

Timo Leinonen

Anturit ja mobiilisovellukset

Insinööri (AMK)

Tieto- ja viestintäteknikka

Kevät 2018



KAJAANIN
AMMATTIKORKEAKOULU
UNIVERSITY OF APPLIED SCIENCES

Tiivistelmä

Tekijä(t): Leinonen Timo

Työn nimi: Anturit ja mobiilisovellukset

Tutkintonimike: Insinööri (AMK), tieto- ja viestintätekniikka

Asiasanat: BLE, tiedonsiirto

Tässä opinnäytetyössä laadittiin langatonta tiedonsiirtoteknologiaa BLE:tä (Bluetooth Low Energy) soveltava järjestelmä. Järjestelmässä BLE-laitteiden avulla kerätään ja lähetetään tarvittavaa dataa Android-käyttöjärjestelmän tabletin mobiilisovellukseen, käsittelyä ja tulostamista varten. Opinnäytetyössä tutustuttiin samalla eri anturisovelluksiin, mobiilisovelluksien kehitysympäristöihin ja langattomiin lyhyen kantaman tiedonsiirtoteknologioihin.

Järjestelmän toteuttamiseksi käytettiin aiempaa KajaPro-yrityksen ohjelmistotuotteella laadittua mobiilisovellusta ja kahta BLE-laitetta. Tuotteen avulla mobiilisovellus pystyy muodostamaan käyttöliittymäpalvelun kautta yhteyden työn järjestelmässä käytettävään BLE-laitteeseen.

Tässä työssä mobiilisovelluksen toimintaa jouduttiin parantamaan. Työssä perehdyttiin BLE-protokollakerroksen määrittämiin laiterooleihin ja BLE-laitteet asetettiin työn kannalta tarpeellisiin rooleihin. Laitteiden rooleiksi valittiin lähettäjä ja yhdistelmälaitteen roolit. Lähettäjäroolissa mainostetaan dataa tarkkailevalle yhdistelmäroolin laitteelle. Yhdistelmäroolista lähetetään kerätty data yhteydessä olevan mobiilisovelluksen käsittelyyn. Roolien aikaansaamiseksi internetistä ladattiin esimerkkiprojekteja laitteiden rooleista. Laitteiden ja mobiilisovelluksen kehitystä tehtiin niiden kehitykseen tarkoitetuissa kehitysympäristöissä. Kehityksessä ohjelmointikielenä käytettiin C++:aa.

Järjestelmän BLE-laitteiden roolit vaihdettiin onnistuneesti laiterooleihin. Lähettäjäroolin laitteesta mainostetaan dataa oikein ja yhdistelmärooliin saadaan yhteys mobiilisovelluksesta.

Abstract

Author(s): Leinonen Timo

Title of the Publication: Sensors and Mobile Applications

Degree Title: Bachelor of Engineering, ICT engineer

Keywords: BLE, communication

In this Bachelor's thesis, a system was developed which uses wireless communication BLE (Bluetooth Low Energy) technology. In the system, BLE devices were utilized to collect and send necessary data to a tablet that uses Android operating system. The tablet has a mobile application which processes the received data and prints it out. The thesis also examined different sensor applications, development environments for mobile applications and short range wireless communication technologies.

A prior mobile application which was built with KajaPro Company's own software tool and two BLE devices were used for developing the system. The software product ensures that through a user interface the mobile application can establish a connection to BLE device used in the system

In the thesis, the mobile application's operation had to be improved. Familiarizing with the device roles that BLE protocol layers define was needed and the BLE devices were set roles, necessary for the system. Broadcaster and combination roles were chosen to the devices. Broadcaster broadcasts its data to the observing combination role device. The device in combination role sends the collected data to the mobile application, which has created a link with it, for processing. Example projects of device roles were downloaded from the internet to set up the roles for the devices. The development of the devices and mobile application were done in development environments meant for development. C++ programming language was used in the development.

Changing the roles of the devices in the system was successful. The device in broadcaster role broadcasts the data in the right way and the mobile application manages to create a connection with the device in the combination role.

Sisällys

1	Johdanto	1
2	Mobiilisovelluksia hyödyntäviä anturisovelluksia	2
2.1	Aktiivisuuden seuranta	4
2.2	Synkroninen jalkojen paineen mittausjärjestelmä	6
2.3	Terveydenhuolto & GlucoWise	7
2.4	Beacon.....	8
2.5	IIoT & SmartBond.....	10
2.6	MetaWear	11
3	Kehitystyökaluja	12
3.1	Eclipse	12
3.2	Android Studio.....	13
4	Langaton tiedonsiirto.....	15
4.1	BT	15
4.2	BLE	16
4.3	ANT/ANT+	18
5	Langattomassa tiedonsiirrossa käytettäviä protokollia	19
5.1	UDP	19
5.2	TCP.....	19
6	Työn tausta, tavoite ja käytettävät välineet	20
6.1	ReGui.....	20
6.2	CC2650 SensorTag	22
6.3	Code Composer Studio	22
6.4	Visual Studio	23
7	Työn suunnittelu, toteutus ja testaus	24
7.1	BLE-laiteroolit.....	25
7.2	Työn kulku.....	26
7.3	Roolien testaaminen	29

8	Jatkokehitys	32
9	Työn tavoitteiden saavuttaminen	33
10	Yhteenveto	34
	Lähteet	35

Symboliluettelo

BLE	Bluetooth Low Energy
BT	Bluetooth
IDE	Integrated Development Environment
IoT	Internet of Things
IIoT	Industrial Internet of Things
ReGui	Remote Graphical User Interface
TCP	Transmission Control Protocol
UDP	User Datagram Protocol

1 Johdanto

Tämän opinnäytetyön toimeksiantajana toimii KajaPro-yritys. KajaPro on Kajaanissa sijaitseva yritys, joka tarjoaa ohjelmistokehityksen alihankintapalveluja ja asiantuntijapalveluja. Yrityksen ohjelmistokehityksen erikoistaitoihin kuuluvat ohjelmistoratkaisut liittyen matkapuhelimiin, tukiasemiin ja tietoliikenteeseen. Yrityksellä on myös osaamista sulautetuista järjestelmistä. [1.] Yrityksen toimitusjohtaja on Arto Karjalainen.

Opinnäytetyön taustana on KajaPro-yrityksessä suoritettu työharjoittelu. Työharjoittelun aikana tutkittiin ja perehdyttiin langattoman tiedonsiirtoteknologian BLE-(Bluetooth Low Energy) toimintaan. Kuvan saamiseksi teknologian toiminnallisuudesta Android-sovelluksia kehitettiin BLE:tä käyttävän laitteen ympärille. Sovellukset laadittiin Android kehitykseen tarkoitetuilla työkaluilla.

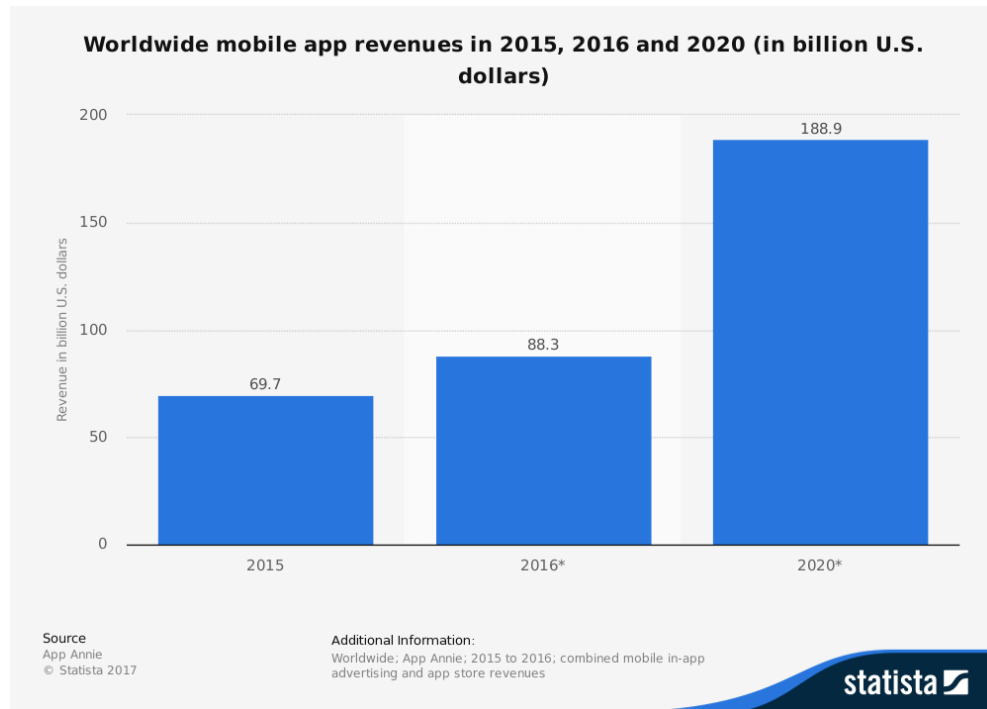
Opinnäytetyön tavoitteena on toteuttaa BLE-teknologiaa hyödyntävä järjestelmä, missä kahdella BLE-teknologiaa soveltavalla laitteella lähetetään tarvittavaa dataa Android-käyttöjärjestelmää käyttävälle tabletille. Tabletti käyttää valmista, työharjoittelun aikana laadittua mobiilisovellusta laitteen datojen vastaanottamiseen ja käsittelyyn. Samalla tässä opinnäytetyössä tarkoituksena on tutkia, millaisia anturisovelluksia on saatavilla mobiilisovelluksiin, mitä IDE:tä (Integrated Development Environment) on mobiilisovelluksien tekoon ja mitä käytettäviä lyhyen kantaman langattomia tiedonsiirtoteknologioita on olemassa.

Opinnäytetyön lisätavoite oli laatia työn järjestelmään pilvipalvelutoteutus. Mobiilisovelluksesta siirretään saatu data pilveen, josta tiedot ovat haettavissa. Tätä toiminnallisuutta ei lähdetty työn järjestelmään laatimaan, sillä muuten työstä olisi tullut liian laaja.

Opinnäytetyön toteutuksessa hyödynnettiin KajaPro-yrityksen omaa ohjelmistotuotetta. Tuotetta on käytetty tabletin mobiilisovelluksen laatimiseen. Ohjelmistotuote mahdollisti käyttöliittymäpalvelun rakentamisen ja yhteyksien luomisen mobiilisovelluksen ja BLE-teknologiaa käyttävien laitteiden välille. Työssä BLE-laitteiden kehitys tapahtui niiden kehitykseen tarkoitettussa IDE:ssä. Mobiilisovellusta kehitettiin työssä toimimaan paremmin ja kehitys tapahtui vastaavasti omassa IDE:ssä. Ohjelmointikielinä laitteiden ja mobiilisovelluksen kehittämisessä käytettiin C++:aa.

2 Mobiilisovelluksia hyödyntäviä anturisovelluksia

Mobiilisovellukset ovat hallinneet markkinoita. Smart Insihtin mukaan 90 prosenttia ihmisten käyttämästä ajasta mobiililaitteilla kuluu mobiilisovellusten käyttöön. [2.] Alla olevassa kuvassa on esitetty maailmanlaajuisesti mobiilisovellusten tuottamat tulot Yhdysvaltain dollareissa vuosilta 2015, 2016 ja 2020.



Kuva 1. Mobiilisovellusten tulot vuosilta 2015, 2016 ja 2020. [3.]

Kuvassa 1 vuoden 2015 mobiilisovellusten tulo oli 69,7 miljardia Yhdysvaltain dollaria. Vuotta myöhemmin tulot kasvoivat 88,3 miljardiin dollariin. Vuoteen 2020 mennessä mobiilisovellusten on ennustettu tuottavan 188,9 miljardia dollaria tuloina. [3.]

Mobiilisovelluksilla on ollut vaikutusta IoT:n (Internet of Things) kasvuun. Sanalla IoT tarkoitetaan sitä, että kaikkiin laitteisiin on kiinnitetty anturi, jossa datan lähetys ja vastaanotto tapahtuvat internetin yli. Tällä hetkellä mobiilisovellukset toimivat päätepesteinä eli niitä käytetään esimerkiksi musiikin kuuntelussa ja sosiaalisen median käytössä. Päätepesteiden sijaan mobiilisovellukset mobiililaitteissa, esimerkiksi älypuhelimet ja tabletit, toimivat väylänä, jota kautta IoT:hen pohjautuvia laiteita voidaan hallita ja seurata niiltä tulutta dataa, esimerkiksi huoneen lämpötilaa. Mobiilisovelluksia voidaan käyttää myös itse anturina hyödyntämällä mobiililaitteen sisäänrakennettuja ominaisuuksia. Esimerkiksi pu-

helimen GPS-paikannintietoja käytetään ihmisen sijainnin määrittämisessä, joka välitetään taustalla. Hyödyntämällä mobiilisovelluksia IoT-ympäristössä luodaan rakennetta antureiden tiedon kululle esineiden läpi vastaanottavalle mobiilisovellukselle tosiajassa. [2.] [4.]

Anturitekniikkaa käytetään nykypäivänä lähes kaikkialla jokapäiväisessä elämässä esimerkiksi turvallisuuden seuraamisessa. Anturitekniikan käyttö mikroskooppi- ja nanotekniikalla antaa anturitekniikkaa hyödyntäville järjestelmille sovelluksen lääketieteen alalla lääkeaineiden seurantavälineenä, joka parantaa potilaiden kriittistä hoitamista. Lääketieteen alalla anturitekniikkaa käytetään myös diagnostiikka- ja huumeidenvalvontatyökaluina. BBC Research on laatinut tutkimuksen, jossa tutkitaan anturitekniikan kasvua. Tutkimus myös tarjoaa anturitekniikan käyttäjille tietoa anturitekniikan mahdollisista tulevaisuuden käyttökohteista. Kun tutkimustyötä suoritettiin antureiden historiasta ja käyttökohteista, arvioitiin anturitekniikan tuottamia tuloja markkinoilta maailmanlaajuisesti. Tulojen määrittämiseksi tietoja kerättiin antureita tuottavien yritysten henkilökunnalta. Antureiden tuottamat tulot vuonna 2015 olivat 110,4 miljardia Yhdysvaltain dollaria. Tulojen määrään uskotaan kasvavan vuoteen 2022 mennessä 240,3 miljardiin dollariin. [5.]

Verrattuna tavalliseen anturiin älykäs anturi kykenee muokkaamaan havaittuja tuloksia ja tuottamaan tulkitsevampaa tietoa. Älykkäiden antureiden tuottama tieto auttaa parantamaan anturijärjestelmän suorituskykyä. Tuloksien eheydellä ja luotettavuudella luodaan sovelluksia muun muassa lääketieteeseen ja kuluttajien tuotteisiin. Eräs esimerkki tuotteesta lääketieteessä, joka jo soveltaa älykästä anturitekniikkaa, on PillCam. PillCam on israelilaisen yrityksen kehittämä. Pillerin kokoisen kameran avulla voidaan saada kuvaa ruokatorvesta ja muista pienistä elimistä. Näin rauhoitteiden ja endoskooppisten menetelmien käyttöä tähystyksessä ei tarvita. [5.]

Älypuhelimissa käytetään useita antureita, kuten kiihtyvyyssanturia ja gyroskooppia. Älypuhelimien sisältämien antureiden määrän odotetaan kasvavan tulevaisuudessa. Älypuhelimet sisältäisivät esimerkiksi ilmanpaine- ja kiihtyvyyssantureita älypuhelimien käyttäjän aktiviteettien seuraamiseen. Tutkimusta tehdään myös, kuinka tietyillä älypuhelimien sijoitettavilla antureilla havaitaan ympärillä olevia saasteita sen hetkessä ympäristössä. Tutkijat arvioivat, että vuoteen 2050 mennessä älykkäät anturit ja muut vastaavat anturijärjestelmät muodostavat suuren osan maailman jatkuvasti kasvavasta taloudesta. [5.]

Jos laitteella on kyky muodostaa yhteys älypuhelimien tai tablettiin, pidetään sitä älykkäänä. Älykkäillä laitteilla jokapäiväiset tehtävät helpottuvat ja nopeutuvat; esimerkiksi sy-

dämen sykettä voidaan seurata älykellon kautta. Älykkäitä laitteita käytetään terveydenhuollon-, IT- ja turvallisuuden aloilla. Älykäs laite koostuu antureista, prosessorista, ohjelmistosta tai toisin sanoen sovelluksesta, joka yhdistää älykkään laitteen älypuhelimeen. Älykäs laite tarvitsee myös keinon yhdistää älypuhelimeen. Yhdistämisen muodostamiseksi käytetään lukuisia langattomia teknologioita kuten BT (Bluetooth), Wi-Fi ja BLE. [6.]

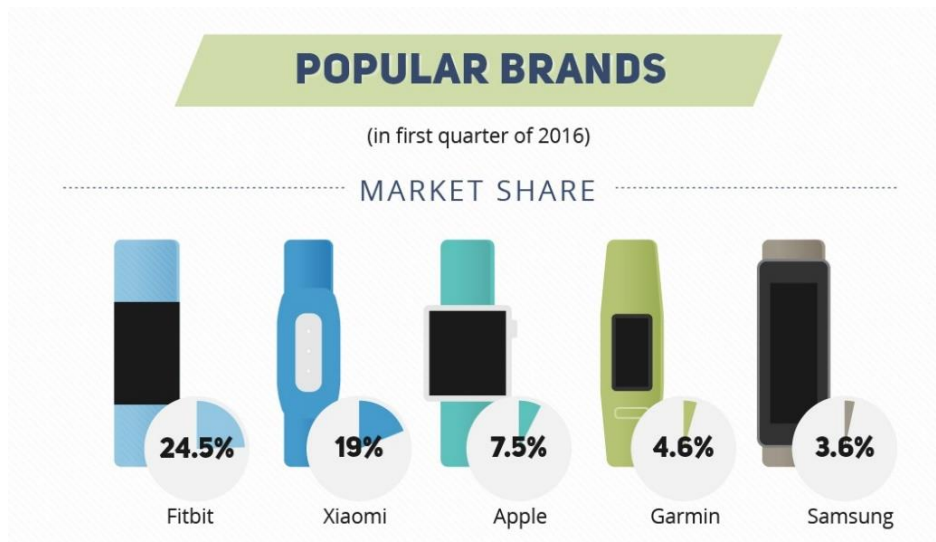
Langattomasta tiedonsiirtoteknologiasta BLE tulee olemaan olennainen osa arkipäiväisessä elämässä. Kuluttajat hyötyvät sovelluksista, jotka käyttävät BLE:tä. Kuluttajat voivat esimerkiksi seurata sydämen sykettä älypuhelimesta. Käyttämällä BLE:tä puhelinsovellusten tekemiseen puhelinsovellusten laatijat hyötyvät BLE:n alhaisesta virrankulutuksesta ja sen monipuolisesta toiminnasta eri käyttöjärjestelmissä. BLE rohkaisee myös yrityksiä laatimaan BLE-käyttöisiä laitteita eri aloille. [7.]

Alla on esitelty lyhyen matkan langattomaa tiedonsiirtoteknologiaa käyttäviä anturisovelluksia, jotka käyttävät puhelinsovelluksia.

2.1 Aktiivisuuden seuranta

Aktiivisuuden seurannassa käytetään aktiivisuusranneketta eli kuntoranneketta, joka on ranteessa pidettävä puettava tietokone. Rannekkeiden tarkoitus on mitata askelmäärää ja unenlaatua. Käyttämällä askelmittaria rannekkeet pystyvät laskemaan kuljetun matkan pituuden ja poltettujen kaloreiden määrän. Rannekkeen tietoja voidaan seurata joko rannekkeen omasta näytöstä tai mobiililaitteen sovelluksen kautta. Tarkempien tietojen seuraaminen edellyttää aktiivisuusrannekkeen kytkemistä tietokoneeseen. [8.] Aktiivisuuden seuraamisessa ei käytetä pelkästään aktiivisuusrannekkeita. Älypuhelimet, älykellot ja sykemittaritkin soveltuvat hyvin aktiivisuuden seuraamiseen. [9.]

Kuvassa 2 on esitetty vuoden 2016 ensimmäisen neljänneksen suosituimmat markkinoilla olevat tuotemerkit, jotka tarjoavat laitteita aktiivisuuden seuraamiseen.



Kuva 2. Suosituimmat tuotemerkit aktiivisuuden mittaamiseen vuoden 2016 ensimmäisestä neljänneksestä. [10.]

Kuvassa 2 Fitbit oli vuoden 2016 ensimmäisen neljänneksen johtava yritys aktiivisuuden seurantaan tarkoitetuilla markkinoilla. Fitbitin jälkeen suosituimmat tuotemerkit olivat Xiaomi, Apple, Garmin ja Samsung.

Aktiivisuuden seuraamiseen tarkoitettujen laitteiden toiminta perustuu kolmiulotteiseen kiihtyvyyssanturiin. Kiihtyvyyssanturin avulla pystytään mittamaan liikkeen alkua, loppumista ja sen voimakkuutta. Esimerkiksi yrityksen Jawbone UP aktiivisuusrannekkeen kiihtyvyyssanturi koostuu kahdesta sähköisesti varatusta levyistä ja vastapainosta levyjen välissä. Vastapaino sijoittuu keskelle, kun anturi on tasapainossa. Liikkeen aikana vastapainon sijainti vaihtuu levyltä toiselle, jonka liikkeen anturi kerää talteen. [11.]

Tämän jälkeen anturin keräämät tiedot käsitellään algoritmien avulla, jotta tieto näkyisi laitteen ruudulla oikeanlaisena. Kerättyjen tietojen tarkkuus varmistetaan vertaamalla tietoja muun laitteen vastaavaan mittaamaan tietoon. Aktiivisuusrannekkeet ja yleensä puettavat laitteet lähettävät dataa mobiililaitteen sovelluksiin eri tavalla. Fitbit-aktiivisuusranneke lähettää sille tarkoitettuun sovellukseen dataa BT-tekniikan avulla. [11.]

2.2 Synkroninen jalkojen paineen mittausjärjestelmä

Jalanpohjan paineen mittaamista liikkeen aikana voidaan esimerkiksi hyödyntää tautien kehityksen seuraamisessa. Varhaiset tähän tarkoitukseen tarkoitetut järjestelmät käyttivät mitattavan henkilön jalan mustepohjia tunnistaakseen jalan kosketuksen jakautumisen. Käytössä on ollut myös paineherkkiä kalvoja seisovan staattisen paineen mittaukseen. Kummatkin järjestelmät tuottivat tulokseksi yhden kuvan, joka oli kokoelma laitteen näkemästä paineesta eli mitään tietoa muuttuvasta paineesta ei ollut. Pedobarograafiset järjestelmät olivat ensimmäisiä, jotka sisälsivät tietoa muuttuvasta paineesta. Datan analysointi laitteilla oli hankalaa, ja niitä ei sen vuoksi ole paljon käytössä. Voimamuuntimet, signaalin ohjausjärjestelmän ohjaamana, ovat luoneet helppokäyttöisiä järjestelmiä, jotka sisältävät tietoa paineen muuttumisesta. [12, s. 143.]

Jalanpohjien paineen mittaamiseen tarkoitetut laitteet voidaan jakaa kahteen ryhmään. Ensimmäinen ryhmä koostuu alustapohjaisista järjestelmistä, jotka on sijoitettu lattiaan. Alustat sisältävät tekijöitä paineen tunnistamiseksi. Alustoja käytetään sekä staattisesti seisovan että muuttuvan paineen perustuvissa tutkimuksissa. Etuna alustalla on sen helppokäyttöisyys, mutta luonnollisen paineen tuottaminen vaatii perehtymistä laitteeseen etukäteen. Toisen ryhmän muodostavat kengät, joihin on sulautettu antureita. Järjestelmä mahdollistaa suorittaa suuren valikoiman tutkimuksia eri kävelytilanteista. Ongelmana anturikenkäjärjestelmälle on mahdollinen anturin liukuminen ulos kengästä ja alueellinen epätarkkuus kuvan muodostamisessa verrattuna alustapohjaisiin järjestelmiin, koska käytössä on vähemmän antureita. [13.]

Eräässä tutkimuksessa on laadittu synkroninen jalkojen painetta mittaava järjestelmä. Tutkimuksen järjestelmä on kykenevä mittaamaan painetta kummastakin jalasta samanaikaisesti, mihin muut langattomat jalanpaineen mittaamiseen tarkoitetut järjestelmät eivät ole kyenneet. Järjestelmän on rakentanut Chung Yuan Christian-yliopisto. Tutkimuksen mukaan moneen anturikenkäjärjestelmään on asetettu paljon antureita. Tehoa kuluu paljon ja siirrettävän datan määrä on suuri, mikäli halutaan saada tarkkoja mittaustietoja mittauksen aikana. Tämän vuoksi tutkimuksessa käytetään viittä paineanturia järjestelmän rakentamiseksi vähentämään tehonkulutusta ja siirrettävän datan määrää. [14, s. 1–2.]

Anturit ovat sijoitettuna kummankin kengän pohjalliseen. Tutkimuksen järjestelmä käyttää anturien tiedonsiirtoon ja langattomuuden saavuttamiseksi vähävirtaista Texas Instrumentsin BLE-käyttöistä CC2650 mikroprosessoria kummassakin kengän pohjallisessa. Tutkimuksen järjestelmälle laadittiin myös mobiilisovellus, joka muodostaa yhteyden kumpaankin CC2650:een yhtä aikaa. Mobiilisovellus vastaanottaa, varastoi tiedot ja esittää

vielä kerätyt tiedot mobiililaitteen näytöllä. Järjestelmän testauksessa saatiin aikaan tyyppilliset jalanpohjista aiheutuvat signaalit ja yhtään dataa ei menetetty tiedonsiirrossa. Tutkimuksen laatimalla mittausjärjestelmälle aiotaan laatia lisätutkimuksia ja havainnollistaa järjestelmän käytettävyyttä esimerkiksi seuraamalla potilaan tilan paranemista kuntoutuksen aikana. Tutkimuksessa laadittua mobiilisovellusta käytettäisiin vertaamaan potilaiden jalanpohjasta aiheutuvaa painetta tavallisiin arvoihin normaalin liikkeen aikana. [14, s. 1–2.]

2.3 Terveydenhuolto & GlucoWise

Terveydenhuollon ala on uusien teknologioiden varhaisessa vaiheessa oleva käyttäjä, aina ensimmäisistä leikkaustyökaluista nykypäivän laitteisiin, kuten puettaviin seurantalaitteisiin. Uuden teknologian nopea käyttöönotto jatkuu yhä. Laitteiden pienemisen myötä tehokasta teknologiaa voidaan sijoittaa kaikkialle. Uusia käyttökohteita luodaan entisille langallisille terveydenhuollon laitteille tekemällä niistä langattomia, eli toisin sanoen IoT tuodaan terveydenhuoltoon. Monet langattomat teknologiat yrittävät saada etuaseman IoT-markkinoilla omilla pidemmällä kantamillaan ja pienemmällä virrankulutuksella. Käytettävältä teknologialta terveydenhuollon IoT-ratkaisuissa vaaditaan vähäistä tehonkulutusta, edullista hintaa ja teknologian saatavuutta. BLE-teknologia on ideaalinen tähän tarkoitukseen esimerkiksi sen vakiintuneen ekosysteemin takia mobiililaitteissa. [15.]

Bluetooth SIG-järjestön tutkimusten mukaan muutaman vuoden sisällä BLE-teknologiaa käytettäisiin langattomissa terveydenhuollon laitteissa niiden tehonkulutuksen alentamiseksi. BLE-teknologiaa soveltavat laitteet tarjoaisivat mahdollisuuden tautien varhaiseen diagnosointiin ja hoitomahdollisuuksia erilaisille vakaville tilanteille. Tämä parantaa viime kädessä lääkehuollon ja lääketieteen palvelujen standardeja. [16.]

BLE-teknologialla on seuraavia sovelluskohteita terveydenhuollossa: verenpaine- ja verensokerimittarit, navigointijärjestelmät ja sairaalatoiminnan kehitys. Verenpaine- ja verensokerimittareiden lähettämiä mitattuja arvoja tarkkaillaan mobiililaitteen kautta. Näillä tiedoilla lääkärit voivat analysoida potilaita jatkokäsittelyä varten. BLE-pohjaiset navigointijärjestelmät helpottavat sairaalassa kulkemista luomalla koko sairaalasta kartan, jota pystyy seuraamaan mobiililaitteen kautta. BLE parantaa sairaalan toimintoja ilmoittamalla sairaalaan henkilökunnalle potilaan tiedoista tai hälytyksistä, kun henkilökunnan väki saapuu työpisteeseensä. [16.]

Gluc Wise on MediWisen kehitteillä oleva glukoosimittari diabetesta sairastaville ihmisille. Gluc Wisen tarkoitus on tehdä vereen liittyvistä mittauksista kivuttomia. Mittaukset suoritetaan laitteen anturiteknologian kautta lävistämättä ihoa. Gluc Wise suorittaa mittauksia niin usein kuin käyttäjä haluaa. Glukoosin mittaamiseksi laite sijoitetaan kevyesti puristamaan ihoa etusormen ja peukalon tai korvolehden väliltä. Tulokset nähdään laitteen näytöltä reaaliajassa. Laitteen keräämät tiedot voidaan langattomasti siirtää BLE:n yli laitteen omaan mobiilisovellukseen tai tietokoneeseen USB-kaapelin kautta. Mobiilisovelluksen tai tietokoneen kautta veren glukoosiarvot voidaan siirtää pilveen talteen. [17.]

Gluc Wise laite toimii lähettämällä korkeataajuisia radiosignaaleja etusormen ja peukalon tai korvolehden alueen ihon läpi. Iho näillä alueilla sisältää riittävän määrä verta ja kudosta on tarpeeksi ohutta radioaaltojen läpipääsemiseksi. Radioaallot vastaanotetaan Gluc Wise-laitteen toisella puolella sijaitsevaan anturiin. Anturi kerää ja analysoi tietoja veren ominaisuuksista lihaksen sisällä. Anturin nano-tason yhdistelmäaerialikalvot tekevät ihosta väliaikaisesti läpinäkyvän radioaalloille helpottamaan ihon läpäisemistä. Tämä varmistaa johdonmukaiset tulokset kaikilta mitatuilta ihmisiltä. [17.]

2.4 Beacon

Beacon on pieni laite, joka toistuvasti lähettää signaaleja lähellä oleville mobiililaitteille. Mobiililaitteiden sovellukset kuuntelevat mainostettavia signaaleja. Kuullessaan sille merkityksellisen signaalin mobiilisovellukset aiheuttavat jonkin toiminnon mobiililaitteessa. Beaconit, voivat vain lähettää dataa. Beaconit toimivat BLE-teknologialla. Beaconit ovat tyypillisesti paristokäyttöisiä ja paristojen käyttöikä vaihtelee kuukausista pariin vuoteen asti. Beaconit ovat kiinni seinissä/katoissa tai ovat piilotettuina muiden esineiden sisällä. Kantama beaconeilla on 70 metriä ilman esteitä. Eräitä beaconeiden valmistajia ovat esimerkiksi Estimote ja Kontakt. [18.]

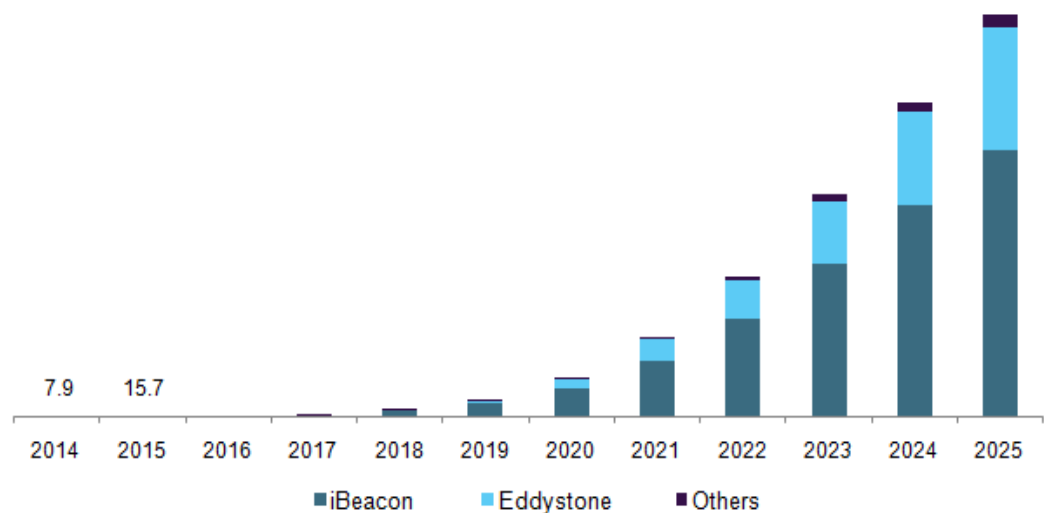
Beaconeilla voidaan paikallistaa ihmisiä ja seurata heidän toimintaa sisätiloissa. Jälleenmyynti on suosituin kohde beaconeiden käytölle. Beaconit luovat uusia kaupankäynnin välisiä vuorovaikutuksia asiakkaiden ja jälleenmyyjien välillä. Jälleenmyyjät voivat kerätä uudella tavalla tietoja asiakkaistaan kaupankäynnin tapahtumien perusteella. Asiakkaat saavat taas tämän perusteella henkilökohtaisia tarjouksia perustuen heidän omiin mieltymyksiinsä ja käytöksiinsä kaupassa. [19.]

Beaconit voidaan luokitella eri tyyppeihin pohjautuen beaconeissa käytettävään virtalähteeseen, protokollaan ja paikannustekniikkaan. Paristovirtalähteen lisäksi beaconit voivat

saada myös virtaa verkkovirrasta tai USB-virtalähteestä. Suurimmalta osalta käytetyn BLE-tekniikan lisäksi beaconit käyttävät muitakin paikannukseen liittyviä teknologioita, kuten Wi-Fi Alliansin kehittämä Wi-Fi Aware. Beaconeissa lähetettävän tiedon sisältö määräytyy käytettävän protokollan mukaan. [20.]

Vuonna 2013 Apple julkaisi iBeacon-protokollan ja kaksi vuotta myöhemmin Google julkaisi Eddystone-protokollan. iBeacon- ja Eddystone-protokollat ovat yhteensopivia Android- ja iOS-käyttöjärjestelmille, jotka tukevat BLE-tekniikkaa. Eddystone-protokolla on kuitenkin alustavapaa, eli se on yhteensopiva kaikkien alustojen kanssa, joilla on jonkin tason tuki beaconeissa käytettävälle BLE-tekniikalle. iBeacon-protokollaa käyttävien beaconeiden lähettämä tieto mobiililaitteelle on beaconin oma uniikki tunnistus. Nämä beaconit tarvitsevat mobiililaitteessa oman sovelluksen, joka herää tekemään jotain vastaanotettuaan tunnisteen. Eddystone-protokollaa käyttävät beaconit lähettävät omien uniikkien tunnistusten lisäksi internetosoitteita, jotka eivät tarvitse omaa mobiilisovellusta aukaistakseen internetosoitteen sivun. Myös beaconin omasta tilasta lähetetään tietoa, joka voi olla tietoa esimerkiksi patterien virran määrästä. [20.] [21.]

Alla olevassa kuvassa on esitetty beacon-tyyppien markkinat käytettävän protokollan mukaan vuosille 2014–2025.



Kuva 3. Beacon-markkinat Yhdysvalloissa vuosille 2014–2025. [22.]

Yllä olevassa kuvassa erityyppisten beaconeiden yhteinen markkinatulo vuonna 2014 oli 7,9 miljardia Yhdysvaltain dollaria. Vuonna 2015 tulot olivat yhteensä 15,7 miljardia dollaria. Kuvan 3 mukaan iBeacon-protokollaa käyttävät beaconit ovat hallitsevin beacon-tyyppi lähitulevaisuudessa.

Eddystone-protokollaa käyttäville beaconeille odotetaan nopeaa kasvua perustuen sen avoimuuteen ja kykyyn tarjota suuri määrä ratkaisuja läheisyysvuorovaikutuksille ilman, että tarvitaan puhelinsovelluksia [23].

2.5 IIoT & SmartBond

IoT tarkoitti laitteiden muodostamaa verkostoa, jotka keräävät ja jakavat dataa. Data lähetetään pilvipalveluun, jossa se yhdistetään muihin datoihin. Yhdistetty data jaetaan loppukäyttäjien kanssa hyödyllisellä tavalla. IoT:n soveltamista tuotantoteollisuudessa kutsutaan IIoT:ksi (Industrial Internet of Things). IIoT parantaa tuotantoa mahdollistamalla suuremman datamäärän hankkimisen, jonka saanti on nopeampaa ja tehokkaampaa kuin ennen. Tämä data auttaa yrityksiä havaitsemaan puutteellisuuksia ja ongelmia nopeammin säästäten ajassa ja kustannuksissa. [24.] [25.]

IIoT luottaa paljon antureiden keräämään dataan erilaisista järjestelmistä. Dataa käytetään eri ominaisuuksien seuraamiseen. [26.] Esimerkiksi GE:n yhdysvaltaisessa tehtaassa, joka valmistaa natriumnikkeliakkuja, on yli 10 000 anturia sijaiten 180 000 neliömetrin alueella. Anturit on yhdistetty nopeaan sisäiseen paikalliseen tietokoneverkkoon. Anturit seuraavat esimerkiksi jokaisen akun valmistukseen käytettävän energian määrää. Tehdastason työntekijät iPadien kanssa voivat saada haltuunsa antureiden keräämää dataa langattomasti datojen jakelupisteistä, joita on levitetty ympäri tehdasta. [27.]

Moduulit ovat itsenäisistä osista koottavia kokonaisuuksia, jotka antavat nopean ja helpon tavan lisätä antureita suunniteluun. Usein moduulien haasteina on yhdistää ne muuhun laiteverkkoon. Helpoin tapa luoda yhteys on käyttää BLE-teknologiaa. BLE-lähettemien koko on pienentynyt sirun kokoiselle tasolle. Pienen koon ansiosta ne on helppo sijoittaa antureiden rinnalle IIoT-mallissa. Langattomien lähettemien suunnittelija Dialog Semiconductor on valmistanut IoT:hen tarkoitettun SmartBond-anturimoduulin, jonka tarkoitus on helpottaa liikkeentunnistukseen ja ympäristön aistimiseen liittyvien sovellusten tekemistä. [26.]

SmartBond koostuu langattomasta lähettemestä ja kahdestatoista toisistaan riippumattomasta anturista. SmartBond sisältää gyroskoopin, magnetometrin, kiihtyvyyssanturin ja joukon digitaalisia ympäristöantureita, jotka Bosch Sensortec on valmistanut. Yhdistämällä SmartBond-järjestelmän pieni koko sisällytettävän laitteiston ja ohjelmiston kanssa järjestelmään voidaan sisällyttää koodi, joka kokoaa anturien datat yhteen. Datojen ka-

saaminen yhteen tuottaa tarkempia tuloksia. SmartBond-laitteeseen on lisätty sen valmistajan laatima ohjelmakirjasto datojen tiedonhankintaa, hienosäätämistä ja yhdistämistä varten. Yhdistetty data BLE:n välityksellä lähetetään Android- tai iOS-mobiililaitteessa pyörivään mobiilisovellukseen visualisointia varten. [26.]

2.6 MetaWear

MetaWear on yhdysvaltalaisen MblentLab-tuottajan sarja valmiita kehitys- ja tuotantoalustoja kohdistuen ylläpidettäviin laite- ja anturisovelluksiin. MetaWear käyttää Nordic Semiconductor-valmistajan nRF51822-järjestelmäpiiriä, joka antaa MetaWear-alustoille liitäntämahdollisuuden BLE-päätelaitteisiin, kuten älypuhelimiin. MetaWear-laitteisto toimitetaan moduulissa, ja moduulin koko voi halkaisijaltaan olla 9 - 11 tai 17 - 26 millimetriä. MetaWear-moduulin anturit voivat mitata muun muassa kiihtyvyyttä, lämpötilaa, painetta ja korkeutta. Moduulien laaja valikoima antureita antaa MetaWearille paljon sovelluskohteita esimerkiksi terveydenhuollossa ja urheilussa. [28.]

Käyttämällä MetaWearia esimerkiksi nyrkkeilyhanskoissa iskujen nopeutta ja voimakkuutta voidaan mitata. Nyrkkeilijä voi harjoitella näin kätevämmiin seuraamalla arvoja BLE-käyttöisen laitteen mobiilisovelluksen kautta. MetaWearin kohderyhmät ovat ihmiset, jotka haluavat kehittää tai parantaa tuotteitaan tekemällä niistä älykkäitä. Omien antureiden lisääminen MetaWeariin on mahdollista. [28.]

MetaWear-moduulia hallitaan Android- tai iOS-puhelimen avulla. Puhelimissa olevat mobiilisovellukset ohjeistavat MetaWearia tekemään toimintoja. MetaWearin anturien arvot mitataan ja näytetään puhelimesta reaaliajassa. Yhteyden luominen puhelimeen tehdään BLE-tekniikan avulla. Puhelimesta tieto voidaan siirtää eri muodossa esimerkiksi Exceliin datan analysointia varten. [29.] MetaWearista on myös saatavilla versioita, jotka sisältävät antureita terveyden mittaamiseen. MetaHealth-versio esimerkiksi sisältää antureita, joilla voidaan mitata asentoa, ihon lämpötilaa ja sydämen sykettä. [30.]

3 Kehitystyökaluja

Ohjelmoinnissa käytettävät kehitystyökalut ovat tietokoneohjelmia, joita käytetään ohjelmien ja sovellusten laatimiseen, debuggaukseen ja ylläpitämiseen. Yksinkertaisimmat työkalut ovat lähdekoodin editori ja kääntäjä. Muiden työkalujen käyttö riippuu esimerkiksi käytettävästä koodikielestä, kehitystavoista ja keinoista. Työkalut ovat joko erillisiä ohjelmia tai ovat osa yhtä isompaa ohjelmaa, jota kutsutaan integroiduksi kehitysympäristöksi (IDE). [31.]

IDE:n tarkoitus on kasvattaa kehityksen tuottavuutta. Koodin jäsentäminen ja syntaksin tarkistaminen samanaikaisesti nopeuttavat kehityksen kulkua esimerkiksi. Tietyille ohjelmointikielille on omistettu oma IDE, mutta montaa ohjelmointikieltä tukevia IDE:itä on myös olemassa. [32.]

3.1 Eclipse

Eclipse on monialustainen tietokoneohjelmoinnissa käytettävä IDE, joka on laadittu Java-ohjelmointikielillä. Eclipseä käytetään suurimmaksi osaksi Java-sovellusten tekemiseen. Lisäosien avulla Eclipsessä on mahdollista kehittää sovelluksia muilla ohjelmointikielillä, kuten C/C++:lla ja JavaScript-kielellä. Eclipse Foundation on IDE:n kehittäjä. Ensimmäinen versio Eclipsestä 3.0 julkaistiin vuoden 2004 kesällä. Uusin versio on tällä hetkellä 4.7 nimeltään Oxygen, joka julkaistiin vuoden 2017 kesällä. Versio 4.8 Photon on suunniteltu julkaistavaksi vuoden 2018 kesällä. [33.]

Eclipsen kehitystä johti IBM:n tytäryhtiö OTI. Eclipse on kehittynyt aiemmista VisualAge kehitysympäristöistä, jotka pohjautuvat Smalltalk-kieleen. Monet Eclipsen konseptit löytyvät VisualAge MicroEdition IDE:stä, mitä oli vaikea laajentaa uusilla osilla. Korvaajaksi kehitettiin Eclipse, johon pystyi helposti lisäämään rakennettavia tuotteita Eclipsen laajenusmekanismien avulla. [34.]

Eclipse alustan arkkitehtuuri perustuu lisäosien ympärille. Lisäosat ovat joukko koodia tai dataa, jotka edistävät järjestelmän toiminnallisuutta. Toiminnallisuutta tuetaan esimerkiksi koodikirjastojen ja alustan laajennusten avulla. Lisäosat pystyvät määrittämään pisteitä, joihin muut lisäosat voivat lisätä toiminallisuutta. Jokainen Eclipse-alustan sisäinen järjestelmä on jäsenneilty joukoksi lisäosia, jotka aikaansaavat jonkin keskeisen toiminnon. Lisäosat toimittavat luokkakirjastoja, joilla voidaan toteuttaa järjestelmän laajennuksia. [35.]

Eclipsen ohjelmistokehityspakkaus koostuu itse alustasta, Java-kehitystyökaluista ja lisäosien kehitysympäristöstä. Java-kehitystyökalut aikaansaavat Javalla varustetun kehitysympäristön. Lisäosille tarkoitettu kehitysympäristö lisää työkaluja suoraviivaistamaan lisäosien ja laajennusten kehitystä. [35.]

Android-kehitystyökalut on Googlen kehittämä lisäosa Eclipselle, joka lisää tuen Android-kehitykselle. Android-kehitystyökalu tarjoaa yhtenäisen ympäristön, jossa Android-sovelluksia voidaan kehittää. Googlen Android-kehitystyökalut lisäosa oli virallinen kehitysympäristö Androidille, mutta se poistui käytöstä vuonna 2015. Android Studio on uusi korvaava kehitysympäristö Androidille, joka on Googlen tukema. [33.]

3.2 Android Studio

Android Studio on IDE, joka on tarkoitettu Android-käyttöjärjestelmässä olevien sovellusten tekemiseen. Pohjana IDE käyttää IntelliJ IDEA-koodieditoria. Android Studio toimii Eclipsen Android-kehitystyökalujen lisäosan korvaajana Android-sovellusten kehityksessä. Android Studion ominaisuuksiin kuuluu muun muassa joustava gradlepohjainen rakentaminen ja Lint työkalut. Lint työkaluja käytetään suorituskyvyn, käytettävyyden, version yhteensopivuuden ja muiden ongelmien havaitsemiseen. Android Studion kehittäjä on Google. Versio 1.0 Android Studiosta julkaistiin vuoden 2014 joulukuussa. Uusin versio on tällä hetkellä versio 3.0, joka julkaistiin lokakuussa 2017. [36.] [37.]

Kotlin on staattisesti tyyhitetty ohjelmointikieli. Kotlin on suunniteltu toimimaan yhdessä Java-koodin kanssa. Se on riippuvainen Java-koodin olemassa olevasta Java Class Library -kirjastosta. Käyttämällä aggressiivista tyyppipäättelyä muuttujille ja ilmaisuille vähennetään Kotlin-kielessä tarvittavan sanallisuuden määrää suhteessa Javaan. Android Studion versiosta 3.0 alkaen Kotlin on täysin tuettu ohjelmointikieli Android-käyttöjärjestelmässä. Kotlin-ohjelmointikielen kehittäjä on JetBrains. Ensimmäinen versio kielestä 1.0 julkaistiin 15.2.2016. Tästä versiosta alkaen on luvattu pitkäaikaista versioiden taaksepäin yhteensopivuutta. Versio 1.2 julkaistiin 28.11.2017, ja tässä kielen versiossa lisättiin ominaisuus jakaa koodia Java-virtuaalikoneen ja Javascript-alustojen välillä. [38.]

Android-käyttöjärjestelmä on rakennettu ohjelmistopinon mallin kaltaiseksi. Järjestelmän arkkitehtuuri koostuu eri kerroksista, jotka ovat alhaalta ylöspäin lueteltuna: Linux Kernel, ajoaikaympäristö ja siihen kuuluvat koodikirjastot, sovelluskehys ja sekä Android-laitteen omat että kolmannen osapuolen laatimat sovellukset. Linux Kernel on Android-ohjelmis-

topinon alin kerros. Linux Kernel tarjoaa matalan tason ydinjärjestelmäpalveluita, esimerkiksi muistin- ja virranhallintaa. Android Studiolla laaditut Android-sovellukset käännetään tavukoodimuotoon. Kun Android-sovellus ladataan myöhemmin laitteelle, Kernel-kerroksen yläpuolella oleva Android-ajoaikaympäristökerros kääntää tavukoodin ELF-tiedostomuotoon. Android-sovelluksen käynnistyksen yhteydessä ajetaan läpi sovelluksen suoritettava ELF-tiedosto, mikä nopeuttaa sovelluksen suoritusta ja parantaa akun käyttöikää. [39.]

Tavallisten Java-kirjastojen lisäksi Android-kehitysympäristö sisältää myös Android-kirjastoja. Nämä kirjastot on tarkoitettu Android-kehitystä varten. Näitä kehityksessä käytettäviä kirjastoja ovat esimerkiksi sovelluskehyskirjastot, jotka puolestaan taas helpottavat käyttöliittymien rakentamista. Java-pohjaiset koodikirjastot kutsuvat C/C++-pohjaisia koodikirjastoja suorittamaan toimintoja kuten graafista piirtämistä. C/C++-kirjastot sisältävät laajan ja monipuolisen määrän muita toimintoja graafisen piirron lisäksi. Muihin toimintoihin kuuluu muun muassa SQLite-tietokannan hallinta sekä ääni- ja videotoisto. [39.]

Android-ohjelmistopinon toiseksi ylimmän kerroksen muodostaa sovelluskehys, jonka päällä sijaitsee ylin kerros, sovelluskerros. Sovelluskehys on joukko palveluita, jotka muodostavat yhdessä Android-sovellusten suorittamiseen ja niiden hallitsemiseen tarkoitettun ympäristön. Kehyksen käsitteenä on, että Android-sovellukset laaditaan uudelleenkäytettävistä, yhteensopivista ja korvattavista osista. [39.]

Eräitä sovelluskehyn pääpalveluita ovat esimerkiksi aktiviteetti-, resurssien, ilmoitusten, pakettien hallitsijat ja tiedon tarjoajat. Aktiviteettihallitsija hallitsee sovelluksen elinikää. Resurssien hallitsija antaa pääsyn ei-sulautettuihin koodin resursseihin, kuten väriasetuksiin ja käyttöliittymän ulkoasuun. Ilmoitusten hallitsijalla sovellus näyttää hälytyksiä ja ilmoituksia käyttäjälle. Pakettien hallitsijalla sovellukset saavat tietoonsa muut sovellukset, jotka on asennettu laitteelle. Tiedon tarjoajat antavat sovellukselle kyvyn jakaa ja julkaista tietoa muille sovelluksille. [39.]

Android-ohjelmistopinon ylimmän kerroksen muodostaa sovelluskerros, joka koostuu Android-käyttöjärjestelmän omista ja kolmannen osapuolen kehittämistä sovelluksista [39].

4 Langaton tiedonsiirto

Langattomalla tiedonsiirrolla tarkoitetaan vähintään kahden laitteen välistä kommunikointia, jotka eivät ole kaapeleilla yhteydessä toisiinsa. Langatonta tiedonsiirtoa esiintyy nykypäivänä kaikkialla. Laitteet kuten puhelimet ja kaukosäätimet siirtävät tietoa langattomasti. Hankkiutumalla eroon kaapeleista parannetaan liikkuvuutta, asennusaikaa ja säästetään kustannuksissa. Langattomilla teknologioilla on tiedonsiirrossa sovelluksia, joissa dataa on siirrettävänä kilobiteistä useisiin gigabitteihin saakka. [40, s. 1–3.]

Teknologiat luokitellaan useisiin ryhmiin käytettävän etäisyyden perusteella. Paikallisalueen verkossa esimerkiksi rakennuksissa etäisyys on satoja metrejä. Henkilökohtaisissa alueen verkoissa etäisyys on vain muutamia metrejä. Tällä etäisyydellä käytettäviä teknologioita ovat muun muassa BT, BLE ja ANT. [40, s. 1–3.]

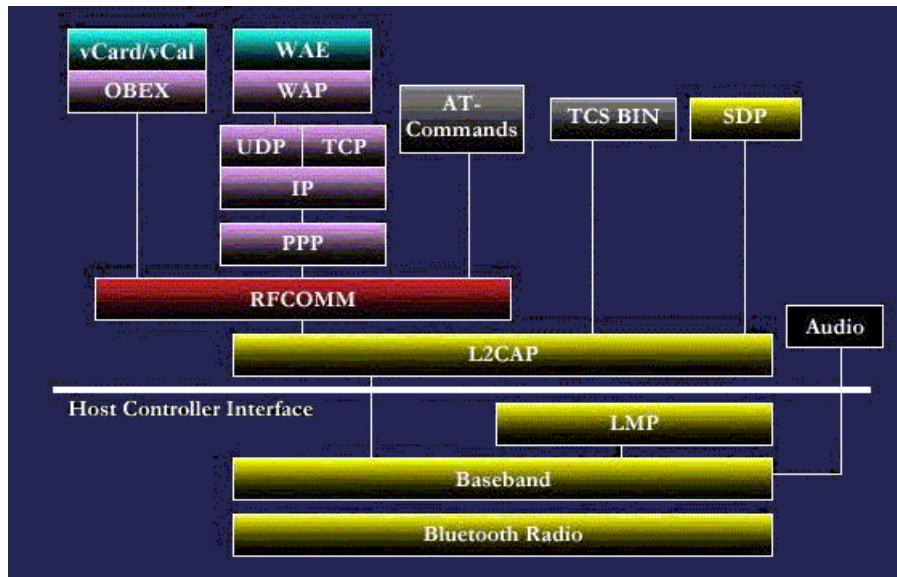
4.1 BT

BT on yleinen lyhyen matkan vähävirtainen ja pienimuotoinen langaton tiedonsiirtoprotokolla. Teknologian avulla laitteet pystyvät keskustelemaan keskenään radioyhteyden yli. BT-teknologiaa käytetään myös yhdistämään laitteita toisiinsa henkilökohtaisessa oman alueen verkossa. [40, s. 16–17.]

Vuonna 1994 Ericsson-yhtiö halusi korvata kommunikoinnissa käytetyt RS-232-kaapelit langattomalla vaihtoehdolla. Muut yritykset kuten Intel ja Nokia olivat samaa mieltä asiasta. Vuonna 1998 yritykset perustivat järjestön Bluetooth SIG kehittämään BT-teknologiaa. Vuotta myöhemmin versio 1.0 BT:stä julkaistiin. Vuonna 2005 julkaistiin versio 2.0 parannetulla datanopeudella. Vuonna 2010 versiossa 4.0 mukaan lisättiin BLE-teknologia. [41.] Tällä hetkellä uusimmassa versiossa 5.0 BT:n toimintoja kasvatetaan IoT:tä varten [42].

BT-teknologialla voidaan muodostaa kahdenlaisia verkkoja: pico- ja scatterverkkoja. Picoverkko on pieni BT-verkko, joka koostuu yhdestä master- ja seitsemästä slave-laitteesta. Kommunikaatio voi tapahtua ainoastaan master- ja slave-laitteiden välillä. Master-laite voi keskustella useamman slave-laitteen kanssa. Scatterverkko muodostuu useammasta picoverkosta. Yhden picoverkon slave-laite voi olla toisen picoverkon master-laite. Tämä laite voi myös toimia solmuna kahden picoverkon välillä. Laite vastaanottaa tietoa sen picoverkon master-laitteelta ja lähettää tiedon toisen verkon slave-laitteelle. [43.]

Bluetooth SIG-järjestön kehittämä protokollapino on esitetty alla.



Kuva 4. BT:n protokollapinon malli. [44.]

Kuvan 4 BT-protokolla koostuu useasta protokollasta ja nämä protokollat muodostavat protokollapinon neljä kerrosta: ydinprotokollat, kaapelin korvaava protokolla, puhelinhallintaprotokolla ja muokkautetut protokollat [44].

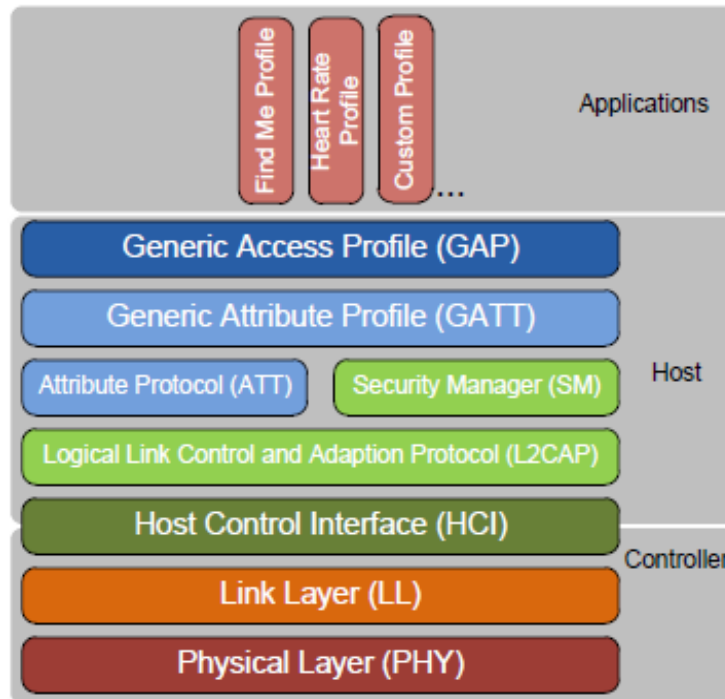
Protokollapinon ytimen muodostavat baseband-, LMP-, L2CAP- ja SDP-protokollat. Näistä protokollista esimerkiksi baseband mahdollistaa fyysisen radioyhteyden BT:n muodostamisessa ja LMP hoitaa BT-laitteiden välisen linkittämisen. [45.] RFCOMM on fyysisen kaapelin korvaava protokolla, ja se vastaa tiedonsiirrosta BT:n perustajuuksalueella. Puhelinhallintaprotokolla TCS BIN ohjaa puhelin-signaaleja datapuheluiden muodostamiseksi BT-laitteiden välille. [46.] Mukautetut protokollat ovat PPP-, TCP- (Transmission Control Protocol), UDP- (User Datagram Protocol), IP-, OBEX- ja WAP-protokollat. BT-laite voi esimerkiksi keskustella internetiin yhdistyneiden laitteiden kanssa TCP/IP-protokollalla ja jakaa objekteja OBEX-protokollalla. [47.]

4.2 BLE

BLE on lyhyen kantaman langaton tiedonsiirtoprotokolla, jonka tarkoitus on huomattavasti vähentää laitteiden virrankulutusta. BLE-teknologia mahdollistaa pienten paristokäyttöisten laitteiden käytön. Nämä laitteet lähettävät ja vastaanottavat pieniä määriä dataa lyhyin väliajoin. BLE-teknologia pitää virrankulutuksen alhaisena käyttämällä vähemmän tehoa

tiedonlähetykseen kuin BT. Nopeampi vaihto valmiustilasta aktiivitilaan BLE-laitteilla säästää energiaa ja mahdollistaa datan lähettämisen pieninä purskeina. BLE-teknologia muodostaa samanlaisia verkkoja kuin BT eli pico- ja scatterverkkoja. [48, s. 218–219.]

Kuvassa 5 on esitetty BLE-protokollapino.



Kuva 5. BLE-protokollapinon malli. [49.]

Kuvan 5 protokolla koostuu kolmesta kerroksesta: ohjain, isäntä ja sovellus. Ohjain on fyysinen laite, joka lähettää ja vastaanottaa signaaleja. Isäntäkerros määrittää, miten laitteet kommunikoivat keskenään. Sovelluskerros sisältää logiikan ja käyttöliittymän sovelluksen käyttäjälle. Tämän kerroksen rakenne riippuu kehitettävästä sovelluksesta. [48, s. 218–219]

BLE-protokollapinon kolme kerrosta sisältävät eri määrän eri kerroksia. Jokaisella kerroksella on vastuu BLE-laitteiden oikein toimivuudesta. Ohjainkerros sisältää fyysisen kerroksen ja linkkikerroksen. Fyysisen kerroksen avulla BLE voi kommunikoida usean kanavan kautta 2,4 GHz radiotaajuusalueella, mitä käytetään yhteyksien muodostamiseen ja tiedon välitykseen. Linkkikerros määrittää laitteiden roolit perustuen loogisiin ryhmiin. Laite voi mainostaa itseään muille laitteille, etsiä mainostavia laitteita, luoda ja hallita yhteyksiä muihin laitteisiin tai hyväksyä yhteyksipyyntöjä muilta laitteilta. Kerroksella on vastuu myös yhteyksien muodostamisesta. HCI-kerros yhdistää fyysisen ja linkkikerroksen isäntäkerrokseen. [50.]

Isäntäkerros koostuu L2CAP, tietoturvaprotokollasta (SM), attribuuttiprotokollasta (ATT), yleisestä attribuutti (GATT) - ja pääsyprofiilista (GAP). Attribuuttiprotokolla (ATT) perustuu laitteen dataan, joka on organisoituna attribuutteihin. Attribuutteja lähetetään palvelinlaitteelta asiakaslaitteelle tämän pyynnöstä. Yleinen attribuuttiprofiili (GATT) sijoittuu attribuuttiprotokollan päälle. Profiili vastaa, kuinka data on organisoitunut ja miten dataa jaetaan sovelluksien välillä. Data on järjestäytynyt attribuuttiprofiilissa eri palveluihin. Jokainen palvelu sisältää yhden tai useamman ominaisuuden, jotka taas pitävät sisällään käyttäjätietoja ja kuvaustietoa datasta. Yleinen pääsyprofiili (GAP) keskittyy esimerkiksi laitteiden rooleihin, niiden väliseen vuorovaikutukseen ja toimintamenetelmiin. [50.]

L2CAP toimii tietopakettien kasaajana ja purkajana. Tietoturvaprotokolla (SM) takaa sen, että tieto siirtyy turvallisesti BLE-yhteyden yli. [49.]

4.3 ANT/ANT+

ANT on vähävirtainen lyhyen matkan langaton tiedonsiirtoprotokolla. Teknologian käyttökohde on urheilu- ja kuntoalueet. Näillä alueilla luodaan henkilökohtaisia alueverkkoja suorituskyvyn ja terveyden mittaamiseksi. [51.] ANT-laitteet voivat toimia vuosia yhdellä kolikkoparistolla. ANT-protokolla pystyy viettämään paljon aikaa hyvin vähävirtaisessa valmiustilassa. ANT herää vain vähäksi aikaa keskustelemaan ja palaa sitten takaisin valmiustilaan. ANT-protokollassa laite voi viestiä yhden tai useamman laitteen kanssa ilman kuittausta viestin saapumisesta. ANT-protokollan viestintä mahdollistaa kuitenkin kuittauksien käytön viestinnässä. ANT-Wireless on protokollan suunnittelija. [52.]

ANT+ on langattomaan tiedonsiirtoon tarkoitettu protokolla, jonka perustana on ANT-protokolla. ANT+-protokolla on ANT+-allianssin suunnittelema ja ylläpitämä. Protokolla mahdollistaa ANT+-laitteiden yhteentoimivuuden verkossa. Yhteentoimivuus helpottaa anturitietojen keräämistä ja selittämistä. Protokolla on suunniteltu anturidatan keräämiseen ja siirtämiseen, mutta protokolla on myös integroitu kauko-ohjaussysteemeihin. [53.] ANT+-laitteille on määritelty oma laiteprofiili. Laitteprofiilista näkee, minkä muiden ANT+-laitteiden kanssa se on yhteensopiva. [54.] ANT-protokolla julkaistiin vuonna 2003 ja ANT+ vuotta myöhemmin. Taustana oli kehittää protokolla, joka kykenee siirtämään anturidataa kelloakkuja käyttäville vastaanottimille. [55.]

5 Langattomassa tiedonsiirrossa käytettäviä protokollia

TCP, UDP, IP ja muut vastaavat protokollat muodostavat internetin datan kuljetuskerroksen TCP/IP-kokonaisuuden vuodesta 1982 lähtien. TCP/IP on ensisijainen valinta digitaalisessa kommunikoinnissa. [56, s. 3.] TCP/IP-kokonaisuudessa TCP- ja UDP-protokollat ovat osa BT-protokollapinon mukautuvia protokollia kuvassa 4.

5.1 UDP

UDP on yksinkertainen kerrostason protokolla. UDP:ta käytetään sovelluksissa, jotka eivät tarvitse luotettavaa tiedonsiirtoa. UDP on osa internetin protokollapinoa ja kuuluu kuljetuskerrokseen. Kuljetuskerros sijoittuu protokollapinossa sovelluskerroksen alapuolelle, joka on internetprotokollapinon ylin kerros. UDP lähettää sovelluskerrokselta tullutta dataa eteenpäin ja suorittaa operaatiota kuten varmenteiden luomisen, jolla vastaanottaja tarkistaa viestin pätevyyden ja viestin lopullisen kohteen osoitteen. [56, s. 4–7.]

UDP:ssä viestit lähetetään ilman yhteyden muodostamista toiseen laitteeseen. Viesteissä ei ole tietoa sen sijainnista ja koosta. Viestien vastaanottamista ei myöskään kuitata. UDP sopii hyvin lyhyiden viestien lähettämiseen. Nämä viestit voivat olla pyyntöjä tai vastauksia yhdeltä lähettäjältä yhdelle vastaanottajalle. Viestien lähettäminen useaan paikkaan on mahdollista. [56, s. 4–7.] Eräs protokollan käyttösovellus on tunneliprotokolla. UDP:n datagramit paketoidaan tunnelin päässä, paketit puretaan toisessa pisteessä ja UDP:n hyötykuorma viedään eteenpäin. [57].

5.2 TCP

TCP:ssä muodostetaan kaksisuuntainen yhteys toisen laitteen kanssa. Yhteys suljetaan viestien vaihdon jälkeen. TCP on kuljetustason protokolla ja vastaanottaa dataa sovelluskerrokselta. TCP toteuttaa luotettavaa tiedonsiirtoa ja suorittaa lukuisia toimintoja sen saavuttamiseksi, muun muassa viestin kuitausta ja virheiden hallintaa. TCP ainoastaan voi lähettää viestejä yhteen osoitteeseen. TCP:n paketit on numeroitu ja paketeilla on määräaika, jolloin niiden on saavuttava määränpäähänsä. Jokainen saapunut paketti kuitataan. Mikäli paketin kuitaus ei saavu lähettäjälle, kopio puuttuneesta paketista lähetetään uudestaan. Viallisia paketteja ei kuitata. [56, s. 4–9.] [58.]

6 Työn tausta, tavoite ja käytettävät välineet

Tämän työn taustana on Kajaanin ammattikorkeakoulun tieto- ja viestintätekniikan koulutusohjelman työharjoittelujakso. Työharjoittelu suoritettiin KajaPro-yrityksessä. Työharjoittelun aikana tutkittiin langattoman tiedonsiirtoteknologian BLE-toimintaa ja käyttöönottoa. BLE-tekniikkaa hyödyntävää CC2650 SensorTag IoT-laitetta käytettiin tähän tarkoitukseen apuna. SensorTag sisältää antureita ja langattoman mikropiirin CC2650. Laitteen ympärille laadittiin sitä hyödyntäviä mobiilisovelluksia mobiilipäätelaitteelle, joka käyttää Android-käyttöjärjestelmää. Mobiilisovellukset kehitettiin sekä Internetistä löytyvien mobiilisovellusten tekemiseen tarkoitettujen IDE:itten että KajaPro:n oman ohjelmistotyökalun ReGui (Remote Graphical User Interface) avulla. Sovellukset yksinkertaisuudessaan etsivät, yhdistävät SensorTagiin ja tulostavat ulos antureiden dataa.

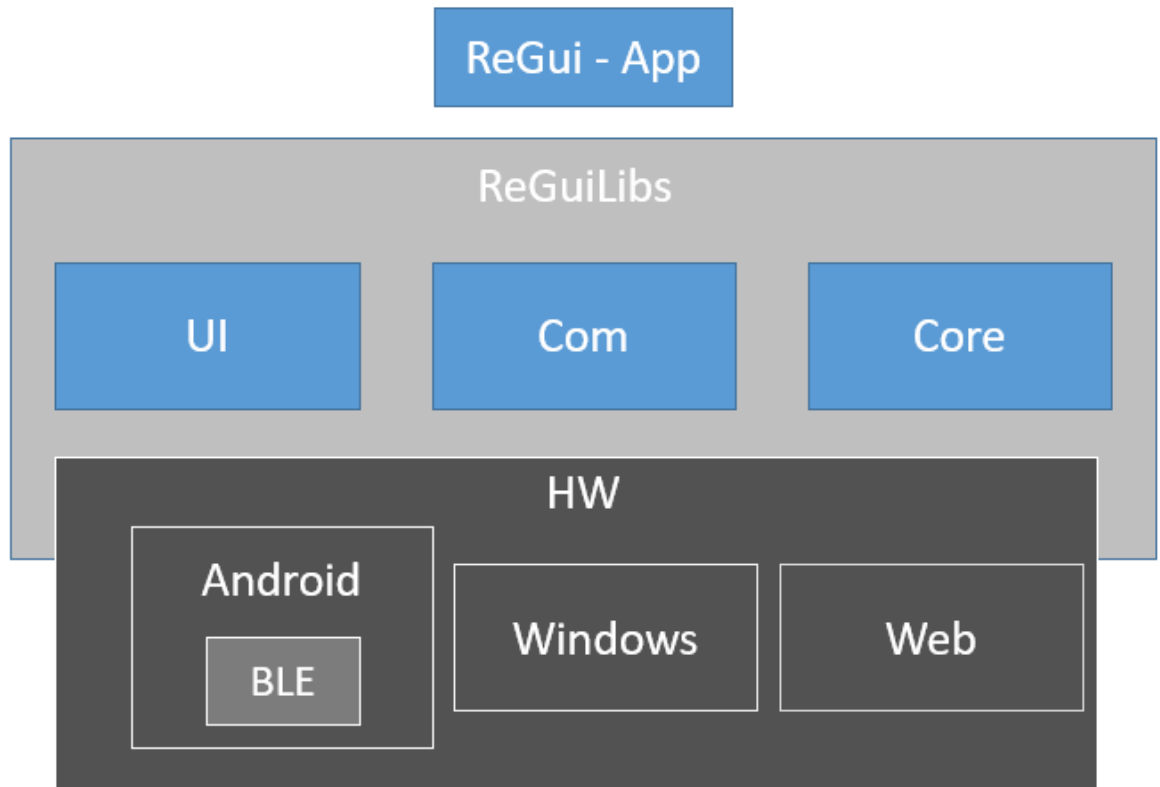
Tämän työn tarkoitus on toteuttaa BLE-järjestelmä, jolla kahden SensorTagin avulla voidaan seurata työkalun lämpötilaa, käyttäjän etäisyyttä työkalusta ja tunnistaa työkalun käyttäjä. Etäisyyttä mitataan seuraamalla SensorTagin signaalin voimakkuutta, ja käyttäjän tunnistaminen tapahtuu SensorTagin omalla laiteosoitteella. Tiedot välitetään BLE-yhteyden yli mobiilipäätelaitteella olevaan mobiilisovellukseen. Mobiilisovelluksena käytetään työharjoittelun aikana ReGui:lla kehitettyä mobiilisovellusta, joka on laadittu SensorTagin ympärille.

6.1 ReGui

KajaPro-yrityksen laatima ohjelmistotuote ReGui tarjoaa mobiilipäätelaitteen avulla käyttöönottoa ulkopuoliselle laitteelle. Käyttöliittymä on yleiskäyttöinen mobiilipäätelaitteessa toimiva mobiilisovellus. Ulkopuolinen laite ja mobiilisovellus voivat kommunikoida keskenään joko langattomasti tai langallisesti. Langaton kommunikointi tapahtuu BT- tai socket-yhteyden kautta. Langallinen kommunikointi tehdään esimerkiksi USB-portin kautta. ReGui kokoaa käyttöliittymän ja tulostaa sen näkyviin mobiilipäätelaitteen näytölle. ReGui:n toiminta pohjautuu laitteistokiihdytettyyn grafiikkaan, jolloin käyttöliittymä tulee jokaisessa laitteessa olemaan samannäköinen. Mobiilisovelluksen käyttöliittymän kautta käyttäjä ohjaa ulkopuolisen laitteen toimintaa, johon ei tarvitse toteuttaa erillisiä ohjausmenetelmiä. [59.]

ReGui on laadittu olemaan yksinomaan yleiskäyttöinen käyttöliittymäpalvelu. Sen toiminta ei ole tuotekohtaisesti sidottu mihinkään. Käyttöliittymäpalveluja tarjotaan kaikille tarvitseville osapuolille. ReGui-sovelluksesta on mahdollista tarpeen mukaan laatia asiakas-käyttöinen versio. [59.]

Kuvassa 6 on esitetty ReGui-ohjelmiston rakenne.



Kuva 6. Ohjelmiston arkkitehtuuri.

Kuvan 6 ohjelmistorakenteen ylimpänä osana on ReGui-sovellus, joka vastaa etäkäyttöliittymän toiminnasta. Se käyttää alemman tason ReGuiLibs-kerroksen tarjoamia kirjastoja. ReGuiLibs-kerrokseen kuuluvat UI-, Com- ja Core-kirjastot. Core-kirjasto antaa ReGui-sovellukselle muun muassa merkkijonojen ja taulukoiden käsittelyyn tarvittavat työkalut. Com-kirjasto antaa yleiset yhteystoiminnallisuudet ReGui-sovellukselle, esimerkiksi sarjaliikenne- ja TCP/IP- toiminnallisuudet. UI-kirjasto antaa käyttöliittymään kuuluvia toimintoja ReGui-sovelluksen käyttöön, esimerkiksi nappeja ja tekstifontteja. ReGuiLibs-kerros keskustelee HW-kerroksen kanssa, joka antaa tälle laitteiston palveluja. HW-kerros sisältää laitteistokirjastoja Android-, Windows- ja Web- laitteista ReGui:lle. [59.]

ReGui:n Android-kirjastoon on lisätty BLE-tekniikan tuki. Tuen lisäys mahdollistaa ReGui-sovelluksen yhteentoimivuuden opinnäytetyössä käytettävän SensorTagin kanssa.

6.2 CC2650 SensorTag

CC2650 SensorTag on BLE-oheislaite. Laitteen on valmistanut Texas Instruments. Laitteen toiminta perustuu langattomaan CC2650-mikrokontrolleriin. Mikrokontrollerin ansiosta SensorTagin virrankulutus on vähäinen. Vähäisellä virrankulutuksella SensorTag toimii vuosia pelkästään yhdellä kolikkoparistolla. SensorTag sisältää myös vähävirtaisia antureita. Laitteessa on muun muassa lämpötila-antureita, kosteusanturi ja kiihtyvyyssanturi. SensorTag sisältää iBeacon-tekniikan ja laitteesta on myös mahdollista aktivoida ZigBee- ja 6LoWPAN-tekniikat. [60.] [61.]

SensorTagin mikrokontrolleri CC2650 kuuluu vähävirtaisten ja 2,4 GHz radiotaajuutta käyttävien laitteiden CC26XX-perheeseen. CC2650:n kohteisiin kuuluvat esimerkiksi BLE- ja ZigBee-etälaitehallintasovellukset. Mikrokontrolleri käyttää pääprosessorinaan 32-bittistä ARM Cortex-M3-prosessoria, joka toimii taajuudella 48 MHz. CC2650-laitteessa on anturien ohjaaja. Vaikka muu järjestelmä olisi lepotilassa, anturien ohjaaja pysyy silti itsenäisesti keräämään dataa antureilta. [62.]

Opinnäytetyössä SensorTagien kehitystä varten käytettiin Code Composer Studio-työkalua.

6.3 Code Composer Studio

Code Composer Studio tai CCS lyhyesti on IDE, jolla voidaan kehittää Texas Instrumentsin sulautettuihin prosessoreihin perustuvia sovelluksia. Se on pääsääntöisesti laadittu sulautettujen projektien suunnittelemiseen. Ohjelmointikielenä, vastaavan kääntäjän kera, käytetään C/C++:aa. Uusimmat versiot IDE:stä pohjautuvat muokkaamattomiin lähdekoodin versioihin Eclipse IDE:stä. Uusin versio 8.0 on julkaistu 9.3.2018. IDE:n kehittäjä on Texas Instruments. [63.] [64.]

CCS tukee Texas Instrumentsin JTAG debug-luotaimia, joita kutsutaan XDS-emulaattoreiksi. Kaikissa CCS:n versioissa tuetaan XDS510- ja XDS560-luokkien luotaimia. Uuden luokan XDS100-luotaimien tuki aloitettiin uusimpien päivitysten myötä IDE:n versioon 3.3. Myöhemmät versiot lisäävät tukea sekä uusille että eri valmistajien JTAG-luotaimille. Muun muassa versiossa 5.2 lisättiin tuki XDS200-luokan debug-luotaimille. [63.]

CCS sisältää TI-RTOS-työkaluympäristön monille Texas Instrumentsin sulautetuille prosessoreille. Työkalu käsittää reaaliaikaympäristön nimeltään TI-RTOS Kernel. TI-RTOS

Kernel rakentuu erillisistä osista. Osat tarjoavat palveluita rajapintojen kautta ja ovat yksilöllisesti muokattavissa. Sulautteluille sovelluksille TI-RTOS-työkalu tarjoaa muun muassa muistinhallintaa ja reaaliaikaista debuggausta TI-RTOS Kernelin kautta. Työkalussa on reaaliaikaympäristön lisäksi osia, jotka tukevat esimerkiksi tiedostojärjestelmiä. [65.]

Opinnäytetyössä käytettiin Visual Studiosta, koska työn ReGui-sovellusta kehitettiin toimimaan paremmin.

6.4 Visual Studio

Visual Studio on Microsoftin kehittämä IDE. IDE:tä voidaan käyttää tietokoneohjelmien ja mobiilisovellusten tekemiseen. Visual Studio tukee 36:ta eri ohjelmointikieltä. Se antaa koodieditorin ja debuggerin tukea melkein kaikkia ohjelmointikieliä. Visual Studion tuemia ohjelmointikieliä ovat esimerkiksi C, C++ ja Visual Basic. Näiden ohjelmointikielien tuet tulevat Visual Studion mukana. Tuot kuten Python- ja Node.js-ohjelmointikielille, asennetaan saatavien lisäosien kautta. Uusin versio, Visual Studio 2017, on julkaistu 7.3.2017. [66.]

Visual Studion koodieditorin ominaisuuksiin kuuluvat lähdekoodin korostaminen ja koodin automaattinen täydennys. Navigointia varten koodissa Visual Studion koodieditori mahdollistaa kirjainmerkkien asettamisen, koodipätkien piilottamisen ja sanojen etsinnän koodista. Kun Visual Studiossa kirjoitetaan koodia, IDE kääntää koodia taustalla. Tämän avulla tunnistetaan virheitä koodin kirjoituksessa ja koodin käänöksissä. Virheet korostetaan koodissa punaisella alleviivauksella. [66.]

Visual Studion debuggeria voidaan käyttää sovelluksille, jotka on laadittu Visual Studion tuetuilla ohjelmointikielillä. Debuggeri voidaan liittää suoritettavaan prosessiin niiden tarkkailuja ja debuggausta varten. Jos prosessin lähdekoodi on saatavilla, debuggeri näyttää koodia suorituksen aikana. Breakpointit lopettavat koodinsuorituksen halutusta kohtaa koodia ja näiden asettamisen debuggeri sallii muuttujien tarkkailemiseen tarkoitettujen vahtien kanssa. [66.]

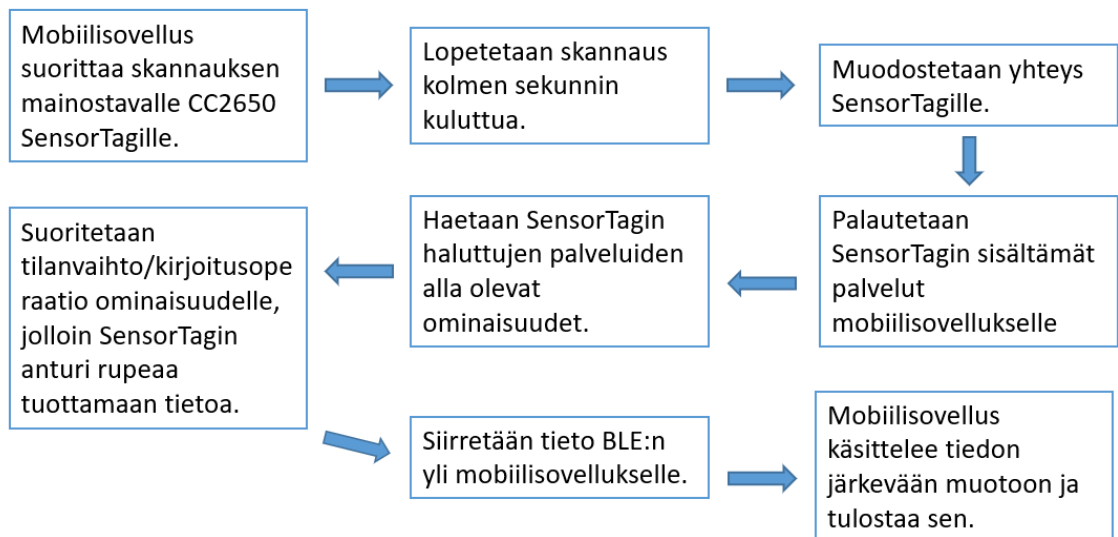
Näitä välineitä käytettiin tämän työn järjestelmän tekemisessä.

7 Työn suunnittelu, toteutus ja testaus

Tämän työn BLE järjestelmän toteutuksessa kummallekin SensorTagille annettiin roolit, jotka on määritetty BLE:n protokollapinossa. Tässä osiossa selitetään työn kulun vaiheet, testaukset ja testauksien tulokset. Työn BLE-järjestelmän rakenne ja BLE-protokollapinon määrittämät laiteroolit selitetään myös tässä osiossa.

Vaiheet BLE-yhteyden muodostamiseksi

Alla olevassa kuvassa on esitetty vaiheina BLE-yhteyden muodostaminen työn mobiilisovelluksen ja SensorTagin välille.



Kuva 7. BLE-yhteyden muodostamisen vaiheet mobiilisovelluksen ja SensorTagin välille

Kuvassa 7 mobiilisovellus suorittaa ensimmäisenä skannausoperaation, jossa se etsii mainostavia BLE-laitteita, tässä tapauksessa SensorTagia. Skannausoperaatio kestää kolme sekuntia. Jos SensorTag ei löydy aikamäärän sisään, skannaaminen on suoritettava uudelleen. SensorTagiin yhdistetään, kun se löydetään. Sen palvelut palautetaan mobiilisovellukselle yhdistämisen jälkeen ja tietoon haetaan haluttujen palvelujen ominaisuudet. Ominaisuusparametrin tilaa vaihdetaan, jolloin SensorTag tulostaa ulos tietoa. Tieto siirretään BLE-yhteyden yli mobiilisovellukselle. Mobiilisovellus käsittelee tietoa muuntamalla sen oikeaan muotoon. Muunnoksen jälkeen mobiilisovellus tulostaa tiedon näkyviin.

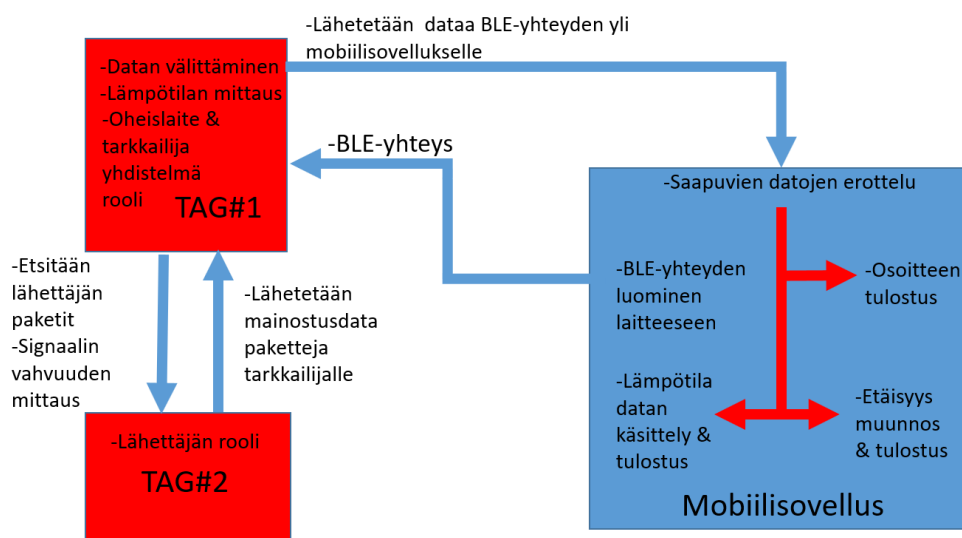
7.1 BLE-laiteroolit

BLE:n yleinen pääsyprofiili (GAP) määrittää neljä roolia, jossa laitteet voivat toimia: keskuslaite, oheislaite, tarkkailija ja lähettäjä. Lähettäjän rooliin määritelty laite lähettää dataa sisältäviä mainospaketteja, joita tarkkailijalaitteet etsivät ja lukevat. Keskuslaite tarkkailijan tavoin etsii mainospaketteja, mutta löydettyään sille sopivan mainospaketin se aloittaa yhteyden muodostamisen lähettämällä yhteyspyynnön yhdelle tai useammalle mainostavalle oheislaitteelle. Kun oheislaite hyväksyy yhteyspyynnön, yhteys muodostetaan laitteiden välille, oheislaite lopettaa itsensä mainostamisen ja keskuslaite aloittaa molempiin suuntiin tapahtuvan tiedonvaihdon laitteiden välillä. Toisin kuin keskuslaite, tarkkailija ei pysty muodostamaan yhteyttä lähettäjään eli se ei tue yhteyden muodostamista. Laitteille pystytään asettamaan myös näiden roolien yhdistelmiä. Esimerkiksi lähettäjä voi toimia myös tarkkailijan roolissa. Lähettäjä näin pystyy lähettämään ja lukemaan muiden lähettäjien mainospaketteja. [67.]

Yleinen attribuuttiprofiili (GATT) asettaa laitteille asiakas- ja palvelinroolit. Roolit ovat käytössä laitteilla, joiden välillä on yhteys. Palvelin sisältää tietoja, jonne asiakkaat lähettävät pyyntöjä tiedon lukemista ja kirjoittamista varten. Yleisen attribuuttiprofiilin roolit ovat riippumattomia yleisen pääsyprofiilin (GAP) keskus- ja oheislaiterooleista. Keskus- ja oheislaitteet voivat olla joko palvelimia tai asiakkaita. [67.]

BLE-järjestelmän rakenne

Kuvassa 8 on esitetty tämän työn BLE-järjestelmän rakenne.



Kuva 8. Työn BLE-järjestelmän rakenne.

Kuten yllä olevasta kuvasta näkee, järjestelmän ensimmäiselle SensorTagille on määritetty oheislaitteen ja tarkkailijan yhdistelmärooli. Toisella SensorTagilla on määritettynä pelkästään lähettäjän rooli. Lähettäjän roolissa SensorTag lähettää tietyin väliajoin mainostusdatapaketteja. Datapaketteihin on lisättyä SensorTagin laitteen osoite. Näitä paketteja etsii yhdistelmäroolin SensorTag mitaten samalla sen signaalin voimakkuutta. Löydettyään paketit se ottaa itselleen talteen pakettien sisältämät tiedot. Yhdistelmäroolin SensorTag pakettien etsinnän lisäksi suorittaa lämpötilamittauksia ja on yhteydessä oheislaiteroolin kautta keskuslaitteeseen.

Yhdistelmäroolin SensorTag välittää keskuslaitteen mobiilisovellukselle lämpötiladatan ohella talteen otettujen mainospakettien sisältämän tiedon ja signaalin voimakkuuden. Mobiilisovellukselle nämä datat saapuvat yhdistelmädatana. Vastaanotettuaan yhdistelmä datan mobiilisovellus suorittaa datojen erottamisen toisistaan. Erottelun jälkeen mobiilisovellus suorittaa datojen käsittelyn. Lämpötiladata muutetaan järkevään muotoon ja signaalin voimakkuuden perusteella lasketaan etäisyys. Laiteosoitetta ei käsitellä. Datojen käsittelyn jälkeen mobiilisovellus tulostaa tiedot näkyviin. SensorTag toimii tämän ja mobiilisovelluksen välisessä yhteydessä yleisen attribuuttiprofiilin (GATT) palvelinroolissa. Mobiilisovelluksen keskuslaite on asiakasroolissa yhteyden aikana.

7.2 Työn kulku

Ensin työssä lähdettiin parantamaan BLE-järjestelmässä käytettävää mobiilisovellusta. Mobiilisovelluksen luokkaa nimeltään "Service" optimoitiin. Luokkaan luodaan objekteja SensorTagin sisältämistä palveluista. SensorTag palauttaa sen sisältämien palveluiden tunnisteet mobiilisovellukselle, kun sovellus hakee niitä. Palveluiden tunnisteet lisätään mobiilisovelluksessa omaan taulukkoon. Taulukon sisältämistä palveluista "Service"-luokkaan luodaan objekteja tässä tapauksessa vain SensorTagin antureiden palveluista. Kun taulukon käsittelyn aikana vuorossa on anturipalvelun tunniste, lisätään tunniste nimen kera uutena objektina "Service"-luokkaan. Luodut objektit palveluista ovat näkyvissä mobiilisovelluksen käyttöliittymän listassa, jossa ne näkyvät nimen perusteella. Objektille nimi tulee lisättävän anturipalvelun tunnisteeseen perusteella. Objekteihin tallentuvat myös anturipalveluiden ominaisuusparametrien tunnisteet. Ominaisuudet sijaitsevat palveluiden alla SensorTagissa. Parametritunnisteet haetaan SensorTagilta. Ominaisuusparametrien tunnisteet tallennetaan oikeisiin objekteihin. Ominaisuuksien tunnisteet tallentuvat objekteissa sijaitseviin taulukoihin mobiilisovelluksessa. Näin voidaan toteuttaa pa-

remmin ominaisuusparametrien kirjoitusoperaatio, koska ominaisuusparametrien tunnisteet sijoittuvat objektien taulukoissa samaan järjestykseen. Jokaisen luodun palveluobjektin kohdalla, sitä vastaavan anturipalvelun käynnistämisen yhteydessä, voidaan käyttää samoja funktioita. Funktiot hakevat kirjoitusoperaatioon tarvittavat ominaisuusparametrit objektien taulukoista aina samasta indeksistä.

Muutoksien jälkeen mobiilisovelluksen koodissa mobiilisovellus saatiin toimimaan dynaamisemmin. Tämän jälkeen aloitettiin roolien vaihtaminen SensorTageille. Työn tavoitteiden saavuttamiseksi rooleiksi valittiin lähettäjän rooli ja yhdistelmärooli, joka on yhdistelmä oheislaitteesta ja tarkkailijasta. Laiteroolien vaihtamiseksi SensorTagit tarvitsevat erillisen niille tarkoitetun Texas Instrumentsin valmistaman "Debugger Devpack"-debuggerin. Code Composer Studio IDE ladattiin debuggerin käyttöönottoa varten. IDE asentaa tietokoneelle tarvittavat XDS110 debug-luotaimen ajurit, joita debuggeri käyttää. Latauksen jälkeen Texas Instrumentsin kotisivuilta asenettiin BLE-ohjelmistopino CC2640 ja CC2650 mikropiiriä käyttäville laitteille. Ohjelmistopino sisältää muun muassa BLE-protokollapinon ja esimerkkikoodeja BLE:n yleisen pääsyprofiilin (GAP) määrittämistä rooleista. Ohjelmistopinon mukana tuli ohjelmistosuunnittelijalle tarkoitettu käyttöohje, johon tutustuttiin. Käyttöohjeessa esimerkiksi selitetään yleisen pääsyprofiilin (GAP) muokkaamiseen liittyviä komentoja ja muuttujia.

Lähettäjäroolin vaihtamiseksi käytettiin valmista ohjelmistopinon mukana tullutta projektia nimeltään "simple_broadcaster_CC2650lp_app". Projekti sisältää esimerkkikoodin nimeltään "simple_broadcaster.c", joka muokkaa laitteen lähettäjän rooliin. Projektille piti antaa tuki lähettäjäroolia varten. Sekä roolin tuen että BLE-ominaisuuksien valitseminen tehdään käytettävän projektin stack-projektissa eli tässä tapauksessa "simple_broadcaster_CC2650lp_stack"-projektissa. Projektissa on kansio nimeltään "TOOLS", jossa puolestaan on tiedosto nimeltään "build_config_opt". Lähettäjäroolin tuen valitsemiseksi tiedostosta valittiin käyttöön määrittäminen "BROADCASTER_CFG". BLE-ominaisuuksien valitseminen tiedosta laittaa stack-projektin käyttämään määrittämiä vastaavia luokkakirjastoja tämän käännöksen aikana. Stack-projekti sisältää BLE-protokollakerroksen rajapinnan, josta sovellukset voivat kutsuvat funktiota esimerkiksi yleisen pääsyprofiilin (GAP) alustamiseen.

Lähettäjäroolin koodiprojekti on laadittu eri alustalle kuin SensorTag, joten projektin kohdealusta oli vaihdettava SensorTag alustaan. Vaihto onnistui yksinkertaisesti muuntamalla projektin asetuksissa ennalta määrätyn alustan merkki haluttuun kohdealustaan. Tässä tapauksessa projektin alustan merkki "CC2650_LAUNCHXL" muutettiin vasta-

maan SensorTag alustan merkkiä "CC2650STK". Alustan valinnalla asetetaan sovitut ajurimuunnokset alustasta riippuen, esimerkiksi sovitaan, miten pinnit määritellään pinni-ajurissa. BLE-ohjelmistopinon käyttöön yhteensopivat alustat on lueteltu "board.c"-tiedostossa. Tiedosto sijaitsee ohjelmistopinossa esimerkkiprojektien "Startup"-kansiossa. Alustan vaihdon jälkeen tehtiin käännökset ensin stack-projektille ja sitten esimerkkikoodin projektille. Käännösten jälkeen projektit ladattiin SensorTagille debugin kautta. Ensin ladattiin stack-projekti ja toisena esimerkkiprojekti.

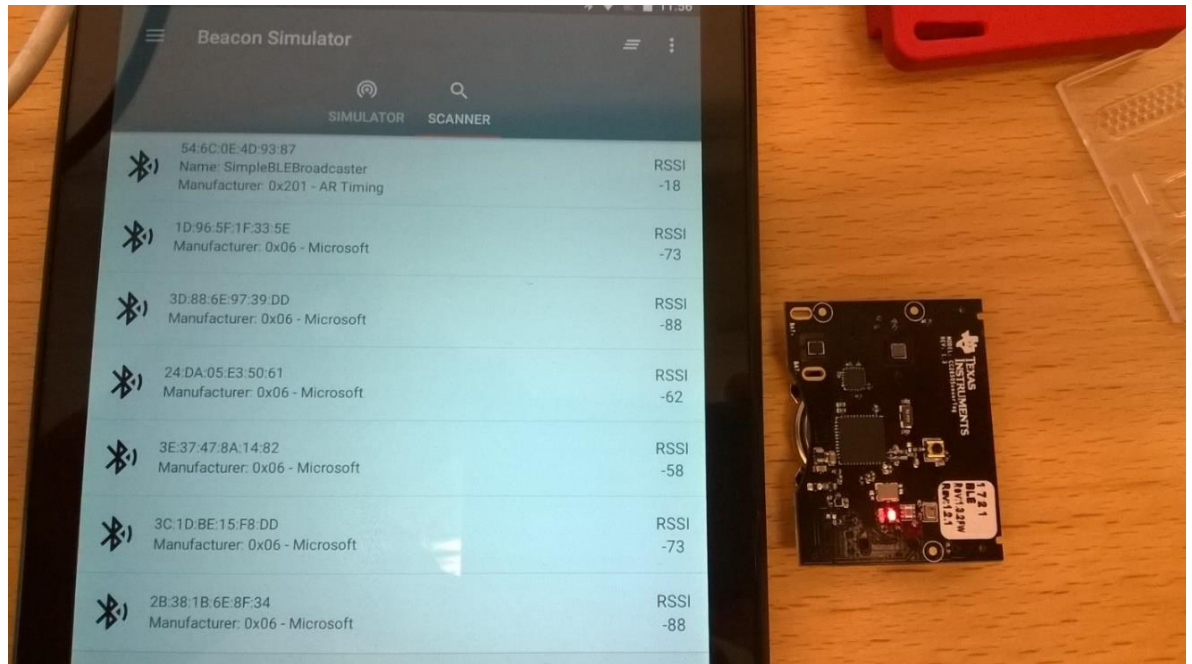
Yhdistelmäroolin asettamiseksi GitHubista ladattiin koodiprojekti nimeltään "simple_peripheral_observer_CC2650lp_app" ja sitä vastaava stack-projekti. Projektin esimerkkikoodi "simple_peripheral_observer.c" saa laitteen toimimaan tarkkailijan ja oheislaitteen yhdistelmäroolissa. Koodi muokkaa BLE:n yleisen pääsyprofiilin (GAP) kerrosta asettamalla yhdistelmäroolin parametrit. Parametrien asettamisella laitteelle asettuvat esimerkiksi halutut mainostus- ja skannausvälit. Samantapainen parametrien asettaminen tehdään myös lähettäjälaitteen roolille esimerkkikoodissa "simple_broadcaster.c".

Yhdistelmäroolin vaihtamiseksi valmiiseen projektiin "sensortag_CC2650stk_app" lähdettiin lisäämään tarkkailijaa, oheislaiteroolin rinnalle, ottamalla mallia tarkkailijan roolin toteutustavasta yhdistelmäroolin esimerkkikoodista "simple_peripheral_observer.c". Muokattava koodiprojekti sisältää SensorTag-sovelluksen alkuperäisen koodin, jossa otetaan käyttöön kaikki SensorTagin anturit. Tarkkailijaroolin lisääminen projektiin tarkoittaa, että SensorTag voi olla yhteydessä mobiilisovellukseen ja etsiä mainosdatapaketteja.

Muokkaus aloitettiin valitsemalla projektille oheislaitteen ja tarkkailijan yhdistelmäroolin tuki. Valinta tehtiin projektia vastaavassa stack-projektin "build_config.opt"-tiedostossa, jossa otettiin käyttöön määrittäminen "DHOST_CONFIG=PERIPHERAL_CFG+OBSERVER_CFG". Projektin kohdealustaa ei vaihdettu, koska muokattava projekti oli laadittu SensorTag-alustalle. SensorTag-esimerkkiprojektissa on "PROFILES"-niminen kansio, jossa on tiedostot "peripheral.h" ja "peripheral.c". Tiedostot muokkaavat yleisen pääsyprofiilin (GAP) toimimaan oheislaiteroolissa määrittämällä roolin funktiot. Funktiota käytetään saavuttamaan roolin toiminnallisuus. Tiedostot korvattiin "peripheral_observer.h"- ja "peripheral_observer.c"-tiedostoilla. Kyseiset tiedostot saatiin yhdistelmäroolin esimerkkiprojektin vastaavasta "PROFILES"-kansioista. Tiedostot muokkaavat yleistä pääsyprofiilia (GAP) tarkkailijan ja oheislaitteen yhdistelmärooliin määrittämällä kummankin roolin tarpeelliset funktiot. Muokkaamisen jälkeen koodiprojekti "sensortag_CC2650stk_app" ja sen stack-projekti käännettiin ja ladattiin SensorTagille debugin kautta.

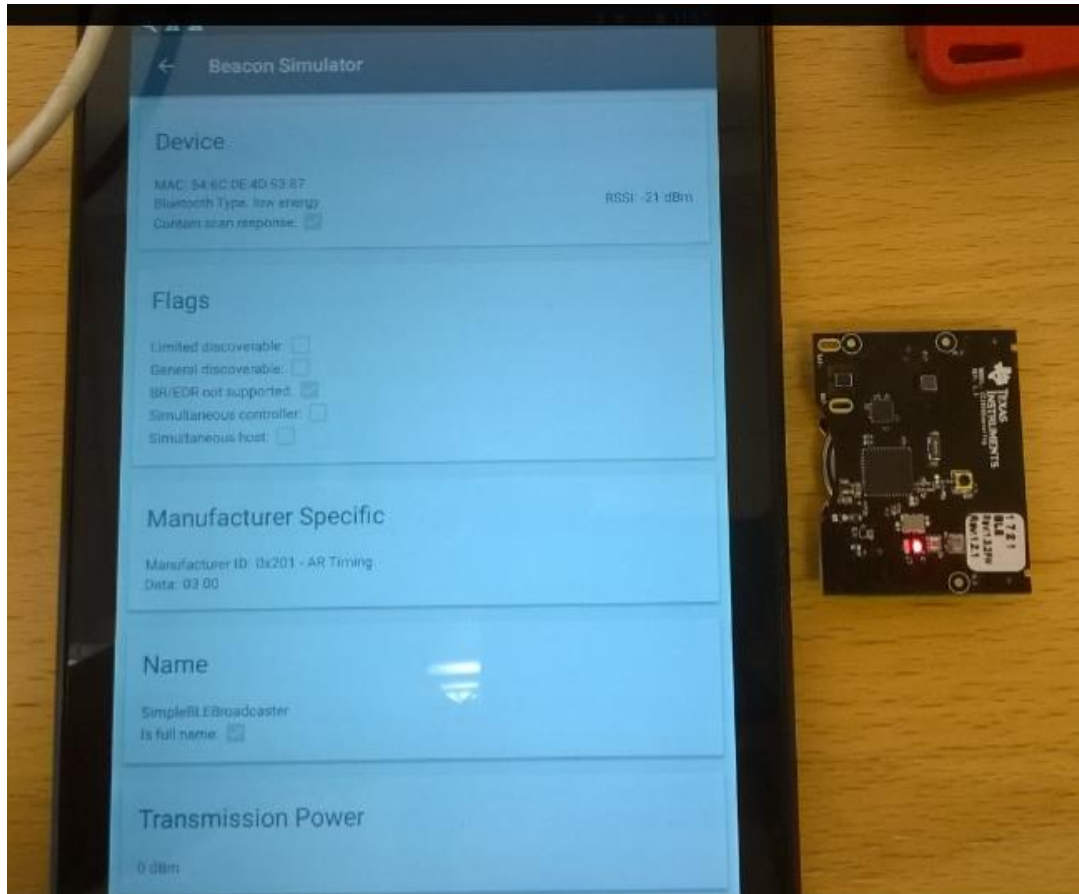
7.3 Roolien testaaminen

Koodiprojektin "simple_broadcaster_CC2650lp_app" ja tämän stack-projektin käännösten lataamisen jälkeen SensorTag asetui lähettäjän rooliin. SensorTag aloitti mainospakettien lähettämisen sovituin aikaväleihin. Lähettäjän toimivuutta testattiin lataamalla mainostusviestejä lukeva mobiilisovellus Google Play-kaupasta nimeltään "Beacon Simulator" (kuva 9).



Kuva 9. Lähettjäroolin testaus "Beacon Simulator"-sovelluksella.

Kuten kuvasta 9 näkee, mobiilisovellus löytää mainostavan SensorTagin. SensorTagissa palaa punainen led valo mainostuksen ajan. Mobiilisovellus löytää SensorTagin nimellä "SimpleBLEBroadcaster". Nimi on määritetty esimerkikoodissa "simple_broadcaster.c". Nimi on osa esimerkikoodissa laadittua vastauspakettia, jolla mainostavat laitteet vastaavat skannauksiin. Mobiilisovellus tulostaa nimen lisäksi SensorTagin osoitteen, valmistajan tunnisteiden ja signaalin vahvuuden. Osoitteen ja valmistajan tunniste tulevat mobiilisovellukselle mainostuksen kautta. Signaalin vahvuuden mittaamisen sovellus hoitaa itse. Kun SensorTagin valitsi mobiilisovelluksesta, tulostettavien laitteiden listasta, aukesi kuvan 10 mukainen näkymä.



Kuva 10. Sovelluksen näyttämät tiedot lähettäjistä.

Kuvassa 10 mobiilisovellus esittää tarkempia tietoja SensorTagista. Mobiilisovellus näyttää mitä lippuja on asetettu, mainospakettien sisältämän datan ja asetetun lähetysoimomakkuuden. Mainospaketteihin lisättävä datan tyyppi esimerkkikoodissa "simple_broadcaster.c" on valmistajaspesifistä. Koska datan tyyppi on esimerkkikoodissa valmistajaspesifinen, mainospaketeissa tulee mukana laitteen valmistajan tunnistetieto. Datatyyppiä seuraa itse mainospakettien data. Koodissa pakettien dataksi asetetaan vain pelkkiä lukuja. Tieto lippujen asetuksista laitteessa tulee mobiilisovellukselle mainospakettien kautta. Koodissa aktiiviseksi lipuksi asetetaan vain se, että laite ei tue tavallista BT-tekniologiaa. Lähetysoimomakkuus on asetettu nolaksi ja tieto tästä välittyy mobiilisovellukselle, ei mainospaketin, vaan vastauspaketin mukana. SensorTag lähettäjäroolissa todettiin toimivan oikein, koska ladattu mobiilisovellus löytää SensorTagin, saa siltä vastauspaketin ja se pystyy lukemaan esimerkkikoodissa määritettyjen mainospakettien sisältämää dataa SensorTagilta.

Toinen SensorTag asettui yhdistelmärooliin, kun projekti "sensortag_CC2650stk_app" ja sitä vastaava stack-projekti käännettiin ja ladattiin SensorTagille. Yhdistelmäroolin testaamisessa käytettiin lähettäjäroolin SensorTagia, jonka mainospaketteja yritettiin lukea. Mainospakettien skannauksen ajaksi yhdistelmäroolin SensorTag laittaa vihreän ledin päälle. Jos skannaus ei onnistu, sytytetään punainen ledi. Skannaus alkaa sivunapin painalluksesta.

Skannauksen käynnistyessä SensorTag sytytti punaisen ledin eli skannaus epäonnistui. Skannaus epäonnistui aina, kun sitä yritti uudelleen. Syy skannaamisen epäonnistumiselle oli annettavien parametrien sopimattomuudessa skannausta suorittavalle funktiolle. Funktion parametrien arvoja muutettiin, mutta skannaus silti epäonnistui. Syytä miksi parametrit eivät kelvanneet funktiolle, ei onnistuttu löytämään.

Yhdistelmäroolin SensorTag kuitenkin pystyi muodostamaan yhteyden järjestelmän mobiilisovellukseen ja lähettämään sille lämpötiladataa. Testauksesta todettiin, että yhdistelmärooli toimii osittain oikein SensorTagissa. Yhdistelmäroolin oheislaittepuoli toimii oikein, mutta tarkkailijan puoli taas ei toimi oikein. Roolien testaamisen jälkeen BLE-järjestelmän rakentaminen jäi kesken ajan puutteen vuoksi.

8 Jatkokehitys

Tässä osiossa kerrotaan, miten työn BLE-järjestelmää kehitettäisiin eteenpäin. Yhdistelmärooli olisi saatava toimimaan täysin SensorTagissa, eli tarkkailijan skannaus on korjattava. Skannaus muutetaan alkamaan SensorTagin käynnistyksen yhteydessä, ja se on koko ajan päällä. Yhdistelmäroolin SensorTagin olisi kuunneltava ja saatava ainoastaan mainospaketteja lähettäjäroolissa olevalta SensorTagilta. Tämä onnistuisi vaihtamalla kummankin SensorTagin mainostukseen ja skannaamiseen käytettävä radiokanava samaksi. Riskinä on, että valitulla radiokanavalla on muita mainostavia lähettäjälaitteita. Silloin yhdistelmäroolin SensorTag saa paketteja näiltä. SensorTagiin joutuisi lisäämään pakettien suodatuksen, jotta paketteja saadaan lähettäjä-SensorTagilta.

Kun yhdistelmärooli olisi kunnossa ja paketteja luetaan vain lähettäjä SensorTagilta, mainospakettien data lisättäisiin omaan, itse luotuun palveluun. Palvelu pitää sisällään yhden kirjoitettavan ominaisuusparametrin, joka sisältää mainostuspakettien datan. Sekä palvelulle että ominaisuudelle generoidaan omat uniikit tunnisteet. Ominaisuutteen on liitettynä ilmoitusten aloittaminen. Ilmoituksilla ilmoitetaan datan saatavuudesta ominaisuudessa mobiilisovellukselle. Jos dataa on saatavilla ja ilmoitukset aloitettu, dataa lähetään ominaisuusparametristä automaattisesti mobiilisovellukselle saatavuuden mukaan. Oma palvelu lisätään SensorTagin muiden palveluiden joukkoon. Palvelu on näkyvissä mobiilisovellukselle sen hakiessa SensorTagin palvelut. Palvelusta tehdään mobiilisovelluksessa objekti palvelun ja ominaisuuden tunnisteista nimen kanssa. Ominaisuuden kirjoitusoperaation jälkeen ominaisuuteen lisätty mainosdata lähetetään mobiilisovellukselle.

Oman palvelun ja ominaisuuksien luominen voi olla työläs prosessi. Jos omaa palvelua ei luo, niin mainosdatan voisi lisätä olemassa olevaan palveluun. Mainosdata sisällytettäisiin BLE-järjestelmässä tässä tapauksessa SensorTagin lämpötila-anturin palveluun. Palveluun lisäys olisi suoraan joko lämpötiladataa pitävään ominaisuuteen tai tekemällä lämpötilapalveluun uuden ominaisuuden mainosdataa varten. Oman ominaisuuden lisääminen palveluun vastaisi ominaisuuden luomista, kun tehdään omaa palvelua, eli ominaisuuden lisäys olisi työläämpää.

Tässä osiossa mainitut jatkokehitys työmahdollisuudet kohdistuisivat lähinnä SensorTag-laitteiden koodien laatimiseen. Mobiilisovellukseen lisätään koodia uuden datan mukaan SensorTagilta. Pilveen tehtävä datan siirto mobiilisovelluksesta kolmannelle osapuolelle voidaan tehdä näkyväksi ja haettavaksi, kun datan siirto toimii mobiilisovellukselle.

9 Työn tavoitteiden saavuttaminen

Tämän työn asetetut tavoitteet saavutettiin osittain. Tarvittavat roolimuuunnokset Sensor-Tageille onnistuivat. Lähettäjäroolin SensorTag lähettää mainospaketteja. Toisen Sensor-Tagin oheislaitteen rooli toimii yhdistelmäroolista, koska mobiilisovellus saa muodostettua siihen yhteyden. Oheislaite lähettää onnistuneesti lämpötiladataa mobiilisovellukselle. Lämpötiladata saadaan muunnettua oikeanlaiseksi ja tulostettua.

Vaikka työssä yhdistelmärooli saatiin vaihdettua, se ei toimi täysin oikein. Tarkkailija ei pysty keräämään lähettäjä-SensorTagin laiteosoitetta ja signaalin vahvuutta talteen, koska tarkkailijan skannaus ei toimi oikein. Tarkkailijapuolen epäonnistumisen vuoksi käyttäjän tunnistus ja signaalin voimakkuuden mittaaminen eivät toteutuneet. Näitä tietoa ei saada välitettyä mobiilisovellukselle. Kun mobiilisovellus ei tiedä signaalin vahvuutta, etäisyyden laskeminen ja tulostus jäävät tekemättä.

Tavoitteiden saavuttamiseksi BLE-järjestelmän työnanto olisi pitänyt ymmärtää paremmin. Järjestelmän työnannosta ensin oletettiin, että mobiilisovellus muodostaa yhteyden kumpaankin SensorTagiin. Mobiilisovellus saisi lämpötiladataa molemmilta Sensor-Tageilta. Se hoitaisi myös signaalin voimakkuuden ja laitteen osoitteen hankkimisen itse. Niinpä aluksi järjestelmää suunniteltiin nämä asiat mielessä. Järjestelmän suunnittelua piti muuttaa, kun oivallettiin järjestelmän toteutuvan laiteroolien avulla. Työnannon ymmärtäminen heti oikein olisi tarkoittanut, että järjestelmää olisi voinut heti lähteä laatimaan ”oikealla” tavalla. Enemmän aikaa olisi ollut laiteroolien koodien perehtymiseen, miettimiseen, miten mainosdataa saadaan mobiilisovellukselle ja roolien kuntoon saamiselle.

Järjestelmä jäi kaipaamaan myös tarkempaa suunnittelua. Suunnittelu tehtiin kiireisesti ajan puutteen vuoksi, minkä takia asioita jäi huomioimatta. Esimerkiksi mainosdatan vienti mobiilisovellukselle huomioitiin vasta yhdistelmäroolin testauksen vaiheessa. Vaikka vientiä ei ehditty toteuttaa, piti toteutustapa miettiä teoriassa työnteon aikana. Paremmalla suunnittelulla olisi saavutettu varmasti enemmän työlle asettuja tavoitteita.

10 Yhteenveto

Tämän opinnäytetyön tarkoituksena oli laatia BLE tiedonsiirtoteknologiaa käyttävä järjestelmä. Kahdella BLE-teknologiaa hyödyntävällä laitteella dataa lähetään Android-käyttöjärjestelmää käyttävän tabletin mobiilisovellukseen, käsiteltäväksi ja tulostettavaksi. Samalla tässä opinnäytetyössä tutustuttiin erilaisiin anturisovelluksiin, mobiilisovelluksen kehittämiseen tarkoitettuihin IDE:ihin ja perehdyttiin lyhyen kantaman langattomiin tiedonsiirtotekniikoihin.

Järjestelmän toteuttamisessa käytettiin aiempaa KajaPro yrityksen ReGui-ohjelmistotuotteella laadittua mobiilisovellusta ja kahta CC2650 SensorTag BLE-laitetta. ReGui:n avulla mobiilisovellus pystyy käyttöliittymäpalvelun kautta saamaan yhteyden SensorTag-laitteeseen.

Tämän työn järjestelmän toteutuksessa jouduttiin parantamaan mobiilisovelluksen toimintaa. Työn teossa piti myös perehtyä BLE-protokollakerrosten määrittämiin laiterooleihin ja asettaa työn kannalta tarvittavat laiteroolit SensorTageille. Toinen SensorTageista on lähettäjän roolissa ja toinen SensorTag on yhdistelmäroolissa. Mobiilisovelluksen parantamisessa hyödynnettiin Visual Studio IDE:tä. SensorTag-laitteiden kehittäminen tapahtui Code Composer Studio IDE:ssä erillisen debuggerin kera. Ohjelmointikielenä mobiilisovelluksen ja laitteiden kehityksessä käytettiin C++:aa.

Internetistä ladattiin BLE-ohjelmistopino, joka sisälsi esimerkkiprojekteja laiterooleista. Pinnon lisäksi GitHubista hankittiin erillinen valmiiksi laadittu esimerkkiprojekti yhdistelmäroolista. Näiden avulla saatiin asetettua kahdelle SensorTagille työssä käytettävät laiteroolit.

Työlle asetetut tavoitteet onnistuttiin saavuttamaan osittain. SensorTag-laitteet asettuivat laiterooleihinsa koodiprojektien ansioista. Lähettäjä-SensorTag mainostaa dataa ja yhdistelmäroolin SensorTagiin mobiilisovellus saa muodostettua yhteyden. Yhdistelmärooliin vajavaisesta toiminnasta johtuen mobiilisovellukselle ei saada lähetettyä lähettäjän mainosdataa. Tämän ja ajan puutteellisuuden takia järjestelmän tekeminen jäi kesken-eräiseksi.

Lähteet

- (1) KajaPro. KajaPro-Ohjelmistokehitys. Available at: <http://www.kajapro.fi/palvelut/ohjelmistokehitys/>. Accessed 15.1. 2018.
- (2) Zola Andrew. How IoT Will Impact Your Mobile App Development. 2017; Available at: <https://www.upwork.com/hiring/for-clients/how-iot-will-impacts-mobile-app-development/>. Accessed 18.1. 2018.
- (3) App Annie, Statista. Worldwide mobile app revenues in 2015, 2016 and 2020 (in billion U.S. dollars). 2016; Available at: <https://www.statista.com/statistics/269025/worldwide-mobile-app-revenue-forecast/>. Accessed 18.1.2018.
- (4) Chavda Ketan. Role of Mobile Apps in revolutionizing the world of IoT. 2016; Available at: <https://www.prismetric.com/role-mobile-apps-revolutionizing-world-iot/>. Accessed 18.1 2018.
- (5) Cuffari Benedette. The Future of Sensor Technology in the Global Market. 2017; Available at: <https://www.azosensors.com/article.aspx?ArticleID=775>. Accessed 19.1.2018.
- (6) BLE Mobile apps. Bluetooth Low Energy (BLE): Make Your Product, a Smart Product. 2017; Available at: <http://www.blemobileapps.com/blog/bluetooth-low-energy-ble-make-product-smart-product/>. Accessed 19.1.2018.
- (7) BLE Mobile Apps. BLE Technology and Mobile Applications. 2017; Available at: <http://www.blemobileapps.com/blog/ble-technology-mobile-applications/>. Accessed 14.1.2018.
- (8) Wikipedia - Aktiivisuusranneke. Available at: <https://fi.wikipedia.org/wiki/Aktiivisuusranneke>. Accessed 20.1.2018.
- (9) Teknavi. Vertailussa aktiivisuusrannekkeet: Polar, Garmin ja FitBit. 2015; Available at: <http://teknavi.fi/elektroniikka/vertailussa-aktiivisuusrannekkeet-polar-garmin-ja-fitbit>. Accessed 20.1.2018.
- (10) MobileSiri. Best Fitness Trackers in 2017 To Keep You Fit, Active & Motivated. 2017; Available at: <http://mobilesiri.com/best-fitness-trackers/>. Accessed 20.1.2018.
- (11) Bogdanov Vik. How wearables work: fitness bracelets and activity trackers. 2015; Available at: <http://intersog.com/blog/tech-tips/how-wearables-work-fitness-bands-and-activity-trackers/>. Accessed 20.1.2018.

- (12) Clinical applications of plantar pressure measurement in pediatric orthopedics. Pediatric Gait: A New Millennium in Clinical Care and Motion Analysis Technology; 2000.
- (13) Abdul Hadi, Abdul Razak, Aladin Zayegh, Rezaul K. Begg, Yufridin Wahab. Foot Plantar Pressure Measurement System: A Review. 2012; Available at: <http://www.mdpi.com/1424-8220/12/7/9884/htm>. Accessed 20.1.2018.
- (14) Portable bluetooth low energy system for synchronize foot pressure measurement. 2017 IEEE 6th Global Conference on Consumer Electronics (GCCE); 2017.
- (15) Svensson Pelle. Bluetooth low energy possibilities in healthcare. 2017; Available at: <https://www.u-blox.com/en/blog/hot/bluetooth-low-energy-possibilities-healthcare>. Accessed 22.1.2018.
- (16) BLE Mobile Apps. Use of Bluetooth Low Energy Technology in Healthcare. 2017; Available at: <http://www.blemobileapps.com/blog/use-bluetooth-low-energy-technology-healthcare/>. Accessed 10.1.2018.
- (17) GlucoWise. Imagine living a healthier life with GlucoWise™. Available at: <http://www.gluco-wise.com/>. Accessed 10.1.2018.
- (18) Olcay. Beacons: Everything you need to know. 2015; Available at: <http://www.pointrlabs.com/posts/beacons-everything-you-need-to-know/>. Accessed 10.1.2018.
- (19) Jurejevcic Dejan. IoT tech deep-dive: The rise of beacon technology. 2015; Available at: <https://iot-analytics.com/rise-of-beacon-technology/>. Accessed 25.1.2018.
- (20) Wikipedia - Types of beacons. Available at: https://en.wikipedia.org/wiki/Types_of_beacons. Accessed 25.1.2018.
- (21) Shubhi Mittal. iBeacon vs Eddystone: Which one works better for your Pilot Project. 2016; Available at: <https://blog.beaconstac.com/2016/01/ibeacon-vs-eddystone/>. Accessed 25.1.2018.
- (22) Grand view research. Bluetooth Beacons Market Analysis By Technology (iBeacon, Eddystone), By End-use (Retail, Travel & Tourism, Healthcare, Financial Institutions), By Region, And Segment Forecasts, 2014 - 2025. 2017; Available at: <https://www.grandviewresearch.com/industry-analysis/bluetooth-beacons-market>. Accessed 25.1.2018.

- (23) Grand view research. Bluetooth Beacon Market Worth \$58.7 Billion By 2025 | CAGR: 95.3%. 2017; Available at: <https://www.grandviewresearch.com/press-release/global-blue-tooth-beacons-market>. Accessed 25.1.2018.
- (24) Inductive automation. The Industrial Internet of Things (IIoT). Available at: <https://inductiveautomation.com/what-is-iiot>. Accessed 26.1.2018.
- (25) Rouse Margaret. Industrial Internet of Things (IIoT). 2015; Available at: <http://internetofthingsagenda.techtarget.com/definition/Industrial-Internet-of-Things-IIoT>. Accessed 26.1.2018.
- (26) European Editors. Using Bluetooth Low Energy to Connect Wearable Wireless Sensors to the Internet of Things. 2017; Available at: <https://www.digikey.fi/en/articles/techzone/2017/may/using-bluetooth-low-energy-connect-wearable-wireless-sensors-iiot>. Accessed 11.1.2018.
- (27) Suhel Dhanani. Industrial Internet of Things (IIoT) and Its Impact on the Design of Automation Systems. 2016; Available at: <https://www.maximintegrated.com/en/app-notes/index.mvp/id/6142>. Accessed 26.1.2018.
- (28) Nordic semiconductor. MetaWear Nordic-powered modules simplify development of Bluetooth Smart wearables and sensor products. 2015; Available at: http://www.nordicsemi.com/node_206/node_305/Product-Related-News/MetaWear-Nordic-powered-modules-simplify-development-of-Bluetooth-Smart-wearables-and-sensor-products. Accessed 30.1.2018
- (29) MbitLab Inc. MetaWear: MetaSensors w/ Temp + Baro + Acc + Gyro + Light. 2015; Available at: <https://www.kickstarter.com/projects/guardyen/metawear-bluetooth-sensors-w-temp-pressure-acc-gyr>. Accessed 11.1.2018.
- (30) MbitLab Inc. MetaSensors. Available at: <https://mbitlab.com/platform/sensors/#>. Accessed 11.1.2018.
- (31) Wikipedia - programming tools. Available at: https://en.wikipedia.org/wiki/Programming_tool. Accessed 12.1.2018.
- (32) What is an integrated development environment (IDE)? Available at: <https://www.veracode.com/security/integrated-development-environments>. Accessed 12.1. 2018.
- (33) Wikipedia - Eclipse. Available at: [https://en.wikipedia.org/wiki/Eclipse_\(software\)](https://en.wikipedia.org/wiki/Eclipse_(software)). Accessed 12.1.2018.

- (34) FAQ Where Did Eclipse come from? 2006; Available at: http://wiki.eclipse.org/FAQ_Where_did_Eclipse_come_from%3F. Accessed 12.1.2018.
- (35) Eclipse. Eclipse - Platform architecture. Available at: <https://help.eclipse.org/neon/index.jsp?topic=%2Forg.eclipse.platform.doc.isv%2Fguide%2Farch.htm>. Accessed 13.2., 2018.
- (36) Meet Android Studio Available at: <https://developer.android.com/studio/intro/index.html>. Accessed 13.1.2018.
- (37) Wikipedia - Android Studio. Available at: https://en.wikipedia.org/wiki/Android_Studio. Accessed 13.1.2018.
- (38) Wikipedia - Kotlin (programming language). Available at: [https://en.wikipedia.org/wiki/Kotlin_\(programming_language\)](https://en.wikipedia.org/wiki/Kotlin_(programming_language)). Accessed 15.4.2018.
- (39) Techotopia. An Overview of the Android Architecture (Android Studio). Available at: [http://www.techotopia.com/index.php/An_Overview_of_the_Android_Architecture_\(Android_Studio\)#Summary](http://www.techotopia.com/index.php/An_Overview_of_the_Android_Architecture_(Android_Studio)#Summary). Accessed 28.2.2018.
- (40) Gupta N. Inside Bluetooth Low Energy. Norwood: Artech House; 2013.
- (41) Nordic Semiconductor. A short history of Bluetooth. 2014; Available at: <https://www.nordicsemi.com/eng/News/ULP-Wireless-Update/A-short-history-of-Bluetooth> Accessed 13.1.2018.
- (42) Bluetooth 5: What it's all about. Available at: <https://www.bluetooth.com/specifications/bluetooth-core-specification/bluetooth5>. Accessed 13.1.2018.
- (43). Dinesh Thakur. Bluetooth - What is Bluetooth? Available at: <http://ecomputer-notes.com/computernetworkingnotes/communication-networks/bluetooth>. Accessed 31.12.2017.
- (44) gc. Bluetooth tutorial-protocol stack. 2013; Available at: <http://www.tutorial-reports.com/wireless/bluetooth/protocolstack.php>. Accessed 1.1.2018.
- (45) gc. Bluetooth tutorial-core protocols. 2013; Available at: <http://www.tutorial-reports.com/wireless/bluetooth/bluetoothcoreprotocols.php>. Accessed 1.1.2018.
- (46) gc. Bluetooth tutorial-Telephony and Cable Replacement Protocol & RFCOMM. 2013; Available at: <http://www.tutorial-reports.com/wireless/bluetooth/telephonycablereplacementprotocol.php>. Accessed 1.1.2018.

- (47) gc. Bluetooth tutorial-adapted protocols. 2013; Available at: <http://www.tutorial-reports.com/wireless/bluetooth/adoptedprotocols.php>. Accessed 1.1.2018.
- (48) SMART sensor network: With Bluetooth low energy and CAN-BUS.; 2017
- (49) Kinnunen Asko. BLE yleiskatsaus v1. Accessed: 3.1.2018
- (50) MikroElektronika. Bluetooth Low Energy - Part 1: Introduction To BLE. 2016; Available at: <https://www.mikroe.com/blog/bluetooth-low-energy-part-1-introduction-ble>. Accessed 3.1.2018.
- (51) Frenzel Lou. What's the Difference Between Bluetooth Low Energy And ANT? 2012; Available at: <http://www.electronicdesign.com/mobile/what-s-difference-between-bluetooth-low-energy-and-ant>. Accessed 5.1.2018.
- (52) Wikipedia - ANT. Available at: [https://en.wikipedia.org/wiki/ANT_\(network\)](https://en.wikipedia.org/wiki/ANT_(network)). Accessed 5.1.2018.
- (53) Wikipedia - ANT+. Available at: <https://en.wikipedia.org/wiki/ANT%2B>. Accessed 4.1.2018.
- (54) ANT+ device profiles. Available at: https://www.thisisant.com/developer/ant-plus/device-profiles/#521_tab. Accessed 5.1.2018.
- (55) ANT- history. Available at: <https://www.thisisant.com/company/d1/history>. Accessed 13.1.2018.
- (56) Carne B. Professional's Guide to Data Communication in a TCP/IP World. Norwood: Artech House; 2004.
- (57) Fairhurst Gorry. The User Datagram Protocol (UDP).2008; Available at: <http://www.erg.abdn.ac.uk/users/gorry/course/inet-pages/udp.html>. Accessed 7.1. 2018.
- (58) Nadeem Unuth. TCP (Transmission Control Protocol) Explained. 2017; Available at: <https://www.lifewire.com/tcp-transmission-control-protocol-3426736>. Accessed 8.1.2018.
- (59) Mika Keränen. ReGui desing document. Accessed 20.4.2018.
- (60) Texas Instruments. CC2650 SensorTag User's Guide; Available at: http://processors.wiki.ti.com/index.php/CC2650_SensorTag_User%27s_Guide. Accessed 20.4.2018.
- (61) Texas Instruments. SimpleLink™ multi-standard CC2650 SensorTag™ kit reference design Available at: <http://www.ti.com/tool/TIDC-CC2650STK-SENSORTAG>. Accessed 20.4.2018.

- (62) Texas Instruments. CC2650 SimpleLink™ Multistandard Wireless MCU. Available at: <http://www.ti.com/product/cc2650>. Accessed 20.4.2018.
- (63) Wikipedia - Code Composer Studio. Available at: https://en.wikipedia.org/wiki/Code_Composer_Studio. Accessed 21.4.2018.
- (64) Texas Instruments. Code Composer Studio (CCS) Integrated Development Environment (IDE). Available at: <http://www.ti.com/tool/CCSTUDIO#descriptionArea>. Accessed 21.4.2018.
- (65) Wikipedia - TI-RTOS. Available at: <https://en.wikipedia.org/wiki/TI-RTOS>. Accessed 21.4.2018.
- (66) Wikipedia - Visual Studio. Available at: https://en.wikipedia.org/wiki/Microsoft_Visual_Studio Accessed 21.4. 2018
- (67) Kinnunen Asko. BLE perusteet. Accessed: 9.4.2018