UposlIMe

Web aplikacija za studentsko omladinske poslove

Arhitekturni projekat

Verzija 1.0

Pregled izmena

|  |  |  |  |
| --- | --- | --- | --- |
| **Datum** | **Verzija** | **Opis** | **Autor** |
| 09.04.2023. | 1.0 | Inicijalna verzija | Pavle Kuzmanović  Luka Đorđević  Stefan Atanasković |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Sadržaj

1. Cilj dokumenta 5

2. Opseg dokumenta 5

3. Reference 5

4. Predstavljanje arhitekture 5

5. Ciljevi i ograničenja arhitekture 5

6. Pogled na slučajeve korišćenja 5

6.1 Dijagrami slučajeva korišćenja 6

6.2 Kratak opis slučajeva korišćenja 8

6.2.1 Prijavljivanje 8

6.2.2 Pretraživanje poslova 8

6.2.3 Filtriranje poslova 8

6.2.4 Prosledjivanje poslova zadruzi 8

6.2.5 Prihvatanje zahteva za posao 8

6.2.6 Odbijanje zahteva za posao 9

6.2.7 Kreiranje oglasa 9

6.2.8 Ažuriranje oglasa (zadruga) 9

6.2.9 Ažuriranje oglasa (poslodavac) 9

6.2.10 Povlačenje oglasa 9

6.2.11 Brisanje oglasa 9

6.2.12 Ocenjivanje 9

6.2.13 Apliciranje na oglas 10

6.2.14 Otkazivanje aplikacije na oglas 10

6.2.15 Prihvatanje zadrugara na posao 10

6.2.16 Kreiranje ugovora 10

6.2.17 Ažuriranje podataka korisničkih profila 10

7. Pogled na logičku arhitekturu sistema 10

7.1 Pregled arhitekture – organizacija paketa i podsistema u slojeve 11

7.1.1 Korisnički interfejs 11

7.1.2 Aplikaciona logika 11

7.1.3 Pristup podacima 11

7.1.4 HTML 11

7.1.5 JavaScript 11

7.1.6 MySQL 12

8. Pogled na procese 12

8.1 Procesi 12

8.1.1 Web čitač 12

8.1.2 Web server 12

8.1.3 Node.js 12

8.1.4 MySQL Server 13

9. Pogled na raspoređivanje sistema 13

9.1 Klijent 13

9.2 Web server 13

9.3 DBMS server 13

10. Pogled na implementaciju sistema 13

10.1 Model domena 13

10.2 Šema baze podataka 14

10.3 Komponente sistema 15

10.3.1 Komponente korisničkog interfejsa 15

10.3.2 Komponente aplikacione logike 15

10.3.3 Komponente za pristup podacima 16

11. Performanse 16

12. Kvalitet 16

Arhitekturni projekat

# Cilj dokumenta

Cilj ovog dokumenta je detaljni opis arhitekture UposliMe portala.

# Opseg dokumenta

Dokument se odnosi na UposliMe portal koji će biti razvijen od strane Dominus-a. UposliMe predstavlja naziv web aplikacije za studentsko omladinske poslove. Sistem će omogućiti zadrugarima efikasno prikazivanje i pretraživanje poslova, kao i apliciranje na iste, dok će poslodavcima pružiti mogućnost brzog i efikasnog pronalaska ispomoći.

# Reference

Spisak korišćene literature:

1. UposliMe – Predlog projekta, SWE-UposliMe-01, V1.0, 2023, Dominus.
2. UposliMe – Planirani raspored aktivnosti na projektu, V1.0, 2023, Dominus.
3. UposliMe – Plan realizacije projekta, V1.0, 2023, Dominus.
4. UposliMe – Vizija sistema, V1.0, 2023, Dominus.
5. UposliMe – Specifikacija zahteva, V3.0, 2023, Dominus.

# 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. UposliMe portal će biti implementiran kao Web aplikacija zasnovana na JavaScript skripting jeziku i MySQL bazi podataka [4].
2. Klijentski deo UposliMe portala će biti optimizovan za sledeće Web čitače: Internet Explorer 6.0 i noviji, Opera 8.0 i noviji, Firefox (Mozilla), Google Chrome, Safari itd [4].
3. Svi zahtevi u pogledu performansi dati u [5] moraju biti uzeti u obzir pri izboru arhitekture i razvoju sistema.
4. Poslovanje medju korisnicima portala i funkcionisanje istog, je omogućeno i kontrolisano od strane zakona propisanih od strane Republike Srbije.

# Pogled na slučajeve korišćenja

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

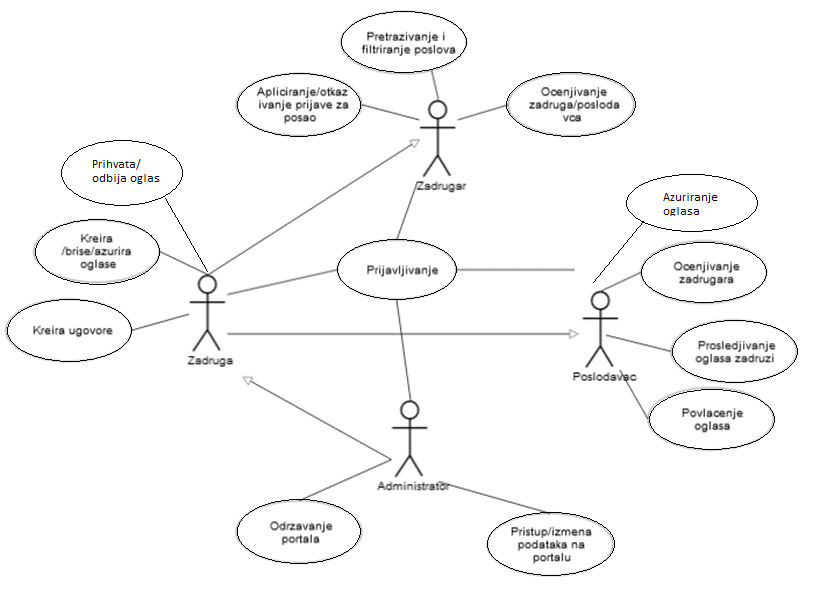
Slučajevi korišćenja UposliMe portala su:

* 6.1 Prijavljivanje
* 6.2 Pretraživanje poslova
* 6.3 Filtriranje poslova
* 6.4 Prosleđivanje oglasa zadruzi
* 6.5 Prihvatanje zahteva za posao
* 6.6 Odbijanje zahteva za posao
* 6.7 Kreiranje oglasa
* 6.8 Ažuriranje oglasa (zadruga)
* 6.9 Ažuriranje oglasa (poslodavac)
* 6.10 Povlačenje oglasa
* 6.11 Brisanje oglasa
* 6.12 Ocenjivanje
* 6.13 Apliciranje na oglas
* 6.14 Otkazivanje aplikacije na oglas
* 6.15 Prihvatanje zadrugara na posao
* 6.16 Kreiranje ugovora
* 6.17 Ažuriranje podataka korisničkih profila

Ove slučajevi korišćenja mogu da iniciraju zadrugari, zadruge, poslodavci ili administrator.

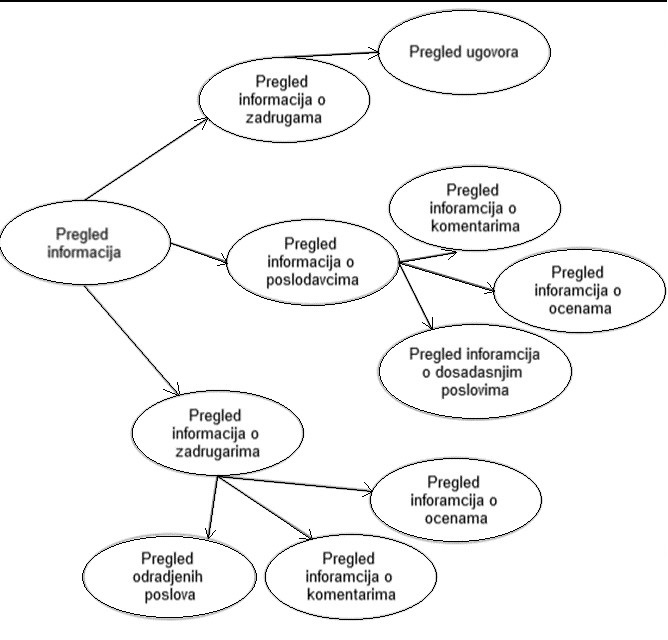
## Dijagrami slučajeva korišćenja

Osnovni UML dijagram koji prikazuje korisnike i slučajeve korišćenja UposliME portala prikazan je na sledećoj slici:

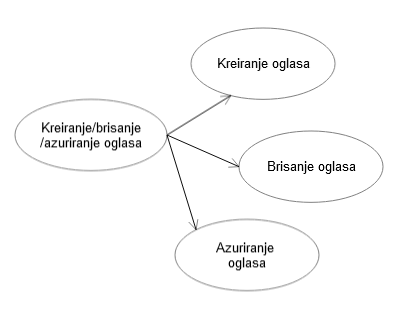


Slučajevi korišćenja pristup/izmena podataka na portalu, kreiranje/brisanje/azuriranje oglasa, apliciranje/otkazivanje prijave za posao obuhvataju složenije radnje koje se mogu razložiti dalje razložiti na pojedinačne slučajeve korišćenja.

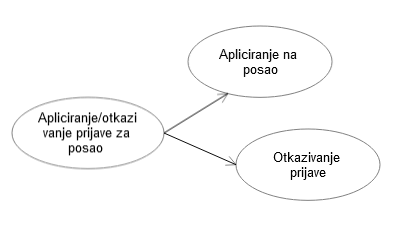
Detaljni UML dijagram za slučaj korišćenja *pristup/izmena podataka na portalu* je prikazan na sledećoj slici:



Detaljni UML dijagram za slučaj korišćenja *kreiranje/brisanje/azuriranje oglasa* je prikazan na sledećoj slici:



Detaljni UML dijagram za slučaj korišćenja *apliciranje/otkazivanje prijave za posao* je prikazan na sledećoj slici:



## Kratak opis slučajeva korišćenja

### Prijavljivanje

**Kratak opis:**

Prijavljivanje korisnika na portal u cilju pristupa specifičnim funkcijama koje zahtevaju autorizaciju

**Akteri koji iniciraju slučaj korišćenja:**

Zadrugar, Zadruga, Poslodavac, Administrator.

### Pretraživanje poslova

**Kratak opis:**

Prikaz stranice portala sa oglasima za poslove.

**Akteri koji iniciraju slučaj korišćenja:**

Zadrugar, Zadruga, Administrator.

### Filtriranje poslova

**Kratak opis:**

Prikaz stranice portala sa oglasima za poslove koji ispujnavaju zadate uslove.

**Akteri koji iniciraju slučaj korišćenja:**

Zadrugar, Zadruga, Administrator.

### Prosledjivanje oglasa zadruzi

**Kratak opis:**

Poslodavac nakon popunjavanje formulara za postavljanje oglasa, klikom na dugme prosledjuje oglas odabranoj zadruzi na pregled.

**Akteri koji iniciraju slučaj korišćenja:**

Poslodavac.

### Prihvatanje zahteva za posao

**Kratak opis:**

Zadruga prihvata zahtev za oglasavanje posla.

**Akteri koji iniciraju slučaj korišćenja:**

Zadruga, Poslodavac**.**

### Odbijanje zahteva za posao

**Kratak opis:**

Zadruga odbija zahtev za oglašavanje posla.

**Akteri koji iniciraju slučaj korišćenja:**

Zadruga, Poslodavac**.**

### Kreiranje oglasa

**Kratak opis:**

Prihvatanjem oglasa na reviziji, oglas se automatski objavljuje.

**Akteri koji iniciraju slučaj korišćenja:**

Zadruga.

### Ažuriranje oglasa

**Kratak opis:**

Prihvatanje/odbijanje zahteva od poslodavca za izmenu oglasa.

**Akteri koji iniciraju slučaj korišćenja:**

Zadruga.

### Ažuriranje oglasa

**Kratak opis:**

Poslodavac prosledjuje zahtev za izmenu oglasa.

**Akteri koji iniciraju slučaj korišćenja:**

Poslodavac.

### Povlačenje oglasa

**Kratak opis:**

Poslodavac odabirom odgovarajuceg dugmeta uklanja oglas.

**Akteri koji iniciraju slučaj korišćenja:**

Poslodavac.

### Brisanje oglasa

**Kratak opis:**

Zadruga odabirom odgovarajuceg dugmeta uklanja oglas.

**Akteri koji iniciraju slučaj korišćenja:**

Zadruga.

### Ocenjivanje

**Kratak opis:**

Korisnici putem odgovarajuceg interfejsa medjusobno ostavljaju komentare i daje ocene.

**Akteri koji iniciraju slučaj korišćenja:**

Zadrugar, Poslodavac.

### Apliciranje na oglas

**Kratak opis:**

Zadrugar aplicira na odabrani oglas.

**Akteri koji iniciraju slučaj korišćenja:**

Zadrugar.

### Otkazivajne aplikacija na oglas

**Kratak opis:**

Zadrugar otkazuje aplikaciju na odabrani oglas.

**Akteri koji iniciraju slučaj korišćenja:**

Zadrugar.

### Prihvatanje zadrugara na posao

**Kratak opis:**

Poslodavac iz liste apliciranih zadrugara bira one koje zeli da prihvati na odabrani posao.

**Akteri koji iniciraju slučaj korišćenja:**

Poslodavac.

### Kreiranje ugovora

**Kratak opis:**

Odabrana zadruga po isteku oglasa i po odabiru zeljenih radnika od strane poslodavca, dobija potrebne informacije za kreiranje ugovora.

**Akteri koji iniciraju slučaj korišćenja:**

Zadruga.

### Ažuriranje podataka korisničkih profila

**Kratak opis:**

Korisnik odlaskom na svoj profil može ažurirati postojeće podatke.

**Akteri koji iniciraju slučaj korišćenja:**

Zadruga, zadrugar, poslodavac.

# 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.

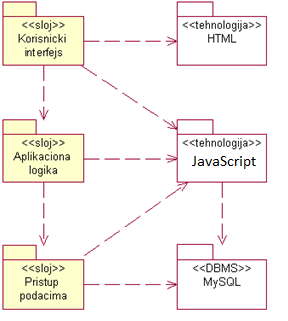
Logički pogled na UposliMe portal obuhvata 3 glavna paketa: Korisnički interfejs, Aplikaciona logika, Pristup podacima.

Paket *Korisnički interfejs* sadrži Web stranice, JavaScript skripte 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 JavaScript skripte zadužene za realizaciju funkcionalnosti specifičnih za domen sistema koji se razvija.

Paket *Pristup podacima* sadrži JavaScript skripte koje predstavljaju 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 sadržane sve HTML, multimedijalni sadržaji i JavaScript skripte koje generišu HTML stranice preko kojih korisnici komuniciraju sa sistemom.

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

### Aplikaciona logika

sloj

Sloj aplikacione logike je srednji sloj u troslojnoj arhitekturi UposliMe portala. Sadrži JavaScript skripte koje realizuju funkcionalnost karakterističnu za domen primene portala i uspostavljaju vezu između korisničkog interfejsa i sloja za pristup podacima.

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

### Pristup podacima

sloj

Sloj za pristup podacima se nalazi na dnu troslojne arhitekture i sadrži JavaScript skripte zadužene za pribavljanje, dodavanje i ažuriranje podataka koji se čuvaju u MySQL bazi podataka.

Ovaj sloj ne zavisi od drugih slojeva, ali je zavisan od paketa JavaScript i MySQL baza 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.

### JavaScript

tehnologija

Tehnologija JavaScript-a 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.

### MySQL

DBMS

MySQL predstavlja sistem za upravljanje bazama podataka koji će se koristiti za realizaciju UposliMe portala.

# Pogled na procese

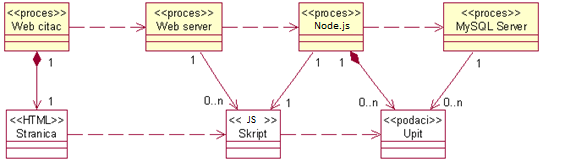
U ovom odeljku je sadržan pogled na procesnu arhitekturu sistema. Ovaj opis treba da sadrži specifikaciju različitih zadataka (procesa i niti) uključenih u rad sistema. Takođe je potrebno dati dijagrame koji pokazuju njihovu interakciju i konfiguraciju. Dodela objekata i klasa na određene zadatke takođe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

Web aplikacije zasnovane na JavaScript-u imaju relativno jednostavan procesni model koji je u potpunosti pod kontrolom Web servera. Sa stanovišta projektanta JavaScript Web aplikacije nije potrebno voditi računa o načinu rada Web servera i načinu izvršavanja skripti.

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

## Procesi

Na sledećem UML dijagramu klasa prikazani su procesi koji učestvuju u izvršenju UposliMe portala. Dijagram je opšteg tipa i može se primeniti na bilo koju Web aplikaciju zasnovanu na JavaScript-u i MySQL bazi podataka.



### Web čitač

Web čitač je proces koji izvršava funkcionalnost aplikacije za prikaz HTML stranica dobijenih od nekog Web servera. U najopštijem slučaju Web čitač u jednom trenutku može da prikazuje samo jednu HTML stranicu.

Web čitač zavisi od Web servera koji generiše i vraća odgovarajuću HTML stranicu na zahtev.

### Web server

Web server je proces koji izvršava funkcionalnost opsluživanja zahteva prispelih sa više Web čitača. Ukoliko je zahtevana stranica JavaScript skript, Web server inicira izvršenje Node.js procesa koji obrađuje odgovarajući skript i generiše sadržaj koji se vraća čitaču. Web server može paralelno da inicira veći broj Node.js procesa.

### Node.js

Node.js proces obavlja posao obrade zadatog JavaScript skripta i generiše odgovarajući tekstualni sadržaj koji Web server šalje Web čitaču. Za izvršenje JavaScript skripta ovaj proces može da zahteva usluge MySQL servera-a. Komunikacija između Node.js procesa i MySQL servera se obavlja preko prosleđivanja upita i vraćanja rezultat.

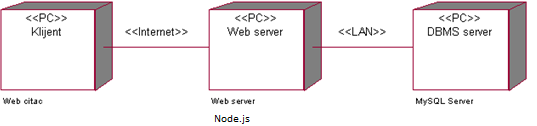
### MySQL Server

MySQL Server je proces koji izvršava funkcionalnost MySQL 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 UposliMe portala.



## Klijent

Pristup UposliMe portal 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 kome se izvršava Web server opslužuje više klijenata koji pristupaju preko Interneta. Pored osnovnog procesa koji realizuje funkcionalnost Web servera, na ovom računaru mogu da se izvršavaju i procesi Node.js koji vrše obradu zadatih JavaScript skripti. U najopštioj konfiguraciji DBMS se izvršava na posebnoj mašini koja je sa Web serverom u lokalnoj mreži (LAN).

## DBMS server

DBMS server je računar na kome se izvršava MySQL 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).

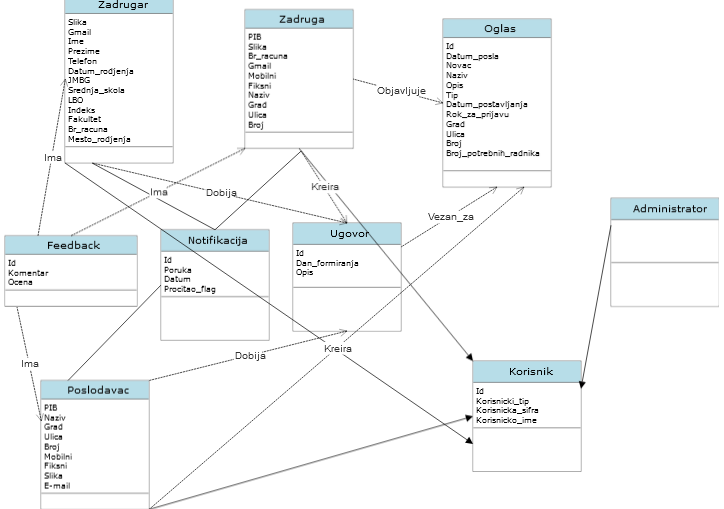
# Pogled na implementaciju sistema

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

## Model domena

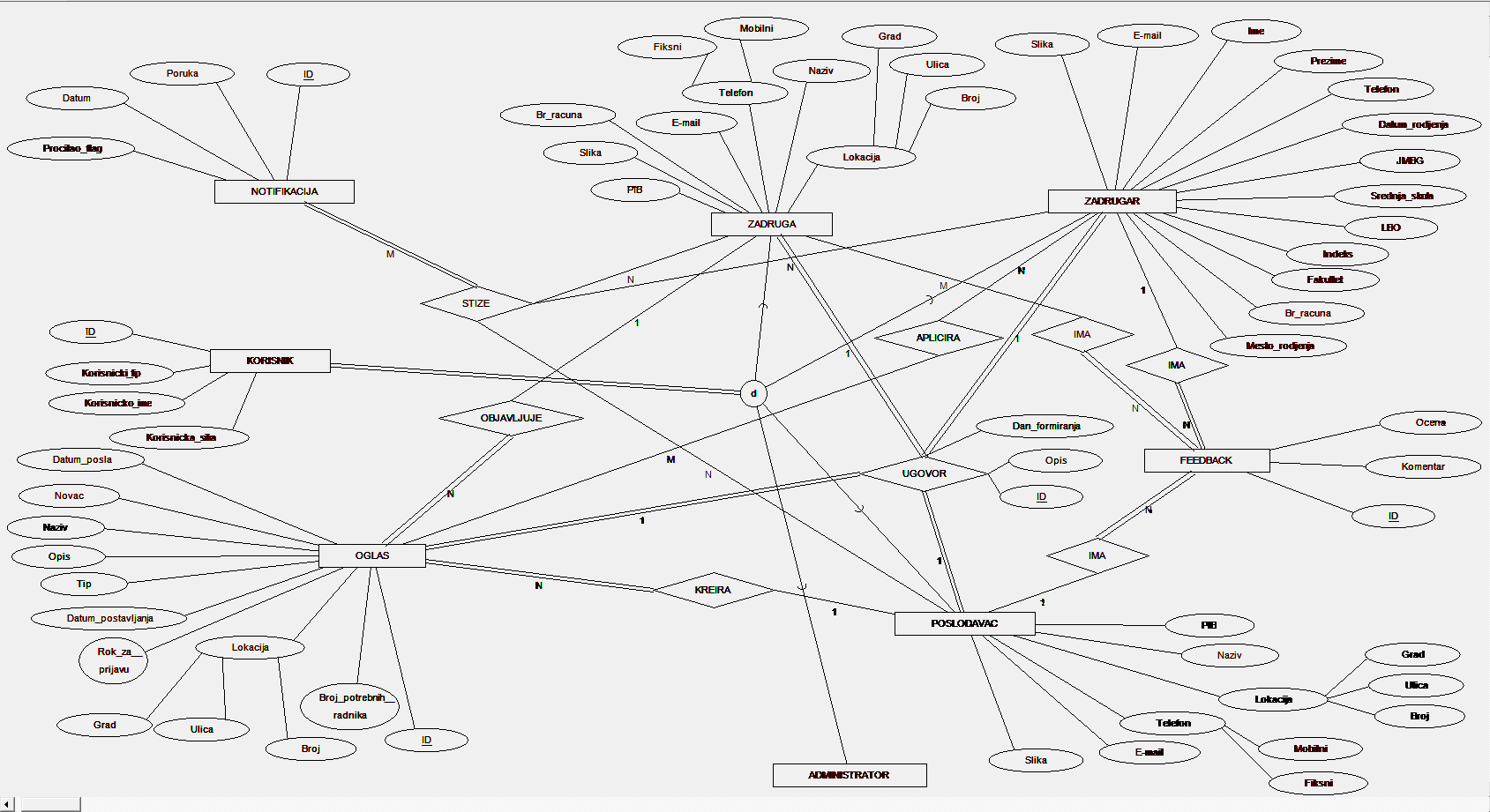
Model domena za koji se UposliMe portal projektuje je ilustrovan UML dijagramom klasa. U njemu su prikazane domenske klase, neki od njihovih atributa, kao i veze koje se mogu identifikovati između njih.

Model domena predstavlja osnovu za projektovanje baze podataka, ali i identifikaciju nekih od komponenti (JavaScript) koje će biti implementirane.



## Šema baze podataka

Detaljna šema baze podataka je prikazana na sledećem EER dijagramu. Dijagram je kreiran korišćenjem SharpEER softvera.



## Komponente sistema

*NAPOMENA: UposliMe projekat nije kompletno implementiran, tj. implementiran je na nivou arhitekturnog prototipa. Zbog ovoga sledeću sekciju trebate shvatiti kao ilustraciju kako bi dokument trebao da izgleda. Opis komponenti sistema za vaše projekte mora biti kompletan! Procena složenosti vašeg rešenja će zavisiti od opisa sistema koji ovde date.*

Komponente sistema UposliMe portala su JavaScript skripti čiji će pregled biti dat po arhitekturnim slojevima. Za ilustraciju će biti korišćeni UML dijagrami komponenti, ali i dijagrami klasa. U slučajevima gde je JavaScript skript prikazan kao klasa atributi predstavljaju ulazne podatke koji se uzimaju iz GET ili POST dela HTTP poruke, dok metodi predstavljaju funkcije definisane u okviru skripta.

### Komponente korisničkog interfejsa

Dizajn korisničkog interfejsa je obuhvaćen dvema komponentama:

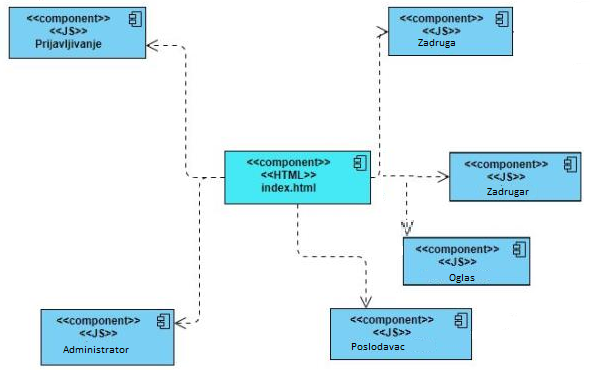


Komponenta **index.html** je implementira stranicu portala čiji sadržaj može da varira od parametra koji joj se proslede u zahtevu.

Komponenta **main.css** 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 **index.html** komponente korisničkog interfejsa. Na taj način zadržavaju sva podešavanja stila definisana u ovom skriptu. Na sledećem dijagramu su prikazane komponente ovog sloja i njihove međusobne zavisnosti:



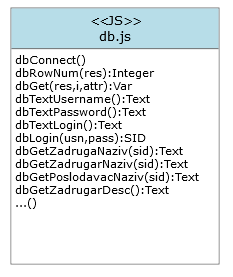
... opis svih komponenti sa dijagrama sa detaljnim dijagramima po potrebi.

### Komponente za pristup podacima

Pristup bazi podataka je u potpunosti zatvoren u funkcije koje su definisane u okviru JavaScript skripta **db.js**. Pomenuti skript se uključuje na početku index.php-a, tako da su sve funkcije za pristup podacima dostupne svim komponentama.



Na sledećem UML dijagramu klasa pobrojane su funkcije za pristup podacima iz baze:



Navedene funkcije obavljaju sledeće zadatke:

* dbConnect – povezivanje na bazu podataka, poziva se u samom skriptu
* dbRowNum – vraća broj slogova sadržanih u rezultatu upita
* dbGet – vraća vrednost polja iz rezultata upita
* dbText... – vraća vrednost pojedinih konstanti
* dbLogin – loguje korisnika i vraća SID
* dbGetZadrugaNaziv – vraća naziv zadruge
* dbGetZadrugarNaziv – vraća naziv zadrugara
* dbGetZadrugarDesc – vraća informacije o zadrugaru
* ... – ostale funkcije

# 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 do 100 simultanih pristupa korisnika portalu.
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. UposliMe portal 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.