

**Aplikacija za nekretnine**

**Predmet: CS230 – Distribuirani sistemi**

Školska godina 2018/19.

**Dokumentacija**

Ime i prezime: Aleksandra Jovanović

Broj indeksa: 1985

*Beograd, Avgust 2019.*

Sadržaj

[1. Uvod 2](#_Toc18345626)

[1.2. Područje 2](#_Toc18345627)

[1.3. Metodologija izrade 2](#_Toc18345628)

[1.4. Definicije, akronimi i skraćenice 2](#_Toc18345629)

[1.5. Reference 3](#_Toc18345630)

[2. Opis sistema 4](#_Toc18345631)

[2.1. Osnovne funkcije i cilj projekta 4](#_Toc18345632)

[2.2. Zahtevi korisnika 5](#_Toc18345633)

[2.3. Ograničenja 5](#_Toc18345634)

[3. Specifikacija zahteva 5](#_Toc18345635)

[3.1. Funkcionalni zahtevi korisnika - agenata 5](#_Toc18345636)

[3.1.1. Registracija 6](#_Toc18345637)

[3.1.2. Prijava na sistem 6](#_Toc18345638)

[3.1.3. Pregled unetih oglasa 6](#_Toc18345639)

[3.1.4. Brisanje oglasa 6](#_Toc18345640)

[3.1.5. Ažuriranje oglasa 6](#_Toc18345641)

[3.1.6. Kreiranje oglasa 7](#_Toc18345642)

[3.2. Funkcionalni zahtevi klijenata 8](#_Toc18345643)

[3.2.1 Pregled najnovijih oglasa 8](#_Toc18345644)

[3.2.2. Pretraga oglasa 8](#_Toc18345645)

[3.2.3. Kontaktiranje agenta 8](#_Toc18345646)

[3.3. Nefunkcionalni zahtevi 8](#_Toc18345647)

[3.3.1 Performanse 8](#_Toc18345648)

[3.3.2 Bezbednost 9](#_Toc18345649)

[3.3.3 Zaštita 9](#_Toc18345650)

[3.3.4 Raspoloživost 9](#_Toc18345651)

[3.3.5 Lakoća održavanja 9](#_Toc18345652)

[3.4. Slučajevi korišćenja 9](#_Toc18345653)

[3.4.1. Slučajevi korišćenja – agent 9](#_Toc18345654)

[3.4.2. Slučajevi korišćenja – klijent 16](#_Toc18345655)

[3.5. Konceptualni model baze podataka 19](#_Toc18345656)

[3.6. Fizički model 20](#_Toc18345657)

[3.7. Izgled baze podataka 20](#_Toc18345658)

[3.8. Klasni dijagram 21](#_Toc18345659)

[3.9. Sistem sekvencijalni dijagrami 22](#_Toc18345660)

[3.9.1 Prijavljivanje na sistem 22](#_Toc18345661)

[3.9.2 Pregled unetih oglasa 23](#_Toc18345662)

[3.9.3 Ažuriranje oglasa 24](#_Toc18345663)

[3.9.4 Unos novog olasa 25](#_Toc18345664)

[4. Arhitektura sistema 25](#_Toc18345665)

[4.1. Infrastrukturni dijagram 26](#_Toc18345666)

[5. Implementacija i testiranje 27](#_Toc18345667)

[5.1. Implementacija sistema 27](#_Toc18345668)

[5.2. Slike ekrana Sistema 31](#_Toc18345669)

[5.2.1. Forma za kreiranje korisničkog naloga 31](#_Toc18345670)

[5.2.2. Početna strana – pregled svih dostupnih oglasa 31](#_Toc18345671)

[5.2.3. Prikaz detalja željenog oglasa 31](#_Toc18345672)

[5.2.4. Izmena već kreiranog oglasa 32](#_Toc18345673)

[5.2.5. Unos kategorije oglasa 33](#_Toc18345674)

[5.2.6. Pregled kategorije oglasa 34](#_Toc18345675)

[5.2.7. Unos grada 35](#_Toc18345676)

[5.2.8. Unos u tabelu područje 37](#_Toc18345677)

[5.2.9. Unos I izgled tabele informacije 39](#_Toc18345678)

[5.2.10. Izgled tabele tip 40](#_Toc18345679)

[5.2.11. Unos novog oglasa 40](#_Toc18345680)

[5.3. Testiranje sofrvera 41](#_Toc18345681)

[5.4. Evolucija sistema 42](#_Toc18345682)

[6. Zaključak 42](#_Toc18345683)

# 1. Uvod

Za potrebe izrade projektnog zadatka iz predmeta CS230 – Distribuirani sistemi biće prikazane sve faze razvoja jednog softverskog *sistema agencije za nekretnine.*

Tema projekta je dokumentovanje i implementacija veb aplikacije koju će koristiti agencija za nekretnine kao stalan softver koji treba biti uključen u svakodnevno poslovanje.

## 1.2. Područje

Dokumentacija sistema sadrži:

* Opšti opis sistema
* Prikaz i detaljan opis funkcionalnih i nefunkcionalnih zahteva
* Prikaz i detaljan opis dijagrama slučaja korišćenja (UseCase Diagram)
* Uvid u apstraktne dijagrame sekvenci za sve slučajeve korišćenja
* Prikaz i detaljan opis klasnog dijagrama
* Prikaz i opis aplikacijskog i infrastrukturnog dijagrama
* Opis implementacije i testiranja sistema
* Plan evaluacije sistema

## 1.3. Metodologija izrade

Što se tiče razvojnih metodologija s obzirom da se radi o sistemu srednje veličine u kom su zahtevi od starta duboko i precizno definisani opredelila sam se za agilnu metodu razvoja softvera koja bi se manifestovala kroz inkrementalni razvoj. Dakle, definisaće se početni zahtevi koje treba u realizovati u što većoj meri, a zatim će se konkretna verzija softvera dublje analizirati i proširivati u skladu sa dodatnim zahtevima i potrebama. Uz sam agilni pristup vođenja projekta biće primenjen i „scrum“ metodologija iz razloga što se akcentuje timski rad što podrazumeva zajedničko donošenje odluka na redovnim sastancima.

## 1.4. Definicije, akronimi i skraćenice

Kako bi sama dokumentacija bila razumnija i jasnija potrebno je upoznati se sa terminima i stranim pojmovima koji će se u nastavku dokumentacije nalaziti. U nastavku će neki od termina i akronima biti zasebno opisani:

* Agilni model – Jedan od pristupa razvoju softvera koji se oslanja na inkrementalni i iterativni metod razvoja;
* Inkrementalni model - Proces razvoja koji se zasniva na postupku postepenog razvoja novih verzija softvera;
* PowerDesigner – Kolaborativni alat za modelovanje sistema, razvijen od strane Sybase-a, a trenutno u vlasništvu firme SAP;
* UML – Unified Modeling Laguage je standardni jezik za specifikaciju, vizuelizaciju, konstruisanje i dokumentovanje softverskog sistema;
* NetBeans IDE 8.2. - Integrisano razvojno okruženje koje pruža sveobuhvatne pogodnosti za programere prilikom razvoja softvera;

## 1.5. Reference

[1] PowerDesigner, <https://www.sap.com/westbalkans/products/powerdesigner-data-modeling-tools.html>;

[2] Java, <https://docs.oracle.com/javase/tutorial/java/concepts/>;

[3] Software engineering patterns, <https://www.geeksforgeeks.org/software-design-patterns/>;

[4] UML diagrams, <https://www.tutorialspoint.com/uml/index.htm>;

[5] JUnit Test Framework, <https://www.tutorialspoint.com/junit/junit_test_framework.htm>;

# 2. Opis sistema

Prema zakonu o nepokretnostima, u nepokretnosti spadaju: zemljište (poljoprivredno, građevinsko, šume, šumsko zemljište), zgrade, posebni delovi zgrada, stanovi, poslovne prostorije, garaže, garažna mesta i drugi objekti.

Pošto obavljaju poslove kupoprodaje i izdavanja nepokretnosti, agencije za promet nekretnina imaju potrebu da vode evidenciju o nepokretnostima, njihovim vlasnicima i klijentima koji su ili kupci ili su zainteresovani za iznajmljivanje nekretnine. Potrebno je voditi evidenciju i o drugim podacima vezanim za poslove kupo-prodaje ili iznajmljivanja nekretnina.

Agencije uglavnom vrše delatnost kao posrednici prilikom kupoprodaje ili iznajmljivanja. Naknada koju agencija potražuje na osnovu usluga koje pruža se definiše ugovorom između vlasnika nepokretnosti i same agencije. To može biti procenat od vrednosti nepokretnosti ili na drugi način definisana naknada.

Agenciju zanima da za svakog vlasnika zna ime, adresu i telefon.

Za svaku nepokretnost koja se nalazi u ponudi agencije, treba da postoje podaci o lokaciji na kojoj se nalazi tj. Država, grad, ulica u kojoj je pomenuta nepokretnost.

Pored toga za sam objekat koji se izdaje ili prodaje treba da je poznat tip tog objekta, površina na kojoj se prostire, cena, ko je vlasnik i na kojoj je lokaciji. Ukoliko sam objekat ima neke dodatne kakakteristike kao što su terasa, parking mesto, klima, grejanje ili kablovska – potrebno je i to navesti kao Dodatno.

Neki vlasnici žele da uz ponudu istaknu i fotografiju objekta. Treba im i to omogućiti tako što ce svaka fotografija koju agencija dobije imati svoj ID i njoj pridružen ID objekta na koji se odnosi.

Agenciji se javljaju klijenti koji žele da kupe ili iznajme nekretninu. I klijenti ostavljaju agenciji iste podatke kao i vlasnici nekretnina.

Na kraju, vlasnici nekretnina uz posredstvo agencije potpisuju sa klijentima ugovor o zakupu ili prodaji.

Svaki ugovor ima svoj redni broj, datum nastanka, datum kada ističe. U ugovoru se navodi ko je vlasnik nekretnine, ko klijent i navodi se iznos na koji se ugovor sklapa, kao i naknada koja se isplaćuje agenciji.

## 2.1. Osnovne funkcije i cilj projekta

Glavni problem koji treba da reši ovaj projekat je da omogući jednostavno i brzo plasiranje informacija koje se odnose na nekretnine u obliku oglasa od strane agencije za nekretnine i sa druge strane lak i efikasan pristup informacijama od strane potencijalnih klijenata agencije. Cilj projekta je da se zaposlenima u agenciji za nekretnine olakša poslovanje i omogući bolja interakcija sa potencijalnim klijentima.

## 2.2. Zahtevi korisnika

Za samo korišćenje sistema nije potrebno napredno poznavanje računarskih sistema, kao i rada nad računarima jer je sistem dosta jednostavan, intuitivan i lak za korišćeće, a user-friendly interfejs to upotpunjava i čini proces upravljanja sistemom za stambene zajednice dosta bržim i nezahtevnim.

## 2.3. Ograničenja

Treba napomenuti da je sistem rađen na srpskom jeziku za naše govorno područje, i nema opciju za izbor nekog drugog jezika.

Ograničenja su:

* Sistem mora biti zaštićen
* Baza podataka mora bizi zaštićena
* Ukoliko dođe do grešaka sistem treba da ih obradi
* Potrebno je popuniti sva polja, a ukoliko dođe do greške sistem bi obavestio korisnika

# 3. Specifikacija zahteva

Specifikacija zahteva sadrži sve funkcionalne i nefunkcionalne zahteve sistema i osnovne slučajeve korišćenja sistema. Funkcionalni zahtevi predstavljaju izjave o servisima koje sistem treba da obezbedi, dok se nefunkcionalni primenjuju na ceo sistem, a ne na pojedina svojstva sistema ili servise.

Analizirajući sve funkcionalnosti koje trebaju biti obezbeđene kroz ovaj sistem možemo razlikovati tri osnovna modula za tri korisničke grupe. Prvu grupu bi predstavljali klijenti koji imaju potrebe za uslugama agencije za nekretnine, drugu službenici koji imaju zadatak da ažuriraju i dodaju sadržaj koji će klijenti pregledati i treću administrator koji će pružati pomoć agenciji prilikom rada sa sistemom tako što će u saradnji sa agencijom definisati potrebne parametre kako bi službenicima olakšao i konkretizovao obavljanje njihovog dela posla.

Svaki od pomenutih modula će biti posebno analiziran i projektovan prema svojoj korisničkoj ciljnoj grupi.

Tri role koje se mogu razlikovati u okviru sistema su:

1. Službenik – agent (u okviru ove role službenika spada i administrator)
2. Klijent

## 3.1. Funkcionalni zahtevi korisnika - agenata

Da bi mogao uspešno da koristi funkcionalnosti koje ovaj sistem pruža, svaki korisnik - agent mora najpre posedovati sopstveni korisnički nalog, što se realizuje kroz unošenje podataka u formu za registraciju. Nakon uspešne registracije, korisnik se loguje na sistem i u mogućnosti je da otpočne korišćenje sistema. U nastavku se nalaze objašnjenja svih funkcionalnih zahteva korisnika pojednično.

### 3.1.1. Registracija

Da bi mogao uspešno da koristi funkcionalnosti koje ovaj sistem pruža, svaki korisnik mora najpre posedovati sopstveni korisnički nalog.

### 3.1.2. Prijava na sistem

Korisnik može da se prijavi na sistem nakon uspešne registracije tako što unosi svoje jedinstvene pristupne podatke (e-mail i lozinku) u polja forme, predviđena za to, na početnoj strani. Nakon uspešne prijave na sistem, korinik je u mogućnosti da koristi sve funkcionalnosti sistema.

### 3.1.3. Pregled unetih oglasa

Agent će moći da pregleda sve oglase koji su uneti od strane njega ili ostalih službenika. Ova funkcionalnost je neophodna kako bi agent mogao da vodi evidenciju o svim oglasima koji su uneti kao i njihovim definisanim informacijama. Ova funkcionalnost uključuje mogućnost pregleda oglasa pomoću filtera, a na nju će se nadovezivati funkcionalnosti ažuriranja informacija o oglasu i brisanja unetih oglasa.

### 3.1.4. Brisanje oglasa

Brisanje oglasa je funckionalnost namenjena službeniku i nudi se u okviru pretrage oglasa. Dakle, službenik nakon što odabere neki od unetih oglasa ima mogućnost i da ga izbriše ukoliko želi. Ova funkcionalnost je potrebna ukoliko službenik želi da ukloni oglas koji više nije u ponudi agencije.

### 3.1.5. Ažuriranje oglasa

Ažuriranje oglasa je takođe jedna od primarnih funkcija u okviru službeničkog modula i pruža agentu mogućnost da ažurira odnosno modifikuje informacije o oglasu. U okviru ažuriranja se takođe uključuje mogućnost dodavanja novih informacija o oglasu.

#### 3.1.5.1. Unos dodatnih informacija

Da bi službenik mogao da unese oglas mora da popuni neke neophodne informacije kako bi se oglas uspešno uneo, ali takođe treba da ima mogućnost da pored osnovnih informacija dodaje i neke specifične informacije koje su važne za konkretan oglas.

#### 3.1.5.2. Izmene već unetih informacija

Agent može da menja informacije u okviru oglasa što takođe predstavlja proširujuću funkcionalnost u okviru pretrage s obzirom da je prvo potrebno da ga locira (putem pretrage), a tek onda može da menja definisane informacije. Ova funkcionalnost je neophodna ukoliko je agent uneo oglas definišući joj neku informaciju pogrešno kao i ukoliko agencija iz bilo kog razloga želi da promeni neke od trenutnih informacija o oglasu.

#### 3.1.5.3. Brisanje već unetih dodatnih informacija

Službeniku treba omogućiti da briše neke od informacija koje je dodao ukoliko one više nisu potrebne za određeni oglas.

#### 

### 3.1.6. Kreiranje oglasa

Unos oglasa je takođe jedna od primarnih funkcionalnosti u okviru modula za službenike i omogućava službeniku da dodaje nove oglase tako što definiše neophodne informacije za oglas. Definisanje neophodnih informacija o oglasu se realizuje kroz popunjavanje forme od strane službenika.

#### 3.1.6.1. Popunjavanje forme za unos novog oglasa

Popunjavanje forme za unos informaicija o oglasu je podrazumevana aktivnost koja uključuje dodavanje novih oglasa. Polja koja se moraju popuniti da bi se kreirao nov oglas su:

* grad
* deo grada
* tip oglasa
* kategorija
* cenu
* površina
* adresa

Pored obaveznih službeniku su na raspolaganju i opciona polja koja takođe mogu pružiti važne informacije o oglasu.

* Naslov, lokacija I opis

## 3.2. Funkcionalni zahtevi klijenata

Klijenti će pomoću aplikacije moći da pregledaju trenutne ponude nekretnina koje nudi agencija. Modul namenjen klijentima agencije bi predstavljao klasičan veb sajt sa dinamičkim sadržajem koji će se unositi sa u okviru modula za službenike. Posetilac sajta može da bude bilo koji korisnik koji želi da kupi ili iznajmi nekretninu i za to mu nisu potrebni registracija i korisnički nalog.

### 3.2.1 Pregled najnovijih oglasa

Potrebno je automatski nakon što klijent pristupi sajtu prikazati oglase koji su najkasnije dodati na sajt. Ne očekuje se da se ova funkcionalnost koristi u velikoj meri s obzirom da su klijenti uglavnom zainteresovani za konkretne oglase, ali ideja krene sa prikazom najnovijih oglasa na početnoj strani sajta.

### 3.2.2. Pretraga oglasa

Potrebno je omogućiti korisniku da definiše filtere tj. kriterijume pomoću kojih će moći da pregleda specifične oglase u zavisnosti od njegovog interesovanja i potreba.

### 3.2.3. Kontaktiranje agenta

Potrebno je klijentu omogućiti da ukoliko želi može kontaktirati agenciju slanjem email-a direktno sa sajta. Pored toga u okviru sajta trebaju biti dostupni ostali telefonski i email kontakti agencije.

## 3.3. Nefunkcionalni zahtevi

Nefunkcionalni zahtevi su ograničenja servisa ili funkcija koje sistem obezbeđuje. Mogu a obuhvate vremenska ograničenja, ograničenja procesa razvoja i ograničenja nametnuta od strane standarda. Nefunkcionalni zahtevi se primenjuju na ceo sistem a ne na njegove pojedine delove. To su najčešće zahtevi o performansama.

### 3.3.1 Performanse

Kada su u pitanju perfomranse potrebno je da se veb strane i sadržaj iz baze učitavaju i prikazuju optimalnom brzinom. Ova brzina zavisi i od internet konekcije koju korisnici imaju tako da se ne može garantovati maksimalni vremenski interval učitavanja i prikaza strana što ujedno nije neophodna stavka s obzirom da sistem po svojoj nameni ne spada u kritične sisteme. Zahtevi za sadržajem u bazi će se realizovati preko jednostavnijih upita tako da se neće troštiti puno vremena za učitavanje strana.

### 3.3.2 Bezbednost

Po pitanju bezednosti treba voditi računa je da se onemogući neautentifikovan i neautorizovan pristup administratorskom i službeničkom nalogu. Pored toga pošto je sa klijentske strane omogućeno slanje email-a agenciji direktno sa sajta potrebno je obezbediti prevenciju nekontrolisanog slanja email-ova tako što će se implenetirati neki od proverenih anti-bot servisa.

### 3.3.3 Zaštita

Kritične akcije koje se nude u okviru sistema predstavljaju privilegije unosa sadržaja u bazu. Sadržaj koji će se unositi će se svakako prikazivati na sajtu namenjenom klijentima tako da se ne može time ne može ugroziti poslovni prosperitet firme iz razloga što se u bazi ne čuvaju poverljive informacije. Nije potrebno obezbediti visok stepen zaštite i samim tim što se mogućnost manipulacije sa sadržajem baze podataka ograničava na module za službenike i administratore koji su obezbeđeni autentikacijom i autorizacijom, biće u potpunosti zadovoljiti potrebni stepen zaštite.

### 3.3.4 Raspoloživost

Sajt treba da bude dostupan za korišćenje na raznim tipovima uređaja poput tablet urađaja i mobilnih telefona što zahteva da ga karakteriše prilagodljiv (responsive) dizajn.

### 3.3.5 Lakoća održavanja

Iako sistem ima jasno definisane i u nekoj meri ograničene zahteve kao i za svaki drugi softver treba obezbediti neophodnu modularnost koja će olakšati razvoj eventualnih naknadnih funkcionalnosti tako što treba težiti da se u što većoj meri koriste postojeće tj. već razvijene komponente.

## 3.4. Slučajevi korišćenja

Slučajevima korišćenja prikazuju se sve funkcionalnosti koje imaju korisnici sistema. Svaki slučaj korišćenja ima učesnike, opis, preduslove koje je potrebno ispuniti, osnovni scenario, alternativni scenario, izuzetke i postuslove.

Slučajevi korišćenja će biti podeljeni u dve kategorije:

* Slučajevi korišćenja – korisnik - agent
* Slušajevi korišćenja – admnistrator

### 3.4.1. Slučajevi korišćenja – agent

Na slici 3.4.1. možemo videti UML prikaz svih slučajeva korišćenja agenata.



Slika 3.4.1. UseCase model za agenta

U nastavku sledi detaljan opis svakog slučaja korišćenja.

#### 3.4.1.1. Prijava na sistem

Potrebno je najpre agent da obavi prijavu kako bi pristupio svom modulu i mogao da obavlja funkcionalnosti koje su mu namenjene. Cilj autentikacije službenika je da se onemogući neutorizovano obavljanje funkcionalnosti koje su namenjene službeniku, što uključuje manipulisnje sa sadržajem baze podataka.

|  |  |  |  |
| --- | --- | --- | --- |
| **Preduslovi** | **Način realizacije** | **Greške** | **Šta je postignuto** |
| Da bi službenik mogao da pristupi svom modulu potrebno je da ima link do login forme koji će mu programer obezbediti gde će obaviti prijavu. | U okviru login forme službenik treba da upiše korisničko ime i šifru. | Pogrešan unos korisničkog imena ili izostavljanje istog će sprečiti službenika da pristupi svom modulu. | Službeniku se nakon uspešno obavljenog prijavljivanja na sistem omogućuje da obavlja ostale funkcionalnosti definisane u okviru njegovog modula. |
| Da bi službenik mogao da uspešno prođe procese autentikacije i autorizacije neophodno je da mu administrator prvo kreira nalog definišući mu korisničko ime i šifru. | Ukoliko prođe autentikaciju i autorizaciju službenik pristupa svom modulu. | Pogrešan unos šifre ili izostavljanje iste će takođe sprečiti službenika da pristupi svom modulu. |  |

#### 3.4.1.2. Pregled oglasa

|  |  |  |  |
| --- | --- | --- | --- |
| **Preduslovi** | **Način realizacije** | **Greške** | **Šta je postignuto** |
| Da bi mogao da pregleda oglase definišući filtere za pretragu službenik mora uspešno da pristupi svom nalogu unošenjem odgovarajućeg korisničkog imena i širfre. | Kada pristupi svom modulu, službeniku će se na početnoj strani izlistati svi uneti oglasi po naslovima i pored toga će mu biti ponuđeni svi filteri za pretragu koji su dostupni. | Ukoliko službenik pogrešno definiše neki filtera za pretragu to će se manifestovati tako što mu se u listi neće prikazati ni jedan oglas (filteri će se definisati većinom preko padajućih listi, check ili radio button-a tako da će se maksimalno smanjiti mogućnost pravljenja grešaka). | Definisanjem filtera za pretragu službeniku se nude samo oglasi koji zadovoljavaju kriterijume koje je zadao i na taj način može pregledati samo specifične oglase. |
|  | Kao filtere pretrage službenik može definisati sledeće: kategoriju, tip oglasa, grad, deo grada, dva polja za unos opsega cene i unos opsega kvadrature (od-do). Takođe službenik može koristiti i pretragu zadavanjem reči koje treba da se pronađu u bilo kojoj informaciji oglasa. |  | Nakon prikaza liste odabranih oglasa korisnik može selektovati oglase kako bi nad njima radio ostale funkcionalnosti koje su mu namenjene (brisanje, menjanje). |
|  | Nakon definisanja pretrage službenik pritiskom na dugme zadaje pretragu nakon čega će mu se lista oglasa ograničiti samo na oglase koji zadovoljavaju kriterijume pretrage. |  |  |

#### 3.4.1.3. Brisanje oglasa

|  |  |  |  |
| --- | --- | --- | --- |
| **Preduslovi** | **Način realizacije** | **Greške** | **Šta je postignuto** |
| Da bi mogao da briše oglase službenik mora uspešno da pristupi modulu unošenjem odgovarajućeg korisničkog imena i šifre na login strani. | Prvo potrebno locirati oglas za brisanje u okviru liste svih oglasa ili ga naći korišćenjem filtera tj. pretrage. | Službenik ne može vršiti brisanje oglasa koji nije lociran, odnosno pronađen u listi svih oglasa. | Brisanjem oglasa se sa sajta kao i iz baze uklanjaju svi oglasi koji više nisu u ponudi agencije ili iz bilo kog razloga ne treba da se nalaze na sajtu. |
| Da bi se oglas mogao brisati potrebno je da prvo bude kreiran od strane službenika. | Nakon selektovanja oglasa potrebno je odabrati opciju za brisanje i potvrditi brisanje kako bi se sprečilo slučajno brisnje oglasa. |  |  |
| Neophodno je selektovati (naći putem pretrage) neki od oglasa koje želi da obriše inače opcija za brisanje neće biti ponuđena. |  |  |  |

#### 3.4.1.4. Ažuriranje oglasa

|  |  |  |  |
| --- | --- | --- | --- |
| **Preduslovi** | **Način realizacije** | **Greške** | **Šta je postignuto** |
| Da bi službenik mogao da dodaje nove informacije za oglase potrebno je da uspešno pristupi modulu unošenjem odgovarajućeg korisničkog imena i šifre na login strani. | Kada službenik nađe oglas kome želi da doda nove informacije potrebno je da ga selektuje nakon čega će mu se otvoriti nova strana u okviru koje su prikazane trenutne vrednosti svih informacija. | Ukoliko službenik pokuša da doda informaciju koja je već definisana za dati oglas potrebno ga je upozoriti odgovarajućom porukom. | Sve informacije koje službenik doda za određeni oglas će biti prisutne na sajtu nakon ažuriranja. |
| Kako bi se dodavale nove informacije takođe je neophodno da oglas bude prethodno kreiran od strane službenika. | Potrebno je da korisnik izabere opciju za dodavanje informacija nakon čega će mu se prikazati dva polja u kojima treba da unese naziv informacije tj. podatka o oglasu i vrednost za novokreirani podatak respektivno. |  |  |
| Neophodno je selektovati i (naći putem pretrage) oglas kojem će se unositi dodatne informacije. | Nakon unosa treba da potvrdi dodavanje informacije klikom na dugme. |  |  |

#### 3.4.1.5. Izmena već unetih informacija

|  |  |  |  |
| --- | --- | --- | --- |
| **Preduslovi** | **Način realizacije** | **Greške** | **Šta je postignuto** |
| Da bi službenik mogao da menja informacije oglasa mora uspešno da pristupi modulu unošenjem odgovarajućeg korisničkog imena i šifre na login strani. | U okviru liste svih oglasa ili korišćenjem filtera tj. pretrage, potrebno je locirati oglas kom će se menjati informacije. | Službenik može greškom (slučajno) izmeniti vrednost nekog oglasa, ali mu se ne sme dozvoliti čuvanje informacija bez potvrde klikom na dugme. | Izmenom trenutnih informacija oglasa službenik samo redefiniše oglase i time se ne dodaju novi oglasi u bazi odnosno na sajt. |
| Takođe je neophodno selektovati (ili naći putem pretrage) oglas kojem će se menjati informacije. | Klikom na oglas će se otvoriti posebna strana sa formom koja sadrži sva polja popunjena sa aktuelnim vrednostima. | Ukoliko službenik obriđe vrednost informacije za neko od obaveznih polja potrebno je mu je ispisati poruku da je obrisao obaveznu informaciju kao i naglasiti koja informacija je u pitanju. |  |
|  | Službenik može izmeniti sva polja koja želi i zatim potvrditi izmene pritiskom na dugme. |  |  |

#### 3.4.1.5. Popunjavanje forme za unos novog oglasa

|  |  |  |  |
| --- | --- | --- | --- |
| **Preduslovi** | **Način realizacije** | **Greške** | **Šta je postignuto** |
| Da bi službenik izvršio funkcionalnost dodavanja oglasa mora da bude ulogovan na modul za službenike što takođe podrazumeva unošenje validnog korisničkog imena i šifre u login formu. | Odabira opcije za nov oglas službeniku se otvara forma u kojoj treba da definiše osnovne podatke o oglasu. | Ukoliko službenik propusti unos nekog od obaveznih polja potrebno je prikazati mu poruku i naznačiti polja koja je propustio. | Nakon uspešnog dodavanja novog oglasa oglas sa svim definisanim informacijama će biti dostupan na sajtu gde ga klijenti mogu videti odnosno pretražiti. |
| Nakon uspešne prijave potrebno je odabrati opciju za dodavanje novog oglasa u meniju. | Obavezni podaci koje službenik mora da popuni kako bi kreirao oglas su: kategorija, tip oglasa, grad i deo grada (ukoliko postoji), cena porvšina i adresa nekretnine. |  | Kreiranje novog oglasa otvara mogućnost službeniku da vrši funkcionalnosti brisanja i ažuriranja oglasa (dodavanje novih informacija, dodavanje slika i promena postojećih informacija) |
|  | Pored osnovnih službenik može definisati sledeće podatke: naslov, lokacija, opis, link do goole map-e i takođe može čekirati neki od marekra koji označavaju da se radi o specijalnoj ponudi(smanjena početna cena oglasa), novogradnji ili hitnoj prodaji. |  | Definisane informacije koje službenik popunjava prilikom unosa novog oglasa će se koristiti prilikom pretrage oglasa korišćenjem filtera tako što će predstavljati konkretne vrednosti parametara (kriterijuma) pretrage. |
|  | Nakon unosa svih željenih informacija potvrđuje se unos i ukoliko podaci prođu validaciju oglas se dodaje na sajt kao i u bazu podataka. |  |  |

### 3.4.2. Slučajevi korišćenja – klijent

Iako klijenti imaju najmanje slučaja upotrebe svrha sistema da se njima obezbede potrebne informacije koje se tiču nekretnina. Pored toga je potrebno da se pomenute informacije brzo i jednostavno pronalaze.

Kao što se može primetiti na slici 1 za klijenta su namenjena dva osnovna slučaja korišćenja, a to su pregled oglasa kao i kontakt sa agencijom. Dakle, klijent koristi sistem kao klasičan veb sajt sa dinamičkim sadržajem koji u ovom slučaju predstavlja oglase koji će se dodavati od strane agencije.

U okviru aktivnosti pregledanja oglasa su uključene aktivnosti pregledanja najnovijih oglasa kao i pregledanje oglasa korišćenjem filtera za pretragu oglasa.



*Slika 3.4.2 – Slučajevi korišćenja klijenta*

#### 3.4.2.1. Pregled oglasa

|  |  |  |  |
| --- | --- | --- | --- |
| **Preduslovi** | **Način realizacije** | **Greške** | **Šta je postignuto** |
| Klijent mora da pristupi veb sajtu unošenjem url adrese ili linkovanjem na sajt u okviru veb pretraživača. | Nakon što pristupi veb sajtu klijentu se prikažu najnoviji oglasi koje može detaljnije da pregleda klikom na jedan od njih. | Ukoliko uspešno locira sajt u okviru ovog slučaja upotrebe klijent ne može da napravi grešku. | Postiže se to da korisnik jednostavnim odabirom oglasa može videti sve dostupne informacije o odgovarajućem oglasu. |
|  | Kada odabere neki od oglasa otvara mu se nova veb strana sa prikazom svih unetih informacija vezanih za odabrani oglas. |  |  |

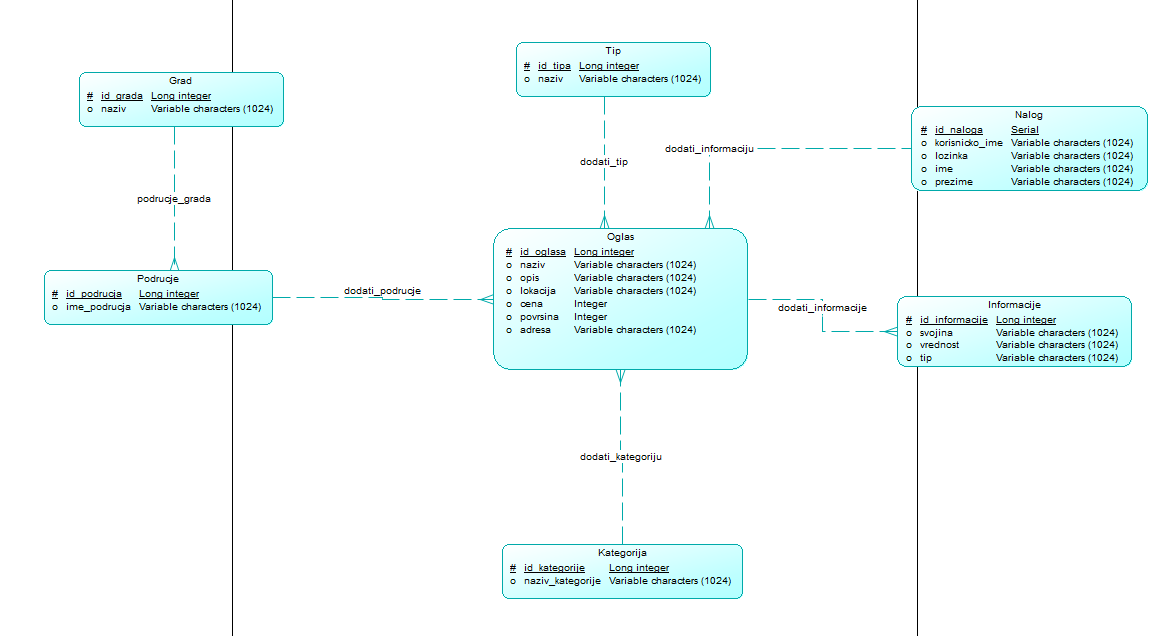
#### 3.4.2.2. Kontakt sa agencijom

|  |  |  |  |
| --- | --- | --- | --- |
| **Preduslovi** | **Način realizacije** | **Greške** | **Šta je postiguno** |
| Potrebno je da klijent pristupi sajtu u okviru veb pretraživača. | Klijent bira u meniju opciju kontakt. | Pogrešno ukucan email ili strane klijenta prouzrokuje problem da agencija neće moći da odgovori na email. | Omogućeno klijentu da kontaktira agenciju slanjem email-a direktno sa sajta. |
|  | Nakon što se klijentu prikaže kontakt forma, potrebno je da popuni polja u kontakt formi (ime i prezime, email, telefon, tekst poruke). | Pogrešno ukucan broj telefona od strane klijenta prouzrokuje problem da agencija neće moći telefonski da kontaktira klijenta. | Takođe se u kontakt segmentu ostavljaju odgovarajući kontakti agencije (email i telefon) kako bi klijent i na taj način mogao da kontaktira agenciju. |
|  | Da bi poslao email klijent mora da reši anti-bot test tako što treba da u odgovarajuće polje da unese brojeve koje vidi na slici (ReCaptcha). | Ukoliko klijent ne unese odgovarajću vrednost u polje za anti-bot proveru email neće biti poslat i klijent će morati ponovo da pokuša. |  |
|  |  | Ako klijent ne popuni ni jedno polje kontakt forme takođe neće moći da pošalje email i ispisaće mu se odgovarajuća poruka. |  |

#### 3.4.2.3. Pretraživanje pomoću filtera

|  |  |  |  |
| --- | --- | --- | --- |
| **Preduslovi** | **Način realizacije** | **Greške** | **Šta je postignuto** |
| Kako bi koristio pretragu oglasa pomoću filtera klijent mora da pristupi veb sajtu unutar veb ptretraživača. | Klijentu se nudi forma preko koje može da definiše kriterijume (filtere) za pretragu.  U kriterijume spadaju kategorija, tip oglasa, grad i deo grada, opseg cene i površine. | S obzirom da su klijentu kao filteri ulavnom ponuđene konkretne vrednosti smanjuje se verovatnoća pogrešno definisanog filtera. | Postignuto je to da korisnik pregleda samo oglase koji ga zanimaju tj. koji mu odgovaraju po pitanju kategorije, tipa, grada i dela grada, cene i površine. |
|  | Kategorija, tip, grad i deo grada korisnik definiše odabirom u padajućoj listi, ček boksu ili radio button-u, a za cenu i površinu opseg unosi preko input polja. Nije obavezno definisanje svih filtera. | Ukoliko klijent loše definiše neki od filtera to će prouzrokovati da mu se kao rezultat vrati prazna lista oglasa kao i poruka da nema oglasa koji zadovoljavaju kriterijume. | Iz liste oglasa koji zadovoljavaju kriterijume pretrage korisnik može jednostavnim odabirom (klikom) oglasa videti sve dostupne informacije o tom oglasu. |
|  | Nakon izlistavanja oglasa koji zadovoljavaju definisane kriterijume potrebno je da klijent klikom izabere konktretan oglas kako bi video sve informacije o njemu. |  |  |

## 3.5. Konceptualni model baze podataka



*Slika 3.5 – Konceptualni model*

Na osnovu modela možemo zaključiti da je u bazi podataka ključni entitet Oglas (Ad), kao što je već pomenuto prilikom prikaza klasnog dijagrama, i da je većina ostalih entiteta direktno povezana sa njim. Entitet Oglas logički predstavlja oglas sa svim definisanim atributima koji pružaju bitne informacije o tom oglasu. Atributi naslov, opis, cena, površina, adresa se unose od strane službenika preko forme u bazu podataka i pretstavljaju osnovne karakteristike oglasa. Pored pomenutih atributa potrebno je definisati i ostale entitete koji su u relaciji sa entitetom oglas kako bi se svaki oglas potpuno definisao.

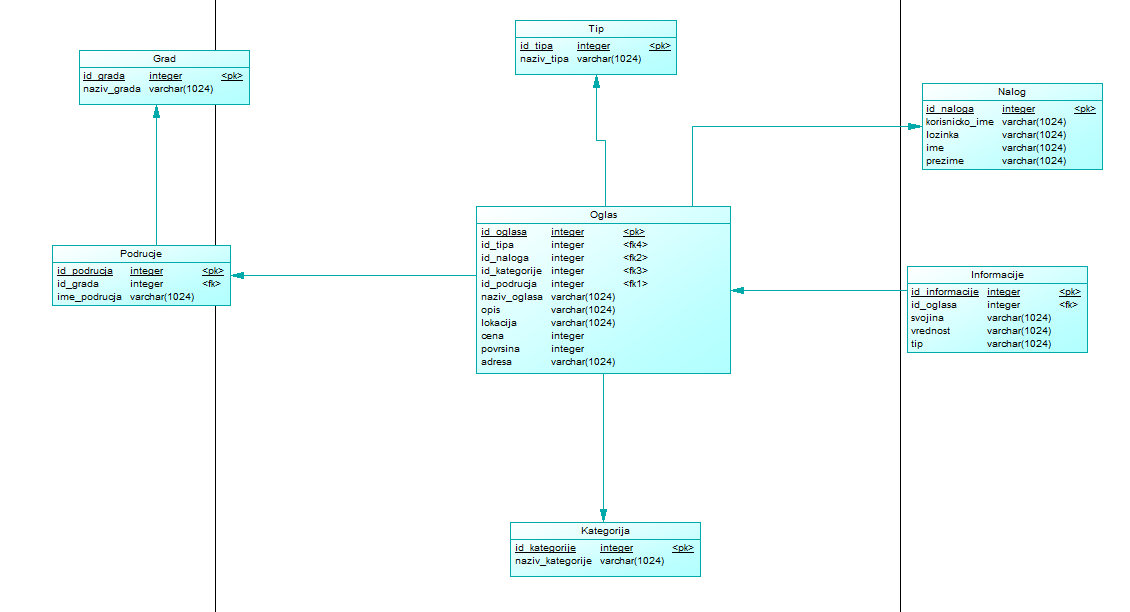
Grad i Deo grada (City i Area) su entiteti kojima se opisuje u kom gradu kao i u kojem tačno delu grada se nalazi nekretnina iz oglasa. Ove dve tabele su u relaciji jedan prema više s obzirom da u svakom gradu možemo imati više delova grada u kojima se mogu nalaziti nekretnine. Identifikator (primarni ključ) iz tablele Deo grada se spušta u centralnu tabelu Oglas i na taj način je pored dela grada dostupan i konkretan grad u kome se nalazi nekretnina.

U okviru entiteta Tip oglasa (Type) se očekuje da se dodaju razni tipovi koji se odnose na oglas i oni nam govore o tome da li se u oglasu radi o prodaji, izdavanju nekretnine ili nekom drugom tipu oglasa. Primarni ključ tabele Tip se takođe spušta u tabelu oglas.

Entitet Kategorija (Category) nam omogućava da definišemo koji tip ili vrsta nekretnine je definisana u oglasu i vrednosti atributa ime (name) u okviru tabele Kategorija mogu biti: stan, kuća, lokal, garsonjera, plac, zemljište itd.

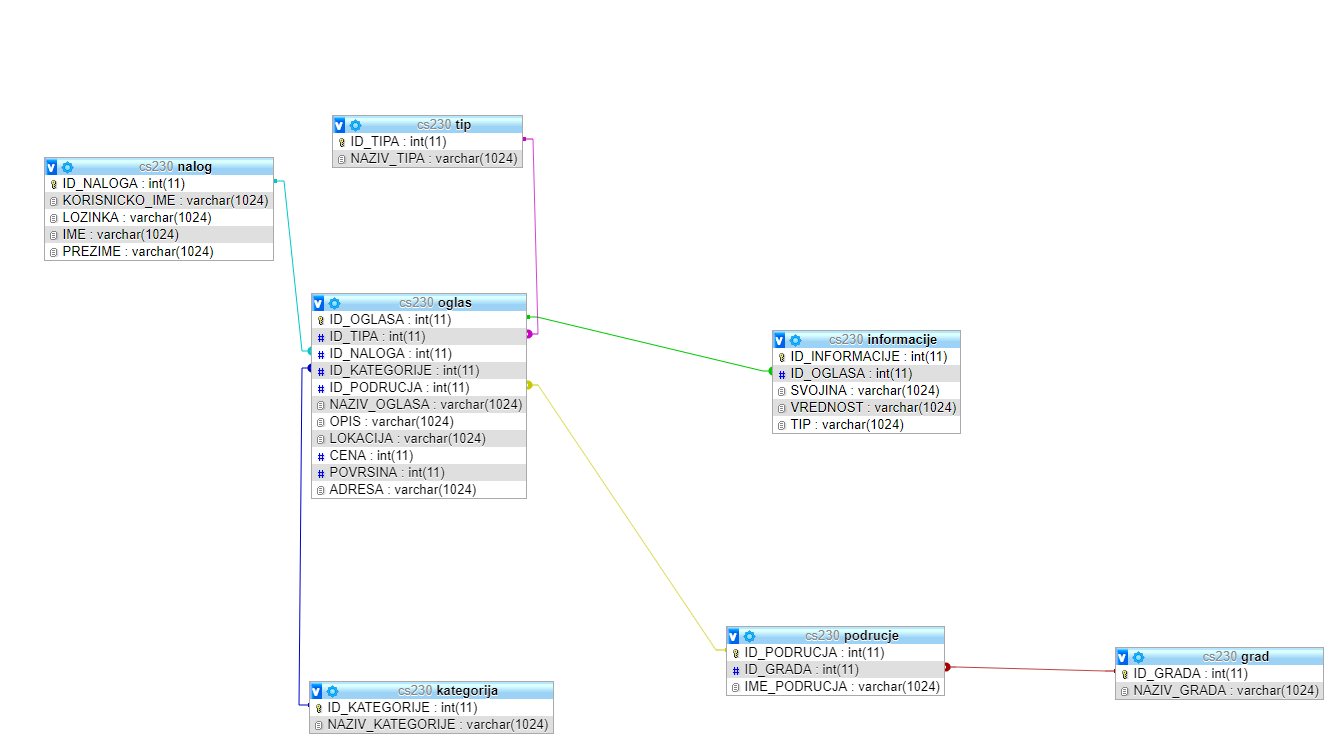
Entitet Informacije (Info) zapravo predstavlja tabelu sa dodatnim informacijama koje službenik može da doda za svaki oglas. Dodatne informacije se mogu opciono uneti za svaki oglas kojem je to potrebno. Kako bi se znalo na koji oglas se odnose dodatne informacije potrebno je da se u tabelu Info prenese primarni ključ odgovarajućeg oglasa (relacija između tabela Oglas i Info je jedan prema više).

## 3.6. Fizički model



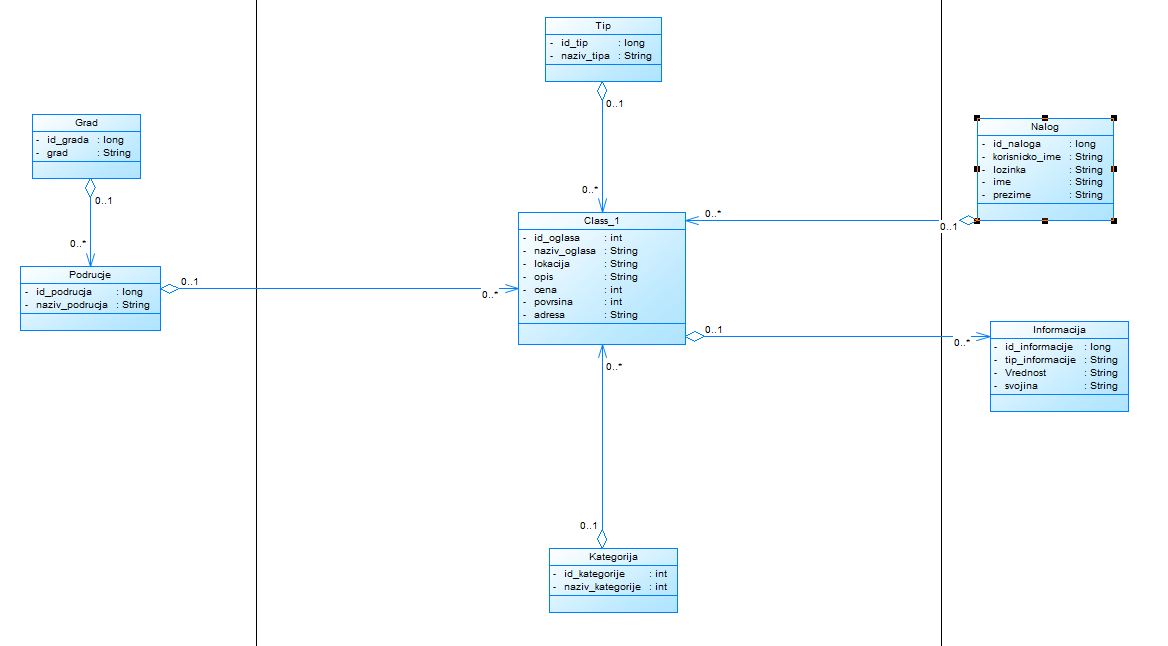
*Slika 3.6 – Fizički model*

## 3.7. Izgled baze podataka



*Slika 3.7 – Izgled baze podataka sa relacijama*

## 3.8. Klasni dijagram



*Slika 3.8 – Klasni dijagram*

## 3.9. Sistem sekvencijalni dijagrami

Za svaki od osnovnih scenarija koji su opisani biće reprezentovan dijagram sekvenci i pored toga će biti opisan tok interakcije u izuzetnim situacijama.

### 3.9.1 Prijavljivanje na sistem



*Slika 3.9.1 – SSD prijave na sistem*

### 3.9.2 Pregled unetih oglasa



*Slika 3.9.2 – SSD pregleda unetih oglasa*

### 3.9.3 Ažuriranje oglasa



*Slika 3.9.3 – SSD ažuriranja oglasa*

### 3.9.4 Unos novog olasa



*Slika 3.9.4 – SSD unosa novog oglasa*

# 4. Arhitektura sistema

Arhitektura sistema nam pokazuje kako bi sistem trebao da bude organizovan i njegovu ukupnu strukturu. Uz pomoć dijagrama arhitekture sistema utvrđuju se glavne strukturne komponente sistema kao i veze između njih. Na slikama ispod biće prikazani UML aplikacijski dijagram, kao i infrastrukturni dijagram.



## 4.1. Infrastrukturni dijagram



Dijagram sa slike se odnosi na krajnji dijagram infrastrukture sistema. Pošto je kod ovog projekta implementirana osnovna verzija sistema koja omogućava osnovne funkcionalnosti sistema.

Ovim dijagramom su predstavljeni svi elementi sistema koji omogućuju da sistem pravilno funkcioniše. Takođe, na dijagramu se mogu videti načini i forme preko kojih se može pristupiti sistemu kao i uređaji preko kojih bi samo pristupanje bilo omogućeno.

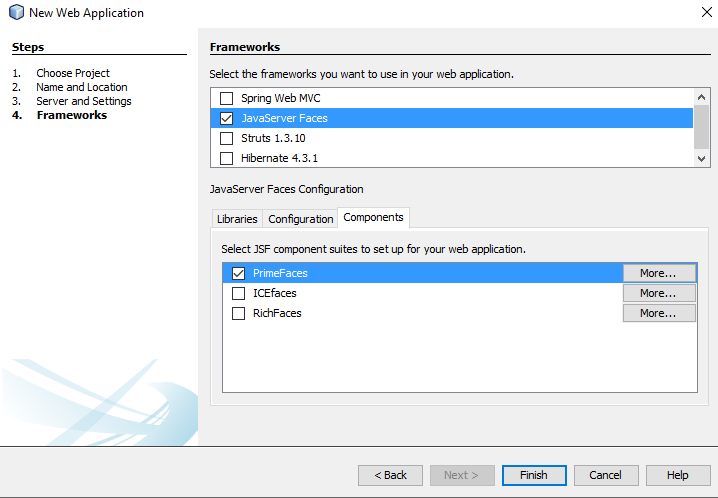
Kako bi korisnici i administrator pristupili sistemu morali bi imati brzu internet konekciju. Zbog same sigurnosti sistema nalazio bi se i Firewall koji bi obezbedio sigurnost samih podataka sistema i sprečio neželjene upade u isti.

# 5. Implementacija i testiranje

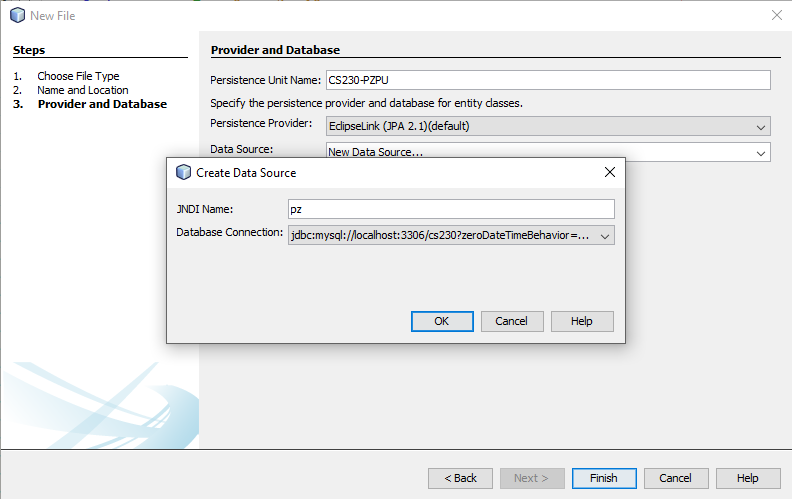
U ovom delu, biće opisani desing patterni tj. šabloni koji su korišćeni u ovom sistemu i tehnologija i metodi koji su se koristili prilikom implementacije. Naravno biće prikazan sam interfejs i predstavljeni moduli sistema. Na kraju, biće više reči o funkcionalnostima koje bi se mogle, u nekim kasnijim, verzijama dodati.

## 5.1. Implementacija sistema

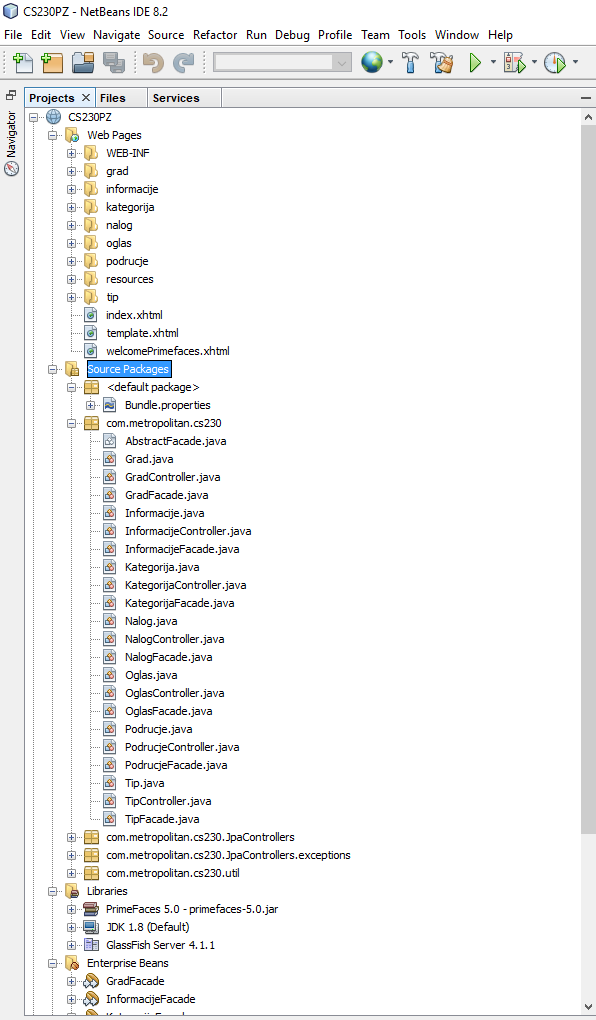
Ovaj sistem je rađen u Javi kao web aplikacija, razvojno okruženje NetBeans 8, a korišćeni su SQL upiti za komunikaciju sa bazom. Za izgled aplikacije, kao šablon, korišćen je PrimeFace Facelet.



*Slika 5.1.1 – Šablon PrimeFaces*

**

*Slika 5.1.2. – Podešavanja konekcije sa bazom podataka*

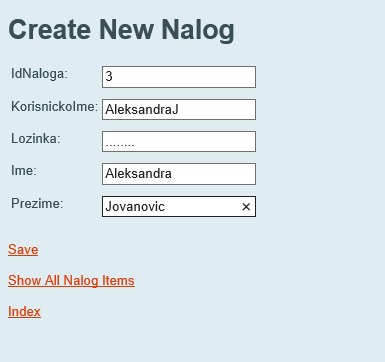
**

*Slika 5.1.3. – Hijerarhija klasa unutar projekta*

## 5.2. Slike ekrana Sistema

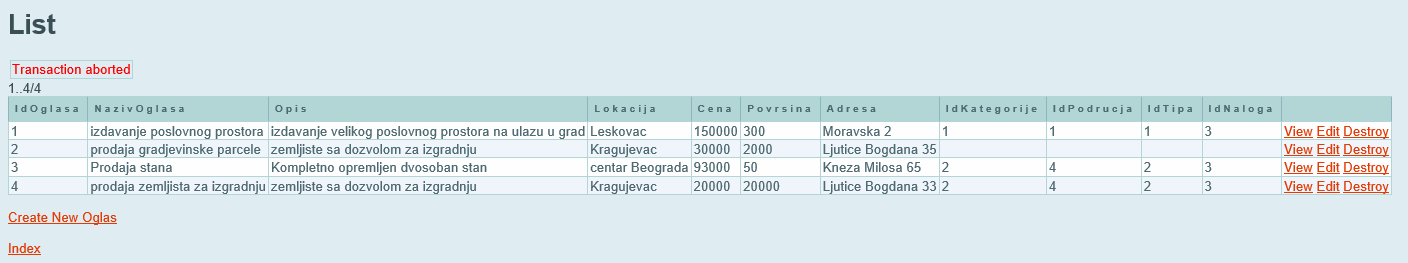
U nastavku će biti prikazan izgled same aplikacije za nekretnine.

### 5.2.1. Forma za kreiranje korisničkog naloga



*Slika 5.2.1. – Kreiranje korisničkog naloga*

### 5.2.2. Početna strana – pregled svih dostupnih oglasa



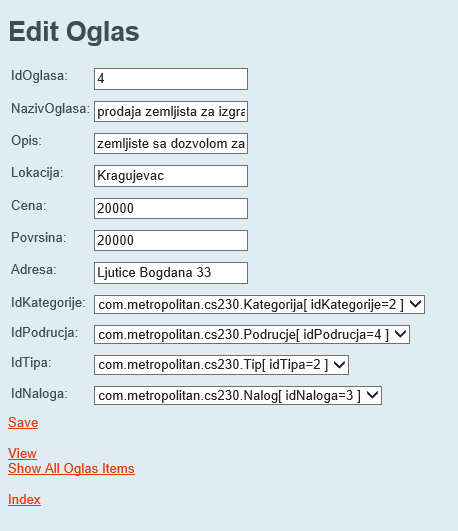
*Slika 5.2.2. – Dostupni oglasi u bazi*

### 5.2.3. Prikaz detalja željenog oglasa



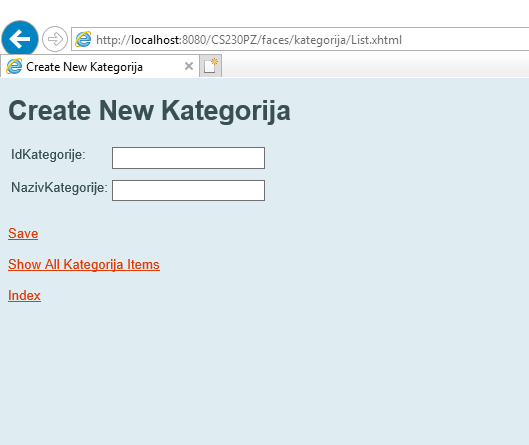
*Slika 5.2.3. – Informacije o oglasu*

### 5.2.4. Izmena već kreiranog oglasa



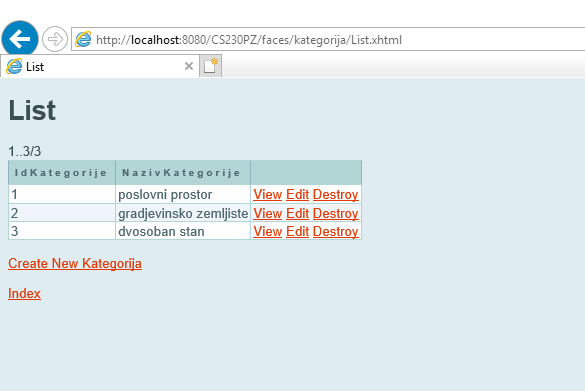
*Slika 5.2.4. – Izmena postojećeg oglasa*

### 5.2.5. Unos kategorije oglasa



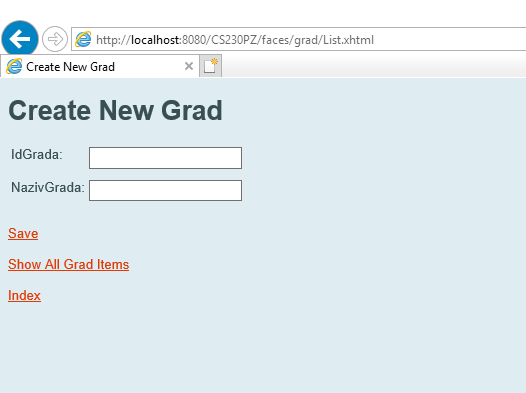
*Slika 5.2.5. – Unos kategorije oglasa*

### 5.2.6. Pregled kategorije oglasa

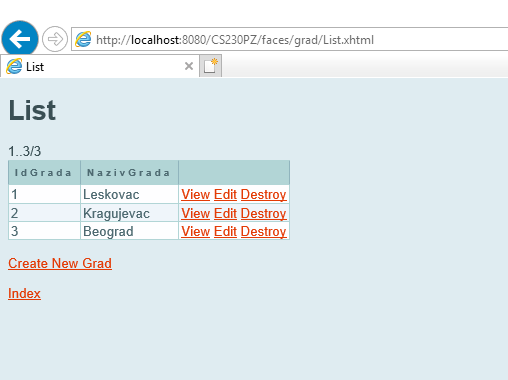


*Slika 5.2.6. – Prikaz kategorija*

### 5.2.7. Unos grada

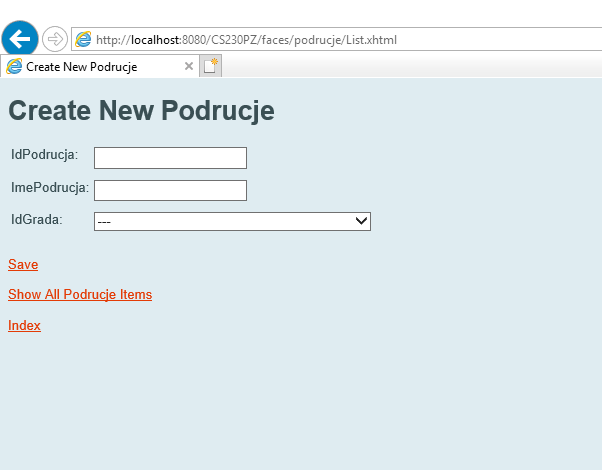


*Slika 5.2.7. – Unos grada za oglas*

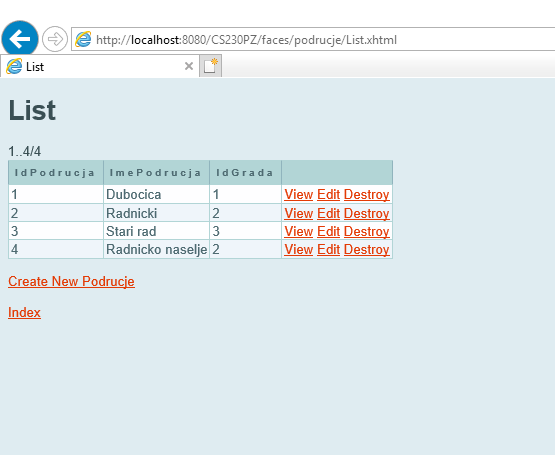
**

*Slika 5.2.8. – Izgled tabele grad*

### 5.2.8. Unos u tabelu područje



*Slika 5.2.9. – Unos u tabelu područje*

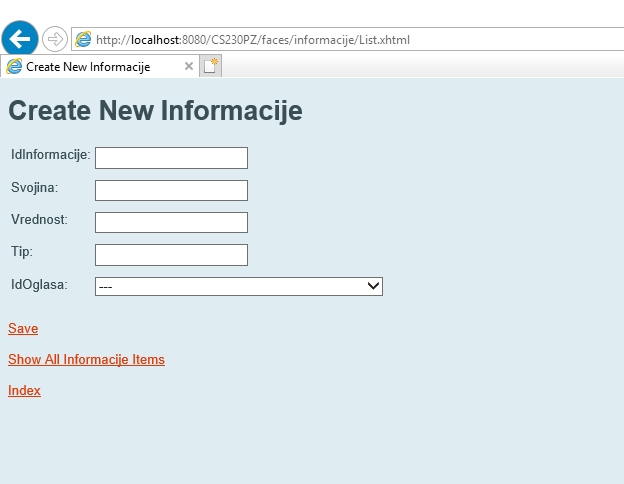
**

*Slika 5.2.10. – Izgled tabele područje*

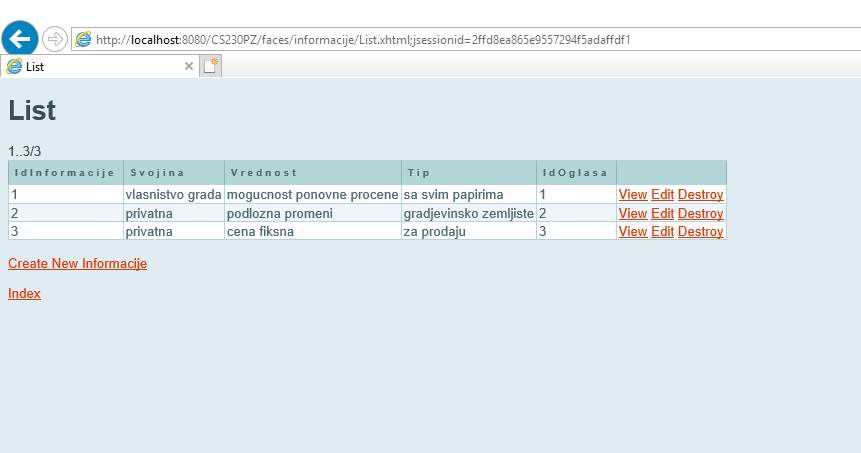


*Slika 5.2.11. – Povezivanje grada i područja*

### 5.2.9. Unos I izgled tabele informacije

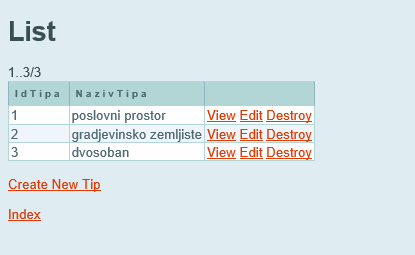


*Slika 5.2.12. – Unos u tabelu informacije*

**

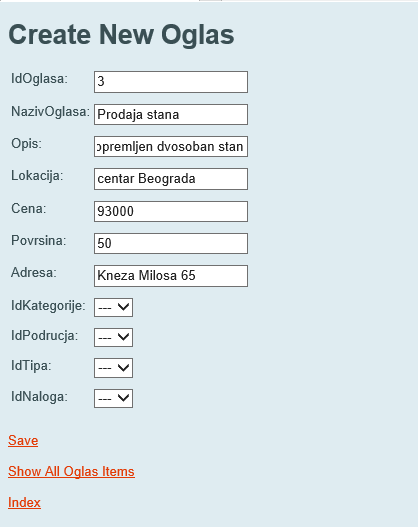
*Slika 5.2.13. – Izgled tabele informacije*

### 5.2.10. Izgled tabele tip



*Slika 5.2.14. – Izgled tabele tip*

### 5.2.11. Unos novog oglasa



*Slika 5.2.15. – Unos novog oglasa*

## 5.3. Testiranje sofrvera

Budući da je sam sistem rađen u programskom jeziku Java, i testiranje se sprovelo pomoću automatizovanih Junit testova koji su ugrađeni u NetBeans razvojno okruženje. S obzirom na to da Junit spada u jedinično testiranje, sam proces obuhvatio je testiranje svake klase koja ulazi u sastav sistema.

Testiranje je proces prilikom koga se u sistemu pronalaze greške. Testiranje sofrvera predstavlja vrlo bitno aktivnost prilikom samog razvoja softvera i od veliko je značaja za sigurnost celog tima, koji učestvuje u projektovanju i implementaciji sistema, da su napravili sistem koji su zapravo i hteli. Samo testiranje softvera je dobilo na značaju kada je vrednost softverskih proizboda počela da raste.

Testiranje sistema bi obuhvatilo testiranje svake komponente kao i kasnije celokupnog sistema. Kod testiranja interfejsa, testirali bi se pojedinačni objekti koji realizuju određene operacije navedene u interfejsu. Razlog testiranja interfejsa jeste često loše korišćenje interfejsa, nerazumevanja kao i zbog neusklađenosti brzina rada komponenti koje su u vezi posredstvom interfejsa.

Testiranje celokupnog sistema nam omogućava da proverimo da li su sve komponente kompatibilne odnosno da li ispavno međusobno komuniciraju. Prilikom ovog testiranja sve komponente se integrišu u celokupan sistem i nad njima se vriši testiranje.

## 5.4. Evolucija sistema

Kao što je već spomenuto, sam sistem predstavlja osnovnu verziju sistema kojom je omogućeno objavljivanje i pracenje nekretnina online.

Naravno, kada bude gotov ceo sistem i pušten u rad, čekali bi se odgovori i komentari klijenata koji bi ukazali na to šta bi trebalo dodati a šta korigovati kod sistema. Takođe, dobra ideja bi bila implementiranje ovog sistema i za mobilene uređaje, kako bi se korisnicima omogućio pristup sistemu sa bilo kog mesta, gde bi posedovali internet konekciju.

# 6. Zaključak

Projektni rad je obuhvatio dokumentaciju sistema za upravljanje pregledom nekretnina, kao i implementaciju osnovne funkcionalne verzije sistema. Različite vrste dijagrama koje su sadržane u ovoj dokumentaciji omogućavaju uvid u način na koji je sam sistem implementiran i sadrži delove koji tek treba da se uvedu u sistem.

Sama dokumentacija, zahvaljujući dijagramima i njihovim detaljnim opisima, omogućava razvojnom timu da održavaju i rekonstruišu sistem tako što bi dodali sve potrebne stavke koje još uvek nisu uvedene, a spomenute su u nekim dijagramima.

Dokumentacija je od velikog značaja za korisnike sistema jer je detaljno opisan način na koji sistem radi, ali i ljudima koji bi učestvovali u njegovom daljem razvoju jer se preko dijagrama lako mogu snaći sa samim načinom izrade i planovima daljeg razvoja sistema.