WV

Wedding venue

Arhitekturni projekat

Verzija 1.0



Pregled izmena

|  |  |  |  |
| --- | --- | --- | --- |
| **Datum** | **Verzija** | **Opis** | **Autor** |
| 18.04.2019. | 1.0 | Inicijalna verzija | Stefan,Jovana,Pavle |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

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 Potvrđivanje dolaska 8

6.2.2 Poziv konobara 8

6.2.3 Izbor menija 8

6.2.4 Prijavljivanje 9

6.2.5 Uvid u dolazne pozive 9

6.2.6 Uvid u listu poručenih menija 9

6.2.7 Pronalazak mesta pristiglog gosta 9

6.2.8 Generisanje računa proslave 9

6.2.9 Generisanje rasporeda sedenja 9

6.2.10 Prikaz rasporeda stolova i gostiju 9

6.2.11 Unos, prikaz i ažuriranje podataka o proslavama 9

6.2.12 Dodavanje porodice 9

6.2.13 Dodavanje člana porodice 9

6.2.14 Ažuriranje podataka o porodici 9

6.2.15 Ažuriranje podataka o konkretnom članu porodice 10

6.2.16 Brisanje cele porodice 10

6.2.17 Brisanje konkretnog člana porodice 10

6.2.18 Kreiranje, arhiviranje i brisanje zaposlenih 10

6.2.19 Uvid u slobodne dane 10

6.2.20 Izbor slobodnog dana 10

6.2.21 Rezervisanje datume 10

6.2.22 Unos broja gostiju 10

6.2.23 Unos osnovnih podataka o mladencima 10

6.2.24 Odabir rasporeda stolova 10

6.2.25 Brisanje proslave 10

6.2.26 Unos, prikaz i ažuriranje podataka od važnosti 11

7. Pogled na logičku arhitekturu sistema 11

7.1 Pregled arhitekture – organizacija paketa i podsistema u slojeve 12

7.1.1 Korisnički interfejs 12

7.1.2 Aplikaciona logika 12

7.1.3 Pristup podacima 12

7.1.4 React-Native-JSX 13

7.1.5 MySQL 13

8. Pogled na procese 13

8.1 Procesi 13

8.1.1 Mobilni telefon-Mobilna aplikacija 13

8.1.2 Web server 14

8.1.3 PHP-CGI 14

8.1.4 MySQL Server 14

9. Pogled na raspoređivanje sistema 14

9.1 Klijent 14

9.2 Web server 14

9.3 DBMS server 14

10. Pogled na implementaciju sistema 14

10.1 Model domena 15

10.2 Šema baze podataka 15

10.3 Komponente sistema 16

10.3.1 Komponente korisničkog interfejsa 16

10.3.2 Komponente aplikacione logike 17

10.3.3 Komponente za pristup podacima 19

11. Performanse 20

12. Kvalitet 21

Arhitekturni projekat

# Cilj dokumenta

Cilj ovog dokumenta je detaljni opis arhitekture WV mobilne aplikacije.

# Opseg dokumenta

Dokument se odnosi na WV mobilnu aplikaciju koja će biti razvijena od strane SJP tima. WV predstavlja skraćenicu za Wedding venue. Namena aplikacije je olakšavanje organizacije i delimično automatizovanje procesa organizovanja i same proslave venčanja.

# Reference

Spisak korišćene literature:

1. WV – Predlog projekta, V1.0, 2019, SJPTeam.
2. WV – Planirani raspored aktivnosti na projektu, V1.0, 2019, SJPTeam.
3. WV – Plan realizacije projekta, V1.0, 2019, SJPTeam.
4. WV – Vizija sistema, V1.0, 2019, SJPTeam.
5. WV – Specifikacija zahteva, V1.1, 2019, SJPTeam.

# 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. WV-Wedding venue će biti implementirana kao mobilna aplikacija zasnovana na JavaScript, PHP skripting jeziku i MySQL bazi podataka [4].
2. Klijentski deo WV mobilne aplikacije će biti optimizovan za sve mobilne i tablet uređaje koji su temeljeni na Android 8.0 platformi i novijim. [4].
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].

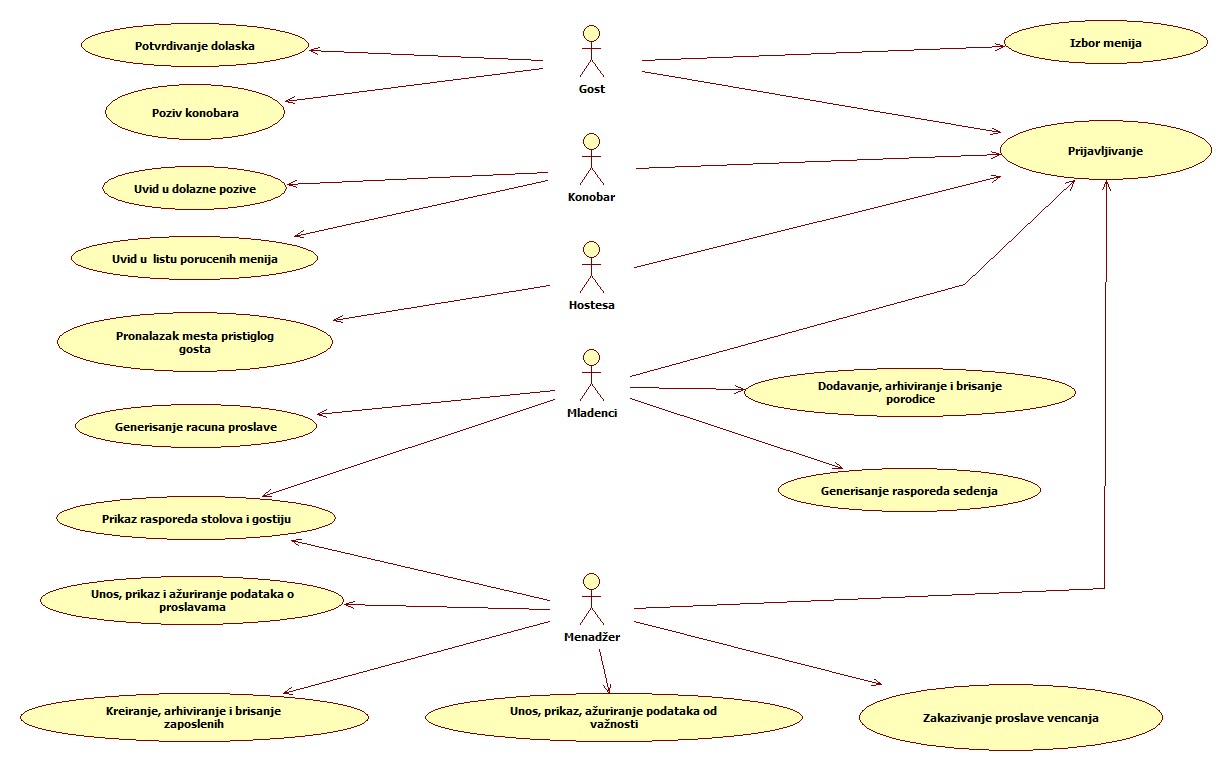
Slučajevi korišćenja WV mobilne aplikacije su:

* Potvrđivanje dolaska
* Izbor menija
* Poziv konobara
* Prijavljivanje
* Uvid u dolazne pozive
* Uvid u listu poručenih menija
* Pronalazak mesta pristiglog gosta
* Generisanje računa proslave
* Generisanje rasporeda sedenja
* Prikaz rasporeda stolova i gostiju
* Unos prikaz i ažuriranje podataka o proslavama
* Kreiranje, arhiviranje i brisanje zaposlenih
* Unos, prikaz, ažuriranje podataka od važnosti
* *Dodavanje, arhiviranje i brisanje porodice*
  + *Dodavanje porodice*
    - Dodavanje člana porodice
  + *Ažuriranje podataka o porodici*
    - Ažuriranje podataka o konkretnom članu
  + *Brisanje porodice*
    - Brisanje cele porodice
    - Brisanje konkretnog člana
* *Zakazivanje proslave venčanja*
  + Uvid u slobodne dane
  + Izbor slobodnog dana
  + *Kreiranje proslave*
    - *Detaljan unos podataka*
      * Unos broja gostiju
      * Unos osnovnih podataka o mladencima
      * Odabir rasporeda stolova
  + Brisanje proslave

Ove slučajevi korišćenja mogu da iniciraju gost na venčanju, konobar, hostesa, mladenci ili menadžer.

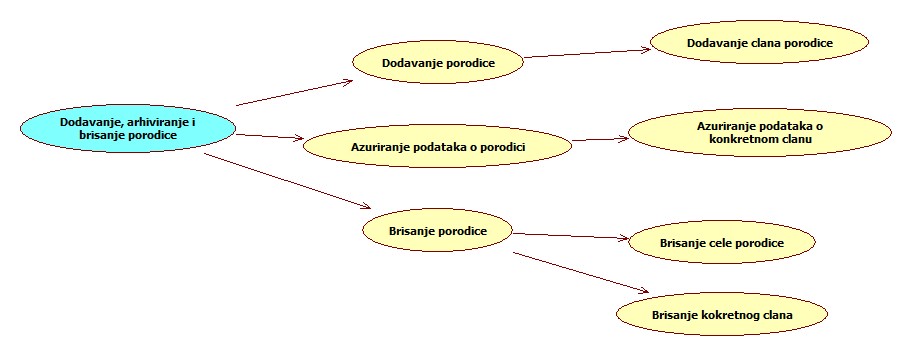
## Dijagrami slučajeva korišćenja

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

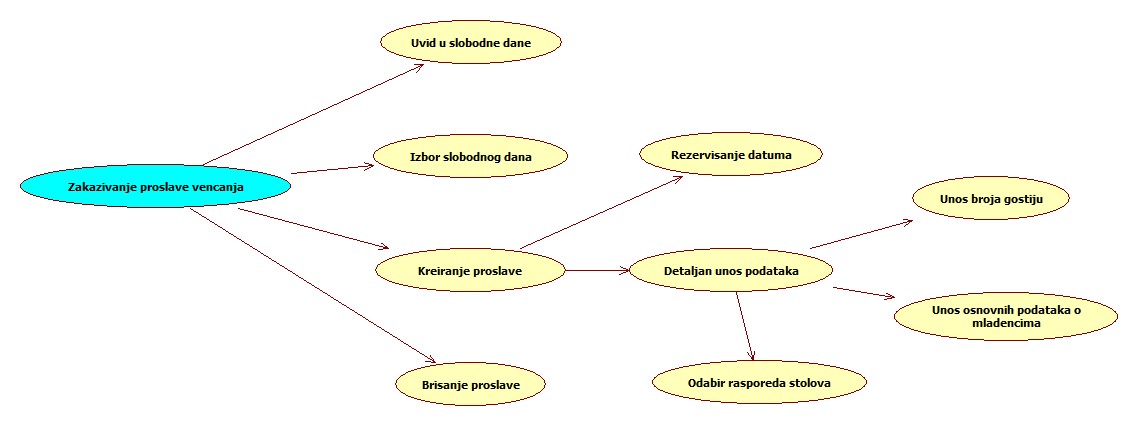


Slučajevi korišćenja *dodavanje, arhiviranje i brisanje porodice* i *zakazivanje proslave venčanj*aobuhvataju složenije radnje koje se mogu razložiti dalje razložiti na pojedinačne slučajeve korišćenja.

Detaljni UML dijagram za slučaj korišćenja *dodavanje, arhiviranje i brisanje porodice*  je prikazan na sledećoj slici:



Detaljni UML dijagram za slučaj korišćenja *zakazivanje proslave venčanja*  je prikazan na sledećoj slici:



## Kratak opis slučajeva korišćenja

### Potvrđivanje dolaska

Kratak opis: Potvrđivanje dolaska pozvanog gosta na proslavu venčanja.

Akteri koji iniciraju slučaj korišćenja: Gost .

### Poziv konobara

Kratak opis: Gost po potrebi poziva konobara.

Akteri koji iniciraju slučaj korišćenja: Gost .

### Izbor menija

Kratak opis: Gost bira meni koji mu se dopada od već ponuđenih menija.

Akteri koji iniciraju slučaj korišćenja: Gost.

### Prijavljivanje

Kratak opis: Prijavljivanje korisnika u aplikaciju u cilju pristupa funkcijama aplikacije.

Akteri koji iniciraju slučaj korišćenja: Gost , Konobar, Hostesa, Mladenci, Menadžer .

### Uvid u dolazne pozive

Kratak opis: Prikaz dolaznih poziva od strane gostiju.

Akteri koji iniciraju slučaj korišćenja: Konobar.

### Uvid u listu poručenih menija

Kratak opis: Prikaz liste poručenih menija.

Akteri koji iniciraju slučaj korišćenja: Konobar.

### Pronalazak mesta pristiglog gosta

Kratak opis: Hostesa unosom specijalnih ključeva koje identifikuju gosta pronalazi mesto u sali gde gost treba biti smešten.

Akteri koji iniciraju slučaj korišćenja: Hostesa.

### Generisanje računa proslave

Kratak opis: Generisanje računa proslave venčanja mladencima.

Akteri koji iniciraju slučaj korišćenja: Mladenci.

### Generisanje rasporeda sedenja

Kratak opis: Generisanje rasporeda sedenja gostiju.

Akteri koji iniciraju slučaj korišćenja: Mladenci.

### Prikaz rasporeda stolova i gostiju

Kratak opis: Prikaz rasporeda stolova u sali za određeni broj gostiju.

Akteri koji iniciraju slučaj korišćenja: Mladenci, Menadžer.

### Unos, prikaz i ažuriranje podataka o proslavama

Kratak opis: Unos ,prikaz i mogućnost ažuriranja podataka o proslavama.

Akteri koji iniciraju slučaj korišćenja: Menadžer.

### Dodavanje porodice

Kratak opis: Dodavanje novih porodica koji će biti pozvane.

Akteri koji iniciraju slučaj korišćenja: Mladenci.

### Dodavanje člana porodice

Kratak opis: Unos određenog člana porodice u sistem koji će biti pozvani.

Akteri koji iniciraju slučaj korišćenja: Mladenci.

### Ažuriranje podataka o porodici

Kratak opis: Ažuriranje osnovnih podataka o porodici.

Akteri koji iniciraju slučaj korišćenja: Mladenci.

### Ažuriranje podataka o konkretnom članu porodice

Kratak opis: Ažuriranje osnovnih podataka o konkretnom članu porodice.

Akteri koji iniciraju slučaj korišćenja: Mladenci.

### Brisanje cele porodice

Kratak opis: Mogućnost mladenaca da izbrišu celu porodicu iz sistema iz nekih ličnih razloga.

Akteri koji iniciraju slučaj korišćenja: Mladenci.

### Brisanje konkretnog člana porodice

Kratak opis: Brisanje konkretnog člana porodice.

Akteri koji iniciraju slučaj korišćenja: Mladenci.

### Kreiranje, arhiviranje i brisanje zaposlenih

Kratak opis: Opcija menadžera restorana da kreira naloge zaposlenim radnicima, arhivira, i brise naloge zaposlenim radnicima.

Akteri koji iniciraju slučaj korišćenja: Menadžer.

### Uvid u slobodne dane

Kratak opis: Pregled slobodnih dana za realizaciju proslave.

Akteri koji iniciraju slučaj korišćenja: Menadžer.

### Izbor slobodnog dana

Kratak opis: Izbor jednog od slobodnih dana za realizaciju venčanja.

Akteri koji iniciraju slučaj korišćenja: Menadžer.

### Rezervisanje datume

Kratak opis: Rezervisanje jednog od slobodnih dana za realizaciju venčanja.

Akteri koji iniciraju slučaj korišćenja: Menadžer.

### Unos broja gostiju

Kratak opis: Unos broja gostiju koji će biti pozvani na proslavu venčanja.

Akteri koji iniciraju slučaj korišćenja: Menadžer.

### Unos osnovnih podataka o mladencima

Kratak opis: Unos podataka od važnosti vezani za mladence.

Akteri koji iniciraju slučaj korišćenja: Menadžer.

### Odabir rasporeda stolova

Kratak opis: Izbor jednog od mogućih rasporeda stolova koji će se primenjivati na proslavi venčanja.

Akteri koji iniciraju slučaj korišćenja: Menadžer.

### Brisanje proslave

Kratak opis: Otkazivanja i brisanje proslave venčanja iz sistema.

Akteri koji iniciraju slučaj korišćenja: Menadžer.

### Unos, prikaz i ažuriranje podataka od važnosti

Kratak opis: Unos, prikaz ili ažuriranje osnovnih podataka o restoranu(cenovnik , meniji, kapacitet) .

Akteri koji iniciraju slučaj korišćenja: Menadžer.

# 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.

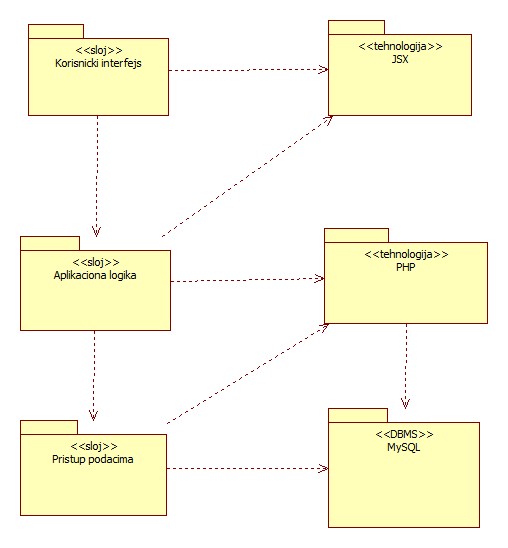
Logički pogled na WV mobilnu aplikaciju obuhvata 3 glavna paketa: Korisnički interfejs, Aplikaciona logika, Pristup podacima.

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

Paket *Aplikaciona logika* predstavlja srednji sloj sistema koji sadrži JSX skripte, 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 aplikacije WV. Korisnički interfejs je realizovan pomoću react native komponenti i JSX sintakse za formatiranje sadržaja. Konkretan stil svake komponente realizovan je pomoću CSS-a.

Sloj korisničkog interfejsa zavisi od sloja aplikacione logike.

### Aplikaciona logika

sloj

Sloj aplikacione logike je srednji sloj u troslojnoj arhitekturi WV-Wedding Venue mobilne aplikacije. Sadrži JavaScript skripte i PHP skripte koje realizuju funkcionalnost karakterističnu za domen primene mobilne aplikacije 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-Native-JSX

Tehnologija

Framework koji obezbedjuje gradivne elemente stranici kao i odgovarjući prikaz na stranici aplikacije ,što naravno uključuje i dizajn same aplikacije. Koristi se za prikaz traženih informacija kao i informacija dobijenih od servera kao i unos informacija na osnovu kojih se prave izmene u samoj bazi podataka a time i u samoj aplikaciji.JSX je novitet koji se koristi da opiše strukturu stranice (react native komponente).Omogućava kombinaciju JavaScript i HTML tag sintakse kako bi se na najbolji način opisala željena struktura stranice (komponente).

*PHP*

tehnologija

Tehnologija PHP-a obezbeđuje mehanizam za pisanje i izvršavanje skripti na strani servera. Ove skripte mogu da generišu kod sa HTML elementima čitljivim za JSX fajlove 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 WV-Wedding Venue mobilne aplikacije.

# Pogled na procese

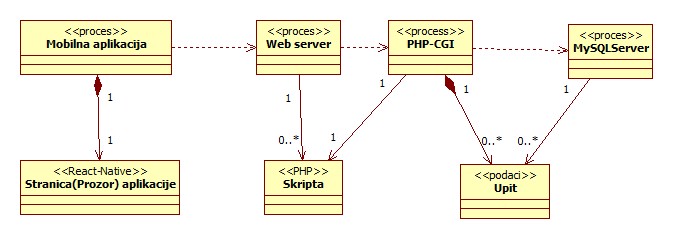
U ovom odeljku je sadržan pogled na procesnu arhitekturu sistema. Ovaj opis treba da sadrži specifikaciju različitih zadataka (procesa i niti) uključenih u rad sistema. Takođe je potrebno dati dijagrame koji pokazuju njihovu interakciju i konfiguraciju. Dodela objekata i klasa na određene zadatke takođe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

U nastavku je dat opis procesa uključenih u izvršenje WV mobilne aplikacije koji je sačinjen od Web servera ,MySQL baze podataka koji obrađuju upite i vraćaju odgovarajuće podatke za sve tipove korisnika koje na osnovu rezultata upita generišu odgovarajući prikaz.

Ilustracije radi u nastavku je dat opis procesa uključenih u izvršenje WV mobilne aplikacije.

## Procesi

Na sledećem UML dijagramu klasa prikazani su procesi koji učestvuju u izvršenju WV mobilne aplikacije. Dijagram je opšteg tipa i može se primeniti na većinu mobilnih aplikacija zasnovih na React native framework-u, JavaScript, PHP i MySQL bazi podataka.



### Mobilni telefon-Mobilna aplikacija

Mobilna aplikacija je aplikacija koja se skida i instalira sa odgovarajućih portala za odgovarajući operativni sistem ,prikazuje najviše jednu stranicu u jednom trenutku sadržaj stranice je većim delom određen podacima dobijenim od Web server-a , za rad naše aplikacije neophodan je pristup Internetu.

### Web server

Web server je proces koji izvršava funkcionalnost opsluživanja prispelih zahteva. 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.

### PHP-CGI

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

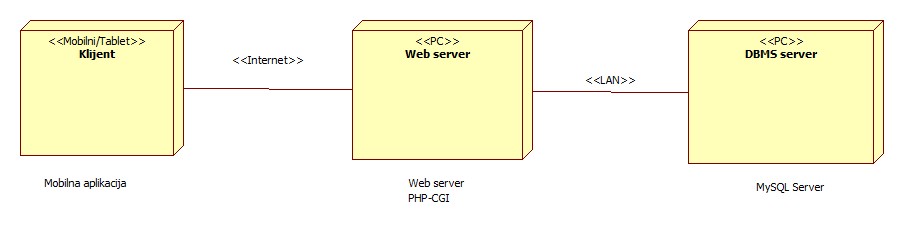
### MySQL Server

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 WV-Wedding venue mobilne aplikacije.



## Klijent

Pristup WV-Wedding venue mobilnoj aplikaciji se obavlja preko klijentskih mobilnih telefona, tableta na kojima se download-uje i izvršava aplikacija. 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 PHP-CGI koji vrše obradu zadatih PHP 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).

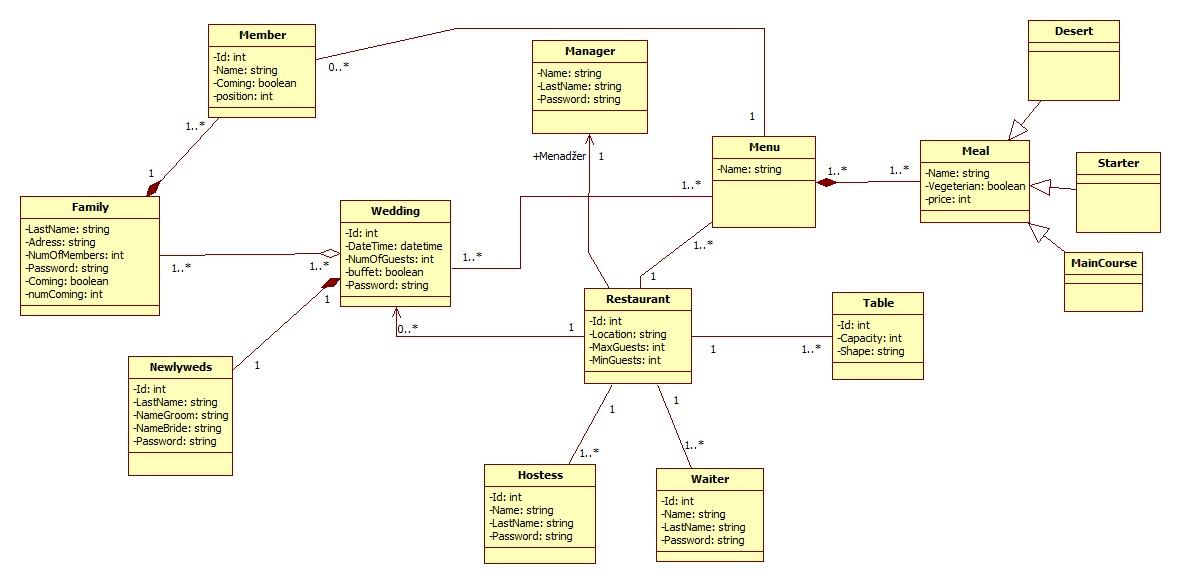
# Pogled na implementaciju sistema

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

## Model domena

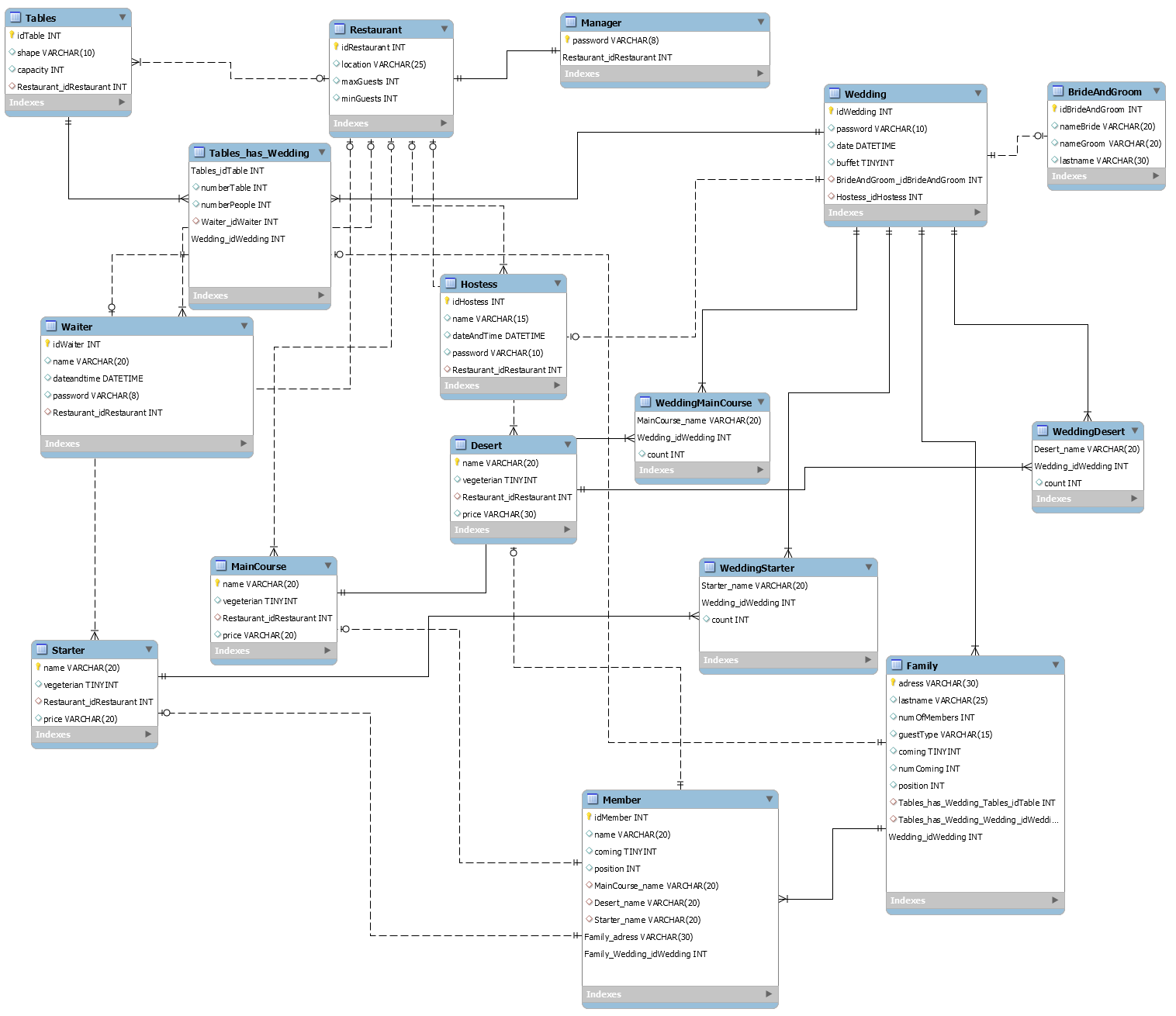
Model domena za koji se WV-Wedding Venue mobilna aplikacija projektuje je ilustrovan UML dijagramom klasa. U njemu su prikazane domenske klase, neki od njihovih atributa, kao i veze koje se mogu identifikovati između njih.

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 MySQLWorkbench-a ,čijim se eksportovanjem dobila i baza podataka koja se ubacuje na server.

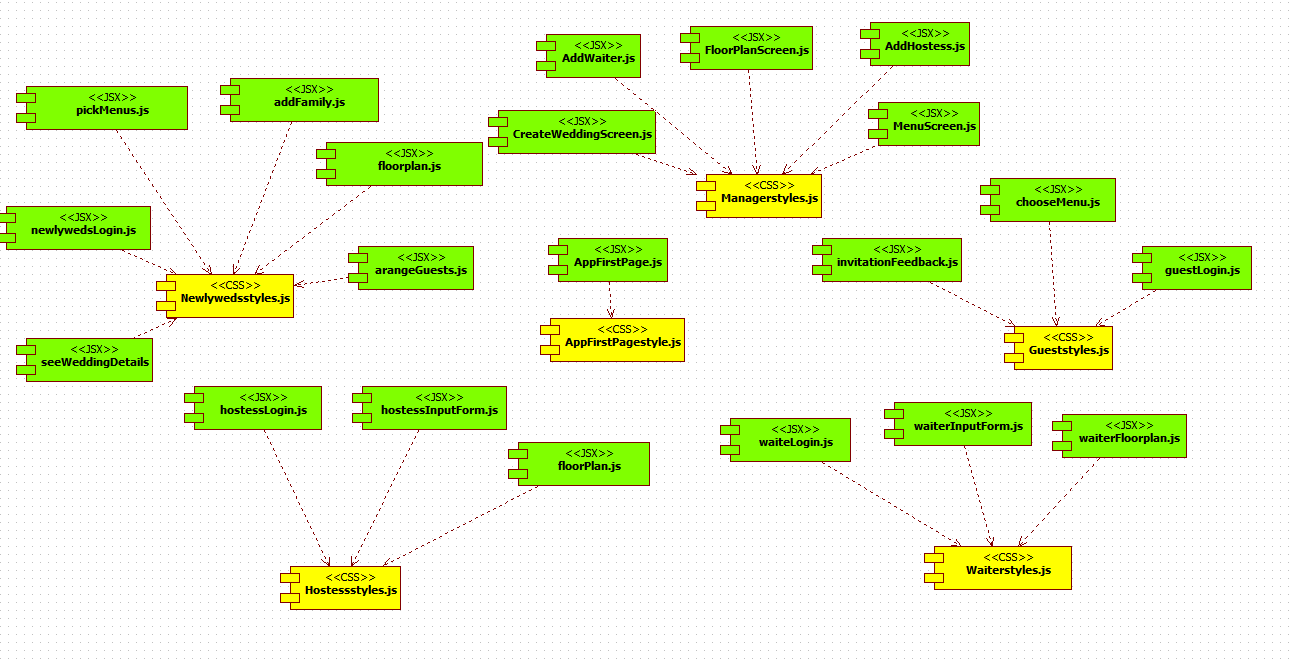


## Komponente sistema

Komponente sistema WV-Wedding Venue mobilne aplikacije su JSX, JavaScript i PHP skripte čiji će pregled biti dat po arhitekturnim slojevima. Za ilustraciju će biti korišćeni UML dijagrami komponenti, ali i dijagrami klasa.

### Komponente korisničkog interfejsa

Dizajn korisničkog interfejsa je obuhvaćen sa više komponenti:



Komponenta **Newlywedsstyle.js** predstavlja opis stilova za pojedine JSX elemente koji se javljaju na stranici koju vide mladenci .

Komponenta **Managerstyle.js** predstavlja opis stilova za pojedine JSX elemente koji se javljaju na stranici koju vidi menadžer .

Komponenta **Waiterstyle.js** predstavlja opis stilova za pojedine JSX elemente koji se javljaju na stranici

koju vide konobari .

Komponenta **Gueststyle.js** predstavlja opis stilova za pojedine JSX elemente koji se javljaju na stranici

koju vide gosti .

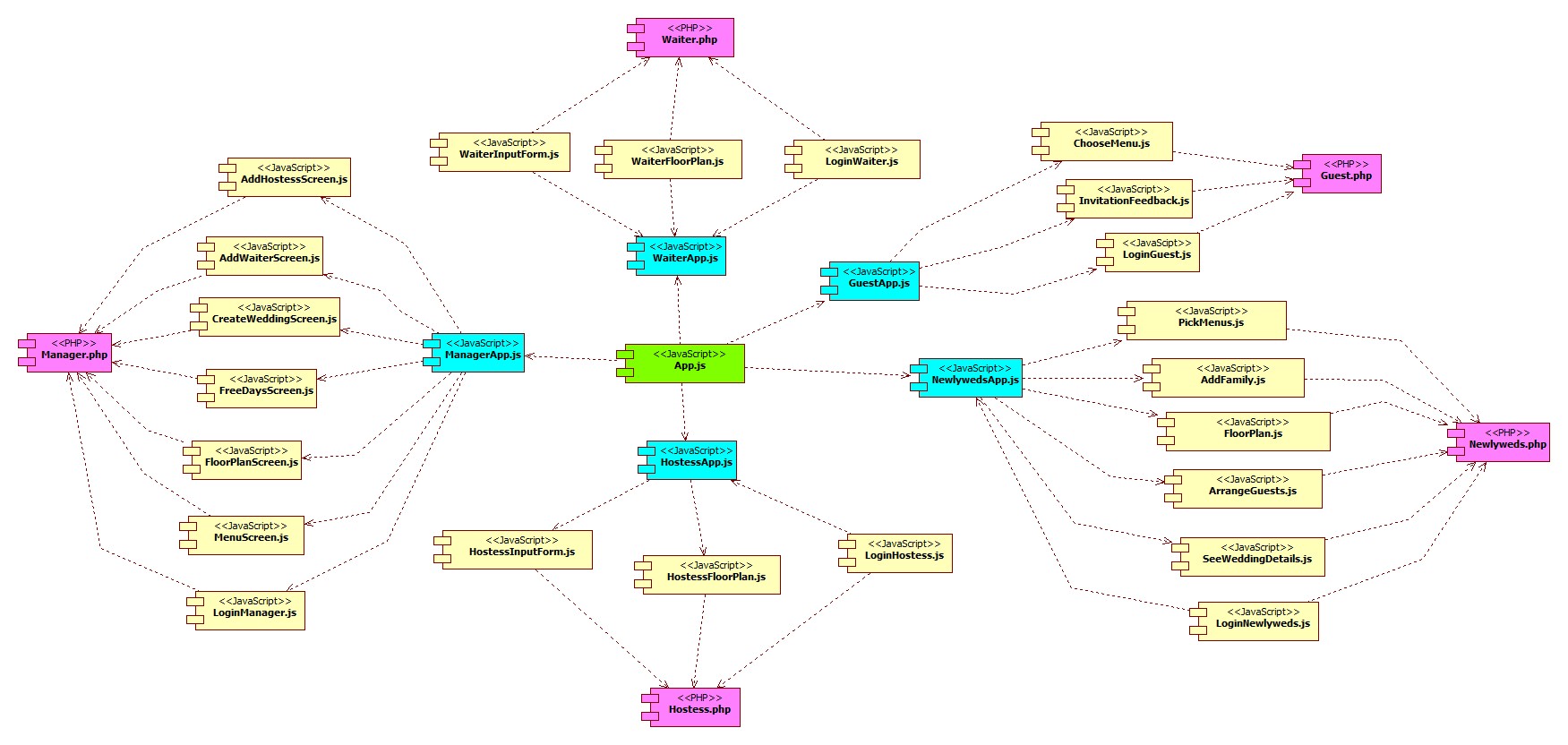
Komponenta **Hostessstyle.js** predstavlja opis stilova za pojedine JSX elemente koji se javljaju na stranici

koju vide hostese .

Komponenta **AppFirstPagestyle.js** predstavlja opis stilova za pojedine JSX elemente koji se javljaju na početnoj stranici .

### Komponente aplikacione logike

Komponente koje realizuju domen problema se uključuju isključivo preko **App.js** komponente korisničkog interfejsa. Na sledećem dijagramu su prikazane komponente ovog sloja i njihove međusobne zavisnosti:



Komponente koje realiziju domen problema menadžera se uključuju isključivo preko **ManagerApp.js .**

**AddHostessScreen.js** realizuje dodavanje hostese u sistem.

**AddWaiterScreen.js** realizuje dodavanje konobar u sistem.

**CreateWeddingScreen.js** realizuje unos osnovnih podataka o venčanju i zakazivanje.

**FreeDaysScreen.js** realizuje problem pregleda slobodnih dana za realizaciju venčanja.

**FlorPlanScreeen.js** izbor rasporeda stolova za određeni broj gostiju.

**MenuScreen.js** kreiranje, unos osnovnih podataka o menijima.

**LoginManager.j**s realizuje domen verifikacije i prijavljivanja menadžera restorana.

**Manager.php**  pruža podršku za rad sa bazom podataka funkcijama menadžera.

Komponente koje realizuju domen problema konobara se uključuju isključivo preko **WaiterApp.js.**

**LoginWaiter.js** realizuje domen verifikacije i prijavljivanja konobara restorana.

**WaiterInputForm.js**  na ovoj stranici konobar unosi podatke od značaja za posao koji obavlja.

**WaiterFloorPlan.js** realizuje prikaz rasporeda stolova, poziva sa kog stola dolaze i koje menije treba servirati.

**Waiter.php** pruža podršku za rad sa bazom podataka funkcijama konobara.

Komponente koje realiziju domen problema pozvanog gosta se uključuju isključivo preko **GuestApp.js .**

**LoginGuest.js** realizuje domen verifikacije i prijavljivanja pozvanog gosta.

**ChooseMenu.js** realizuje izbor menija koji gost želi da mu bude serviran.

**InvitationFeedback.js** pruža gostu da potvrdi ili otkaže svoj dolazak na proslavu venčanja.

**Guest.php** pruža podršku za rad sa bazom podataka funkcijama pozvanog gosta.

Komponente koje realiziju domen problema hostese se uključuju isključivo preko **HostessApp.js.**

**LoginHostess.js** realizuje domen verifikacije i prijavljivanja hostese.

**HostessInputForm.js** na ovoj stranici hostesa unosi podatke od značaja za posao koji obavlja.

**HostessFloorPlan.js** realizuje prikaz rasporeda stolova i prikaz informacija od važnosti za njen posao.

**Hostess.php** pruža podršku za rad sa bazom podataka funkcijama hostese.

Komponente koje realiziju domen problema mladenaca se uključuju isključivo preko **NewlywedsApp.js.**

**LoginNewlyweds.js** realizuje domen verifikacije i prijavljivanja mladenaca.

**PickMenus.js** realizuje izbor menija koji će se posluživati na proslavi venčanja.

**AddFamily.js** ova stranica služi za unos o porodicama i njenim članovima koji će biti pozvani na proslavu venčanja.

**FloorPlan.js** omogućuje mladencima uvid u raspored stolova u sali i korigovanje rasporeda.

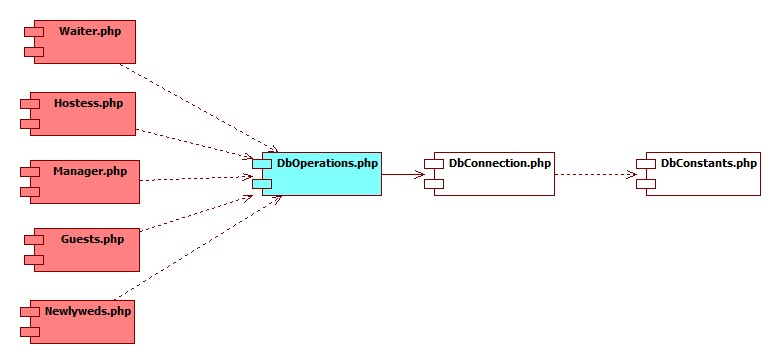
**ArrangeGuests.js** realizuje domen generisanja rasporeda sedenja gostiju.

**SeeWeddingDetails.js** realizuje pregled podataka od važnosti za mladence.

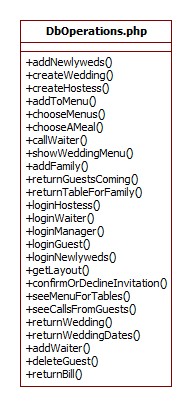
**Newlyweds.php** pruža podršku za rad sa bazom podataka funkcijama mladenaca.

### Komponente za pristup podacima

Pristup bazi podataka je u potpunosti zatvoren u funkcije koje su definisane u okviru PHP skripta **DbOperations.php**. Pomenuti skript se uključuje na početku skripte namenjene svakom korisniku tako da su sve funkcije za pristup podacima dostupne svim komponentama. Osim toga postoje posbne skripte jedna **DbConstants.php** koja sadrži konstante neophodne za povezivanje sa bazom podataka , kao i **DbConnection.php** koja na osnovu konstanti yadatih u **DBConstants.php** kreira konekciju i na osnovu te konkecije ostale skripte mogu da pozivaju odgovarajuće funkcije.



Na sledećem UML dijagramu klasa pobrojane su funkcije za pristup podacima iz baze:



Navedene funkcije obavljaju sledeće zadatke:

* addNewlyweds–dodavanje mladenaca
* createWedding– kreira venčanje
* createHostess-kreira hostesu
* addToMenu-dodavanje u meni
* chooseMenus-izbor menija
* chooseAMeal-izbor obroka
* callWaiter-pozovi konobara
* showWeddingMenu-prikaz menija svadbenog
* addFamily-dodaj porodicu
* returnGuestsComing-vrati goste koji dolaze
* returnTableForFamily-vrati sto na kome sedi porodica
* getLayout-vraćanje rasporeda(položaja stolova)
* confirmOrDeclineInvitation-potvrditi ili odbiti poziv na venčanje
* seeMenuForTables-uvid u izabrana jela sa jednog stola
* seeCallsFromGuests-videti pozive od gostiju
* returnWedding- vraćanje svih informacija o venčanju
* returnWeddingDates-vraćanje zauzetih datuma za venčanje
* addWaiter-dodavanje konobara
* deleteGuest-brisanje gosta
* returnBill-vraćanje računa

# 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 500 simultanih pristupa mobilnoj aplikaciji
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. WV mobilna 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.