

# Treškviz

## Model komunikacije

Aleksa Stančev 17434

Lazar Ignjatović 17124

## Sadržaj

1. Izabrana tehnologija.....	2
2. Model komunikacije.....	2

## 1. Izabrana tehnologija

Za izradu komunikacionog sloja Treš kviz aplikacije smo koristili Microsoft-ovu biblioteku SignalR. Ova biblioteka se zasniva na RPC (remote procedure call) principu, i na taj način omogućava da klijentska strana poziva serverske metode i obratno. Za ostvarivanje komunikacije između klijentske i serverske strane ova biblioteka koristi websocket protokol.

SignalR za apstrakciju komunikacije nudi klasu Hub, koju je potrebno naslediti. Klase koje je nasleđuju sadrže metode koje pozivaju klijenti, korišćenjem klijentske SignalR biblioteke. Za pozivanje određene metode potrebno je proslediti njen naziv, kao i odgovarajući broj parametara. Pozivi klijentskih metoda iz serverske aplikacije se vrše po istom principu. U klijentskim aplikacijama se pozivi ovih metoda tretiraju kao događaji.

Klijentska aplikacija za povezivanje na željeni Hub koristi endpoint, koji je tom Hub-u namenjen u serverskoj aplikaciji.

Serverska aplikacija za izbor klijenata, čije će metode pozivati, koristi mogućnost SignalR-a za kreiranje grupa klijentskih konekcija i pozivanja određene metode za sve pripadnike jedne grupe. Klijenti, nakon određenih akcija, bivaju raspoređeni u grupe (lobi, sobe za kreiranje partije, kao i same partije), pozivaju metode Hub-ova i u realnom vremenu dobijaju obaveštenja o akcijama ostalih pripadnika grupe.

## 2. Model komunikacije

Serverska aplikacija sadrži četiri klase izvedene iz Hub klase. Svaka od njih sadrži skup metoda koje omogućavaju određen segment komunikacije. Svaka od tih klasa je navedena ispod, zajedno sa kratkim opisom njene nadležnosti i njenih funkcija.

### 1. Authentication Hub:

Klasa zadužena za prijavljivanje, registraciju korisnika i odjavljivanje.

Metode Login, Register i Logout vrše prijavljivanje, registrovanje i odjavljivanje klijenata, respektivno.

### 2. LobbyHub:

Klasa zadužena slanje obaveštenja svim klientima u lobiju o dostupnim sobama kao i pretragu soba na osnovu ključa.

Metoda RoomsUpdate šalje klijentima obaveštenja o dostupnim javnim sobama, a metoda FindRoom vrši pronalaženje sobe sa prosleđenim ključem.

### 3. RoomHub:

Klasa zadužena za pristup, napuštanje, ažuriranje stanja sobe i pokretanje partije.

Metoda JoinRoom i LeaveRoom dodaje odnosno uklanja pozivajućeg klijenta iz sobe sa prosleđenim ključem.

Metoda ModifyRoom vrši ažuriranje stanja sobe na osnovu prosleđenog objekta.

Metoda MarkReady označava da je pozivajući klijent spreman za početak igre,

Metoda StartGame započinje partiju.

#### 4. MatchHub:

Klasa zadužena za pristup, napuštanje, ažuriranje stanja partije, prihvatanje odgovora i određivanje pobednika.

Metoda JoinMatch i LeaveMatch dodaje odnosno uklanja pozivajućeg klijenta iz partije sa prosleđenim ključem.

Metoda SubmitAnswer prihvata odgovor na tekuće pitanje u partiji od strane pozivajućeg klijenta.

Metoda MatchUpdate obaveštava sve članove partije o promenama stanja iste.

Metoda DeclareWinner obaveštava članove partije o pobedniku.

Primer akcije označavanja spremnosti igrača za početak partije ilustrovan sekvencijalnim dijagramom:

