



Univerzitet u Sarajevu
Elektrotehnički fakultet Sarajevo
Odsjek za računarstvo i informatiku



Deseti projektni zadatak

Objektno orijentisana analiza i dizajn

Ime i prezime: Din Švraka 18857
 Nejla Bečirspahić 18835
 Amina Pandžić 18808
Grupa: šest (6) / DNACityGuide
Datum: 03.06.2022.

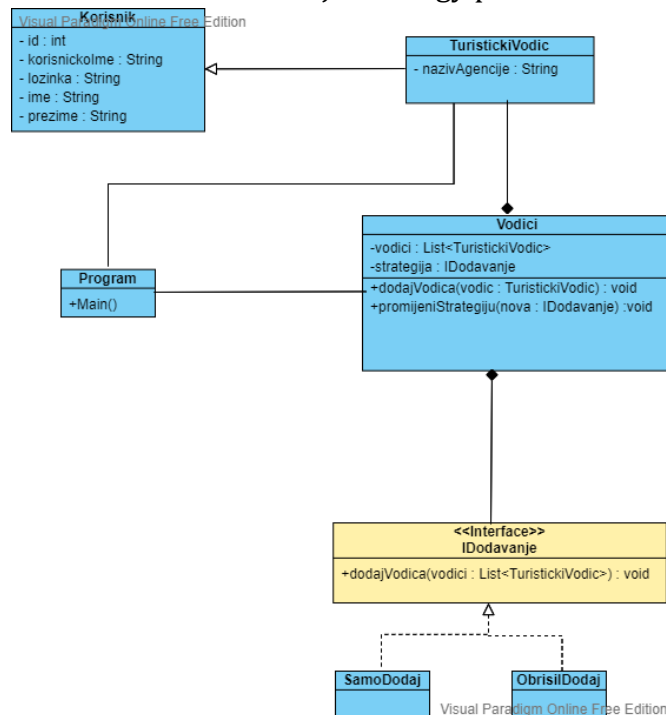
1. Paterni ponašanja

1.1. Strategy

Kada vlasnik agencije (koji je ujedno i administrator sistema) bude želio izbrisati neki od profila trenutno ponuđenih turističkih vodiča, to mjesto će se automatski osloboditi. Budući da imamo uvijek mjesta za 3 turistička vodiča, kada vlasnik bude želio dodati novog vodiča, postojat će dvije strategije. Prva strategija bi mogla biti da vodič zauzme poziciju na kojoj se prije nalazio vodič koji je izbrisan ili druga strategija da se novi vodič uvijek postavi na zadnje mjesto, a prethodni vodiči pomjere prema lijevo za jedno mjesto.



Slika 1. Ilustracija Strategy paternna



Slika 2. Strategy pattern

1.2. State

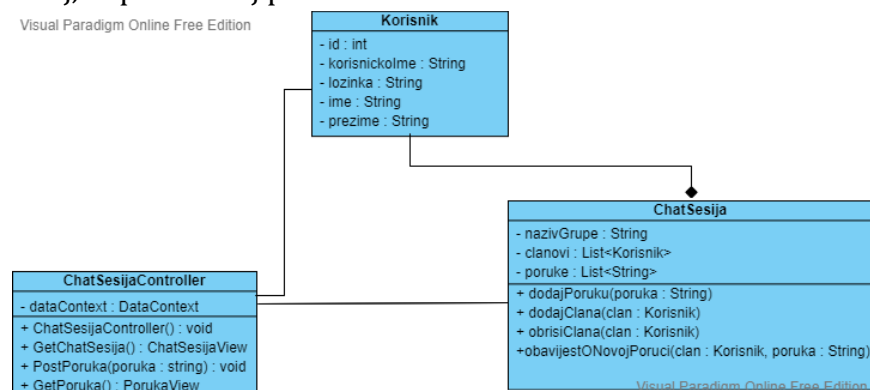
State patern je moguće primijeniti ukoliko bismo omogućili korisniku pristup aplikaciji dok je on „offline“. U tom slučaju, korisnik bi mogao samo vršiti pregled svi turističkih atrakcija, tura, vodiča, suvenira... Međutim ne bi imao mogućnost rezervacije tura, kupovine ulaznica i pristup chat sesiji. S druge strane, korisnik koji pristupa aplikaciji dok je online, ima pravo na pristup svim onim funkcionalnostima koje su u skladu sa njegovom dodijeljenom ulogom u aplikaciji, koje smo u prethodnim taskovima već detaljno razradili.

1.3. Template method

Ono što je zajedničko za turiste koji su državljani Bosne i Hercegovine i turiste strance jeste to što se oba moraju registrovati i unijeti svoje podatke. Međutim, kada se turist opredijeli za jednu od ove dvije opcije, tada se u pozadini sistema mijenja ono što je naslijeđeno, iako na prvi pogled između njih nema nikakve razlike. Sličnosti ćemo samo navesti, a na razlikama ćemo se zadržati. Neke od sličnosti su online kupovina ulaznica za posjetu određenih atrakcija i online kupovina City Pass paketa. Međutim, strani turisti imaju „extra“ mogućnost online rezervacije tura i posjete atrakcijama sa vodičima. Dakle, ovu funkcionalnost zabranjujemo u pozadini sistema domaćim, a omogućavamo stranim turistima.

1.4. Observer

Ovaj patern je izuzetno primjenjiv u našoj aplikaciji, i to u dijelu „Q&A sesija“. Naime, kada jedan korisnik pošalje poruku u određenoj ChatSesiji, svi članovi te sesije će dobiti obavijest o novoj, nepročitanoj poruci.



Slika 3. Observer pattern

1.5. Iterator

Kada vlasnik agencije želi da pregleda prijavljene komentare preko svog naloga, komentare može pregledati na više načina, odnosno komentari i prijave mogu biti poredani od najstarijeg do najnovijeg, prema abecednom redu korisnika koji su prijavili komentare, randomly poredani itd. Ovo je upravo niz primjera primjene Iterator paterna čija je osnovna logika omogućavanje prolaska kroz kolekciju objekata koja nije niz niti lista.

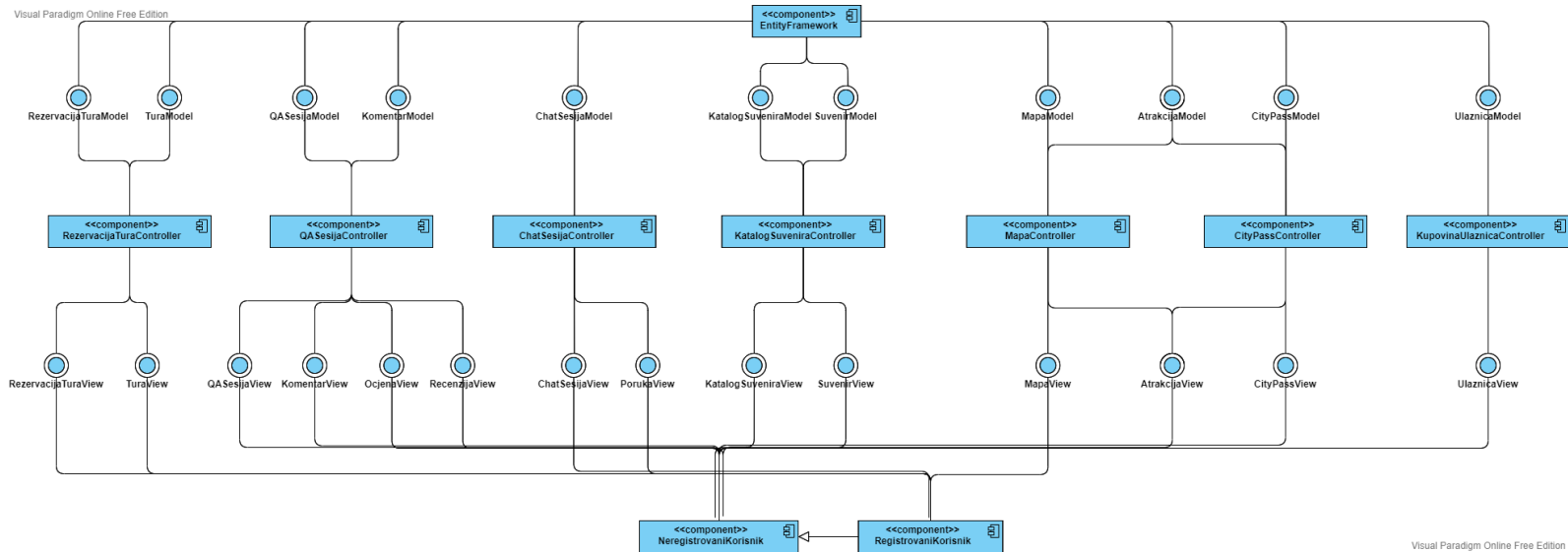
1.6. Chain of responsibility

Primjer lančanog dešavanja akcija unutar naše aplikacije je prilikom rezervacije ture za posjetu atrakcijama sa turističkim vodičima za strane turiste. Strani turist se mora prvenstveno prijaviti na aplikaciju (ili registrovati ako već nema prije napravljeni account), a nakon toga odabira opciju odabira željene ture. Nakon toga, imaće mogućnost odabira jedne od pet ponuđenih tura. Poslije izvršenog biranja ture, turist je rezerviše, dobija potvrdu o (ne)uspješnoj rezervaciji i tek tada može pristupiti u CHAT sobu sa turističkim vodičem odgovornim i zaduženim za odabranu turu. Dakle, time je ispunjena osnovna svrha Chain of responsibility paterna, odnosno primjena procesa gdje je veoma važno definisati korake i njihov redoslijed (za pristup CHAT sobi su se morali izvršiti svi gore, prethodno navedeni koraci).

1.7. Medijator

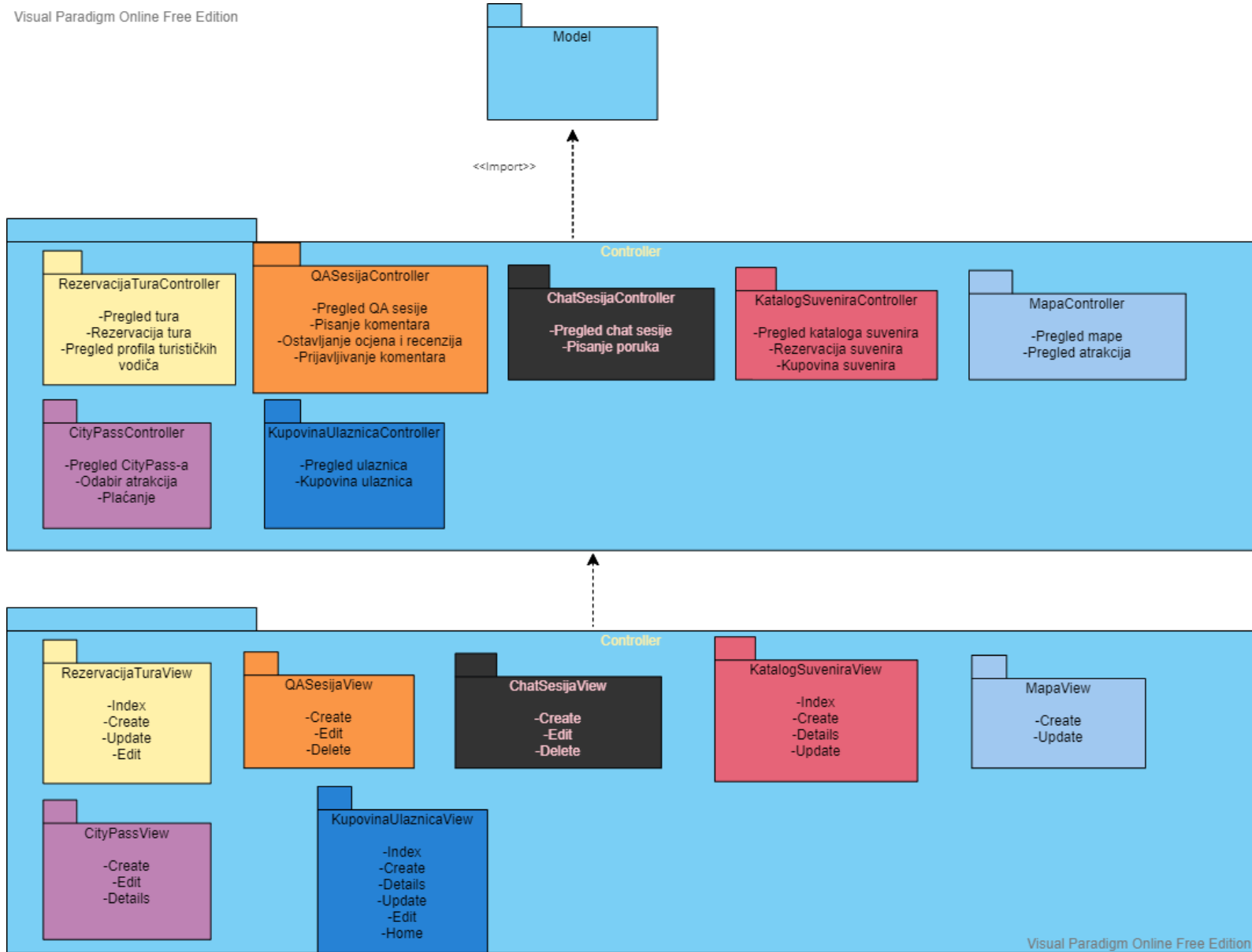
Primjer upotrebe Medijator paterna bi se mogao predstaviti preko rezervacije tura posjeta atrakcijama sa turističkim vodičima. Broj turista koji se mogu prijaviti za određenu turu je ograničen. Shodno tome, zamislimo situaciju da je ostalo još jedno mjesto i dva zainteresovana strana turista koji ne znaju jedan za drugog. Oba su došla do dijela za potvrdu zahtjeva o rezervaciji, ali je prvi turist bio brži u odnosu na drugog i kliknuo opciju rezerviši. Sada drugi turist više nema mogućnost rezervisanja iako ne zna razlog. U tom slučaju turist dobija informaciju o grešci i upućuje se da se vrati na početak. Na ovaj način kreirali smo centralizovan sistem (sabrnicu) koja smanjuje broj direktnih veza između klasa, uz napomenu da je gore naveden samo primjer (nešto što bi se moglo desiti, i zbog čega bi se aplikacija u budućnosti mogla nadograditi, dodatno razviti i poboljšati).

2. Dijagram komponenti



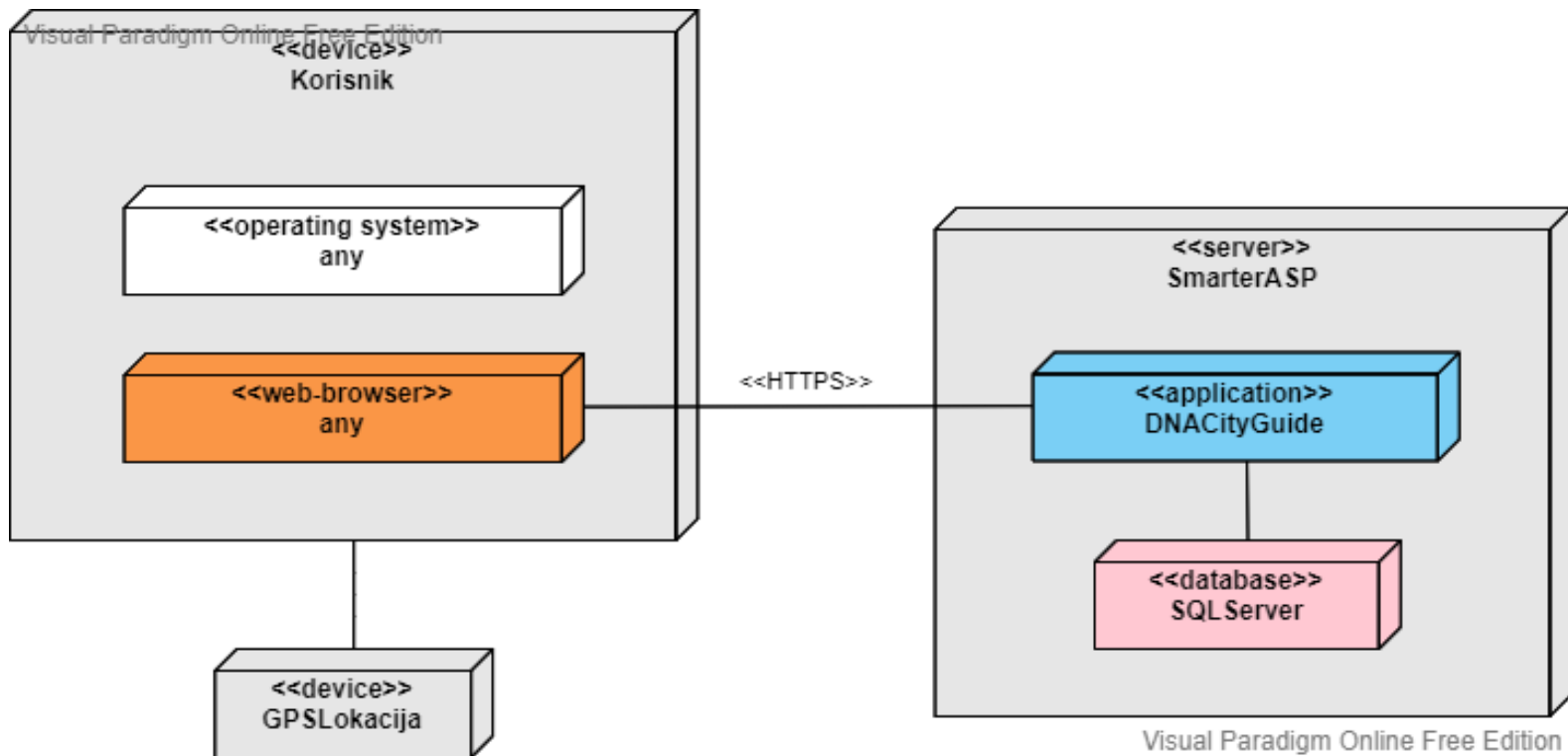
Slika 4. Dijagram komponenti

3. Dijagram paketa



Slika 5. Dijagram paketa

4. Dijagram raspoređivanja



Slika 6. Dijagram raspoređivanja