LanTern

Learn for Term

Arhitekturni projekat

Verzija 1.1

Pregled izmena

|  |  |  |  |
| --- | --- | --- | --- |
| **Datum** | **Verzija** | **Opis** | **Autor** |
| 09.05.2021. | 1.0 | Inicijalna verzija. | Dušan, Jovana, Snežana |
| 13.05.2021. | 1.1 | Prerađena verzija. | Snežana |
|  |  |  |  |
|  |  |  |  |

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 Registrovanje na platformu 8

6.2.2 Prijavljivanje na platformu 8

6.2.3 Pregled oglasa po gradu 8

6.2.4 Pregled oglasa sortiranih po ceni časa 8

6.2.5 Pregled oglasa po kategoriji 8

6.2.6 Čuvanje oglasa 9

6.2.7 Uklanjanje oglasa iz sačuvanih oglasa 9

6.2.8 Kontaktiranje autora oglasa 9

6.2.9 Kontaktiranje administratora 9

6.2.10 Pregled profila korisnika 9

6.2.11 Ažuriranje podataka o korisniku 9

6.2.12 Postavljanje novog oglasa 9

6.2.13 Ažuriranje postojećeg oglasa 9

6.2.14 Brisanje postojećeg oglasa 9

6.2.15 Dodavanje novih kategorija 9

6.2.16 Brisanje postojećih kategorija 9

6.2.17 Dodavanje novih gradova 9

6.2.18 Brisanje postojećih gradova 9

6.2.19 Brisanje oglasa 10

6.2.20 Brisanje korisničkog naloga 10

7. Pogled na logičku arhitekturu sistema 10

7.1 Pregled arhitekture – organizacija paketa i podsistema u slojeve 10

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 CSS 11

7.1.6 JavaScript 11

7.1.7 PHP 11

7.1.8 MySQL 11

8. Pogled na procese 11

8.1 Procesi 12

8.1.1 Web browser 12

8.1.2 Web server 12

8.1.3 MySQL Server 12

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 15

10.3 Komponente sistema 15

10.3.1 Komponente korisničkog interfejsa 15

10.3.2 Komponente aplikacione logike 17

10.3.3 Komponente za pristup podacima 18

11. Performanse 19

12. Kvalitet 19

Arhitekturni projekat

# Cilj dokumenta

Cilj ovog dokumenta je detaljni opis arhitekture **LanTern – Learn for Term** *Web* platforme.

# Opseg dokumenta

Dokument se odnosi na **LanTern** *Web* platformu koja će biti razvijena od strane **3STeam**-a. **LanTern** predstavlja skraćenicu za **Learn for Term**. Namena sistema je efikasno pronalaženje predavača za privatne časove.

# Reference

Spisak korišćene literature:

1. LanTern – Predlog projekta, 3S-LanTern-01, V1.5, 2021, 3STeam
2. LanTern – Vizija sistema, 3S-LanTern-02, V1.5, 2021, 3STeam
3. LanTern – Plan realizacije projekta, 3S-LanTern-03, V1.3, 2021, 3STeam
4. LanTern – Planirani raspored aktivnosti na projektu, V1.0, 2021, 3STeam
5. LanTern – Specifikacija zahteva, V1.1, 2021, 3STeam

# 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 raspoređivanje sistema i pogled na implementaciju sistema. 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. **LanTern** *Web* platforma će biti implementirana kao *Web* aplikacija, koja je na serverskoj strani zasnovana na **PHP** skripting jeziku, a na klijentskoj će biti iskorišćen template (skinut sa Web sajta https://elements.envato.com/web-templates) koji je kreiran pomoću tehnologija **HTML, CSS** i **JavaScript**. Pored toga, koristi **MySQL** bazu podataka.
2. Klijentski deo **LanTern** *Web* platforme će biti optimizovan za naredne *Web browser*-e: Microsoft Edge, Google Chrome i Mozilla Firefox [2].
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 definisanih u specifikaciji zahteva [5].

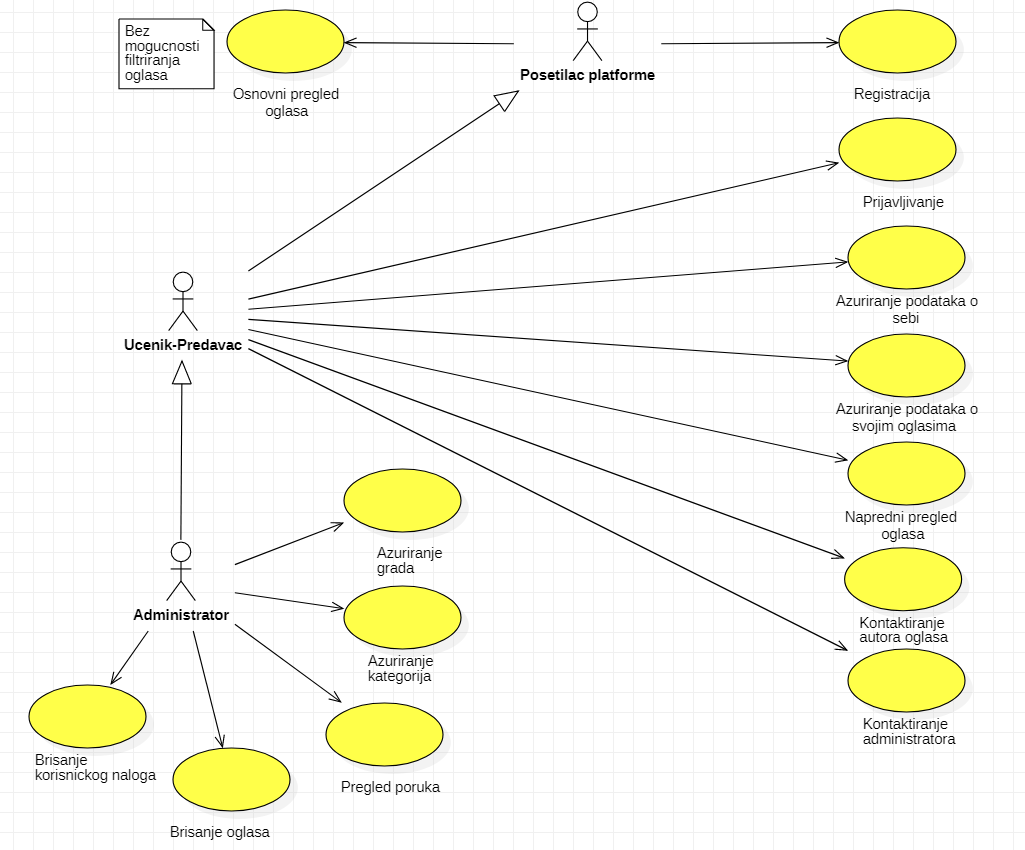
Slučajevi korišćenja **LanTern** *Web* platforme su:

* *Osnovni pregled oglasa*
  + Pregled svih oglasa bez mogućnosti filtriranja
* *Registrovanje*
  + Registrovanje na platformu
* *Prijavljivanje*
  + Prijavljivanje na platformu
* *Ažuriranje podataka o sebi*
  + Ažuriranje podataka o korisniku
* *Ažuriranje podataka o svojim oglasima*
  + Postavljanje novog oglasa
  + Ažuriranje postojećeg oglasa
  + Brisanje postojećeg oglasa
* *Napredni pregled oglasa*
  + Pregled oglasa po kategoriji
  + Pregled oglasa po gradu
  + Pregled oglasa sortiranih po ceni časa
  + Pregled profila autora oglasa
  + Čuvanje oglasa
  + Pregled sačuvanih oglasa
    - Uklanjanje oglasa iz sačuvanih oglasa
* *Ažuriranje podataka o kategorijama*
  + Dodavanje novih kategorija
  + Brisanje postojećih kategorija
* *Ažuriranje podataka o gradovima*
  + Dodavanje novih gradova
  + Brisanje postojećih gradova
* Brisanje oglasa
* Brisanje korisničkog naloga

Ove slučajeve korišćenja mogu da iniciraju: Posetilac platforme, Učenik-Predavač, ili Administrator.

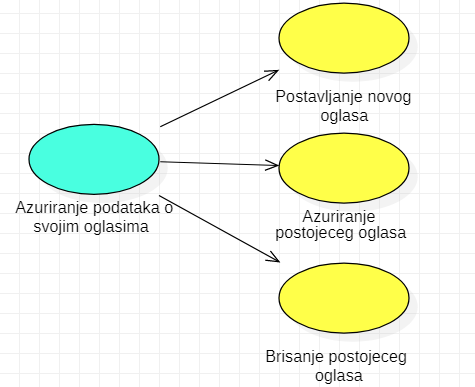
## Dijagrami slučajeva korišćenja

Osnovni *UML* dijagram koji prikazuje korisnike i slučajeve korišćenja **LanTern** *Web* platforme prikazan je na sledećoj slici:

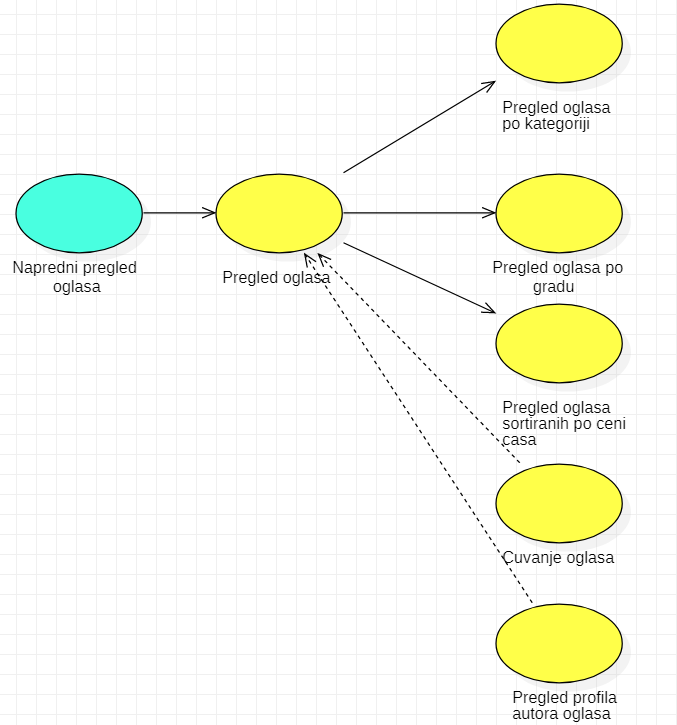


Slučajevi korišćenja: Ažuriranje podataka o svojim oglasima, Napredni pregled oglasa, Ažuriranje grada i Ažuriranje kategorije obuhvataju složenije radnje koje se mogu razložiti dalje na pojedinačne slučajeve korišćenja.

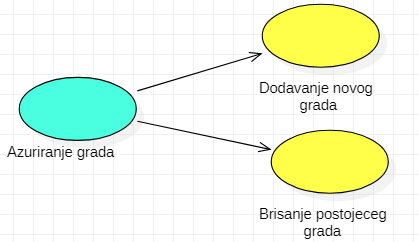
Detaljni *UML* dijagram za slučaj korišćenja Ažuriranje podataka o svojim oglasima je prikazan na sledećoj slici:



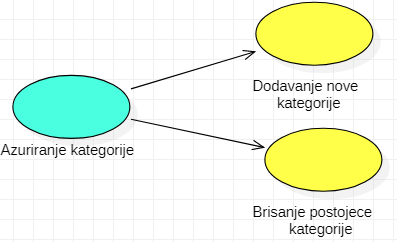
Detaljni *UML* dijagram za slučaj korišćenja Napredni pregled oglasa je prikazan na sledećoj slici:



Detaljni *UML* dijagram za slučaj korišćenja Ažuriranje grada je prikazan na sledećoj slici:



Detaljni *UML* dijagram za slučaj korišćenja Ažuriranje kategorije je prikazan na sledećoj slici:



## Kratak opis slučajeva korišćenja

### Registrovanje na platformu

**Kratak opis:** Registrovanje korisnika na platformu u cilju korišćenja naprednih funkcionalnosti sistema.

**Akteri koji iniciraju slučaj korišćenja:** Posetilac platforme, Učenik-Predavač, Administrator.

### Prijavljivanje na platformu

**Kratak opis:** Prijavljivanje korisnika na platformu u cilju korišćenja naprednih funkcionalnosti sistema, među kojima je i postavljanje oglasa.

**Akteri koji iniciraju slučaj korišćenja:** Učenik-Predavač, Administrator.

### Pregled oglasa po gradu

**Kratak opis:** Prikaz svih oglasa po izabranom gradu.

**Akteri koji iniciraju slučaj korišćenja:** Učenik-Predavač, Administrator.

### Pregled oglasa sortiranih po ceni časa

**Kratak opis:** Prikaz svih oglasa sortiranih po ceni časa.

**Akteri koji iniciraju slučaj korišćenja:** Učenik-Predavač, Administrator.

### Pregled oglasa po kategoriji

**Kratak opis:** Prikaz svih oglasa koji pripadaju kategoriji od interesa.

**Akteri koji iniciraju slučaj korišćenja:** Učenik-Predavač, Administrator.

### Čuvanje oglasa

**Kratak opis:** Prijavljeni korisnici imaju mogućnost da sačuvaju oglase, kako bi ih pogledali kasnije.

**Akteri koji iniciraju slučaj korišćenja:** Učenik-Predavač, Administrator.

### Uklanjanje oglasa iz sačuvanih oglasa

**Kratak opis:** Prijavljeni korisnici imaju mogućnost da uklone oglas iz liste sačuvanih oglasa, ukoliko odluče da im taj oglas iz nekog razloga više nije potreban.

**Akteri koji iniciraju slučaj korišćenja:** Učenik-Predavač, Administrator.

### Kontaktiranje autora oglasa

**Kratak opis:** Korisnik ima mogućnost kontaktiranja autora oglasa.

**Akteri koji iniciraju slučaj korišćenja:** Učenik-Predavač, Administrator.

### Kontaktiranje administratora

**Kratak opis:** Korisnik može da kontaktira administratora u slučaju da ima nekih bitnih pitanja.

**Akteri koji iniciraju slučaj korišćenja:** Učenik-Predavač, Administrator.

### Pregled profila korisnika

**Kratak opis:** Prikaz profila autora određenog oglasa.

**Akteri koji iniciraju slučaj korišćenja:** Učenik-Predavač, Administrator.

### Ažuriranje podataka o korisniku

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

**Akteri koji iniciraju slučaj korišćenja:** Učenik-Predavač, Administrator.

### Postavljanje novog oglasa

**Kratak opis:** Prijavljeni korisnici mogu da postave novi oglas.

**Akteri koji iniciraju slučaj korišćenja:** Učenik-Predavač, Administrator.

### Ažuriranje postojećeg oglasa

**Kratak opis:** Prijavljeni korisnici mogu da ažuriraju svoje postojeće oglase.

**Akteri koji iniciraju slučaj korišćenja:** Učenik-Predavač, Administrator.

### Brisanje postojećeg oglasa

**Kratak opis:** Prijavljeni korisnici mogu da obrišu svoje oglase.

**Akteri koji iniciraju slučaj korišćenja:** Učenik-Predavač, Administrator.

### Dodavanje novih kategorija

**Kratak opis:** Korisnik može da dodaje nove kategorije od interesa kako bi unapredio pretragu.

**Akteri koji iniciraju slučaj korišćenja:** Administrator.

### Brisanje postojećih kategorija

**Kratak opis:** Korisnik može da ukloni kategoriju, ako je to potrebno.

**Akteri koji iniciraju slučaj korišćenja:** Administrator.

### Dodavanje novih gradova

**Kratak opis:** Korisnik može da dodaje nove gradove kako bi unapredio pretragu.

**Akteri koji iniciraju slučaj korišćenja:** Administrator.

### Brisanje postojećih gradova

**Kratak opis:** Korisnik može da ukloni grad, ako je to potrebno.

**Akteri koji iniciraju slučaj korišćenja:** Administrator.

### Brisanje oglasa

**Kratak opis:** Korisnik platforme ima mogućnost da obriše oglas koji je postavljen.

**Akteri koji iniciraju slučaj korišćenja:** Administrator.

### Brisanje korisničkog naloga

**Kratak opis:** Korinik platforme ima mogućnost da obriše korisničke naloge prijavljenih korisnika.

**Akteri koji iniciraju slučaj korišćenja:** Administrator.

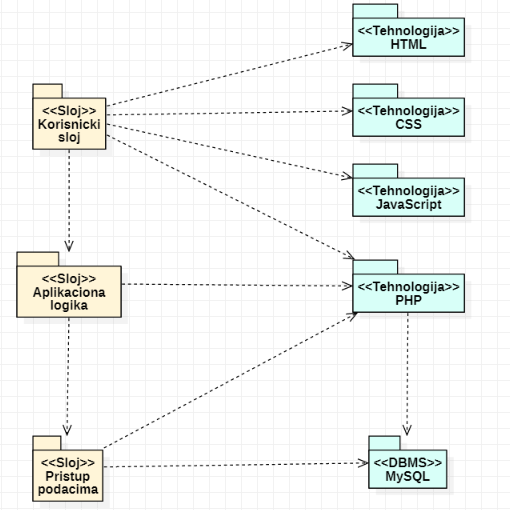
# Pogled na logičku arhitekturu sistema

U ovom odeljku je dat pregled logičke arhitekture sistema. Ovaj pogled sadrži opis najznačajnijih klasa, njihove organizacije u pakete i podsisteme, kao i organizaciju 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 podsistema, paketa ili slojeva moguće je uključiti i odgovarajuće dijagrame.

Logički pogled na **LanTern** *Web* platformu obuhvata sledeće pakete:

* Paket *Korisnički interfejs* – Sadrži *Web* stranice, **PHP** skripte, **JavaScript** skripte, kao i multimedijalni sadržaj koji realizuje grafički dizajn i forme preko kojih korisnici sistema komuniciraju sa sistemom.
* Paket *Aplikaciona logika* – Sadrži neophodne **PHP** skripte, zadužene za realizaciju funkcionalnosti specifičnih za domen sistema koji se razvija, kao i korišćenje Ajax tehnologije.
* 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 je zadužen za realizaciju korisničkog interfejsa platforme. Sadrži sve **HTML** i **CSS** fajlove, **JavaScript** skripte, **PHP** skripte koje služe za generisanje delova *Web* stranica, kao i multimedijalnih sadržaja.

Sloj *Korisnički interfejs* zavisi od sloja aplikacione logike, kao i paketa *HTML*, *CSS*, *JavaScript* i *PHP*.

### Aplikaciona logika

**Sloj**

Sloj zadužen za realizaciju funkcionalnosti specifičnih za domen sistema koji se razvija. Sadrži neophodne **PHP** skripte koje realizuju funkcionalnost karakterističnu za specifični domen, i uspostavljaju vezu između korisničkog interfejsa i sloja za pristup podacima.

Sloj *Aplikaciona logika* zavisi od sloja *Pristup podacima* i *PHP* paketa.

### Pristup podacima

**Sloj**

Navedeni sloj je zadužen za pristup, dodavanje i ažuriranje podataka koji se čuvaju u MySQL bazi podataka.

Sloj *Pristup podacima* ne zavisi ni od jednog drugog sloja. Ali, zavisi od paketa *PHP*, *MySQL* baza podataka.

### HTML

**Tehnologija**

*HTML* definiše gradivne elemente *Web* stranica koje omogućavaju prikaz formatiranih informacija i realizaciju formi za unos i ažuriranje podataka.

### CSS

**Tehnologija**

*CSS* obezbeđuje stilove za gradivne elemente *Web* stranica.

### JavaScript

**Tehnologija**

*JavaScript* tehnologija omogućava pisanje i izvršavanje skripti na klijentskoj strani. Takođe, mogu se koristiti za slanje zahteva za pribavljanje podataka iz baze podataka ili slanje podataka koji bi bili smešteni u bazu isključivo uz korišćenje **Ajax** tehnologije.

### PHP

**Tehnologija**

Tehnologija **PHP**-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 **LanTern** *Web* platforme.

# Pogled na procese

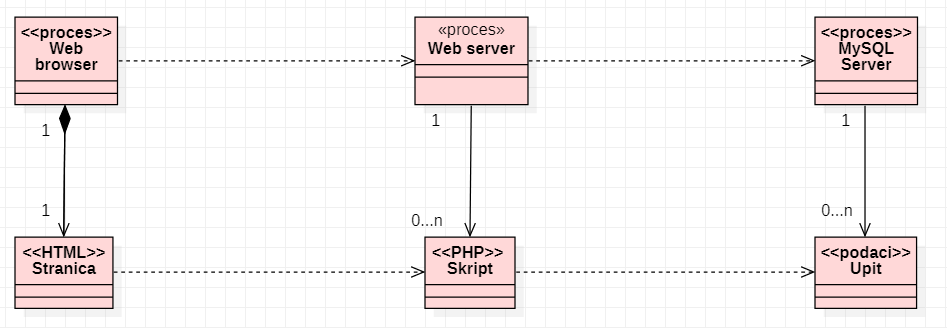
U ovom odeljku je sadržan pogled na procesnu arhitekturu sistema. Ovaj opis treba da sadrži specifikaciju različitih zadataka uključenih u rad sistema. Takođe je potrebno dati dijagrame koji pokazuju njihovu interakciju i konfiguraciju.

Web aplikacije zasnovane na **PHP**-u imaju relativno jednostavan procesni model koji je u potpunosti pod kontrolom Web servera. Sa stanovišta projektanta PHP 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 **LanTern** *Web* platforme.

## Procesi

Na sledećem *UML* dijagramu klasa prikazani su procesi koji učestvuju u izvršenju **LanTern** *Web* platforme. Dijagram se može primeniti i na neke druge *Web* aplikacije koje su zasnovane na **PHP**-u i **MySQL** bazi podataka.



### Web browser

*Web browser* predstavlja proces koji izvršava funkcionalnost aplikacije za prikaz *HTML* *stranica* čiji su podaci dobijeni od *Web* servera. Opšti slučaj obuhvata to da *Web browser* 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

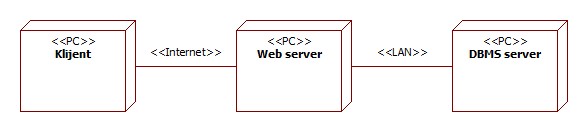
Web server je proces koji izvršava funkcionalnost opsluživanja zahteva prispelih sa više Web browser-a. Ukoliko je zahtevana stranica PHP skript, Web server inicira izvršenje procesa koji obrađuje odgovarajući skript i generiše sadržaj koji se vraća browser-u.

### MySQL Server

*MySQL Server* predstavlja 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 inicirao.

# Pogled na raspoređivanje sistema

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

Na sledećoj slici dat je *UML* dijagram raspoređivanja **LanTern** *Web* platforme.

## Klijent

Pristup **LanTern** *Web* platformi se obavlja preko klijentskih računara na kojima se izvršava ***Web browser***. Za povezivanje između klijenata i *Web* servera koristi se *Internet* infrastruktura, pa nema ograničenja u pogledu lokacije na kojoj se klijent nalazi.

## Web server

Računar na kome se izvršava *Web* *server* opslužuje više klijenata koji pristupaju preko Interneta. Na ovom računaru može da se izvršava obrada neophodnih ***PHP*** *sripti*. U opštoj konfiguraciji *DBMS* se izvršava na posebnoj mašini koja je sa *Web* serverom u lokalnoj mreži (*LAN*).

## DBMS server

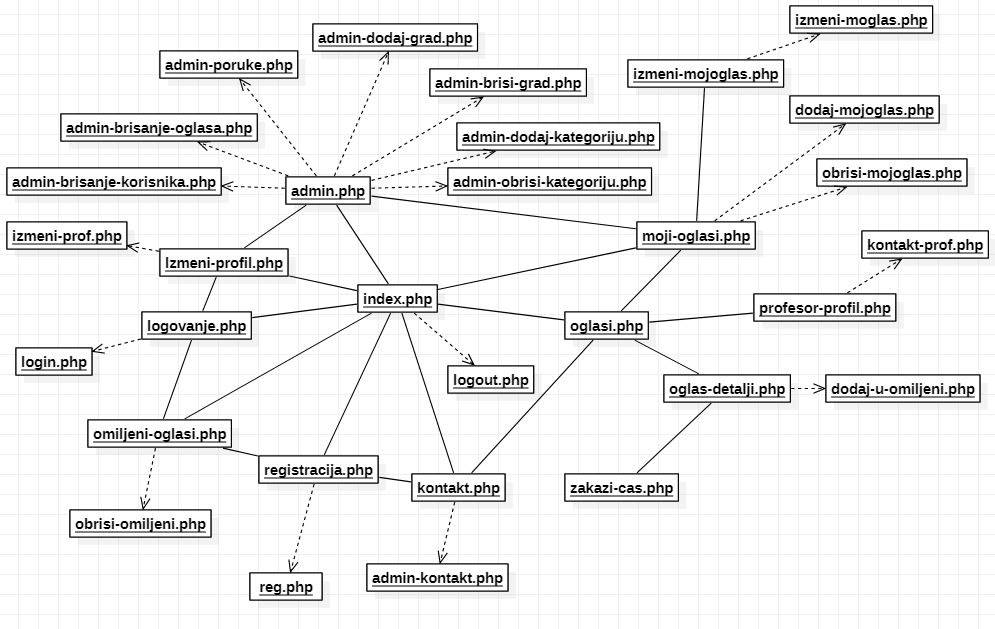
*DBMS server* je računar na kome se izvršava ***MySQL Server*** proces koji realizuje funkcionalnost sistema za upravljanje bazama podataka. Zbog sigurnosti podataka koji se na ovom računaru čuvaju, pristup bazi je ograničen samo na računare iz lokalne mreže (*LAN*).

# Pogled na implementaciju sistema

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

## Model domena

Model domena za koji se **LanTern** *Web* platforma projektuje je ilustrovan sledećim dijagramom. Koristiće komponente (PHP skripte) za komunikaciju sa bazom podataka. Model domena predstavlja osnovu za identifikaciju komponenata (PHP skripti) koje će biti implementirane kao i veze koje se mogu identifikovati između njih.

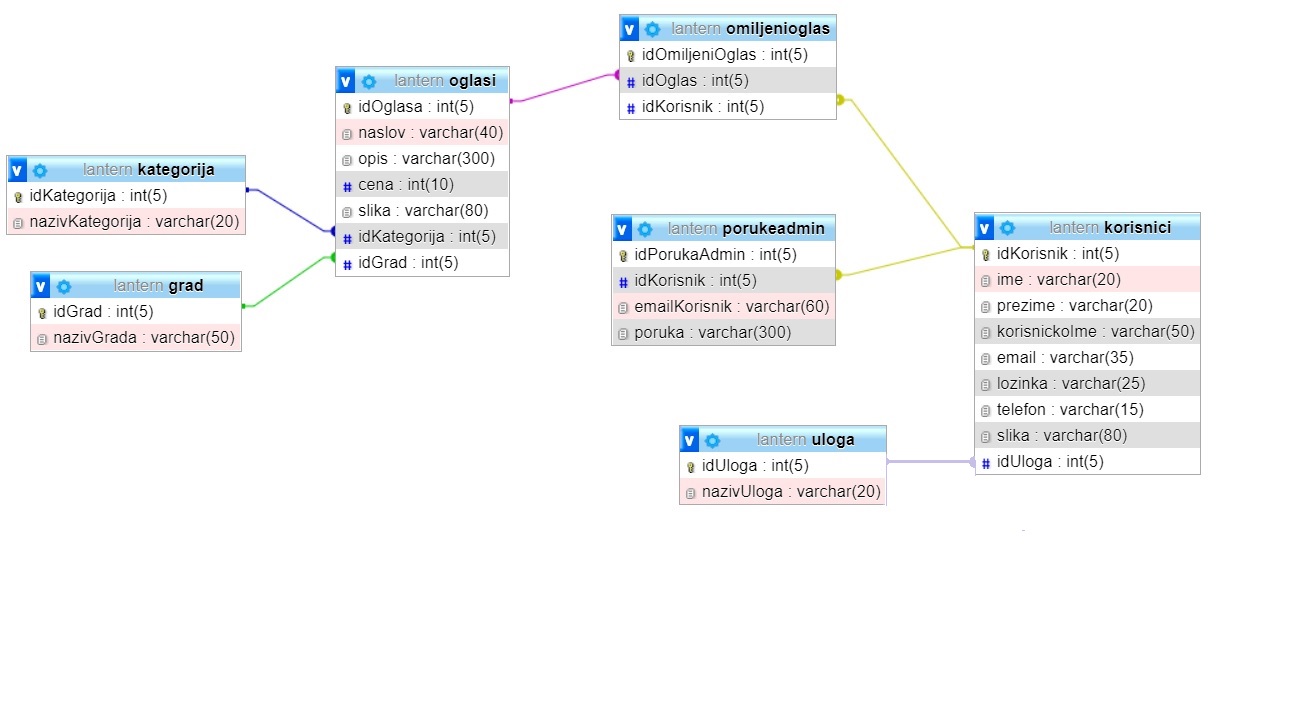


**Napomena:** Kada se korisnik uloguje na svakoj stranici pojavljuje se link u navigacionom meniju, koji poziva funkciju logout.php.

Takođe, sve stranice su međusobno povezane i može im se pristupiti sa bilo koje stranice ovog sajta.

## Šema baze podataka

Detaljna šema baze podataka je prikazana na sledećem dijagramu. Baza podataka i dijagram su kreirani korišćenjem XAMPP aplikacije, u okruženju phpMyAdmin, korišćenjem ***MySQL*** *servera*.



## Komponente sistema

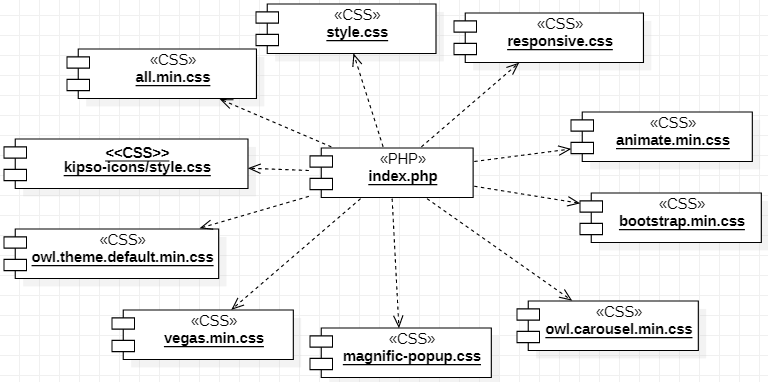
Komponente sistema **LanTern** *Web* platforme su prvenstveno **PHP** skripte čiji će pregled biti dat po arhitekturnim slojevima. Za ilustraciju će biti korišćeni *UML* dijagrami komponenti.

### Komponente korisničkog interfejsa

CSS stilovi i JavaScript skripte su identifikovane iz preuzetog template-a.

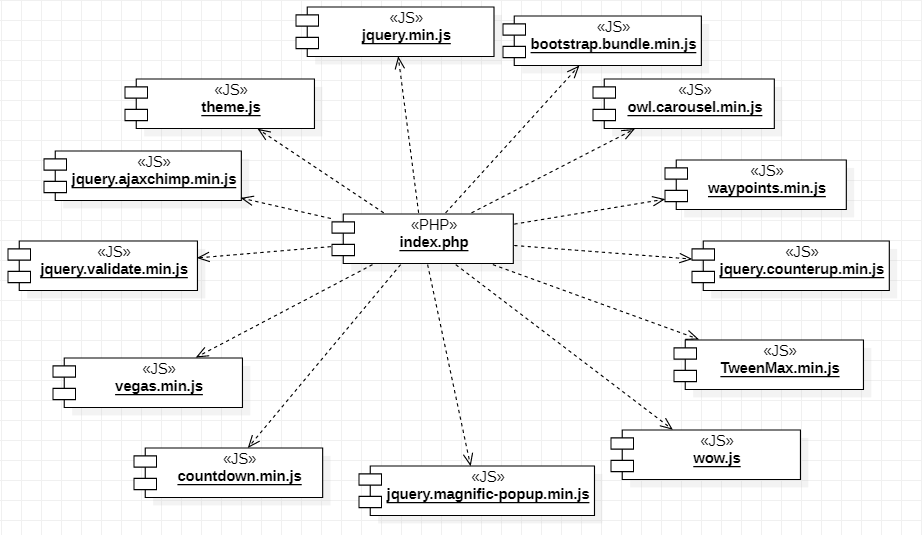
* **CSS stilovi**

Dizajn korisničkog interfejsa je obuhvaćen sledećim komponentama:



* **JavaScript skripte**

Dizajn korisničkog interfejsa je obuhvaćen sledećim komponentama:



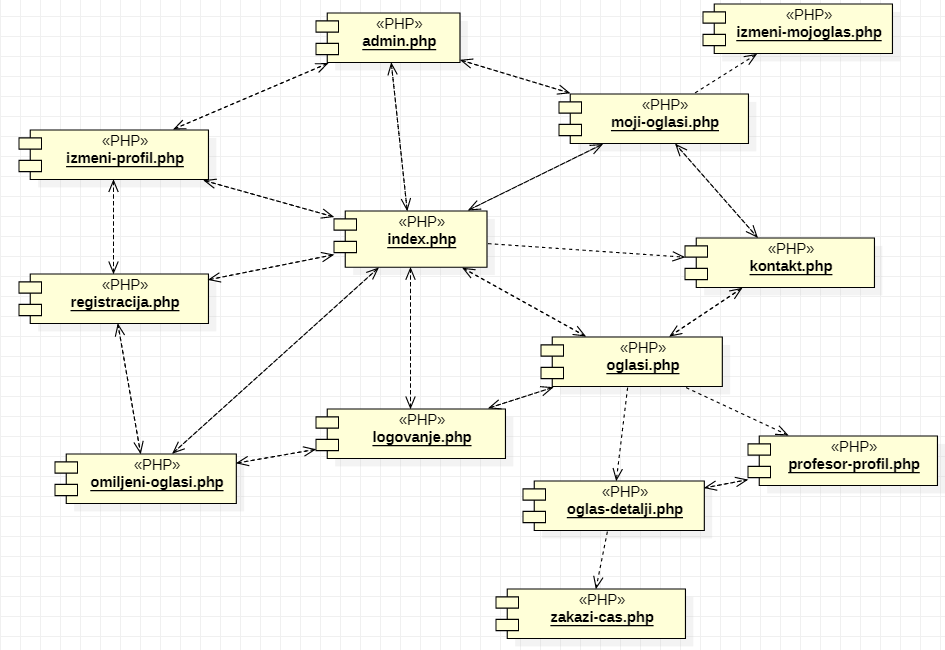
**Napomena:** Komponente korisničkog interfejsa koje imaju iste CSS stilove, kao i JavaScript skripte su: *index.php, admin.php, izmeni-mojoglas.php, kontakt.php, logovanje.php, moji-oglasi.php, oglas-detalji.php, oglasi.php, omiljeni-oglasi.php, profesor-profil.php, registracija.php, zakazi-cas.php.*

Ovde povodom realizacije je šemantski prikazan izgled samo index.php komponente koja je povezana sa odgovarajućim CSS stilovima, kao i JavaScript skriptama. Ostale komponente (php komponente) su identično povezane sa CSS stilovima i JavaScript skriptama kao i index.php komponenta.

### Komponente aplikacione logike

Komponente koje realizuju domen problema mogu da se uključe preko većine komponenti korisničkog interfejsa. Uključenje komponenti će se obavljati pomoću navigacionog menija.

Na sledećem dijagramu su prikazane komponente ovog sloja i njihove međusobne zavisnosti:



**Napomena:** Ako je korisnik ulogovan kao aministrator u navigacionom meniju će se pojaviti dodatna komponenta “Admin”. Ako korisnik nije ulogovan neće mu biti prikazane komponente koje su zaštićene sesijama.

**index.php** je komponenta na kojoj se nalazi informativni sadržaj, preko nje može da se pristupi sledećim komponentama: index.php, oglasi.php, izmeni-profil.php, omiljeni-oglasi.php, kontakt.php, moji-oglasi.php.

**oglasi.php** je komponenta na kojoj se nalaze svi oglasi, preko nje može da se pristupi sledećim komponentama: index.php, moji-oglasi.php, izmeni-profil.php, omiljeni-oglasi.php, oglas-detalji.php, profesor-profil.php, kontakt.php

**oglasi-detalji.php** je komponenta na kojoj se nalaze svi podaci o izabranom oglasu, preko nje može da se pristupi sledećim komponentama: index.php, moji-oglasi.php, izmeni-profil.php, omiljeni-oglasi.php, zakazi-cas.php, profesor-profil.php, kontakt.php

**zakazi-cas.php** je komponenta na kojoj se nalaze svi podaci o izabranom oglasu, preko nje može da se pristupi sledećim komponentama: index.php, moji-oglasi.php, izmeni-profil.php, omiljeni-oglasi.php, kontakt.php

**profesor-profil.php** je komponenta na kojoj se nalaze svi podaci o izabranom oglasu, preko nje može da se pristupi sledećim komponentama: index.php, moji-oglasi.php, izmeni-profil.php, omiljeni-oglasi.php, kontakt.php, oglas-detalji.php

**logovanje.php** je komponenta na kojoj se nalazi forma za logovanje, preko nje može da se pristupi sledećim komponentama: registracija.php, kontakt.php, oglasi.php

**registracija.php** je komponenta na kojoj se nalazi forma za registraciju, preko nje može da se pristupi sledećim komponentama: registracija.php, kontakt.php, oglasi.php

**moji-oglasi.php** je komponenta na kojoj se nalaze oglasi koje je korisik postavio, preko nje može da se pristupi sledećim komponentama: index.php, oglasi.php, izmeni-profil.php, omiljeni-oglasi.php, kontakt.php, oglas-detalji.php, izmeni-mojoglas.php

**izmeni-mojoglas.php** je komponenta na kojoj se nalazi forma za ažuriranje postavljenog oglasa, preko nje može da se pristupi sledećim komponentama: index.php, oglasi.php, izmeni-profil.php, omiljeni-oglasi.php, kontakt.php, moji-oglasi.php

**izmeni-profil.php** je komponenta na kojoj se nalazi forma za ažuriranje profila korisnika, preko nje može da se pristupi sledećim komponentama: index.php, oglasi.php, izmeni-profil.php, omiljeni-oglasi.php, kontakt.php, moji-oglasi.php

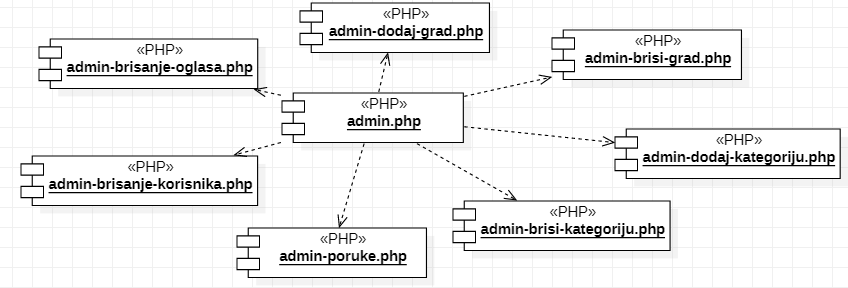
**omiljeni.php** je komponenta na kojoj se nalaze svi oglasi koji su omiljeni, preko nje može da se pristupi sledećim komponentama: index.php, oglasi.php, izmeni-profil.php, omiljeni-oglasi.php, kontakt.php, moji-oglasi.php

**kontakt.php** je komponenta na kojoj se nalazi forma za kontakt aministratora, preko nje može da se pristupi sledećim komponentama: index.php, oglasi.php, izmeni-profil.php, omiljeni-oglasi.php, kontakt.php, moji-oglasi.php

**admin.php** je komponenta na kojoj se nalazi forma za kontakt aministratora, preko nje može da se pristupi sledećim komponentama: index.php, oglasi.php, izmeni-profil.php, omiljeni-oglasi.php, kontakt.php, moji-oglasi.php.

### Komponente za pristup podacima

Pristup bazi podataka se obavlja kroz fajlove koji se nalaze u okviru foldera “funkcije“. U tim fajlovima se nalazi isključivo **PHP** kod. Korišćenjem zadatih upita pristupamo podacima koji se nalaze u našoj bazi podataka.



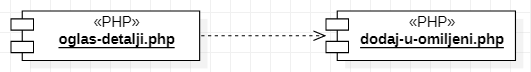


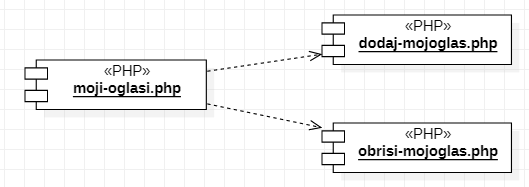














# Performanse

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 platformi.
2. Vreme potrebno za pristupanje bazi podataka u cilju izvršenja 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 specificiranih u zahtevima u pogledu pouzdanosti [5]:

1. **LanTern** *Web* platforma će biti dostupna 24h dnevno, 7 dana u nedelji. Vreme kada platforma nije dostupna ne sme da pređe 10%.
2. Srednje vreme između dva sukcesivna otkaza ne sme da padne ispod 120 sati.