Tietoliikenteen sovellusprojekti, syksy 2024: kiihtyvyysanturidataa käsittelevä järjestelmä

Projektin tavoitteena oli tehdä nRF5340-kehitysalustalle ohjelma, joka ottaa dataa kiihtyvyysanturilta, tekoälyn avulla päättelee siitä anturin asennon ja lähettää tämän langattomasti IoT-laitteelle, joka välittää tiedon eteenpäin tietokantaan. Tietokannasta tiedot pystyy lukemaan HTTP-rajapinnasta. (Kuva 1.)

A diagram of a computer

Description automatically generated

KUVA 1. Järjestelmän arkkitehtuuri

Lisäksi kurssilla opeteltiin Scrum- ja Kanban-metodien käyttöä projektin yhteydessä.

Projekti suoritettiin toisen vuosikurssin syksyllä OAMK:n tietotekniikan koulutusohjelman laiteläheisen ohjelmistokehityksen opinnoissa.

# Ryhmän jäsenet

[@Nornalite](https://github.com/Nornalite/)

[@baarnuo](https://github.com/baarnuo/)

# Projektin vaiheet

*A computer with wires connected to it

Description automatically generated*

KUVA 2. Laitteistomme

## nRF 5340DK

Luimme Nordicin kehitysalustalla kiihtyvyysanturilta ADC-kirjaston avulla suuntadataa, ja hyödynsimme alustan Bluetooth Low Energy -yhteyttä anturidatan eteenpäin lähetykseen. Loimme alustalle kiihtyvyysanturipalvelun, jota laite mainostaa ympäröiville BLE-laitteille. Ulkoinen laite voi muodostaa yhteyden alustaamme tunnistettuaan sen palvelun UUID-numeron perusteella ja tilata mittaustiedot. Aina mittauksen suoritettuaan alustamme lähettää mittaustulokset sisältävän ilmoituksen tilaajalle.

Aloituspisteinä toimivat opettajan tarjoama adc-lukuohjelma ja Nordic Semiconductorin Bluetooth Low Energy Fundamentals -kurssin esimerkit.

## Koneoppiminen

Opetimme ja toteutimme kaksi koneoppimisalgoritmia suuntatiedon laskemiseen.

Ensisijaisena algoritmina toimi k-keskiarvoklusterointi, jossa arvotut datakeskittymien keskipisteet hakeutuvat kierros kierrokselta kohti lähimpiä datapisteitä (kuva 3.) Algoritmin luomisen ja koulutuksen jälkeen toimme lasketut keskipisteet .h-tiedostoon, josta käytimme niitä luetun datan lähimmän keskipisteen eli suunnan määritykseen.

**A graph of a graph of a graph

Description automatically generated with medium confidence**

KUVA 3. K-keskiarvoklusterointialgoritmi laskennan ensimmäisen kierroksen jälkeen

Lisätehtävänä opetin Tensorflow-kirjaston avulla yksinkertaisen neuroverkon lajittelemaan datapisteitä (kuva 4.) Hyviä tuloksia antoi herätteiden luokitteluun tarkoitetun Catergorical Crossentropy -häviöfunktion ja suurista korjausliikkeistä aloittavan, jatkuvasti tarkentuvan Exponential Decay -optimisaattorin käyttö. Opetuksen jälkeen toin painokertoimet ja bias-arvot .h-tiedostoon, josta niitä sai käytettyä nRF 5340 -alustalle luodussa neuroverkkoa matkivassa ohjelmassa mittaustulosten luokitteluun.

A screenshot of a computer program

Description automatically generated

KUVA 4. Neuroverkkomalli

## Raspberry Pi

Etsimme bleak-kirjastoon pohjautuvalla ohjelmalla kiihtyvyysanturipalvelumme ympäröivien BLE-mainosten joukosta. Ilmoituksia vastaanottaessaan ohjelmamme purkaa datan ja lähettää sen eteenpäin tietokantaamme mysql-kirjaston avulla.

## Linux-serveri ja MySQL-tietokanta

Saimme koululta allokoidut henkilökohtaiset Linux-serverit, joita käytimme Linuxin opetteluun (ja sitä tukevien lisätehtävien tekoon). Loimme servereillemme myös omat MySQL-tietokannat datan säilytykseen sekä Apache-serveriin perustuvia HTTP-rajapintoja datan hakemiseen tietokannasta.

## Muut taidot

Kurssilla seurattiin Scrum-metodologian oppeja; projekti jaettiin pieniin, viikottaisiin osioihin ja osioiden edistymistä seurattiin päivittäisillä ryhmäpalavereilla sekä viikottaisilla useampien ryhmien kokouksilla. Tehtävien seuraamisessa apuna oli GitHubin projektiosion Kanban-taulu.

# Yhteenveto

Projekti sujui hyvin ja pysyimme projektin aikana aikataulussa. Saimme sekä projektin perusosion että lisätehtävät valmiiksi.