

Ennakkomatskuja

1. Javascript

Kurssin alussa kerrataan javascriptia, HTML:ää sekä CSS:ää. Alla linkkejä ennakkoon näistä aiheista.

HTML: <https://www.w3schools.com/html/default.asp>

CSS: <https://www.w3schools.com/css/default.asp>

JS: <https://www.w3schools.com/js/default.asp>

Modern Javascript Tutorial: <https://javascript.info/>
<https://exploringjs.com/>

<https://2ality.com>

Ennakkoon tehtäviä, joilla voi mitata osaamistaan:

https://www.w3schools.com/js/exercise_js.asp

Ehdottomasti kannattaa aloittaa myös:

<https://www.freecodecamp.org/>

Tuolla voi suorittaa myös LinkedIniin lisättäviä sertifikaatteja

2. NodeJS

Kurssin pääasiallisena backendinä toimii NodeJS ja siihen liittyvät kirjastot.

Mikä on NodeJS:

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

NodeJS lataaminen omalle koneelle:

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

NodeJS:n perustoimintalogiikka:

<https://nodejs.org/en/guides/event-loop-timers-and-nexttick/>

Node Package Manager:

<https://nodejs.org/en/learn/getting-started/an-introduction-to-the-npm-package-manager>

Intro to Node apps:

<https://nodejs.org/en/learn/getting-started/introduction-to-nodejs>

3. Express

ExpressJS on minimalistinen NodeJS:n päällä toimiva web framework, joka huomattavasti helpottaa erilaisten webapplikaatioiden tekemistä. Se on perusrakenteeltaan yksinkertainen, mutta helposti laajennettava. Expressin ympärille on kehittynyt suuri joukko erilaisia kirjastoja, jotka toimivat Expressin kautta toistensa kanssa avustaen eri tyyppisten tehtävien, kuten käyttäjänhallinnan, sessionhallinnan, tietokantayhteyksien ja vastaavien kanssa.

ExpressJS:

<https://expressjs.com/>

Express Hello World:

<https://expressjs.com/en/starter/hello-world.html>

Perusrouttaus:

<https://expressjs.com/en/guide/routing.html>

Expressin kanssa toimivia muita kirjastoja:

<https://www.passportjs.org/>

<https://www.npmjs.com/package/connect-mongo>

<https://www.npmjs.com/package/express-session>

4. React

React on yksi suosituimmista frontend-kirjastoista. Se on Facebookin kehittämä ja Vercelin ylläpitämä. React on minimalistinen eli se keskittyy vain siihen miltä UI näyttää ja miten itse UI toimii. Kaikki muu toiminnallisuus tuodaan React applikaatioon muiden kirjastojen kautta. React on komponenttipohjainen eli koodia ei tarvitse toistaa komponentteja käytettäessä.

React:

<https://react.dev/>

Nykyinen React lataustyökalu:

<https://vitejs.dev/>

(Create React Appia ei enää kehitetä)

Miten React applikaatioiden koodamiseen lähdetään:

<https://react.dev/learn/thinking-in-react>

Reactin oma "html" eli JSX:

<https://react.dev/learn/writing-markup-with-jsx>

Lisäkirjastot, joita käytetään kurssin aikana:

React Router eli "client side routing" fronttirouttausta, juuri muuten päivitetty:

<https://reactrouter.com/en/main>

Uusi tapa React Routerille on luoda erikseen Router eikä käyttää ns suoraan purkista BrowserRouteria.

Tutoriaali uudesta tavasta toimia:

<https://reactrouter.com/en/main/start/tutorial>

Redux ja Redux Toolkit eli applikaation tilanhallintaa:

<https://redux.js.org/>

Reduxiin aputyökaluna Redux Toolkit, joka poistaa turhaa bulkkikoodia.

<https://redux-toolkit.js.org/>

5. MongoDB

MongoDB on dokumenttipohjainen NoSQL kanta, jonka erityinen vahvuus on webbapplikaatioissa sen nopeiden lukuominaisuuksien takia. Kurssilla käytetään MongoDB:n tarjoamaa pilvipalvelua (ilmainen, ei luottokorttia) Mongo Atlasta. Lisäksi käytetään NodeJSn kirjastoa MongooseJS hoitamaan yhteydet Mongoon:

MongoDB: <https://www.mongodb.com/>

Tutoriaali: <https://www.tutorialspoint.com/mongodb/index.htm>

MongooseJS: <https://mongoosejs.com/>

