Teach and learn

Arhitekturni projekat

Verzija 1.0

Pregled izmena

|  |  |  |  |
| --- | --- | --- | --- |
| **Datum** | **Verzija** | **Opis** | **Autor** |
| 17.05.2021. | 1.0 | Inicijalna verzija | MNMteam |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Sadržaj

1. Cilj dokumenta 4

2. Opseg dokumenta 4

3. Predstavljanje arhitekture 4

4. Ciljevi i ograničenja arhitekture 4

5. Pogled na slučajeve korišćenja 4

5.1 Dijagrami slučajeva korišćenja 4

5.2 Kratak opis slučajeva korišćenja 7

5.2.1 Pregled informacija 7

5.2.2 Kreiranje naloga 7

5.2.3 Prijavljivanje na sistem 7

5.2.4 Slanje privatnih poruka 7

5.2.5 Formiranje grupa 7

5.2.6 Rukovođenje serverom 7

6. Pogled na logičku arhitekturu sistema 7

6.1 Pregled arhitekture – organizacija paketa i podsistema u slojeve 8

6.1.1 Korisnički interfejs 8

6.1.2 Aplikaciona logika 8

6.1.3 Pristup podacima 8

6.1.4 Angular 8

6.1.5 .NET 8

6.1.6 MySQL 9

7. Pogled na procese 9

7.1 Procesi 9

7.1.1 Web čitač 9

7.1.2 Web server 9

7.1.3 .NET 9

7.1.4 MySQL Server 9

8. Pogled na raspoređivanje sistema 10

8.1 Klijent 10

8.2 Web server 10

8.3 DBMS server 10

9. Pogled na implementaciju sistema 10

9.1 Model domena 10

9.2 Šema baze podataka 10

Arhitekturni projekat

# Cilj dokumenta

Cilj ovog dokumenta je detaljni opis arhitekture Teach and learn apikacije.

# Opseg dokumenta

Dokument se odnosi na Teach and learn aplikaciju koja će biti razvijena od strane MNMteam-a. Namena sistema je efikasno udruživanje studenata rati lakseg i efikasnijeg učenja.

Reference

Spisak korišćene literature:

1. Teach and learn – Predlog projekta, MNMteam.
2. Teach and learn – Vizija sistema, MNMteam.
3. Teach and learn – Plan realizacije projekta, MNMteam.
4. Teach and learn – Specifikacija zahteva, MNMteam.

# 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. Teach and learn aplikacija će biti implementirana kao Web aplikacija zasnovana na Angular front-endu, .Net back-endu i MySQL bazi podataka.
2. Klijentski deo Teach and learn aplikacije će biti optimizovan za sledeće Web čitače: Google Chrome, Opera 8.0 i noviji, kao i Firefox (Mozilla) .
3. Svi zahtevi u pogledu performansi dati u [5] moraju biti uzeti u obzir pri izboru arhitekture i razvoju sistema.

# Pogled na slučajeve korišćenja

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

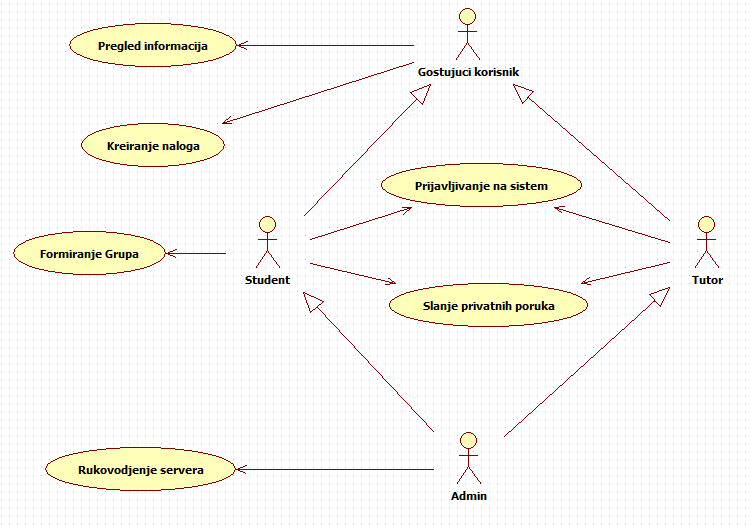
Slučajevi korišćenja Teach and learn aplikacije su:

* Pregled informacija
  + Broj korisnika
  + Trenutno formirane grupe
* Kreiranje naloga
* Prijavljivanje na sistem
* Slanje privatnih poruka
* Formiranje grupa
* Rukovođenje serverom

Ove slučajeve korišćenja mogu da iniciraju student, tutor, gost, admin.

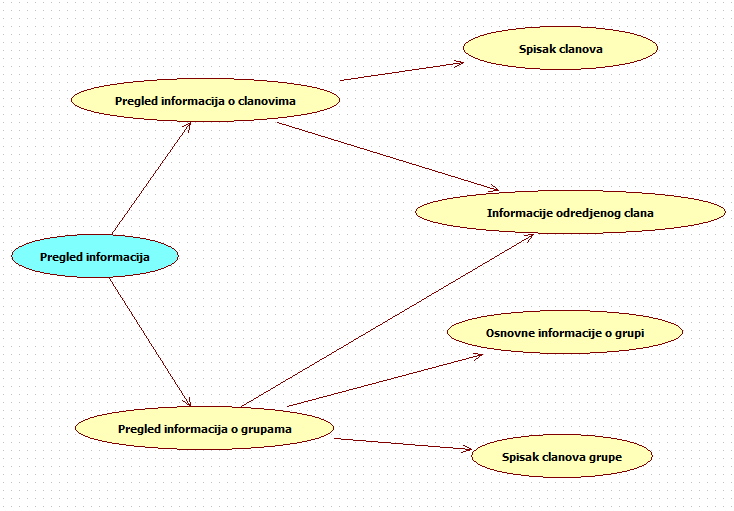
## Dijagrami slučajeva korišćenja

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

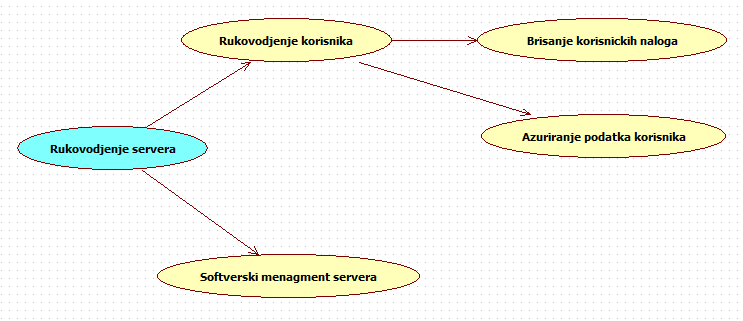


Slučajevi korišćenja *pregled informacija*, *formiranje grupa, rukovodjenje servera, slanje privatnih poruka* i *prijavljivanje na sistem* obuhvataju složenije radnje koje se mogu razložiti dalje razložiti na pojedinačne slučajeve korišćenja.

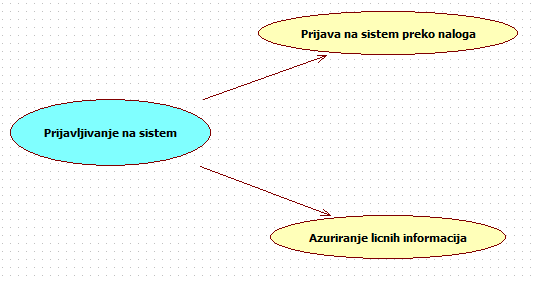
Detaljni UML dijagram za slučaj korišćenja *pregled informacija* je prikazan na sledećoj slici:



Detaljni UML dijagram za slučaj korišćenja *rukovođenje servera* je prikazan na sledećoj slici:



Detaljni UML dijagram za slučaj korišćenja *prijavljivanje na sistem* je prikazan na sledećoj slici:



## Kratak opis slučajeva korišćenja

### Pregled informacija

Kratak opis: Prikaz podataka kao sto su broj korisnika i trenutno formirane grupe.

### Kreiranje naloga

Kratak opis: Kreiranje naloga korisnika.

### Prijavljivanje na sistem

Kratak opis: Prijavljivanje korisnika na postojeći nalog

### Slanje privatnih poruka

Kratak opis: Mogucnost komunikacije preko privatnih poruka.

### Formiranje grupa

Kratak opis: Kreiranje grupa za saradnju

### Rukovođenje serverom

Kratak opis: Administratorske funkcionalnosti

# 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.

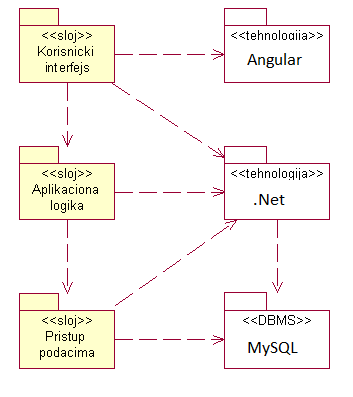
Logički pogled na Teach and learn aplikaciju obuhvata 3 glavna paketa: Korisnički interfejs, Aplikaciona logika, Pristup podacima.

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

Paket *Pristup podacima* sadrži .NET 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 aplikacije. U njemu su sadržane sve HTML stranice, multimedijalni sadržaji i .NET koje generišu HTML stranice preko kojih korisnici komuniciraju sa sistemom.

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

### Aplikaciona logika

sloj

Sloj aplikacione logike je srednji sloj u troslojnoj arhitekturi Teach and learn aplikacije. Sadrži C# skripte koje realizuju funkcionalnost karakterističnu za domen primene aplikacije i uspostavljaju vezu između korisničkog interfejsa i sloja za pristup podacima.

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

### Pristup podacima

sloj

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

### Angular

tehnologija

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

### .NET

tehnologija

Tehnologija .NET obezbeđuje mehanizam za pisanje i izvršavanje skripti na strani servera. Ove skripte 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 Teach and learn aplikacije.

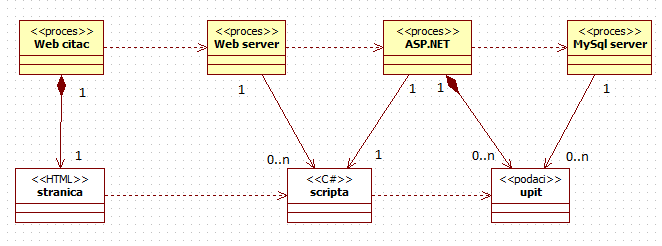
# Pogled na procese

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

Ilustracije radi u nastavku je dat opis procesa uključenih u izvršenje Teach and learn kao Web aplikacije.

## Procesi

Na sledećem UML dijagramu klasa prikazani su procesi koji učestvuju u izvršenju Teach and learn aplikacije. Dijagram je opšteg tipa i može se primeniti na bilo koju Web aplikaciju zasnovanu na .NET-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 C# skript, Web server inicira izvršenje .NET procesa koji obrađuje odgovarajući skript i generiše sadržaj koji se vraća čitaču. Web server može paralelno da inicira veći broj .NET procesa.

### .NET

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

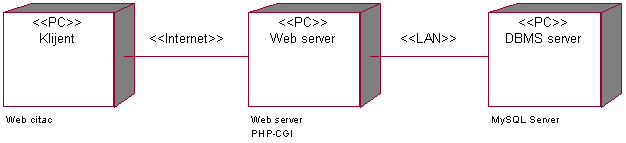
### MySQL Server

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 Teach and learn aplikacije.



## Klijent

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

## Web server

Računar na 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 .NET koji vrše obradu zadatih C# 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 Teach and learn aplikacije ovaj odeljak sadrži model domena, šemu baze podataka i prikaz komponenti sistema razvrstanih u ranije identifikovane pakete.

## Model domena

Model domena za koji se Teach and learn aplikacije 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 (C# 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.

