Tekninen määrittely/ suunnitteludokumentti

1pOHKE projektityö, Urheiluvälinevuokraamo

Tekijä: Jari Mikkonen

Sisällysluettelo

[1 Johdanto 3](#__RefHeading___Toc3060_1377255538)

[1.1 Dokumentin sisältö 3](#__RefHeading___Toc3062_1377255538)

[1.2 Dokumentin tarkoitus 3](#__RefHeading___Toc3064_1377255538)

[2 Palvelun yleiskuvaus 3](#__RefHeading___Toc3066_1377255538)

[2.1 Laiteympäristö 3](#__RefHeading___Toc3068_1377255538)

[2.1 Ohjelmointiympäristö 3](#__RefHeading___Toc3070_1377255538)

[3 Palvelun tekninen toteutus ja arkkitehtuuri 3](#__RefHeading___Toc3072_1377255538)

[3.1 Palvelun komponentit 4](#__RefHeading___Toc3074_1377255538)

[3.1.1 Tallennuspalvelu 4](#__RefHeading___Toc3076_1377255538)

[3.1.2 Taustapalvelu 4](#__RefHeading___Toc3078_1377255538)

[3.1.3 Käyttöliittymä 4](#__RefHeading___Toc3080_1377255538)

[3.2 Arkkitehtuuri 4](#__RefHeading___Toc3082_1377255538)

[4 Tietokannan arkkitehtuuri 6](#__RefHeading___Toc3084_1377255538)

[4.1 Tietokannan tekninen toteutus 6](#__RefHeading___Toc3086_1377255538)

[4.2. Tietokannan taulut ja kentät 7](#__RefHeading___Toc3088_1377255538)

[5. Liitteet 7](#__RefHeading___Toc3090_1377255538)

# 1 Johdanto

Tässä luvussa käydään läpi dokumentin sisältö ja tarkoitus.

## 1.1 Dokumentin sisältö

Tämä dokumentti sisältää palvelun yleiskuvauksen, tietoa palvelun teknisestä toteutuksesta sekä arkkitehtuurista

## 1.2 Dokumentin tarkoitus

Dokumentin tarkoituksena on piirtää kuvaa toimintatavoista, joilla palvelu on toteutettu.

# 2 Palvelun yleiskuvaus

Tässä luvussa käydään läpi palvelua seuraavilla tasoilla: Laiteympäristö sekä ohjelmointiympäristö.

## 2.1 Laiteympäristö

Palvelu on suunniteltu ajettavaksi docker-konteissa. Docker mahdollistaa ohjelmistokehityksen niin, että ajoympäristö pysyy samana riippumatta siitä millä alustalla ohjelmistoa ajetaan. Sovellus on tehty käyttäen Dockerin versiota 20.10.5.

Palvelu pyörii tällä hetkellä Amazon EC2-virtuaalipalvelimella, jossa käyttöjärjestelmänä on Debian 5.10.0-21-cloud-amd64.

## 2.1 Ohjelmointiympäristö

Ohjelmointiympäristöksi on valikoitunut avoimen lähdekoodin NodeJS-alusta. NodeJs on alustariippumaton ajoympäristö, jolla voidaan ajaa javascript-koodia. Ohjelmointikielenä on käytössä TypeScript, joka mahdollistaa staattisen tyypityksen javascriptiin sekä joka parantaa koodin luettavuutta ja käytettävyyttä. TypeScript-koodi transpiloidaan kääntäjällä normaaliksi javascript-koodiksi, jota Node-alusta osaa ajaa.

# 3 Palvelun tekninen toteutus ja arkkitehtuuri

Tässä luvussa käydään läpi palvelun teknistä toteutusta ja arkkitehtuuria.

## 3.1 Palvelun komponentit

Palvelu on tehty kolmesta komponentista jotka ovat tallennuspalvelu, taustapalvelu sekä käyttöliittymä.

### 3.1.1 Tallennuspalvelu

Tallennuspalvelu on tehty MariaDB relaatiotietokannalla.

### 3.1.2 Taustapalvelu

Taustapalveluun on valikoitunut ExpressJs-taustaverkkosovelluskehys. Express mahdollistaa RESTful-rajapintojen rakentamisen javascript-ohjelmointikielellä. Taustapalvelun vastuulla on toimia välittäjänä tietokannan ja käyttöliittymän välillä.

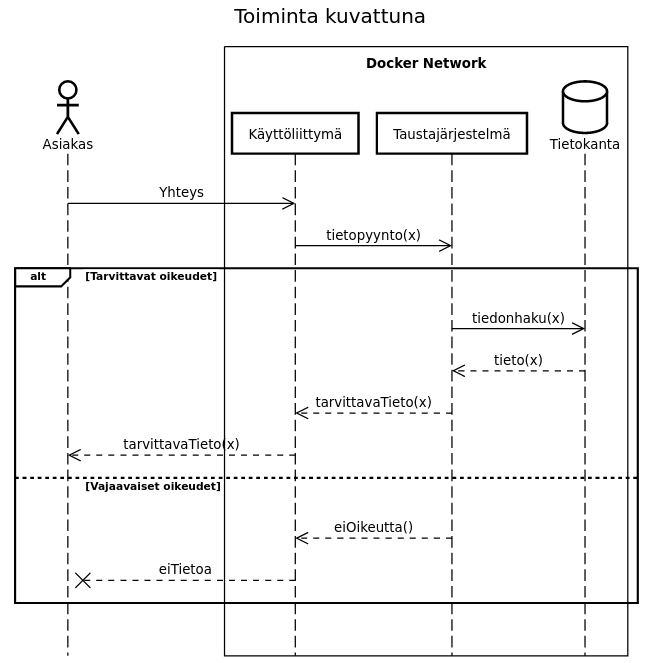
Taustapalvelussa on yksi erittäin kriittinen osa, käyttäjätietojen hallinta. Käyttäjän salasanat eivät saa olla luettavissa ulospäin pelkkänä tekstinä vaan salasanat on kryptattava. Salasanojen kryptaamiseen on valikoitunut Bcrypt-kirjasto, jota voidaan käyttää Node-alustalla. Bcrypt-kirjasto salaa käyttäjän salasanan käyttäen apuna “suolakierroksia” sekä merkkijonoa, joka on ainoastaan taustajärjestelmän tiedossa. Tämän kirjaston avulla salasanat ovat tallennettuna tietokantaan kryptattuna, joten ulkopuolisen ei ole mahdollista saada käyttäjän salasanaa selville vaikka joku onnistuisi kaappaamaan tietokannan.

### 3.1.3 Käyttöliittymä

Palvelun käyttöliittymä on kuten taustajärjestelmä Node-alustan päällä. Käyttöliittymän rakennukseen on käytetty React-kirjastoa, jolla käyttöliittymäsuunnittelu on erittäin helppoa.

## 3.2 Arkkitehtuuri

Palvelun arkkitehtuuri on tehty käytettäen Docker-”kontteja”. Dockerin avulla on luotu yksityinen verkko, johon ulkopuolelta ainoastaan käyttöliittymään on pääsy. Taustajärjestelmä pitää huolen siitä, että palvelun asiakkaalla on oltava tarvittavat oikeudet pyydettyyn tietoon. Alla olevassa kuvassa 1 on havannollistettu järjestelmää.



Kuva 1.

# 4 Tietokannan arkkitehtuuri

Tässä luvussa käydään läpi tietokanta-arkkitehtuuria

## 4.1 Tietokannan tekninen toteutus

Palvelun tietokanta on rakennettu käyttäen MariaDB-relaatiotietokantaa. Tietokanta ajetaan docker-kontissa, johon ainoastaan taustajärjestelmällä on pääsy.

## 4.2. Tietokannan taulut ja kentät

Tietokannan taulut ja kentät ovat kuvattu liitteessä 1.

# 5. Liitteet

Liite 1. Tietokantakaavio