****

**React.js**

**Predmet: Klijent Server Sistemi**

**Profesor: Student:**

**dr Mirko Kosanović Miloš Marković**

**Miloš Kosanović rer 26/16**

**Niš 26.04.2019.**

SADRŽAJ

[1. Uvod - 3 -](#_Toc512514140)

[2. Instalacija i podešavanje projekta - 3 -](#_Toc512514141)

[2.1. Instaliranje modula - 3 -](#_Toc512514142)

[3. Struktura projekta - 3 -](#_Toc512514143)

[3.1 Serverski deo - 4 -](#_Toc512514144)

[3.2 Klijentski deo - 4 -](#_Toc512514145)

[3.3 Komunikacija - 5 -](#_Toc512514147)

[4. Rad aplikacije - 7 -](#_Toc512514148)

[4.1 Opis implementacije - 7 -](#_Toc512514149)

[4.2 Opis funkcionalnosti – korisničko uputstvo - 7 -](#_Toc512514150)

[5. Literatura - 9 -](#_Toc512514151)

# 1. Uvod

U ovom projektu obrađen je protitip online table za pisanje (writebook). Tehnologije koje su korišćene na klijentskoj strani su HTML, CSS i **React** framework dok je na serverskoj strani korišćen NodeJS i Express.js. Takođe koristimo React biblioteku za frontend.. Alati koji su potrebni za izradu projekta su web browser, neki text editor i terminal.

# 2. Instalacija i podešavanje projekta

Pre svega moramo imati instaliran Node.js na nasem računaru.Zatim se koristi node package installer.Za ovaj projekat je korišćen Yarn kao package installer,koji je alternativa npm-a.Projekat se može pokrenuti i sa npm-om.

## 2.1. Instaliranje modula

Organizacija koda u jednom fajlu, čini programiranje veoma komplikovanim. Iz tog razloga se u JavaScript-u kao i u mnogim drugim programskim jezicima, programski kod deli u manje celine tzv. module.

JavaScript modul od kojeg zavisi rad drugog modula se zove “zavisnost” (eng. “dependency”).

U ovom projektu nalazi se **package.json** fajl, to je fajl koji se inicijalizuje sa projektom, naravno na zahtev programera, i ukoliko popunimo sve informacije ispravno biće kreiran fajl. Zatim kad krenemo da instaliramo nove module i ukoliko upotrebimo neke ključne reči (-S , -save) prilikom instalacije u ovom fajlu biće nam upisani svi moduli koje koristimo za našu aplikaciju.Ukoliko je sve to ispravno kreirano, da bi smo na nekom drugom računaru pokrenuli i instalirali module, potreno je ukucati sledeću naredbu **npminstall**ova naredba prvo pretražuje **package.json** fajl i u njemu traži i instalira sve dependence (tj. Module ili biblioteke) koji su potrebni za ovaj projekat.

# 3. Struktura projekta

Projekat u sebi sadrži korenski (engl. Root) direktorijum '/' koji u sebi sadrži '/client' i '/server ' direktorijum,u kojima se nalaze node moduli i .json filovi.U direktorijumu /client' nalazi se /src' gde su smesteni svi html i css dokumenti.

## 

## 3.1 Serverski deo

**Node.js** je radno okruženje JavaScript programskog jezika koje se koristi na strani web servera. JavaScript se tradicionalno koristi na strani klijenta – tzv. front-end razvoj, a Node je omogućio da se JS koristi i za back-end razvoj odnosno serversko izvršavanje koda.

Odmah da napomenemo i u čemu Node.js blista – to su tzv. real-time (u realnom vremenu) aplikacije kod kojih se vrši razmena velike količine podataka. Primer bi bile web aplikacije namenjene timskom radu poput Google Docs ili Dropbox Paper. Postoje i situacije u kojima nije preporučljivo koristiti Node.js – tipičan primer je bilo koja aplikacija koja je „teška“ za CPU, odnosno koja obavlja veliki broj kalkulacija.

**Node.js** ima veliki broj modula – **JavaScript** biblioteka ili funkcija koje obavljaju određene specifične zadatke i koje možete pridodati svojoj aplikaciji. Postoje neki ugrađeni moduli, dok su drugi dostupni pre svega putem NPM repozitorijuma – Node Package Manager. NPM nije ograničen samo na Node.js – sada je to univerzalni JavaScript repozitorijum softvera i dostupni su moduli i za back-end i za front-end razvoj, React, Angular, Vue.js itd.

**Express.js** je Node.js web framework. Express olakšava i ubrzava mnoge radnje i spasava nas pisanja suvišnih linija koda prilikom kreiranja Node.js aplikacija.

Pre nego sto krenemo sa Reactom i klijentskom stranom,trebalo bi podignuti backend server da podrži našu aplikaciju.U ovom projektu ćemo koristiti Express uz NodeJS kako bismo kreirali API.

## 3.2 Klijentski deo

Na klijentskom delu korišćen je HTML kao opisni jezik specijalno namenjen opisu veb stranica. Pomoću njega se jednostavno mogu odvojiti elementi kao što su naslovi, paragrafi, citati itd. Korišćen je i CSS "Cascading Style Sheets", a služi za definisanje stilova koji određuju izgled HTML elemenata (font, boje, pozadine, razmake...).

Ti stilovi se nadovezuju u "Style Sheets", externe fajlove sa .css extenzijom, ili jednostavno se pišu u zaglavlju HTML dokumenta ili čak inline, tj. na samim elementima.

Externi stilovi su i najbolji jer vam omogućuju uštedu vremena pri radu i pri redizajniranju Vašeg HTML dokumenta.

React je JavaScript biblioteka koju je Facebook napravio kako bi pokrenuo izradu jedno-stranih aplikikacija (Single-Page Applications SPA).Nakon sto je Facebook objavio React,biblioteka je postala je veoma popularna širom sveta i masovno prihvaćena od strane developera.Danas veliki broj sajtova koristi React za svoju izradu(Airbnb,Netflix).

React koristi JSX sintaksu,skraćeno od JavaScript XML,sintaksni dodatak koji dozvoljava developerima da koriste XML kako bi opisali strukturu korisničkog interfejsa.

Komponente u Reactu su najbitniji delovi koda.Sve sa čime se susrećeš u React aplikaciji je deo komponente.

## 3.3 Komunikacija

Prvo smo importovali u aplikaciju nekoliko modula, uzmimo za primer Express modul. Njega smo uvezli i sačuvali ga u varijabli koju smo nazvali express. Moduli se uvek uključuju uz pomoć komande **require**. Zatim smo „pozvali” funkciju Express modula pod imenom **express()** i sačuvali smo tu funkciju u app varijabli.



Na kraju imamo app.listen(port), ovde kažemo serveru da “sluša” na nekom portu, u našem slučaju port incijalizovan kao konstanta : **3200** .

# 4. Rad aplikacije

## 

## 4.1 Opis implementacije

U fajlu **App.js** prvo što radimo, importujemo sve potrebne foldere i fajlove. Importujemo naše rute iz foldera router. Takodje importujemo nas fajl Todo u kome se nalaze naši stilovi.

Z

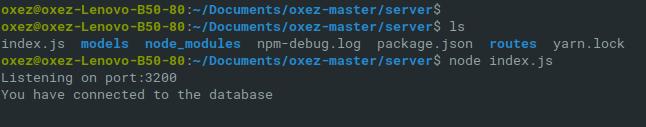
a serverski deo korišćen je **Express**framework. Instaliramo Express preko naredbe npm install express –save jer želimo da modul bude zapamćen u dependency listi.

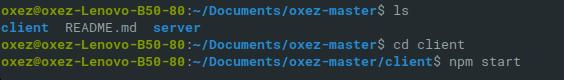
## 4.2 Opis funkcionalnosti – korisničko uputstvo

Sve funkcionalnost Node-a se nalazi u modulima. Moduli su JavaScript fajlovi koji sadrže funkcionalnost. Node ima bogatu bilblioteku ugrađenih (built-in) modula. Možete napisati i dodavati svoje module.

Potrebno je prvo instalirati NodeJS sajta https://nodejs.org , ako već nemamo instaliranu ovu serversku JavaScript platformu.

Nakon instaliranja NodeJS instaliramo sve potrebne module,biblioteke,pakete (dependencies) uz pomoc komande ‘yarn’ ili ‘npm install’ i na klijenskoj i serverskoj strani.

Prvo pokrenuti server uz pomoć komande 'node index.js’, automatski ce se povezati sa mongo bazom podataka.



Zatim u folderu na klijentskoj strani pokrenuti iz pomoć komande ‘npm start’ .

# Aplikacija ce se pokrenuti na localhostu u vašem browseru.Kao na slici ispod.

# C:\Users\Buzejic\Downloads\Screenshot from 2019-05-26 23.46.42.jpeg

# 5. Literatura

1. <https://nodejs.org/en/>

2.<https://docs.npmjs.com/cli/install.html>

3. <https://reactjs.org/>

4.<https://medium.com/javascript-in-plain-english/full-stack-mongodb-react-node-js-express-js-in-one-simple-app-6cc8ed6de274>

5. <https://jenkins.io/doc/tutorials/build-a-node-js-and-react-app-with-npm/>

6. <https://expressjs.com/>

7. <https://stackoverflow.com>

8. <https://auth0.com/blog/react-tutorial-building-and-securing-your-first-app/>

9. <https://www.w3schools.com>