PetBazaar

Website za prodaju kućnih ljubimaca i opreme

Detaljni arhitekturni projekat

Verzija 1.0

Pregled izmena

|  |  |  |  |
| --- | --- | --- | --- |
| **Datum** | **Verzija** | **Opis** | **Autor** |
| 25.12.2024. godine | 1.0 | Inicijalna verzija | Tijana |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

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 7

6.2.1 Pretraživanje modela 7

6.2.2 Filtriranje modela 7

6.2.3 Dodavanje proizvoda u korpu 7

6.2.4 Ocenjivanje 7

6.2.5 Pregled Korisnika 8

6.2.6 Prijavljivanje 8

6.2.7 Ažuriranje podataka o sebi 8

6.2.8 Kreiranje novog naloga 8

6.2.9 Brisanje postojećeg naloga 8

6.2.10 Unos modela 8

6.2.11 Brisanje postojećeg modela 8

6.2.12 Ažuriranje podataka o modelu 8

7. Pogled na logičku arhitekturu sistema 8

7.1 Pregled arhitekture – organizacija paketa i podsistema u slojeve 9

7.1.1 Korisnički interfejs 9

7.1.2 Aplikaciona logika 9

7.1.3 Pristup podacima 9

7.1.4 React 9

7.1.5 Node js 9

7.1.6 MongoDB 9

8. Pogled na procese 10

8.1 Procesi 10

8.1.1 Web čitač 10

8.1.2 Web server 10

8.1.3 MongoDB Server 10

9. Pogled na raspoređivanje sistema 10

9.1 Klijent 11

9.2 Web server 11

9.3 Database server 11

10. Pogled na implementaciju sistema 11

10.1 Model domena 11

10.2 Šema baze podataka 12

10.3 Komponente sistema 12

10.3.1 Komponente korisničkog interfejsa 13

10.3.2 Komponente aplikacione logike 14

10.3.3 Komponente za pristup podacima 14

11. Performanse 15

12. Kvalitet 15

Detaljni arhitekturni projekat

# Cilj dokumenta

Cilj ovog dokumenta je detaljni opis arhitekture PetBazaar portala.

# Opseg dokumenta

Dokument se odnosi na PetBazaar portal koji će biti razvijen od strane PetShopTeam-a. PetBazaar predstavlja skraćenicu za Website za prodaju ljubimaca I opreme. Namena sistema je efikasno prezentovanje, kreiranje i održavanje sadržaja vezanih za ljubimce I prodaju opreme.

# Reference

Spisak korišćene literature:

1. PetBazaar – Predlog projekta, PetShopTeam-PetBazaar-01, V1.0, 2024, PetShopTeam.
2. PetBazaar – Planirani raspored aktivnosti na projektu, V1.0, 2024, PetShopTeam.
3. PetBazaar – Plan realizacije projekta, V1.0, 2024, PetShopTeam.
4. PetBazaar – Vizija sistema, V1.0, 2024, PetShopTeam.
5. PetBazaar – Specifikacija zahteva, V1.0, 2024, PetShopTeam.

# Predstavljanje arhitekture

Arhitektura sistema u dokumentu je prikazana kao serija pogleda na sistem: pogled na slučajeve korišćenja, 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. PetBazaar portal će biti implementiran kao Web aplikacija zasnovana na PHP skripting jeziku i MySQL bazi podataka [4].
2. Klijentski deo PetBazaar portala 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.
4. Klasifikaciju publikacija treba standardizovati sa postojećom tipologijom propisanom od strane Ministarstva za nauku.

# Pogled na slučajeve korišćenja

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

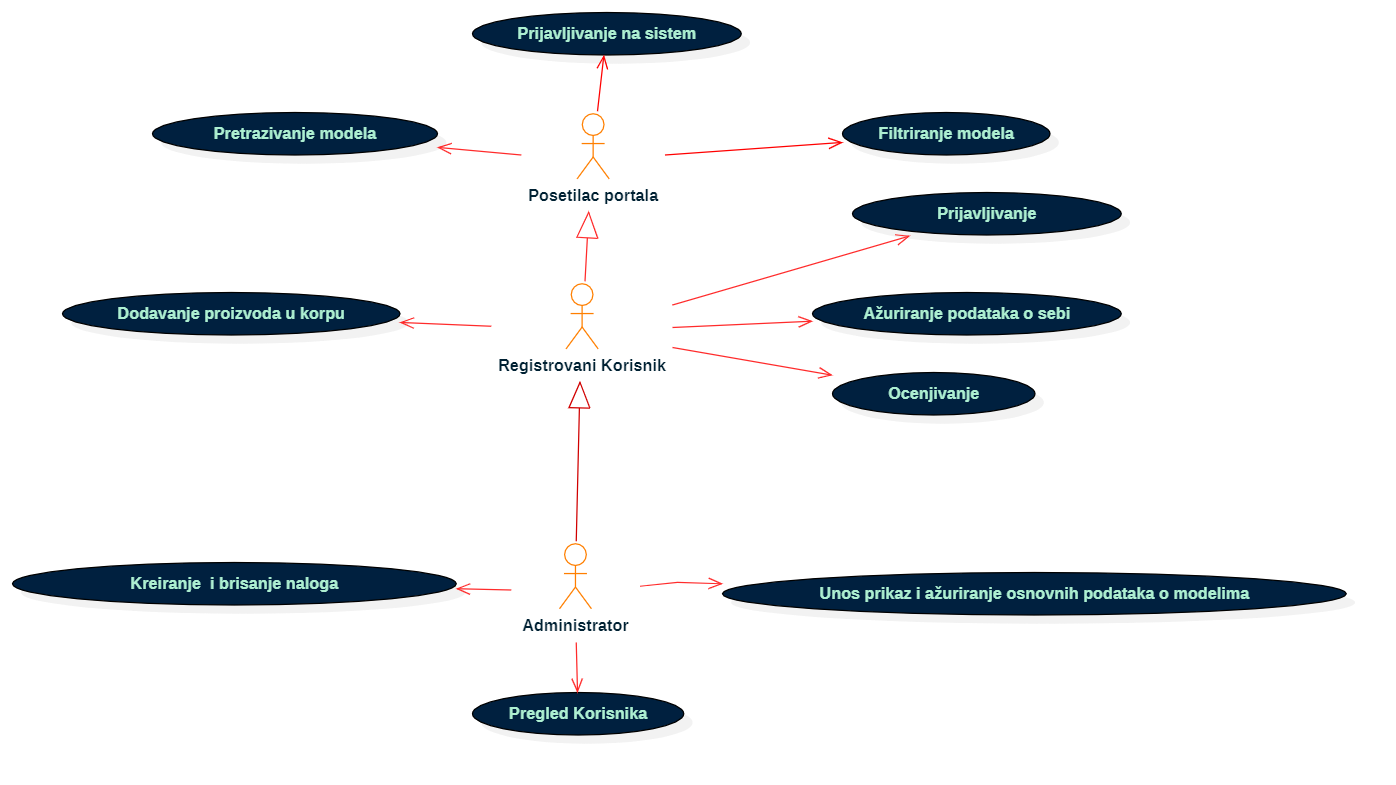
Slučajevi korišćenja PetBazaar portala su:

* *Pregled informacija*
  + Pregled osnovnih podataka o laboratoriji
  + *Pregled podataka o članovima*
    - Pregled spiska članova
    - Pregled podataka o određenom članu laboratorije
  + *Pregled podataka o publikacijama*
    - Pregled publikacija po autoru
    - Pregled publikacija po tipu
    - Pregled publikacija po godini
  + *Pregled podataka o projektima*
    - Pregled spiska projekata
    - Pregled podataka o određenom projektu
* Prijavljivanje
* Ažuriranje podataka o članu
* *Ažuriranje podataka o publikacijama*
  + Dodavanje nove publikacije
  + Brisanje postojeće publikacije
* Ažuriranje osnovnih podataka o laboratoriji
* *Kreiranje brisanje i arhiviranje članova*
  + Kreiranje novog člana
  + Brisanje postojećeg člana
  + Arhiviranje postojećeg člana
* Kreiranje projekta i postavljanje vođe
* Ažuriranje podataka o projektu

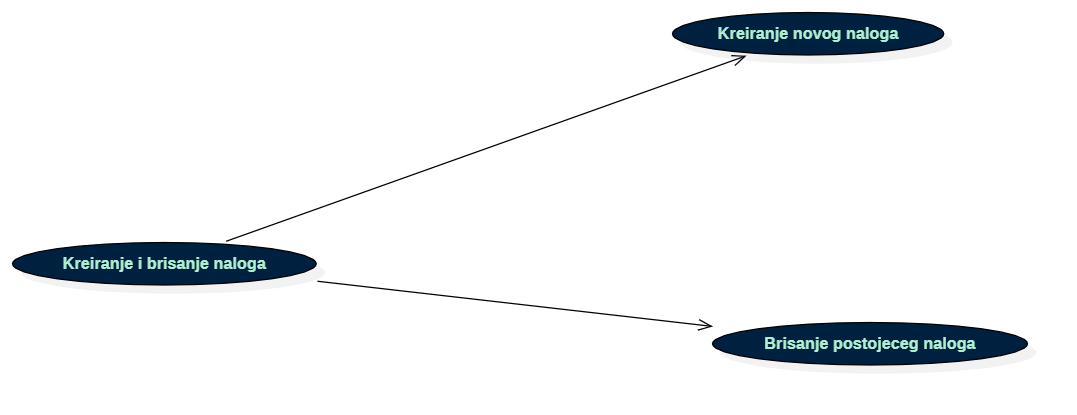
Ove slučajevi korišćenja mogu da iniciraju posetilac portala, član laboratorije, šef laboratorije, vođa projekta ili administrator.

## Dijagrami slučajeva korišćenja

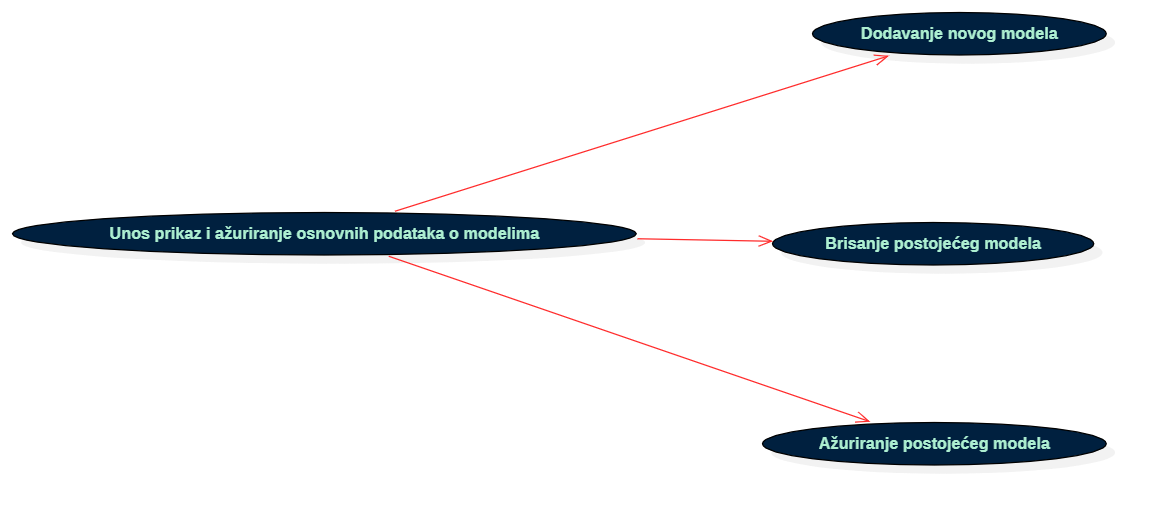
Osnovni UML dijagram koji prikazuje korisnike i slučajeve korišćenja PetBazaar portala prikazan je na sledećoj slici:



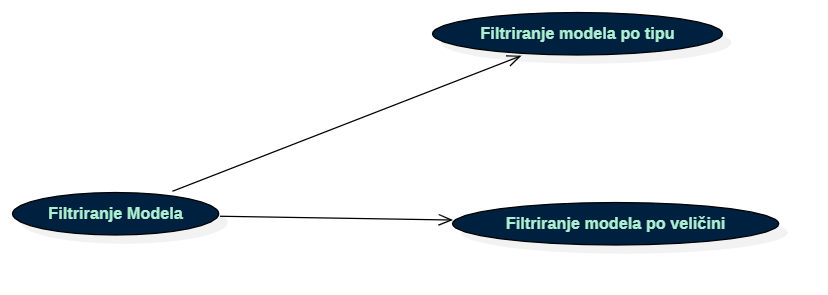
Detaljni UML dijagram za slučaj korišćenja *kreiranje i brisanje naloga* je prikazan na sledećoj slici:



Detaljni UML dijagram za slučaj korišćenja *prikaz i ažuriranje podataka o modelima*  je prikazan na sledećoj slici:



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



## Kratak opis slučajeva korišćenja

### Pretraživanje modela

Kratak opis: Mogućnost pretraživanja po rayličitim tipovima modela.

Akteri koji iniciraju slučaj korišćenja:Posetilac portala,Registrovani korisnik i Administrator.

### Filtriranje modela po tipu

Kratak opis: Filtriranje proizvoda na osnovu tipa.

Akteri koji iniciraju slučaj korišćenja: Posetilac portala, Registrovani korisnik,Administrator.

### Filtriranje modela po veličini

Kratak opis: Filtriranje proizvoda na osnovu veličine.

Akteri koji iniciraju slučaj korišćenja: Posetilac portala, Registrovani korisnik,Administrator.

### Dodavanje proizvoda u korpu

Kratak opis: Dodavanje prozvoda koje korisnik želi da poruči.

Akteri koji iniciraju slučaj korišćenja: Registrovani korisnik,Administrator

### Ocenjivanje

Kratak opis: Ocenjivanje dostupnih modela..

Akteri koji iniciraju slučaj korišćenja: Registrovani korisnik,Administrator.

### Pregled Korisnika

Kratak opis: Prikaz stranice sa informacijama o određenom korisniku.

Akteri koji iniciraju slučaj korišćenja: Administrator.

### Prijavljivanje

Kratak opis: Prijavljivanje korisinika na portalu u cilju pristupa specifičnim funkcijama koje zahtevaju autorizaciju.

Akteri koji iniciraju slučaj korišćenja: Registrovani korisnik, Administrator.

### Ažuriranje podataka o sebi

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

Akteri koji iniciraju slučaj korišćenja: Registrovani korisnik, Administrator.

### Kreiranje novog naloga

Kratak opis: Kreiranje korisničkog naloga za novog korisnika.

Akteri koji iniciraju slučaj korišćenja: Posetilac portala, Administrator.

### Brisanje postojećeg naloga

Kratak opis: Brisanje korisničkog naloga i podataka za postojećeg korisničkog naloga.

Akteri koji iniciraju slučaj korišćenja: Šef laboratorije, Administrator.

### Unos modela

Kratak opis: Unos novog modela od strane administratora.

Akteri koji iniciraju slučaj korišćenja:Administrator.

### Brisanje postojećeg modela

Kratak opis: Brisanje postojećeg modela od strane administratora.

Akteri koji iniciraju slučaj korišćenja:Administrator.

### Ažuriranje podataka o modelu

Kratak opis: Izmena podataka o modelu od strane administratora.

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.

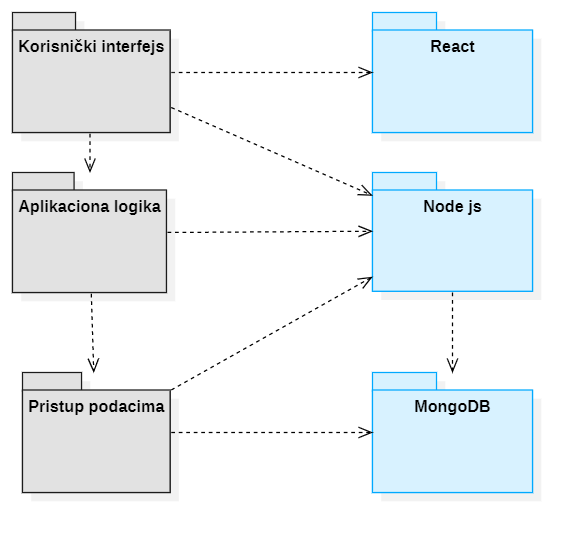
Logički pogled na Actiwear portal obuhvata 3 glavna paketa: Korisnički interfejs, Aplikaciona logika, Pristup podacima.

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

Paket *Pristup podacima* sadrži PHP 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 PHP skripte koje generišu HTML stranice preko kojih korisnici komuniciraju sa sistemom.

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

### Aplikaciona logika

sloj

Sloj aplikacione logike je srednji sloj u troslojnoj arhitekturi PetBazaar portala. Sadrži PHP 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 PHP paketa.

### Pristup podacima

sloj

Sloj za pristup podacima se nalazi na dnu troslojne arhitekture i sadrži PHP 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 PHP i MySQL baza podataka.

### React

tehnologija

Tehnologija React-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.

### Node js

tehnologija

Tehnologija Node js-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.

### MongoDB

DBMS

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

# Pogled na procese

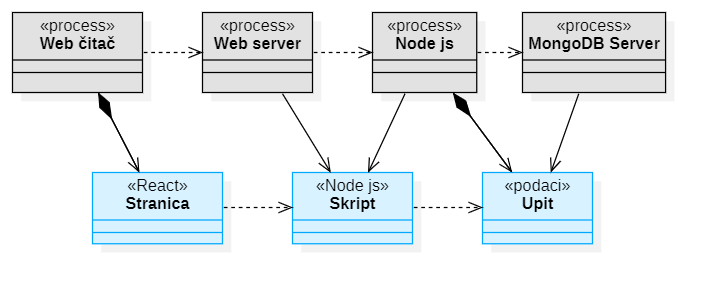
U ovom odeljku je sadržan pogled na procesnu arhitekturu sistema. Ovaj opis treba da sadrži specifikaciju različitih zadataka (procesa i niti) uključenih u rad sistema. Takođe je potrebno dati dijagrame koji pokazuju njihovu interakciju i konfiguraciju. Dodela objekata i klasa na određene zadatke takođe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

Web aplikacije zasnovane na React-u imaju relativno jednostavan procesni model koji je u potpunosti pod kontrolom Web servera..

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

## Procesi

Na sledećem UML dijagramu klasa prikazani su procesi koji učestvuju u izvršenju PetBazaar portala. Dijagram je opšteg tipa i može se primeniti na bilo koju Web aplikaciju zasnovanu na React,Express i Mongoose 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.

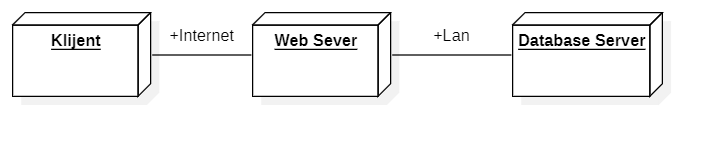
### MongoDB Server

MongoDB 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 PetBazaar portala.



## Klijent

Pristup PetBazaar portalu 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. U najopštioj konfiguraciji DBMS se izvršava na posebnoj mašini koja je sa Web serverom u lokalnoj mreži (LAN).

## Database server

Database 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).

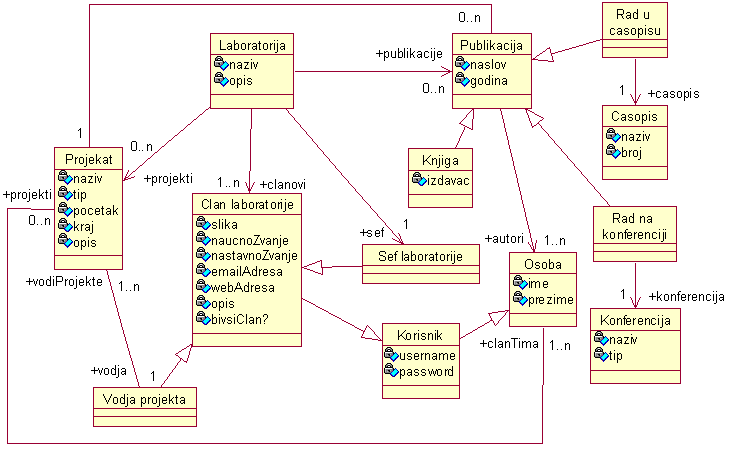
# Pogled na implementaciju sistema

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

## Model domena

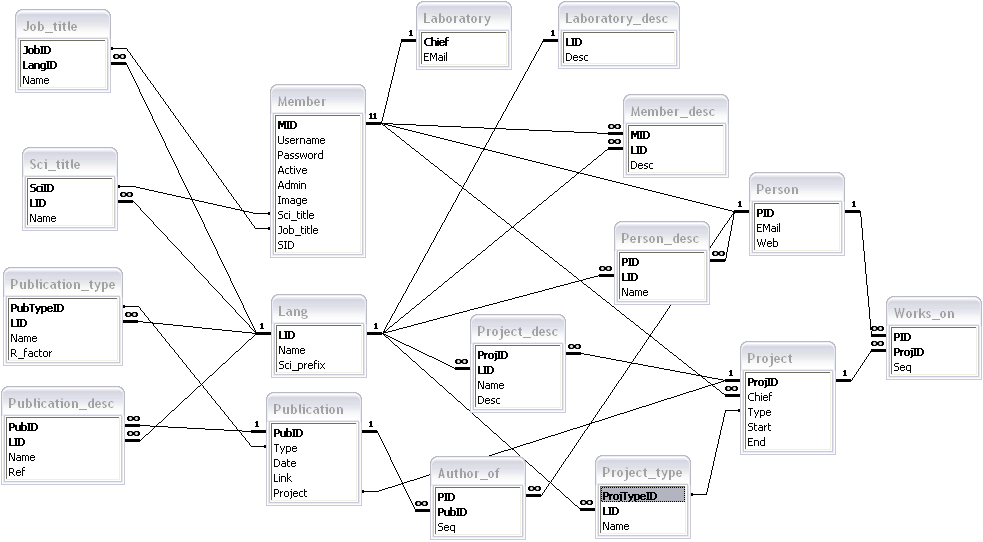
Model domena za koji se PetBazaar portal projektuje je ilustrovan UML dijagramom klasa. U njemu su prikazane domenske klase, neki od njihovih atributa, kao i veze koje se mogu identifikovati između njih.

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



## Šema baze podataka

Detaljna šema baze podataka je prikazana na sledećem dijagramu. Baza podataka i dijagram su kreirani korišćenjem *MS Access*-a, dok je migracija na MySQL obavljena pomoću *MySQL Migration Toolkit*-a.

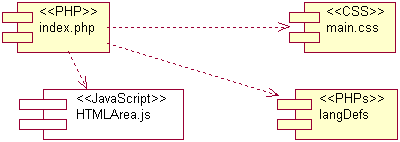


## Komponente sistema

Komponente sistema PetBazaar portala su PHP 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 PHP 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.php** 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.

Komponenta **langDefs** predstavlja grupu PHP skriptova kojima se definišu konstante na različitim jezicima. Konvencija je da se ovi skriptovi nazivaju po ID-u jezika. Primer skripta koji definiše konstante za srpski jezik (ćirilica) je sledeće sadržine:

<?php

$L\_TITLE = "Лабораторија за рачунарску графику и ГИС";

$L\_USERNAME = "Корисничко име:";

$L\_PASSWORD = "Лозинка:";

$L\_LOGIN = "Пријави се";

$L\_LOGOUT = "Одјави се";

$L\_EDITSTART = "Укључи уређивање";

$L\_EDITEND = "Искључи уређивање";

$L\_SAVECHANGES = "Сачувајте измене";

$L\_HOME = "О лабораторији";

$L\_MEMBERS = "Чланови";

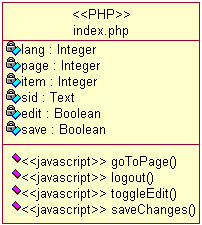
$L\_PUBLICATIONS = "Публикације";

$L\_PROJECTS = "Пројекти";

?>

Komponenta **HTMLArea.js** predstavlja OpenSource HTML editor koji se koristi za ažuriranje složenijih opisa.

Parametri koji utiču na izbor i jezik za prikaz stranice ilustrovani su sledećim dijagramom klasa:

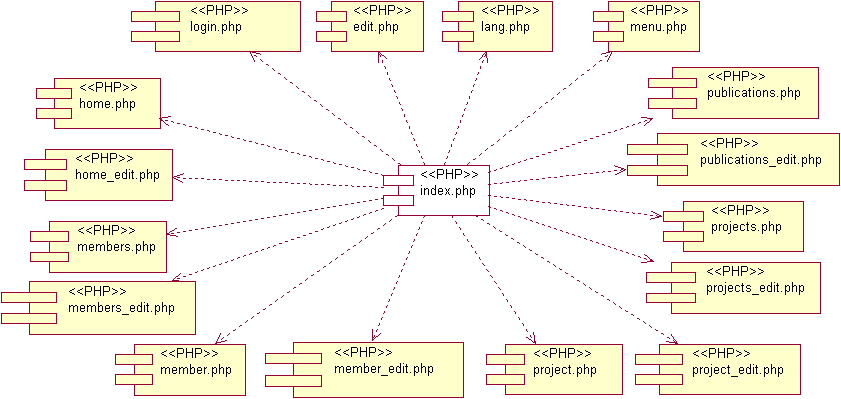


Značenje atributa je sledeće:

* lang – identifikator jezika (JID iz tabele Jezik)
* page – interni identifikator stranice (o laboratoriji, članovi, publikacije, projekti)
* item – identifikator stavke koja se detaljno prikazuje na stranici (određeni član ili projekat)
* sid – identifikator sesije kada je korisnik ulogovan
* edit – definiše da li je uključeno uređivanje
* save – definiše da li treba izvršiti snimanje podataka sa tekuće stranice/stavke

### Komponente aplikacione logike

Komponente koje realizuju domen problema se uključuju isključivo preko **index.php** 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:

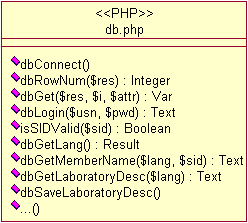


### Komponente za pristup podacima

Pristup bazi podataka je u potpunosti zatvoren u funkcije koje su definisane u okviru PHP skripta **db.php**. 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
* dbLogin – loguje korisnika i vraća SID
* isSIDValid – vrši proveru validnosti SID-a
* dbGetLang – vraća rezultat upita nad jezicima definisanim u bazi podataka
* dbGetMemberName – vraća ime člana po jeziku i SID-u
* dbGetLaboratoryDesc – vraća opis laboratorije za zadati jezik
* dbSaveLaboratoryDesc – čita i snima u bazu podataka opis laboratorije na svim jezicima
* ... – 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 1000 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. PetBazaar 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.