**Arhitekturni dizajn softverskog sistema**

**eBasket**

**Mihailo Rašić, 15850**

**Aleksandar Trajković, 15944**

**Opis projekta**

Aplikacija eBasket je Web aplikacija koja će imati dva tipa korisnika, trenera i klijenta, i koja će se prvenstveno baviti iscrtavanjem, čuvanjem i uvežbavanjem košarkaških akcija od strane trenera i igrača koji su registrovani u istom klubu.

**Funkcionalni zahtevi**

1. Registrovanje korisnika sistema, gde je moguće imati različite uloge, trener i igrač.
2. Klijentu treneru je omogućeno pravljenje nove košarkaške akcije.
3. Klijentu igraču je omogućeno i da započne uvežbavanje postojećih akcija.
4. Igrači mogu da odigravaju poteze tako što pomeraju svoju figuru po terenu.
5. Klijent trener će imati uvid u uspešnost treninga.

**Nefunkcionalni zahtevi**

1. Pristupačnost – web aplikacija ne zavisi od platforme na kojoj se izvršava

2. Dostupnost – pošto se aplikacija nalazi na serverima koji garantuju

dosupnost 24/7, te će naša aplikacija biti dostupna uvek svim korisnicima

3. Sigurnost – svaki korisnik ima svoj nalog i niko drugi sem njega nema

pristup tim podacima

4. Pouzdanost

5. Performanse – potrebno je smanjiti vreme odziva i prilagoditi propusnu moć sistemu koji se realizuje.

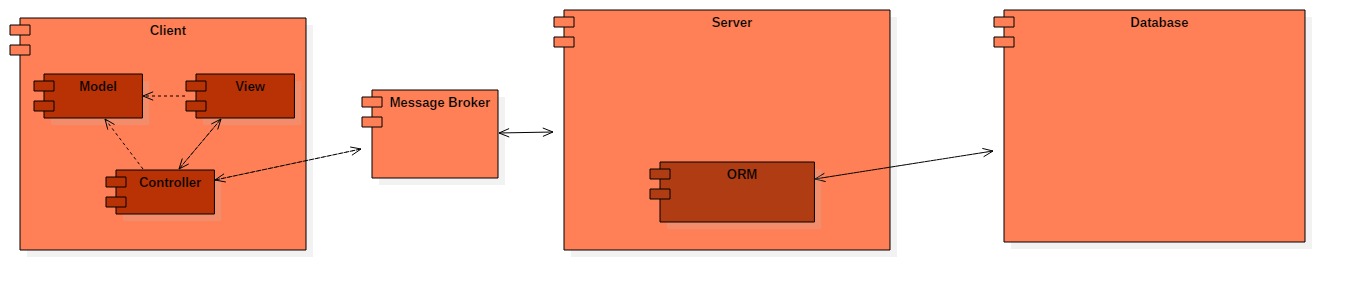
**Arhitekturni dizajn**

**Strukturni pogled**

Prikazuje strukturu sistema navodeći komponente sistema.

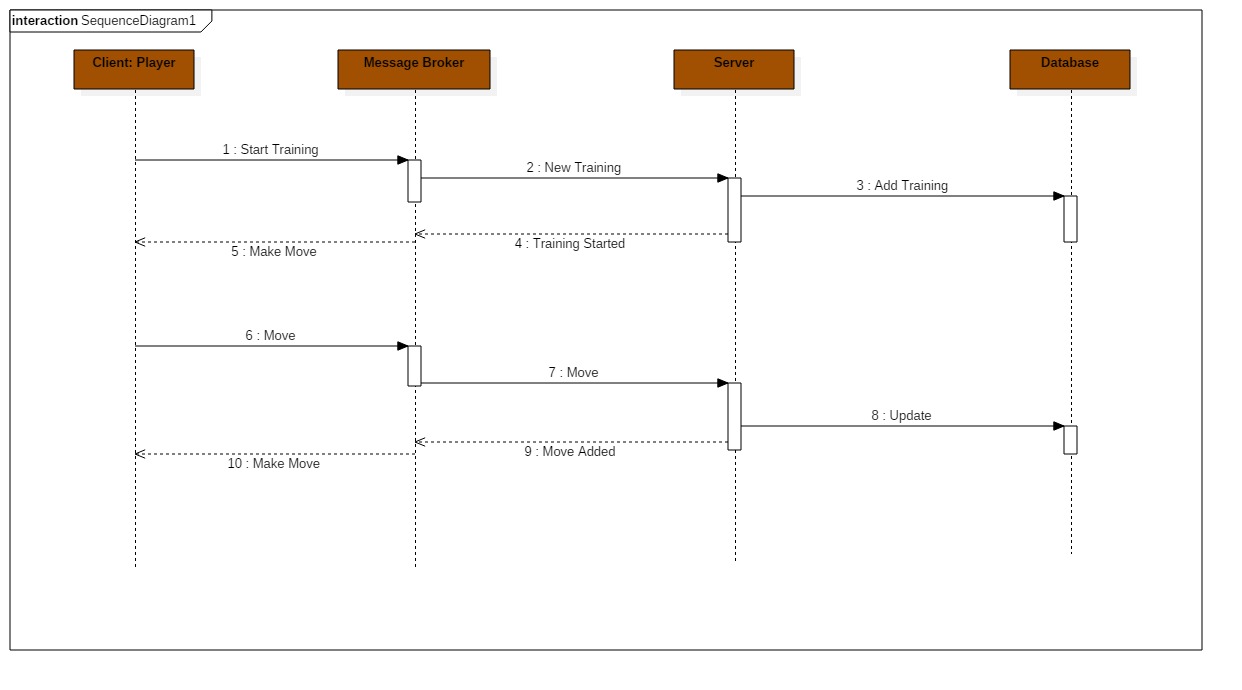
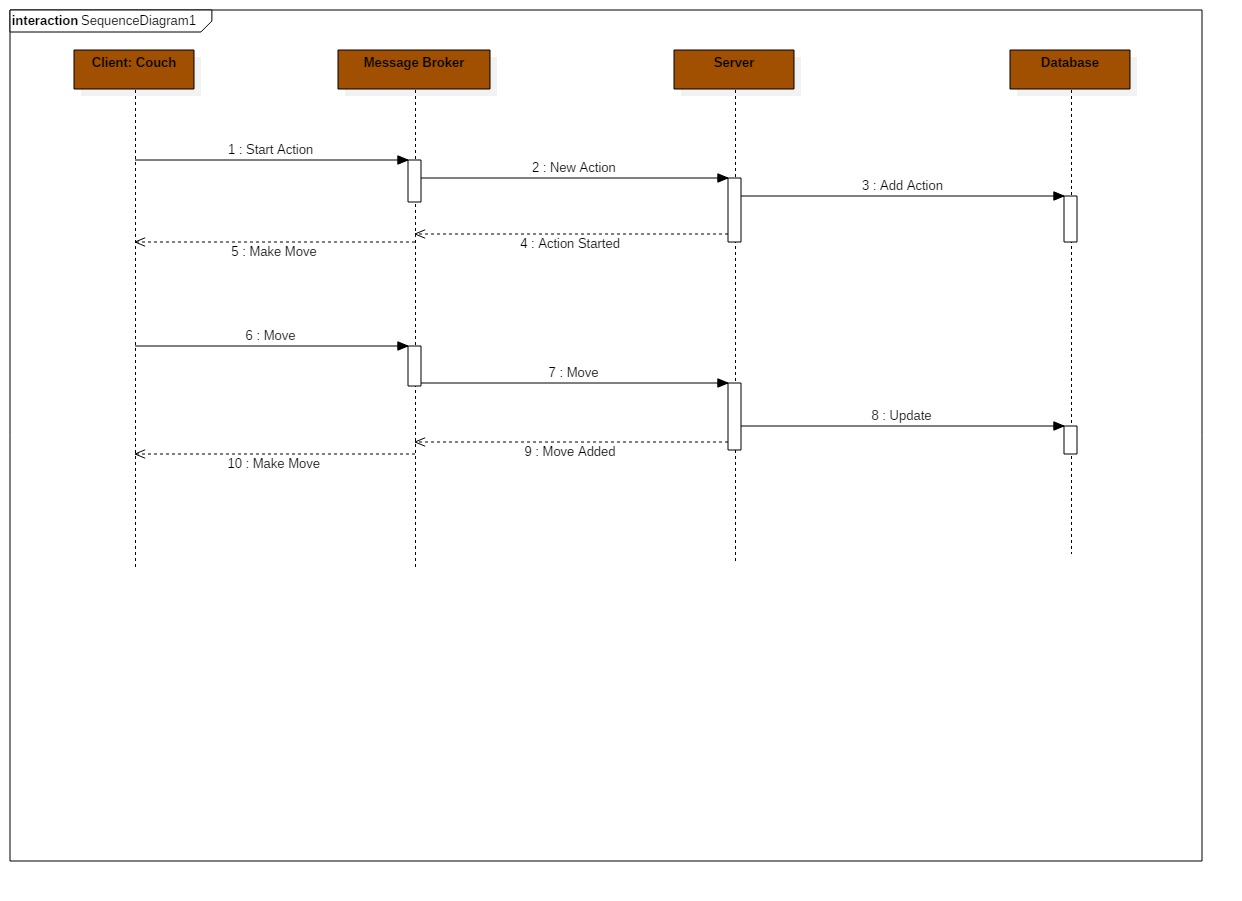
Klijent ima unutrašnju stukturu zasnovanu na MVC projaktnom obrascu.

Klijent komicira sa serverom preko Controller-a koristeći Message Broker. Server preko ORM-a komunicira sa bazom podataka.



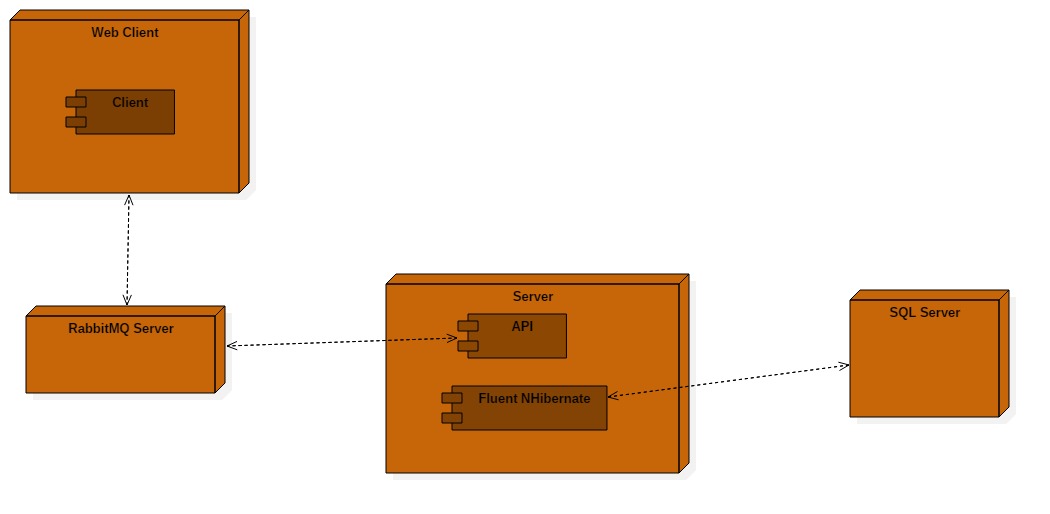
**Bihevioralni pogled**

Bihevioralni pogled opisuje interakciju između komponenti sistema prilikom pravljenja i uvežbavanja akcija. Prvo se upisuje akcija u bazu podataka, a nakon toga se ažurira prilikom svake promene.



**Alokacioni pogled**

Alokacioni pogled prikazuje raspored komponenti na čvorovima tj. na stvarnim računarima. Client aplikacija se otvara u Web pretraživaču . Za komunikaciju sa serverom se koristi RabbitMQ server. Komunikacija između servera i baze se odvija pomoću mapiranja objekata putem FluentNHibernate-a.



**Arhitekturni obrasci**

**Publish-Subscriber**

**Singleton**

**MVC**

**Specifikacija biblioteka i programskih okvira**

**NodeJS MVC**

**RabbitMQ**

**AMQP library**

**FluentNHibernate**

**MS SQL DBMS**