Great Generals

Arhitektura i projektovanje softvera

Nemanja Milosavljević - 15747

Bojan Pavlović - 15800

Sadržaj

[Opis projekta](#_mavxhjbshaz6)

[Funkcionalni zahtevi](#_3km4w33wtr8q)

[Nefunkcionalni zahtevi](#_wjp4zv242syc)

[Arhitekturni zahtevi](#_hn63frjztguk)

[Arhitekturni šabloni](#_ionlxlyfxavn)

[Arhitekturni dizajn](#_gqiiqn4uhr)

[Arhitekturni obrasci](#_t3htmz2mljzk)

[Specifikacija biblioteka i programskih okvira](#_xabfbolj4j05)

[Dizajn obrasci koje će biti korišćeni](#_w9zs524xy5c3)

# Opis projekta

*Great Generals* predstavlja *multiplayer* igru za više igrača realizovanu u vidu desktop aplikacije.

Na početku igre igračima se dodeljuje određena teritorija obeležena jedinstvenom bojom. Teritorija kojom igrač raspolaže predstavljena je u vidu šestougaonika koje je moguće selektovati. Na početku igre svaki igrač na raspolaganju ima kralja i par grupa neobučenih vojnika (jedinica). Desnim klikom na jedinicu (šestougao na kome se nalazi nacrtan određeni tip vojnika) otvara se meni sa mogućim opcijama za selektovanu jedinicu.

Mogućnosti jedne jedinice su: pomeranje po tabli, kako po svojoj tako i po tuđim teritorijama pri čemu brzina kretanja jedinice zavisi od njenog tipa i terena po kome se kreće, napad na protivničko polje i napad na protivniču armiju ili kreiranje logora na trenutnom mestu grupe. Nakon kreiranja logora vojnici se mogu trenirati kako bi im se povećalo iskustvo i verovatnoća za pobedu u naradnim bitkama. Sem treninga vojnici se u logoru mogu i lečiti nakon bitke.

Nove jedinice kreiraju se odabirom praznog polja a zatim i tipa vojnika od kojih će se novonastala jedinica sastojati. Vreme potrebno za kreiranje nove jedinice zavisi od izabranog tipa vojnika (kreiranje snažnijih odnosno spremnijih vojnika zahteva više vremena). Za kreiranje nove jedinice potrebna je određena količina novca srazmerna tipu jedinice.

Svaki igrač može videti samo par linija protivničkih šestougaonika koji se nalaze blizu granice kako bi raspored protivničkih jedinica ostao nepoznat. Takođe pri kretanju kroz neprijateljsku teritoriju igraču su otkrivena samo polja u blizini njegovih jedinica.

U toku igre svim igračima je na raspolaganu mogućnost kreiranja *chat-ova* sa odabranim igračima.

Izborom nekog od polja sa neprijateljske teritorije i opcije za kreiranje saveza, uz saglasnost druge strane, moguće je kreirati prijateljske savez sa ostalim igračima. Igrači iz istog saveza mogu zajedno učestvovati u borbama pri čemu se njihove jedinice neće sukobljavati. Savez takođe može biti sklopljen i između više igrača.

Ukoliko neka od neprijateljskih jedinica napadne polje na kome se nalazi kralj i dođe do pobede neprijateljske jedinice, pobednik dobija svu teritoriju koju je posedovao poraženi igrač i 70% novca poraženog dok se jedinice poraženog igrača gube.

U toku cele igre količina novca u posedstvu igrača se povećava kako bi igrači imali mogućnost kreiranja novih jedinica. Nakon osvajanja neprijateljskih teritorija igrač dobija određenu količinu novca.

Nakon započetog sukoba između dve armije moguća su 2 ishoda: pobeda jedne od jedinica pri čemu pobednik zadržava teritoriju na kojoj se sukob odvio ili predaja jedne od jedinica pri čemu polje na kome se sukob odvio takođe zadržava pobednik dok jedinica koja se predala ostaje oslabljena.

Smatra se da je igrač izgubio partiju ukoliko ne poseduje nijednu teritoriju ili je njegov kralj poražen (pri čemu takođe gubi svu teritoriju tako da je ispunjen prvi uslov).

Igra se smatra završenom u trenutku kada celokupna teritorija pripada jednom igraču.

# Funkcionalni zahtevi

1. Povezivanje klijenta i servera
2. Kreiranje korisničkih naloga
3. Prijavljivanje uz pomoć već kreiranih naloga
4. Kreiranje igre sa određenom mapom sa igračima koji su izabrali istu
5. Igranje igre
   1. Ispravan rad svih klijentu dostupnih funckionalnosti (klijentski deo aplikacije)
   2. Sinhronizacija pogleda igrača u jedoj partiji (serverski deo aplikacije)
6. Pamćenje podataka o igri ukoliko je korisnik odabrao opciju pamćenja igre
7. Komunikacija između klijenata.

# Nefunkcionalni zahtevi

1. Pouzdanost
2. Skalabilnost
3. Minimalno kašnjenje između odigravanja poteza i osvešavanja prikaza kod ostalih igrača.
4. Minimalno korišćenje podataka učesnika igre (internet podaci)
5. Mogućnost lakog dodavanja dodatnih mapa za igru
6. Mogućnost lakog dodavanja novih tipova „jedinica“
7. Podržavanje velikog broja igrača
8. Modifikabilnost

# Arhitekturni zahtevi

* Korišćenje relacione baze podataka
* Skrivena šema baze podataka
* Slabo spregnute komponente

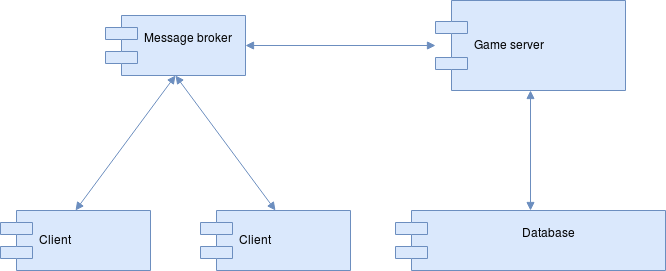
# Arhitekturni šabloni

1. Publish&Subscriber- koji je sadržan u message broker-u
2. MVC na klijentu
3. Klijent server
4. Repozitorijum

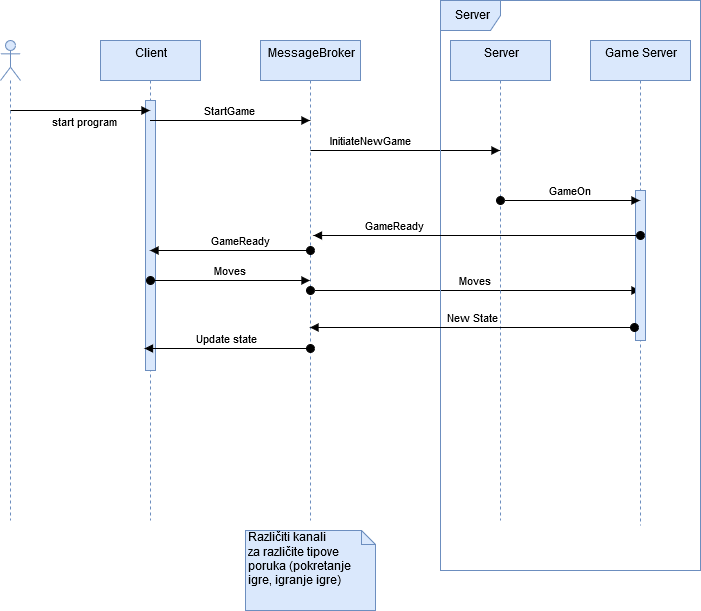
# 

# Arhitekturni dizajn

* Strukturni dijagram



* Sekvencijalni dijagram



# Specifikacija biblioteka i programskih okvira

* JavaFx
* RabbitMQ
* Hibernate

# Dizajn obrasci koji će biti korišćeni

* Command - komunikacija između server proxy-ja i klijentskog kontrolera kao i između klijentskog kontrolera i klijentskog pogleda (view) obavlja se prosleđivanjem adekvatnih komandi.
* Observer – obrada događaja koji su se desili u klijentskom pogledu (view), obrada pristiglih poruka sa message broker-a.
* Strategy – način kretanja i napada određenih „jedinica“, postavljanje mogućih opcija (akcija) nad selektovanim poljem.
* Prototype – kriranje novih „jedinica“ na osnovu zadatog imena (string)

## Potencijalni problemi u implementaciji

Kašnjenje u komunikaciji zbog opterećenosti servera.

Konzistencija podataka.