

## Igra zmijice (Projektan zadatak)

# Predmet CS324 - Skripting jezici

Profesori:

Dr. Igor Franc

Asistent:

Katarina Banovic Student: Nikola Nesovanovic 4284

# Sadrzaj:

## 1.Uvod……………………………………………………………………

## 2.Opis koriscenih tehnologija………………………………………….

## 2.1. React………………………………………………………….

## 2.2. NodeJS……………………………………………………….

## 2.3. Node Package Manager…………………………………....

## 3.Struktura projekta…………………………………………………….

## 4.Pokretanje projekta…………………………………………………..

## 5.Zakljucak……………………………………………………………....

## 6.Reference……………………………………………………………..

1.Uvod

Cilj projektnog zadatka ‘Igra zmijice’ je da se napravi klon klasicne ‘snakes’ igrice za internet pretrazivace, koja je napravljena pomocu tehnologija koji omogucavaju dalji development i integraciju sa lakocom. Posto je korisnicki interfejs radjen pomocu react-a, igricu je moguce lako prilagoditi za android i ios platforme, koristeci react-native tehnologije.

**NodeJS** – NodeJS je programski jezik pisan u JavaScript-u koji je korišćen kao rešenje za serverski deo našeg projekta. Koristi Google-ov V8 JavaScript engine kao način interpetiranja. Open source je i takodje je napravljen da funckioniše na svakoj platformi.

**React** - React je javascript biblioteka koja se koristi za dinamicno i interaktivno razvijanje korisnickih interfejsa za veb pretrazivace.

**Node package manager** - Npm je menadzer biblioteka koji sa jednom komandom moze da instalira sve biblioteke koji su deo node okruzenja ili se koriste za odredjeni projekat

2. Opis koriscenih tehnologija:

## 2.1. React:

React je javascript biblioteka koja se koristi za dinamicno i interaktivno razvijanje korisnickih interfejsa za veb pretrazivace.

## 2.2. NodeJS:

NodeJS je programski jezik pisan u JavaScript-u koji je korišćen kao rešenje za serverski deo našeg projekta. Koristi Google-ov V8 JavaScript engine kao način interpetiranja. Open source je i takodje je napravljen da funckioniše na svakoj platformi.

## 2.3. Node Package Manager:



Node Package Manager je podrazumevani menadžer paketa za JavaScript runtime environment NodeJS. Od 0.6.3 verzije NodeJS-a, Node Package Manager je automatski instaliran sa okruženjem i da bi se danas koristio dovoljno je samo instalirati NodeJS okruženje. Node Package Manager je potpuno napisan u JavaScript-u i razvijen je od strane Isaac Z. Schlueter-a. Prvo izdanje Node Package Manager-a je objavljeno 12.1.2006, dok je poslednje stabilno izdanje objavljeno 6.5.2016 godine. Sve Node Package Manager komande je moguće pokrenuti iz terminala ili Command Prompt-a u zavisnosti od operativnog sistema korisnika.

3. Struktura projekta:

## 3.1. Opis strukture direktorijuma i fajlova:

Struktura foldera i fajlova je jako bitna za projekat kada na njemu rade vise osoba ili se development na projektu nastavlja posle duzeg vremena. Node package manager sam odvaja sve NodeJS module u node\_modules folder, kao i odvaja sve korisnicke fajlove u public i src foldere (public za fajlove koji su vidljivi krajnjem korisniku i src za ‘back-end’ fajlove).

### 3.1.1. Node\_modules direktorijum:

Node package manager pri pokretanju npm i komande kreira preko 1000 foldera sa svim mogucim bibliotekama i modulima koje je moguce koristiti sa NodeJS-om. Takodje ukoliko zelite u json konfiguracionim fajlovima mozete specifirati koje je biblioteke potrebno instalirati za vasu aplikaciju.

### 3.1.2. Public direktorijum:

Public direktorijum ne sadrzi javascript, css ili slicne korisnicke fajlove, od kojih se sastoji glavni deo nase aplikacije. U njemu se nalaze samo krajnji fajlovi koji su dostupni krajnjem korisniku.

**Index.html** je prosto template koji sadrzi html kod, koji sadrzi meta informacije o nacinu prikazivanja aplikacije na stranici, kao i ‘root div’ u koji cemo smestati sav react generisan kod.

**Manifest.json** je fajl koji sadrzi informacije o aplikaciji poput imena, ikone kao i osnovnih nacina prikaza.

**Favicon.ico** je ikonica aplikacije tj stranice.

### 3.1.3. Src direktorijum:

Src direktorijum sadrzi sve skripte koje kontrolisu rad aplikacije.

**useInterval.js** je fajl koji sadrzi specijalni ‘hook’ koji je razvio Dan Abramov. ‘Hook’-ovi su specijalne komande nalik event listener-ima. Oni vezuju promenljivu sa specijalnom funkcijom, koja menja vrednost promenljive pri njenom pozivu i usput pozivaju ponovni render react elemenata. ‘useInterval’ je hook koji se poziva posle odredjenog intervala vremena i u ovom programu ga koristima kako bi omogucili da se ‘gameLoop’ funkcija poziva posle konstantom odredjenog vremenskog perioda.

**Index.js** prosto vezuje react elemente generisane **app.js** fajlom, sa root div elementom koji se nalazi u **index.html** fajlu.

**Constants.js** je fajl koji sadrzi konstante koje kontrolisu pocetnu poziciju zmije i jabuke, kolicinu i velicinu tajlova i brzinu zmije. Takodje ovde se nalaze pomocne konstante poput ‘key code’-ova strelica kao i vektora direkcije koji odgovaraju pravcu strelice.

**App.js** je fajl u kome se nalazi glavni deo koda.

Prvi deo fajla su Hook-ovi koji vezuju funkciju za konstantu i svaki put kada se pozove ta funkcija, vracaju novu konstantu na mestu stare.

Drugi deo fajla sadrzi funkcije od kojih su glavne gameLoop, startGame i endGame, i one upravljaju glavnim tokom igre i jedine pozivaju ostale funkcije i hookove.

I za kraj fajl se zavrsava sa return kljucnom reci koja vraca html elemente index.js fajlu koji ih smesta u root div.

4. Pokretanje projekta:

## 4.1. Instalacija aplikacije:

Da bi instalirali aplikaciju potrebno je prvo skinuti najnoviju verziju node-a sa: <https://nodejs.org/en/download/>. U node je automaski ukljucen i npm koji cemo koristiti za instaliranje svih ostalih biblioteka koje se koriste u projektu. Sam projekat se moze naci na <https://github.com/SimpleBrah/ProjekatSkriptingJezici/>, i potrebno je skinuti projekat u zip formatu, a zatim ga unzipovati negde na racunaru. Kada se skine i instalira, potrebno je otvoriti command prompt, i premestiti se u folder u koji smo unzipovali projekat, i zatim pokrenuti komandu **npm i**. Sada imamo sve fajlove koji su potrebni za pokretanje aplikacije.

## 4.2. Pokretanje aplikacije:

Aplikaciju pokrecemo pomocu komande **npm start** koja ce pokrenuti lokalni server na adresi: localhost:3000, i zatim otvoriti default pretrazivac na toj adresi. Ukoliko je neki drugi ili vise pretrazivaca ili prozora vec otvoreno pre pokretanja ove komande, aplikacija ce se otvoriti u prozoru koji je poslednji bio fokusairan, pa proizvoljno pre pokretanja ove komande, mozete otvoriti preferirani prozor pretrazivaca, kako bi ste kontrolisali gde se aplikacija otvara.

## 4.3. Koriscene aplikacije:

Glavni deo programa je canvas element koji se nalazi sa leve strane prozora. U njemu se nalazi sama igra, gde zmiju kontrolisete putem strelica na tastaturi. Sa desne strane se nalazi forma za pokretanje nove partije, gde mozete promeniti framerate igrice (veci broj znaci da je igrica sporija), velicinu mape (izrazenu preko kolicine polja na x i y osi), kao i wrap mode tj. Da li ce igrac izgubiti partiju kada udari u ivicu ekrana ili ce se pojaviti na paralelnoj ivici ekrana i nastaviti igru. Kada zavrsite partiju dobijate score koji se nalazi na vrhu forme i mozete pokrenuti novu partiju sa podesavanjima koji se nalaze u furmi pomocu dugmeta koji se nalazi na vrhu forme. Povecavanje velicine povrsine za igranje ce povecati i sam canvas element pa pazite pri menjanju tih vrednosti ako vam je rezolucija manja.

5. Zakljucak:

Ova literatura predstavlja dokumentaciju izrade aplikacije, kao i uputstvo za instaliranje, pokretanje i koriscenje aplikacije.

Aplikacija je pravljena pomocu tehnologija koje omogucavaju lako skaliranje projekta, kao i podizanje servera koji ce omoguciti da se aplikaciji u gotovom stanju pristupi sa interneta. Korisnicki interfejs i korisnicki dozivljaj (UI i UX), su prioriteti pri daljem razvoju ove aplikacije, a dodavanje novih gameplay elemenata je lako za dodavanje zbog funkcionalnih react elemenata, koji nemaju nuspojave jer ne menjaju stanje u bilo kom trenutku. Sama aplikacija ne koristi u potpunosti funkcionalnu paradigmu programiranja, ali su sve komande u glavnom app.jss fajlu potpuno funkcionalne.

6. Reference:

## React: <https://reactjs.org/>

## Github: <https://github.com/>

## NodeJS: <https://nodejs.org/en/>

## UseInterval custom hook: <https://www.npmjs.com/package/use-interval>