

GeekHouse
Online sajt o filmovima, serijama i video igrama

Specifikacija zahteva

Verzija 1.0

GeekHouse	Verzija:1.0
Detaljni arhitekturni projekat	Datum: 07. decembar 2019.
CHAOS-GeekHouse-05	

Pregled izmena

Datum	Verzija	Opis	Autor
07. decembar 2019.	1.0	Inicijalna verzija	Adem Mulić
			Ena Trtovac

GeekHouse	Verzija:1.0
Detaljni arhitekturni projekat	Datum: 07. decembar 2019.
CHAOS-GeekHouse-05	

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	6
6.1. Dijagrami slučajeve korišćenja	6
6.1.1 Prijavljivanje na sistem.....	9
6.1.2 Prikaz članova sistema.....	9
6.1.3 Ažuriranje članova sistema	9
6.1.4 Brisanje članova sistema.....	9
6.1.5 Prikaz snimaka i pisanih članaka.....	9
6.1.6 Dodavanje video snimaka ili pisanih članaka.....	9
6.1.7 Brisanje video snimaka ili pisanih članaka	9
6.1.8 Dodavanje video igara	9
6.1.9 Dodavanje filmova.....	9
6.1.10 Dodavanje serija	9
6.1.11 Brisanje video igara	10
6.1.12 Brisanje filmova	10
6.1.13 Brisanje serija.....	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.....	12
7.1.5 PHP.....	12
7.1.6 MySQL.....	12
8. Pogled na procese	12
8.1. Procesi	12
8.1.1 Web čitač.....	13
8.1.2 Web server	13
8.1.2 MySQL	13
9. Pogled na raspoređivanje sistema	13

GeekHouse	Verzija:1.0
Detaljni arhitekturni projekat	Datum: 07. decembar 2019.
CHAOS-GeekHouse-05	

9.1.	Klijent.....	14
9.2	Web server.....	14
9.3	DBMS server	14
10.	Pogled na implementaciju sistema	14
10.1.	Model domena.....	14
10.2.	Šema baze podataka.....	14
10.3.	Komponente sistema.....	15
10.1.1	Komponente korisničkog interfejsa	15
10.1.2	Komponente aplikacione logike	15
10.1.3	Komponente za pristup podacima.....	15
11.	Performanse	15
12.	Kvalitet.....	15

GeekHouse	Verzija:1.0
Detaljni arhitekturni projekat	Datum: 07. decembar 2019.
CHAOS-GeekHouse-05	

Detaljni arhitekturni projekat

1. Cilj dokumenta

Cilj ovog dokumenta je detaljni opis arhitekture web aplikacije GeekHouse.

2. Opseg dokumenta

Dokument se odnosi na GeekHouse aplikaciju koja će biti razvijena od strane CHAOS-a. GeekHouse je online aplikacija o filmovima, serijama i video-igrama.

3. Reference

Spisak korišćene literature:

- GeekHouse – Predlog projekta, CHAOS GeekHouse, V1.0, 2019, CHAOS.
- GeekHouse – Planirani raspored aktivnosti na projektu, V1.0, 2019, CHAOS.
- GeekHouse – Plan realizacije, CHAOS GeekHouse, V1.0, 2019, CHAOS.
- GeekHouse – Vizija sistema, V1.0, 2019, CHAOS.
- GeekHouse – Specifikacija zahteva, V1.0, 2019., CHAOS.

4. 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.

5. Ciljevi i ograničenja arhitekture

Ključni zahtevi i sistemska ograničenja koja imaju značajan uticaj na izbor arhitekture i projektovanje sistema su:

- GeekHouse će biti web aplikacija implementirana kao web stranica zasnovana na PHP programskom jeziku. Za kreiranje baza se koristi XAMPP local developmet server.
- GeekHouse web stranici će svako imati pristup usled postojanja internet veze i korišćenjem odgovarajuće URL adrese.
- Svi zahtevi u pogledu performansi dati u dokumentu [5] moraju biti uzeti u obzir pri izboru arhitekture i razvoju sistema.

GeekHouse	Verzija:1.0
Detaljni arhitekturni projekat	Datum: 07. decembar 2019.
CHAOS-GeekHouse-05	

6. Pogled na slučajeve korišćenja

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

Slučajevi korišćenja Pharmacy Software aplikacije su:

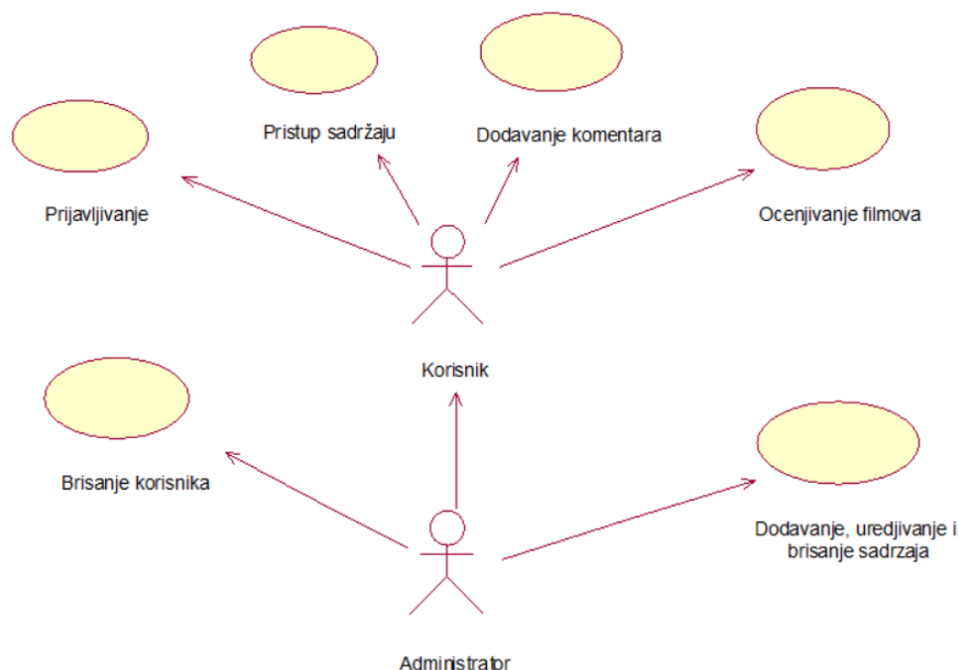
- ✓ Prijavljivanje na sistem;
- ✓ Prikaz, ažuriranje, brisanje članova sistema;
 - ✓ Prikaz članova sistema;
 - ✓ Ažuriranje članova sistema;
 - ✓ Brisanje članova sistema;
- ✓ Prikaz, dodavanje, brisanje video snimaka ;
 - ✓ Prikaz video snimaka;
 - ✓ Dodavanje video snimaka;
 - ✓ Brisanje video snimaka;
- ✓ Prikaz, dodavanje, brisanje pisanih članaka;
 - ✓ Prikaz pisanih članaka;
 - ✓ Dodavanje pisanih članaka;
 - ✓ Brisanje pisanih članaka;
- ✓ Prikaz, dodavanje, brisanje komentara;
 - ✓ Prikaz komentara;
 - ✓ Dodavanje komentara;
 - ✓ Brisanje komentara;
- ✓ Prikaz, dodavanje, brisanje sadržaja;
 - ✓ Prikaz sadržaja;
 - ✓ Dodavanje sadržaja;
 - ✓ Brisanje sadržaja;
- ✓ Pretraga sadržaja web stranice;

Ove slučajeve korišćenja mogu da iniciraju korisnici sistema (Administrator, korisnik).

6.1. Dijagrami slučajeva korišćenja

Na slici 6.1.1 prikazan je dijagram slučajeva korišćenja i aktera.

GeekHouse	Verzija:1.0
Detaljni arhitekturni projekat	Datum: 07. decembar 2019.
CHAOS-GeekHouse-05	

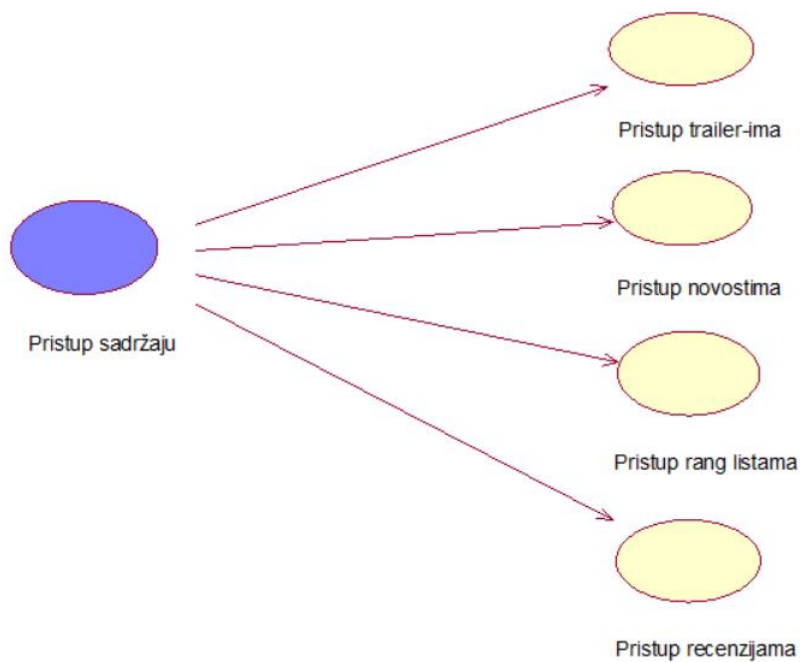


Slika 6.1.1 Use Case dijagram Pharmacy Software aplikacije

Slučaj korišćenja **Pristup sadržaju** obuhvata složenije radnje koje se mogu dalje razložiti na pojedinačne slučajeve korišćenja.

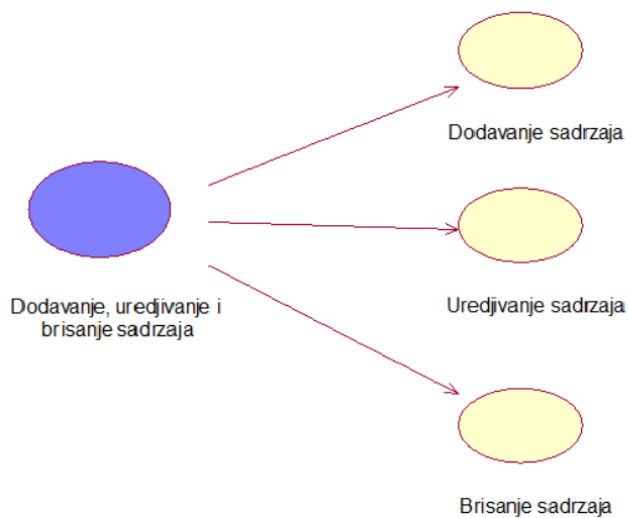
Detaljni UML dijagram za slučaj korišćenja **Pristup sadržaju** prikazan je na sledećoj slici:

GeekHouse	Verzija:1.0
Detaljni arhitekturni projekat	Datum: 07. decembar 2019.
CHAOS-GeekHouse-05	



Slika 6.1.2 Slučaj korišćenja Dodavanje, uređivanje i brisanje sadržaja obuhvata složenije radnje koje se mogu dalje razložiti na pojedinačne slučajeve korišćenja.

Detaljni UML dijagram za slučaj korišćenja Dodavanje, uređivanje i brisanje sadržaja prikazan je na sledećoj slici:.



Kratak opis slučajeva korišćenja

GeekHouse	Verzija:1.0
Detaljni arhitekturni projekat	Datum: 07. decembar 2019.
CHAOS-GeekHouse-05	

6.1.1 Prijavljivanje na sistem

- Prijavljivanje korisnika na aplikaciju u cilju pristupa sadržaju same aplikacije
- Akteri: Administrator, korisnik.

6.1.2 Prikaz članova sistema

- Kratak opis: prikaz korisnika sistema.
- Akteri: Administrator.

6.1.3 Ažuriranje članova sistema

- Kratak opis: dodavanje administratorskih privilegija.
- Akteri: Administrator.

6.1.4 Brisanje članova sistema

- Kratak opis: brisanje korisnika sistema.
- Akteri: Administrator.

6.1.5 Prikaz snimaka i pisanih članaka

- Kratak opis: pregled video snimaka I pisanih članaka.
- Akteri: Administrator , Korisnik.

6.1.6 Dodavanje video snimaka ili pisanih članaka

- Kratak opis: dodavanje novih novih video snimaka ili pisanih članaka.
- Akteri: Administrator, Korisnik.

6.1.7 Brisanje video snimaka ili pisanih članaka

- Kratak opis: Brisanje video snimaka ili pisanih članaka.
- Akteri: Administrator.

6.1.8 Dodavanje video igara

- Kratak opis: Dodavanje video igara.
- Akteri: Administrator.

6.1.9 Dodavanje filmova

- Kratak opis: Dodavanje filmova.
- Akteri: Administrator.

6.1.10 Dodavanje serija

- Kratak opis: Dodavanje video igara.
- Akteri: Administrator.

GeekHouse	Verzija:1.0
Detaljni arhitekturni projekat	Datum: 07. decembar 2019.
CHAOS-GeekHouse-05	

6.1.11 Brisanje video igara

- Kratak opis: Dodavanje video igara.
- Akteri: Administrator.

6.1.12 Brisanje filmova

- Kratak opis: Dodavanje filmova.
- Akteri: Administrator.

6.1.13 Brisanje serija

- Kratak opis: Dodavanje serija.

6.1.14 Pretraga web stranice

- Kratak opis: Pretraga web stranice.
- Akteri: Administrator i Korisnik.

6.1.15 Uređivanje profila

- Kratak opis: Uređivanje profila.
- Akteri: Administrator I korisnik.

7. 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 GeekHouse web aplikaciju obuhvata 3 glavna paketa: Korisnički interfejs, Aplikaciona logika, Pristup podacima.

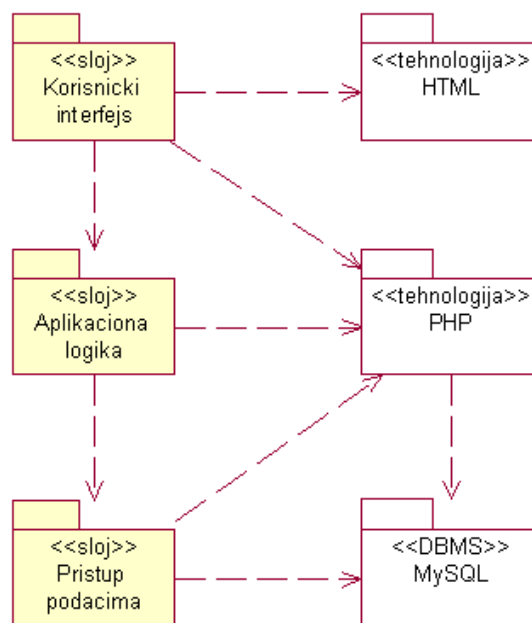
Paket Korisnički interfejs sadrži Web stranice, PHP skripte i multimedijalni sadržaj koji realizuju grafički dizajn i forme preko kojih korisnici sistema komuniciraju sa sistemom.

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.

7.1. Pregled arhitekture—organizacija paketa i podsistema u slojeve

GeekHouse	Verzija:1.0
Detaljni arhitekturni projekat	Datum: 07. decembar 2019.
CHAOS-GeekHouse-05	



Slika 7.1 Organizacija paketa i podsistema u slojeve

7.1.1 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.

7.1.2 Aplikaciona logika

sloj

Sloj aplikacione logike je srednji sloj u troslojnoj arhitekturi GeekHouse web aplikacije. 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.

7.1.3 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 zavisian od paketa PHP i MySQL baza podataka.

GeekHouse	Verzija:1.0
Detaljni arhitekturni projekat	Datum: 07. decembar 2019.
CHAOS-GeekHouse-05	

7.1.4 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.

7.1.5 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.

7.1.6 MySQL

Alat za kreiranje baze podataka

DBMS

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

8. 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.

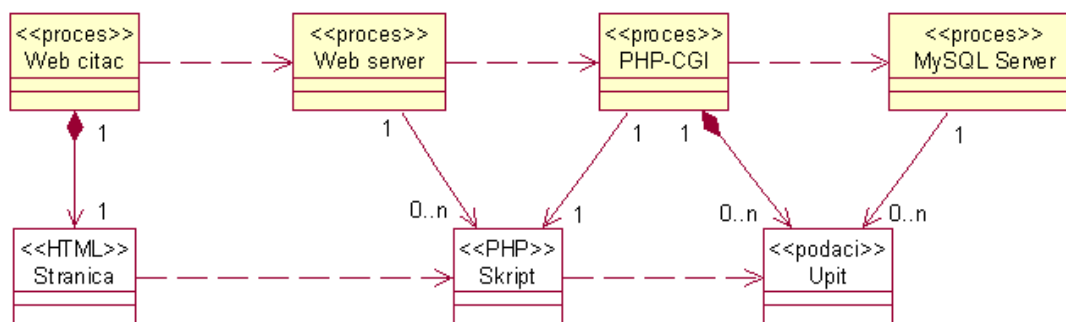
Web aplikacije zasnovane na PHP-u imaju relativno jednostavan procesni model koji je u potpunosti pod kontrolom Web servera. Sa stanovišta projektanta PHP Web aplikacije nije potrebno voditi računa o načinu rada Web servera i načinu izvršavanja skripti.

Ilustracije radi u nastavku je dat opis procesa uključenih u izvršenje GeekHouse-a kao Web aplikacije.

8.1. Procesi

Na sledećem UML dijagramu klasa prikazani su procesi koji učestvuju u izvršenju GeekHouse web aplikacije.

GeekHouse	Verzija:1.0
Detaljni arhitekturni projekat	Datum: 07. decembar 2019.
CHAOS-GeekHouse-05	



Slika 8.1.1 Procesi koji učestvuju u izvršenju GeekHouse web aplikacije

8.1.1 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.

8.1.2 Web server

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

8.1.2 MySQL

proces

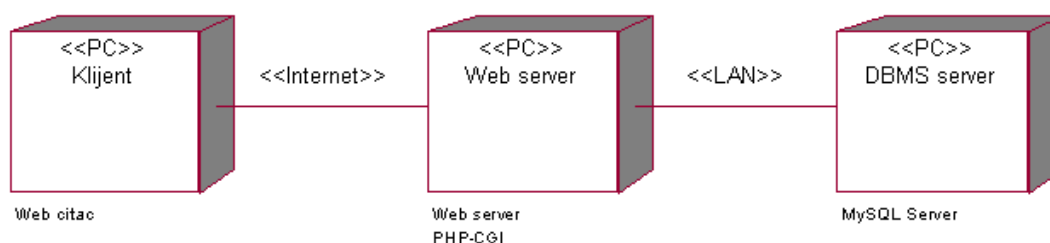
SQL (*Structured Query Language*) je relacioni upitni jezik. Relacije se kreiraju jednom naredbom i odmah su dostupne, što ga čini jednostavnim za korišćenje. Uniforman je, jer se svi podaci i rezultati operacija prikazuju u vidu tabele i omogućava interaktivno i klasično programiranje.

9. 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 Pharmacy Software aplikacije.

GeekHouse	Verzija:1.0
Detaljni arhitekturni projekat	Datum: 07. decembar 2019.
CHAOS-GeekHouse-05	



Slika 9.1 Pogled na raspoređivanje sistema

9.1. Klijent

Pristup GeekHouse web stranici 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.

9.2 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 PHP-CGI koji vrše obradu zadatih PHP skripti. U najopštijoj konfiguraciji DBMS se izvršava na posebnoj mašini koja je sa Web serverom u lokalnoj mreži (LAN).

9.3 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).

10. Pogled na implementaciju sistema

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

10.1. Model domena

Model domena za koji se Parking Software sistem 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

10.2. Šema baze podataka

Detaljna šema baze podataka je prikazana je na sledećem dijagramu. Dijagram je kreiran korišćenjem MySQLWorkbench-a.

GeekHouse	Verzija:1.0
Detaljni arhitekturni projekat	Datum: 07. decembar 2019.
CHAOS-GeekHouse-05	

10.3. Komponente sistema

10.1.1 Komponente korisničkog interfejsa

10.1.2 Komponente aplikacione logike

10.1.3 Komponente za pristup podacima

11. Performanse

Izabrana arhitektura softvera podržava zahteve u pogledu broja korisnika koji mogu pristupati sistemu i vremena odziva za pristup bazi podataka specificirane u zahtevima u pogledu performansi [5]:

- Vreme potrebno za pristupanje bazi podataka u cilju izvršenje nekog upita ne sme da bude veće od 5 sekundi.
- Broj korisnika je ograničen u zavisnosti od potreba vlasnika. Vlasnik će voditi evidenciju o broju korisnika sistema.

Zahtevane performanse su zadovoljene izborom tehnologija na kojima će sistem biti razvijen i definisane hardverske platforme [5].

12. Kvalitet

Izabrana arhitektura softvera podržava zahteve u pogledu dostupnosti i srednjeg vremena između otkaza specificirane u zahtevima u pogledu pouzdanosti [5]:

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