*Jesenji semestar, 2016/17*



*PREDMET: SE201 – Uvod u softversko inženjerstvo*

# Projektni zadatak na temu

**Sistem predaje predispitnih obaveza na Univerzitet Metropolitan**

Profesor: Ime i prezime:

**Prof. Dragan Domazet Viktor Cvetanovic**

Asistent: Broj indeksa:

**Jovana Jović 4421**

# 1. Uvod

Cilj dokumenta je da detaljno predstavi sistem za predaju predispitnih obaveza na Univerzitet Metropolitan. Ovaj projektni zadatak baviće se definisanjem osnovnih funkcionalnih i nefunkcionalnih zahteva koje softver mora da zadovolji kako bi se smatrao uspešnim. Kako bi projekat bio realizovan biće objašnjen putem korišćenja dijagrama(klasni, use-case, sekvencijalni) koji bi trebalo detaljnije da objasne način i zamisao funkcionisanja sistema. Za potrebe izrade dijagrama koristiće se alat **PowerDesigner.** U projektnom zadatku biće predstavljene i mogućnosti svih korisnika sistema. Korisnici ovog sistema su profesor I student.

Projekat je tako zamišljen da olakša i poboljša predaju predispitnih obaveza na Univerzitet Metropolitan.

# 2. Opis i svrha dokumenta

Svrha dokumenta je da jasno opiše moguće funkcionalnosti i mogućnosti softvera za prelazak studenata na Univerzitet Metropolitan. Dokument se sastoji od opisa sistema sa identifikovanim klasama korisnika i njihovim ovlašćenjima, detalje zahteva u kojima su opisani funkcionalni i nefunkcionalni zahtevi. Dokument će se sastojati i od specifikacije dizajna u okviru kog se nalaze dijagrami slučajeva korišćenja, klasni dijagram i svih mogućih sekvencijalnih dijagrama koji su vezani za rad sistema. Nakon toga, dokument će dati opis funkcionisanja sistema u vidu slika.

# 3. Opis sistema i metodologija rada

.

## 3.1. Funkcije softvera

Funkcije sistema su:

* Pregled podataka
* Vraćanje dokumenta na doradu
* Prosleđivanje dokumenta na obradu
* Unos broja ects bodova
* Pregled prosleđenog dokumenta
* Vraćanje zahteva zbog nedostatka dokumentacije
* Odbijanje zahteva
* Evidentiranje predloga o prelazu
* Preuzimanje fajla o položenim predmetima sa prethodne ustanove

## 3.2. Korisnici sistema

Postoje dve vrste korisnika sistema:

1. Profesor
2. Student

|  |  |
| --- | --- |
| 3.2.1. | Opis sistema za profesora |

Profesor ima mogućnost pregleda predispitnih obaveza na jednom mestu I tacno ce moci za svaku definisanu predispitnu obavezu da zna da li je odredjeni student odradio taj zadatak. Profesor ce na ovaj nacin u svakom trenutku imati validne informacije koje ce se pamtiti u sistemu. Profesor moze da oceni svaku predispitnu obavezu kao I do sada.

|  |  |
| --- | --- |
| 3.2.2. | Opis sistema za studente |

Studenti ce pored sadasnjih opcija za samo pregled predispitnih obavezi moci da predaju predispitne obaveze preko sistema ISUM koji je glavni informacioni sistem za Univerzitet Metropolitan. Studenti ce na ovaj nacin imati uvek u uvid predispitne obaveze koje su predali I kada su definisani rokovi za te predispitne obaveze. Studenti nece morati vise da koriste mail kako bi slali predispitne obaveze I na taj nacin se smanjuju sanse za greske.

## 3.3. Metodologija izrade

Metodologiju koju ćemo koristiti za razvoj sistema je model vodopada.

Model vodopada je proces koji se mora planirati i odrediti termin za sve aktivnosti procesa pre nego što se počne njegovo izvršavanje. Koristi se u slučajevima kada su svi zahtevi dobro definisani, jasni i stabilni odnosno ne menjaju se za vreme razvoja softvera. Sistem koji će se obrađivati u ovom projektu zahteva dokumentaciju što zahteva i model vodopada. Model vodopada se koristi u velikim softverskim sistemima koji moraju biti vrlo pouzdani.

# 4. Specifikacija zahteva

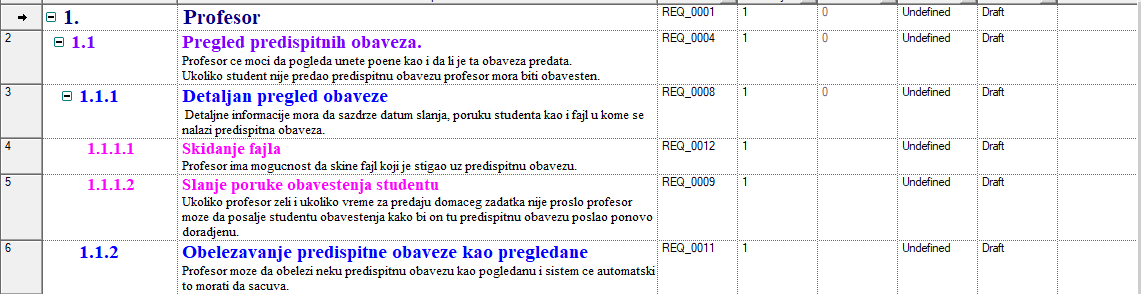
Specifični softverski zahtevi predstavljaju način na koji bi bilo potrebno razvijati neki sistem. Dobro definisanje specifičnih zahteva dovodi do ispravnog rešenje. U ovaj tip softverskih zahteva spadaju funkcionalni i ne-funkcionalni zahtevi.

## 4.1. Funkcionalni zahtevi

U delu funkcionalnih zahteva biće prikazani svi funkcionalni zahtevi koje sistem treba da zadovolji. Funkcionalni zahtevi su podeljeni na funkcionalne zahteve koje mora da zadovolji profesor I na funkcionalne zahteve koje mora da zadovolji student. Za izradu i prikaz funkcionalnih zahteva se koristi Requirements model koji zadovoljava sve standarde za pisanje istih generisan u PowerDesigner alatu.

|  |  |
| --- | --- |
| 4.1.1. | Funkcionalni zahtevi profesora |

U ovom poglavnju prikazani su funkcionalni zahtevi koje profesor treba da zadovolji. Svi zahtevi koje profesor pruža se nalaze na slici 4.2.1.1.



Slika 4.2.1.1 – Funkcionalni zahtevi profesora

### 4.2.1.1 **Pregled predispitnih obaveza.**

Profesor ce moci da pogleda unete poene kao i da li je ta obaveza predata.

Ukoliko student nije predao predispitnu obavezu profesor mora biti obavesten.

4.2.1.2 **Detaljan pregled obaveze**

### Detaljne informacije mora da sazdrze datum slanja, poruku studenta kao i fajl u kome se nalazi predispitna obaveza.

### 4.2.1.3 **Skidanje fajla**

Profesor ima mogucnost da skine fajl koji je stigao uz predispitnu obavezu.

### 4.2.1.4 **Slanje poruke obavestenja studentu**

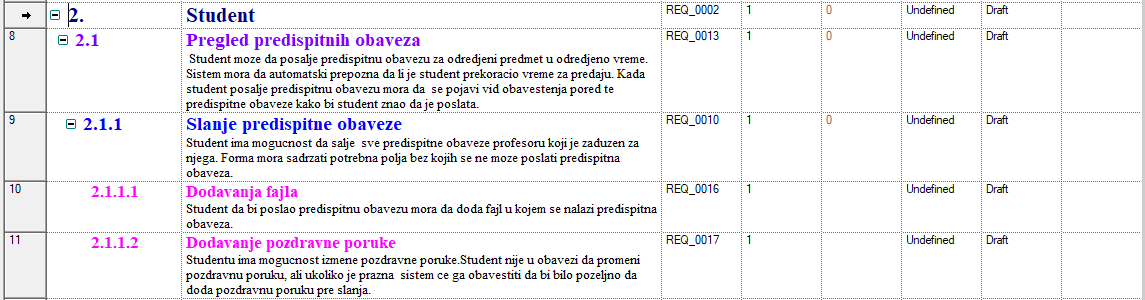
Ukoliko profesor zeli i ukoliko vreme za predaju domaceg zadatka nije proslo profesor moze da posalje studentu obavestenja kako bi on tu predispitnu obavezu poslao ponovo doradjenu.

4.2.1.5 **Obelezavanje predispitne obaveze kao pregledane**

Profesor moze da obelezi neku predispitnu obavezu kao pogledanu i sistem ce automatski to morati da sacuva.

|  |  |
| --- | --- |
| 4.2.2. | Funkcionalni zahtevi studenta |

U ovom poglavnju prikazani su funkcionalni zahtevi koje student treba da zadovolji. Svi zahtevi koje student pruža se nalaze na slici 4.2.2.1.



Slika 4.2.2.1 – Funkcionalni zahtevi studenta

4.2.2.1 **Pregled predispitnih obaveza**

Student moze da posalje predispitnu obavezu za odredjeni predmet u odredjeno vreme. Sistem mora da automatski prepozna da li je student prekoracio vreme za predaju. Kada student posalje predispitnu obavezu mora da se pojavi vid obavestenja pored te predispitne obaveze kako bi student znao da je poslata.

4.2.2.2 **Slanje predispitne obaveze**

#### Student ima mogucnost da salje sve predispitne obaveze profesoru koji je zaduzen za njega. Forma mora sadrzati potrebna polja bez kojih se ne moze poslati predispitna obaveza.

4.2.2.3 **Dodavanja fajla**

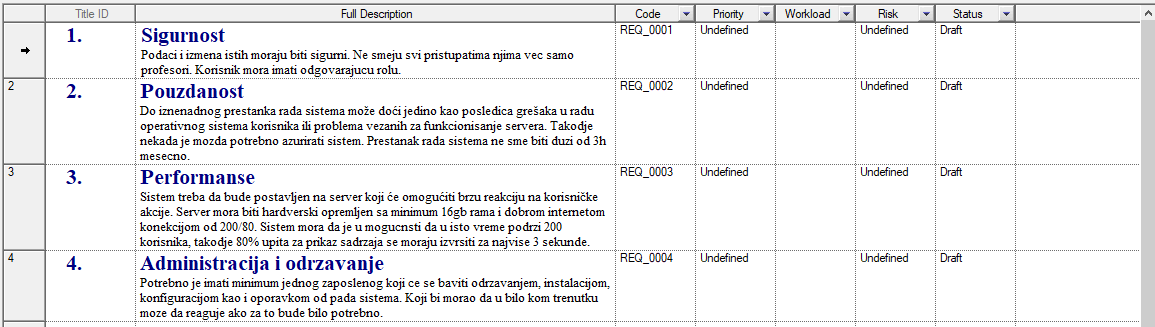
#### Student da bi poslao predispitnu obavezu mora da doda fajl u kojem se nalazi predispitna obaveza.

4.2.2.4 **Dodavanje pozdravne poruke**

Studentu ima mogucnost izmene pozdravne poruke.Student nije u obavezi da promeni pozdravnu poruku, ali ukoliko je prazna sistem ce ga obavestiti da bi bilo pozeljno da doda pozdravnu poruku pre slanja.

|  |  |
| --- | --- |
| 4.2.4. | Nefunkcionalni zahtevi |

Nefunkcionalni zahtevi su zahtevi koji su vezani za ceo sistem. Pod nefunkcionalnim zahtevima spadaju sigurnost, performanse, pouzdanost, lakoća upotrebe.



Slika 4.2.4.1 – Nefunkcionalni zahtevi sistema

4.2.4.1 **Sigurnost**

Podaci i izmena istih moraju biti sigurni. Ne smeju svi pristupatima njima vec samo profesori. Korisnik mora imati odgovarajucu rolu.

4.2.4.2 **Pouzdanost**

#### Do iznenadnog prestanka rada sistema može doći jedino kao posledica grešaka u radu operativnog sistema korisnika ili problema vezanih za funkcionisanje servera. Takodje nekada je mozda potrebno azurirati sistem. Prestanak rada sistema ne sme biti duzi od 3h mesecno.

#### 

4.2.4.3 **Performanse**

Sistem treba da bude postavljen na server koji će omogućiti brzu reakciju na korisničke

akcije. Server mora biti hardverski opremljen sa minimum 16gb rama i dobrom internetom konekcijom od 200/80. Sistem mora da je u mogucnsti da u isto vreme podrzi 200 korisnika, takodje 80% upita za prikaz sadrzaja se moraju izvrsiti za najvise 3 sekunde.

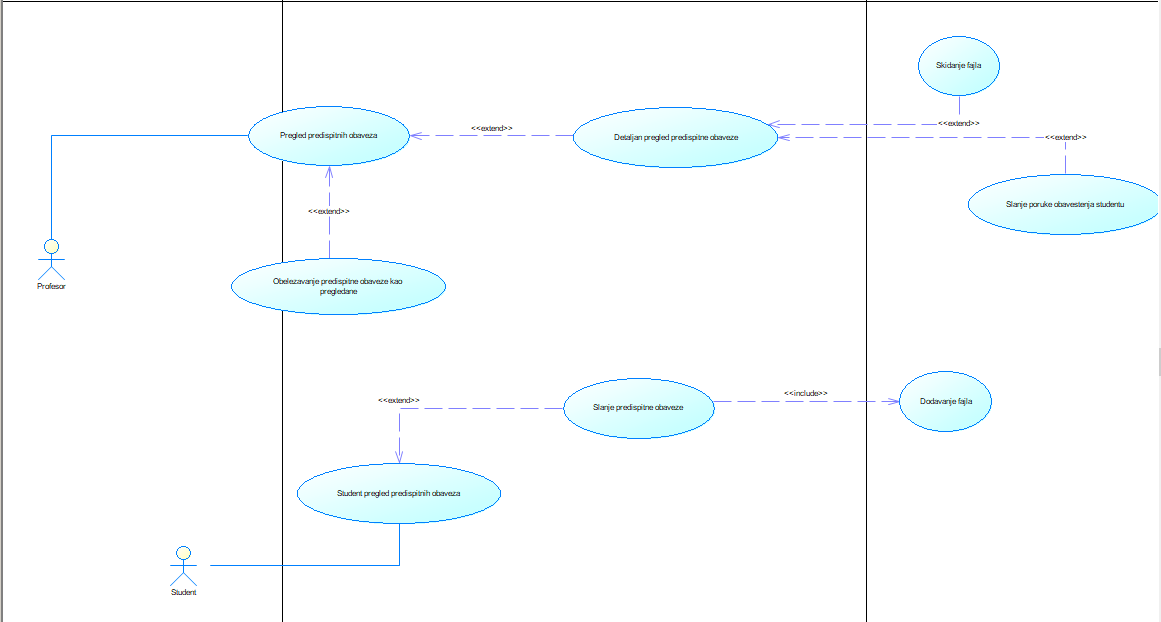
4.2.4.4 **Administracija i odrzavanje**

Potrebno je imati minimum jednog zaposlenog koji ce se baviti odrzavanjem, instalacijom, konfiguracijom kao i oporavkom od pada sistema. Koji bi morao da u bilo kom trenutku moze da reaguje ako za to bude bilo potrebno.

#### 

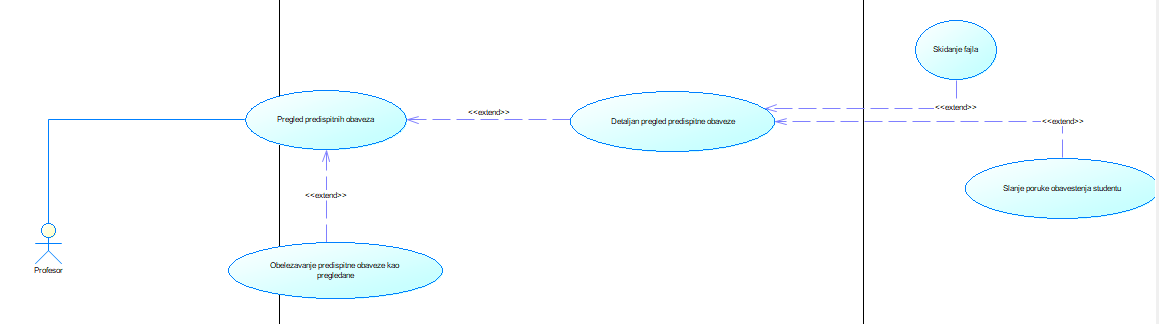
## 4.3. Slučajevi korišćenja

Slučajevi korišćenja predstavljaju akcije koje akteri mogu da izvrše sa funkcionalnostima softvera. Slučajevi korišćenja su prikazani u UseCase dijagramu koristeći PowerDesigner alat.



Slika 4.3.1 – Slučajevi korišćenja svih aktera sistema

|  |  |
| --- | --- |
| 4.3.1. | UseCase profesora |



Slika 4.3.1.1 – Slučajevi korišćenja profesora

### 4.3.1.1 Pregled predispitnih obaveza

* **Opis slučaja korišćenja:**

Profesor moze da pregleda predispitne obaveze

* **Preduslovi:**

1.Korisnik mora biti ulogovan i mora da poseduje odgovaracuju rolu

* **Osnovni scenario:**

1.Profesor bira skolsku godinu

2.Profesor bira predmet

3.Profesor bira obavezu

4.Profesor bira broj obaveze

5.Profesor bira studenta

6.Profesor pritiska dugme prikazi

7.Sistem vraca poene studenta na toj obavezi kao i da li je poslao domaci zadatak

8.Profesor klikce na dugme za detaljan pregled predispitne obaveze

* **Postuslovi**:

1.Profesor je uspesno pogledao predispitne obaveze

### 4.3.1.2 Obelezavanje predispitne obaveze kao pregledane

* **Opis slučaja korišćenja:**

Profesor moze da obelezi predispitnu obavezu kao pregledanu

* **Preduslovi:**

1.Korisnik mora biti ulogovan i mora da poseduje odgovaracuju rolu

* **Osnovni scenario:**

1.Profesor pritiska checkbox dugme

2.Sistem automatski cuva profesorov izbor

* **Postuslovi:**

1.Profesor je uspesno pregledao predispitnu obavezu

### 4.3.1.3 Detaljan pregled predispitne obaveze

* **Opis slučaja korišćenja:**

Profesor gleda detaljne informacije o predispitnoj obavezi.

* **Preduslovi:**

1.Korisnik mora biti ulogovan i mora da poseduje odgovaracuju rolu

* **Osnovni scenario:**

1.Profesor pritiska dugme prikazi

2.Sistem vraca detaljne informacije o predispitnim obavezama[ALT-1][ALT-2]

3.Profesor klikom na dugme Zatvori zatvara detaljan pregled predispitne obaveze

* **Postuslovi:**
  1. Profesor je pogledao detaljan pregled predispitne obaveze.

### 4.3.1.4 Skidanje fajla

* **Opis slučaja korišćenja:**

Profesor skida fajla u kom se nalazi predispitna obaveza

* **Preduslovi:**

1.Korisnik mora biti ulogovan i mora da poseduje odgovaracuju rolu

* **Osnovni scenario:**

1.Profesor klikom na dugme Preuzmi otvara popup prozor

2. Profesor bira gde zeli da sacuva fajl

3.Profesor klikom na potvrdi uspesno cuva fajl

* **Postuslovi:**
  1. Profesor je uspesno skinuo fajl

### 4.3.1.5 Slanje poruke obavestenja studentu

* **Opis slučaja korišćenja:**

Profesor salje poruku obavestenja studentu

* **Preduslovi:**

1.Korisnik mora biti ulogovan i mora da poseduje odgovaracuju rolu

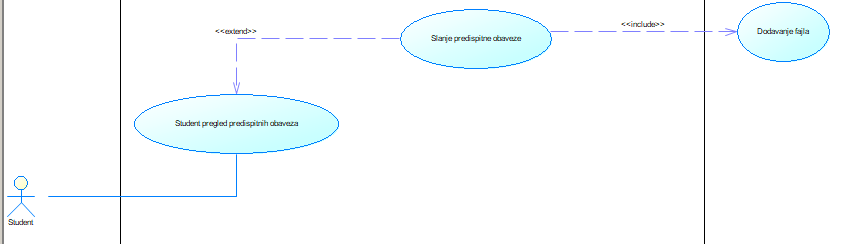
* **Osnovni scenario:**

1.Profesor popunjava polje za tekst poruke

2.Profesor klikom na posalji salje poruku studentu

* **Postuslovi:**
  1. Profesor je uspesno poslao poruku obavestenja

|  |  |
| --- | --- |
| 4.3.2. | UseCase studenta |



Slika 4.3.2.1 – Slučajevi korišćenja studenta

### 4.3.2.1 Student pregled predispitnih obaveza

* **Opis slučaja korišćenja:**

Student pregleda svoje predispitne obaveze

* **Preduslovi:**
  1. Student je ulogovan na sistem I ima odgovarajucu rolu.
* **Osnovni scenario:**

1.Student bira predmet

2.Student bira obavezu

3.Student bira broj obaveze

4.Student pritiska dugme Pregledaj

5.Sistem vraca sve predispitne obaveze koje je student izabrao[ALT-1]

* **Postuslovi:**

1. Podaci su pregledani.

### 4.3.2.2 Slanje predispitne obaveze

* **Opis slučaja korišćenja:**

Student moze da posalje predispitnu obavezu preko sistema.

* **Preduslovi:**

1.Student je ulogovan na sistem I ima odgovarajucu rolu.

* **Osnovni scenario:**

1.Student pritiska dugme prikazi

2.Sistem popunjava studentu pozdravnu poruku, sistem vraca informaciju do kada je odredjen rok za tu predispitnu obavezu kao i ostale informacije

4.Student pritiska dugme Posalji da posalje predispitnu obavezu[ALT-1][EX-1]

* **Postuslovi:**

1. Student je uspesno poslao predispitnu obavezu

### 4.3.2.3 Dodavanje fajla

* **Opis slučaja korišćenja:**

Student dodaje fajl prilikom slanja predispitne obaveze.

* **Preduslovi:**

1. Student je ulogovan na sistem I ima odgovarajucu rolu

* **Osnovni scenariji:**

1.Student pritiska dugme dodaj fajl

2. Sistem otvara prozor

3.Student bira koji fajl zeli da doda[EX-1][EX-2]

* **Postuslovi:**

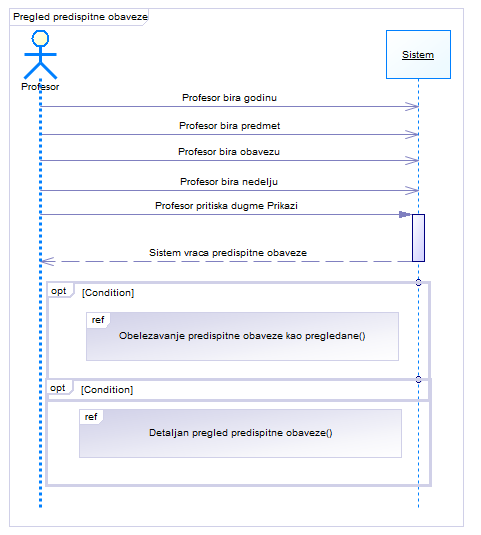
1. Student je uspesno dodao fajl

## 4.4. Apstraktni dijagram sekvenci

Apstraktni dijagram sekvenci ili sistem sekvencijalni dijagrami su interakcioni dijagrami koji detaljno opisuju kako se neke operacije izvršavaju. U ovom projektnom zadataku, sistem sekvencijalni dijagrami će biti vezani za sve operacije koje smo objasnili u dijagramu slučajeva korišćenja sistema.

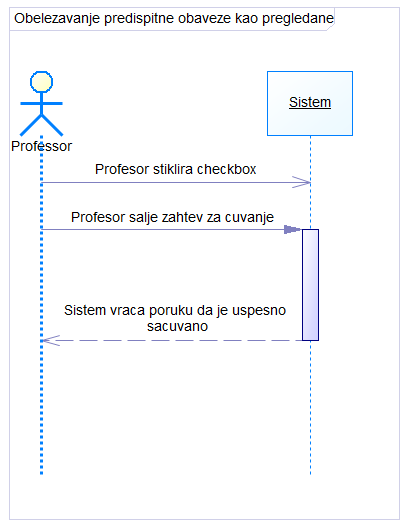
|  |  |  |
| --- | --- | --- |
| 4.4.1. |  | Sistem sekvencijalni dijagram za profesora |

### 4.4.1.1 Pregled predispitnih obaveza



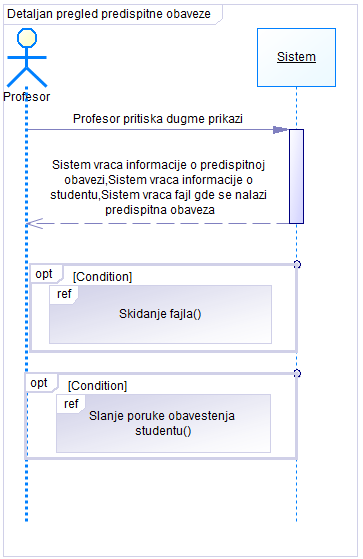
Kako bi profesor pregledao predispitne obaveze on mora da unese parametre prema kojim se pretrazuju predispiotne obaveze u sistemu. Kada to uradi potrebno je potvrditi klikom na dugme Potvrdi. Sistem ce prilikom te interakcije vratiti sve predispitne obaveze koje je profesor izabrao.

### 4.4.1.2 Obelezavanje predispitne obaveze kao pregledane



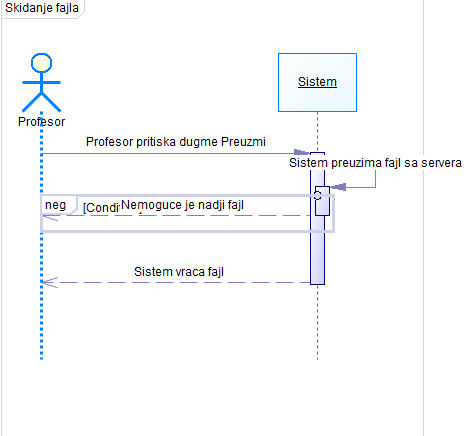
Profesor prilikom ove interakcije sa sistemo obelezava predispitnu obavezu kao pregledanu I sistem to automatski cuva.

### 4.4.1.3 Detaljan pregled predispitne obaveze



Kada je profesor izabrao koju predispitnu obavezu zeli detaljno da pregleda dobice ovakvu vrstu interakcije. Opciono moze da skine fajl ili da posalje poruku obavestenja.

### 4.4.1.4 Skidanje fajla



Profesor ovom interakcijom skida fajl.

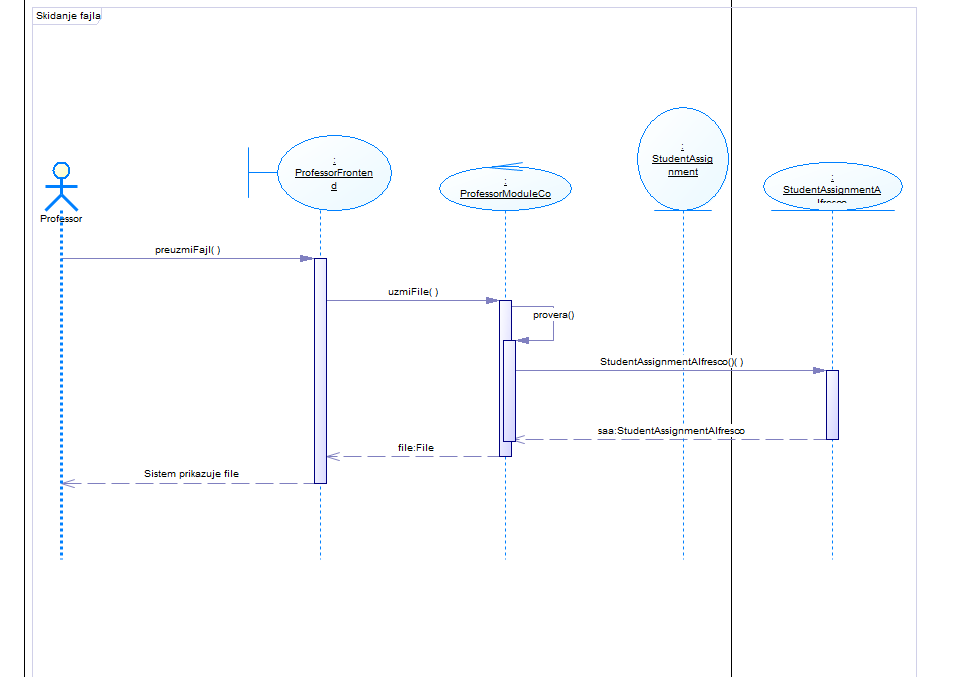
### 4.4.1.5 Slanje poruke obavestenja studentu

### 

Profesor unosi poruku po zelji I pritiskom na dugme posalji sistem dobija potrebne informacije za slanje poruke studentu.

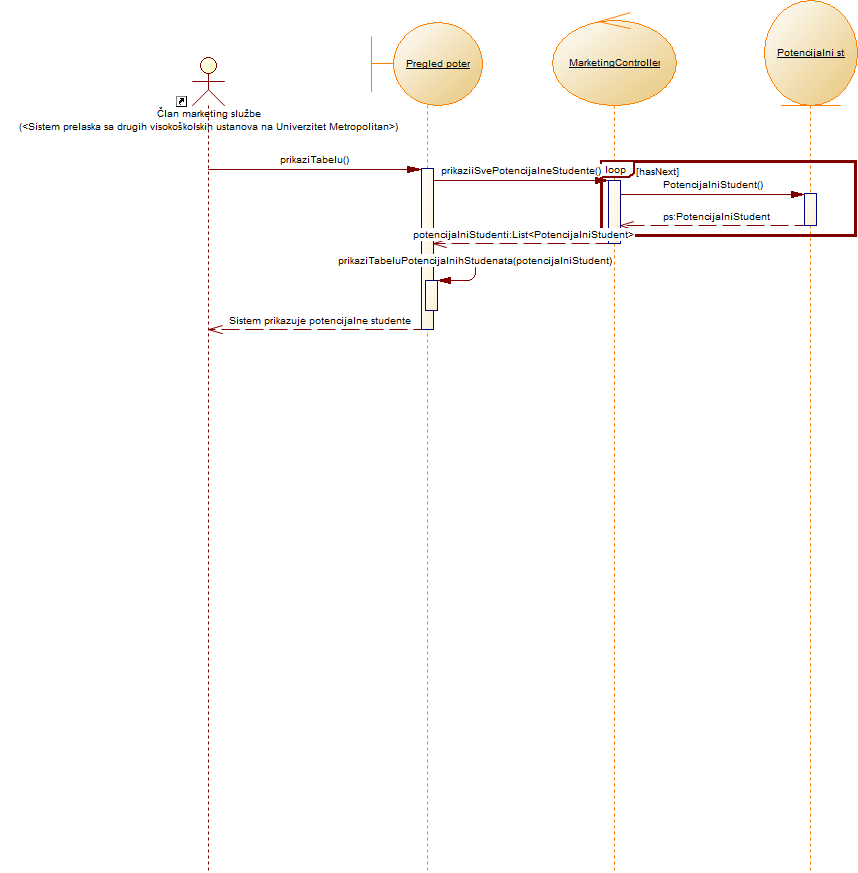
## 4.5. Sekvencijalni dijagram(nivo upotrebljenih klasa)

U narednom delu projektnog zadatka biće predstavljeni sekvencijalni dijagrami na nivou upotrebljenih klasa na kojima se prikazuju sve klase koje učestvuju u određenom procesu. Na dijagramu će biti prikazan celokupan process.



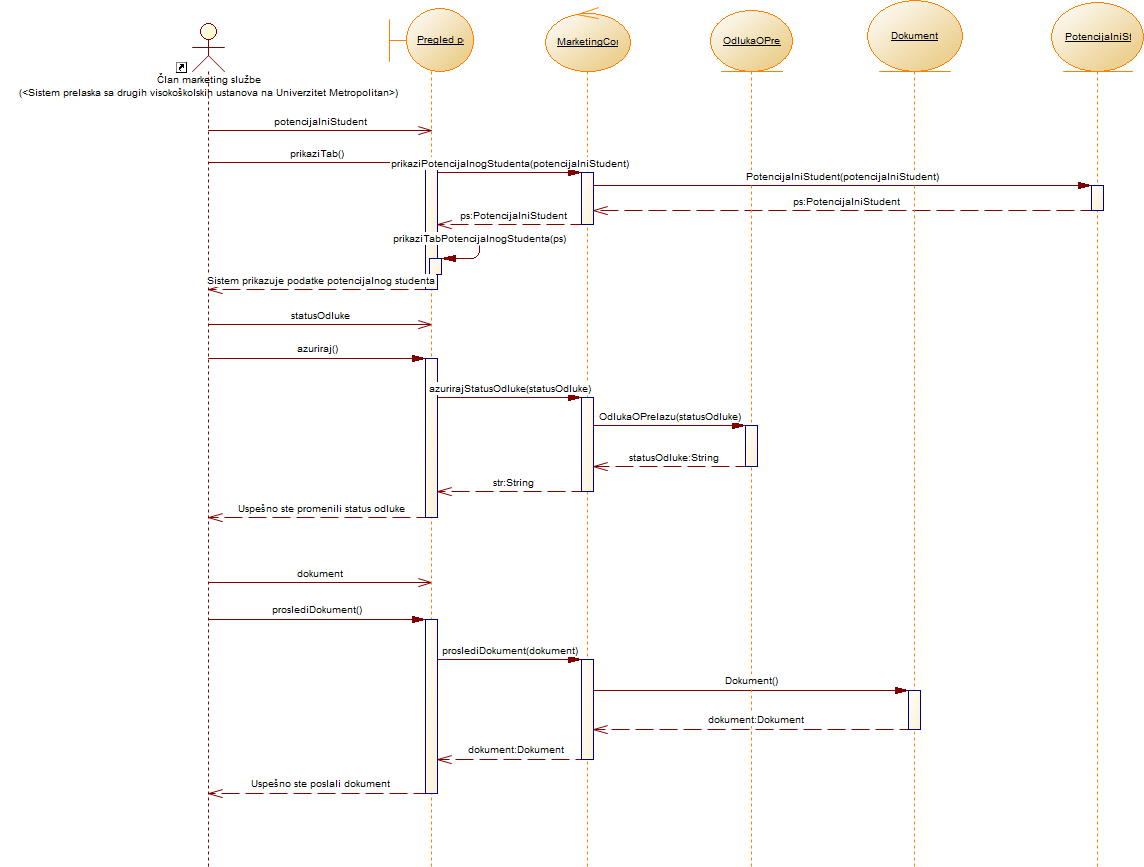
Slika 4.5.1 – Prvi slučaj skidanje fajla

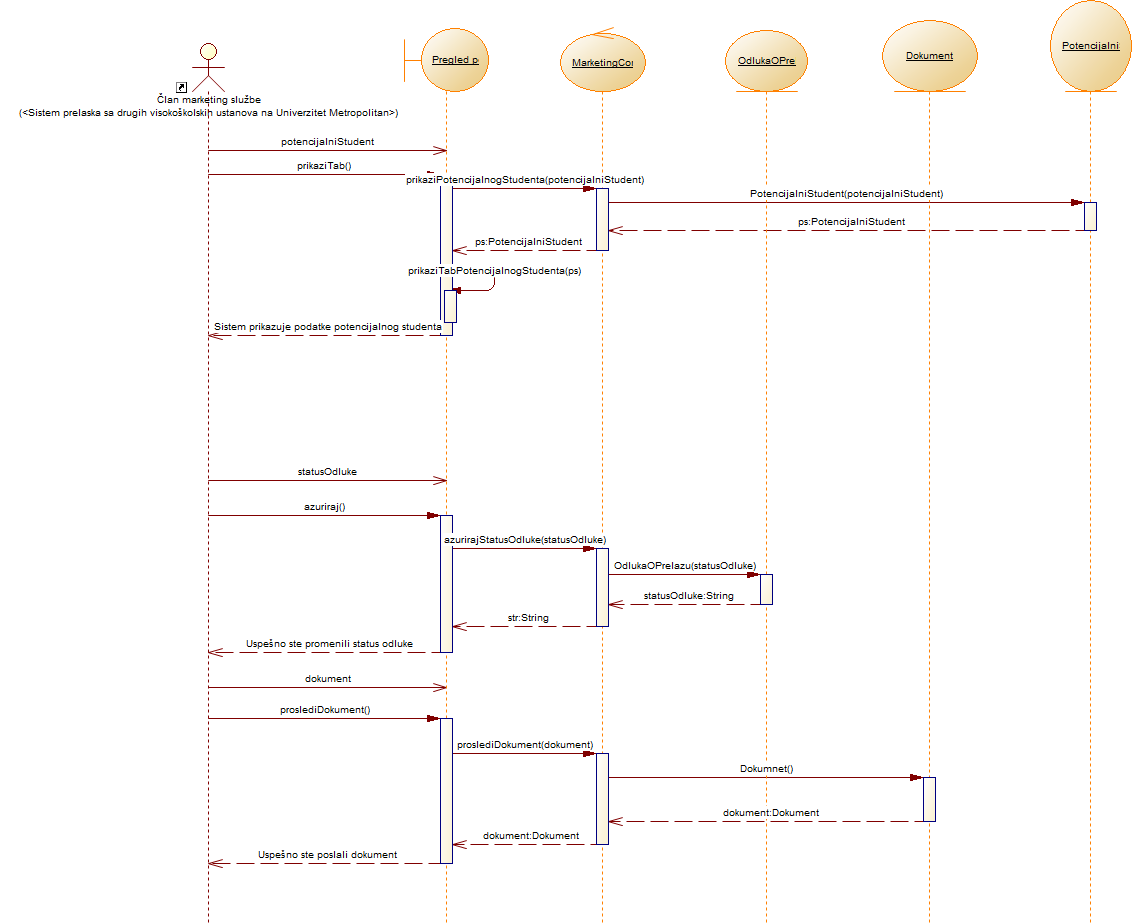
Prvi slučaj Skidanje fajla **[[4.3.1.4.]](#_4.3.1.4_Unos_broja)**.

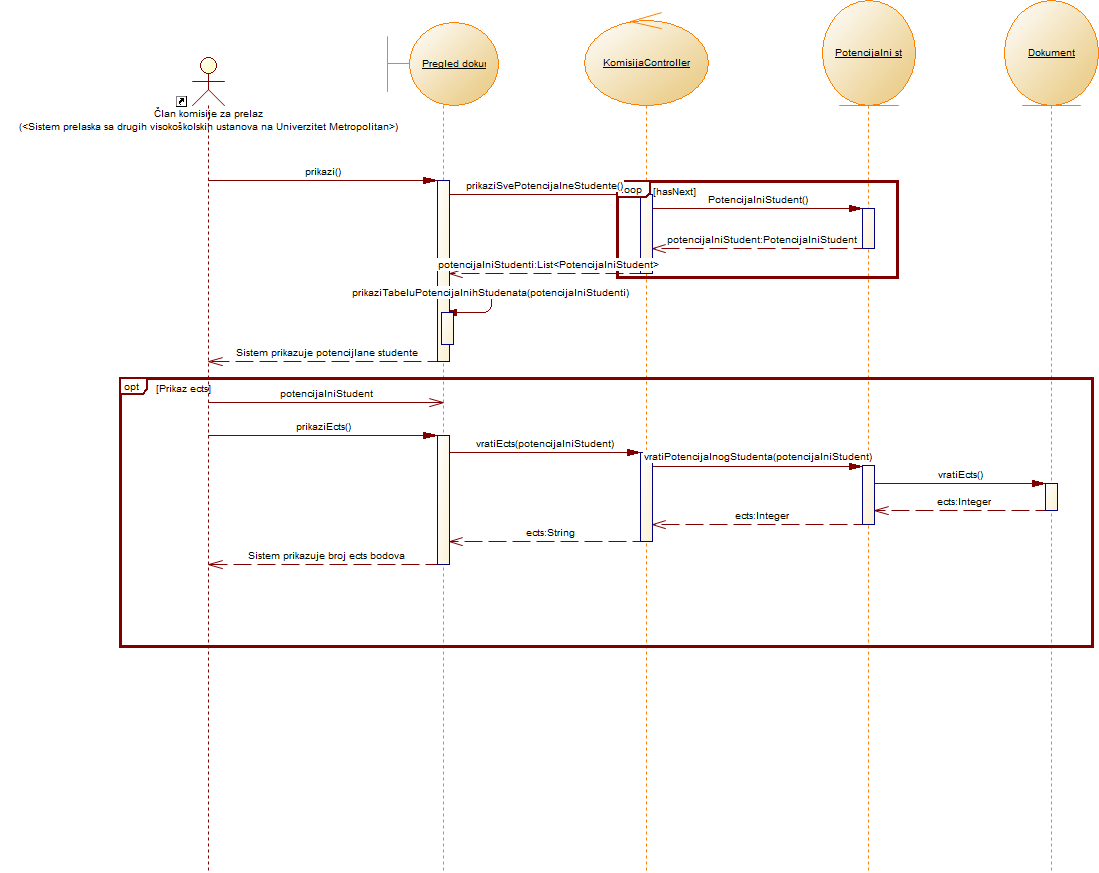


Slika 4.5.2 – Drugi slučaj pregled podataka

Drugi slučaj pregled podataka **[[4.3.1.1]](#_4.3.1.1_Pregled_podataka).**.

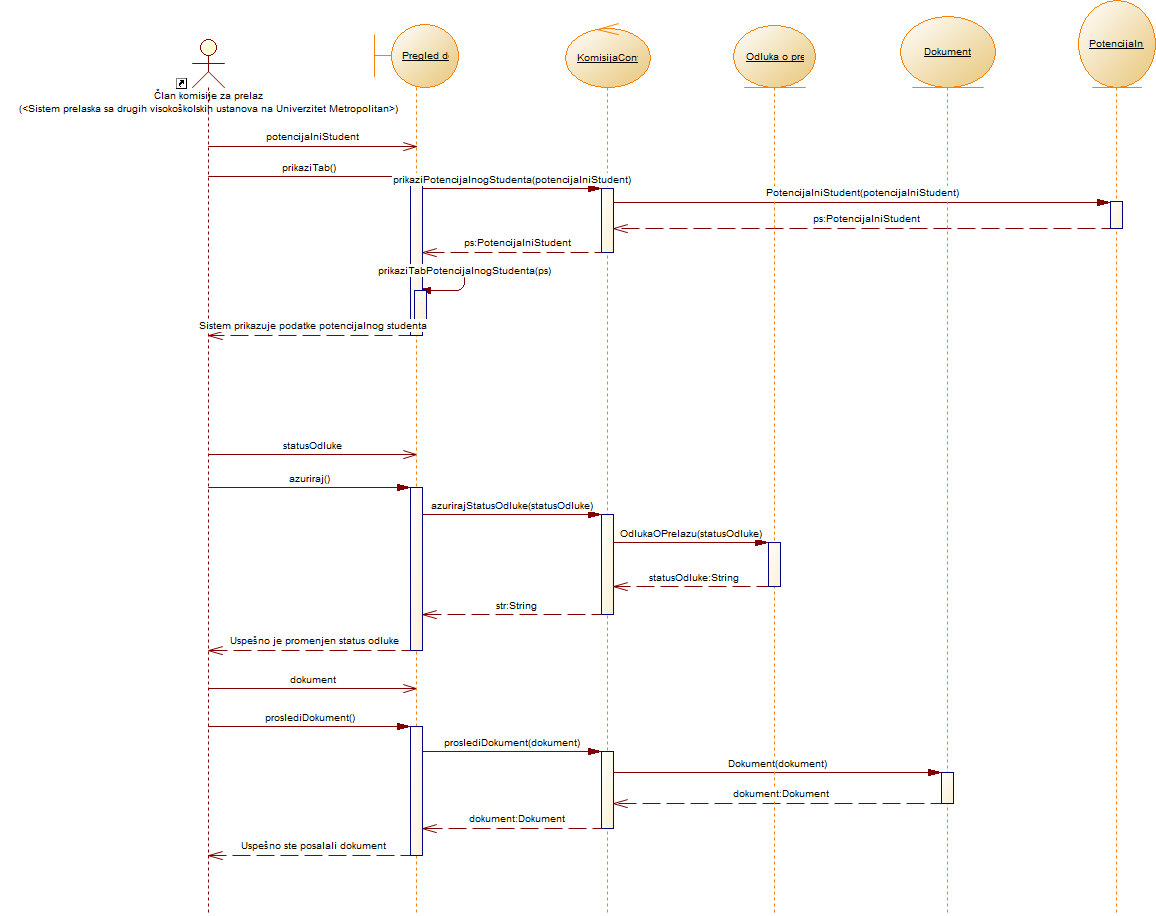


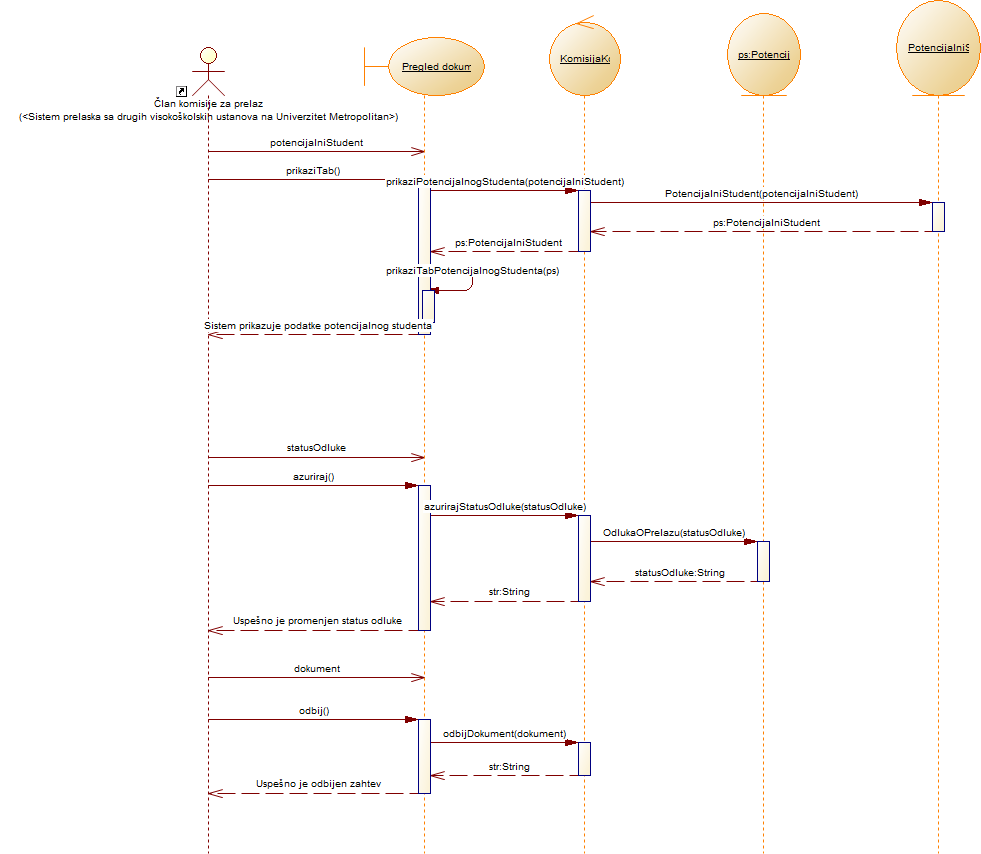


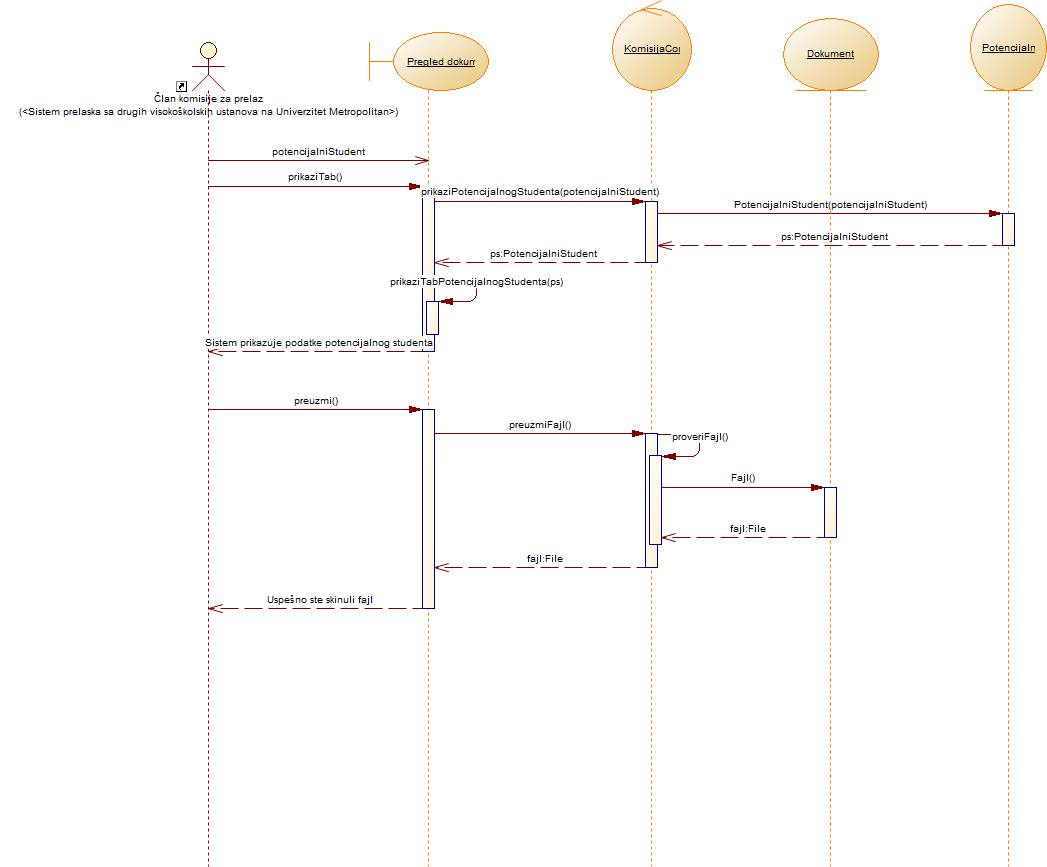


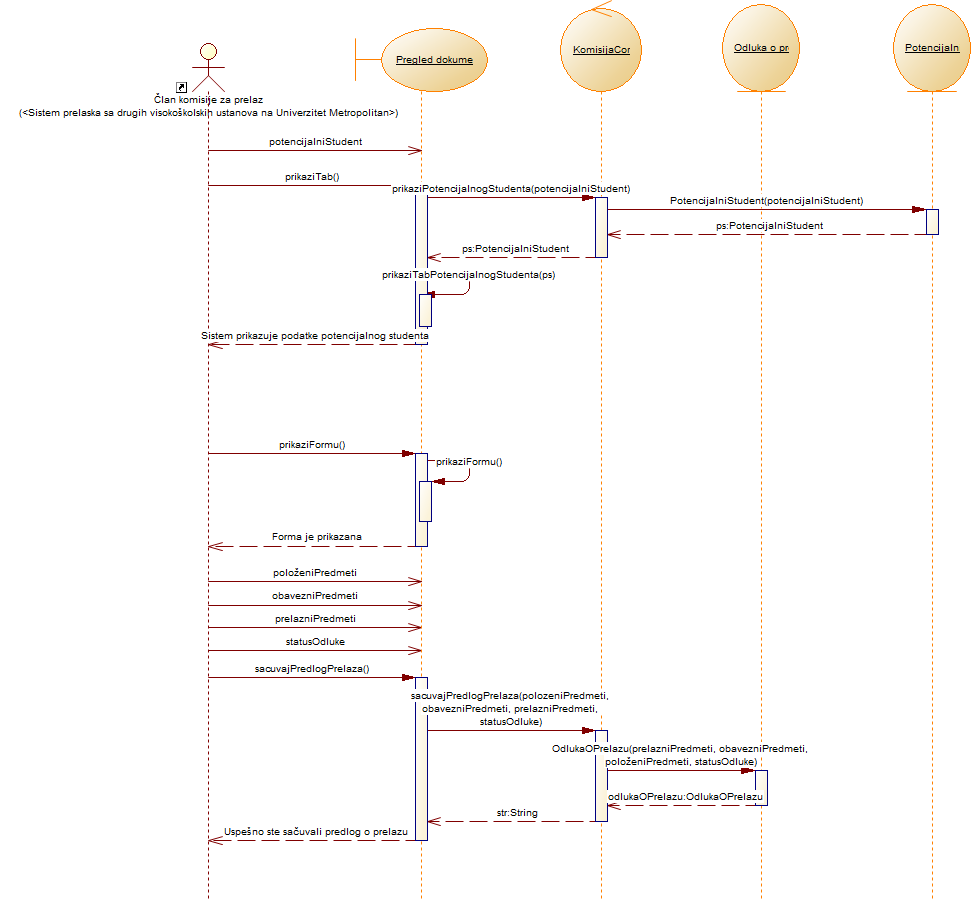
Slika 4.5.5 – Peti slučaj pregled prosleđenog dokumenta

Peti slučaj pregled prosleđenog dokumenta **[[4.3.2.1]](#_4.3.2.1_Pregled_prosleđenog)**.



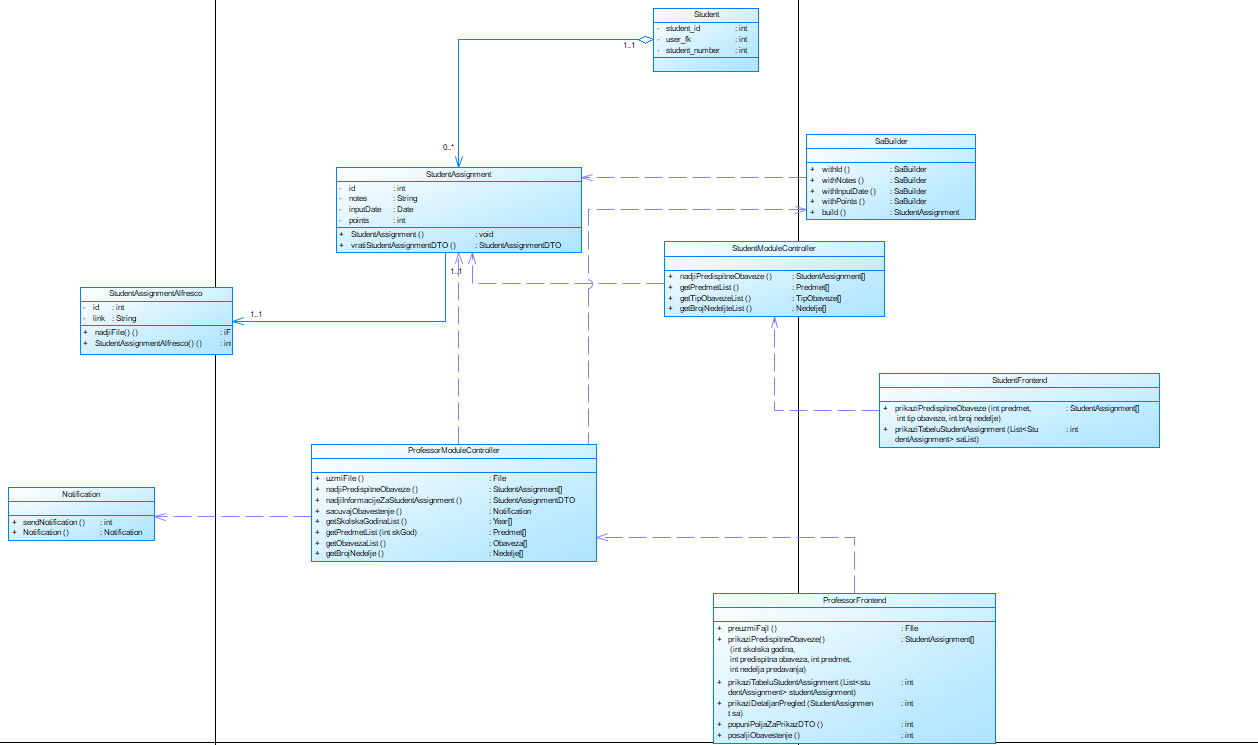






## 4.6. Klasni dijagram

U klasnom dijagramu se nalaze predviđene klase za kreiranje objekata kako bi se omogućilo potpuno funkcionisanje sistema. Pored samih klasa prikazane su i veze među njima.



|  |  |
| --- | --- |
| 4.6.1. | Metode klasnog dijagrama |

U listu metoda ćemo uvrstiti samo one metode koje nisu getter ili setter metode.

nadjiPredispitneObaveze

nadjiInformacijeZaStudentAssignment

sacuvajObavestenje

getSkolskaGodinaList

getPredmetList

getObavezaList

getBrojNedeljeName

preuzmiFajl

prikaziPredispitneObaveze()

prikaziTabeluStudentAssignment

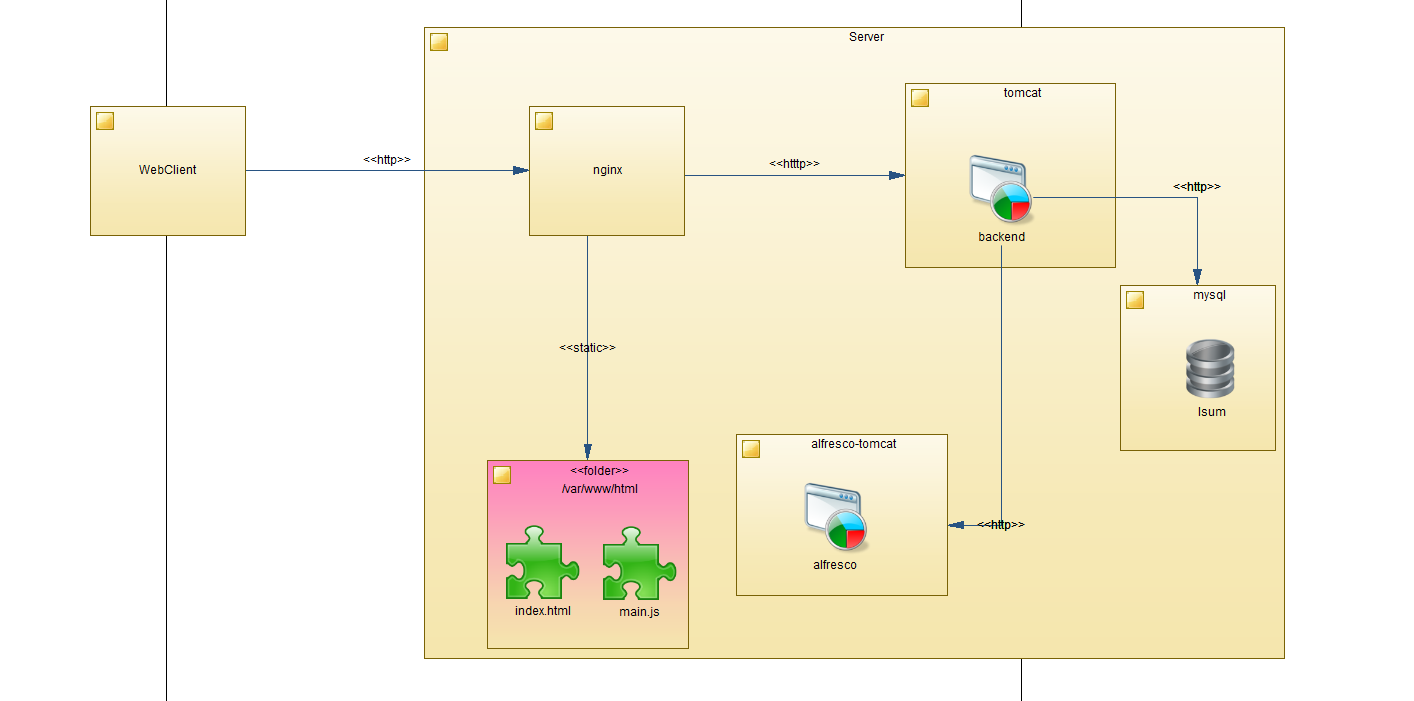
prikaziDetaljanPregled

popuniPoljaZaPrikazDTO

posaljiObavestenje

## 4.7. Arhitektura sistema

|  |  |
| --- | --- |
| 4.7.1. | Dijagram arhitekture |



Slika 4.7.1.1 – Dijagram arhitekture

Dijagram arhitekture čine baza podataka, klijentska I serverska aplikacija sa svojim komponentama I internet sa pretraživačem.

# 5. Zaključak

Ovaj sistem će omogućiti brz I lak prelazak potencijalnih studenata na Univerzitet Metropolitan.

# 6. Reference

**1.** SE201- Uvod u softversko inženjerstvo, Prof.dr Dragan Domazet.