

KURSSIPROJEKTI

CT30A3203_01.06.2020 Web Applications

Samu Salko, 0598609

Ohjelma on toteutettu Noden Express-frameworkilla, ja tietojen tallennukseen käytetään MongoDB:tä (verkkopalvelu tai paikallinen). Expressin rungon mukaisesti ohjelmisto on toteutettu MVC-mallia mukaillen, joskin vastuualueet paisuivat paikoitellen suuriksi.

Parhaana puolena pitäisin käyttäjän kirjautumista ja autentikointia passport:ia sekä JSON web tokeneita hyväksikäyttäen. Aluksi käyttäjätili täytyy luoda sign-up -sivulla, jotta palveluun voi kirjautua. Kirjautumisen yhteydessä käyttäjälle annetaan tunnistautumiseen käyvä "token", joka sisältää tiedon käyttäjästä. Tämä tallennetaan käyttäjän puolella evästeisiin. Valitettavasti nämä tokenit eivät kovin nopeasti vanhene (ehkä koskaan), jolloin kerran luotu token käy aina yhden käyttäjän tunnistamisessa.

Ohjelma käyttää npm:ää, jolloin sen käyttämien pakettien lataaminen onnistuu helposti. Omalta koneelta vaatii latauksen Gitistä (<https://github.com/Kronkeli/Chatbox.git>), jonka jälkeen npm-komennot "npm install" sekä "npm start". Jos haluaa käyttää MongoDB:tä omalta koneelta, vaihtaa tiedostosta "app.js" muuttujan "mongoURL" arvon paikalliseksi (var mongoURL = "mongodb://localhost:27017/chatbox").

Luokkakaavio on kuvana. Pääpiirteittäin:

- postController.js
 - o Hallitsee "post"ien haun tietokannasta
- index.js (sekä auth.js)
 - o Vastaa käyttäjän luomisesta sekä kirjautumisesta
- wall.js
 - o Ohjaa "post"ien hakua ja näyttämistä wall.pug:ssa
- main.js (public/javascripts)
 - o Hallitsee XmlHttpRequestteja koskien "post"ien hakua ja päivittämistä.
- main.scss(->main.css)
 - o Tuotu mukaan Front-end-kurssilta tekemään frontista kauniimpi

Lisäpisteitä projektista:

- +5 Running your application in a Docker container (or having the application Docker compatible)
- +5 Running your application in Rahti
- +5 Using a database, such as Mongo, Redis, or any SQL-compatible
- +3 Use an ORM and models in backend, such as the Mongoose (MongoDB) or Sequelize (SQL)
- +3 Using XHR (also know as AJAX/AJAJ) for data transfer and frontend synchronization
- +5 User registration, authentication, and password storage
- -5 Minor bug, one crash during a non-essential feature, other small problem (käyttäjän inputteja ei validoitu/sanitoitu, koska en osannut tehdä sitä javascriptissä ennen setRequestHeaderia) -> ei voi laittaa esim. € julkaisuun
- +2 Sass:in käytöstä, joka jäi kuitenkin vähäiseksi.
- = 23 pistettä

Luokkakaavio kansioittain

