NightEvents

Web aplikacija za lakšu organizaciju noćnih događaja u klubovima

Arhitekturni projekat

Verzija 1.0

Pregled izmena

|  |  |  |  |
| --- | --- | --- | --- |
| **Datum** | **Verzija** | **Opis** | **Autor** |
| 19.04.2023. | 1.0 | Inicijalna verzija | Luka Gavrilović  Mihajlo Mihajlović  Aleksa Živković  Andriana Petrović |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

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 Izbor jezika 8

6.2.2 Pregled osnovnih podataka o laboratoriji 8

6.2.3 Pregled spiska članova 8

6.2.4 Pregled podataka o određenom članu laboratorije 8

6.2.5 Pregled publikacija po autoru 8

6.2.6 Pregled publikacija po tipu 8

6.2.7 Pregled publikacija po godini 8

6.2.8 Pregled spiska projekata 8

6.2.9 Pregled podataka o određenom projektu 8

6.2.10 Prijavljivanje 9

6.2.11 Ažuriranje podataka o članu 9

6.2.12 Dodavanje nove publikacije 9

6.2.13 Brisanje postojeće publikacije 9

6.2.14 Ažuriranje osnovnih podataka o laboratoriji 9

6.2.15 Kreiranje novog člana 9

6.2.16 Brisanje postojećeg člana 9

6.2.17 Arhiviranje postojećeg člana 9

6.2.18 Kreiranje projekta i postavljanje vođe 9

6.2.19 Ažuriranje podataka o projektu 9

7. Pogled na logičku arhitekturu sistema 9

7.1 Pregled arhitekture – organizacija paketa i podsistema u slojeve 10

7.1.1 Korisnički interfejs 10

7.1.2 Aplikaciona logika 10

7.1.3 Pristup podacima 10

7.1.4 HTML 10

7.1.5 PHP 11

7.1.6 MySQL 11

8. Pogled na procese 11

8.1 Procesi 11

8.1.1 Web čitač 11

8.1.2 Web server 11

8.1.3 PHP-CGI 11

8.1.4 MySQL Server 12

9. Pogled na raspoređivanje sistema 12

9.1 Klijent 12

9.2 Web server 12

9.3 DBMS server 12

10. Pogled na implementaciju sistema 12

10.1 Model domena 12

10.2 Šema baze podataka 13

10.3 Komponente sistema 14

10.3.1 Komponente korisničkog interfejsa 14

10.3.2 Komponente aplikacione logike 14

10.3.3 Komponente za pristup podacima 15

11. Performanse 16

12. Kvalitet 16

Arhitekturni projekat

# Cilj dokumenta

Cilj ovog dokumenta je detaljni opis arhitekture NightEvents Web aplikacije.

# Opseg dokumenta

Dokument se odnosi na NightEvents aplikaciju koja će biti razvijena od strane MAALTeam-a. NightEvents predstavlja skraćenicu za Sistem koji će omogućiti lakšu organizaciju i prisustvo događajima. Namena sistema je efikasno prezentovanje, kreiranje i održavanje sadržaja vezanih za organizaciju muzičkih događaja na određenom području.

# Reference

Spisak korišćene literature:

1. NightEvents – Predlog projekta, SWE-NightEvents-01, V1.0, 2023, MAALTeam.
2. NightEvents – Planirani raspored aktivnosti na projektu, V1.0, 2023, MAALTeam.
3. NightEvents – Plan realizacije projekta, V1.0, 2023, MAALTeam.
4. NightEvents – Vizija sistema, SWE-NightEvents-03, V1.0, 2023, MAALTeam.
5. NightEvents – Specifikacija zahteva, V1.0, 2023, MAALTeam.

# 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. NightEvents aplikacija će biti implementirana kao Web aplikacija zasnovana na C# jeziku za backend, HTML, CSS i JavaScript za frontend i MySQL bazi podataka (Azure Data Studio) [4].
2. Klijentski deo NightEvents aplikacije će biti optimizovan za sve Web čitače.
3. Svi zahtevi u pogledu performansi dati u [5] moraju biti uzeti u obzir pri izboru arhitekture i razvoju sistema.

# Pogled na slučajeve korišćenja

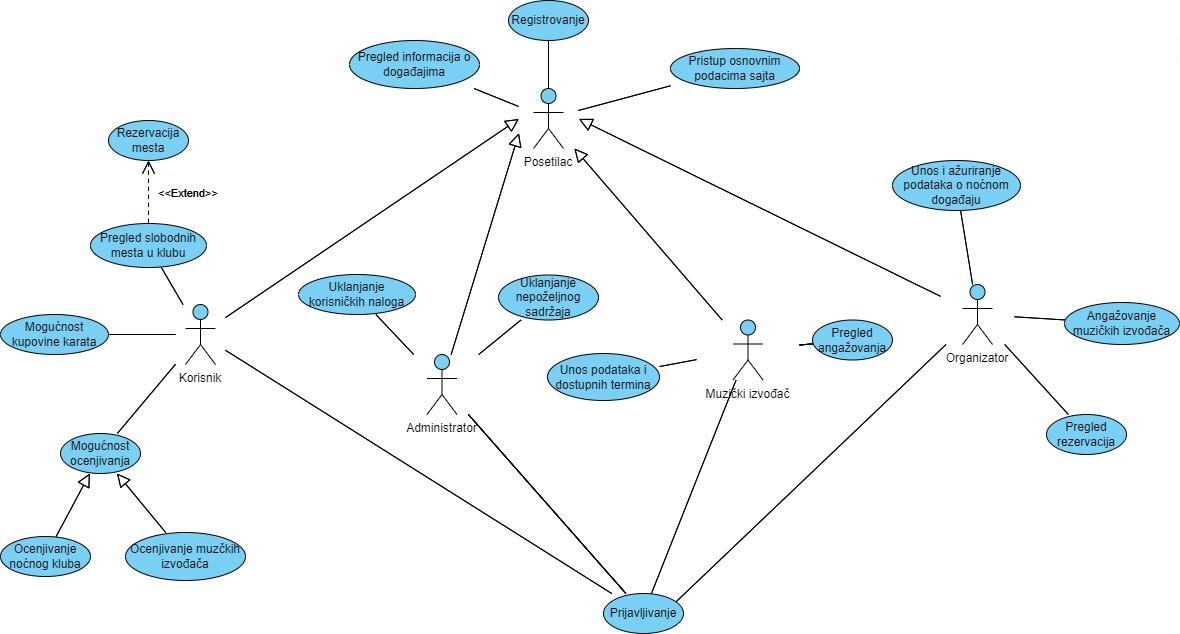
U ovom odeljku je dat pogled na slučajeve korišćenja definisane u specifikaciji zahteva [5].

Slučajevi korišćenja NightEvents portala su:

* *Pregled informacija o događajima*
  + Pregled podataka po određenom kriterijumu
  + Pregled podataka o izabranom događaju
  + Pregled dostupnih mesta
* *Registrovanje*
  + Unos imena
  + Unos prezimena
  + Unos e-mail adrese
  + Unos korisničkog imena
  + Unos šifre
* Pristup osnovnim podacima sajta
* Pregled slobodnih mesta u klubu
* Mogućnost kupovine karata
* *Mogućnost ocenjivanja*
  + Ocenjivanje noćnog kluba
  + Ocenjivanje muzičkih izvođača
* Prijavljivanje
* Uklanjanje korisničkih naloga
* Uklanjanje nepoželjnog sadržaja
* Unos podataka i dostupnih termina
* Pregled angažovanja
* Angažovanje muzičkih izvođača
* Unos i ažuriranje podataka o noćnom događaju
* Pregled rezervacija

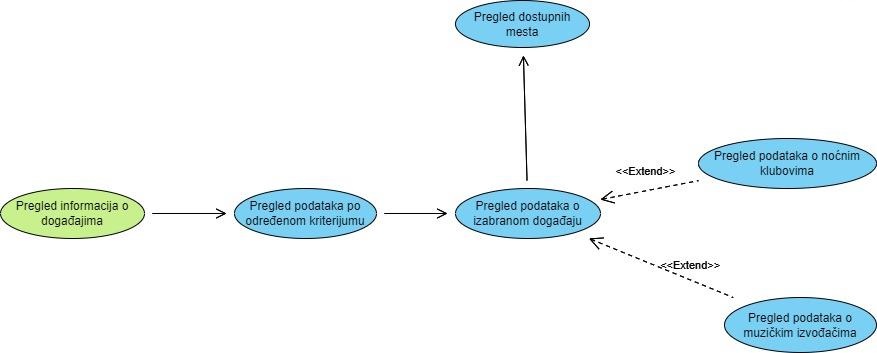
Ove slučajevi korišćenja mogu da iniciraju posetilac aplikacije, korisnik, organizator, muzički izvođač ili administrator.

## Dijagrami slučajeva korišćenja

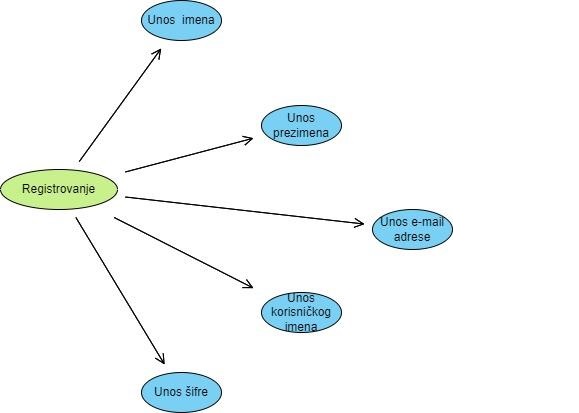
Osnovni UML dijagram koji prikazuje korisnike i slučajeve korišćenja NightEvents aplikacije prikazan je na sledećoj slici:

Slučajevi korišćenja *pregled informacija* *o događajima* i *registrovanje* obuhvataju složenije radnje koje se mogu dalje razložiti na pojedinačne slučajeve korišćenja.

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



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



## Kratak opis slučajeva korišćenja

### Registracija

Kratak opis: Registracija je neophodna radi kreiranja naloga.

Akteri koji iniciraju slučaj korišćenja: Administrator, Posetilac, Organizator, Muzički izvođač.

### Pregled podataka po određenom kriterijumu

Kratak opis: Prikaz stranice sa informacijama o noćnim događajima koji zadovoljavaju unete kriterijume.

Akteri koji iniciraju slučaj korišćenja: Posetilac aplikacije, Korisnik, Organizator, Muzički izvođač, Administrator.

### Pregled podataka o izabranom događaju

Kratak opis: Podaci o noćnom događaju, koji je prethodno izabran po određenom kriterijumu, mogu biti prikazani.

Akteri koji iniciraju slučaj korišćenja: Posetilac aplikacije, Korisnik

### Pregled podataka o noćnim klubovima

Kratak opis: Prikaz stranice Web aplikacije sa spiskom noćnih klubovima na određenom području.

Akteri koji iniciraju slučaj korišćenja: Posetilac aplikacije, Organizator, Muzički izvođač, Administrator.

### Pristup osnovnim podacima sajta

Kratak opis: Prikaz početne stranice sajta gde se nalaze najosnovnije informacije. Cilj je da se posetiocima, dakle korisnicima koji nisu logovani ne daje mnogo informacija.

Akteri koji iniciraju slučaj korišćenja: Posetilac aplikacije, Korisnik, Administrator, Organizator, Muzički izvođač.

### Pregled slobodnih mesta u klubu

Kratak opis: Pregled stranice sa grafičkim prikazom dostupnih mesta u klubu.

Akteri koji iniciraju slučaj korišćenja: Korisnik.

### Mogućnost kupovine karata

Kratak opis: Prikaz stranice sa događajem za koji se može kupiti karta.

Akteri koji iniciraju slučaj korišćenja: Korisnik.

### Mogućnost ocenjivanja

Kratak opis: Posetioci žurke moći će da ocene noćni klub u kome su prisustvovali, kao i muzičke izvođače koji su bili zaduženi za taj noćni događaj.

Akteri koji iniciraju slučaj korišćenja: Korisnik.

### Unos podataka i dostupnih termina

Kratak opis: Unos podataka o muzičkim izvođačima i dodavanje slobodnih termina za njihove nastupe.

Akteri koji iniciraju slučaj korišćenja: Muzički izvođač.

### Pregled angažovanja

Kratak opis: Prikaz stranica sa spiskom događaja na kojima je angažovan muzički izvođač.

Akteri koji iniciraju slučaj korišćenja: Muzički izvođač.

### Unos i ažuriranje podataka o noćnom događaju

Kratak opis: Podaci koji se unesu kao podaci vezani za noćni događaj nisu fiksni, mogu da uslede promene.

Akteri koji iniciraju slučaj korišćenja: Organizator.

### Angažovanje muzičkih izvođača

Kratak opis: Biranje dostupnih muzičkih izvođača za nastup na događaju koji organizator organizuje.

Akteri koji iniciraju slučaj korišćenja: Organizator.

### Pregled rezervacija

Kratak opis: Za sve rezervacije u noćnom klubu postoji mogućnost njihovog pregleda dostupnosti samog organizatora.

Akteri koji iniciraju slučaj korišćenja: Organizator.

### Brisanje postojećih naloga

Kratak opis: Administrator ima tu privilegiju da ukloni određene naloge.

Akteri koji iniciraju slučaj korišćenja: Administrator.

### Prijavljivanje

Kratak opis: Nakon registrovanja, postoji mogućnost prijavljivanja i prijavljivanjem otvara se nova stranica sa novim funkcionalnostima. Prijavljivanje korisnika na aplikaciju u cilju pristupa specifičnim funkcijama koje zahtevaju autorizaciju.

Akteri koji iniciraju slučaj korišćenja: Korisnik, Administrator, Organizator, Muzički izvođači.

# 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.

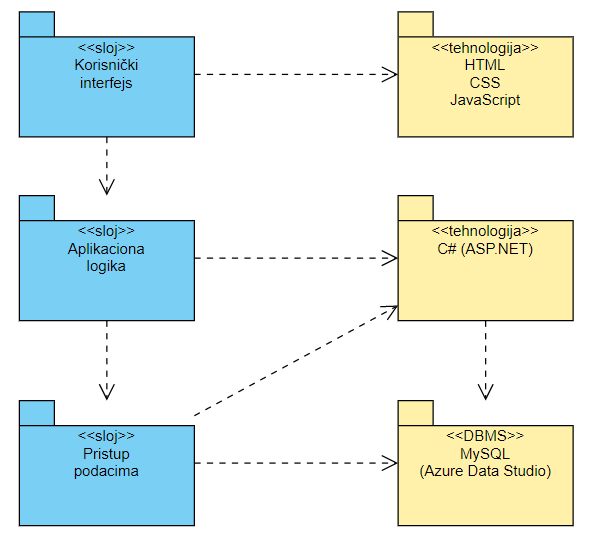
Logički pogled na NightEvents aplikaciju obuhvata 3 glavna paketa: Korisnički interfejs, Aplikaciona logika, Pristup podacima.

Paket *Korisnički interfejs* sadrži Web stranice HTML/CSS , Bootstrap.

Paket *Aplikaciona logika* predstavlja srednji sloj sistema koji sadrži funkcije u C# zadužene za realizaciju funkcionalnosti specifičnih za domen sistema koji se razvija.

Paket *Pristup podacima* sadrži C# funkcije 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 stranice preko kojih korisnici komuniciraju sa sistemom. Stranice su realizovane uz pomoć HTML/CSS-a, Bootstrap-a, JavaScript-a.

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

### Aplikaciona logika

sloj

Sloj aplikacione logike je srednji sloj u troslojnoj arhitekturi NightEvents aplikacije. Sadrži C# funkcije 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 C# tehnologije.

### Pristup podacima

sloj

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

### HTML, CSS, JavaScript

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.

Tehnologija CSS-a definiše izgled HTML elemenata.

Tehnologija JavaScript-a definise funkcionalnost aplikacije i interakciju sa korisnikom.

### C# (ASP.NET)

tehnologija

Tehnologija C# obezbeđuje mehanizam za pisanje i izvršavanje funkcija na strani servera. Ove funkcije će se koristiti kako za izvršavanje funkcionalnosti koje korisnik zatraži, tako i za pristup podacima iz baze podataka, njihovu izmenu i brisanje.

### MySQL (Azure Data Studio)

DBMS

MySQL predstavlja sistem za upravljanje bazama podataka koji će se koristiti za realizaciju NightEvents aplikacije.

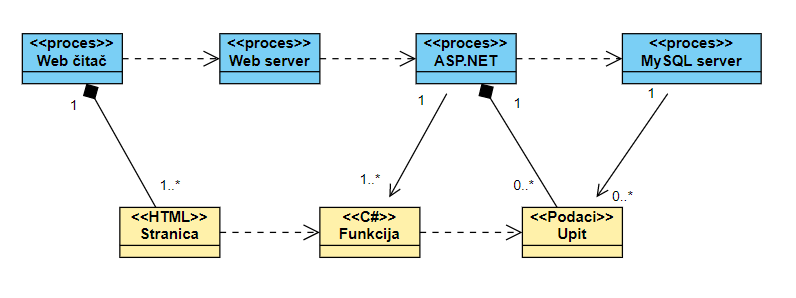
# Pogled na procese

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 NightEvents aplikacije kao Web aplikacije.

## Procesi

Na sledećem UML dijagramu klasa prikazani su procesi koji učestvuju u izvršenju NightEvents aplikacije. Dijagram je opšteg tipa i može se primeniti na bilo koju Web aplikaciju zasnovanu na C# (ASP.NET) 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.

### ASP.NET

ASP.NET obavlja posao obrade zadate preko upita i C# funkcija, koje generišu odgovarajući sadržaj koji Web server šalje Web čitaču. Za izvršavanje može da zahteva usluge MySQL server-a.

Komunikacija izmedju klijentske aplikacije i DBMS-a realizuje se uz pomoć ASP.NET i MySQL.

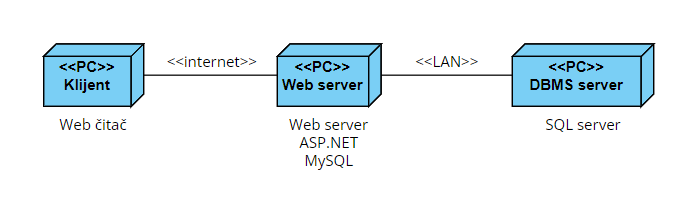
### MySQL Server

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 NightEvents aplikacije.



## Klijent

Pristup NightEvents 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. Računar će da sadrži proces koji realizuje funkcionalnost Web servera. Web server može da bude na istom računaru kao i DBMS ili može da bude povezan sa njim.

## 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).

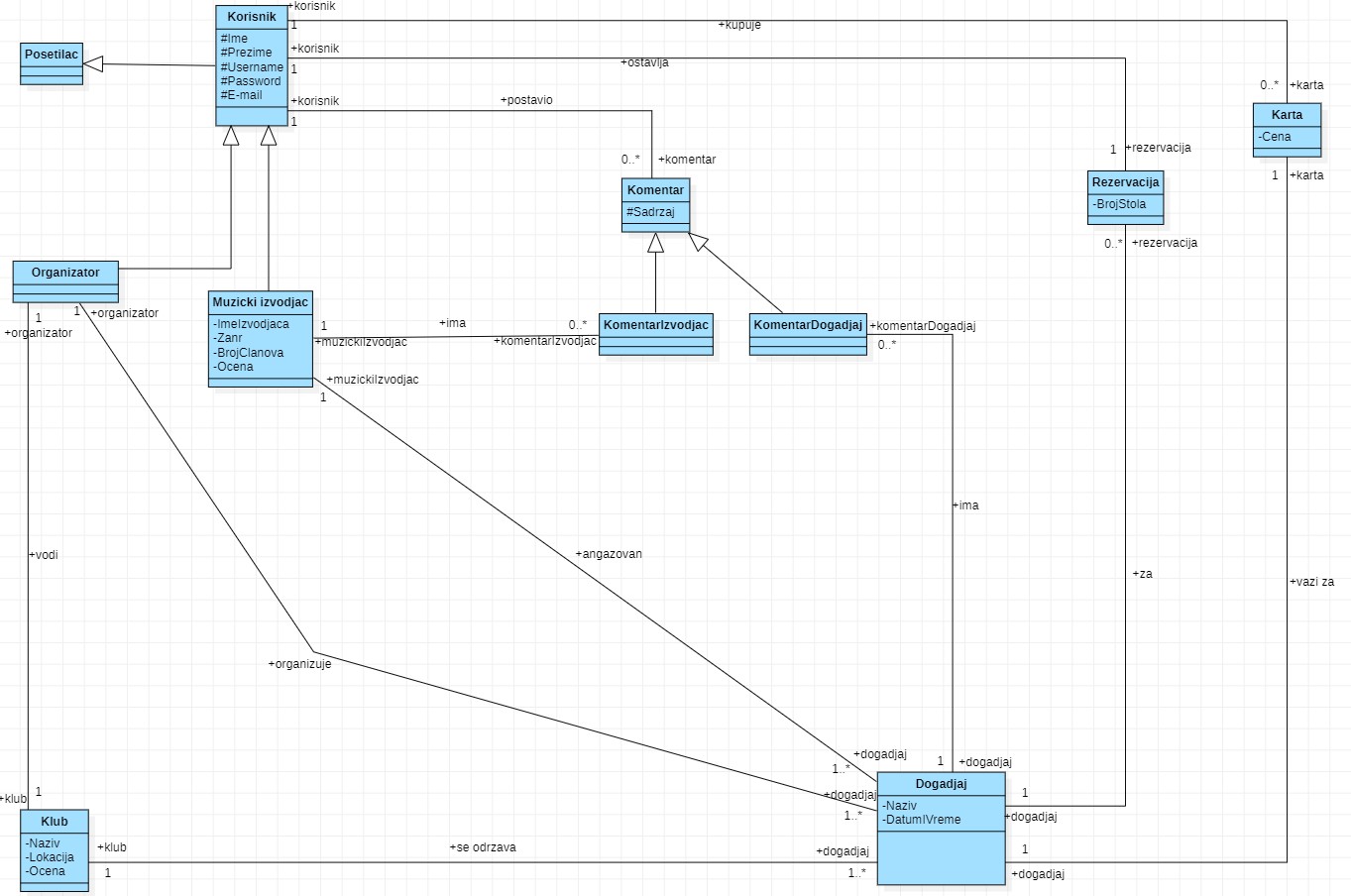
# Pogled na implementaciju sistema

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

## Model domena

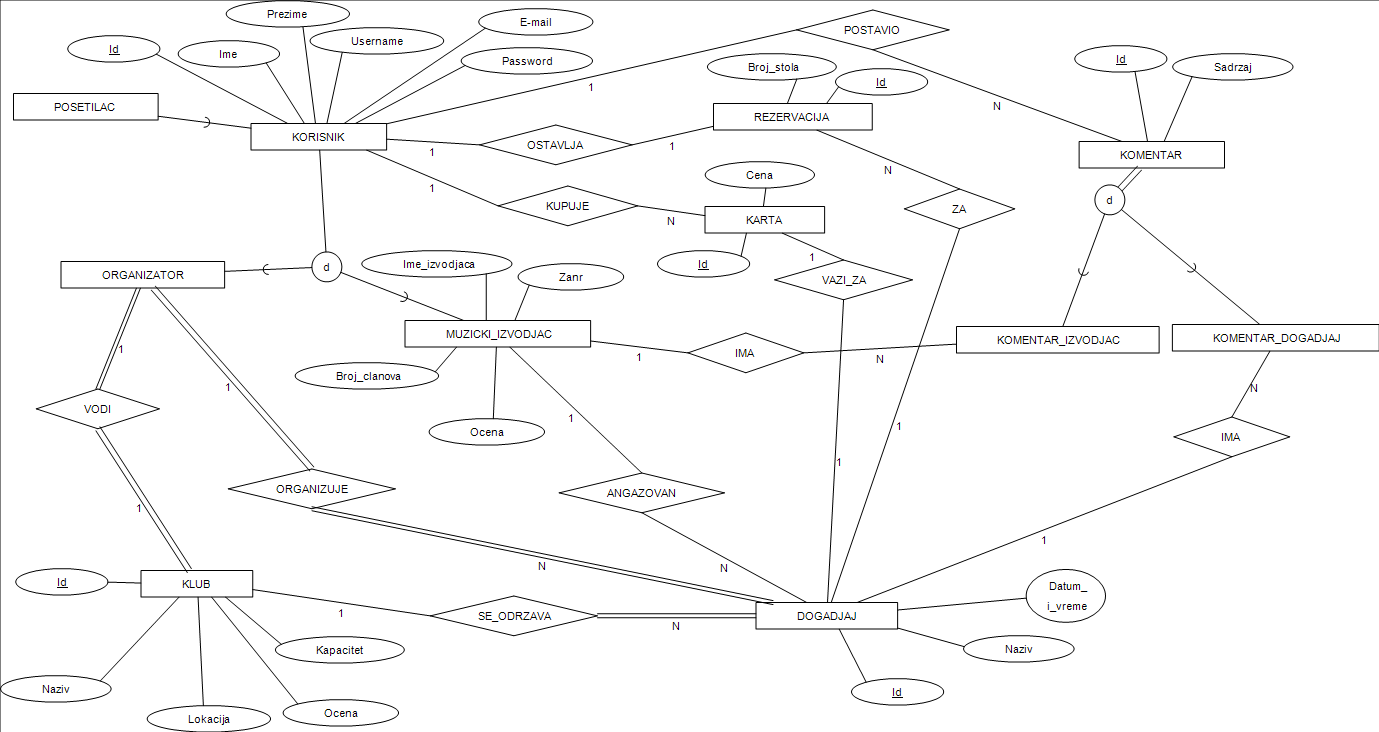
Model domena za koji se NightEvents aplikacija projektuje je ilustrovan UML dijagramom klasa. U njemu su prikazane domenske klase, neki od njihovih atributa, kao i veze koje se mogu identifikovati između njih.

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



## Šema baze podataka

Detaljna šema baze podataka je prikazana na sledećem dijagramu. Baza podataka i dijagram su kreirani korišćenjem *SharpEER programa*.

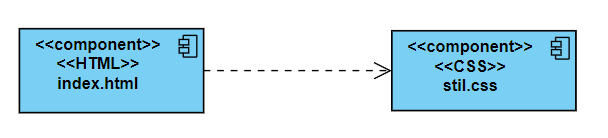


## Komponente sistema

Komponente sistema NightEvents aplikacije su JavaScript funkcije 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:

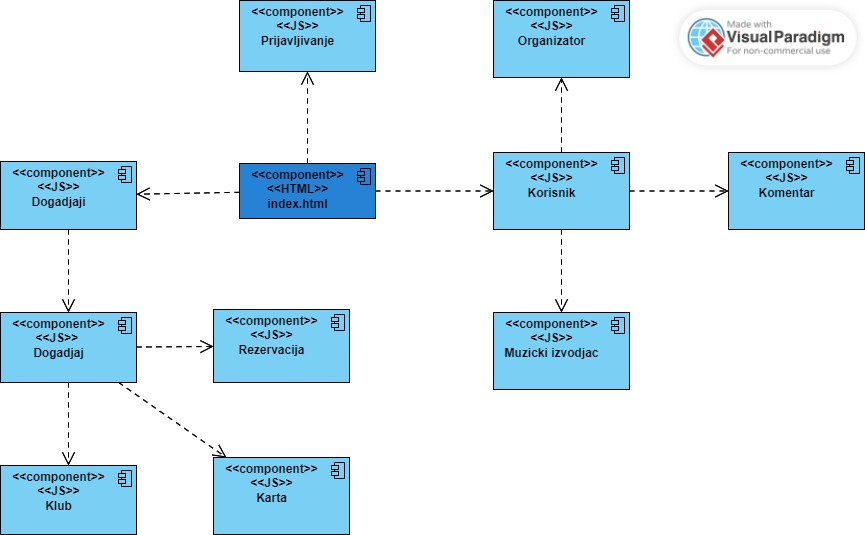


Komponenta **index.html** je implementira stranicu portala čiji sadržaj može da varira od parametra koji joj se proslede u zahtevu.

Komponenta **stil.css** predstavlja opis stilova za pojedine HTML elemente koji se javljaju na različitim stranicama.

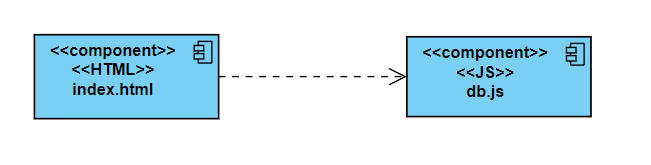
### Komponente aplikacione logike

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:

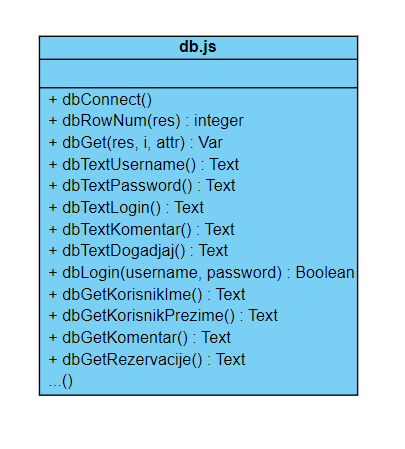


### Komponente za pristup podacima

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.html-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 true ili false
* dbGetKorisnikIme – vraća ime korisnika
* dbGetKorisnikPrezime – vraća prezime korisnika
* dbGetKomentar – vraća komentar
* dbGetRezervacije – vraća sve rezervacije
* ... – 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:

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.

# Kvalitet

Izabrana arhitektura softvera podržava zahteve u pogledu dostupnosti i srednjeg vremena između otkaza specificirane u zahtevima u pogledu pouzdanosti:

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