

# Clash of the Superheroes II

Cilj ovog zadatka je da polaznici Factory frontend akademije iskoriste sve ono što su naučili kroz ovaj period održavanja radionica (HTML-a SASS-a, javascript, jQuery) te da nadograde prethodni zadatak (*Clash of the Superheroes*) tako da naprave grafičko sučelje za igru, u kojem će omogućiti interakciju korištenjem jQuery. Pri realizaciji zadatka bilo bi dobro da se nastoji držati razdvojena javascript logika same igre od javascript/jQuery logike koja korisniku omogućava interaktivnost.

*Kada govorimo o potrebi razdvajanja ove dvije logike, mislim na to da se izbjegava pisanje poslovne logike igre unutar callback funkcija evenata i slično; umjesto toga jQuery dio koda bi morao koristiti API vaše igre (API su public metode na vašim objektima koje omogućuju interakciju i upravljanje tim objektima izvana - u ovom slučaju je to interakcija korisnika preko web sučelja).*

Npr:

```
// Ovo je samo koncept kako se može razdvojiti poslovna logika igre
import { Turnir } from './my-game'
import HeroRepository from './HeroRepository'

const heroRepository = new HeroRepository({apiKey: 'xxxxxxxxxxxxxxxx'});
const turnir = new Turnir();

// turnir.onMatchEnd = ...
// turnir.onGroupEnd = ...

$('.hero').click(async function () {
  turnir.signIn(
    await heroRepository.getById(
      $(this).data('id')
    )
  )
})

$('.play').click(() => turnir.start())
```

# Zadatak

Potrebno je iskodirati template koji je dostupan na [linku](#)

Design se sastoji od dva screen-a

1. Početni screen
2. Tournament bracket screen

## Početni screen

Početni screen se sastoji od slider-a, buttona za nasumičan odabir heroja, buttona za početak turnira;

Po učitavanju stranice, potrebno je dohvatiti 10 nasumičnih heroja;

U odabranom setu od 10 heroja ne smije biti duplikata;

Slider korisniku omogućava odabir jednog od 10 nasumično odabranih heroja;

Svih 10 heroja prikazuju se u vidu thumbnailova ispod slidera;

Po defaultu, odabran je prvi heroj sa liste;

Slider se mijenja klikom na strelice, ili klikom na određeni thumbnail;

Lijeva strelica je disablevana ako ne postoji niti jedan heroj sa lijeve strane trenutnog prikazanog heroja

Desna strelica je disablevana ako ne postoji niti jedan heroj sa desne strane trenutno prikazanog heroja

Klikom na button za nasumičan odabir heroja, bira se jedan od 10 dostupnih heroja i prikazuje se u slideru

Klikom na button “Nastavi” otvara se slijedeći screen i započinje turnir;

Korisnikov heroj je onaj koji je bio odabran/prikazan na slideru u trenutku klika na button “Nastavi”

Slider se smije realizirati i korištenjem lib-a (*npr: slick slider*)

## Tournament bracket screen

Pokretanjem igre, potrebno je nasumično odabrati preostalih 7 heroja za turnir;

U turniru ne mogu sudjelovati dva ista heroja;

Heroji se u svakoj grupi “igraju” na dva dobivena meča;

Pauza između svakog meča u nekoj grupi je 2 sekunde;

Meč se odigrava istovremeno u svim grupama;

Rezultat u bracket-u je potrebno ažurirati nakon svakog meča;

Kada heroj neke grupe dosegne rezultat “2”, tada prelazi u slijedeću rundu;

Potrebno je napraviti animaciju tako da se slika pobjednika neke grupe translatira u slot/kružić slijedeće runde;

Slika heroja koji je izgubio u grupi treba biti zatamnjena (*kao u designu*) samo za tu grupu;

## Dodatne informacije

- **Meč** - najmanja cjelina igre, označava jedan sukob dva heroja
- **Grupa** - predstavlja cjelinu u kojoj 2 heroja igraju na N dobivenih mečeva
- **Runda** - predstavlja faze turnira, gdje se svaka faza turnira sastoji od N grupa; Turnir od 8 sudionika ima 3 runde: četvrtfinale, polufinale i finale

---

### ***Migracija između 2 screen-a***

*Za prikaz i migraciju sa jednog screena na drugi screen, nije potrebno koristiti neki od javascript frameworka ili rutera, dovoljno je iskoristiti css svojstvo za sakrivanje containera*

---

### ***Animacija kod pobjede heroja u nekoj grupi***

*kod animacije za pobjednika neke grupe, potrebno je obratiti pažnju da se u prvoj i posljednjoj rundi, osim translacije, mora odraditi i scale slike.*

---

### ***Različit broj borbi u grupi***

*S obzirom da jedan par može odigrati s rezultatom 2:0, a drugi 1:2, to nas vodi do problema što pobjednik prvog para mora “sačekati” pobjednika iz drugog para (razmotriti upotrebu promisa)*

---

### ***Kontrola brzine izvršavanja/odigravanja***

*Razmotriti update igre koju ste napravili, tako da se uvedu koraci gdje je 1 korak = 1 meč (s obzirom da je meč, najmanja cjeina/događaj), na taj način bi mogli definirati neki interval i pozivati metodu nextStep() svakih N sekundi (neka vrsta takta kojom brzinom se odigravaju mečevi, a samim time kompletan turnir)*

---