

## Kuvaus ohjelmasta

### Mitä ohjelma sisältää?

Harkkatyö on yksinkertainen verkkosivu, jossa käyttäjä pystyy luomaan käyttäjiä, kirjautumaan sisään, selaamaan koodin pätkiä, selaamaan kommentteja, sisään kirjautuneena käyttäjä pystyy myös luomaan uusia pätkiä sekä uusia kommentteja. Käyttäjää luodessa salasanan täytyy noudattaa vahvan salasanan määritelmiä. Liikkuminen eri sivujen välillä on toteutettu valikon avulla. Käyttäjät, pätkät ja kommentit tallennetaan MongoDB tietokantaan omina kokoelminaan.

### Miksi nämä asiat on sisällytetty ohjelmaan?

Ominaisuudet sisällytettiin ohjelmaan, jotta se sisältäisi hyödyllisiä ominaisuuksia ja täyttäisi minimi vaatimukset.

Ohjelma sisältää minimivaatimukset eli haetut pisteet ovat seuraavanlaiset.

Feature	Max points
Basic features with well written documentation	25
Timestamp when created is stored and shown with posts/comments	1
<b>Total</b>	<b>26</b>

## Ohjelman toteutus

### Ohjelman toteutukseen käytettiin seuraavia

- JavaScript
  - o pakollinen
- Node.js
  - o pakollinen
- Express
  - o ei pakollinen, mutta sitä on käytetty koko kurssin ajan.
- MongoDB
  - o jokin tietokanta oli pakollinen, MongoDB oli tietokanta, josta oli eniten kokemusta JavaScriptin yhteydessä
- mongoose
  - o Helpottamaan MongoDB:n käyttöä
- Morgan
  - o tulostaa html kutsuja, joka helpotti koodaamista
- Multer
  - o monirivisen formdatan lähettämistä varten
- JSON Web Token
  - o autentikaatiota varten

- bcrypt
  - o salasanan suolausta ja hashausta varten
- Pug
  - o pug oli mielestäni selkeä ja helppokäyttöinen template engine
- materialize
  - o sivu näyttää paljon paremmalta ja on responsiivinen eri kokoisilla näytöillä.

### Ohjelmistokehitystyökalut

Ohjelma kirjoitettiin Visual Studio Codessa. Git versionhallinta.

### Dokumentoitityökalut

Dokumentointiin käytettiin Microsoft Officea.

### Installation guidelines

- npm install
- npm start

### User manual

- Sivun näkyy osoitteessa <http://localhost:3000/>
- Sivuston kotisivulla käyttäjä näkee valikon, josta pystyy kirjautumaan sisään, rekisteröitymään, katsomaan julkaistuja koodin pätkiä.
- **Home** sivulla käyttäjä näkee valikon. Sisään kirjautuneena käyttäjä pystyy kirjautumaan ulos logout napista, joka tulee näkyviin home sivulla.
- **Register** sivulla käyttäjä pystyy luomaan uuden käyttäjän. Käyttäjänimenä toimii sähköpostiosoite ja salasan täytyy noudattaa vahvan salasan määritelmiä.
- **Login** sivulla käyttäjä pystyy kirjautumaan sisään luomallaan käyttäjällä ja salasanalla.
- **Posts** sivulla käyttäjä näkee listan julkaistuista koodinpätkistä, jos sellaisia on. Koodin pätkien perässä on linkki, josta näkee kyseisen postauksen kommentit. Jos käyttäjä on kirjautunut sisään sivun alaosaan, näkyy kenttä, jossa pystyy luomaan uusia postauksia.
- **Comments** sivulla käyttäjä näkee kommentit kyseiselle koodinpätkälle. Sisään kirjautuneena käyttäjä pystyy luomaan uusia kommentteja. Kommentin luomisen "Add comment for post with this title" -kenttään tulee sijoittaa halutun postauksen title. Esim. jos haluaa kommentoida koodinpätkää, jonka title on "statement" täytyy kyseiseen kenttään sijoittaa "statement".