XChange

Arhitekturni projekat

Verzija 1.0

Pregled izmena

|  |  |  |  |
| --- | --- | --- | --- |
| **Datum** | **Verzija** | **Opis** | **Autor** |
| 27.04.2023 | 1.0 | Inicijalna verzija | Mihajlo Bencun, Uroš Miladinović, Luka Stojković |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Sadržaj

[1. Cilj dokumenta 3](#_Toc133507896)

[2. Opseg dokumenta 3](#_Toc133507897)

[3. Reference 3](#_Toc133507898)

[4. Predstavljanje arhitekture 3](#_Toc133507899)

[5. Ciljevi i ograničenja arhitekture 3](#_Toc133507900)

[6. Pogled na slučajeve korišćenja 3](#_Toc133507901)

[6.1 Dijagrami slučajeva korišćenja 4](#_Toc133507902)

[6.2 Kratak opis slučajeva korišćenja 6](#_Toc133507903)

[6.2.1 Pregled oglasa 6](#_Toc133507904)

[6.2.2 Pregled korisničkih profila 6](#_Toc133507905)

[6.2.3 Filtriranje oglasa po kategorijama 6](#_Toc133507906)

[6.2.4 Filtriranje oglasa po „Pretrazi“ 6](#_Toc133507907)

[6.2.5 Prijavljivanje 6](#_Toc133507908)

[6.2.6 Izmena svog profila 6](#_Toc133507909)

[6.2.7 Postavljanje oglasa 6](#_Toc133507910)

[6.2.8 Brisanje oglasa 6](#_Toc133507911)

[6.2.9 Modifikacija podataka o oglasu 6](#_Toc133507912)

[6.2.10 Interakcija sa korisnicima 6](#_Toc133507913)

[6.2.11 Ocenjivanje korisnika 6](#_Toc133507914)

[6.2.12 Pretplaćivanje na oglas 6](#_Toc133507915)

[6.2.13 Prijava profila 7](#_Toc133507916)

[6.2.14 Analiza prijave 7](#_Toc133507917)

[6.2.15 Privremeno suspendovanje naloga 7](#_Toc133507918)

[6.2.16 Trajno brisanje naloga 7](#_Toc133507919)

[6.2.17 Brisanje oglasa koji krši pravila 7](#_Toc133507920)

[6.2.18 Interakcija korisničke podrške i korisnika 7](#_Toc133507921)

[6.2.19 Održavanje aplikacije 7](#_Toc133507922)

[7. Pogled na logičku arhitekturu sistema 7](#_Toc133507923)

[7.1 Pregled arhitekture – organizacija paketa i podsistema u slojeve 8](#_Toc133507924)

[7.1.1 Korisnički interfejs 8](#_Toc133507925)

[7.1.2 Aplikaciona logika 8](#_Toc133507926)

[7.1.3 Pristup podacima 8](#_Toc133507927)

[7.1.4 HTML 8](#_Toc133507928)

[7.1.5 NestJS 8](#_Toc133507929)

[7.1.6 MongoDB 9](#_Toc133507930)

[8. Pogled na procese 9](#_Toc133507931)

[8.1 Procesi 9](#_Toc133507932)

[8.1.1 Web čitač 9](#_Toc133507933)

[8.1.2 Statički server 9](#_Toc133507934)

[8.1.3 API server 9](#_Toc133507935)

[8.1.4 Angular 10](#_Toc133507936)

[8.1.5 MongoDB Server 10](#_Toc133507937)

[9. Pogled na raspoređivanje sistema 10](#_Toc133507938)

[9.1 Klijent 10](#_Toc133507939)

[9.2 Web server 10](#_Toc133507940)

[9.3 DBMS server 10](#_Toc133507941)

[10. Pogled na implementaciju sistema 10](#_Toc133507942)

[10.1 Model domena 10](#_Toc133507943)

[10.2 Šema baze podataka 11](#_Toc133507944)

[10.3 Komponente sistema 12](#_Toc133507945)

[10.3.1 Komponente korisničkog interfejsa 12](#_Toc133507946)

[10.3.2 Komponente aplikacione logike 12](#_Toc133507947)

[10.3.3 Komponente za pristup podacima 12](#_Toc133507948)

[11. Performanse 12](#_Toc133507949)

[12. Kvalitet 12](#_Toc133507950)

Sadržaj

Arhitekturni projekat

# Cilj dokumenta

Cilj ovog dokumenta je detaljni opis arhitekture XChange aplikacije.

# Opseg dokumenta

Dokument se odnosi na XChange aplikaciju koja će biti razvijena od strane Posade M-84. Namena aplikacije jeste omogućavanje online trgovine bez posredstva novca.

# Reference

Spisak korišćene literature:

1. XChange – Predlog projekta, Posada M-84, V1.0, 2023.
2. XChange – Planirani raspored aktivnosti na projektu, V1.0, 2023, Posada M-84.
3. XChange – Plan realizacije projekta, V1.0, 2023, Posada M-84.
4. XChange – Vizija sistema, V1.0, 2023, Posada M-84.
5. XChange – D04\_Spec\_Zahteva, Posada M-84.

# Predstavljanje arhitekture

Arhitektura sistema u dokumentu je prikazana kao serija pogleda na sistem: pogled na slučajeve korišćenja, pogled na logičku arhitekturu sistema, pogled na procese, pogled na razmeštaj komponenti sistema i pogled na implementaciju. Ovi pogledi su predstavljeni odgovarajućim UML dijagramima.

# Ciljevi i ograničenja arhitekture

Ključni zahtevi i sistemska ograničenja koja imaju značajan uticaj na izbor arhitekture i projektovanje sistema su:

1. XChange aplikacija će biti implementirana kao Web aplikacija zasnovana na MEAN stack-u. [4].
2. Klijentski deo XChange-a će biti optimizovan za sledeće Web čitače: Internet Explorer 6.0 i noviji, Opera 8.0 i noviji, kao i Firefox (Mozilla) [4].
3. Svi zahtevi u pogledu performansi dati u [5] moraju biti uzeti u obzir pri izboru arhitekture i razvoju sistema.

# Pogled na slučajeve korišćenja

U ovom odeljku je dat pogled na slučajeve korišćenja definisane u specifikaciji zahteva [5].

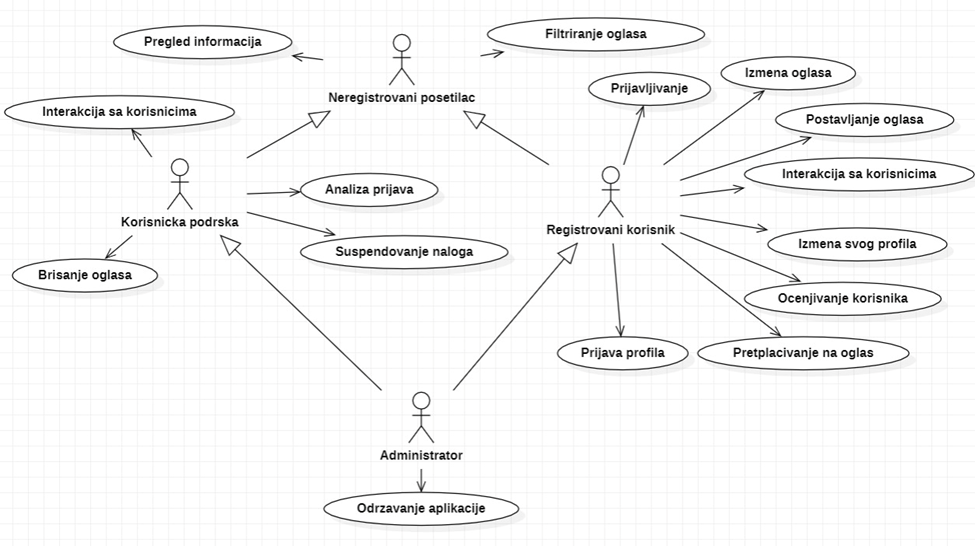
Slučajevi korišćenja XChange aplikacije su:

* *Pregled informacija*
  + *Pregled tuđih profila*
  + *Pregled oglasa*
* *Filtriranje oglasa*
  + *Filtrirajne po kategorijama*
  + *Filtriranje pomoću „Search“*
* *Prijavljivanje*
* *Izmena svog profila*
* *Postavljanje oglasa*
* *Brisanje oglasa*
* *Izmena oglasa*
  + *Brisanje oglasa*
  + *Modifikacija podataka o oglasu*
* *Interakcija sa korisnicima*
* *Ocenjivanje korisnika*
* *Pretplaćivanje na oglas*
* *Prijava profila*
* *Analiza prijave*
* *Suspendovanje naloga*
  + *Privremeno suspendovanje*
  + *Trajno brisanje naloga*
* *Brisanje oglasa koji krši pravila*
* *Interakcija korisničke podrške i korisnika*
* *Održavanje aplikacije*

Ove slučajevi korišćenja mogu da iniciraju posetilac, registrovani korisnik, korisnička usluga ili administrator.

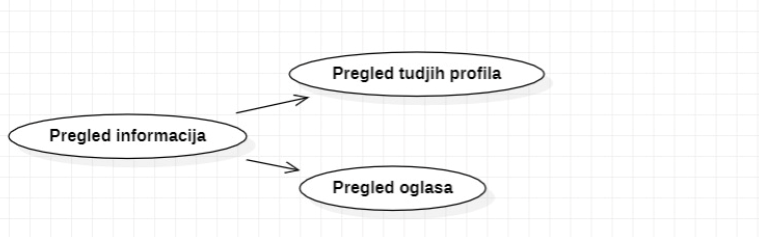
## Dijagrami slučajeva korišćenja

Osnovni UML dijagram koji prikazuje korisnike i slučajeve korišćenja XChange aplikacije prikazan je na sledećoj slici:



Slučajevi korišćenja *pregled informacija* i *izmena oglasa* i *suspendovanje naloga* i *filtriranje oglasa* obuhvataju složenije radnje koje se mogu razložiti dalje razložiti na pojedinačne slučajeve korišćenja.

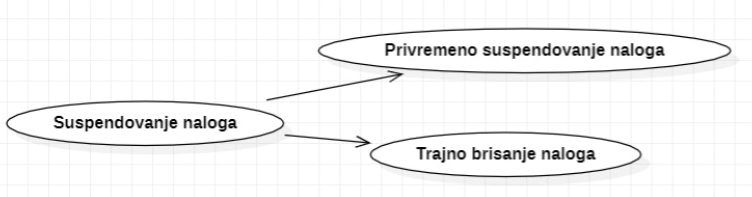
Detaljni UML dijagram za slučaj korišćenja *pregled informacija* je prikazan na sledećoj slici:



Detaljni UML dijagram za slučaj korišćenja *izmena oglasa* je prikazan na sledećoj slici:



Detaljni UML dijagram za slučaj korišćenja *suspendovanje naloga* je prikazan na sledećoj slici:



Detaljni UML dijagram za slučaj korišćenja *filtriranje oglasa* je prikazan na sledećoj slici:



## Kratak opis slučajeva korišćenja

### Pregled oglasa

Kratak opis**:** Prikaz stranice sa oglasima ili stranice sa konkretnim oglasom.

Akteri**:** Neregistrovani korisnik, registrovani korisnik, korisnička podrška, administrator.

### Pregled korisničkih profila

Kratak opis: Prikaz stranice sa informacijama nekog registrovanog korisnika..

Akteri koji iniciraju slučaj korišćenja: Neregistrovani korisnik, registrovani korisnik, korisnička podrška, administrator.

### Filtriranje oglasa po kategorijama

Kratak opis: Filtriranje oglasa po željenim kategorijama od ponuđenih.

Akteri koji iniciraju slučaj korišćenja: Neregistrovani korisnik, registrovani korisnik, korisnička podrška, administrator.

### Filtriranje oglasa po „Pretrazi“

Kratak opis: Filtriranje oglasima prema imenu ili drugim informacijama koje zada korisnik.

Akteri koji iniciraju slučaj korišćenja: Posetilac portala, Član laboratorije, Šef laboratorije, Vođa projekta, Administrator.

### Prijavljivanje

Kratak opis: Prijavljivanje korisnika na sajt u cilju pristupa specifičnim funkcijama koje zahtevaju autorizaciju.

Akteri koji iniciraju slučaj korišćenja: Registrovani korisnik, korisnička podrška, administrator.

### Izmena svog profila

Kratak opis: Ažuriranje podataka o sebi od strane prijavljenog korisnika.

Akteri koji iniciraju slučaj korišćenja: Registrovani korisnik, korisnička podrška, administrator.

### Postavljanje oglasa

Kratak opis: Korisnik kreira novi oglas i navodi detalje o njemu.

Akteri koji iniciraju slučaj korišćenja: Registrovani korisnik, administrator.

### Brisanje oglasa

Kratak opis: Korisnik koji je kreirao oglas može da ga ukloni, ili oglas može ukloniti korisnička podrška ukoliko krši pravila korišćenja.

Akteri koji iniciraju slučaj korišćenja: Registrovani korisnik, korisnička podrška, administrator.

### Modifikacija podataka o oglasu

Kratak opis: Korisnik koji je kreirao oglas ima opciju izmena podataka o tom oglasu.

Akteri koji iniciraju slučaj korišćenja: Registrovani korisnik, administrator.

### Interakcija sa korisnicima

Kratak opis: Korisnici mogu međusobno da komuniciraju putem poruka.

Akteri koji iniciraju slučaj korišćenja: Registrovani korisnik, korisnička podrška, administrator.

### Ocenjivanje korisnika

Kratak opis: Moguće je ostaviti numeričku ocenu kao i kratak opis interakcije sa drugim korisnikom koji će biti javno vidljivi.

Akteri koji iniciraju slučaj korišćenja: Registrovani korisnik, administrator.

### Pretplaćivanje na oglas

Kratak opis: Korisnici se mogu pretplatiti na oglas kako bi dobili obaveštenja putem e-maila za sve promene vezane za taj oglas.

Akteri koji iniciraju slučaj korišćenja: Registrovani korisnik, administrator.

### Prijava profila

Kratak opis: Korisnici mogu da prijave oglas i profil za koji smatraju da krši pravila korišćenja.

Akteri koji iniciraju slučaj korišćenja: Registrovani korisnik, administrator.

### Analiza prijave

Kratak opis: Korisnička podrška analizira prijave i utvrđuje da li je došlo do prekršaja.

Akteri koji iniciraju slučaj korišćenja: Korisnička služba, administrator.

### Privremeno suspendovanje naloga

Kratak opis: U skladu sa prekršajem, nalog može biti privremeno suspendovan.

Akteri koji iniciraju slučaj korišćenja: Korisnička podrška, administrator.

### Trajno brisanje naloga

Kratak opis: U skladu sa prekršajem nalog može biti trajno obrisan.

Akteri koji iniciraju slučaj korišćenja: Korisnička služba, administrator.

### Brisanje oglasa koji krši pravila

Kratak opis: U skladu sa prekršajem, može biti obrisan oglas.

Akteri koji iniciraju slučaj korišćenja: Korisnička služba, administrator. Brisanje oglasa koji krši pravila

### Interakcija korisničke podrške i korisnika

Kratak opis: Korisnička služba može uspostaviti komunikaciju sa korisnikom čiji je profil prijavljen.

Akteri koji iniciraju slučaj korišćenja: Korisnička služba, registrovani korisnik, administrator.

### Održavanje aplikacije

Kratak opis: Administratori održavaju funkcionalnost aplikacije, testiraju nove funkcionalnosti.

Akteri koji iniciraju slučaj korišćenja: Administrator.

# Pogled na logičku arhitekturu sistema

U ovom odeljku je dat pregled logičke arhitekture sistema. Ovaj pogled sadrži opis najznačajnijih klasa, njihove organizacije u pakete i podsisteme, i organizacija podsistema u slojeve. U cilju opisivanja dinamičkih aspekata arhitekture, ovaj odeljak može da uključi opise realizacije najznačajnijih slučajeva korišćenja. Da bi se ilustrovala veza između arhitekturno značajnih klasa, podsistema, paketa ili slojeva moguće je uključiti i odgovarajuće dijagrame klasa.

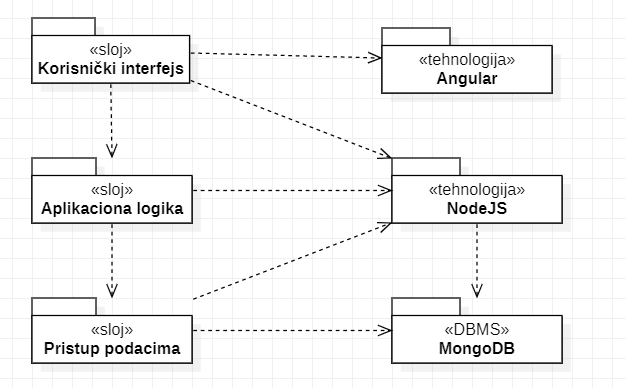
Logički pogled na XChange aplikaciju obuhvata 3 glavna paketa: Korisnički interfejs, Aplikaciona logika, Pristup podacima.

Paket *Korisnički interfejs* sadrži Web stranice, Angular kod i multimedijalni sadržaj koji realizuju grafički dizajn i forme preko kojih korisnici sistema komuniciraju sa sistemom.

Paket *Aplikaciona logika* predstavlja srednji sloj sistema koji sadrži NodeJS proces (u NestJS framework-u) zadužen za realizaciju funkcionalnosti specifičnih za domen sistema koji se razvija.

Paket *Pristup podacima* takođe sadrži NodeJS proces koji predstavlja interfejs za pristup, dodavanje i ažuriranje podataka koji se čuvaju u bazi podataka.

## Pregled arhitekture – organizacija paketa i podsistema u slojeve



### Korisnički interfejs

sloj

Ovaj sloj realizuje korisnički interfejs portala. U njemu su sadržani Angular aplikacija i multimedijalni sadržaji preko kojih korisnici komuniciraju sa sistemom.

Sloj korisničkog interfejsa zavisi od sloja aplikacione logike, kao i paketa Angular.

### Aplikaciona logika

sloj

Sloj aplikacione logike je srednji sloj u troslojnoj arhitekturi aplikacije. Sadrži NodeJS server u okviru NestJS framework-a koji realizuje funkcionalnost karakterističnu za domen primene aplikacije i uspostavljaju vezu između korisničkog interfejsa i sloja za pristup podacima.

Ovaj sloj zavisi od sloja za pristup podacima i NodeJS paketa.

### Pristup podacima

sloj

Sloj za pristup podacima se nalazi na dnu troslojne arhitekture i sadrži NodeJS kod zadužen za pribavljanje, dodavanje i ažuriranje podataka koji se čuvaju u MongoDB bazi podataka.

Ovaj sloj ne zavisi od drugih slojeva, ali je zavisan od paketa NodeJS i MongoDB baze podataka.

### HTML

tehnologija

Tehnologija HTML-a definiše gradivne elemente stranica koje se prikazuju u Web čitaču i koje omogućavaju prikaz formatiranih informacija i realizaciju formi za unos i ažuriranje podataka.

### NestJS

tehnologija

Tehnologija NestJS framework-a za Javascript obezbeđuje mehanizam za pisanje i izvršavanje skripti na strani servera. Ove skripte mogu da generišu HTML kod koji realizuje korisnički interfejs i pristupaju bazi podataka u cilju pribavljanja, unosa i ažuriranja podataka.

### MongoDB

DBMS

MongoDB predstavlja sistem za upravljanje NoSQL bazama podataka koji će se koristiti za realizaciju aplikacije.

# Pogled na procese

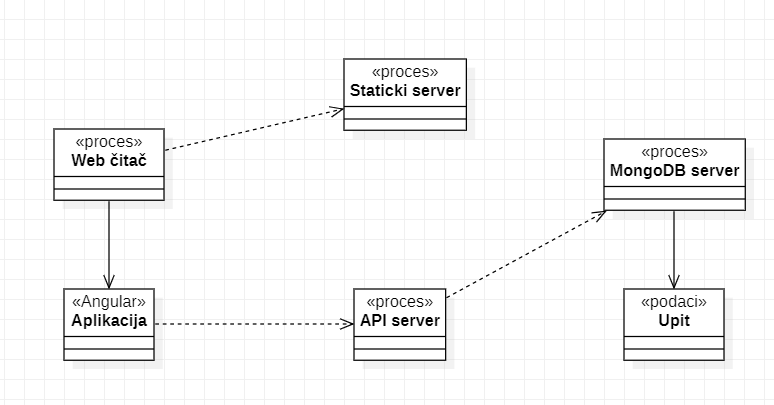
U ovom odeljku je sadržan pogled na procesnu arhitekturu sistema. Ovaj opis treba da sadrži specifikaciju različitih zadataka (procesa i niti) uključenih u rad sistema. Takođe je potrebno dati dijagrame koji pokazuju njihovu interakciju i konfiguraciju. Dodela objekata i klasa na određene zadatke takođe spada u opis procesne arhitekture.

Web aplikacije zasnovane na MEAN stack-u funkcionišu tako što se klijentu dostave statički podaci neophodni za učitavanje Angular aplikacije na pregledaču. Potom se API pozivi rutiraju na NodeJS server koji

Ilustracije radi u nastavku je dat opis procesa uključenih u izvršenje XChange kao Web aplikacije.

## Procesi

Na sledećem UML dijagramu klasa prikazani su procesi koji učestvuju u izvršenju XChange aplikacije. Dijagram je opšteg tipa i može se primeniti na bilo koju Web aplikaciju zasnovanu na NestJS-u i MongoDB bazi podataka.



### Web čitač

Web čitač će sa statičkog servera dobiti kompletnu Angular aplikaciju koja će se izvršavati na klijentskoj strani.

### Statički server

Server koji sadrži Angular aplikaciju i servira statičke podatke.

### API server

Node server sa kojim komunicira Angular aplikacija za pristup bazi podataka i dodatne backend funkcionalnosti.

### Angular

Angular aplikacija predstavlja kompletan front-end i logiku aplikacije koja će se izvršavati na klijentskoj strani.

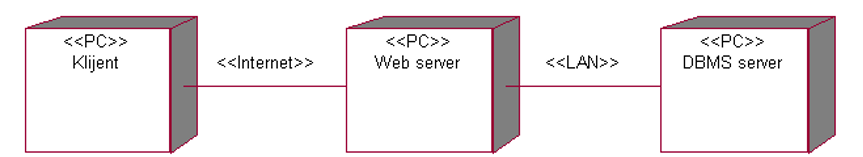
### MongoDB Server

MongoDB Server je proces koji izvršava funkcionalnost MongoDB sistema za upravljanje bazama podataka. Ovaj proces može konkurentno da prihvati određen broj upita, izvrši ih nad bazom podataka i vrati rezultate procesu koji je upite postavio.

# Pogled na raspoređivanje sistema

Pogled na raspoređivanje sistema prikazuje različite fizičke čvorove za najopštiju konfiguraciju sistema. Fizičkim čvorovima koji predstavljaju procesore vrši se dodeljivanje identifikovanih procesa.

Na sledećoj slici dat je UML dijagram raspoređivanja PeNcIL portala.



Web čitač Node API server

## Klijent

Korišćenje XChange aplikaciji se obavlja preko klijentskih računara na kojima se izvršava Web čitač. Za povezivanje između klijenta i Web servera koristi se Internet infrastruktura tako da nema ograničenja u pogledu lokacije klijenta.

## Web server

Računar na kom se izvršava statički server opslužuje klijente statičkim kodom Angular aplikacije. Pored njega, postoji i Node server koji opslužuje API zahteve klijenata, na istom ili drugom računaru u mreži. Serverima se pristupa preko Interneta.

## DBMS server

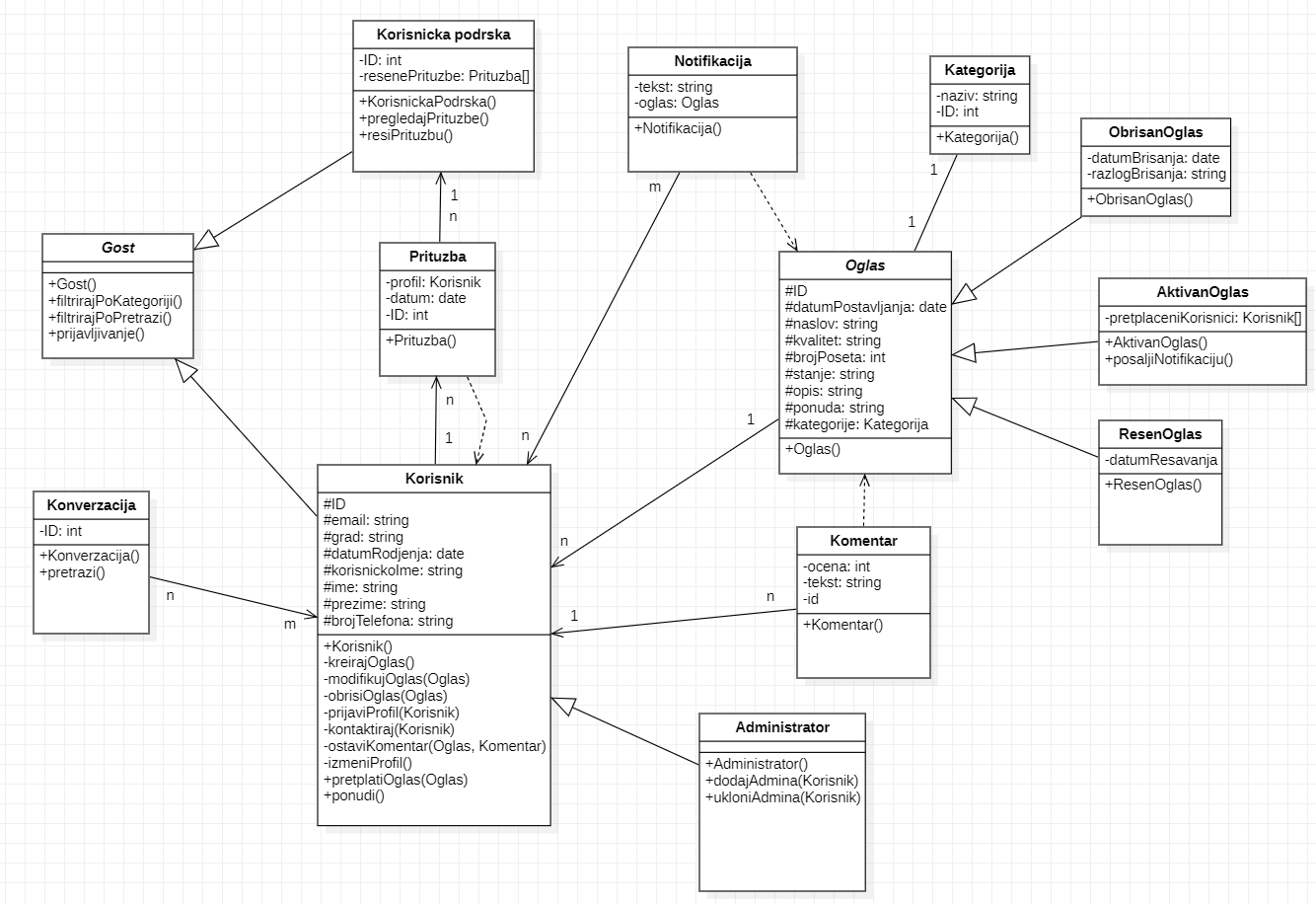
DBMS server je računar na kome se izvršava MongoDB Server proces koji realizuje funkcionalnost sistema za upravljanje bazama podataka. Zbog sigurnosti podataka koji se na ovom računaru čuvaju pristup bazi je ograničen samo na računare iz lokalne mreže (LAN).

# Pogled na implementaciju sistema

Pogled na implementaciju prikazuje različite aspekte bitne za implementaciju sistema. U slučaju XChange aplikacije ovaj odeljak sadrži model domena, šemu baze podataka i prikaz komponenti sistema razvrstanih u ranije identifikovane pakete.

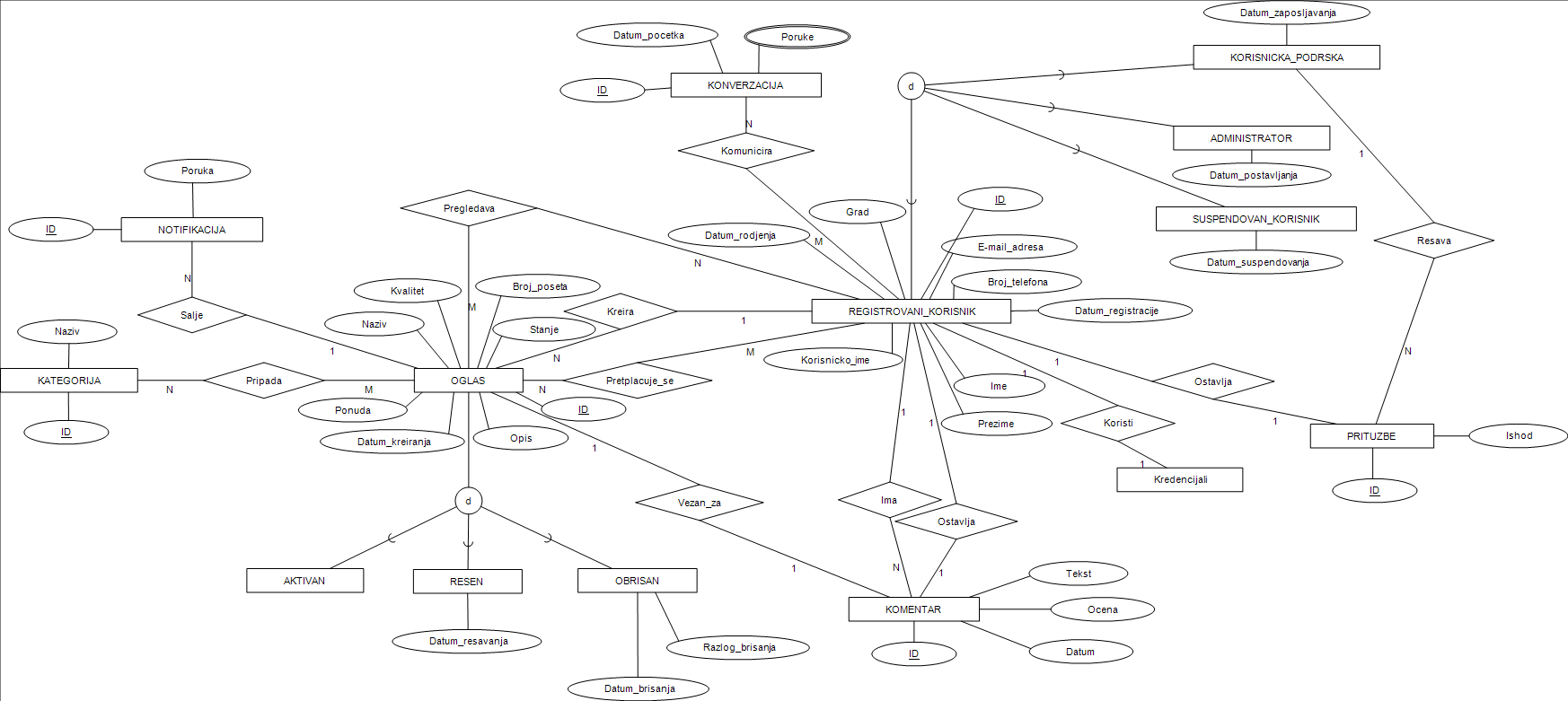
## Model domena

Model domena za koji je projektovana XChange aplikacija predstavljen je UML dijagramom klasa na najvišem nivou funkcionalnosti. Tu su prikazane klase, neki njihovi atributi i funkcionalnosti kao i povezanosti između njih. Ovo predstavlja osnov za projektovanje baze podataka ali i komponenti logike koje će biti implementirane.



## Šema baze podataka

Detaljna šema baze podataka je prikazana na sledećem dijagramu. Baza podataka i dijagram su kreirani korišćenjem *#EER*-a.



## Komponente sistema

Komponente sistema XChange aplikacije su Angular komponente čiji će pregled biti dat po arhitekturnim slojevima. Za ilustraciju će biti korišćeni UML dijagrami komponenti, ali i dijagrami klasa.

### Komponente korisničkog interfejsa

Dizajn korisničkog interfejsa je obuhvaćen dvema komponentama:



Komponenta **main.ts** je implementira stranicu aplikacije tako što predstavlja bootstrap za početni Angular app-module i glavni .scss fajl za stilove.

Komponenta **styles.scss** predstavlja opis stilova za pojedine HTML elemente koji se javljaju na različitim stranicama.

### Komponente aplikacione logike

Komponente koje realizuju domen problema se uključuju isključivo preko **app-component** komponente aplikacije. Komponente aplikacione logike su obeležene „service“ kako bi se označilo da se odnose na logiku.

### Komponente za pristup podacima

Pristup bazi podataka je izdvojen u svoje komponente. Pomenute komponente se uključuju u app-module, tako da su sve funkcije za pristup podacima dostupne svim komponentama.

Podaci su modelirani komponentama koje su izdvojene u svoj folder …/models.

# Performanse

Izabrana arhitektura softvera podržava zahteve u pogledu broja korisnika koji mogu simultano pristupati sistemu i vremena odziva za pristup bazi podataka specificirane u zahtevima u pogledu performansi [5]:

1. Sistem će da podrži veoma veliki broj simultanih pristupa korisnika.
2. Vreme potrebno za pristupanje bazi podataka u cilju izvršenje nekog upita ne sme da bude veće od 5 sekundi.

Zahtevane performanse su zadovoljene izborom tehnologija na kojima će sistem biti razvijen i definisane hardverske platforme [5].

# Kvalitet

Izabrana arhitektura softvera podržava zahteve u pogledu dostupnosti i srednjeg vremena između otkaza specificirane u zahtevima u pogledu pouzdanosti [5]:

1. XChange aplikacija će biti dostupan 24 časa dnevno, 7 dana u nedelji. Vreme kada portal nije dostupan ne sme da pređe 10%.
2. Srednje vreme između dva sukcesivna otkaza ne sme da padne ispod 120 sati.