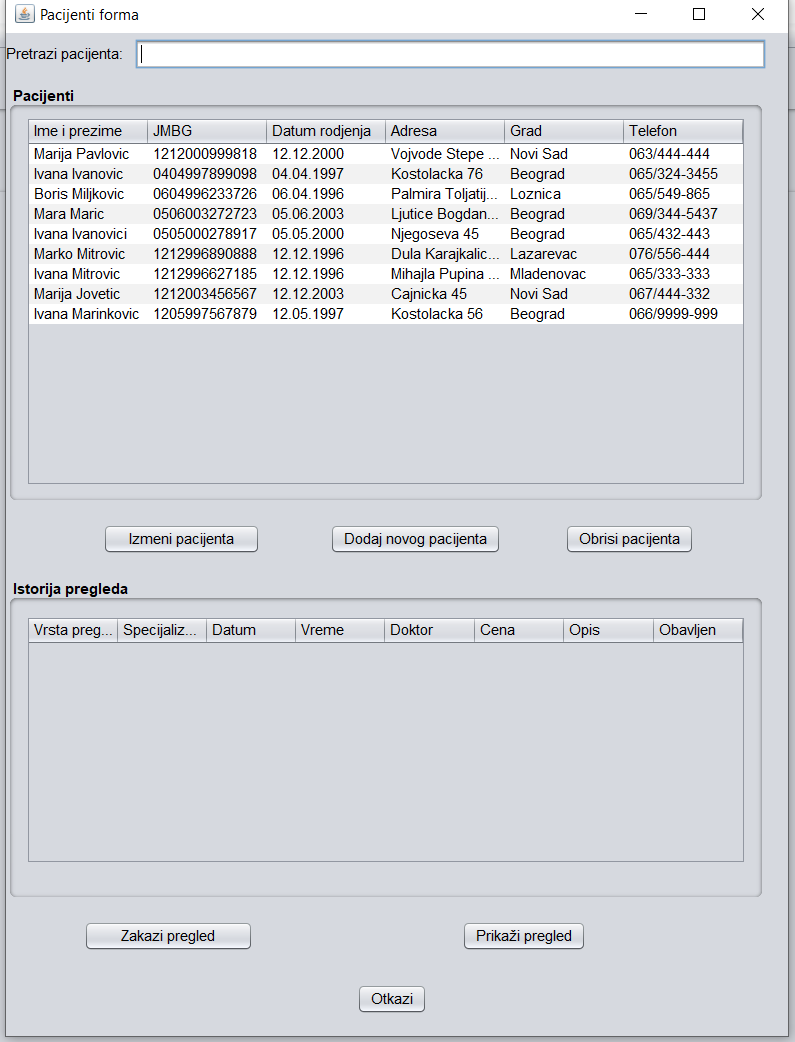
Službenik

**Učesnici SK**

Službenik i sistem (program)

**Preduslov**: Sistem je uključen i službenik je ulogovan pod svojom šifrom. Učitana je lista pacijenata. Sistem prikazuje formu za rad sa pacijentima.

Opis: Klikom na stavku menija Pacijenti->Pretraži pacijente poziva se SO VratiListuPacijenata (List<Pacijent>)

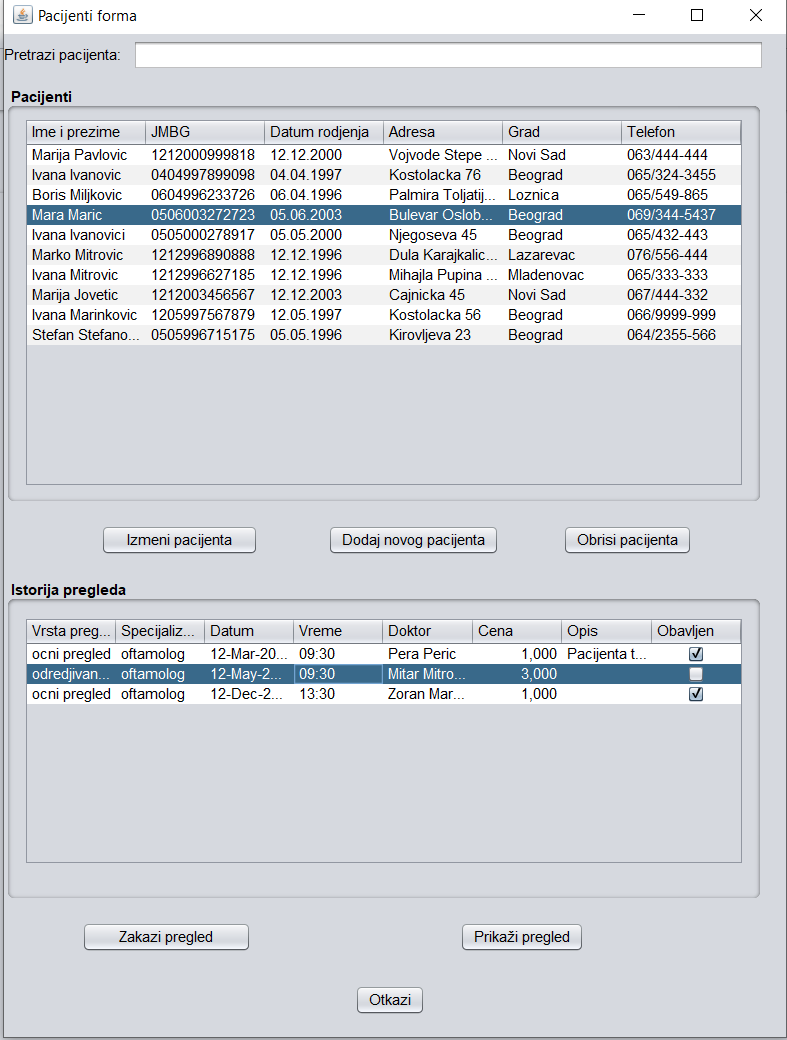


**Osnovni scenario SK**

1. Službenik bira pacijenta. (APUSO)
2. Službenik poziva sistem da nađe sve podatke (preglede) o izabranom pacijentu. (APSO)

Opis: Klikom na pacijenta poziva se SO VratiListuPregleda (Pacijent, List<Pregled>)

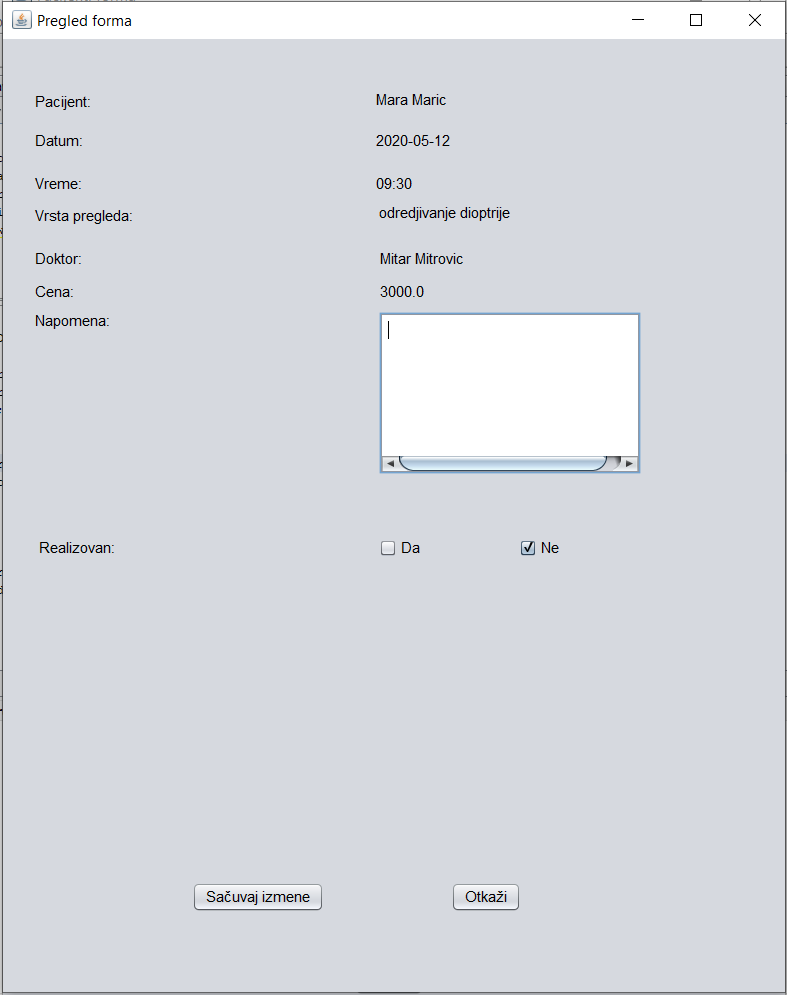
1. Sistem traži podatke o izabranom pacijentu. (SO)
2. Sistem prikazuje službeniku podatke o izabranom pacijentu (preglede). (IA)
3. Službenik bira pregled. (APUSO)



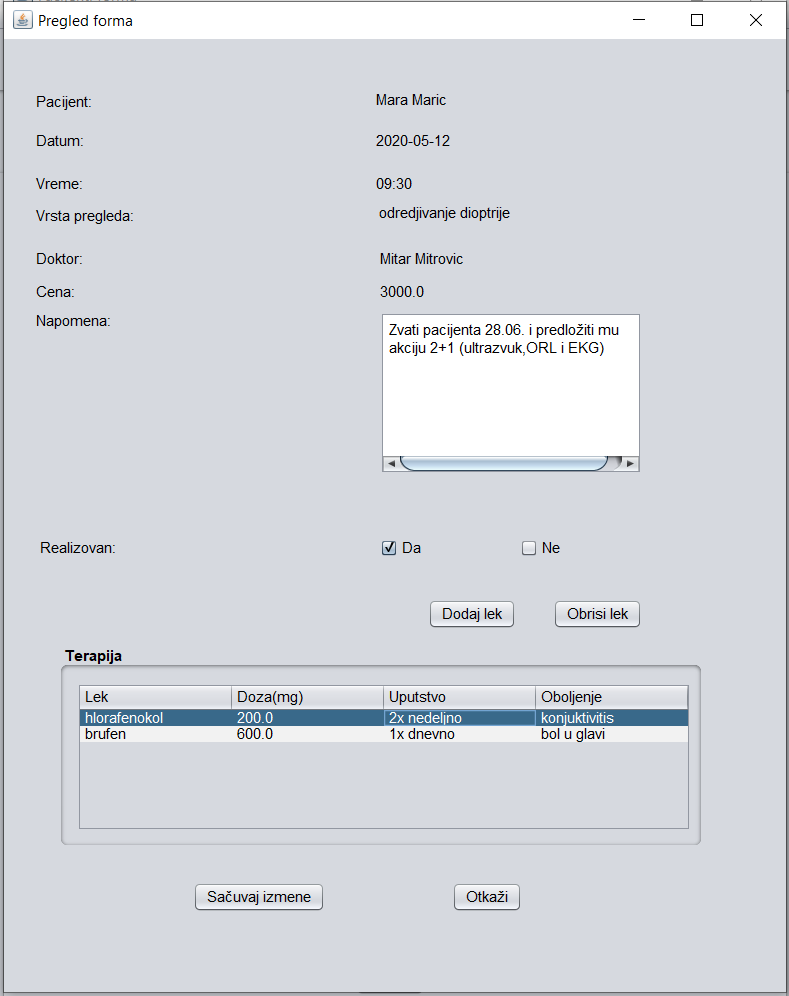
1. Službenik poziva sistem da prikaže sve podatke o izabranom pregledu. (APSO)

Opis: Klikom na dugme Prikaži pregled pozivaju se SO UčitajPregled (Pregled), otvara se forma za rad sa pregledom i ona poziva SO VratiListuLekova (List<Lek>).

1. Sistem traži podatke o izabranom pregledu. (SO)
2. Sistem prikazuje službeniku podatke o odabranom pregledu. (IA)



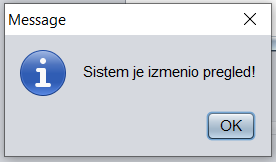
1. Službenik menja podatke o pregledu (unosi terapije i napomenu). (APUSO)



1. Službenik kontroliše da li je korektno uneo podatke o obavljenom pregledu. (ANSO)
2. Službenik poziva sistem da izmeni podatke o pregledu pacijenta. (APSO)

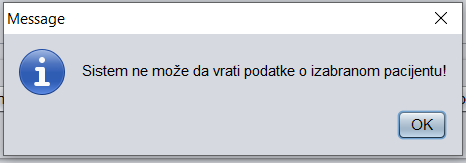
Opis: Klikom na dugme Sačuvaj izmene poziva se SO IzmeniPregled (Pregled) i forma za rad sa pregledima poziva SO VratiListuPregleda (Pacijent)

1. Sistem menja podatke o pregledu. (SO)
2. Sistem prikazuje službeniku poruku: “Sistem je izmenio pregled! “. (IA)

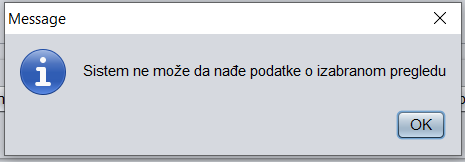


**Alternativna scenarija**

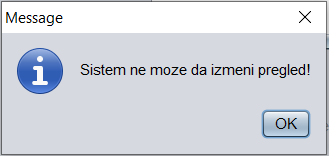
4.1 Ukoliko sistem ne može da nađe podatke o izabranom pacijentu on prikazuje službeniku poruku: “Sistem ne može da vrati podatke o izabranom pacijentu!” Prekida se izvršenje scenarija. (IA)



8.1 Ukoliko sistem ne može da nađe podatke o izabranom pregledu on prikazuje službeniku poruku: “Sistem ne može da nađe podatke o izabranom pregledu” Prekida se izvršenje scenarija. (IA)



13.1 Ukoliko sistem ne može da izmeni podatke o obavljenom pregledu, on službeniku prikazuje poruku: “Sistem ne može da izmeni pregled! “. (IA)



**SK6: Slučaj korišćenja: Pretraga pregleda**

**Naziv SK**

Pretraga pregleda

**Akteri SK**

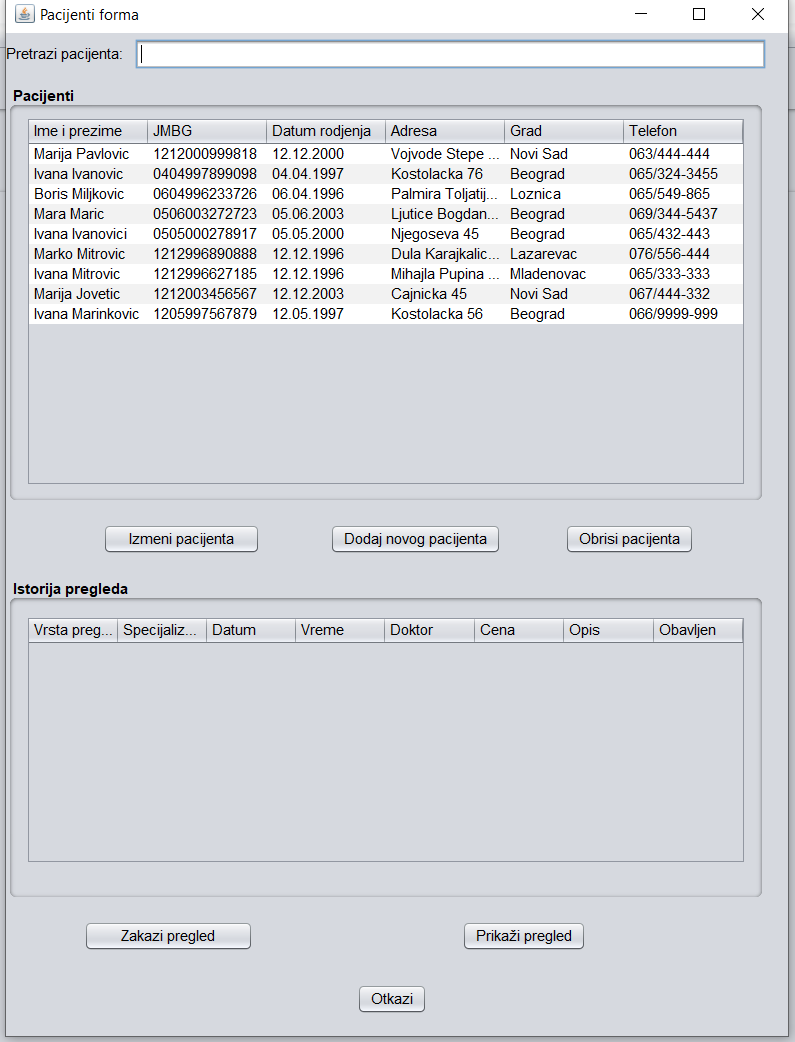
Službenik

**Učesnici SK**

Službenik i sistem (program)

**Preduslov:** Sistem je uključen i službenik je ulogovan pod svojom šifrom. Sistem prikazuje formu za rad sa pacijentima. Učitana je lista pacijenata.

Opis: Klikom na stavku menija Pacijenti-> Pretraži pacijente poziva se SO VratiListuPacijenata (List<Pacijent>) i otvara se forma za rad sa pacijentima.

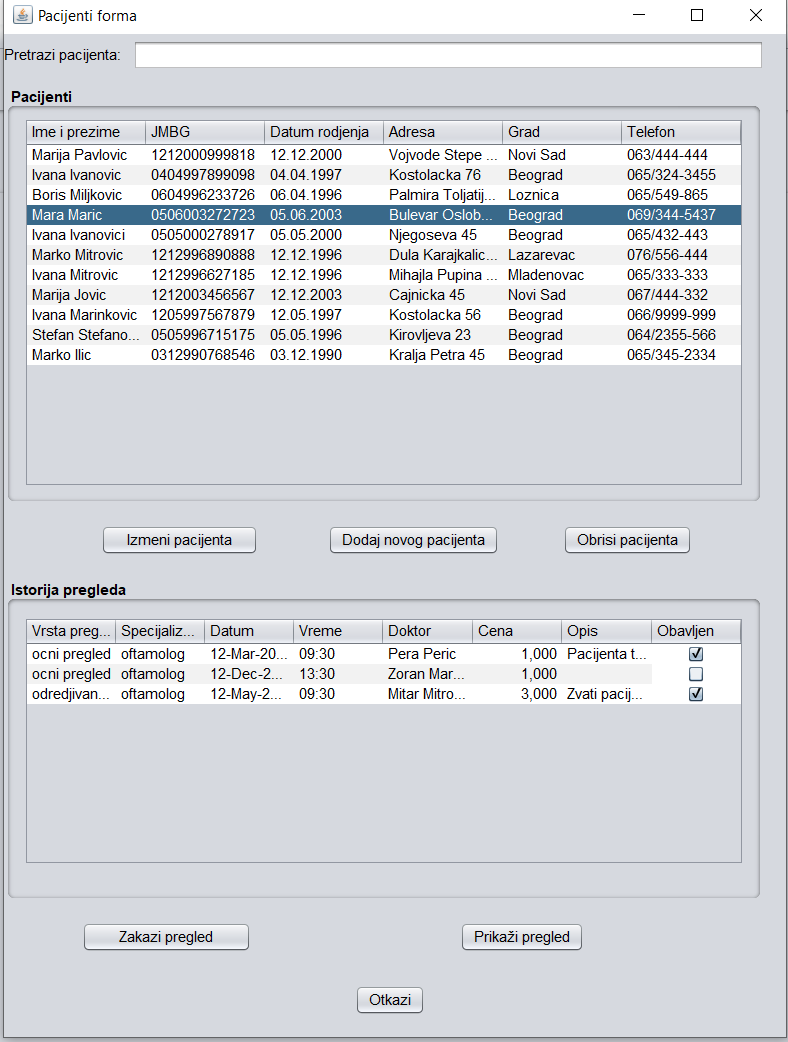


**Osnovni scenario SK**

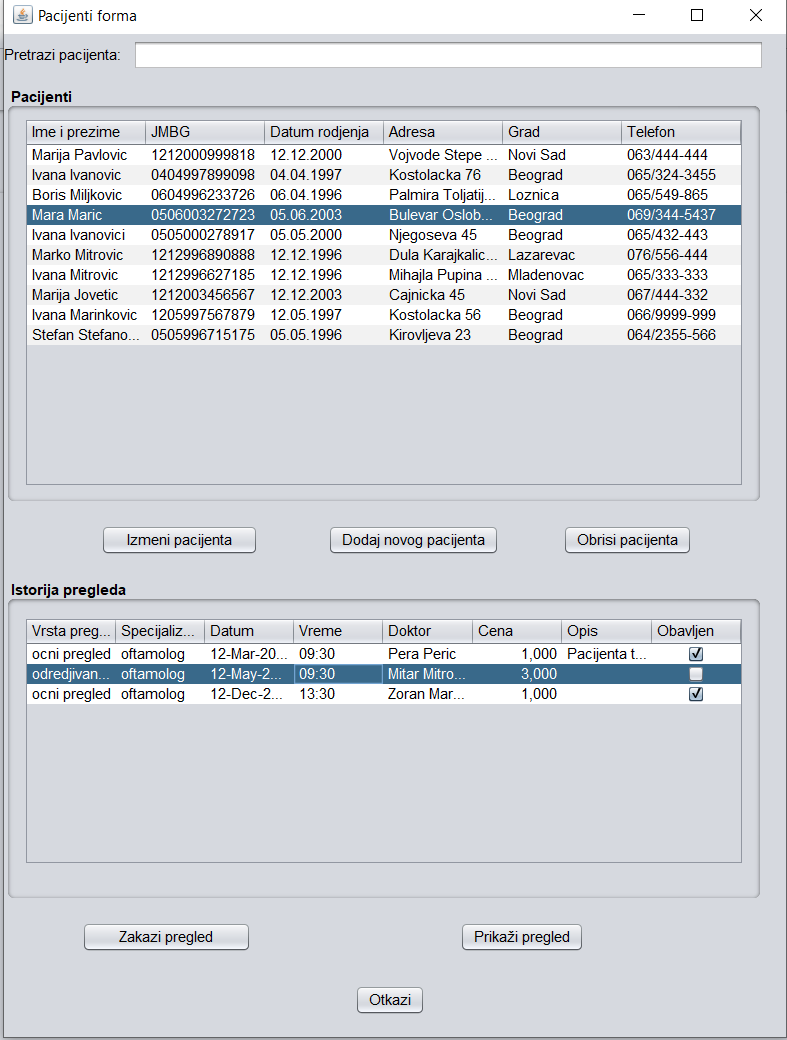
1. Službenik bira pacijenta. (APUSO)
2. Službenik poziva sistem da prikaže sve podatke (istoriju pregleda) o izabranom pacijentu. (APSO)

Opis: Klikom na pacijenta poziva se SO VratiListuPregleda (Pacijent, List<Pregled>)

1. Sistem traži podatke o izabranom pacijentu. (SO)
2. Sistem prikazuje službeniku sve podatke o odabranom pacijentu. (IA)



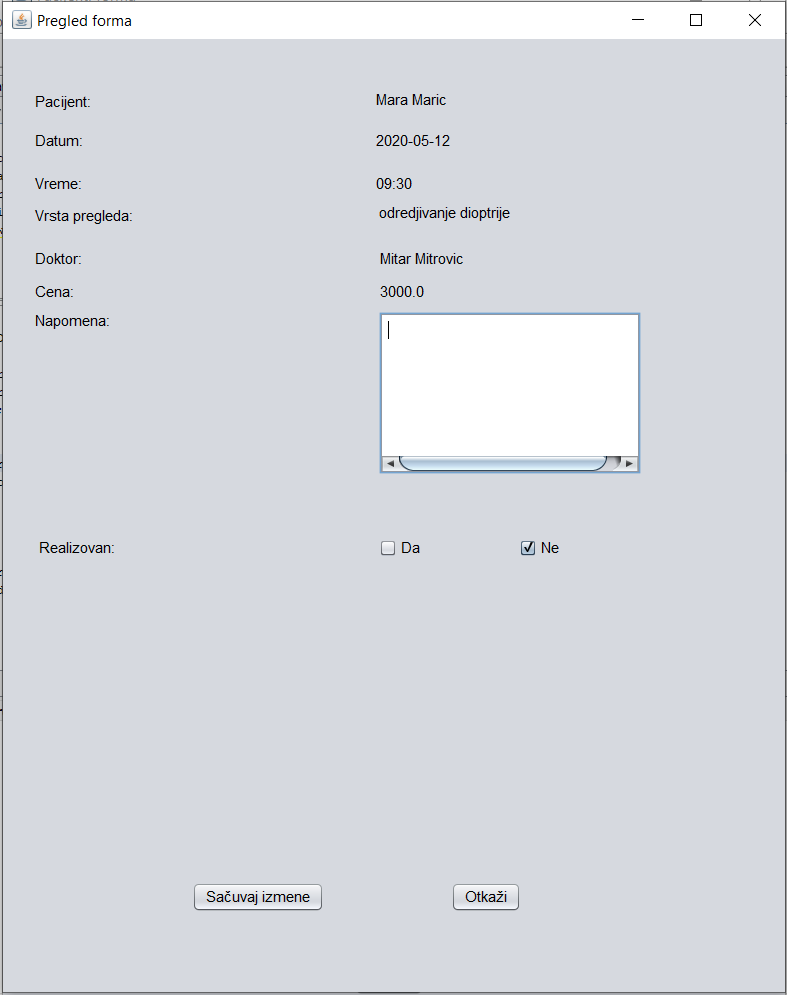
1. Službenik bira pregled koji želi da prikaže. (APUSO)



1. Službenik poziva sistem da učita podatke o izabranom pregledu. (APSO)

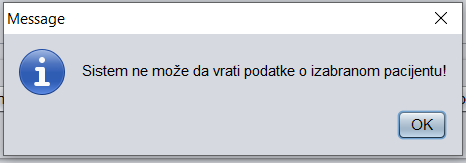
Opis: Klikom na dugme Prikaži pregled poziva se SO UčitajPregled (Pregled) i otvara se forma za prikaz pregleda. Ona poziva SO VratiListuLekova (List<Lek>)

1. Sistem traži podatke o izabranom pregledu. (SO)
2. Sistem prikazuje službeniku sve podatke o odabranom pregledu. (IA)

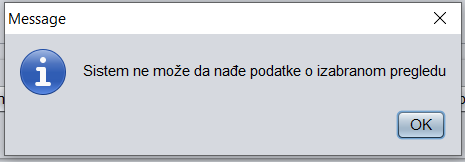


**Alternativni scenariji**

* 1. Ukoliko sistem ne može da nađe podatke o pacijentu, on službeniku prikazuje poruku: “Sistem ne može da nađe podatke o izabranom pacijentu! “. (IA)



* 1. Ukoliko sistem ne može da nađe podatke o pregledu, on službeniku prikazuje poruku: “Sistem ne može da nađe podatke o izabranom pregledu “. (IA)



**SK 7: Slučaj korišćenja – Unos novog doktora**

**Naziv SK**

Unošenje novog doktora

**Aktori SK**

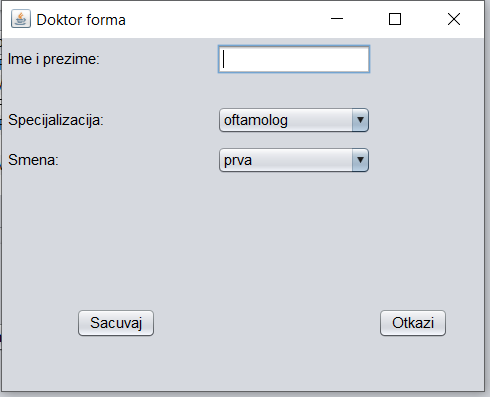
Službenik

**Učesnici SK**

Službenik i sistem (program)

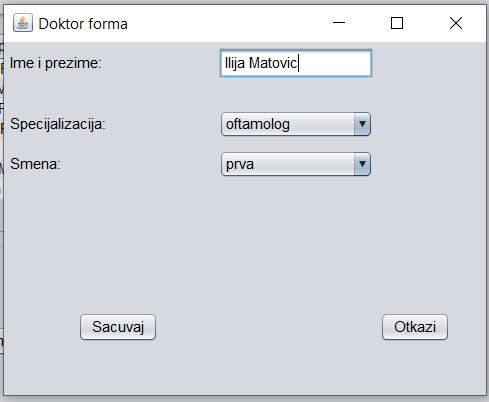
**Preduslov:** Sistem je uključen i službenik je ulogovan pod svojom šifrom. Sistem prikazuje formu za rad sa doktorom.

Opis: Klikom na dugme Dodaj doktora otvara se forma za dodavanje novog doktora. Forma poziva SO VratiListuSpecijalizacija (List<Specijalizacija>) i SO VratiListuSmena(List<Smena>)



**Osnovni scenario SK**

1. Službenik unosi podatke o doktoru. (APUSO)



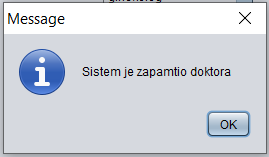
2. Službenik kontroliše da li je korektno uneo podatke u doktora. (ANSO)

3. Službenik poziva sistem da zapamti podatke o doktoru. (APSO)

Opis: Klikom na dugme Sačuvaj poziva se SO ZapamtiDoktora (Doktor) i forma za rad sa doktorima poziva SO VratiListuDoktora (List<Doktor>)

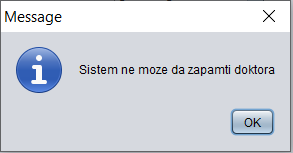
4. Sistem pamti podatke o doktoru. (SO)

5. Sistem prikazuje službeniku zapamćenog doktora i poruku: “Sistem je zapamtio doktora”. (IA)



**Alternativna scenarija**

5.1 Ukoliko sistem ne može da zapamti doktora on prikazuje službeniku poruku: “Sistem ne može da zapamti doktora”. (IA)



**SK 8: Slučaj korišćenja- Izmena doktora**

**Naziv SK**

Izmena doktora

**Autori SK**

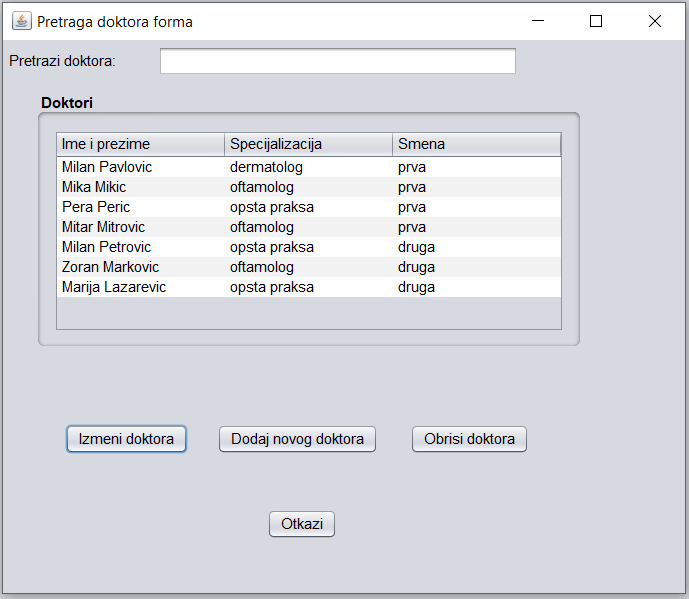
Službenik

**Učesnici SK**

Službenik i sistem (program)

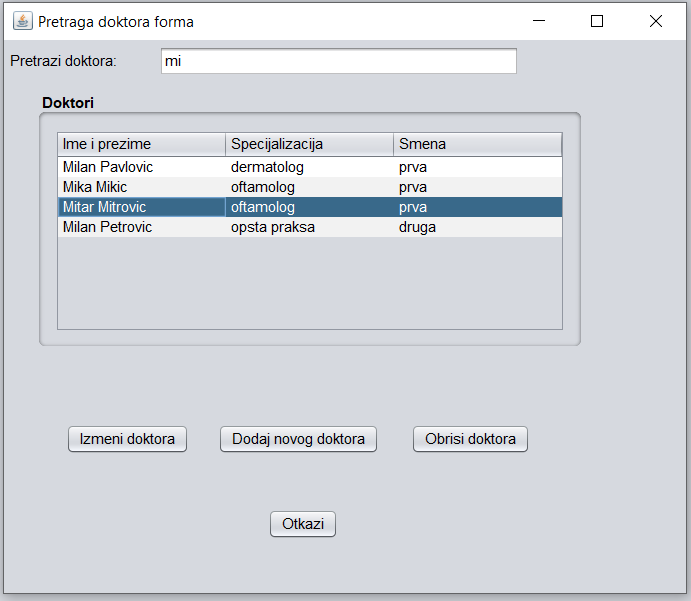
**Preduslov:** Sistem je uključen i službenik je ulogovan pod svojom šifrom. Učitana je lista doktora i sistem prikazuje formu za rad sa doktorima.

Opis: Klikom na stavku menija Doktori->Pretraži doktore otvara se forma za rad sa doktorima i poziva se SO VratiListuDoktora (List<Doktor>)



**Osnovni scenario SK**

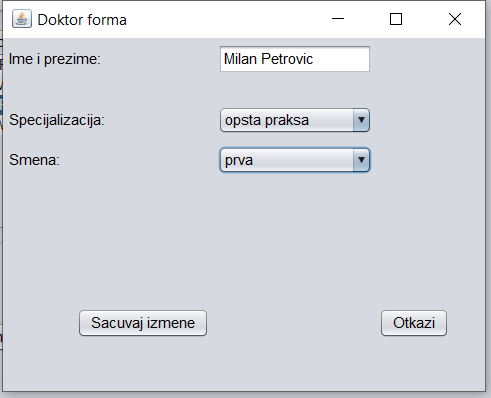
1. Službenik bira doktora. (APUSO)



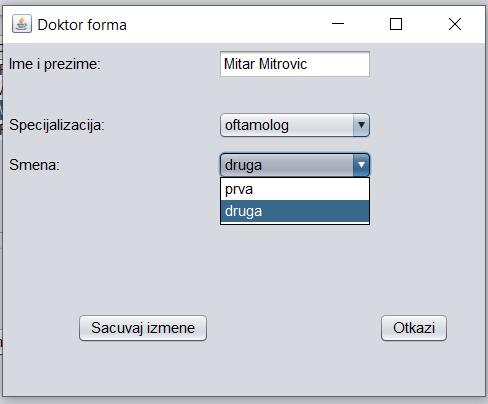
1. Službenik poziva sistem da prikaže sve podatke o izabranom doktoru. (APSO)

Opis: Klikom na dugme Izmeni doktora poziva se SO UčitajDoktora (Doktor) i i otvara se forma za izmenu doktora. Ona poziva SO VratiListuSpecijalizacija (List<Specijalizacija>) i SO VratiListuSmena (List<Smena>)

1. Sistem traži podatke o izabranom doktoru. (SO)
2. Sistem prikazuje službeniku podatke o odabranom doktoru. (IA)



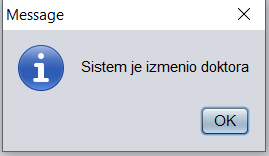
1. Službenik unosi (menja) podatke o doktoru. (APUSO)



1. Službenik kontroliše da li je korektno uneo podatke o doktoru. (ANSO)
2. Službenik poziva sistem da izmeni podatke o doktoru. (APSO)

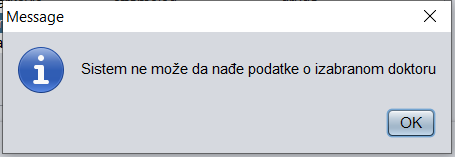
Opis: Klikom na dugme Sačuvaj izmene poziva se SO IzmeniDoktora (Doktor) i forma za rad sa doktorima poziva SO VratiListuDoktora (List<Doktor>)

1. Sistem menja podatke o doktoru. (SO)
2. Sistem prikazuje službeniku i poruku: “Sistem je izmenio doktora “ . (IA)

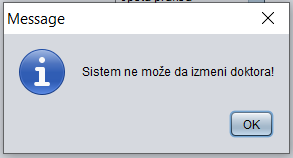


**Alternativni scenariji**

4.1 Ukoliko sistem ne može da nađe podatke o doktoru, on službeniku pikazuje poruku: “Sistem ne može da nađe podatke o izabranom doktoru”. Prekida se izvršenje scenarija. (IA)



9.1 Ukoliko sistem ne može da izmeni podatke o doktoru, on prikazuje službeniku poruku: „Sistem ne može da izmeni doktora! “. (IA)



**SK9: Slučaj korišćenja- Brisanje doktora**

**Naziv SK**

Brisanje doktora

**Autori SK**

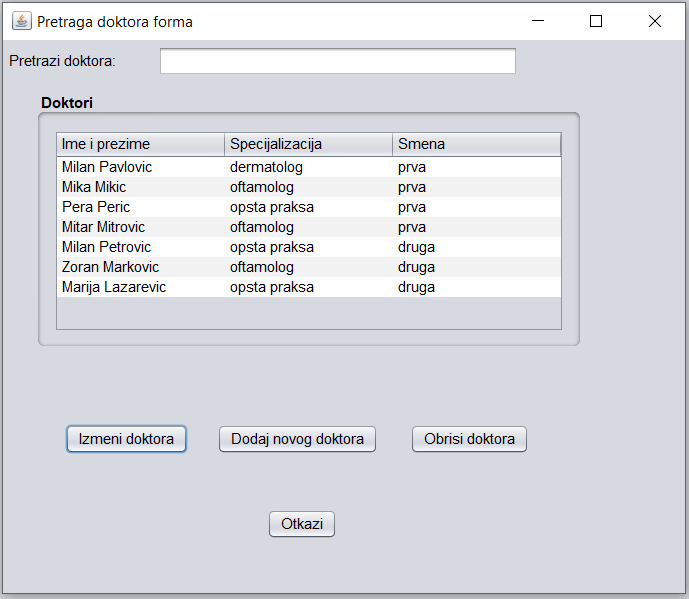
Službenik

**Učesnici SK**

Službenik i sistem (program)

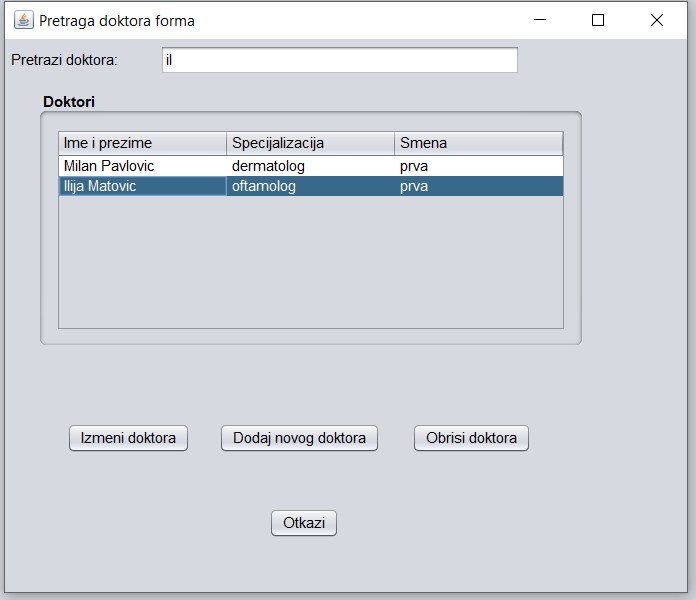
**Preduslov:** Sistem je uključen i službenik je ulogovan pod svojom šifrom. Učitana je lista doktora. Sistem prikazuje formu za rad sa doktorima.

Opis: Klikom na stavku menija Doktori->Pretraži doktore otvara se forma za rad sa doktorima i poziva se SO VratiListuDoktora (List<Doktor>)



**Osnovni scenario SK**

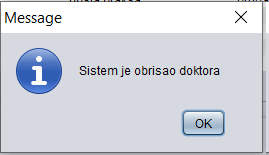
1. Službenik bira doktora kojeg želi da obriše. (APUSO)



1. Službenik poziva sistem da obriše podatke o doktoru. (APSO)

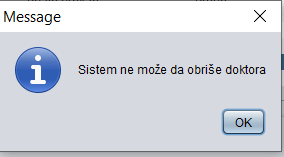
Opis: Klikom na dugme Obriši doktora poziva se SO ObrišiDoktora (Doktor) i forma za rad sa doktorima poziva SO VratiListuDoktora (List<Doktor>)

1. Sistem briše doktora i pamti izmene. (SO)
2. Sistem prikazuje službeniku poruku: “Sistem je obrisao doktora “. (IA)



**Alternativni scenariji**

4.1. Ukoliko sistem ne može da obriše doktora, on prikazuje službeniku poruku: „Sistem ne može da obriše doktora “. (IA)



**SK 10: Slučaj korišćenja – Logovanje službenika**

**Naziv SK**

Logovanje službenika

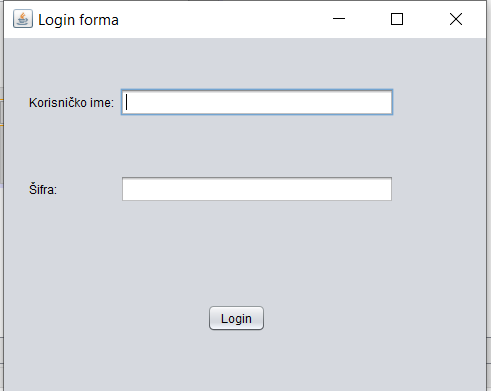
**Aktori SK**

Službenik

**Učesnici SK**

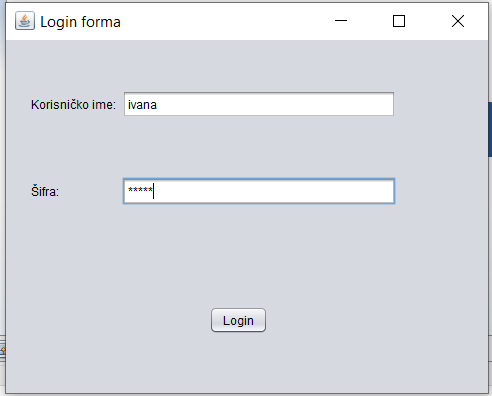
Službenik i sistem (program)

**Preduslov:** Sistem je uključen i prikazuje formu za logovanje službenika.



**Osnovni scenario SK**

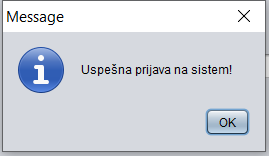
1. Službenik unosi podatke za autentifikaciju službenika. (APUSO)



1. Službenik poziva sistem da pronađe službenika sa zadatim podacima. (APSO)

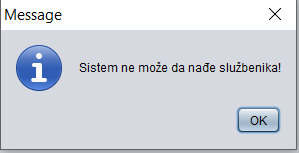
Opis: Klikom na dugme Login poziva se SO NadjiSlužbenika (Službenik)

1. Sistem pretražuje službenike. (SO)
2. Sistem prikazuje poruku “Uspešna prijava na sistem!” i omogućava pristup sistemu. (IA)



**Alternativna scenarija**

4.1. Ukoliko sistem ne može da nađe službenika prikazuje poruku “Sistem ne može da nadje službenika!”. (IA)

****

### 3.2.2 Projektovanje kontrolera korisničkog interfejsa

Ekranska forma prosleđuje podatke dobijene od kontrolera korisničkog interfejsa. U skladu sa tim kontroler korisničkog interfejsa odgovoran je da:

a) prihvati podatke koje šalje ekranska forma

b) konvertuje podatke (koji se nalaze u grafičkim elementima) u objekat koji predstavlja ulazni argument SO koja će biti pozvana

c) šalje zahtev za izvršenje SO do aplikacionog servera (softverskog sistema)

d) prihvata objekat (izlaz) softverskog sistema koji nastaje kao rezultat izvršenja SO i

e) konvertuje objekat u podatke grafičkih elemenata.

## 3.3 Projektovanje aplikacione logike

[1]

Osnovni delovi aplikacione logike su:

• Kontroler

• poslovna logika

• deo zadužen za komunikaciju sa bazom

• deo zadužen za komunikaciju sa klijentom.

Poslovna logika se sastoji iz domenskih klasa i klasa sistemskih operacija.

Kontroler podiže serverski soket u delu za komunikaciju sa klijentom. Kontroler se prema poslovnoj logici odnosi kao klijent. Sistemske operacije sa kojima je kontroler povezan u klijentskom su odnosu prema brokeru baze podataka. Broker baze podataka izvršava osnovne operacije nad bazom podataka. Transakcijom upravljaju sistemske operacije. Po izvršenju sistemske operacije, kontroleru se vraća signal, potom kontroler kreira transferni objekat koji vraća objektu za komunikaciju sa klijentom. Odeljak za komunikaciju sa klijentom prosleđuje objekat u mrežu.

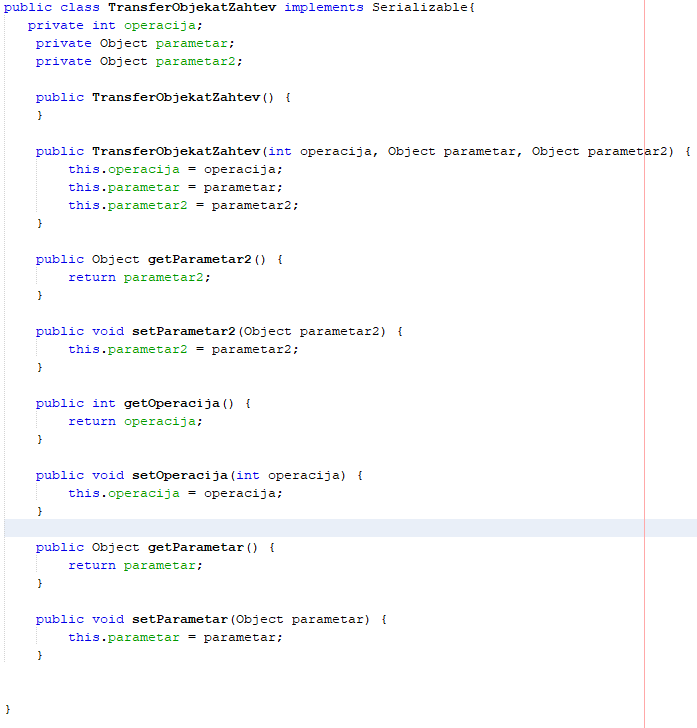
Slanje i primanje podataka od klijenta se ostvaruje preko soketa. Klijent šalje zahtev za izvršenje neke od SO do odgovarajuće niti, koja je povezana sa tim klijentom. Nit prima zahtev i dalje ga preusmerava do klasa koje su odgovorne za izvršenje sistemskih operacija. Nakon izvršenja SO rezultat se vraća do aplikacione logike, odnosno do niti, koja taj rezultat šalje nazad do klijenta.

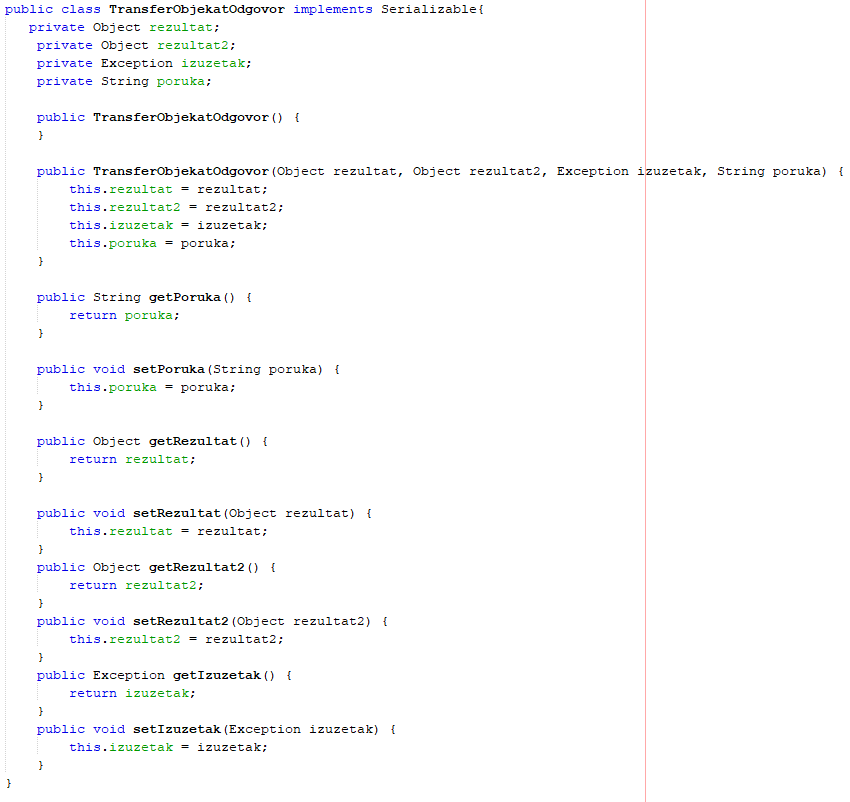
### 3.3.1 Komunikacija sa klijentom

Deo za komunikaciju podiže serverski soket koji osluškuje mrežu. Kada klijent uspostavi konekciju server generiše nit koja će biti odgovorna za dvosmernu vezu sa klijentom.

Klijent šalje zahtev za izvršenje neke od sistemskih operacija, odgovarajuća nit prihvata zahtev i prosleđuje ga do kontrolera aplikacione logike. Nakon izvršenja sistemske operacije rezultat se prosleđuje klijentu.

Komunikacija između klijenta i servera se obavlja razmenom objekta TransferObjekatOdgovor i TransferObjekatZahtev.

****

****

### 3.3.2. Kontroler

Kontroler aplikacione logike prihvata zahteve za izvršenje sistemskih operacija i iste prosleđuje do konkretne sistemske operacije. Nakon izvršenja sistemske operacije kontroler prihvata odgovor i vraća nazad pozivaocu (niti klijenta).

### 3.3.3. Ponašanje softverskog sistema

Sistemske operacije su povezane sa logikom problema. Svaka sistemska operacija nasleđuje opštu sistemsku operaciu koja sadrži sledeće metode:

ucitajDriver(),

otvoriKonekciju(),

commitTransakcije(),

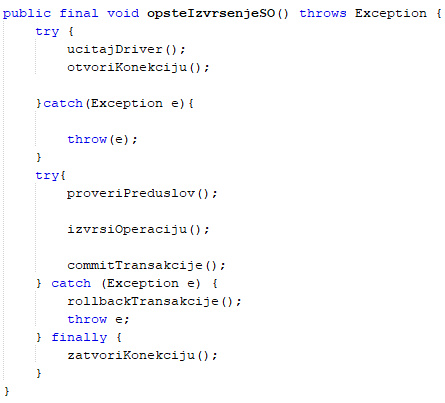
rollbackTransakcije(),

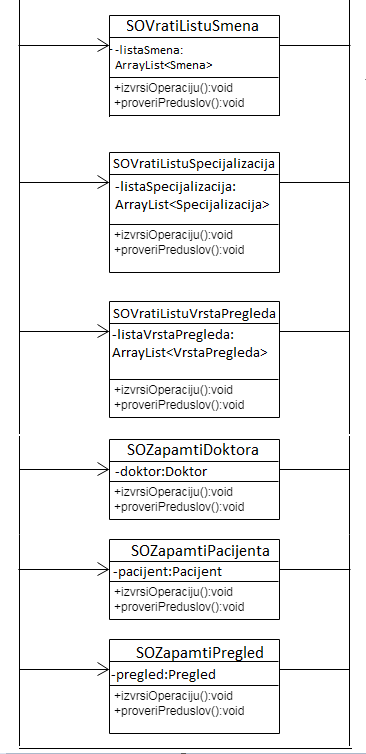
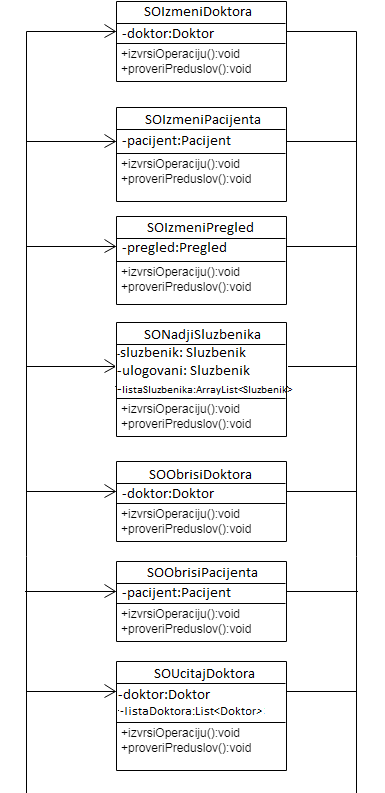
proveriPreduslov(),

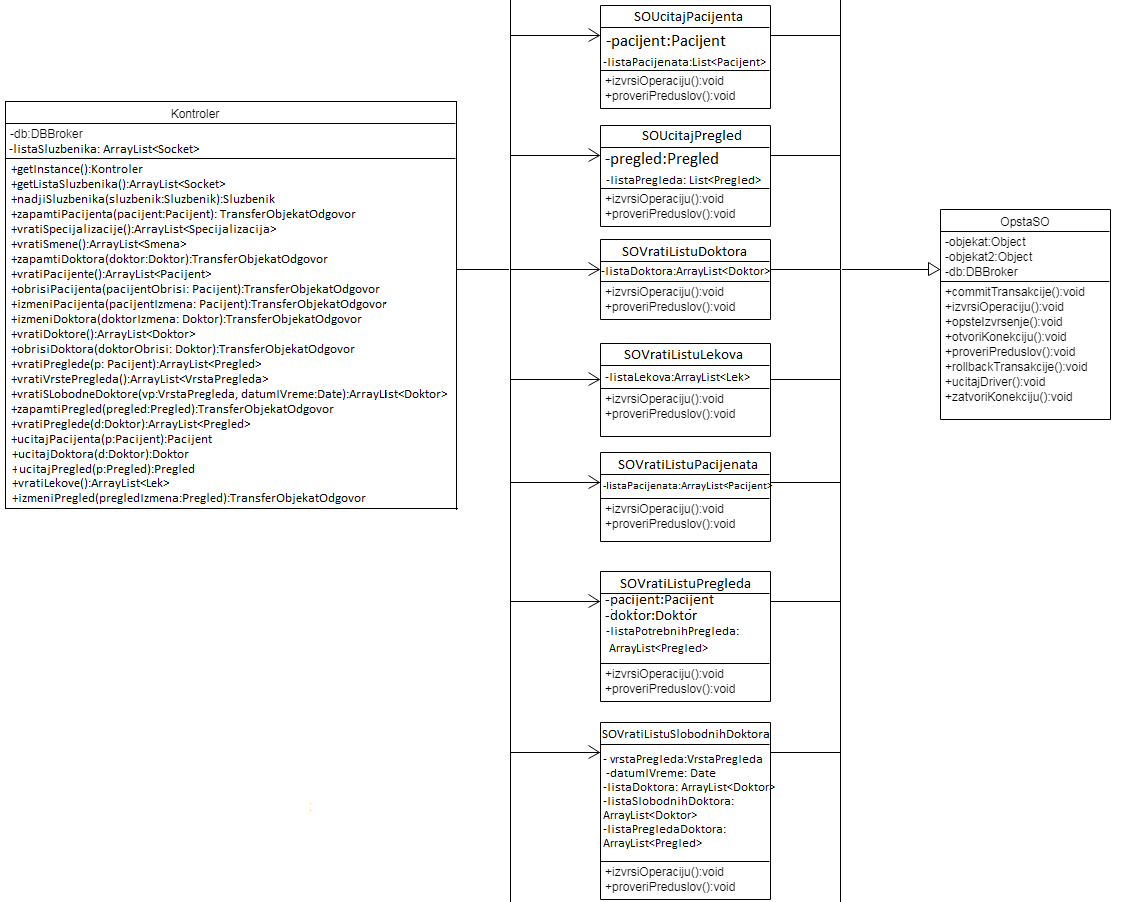
izvrsiOperaciju(),

opsteIzvrsenjeSO ()

Posle instanciranja odgovarajuće SO klase, kontroler nad instancom poziva metodu za opšte izvršenje svake SO:



****

****

### 3.3.4. Broker baze podataka i domenske klase

Klasa DBBroker predstavlja perzistentni okvir koji posreduje u svim operacijama nad bazom podataka i realizuje sledeće metode:

1. public void ucitajDriver ()

2. public void otvoriKonekciju ()

3. public void zatvoriKonekciju ()

4. public void commitTransakcije ()

5. public void rollbackTransakcije ()

6. public void insert (OpstiDomenskiObjekat odo)

7. public void update (OpstiDomenskiObjekat odo)

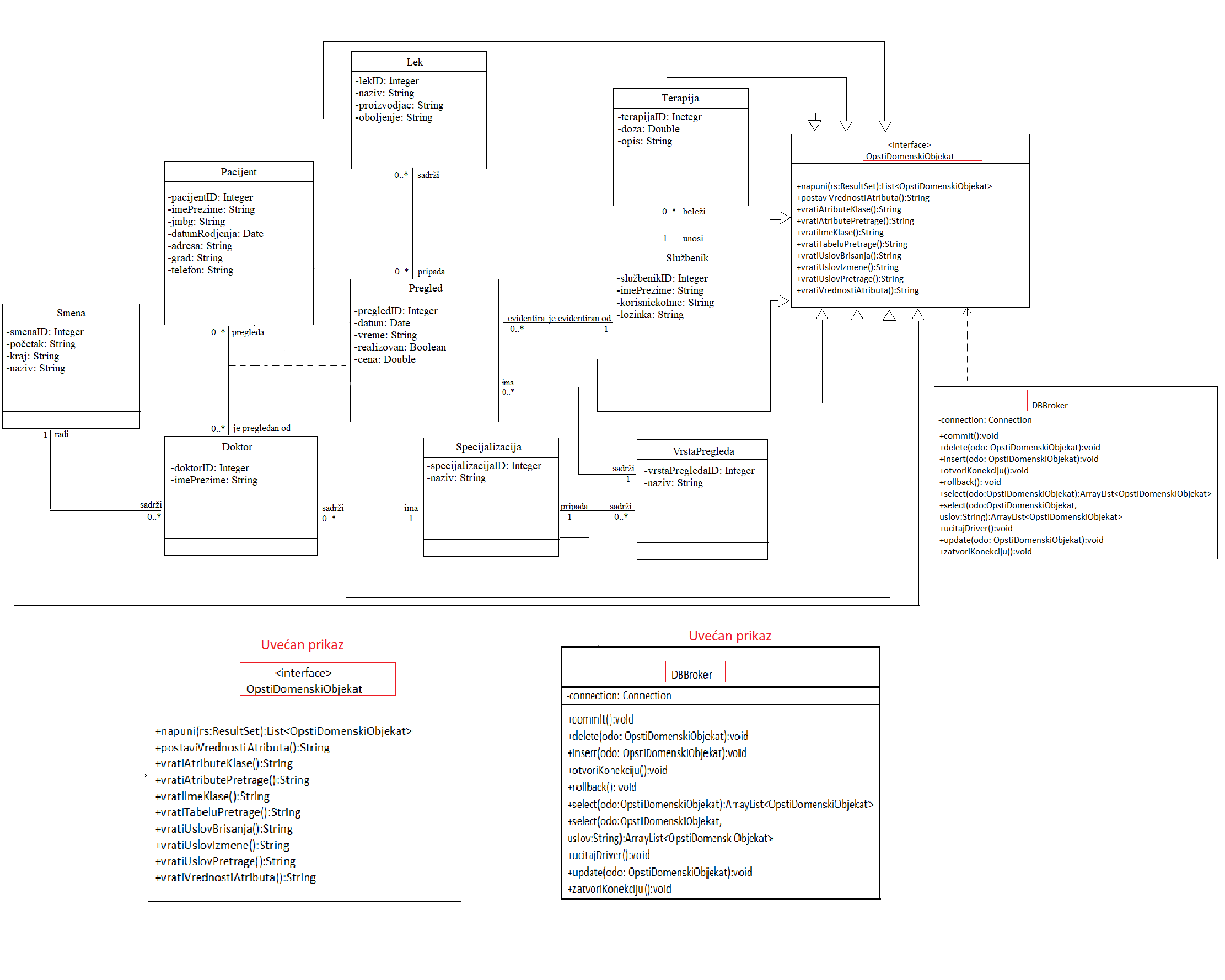
8. public void delete (OpstiDomenskiObjekat odo)

9. public ArrayList< OpstiDomenskiObjekat > select (OpstiDomenskiObjekat odo)

10. public ArrayList< OpstiDomenskiObjekat > select (OpstiDomenskiObjekat odo, String uslov)

Sve metode DBBroker klase su projektovane kao generičke, što znači da mogu da prihvate različite domenske objekte preko parametara. Na taj način postižemo da u DBBroker klasi nemamo implementaciju pojedinačnih metoda za svaku domensku klasu.

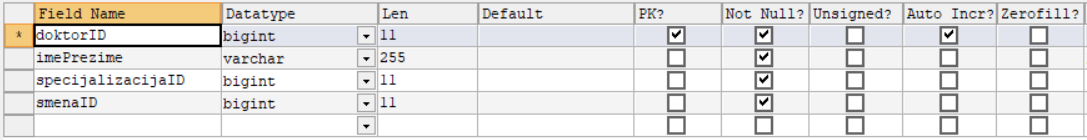
U procesu pravljenja DBBroker klase dobili smo metode interfejsa OpstiDomenskiObjekat. Svaka klasa iz domena implementira dati interfejs, i sve njegove metode. Na taj način je omogućeno da metode klase DBBroker budu generičke i da se DBBroker povezuje sa interfejsom OpstiDomenskiObjekat.



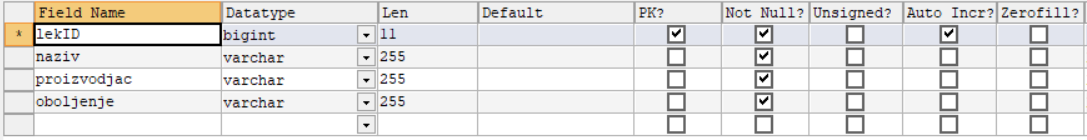
## 3.4. Projektovanje baze podataka

Na osnovu softverskih klasa strukture projektovane su tabele (skladišta podataka) relacionog si stema za upravljanje bazom podataka:

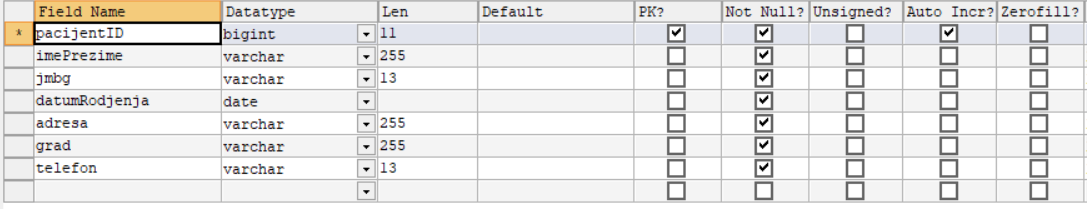
**Tabela Doktor**

****

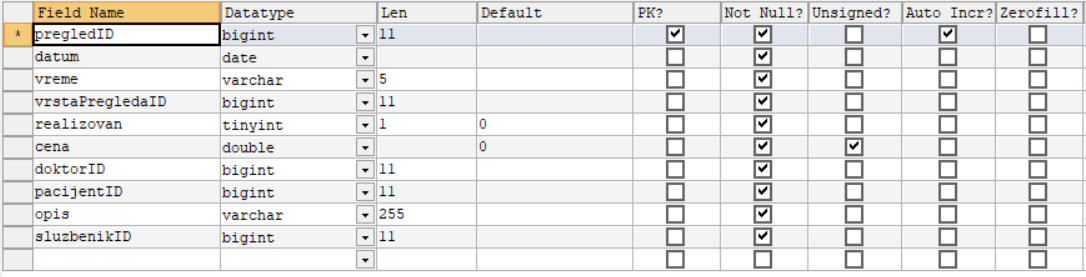
**Tabela Lek**

****

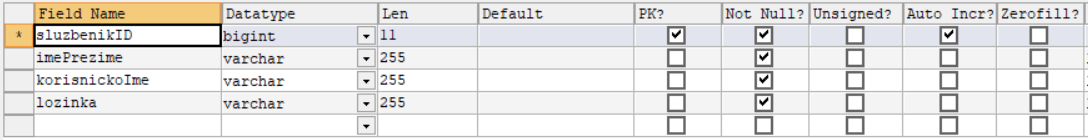
**Tabela Pacijent**

****

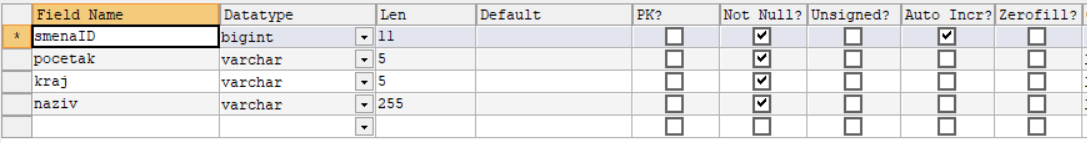
**Tabela Pregled**

****

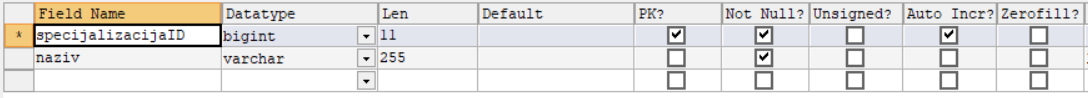
**Tabela Sluzbenik**

****

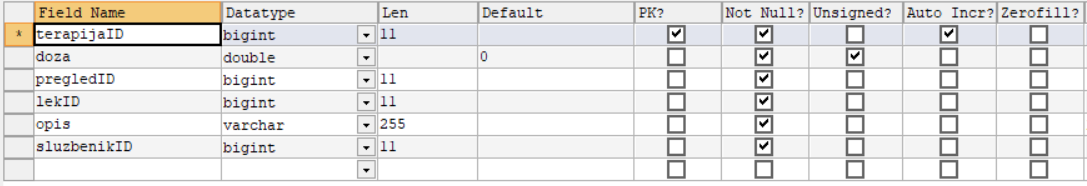
**Tabela Smena**

****

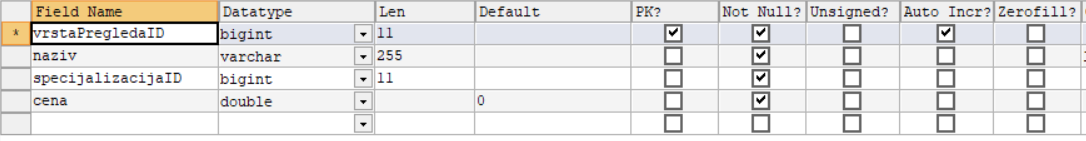
**Tabela Specijalizacija**

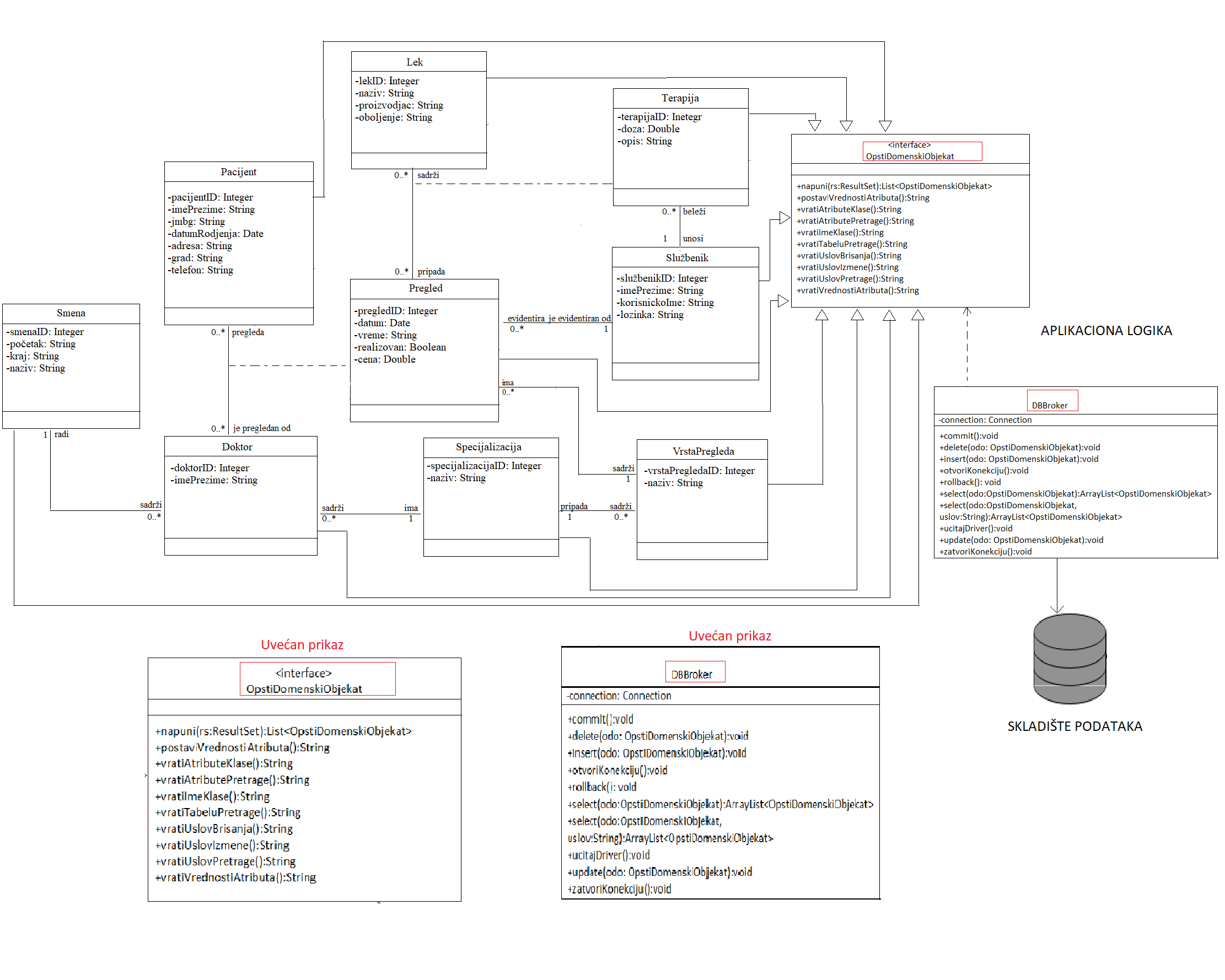
****

**Tabela Terapija**

****

**Tabela VrstaPregleda**

****

****

# 4. Implementacija

Softverski sistem, rezultat ovog rada, razvijen je u programskom jeziku Java i projektovan je kao klijent-server aplikacija. Kao razvojno okruženje korišćen je NetBeans IDE 8.2. RC, a kao sistem za upravljanje bazom podataka korišćen je XAMPP.

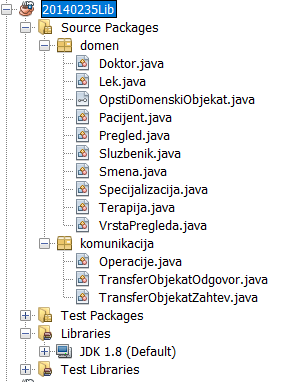
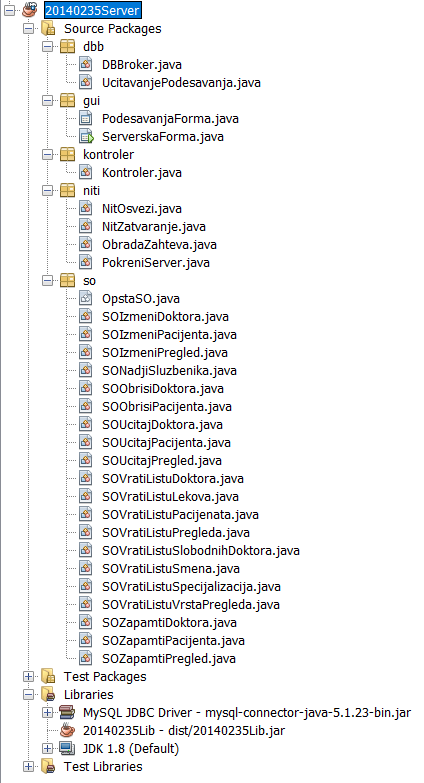
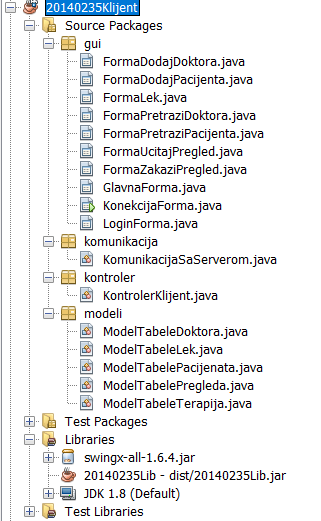
Čitav sistem je realizovan u tri projekta:

20140235 Klijent

20140235 Lib

20140235 Server

U projekat klijenta i servera dodata je .jar datoteka projekta koji sadrži zajedničke domenske objekte.



# 5. **Testiranje**

Testiranje aplikacije i svih njenih slučajeva korišćenja je rađeno ručno, unošenjem mogućih varijanti. Uprogram su unošene logične, nelogične, poželjne i nepoželjne vrednosti. Ukoliko je došlo do sintaksnog ili logičkog propusta, uočen nedostatak je bio ispravljen.

# 6. Literatura

1. dr Siniša Vlajić, Projektovanje softvera (skripta), Beograd, 2015