



**Jesenji semestar  
2016/17 PROJEKTNI ZADATAK**

**Dokumentacija MetChat sistema**

**AUTORI  
Ajla Džeković – 2470  
Velibor Bačujkov - 2493**

ajla.dzekovic.2470@metropolitan.ac.rs  
velibor.bacujkov.2493@metropolitan.ac.rs

Beograd, Oktobar 2016

## Table of Contents

1. Uvod .....	5
1.1 Svrha .....	5
1.2 Područje projekta .....	5
1.3 Definicije, akronimi i skraćenice .....	5
1.4 Reference .....	5
1.5 Opis problema .....	5
2. Generalni opis sistema .....	2
2.1 Perspektiva proizvoda .....	2
2.2 Funkcije proizvoda .....	2
2.4 Generalna ograničenja .....	3
2.5 Pretpostavke i ograničenja .....	3
3. Specifični zahtevi .....	3
3.1 Zahtevi eksternog interfejsa .....	3
3.1.1 Korisnički interfejs .....	3
3.1.2 Hardverski interfejs .....	3
3.1.4 Komunikacioni interfejs .....	4
3.1.5 Registracija korisnika .....	4
3.2 Funkcionalni zahtevi .....	4
3.2.1 Korisnik ima pristup sistemu .....	4
3.2.2 Korisnik kreira svoj nickname .....	4
3.2.3 Korisnik može da se priključi javnom chat-u .....	5
3.2.4 Korisnik može da ima privatni chat sa drugim korisnicima .....	5
3.2.5 Korisnik može da vidi ostale korisnike na mreži .....	5
3.2.6 Korisnik može da vidi kada se neki drugi korisnik priključi chatu .....	5
3.2.7 Korisnik može da vidi kada neko od korisnika ode sa sistema .....	5
3.2.8 Korisnik može da vidi kada neki drugi korisnik kuca poruku u tom chat-u .....	6
3.2.9 Korisnik može da pošalje prilog .....	6
3.2.10 Korisnik može da pošalje prilog sa lokalnog uređaja .....	6
3.3 Dijagrami slučajeva koriscenja .....	7
3.3.1 Dijagram slučaja korišćenja od strane korisnika .....	7
3.4 Nefunkcionalni zahtevi .....	10
3.4.1 Performanse .....	10
3.4.2 Pouzdanost .....	10
3.4.3 Upotrebljivost .....	10

3.4.4 Integritet .....	10
3.5 Zahtevi baze podataka .....	11
3.5.1 Konceptualni model baze podataka.....	11
3.5.2 Fizicki model baze podataka .....	12
3.5 Dijagrami aktivnosti .....	12
3.5.1 Dijagram aktivnosti za pristup sistemu .....	14
3.5.2 Dijagram aktivnosti za slanje privatnih poruka.....	15
3.5.3 Dijagram aktivnosti za slanje javnih poruka.....	16
3.5.4 Dijagram aktivnosti za slanje priloga .....	17
3.5.5 Dijagram aktivnosti za slanje lokalnih priloga.....	18
3.5.6 Dijagram aktivnosti za pregled korisnika .....	19
3.5.7 Dijagram aktivnosti za citanje poruka.....	20
3.5.8 Dijagram aktivnosti za status pisanja .....	21
4. Modeli analize sistema.....	22
4.1 Dijagrami sekvenci .....	22
4.2.1 Dijagram sekvenci za pristup .....	23
4.2.2 Dijagram sekvenci za slanje privatnih poruka.....	24
4.2.3 Dijagram sekvenci za slanje javnih poruka.....	25
4.2.4 Dijagram sekvenci za slanje priloga .....	26
4.2.5 Dijagram sekvenci za lokalno slanje priloga.....	27
4.2.5 Dijagram sekvenci za pregled korisnika .....	28
4.2.6 Dijagram sekvenci za citanje poruka.....	28
4.2.7 Dijagram sekvenci za prikaz statusa pisanja .....	29
4.3 Dijagrami komunikacije.....	30
4.3.1 Dijagram komunikacije za pristup.....	30
4.3.2 Dijagram komunikacije za slanje privatnih poruka .....	31
4.3.3 Dijagram komunikacije za slanje javnih poruka .....	31
4.3.4 Dijagram komunikacije za slanje priloga.....	32
4.3.6 Dijagram komunikacije za slanje lokalnih priloga .....	32
4.3.8 Dijagram komunikacije za citanje poruka .....	33
4.3.9 Dijagram komunikacije za status pisanja .....	34
5. Implementacioni dijagrami .....	35
5.1 Dijagram realizacije.....	35

5.2 Infrastrukturni dijagram.....	36
5.3 Dijagram arhitekture.....	37
5.4 Dijagram mreže.....	38
6. Plan testiranja .....	39
7. Zaključak.....	39

## 1. Uvod

U okviru ovog dokumenta biće detaljno objašnjeni principi funkcionisanja sistema chat aplikacije. Oni će biti objašnjeni kroz niz dijagrama i tabela. Softver koriscen u izradi je PowerDesigner, a programski jezik u kome se sistem razvijao je Spring Boot I Angular 2.

### 1.1 Svrha

Glavna svrha ovog sistema je da pruži mesto na kome će korisnici da komuniciraju, kako u privatnom, tako i u javnom chat prozoru.

### 1.2 Područje projekta

- (1) Ime sistema je “MetChat”.
- (2) Ovaj sistem predstavlja prostor za razmenu poruka.
- (3) Sistem pruža korisnicima mogućnost online razgovora.

### 1.3 Definicije, akronimi i skraćenice

MT - MetChat

LAN – Local Area Network

HDD – Hard Disc Drive

CSS – Cascading Style Sheets – jezik formatiranja

HTML - HyperText Markup Language

### 1.4 Reference

- (1) Software engineering – A Practitioner’s Approach (Roger S. Pressman)
- (2) Software engineering (James Peters)
- (3) Software engineering – Tenth Edition, Pearson Education (Ian Sommerville)

### 1.5 Opis problema

MT sistem je projekat koji simulira osnovne funkcionalnosti jedne chat aplikacije. Sistem omogućava korisniku da pristupi sistemu. Pomoću tog naloga korisnik pristupa sistemu i dobija sve privilegije korisnika sistema.

Korisnik dobija mogućnost da razmenjuje poruke sa drugim korisnicima sistema. Poruke mogu biti u okviru teksta, priče, pesme, i ostalih tekstualnih objava, kao i slika, fajlova ili linkova. Korisnik ima mogućnost da te poruke šalje drugim korisnicima. Takođe, korisnik ima mogućnost da čita poruke drugih korisnika kao i uvid u listu ostalih korisnika koji koriste sistem.

## 2. Generalni opis sistema

Sistem podrazumeva aplikaciju za unošenje informacija o korisniku i samom MT sadržaju. Sistem se sastoji od veb aplikacije koja je dostupna na svim računarima. Sistem omogućava korisnicima brz unos slanje željenog sadržaja i razmena istog sa drugim korisnicima sistema.

### 2.1 Perspektiva proizvoda

MT je nezavisna aplikacija. Interfejs sistema, korisnički interfejs i hardverski interfejs su definisani sledećim redosledom:

#### 2.1.1. Interfejs sistema:

Korisnik sistema je u mogućnosti da deli detalje dostupne u bazi podataka kroz internet konekciju.

#### 2.1.2. Korisnički interfejs:

Format ekrana i struktura liste objava treba biti takva da je svim korisnicima lako dostupna i jednostavna za korišćenje. Sistem mora biti korisnički jednostavan i interaktivan. Funkcionalnosti koje sistem obezbeđuje treba adaptirati za sve korisnike sistema.

#### 2.1.3. Hardverski interfejs:

Sistem ne zahteva nikakvu hardversku podršku, već se nalazi na mreži.

#### 2.1.4. Softverski interfejs:

Ime jezika: Angular 2 i Spring Boot.

#### 2.1.5. Komunikacioni interfejs:

Korisnici komuniciraju kroz LAN mrežu različitim klijentskim sistemima.

#### 2.1.6. Memorijski zahtevi:

Sistem zahteva minimalnu količinu RAM memorije i HDD prostora za klijentske sisteme.

#### 2.1.7. Operacije:

Sistem obezbeđuje korisniku odabir nickname-a pomoću koga korisnik pristupa sistemu. Mora postojati back-up podataka u slučaju oporavka sistema.

### 2.2 Funkcije proizvoda

Glavne funkcije uključuju:

- Login.
- Public chat.
- Privatan chat.
- Prikaz kada se neko priključi chatu.
- Prikaz kada neko ode sa chata.
- Prikaz kada neko kuca poruku.

### 2.3 Zahtev korisnika

Nije potrebno nikakvo tehničko znanje ili iskustvo u upravljanju sistemom sa korisničke strane.

## 2.4 Generalna ograničenja

- Obavezno je da sva polja budu popunjena i da ne sadrže nepotrebne podatke.
- Korisnik mora imati internet konekciju.
- Sistem mora biti korisnički jednostavan i da prikazuje sve potrebne poruke i poruke greške.
- Sistem ne sadrži interfejs ka nekim drugim sistemima.
- Mora biti u mogućnosti da izvršava više paralelnih operacija odjednom.
- Mora se izbegavati redudatnost podataka i praznih unosa.
- Sistem uvek mora biti adekvatno zaštićen.

## 2.5 Pretpostavke i ograničenja

Pretpostavlja se da će korisnik već imati ideju za koju svrhu će koristiti sistem. Buduće promene, prikaz objava, nadogradnja sistema, i sve ostale promene u okviru sistema relevantnim za korisnike, korisnici će biti obavešteni.

Obaveza je da se unapred korisnicima da do znanja kada sistem neće biti dostupan i iz kojih razloga (npr. instalacija nove verzije sistema).

## 3. Specifični zahtevi

### 3.1 Zahtevi eksternog interfejsa

#### 3.1.1 Korisnički interfejs

Interfejs mora biti lak za razumevanje. Korisnički interfejs uključuje:

- Organizaciju ekrana: Uvodni ekran će biti prikazan prvi, što će dozvoliti korisnicima da biraju između dve opcije: unos nickname-a ili izlazak sa sistema.
- Organizacija prozora: Kada korisnik odabere neku od opcija, tada će informacije biti vezane za tu određenu opciju, kao i mogućnost otvaranja u novom prozoru, što obezbeđuje prikaz više prozora odjednom.
- Format podataka: Podaci uneti od strane korisnika će biti alfa-numericki.
- Izlazna poruka: Kada se jave neki izuzetci ili greške prikazaće se poruka sa detaljima o grešci i način na koji se greška može ispraviti.

#### 3.1.2 Hardverski interfejs

Sistem mora podržati određene input-output uređaje:

Naziv	Opis	Input/Output
Monitor	Prikaz sistema i mogućnosti koje sistem pruža korisniku.	Output
Tastatura	Unos podataka	Input
Štampač	Štampanje zahtevanih podataka od strane korisnika	Output

### 3.1.3 Softverski interfejs

Sistem je nezavistan softver i ne uključuje dodatne interfejse.

### 3.1.4 Komunikacioni interfejs

Svaki korisnik mora imati pristup internet konekciji i mora imati pristup samo serveru sistema a nikako nekom od klijentskih sistema.

### 3.1.5 Registracija korisnika

Sistem neće tražiti od korisnika da prođe kroz proces registracije već samo odabir nickname-a da bi imao pristup sistemu.

## 3.2 Funkcionalni zahtevi

	Title ID	Full Description	Code	Priority	Workload	Risk	Status
1	1.	Korisnik ima pristup sistemu	REQ_0001	Undefined		Undefined	Draft
2	2.	Korisnik kreira svoj nickname	REQ_0002	Undefined		Undefined	Draft
3	3.	Korisnik može da se prikljuci javnom chatu	REQ_0003	Undefined		Undefined	Draft
4	4.	Korisnik može da ima privatan chat sa drugim korisnicima	REQ_0004	Undefined		Undefined	Draft
5	5.	Korisnik može da vidi ostale korisnike na mrezi	REQ_0005	Undefined		Undefined	Draft
6	6.	Korisnik može da vidi kada neki od korisnika ode sa mreze	REQ_0006	Undefined		Undefined	Draft
7	7.	Korisnik može da vidi kada se drugi korisnici prikljuce chatu	REQ_0007	Undefined		Undefined	Draft
8	8.	Korisnik može da vidi kada neki drugi korisnik kuca poruku u tom chatu	REQ_0008	Undefined		Undefined	Draft
9	9.	Korisnik može da pošalje prilog	REQ_0009	Undefined		Undefined	Draft
→	10.	Korisnik može da pošalje prilog sa lokalnog uređaja	REQ_0010	Undefined		Undefined	Draft

Slika 1. Funkcionalni zahtevi sistema

### 3.2.1 Korisnik ima pristup sistemu

#### 3.2.1.1 Uvod

Svrha ovog sistema je da omogući korisnicima da pristupe sistemu.

#### 3.2.1.2 Ulaz

Korisnik ima uvid u početni ekran sistema.

#### 3.2.1.3 Proces

Ulazni podaci se čuvaju u bazi podataka.

#### 3.2.1.4 Izlaz

Korisnik jasno zna kako da pristupi sistemu.

### 3.2.2 Korisnik kreira svoj nickname

#### 3.2.2.1 Uvod

Svrha ovog sistema je da omogući korisnicima da odaberu svoj nickname.

#### 3.2.2.2 Ulaz

Korisnik popunjava polje za kreiranje nickname-a.

#### 3.2.2.3 Proces

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

#### 3.2.2.4 Izlaz

Korisnik je pristupio sistemu.



### **3.2.3 Korisnik može da se priključi javnom chat-u**

#### **3.2.3.1 Uvod**

Svrha ovog zahteva je da omogući korisnicima da se priključe javnom chat-u.

#### **3.2.3.2 Ulaz**

Korisnik se odabirom nickname-a automatski priključuje javnom chat-u.

#### **3.2.3.3 Izlaz**

Korisnik ima pristup javnom chat-u.

### **3.2.4 Korisnik može da ima privatan chat sa drugim korisnicima**

#### **3.2.4.1 Uvod**

Svrha ovog sistema je da omogući korisnicima da kreiraju privatan chat sa drugim korisnicima.

#### **3.2.4.2 Ulaz**

Korisnik bira drugog korisnika sa kojim otvara privatan chat.

#### **3.2.4.3 Proces**

Podaci o kreiranom chatu i sadržaju se prikazuju na sistemu i čuvaju u bazi podataka.

#### **3.2.4.4 Izlaz**

Korisnik je kreirao privatan chat.

### **3.2.5 Korisnik može da vidi ostale korisnike na mreži**

#### **3.2.5.1 Uvod**

Svrha ovog sistema je da omogući korisnicima da pretražuju ostale korisnike koji su na sistemu.

#### **3.2.5.2 Ulaz**

Korisnik iz liste korisnika bira željenog korisnika

#### **3.2.5.3 Proces**

Svaka izmena se čuva u bazi podataka.

#### **3.2.5.4 Izlaz**

Korisnik je uspešno pronašao željenog korisnika.

### **3.2.6 Korisnik može da vidi kada se neki drugi korisnik priključi chatu**

#### **3.2.6.1 Uvod**

Svrha ovog sistema je da prikaže kada se neko priključi chat-u.

#### **3.2.6.2 Ulaz**

Korisnik dobija poruku da je neki korisnik pristupio chat-u.

#### **3.2.6.4 Izlaz**

Korisnik ima uvid u to ko se priključuje chat-u.

### **3.2.7 Korisnik može da vidi kada neko od korisnika ode sa sistema**

#### **3.2.7.1 Uvod**

Svrha ovog sistema je da prikaže kada neko ode iz chat-a.

#### **3.2.7.2 Ulaz**

Korisnik dobija poruku kada se neko od korisnika odjavi sa chat-a.

#### **3.2.7.3 Izlaz**

Korisnik ima uvid u to ko odlazi iz chat-a.

### **3.2.8 Korisnik može da vidi kada neki drugi korisnik kuca poruku u tom chat-u**

#### **3.2.8.1 Uvod**

Svrha ovog sistema je da prikaže kada neki korisnik kuca poruku u trenutnom chat-u

#### **3.2.8.3 Proces**

Prikaz kucanja poruke

#### **3.2.8.4 Izlaz**

Korisnik vidi kada neki korisnik kuca poruku u tom chatu

### **3.2.9 Korisnik može da pošalje prilog**

#### **3.2.8.1 Uvod**

Svrha ovog sistema je da omoguci korisnicima da šalju priloge u chatu

#### **3.2.8.2 Ulaz**

Korisnik daje putanju priloga (ex. URL)

#### **3.2.8.3 Proces**

Sistem obrađuje prilog i šalje ostalim korisnicima.

#### **3.2.8.4 Izlaz**

Korisnik je poslao prilog.

### **3.2.10 Korisnik može da pošalje prilog sa lokalnog uređaja**

#### **3.2.8.1 Uvod**

Svrha ovog sistema je da omoguci korisnicima da šalju priloge u chatu sa lokalnog uređaja

#### **3.2.8.2 Ulaz**

Korisnik otvara pretraživanje uređaja

#### **3.2.8.3 Proces**

Korisnik bira prilog sa uređaja koji želi da pošalje

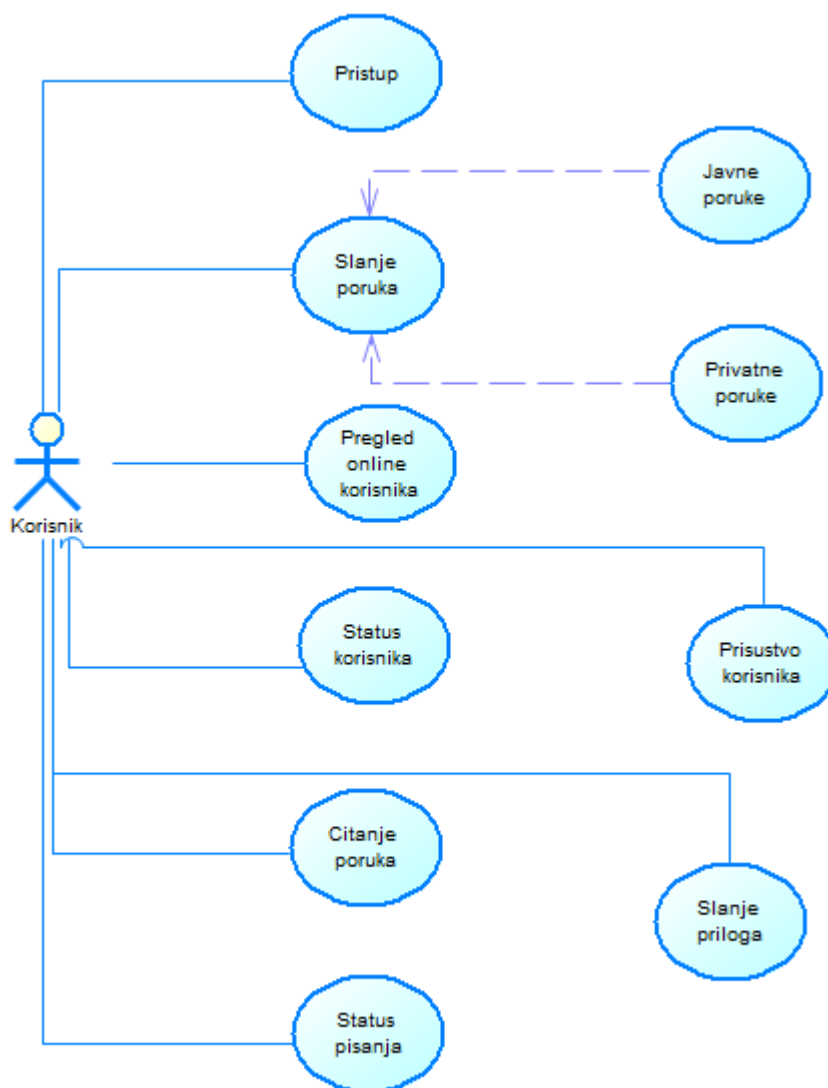
#### **3.2.8.4 Izlaz**

Korisnik je poslao prilog

### 3.3 Dijagrami slučajeva koriscenja

1. Pristup sistemu – Proces počinje od strane korisnika. Pruža mogućnost da korisnik kreira svoj nickname za pristupanje sistemu.
2. Prijavljivanje – Korisnik može da pristupi sistemu odabirom svog nickname-a. Ukoliko su uneti podaci neispravni, korisnik će biti u mogućnosti da ponovo unese ispravne korisničke podatke.
3. Chat – Korisnik ima moćnost korišćenja javnog chat prostora kao i kreiranja privatnih chat soba sa ostalim korisnicima sistema.
4. Čitanje poruka i status – Korisnik ima mogućnost čitanja poruka drugih korisnika i uvid u to ko je od korisnika na mreži.

#### 3.3.1 Dijagram slučaja korišćenja od strane korisnika



Slika 2. Dijagram slučajeva korišćenja korisnika

UseCase	Preduslovi	Koraci	Izuzeci	Postuslovi
Pristup	Korisnik mora imati pristup internet konekciji.	Korisnik pri ulasku na sistem bira svoj nickname i popunjava nickname polje.	Korisnik sa istim nickname-om vec postoji na mreži.	Korisnik je pristupio sistemu.
Javni chat	Korisnik mora da ima svoj nickname i da je pristupio sistemu.	Korisnik pri ulasku na sistem biva preusmeren u javni chat vidljiv svim korisnicima sistema.	Korisnik nije validno pristupio sistemu. Korisnik nema pristup internet konekciji.	Korisnik ima uvid u javni chat sistema.
Privatan chat	Korisnik mora da ima svoj nickname i da je pristupio sistemu.	Korisnik iz liste korisnika bira željenog korisnika sa kojim kreira privatan chat.	Traženi korisnik ne postoji. Korisnik nije validno pristupio sistemu.	Korisnik je uspešno kreirao privatan chat.
Pregled online korisnika	Korisnik mora da ima svoj nickname i da je pristupio sistemu.	Korisnik ima listu sa svim korisnicima koji koriste sistem	Korisnik nema pristup internet konekciji	Korisnik ima uvid u listu korisnika
Status korisnika	Korisnik mora da ima svoj nickname i da je pristupio sistemu.	Korisnik može da vidi kada neko od korisnika ode ili se priključi chat-u		Korisnik ima uvid u status drugih korisnika
Slanje priloga	Korisnik mora da ima svoj nickname i da je pristupio sistemu.	Korisnik može da šalje prilog u formatu URL-a	Prilog nije validnog formata	Korisnik je poslao URL priloga
Slanje lokalnih priloga	Korisnik mora da ima svoj nickname i da je pristupio sistemu.	Korisnik može da šalje priloge sa lokalnog prostora, sopstvenog PC-a, servera	Prilog nije validan. Korisnik nema pristup internet konekciji.	Korisnik je poslao prilog putem lokalnog odabira priloga.

Čitanje poruka	Korisnik mora da ima svoj nickname i da je pristupio sistemu.	Korisnik može da čita poruke drugih korisnika	Korisnik nema pristup internet konekciji	Korisnik je uspešno pročitao poruke drugih korisnika sistema.
Status pisanja	Korisnik mora da ima svoj nickname i da je pristupio sistemu.	Korisnik može da vidi ukoliko neko od drugih korisnika piše u tom chat-u.	Korisnik nema pristup internet konekciji	Korisnik ima uvid u status pisanja drugih korisnika.

### 3.4 Nefunkcionalni zahtevi

	Title ID	Full Description	Code	Priority	Workload	Risk	Status
1	1.	Administrator baze podataka upravlja korisničkim podacima	REQ_0001	Undefined		Undefined	Draft
2	2.	Administrator baze podataka upravlja porukama korisnika	REQ_0002	Undefined		Undefined	Draft
3	3.	Sistem treba biti dostupan u svakom trenutku	REQ_0003	Undefined		Undefined	Draft
4	4.	Sistem mora prikazivati informacije u realnom vremenu	REQ_0004	Undefined		Undefined	Draft
5	5.	Veb interfejs mora biti interaktivan i lak za korišćenje	REQ_0005	Undefined		Undefined	Draft
6	6.	Veb interfejs mora imati laku navigaciju	REQ_0006	Undefined		Undefined	Draft
→	7.	Samo sistem administrator ima pravo da promeni parametre sistema	REQ_0007	Undefined		Undefined	Draft

Slika 3. Nefunkcionalni zahtevi

#### 3.4.1 Performanse

3.4.1.1 Vreme odaziva MT sistema treba biti do 4 sekunde u najboljem slucaju. Vreme odaziva sistema se odnosi na vreme čekanja dok se ne pristupi sistemu, i dok sistem ne povuče podatke iz baze podataka.

3.4.1.2 Sistem ne treba da prikazuje vreme odaziva, broj korisnika sistema, kao ni njihov rast.

3.4.1.3 Sistem mora radi u realnom vremenu i da podnese više korisnika odjednom.

#### 3.4.2 Pouzdanost

3.4.2.1 Sistem treba da bude dostupan 24 časa, u svakom trenutku.

3.4.2.2 Sistem će uvek pružati informacije u realnom vremenu.

3.4.2.3 Sistem treba da ima visok stepen tolerancije. Na primer, ukoliko korisnik unese pogrešne podatke, sistem ne treba da prestane sa radom, već da prepozna pogrešan unos i korisnika obavesti porukom o grešci.

3.4.2.4 Sistem mora da bude u mogućnosti da se oporavi u slučaju kvara hardvera, nestanka struje i drugih prirodnih katastrofa i vraćanje baze podataka u prvobitno ispravno stanje.

#### 3.4.3 Upotrebljivost

3.4.3.1 Sistem treba da obezbedi lako pristupačan grafički interfejs koji je lak za korišćenje za sve korisnike sistema.

3.4.3.2 Veb interfejs treba da bude intuitivan i lak za navigaciju. Korisniku meni i opcije treba da budu laki za razumevanje.

3.4.3.3 Sva obaveštenja o grešci koje proizvodi sistem treba da budu jasne, kratke, smislene i razumljive.

#### 3.4.4 Integritet

3.4.4.1 Samo administrator sistema ima pravo da promeni parametre sistema, kao što je menjanje polise korišćenja, itd. Sistem treba da bude obezbeđen i da koristi enkripciju za zaštitu baze podataka.

3.4.4.2 Korisnik treba da potvrdi autentifikaciju pre nego što dobije pristup bilo kojim ličnim detaljima.

### 3.4.5 Samostalnost

3.4.5.1 Sistem će koristiti minimalni broj drugih sistema, kao što je baza podataka.

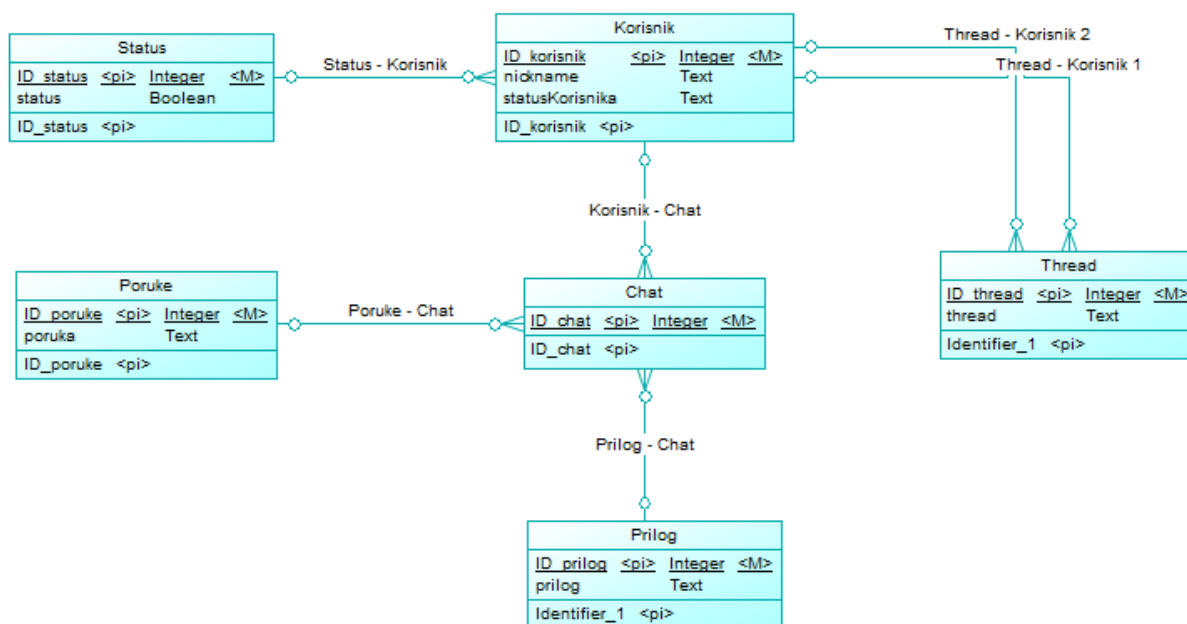
### 3.4.6 Portabilnost

3.4.6.1 Sistemu je moguće pristupiti sa bilo kog računara koji ima stabilnu internet konekciju.

## 3.5 Zahtevi baze podataka

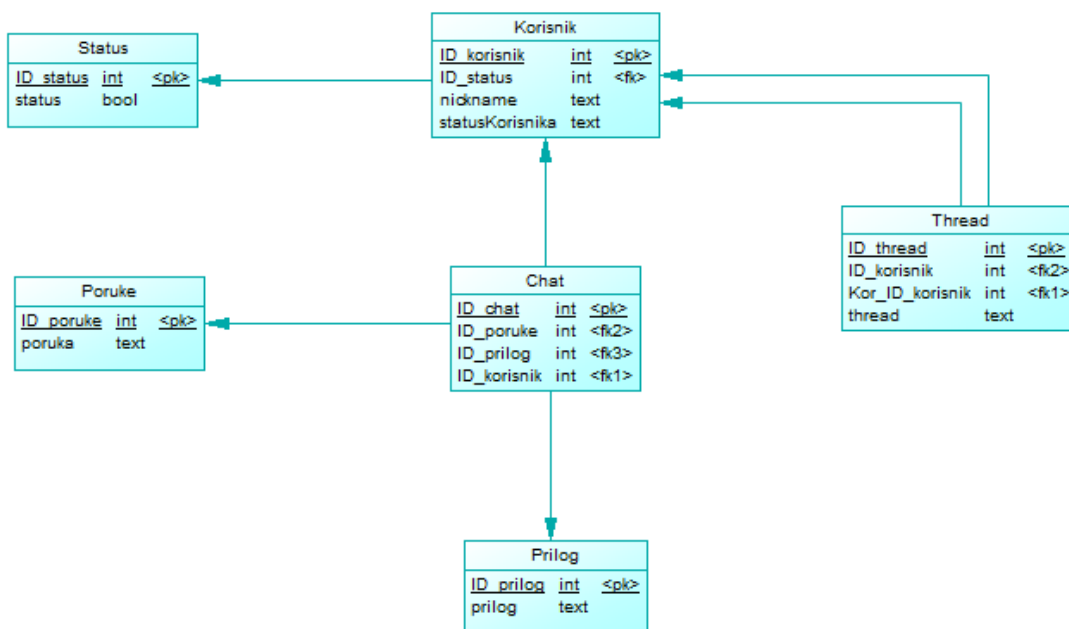
MongoDB baza podataka, fleksibilnog sadržaja.

### 3.5.1 Konceptualni model baze podataka



Slika 4. Konceptualni model

### 3.5.2 Fizicki model baze podataka



Slika 5. Fizicki model

### 3.5 Dijagrami aktivnosti

Dijagram aktivnosti prikazuje tok aktivnosti koju izvršavaju objekti, eventualno i tok objekata između koraka aktivnosti.

Aktivnost je specifikacija ponašanja koje se izvršava kroz tok izvršenja nekih podaktivnosti.

Akcija je osnovna jedinica specifikacije ponašanja koja predstavlja neku transformaciju ili obradu u sistemu. Predstavlja jedan korak u aktivnosti sistema. Akcija se dešava samo jednom na datom mestu unutar date aktivnosti.

Dijagrami aktivnosti sadrže:

- Akcije i aktivnosti
- Objekte
- Slanje događaja
- Prihvatanje događaja
- Vremenske događaje
- Kontrolne čvorove
- Početni tok
- Završni tok
- Kraj toka
- Čvorove
- Grane
- Prelaz između akcija
- Sekvencijalna grananja

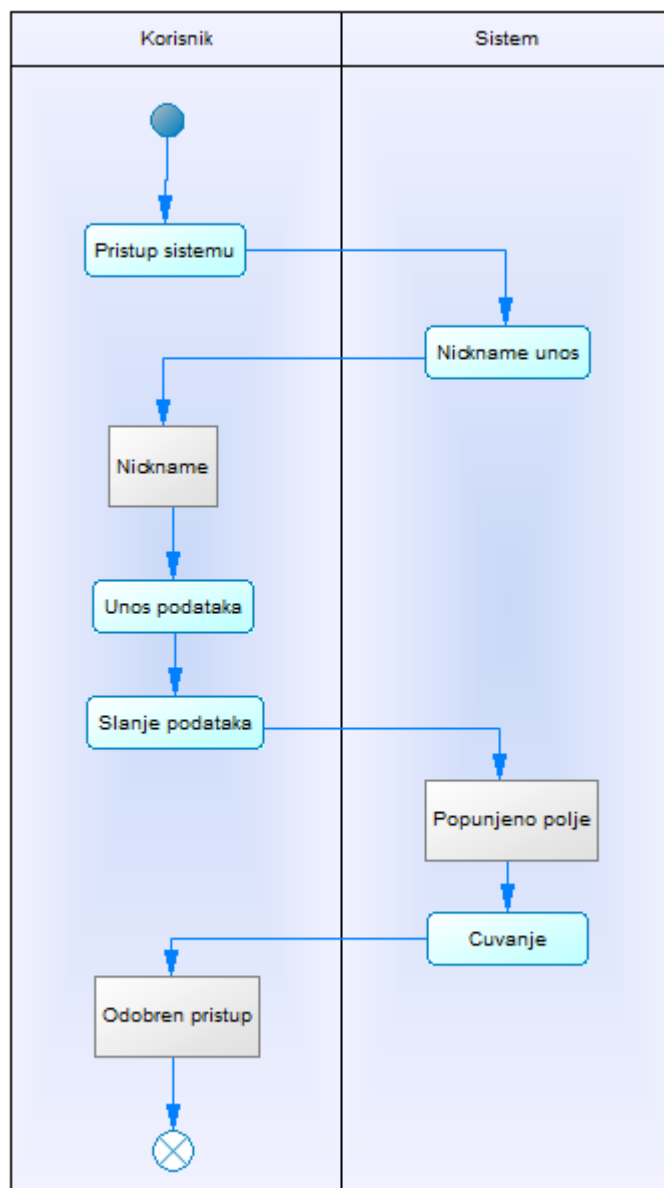


Izvršenje akcije koja pokreće neku aktivnost obuhvata izvršenje te aktivnosti i njenih akcija. Akcija se može posmatrati i kao diskretan element i kao složeno ponašanje. Kao deo strukture u modelu aktivnosti, akcija je diskretan element, a kao deo specifikacije ponašanja, akcija može pokrenuti ponašanje ili neku aktivnost.

Akcija ili aktivnost se može pokrenuti jer je izvršeno izvršavanje nekih prethodnih akcija ili aktivnosti, objekat je postao dostupan ili se dogodio neki spoljašnji događaj.

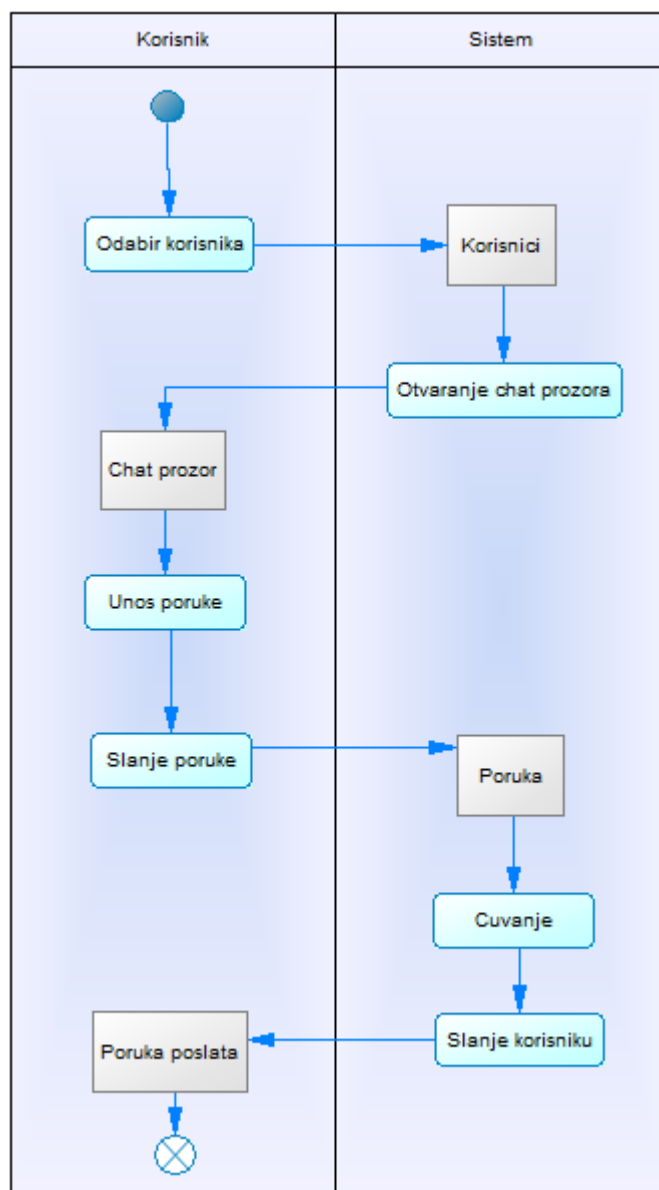
Kraj izvršavanje jedne akcije omogućava izvršenje neke druge akcije ili skupa akcija.

### 3.5.1 Dijagram aktivnosti za pristup sistemu



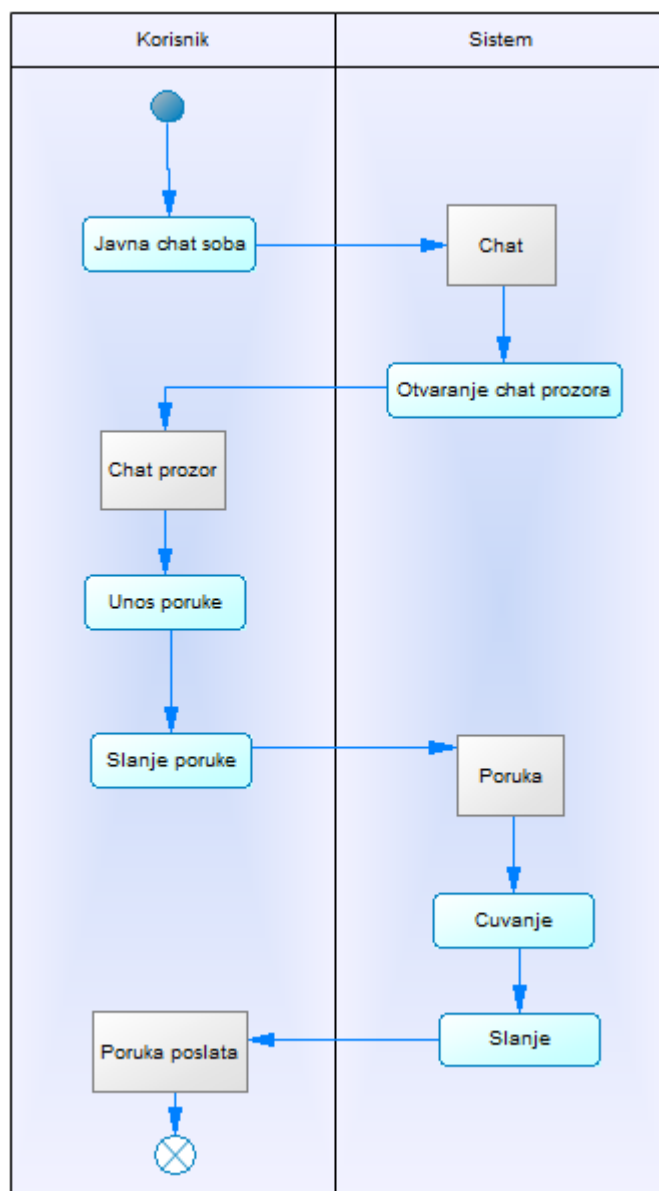
Slika 6. Dijagram aktivnosti za pristup sistemu

### 3.5.2 Dijagram aktivnosti za slanje privatnih poruka



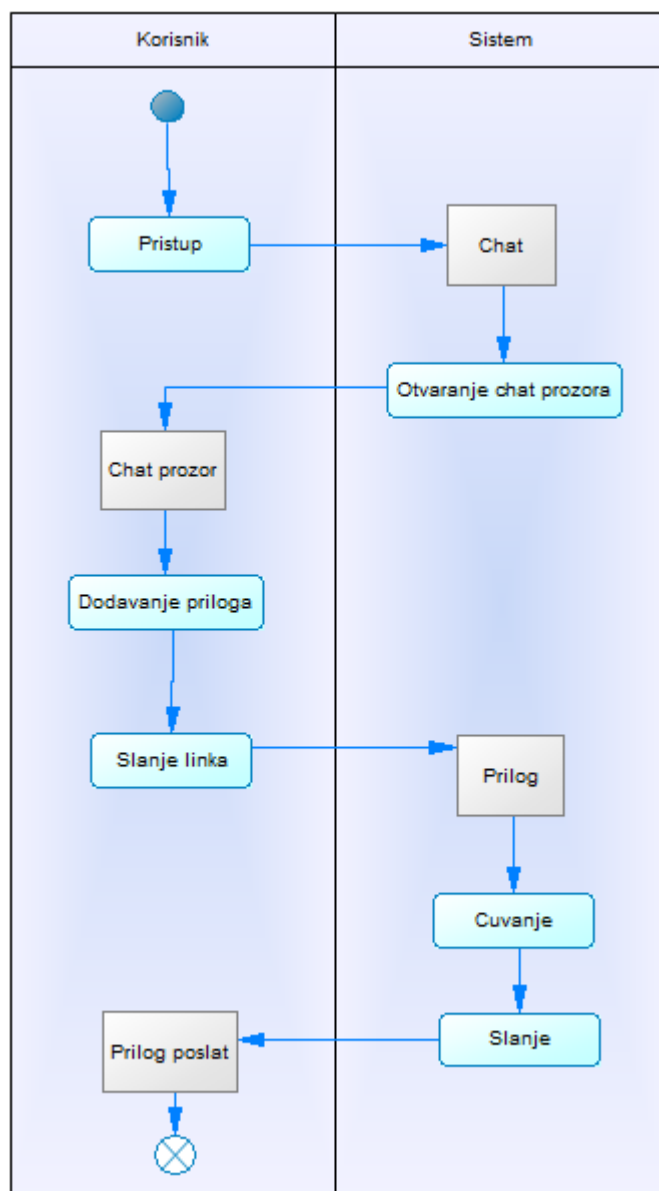
Slika 7. Dijagram aktivnosti za slanje privatnih poruka

### 3.5.3 Dijagram aktivnosti za slanje javnih poruka



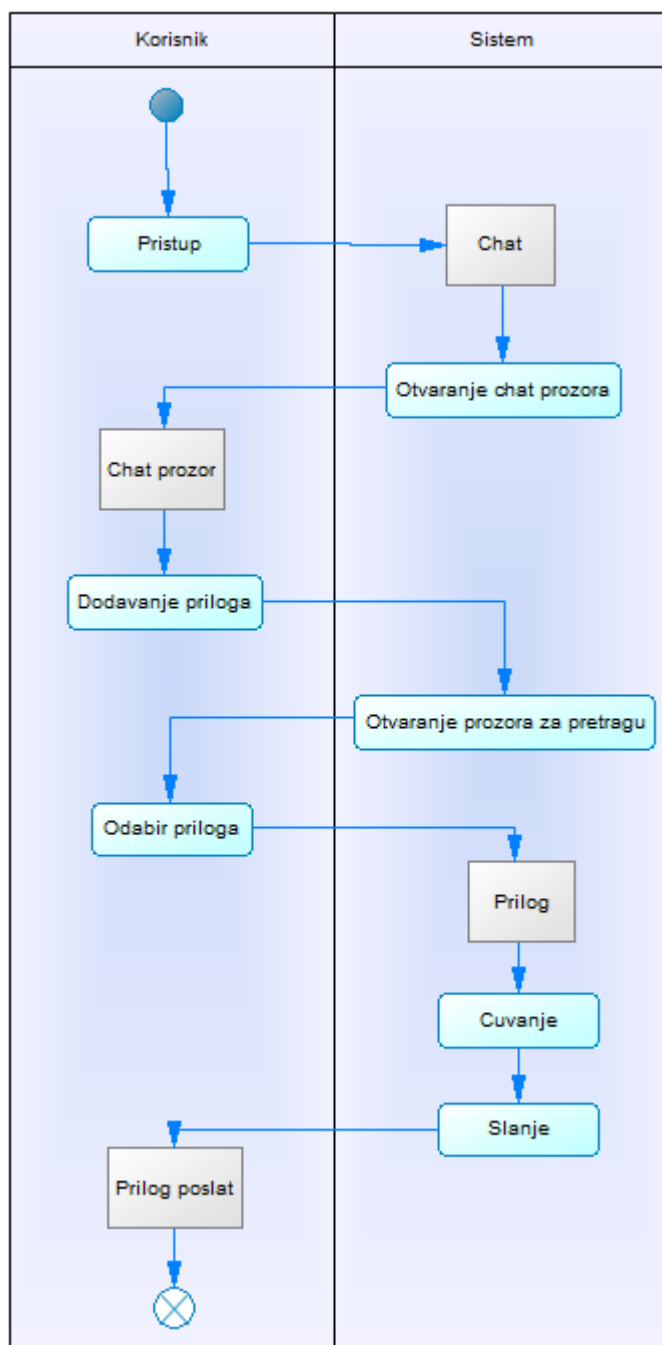
Slika 8. Dijagram aktivnosti za slanje javnih poruka

### 3.5.4 Dijagram aktivnosti za slanje priloga



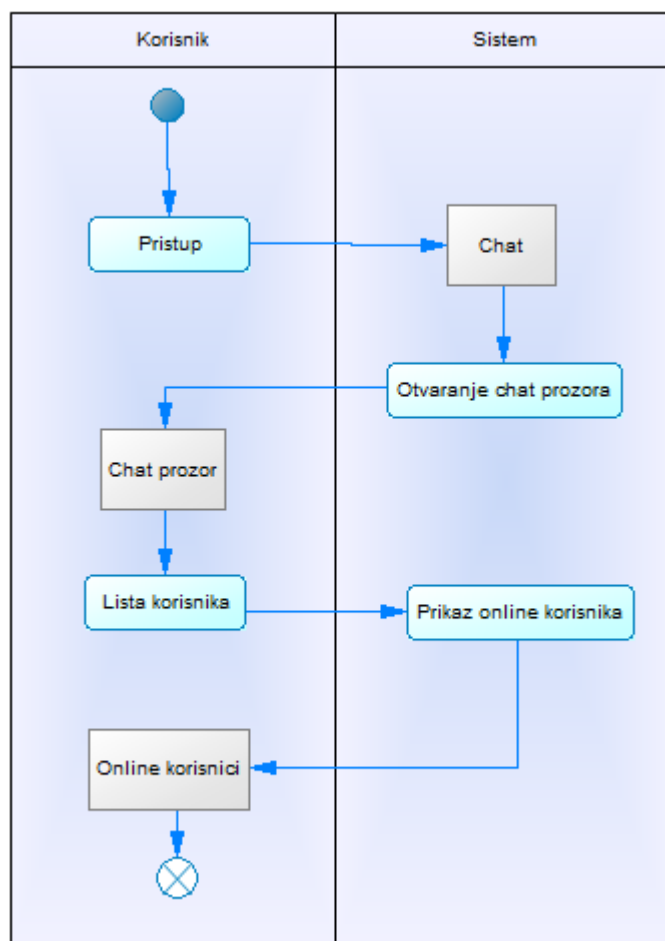
Slika 9. Dijagram aktivnosti za slanje priloga

### 3.5.5 Dijagram aktivnosti za slanje lokalnih priloga



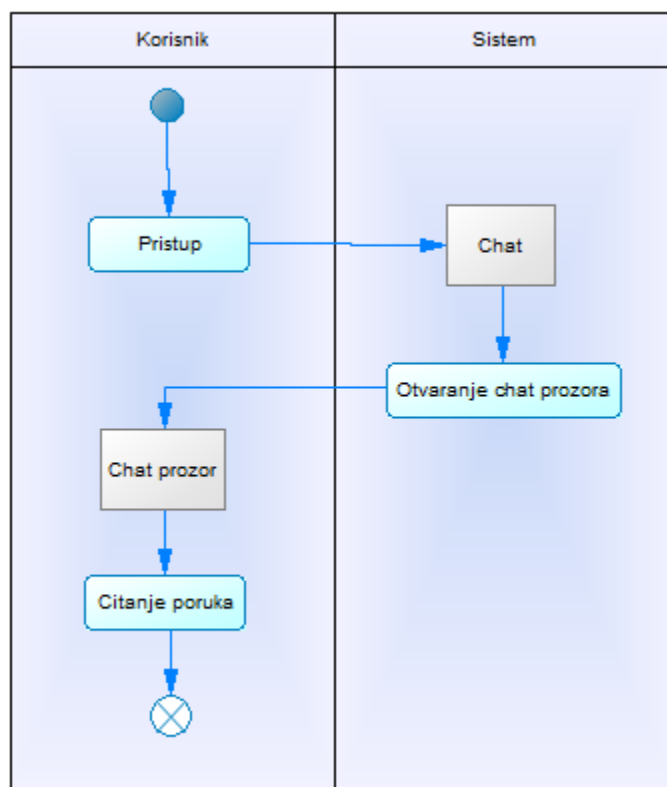
Slika 10. Dijagram aktivnosti za slanje lokalnih priloga

### 3.5.6 Dijagram aktivnosti za pregled korisnika



Slika 11. Dijagram aktivnosti za pregled korisnika

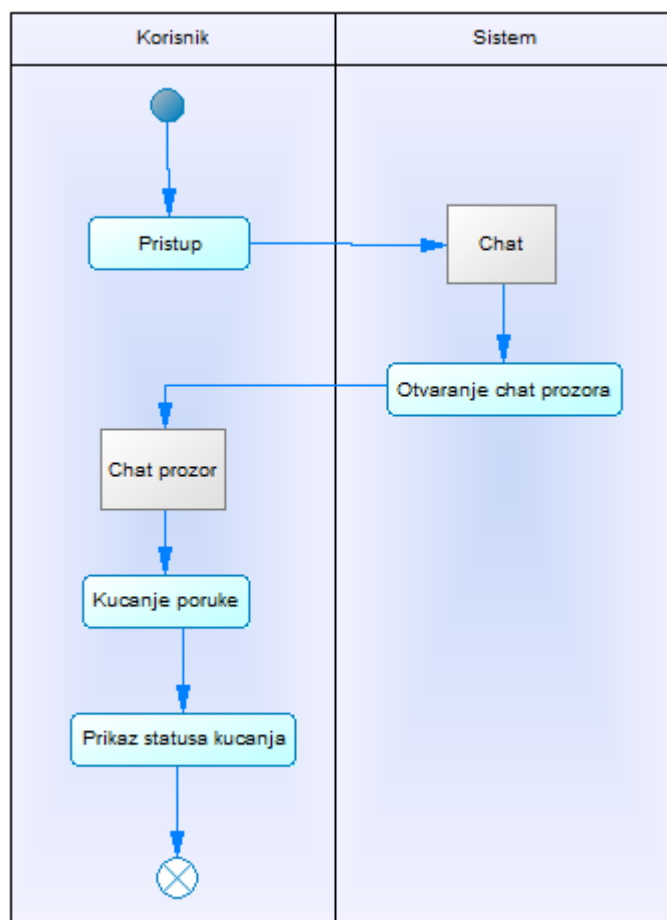
### 3.5.7 Dijagram aktivnosti za čitanje poruka



Slika 12. Dijagram aktivnosti za čitanje poruka



### 3.5.8 Dijagram aktivnosti za status pisanja



Slika 13. Dijagram aktivnosti za status pisanja

## 4. Modeli analize sistema

### 4.1 Dijagrami sekvenci

Sekvencijalni dijagram prikazuje interakciju između objekata uređenih u vremensku sekvencu. Najopštija primena sekvencijalnih dijagrama je predstavljanje detaljne interakcije objekata koji se pojavljuju u jednom slučaju upotrebe ili jednoj operaciji.

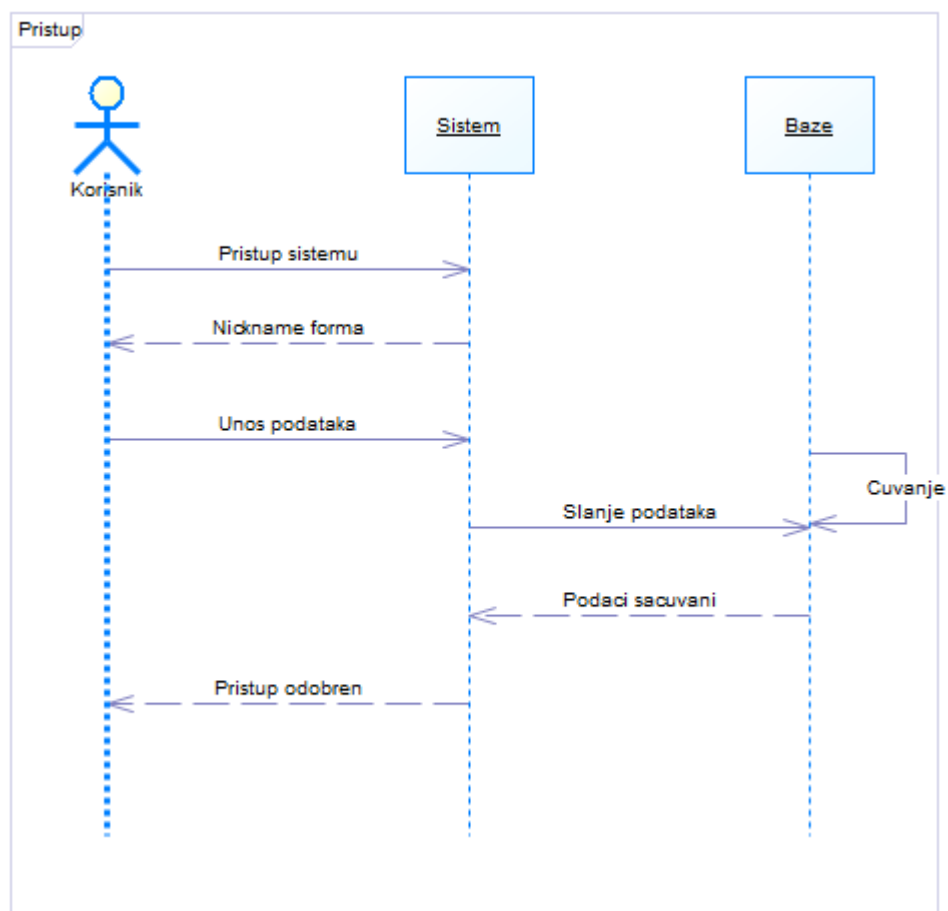
Prikazuje jedan scenario koji obuhvata izvestan broj objekata i poruka koje oni razmenjuju u okviru slučaja upotrebe. Korišćenjem sekvencijalnog dijagrama može se opisati koje interakcije se izvršavaju kada se pojedinačni slučaj upotrebe izvršava i po kom redosledu se te interakcije izvršavaju. Dijagram objekata specificira objekte koji sarađuju i veze između njih.

Dijagrami sekvence predstavljaju ponašanje slučaja upotrebe koristeći kao osnovu klase.

Dijagram sekvenci ima različite vrste poruke:

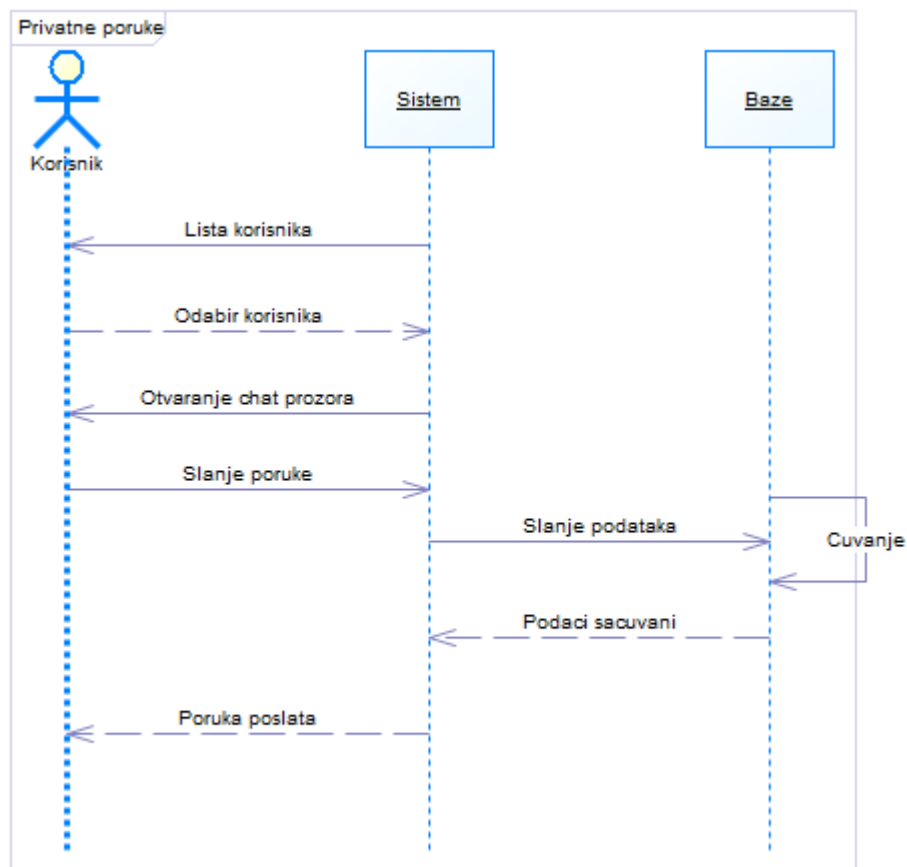
- Sinhrona poruka izaziva da pozvana operacija suspenduje izvršavanje dok joj se fokus kontrole ne vrati.
- Asinhrona poruka ne izaziva prekid izvršenja dok se čeka na povratak. Kada se pošalje sinhrona poruka, operacije u oba objekta mogu da se vrše istovremeno.
- Povratna poruka

#### 4.2.1 Dijagram sekvenci za pristup



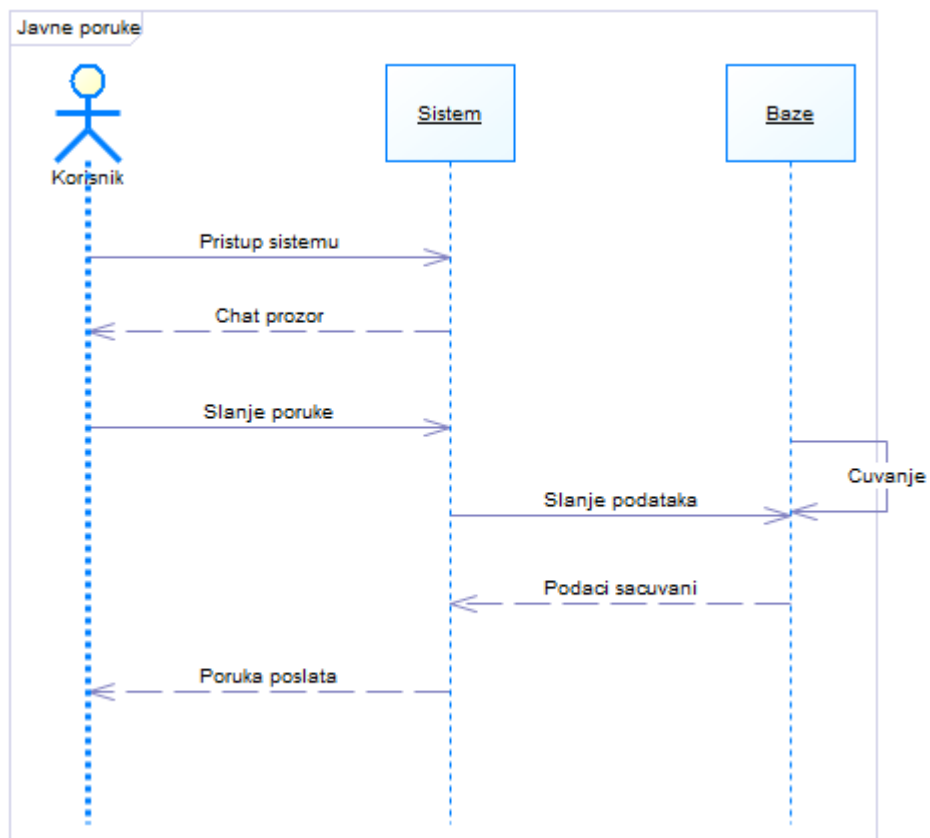
Slika 14. Dijagram sekvenci za pristup

#### 4.2.2 Dijagram sekvenci za slanje privatnih poruka



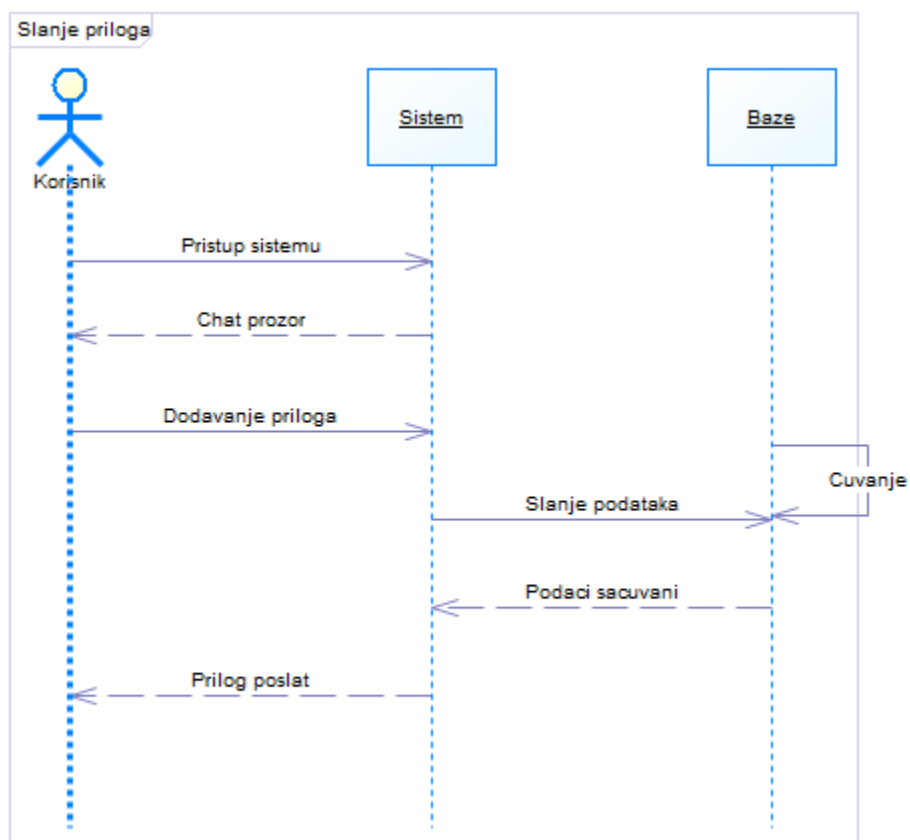
Slika 15. Dijagram sekvenci za slanje privatnih poruka

#### 4.2.3 Dijagram sekvenci za slanje javnih poruka



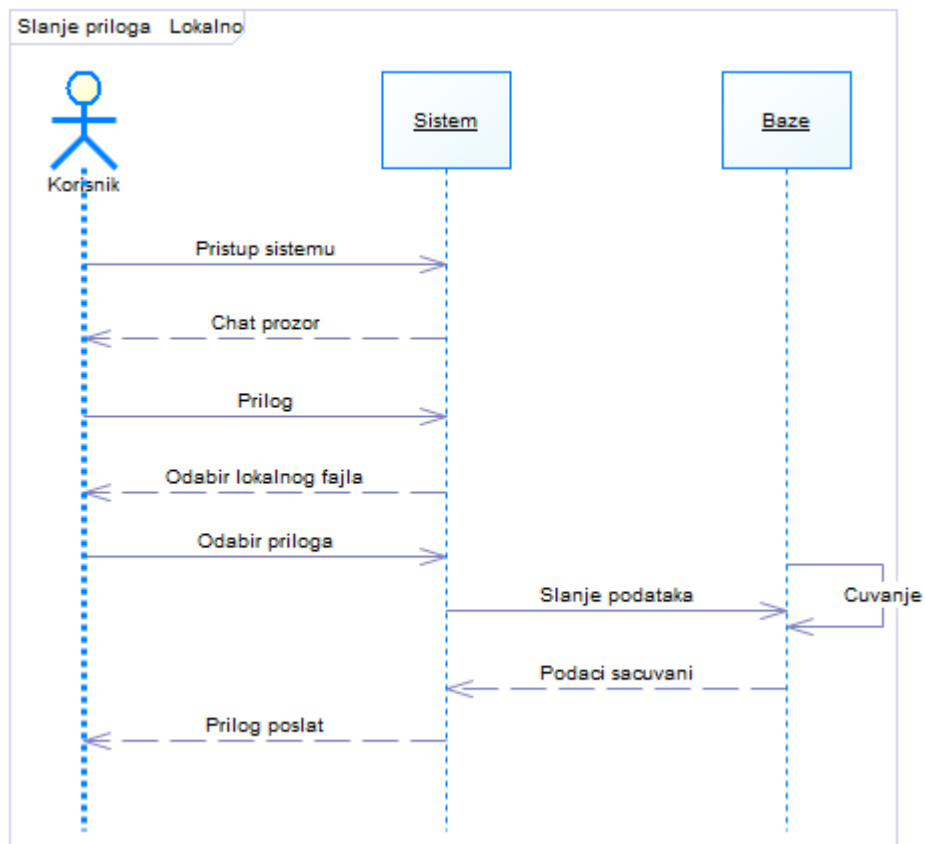
Slika 16. Dijagram sekvenci za slanje javnih poruka

#### 4.2.4 Dijagram sekvenci za slanje priloga



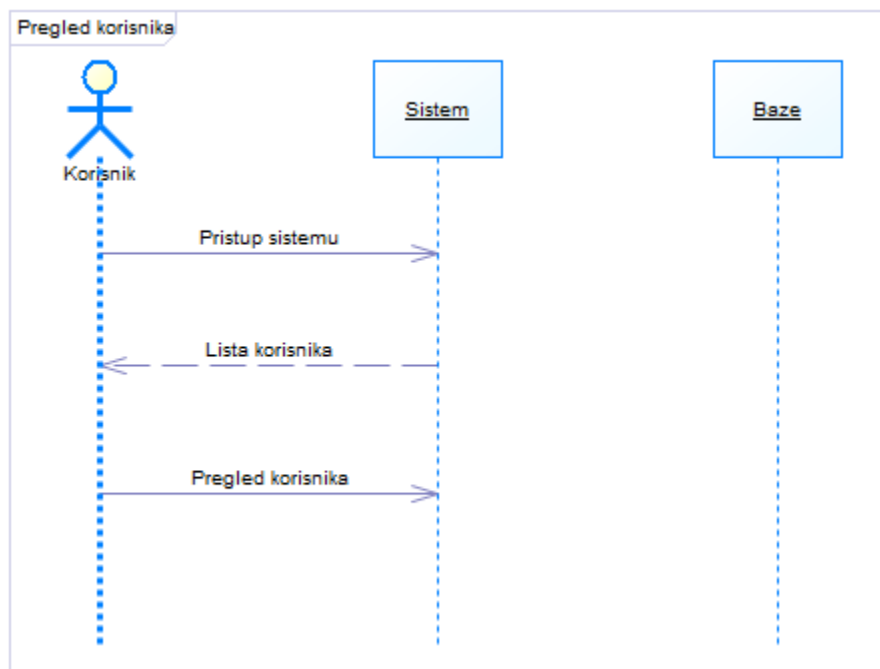
Slika 17. Dijagram sekvenci za slanje priloga

#### 4.2.5 Dijagram sekvenci za lokalno slanje priloga



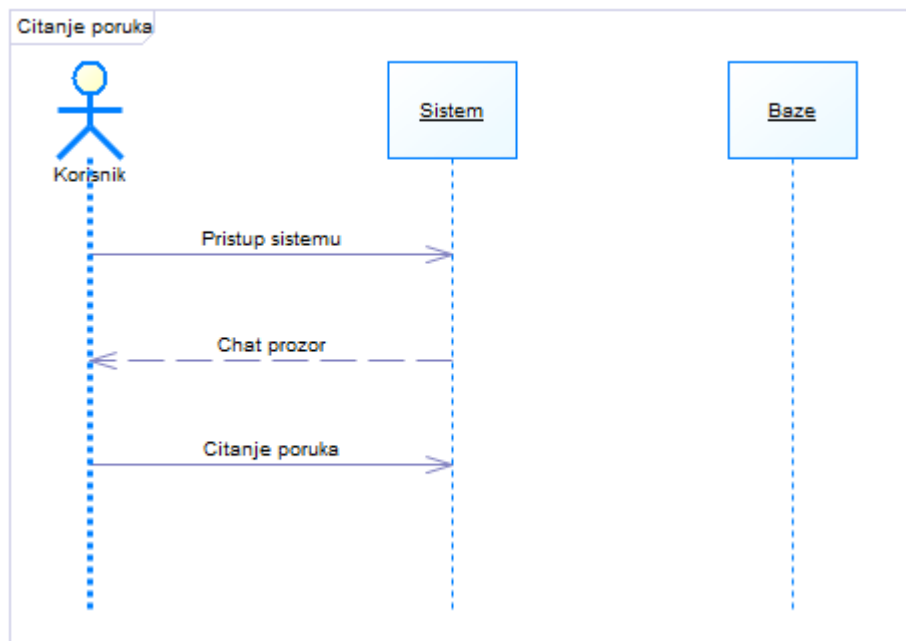
Slika 18. Dijagram sekvenci za lokalno slanje priloga

#### 4.2.5 Dijagram sekvenci za pregled korisnika



Slika 19. Dijagram sekvenci za pregled korisnika

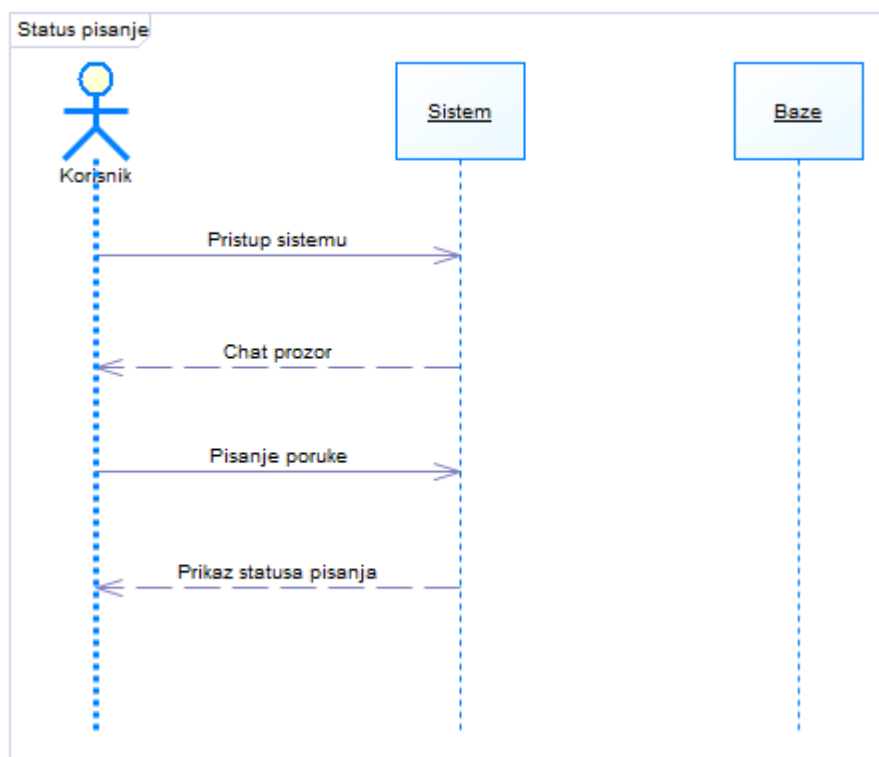
#### 4.2.6 Dijagram sekvenci za citanje poruka



Slika 20. Dijagram sekvenci za čitanje poruka



#### 4.2.7 Dijagram sekvenci za prikaz statusa pisanja



Slika 21. Dijagram sekvenci za prikaz statusa pisanja

## 4.3 Dijagrami komunikacije

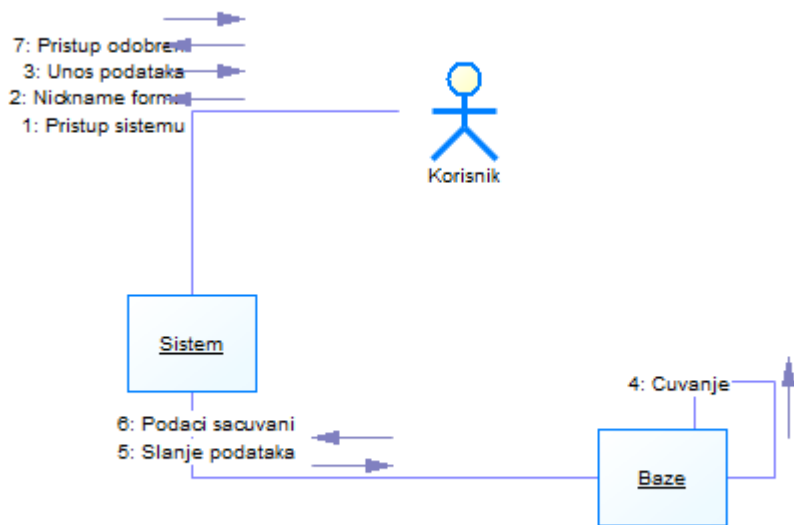
Komunikacioni dijagram opisuje sradnju i komunikaciju između objekata u sistemu. Saradnja se prikazuje međusobnim vezama između objekata.

Za razliku od dijagrama sekvenci, na dijagramu saradnje je naglašena veza između objekata. Moguće je automatski prevesti jedan oblik u drugi – dijagram sekvenci u dijagram saradnje i obrnuto.

Dijagrami sekvenci obezbeđuju način da se scenario posmatra u vremenski određenom redosledu. Klijenti mogu lako da čitaju i razumeju takvu vrstu dijagrama, stoga su korisni u raznim fazama analize.

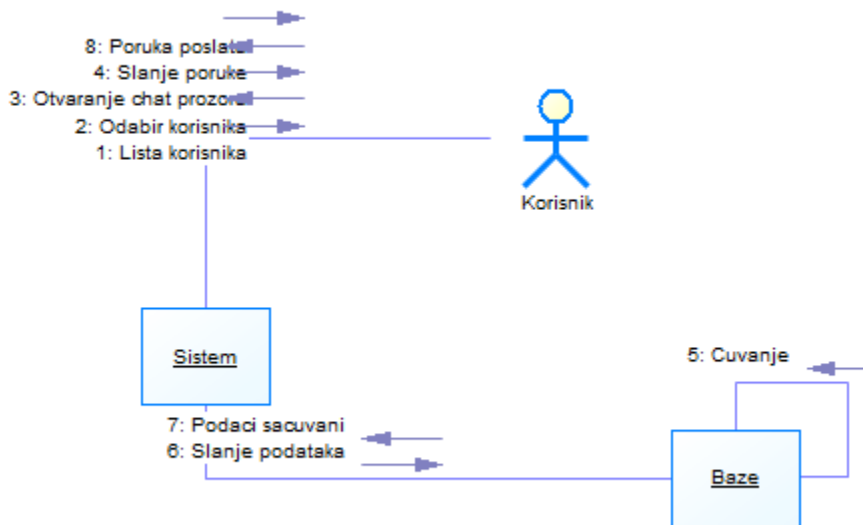
Dijagrami saradnje pružaju uvid u širu sliku o scenariju pošto ističu veze između objekata. Dijagrami se više koriste u projektnoj fazi razvoja, kada se projektuju realizacije relacija.

### 4.3.1 Dijagram komunikacije za pristup



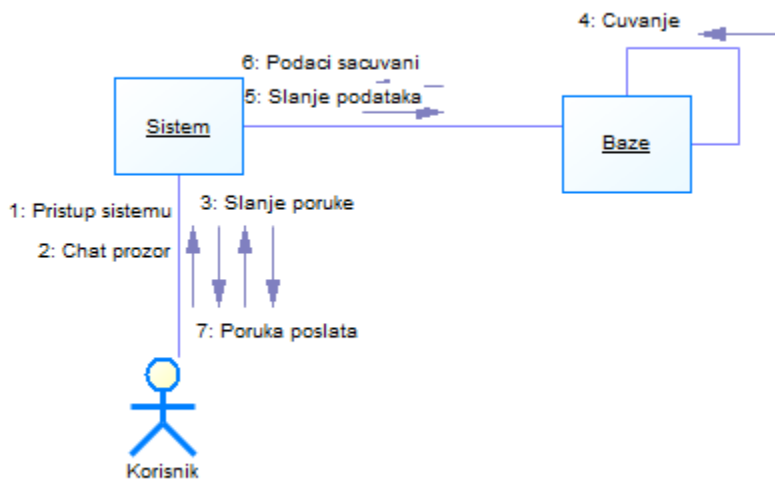
Slika 22. Komunikacioni dijagram za pristup

#### 4.3.2 Dijagram komunikacije za slanje privatnih poruka



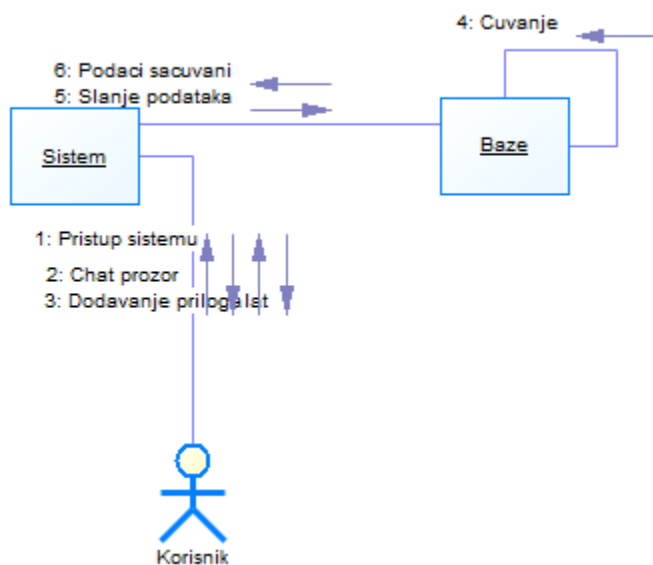
Slika 23. Komunikacioni dijagram za slanje privatnih poruka

#### 4.3.3 Dijagram komunikacije za slanje javnih poruka



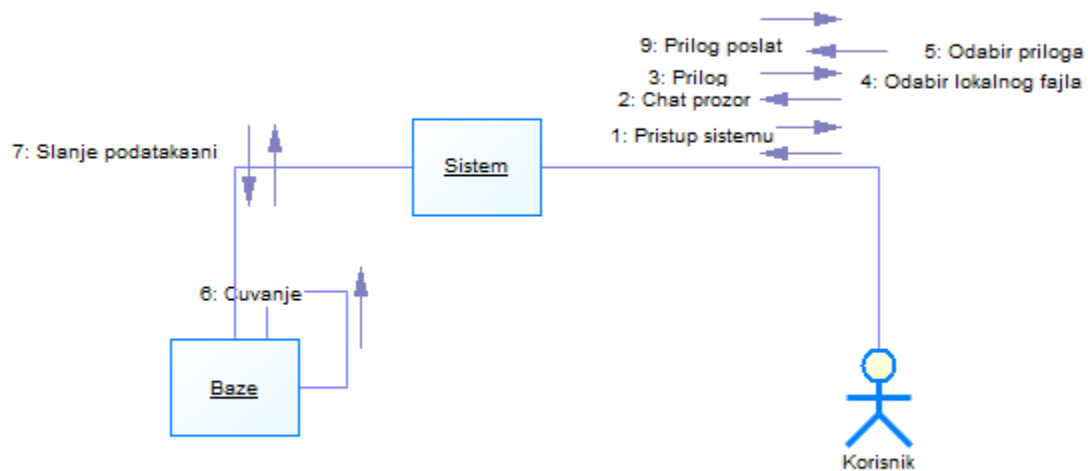
Slika 24. Komunikacioni dijagram za slanje javnih poruka

#### 4.3.4 Dijagram komunikacije za slanje priloga



Slika 25. Komunikacioni dijagram za slanje priloga

#### 4.3.6 Dijagram komunikacije za slanje lokalnih priloga



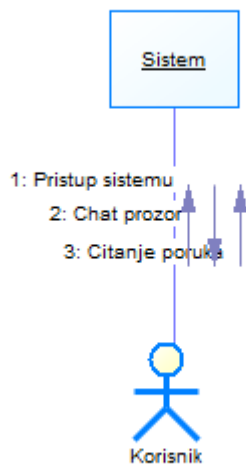
Slika 26. Komunikacioni dijagram za lokalno slanje priloga

#### 4.3.7 Dijagram komunikacije za pregled korisnika



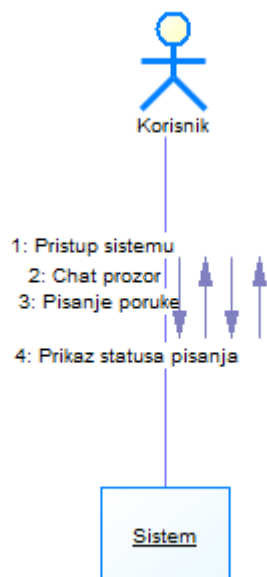
Slika 27. Komunikacioni dijagram za pregled korisnika

#### 4.3.8 Dijagram komunikacije za čitanje poruka



Slika 28. Komunikacioni dijagram za čitanje poruka

#### 4.3.9 Dijagram komunikacije za status pisanja

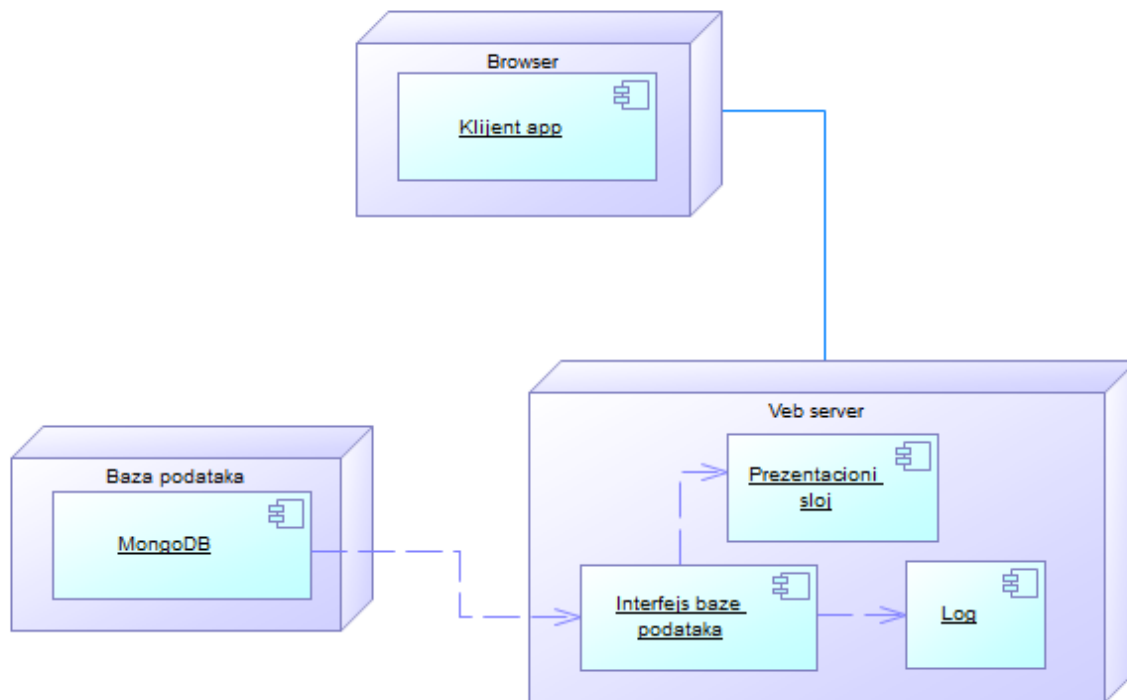


Slika 29. Komunikacioni dijagram za prikaz statusa pisanja

## 5. Implementacioni dijagrami

### 5.1 Dijagram realizacije

Dijagram realizacije (Deployment diagram) predstavlja hardverski način realizacije rešenja, uređaje i komunikaciju. Definiše mrežnu konfiguraciju, alokaciju (raspored) čvorova i mehanizam distribucije.



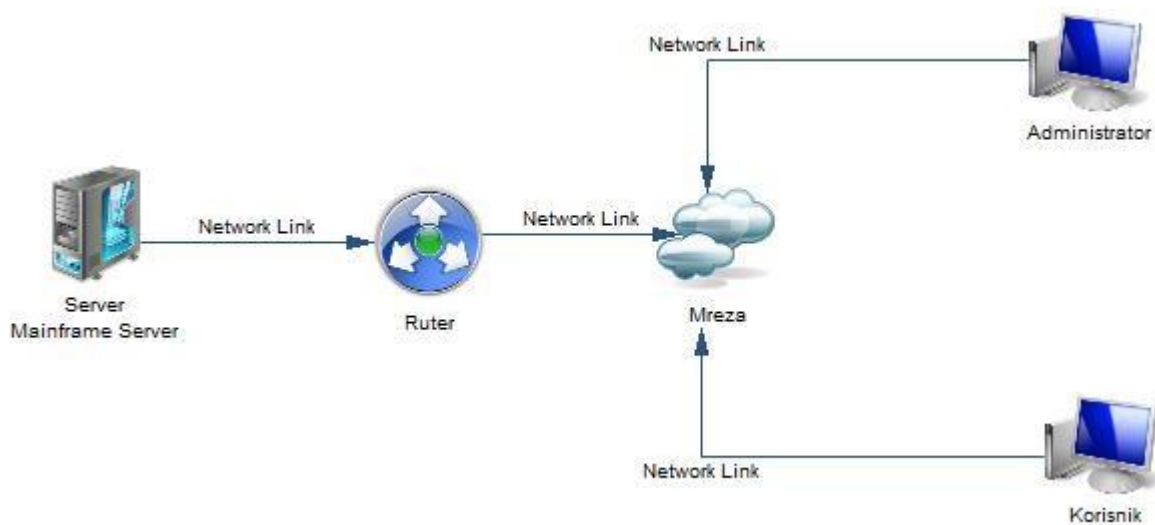
Slika 30. Dijagram realizacije

## 5.2 Infrastrukturni dijagram

Infrastrukturni dijagram predstavlja unutrašnju strukturu elementa. Uključuje elemente koji su neophodni za funkcionisanje jednog veb sistema.

Administratori i korisnici sistema pristupaju sistemu preko interneta sa bilo kog računara, samim tim moraju imati stabilnu internet konekciju radi najboljeg odaziva sistema. Klijenti mogu da pristupe samo onom delu sistema koji je njima namenjen.

Da bi postojao pristup veb sistemu, potrebno je da bude postavljen server preko koga će klijenti moći da mu pristupe.

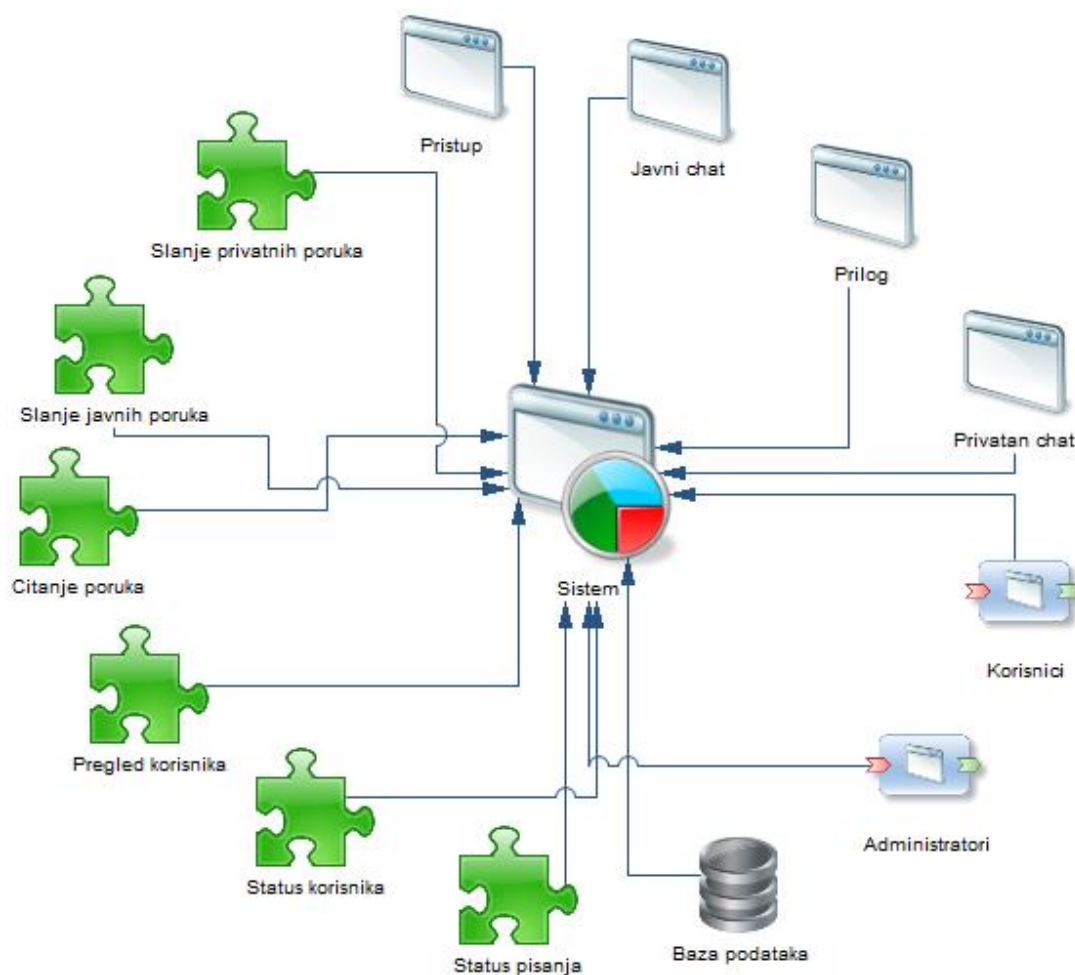


Slika 31. Infrastrukturni dijagram



### 5.3 Dijagram arhitekture

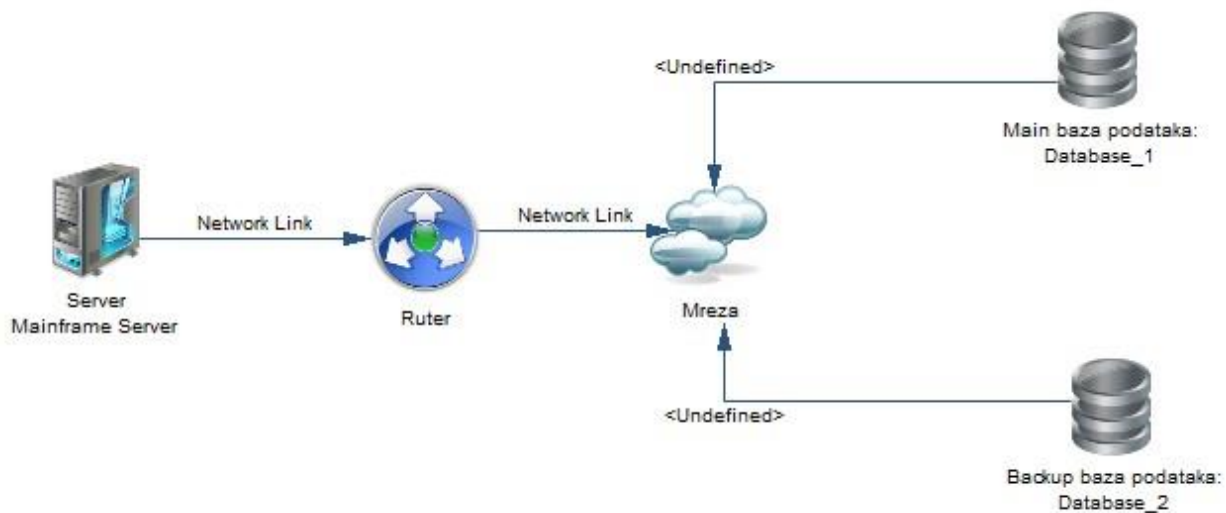
Dijagram arhitekture predstavlja arhitekturu jednog veb sistema. Pored osnovnih delova za funkcionisanje jednog veb sistema, tu je baza podataka koja predstavlja MongoDB bazu podataka sa osnovnim mogućnostima manipulacije nad podacima u bazi podataka (čitanje, upisivanje, ažuriranje i brisanje).



Slika 32. Dijagram arhitekture

## 5.4 Dijagram mreže

Dijagram mreže predstavlja osnovnu arhitekturu baze podataka sistema. Za svaki deo baze podataka postoji backup gde se čuvaju podaci u slučaju gubitaka prvobitnih podataka glavne baze podataka sistema.



Slika 33. Dijagram mreže

## 6. Plan testiranja

U okviru plana testiranja sistema postoji nekoliko ključnih slojeva koje treba testirati na osnovu kojih se može garantovati kvalitet funkcionalnosti i integritet podataka.

Prvo deo testiranja biće testiranje funkcionalnosti samog sistema i interakcije sistema sa korisnikom. Ovim testovima biće pokrivene sve ključne funkcionalnosti interfejsa sistema i interakcije sa korisnikom.

Drugi deo testiranja jesu upiti ka bazi podataka. U ovom delu biće korišćena test baza podataka. Baza će biti identična bazi namenjenoj za produkciju, ali će biti popunjena potrebnom količinom podataka kako bi testiranje u što većoj meri odgovaralo realnom redu sistema sa bazom podataka. Prebacivanje sa jedne baze na drugu se vrši jednostavno, promenom naziva baze u okviru konfiguracionog fajla. Biće testirane osnovne CRUD operacije (engl. Create, Read, Update, Delete) jer one predstavljaju glavne operacije samog sistema, i bez njihovog funkcionisanja sistem nije u stanju da dostigne fazu testiranja. Testiranje uključuje proveru rada i funkcionalnosti veb servisa.

Nakon izvršenog testiranja, projekat će biti pušten u beta fazu i ima za cilj testiranje krajnjeg proizvoda iz ugla krajnjeg korisnika. U ovoj fazi se ne testiraju tehnički aspekti funkcionalnosti, već korisničko iskustvo, kao i eventualne greške koje su nastale kroz razvoj sistema, a nisu otkrivene u ranijim fazama testiranja.

## 7. Zaključak

Ovaj dokument predstavlja idejni koncept projekta MetChat sistema i ne predstavlja završnu dokumentaciju projekta. Kroz dalji razvoj projekta, delovi ovog koncepta će se menjati, gde će biti dodeljene nove funkcionalnosti, proces implementacije, testiranje i uvid u prve prototipe sistema.