Autori

Branka stanković, Filip adamović, sukara miloš, nikola blagojević

Ticketeer

Dijagram klasa

Sadržaj

[Uvod 1](#_Toc12797950)

[Dijagram klasa za rad sa podacima 3](#_Toc12797951)

[Dijagram klasa za rad sa korisničkim nalozima 4](#_Toc12797952)

[Dijagram klasa za rad sa rezervacijama 5](#_Toc12797953)

[Dijagram klasa za rad sa salama 6](#_Toc12797954)

[Dijagram klasa za rad sa događajima 8](#_Toc12797955)

# Uvod

Ticketeer je web aplikacija za rezervaciju ulaznica. Sistem je moguće prilagoditi za različite klijente i vrste događaja. U zavisnosti od naručioca sistema, kreraju se sale sa odjeljcima u odgovarajućem rasporedu i brojem mjesta. Cilj sistema je da omogući lakše, brže i efikasnije rezervisanje ulaznica za događaje. Naručioc sistema će imati lakši uvid u rezervacije i prodaju ulaznica, dok će korisnicima rezervacija ulaznica biti dostupnija.

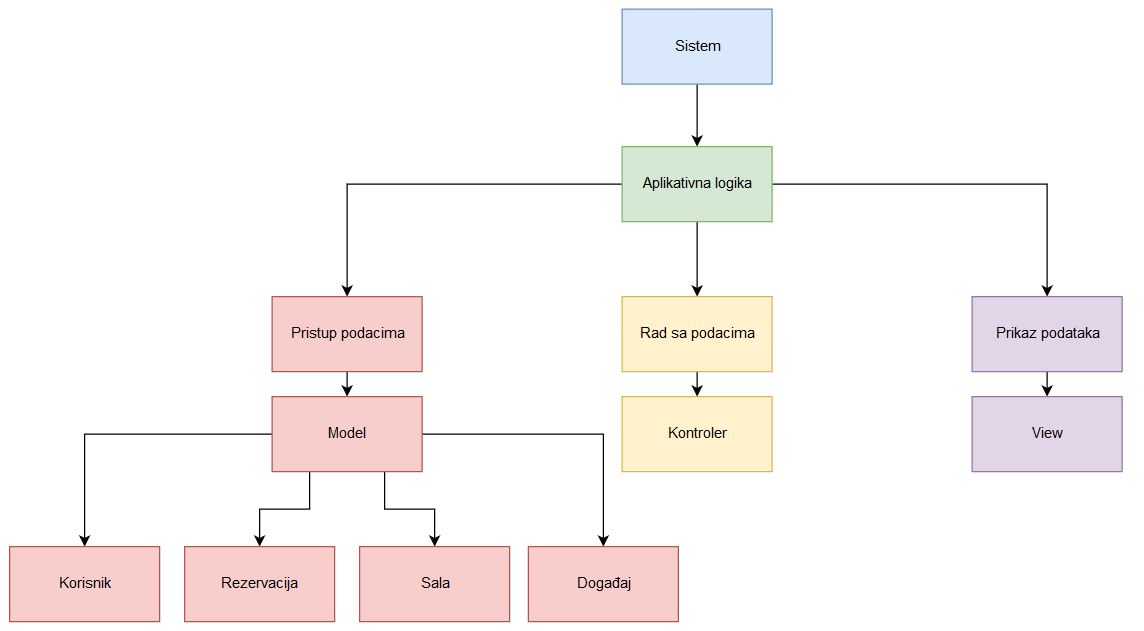
Mogućnost rezervisanja će biti dostupna korisnicima koji imaju registrovan nalog, dok će neregistrovani korisnici biti u mogućnosti da pregledaju repertoar i događaje, ali neće moći rezervisati ulaznicu.

Ticketeer je klijent-server aplikacija koja podržava veći broj funkcija za interakciju korisnika sa sistemom i odgovarajućim grafičkim korisničkim interfejsom (GUI). Ticketeer nudi ograničen broj funkcionalnosti i za korisnike koji nisu registrovani na sistem.

Ovaj softver je dizajniran tako da ispuni sledeće projektne ciljeve:

* Pouzdanost podrazumjeva da je softver u mogućnosti da izvrši traženu funkciju pod određenim uslovima u određenom vremenskom periodu.
* Ponovna upotreba
* Efikasnost se ogleda u nastojanju da se putem softvera omogući brza i laka rezervacija ulaznica.
* Prenosivost se ogleda u mogućnosti korišćenja softvera i svih njegovih funkcionalnosti na različitim uređajima i u različitim okruženjima.
* Prilagodljivost se ogleda u mogućnosti ovog softvera da se primjeni za različite naručioce i različite tipove događaja.
* Robusnost predstavlja mogućnost sistema da radi pri velikim opterećenjima ili da reaguje na nevalidne situacije.
* Laka upotreba se zasniva na jednostavnom i intuitivnom dizajnu, što omogućava korišćenje sistema na jednsotavan način.
* Lako održavanje se odnosi na mogućnost zamjene i nadogradnje hardvera i softvera.
* Niska cijena u zavisnosti od zahtjeva naručioca

Na sljedećem dijagramu je prikazana podjela sistema na logičke podsisteme po kojima su predstavljeni dijagrami klasa.

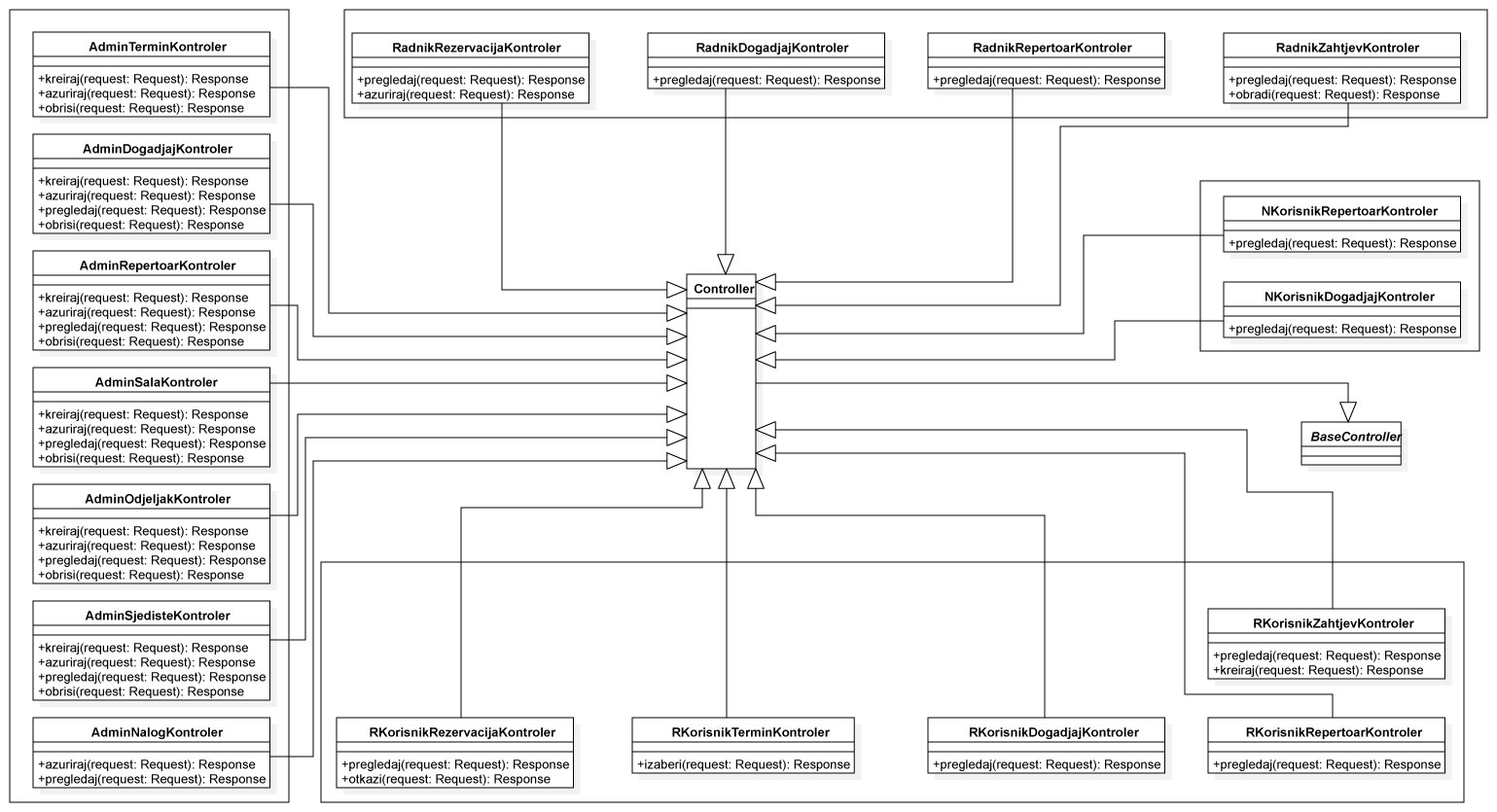


Slika Podjela sistem na logičke cjeline

Sa dijagrama (*Slika 1*) se uočava da se aplikativna logika sistema dijeli na tri dijela. Dio za pristup podacima služi kao perzistentni sloj sistema. Rad sa podacima predstavlja dio aplikacije za kontrolu podataka i vraćanje odgovarajućih prikaza korisnicima. Prikaz podataka čini skup pregleda (*view*-ova) koji predstavljaju reprezentaciju datih podataka u zavisnosti od trenutnog korisnika.

# Dijagram klasa za rad sa podacima

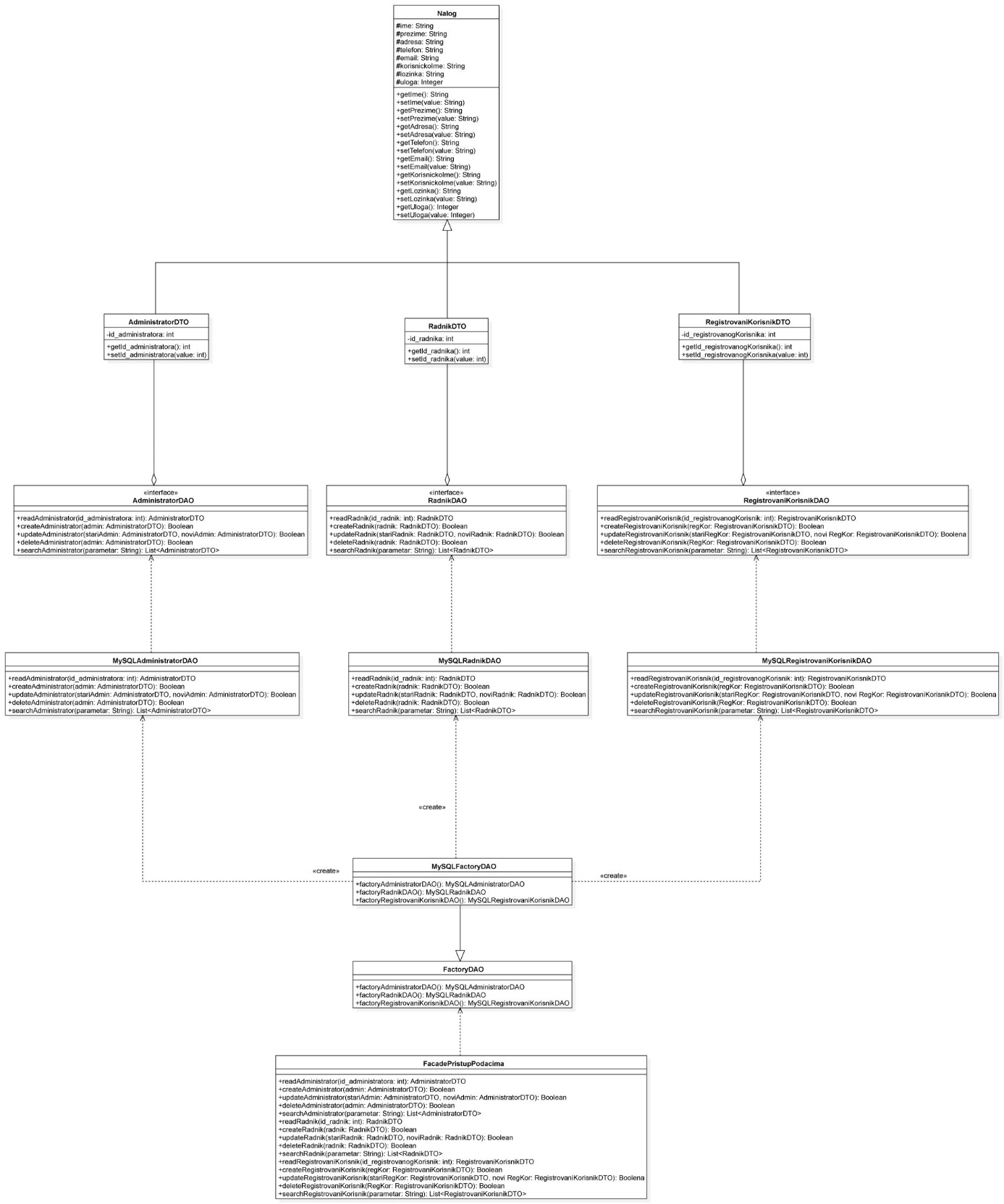
Na slici 2 je prikazan dijagram klasa dijela za rad sa podacima. Funkcionalnosti *Controller*-a i *BaseController*-a su riješeni na nivou *framework*-a. U zavisnosti od uloge korisnika sistem generiše odgovarajući kontroler. Kontroleri dalje omogućavaju određene operacije nad datim podacima. Svaka metoda svih kontrolera kao parametar medote primaju *Request* objekat koji predstavlja odgovarajući zahtjev poslan od strane korisnika aplikacije. *Response* predstavlja odgovor aplikacije na dati zahtjev, u zavisnosti da li se pristupa preko *api* ili *web* ruta, *Response* objekat će vratiti *JSON* response ili odgovarajući *View*.



Slika Rad sa podacima

# Dijagram klasa za rad sa korisničkim nalozima

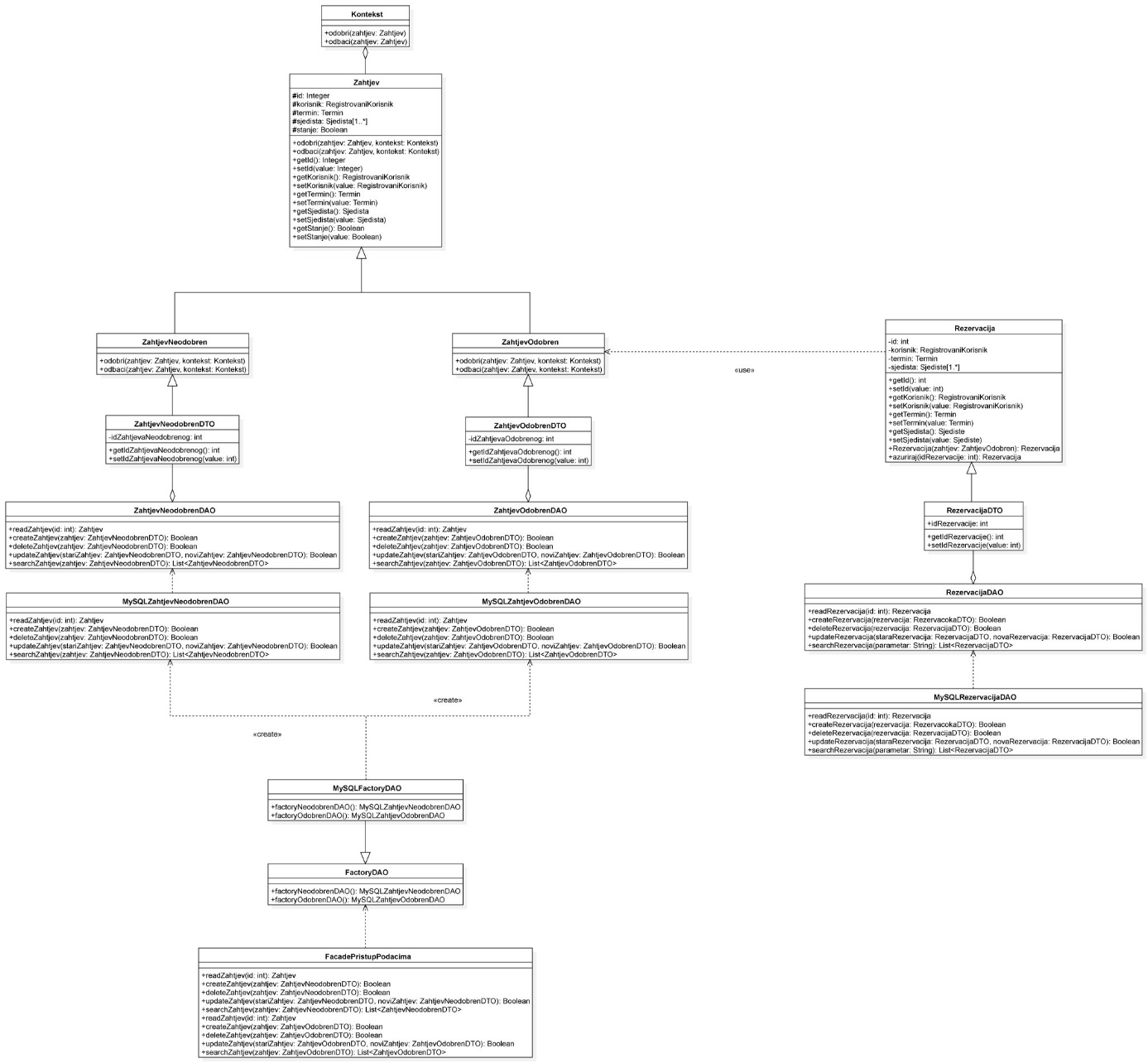
Na slici 3 je prikazan dijagram klasa za rad sa korisničkim nalozima. Na dijagramu su korišćeni projektni obrasci fasada i fabrički metod. Fasada obezbjeđuje jedinstveni interfejs za cjelokupni sistem. Fabrički metod definiše interfejs za kreiranje objekata, ali dozvoljava da izvedena klasa odluči koju će klasu da instancira.



Slika Rad sa korisničkim nalozima

# Dijagram klasa za rad sa rezervacijama

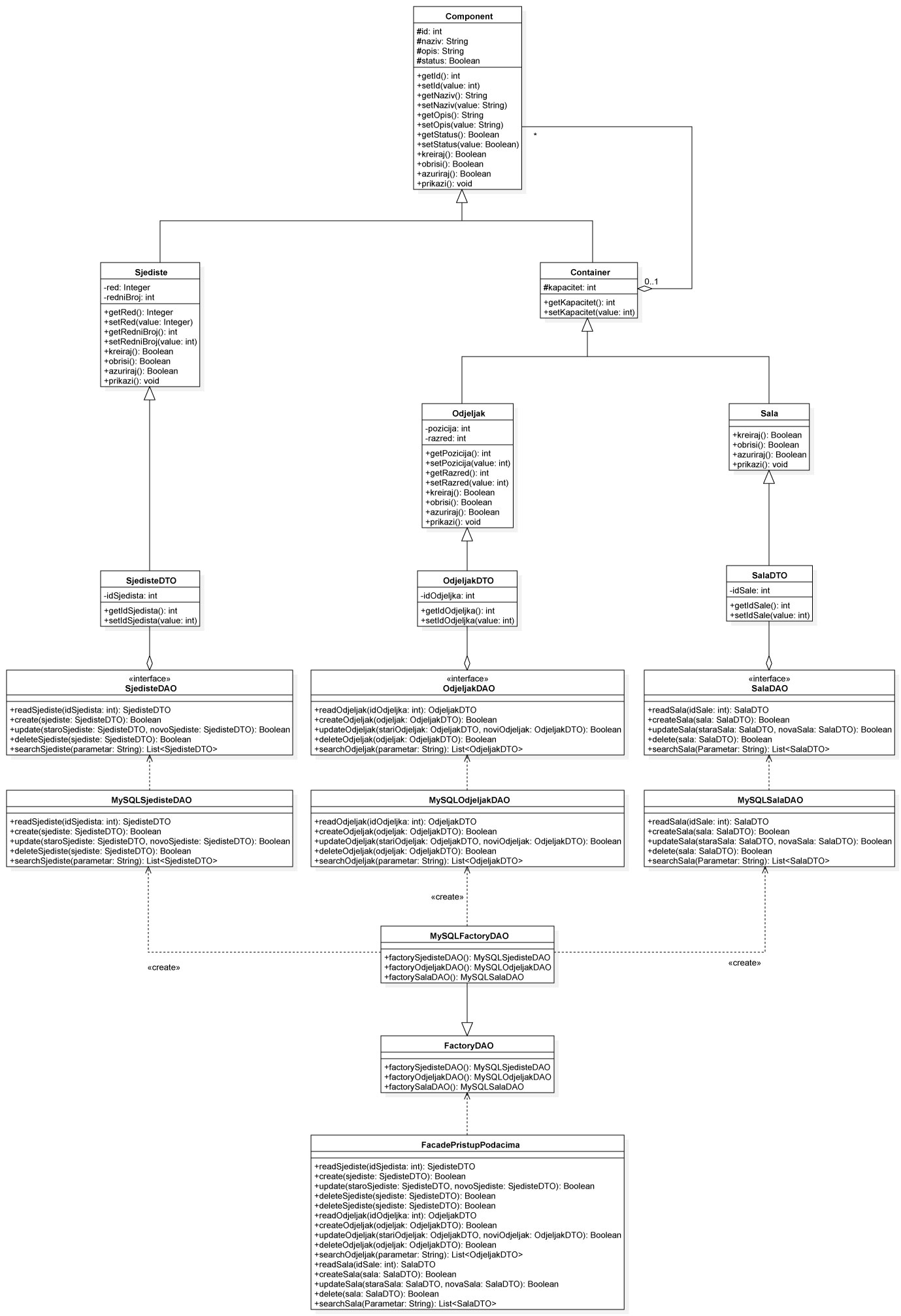
Na slici 4 je prikazan dijagram klasa za rad sa rezervacijama. Na ovom dijagramu korišćeni su projektni obrasci fasada i stanje. Projektni obrazac stanja je korišćen za prikaz zahtjeva za rezervaciju. U zavisnosti od stanja zahtjeva, moguće je kreirati odgovarajuću rezervaciju ili poslati obavještenje korisniku.



Slika Rad sa rezervacijama

# Dijagram klasa za rad sa salama

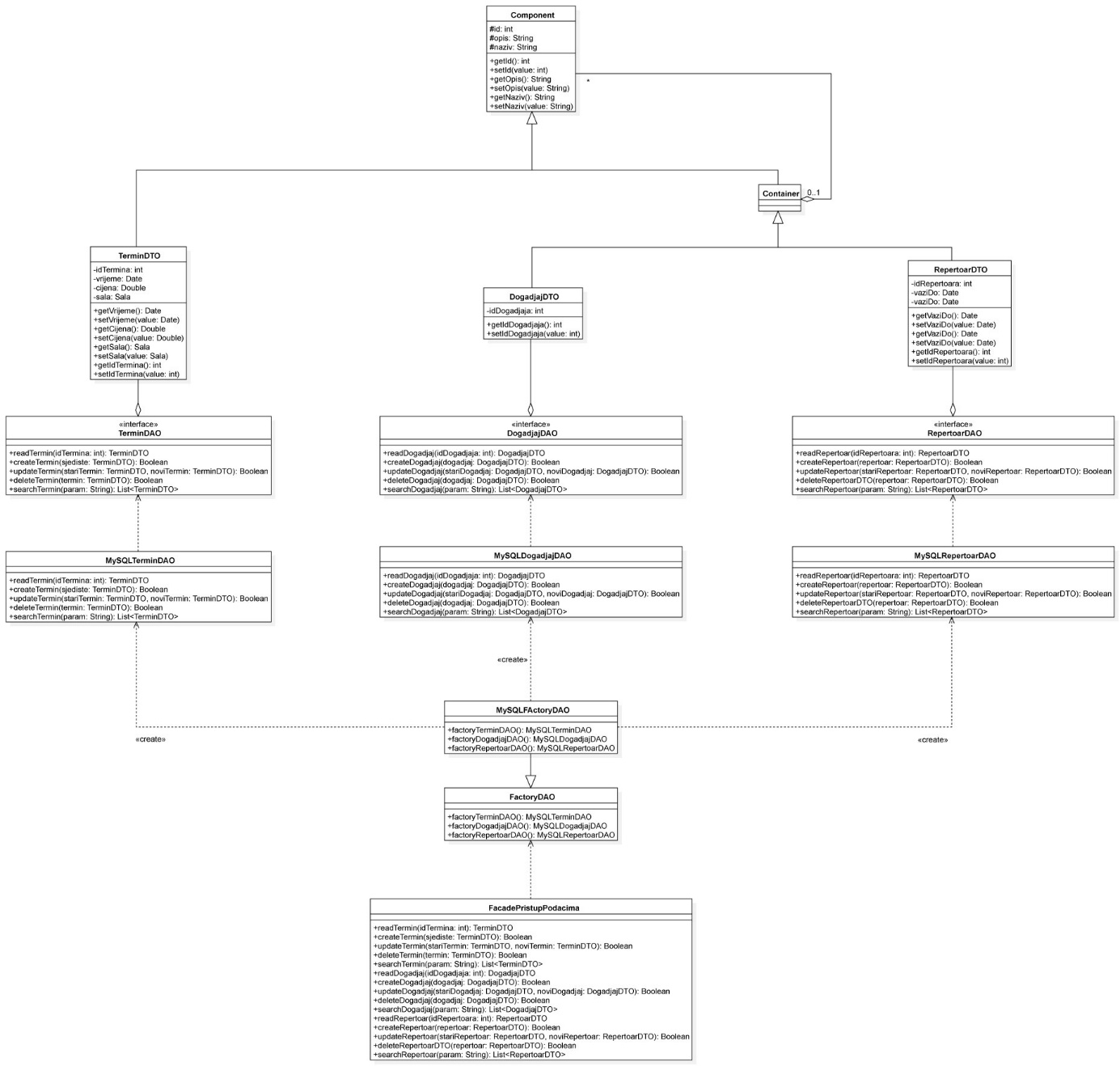
Na slici 1.5 je prikazan dijagram klasa za rad sa salama. Na dijagramu su korišćeni projektni obrasci fasada i kompozicija. Projektni obrazac kompozicije je korišćen za prikaz sala, odjeljaka i sjedišta u odjeljcima. Sale i odjeljci su generalizovani kao kontejnerske klase, dok sjedište predstavlja prostu komponentu (*Leaf*). Sale se sastoje od odjeljaka, dok se odjeljci sastoje od sjedišta.



Slika Rad sa salama

# Dijagram klasa za rad sa događajima

Na slici 1.6 je prikazan dijagram klasa za rad sa događajima. Na dijagramu su korišćeni projektni obrasci fasada i kompozicija. Projektni obrazac kompozicija je korišćen za prikaz repertoara, događaja i termina. Repertoari i događaji su generalizovani kao kontejnerske klase, dok termin predstavlja prostu komponentu (*Leaf*). Repertoari sadrže događaje, dok događaji sadrže različite termine.



Slika Rad sa događajima