

Harjoitustyön raportti- Reenikirja

Jaakko Ikäheimo TVT19KMO

Sivun osoite: <http://www.students.oamk.fi/~c8ikja00/reenikirja/> (suositellen Chromea/Firefoxia)

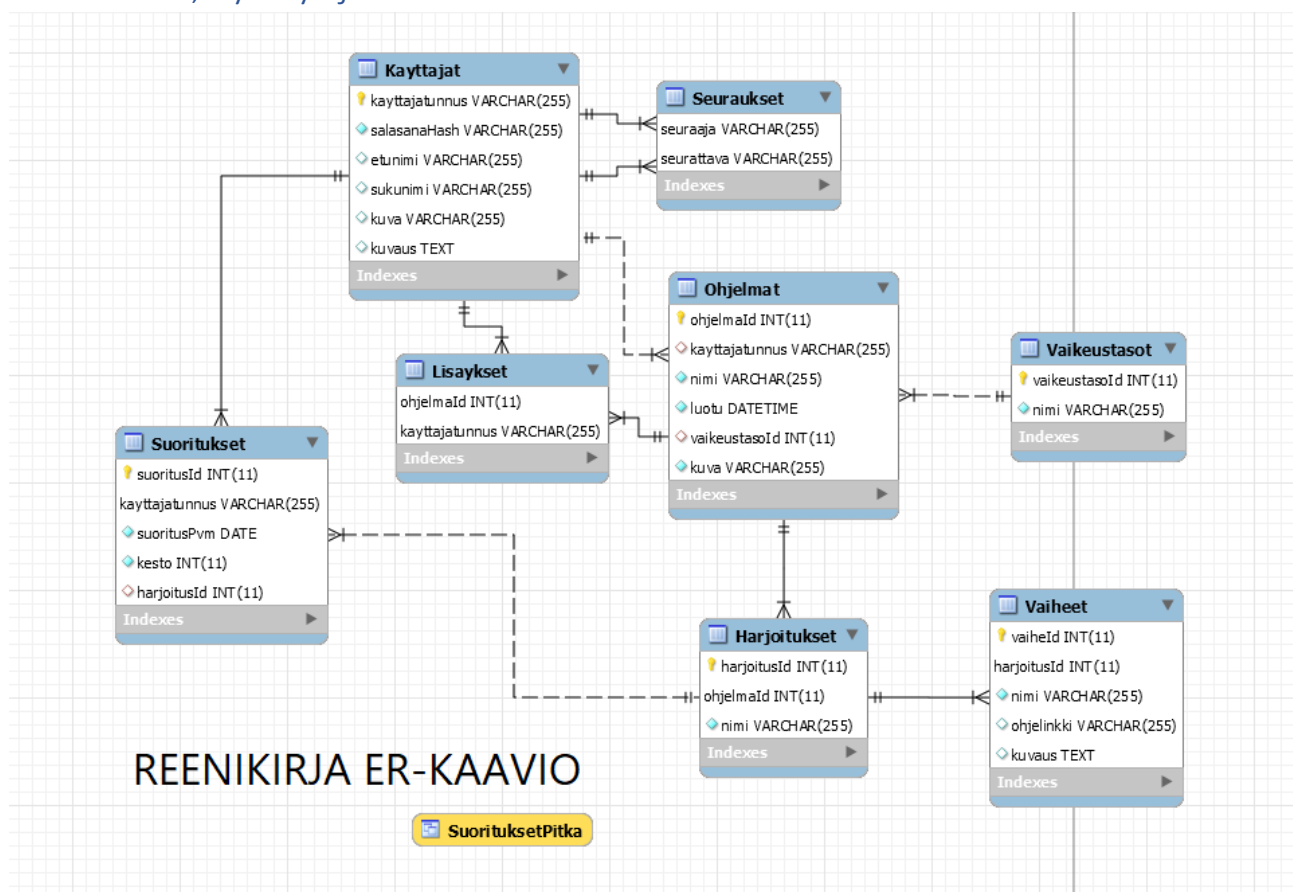
Sivun API: <http://www.students.oamk.fi/~c8ikja00/reenikirja/Api/> (tuolta voi valita nuo "reitit" suoraan)

GitHub-osoite: <https://github.com/JIkaheimo/ohjelmistokehityksen-perusteet-s19>

Yleistä

Harjoitustyöni aiheena oli toteuttaa Reenikirja-nettisivusto, minkä avulla kirjautuneet käyttäjät pystyvät luomaan, lisäämään ja selailemaan muiden käyttäjien luomia treeniohjelmia. Käyttäjät pystyvät myös selaamaan toistensa julkisia profiilisivuja, sekä kirjaamaan ylös suorittamiaan ohjelmien harjoituksia.

Tietokanta, kyselyt ja API



!HOX! Kaikki tietokantaan ja tietokannan kanssa kommunikointiin tarvittava löytyy Api-kansion alta.

Reenikirjan tietokannan rakenne on aika simppele. **Käyttäjät**-taulu pitää kirjata sovellukseen rekisteröityneistä käyttäjistä. Käyttäjillä tulee olla rekisteröityessään uniikki käyttäjätunnus, sekä salasana

myöhempiä kirjautumista varten. Käyttäjä-taulun tietueet ovat riippumattomia tietokannan muista tietueista.

Seuraukset-taulu sisältää tiedon käyttäjien seuraamista muista käyttäjistä (voi seurata myös itseään). Tässä seuraaja ja seurattava muodostavat taulun pääavaimen, sillä käyttäjä ei pysty seuraamaan toista moneen kertaan. Samalla seurausta ei pysty olemaan olemassa ilman seurattavaa ja seuraavaa käyttäjää. Käyttäjä pystyy seuraamaan tässä montaa käyttäjää ja seuratulla pystyy olemaan monta seuraajaa.

Ohjelmat-taulu sisältää tiedon käyttäjien luomista treeniohjelmissa. Treeniohjelmalla tulee olla ainakin nimi, kuva (oletuksena käytetään placeholder-kuvaa) sekä hetki minä ohjelma on luotu. Ohjelma pystyy luomisensa jälkeen olla olemassa ilman sitä luonutta käyttäjää. Käyttäjällä pystyy myös olemaan luotuna monta ohjelmaa, muttei ohjelmalla kuin vain yksi käyttäjä. Tämän päätöksen takana oli se, että ohjelma halutaan pitää tietokannassa, vaikkei sitä luonutta käyttäjää enää olisi olemassa.

Lisaykset-taulu pitää yllä tietoa käyttäjien lisäämistä (treeni)ohjelmissa. Lisäystä ei pysty olemaan olemassa ilman että sillä on käyttäjä ja ohjelma. Käyttäjällä pystyy olemaan lisättynä monta eri ohjelmaa ja ohjelma pystyy olla lisättynä monella eri käyttäjällä.

Harjoitukset-taulu sisältää tiedon ohjelmien eri harjoituksista. Harjoituksella tulee olla nimi sekä sen sisältävän ohjelman id. Harjoitusta ei pysty olemaan ilman ohjelmaa. Tässä ohjelmalla voi olla monta harjoitusta, mutta harjoituksella vain yksi ohjelma.

Vaiheet-taulu on periaatteessa samantapainen kuin Harjoitukset-taulu... Vaiheella pystyy olemaan vain yksi harjoitus, mutta harjoituksella monta vaihetta. Vaihetta ei pysty olemaan olemassa ilman sitä sisältävää harjoitusta. Vaiheella tulee olla ainakin sen sisältävän harjoituksen id, ja nimi.

Suoritukset-taulu pitää yllä tietoa käyttäjien suorittamista harjoituksista. Käyttäjällä pystyy olemaan monta suoritusta, mutta suorituksella vain yksi käyttäjä. Samalla tavalla suorituksella voi myös olla vain yksi (tai 0) harjoitus, mutta harjoituksella monta suoritusta. Suoritukset halutaan tässä säilyttää, vaikka siihen liitetty harjoitus olisikin poistettu...

Vaikeustasot-taulu sisältää ohjelmien eri vaikeustasot. Vaikeustasolla tulee olla uniikki nimi (olisi voinut käyttää pääavaimena) sekä id. Tämä taulu on muista riippumaton...

Taulujen luonti: Löytyy create.sql -tiedostosta

Taulujen täyttö: Löytyy insert.sql -tiedostosta.

Taulujen kyselyt: Nämä löytyvät projektin Api -> Queries kansion alla olevien tiedostojen yläreunasta. Näissä toki pitää korvata :-alkuiset sanat oikeilla arvoilla... Samoista tiedostoista löytyy myös kyselyjen muodostus PHP:n avulla.

API: Tämä on periaatteessa bonusta. Heti Api-kansion alta löytyvät PHP-tiedostot muodostavat sovelluksen API:n, mitä pystyy testaamaan esimerkiksi seuraavilla osoitteilla:

Haetaan kaikki ohjelmat:

<http://www.students.oamk.fi/~c8ikja00/reenikirja/Api/ohjelmat.php>

Haetaan tietty ohjelma sisällön kera:

<http://www.students.oamk.fi/~c8ikja00/reenikirja/Api/ohjelmat.php?id=4>

Haetaan serverillä generoitu HTML-elementti:

<http://www.students.oamk.fi/~c8ikja00/reenikirja/Api/suoritukset.php?id=3&tr=3>

Pohdintaa

Onnistuin toteutuksessa mielestäni yllättävän hyvin, ottaen huomioon omien töiden määrän sekä työlle annettujen deadlinejen aikavälin. Sain melkein kaikki sivuston toiminnot valmiiksi, lukuunottamatta ohjelmien kuvien kustomointia. Sain myös toteutettua harjoituksen sivutuotteena pienimuotoisen "REST"-apin, ja muutinkin siksi sivustoni kommunikoimaan tietokannan kanssa AJAXin avulla.

Olin ennen harjoitusta käyttänyt jo jonkin verran HTML + CSS + JavaScriptia, joten ne olivat entuudestaan jo minulle tuttuja. PHP:n käyttö oli itselleni vähemmän tuttua, joten sen opettelu oli minulle ihan uutta, mutta samalla mielenkiintoista. Aiemmin olen käyttänyt näissä hommissa enimmäkseen nettikehityshommissa MERN-pinoa.

Nettisivuston ulkoasua olisi vielä voinut hioa mielestäni mm. mobiiliystävällisemmäksi ja lisätä jonkin sortin animaatioita, kun esimerkiksi taulukon rivien "häivyttäminen" poistettaessa. CSS ja JavaScriptin polyfillaus vanhoille selaimille ja Internet Explorerille jäi myös vähän kesken, suosittelenkin siksi käyttämään sivustoa vaikka Chromella tai Firefoxilla. Nettisivuston kehityksen myötä heränneiden "lisäominaisuuksien" implementointi mutkitti myös hieman kehitysprosessia.

Osa suunnittelemastani JavaScript-funktionaalisuudesta jäi myös toteuttamatta, kuten taulukoiden poistaminen, jos yhtään riviä ei jää jäljelle. Nämä kuitenkin ovat aika pieni seikkoja harjoituksen kannalta, sillä päätarkoituksenahan tässä oli käyttää HTML + CSS + PHP.