



TAMPEREEN
AMMATTIKORKEAKOULU

TAMKin HYVINVOINNIN ETÄMITTAUSRATKAISU

IoT-laboratorio

Mika Helonsalo

Juho Kauppi

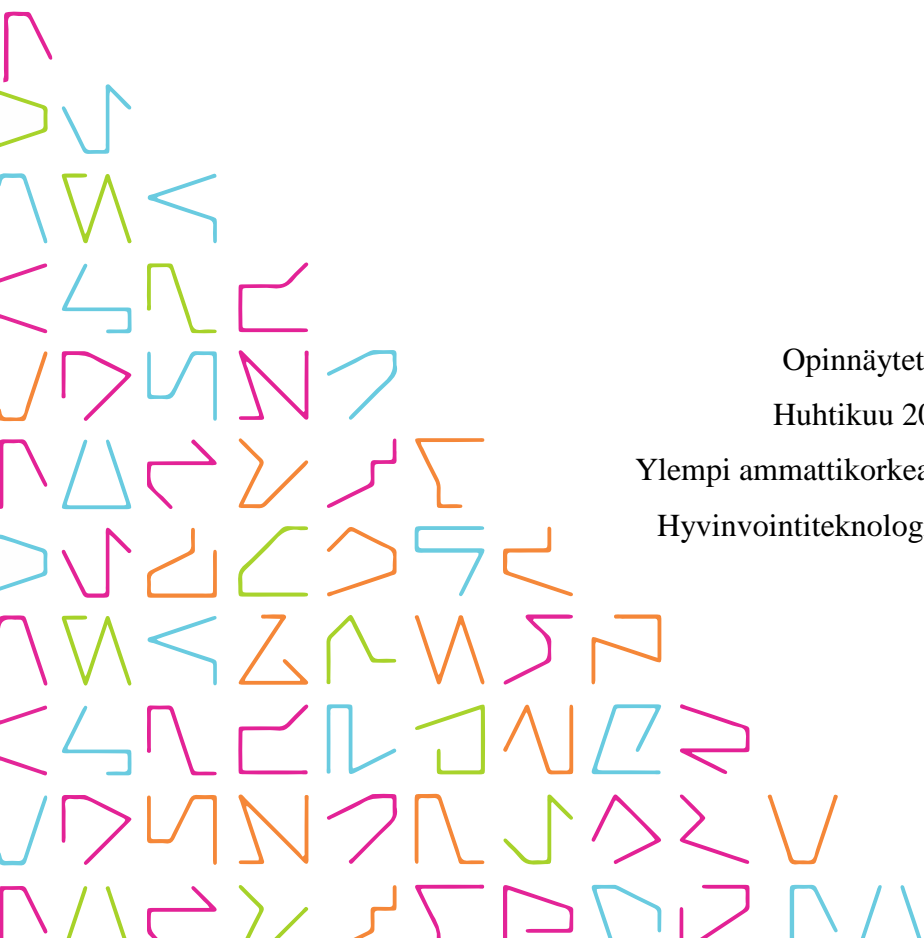
Timo Salminen

Opinnäytetyö

Huhtikuu 2017

Ylempi ammattikorkeakoulututkinto

Hyvinvointiteknologian koulutus



TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Ylempi ammattikorkeakoulututkinto
Hyvinvointiteknologian koulutus

HELONSALO, MIKA & KAUPPI, JUHO & SALMINEN, TIMO:

TAMKin Hyvinvoinnin etämittausratkaisu

IoT-laboratorio

Opinnäytetyö 115 sivua, joista liitteitä 11 sivua

Huhtikuu 2017

Opinnäytetyön toimeksiantajana oli Tampereen ammattikorkeakoulun Hyvinvointiteknologian YAMK koulutus. Työn tavoitteena oli suunnitella Hyvinvoinnin Etämittauksen sisältöjä sekä toimintoja IoT-laboratorion yhteyteen ja uusien opiskelijoiden käyttöön. Tämä hyvinvoinnin tai terveyden etämittaustutkimus sekä -ratkaisu palvelee opiskelijoita uudenaikaisena oppimisympäristönä. Tutkimuksessa on suunniteltu uudenlaisia tapoja suorittaa asiakkaan hyvinvoinnin tai terveyden seuranta etämittauksen avulla. Mukana tutkimuksessa oli Libelium e-Health-Kit sensoreita ja mikrokontrollereita, Arduino Mega 2560 ja Raspberry Pi 3 model B. Teoriaosuudessa on tutustuttu käytössä oleviin laitteisiin, ohjelmistosuunnittelun periaatteisiin, potilastiedon turvallisuuden vaatimuksiin ja toteutuksen arkkitehtuuriin sekä integraatioon.

Tutkimuksessa on keskitytty etämittauksiin, joita voitaisiin suorittaa helposti asiakkaan kotona. Alussa päähuomion vei IoT-alustan valinta, koska se vaikuttaisi kaikkein eniten ratkaisuumme kerätä tietoa sensoreista. Tietoa aihealueesta on hankittu osallistumalla seminaareihin, yritysvierailuille, henkilöhaastatteluille, lukemalla alan kirjallisuutta ja perehtymällä luotettaviin internetin tietolähteisiin. Kehitystehtäväksi muodostui määrittellä tapa käyttää e-Health-Kit sensoreita ja tehdä hankintaehdotus uusista sensoreista opetuskäyttöön. Kehitystyö osuuden ideat on luotu yhdessä työryhmän kesken ja tutkimusosuuden suoritus jakaantui osallisten osaamisen perusteella. IoT-alustan valinnassa päädyttiin Wapicen tekemään IoT-Ticket -alustaan, koska sitä näytti olevan helppo oppia käyttämään ja se soveltui hyvin käyttöön eri Tampereen ammattikorkeakoulun koulutusohjelmissa. Terveyden etämittauksen pilottiratkaisu on toteutettu käyttämällä e-Health-Kit sensoreista happisaturaatio- ja kaatumisanturia sekä mikrokontrollereita, että IoT-Tickettiä.

Hyvinvointiteknologian YAMK koulutuksen pitäisi panostaa lisää ajankohtaisten terveysalan ratkaisujen hankintaan ja käyttöön opetuksessa. Tällä tavalla uusien aikaa ja rahaa säästävien digitaalisten hoitovälineiden käyttöönotto opiskelijoiden työpaikoilla myös nopeutuu. Tiiviimpi yhteistyöhön terveysalan digitaalisia seurantaratkaisuja toteuttavien yritysten kanssa nopeuttaisi koulun omien ratkaisujen kehitystä ja tarjoaisi samalla oppilaille mahdollisuuden verkostoitua alan toimijoiden kanssa.

Asiasanat: hyvinvoinnin etäseuranta, IoT-alusta, käyttöliittymä, potilaan tietoturva

ABSTRACT

Tampereen ammattikorkeakoulu
Tampere University of Applied Sciences
Master's Degree in Wellbeing Technology

HELONSALO, MIKA; KAUPPI, JUHO & SALMINEN, TIMO:

TAMK Solution for Remote Wellbeing Monitoring

IoT-laboratory

Master's thesis 115 pages, appendices 11 pages

April 2017

The aim of this study was to collect information about wellbeing remote monitoring technologies and develop new solutions to be used as part of Wellbeing Technology teaching. This study will serve forthcoming students as a new kind of educational environment.

Libelium e-Health-Kit sensors were purchased earlier for use in TAMK IoT-laboratory. Microcontrollers Raspberry Pi 3 model B and Arduino Mega 2560 was needed also in this work. The target was to collect data from e-Health sensors and present it in the user interface. In theory part was studied knowledge related on Health remote monitoring, platform creation, application forming and patient information security.

The first TAMK remote health monitoring system was developed and it uses the Wapice IoT-Ticket as user interface. This is a good base for a new student works and theses. The study include also proposals for better health monitoring devices in educational use. These new devices would improve the quality of Wellbeing Technology teaching. In this Thesis was planned also a new application idea which collects information from a user's own sensors and sends data to the patient database.

Wellbeing technology teaching need to improve the knowledge of latest innovations in patient monitoring. Teaching of new technologies is based too much on knowledge searched from internet. To get new innovators to Finnish healthcare business one must be aware of the newest technology available and be also able to use it.

Key words: remote wellbeing monitoring, IoT-platform, user interface, patient information security

SISÄLLYS

TIIVISTELMÄ	2
1 JOHDANTO.....	7
2 OPINNÄYTETYÖN TAUSTA	9
2.1 Työn tavoite ja tarkoitus	9
2.2 Asiakas, kohderyhmä ja yhteistyökumppanit	11
2.3 Tutkimuskysymykset	12
2.4 Tutkimusmenetelmä.....	12
3 HYVINVOINNIN ETÄMITTAUS.....	13
3.1 Tiedonsiirtolaitteisto	14
3.1.1 Arduino	14
3.1.2 Raspberry Pi	17
3.2 Sensorit	19
3.2.1 SpO2 eli happisaturaation ja sykkeen mittaus	20
3.2.2 Verensokeri	21
3.2.3 Verenpaine	22
3.2.4 Kaatumisanturi	23
3.2.5 Hengitysilman määrä	24
3.3 Langattomat tietoverkot.....	25
3.3.1 Bluetooth.....	25
3.3.2 WLAN.....	26
3.3.3 5 G.....	27
3.3.4 LoRa.....	28
3.3.5 IPv6	29
3.4 IoT-alustan valinta	30
3.5 Käyttöliittymä – Kuntoutusohjelma.....	32
3.5.1 Käyttöliittymän suunnittelun vaiheita	32
3.5.2 Käyttöliittymän elementtien asettelu	34
3.5.3 Käyttöliittymän estetiikka	34
3.5.4 Käyttöliittymän testaus ja testimenetelmän valinta.....	36
3.5.5 Pilottitesti	41
3.5.6 Käyttäjäkokemukseen vaikuttavat tekijät	42
3.6 Tietoturva ja tietosuoja	45
3.6.1 Mihin tietoturvaa tarvitaan.....	45
3.6.2 Tietoturvan tavoitteet	46
3.6.3 Tietosuoja.....	47
3.6.4 Terveystietoturvan erityispiirteitä	48

3.7	Kokonaisarkkitehtuuri	51
3.8	Integraatio	52
4	HYVINVOINNIN ETÄMITTAUKSEN TOTEUTUS.....	54
4.1	IoT-alustan valintaperusteet.....	54
4.2	Tiedonsiirto sensoreista IoT-Ticketille.....	60
4.2.1	Sensoritiedon kerääminen	60
4.2.2	Tiedonsiirron ohjelmistoratkaisu	63
4.2.3	IoT – Ticket UI.....	68
4.3	IoT-Ticket matkapuhelimessa.....	71
4.4	Terveysdatan mittaaminen ihmisestä.....	74
4.4.1	Veren happisaturaatio.....	74
4.4.2	Verensokeri	75
4.4.3	Verenpaine	77
4.4.4	Kaatumisanturi	78
4.4.5	Hengitysilma määrä	79
5	Jatkokehitysajatuksia ja suosituksia	80
5.1	Käyttöliittymän suunnittelu	80
5.2	Arkkitehtuuri ja integraatio.....	85
5.3	Tietoturvan toteutus	87
5.3.1	Oppilaitoskäyttö	87
5.3.2	Potilaskäyttö.....	88
5.3.3	Yhteenveto	88
5.4	Ostosuosituksia	90
5.4.1	Verensokerin mittaus	90
5.4.2	Verenpaineen mittaus.....	92
5.4.3	Tuotepäheratkaisut.....	93
6	POHDINTA.....	97
	LÄHTEET.....	100
	LIITTEET	105
	Liite 1. etäk_arduino.ino	105
	Liite 2. main.cpp.....	109
	Liite 3. SendWap.c	114

LYHENTEET JA TERMIT

5G	The fifth generation, seuraava verkkoteknologiasukupolvi nykyisen 4G jälkeen
Applikaatio	Mobiilisovellus älypuhelimelle, tabletille tai PC:lle
ASDL	Asymmetric Digital Subscriber Line -verkkokytintekniikka
Big-data	valtava datamäärä, joka kerätään eri tietolähteistä
BT	Bluetooth, langaton verkkoteknologia
e-Health-Kit	Libeliumin toteuttama sensorisarja terveyden mittaamiseen
EKG	Elektrokardiogrammi eli sydänsähkökäyrä
DDoS	Distributed Denial-of-Service. Kyberturvahyökkäys. joka käyttää useampaa kuin yhtä IP-osoitetta
GDPR	General Data Protection Regulation, EU:n uusi tietosuojasetus, joka astuu voimaan 25.05.2018
IoT	Internet of Things, Esineiden tai asioiden internet
IP-osoite	Internet Protocol, tietokoneen osoite internet-verkossa
IT	Informaatioteknologia eli tietotekniikka
LPWA	Low Power Wide Area, vähän virtaa kuluttava pitkän kantaman IoT-verkko
Mock-Up	Malli toteutettavasta ratkaisusta
REST	Representational State Transfer, arkkitehtuurimalli web-pohjaisten ohjelmarajapintojen toteuttamiseksi
Sensori	sama kuin anturi, mittaa fyysikaalista tai kemiallista suuretta
SD	Secure Digital, muistikorttityyppi
SpO2	Peripheral capillary oxygen saturation, happisaturaatio
SSL	Secure Sockets Layer. Tietoverkkosalausprotokolla.
USB	Universal Serial Bus, sarjaväyläarkkitehtuuri oheislaitteiden liittämiseksi tietokoneeseen
VPN	Virtual Private Network, yksityinen salattu verkkoliittymä
WLAN	Wireless local area network, langaton lähiverkko

1 JOHDANTO

Hyvinvoinnin etämittaus on erilaisten etäteknologiaa hyödyntävien sovellusten ja laitteistojen tavoitteellista käyttöä asiakkaan terveydentilan seuraamiseksi. Tällaisia laitteistoja ovat mm. erilaiset sensorit, älypuhelimet ja tietokoneet. Etäkuntoutus on osa etämittausta ja sitä ohjaavat kuntoutuksen ammattilaiset. He seuraavat ja opastavat oikeanlaiseen kuntoutustapahtumaan tätä varten kehitettyjen ohjelmistojen tai applikaatioiden avulla. Etäkuntoutuksella on selkeä tavoite sekä alku ja loppu, kuten muullakin kuntoutuksella. (Salminen, Hiekkala & Stenberg 2016, 11)

Internet of Things (IoT) eli esineiden tai sensorien internet on ratkaisu, jolla haluttujen tapahtumien tiedot saadaan kerättyä talteen ja näitä tietoja analysoimalla kehitettyä uusia ratkaisuja. Kun sensorit liitetään internettiin, niin tiedon keräämisen lisäksi näiden toimintaa voidaan ohjata ja seurata.

Etäkuntoutuksen mittausta suunnitteleva opinnäytetyö tehtiin kolmen opiskelijan yhteishankkeena. Työn tilaajana oli Tampereen Ammattikorkeakoulun (TAMK) Hyvinvointiteknologian YAMK-koulutusosasto. Tutkimme aihetta etäkuntoutuksen ja hyvinvoinnin etäseurannan näkökulmasta. Tavoitteenamme oli kehittää uusia ratkaisuja etämittauksen opetuksen toteuttamiseksi ja kehittämiseksi.

Työn tavoitteena oli suunnitella hyvinvoinnin etämittaussisältöjä sekä toimintoja TAMK IoT-laboratorion ja tulevien opiskelijoiden käyttöön. Hyvinvoinnin etämittauksen ratkaisu palvelisi opiskelijoiden oppimisympäristönä ja testausympäristönä. Kehitystyön aikana esille tulleet ongelmat ja niiden ratkaisuideat tarjoaisivat uusia mahdollisuuksia toteuttaa opinnäytetöitä Hyvinvointiteknologian opetuksen kehittämiseksi.

Mika Helonsalo määrittä tietoa tuottavia sensoriratkaisuja ja tiedonsiirtoa lähimpään yhteydessä olevaan laitteeseen sekä IoT-alustalle. Käytettävissämme olevat sensorit olivat johdoilla yhteydessä tiedonsiirtoalustaan. Sensorien tarkoitus on tallentaa asiakkaan mitaustapahtuma. Sensoreista kerätty tieto siirretään internetin kautta palvelimelle, johon raakadata tallennetaan myöhempiä analyysyjä varten. Lisäksi hän toimi projektijohtajana edistäen kokonaisuuden suunnittelua, pysymistä aikataulussa ja järjestämällä tapaamisia yhteistyökumppaneiden kanssa.

Timo Salminen määritteli käyttöliittymän suunnitteluun liittyviä ratkaisuja sekä kehitti ohjelmiston, jolla tietoa siirretään asiakasta mittaavasta sensorista kuntoutusohjelmaan. Ohjelmiston kehitystyön tavoite oli mahdollistaa ja toteuttaa valituilta sensoreilta saadun tiedon siirtäminen IoT-alustalle.

Juho Kauppi perehtyi aiheisiin laitteiston ja ohjelmiston käytettävyys, integrointi sekä tietoturvaan ja tietosuojaan. Hän varmisti ohjelmiston, laitteiston ja seurannan toimimisen saumattomasti. Käyttökokemuksen pitää olla miellyttävä ja tuloksellinen.

Opinnäytetyömme aikana saimme luotua ensimmäisen sensoritiedon keräysratkaisun TAMK opiskelijoiden käyttöön, jolla voi seurata terveysdatan kertymistä ja tallentaa sitä jatkoanalysejä varten. Seuraavia opinnäytetöitä varten on kasattu myös tietoa applikaatioiden ja käyttöliittymän suunnittelusta, tiedonsiirtoratkaisun arkkitehtuurista ja integraatiosta sekä ehdotus uusista sensorihankinnoista.

2 OPINNÄYTETYÖN TAUSTA

2.1 Työn tavoite ja tarkoitus

Työn tavoitteena oli suunnitella hyvinvoinnin etämittaussisältöjä sekä toimintoja TAMKin IoT-laboratorion kautta opiskelijoiden käyttöön. Etämittausratkaisumme palvelevat opiskelijoiden oppimis- ja testausympäristönä. Suunnittelimme terveyden etämittauksen menetelmiä kotona kuntoutettavien käyttöön e-Health-Kit sensori-pakettia käyttäen. Lisäksi tutkimme tiedonsiirtoa, tiedon keräämistä sekä visualisointia, että käytettävyyttä hoidon analysoimiseksi. TAMKilla oli Texas Instruments yhteistyön tuloksena suunnitella IoT-laboratorio, jossa olisi heidän tuotteillaan varustettu välineistö käytettäväksi langattomien antureiden kanssa.

Tutkittavat ja analysoidtavat sensorit valikoituivat sen mukaan, mitä Tampereen ammattikorkeakoulu oli hankkinut keväällä 2016. Silloin oli ostettu Libeliumin e-Health Sensor Kit, jota on esitelty kuvassa 1 ja 2. Sensoripaketti sisälsi seuraavat mittarit tai anturit (libelium.com/top_50_iot_sensor_applications_ranking):

- pulssin ja veren happipitoisuuden mittausta (SpO2)
- verensokerin mittausta
- verenpainemittari
- potilaan asennon tunnistus (kiihtyvyydenmittaus)
- hengitysilman määrän mittausta
- antureita EKG-mittausta varten



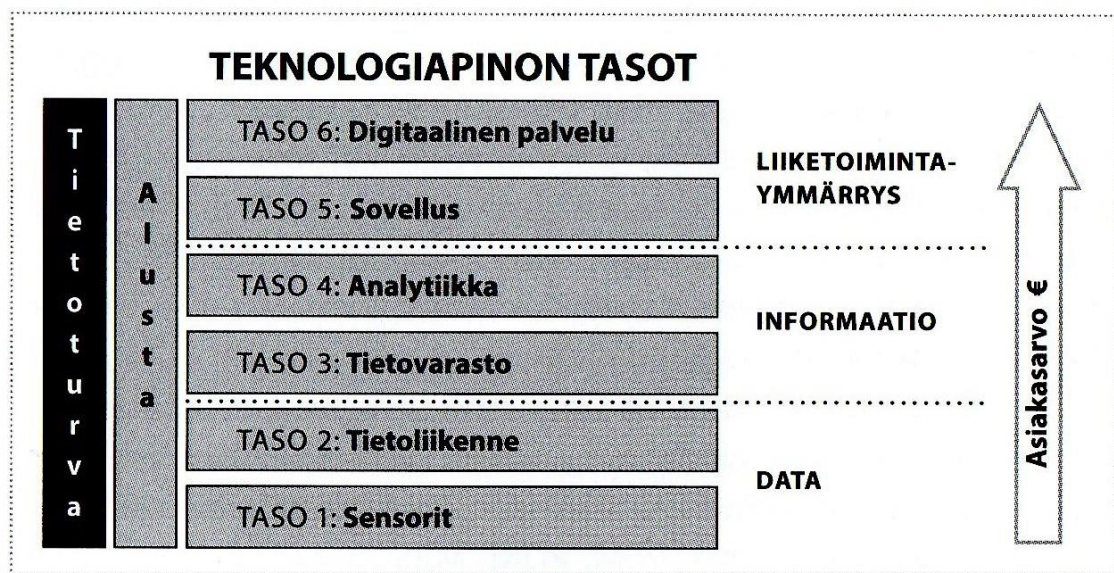
KUVA 1 ja KUVA 2 Libelium e-Health-Kit sensoripaketti (kuva: Mika Helonsalo 2016)

Tutkimme e-Health-Kit sensoreiden hyödyntämisen mahdollisuuksia hyvinvoinnin etämittauksessa. e-Health-Kit paketista oli myös uudemman version tilaus suunnitelmassa,

mutta sitä emme ehtineet saamaan käyttöömmee. Tutkimuksessa mukana olleista sensoreissa tutkimme niiden toimintaperiaatetta, käytännön laatua ja miten niitä voidaan käyttää.

Tavoitteenamme oli myös kehittää uusia etämittauksen ratkaisuja, joilla voidaan auttaa esim. ikäihmistä selviämään pidempään kotona. Toteutimme ensimmäisen version laitekokonaisuudesta, jolla voidaan harjoitustöiden yhteydessä kerätä terveydentilaan liittyvää tietoa. Samalla arvioimme saatujen mittaustulosten oikeellisuutta ja käyttökelpoisuutta.

Teollinen internet kirjassa esiteltiin Teknologiaapinon eri tasot uusien IoT-palvelujen kehittämisen pohjaksi ja tämä ajatusmalli on esitetty kuviossa 1. Tämän esityksen ja jo aiemmin itse päättelöllä tärkeiksi asioiksi poimittujen palasten avulla muodostui teoriaosuutemme. Päätimme jättää pääosin pois uusien digitaalisten palvelujen suunnitteluosuuden, koska työme tavoitteena oli kehittää alusta alkaen TAMK tarjoamaa oppimisympäristöä hyvinvointiteknologiassa. Alussa piti tehdä ratkaisut käytettävistä teknologioista ja alkaa kehittää uusia ratkaisuja niiden pohjalta.



KUVIO 1 Teknologiaapinon tasot, tarvittavat asiat suunniteltaessa Etämittauspalveluja (Collin & Saarelainen 2016, 143)

Tiedonsiirtomenetelmissä keskityttiin niihin, jotka vaikuttivat oleellisimmilta suunniteltaessa terveyden etämittauksen hyvinvointipalveluja. Terveydenhoidon tiedonsiirrossa on

tärkeää mm. turvallinen tapa siirtää tietoa. Analysoitavaksi päätyivät BT, WLAN, 5G sekä LoRa-verkko, jota mainostetaan hyvänä ratkaisuna IoT-ratkaisujen tarpeisiin.

IoT-alustan valinnassa pyrittiin ensisijaisesti huomioimaan hyvinvoinnin (terveyden) etämittaamisen tarpeet. Lopullisessa ratkaisussa otimme huomioon myös sen, että mikä tarjolla olevista vaihtoehdoista soveltuu laajasti eri TAMK koulutusalojen käyttöön.

Opinnäytetyön suunnitteluvaiheessa oli tarkoitus toteuttaa myös Mock-Up tarvittavasta käyttöliittymästä analysoida dataamme. Täten halusimme toteuttaa varsin laajan selvityksen nykyaikaisen käyttöliittymän suunnittelun periaatteista, sen luomasta käyttäjäkokemuksesta sekä testauksesta, että käyttöönotosta.

Potilastietoja käsiteltäessä on tärkeää järjestää tietoturva ja tietosuoja kuntoon eri osaluilla suunniteltavassa palvelussa. Miten potilasdata huomioidaan ja mitä tietoturvan toteutukseen vaaditaan. Vuonna 2018 on myös tulossa voimaan uusi henkilötietoja käsittelevä tietosuoja-asetus (GDPR), joten nyt oli erinomainen ajankohta paneutua myös tämän tuomiin uusiin vaatimuksiin sekä periaatteisiin.

Kokonaisarkkitehtuurin ja ohjelmistojen integraation suunnittelu liittyy oleellisena osana toteutettavan uuden ohjelmistopalvelun kokonaisuuteen. Toteutimme applikaatioideamme tässä työssä ideamäärittelyn tasolle, josta IT-osaston opiskelijat pääsevät sitä jatkokehittämään oikeaksi ohjelmistotuotteeksi.

2.2 Asiakas, kohderyhmä ja yhteistyökumppanit

Työn asiakkaana ovat Tampereen ammattikorkeakoulun hyvinvointiteknologian opiskelijat. Työn yhteistyökumppaneiksi voisi nimetä TAMK IoT-laboratorion, Hyvinvointiklinikan (hyvinvointiklinikka.tamk.fi) ja Wapicen (wapice.com), joka tarjosi meille IoT-alustansa käyttöömme. Sen kaupallinen nimitys on IoT-Ticket.

2.3 Tutkimuskysymykset

Tutkimuksen kannalta alussa oleellisia kysymyksiä olivat:

1. Mitä voimme mitata TAMKIn hankkimien Health-Kit sensorien avulla
2. Millaisen IoT-alustan valitsemme käyttöömmme
3. Saammeko kerätyn tiedon TAMKille tietopankiksi
4. Kuinka tiedonsiirto sensoreilta data-pankkiin toteutetaan
5. Kuinka teemme tiedon analysoinnin havainnolliseksi
6. Millainen Applikaatio tarvitaan käyttöliittymäksi sovellukseen
7. Mitkä ovat teknisen tietoturvan ja tietosuojan vaatimukset
8. Millainen on kokonaisarkkitehtuuri ja Integraatio
9. Millaisia palveluja voidaan luoda valittujen ratkaisujen avulla

2.4 Tutkimusmenetelmä

Opinnäytetyömme teossa meillä oli konstruktiiivinen tutkimusote. Meillä oli jo alussa tiedossa tutkimustyömme tavoite kehittää hyvinvointiin liittyvän sensoritiedon hankintaa ja sen siirtoa IoT-alustalle. Tutkimustyö toteutettiin läheisessä yhteistyössä tilaajien, yhteistyökumppaneiden ja projektitiimimme kanssa. Työmme tulokset perustuvat aiemmin suoritettuihin tutkimuksiin ja todelliseen käyttöön kehitettyihin ratkaisuihin. Teoreettisen tiedon hankinnassa käytimme hyväksemme kirjallisuustutkimusta, asiantuntijoiden haastatteluja, osallistuimme aiheeseen liittyviin seminaareihin, teimme yritysvierailuja ja etsimme artikkeleita sekä teknologian esittelyitä internetistä. Toteutimme ratkaisun hyödyntäen koululle hankittua tietotekniikkaa ja ohjelmistoja. Tiedonkeräyslaitteiston toimivuuden testasimme projektitiimin voimin ja muutimme sitä niin, että tulevatkin käyttäjät sitä helposti osaisivat käyttää.

3 HYVINVOINNIN ETÄMITTAUS

Mika Helonsalo

Etämittausta hyödyntämällä terveystalvveluja järjestävä taho pystyy seuraamaan asiakkaansa suorittamia mittauksia omasta toimipisteestään. Tämä mahdollistaa entistä useampien auttamisen samanaikaisesti olemassa olevalla henkilökunnalla. Tärkeimpänä syynä siirtyä etämittauksiin näyttää olevan tällä hetkellä syntyvä kustannussäästö. Asiakkaiden ei enää tarvitse matkustaa tutkimuksiin ja ammattilaisten aika vapautuu muihin tehtäviin. Erityisen paljon hyötyä etäpalveluista on syrjäseuduilla asuville ja liikuntarajoitteisille, kun mittaukset ja harjoitteet voi suorittaa esim. kotona. Tavoitteena on myös entistä tyytyväisemmät asiakkaat, varsinkin jos he saavat nopeasti palautetta edistymisestään ja kokevat haluavansa osallistua palvelun järjestämiseen. (Elisa 2016 & Etäkuntoutus, 93)

Mittauksissa syntyvä tieto siirtyy suoraan pilvipalveluun, jonne kertynyt raakadata tallennetaan. Hyvä palvelu (applikaatio) myös muistuttaa suorittamaan mittaukset ajallaan sekä kertoo mittaustapahtuman onnistumisesta. Raakadatan saaminen ei sinällään vielä riitä vaan se täytyy analysoida, jotta siitä on hyötyä potilaan tilan tulkitsemiseksi. Tapamme UKK-instituutin fysioterapeuttien ja alalla jo työskentelevien palveluntarjoajien (Nokia Health, PSHP, jne.) mukaan hoitavat lääkärit haluavat saada nopeita neuvoja potilaan tilan edistämisestä eli hyviä yhteenvetoja hoitotilanteesta. Uuden palvelun halutaan säästävän työaikaa, ei tuovan päällekkäisiä analyyssejä. Jotta raakadata saadaan hyödyttämään mahdollisimman paljon palveluntarjoajaa, niin tarvitaan hyviä algoritmeja tuottamaan hyödyllisiä analyyssejä.

Etäkuntoutus tai elintoimintojen mittaus voidaan jakaa seuranta-ajankohdan perusteella reaaliaikaiseen ja ajasta riippumattomaan. Reaaliaikaisessa etäkuntoutuksessa kuntoutuja ja terveystalvveluja järjestävä taho (esim. fysioterapeutti) ovat reaaliaikaisesti yhteydessä toisiinsa etäteknologian sovellusten avulla. Ajasta riippumattomassa etäkuntoutuksessa kuntoutujat suorittavat omatoimisesti hänelle määrätyt harjoitteet ja järjestävä taho sitten myöhemmin analysoi tulokset sekä antaa palautetta suorituksesta. (Salminen, Hiekkala & Stenberg 2016)

Reaaliaikaisessa etäkuntoutuksessa voidaan hyödyntää mm. videoneuvottelua, puhelinta tai tablettia. Ajasta riippumattomana etämenetelmänä voidaan käyttää esimerkiksi virtuaalista kuntoutusta tai verkkopohjaista sovellusta. Virtuaalinen kuntoutus soveltuu myös reaaliaikaiseksi menetelmäksi ja siihen voidaan lisätä tarkkaa tietoa kerääviä sensoreita. Etäteknologian avulla fysioterapeutti voi etukäteen määritellä terapian ajoituksen, keston, liikeratojen laajuuden ja suoritustehon. Teknologian omaksumiseen osaksi kuntoutusta vaikuttaa sen helppo käyttöönotto, saatavuus sekä varsinkin toimivuus etämittaustilanteissa. Koska ihmisten välinen suora vuorovaikutus puuttuu etäkuntoutusta toteutettaessa, niin on tärkeää luoda aluksi hyvä keskusteluyhteys ja luottamus kuntouttajan sekä kuntoutettavan välille. Uuden toimintatavan omaksumista käyttöön helpottaa, jos ensimmäinen käyntikerta on kasvokkain ja siinä käydään lävitse tavoitteita ja opastetaan teknologian käyttöön. (Vuononvirta 2011 & 2016)

3.1 Tiedonsiirtolaitteisto

Sensoreilta kerätty tieto täytyy toimittaa palvelimelle jatkokäyttöä varten. Tässä kappaleessa on esitelty tiedonsiirtoratkaisumme kannalta oleelliset komponentit.

3.1.1 Arduino

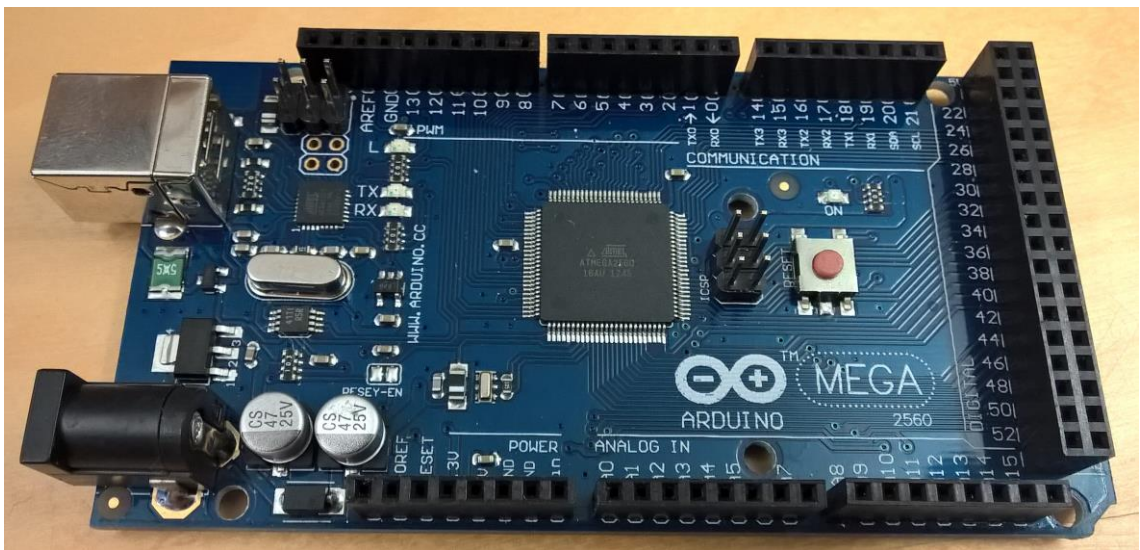
Mika Helonsalo

Arduinon mikrokontrolleri kehitettiin Pohjois-Italiassa 2005 suunnittelua opiskelevien edulliseksi ja helppokäyttöiseksi sekä vuorovaikutteisten harjoitustöiden kokeilujen alustaksi. Arduinosta on ollut monia kehitysversioita, kuten NG, Diecimila, Duemilanove ja Uno. Meidän käyttöömmme oli hankittu Megan 2560-versio, joka on esitelty kuvassa 3. Sen veroton hinta on tällä hetkellä noin 50€. Konsepti perustuu avoimen kehitysalustan käyttöön ja ohjelmointi on myös suunniteltu aloittelijoita ajatellen helpoksi. Internetistä löytyy ladattavaksi valmiita kehitysympäristöjä Win-, Mac- ja Linux-käyttöjärjestelmille ja ohjelmistoratkaisuja eri toimintojen toteuttamiseksi. Tiedonsiirto alustalle ohjelmiston syöttämiseksi ja kerätyn datan lähettämiseksi eteenpäin hoidetaan USB-kaapelin avulla. Se tarvitsee toimiakseen ulkoisen virtalähteen, joka voi olla USB-yhteys päätietokoneeseen. Koska kyseessä on edullisista komponenteista tehty mikrotietokone, niin siinä

on vain 8-bittinen ATmega mikroprosessori, 265kB Flash-muisti ja RAM muistia 8kB. Sensorien liittämiseksi löytyy 6 sarjaporttia sekä 54 pinninen GPIO (General Purpose I/O) väylä.

Arduinon ohjelmointi tapahtuu IDE-ohjelmalla (Integrated Development Environment). Ohjelmointikielenä on hieman muokattu C++, Wiring-sovelluslualustalta. Ohjelmiston syöttö tapahtuu USB-kaapelin kautta toiselta tietokoneelta. Ohjelma tallentuu Flash-muistille, jos vain se on kirjoitettu virheettömästi eli läpäissyt tarkistuksen ilman virhekoodeja. Tyypillisimmät virheilmoitukset liittyvät Pinnien sijaintiin, joten kytkentöjen paikat kannattaa tarkistaa piirilevyn numeroinnista. Koodi suorittaa tavallisesti kahta vaihetta. (McEwen & Cassimally, 101)

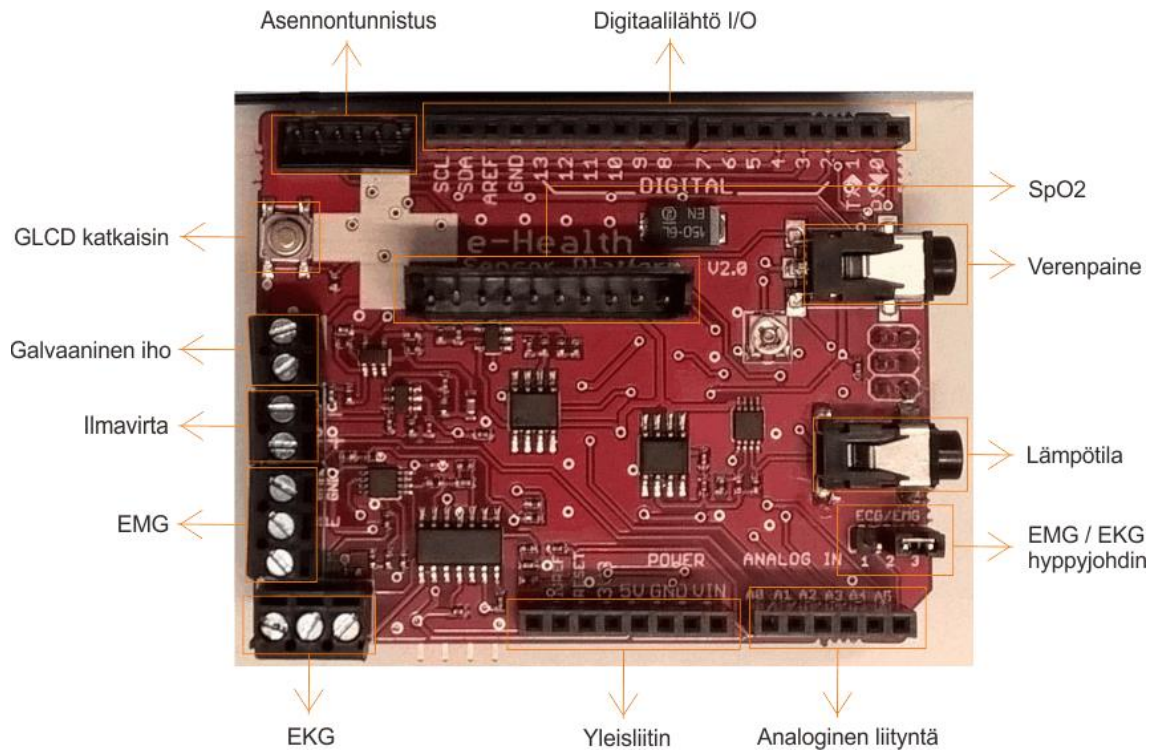
1. asennus (setup) ajetaan, kun Arduino käynnistyy. Voit käyttää tätä asentaaksesi I/O pinnit lähettämään tai vastaanottamaan tietoa.
2. silmukka (loop) suorittaa sekä toistaa ohjelmaa mm. datan keräämiseksi sensorilta



KUVA 3 Arduino Mega2560 sensori-alusta (kuva: Mika Helonsalo 2016)

Arduinon sensorialusta soveltuu hyvin analogisen datan vastaanottoon, kuten pulssin. Se ei kuitenkaan sisällä suoraan yhteyttä internetiin, jossa tieto liikkuu digitaalisena. Tätä varten on olemassa liitettävä moduuli, jolla Ethernet tai WLAN internet-yhteys voidaan toteuttaa. Meillä ei kuitenkaan ollut käytössämme tällaista lisäosaa, joten tiedonsiirto päätettiin toteuttaa USB-kaapelin kautta Raspberry Pi:lle, jossa se muutettiin myös digitaaliseen muotoon. Lisäksi huomasimme tarvitsevämme e-Health-Kit sensoreiden liityntöjen

toteutukseen oman e-Health Sensor Platform moduulin. Kuvasta 4 selviää käytettävissä olevat liitännärajoitukset.



KUVA 4 e-Health Sensor liityntä rajoitukset (kuva: Mika Helonsalo 2017)

3.1.2 Raspberry Pi

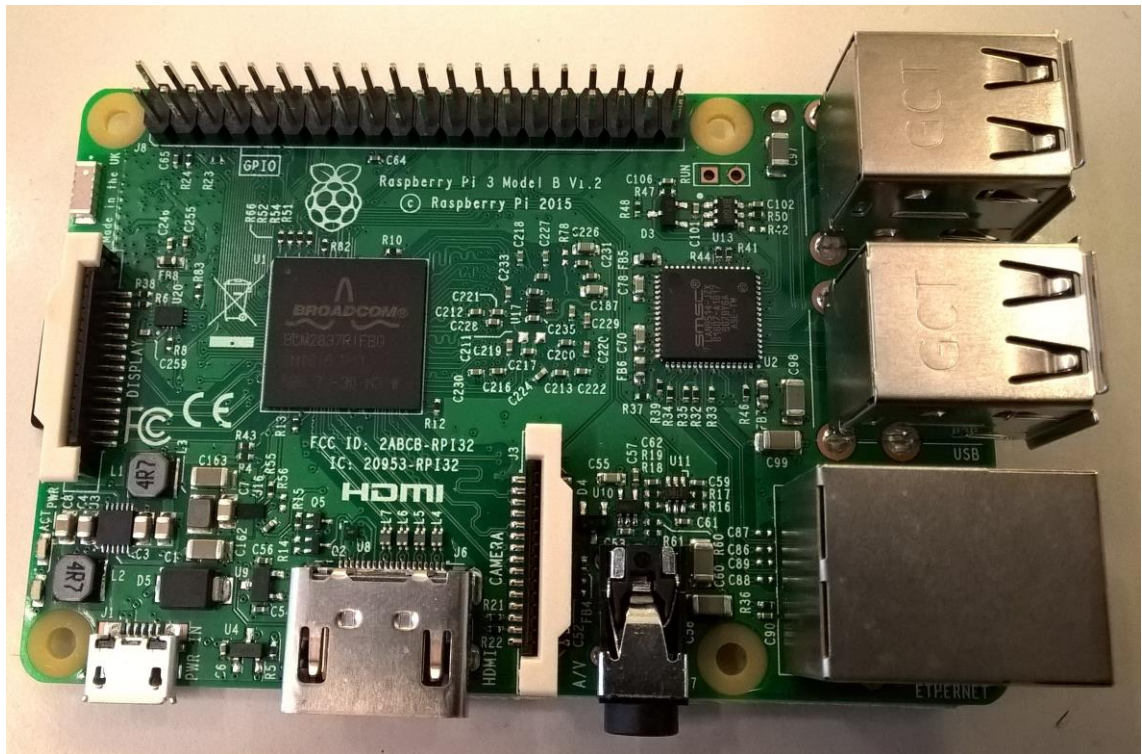
Timo Salminen

Raspberry Pi kehitys aloitettiin Britanniassa 2006 opetuskäyttöön. Tarkoitus oli luoda tietokone, joka on riittävän pieni ja edullinen ohjelmoinnin harjoitteluun ja kokemuksen hankkimiseen. Raspberry Pi Foundation on perustettu hoitamaan tämän mikrokontrollerin kehitystä (raspberrypi.org).

Saimme käyttöömmme Raspberry Pi 3 B-mallin mikrokontrollerin, joka on esitelty kuvassa 5. Raspberryn avulla saamme muutettua analogisen signaalin digitaaliseksi ja lähetykselpoiseksi IoT-alustaan. Veroton hinta kaupassa on tällä hetkellä noin 45 euroa. Raspberry Pi on varsin yleiskäyttöinen mikrokontrolleri verrattuna Arduinoon mm. sen monipuolisten liitännöiden ansiosta ja moninkertaisen tehokkuuden ansiosta. Siihen voi ladata ihan oikean käyttöjärjestelmän, jolla voi mennä internettiin (sisäänrakennettu Ethernet) ja käyttää hiirtä, näppäimistöä sekä näyttöä. RAM muistia versiossamme on 512Mb sekä Broadcomin kaksi-ytiminen prosessori. Virran syöttö tapahtuu suoraan [USB](#)-kaapelin kautta, mutta muun kuin kannettava tietokoneen käyttämistä virtalähteenä suositellaan.

Käyttöohjelma tallennetaan SD (Secure Digital) muistikortille, jota käytetään muistikorttipaikan kautta. Ohjelmointikieleksi Raspberry Pi:n säätio suosittelen käyttämään Pythonia ja käyttöjärjestelmäksi Raspbiania, mutta muitakin vaihtoehtoja voi käyttää. Raspberry Pi:lle on saatavilla monta eri vaihtoehtoa graafiselle käyttöliittymälle. Näitä graafisia käyttöliittymiä ovat mm. Rasbian ja Noobs, joita Raspberry Pi säätio pitää yllä. Näitä kahta graafista käyttöliittymää voidaan pitää helppokäyttöisinä, sekä sellaisina käyttöliittyminä ja hyvin tuettuina. Muita käyttöliittymiä Raspberry Pi:lle on mm: Ubuntu Mate, Snappy Ubuntu Core, Windows 10 IOT Core, OSMC, Librelec, Pinet, Risc OS. Mielenkiintoinen valinta käyttöliittymäksi olisi ollut Microsoftin Windows 10 IOT, joka pitää sisällään kehitysympäristön pienemmille laitteille, joissa on näyttö tai ei näyttöä lainkaan. Windows IOT on Windows 10 käyttöjärjestelmästä räätälöity versio Raspberry Pi kaltaisia mikrokontrollereita. Windows 10 IOT on ehdottomasti tutustumisen arvoinen ja tutkimisen arvoinen käyttöliittymä, koska kehitysympäristössä on helposti käytettävissä muut työkalut kuten Visual Studio, koodin ajaminen etänä ja jopa koodin pysäyttäminen etänä on mahdollista (Windows 10 Iot Core, 21.2.2017). Työssä on keskitytty Noobs käyttöjärjestelmän käyttämiseen.

Mikrokontrollerin vakio IP-osoite on raspberrypi.local, jolloin sen löytäminen helpottuu. Windows SSH-client ohjelmana toimii hyvin Putty, jota mekin käytimme ensimmäisen tiedonsiirtohavainnoinnin yhteydessä. Koska Raspberry Pi:ssä ei ole analogisia (ADC) sisääntulo-liittimiä (input), niin emme voi käyttää suoraan Health-Kit sensoreitamme. Tämän vuoksi välitämme sensoritietomme Arduinin kautta Raspberry:lle (McEwen & Cas-simally, 111)



KUVA 5 TAMK hankkima Raspberry Pi 3 Model B (kuva: Mika Helonsalo 2016)

3.2 Sensorit

Mika Helonsalo

Sensoreilla eli antureilla on sama ongelma kuin muullakin elektroniikalla eli käytettävyys heikkenee vanhenemisen seurauksena. Toisin sanottuna mittaustulokset eivät pysy samoina koko käyttöaikaa vaan tarkkuus pikkuhiljaa heikkenee. (Collin & Saarelainen 2016, 156)

Mittaustulos voi olla myös jo heti alussa väärä, jos sensoria ei ole kalibroitu oikein eli tarkistettu. Mittaustuloksen oikeellisuus varmistetaan vertaamalla tulosta oikeaksi todettuun referenssilaitteeseen. Antureiden tuotannossa on aina vaihtelua aiheuttavia tekijöitä, joten vain tarkistamalla mittaustulos voidaan osoittaa tuloksien käyttökelpoisuus. Terveysthuollon mittarit on kalibroitava kahden vuoden välein. Etämittausratkaisussa oleelliseksi kysymykseksi muodostuu, että voiko mittarin tai sensorin kalibroida etäyhteyden avulla. (Hyödynmaa 2016)

Mahdollisesti suurin vaikutus datan luotettavuuteen on kuitenkin käyttäjän suorittamalla mittauksilla. Tässä yhteydessä voidaan puhua mittausepävarmuudesta. Tähän liittyen on esitettävä ainakin seuraavat kysymykset:

- Suoritetaanko mittaus oikein ja ohjeiden mukaisesti?
- Toimiiko esim. verenpainemittari oikein mittauksen yhteydessä?
- Onko mittausolosuhteet tai mitattava sellaisia, että mittaustulosta voidaan pitää luotettavina?

Continuan (continuaalliance.org/products/product-showcase) sivustolla voi suorittaa hakuja sertifioituista terveysalan tuotteista, mm. laitetypin, valmistajan ja laitteen tarjoaman ulkoisen rajapinnan mukaan. Tällä hakukoneella löytyvät vain Continua-sertifioidut laitteet.

3.2.1 SpO2 eli happisaturaation ja sykkeen mittaus

Sormipulssioksimetri (SpO2) soveltuu happisaturaation ja sykkeen mittaamiseen. Käytössämme ollut e-Health-Kit paketin SpO2-mittari on esitelty kuvassa 6 ja 7. Reaaliaikainen happisaturaation mittaus on potilasseurannassa yleinen rutiinimenetelmä ja erityisen tärkeää happihoitoa saavien potilaiden sekä hengitystie- ja sydänvaivoista kärsivien henkilöiden seurannassa. Pulssioksimetrejä käytetään myös urheilussa harjoittelun keston kontrollointiin. Laite lähettää valosäteitä kahdella taajuudella: 660nm (punainen) ja 940nm (infrapunainen), joiden absorptioon perusteella laite määrittää sekä hemoglobiinin että hemoglobiinin hapen arvot.

Veren Happisaturaatio siis määritetään hapettoman ja hapellisen hemoglobiinin suhteesta. Happisaturaation arvo kertoo hapen riittävydestä. Punaisten verisolujen mukana kulkee happea kudoksiin ja tästä veri saa myös värinsä. Hemoglobiinia on kahdenlaista; Ensimmäistä johon on kiinnittynyt happea, kutsutaan oksi-hemoglobiiniksi (HbO2) ja toista de-oksi-hemoglobiiniksi (Hb). Hb absorboi itseensä enemmän näkyvää punaista valoa kuin HbO2 ja HbO2 taas absorboi enemmän infrapunavaloa. Veren happisaturaatio (SpO2) tarkoittaa oksi-hemoglobiinin ja alentuneen happitason hemoglobiinin suhdetta. Se voidaan laskea kaavasta $SpO2 = HbO2 / (Hb + HbO2)$. (Elektroniikkalehti 2015)



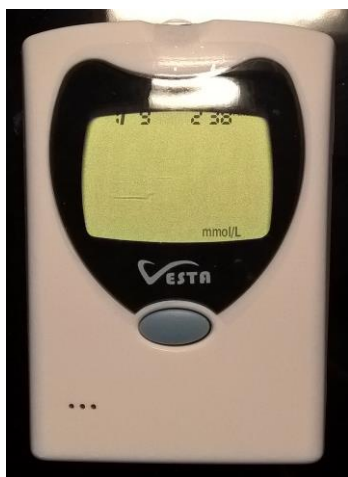
KUVA 6 ja KUVA 7 Sormipulssioksimetri, SpO2 (kuva: Mika Helonsalo 2016)

Sormipulssioksimetrin tyypillisiä tarkkuusarvoja mitattaessa aikuisten SpO2:ta on:

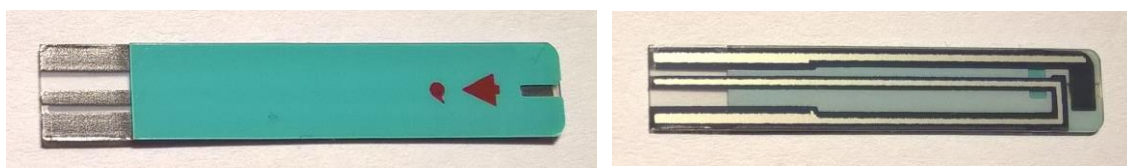
- Happisaturaatio
 - o resoluutio eli lukeman tarkkuus 1% mittausvälillä 35 – 99%
 - o mittausepävarmuus välillä 70 – 90% on +/-2% ja sitä alle ei määritelty
- Pulssi
 - o resoluutio mittausvälillä 30-240 on 1 lyönti/ minuutti
 - o mittausepävarmuus on +/-2 lyönti/ minuutti

3.2.2 Verensokeri

Libelium e-Health-Kit paketin mukana saapui Vesta-verensokerimittari, joka on esitelty kuvassa 8. Mittarimme mittaisi verensokerin määrän suoraan verinäytteestä, joten sen analysointia varten paketissa oli myös mukana kuvan 9 kaltaisia testiliuskoja.



KUVA 8. e-Health-Kit Vesta-verensokerimittari. (kuva: Mika Helonsalo 2017)



KUVA 9 e-Health-Kit mukana tullut testiliuska (kuva: Mika Helonsalo 2016)

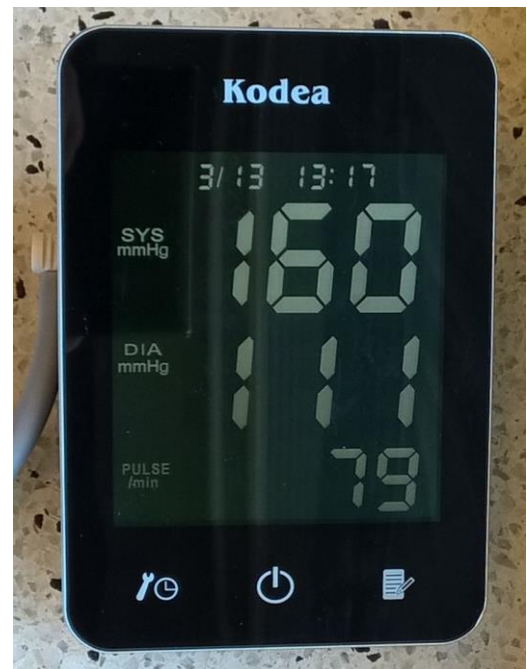
Mittarin mittaustulosten oikeellisuus täytyy määräjain tarkistaa. Suomessa myynnissä olevien verensokerimittarien lukemat saavat vaihdella noin +/- 15-20%, vaikka tavoite-tarkkuutena on +/-10%. Jos mittarin näyttämä on 10mmol/L, niin todellinen tulos on vä-lillä 9-11mmol/L. (Vesterlund 2012)

ISO15197-standardi asettaa mittausten tarkkuusvaatimukset:

- a) alle 4,2mmol/L hyväksyntäraja on enimmillään $\pm 0,83$ mmol/L
- b) yli 4,2mmol/L hyväksyntäraja on enimmillään $\pm 20\%$

3.2.3 Verenpaine

TAMK hankkima Kodea KD-202F malli mittaa verenpaineen paineenvaihtelun perusteella. Jos sydämenlyönti on heikko tai epäsäännöllisen, niin voi olla vaikeuksia saada tehtyä luotettava mittaus. Koemittauksia tehtäessä painemansetti kiristyikin epämukavan kireälle, jolloin teki mieli keskeyttää mittaustapahtuma. Käytössämme ollut painemansetti on esitelty kuvassa 10 ja itse mittari kuvassa 11.



KUVA 10 ja 11. Painemansetti ja Kodea-verenpainemittari (kuvat: Mika Helonsalo 2017)

Automaattinen verenpainemittari osaa itse määrittää systolisen (yläpaine) ja diastolisen (alapaine) paineen. Kotimittari tulisi kalibroida joka toinen vuosi. (tohtori.fi)

Verenpainemittarimme tarkkuusarvoja mitattaessa:

- Paineen mittausalue 30 – 280 mmHg
 - resoluutio on 1 mmHg
 - mittaustarkkuus on +/-3 mmHg
- Pulssi mittausväli on 40 – 199 lyönti/ minuutti
 - resoluutio on 1 lyönti/ minuutti
 - mittaustarkkuus +/-5 %

3.2.4 Kaatumisanturi

Potilaan kaatumisen ilmaisin perustuu potilaan asennon tuntemiseen ja sen muutoksen mittaamiseen kiihtyvyyssanturilla. TAMK:n käyttöön saatu kaatumisanturi on kuvassa 12. (Libelium 2016)

Tarvittava data syntyy kiihtyvyyssanturin mittaustuloksesta, liikkeen nopeus kolmen akselin sijaintitietona. Pystyakselilla tapahtuva mittaus suhteuttaa liikkeen Maan keskipisteeseen suuntautuvaan vetovoimaan. Mittaustietoa analysoimalla määritetään potilaan asento ja sen tilanteen muutos. (Collin & Saarelainen 2016, 157)

Oleellisinta on löytää se tieto tai poikkeama tuloksissa joka määrittää, että potilas on kaatunut. Tämä tärkeä tieto voi olla, vaikka hidastuvuus, joka syntyy potilaan törmätessä lattiaan (maahan).



KUVA 12. e-Health-Kit sensori kaatumisen todentamiseen (kuva: Mika Helonsalo 2016)

3.2.5 Hengitysilman määrä

Ilmavirtasensorin avulla voidaan seurata potilaan hengityksen ilmavirran määrää, esim. silloin kun tarvitaan hengitysapua. e-Health-Kit paketissa ollut sensori on tarkoitettu nenän kautta tapahtuva hengityksen seurantaan ja tämä on esitelty kuvassa 13.



KUVA 13. Hengitysilman määrä mittaus (kuva: Mika Helonsalo 2016)

3.3 Langattomat tietoverkot

Mika Helonsalo

Sensorien keräämä tieto eli data täytyy lähettää IoT-alustalle, jotta sitä voidaan seurata ja analysoida. Tässä luvussa tarkastellaan muutamaa langatonta tiedonsiirtotapaa, jotka ovat tärkeimpiä lähitulevaisuudessa terveysdataa kerättäessä. Käsittelemme tarpeellisia tietoverkkoja tiedon alkulähteestä eteenpäin, jolloin sensorista lähtien tarvittavat langattomat verkot ovat Bluetooth, WLAN ja 5G. Lisäyksenä alkuperäiseen suunnitelmaan tuli tarkasteltavaksi LoRa-verkko, koska myös Tampereella ollaan sellaista toteuttamassa Smart-City palvelujen tuottamiseksi. Nämä verkot voidaan jakaa paikallisiin lan-verkkoihin, joita ovat Bluetooth ja WLAN. Pitkän kantaman wan-verkkoja edustaa 5G ja LoRa. Langattomien verkkojen ratkaisut kuluttavat yleensä enemmän virtaa kuin langallisten, joten akkujen käyttöaika on sensorien käytön kannalta suurin ongelma. Esineiden internetin ratkaisujen yleistyessä tavoite on kehittää verkkoteknologioita enemmän tämän käyttötarpeen ratkaisuja palvelemaan. Potilastietojen siirtämisessä on tärkeää myös valita turvallinen tapa siirtää tietoa. Tietoturvaratkaisuja käsitellään hieman tarkemmin kunkin ratkaisun yhteydessä ja myöhemmin kappaleissa 3.6 ja 4.2.3. (Collin & Saarelainen 2016)

3.3.1 Bluetooth

Bluetooth Low Energy (BT LE tai BT 4.0) on lyhyen kantaman langaton verkko, joka soveltuu yhteyden muodostamiseen sensorista päätelaitteeseen eli esim. älypuheliimeen. Verkon kantama on yleensä enintään muutaman 10 metriä ja taajuutena 2,4 GHz. Saa- vutettava datan siirtonopeus on noin luokkaa 1 Mb/s. Tällä hetkellä on yleisimmin käytössä 2014 julkaistu versio 4.2. (Collin & Saarelainen, 171-173)

Kuten nimikin antaa ymmärtää, niin versio BT 4.0 on vähän virtaa kuluttava versio Bluetoothista. Se on toteutettu kolikon kokoisella BLE-piirillä, jotka kuluttavat virtaa vain mikroampeereja. Nappiparisto siis riittää helposti kahden vuoden käyttöön. BT paras ominaisuus on yhteensopivuus erilaisten mobiililaitteiden ja käyttöjärjestelmien kanssa sekä sovelluskehityksen helppous. Bluetoothin vahvuuksiin sisältyy tietoturva, kun siinä voidaan käyttää tietoliikenteessä 128-bittistä AES-salausta. Eri laitteet yhdistetään toisiinsa hyväksymällä yhteydenottopyyntö sekä todentamalla käyttäjän oikeutus samalla ja

valitulla avainkoodilla. BT yhteyttä käytettäessä suurin turvariski on itse laitteen käyttäjä, jos hän hyväksyy yhteyspyyntöjä tuntemattomista laitteista. (McEven, 98 & Hannula, 12)

Tämän hetkinen heikkous Bluetooth teknologiassa on, että tietoliikenne kulkee sensoreilta vain yhteen vastaanottavaan laitteeseen. BT5 versioon on tulossa mukaan Mesh-ominaisuus, jolloin yksittäiset BT-piirit voidaan liittää viestimään myös toistensa kanssa. BT Smart Mesh standardia ei ole vielä vahvistettu. (Collin & Saarelainen, 172) BT5 määrittely valmistui 2016 ja sitä odotetaan ensimmäisiin laitteisiin 2017 aikana. Tämä versio mahdollistaa 2Mb/s tiedonsiirtonopeuden ja kantomatka nelinkertaistuu 4.0 versioon nähden. Virrankulutusta on myös saatu pienennettyä ja sitä on mahdollista alentaa vielä lisää tiedonsiirron nopeutta laskemalla. (Hannula 2016) Versiossa 5 on myös parannettu paikantiedon määrittelyä BT-majakoiden avulla esim. sisätiloissa. Näillä majakoilla voidaan myös jakaa viestejä lähellä oleville toisille Bluetooth laitteille. (Teknavi 2016)

Bluetoothin kehityksestä vastaa Bluetooth SIG, joka on alan toimijoista koostuva yhteistyöelin. Bluetooth, Specification adopted documents, bluetooth.com/specifications/adopted-specifications.

3.3.2 WLAN

Kun sensorin keräämä data (tieto) on siirretty esim. taulutietokoneelle tai älypuhelimelle, niin sieltä se täytyy välittää internet kautta yhdistetylle tallentimelle. Sisätiloissa tähän soveltuu hyvin WLAN-verkko, jonka nopeus on useita 10Mb/s. Verkon kantama on yleensä enintään 100m. Sensori voi olla myös kytkettynä suoraan WLAN verkkoon, mutta tätä ratkaisua on käytetty vähemmän kuin Bluetoothia sen suuremman virrankulutuksen takia. (Collin & Saarelainen, 165 ja 171)

WLAN on lyhenne sanoista Wireless Local Area Network eli langaton lähiverkko. Sen pohjaltehdyissä tuotteissa käytetään usein kaupallista nimitystä Wi-Fi. WLAN on vaihtoehto kaapeleilla toteutetulle lähiverkolle paikoissa, joissa kaapelointi on vaikeaa tai mahdotonta. IEEE-järjestön standardointi ryhmässä 802.11 on tehty eri WLAN-standardeja; 802.11a, 802.11b, 802.11c, 802.11g sekä uusimpana 802.11n. Eri standardin sukupolvilla on erilainen soveltuvuus IoT-ratkaisuihin. Tällä hetkellä yleisesti käytetyssä 802.11ac standardissa ei ole otettu huomioon pienen virrankulutuksen tarvetta ja signaali

heikkenee nopeasti rakennusten seinien läpi kulkiessaan. Version 802.11g radiorajapinnan maksimisiirtonopeus on 54 Mb/s.

Uusin 802.11ah-standardi (WiFi HaLow) on kehitetty IoT-käyttöä varten. Sen vahvuuksia on vähäinen virrankulutus, rakenteiden läpäisykyky, jopa kilometrin kantama ja mahdollisuus liittää suuri määrä laitteita samaan tukiasemaan. Protokolla on kehitetty tukemaan laitteiden välistä viestintää ja suurin tiedonsiirtonopeus jopa 300 Mb/s. Se käyttää alle 1 G taajuusalueita, kun aiemmat ovat toimineet joko 2,4 GHz tai 5 GHz taajuuksilla. Signaalin kantama kasvaa ja seinän läpäisykyky paranee, kun taajuus laskee. 802.11ah-standardi julkaistiin vuonna 2016, mutta tuotteiden sertifiointi tavoite on aloittaa vasta 2018. (Collin & Saarelainen, 165 & 171)

Suojaamattoman avoimen verkon kautta luotu yhteys on turvallinen, kun tietokoneen ja palvelun välillä käytetään SSL-suojausta. SSL-suojatut verkkosivut tunnistaa HTTPS-protokollasta. Myös VPN-yhteydet ovat salattuja.

3.3.3 5 G

5G on nimitys seuraavalle mobiiliverkkoteknologian sukupolvelle, joka tulee käyttöön nykyisen 4 G jälkeen noin 2020. Verkon standardisointi on vielä kesken ja ensimmäinen vaihe siitä valmistuu 2018. Toisen standardisointivaiheen on määrä valmistua 2020. Ensimmäinen 5G-yhteys tapahtui Oulussa laboratorioympäristössä 23.joulukuuta.2016. Oulussa otetaan käyttöön myös ensimmäinen 5G testiverkko, mutta laajemmin se tulee käyttöön vasta 2020-luvulla. Ensimmäinen kaupallinen verkko on tarkoitus avata jo 2018 talviolympialaisia varten Etelä-Korean Pyeongchangissa, vaikka standardisointi on vielä kesken. Ensimmäiset päätelaiteet ovat varsin kookkaita mobiililaajakaistalaitteita. (Teknavi 2017)

Tämän kehitysversion tavoitteena on mahdollistaa jopa 100 kertainen tiedonsiirto aikaisempaa verrattuna. Toinen keskeinen 5G tavoite on pienempi virrankulutus, jolloin langattomat sensorit toimisivat nykyiselläkin akkuteknologialla pidempään. Siirrettävän tiedon tietoturva paranee, kun tietopakettien siirtoaika eli latenssi saadaan lyhennettyä 4G/LTE verkon nykyisestä 10 ms noin 1 ms pituiseksi (Collin & Saarelainen, 177). Tietoliikennetekniikassa puhutaan latenssista, joka tarkoittaa sitä aikaa jonka kestää tietopakettien siirtyä lähettäjältä vastaanottajalle ja takaisin. Lähiverkoissa tyypillinen latenssin

on jo alle 1 ms ja ADSL-yhteydellä toteutetussa internetpalvelussa 8-16 ms. Lyhyt vasteaika mahdollistaa reaaliaikaisen ohjauksen tai säädön laitteille sekä mahdollistaa suuremman määrän laitteita yhtä aikaa verkossa. Huonona puolena 5G:ssä on, että tiedonsiirtomatka lyhenee, kun taajuuksia nostetaan. Tällä hetkellä on suunnitelmassa käyttää taajuusalueina 28 ja 35 GHz, mutta myös matalammat taajuudet kuten 3,5 GHz ja 700 MHz voivat olla myöhemmin mahdollisia. Lisäksi rakennukset heikentävät nopeasti signaalin voimakkuutta, jolloin tukiasemia täytyy sijoittaa entistä tiheämpään. (Latva-aho 2015)

3.3.4 LoRa

Suomeen LoRa-verkkoa rupesivat rakentamaan yhdessä Espotel (nykyinen Etteplan Oy) ja Digita vuonna 2015 ja nykyisin se kattaa joko koko Suomen. Teknologian verkolle on toimittanut Actility (actility.com). Teknologiaa kehitetään LoRa Alliance-järjestö alla (lora-alliance.org) ja se on standardipohjainen esineiden Internet. (Elektroniikkalehti 2015)

Digitan rakentama verkko on teknologialtaan LPWA (Low Power Wide Area). Tähän kuuluu kolme eri radiostandardia, jotka ovat LoRa, Sigfox ja Weightless. Tässä osiossa käsittelemme tarkemmin ainoastaan LoRa-verkkoa, koska sellaista on Wapice rakentamassa Tampereelle. Puhekielen nimitys tälle verkolle tulee siinä hyödynnettävästä teknologiasta LoRaWAN (Long Range Wide Area Network). Tämä on ensimmäinen esineiden internetiä varten kehitetty verkkoteknologia maailmassa. Käytettävä teknologia on muuten avoimesti saatavissa, paitsi radiopiiriksi käy ainoastaan Semtechin valmistama. LoRa verkko on pitkän kantaman tietoverkko, joka tähtää IoT-ratkaisujen käytön lisäämiseen ja kehittämiseen. Tietoliikenne on kaksisuuntaista, jolloin se mahdollistaa esim. ohjelmistojen päivityksen etänä. Verkon kantama ulkona on tukiasemasta 15-20 km ja lähetystaajuutena Euroopassa käytetään 868 MHz. Matalataajuinen signaali läpäisee myös melko hyvin seinärakenteita. Verkossa käytettävät langattomat sensorit ovat edullisia, pitkäikäisiä ja helppoja ottaa käyttöön. Niiden lähettimet kuluttavat minimaalisesti virtaa, jolloin akunkesto voi olla jopa 10 vuotta. Pitkästä akunkestosta seuraa myös, että niistä tulee käytännössä huoltovapaita. Tiedonsiirtonopeus LoRaWAN verkossa on välillä 0.3 - 50 kB/s ja soveltuu parhaiten sellaisiin ratkaisuihin, joissa tarvitsee lähettää vain muutamia viestejä tunnin aikana. Melko alhainen tiedonsiirtonopeus pienentää myös virrankulutusta. Ensimmäisinä käyttökohteina on ajateltu olevan Smart-city sovellukset, kuten

älykäs valaistus, säätiedot, olosuhteiden valvonta kiinteistöissä ja liikenneuhkien seuranta. (Wapice 2016, Digita 2016, Postscapes 2017 & TIVI 2016)

Kiinteistöjen sensoreita ajatellen Semtech (semtech.com) on kehittänyt Pico-solutyyppisen tukiaseman, jolla saadaan aikaiseksi pienikokoinen LoRa-verkko. Tällä ratkaisulla saadaan suurempi peittoainen verkko toimimaan paremmin sisätiloissa ja lisättyä jo olemassa olevan peittoaluetta. Myynti alkaa tammikuussa 2017. Alhaisen virrankulutuksen ansiosta tässä verkossa toimiva sensori toimii paristolla jopa 20 vuoden ajan. Antureiden virrankulutus on erittäin alhainen, mikä tarkoittaa tyypillisesti 5-15 vuoden paristonkestoa. (Elektroniikkalehti 2016)

Tietoturva perustuu moninkertaiseen salaukseen. Verkkoon pääsyyn tarvitaan 64-bittinen avain. Applikaatiolle tarvitaan myös oma 64-bittinen avain. Laitekohtaisesti tarvitaan 128-bittinen salausavain. (LoRa Alliance 2017)

LoRa-verkon suurin heikkous terveyden etäseurannan verkkona on sen melko pieni tiedonsiirron nopeus. Ikäihmisen kodin tapahtumien, kuten lämpötila, kosteus, hälytykset, seurantaan taas se voi olla täysin oikea ratkaisu.

3.3.5 IPv6

Internetiin kytkettävä laite tarvitsee yhteyden muodostamiseksi IP-osoitteen (Internet Protokolla). Verkkoprotokollan avulla verkon kautta tietopaketteja vaihtavat koneet saadaan ymmärtämään toisiaan. Protokolla määrittelee siirrettävän tiedon laittamisen pakettiin, lähettäjän ja vastaanottajan yhteystietojen koodauksen sekä yksilöi tietopaketin. Vain samaa protokollaa käyttävät laitteet voivat keskustella keskenään verkon kautta. (Syrjänen 2010)

IPv6 protokolla on tulossa ja soveltuu paremmin Hyvinvointiklinikkamme datasiirtoon kuin nykyinen IPv4. Siinä voidaan suojautua aiempaa paremmin yhteyden väärinkäytöksiltä. Se huomioi paremmin myös paristotoimiset laitteet. Osoitteiden pituus kasvaa 128-bittiin ja samalla osoiteavaruus laajenee eli kaikille laitteille riittää oma osoite. Tähän saadaan yhdistettyä myös automaattinen asennuksen tuki verkon yli. (DNA 2016)

3.4 IoT-alustan valinta

Mika Helonsalo

Alustalla eri lähteistä saatu data kootaan yhteen. Alusta voi myös sisältää koko teknologisen infrastruktuurin sekä toiminnallisuuden. Alustan avulla sinulla on näkymä sensoriverkkoon ja niiden liittymisestä muuhun järjestelmään. (Collin & Saarelainen, 227)

Aluksi määrittelimme työryhmän kanssa IoT-alustat, joita voisimme mahdollisesti käyttää TAMK IoT-laboratoriossa. Keskusteluissa TAMK opettajien ja tapaamiemme asiantuntijoiden kanssa muodostui syyskuussa seuraava lista tutkittavista ehdokkaista. Alustan valintaan liittyvistä vaatimuksista kysyimme mielipidettä myös TAMK IT-osaston opettajilta. Listamme harkittavista vaihtoehdoista muodostui seuraavaksi:

1. Elisan IoT-alusta Thingworx (thingworx.com)
2. Wapice IoT-Ticket (wapice.com/fi/tuotteet/iot-ticket-internet-of-things-ratkaisut)
3. TAMKin osuuskunnan oma ratkaisu
4. Amazon AWS IoT (aws.amazon.com/iot)
5. Microsoft Azure IoT (azure.microsoft.com)
6. IBM Bluemix Watson (ibm.com/cloud-computing/bluemix/watson)

Kaikki tarjolla olevat pilvipalvelut toimisivat lähes yhtä hyvin Palvelumme kehityksen pohjana, joten eri vaihtoehtoja kannattaa vertailla. Teollinen internet kirjassa esiteltiin IoT-alustan valintaan liittyviä tärkeitä näkökantoja, jotka on eritelty kuviossa 2.

Hyvän alustan kahdeksan vaatimusta

3. Tietokanta	8. Ulkoiset rajapinnat	
	5. Analytiikka	7. Muut työkalut
	6. Datan visualisointi	
	4. Prosessointi ja toimintojen hallinta	
	2. Laitehallinta	
	1. Liitettävyys ja normalisointi	

KUVIO 2 Hyvän alustan 8 vaatimusta (iot-analytics.com 2016)

Hyvältä IoT-alustalta vaadittavat ominaisuudet näyttäisi voivan jakaa kahdeksaan eri huomioitavaan kohtaan. (Collin & Saarelainen, 230 & IoT-analytics 2016)

- 1) liitettävyys sensoriverkkoon
 - a. Sensorien pitää olla helposti yhdistettävissä tiedonsiirtoverkkoon ja rekisteröitävissä järjestelmään
- 2) Laitteiden hallinta ja sääntöjen muodostaminen
 - a. Alustan avulla on pystyttävä valvomaan, että päätelaite toimii oikein, tuottaa dataa odotetulla tavalla, ohjelmisto on ajan tasalla ja pystyssä
- 3) Tietokanta ja ns. pilvipalvelu
 - a. Kerätyn datan tallennus on oleellinen asia. Alustan on selvittävä suuresta määrästä ja eri laatuista dataa.
 - b. TAMK yksi oleellisista vaatimuksista oli, että tiedot pitää saada suoraan omaan käyttöön ja omistukseen.
- 4) Datan prosessointi ja yhdistäminen käyttökelpoiseksi kokonaisuudeksi
 - a. Datasta täytyy löytää oleelliset poikkeamat ja näiden perusteella havaita oleelliset muuttujat
- 5) Datan analysointi ja hallinta
 - a. raja-arvojen ylityksien ja poikkeamien täytyy laukaista hälytykset
- 6) Datan visualisointi ja raportointi
 - a. Jotta data tuottaisi uutta arvokasta tietoa, niin sitä täytyy käsitellä havainnollisempaan muotoon.
 - b. visualisointi voi olla havainnollisia mittaristoja ja graafisia yhteenvetoja datasta
- 7) Tietoturva ja tietosuoja, pääsynhallinta ja tunnistaminen
 - a. kenellä on pääsyoikeus kerättyyn dataan
 - b. käyttäjät täytyy tunnistaa ja käytön historia täytyy tallentaa
- 8) ulkoiset rajapinnat ja sovelluskehityksen työkalut
 - a. Rajapinnat ulkoisiin tietojärjestelmiin, jotta integroinnit toisiin järjestelmiin onnistuu.

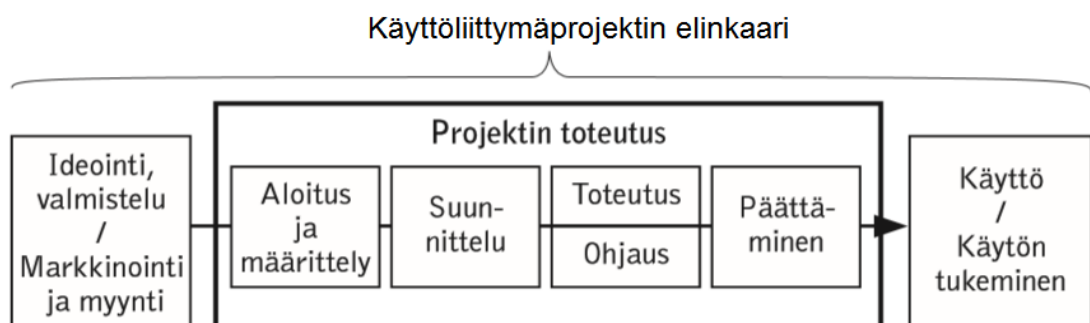
3.5 Käyttöliittymä – Kuntoutusohjelma

Timo Salminen

Käyttöliittymä on osa laitetta, ohjelmistoa, jonka avulla käyttäjä käyttää laitetta tai tuotetta. Käyttöliittymä on se osa, minkä käyttäjä näkee tietokoneen, matkapuhelimen tai tabletin – tietokoneen ruudulla. Käyttöliittymä nykyään tarkoittaa graafista käyttöliittymää. Käyttöliittymän oleellisena osana perustuu myös se, millä tavalla käyttöliittymää käytetään, tietokoneen näppäimistöä ja hiirtä, matkapuhelimen tai tabletti – tietokoneen kosketusnäyttöä käyttäen. Hyvä käyttöliittymä helpottaa tehtävän käsittelyä ja loppuun viemistä ilman, että käyttöliittymään tarvitsisi kiinnittää tarpeetonta huomiota. Hyvää käyttöliittymä on myös miellyttävä käyttää. (Käyttöliittymä: Käyttöliittymän kuvaus)

3.5.1 Käyttöliittymän suunnittelun vaiheita

Käyttöliittymän suunnittelussa on kyse käyttäjäkokemuksen sekä käytettävyyden maksimoimisesta. Käyttöliittymän suunnittelussa pyritään siihen, että käyttöliittymä on yksinkertainen ja mahdollisimman tehokas, jotka täyttävät käyttäjän vaatimukset. Kyseistä lähestymistapaa voidaan pitää myös käyttäjäkeskeisenä mallintamisena käyttöliittymän suunnittelussa. (Käyttäjäystävällisen sovelluksen suunnittelu, Wiio) Käyttöliittymän suunnitteluun tulee käyttää aikaa ja hyvän käyttöliittymän suunnitteluun voidaan käyttää myös tavallisen projektin elinkaaren esimerkkiä, kuviossa 3 kuvataan tavallisen projektin elinkaari.



KUVIO 3 Käyttöliittymäprojektin elinkaari (Artto, Martinsuo & Kujala 2006)

Projektin elinkaarta voidaan käyttää apuna käyttöliittymän suunnittelun vaiheisiin. Käyttöliittymän ideoinnissa tulee ottaa huomioon loppukäyttäjät. Loppukäyttäjää voi olla eri ammattiryhmistä, jolloin on tärkeää, että käyttöliittymä on helposti laajennettavissa, sekä kaikkien sidosryhmien tarpeet ja odotukset huomioidaan. Aloituksessa ja määrittelyssä tulisi ottaa huomioon mitä ongelmakohtia käyttöliittymän toteutuksessa voi ilmetä ja käyttöliittymän tavoitteet, jotka käyttöliittymän tulee toteuttaa. Suunnitteluvaiheessa käyttöliittymän toteutussuunnitelma voidaan laatia ja toteutukseen liittyvät resurssit, aikataulu sekä kustannukset ovat selvillä. Suunnitteluvaiheessa käyttöliittymään liittyvät vaatimukset ovat selkeytyneet ja voidaan edetä toteutusvaiheeseen. Toteutusvaihe tarkoittaa teknistä toteutusta, sekä koko projektiryhmän työskentelyä. Toteutusvaiheessa tulee siis ymmärtää koko käyttöliittymäprojektin toteutus. Ohjausvaihe on rinnakkainen toimenpide toteutusvaiheen kanssa, jossa seurataan käyttöliittymän etenemistä teknisiin spesifikaatioihin, aikatauluun sekä kustannuksiin. Ohjausvaiheessa voidaan puuttua poikkeamiin tavoitteisiin ja suunnitelmiin. Käyttöliittymän ohjausvaihe pitääkin sisällään monia iteraatioita teknisestä toteutuksesta testaukseen, raportoinnin työkaluin. Jos käyttöliittymän tekninen toteutus ei etene suunnitelmien mukaan, tilanne analysoidaan ja tehdään tarvittavat muutokset käyttöliittymän projektisuunnitelmaan. Päättäminen on erittäin tärkeä vaihe, jossa on saatu käyttöliittymä valmiiksi. Kaikki käyttäjäryhmät ovat hyväksyneet käyttöliittymän, joka on otettu käyttöön. Käyttöliittymästä ja sen toimivuudesta tulee kerätä palautetta, jonka perusteella voidaan arvioida, että miten hyvin käyttöliittymä toimii. (Artto, Martinsuo & Kujala 2006)

Yksi tärkeimmistä suunnitteluvaiheista on käyttöliittymän visuaalinen suunnittelu, joka on merkittävämpää kuin usein ajatellaan. Visuaalisella suunnittelulla taataan uusien käyttäjien osaaminen. Visuaalinen suunnittelu vaikuttaa myös suurelta osin käytön tehokkuuteen. Käyttöliittymän visuaalisessa suunnittelussa ei silti saa unohtaa käyttöliittymän tärkeintä tehtävää, joka on sen tarjoama sisältö sekä toimivuus, vaikka värien hallinta ja kokonaisuuden sommittelu onkin tärkeä osa visuaalista suunnittelua. (Sinkkonen, Kuoppala, Parkkinen & Vastamäki 2006, 155)

3.5.2 Käyttöliittymän elementtien asettelu

Käyttöliittymän elementtien asettelu on tärkeä osa käyttöliittymää. Tuotteen käyttöliittymän elementtien asetteluun tavoite on huolehtia niin hyvästä kommunikaatiosta käyttäjän ja tuotteen välillä kuin mahdollista, niin intuitiivista kuin tehokasta. Asetteluun on otettava huomioon käyttäjän tavoitteista ja suoritettavista tehtävistä. Hyviä keinoja elementtien asetteluun on typografia, värit, tiedon organisointi sekä taustan käyttö. Käyttöliittymän kehittäjällä asetteluun voi vaikuttaa myös asioiden hierarkkinen näyttäminen, siten, että asettelu vastaa elementtien takan olevien käsitteiden todellisia suhteita. (Sinkkonen, Kuoppala, Parkkinen, Vastamäki 2006, 155. Parviainen & Sorjanto 2012, 25)

Asetteluun tasapaino riippuu kontrastin ja tyhjän tilan käytöstä, elementtien muodosta ja sijoittelusta, liikkeestä ja väreistä. Hyvässä käyttöliittymässä kaikki olennainen tulee esille liioittelematta, mutta selkeästi.

3.5.3 Käyttöliittymän estetiikka

Käyttöliittymän esteettisen visuaalisen suunnittelun peruseriaatteen ovat samat kuin hyvän toimivuuden – selkeys, johdonmukaisuus, miellyttävä ulkonäkö ja yksinkertaisuus. Esteettisiin asioihin toki liittyy muitakin arvoja kuin toimivuus – visuaalinen identiteetti, ammattimaisuus, tasapaino, harmonia, jännitteet sekä arvaamattomuus, ovat estetiikkaan liittyviä asioita. Estetiikka voidaan jakaa kolmeen periaatteeseen, jolla ulkonäköä voidaan suunnitella:

- Yhtenäinen suunnittelu, siten että elementit ovat osa samaa visuaalista kokonaisuutta.
- Käyttöliittymän suunnittelussa otetaan huomioon se mihin, milloin ja miten käyttäjän huomio kiinnitetään.
- Käyttöliittymä soveltuu sen kanssa käytävään vuorovaikutukseen

Esteettisesti kauniisti suunniteltu käyttöliittymä on osa kokonaisuutta ja koko tuotteen miellyttävyyttä. Jokainen käyttäjä etsii kauneutta, mutta toisille esteettiset arvot eivät näytele niin suurta osaa kokonaisuudesta kuin toisille. Toiset voivat arvostaa taidokkaasti tehtyjä elementtejä, koristeluita, kun taas toiset arvostavat yksinkertaisuutta. Joku ajattelee enemmän kokonaisuutta, kun toinen haluaa nähdä pieniä yksityiskohtia. Onkin mielenkiintoista, että estetiikka muuttuu ajan saatossa ja kuinka kauneusarvot vaihtelevat eri kulttuurien välillä. Käyttöliittymää suunniteltaessa onkin otettava huomioon eri kulttuurit

ja eri ammattiryhmät ja se mitä kukin arvostaa. Tämä osa-alue voikin olla erittäin hankalasti toteutettavissa kaikille sopivaksi. (Sinkkonen, Kuoppala, Parkkinen & Vastamäki 2006, 156 - 157)

Käyttöliittymän esteettisenä tavoitteena on siis sanoman pukeminen tehokkaaseen, harmittuun graafiseen ulkoasuun ottaen huomioon aiheen, vastaanottajan, resurssit sekä käyttötilanteen. Suunnittelija on hyvä lähteä liikkeelle siitä, mitä elementtejä käyttöliittymään tarvitaan. Mikä on elementtien merkityksellinen järjestys, hierarkia, tärkeys, pituus, muoto ja miten sisältö otetaan huomioon järkevästi. (Sinkkonen, Kuoppala, Parkkinen & Vastamäki 2006, 157). Käyttöliittymän esteettisyydessä suunnittelijan on hyvä ottaa huomioon myös käyttöliittymän muotoon vaikuttavia tekijöitä, joita ovat: läheisyys, samankaltaisuus, jatkuvuus, päättäminen, kuvaavuus sekä yleinen uskomus. (Johnson 2010, 12).

Käyttöliittymää suunniteltaessa voidaan todeta, että eleganssi ja yksinkertaisuus saadaan aikaan huolehtimalla tuotteen yhtenäisestä suunnittelusta, viimeistelystä ja sopivuudesta tehtävään. Elementtien ja kokonaisuuksien suhteet ja kontrastit saadaan aikaan huolehtimalla, että lopputulos on harmoninen, selkeä ja siinä on tarpeeksi jännitettä, että käyttöliittymä ei ole tylsä. (Sinkkonen, Kuoppala, Parkkinen & Vastamäki 2006, 158 - 159)

Erilaiset kolmiulotteiset miljööt auttavat usein käyttäjää ymmärtämään, miten pitää toimia. Ihminen reagoi nopeammin ja paremmin kolmiulotteisiin kuvakkeisiin, mutta on hyvin vaativaa rakentaa hyvä kolmiulotteinen ympäristö, jossa kaikki ratkaisut toimisivat moitteettomasti. Mitä monimutkaisempi kokonaisuus on kyseessä, niin sen haasteellisempi ja sitä enemmän ammattitaitoa käyttöliittymän tekeminen vaatii. Suunnittelija voi itse käyttää apuna siristystestiä, jossa käyttöliittymän tasapainoisuutta voidaan varmentaa vetäytymällä hieman kauemmaksi liittymästä, siristetään silmiä niin, etteivät sommittelun yksityiskohdat näy ja tarkistetaan ovatko elementit tasapainoisesti käyttöliittymän ikkunassa. Toinen havainnollinen testi on x-testi, jossa katsotaan käyttöliittymän asetelua ja toiminnallisuutta. Käyttöliittymän tekstit korvataan x-kirjaimilla ja testataan ymmärtääkö käyttöliittymän merkitys ja toimita siinäkin tapauksessa. (Sinkkonen, Kuoppala, Parkkinen & Vastamäki 2006, 160 - 161)

3.5.4 Käyttöliittymän testaus ja testimenetelmän valinta

Seuraavissa kappaleissa käydään läpi erilaisia käyttöliittymän testaukseen liittyviä keskeisiä tekijöitä, sekä erilaisia testausmenetelmiä, joita voidaan käyttää hyväksi käyttöliittymän testauksessa.

Projektipäällikön, käyttöliittymän suunnittelijoiden ja muiden projektin jäsenten on käyttöliittymäprojektin edetessä ottaa huomioon käyttöliittymän testaus. Käyttöliittymän testaus tuleekin olla osa tuotekehitysprosessia. Käyttöliittymän tai tuotteen käytettävyyden testaaminen ja varmistaminen on yksi tärkeimmistä osa-alueista, jota ei voida unohtaa tai laiminlyödä käyttöliittymäprojektin edetessä. Keskeisimpiä menetelmiä ovat käyttäjien tehtävien, osaamisen ja toimintaympäristön selvittäminen ja dokumentointi sekä käytettävyyden arviointi. Käytettävyydestä on yksi parhaista tavoista testata käyttöliittymää, niin kuin mitä tahansa tuotetta. Käytettävyyden testaaminen on joustava menetelmä, josta saadaan helposti käyttöliittymästä monenlaista tietoa, sekä käytettävyydestä on ainoa objektiivinen mittaustapa, jolla tuotteen käytettävyys voidaan todeta. Käytettävyydestä tulee ottaa kuitenkin huomioon se, että käytettävyys ja käyttäjien mielipiteet tuotteesta ovat eri asia. Mielipiteitä kysyttäessä käyttöliittymästä tai tuotteen toimintatavasta saadaan mielipiteitä, niin paljon kuin on vastaajia, kun taas käytettävyydestä ilmenee helposti jo pienellä käyttäjämäärällä, missä tuotteen tai käyttöliittymän ongelmakohdat piilevät. (Sinkkonen, Kuoppala, Parkkinen & Vastamäki 2006, 275 - 280)

Käytettävyydestä perimmäinen tarkoitus on tehdä tuotteen käyttölaadusta parempi seuraamalla käyttäjää tilanteessa, joka muistuttaa aitoa tilannetta mahdollisimman paljon. Käytettävyydestä onkin hyvä pitää mukana jo käyttöliittymän tai tuotteen kehitysvaiheessa. Kehitysvaiheessa tehtävät käytettävyydestä ohjaavat ja palvelevat suunnittelijoita paremmiksi, kun testaus otetaan huomioon jo tässä vaiheessa. Käytettävyydestä voidaan jakaa kahteen eri tarkoitukseen: kehitystestien sekä hyväksymistestien. Kehitystestit tähtäävät mahdollisimman hyvään lopputulokseen. Hyväksymistestit voidaan vielä tarkemmin jakaa kahteen eri osaan: tuotehyväksyntään, joka tehdään käyttöliittymälle tai tuotteelle yrityksen sisällä ja lopulliseen hyväksyntään, jossa on mukana niin asiakas kuin itse oma yritys. Kaikissa tilanteissa käyttöliittymä tai tuote testataan ja todetaan toimivaksi. Käyttöliittymän testaus ja käytettävyydestä tulisivat viimeistään aloittaa siinä vaiheessa, kun ensimmäinen prototyyppi on valmistunut. (Sinkkonen, Kuoppala, Parkkinen & Vastamäki 2006, 276 - 280)

Testauksen suorittaminen käytettävyytestissä voidaan karkeasti jakaa kolmeen eri osaan:

1. Testin järjestäminen ja testaussuunnitelman laatiminen
2. Testin suorittaminen
3. Testin analysointi ja testiraportin laatiminen

Testauksessa kannattaa ottaa huomioon, sekä kvantitatiivinen, että kvalitatiivinen testaus. Kvantitatiivisella testauksella mitataan käyttöliittymän laatua käytettävyystavoitteisiin verraten. Kvalitatiivisella pyritään löytämään käyttöliittymästä niin monta käytettävyydeltään ongelmallista kohtaa kuin mahdollista. (Sinkkonen, Kuoppala, Parkkinen & Vastamäki 2006, 280 - 281)

Testisuunnitelman ja testauksen aloittamisessa tulee ottaa huomioon eri työvaiheita, joita ovat (Sinkkonen, Kuoppala, Parkkinen & Vastamäki 2006, 281):

- Testin tavoitteiden selvittäminen
- Käytettävyyksivaatimusten selvittäminen
- Käyttöliittymään ja tuotteeseen tutustuminen
- Testattavien toimintojen valinta
- Testikäyttäjien valinta ja niiden määrä
- Testitehtävien laatiminen
- Testausmenetelmän valinta
- Koulutustarpeen selvittäminen ennen testiä
- Apumateriaalin läsnäolo testauksessa

Testitavoitteiden määrittäminen on todella tärkeä ja aluksi onkin selvitettävä mitä testiltä itsessään halutaan. On hyvä kohdentaa testit oikealle käyttäjäryhmälle. Onko kyseessä kokenut käyttäjä vai onko käyttäjä kokematon. Käyttöliittymää testattaessa hyviä yleisiä huomioonotettavia tavoitteita ovat yleinen käytettävyys, sopivuus kaikille käyttäjille, niin kokeneelle kuin kokemattomalle käyttäjällekin, opittavuus ja virheensieto. Testaamisessa tulisi valita sopivat mittarit, joilla tavoitteita tullaan mittaamaan. Testeissä käytettäviä mittareita voivat olla muun muassa (Sinkkonen, Kuoppala, Parkkinen & Vastamäki 2006, 282 - 283):

- Testeihin kulutettu aika
- Virheiden määrä
- Virheistä toipuminen ohjelmistollisesti sekä niistä toipumiseen kulutettu aika
- Testikäyttäjän omat virheet
- Tarvitsiko käyttäjä apua
- Käyttäjän asenne käyttöliittymää testattaessa
- Annettavaa testiä ei pystytty suorittamaan lainkaan

Käyttäjien määrän valitseminen on yksi osa testausta. Testikäyttäjää pitää olla enemmän kuin yksi, yleensä kolmesta kuuteen henkilöä. Vakavimmatkin virheet käyttöliittymästä voidaan löytää jo kahdella testihenkilöllä, mutta varmemman lopputuloksen antaa suurempi testihenkilöiden määrä. Suurempi testihenkilöiden määrä lisää löydettyjen käytettävyysohjelmien määrää. Testihenkilöiden määrä ei kuitenkaan saa kasvaa kohtuuttoman suureksi, ettei testauksesta ja testien analysoinnista tule liian raskasta. Olisi siis löydettävä sellainen testaajien määrä ja laatu, jolla annetusta testitehtäväjoukosta löydetäisiin mahdollisimman paljon käyttöliittymässä olevista ongelmista.

Testikäyttäjiksi valitaan sellaisia tuotteen tai käyttöliittymän tulevia asiakkaita, jotka eivät ole olleet mukana tuotteen kehittämisessä. Testikäyttäjän tärkeä ominaisuus on rohkeus ilmaista itseään. Testikäyttäjän tulisi kuitenkin tuntee työ ja toiminta, johon käyttöliittymä on tarkoitettu. (Sinkkonen, Kuoppala, Parkkinen & Vastamäki 2006, 283 - 284)

Testitehtävien laadinta on erittäin tärkeä osa testausta, joka pitää huolellisesti suunnitella. Testattavat kohteet tulee listata, ja joissa katetaan koko käyttöliittymän testaus. Testitehtävissä tulee olla helppoja, niin kuin monimutkaisiakin tehtäviä, ottaen huomioon käyttöliittymän rajoitukset. Testitehtävät eivät saa koskaan käyttää suoraan tuotteesta näkyviä termejä, koska testikäyttäjät seuraavat testitehtävän termejä poikkeuksetta, koska ne vievät heidät joko oikeaan tai väärään suuntaa. Testikäyttäjä halutaankin ennemmin itse ymmärtää mitä kulloinkin tulee tehdä. (Sinkkonen, Kuoppala, Parkkinen & Vastamäki 2006, 285)

On erittäin tärkeää, että käyttöliittymän testaukseen valitaan siihen soveltuva testimenetelmä. Tässä kappaleessa keskitytäänkin tavanomaisimpiin käytettäviin testimenetelmiin. Kyseiset testimenetelmät ovat (Sinkkonen, Kuoppala, Parkkinen & Vastamäki 2006, 286):

- Ääneen ajattelu
- Paritestit
- Yhteisläpikäynti
- Jälkikäteen haastattelu
- Jälkeenpäin kommentointi
- Pikkutestit, osatesti, käsitelistat
- Ryhmäläpikäynti
- Vapaa läpikäynti

Ääneen ajattelussa testaajat toimivat yksi kerrallaan kertoen mitä ovat tekemässä ja miksi tekevät juuri tällä tavalla. Tämä on yksi käytetyimmistä ja käyttökelpoisimmista menetelmistä. Tässä tavassa ongelmana on se, että henkinen kuormitus kasvaa ja puhuminen voi olla hankalaa. Menetelmä vaatii hyvää ohjaajaa, joka osaa pitää testaustunnelman rentona, mutta ei vaikuta itse testituloksiin. Testaustilanne voidaan taltioida, jolloin syiden ja muistiinpanojen kirjaaminen on huomattavasti helpompaa.

Paritestissä järjestelmää tai käyttöliittymää käyttää kaksi testikäyttäjää yhtä aikaa ja keskustelevat tuotteesta keskenään. Menetelmä on huomattavasti samantyylinen, kuin ääneen ajattelu, mutta tässä tapauksessa käyttäjät keskustelevat keskenään. Paritestissä tulee valita testikäyttäjät huolellisesti siten, että molemmat ovat tasavertaisessa asemassa testattavaan tuotteeseen, jolloin molempien mielipiteet tulevat hyvin esille. Videointi on tavallinen tallennustapa tälle menetelmälle, josta on helppo jälkepäin tarkistaa syitä ja seurauksia. Käyttäjät voidaan myös haastatella paritestin jälkeen.

Yhteisläpikäynnissä ohjaaja ja testikäyttäjä etenevät testeissä keskustellen käyttöliittymästä tai tuotteesta. Ohjaajan tehtävänä on aktiivisesti selvittää testikäyttäjän ymmärrystä testattavasta kohteesta. Ohjaajan tehtäviin kuuluu enenemässä määrin kysellä testikäyttäjältä miksi käyttäjä toimi, kuten toimi kussakin tilanteessa. Erityistä tarkkaavaisuutta tulee kuitenkin käyttää testikäyttäjän keskeyttämisiin, koska tämä liiallinen keskeyttäminen voi häiritä käyttäjää huomattavan paljon, jolloin tilanteen luontevuus katoaa. Jos käyttöliittymässä tai tuotteessa on huomattavia puutteita, silloin yhteisläpikäyntimenetelmä on oivallinen valinta, koska silloin aktiivisella ohjaamisella ja osallistumisella voidaan kuitenkin saada tarvittava tieto tuotteen sen hetkisistä puutteista keskittymällä oleellisiin toimintoihin. Ohjaajan taitoa vaatii myös se, että hän ei johdattele testikäyttäjää. Videointi on suotavaa myös tässä menetelmässä.

Jälkikäteen haastattelussa tekee tarvittavat testitehtävät itsekseen ja lopuksi testikäyttäjät haastatellaan tai mahdollisesti voidaan käyttää kyselylomaketta. Testaustapa on sama kuin kvantitatiivisissa testeissä. Samalla voidaan selvittää myös testikäyttäjän tyytyväisyyttä käyttöliittymään tai tuotteeseen. Tällä menetelmällä voidaan myös saada selville hyvin mahdollisia käyttövirheitä ja kognitiivista kuormitusta testitilanteessa. Tämä menetelmä ei myöskään tarvitse ohjaajaa paikalle, mutta testikäyttäjän motivaation on hyvä olla korkea, että mahdollinen tieto saadaan kerättyä. Menetelmä vaatii myös huomattavaa ja huolellista suunnittelua.

Jälkeenpäin kommentoinnissa testikäyttäjä tekee tehtävät ilman ohjausta. Testaustilanne kuitenkin taltioidaan, jonka jälkeen testikäyttäjä ja ohjaaja katsovat tallenteen, josta käyttäjä kommentoi tilanteita tallenteelta. Tällä tavalla saadaan selville käytön ongelmakohtia. Menetelmä sopii myös hyvin kvantitatiiviseen testaukseen. Seuraavat puutteet kuten, käyttäjä ei muista kaikkea, käyttäjä saattaa muuttaa mielipidettään tai kaunistella mielipidettään ja myös se, että testikäyttäjä tietää enemmän selostaessaan, kuin tiesi testitilanteessa itse varjostavat jälkeenpäin kommunikointia.

Pikkutestit, osatesti, käsitelikat, joissa testikäyttäjä tekee kynällä ja paperilla annetun tehtävän ja kommentoi ohjaajalle miksi ja mitä teki. Tuloksien läpikäynti tulee käydä läpi testikäyttäjän kanssa.

Ryhmäläpikäynnissä testikäyttäjät, ohjaaja, suunnittelija käyvät testitehtävät läpi yhdessä käyttäen käyttöliittymän kuvia. Tämä sopii erityisen hyvin prototyypitestaukseen, jossa kaikkea toiminallisuutta ei ole vielä ehditty tehdä valmiiksi. Tässä menetelmässä on erityisen hyvää se, että testikäyttäjät ja suunnittelijat ovat samassa tilassa keskustelemassa käyttöliittymästä. Testikäyttäjät pystyvät myös vapaammin kertomaan mielipiteitään paperilla esitetyistä kuvista, kuin valmiista tuotteesta.

Vapaassa läpikäynnissä testikäyttäjä kokeilee testattavaa ohjelmaa tai järjestelmää rauhassa. Testin ohjaaja ei puutu testin kulkuun millään tavalla. Tämä menetelmä on syytä jättää kuitenkin hyvin valmiille käyttöliittymälle tai prototyypille, koska kaikkien toimintojen olisi syytä löytyä jo tässä vaiheessa käyttöliittymästä tai tuotteesta itsestään. Testin ohjaajan on tunnettava käyttöliittymä erittäin hyvin, jotta testin ohjaaja voi seurata ja havainnoida miten hyvin testikäyttäjä ohjelman käyttämisestä suoriutuu. (Sinkkonen, Kuoppala, Parkkinen & Vastamäki 2006, 285 - 287)

3.5.5 Pilottitesti

Pilottitesti on otettu yhtenä kokonaisuutena, koska se on yksi erittäin tärkeä osa-alue käyttöliittymää tehtäessä. Pilottitestin tarkoitus on varmistaa käytettävyydestin toimivuus sekä testausympäristön sopivuus käytettävyydestiin. (Sinkkonen, Kuoppala, Parkkinen & Vastamäki 2006, 288)

Pilottitestiin valmistauduttaessa on hyvin tavallista, että testitehtäviä korjataan pilottitestivaiheessa. Pilottitestissä siis korjataan testitehtäviä, täydennetään haastattelukysymyksiä, koekäytetään testitehtävät ja mitataan niiden suorittamiseen menevä aika. Pilottitestissä ohjaajalle ja muille testiin osallistujille on hyvä ottaa mukaan muistilista, jotka pitää tehdä ennen testiä, testin aikana ja testin jälkeen. Testisuoritus voidaan jakaa seuraavasti (Sinkkonen, Kuoppala, Parkkinen & Vastamäki 2006, 288 - 290):

- testitilanteen selvittäminen käyttäjälle
- alkukysely
- testitehtävien tekeminen
- visuaalinen läpikäynti
- loppuhaastattelu

Testitilanteen selvittäminen testikäyttäjälle käyttöliittymän testaamisessa on hyvin tärkeä, koska tässä vaiheessa tuotteen tai käyttöliittymän valmiusastetta ei ole tarkoitus testata käyttäjän selviytymistä tehtävistä vaan itse käyttöliittymää. Ohjaajan on selvitettävä testikäyttäjälle, että testien aikana on mahdollista lähettää suunnittelijoille palautetta käyttöliittymän toimivuudesta. On hyvä myös kerrata, että testihenkilön tärkeyttä, koska tämän avulla käyttöliittymästä saadaan mahdollisimman helppokäyttöinen ja ymmärrettävä käyttöliittymä.

Alkukyselyn tarkoituksena on selvittää käyttäjien taustaa sekä selvittää heidän ennakkosenteitään. Kun otetaan huomioon kuntoutusohjelmassa käytettävän käyttöliittymän testaus, niin tässä erityisesti tulee ottaa huomioon testikäyttäjien osaaminen testin kohdealueelta ja tietokoneen, niin kuin mobiili applikaatioiden käyttömäärä.

Testitehtävien tekemiseen tulee varata aikaa, koska riippuen käyttäjän iästä tai käyttöliittymän ymmärrettävyydestä tulee aikaa olla suorittaa tarvittavat testitehtävät. Ohjaajan on hyvä käydä alkutilanne läpi testikäyttäjän kanssa. Erityisesti kuntoutusohjelmaa otettaessa huomioon, voi tilanne hyvinkin olla siten, että tilanteet tai tehtävät syntyvät testitehtävien etenemisellä.

Visuaalinen läpikäynnissä testikäyttäjä kertoo, että miten hän omasta mielestään suoriutui testitehtävistä ja miten hän ymmärsi tehtävät ja tulkitse näkemänsä käyttöliittymässä. Läpikäynnissä testikäyttäjä ymmärtää miten hänen olisi pitänyt toimia missäkin tilanteessa ja tässä vaiheessa saadaankin hyvää palautetta siitä, että miten käyttöliittymää tai tuotetta on muutettava, jotta siitä tulisi helpommin ymmärrettävä tai helppokäyttöisempi.

Loppuhaastattelu on syytä aloittaa aina kysymällä testikäyttäjän tuntemuksia testattavasta käyttöliittymästä tai tuotteesta. Hyvä käyttäjän tunnelma itse tuotteesta on erittäin oleellinen osa mihin lopputuotteen kanssa kannattaa pyrkiä. Loppuhaastattelun tulisikin olla mahdollisimman vapaa ja testikäyttäjän tulisikin ilmaista itseään mahdollisimman vapaasti. Suunnittelijoilla tulisi myös olla mahdollisuus haastatella testikäyttäjää, koska se voi olla erittäin suuressa arvossa, kuitenkin ottaen huomioon, että hän ei saa vaikuttaa testikäyttäjän antamaan palautteeseen tai lopputulokseen. (Sinkkonen, Kuoppala, Parkkinen & Vastamäki 2006, 288 - 291)

3.5.6 Käyttäjäkokemukseen vaikuttavat tekijät

Käyttöliittymä, mobiiliapplikaatio, sovellus voi mukautua eri lähdetietojen perusteella, kuten käyttäjäprofiilin, toimintatapojen, ympäristön, ajan ja ympärillä olevien teknologioiden mukaan. Käyttäjä itse ei välttämättä edes tiedosta tai huomaa miksi ja minkä mukaan sovellus kulloinkin mukautuu. Pitääkin huomioida käyttöliittymää tai mobiiliapplikaatiota tehtäessä tuleeko käyttäjän huomata sovelluksen mukautuminen ja mitä jos käyttäjän mukaan sovellus mukautuukin tiedostamattomasti. (Kaasinen & Norros 2007, 197 - 198). Mukautuvan vuorovaikutuksen suunnittelussa korostuvat käytettävyyys ja kuten käyttäjäkokemustavoitteet, huomaamattomuus, ennustettavuus, hallittavuus ja näkyvyys. (Jameson, 1-3)

Näkyvyys on sitä, että miten sovellus toimii ja miten käyttäjä itse ymmärtää sovelluksen toiminnan. Näkyvyys on sitä, että sovellus tai käyttöliittymä informoi käyttäjää siitä, mitä on tekemässä ja miksi. Tällä parannetaan käyttäjän ymmärrettävyyttä, jolloin sovellusten ja käyttöliittymän käyttäminen helpottuu huomattavasti. (Kaasinen & Norros 2007,198)

Huomaamattomuus on sitä, että käyttöliittymä tai sovellus ei saa viedä käyttäjän ajatuksia liikaa muualle siitä työstä mitä hän oli tekemässä. Alkuperäisen työtehtävän suorittaminen ei keskeydy. On haastavaa löytää sellainen raja missä näkyvyys ja hallittavuus ovat

harmoniassa, jossa käyttäjä osaa arvioida sovelluksen tai käyttöliittymän ennustettavuutta. Tällöin käyttäjä tuntee myös sen, että hän hallitsee teknologian. (Kaasinen & Norros 2007, 199)

Ennustettavuus on sitä, miten käyttäjä voi ennustaa toimintansa vaikutukset. Kun käyttöliittymää tai sovellusta käytetään käyttäjä olettaa tiettyjen tapahtumien tapahtuvan. Mukautuvan sovelluksen tai käyttöliittymän suunnittelu on huomattava haastavaa, koska ennustettavuus käyttäjän näkökulmasta ei ole jokaisessa tapauksessa samanlainen tai toiminnot eivät tapahdu samalla tavalla loogisessa järjestyksessä. Ympäristö on hyvä esimerkki siitä, millä tavalla sovellus tai käyttöliittymä voi mukautua käyttäjän sitä odottamatta. Käyttäjäkokemukseen siis vaikuttaa suuresti se, että käyttäjä olettaa, että sovellus toimii niin kuin on olettanut. (Kaasinen & Norros 2007, 198)

Yksityisyyteen ja turvallisuuteen liittyvä asiat ovat erittäin kriittisiä osatekijöitä hyvän käyttäjäkokemuksen saamiseksi. Käyttäjälle pitää selkeästi ilmaista mitä tietoja käyttäjältä kerätään ja miksi, varsinkin kun kyse on henkilökohtaisista tiedoista. (Kaasinen & Norros 2007, 199 -200)

Hallittavuudessa on kyse siitä, miten ja millä tavalla käyttäjä voi hallita sovellusta tai käyttöliittymää. Mukautuvat käyttöliittymät ja sovellukset tulisi rakentaa siten, että käyttäjä tuntee hallitsevansa asioita ja olevansa vastuussa niistä. Kun käyttäjä tuntee olevansa itse vastuussa ja hallitsee asioita, niin vaikuttaa se suuresti käyttäjäkokemukseen. Suunnittelijoiden tulisi kehittää käyttöjärjestelmästä tai sovelluksessa mahdollisimman helpokäyttöinen, että tämä tunne saavutetaan, jossa hallittavuus säilyy itse käyttäjällä. Hallittavuudessa ominaista on myös se, että sovellus tai käyttöjärjestelmä voi oppia eri tilanteista ja mukauttaa sovellusta toimimaan oppimansa perusteella. Sovelluskehittäjät voivat, myös tehdä käyttöjärjestelmästä tai sovelluksesta muokattavan, jolloin käyttäjän hallittavuus kasvaa. Käyttäjien itse sallimat muokattavuudet tulee kuitenkin pitää hyvin helpokäyttöisinä, eikä vaadi liikaa teknistä asiantuntemusta. (Kaasinen & Norros 2007, 199)

Kokemuksen kokonaislaajuus on kaiken kattava osa käyttäjäkokemukseen vaikuttava tekijä. Kokemuksen kokonaislaajuudella voidaan tarkoittaa sitä, että käyttäjä voidaan yllättää positiivisesti ja odottamattomasti. Sovelluksen on helppo toimia käyttäjän antamien ohjeiden mukaan, siitä mistä tämä on kiinnostunut. (Kaasinen & Norros 2007, 199)

Käyttöliittymien odotetaan nopeuttavan ja helpottavan käyttäjän toimintatapoja, mutta hallittavuuden, näkyvyyden, yksityisyyden ja turvallisuuden tunteen puutteet tuntuvat siis negatiivisena puolena. Tasapainoilu näiden kaikkien asioiden välillä on siis haasteellista. Suunnittelijoiden, sekä sovelluskehittäjien ja valmistajien tulisikin muistaa, että on miltei mahdotonta tyydyttää kaikkia ihmisiä ja käyttäjiä.

3.6 Tietoturva ja tietosuoja

Juho Kauppi

Tietoturva ja tietosuoja ovat kaksi eri asiaa. Tietosuoja perustuu lakiin, käytänteisiin ja hyviin tapoihin. Tietoturva perustuu teknisiin ja organisatorisiin ratkaisuihin. Tietoturvasta puhuttaessa tarkoitetaan usein myös tietosuojan käytänteitä. Tietosuoja saadaan kuntoon vain hyvillä tietoturvan ratkaisuilla.

Tietoturva kattaa kaiken sen, mikä liittyy tietojen saatavuuteen, oikeellisuuteen sekä tietojen luottamuksellisuuden säilymiseen käsittelyn, säilytyksen ja tiedonsiirron aikana. (Järvinen 2002)

Tietosuojalla eli yksityisyyden suojalla tarkoitetaan ihmisen henkilötietojen sekä henkilökohtaiseen toimintaan liittyvien tietojen keräämisen ja käsittelyn rajoittamista niin, ettei henkilön yksityisyys turhaan vaarannu. (Järvinen 2002)

3.6.1 Mihin tietoturvaa tarvitaan

Tietoturva suojaa sisäisiä tietoja ulkopuolisilta toimijoilta. Tietoturva on dokumenttien ja viestien turvaamista, sekä se on kokonaisuus, johon liittyvät sekä tiedonkäsittelylaitteiden fyysinen turvallisuus, että ihmisten toiminta organisaatiossa. Tietoturvalla pyritään estämään tietojen, järjestelmien tai palvelujen luvaton selville saanti tai käyttöönotto. Tietojen ja tiedonkäsittelyresurssien on pysyttävä käytettävissä ja käyttökelpoisina valtuutetuille käyttäjille. Internet avoimena järjestelmänä antaa mahdollisuuden hyökkäyksiin yritysten ja organisaatioiden tietojärjestelmiä kohtaan.

Internetin tietoturva voidaan myös jakaa kahteen eri luokkaan seuraavasti:

- 1) Tietoliikenteen suojaaminen
 - Estetään kahden laitteen välisen liikenteen kuunteleminen
 - Tietoliikenne suojataan salaamisen eli kryptografian avulla
- 2) Palvelinkoneen ja laitteiden (IoT) suojaaminen
 - Estetään asiaankuulumattomien pääsy palvelimelle
 - Kone eristetään Internetistä palomuurin avulla
 - Käyttäjien tunnistaminen

Edellisten lisäksi on hyvä huolehtia tietojen turvaaminen sisäisten ja inhimillisten uhkien varalta sekä luonnonilmiöiltä.

3.6.2 Tietoturvan tavoitteet

1. Luottamuksellisuus (Confidentiality)

- kukaan ulkopuolinen ei tiedä osapuolten välisestä viestinnästä
- salattu viesti on ulkopuoliselle käsittämätön
- käyttäjät pystytään todentamaan
- tieto salataan (Encryption)

2. Eheys (Integrity)

- tieto ei muutu matkalla
- tieto säilyy kokonaisena
- esim. www-sivulle tunkeutujat (murtautuminen, virukset)
- esim. sähköpostin sisällön ja lähettäjätietojen manipulointi
- esim. viallinen levy - käsittelyvirheet tmv.
- tärkeää lokitiedostojen arkistoinnissa

3. Saatavuus (Availability)

- verkkoyhteyksien toimivuus online
- tietojen käytettävyys; varmuuskopiointi
- laitteiden toimintavarmuus; UPS -laite sähkökatkoksiin
- 'ennen vanhaan' tallennettujen tiedostojen käytön varmistaminen
- esim. palvelunestohyökkäyksillä voidaan häiritä

4. Todennus (Authentication)

- viestin lähettäjän (olion) henkilöllisyyden todistaminen; olio voi olla laite, www-sivu, ohjelmakoodi
- digitaalinen allekirjoituksen lisääminen sähköpostiviestiin tehostaa viestin autentikointia; lähettäjän kun on helppo väärentää sähköpostiosoite
- käyttäjät todennetaan salasanoin
- laite todennetaan toiselle laitteelle salauksen sovelluksin
- verkkotiedon todennus on vaikeaa
- nettipalvelu todennetaan osoitteen perusteella; SSL-protokollan avulla parempi varmuus
- GSM-laite todennetaan tukiasemaverkostolle
- ohjelmakoodin todentaminen

5. Pääsynvalvonta (Access control)

- vain todennetut henkilöt pääsevät järjestelmään
- pääsynvalvontaan liittyy käytön seuranta

6. Kiistämättömyys (Non-Repudiation)

- viestin lähettäjä ei voi kiistää lähettäneensä viestiä
- transaktioiden aikatiedon tallennus

Tietoturva on asioiden oikein tekemistä. (Elisa 2016)

3.6.3 Tietosuoja

EU:n yleinen tietosuoja-asetus astuu voimaan 25.5.2018 alkaen. Asetus korvaa henkilö-tietodirektiivin ja kansalliset tietosuojalait (Suomessa henkilötietolain). Tietosuoja-asetus muuttaa merkittävästi henkilötietojen käsittelyyn liittyviä oikeuksia ja velvollisuuksia. Tässä työssä on erityisesti paneuduttu uuden tietosuoja-asetuksen (GDPR) vaikutuksiin.

Tietosuoja-asetus tuo saman sääntelyn koko EU:n alueelle; yrityksille ja rekisteröidyille tulee yhden luukun periaate. Käsittelijät ovat itsenäisessä vastuuasemassa suhteessa rekisteröityihin ja valvontaviranomaisiin. Valvontaviranomaisille tulee sakotusoikeus, jos organisaation tietosuoja-asiat eivät ole kunnossa. Rekisterinpitäjän ja käsittelijän on näytettävä toteen, miten tietosuojaa toteutetaan organisaatiossa.

Rekisteröidyn henkilön oikeus määrätä henkilötiedoistaan vahvistuu

- Oikeus tutustua omiin henkilötietoihin parantuu
- Henkilötietojen käsittelyyn tarvitaan henkilön yksilöity, tietoinen ja yksiselitteinen suostumus
- Oikeus siirtää tietoja palveluntarjoajalta toiselle (Data Portability)
- Oikeus saada korjattua mahdolliset virheelliset henkilötiedot tai poistaa ne
- Oikeus rajoittaa henkilötietonsa käsittelyä
- Oikeus ”tulla unohdetuksi”
- Alaikäisten henkilötietojen käsittelyyn vanhempien lupa

Henkilötietojen käsittely on lainmukaista, jos sille on vähintään yksi seuraavista perusteista (6. Artikla):

- Rekisteröidyn suostumus
- Sopimuksen täytäntöön panemisen tai edeltävien toimenpiteiden toteuttaminen
- Rekisterinpitäjän lakisääteisen velvoitteen noudattaminen
- Rekisteröidyn tai toisen henkilön elintärkeiden etujen suojaaminen
- Rekisterinpitäjän tai kolmannen osapuolen oikeutettujen etujen toteuttamiseksi, paitsi milloin henkilötietojen suoja edellyttävät rekisteröidyn edut tai perusoikeudet ja -vapaudet syrjäyttävät tällaiset edut, erityisesti jos rekisteröity on alaikäinen
- Tarpeen yleisen edun vuoksi tai rekisterinpitäjälle kuuluvan julkisen vallan käyttämiseksi toteutetun tehtävän suorittamiseksi

3.6.4 Terveysalan tietoturvan erityispiirteitä

Terveydenhuollon tietosuojaa ja salassapitoa koskevat asiat ovat lakisääteisiä. Terveydenhuollossa asiakas- ja potilassuhteiden perusta on tiedon luottamuksellisuus ja salassapitovelvollisuus. Laki terveydenhuollon ammattihenkilöstöstä (1994, 18§) velvoittaa terveydenhuollon ammattihenkilön tuntemaan ammattitoimintaansa koskevat säännökset ja määräykset. Myös opiskelijoita velvoittavat salassapito ja henkilötietojen asianmukainen käsittely.

Salassapitovelvollisuudesta säädetään julkisuuslaissa ja laissa terveydenhuollon ammattihenkilöistä. Salassapitovelvollisuus on ikuinen ja se säilyy myös ammatinharjoittamisen jälkeenkin. Salassapitovelvollisuudella on kolme muotoa: asiakirjasalaisuus, vaitiolovelvollisuus sekä tietojen hyväksikäyttökielto (Laki viranomaisten toiminnan julkisuudesta 1999, 22-23§).

Etäkuntoutuksen tai etäseurannan potilastietoa koskee samalainen luottamuksellisuus ja salassapito velvollisuus kuin perinteisesti, kasvokkain, tapahtuvassa toiminnassa. Tämä tarkoittaa, että laitteistojen ja tiedonsiirtovälineiden valinnassa ja käytössä on huomioitava turvallisuus ja salassapito. Jos tieto varastoidaan eli tallennetaan myöhempää käyttöä ajatellen, niin potilaan täytyy olla tietoinen tästä ja luottaa että tietoturvasuus huomioidaan asianmukaisesti. Kolmannet osapuolet eivät saa päästä käsiksi seurattavan tietoihin ja mahdollisesti tunnistamaan asiakasta. (Etäkuntoutus 2016, 31 ja 38)

Terveysalan tietoturvan ratkaisuihin pyritään optimoimaan turvallisuus, toiminnallisuus ja kustannukset niin, että saadaan toteutettua kokonaisedullinen ratkaisu. Tietoturvan toteutus pyrkii suojaamaan tiedot, tietoliikenneyhteydet, ohjelmistot ja laitteistot, niin fyysisesti kuin teknisesti ja toiminnallisestikin. Mahdolliset uhat eivät saa aiheuttaa merkittävää lisäriskiä ja turvallisuus on oltava kunnossa sekä normaaleissa että poikkeustilanteissa. On tärkeä miettiä palveluja suunniteltaessa, että mitä potilasdataa kerätään ja säilytys järjestetään. (Sylander 2016)

Potilastietojen siirtämiseksi on tärkeää valita turvallinen tapa siirtää tietoa. Tietoturva perustuu moninkertaiseen salaukseen. Potilastietojen suojaamiseksi voidaan käyttää monenlaisia ratkaisuja, kuten:

- Avoimien verkkojen kanssa käytetään VPN-yhteyttä, jolla saadaan salattua tietoliikenne ja loppukäyttäjän sijainti. Yleinen tällainen salausteknologia on SSL, joka suojaa tietokoneen ja palvelun välisen yhteyden.
- Eri laitteet yhdistetään toisiinsa hyväksymällä yhteydenottoopyyntö sekä todentamalla käyttäjän oikeutus avainkoodilla
- Laitekohtaisesti vaaditaan 128-bittinen salausavain
- Verkkoon pääsyyn vaaditaan vähintään 64-bittinen avainkoodi
- Applikaatioiden käyttäjiltä vaaditaan myös oma 64-bittinen avainkoodi
- Kirjautuminen palveluun omilla pankkikohtaisilla tunnuksilla ja koodilla on nykyisin varsin yleinen ja helppo tapa toteuttaa käyttäjän tunnistus

Suurin turvallisuus riski on kuitenkin itse laitteen käyttäjä, jos hän hyväksyy yhteyspyyntöjä tuntemattomista laitteista tai lähteistä. On tärkeää määritellä, että kenellä on pääsy-oikeus kerättyyn dataan. Käyttäjät täytyy tunnistaa ja käytön historia täytyy tallentaa. Osa tätä kokonaisuutta on tietoturvapoliittikka, jossa arvioida mahdolliset, tärkeimmät ja todennäköisimmät uhat.

Terveysalan tietoturvan toteutukseen liittyy myös paljon alakohtaista lainsäädäntöä. Palveluntuottajien kannalta on oleellista tutkia, että minkä maan lainsäädäntöä täytyy noudattaa. Tähän vaikuttaa se, että missä tiedon säilytykseen tarvittavat palvelimet sijaitsevat. (Etäkuntoutus 2016, 202) Suomen lainsäädännöstä kannaltamme tärkeitä ovat:

- Terveydenhuoltolaki 30.12.2010/1326
 - o 9§ potilastietorekisteri ja potilastietojen käsittely
- Henkilötietolaki 523/1999
 - o 33§ vaitiolovelvollisuus
 - o 34§ henkilörekisterin hävittäminen
 - o 35§ henkilötietojen siirto arkistoon
- Laki sosiaali- ja terveydenhuollon asiakas-tietojen sähköisestä käsittelystä 159/2007
 - o 1§ lain tarkoitus
 - o 2§ soveltaminen
 - o 4§ asiakastietojen käytettävyys ja säilyttäminen
 - o 5§ käytön ja luovutuksen seuranta
 - o 6§ potilasasiakirjojen tietorakenteet
 - o 7§ suunnittelu-, tutkimus- ja tilastotieto

- 8§ tunnistaminen
- 9§ asiakirjan sähköinen allekirjoittaminen
- 10§ potilastietojen luovuttaminen
- 13§ potilaan suostumus
- 14§ valtakunnalliset tietojärjestelmäpalvelut
- 15§ velvollisuus liittyä tietojärjestelmäpalvelujen käyttäjäksi
- 17§ potilaan informointi
- 18§ asiakkaan tiedonsaantioikeus
- 20§ ohjaus, valvonta ja seuranta

3.7 Kokonaisarkkitehtuuri

Juho Kauppi

Kokonaisarkkitehtuuri määrittelee ja mallintaa järjestelmän, joka suunnitellaan asiakkaiden käyttöön. Arkkitehtuurin määrittely etenee vaiheittain tarkentuen ja siitä voidaan erottaa ainakin seuraavat vaiheet.

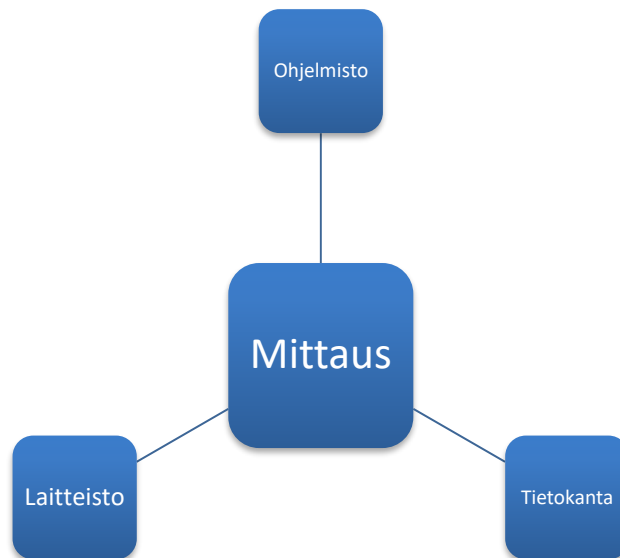
- 1) Määritellään asiakkaan tarpeet ja vaatimukset eli käyttö.
- 2) suunnittelu vaiheessa määritellä toiminnallisuus.
- 3) Toteutus vaiheessa rakennetaan järjestelmä ja testataan sitä
- 4) Järjestelmää täytyy käytön aikana hallita ja tarvittaessa suorittaa muutoksia

Tietoliikenteellekin on hyvä olla olemassa oma arkkitehtuurinen suunnitelma. Kaikkea raakadataa ei kannata siirtää sensorilta tietoverkon yli analysoitavaksi IoT-alustalle, koska se rasittaa turhaan siirtokapasiteettia. Sujuva ja luotettava toiminta edellyttää verkon perusrakenteen eli laitteiden liittämisen toisiinsa hyvää suunnittelua ja toteutusta. On mietittävä jo etukäteen, että mikä tieto on tarpeen lähettää analysoitavaksi ja mitä kannattaa tallentaa myöhempää käyttöä varten. Yleinen tapa välttää turhan tiedon lähetystä on välittää raakadata useilta sensoreilta ensin lähiverkon yhdystietokoneelle, joka suorittaa datan alkuanalysoinnin ennen lähetystä tallentavalle pilvipalvelimelle. Tätä mallia kutsutaan Gateway- eli yhdyskäytävämalliksi. Raakadatan suodatus paikallisesti Gateway-arkkitehtuurin mallin mukaan vähentää tietoliikenteen määrää ja säästää näin resursseja, kun ylisuureen tiedonsiirtokapasiteettiin ei tarvitse investoida. Suodatuksen etuna on myös että se tiivistää oleellisen datan määrää ja näin monesti analysoinnin kannalta tärkeät häiriöt tulevat paremmin huomatuiksi. Gateway-tietokoneet myös pystyvät puskuroimaan dataa esim. tietoliikenne häiriöiden yhteydessä ja näin datan keräily sensoreilta ei turhaan keskeydy. Tästä on hyötyä myös integroinnissa, koska Gateway-tietokone voi huolehtia protokollien yhtenäistämistä. (Collin & Saarelainen, 190)

3.8 Integraatio

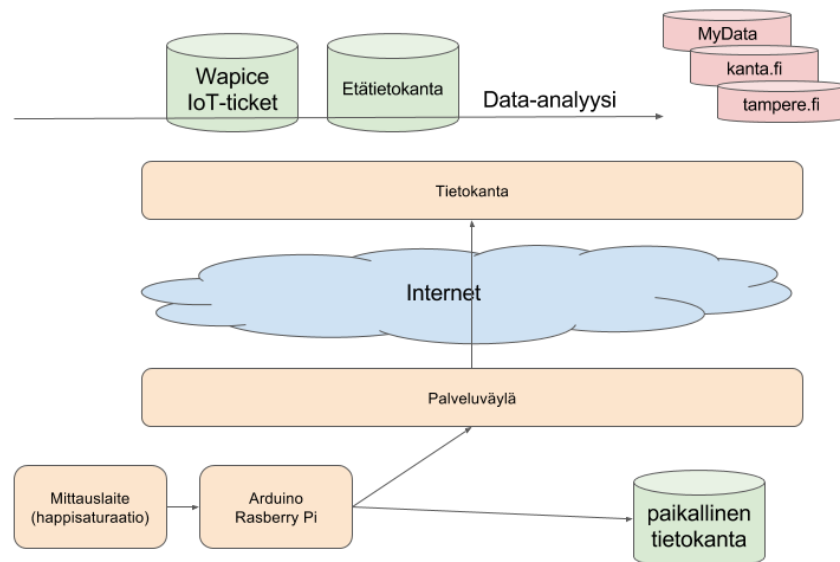
Juho Kauppi

Integraatio on erillisten osioiden yhdistämistä tai järjestämistä yhdeksi kokonaisuudeksi. Palvelukokonaisuus muodostuu seuraavista osa-alueista; ohjelmisto, laitteisto, mittaus ja tietokanta. Palvelukokonaisuutta on kuvattu kuviossa 4.



KUVIO 4 Palveluintegraatio (Juho Kauppi 2017)

Integraatiossa mittaustapahtuma suoritetaan mittauslaitteistolla (esimerkiksi pulssi tai happisaturaatio) siihen liitettyllä ohjelmistolla, joka lähettää mittaustiedot internetin välityksellä tietokantaan. Ohjelmistoa kutsutaan palveluarkkitehtuurissa palveluväyläksi, johon mittalaitteet liitetään. Ohjelmistokokonaisuuteen kuuluu kolme keskeistä toimintoa; itse mittaustapahtuma, mittaustuloksen lähetys internetin ylitse (Internet of Things -tapahtuma) ja tiedon vastaanotto tietokantaan.



KUVIO 5 Integraatioarkkitehtuuri (Juho Kauppi 2017)

Hyvä integraatioarkkitehtuuri rakentuu modulaarisesti eli arkkitehtuurin eri osat keskustelevat keskenään rajapintojen kautta. Rajapinta tulisi olla avoin eli toteuttaa jotain yleisesti tunnettua menetelmää kommunikointiin. Tällöin arkkitehtuurin eri osakokonaisuuksia pystytään vaihtamaan ilman, että kokonaisuus kärsii. Etenkin oppilaitoksen kehittämissympäristönä modulaarisuus tuo etuja, kun ei tarvitse olla toimittajasidonnainen vaan voidaan testata erilaisia laitteita tai teknologioita. Kehitteillä olevaan tiedonsiirto ratkaisuumme soveltuva integraatioarkkitehtuurimalli on kuvailtu kuviossa 5.

4 HYVINVOINNIN ETÄMITTAUKSEN TOTEUTUS

Datasiirron pilotti osuudessa suunnittelimme ja toteutimme TAMK:n ensimmäisen terveystiedon siirron ratkaisun valituilta sensoreilta IoT-alustalle.

4.1 IoT-alustan valintaperusteet

Mika Helonsalo

Aluksi kiinnostuimme erityisesti Elisan tarjoamasta IoT-alustasta, koska heillä oli jo tarjolla etämittauspalveluja terveysalalle. Elisa esitteli meille yhteistyökumppaninsa kehittämää Thingworx-alustaa. Se vaikutti olevan alun perin suunniteltu ja soveltuvan hyvin esim. kuorma-autojen rahdin seurantaan. Elisa oli kouluttanut muutamia TAMK:n opettajia Thingworx-alustan käyttöön sekä kouluttajiksi. Thingworx-käytön oppimisen sanottiin kuitenkin olevan hidas prosessi, joten olimme avoimia myös muille vaihtoehdoille.

Halusimme valita koululle alustan, joka soveltuisi käytettäväksi kaikilla osastoilla. Talotekniikka oli jo tutkinut Wapicen IoT-Ticket-soveltuvuutta ja ottanut sen osittain käyttöönsä. Alihankinta2016-messuilla CGI:n konsultti suositteli IoT-Ticketiä käyttööme. Wapicen edustajat vakuuttivat, että heidän alustansa soveltuisi myös terveyden etämittaussovelluksiin, vaikka siihen sitä ei ennen ole käytettykään. Järjestimme koululla esittelyn, jossa oli mukana talotekniikan, IT-puolen ja hyvinvointiteknologian opettajia. IoT-Ticket-käyttöä harjoitelleet TAMK:n opettajat vakuuttivat helposta ja nopeasta käyttöönotosta sekä joustavasta järjestelmästä. Markus Mäkelä Wapicelta saapui pitämään esittelyn IoT-Ticketistä 3.11.2016.

Seuraavaksi analysoimme IoT-Ticket-alustan käyttökelpoisuutta kahdeksan eri huomiotavan kohdan menetelmällä.

1) Liitettävyys sensoriverkkoon

Sensorien pitää olla helposti yhdistettävissä verkkoon ja rekisteröitävissä järjestelmään. Sensorien liittäminen osaksi tiedonkeruuverkkoa tapahtuu Dashboard-käyttöliittymällä suoraan verkkoselaimessa. Eri sensoreiden yhdistämismahdollisuuksia on kuvattu kuviossa 6. IoT-Ticket:iin voidaan käytännössä lähettää tietoja kolmella eri tavalla (Vehmaanperä 2016):

- Liitetään sensorit Wapicen oman VRM 247+ kautta. Laite sisältää fyysisiä liittymisiä erilaisille antureille ja toimilaitteille, joten sillä voidaan sekä lukea että kirjoittaa tietoa kenttälaitteille. Laitteessa on ohjelmistotasolla tuki muun muassa seuraaville tavoille: 1-Wire, CAN, IEC 61850, ISO 11783, OPC UA sekä mA ja V tuloille ja lähdöille.
- Rest API rajapinnan kautta, jonka avulla voidaan liittää lähes mikä tahansa laite osaksi kokonaisuutta. Tällainen liityntä on esim. IoT-Tracker applikaatiolla, jolla älypuhelimien sensorit saadaan tuotua osaksi tiedonkeruuta.
- Suoraan IoT-Ticket.com -palvelimen kommunikointi rajapinnan kautta. Palvelin voi kommunikoida suoraan OPC- DA/UA järjestelmien kanssa sekä käyttää Modbus, MQTT ja SNMP protokollia.

IoT-Ticket.com
Connecting to Ecosystem

Easy data acquisition:

- 1-wire protocol
- Analog and digital inputs and outputs
- BACnet
- CAN
- CANopen
- OPC
- OPC-UA
- Modbus
- Modbus TCP
- IEC 61850
- ISO 11783
- USB
- OBD
- REST (JSON and XML)
- (and many more)

Sensor support:

- Accelerometer (Digital, triaxial, 16 bit, 2g/±4g/±6g/±8g/±16g)
- GPS
- Temperature

Develop own IoT-Ticket.com device:

- IoT Ticket API

3rd party interfaces:

- Matlab/Simulink
- R

The slide also features images of various IoT devices and a Wapice logo in the bottom right corner.

KUVIO 6 IoT-Ticket yhteydet ekosysteemiin (Wapice 2016)

WRM 247+ on itsenäinen laite, jolla voidaan toteuttaa etäyhteys sensorin ja IoT-Ticket palvelimen välille. Se hallinnoi, laskee ja ohjaa sensoreita. Tieto voidaan lähettää 3G-, WLAN- tai BT-yhteyden kautta. Lisää tietoa tästä laitteesta löytyy wapice.com/files/WRM247+.

2) Laitteiden hallinta ja sääntöjen muodostaminen

Alustan avulla täytyy pystyä valvomaan, että päätelaite toimii oikein, tuottaa dataa odotetulla tavalla sekä pystyä toteamaan, että ohjelmisto on ajan tasalla ja pystyssä. Sensorien tuottamaa tietoa päästään tarkastelemaan IoT-Ticket Dashboardin kautta. Tehtyjen mittareiden ja sensorien välille luodaan haluttu yhteys. Tätä pääsimme käytännössä kokeilemaan, kun muodostimme haluamillamme mittareille sensoriliityntöjä. Sensorien liittämisen mittareihin pystyi toteuttamaan reaaliaikaisesti ja niiden tuottama data oli täten nopeasti luettavissamme sekä todettavissa oikeanlaatuisiksi.

3) Tietokanta

Kerätyn datan tallennus on oleellinen asia. Alustan on selvittävä suuresta määrästä eri laatuista dataa. IoT-Ticket on verkkopohjainen alustaratkaisu, jossa voi tallentaa kerätyt tiedot joko Wapicen Big-data Analytics -palvelimelle (pilvipalvelin) tai omalle palvelimelle. TAMKin yksi oleellisista vaatimuksista oli, että tiedot pitää saada suoraan omaan käyttöön ja omistukseen. Oman palvelimen käyttö on järkevää, koska silloin tiedonsiirtonopeus ei ole niin suuri rajoittava tekijä. Valitettavasti emme kerenneet saamaan mukaan ratkaisuumme TAMKin omaa palvelinta, koska se saapui vasta työmme viimeistelyvaiheessa.

4) Datan prosessointi ja yhdistäminen käyttökelpoiseksi kokonaisuudeksi

Datasta täytyy löytää oleelliset poikkeamat ja näiden perusteella havaita oleelliset muutujat. Datan poikkeamien analysointiin ei tämän opinnäytetyön aikataulun puitteissa ollut mahdollisuutta. Keskityimme tiedonkeruun aloituksen mahdollistamiseen. Myöskin datan prosessointia käsittelemme tässä työssä vain teoreettisesti. Huomasimme varsinkin kiihtyvyyssanturin kanssa, että tiedot pitäisi analysoida ja suodattaa ennen lähetystä serverille ja käytettäväksi. Eri sensoreiden tuottama tieto saatiin kuitenkin helposti havainnollistettua Dashboardilla.

5) Datan analysointi ja hallinta

Raja-arvojen ylityksien ja poikkeamien täytyy laukaista hälytykset. Dashboardilla löytyy työkalut raja-arvojen ylitysten ja poikkeamien huomaamiseksi reaaliaikaisesti. Seurattaville suureille asetetaan raja-arvo, jossa hälytys tapahtuu. Toteutuksemme yhteydessä emme katsoneet tarpeelliseksi vielä ottaa hälytyksiä käyttöön, koska ensisijainen tavoitteemme oli ratkaista tiedonvälitys alustalle.

6) Datan visualisointi ja raportointi

Jotta data tuottaisi uutta arvokasta tietoa, niin sitä täytyy pystyä käsittelemään havainnollisempaan muotoon. Visualisointi voi olla havainnollisia mittaristoja tai graafisia yhteenvedoja datasta. Pyrimme havainnollistamaan datan keräystä luomillamme Dashboard-näkymillä ja raporteilla. IoT-Ticketiltä voi lähettää raportin PDF muodossa sähköpostiin haluamasi aikana ja aikavälillä, esim. päivä tai viikkoraportin. Raportteja voi havainnollista valmiina olevilla Widgeteillä sekä lisäämällä selosteita, että omavalintaisia kuvia.

7) Tietoturva ja tietosuoja, pääsynhallinta ja tunnistaminen

On tärkeää määritellä, että kenellä on pääsyoikeus kerättyyn dataan. Käyttäjät täytyy tunnistaa ja käytön historia täytyy tallentaa. IoT-Ticketissä täytyy luoda jokaisen itsenäisen käyttäjän omat tunnukset ja salasana. Käyttäjänä voi tietysti olla myöskin työryhmä, kuten alkuvaiheessa meillä oli yhteiset tunnukset kaikille TAMK-käyttäjille. Tietoa keräävät sensorit yhdistetään IoT-Ticketissä käyttäjän mukaan järjestelmään, joten käyttäjän tunnistaminen on oleellinen asia. Käyttäjähistorian tallennus ei ollut osana tutkimustamme.

Jos oppilaitoskäytössä käyttäjä (käyttäjä, jolta kerätään mittausdataa) tunnistetaan esimerkiksi nimellä, syntyy henkilörekisteri. Tällöin mittausaineistosta syntyy tietosuojan piiriin kuuluva sensitiivinen aineisto. Tällöin käyttäjältä on hyvä pyytää lupa tietojen käsittelyyn oppimisympäristössä erilaisissa koulutukseen liittyvissä asioissa. Suosituksena on, että dataa kerättäisiin anonymisti, jolloin vältetään usealta yksilön tietosuojaan liittyvältä riskiltä.

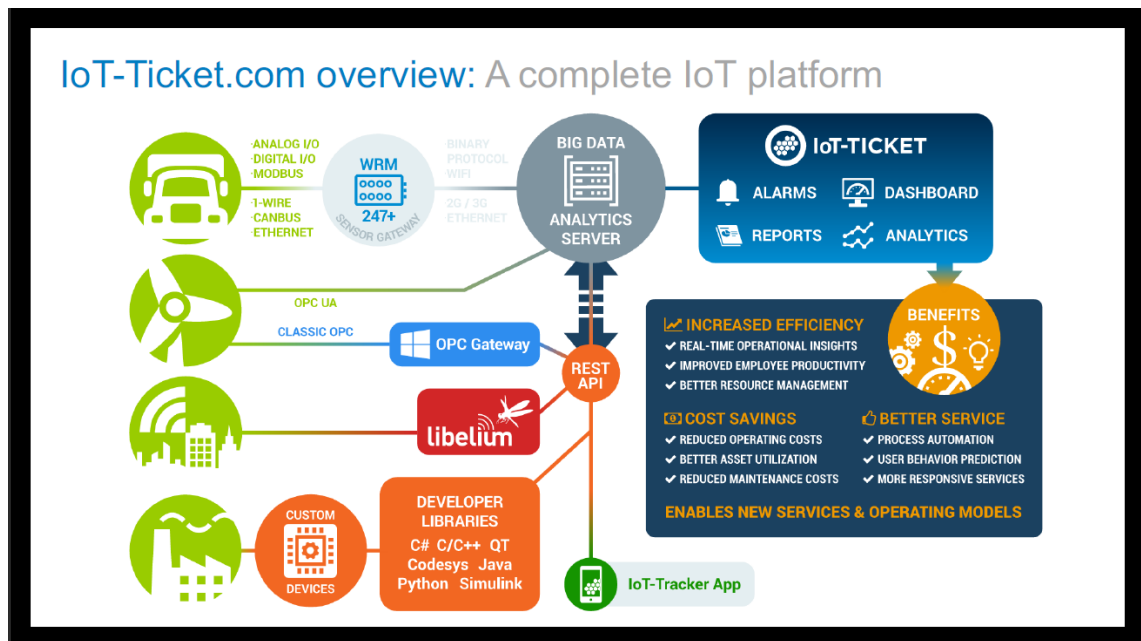
8) Ulkoiset rajapinnat ja sovelluskehityksen työkalut

On tärkeää olla valmiina tarvittavat rajapinnat ulkoisiin tietojärjestelmiin nykyisiä ja tulevia ratkaisuja ajatellen. Integrointi toisiin järjestelmiin pitäisi olla mahdollisimman helppoa, jotta uusia järjestelmiä halutaan ylipäättään ottaa osaksi kokonaisuutta. Kuviossa 7 on esitelty eri ulkoisia rajapintoja, joiden kautta sensorin liittäminen IoT-Tickettiin on mahdollista. Huomasimme jo IoT-Ticketin ensiesittelyssä, että SoTe puolen ratkaisuja varten ei ole HL7- tai sitä uudempaa sopivaa rajapintaa. Ilman HL7-rajapintaa tietoa ei voi välittää suoraan Omakanta-arkistoon. SoTe puolelle ei ole vielä kehitetty omaa toiminnanohjausjärjestelmää IoT-Ticketissä. Koska tällä hetkellä teemme vielä ratkaisua terveydenhoitojärjestelmien määräysten ulkopuolella, niin emme katsoneet HL7-rajapinnan puutteen estävän alustan valintaa käyttöömmee. IoT-Ticketin suurin vahvuus on sen helppokäyttöisyys ja soveltuvuus useille toimialoille.

Libelium-sensorit ovat rajapinnaltaan yhteensopivia IoT-Ticket kanssa, kumppanuus heidän kanssaan on solmittu. Tarvittaisiin kuitenkin vielä kehittää applikaatio, joka osaisi keskustella BT-antureiden kanssa (BT majakat). Koska käytimme tiedonsiirtoratkaisusamme Libeliumin toimittamia sensoreita, niin tämä mahdollisuus auttaa seuraavan vaiheen kehittäjää.

Microsoftin Kinetic liikeohjaimissa on käytössä avoin rajapinta, joten sen välittämä tieto on mahdollista toimittaa IoT-Ticketille. Ajatellen alkuperäistä tavoitettamme kehittää etäkuntoutuksen seuranta, niin tämä on tärkeä mahdollisuus. Hahmontunnistus on nykyisin useimmiten toteutettu juuri Kineticin liikeohjaimilla.

Analytics- ja TAMK-palvelin tukevat Rest-rajapintaa. Käytettävissä on myös kattava määrä muita rajapintoja (Canbus, HTML5, Javascript), joiden olemassaolo mahdollistaa kattavan yhdistämisen eri käyttöjärjestelmää käyttäviin laitteisiin, kuten Windows, iOS- ja Android.

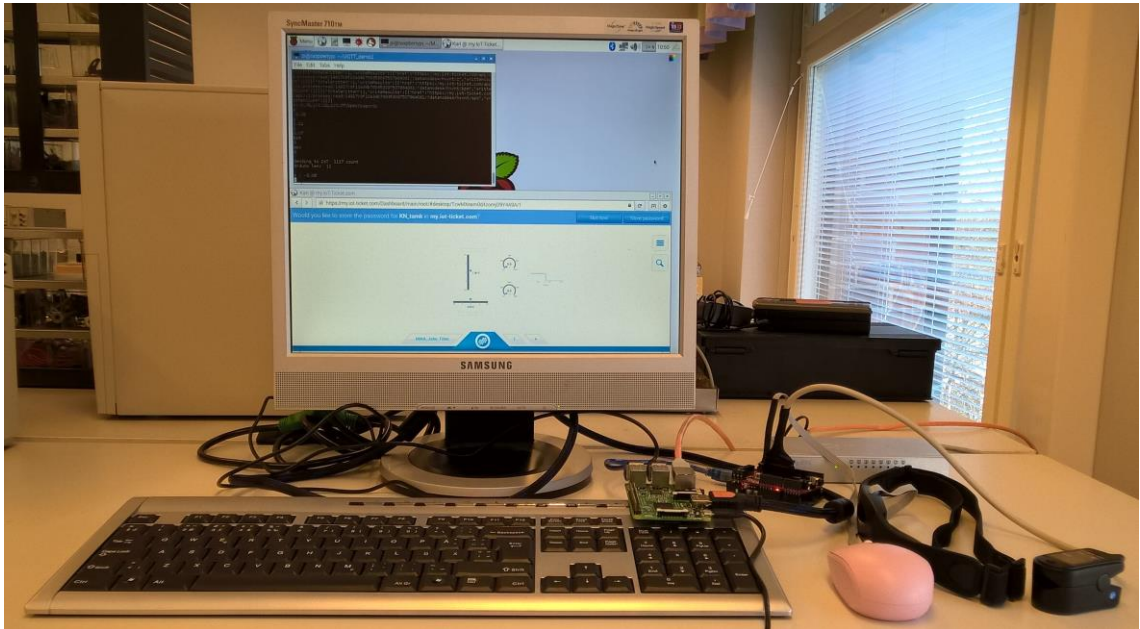


KUVIO 7 IoT-Ticket sovellusalusta (Wapice 2016)

Valintamme sensorien tuottaman tiedon analysointiin oli siis IoT-Ticket (iot-ticket.com) ja samalla siitä tuli ensimmäinen käyttöliittymämme. Eri TAMK-koulutuskokonaisuudet päättivät ottaa myös IoT-Ticket:n kokeiluihin ja käyttöön, joten meillä oli käytössä koko koulun yhteinen alusta sensoritiedon vastaanottamiseen. Wapice lupasi TAMK:n käyttöön alustan ja tiedonsiirtoon tarvittavia välineitä sekä tuotteita, kuten 2kpl VMR247+. (Mäkelä 2016)

4.2 Tiedonsiirto sensoreista IoT-Ticketille

Päätimme toteuttaa referenssiratkaisun IoT-Ticket avulla. Siitä tulisi samalla käyttöliittymä ratkaisuumme. Oheisessa kuvassa 14 näkyy toteutuksemme kokonaisuus.



KUVA 14. Tiedonsiirtoratkaisumme osaset (kuva: Mika Helonsalo 2017)

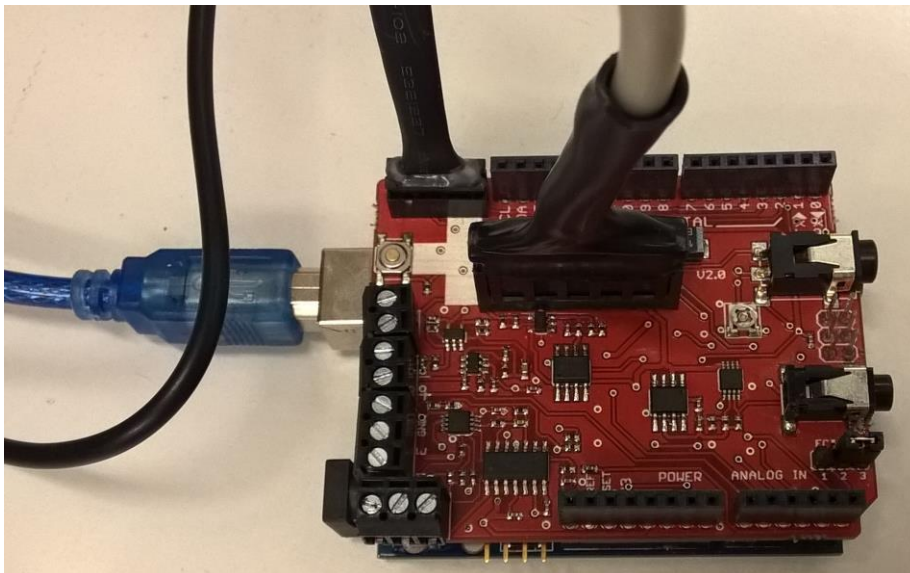
4.2.1 Sensoritiedon kerääminen

Mika Helonsalo & Juho Kauppi & Timo Salminen

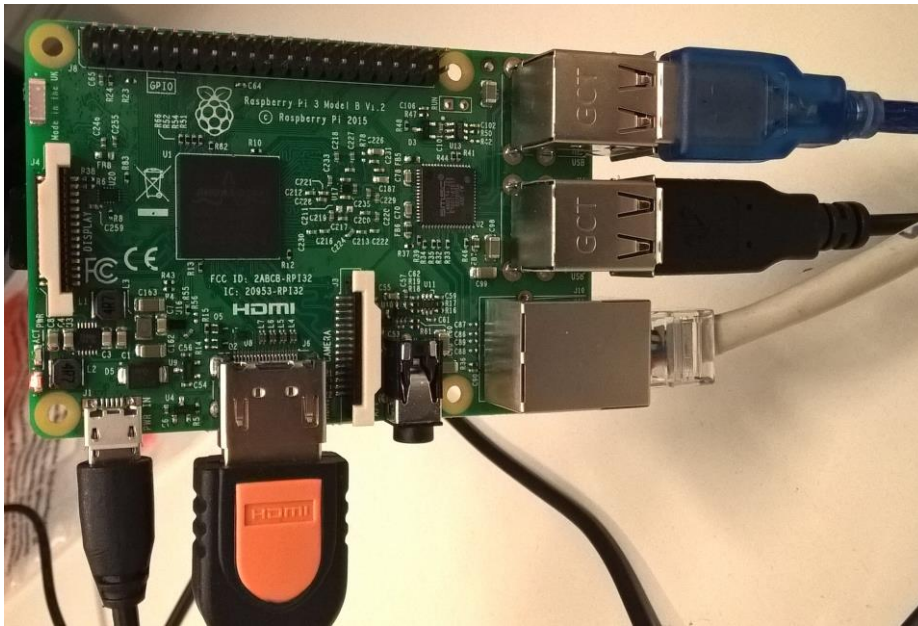
Marraskuussa saimme avuksemme IT-osaston oppilaan Tommi Perälän. Hänen piti tehdä mikrokontrollereihin liittyvä harjoitustyö, joka voisi hyödyttää samalla meidän opinnäytetyön etenemistä. Tätä osiota ei oltu vielä suunniteltu tehtävän lokakuussa tehdyssä työsuunnitelmassa, mutta tiedonsiirron toteutus e-Health-Kit sensoreiden ja mikrokontrollereiden avulla näytti tukevan hyvin päämäärämme oppimisympäristön kehittämisestä. Yhdessä aloimme miettiä toteutusta sille, että mitenkä sensoreista saatava tieto välitettäisiin ensin Arduinin mikrokontrolleriin (KUVA 15) ja sitten Raspberry Pi 3 (KUVA 16) kautta verkkopohjaiseen käyttöliittymään. Tommi kirjoitti tarvittavan tiedonsiirto-ohjelmiston Arduinolle (etäk_arduino.ino, liite 1). Sensoreiksi valitsimme kaatumis- ja SpO2-anturin. Tiedonkeruun sykliksi päätimme valita 1 sekuntia, koska pulssi mitataan tällä jaksolla. Sekä liikeseurorin että SpO2:n keräämä data välittyi hienosti näytölle, joten pys-

tyimme ainakin toteamaan, että ratkaisu on mahdollista saada toimimaan. Kaatumisanturin toimintaa verifioitaessa syklin tulisi kuitenkin olla huomattavasti nopeampi, että voimme olla varmoja tiedon tarkkuudesta, sekä oikeellisuudesta. Kaatumisanturissa olevan kiihtyvyyssanturin suurin mahdollinen päivitystaajuus on noin 800 Hz.

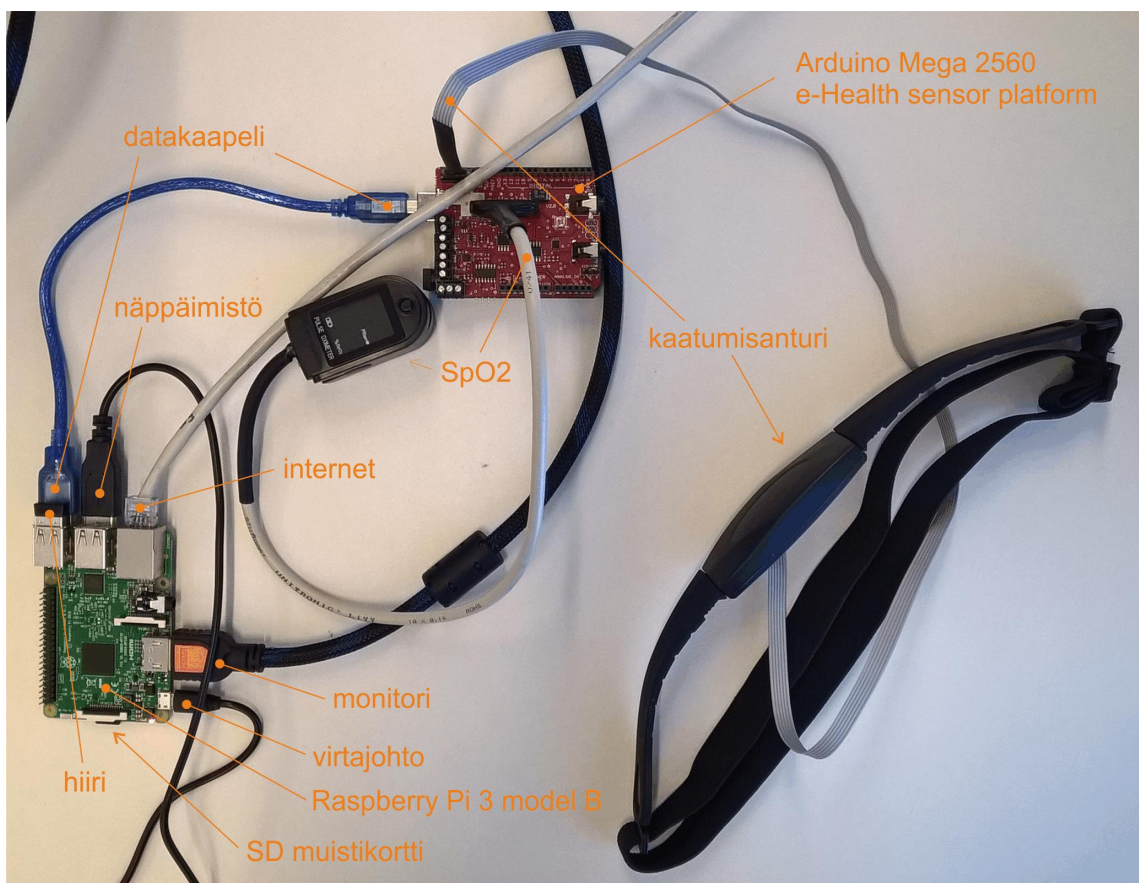
Koska käytössämme oli vain jo koululle hankittuja laitteistoja, niin tiedonsiirtoratkaisutamme tuli täysin langallinen (johdollinen). Syksyllä haaveenamme oli vielä toteuttaa langaton etäkuntoutuksen seurantaan soveltuva ratkaisu. Toteuttamamme ratkaisu soveltuu kuitenkin hyvin oppilaiden kehitysympäristöksi. Työhömmme käytettävissä oleva aika ja tiedonsiirtojärjestelmän keskeneräisyys (vasta kehityksen alussa) mahdollistivat nyt vain toteutuksen rakentamisen SpO₂- (happisaturaatio ja pulssi) ja kaatumisanturin (liikesensori) tiedon keräämiseksi. Kuvassa 17 on esitelty tiedonsiirtoratkaisumme liityntärajapinnat. Työmmme pohjalta jatkavat voivat lisätä järjestelmään uusia sensoreita, kehittää käytettävyyttä ja miettiä mahdollisuuksia toteuttaa tiedonsiirto langattomasti. Bluetoothia hyväksikäyttäen tieto voisi siirtyä langattomasti helposti päätelaitteeseen, kännykkään tai tablettiin. Jatkokehityshankkeita ajatellen perehdyimme teoriaosuudessamme erittäin laajasti myös käyttöjärjestelmän suunnitteluun, kehitykseen ja testaukseen. Seuraavissa hankkeissa joutuu myös pohtimaan lisää asiakastiedon turvallisuuteen liittyä ratkaisuja, joten siitäkkin hankimme jo tuleville kehittäjille varsin laajan tietopaketin.



KUVA 15. Arduino ja e-Health-sensor-platform liitynnät SpO₂ ja liikesensori (kuva: Mika Helonsalo 2017)



KUVA 16. Tiedonsiirron liityntäratkaisu Raspberry Pi 3 (kuva: Mika Helonsalo 2017)



KUVA 17. Liityntärajapinnat SpO2 ja Kaatumisanturi tietojen välittämiseen (kuva: Mika Helonsalo 2017)

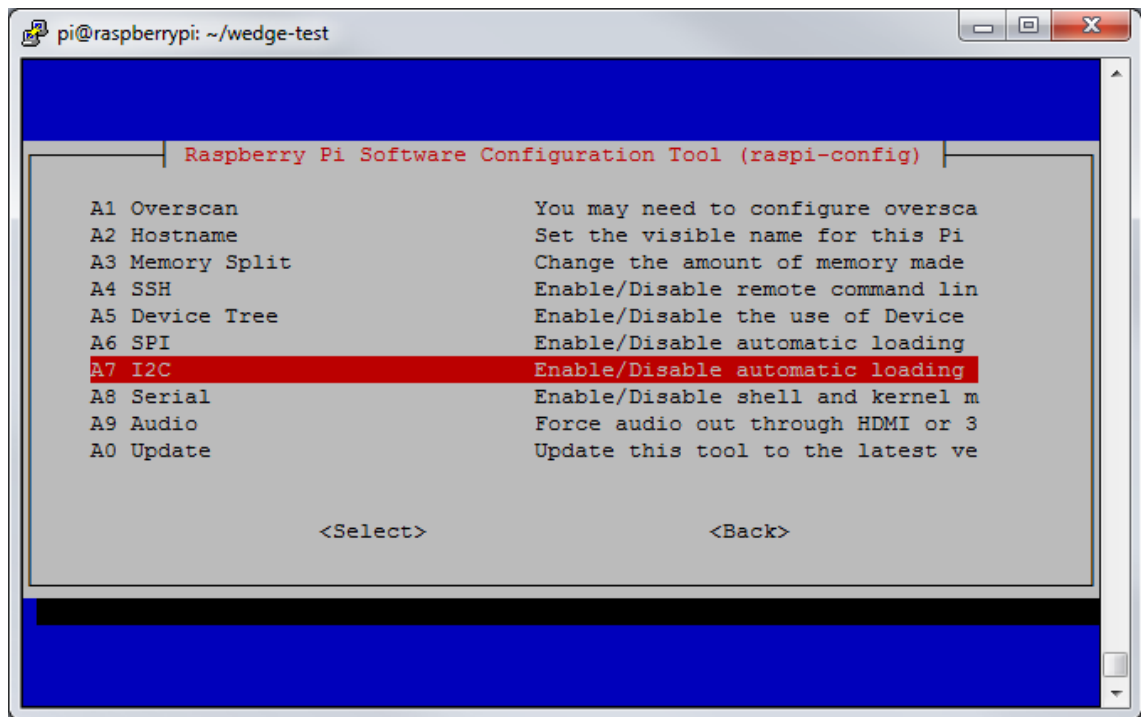
4.2.2 Tiedonsiirron ohjelmistoratkaisu

Timo Salminen

Koska ensimmäisen projektityön tuloksena tieto ei vielä kulkenut IoT-Ticketille asti, niin ratkaisumme toteuttamisen yhdeksi päätehtäväksi jäi toteuttaa toimiva ohjelmisto Raspberry:lle, tiedon saamiseksi Arduinolta ja välittämiseksi Wapicen serverille. Päätimme käyttää pohjana ensimmäisessä projektityössä ollutta ratkaisua liike- ja SpO2-sensoreiden kanssa. Tässä vaiheessa TAMKilla ei ollut vielä käytettävissä omaa IoT-ticket palvelinta tiedonkeruuta varten, joten kaikki keräämämme tieto meni Wapicen omalle palvelimelle ja sieltä saimme sen näkyviin IoT-Ticket käyttöliittymällämme. Wapice lupasi toimittaa myös tarvittavaa lähdekoodia (Source Code) toteutustamme varten, mutta tämä lupaus ei toteutunut työmme aikataulussa.

Sensorilta tieto välittyi Arduinon mikrokontrolleriin, joka ottaa tiedon vastaan analogisena. Tieto muunnetaan digitaaliseen muotoon Raspberryllä ja muotoillaan sopivaksi lähettää IoT-Ticket serverille. Koodi on kirjoitettu C++ ohjelmointikielellä. Raspberry Pi toimintakuntoon saamiseksi saimme image-koodin Kari Naakalta. Tämän jälkeen piti varmistaa, että liikesensorista ja SpO2:sta keräämämme tieto välittyisi oikeanlaisena ja oikeaan osoitteeseen Wapicen palvelimella. Lisäksi sensoreiden tieto piti saada näkyvään KN_Tamk käyttäjänimen alla IoT-Ticket Dashboardilla.

Raspberry Pi:llä halusimme varmistua, että toteuttaisimme ainakin kaksi mahdollista tiedonsiirtotapaa. Tiedonsiirtotapoina käytimme USB – sarjaliityntää, sekä mahdollistimme I2C – väylältä tiedonsiirron Wapicen Iot-Ticket näkymään. I2C-väylän toimimiseksi, Raspberry Pi:lle tarvitsee asentaa kirjasto, jotta tiedon luku onnistuu Arduinolta tai muulta laitteelta, joka toteuttaa tiedon lähettämisen I2C-väylään. Raspberry Pi:ltä tulee ensiksi aktivoida I2C-väylä ennen kuin tiedonsiirto on mahdollista (KUVA 18).



KUVA 18. Raspberry Pi I2C väylän aktivointi (learn.sparkfun.com 2017)

I2C-väylän aktivoimisen jälkeen tulee I2C-rajapinta asettaa toimintakuntoon. Raspberry Pi:lle tulee hakea tarvittavat kirjastot kirjoittamalla Raspberry Pi:n komentoriville komento: `sudo apt-get install -y i2c-tools`. (I2C Interface for Raspberry Pi 2016)

Päätimme kuitenkin käyttää USB-väylää tiedonsiirron välittämiseen Arduinolta Raspberlylle ja sieltä IoT-Ticketille. (Wiring Pi 2017)

Raspberry Pi:lle on saatavilla Wiring Pi kirjasto (Wiring Pi Library 2017), jonka avulla lukeminen USB-sarjaväylästä onnistui helposti. Tämän kirjaston avulla voidaan liittää mikä tahansa laite USB-sarjaväylään ottamatta huomioon millaisia laitteet olisivat. Wiring Pi kirjaston avulla luettaessa tulee ottaa huomioon avattavan kohteen osoite: `"/dev/ttyUSB0"` sekä siirtonopeus (baudrate), joka täytyy olla sama lähettävän laitteen siirtonopeuteen. Jos siirtonopeus on eri, tulee tiedonsiirtoon huomattavasti virheitä ja tiedon vastaanottaminen ei ole tämän jälkeen luotettavalla tasolla. Omassa toteutuksessa tulimme siihen tulokseen, että haluamme päästä vähintään yhden sekunnin läheystaajuuteen, jolloin tiedot luettiin Arduinolta kerran sekunnissa. Arduino puolestaan lähettää tiedon kerran sekunnissa, jolloin Raspberry Pi:n tulee varmistaa, että kaikki tieto on vastaanotettavissa käynnistettäessä.

Onnistuimme ensimmäisen kerran siirtämään pulssitiedon onnistuneesti Dashboardille 9. helmikuuta 2017. Ratkaisumme toimivuus vaati vielä kuitenkin käytettävyyden ja tiedonsiirron ratkaisun parantamista, joten olimme tyytyväisiä Dashboardille saamamme tiedon laatuun vasta 15. helmikuuta tekemämme kokeilujen jälkeen. Samalla oli tarkoitus lähettämään myös päivätasolla raportteja omiin sähköposteihimme, mutta tämä ei onnistunut, koska käyttäjätunnus KN_tamk mahdollisti lähetyksen osoitteeksi laitettavan vain koulun henkilökunnan osoitteita.

Käyttöohjelmistomme Raspberry Pi käyttöön on main.cpp (liite2) ja se on tallennettuna muistikortilla hakemistoon MQTT_demo2. Sensoritietojen lähettämiseksi Wapicen serverille on luotu ohjelma SendWap.c (liite3). Hyvinvointi-ohjelmamme käynnistyy komennolla ”./hyvinvointi”. KUVA 19 esittelee ohjelmistomme käynnistämisen vaiheita.

```

pi@raspberrypi: ~/MQTT_demo2
login as: pi
pi@10.10.206.136's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Mar  7 12:41:19 2017
pi@raspberrypi:~ $ ls
2017-02-15-173748_1280x1024_scrout.png  MQTT_temp_t1  test
Desktop                                Music          Timer1
Documents                              Pictures       tw
Downloads                              pin_int1      tw_test_0
example.c--libcurl                    Public         tw_test_sensors
IoT-Py                                 python_games  Videos
IoT-tic                                rpi-clone    wapice
mosquitto-repo.gpg.key                 saie          wap_test_0
MQTT_demo                              serial_read.py wiringPi
MQTT_demo2                             Templates

pi@raspberrypi:~ $ cd MQTT_demo2
pi@raspberrypi:~/MQTT_demo2 $ ls
curl_komento  Makefile_temp  mqtt_arduino1.cpp  SendTW.c  SendWap.h
main.cpp      Makefile_wap   mqtt_arduino1.h   SendTW.h  SendWap.o
main.o        Makefile_wap_1 mqtt_arduino1.o   SendTW.o
Makefile      mqtt           orig              SendWap.c
pi@raspberrypi:~/MQTT_demo2 $ ./mqtt

```

KUVA 19. Raspberry Pi käyttöohjelmat löytyy hakemistosta MQTT_demo2

```

pi@raspberrypi: ~/MQTT_demo2
File Edit Tabs Help
Sending to IoT 53 count
x:-0.07,y:1.01,z:-0.14;bpm:85;spo:96;
Sending to IoT 54 count
x:-0.07,y:1.01,z:-0.15;bpm:85;spo:96;
Sending to IoT 55 count
x:-0.07,y:1.01,z:-0.14;bpm:86;spo:96;
Sending to IoT 56 count
x:-0.07,y:1.01,z:-0.14;bpm:87;spo:96;
Sending to IoT 57 count
x:-0.07,y:1.02,z:-0.14;bpm:87;spo:96;
Sending to IoT 58 count
x:-0.07,y:1.01,z:-0.14;bpm:88;spo:97;
Sending to IoT 59 count
x:-0.07,y:1.01,z:-0.14;bpm:88;spo:97;
Sending to IoT 60 count

x:-0.07,y:1.01,z:-0.15;bpm:0;spo:0;
x
-0.07
y
1.01
z
-0.15
bpm
0
spo
0
Sending to IoT 8 count
Arduino len: 11

```

KUVA 20 ja 21. Liikesensorin x-, y- ja z-akselin suuntainen kiihtyvyys sekä SpO2 pulssi (bpm) veren happipitoisuus (spo)

Kiihtyvyysanturi ja SpO2 toimivat hyvin kehittämämme ohjelmamme kanssa, tieto välit-
tyi hyvin pienellä viiveellä katseltavaksemme. KUVA 20 ja kuva 21 esittää tätä tiedon-
siirtoa. Kiihtyvyyssi tieto muuttuu jatkuvasti, kun taas pulssi ja happisaturaatio muuttuvat
hitaammin. Kokeilimme määrittää viivettä lyhemmäksi ja pitemmäksi, jotta voimme to-
deta tiedon siirtyvän luotettavasti IoT-Ticketille. Suurempi viive takasi luotettavimman
tiedon välityksen IoT-Ticketille. IoT-Ticketille nopeaa lähetystaajuutta käytettäessä
emme kuitenkaan saaneet käyttöömmme riittävän reaaliaikaista tietoa kiihtyvyysanturista.
Mietimme tätä ongelmaan ja arvelimme tämän johtuvan IoT-Ticketin päivittämistaajuu-
desta ja kuinka nopeasti IoT-Ticket pystyy vastaanottamaan tietoa. Kaatumisanturissa
olevan kiihtyvyysanturin suurin mahdollinen päivitystaajuus on noin 800 Hz. IoT-Ticke-
tin tiedon vastaanottamistaajuuden konfigurointi onkin yksi tulevista kehityskohteista ja
kuinka tämä mahdollistetaan.

Tiedonsiirto Raspberry Pi:ltä IoT-Ticketille tapahtuu libcurl-kirjastoa käyttäen. Libcurl
rajapinta on helposti otettavissa käyttöön, varsinkin synkroninen viestien lähetys on te-
hokas tapa välittää tieto IoT-Ticketille. Tärkeimpiä asetuksia mitä tulee ottaa huomioon,
on itse URL – osoite minne tieto tulee lähettää. URL:n lisäksi tärkeitä konfiguroitavia
parametrejä ovat käyttäjänimi ja salasana, joilla yhteys voidaan muodostaa kyseiseen

IoT-Ticket palveluun. URL:iin täytyy muistaa lisätä myös laitekohtainen ID, jotta yhteydenmuodostus onnistuu. Yksi tärkeimmistä parametreista, joka pitää asettaa, on tiedon siirron välitystapa. Välitystapana käytetään tässä tapauksessa JSON-viestin rakennetta (w3schools, JSON). Tietorakenteemme muodostuu arvopareista, esimerkiksi:

```
{  
  "name" : "spo",  
  "path" : "HVVNT",  
  "v": "97",  
  "unit" : "spo"  
}
```

Tieto lähetetään sellaisenaan IoT-Ticketille, joka osaa parsia tiedon suoraan käytettäväksi ja näytettäväksi IoT-Ticket palvelussa (Libcurl).

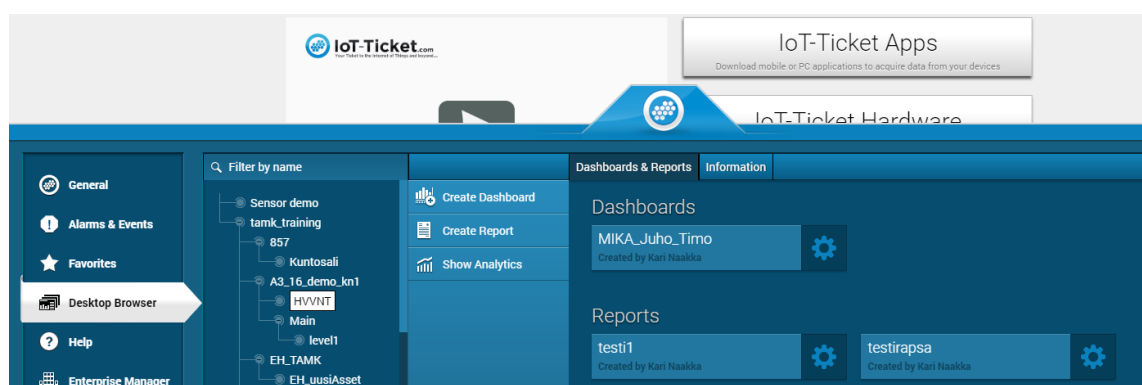
4.2.3 IoT – Ticket UI

Mika Helonsalo & Juho Kauppi

Valitsimme IoT-Ticket käyttöliittymäksemme sekä tiedon keräämiseen, analysointiin, että havainnollistamisen työkaluksi. Haasteeseen liittyviä tutkimuskysymyksiämme tässä vaiheessa olivat:

- Kuinka sensorien välittämä data saadaan näkymään IoT-Ticketissä?
- Kuinka Dashboard näkymä luodaan tiedon havainnollistamiseksi?
- Kuinka saadaan luotua raportteja kerätystä tiedosta?

Jokainen meistä voi luoda oman IoT-Ticket tilin pyytämällä tunnuksia Wapicelta. Käytimme tiedonsiirtoharjoituksessa TAMKille yhteisiksi luotuja tunnuksia. IoT-Tracker kokeilua varten oli käytössä henkilökohtainen kirjautuminen, mutta tätäkin kautta saatava sensoritieto olisi voitu yhdistää KN_tamk tunnusten alle kerättyihin tietoihin. IoT-Ticket kokeilutunnukset voi luoda osoitteessa my.iot-ticket.com/Dashboard/main/root. TAMK sai myös oman IoT-Ticket palvelimen käyttöönsä helmikuussa, vähän ennen käytännön osuutemme valmistumista. TAMK IoT-Ticket serverin käyttöä emme täten sisällyttäneet enää harjoitukseemme. Käytettävillä kirjautumistunnuksilla on merkitystä sen takia, että löydämme tarvitsevamme sensorit ja niiden välittämän tiedon esitettäväksi Dashboardilla. Kuva 22 havainnollistaa käytössä olevien sensorien etsimistä käytettäväksi tiedon lähteenä. Etämittausratkaisumme sensoritiedolle annoimme nimeksi HVVNT.

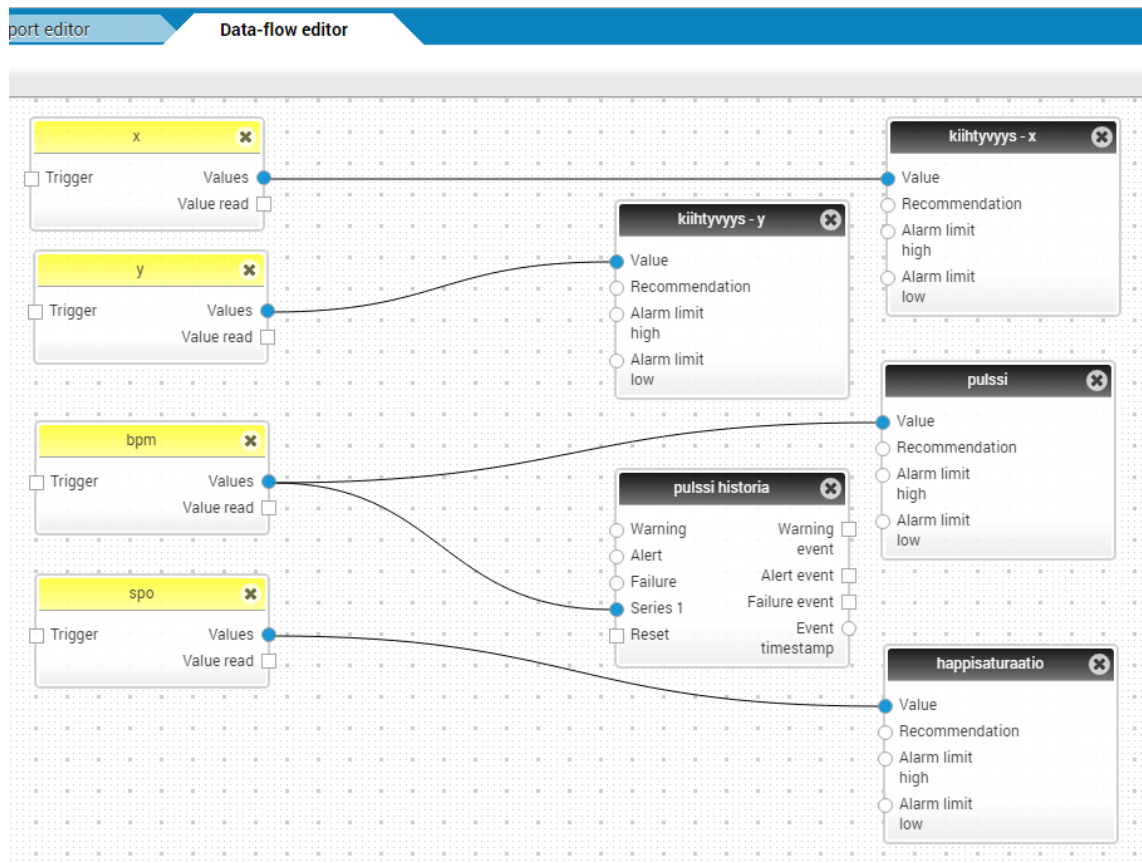


KUVA 22. Kuvakaappaus, näkymä sensoriimme HVVNT IoT-Ticket Dashboardilla

Kun olimme kertoneet haluavamme järjestää seurannan HVVNT sensoreille, niin sitten siirrytään valitsemaan mittareita esitettäväksi Dashboardilla. Ratkaisullamme pystyi mit-

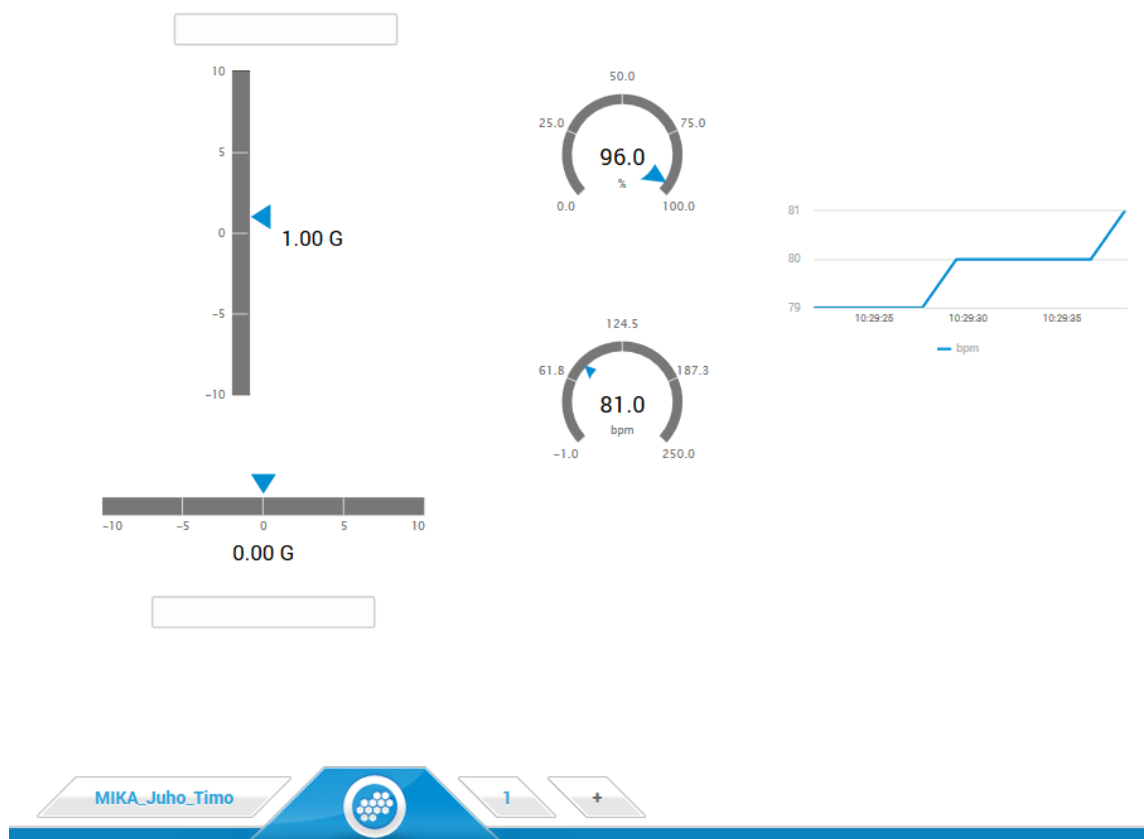
taamaan kiihtyvyystietoja 3-akselin suhteen, pulssia ja happisaturaatiota. Näiden esittämiseen soveltuisi lineaariset ja pyöreät mittarit. Kiihtyvyyden esittämiseen päätimme käyttää lineaarisia mittareita. Pulssin ja happisaturaation esitykseen näytti parhaiten sopivan pyöreät mittarit.

Dashboardille luodut mittarit eivät toimi ilman yhdistämistä oikeanlaiseen sensoritietoon. Tämä yhdistäminen tapahtuu Data-flow editorissa ja sitä on havainnollistava kuva 23. Keltaisella palkilla varustettuna näkyvät saatavilla olevat sensorit, kiihtyvyys x- ja y-akseli & pulssi (bpm) & happisaturaatio (spo). Sensorit yhdistetään luotuihin mittareihin, jotka näkyvä oikeassa reunassa mustalla palkilla. Kummallista oli, että Dashboard näkymästä ei voi suoraan luoda raporttia. Raportti ja Dashboard näkyvät on luotava kumpikin erikseen, vaikka tarvittavat toimenpiteet molemmille on samat.



KUVA 23. Kuvakaappaus, Data-flow editor. Sensoreiden ja mittareiden yhdistäminen.

Kun halutut tiedot on saatu yhdistettyä oikeanlaisiin mittareihin, niin sen jälkeen voidaan vielä lisätä mittareiden luettavuutta. Esille voidaan laittaa mitattavat asiat lisätekstin ja merkitsemällä suuret näkyviin, kuten happisaturaatio-% ja kiihtyvyyden arvo G. Kuva 24 havainnollistaa näkymää Dashboardilla. Jostakin syystä näitä toimenpiteitä ei aina saanut suoritettua samalla tavalla, joten jätimme seurantapohjat yksinkertaisiksi. Mitatuista suureista voidaan luoda raportteja halutulla aikavälillä lähetettäväksi, esim. päivä tai viikkoraportti. Raportin luontia käydään läpi hieman lisää seuraavassa kappaleessa 4.3.

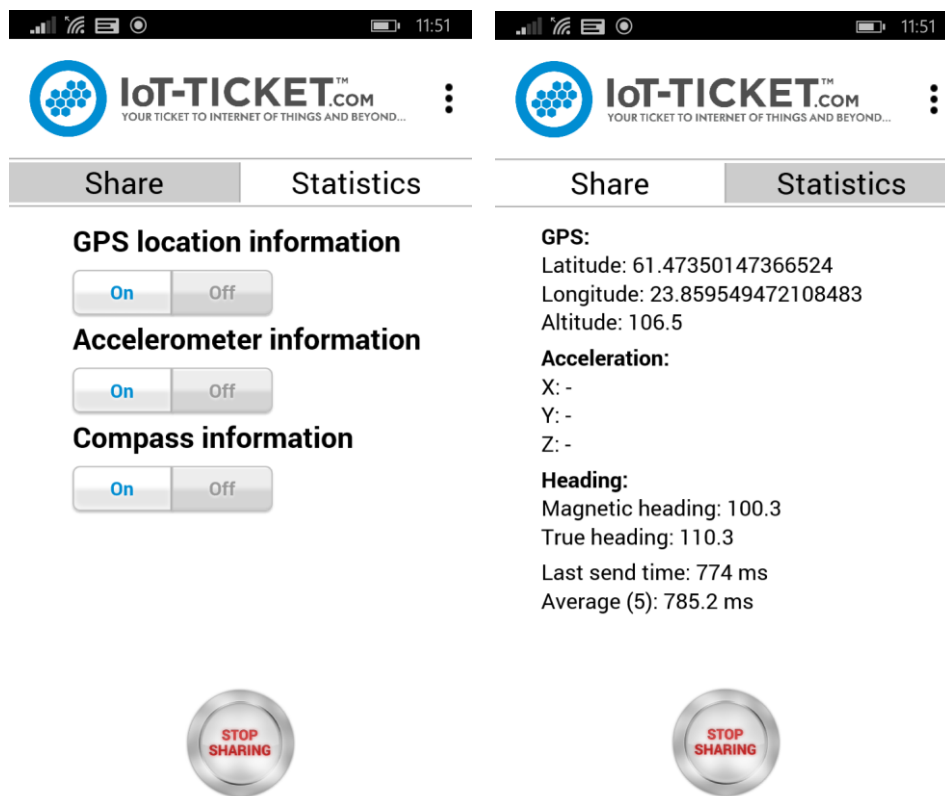


KUVA 24. Kuvakaappaus, Dashboard näkymä, vasemmalla kiihtyvyys ja SpO2. Oikeassa reunassa on pulssin seurantanäkymä

4.3 IoT-Ticket matkapuhelimessa

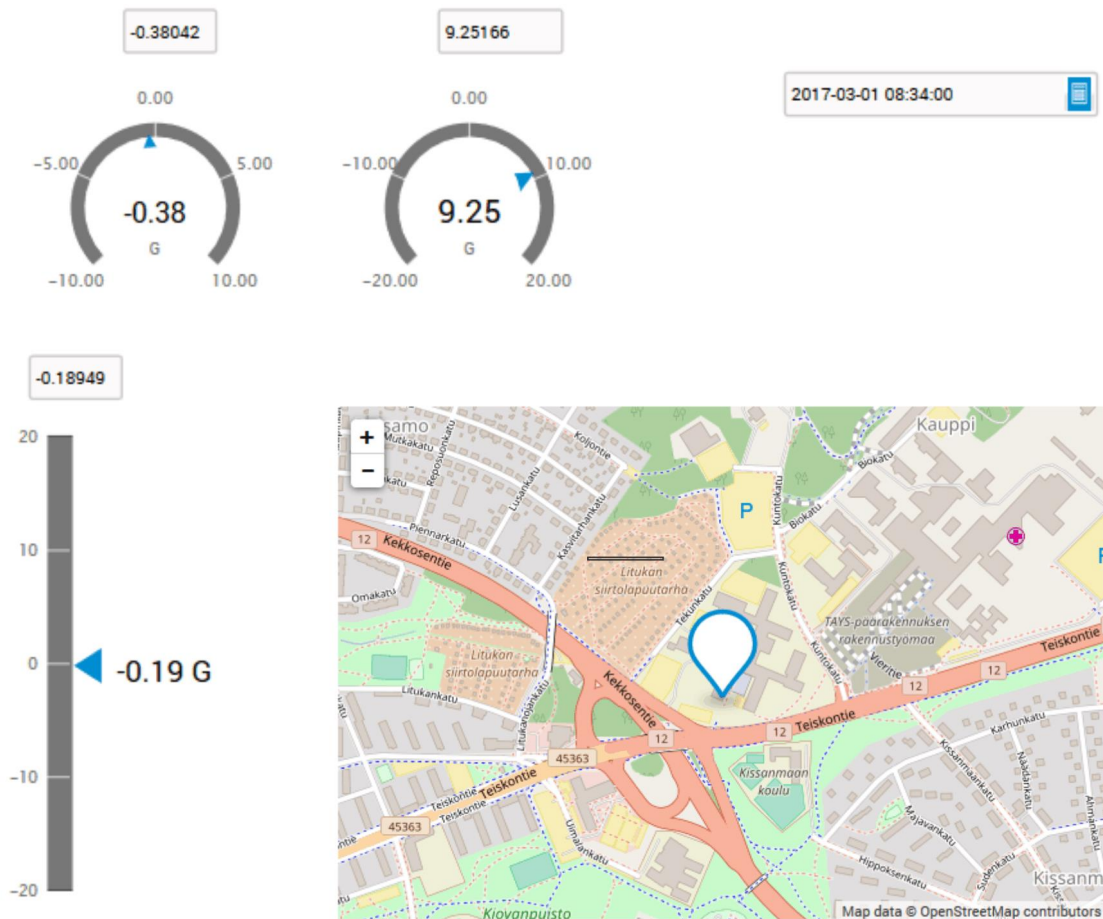
Mika Helonsalo

Kun emme saaneet Wapicelta heti käyttöömme aloituspakettia tai muita tarvikkeita, kuten WRM 247+ sensoriväylää, niin ajatuksena oli lähteä tutkimaan tarkemmin älypuhelimessa toimivaa IoT-Tracker sovellusta. IoT-Tracker mobiilisovelluksen voi ladata Windows- ja Android matkapuhelimelle sekä tableteille. Tämä on varsin yksinkertainen tapa toteuttaa esim. henkilön seuranta puhelimessa olevien sensoreiden avulla. Laitteessa sijaitsevien sensoreiden välittämät paikka, lämpötila, liike (kiihtyvyys), kompassi jne. tiedot saadaan välitettyä IoT-Ticket Dashboardille ja raporttiin. Seurannan alla oleva ihminen voi tietysti päättää, että lähettääkö hänen laitteensa tietoja tietokantaan, lähetys täytyy laittaa aina erikseen päälle. Seurantamahdollisuuksien määrittystä havainnollistaa kuva 25 ja 26. (Iot-ticket.com)



KUVA 25 ja KUVA 26. Kuvakaappaus, Puhelimen IoT-Ticket sovellus (kuva: Mika Helonsalo 2016)

IoT-Tracker kautta saatavaa sensoritietoa havainnollistetaan Dashboardilla ja luomalla raportteja. Mikan puhelimen keräämän datan päivärapportti on nähtävillä kuvassa 27. Raportti kuvaa hetkellistä tilannetta päivän aikana.



KUVA 27. Kuvakaappaus, Raportin voi lähettää PDF muodossa sähköpostiin (kuva: Mika Helonsalo 2017)

Paikannusratkaisu toimi ainakin sen verran hyvin, että oman puhelimensa sijainti päivittyy Dashboardille melko pian yhdistämisestä. Kompassin suuntimatietoja ei tuntunut pystyvän käyttämään mitenkään järkevästi hyödyksi. Ehkä tilanne on toinen, jos toinen henkilö pystyy tarkkailemaan puhelimen kanssa liikkuvaa reaaliaikaisesti hieman pidemmällä aikavälillä. Puhelimen kiihtyvyyssanturit vaikuttivat toimivan paremmin IoT-Ticket kanssa kuin käyttöön saamamme kaatumisanturin. Johtuen kait tiedonsiirron viiveestä Wapicen tietokantaan, emme pystyneet toteamaan esim. pystysuoran kiihtyvyyden (putoaminen) suoraan Dashboardilta lukeman muuttumisena. Kun TAMK saa oman IoT-ticket serverinsä, niin puhelimen sensoritiedon käyttömahdollisuutta kannatta tutkia lisää.

Yritimme myös liittää Fitbit-aktiivisuusrannekkeen puhelimen kautta näkymään IoT-Tracker sensorina. Valitettavasti tämä ei onnistunut ja myös Wapicelta tuli vastaus, että tällainen ei ole mahdollista. Tästä heräsikin ajatus, että tarvitsemme applikaation jolla voi yhdistää omien käytössä olevien sensoreiden / laitteiden tietoa tarkkailtavaksi itselle. Uusimmilla ja kalleimmilla älypuhelimilla voi jo mitata verensokeria, pulssia, käveltyjä askelia, määritellä oma painoindeksi ja arvioida kokonaishyvintia. Siitä ei työryhmälämme ole kokemusta, että saako tällaisen puhelimen sensorin mittamaan pulssin näkymään IoT-Ticket Dashboardilla. Tästäkin seuraava oppilastyö voi ottaa selvää.

4.4 Terveysdatan mittaaminen ihmisestä

Mika Helonsalo

Tässä osiossa käydään lävitse tarvittavia käyttöohjeita työssä mukana olleiden sensorien ja mittarien käyttämiseksi. Havaitimme tämän selvityksen tarpeelliseksi, koska insinööreinä meillä ei ollut tarpeeksi kokemusta suoritettavista terveyden itsemittauksista. Lisäksi tarvitsimme myös lisää tietoa siitä, että miksi on kyseisiä mittausta kannattaa suorittaa.

4.4.1 Veren happisaturaatio

Happisaturaatiolla eli veren happipitoisuudella tarkoitetaan punasoluihin sitoutuneen hapen suhteellista määrää. Valtimoveren happiosapaine ja veren happisaturaatio liittyvät läheisesti toisiinsa. Happiosapaineen kasvaessa lähestyy happisaturaatio 100 %. (*Lääketieteen sanasto 2016*) Normaali happisaturaatio ihmisellä on vähintään 96 %, lievä happivajaus eli hypoksemia 89-95 %, keskivaikea 80-88 % ja vaikea alle 80 %. 100 % on yleensä merkki hyperventilaatiosta eli ylihengityksestä. Valtimoveren happisaturaatiota mitataan potilaan maataessa tai istuessa pulssioksimetrillä. Mittaus tapahtuu joko sormesta tai korvalehdestä valosäteiden avulla ja se ilmoittaa pikkuvaltimon happikyllästeisyyden. (*HUSlab ohjekirja 2016*)

Veren kuljettamalla happimäärällä on vaikutusta esimerkiksi vireystilaan, hengästyvyyteen ja pulssiin. Syöpäpotilailla veren happisaturaatio on normaalia alhaisempi, yleensä noin 60 % luokkaa. Tähän liittyy Warburgin hypoteesi. Nobelisti Tohtori Otto Warburg havaitsi, että laskemalla kudosten happipitoisuutta 35 % kahden vuorokauden ajaksi, normaalit solut muuttuivat syöpäsoluiksi. Happivajaus saattaa olla seurausta esimerkiksi kroonisesta keuhkosairaudesta, sydänsairaudesta, dekompensoituksesta tai astmakohtauksesta (Warburg).

4.4.2 Verensokeri

Verensokeriarvon seuraaminen on tärkeää diabeetikoille ja heidän tulisi osata suorittaa tämä mittaus itse. Tällä hetkellä on yleistä mitata verensokeriarvo suoraan verinäytteestä, joka otetaan sormenpäistä. Verinäytteen otto vaatii neulalla pistämisen, jolloin sormenpäiden iho tulee kipeäksi ja myös kovettuu. Nykyiset verensokerimittarit ovat kuitenkin helppokäyttöisiä ja nopeita, mutta Verensokerimittarit ilmoittavat verensokeriarvot yleensä plasman glukoosiarvona. (Terve-lehti 2016)

Diabetes voidaan todeta sokerirasituskokeessa tai mittaamalla pitkällä aikavälillä verensokerin arvo laskimoverestä. Terveen ihmisen paaston (syömättä ja juomatta edellisestä illasta lähtien) jälkeen mitattu sokeriarvo on 6.0mmol/L tai matalampi. Diabeetikolla paaston jälkeen mitattu sokeriarvo on 7.0mmol/L tai korkeampi. Mittausyksikkö mmol/L kertoo, että kuinka monta millimoolia glukoosipitoisuus on litrassa verta.

Verensokerin seurannalla on tarkoitus ehkäistä diabeetikoille sattuvia vaaratilanteita. Liian korkeat ja alhaiset verensokerin arvot voivat molemmat aiheuttaa vaarallisia tiloja. Mittaukseen tarvittava pieni verimäärä saadaan iskettyä sormesta tähän tarkoitukseen kehitetyllä neulalla eli lansetilla. Suositeltu verinäytteen ottopaikka on keskisormesta tai nimettömästä sormenpäiden sivulta. Ennen näytteenottoa sormenpää täytyy puhdistaa ja ensimmäinen veritippa pyyhitään pois. Vasta toinen veritippa laitetaan sormesta mittausliuskan näytekohdan päälle. Itse mittautapahtuma kestää noin 5 sekuntia. Mittautulokseen vaikuttaa suuresti käyttäjän kokemus mittarin käytöstä. Diabeetikko suorittaa mitauksia niin useasti, että mittausasuorituksen takia virheitä tapahtuu vain vähän. Itse verinäytteen ottoon voi liittyä seuraavia ongelmia; voimakas puristus sormesta sekoittaa kudostnestettä näytteeseen, kylmät tai likaiset sormet, väärin säilytetyt liuskat, mittausajan kohta eli ajoitus ruokailuihin.

Omaseurantaa helpottamaan on tulossa laitteita jotka eivät edellytä verinäytteen ottamista sormenpäistä. Tällaisessa ratkaisussa iholle kiinnitetään pieni ja pyöreä sensori, josta lähtee säie ihon läpi rasvakudokseen. Sensorin rasvakudoksesta mittaama glukoosilukema (verensokeriarvo) luetaan tähän tarkoitukseen olevalla lukulaitteella. Nykyiset sensorit kestävät myös kastumisen ja niiden toiminta-aika on noin kaksi viikkoa. Lukulaite myös ilmoittaa, että mitenkä verensokeri heittelee seurantajakson aikana. Siitä saadaan myös arvio verensokeriarvon tulevasta suunnasta ja tilannekuvan kerätyistä glukoositiedoista,

kaikki tämä helpottaa hoidon suunnittelua. Ilman mittaamista on vaikea tietää tarvittava (pistettävä) insuliiniannos ja kuinka ateriat sekä liikunta vaikuttavat verensokeriarvoon. Ykköstyypin diabeetikot mittaavat verensokeria 4–10 kertaa päivässä (Koskinen 2016).

Kokeilin suorittaa verensokerimittauksen käytössämme olevalla mittarilla helmikuun alussa, kun käytettävissä oli myös kunnollinen lansetti, joka on esitelty kuvassa 28. Lansetti on neula verinäytteen ottamiseksi sormenpäältä. Valitettavasti mittari ei suostunut laskemaan tulosta, vaan mittaus päättyi aina jonkinlaiseen virheilmoitukseen E-U ja sammui sitten OFF ilmoituksen jälkeen. Verinäyteliuskan ja mittarin asennus on esitelty kuvassa 29. Pyysin myös tuttua lähihoitajaa suorittamaan saman mittauksen, jotta varmistuisin mittaustapahtuman oikeasta suorittamistavasta. Valitettavasti emme onnistuneet yhdessäkään suorittamaan mittausta. Libelium e-Health-Kit paketin mukana ei ollut mittarin ohjekirjaa. Vastaavanlainen Vesta-glukoosimittari löytyy cooking-hacks.com sivustolta, mutta käyttöohjekirjaa mittarille ei löytynyt sieltäkään. Ilman käyttöohjekirjan virhetilanne neuvoja en osaa arvata, että miksi mittaus ei onnistunut. Joko mittari on viallinen tai sen mukana tulleet testiliuskat eivät ole toimintakuntoisia. (Cooking-hacks.com 2017)



KUVA 28 ja 29. Lansetti ja Verensokerimittarin laskenta on käynnissä. (kuvat: Mika Helonsalo 2017)

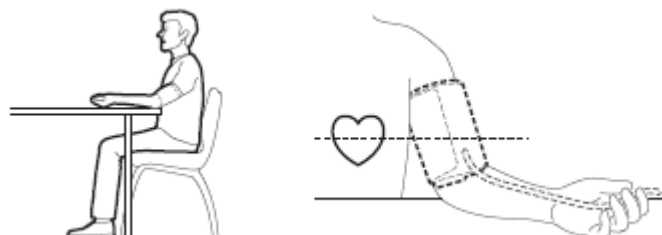
4.4.3 Verenpaine

Oikein tehtynä kotimittaus on hyvä apu verenpaineen seurannassa. Olkavarsimittarit ovat luotettavampia kuin rannemittarit. Rannemittaria käytettäessä on muistettava pitää mittari sydämen tasolla, päällä. Rannemittareita tarjoavat esimerkiksi iHealth & Beurer.

Verenpainemittari ilmoittaa verenpaineen yläarvon ja ala-arvon. Yläpaine (systolinen) ilmoittaa valtimon sisällä olevan paineen sydämen supistuksen aikana. Alapaine (diastolinen) ilmoittaa paineen sydämen lepovaiheen aikana. Verenpaine normaaliarvot ovat alle 130/85 mmHg (millimetriä elohopeaa). Verenpaine on tyydyttävä välillä 130–139/85–89 mmHg ja koholla paineen ollessa vähintään 140/90 mmHg. Kohonnut verenpaine vahingoittaa valtimoita ja aiheuttaa aivohalvauksia ja sydäninfarkteja. (Mustajoki 2015)

Ohje mittauksen tekniseen suorittamiseen olkavarsimittarilla (terve.fi, verenpaine & Koodea, käyttöohjekirja):

- Mittaus tehdään olkavarteen asennettavaa painemansettia käyttäen. Mansetin kumpipussin täytyy olla vähintään 40% ja pituuden vähintään 80% olkavarren mitasta. Yleensä laitteen mukana tulee M-kokoinen mansetti. Väärän kokoinen mansetti aiheuttaa vääriä mittaustuloksia.
- Mittausta edeltävän puolen tunnin aikana vältetään raskaita fyysisiä suorituksia, tupakointia ja kofeiinipitoisten juomien nauttimista.
- Painemansetin tulee sijaita sydämen tasolla, jota on havainnollistettu kuvassa 30
- Kun mittarin mansetti on kiedottu olkavarren ympärille, istutaan rauhassa 5 minuuttia ja mittauskäsi voi levätä pöydällä. Mittauskädeksi valitaan ei dominoiva käsi.
- Verenpaine mitataan kaksi kertaa peräkkäin, ja molemmat mittaustulokset kirjataan seurantatietoihin. Lopullinen tulos on mittausten keskiarvo.
-



KUVA 30. verenpaineen mittausasento (tohtori.fi 2016)

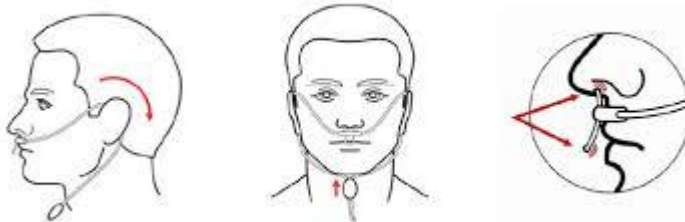
4.4.4 Kaatumisanturi

Kaatumisanturina käytettävä kiintyvyyssanturi voidaan asettaa, vaikka seurattavan rintakehän päälle, joka sijoituspaikkansa puolesta sijaitsee melko vähäisen normaalin liikkeen alueella. Kun potilas kaatuu, niin Kaatumisanturin sovellus tekee hälytyksen IoT-alustalle. Koska kaatumisen määritelmä ei voi olla täysin luotettava, niin sovelluksessa on hyvä olla mahdollisuus perua automaatti hälytys, jos potilas ei tarvitsekaan apua. Kaatumisanturissa voi olla esim. painikkeet väärän hälytyksen peruutukseen ja avun kutsumiseksi paikalle.

Kokeilimme ensimmäisen tiedonsiirtoratkaisun yhteydessä joulukuussa, että mitenkä hyvin saisimme mitattua tietoa näytölle verrattuna heilahduksiin. Aiheutimme kaatumisanturillemme kiihtyvyyssuutoksia liikuttamalla sitä kädessä ylös, alas ja sivulle. Vertasimme tätä tuloksiin, joita piirtyi samaan aikaan puhelimen heilahduksista IoT-Trackerin kautta IoT-Ticket Dashboardille. Näytti siltä, että puhelimesta tuleva tieto reagoi paremmin tapahtumiin, joita aiheutimme. Tämä on tietenkin hieman epävarma tapa vertailla kahden eri sensorin toimintaa toisiinsa, mutta tulimme siihen johtopäätökseen, että e-Health-Kitin mukana tulleen kiihtyvyyssanturin ratkaisu ei ollut paras mahdollinen lähtökohta toteuttaa kaatumisen analysoinnin IoT-Ticket kautta.

4.4.5 Hengitysilma määrä

Hengityksen määrää mitataan esim. kahden karva-sensorin avulla, jotka sijoitetaan potilaan nenään. Mittaava sensori pysyy paikallaan, kun signaalijohdot laitetaan kulkemaan korvien takaa. Sensorin oikea asennusohje on esitelty kuvassa 31. (Cooking-hacks 2017)



KUVA 31. Hengitysilmamäärä mittauslaite sijoittelun ohje

Hengitysilmanmäärän mittausta voidaan käyttää osana potilaan seurantaa esim. tajuttomuustilan aikana. Nenän kautta hengitettyä ilmamäärää seurataan myös uniapnean havaitsemiseksi. Uniapneaa sairastavalla on ominaista hengityksen ilmamäärän väheneminen tai kokonaan pysähtyminen unen aikana. Apnea on ajanjakso, jolloin hengitys pysähtyy vähintään 10 sekunniksi tai pienenee huomattavasti. (Uniapnea.fi)

5 Jatkokehitysajatuksia ja suosituksia

Koska tavoitteenamme oli osaltamme kehittää tulevien hyvinvointiteknologian opiskelijoiden opetusta, niin loimme muutamia kehitysaiheita toteutettavaksi lähitulevaisuudessa. Tampereen ammattikorkeakoululle ostettu e-Health-Kit -paketti osoittautui tutkimuksemme aikana erittäin haastavaksi hyödyntää kerättyä tietoa IoT-alustalle. Kun suunnittelimme ratkaisua kuntoutuksen seurantaan, niin päällimmäisenä toteamuksena oli ke- säkuussa, että sensoreiden pitäisi olla langattomia ja yhdistettävissä Bluetooth-yhteydellä esim. tablettiin, jossa tietoa voisi tarkastella ja lähettää eteenpäin. Ensimmäisistä TAMK:n terveyden etäseurantaratkaisuista tuli nyt hieman kömpelöitä käyttää, koska jouduimme välittämään tietoa useiden mikrotietokoneiden kautta käyttöliittymälle. Meditech2016 messuilla oli esillä monia uusia terveyden etämittausratkaisuja, joista osa pääsi mukaan myös suosituksiimme mahdollisista hankintakohteista.

5.1 Käyttöliittymän suunnittelu

Timo Salminen

Havahduimme tarpeeseen sovelluksesta, jolla voisi yhdistää asiakkaan langattomia sensoreita käyttövälineelle ja välittää tiedot tietokantaan analysoitavaksi. Sairaanhoidohenkilöiden kanssa keskusteltaessa selvisi, että asiakkailla on jo mittareita ja sensoreita käytössä kotona, mutta niiden tietoja ei mitenkään saada välitettyä potilastietojärjestelmiin. Tarvittaisiin avoin rajapinta (esim. Rest API) asiakkaan toteuttaman oman seurantatiedon siirtämiseksi mittareista potilastietokantaan. Oppilastöinä toteutetut sensoriratkaisut hyötyisivät myös tästä, kun työn tuloksia voitaisiin havainnollistaa helposti.

Sovellusta kehitettäessä on huomattavaa hyötyä, jos käytetään hyödyksi projektin elinkaarta. Ideoinnissa olisi erinomaisen tärkeää, että jo tässä vaiheessa olisi mukana sovelluksen suunnittelussa kaikista mahdollisista sidosryhmistä henkilöitä, sairaanhoidohenkilöstöä, sovelluksen kehittäjät ja loppukäyttäjiäkin. Määritellään sovelluksen tavoitteet kaikki sidosryhmät huomioon ottaen ja huomioidaan mahdollisia ongelmakohtia, jotka vaikuttavat sovelluksen kehitykseen. Suunnitteluvaiheessa otetaan selville kaikki resurssit, jotka ovat käytettävissä, ohjelmoijat, fysioterapeutit, lääkärit, sairaanhoitajat, projektipäällikkö jne. Aikataulukutus tulee laatia tässä vaiheessa. Resursoinnissa tulee ottaa huo-

mioon, että kaikilla sidosryhmillä on aikataulun puitteissa mahdollisuus osallistua käyttöliittymän suunnitteluun, testaukseen, ja mahdollisesti uusien ideoiden mahdollistamiseen. Kun suunnitteluvaiheesta edetään toteutusvaiheeseen, tulee ottaa erityisesti huomioon testaus, joka kulkee käsi kädessä koko toteutusvaiheen ajan.

Toteuttamisvaiheessa, jossa aloitetaan itse sovelluksen tai käyttöliittymän tekeminen tulee valita tietyt välitavoitteet. Välitavoitteina voitaisiin pitää luontevasti eri toiminnallisuuksien saavuttamista. Välitavoitteita, joita analysoidaan eri sidosryhmien kanssa:

- Tarvittavat resurssit eri sidosryhmistä ovat selvillä
- Sovelluksen ensimmäisen version kehittäminen tavoitteiden perusteella
 - o Tiedetään mihin loppukäyttäjä tarvitsee sovellusta ja pidetään loppukäyttäjän päämäärä sovellusta kehitettäessä.
 - o Sidoryhmiltä tulevat tavoitteet
 - Sidoryhmät, lääkärit, sairaanhoitajat, fysioterapeutit, loppukäyttäjät ovat antaneet tiedon sovelluksessa tarvittaville ominaisuuksille.
 - Sovelluksessa tapahtuva analytiikka on mahdollista toteuttaa sidoryhmiltä saamien tietojen perusteella. Analytiikkaa ei tarvitse toteuttaa vielä ensimmäiseen läpikäyntiin.
 - Analytiikalla tässä tarkoitetaan esimerkiksi polvileikkauksesta tulevan kuntoutettavan loppukäyttäjän liikerata-analytiikan laatimista, jolla taataan, että liikeradat suoritetaan oikein.
- Sovelluksen ensimmäinen läpikäynti
 - o Voidaan järjestää *ryhmäläpikäyntinä*. Sovelluksesta voidaan näyttää paperikuvia, jolloin saadaan hyödyllistä tietoa siitä, että onko sovelluksen visuaalinen olemus odotetun mukainen.
- Sovelluksen toteuttaminen/ohjelmointi n - kertaa
- Sovelluksen testaus, yhteisläpikäynti tai paritestit eri sidosryhmien kanssa n - kertaa
 - o Ohjaaja ja testaaja etenevät käyttöliittymän/sovelluksen testauksessa yhdessä keskustellen sovelluksesta. Ohjaajan on oltava tässä vaiheessa vielä erittäin aktiivinen, että mahdolliset ongelmakohdat saadaan selvitettyä.
 - o Analytiikan/algorithmien olisi hyvä olla tässä vaiheessa toteutettuna, jotta testikäyttäjät pääsevät näkemään, miten analytiikka sovelluksessa toimii ja onko analytiikka toteutettu oikein

Sovelluksen toteutus- ja testausyklejä voi aikataulun puitteissa olla useampikin kerta. Olisi kuitenkin hyvä pyrkiä useampaan toteutus- ja testausyكليin ja testata ahkerasti ja säännöllisesti. Kun tarvittavat iterointikierrokset on tehty sovelluksen hyvän luotettavuuden ja loppukäyttäjää palvelevaksi, olisi tärkeää järjestää pilottitesti. Pilottitestissä tulee kuntoutettavan asiakkaan, sekä hoitohenkilön, sairaanhoitajan tai fysioterapeutin käydä

läpi testitilanne kuntoutettavalle. Erityisen tärkeää on, että fysioterapeutti tai sairaanhoitaja katsoo ensiksi oikeat liikeradat kuntoutettavan kanssa, jonka jälkeen kuntoutettava toistaisi testitehtävät. Tämän jälkeen kuntoutuja kertoo, miten hän suoriutui testitehtävistä ja miten hän ymmärsi tehtävät. Tässä vaiheessa palaute on vielä tärkeää, koska sovellusta voidaan muuttaa helpommin. On myöskin otettava huomioon, että pelkästään kuntoutuja ei tässä tilanteessa ole ainoa loppukäyttäjä. Sekä itse ohjaaja, fysioterapeutti tai sairaanhoitaja on samalla tavalla loppukäyttäjä tässä tilanteessa. Onkin siis tärkeää saada samat huomiot myös fysioterapeutilta tai sairaanhoitajalta. Lopuksi olisi hyvä haastatella sekä kuntoutuja, että häntä ohjaava henkilö, sairaanhoitaja tai fysioterapeutti. Haastateltavilta on tärkeä saada tuntemukset millaista käyttöliittymää olisi hyvä käyttää. Usein hyvä tunnelma sovelluksesta takaa hyvän lopputuloksen sovelluksesta.

Hyvä käyttäjäkokemus taataan seuraavilla asioilla, joihin on erityisesti kiinnitettävä huomiota, kun sovellusta ja käyttöliittymää toteutetaan:

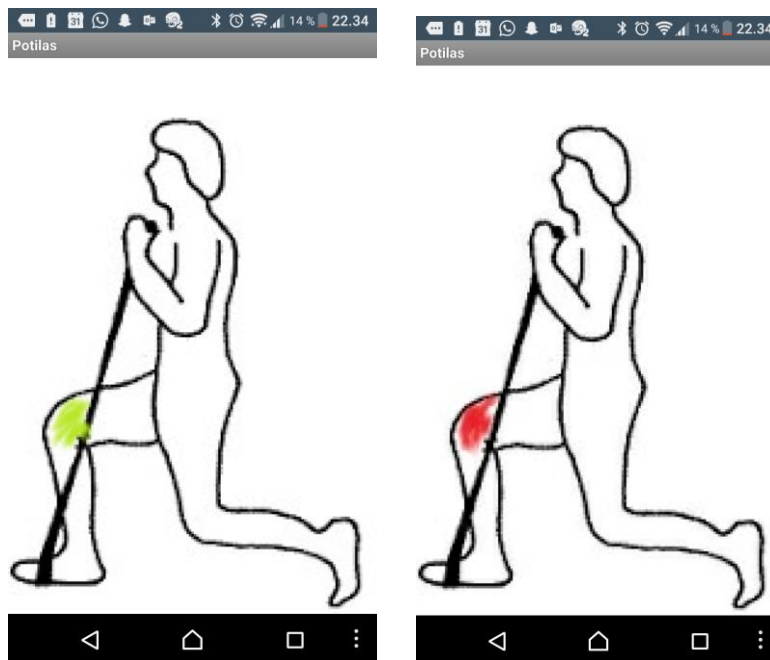
- Sovelluksen näkyvyys on hyvä ja käyttäjä itse ymmärtää sovelluksen toiminnallisuuden.
- Huomaamattomuudella saadaan aikaan se, että kuntoutujan ajatukset eivät suuntaudu väärään suuntaan vaan huomio on itse kuntoutustehtävän tekemisessä.
- Ennustettavuuteen on kiinnitettävä huomiota. Käyttäjä esimerkiksi voi olettaa sovelluksen näyttävän kuntoutettavan ruumiinosan liikkuvan samaan tahtiin kuin hän itse sitä liikuttaa.
- Hallittavuus on yksi osatekijä, joka vaikuttaa käyttäjäkokemukseen. On erittäin tärkeää, että kuntoutettava henkilö tuntee olevansa itse hallitseva tekijä sovelluksen suhteen. Tämä osa-alue vaikuttaa suuresti hyvään käyttäjäkokemukseen.
- Yleisiä perussääntöjä sovellukselle ja käyttöliittymälle ovat, että sovellus on mahdollisimman yksinkertainen ja helppokäyttöinen. Helppokäyttöisyys myös siksi, koska kuntoutettavat asiakkaat voivat olla mistä ikäryhmästä tahansa ja heiltä ei voi vaatia suurta teknistä tuntemusta.

Sovelluksessa huomioon otettavia muita ominaisuuksia:

- Sensoreilta saatava informaation välitys sovellukselle on oltava reaaliaikaista ja nopeasti saavutettavissa (KUVA 32 ja KUVA 323).
 - o Kuntoutuja ehtii tehdä mahdolliset korjausliikkeet, kun ohjelma ja sensorit toimivat nopeasti ja ilmoittavat mahdollisesta virheliikkeestä
 - o Tiedonvälitys tapahtuu langattomasti, esim. Bluetooth
- Käyttöliittymästä tiedon välitys IoT-alustalle,
 - o Mahdollisimman nopea toimitus IoT-alustalle, että hoitaja, fysioterapeutti voi tarvittaessa tehdä mahdolliset havainnot ja parannusehdotukset jopa reaaliajassa
 - o Havaitsimme huomattavia viiveitä omissa testeissämme tiedonsiirrossa IoT-Ticketille

- Sovelluksen helppokäyttöisyys kaikilla ikäryhmillä
 - o Selkeä
 - o Hallittava
- Sovellus on esteettisesti huoliteltu
- Sovellusta on mahdollisimman paljon automatisoitu
 - o Hoitaja kirjaa kuntoutujan tiedot sovellukseen, joilla käyttäjä myös identifioituu IoT-alustaan, johon tieto välitetään
 - o Tiedon salauksesta huolehtiminen, jos käyttäjätietoja välitetään
 - o Hoitaja/ohjaaja liittää sensorit sovellukseen, joita kuntoutujan ei tarvitse itse tämän jälkeen enää liittää
 - o Ilmoitus kuntoutujalle, jos Bluetooth ei ole päällä (KUVA 34)

Seuraava Mock-Up sovellus näyttää peruseräilyä, miten käyttäjälle voidaan näyttää päätelaitteelta, kännykkä tai tabletti, miten polven kuntoutus sujuu. Mock-Up versio ei ota millään tavalla kantaa itse sensoreiden toimintaa tai tarkempaan analyysiin, mihin suuntaan ja syvyydellä tulisi liikkeitä tehdä. Mutta Mock-Up näyttää, sen että polven kuntoutuja näkee helposti, yksinkertaisesti milloin liike tehdään oikein ja milloin väärin. Tarkempi ja huolitellumpi analyysi tarvitaan siinä vaiheessa, kunnes halutaan kuntoutujalle välittää tieto mihin suuntaan pitää liikettä korjata.



KUVA 32 ja 33. Polven kuntoutuspotilas, ilmaistaan, että liike tehdään oikein tai väärin. (huom. suunta mihin suuntaan liikettä tulisi tehdä puuttuu kuvasta)



KUVA 34 Ohjeistetaan kuntoutujaa kytkemään Bluetooth päälle

Kuten jo kerrottu osiossa 4.3.3 Wapicelta löytyy päätelaitteeseen ladattavaksi IoT-Tracker sovellus. Se löytää älypuhelimessa olevat anturit, mutta ei muita. Tarvittaisiin siis applikaatio, jolla voidaan helposti luoda yhteys tabletista tai älypuhelimesta esim. BT-majakan kautta yhdistettävään sensoriin. Applikaation kautta tieto voitaisiin sitten helposti välittää mm. IoT-Ticketille. Tämä applikaatio voisi mahdollisesti antaa myös palautetta sekä raportteja käyttäjälle terveyden kehityksestä. Suurimpien sensorivalmistajien tuotteissa mahdollisesti on jo avoin rajapinta, mutta tätä kuultua toteamusta ei nyt laajemmin lähdetty selvittämään.

Wapicen Markus Mäkelä kertoi, että kyseisen kaltaiseen sovellukseen on joskus tehty demosofta, mutta sitä ei enää sellaisenaan kehitetä. TAMK voi saada lähdekoodin, jos joku opiskelijoista haluaa kehittää sitä edelleen.

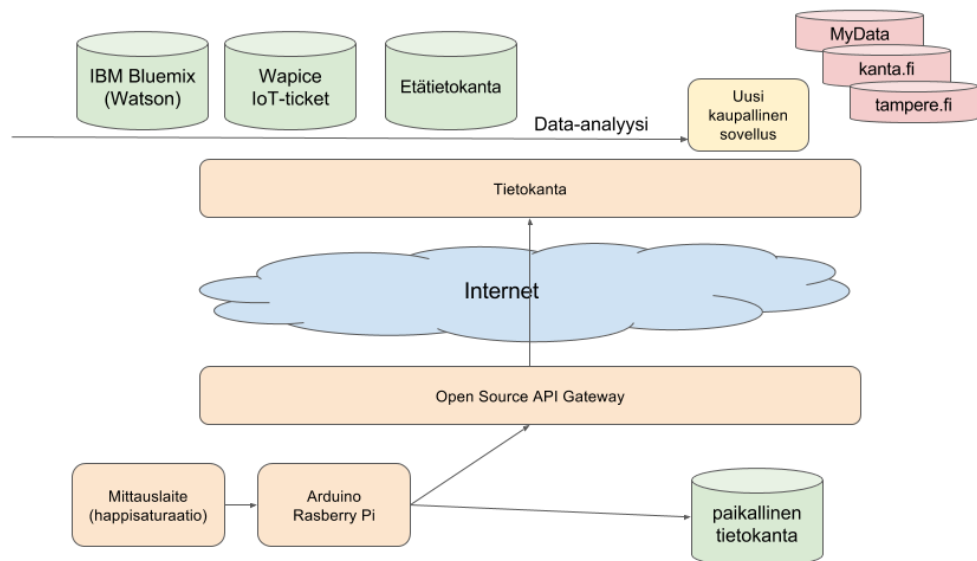
Sensoreilta applikaation kautta IoT-Ticketille asti voidaan käyttää avointa RestAPI-rajapintaa, kun taas potilastietojärjestelmiä ja Omakanta (kanta.fi/omakanta) varten pitäisi olla käytössä HL7 RestAPI-rajapinta.

5.2 Arkkitehtuuri ja integraatio

Juho Kauppi

Palveluväyläratkaisu suositellaan toteutettavaksi avoimen lähdekoodin sovelluksella, esimerkiksi Umbrella API Gateway (apiumbrella.io), jonka etuja on useamman API-ratkaisun (IoT-laitteen) liittäminen ja käyttöoikeuksien hallinnointi yhdestä käyttöliittymästä. APInf.io (apinf.io) on Suomessa tehtävä vastaava ratkaisu, joka on integroitu toimimaan MQTT-rajapintojen kanssa yhteensopivaksi, ja nykyisellään siinä hoidetaan REST-rajapinnat Umbrellan avulla. APInf.io on hallintakerros, jossa yhdestä pisteestä voidaan hallita useaa proxya. Jatkokehitykseen olisi hyvä valita tämän kaltainen yhteistyökumppani.

Kuvio 8 kuvaa Arkkitehtuuriratkaisua jatkotutkimuksia ajatellen.



KUVIO 8 Arkkitehtuuri II (Juho Kauppi 2017)

Wapice tarjoaa datan käsittelyyn analyysityökaluja. Opinnäytetyössä tehdyssä konfiguraatiossa, käytettäessä Wapicen omaa palvelinta, käyttöliittymän hidas reagointi nähtiin ongelmaksiksi. Ratkaisussa käytimme ainoastaan Wapicen IoT-ticketin käyttöliittymätyökaluja. Jatkokehityksenä voisi tutkia myös muita Wapicen tarjoamia IoT-palveluja.

Amazon Web Services (AWS), Google ja IBM Bluemix tarjoaa myös erilaisia työkaluja datan käsittelyyn. Näistä mielenkiintoisin alusta on IBM Bluemix, joka tarjoaa kauppanimellä Watson erilaisia tekoälysovelluksia alustansa päällä. Google on keskittynyt erilaisiin data-analyysin työkaluihin. AWS tarjoaa laskentakapasiteetin ohella myös erilaisia sovelluskehityksen työkaluja.

Opinnäytetyö on hyvä proof-of-concept, joka arkkitehtuuriltaan mahdollistaa erilaisten kombinaatioiden rakentamisen poistamalla ja lisäämällä moduuleja ratkaisuun. Tukeutuminen yhteen sovellustoimittajaan on oppimisympäristönä vaivaton ratkaisu, mutta uusien innovaatioiden kehitysalustana käsiä sitova valinta. Opinnäytetyön tarkoitus oli toteuttaa hyvinvointiteknologian koulutusohjelmaan oppimisympäristö. Työ on poikinnut useita jatkokehityspolkuja, joita on hyvä toteuttaa yhteistyössä myös muiden koulutusohjelmien kanssa.

5.3 Tietoturvan toteutus

Juho Kauppi

Hyvän tietoturvan ja tietosuojan pääperiaatteena kaikki tietoliikenneyhteydet salataan. Jotta oppimisympäristö olisi käytettävyydeltään hyvä tietosuojamielessä, on erikseen määritelty oppilaitoskäyttö ja potilaskäyttö. Oppilaitoskäytössä datasta ei kerätä henkilörekisteriä, jolloin tietosuoja-asetusten huomioitta jättäminen helpottaa käyttöä ja testaamista huomattavasti.

5.3.1 Oppilaitoskäyttö

Helpoin tunnistautumisen toteutusratkaisu oppilaitoskäytössä olisi loppukäyttäjän sormenjälkitunnistus, jossa kone antaa juoksevan tunnistenumeron sormenjälkitunnistuksen perusteella. Näin voitaisiin kerätä mittausdataa anonymisti, kuitenkin tallentaa eri loppukäyttäjiltä huomattava määrä dataa. Data olisi tunnistetiedolla eriteltävissä, mutta henkilön tietosuoja säilyisi anonyminä.

Olisi hyvä, että mittaus tietojen historiadataa erilaisiin kokeilukäyttötarkoituksiin, esimerkiksi mittaus tietodataa voisi käsitellä erilaisin data-analytiikan keinoin ja löytää datamassasta (Big Data) erilaisia syy-yhteyksiä ja kehittää uusia tuoteinnovaatioita, esimerkiksi hyvinvointiteknologian koulutuksen ja Y-kampustoiminnan yhteisyyenä.

Hyvinvointiteknikan laboratorion tulisi olla nimetty pääkäyttäjä, jonka vastuulla olisi laboratorion laitekokonaisuus, sekä osaaminen miten tietosuoja-asiat ja tietoturva toteutetaan laboratorioympäristössä.

On suositeltavaa, että mittausdatalle määritellään joku säilytysaika, esimerkiksi 1 vuosi. Jos dataa kertyy paljon, lyhyempikin aika voi olla perusteltua.

5.3.2 Potilaskäyttö

Jos ja kun hyvinvointitekniikan laboratoriota käytetään aidoissa asiakasprojekteissa. On hyvä huomioida lain vaatimukset potilasdatan käsittelyssä.

Wapicen palvelinalustaa käytettäessä, myös heiltä pitää saada selvitys, miten he käsittelevät alustan dataa, onko datan suojaus lain ja asetusten vaatimalla tavalla tehty. TAMKin omalla palvelimella vastaavanlainen tietosuojaselvitys, kuka pääsee tietoihin käsiksi ja mihin niitä käytetään.

Uuden tietosuojasuoja-asetuksen myötä tiedot kannattaa poistaa asiakassuhteen päättämisen jälkeen palvelimelta. Potilaalla on myös mahdollisuus viedä hänestä kerätty data mukanaan seuraavaan palveluun eli potilaalle voi tarjota mahdollisuutta tietojen hävittämiseen, mukaan ottamista tai mahdollisesti siirtämistä johonkin muuhun palveluun. Kanta.fi-palvelun kehittymisen seuraaminen ja HL7-rajapintavaihtoehdon mukaan ottaminen ympäristöön kannattaa ainakin testata.

5.3.3 Yhteenveto

Suosittelvat roolit laboratoriokäytössä:

- Pääkäyttäjä, tekninen pääkäyttäjä. Luo opiskelijoille tunnukset TAMKin Wapice-ympäristöön. Pitää kirjaa käyttäjistä ja projekteista. Huolehtii, että projektien data myös poistetaan palvelimelta projektin päättyessä. Varmistaa tietosuoja- ja tietoturvakäytänteet hoidetaan asianmukaisesti opiskelijoiden toimesta.
- Opiskelijakäyttäjä, light. Oikeudet tehdä ja hallinnoida omia projektejaan.
- Opiskelijakäyttäjä, heavy. Oikeudet omaan instanssiin eli esimerkiksi muusta opiskelukäytöstä eristetty ympäristö potilasdatasta johtuen.
- Opettajakäyttäjä, vastaava käyttöoikeus kuin opiskelija-heavy. Voi luoda omille opiskelijoilleen ympäristöjä.

Palvelimen on syytä olla normaalisti DDOS-suojattu, SSL-sertifikaattia käyttävä palvelin, jolloin tietoturvan perusominaisuudet ovat kunnossa. Jatkokehitystyönä tulee miettiä, miten tieto saadaan salatusti siirtymään sensoreilta TAMKin palvelimelle.

Mittaustapahtuman yksilöinti esimerkiksi yksilöllisellä tunnistenumeraalla ja sen linkittäminen tietokantatauluun, missä on potilaan tiedot. Jolloin ainoastaan hoitohenkilökunta

pääsee sensitiiviseen henkilötietoon käsiksi. Vain terveydenhuollon ammattihenkilöt, jotka osallistuvat potilaan hoitoon, ovat oikeutettuja tekemään potilasasiakirjamerkintöjä.

Työssämme olemme käyttäneet potilaan mittaamiseen vain ei-lääkinnällisiä, ei-luotettavia mittalaitteita, mutta järjestelmään voidaan liittää myös kalibroituja laitteita, kunhan niissä on tunnettu rajapinta, jolloin mittaustulokset ovat luotettavia ja eteen tulee kysymys, voiko itse tehty mittaus olla luotettava ja voiko mittaustuloksen siirtää automaattisesti esimerkiksi kanta.fi-palveluun, sairaanhoitopiirin potilasrekisteriin vai olisiko omille mittaustuloksille hyvä olla olemassa joku oma tietokantansa, jossa käyttäjä voisi antaa lukulupia vakuutusyhtiölle, terveydenhuollolle jne.

5.4 Ostosuosituksia

Mika Helonsalo

Ensimmäisen väliesityksen jälkeen toivottiin opinnäytetyöhön lisättävien ehdotuksia uusista ja paremmista sensoreista ja laitteista, jotka soveltuisivat hyvin opetustyöhön. Ostosuositukset kappale on kirjoitettu tätä ajatusta mukaillen.

5.4.1 Verensokerin mittaus

Aamulehdestä oli artikkeli, jossa kerrottiin uudelta ratkaisulta seurata verensokeria. Tässä ratkaisussa iholle kiinnitetään pieni sensori, josta lähtee säie ihon läpi rasvakudokseen. Sensorin rasvakudoksesta mittaama verensokeriarvo luetaan tähän tarkoitukseen olevalla lukulaitteella. Tällaisen sensorin patterinkesto on parhaimmillaan 2 viikkoa. Verensokeriarvon tutkiminen rasvakudoksesta on erittäin uusi ratkaisu vielä Suomessa, joten sen käyttämiseen opetustoiminnassa voisi panostaa ja tehdä yhteistyötä alan toimijoiden kanssa. Markkinoilla on ainakin kaksi valmistajaa, joiden ratkaisu tukee rasvakudoksesta mittausta.

Dexom G5 Mobile CGM-järjestelmä on jatkuvaan glukoosin tarkkailuun tarkoitettu etämittausjärjestelmä (dexcom.com). Se soveltuu aikuisille ja vähintään 2-vuotialle lapsille. Lähettimessä on BT-yhteys tietojen lähettämiseksi älylaitteellesi. Se mahdollistaa glukoosiarvojen, trendien ja tietojen etätarkastelun. Se lähettää myös varoituksen puhelimeesi silloin, kun glukoosiarvot ovat menossa liian mataliksi tai korkeaksi, joko hälytysäänenä tai tekstiviestinä. Mobiili Applikaatio on saatavilla iOS- ja Android-käyttöjärjestelmän laitteille. Sensori laitetaan vyötärön tasolle tai käsivarteen mukana tulevan asennuslaitteen avulla. Sensorista menee säie ihon alle ja se kiinnittyy teipillä ihoon. Sensoriin kiinnitetään vielä BT-lähetin, joka välittää tiedon lukulaitteelle. Sensorista on kuva 35. Koska lukulaite on Bluetooth yhteydessä (kantama 6 m) sensoriin, niin sen lukemisen voi suorittaa hieman etempääkin. Anturia on hyväksyttävää käytettävä yhtäjaksoisesti 7 päivän ajan ja on vedenkestävä (Infucare 2017). Dexomin ratkaisun hinta on hieman kalliimpi kuin FreeStyle Libren Flash-toteutus. Maahantuojalta ei saatu vastausta saatuuteen Suomessa.



KUVA 35. Dexcom sensoripaketti käyttövalmiina (dexcom.com 2016)

FreeStyle Libre Flash on Abbott Diabetes Care kehittämä tuote ja se on tulossa kokeilukäyttöön myös Pirkanmaalla (abbottdiabetescare.fi/tuotteemme/freestyle-libre).

Sen saa toistaiseksi Suomessa käyttöön vain terveydenhuollon ammattilaisten kautta. Sensori ja lukulaite maksavat tällä hetkellä saman verran eli noin 60€ molemmat, näistä on kuva 36. Jatkuvan seurannan avulla saamme tietää myös verensokerin vaihtelun päivä ja yön aikana. Lisäksi se kertoo arvion verensokeriarvon kehittymisen suunnalle. Sensorin käyttöaika on 2 viikkoa, jonka jälkeen se on vaihdettava uuteen. Käyttöä helpottaa sen vedenkestävyys. Tietojen lukeminen kestää vain 1 sekunnin ja tapahtuu noin 4 cm etäisyydellä sensorista. Sensori asennetaan käsivarteen tuotepaketissa olevan asennuslaitteen avulla ja sen pitää paikallaan kiinnitysteippi. Sensorista lähtevä 0,4 mm säie menee 5 mm ihon läpi mitatakseen kudokset. Glukoosi arvo tallentuu 15 min välein. (Abbott 2017)



KUVA 36. FreeStyle Libren sensori ja lukulaite (kuva: Mika Helonsalo 2017)

5.4.2 Verenpaineen mittaus

Verenpaineen mittaus painevaihteluun perustuvalla mansetilla oli todella tuskallinen kokemus, kun paine kumipussissa nousi jopa yli 200 mmHg. Medica2016 messuilla pääsi kokeilemaan rannemittaukseen perustuvaa iHealth ratkaisua. Oheisen kokeilun yhteydessä on otettu kuva 37. Siinä keskittymistä vaati ranteen (mittarin) pitäminen sydämen tasolla (rintakehän päällä) 30 sekuntia. Itse mittaustapahtuma oli vähemmän tuskallinen kuin Kodea käsivarsimittarilla, vaikka siinäkin on painemansetti. Mitatut arvot pystyy näkemään heti laitteen näytöllä ja ne tallentuu BT-yhteyden kautta myös tietokoneen muistiin myöhempää tarkastelua varten. Olkavarsimittarit ovat käyttöohjeessa mainittujen tarkkuusarvojen perusteella vielä rannemittareita tarkempia. Tekniikka kehittyy nykyään nopeasti koko ajan tarkemmaksi ja havaintojeni perusteella mittaustilanteessa ratkaisevaan on mitattavan rauhallisuus. Jos mitattava henkilö tuntee mittaustilanteen epämiellyttäväksi, niin todennäköisesti se nostaa verenpainetta mittarin epätarkkuutta enemmän. Etämittauksessa mittarin mittaustarkkuutta oleellisemmaksi tekijäksi todennäköisesti osoittautuu pitkäkestoinen tieto mittaustulosten kehityksestä, onko seurattavan arvot paranemassa vai huononemassa. iHealth BP7 verenpainemittarin verollinen hinta Suomessa on noin 100€ ja saatavilla se on saatavilla elektroniikkaa myyvistä kaupoista (ihealthlabs.com).



KUVA 37. iHealth ranneverenpainemittari (kuva: Mika Helonsalo 2016)

Lisäksi rannemittareita myy ainakin Beurer, Philips ja Withings, mutta heidän mittareistaan työryhmällämme ei ole käyttäjäkokemusta.

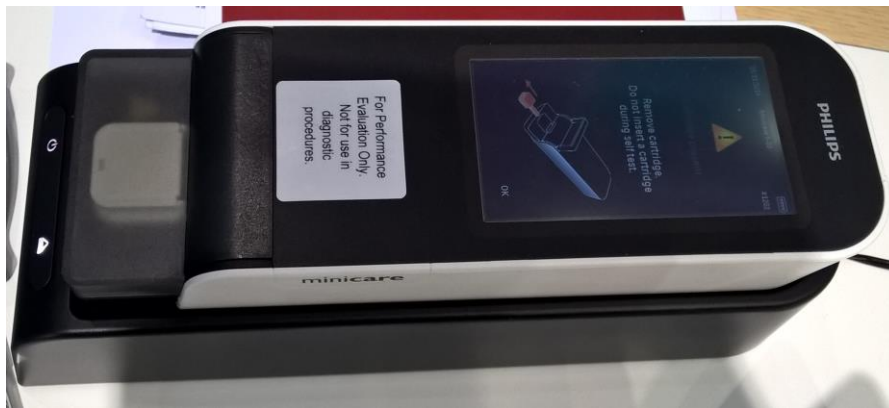
5.4.3 Tuoteperheratkaisut

Opetuksen kannalta voisi olla järkevää hankkia muutamalta valmistajalta heidän koko tuoteperheen ratkaisut kokeiltavaksi ja vertailtavaksi. Tällaisia laajojen henkilökohtaisten mittausvälineratkaisujen valmistajia on ainakin iHealth, Philips, Withings (Nokia) ja Beurer (beurer.com/web/en/products).

Philips tuoteperhe terveyden etämittaukseen käsittää seuraavia ratkaisuja (philips.co.uk/c-m-hs/health-programs):

- verenpaineen ja pulssin mittaus sekä ranteesta että olkavarresta
- vaaka painon mittaukseen
- kelloja sykkeen ja aktiivisuuden mittaukseen
- lääkkeenjakolaatikko, jonka avulla lääkkeiden käyttöä voidaan valvoa
- keskuspuhelin jonka välityksellä saa Päivystys-palveluun yhteyden
- hammasharja joka neuvoo harjaamaan paremmin ja kertoo myös hammaslääkärille, että kuinka hyvin on harjaus suoritettu (kuvat 40 ja 41)
- sydämen toimintoja mittava sensori, jonka toiminta-aika on 4vrk täysin johdottomasti (kuva 39)
- kuumemittari
- lapsen seuranta kameralla
- kaatumisanturi
- paikannussensori, jossa on GPS paikannus
- CRP veriarvon mittaus. Tulos on valmis 2 minuutissa. Toistaiseksi tämä on ainut mittaus (kuva 38)

Kokonaisuutta hallinnoitaan sovelluskaupasta ladattavalla applikaatiolla. Kaatumisriskin analysointiin oli myös oma sovellus, mutta valitettavasti siitä meillä ei ole tarkempaa tietoa saatavilla. Kaikkia näitä ei ole vielä valitettavasti ole myynnissä Suomessa. Pilottimaina heillä on ensin Britannia ja USA.

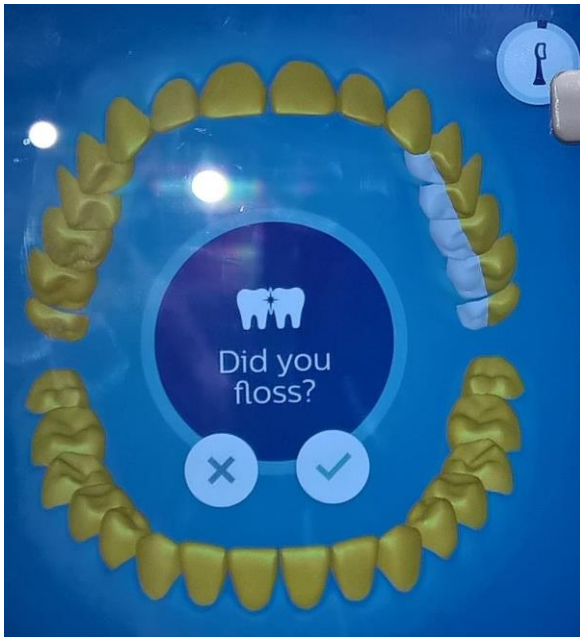


KUVA 38. Philips CRP-verianalyysilaitte (kuva: Mika Helonsalo 2016)



KUVA 39. Philips sydäntoimintojen seuranta sensori (kuva: Mika Helonsalo 2016)

Kaikkein eniten Philips tarjoamista laitteista henkilökohtaiseen hyvinvointiin liittyen herätti kiinnostusta kuitenkin sähköhammasharja, joka opettaa sinua harjaamaan hampaasi oikein. Neuvonta tapahtuu applikaatiossa (kuva 43), jonka voit ladata puhelimeesi. Hammasharjassa (kuva 44) on sensorit paikalle, paineelle ja hankaavalle puhdistukselle.



KUVA 40 ja 41. Philips Sonicare sähköhammasharja ja applikaatio hampaiden puhtauden tarkkailuun (kuva: Mika Helonsalo 2016)

iHealth on panostanut erityisesti laitteidensa ulkonäköön ja ainakin siksi ne herättävät heti ostajan mielenkiinnon. Kaikkein mielenkiintoisin heidän tuotteistaan on mieles-tämme vaaka, joka analysoi kehosi koostumuksen, kuten kokonaispaino, rasva-%, sitou-tunut vesimäärä, lihasmassa, kaloreiden tarve päivässä, BMI (painoindeksi).

Heiltä löytyy seuraava valikoima tuotteita (ihealthlabs.com):

- verenpaineen ja pulssin mittaus sekä ranteesta että olkavarresta
- verensokeri verinäytteen ja liuskan avulla
- aktiivisuusranneke
- happisaturaatio mittaus sormesta
- kehoa analysoiva vaaka

Withings on ranskalainen yritys, jonka Nokia osti 2016. Heidän tuotevalikoimastaan löytyy hienolla designilla tehtyjä itsemittaamisen ratkaisuja. Näyttöpäätteenä käytetään pääosin puhelinta, joten monet ratkaisut ovat yksinkertaisempia kuin kilpailijoilla. Emme ole päässeet tutustumaan tai käyttämään heidän tuotteitaan. Suluissa on mainittu yrityksen oman verkkokaupan verollinen hinta (withings.com).

- verenpainemansetti, jonka mittauspaiikkana on olkavarsi (100€)
- aktiivisuuden mittausranneke (50€)
- otsalämpömittari (100€)
- happisaturaatio ranteesta (100€)
- videovalvonta ja ilmanlaadun seuranta (200€)
- unenlaatu (190€)
- kehoa analysoiva vaaka (180€)
 - o BMI (painoindeksi), pulssi, kehon koostumus

TAMK Hyvinvointiteknologian koulutuksen kannalta erityisen mielenkiintoisen Nokian tekee myös se, että Tampereella ja Espoossa he kehittävät uusia terveysteknologian ratkaisuja. Nokia Digital Health esitteli 1.11.2016 Tekes ja HealthHUB.fi järjestämässä tilaisuudessa käynnissä olevaa kokeilua, jossa seurataan HUS potilaiden sydänkohtauksen jälkeistä elämää. Siinä seurataan heiltä verenpainetta, EKG:ta, aktiivisuutta ja painoa. Tärkeimpiä oppeja hankkeesta on ollut, että potilaat on saatava sitoutumaan sensorien käyttöön esim. jakamalla heille tietoa kehityksestään sekä laitteiden käytettävyyden ol-tava sujuvaa. Olisi erittäin hyödyllistä paikallisen elinkeinoelämän kannalta saada tällaisia kokeiluja tehtäväksi myös PSHP kanssa, koska tällä hetkellä suurin osa uusista teknologia kokeiluista Suomessa näyttää tapahtuvan yhteistyössä HUS kanssa. (Nokia Digital Health 2016)

6 POHDINTA

Toukokuun lopussa 2016 liityimme TAMKin tutkimushanketyöryhmään, jonka piti toteuttaa tiedonkeräys langattomasti Hyvinvointiklinikan ryhmämuotoisen kuntoutuksen laitteista perusteilla olevaan IoT-laboratorioon. Kesäkuussa aihe tarkentui hankkeeksi, jossa suunnittelisimme yhteistyöyritysten kanssa ratkaisun polvikuntoutettavien etäseurantaan. Työmme tarkoitus oli seurata kotikuntoutettavien harjoitusten oikeellisuutta ja rasittavuutta etämittauksen menetelmin. Yhteistyökumppaneiksi oli ehdolla Coxa, Hurlabs, Goodlife, PhysioTools ja Myontec. Teimme tämän hankkeen valmistelua syyskuuhun asti, jolloin alkoi jo näyttää vahvasti siltä, että emme saa yhteistyötä valittujen kumppaneiden kanssa käyntiin ennen kuin vasta vuonna 2017. Toinen suuri ongelma oli tutkimusluvan hakeminen ja saaminen PSHP:ltä polvileikkauspotilaiden käyttämiseksi osana tutkimusta. Aihe oli hyvä ja pyritään aloittamaan seuraavan työryhmän toimesta.

Syyskuussa ideapaperin esittelyn jälkeen päätimme pistää 6 henkilön työryhmän puoliksi, jolloin insinöörit YAMK ryhtyisivät selvittämään ratkaisuja, joita koululle hankitulla e-Health-Kit paketilla voitaisiin toteuttaa. Samalla kehittäisimme uusien opiskelijoiden oppimisympäristöä hyvinvointiteknologian osa-alueella. Määrittelimme työmme sisällöksi: tiedonsiirron toteutus sensorilta käyttöliittymään, tiedon keräyksen tapa, tiedon visualisointi sekä tiedon käytettävyys.

Ryhmätyönä tehty aikuisopiskelijoiden opinnäytetyö on huomattavasti haastavampi kuin yksintehty, koska jokaisen aikataulujen sovittaminen yhteen vie huomattavasti aikaa muusta tekemisestä. Yksin tehdyssä opinnäytetyössä tarvitsee ottaa huomioon vain omat kyvyt, tehtävän antajan toiveet ja ohjaajan neuvot etenemiseksi. Ryhmätyössäkin teoriaosuuden voi kirjoittaa melko itsenäisesti, mutta kehitystehtävää on vaikea jakaa samalla tavalla itsenäisiksi osioiksi. Kehitystehtävien on hyvä tulla luoduksi yhteinen tavoite mielessä, jotta työ ei lähde liiaksi rönsyilemään. Työryhmälle on hyvä valita projektijohtaja, joka järjestee ja sopii muidenkin puolesta tapaamisia työn kannalta tärkeiden yhteistyökumppanien kanssa. Yhteisen esityksen kirjoitus vaatii viimeistelyn ja lähteiden jakamisen pilvipalvelimessa. Ongelmia aiheuttaa erilaiset käyttöohjelmistot, jolloin saman työn muokkaus ei välttämättä onnistukaan ihan helposti. Jouduimme tarkastamaan aina ensin muiden muutokset, ennen kuin ryhdyimme päivittämään omaa uutta osioitamme tekstiin. Vaarana oli esim. kopioida oma osuus vanhaan versioon työstä tai tuhota täysin toisen

tekemät Tyyli-määrytykset, jolloin jo kerran tehdyn päivityksen joutui tekemään uudelleen.

Työn aikana huomasimme, että TAMK:n opetus Hyvinvointiteknologiasta on liikaa teoriatiedon varassa. Koululle oli hankittuna Libeliumin kasaama e-Health-Kit sensoripaketti. Se ei kokemuksiemme mukaan sovellu kovin hyvin nykyaikaiseen opetukseen hyvinvointiteknologian mahdollisuuksista. Ensimmäinen suuri huomiomme oli, että kaikki sensorit vaativat langallisen yhteyden. Arviomme oli, että varsinkin kuntoutuksen seurannan pitäisi tapahtua mahdollisimman huomaamattomasti, jottei siitä olisi haittaa itse tapahtumalle. Tämän takia sensoreiden langattomuus tuntuu ainoalta mahdolliselta toteutustavalta. e-Health-Kit sensorien tuottama tieto oli analogista ja ensin se olisi muunnettava digitaaliseksi, jotta voisimme analysoida sitä. Jouduimme siis toteuttamaan paljon enemmän perustavaa tutkimusta ja kokeiluja tiedon saattamiseksi käyttöliittymäämme kuin aluksi arvelimme. Onneksi saimme käyttöömmme Wapicen IoT-Ticketin, jolloin ainakin ihan oman käyttöliittymän toteutus ei enää ollut päätavoitteemme. Seuraavat työryhmät voivat pohtia lisää järkevän käyttöliittymän tarpeita, kun tiedonsiirron perusratkaisu on kerran selvitetty alusta loppuun. Seuraava versio tiedonsiirtoratkaisusta voisi olla jo langaton ja tieto kannattaa kerätä koulun omalle palvelimelle, jotta datasiirron viive jää mahdollisimman lyhyeksi. Lisäksi olisi mahdollista kokeilla, että toimiiko tiedonsiirron ratkaisumme samalla tavalla myös toisten IoT-alustojen kanssa, kuten Thingworx tai IBM Bluemix (Watson), joka tarjoaa alustan lisäksi myös alustaan liitettyjä tekoälysovelluksia ja analytiikkaa.

Mietimme työssämme ehdotuksen tiedonkeräyksen applikaatioksi. Tekemämme pohjatyön perusteella voisi toteuttaa (ohjelmoida) ihan toimivan ohjelman tiedon keräämiseksi avoimen rajapinnan sensoreilta. Työn aikana kävi selväksi, että tiedonsiirto asiakkaiden omilta sensoreilta ei vielä onnistu potilastietojärjestelmiin. Tämän vuoksi aloimme ideoida uutta applikaatiota millä tiedonkeräys ja lähetys tietokantaan onnistuisi. Nykyaikaisen käyttöliittymän suunnittelun lähtökohdat selventyivät työn aikana ja oli ilo huomata, että voimme käyttää myös IoT-alustaa käyttöliittymänä. Esineiden internetin pohjalta toteutetun palvelun suunnittelu poikkeaa aiemmista hyvinvoinnin palveluista, joten uuden palvelumallin kuvaus tuli myös todelliseen tarpeeseen.

Suosittellemme TAMK Hyvinvointiteknologian koulutusohjelmaa hankkimaan joidenkin kaupallisten yritysten jo valmiiksi mietittyjä ratkaisuja opetuskäyttöön, jotta teknologian

käyttömahdollisuudet tulevat mahdollisen monelle oppilaalle nopeasti selville. Opetusryhmän taustatiedot ovat hyvin erilaisella tasolla (sairaanhoitaja, tradenomi, insinööri AMK), joten ei ole hyvä edellyttää IT-insinöörin tietotaitoa, jotta voi kokea onnistumisen iloa oman terveyden etämittauksessa. Esim. Philipsin ja iHealth tarjoamat langattomat ratkaisut vaikuttivat nopean tutustumisen perusteella hyviltä, koska mittaustiedon näkee nopeasti suoraan laitteelta ja sen lisäksi sen saa välitettyä käyttöliittymälle tarkasteltavaksi myöhemmin. Mahdollinen ongelma tarjolla olevien kaupallisten ratkaisujen kanssa voi olla suljettu käyttörajapinta, joten on hyvä edistää avoimen rajapinnan sensorien kehittämistä, vaikka IT-osaston oppilaiden opinnäytetöinä. Aktiivinen yhteistyö terveysteknologiayritysten kanssa avaa myös mahdollisuuksia saada opetuksen käyttöön nykyaikaisia hoitoratkaisuja ja luo mahdollisuuksia tehdä opinnäytetöitä yhdessä alan yritysten kanssa.

Onnistuimme löytämään uusia mahdollisuuksia toteuttaa hyvinvointiteknologian koulutusohjelman opetusta IoT-laboratorion yhteydessä. Selkeä huomiomme oli, että koulun opetuksessa on herätty melko myöhään tarpeeseen osallistua aktiivisesti etäkuntoutuksen ratkaisujen toteutukseen ja kehittämiseen. Tarve opetuksen ajantasaistamiseen on ollut, mutta toteutukseen ei ole ollut tarpeeksi resursseja. Iloksemme voimme myös todeta, että loimme useita mahdollisia opinnäytetyöaiheita seuraavaksi opetukseen osallistuville. Koulutusohjelman etu olisikin, jos oppilaat tekisivät TAMKille vuosittain opinnäytetöitä aiheeseen liittyen. Uusien terveysteknologioiden kehitystyö on hidasta ja pitkäjänteistä toimintaa.

Haluamme kiittää opettajaamme Lea Saarnia hyvän alkuperäisen opinnäytetyöaiheen ideoinnista sekä ohjaavaa opettajaamme Tony Torpia insinööreille sovitettuna aihealueen rajaamisesta meille sopivaksi. Sitten vielä erityiskiitos IT-osaston opettaja Kari Naakalle IoT-alustan käyttöön liittyvistä neuvoista. Tommi Perälä edisti lopputyötämme tutkimalla sensorien ja mikrokontrollerien käyttöä omana oppilastyönään. Vihtori Lehtonen toimi alussa myös suurena apuna määriteltäessä meille sopivaa IoT-alustaa. Kiitos Teille kaikille ☺.

Opinnäytetyöryhmämme taival oli pitkä, mutta erittäin opettavainen. Työ eteni pienin askelin jatkuvasti muuttuvassa tavoiteympäristössä, mutta lopulta halusimme saada sen yhdessä valmiiksi. Toivomme selvitystyöstämme olevan paljon hyötyä seuraavia oppimiskokonaisuuksia mietittäessä.

LÄHTEET

Abbott. 2017. FreeStyle glukoosimittaus. Luettu 1.2.2017

<https://abbottdiabetescare.fi/tuotteemme/freestyle-libre>

Arto, K., Martinsuo, M. & Kujala, J. 2006. (2. painos: 2008). Projektiliiketoiminta. Helsinki: WSOY.

http://pbgroup.aalto.fi/en/the_book_and_the_glossary/projektiliiketoiminta.pdf

Beurer. 2017. Luettu 4.2.2017

<https://www.beurer.com/web/en/products/bloodpressure/wrist.php>

Collin, J. & Saarelainen, A. 2016, Teollinen internet. Helsinki: Talentum.

Cooking-hacks 2016. Luettu 20.6.2016. <https://www.cooking-hacks.com>

Cooking-hacks. e-Health Sensor Platform for Biometric and Medical applications. Luettu 20.1.2017

<https://www.cooking-hacks.com/documentation/tutorials/ehealth-biometric-sensor-platform-arduino-raspberry-pi-medical>

Cooking-hacks 2017. Glucometer Sensor for e-Health Platform. Luettu 20.1.2017

<https://www.cooking-hacks.com/glucometer-sensor-ehealth-medical>

Digita. 15.12.2015, Actility ja lähetyverkko-operaattori Digita ottaneet koekäyttöön LoRa-verkon. Luettu 19.1.2017.

http://www.digita.fi/medialle/tiedotteet/2015/actility_ja_lahetyverkko-operaattori_digita_ottaneet_koekayttoon_lora-verkon.4293.news

DNA 2016. Mikä on IPv6. Luettu 25.11.2016. <https://www.dna.fi/ipv6>

Elisa. 2016. Terveysthuollon digitalisointi. Helsinki: Elisa Oyj. Luettu 14.9.2016

<https://yrityksille.elisa.fi/terveydenhuollon-digitalisointi>

Elektroniikkalehti. 2015 & 2016. Suomeen perustettiin ensimmäinen IoT-radioverkko. Luettu 5.1.2017.

http://etn.fi/index.php?option=com_content&view=article&id=5510:nyt-voi-rakentaa-iot-verkon-kotiin&catid=13&Itemid=101

Elektroniikkalehti. 2015. Kuinka mitata veren happitasoa puettavalla laitteella? Luettu 15.12.2016

http://etn.fi/index.php?option=com_content&view=article&id=3593:kuinka-mitata-veren-happitasoa-puettavalla-laitteella&catid=26&Itemid=140

Euroopan Parlamentin ja Neuvoston asetus (EU) 2016/679 <http://eur-lex.europa.eu/legal-content/FI/TXT/HTML/?uri=CELEX:32016R0679&from=EN>

EU:n tietosuojauudistus, Eurooppa-neuvosto. 2016. Luettu 25.11.2016.

<http://www.consilium.europa.eu/fi/policies/data-protection-reform>

- EU:n tietosuojauudistus, Valtionvarainministeriö. 2016. Luettu 11.12.2016.
http://vm.fi/artikkeli/-/asset_publisher/lakiuudistus-parantaa-tietosuojaa-eu-ssa-tuore-ra-portti-uudistuksesta-antaa-suosituksia-muutosvaiheeseen
- Hannula K. 2016, Mittalaittevaatimukset ja niiden toiminta terveydenhuollon etämittaamisprosessissa. Tampereen ammattikorkeakoulu. Hyvinvointiteknologia. Opinnäytetyö. Luettu 22.12.2016
http://www.theseus.fi/bitstream/handle/10024/109619/Hannula_Kari.pdf?sequence=1
- HUSlab ohjekirja. 2016. Luettu 15.12.2016 <http://huslab.fi/ohjekirja/2424.html>
- Hyödynmaa, P. 2016. Olosuhdekaappien Kalibrintipalvelujen Kehittäminen. Hyvinvointiteknologian koulutusohjelma. Tampereen ammattikorkeakoulu. Opinnäytetyö.
- I2C Interface for Raspberry Pi, Luettu 15.11.2016.
<https://learn.sparkfun.com/tutorials/raspberry-pi-spi-and-i2c-tutorial>
- iHealth. 2017. Luettu 4.2.2017
<https://ihealthlabs.com/blood-pressure-monitors/wireless-blood-pressure-wrist-monitor-view-bp7s/>
- IoT analytics. 2016. Luettu 15.12.2016. <https://iot-analytics.com/>
- Infucare 2017. Luettu 1.2.2017. <http://www.infucare.se/fi/diabetes-2/dexcom-g5>
- Jameson, A. Adaptive Interfaces and agents, Luettu 24.1.2016.
<http://dfki.de/~jameson/pdf/Jameson08Handbook.pdf>
- Johnson, J. 2010. Designing with the mind in mind: simple guide to understanding user interface design rules, Morgan Kaufmann Publishers/Elsevier
- Jokelainen P. 2011. Hoitohenkilöstön tietosuojaja- ja tietoturvatietämys. Luettu 12.1.2017
http://epublications.uef.fi/pub/urn_nbn_fi_uef-20120015/urn_nbn_fi_uef-20120015.pdf
- Kaasinen, E. ja Norros, L. 2007. Älykkäiden ympäristöjen suunnittelu. Tampere: Tammer-Paino Oy
- Järvinen, P. 2002. Tietoturva & yksityisyys. Docendo.
- Kodea verenpainemittarin käyttöohje. Luettu 23.11.2016
http://www.zowamedical.com.sg/?page_id=314
- Koskinen, A. L. 2016. Diabeteksen hoito mullistuu – Uusi laite mittaa verensokerin ilman verta. Aamulehti 22.4.2016. Luettu 18.1.2017.
<http://www.aamulehti.fi/kotimaa/diabeteksen-hoito-mullistuu-uusi-laite-mittaa-verensokerin-ilman-verta-23599699/>
- Käyttöliittymä, Graphic User Interface, Artikkelit World Heritage Encyclopedia, Luettu 18.10.2016, http://www.worldlibrary.org/articles/eng/Graphic_user_interface
- Latva-aho, M. 2015. Oulun yliopiston langattoman tietoliikenteen tutkimuskeskus.

- Huippunopea 5G tulee jo lähivuosina, näin se muuttaa elämäämme. Luettu 14.1.2017. http://www.iltalehti.fi/digi/2015031519363343_du.shtml
- Libelium. 2016. e-Health Sensor Platform for Biometric and Medical applications. Luettu 5.6.2016. <http://www.libelium.com/130220224710/>
- LoRa Alliance. 2017. Luettu 14.1.2017 <https://www.lora-alliance.org/what-is-lora/technology>
- Lääketieteen sanasto. 2016. Duodecim. Luettu 15.12.2016 <http://www.terveyskirjasto.fi/terveyskirjasto>
- Lääkintälaitedirektiivi - suomenkielinen versio - Neuvoston direktiivi 93/42/ETY
- McEwen, A. & Cassimally, H. 2014. Designing the internet of things. Chichester: Wiley. <http://site.ebrary.com.elib.tamk.fi/lib/tamperepoly/detail.action?docID=10784816>
- MIT App Inventor 2017. Tutustuttu ja käytetty 1.2.2017. <http://appinventor.mit.edu/explore/>
- Mustajoki, P. 2015. Kohonnut verenpaine. Lääkärikirja Duodecim. Luettu 27.12.2016 http://www.terveyskirjasto.fi/terveyskirjasto/tk.koti?p_artikkeli=dlk00034
- Mäkelä, M. 2016. Wapice IoT-ticket esittely 3.11.2016 Tampereen ammattikorkeakoululla.
- Nokia Digital Health & Honkanen, M. 2016. Seminaari 1.11.2016 HealthHUB Tampere. http://www.nokia.com/en_int/about-us/what-we-do/digital-health
- Nuolivirta, M. & Sahila S. 2013. Opinnäytetyö Diakonia-ammattikorkeakoulu. Salassapito ja tietosuoja terveysalalla. Luettu 13.1.2017 <https://publications.theseus.fi/handle/10024/59755>
- Oikeusministeriö: uusi tietosuoja-asetus http://oikeusministerio.fi/fi/index/valmisteuilla/lakihankkeet/informaatio-oikeus/henkilotietojensuojakansallisenlainsaadannontarkistaminen_0.html
- Opi tietosuoja. 2016. Luettu 27.12.2016. <https://opitietosuoja.fi>
- Parviainen, E. & Sorjanto, K 2012. Käyttöliittymän suunnitteluperiaatteita tietokoneille ja mobiililaitteille. Oulun seudun ammattikorkeakoulu. Opinnäytetyö.
- Philips. 2017. Luettu 4.2.2017. <http://www.philips.co.uk/c-m-hs/health-programs>
- Postscapes. 2017. LoRa Network Protocol and Long Range Wireless IoT. Luettu 28.1.2017 <http://www.postscapes.com/long-range-wireless-iot-protocol-lora>
- Tietosuojavaikuttetun toimisto: EU:n tietosuojaudistus <http://www.tietosuoja.fi/fi/index/euntietosuojaudistus.html>

- Saarelainen, A. Esineiden internet vaatii omat verkot. TIVI-lehti, lokakuu 2016. Helsinki: Talentum. Luettu 9.2.2017.
- Salminen, A-L. & Hiekkala, S. & Stenberg, J-H. 2016, Etäkuntoutus. Helsinki: KELA. <https://helda.helsinki.fi/bitstream/handle/10138/161341/Etakuntoutus.pdf?sequence=1>
- Sinkkonen I., Kuoppala, H., Parkkinen, J., Vastamäki R. 2006 3. uudistettu painos, Käytettyvyyden psykologia, Edita Publishing Oy
- Sylander, I. 2016. Organisaatioiden välinen tiedonsiirto. TAMK Opetusmateriaali
- Syrjänen, S. 2010. Protokollat. Luettu 23.11.2016. <https://wiki.helsinki.fi/display/verkko/Protokollat>
- Tarvainen, S. 2016. Elisa Terveydenhuollon palvelut. Esitys 30.06.2016. Helsinki: Elisa Oyj
- Tarvainen, S. & Katz, R. & Kumpulainen, J. & Elovaara, R. 2016. Elisa Etämittaus esitys Pasilan toimistolla 5.9.2016. Helsinki: Elisa Oyj
- Tarvainen, S. & Elovaara, R. 2016. Elisa ja digitalisoitua sosiaali- ja terveydenhuolto. Seminaari Health-HUB Tampere 1.11.2016. Helsinki: Elisa Oyj
- Teknavi. 11.12.2016. Bluetooth 5 laitteissa ensi vuonna tuplasti nopeutta ja nelinkertaisesti kantomatkaa. Luettu 11.1.2017. <http://teknavi.fi/elektronikka/bluetooth-5-laitteissa-ensi-vuonna-tuplasti-nopeutta-ja-nelinkertaisesti-kantomatkaa>
- Teknavi. 30.1.2017. 5G-verkkojen aloitus nähdään jo tämän vuoden aikana. Luettu 30.1.2017. <http://teknavi.fi/elektronikka/ensimmainen-5g-verkko-avautumassa-jo-tana-vuonna>
- Terve-lehti. Verenpaineen mittaaminen ja tavoitearvot. Luettu 27.12.2016 <http://www.terve.fi/verenpaine/verenpaineen-mittaaminen-ja-tavoitearvot>
- Terve-lehti. Verensokerin kotimittaus ja seuranta -potilasohje. Luettu 20.12.2016 <http://www.terve.fi/diabetes-tyyppi-2/81737-verensokerin-kotimittaus-ja-seuranta-potilasohje>
- Tietosuojavaltuutetun toimisto. 2017. Luettu 16.12.2016. <http://www.tietosuoja.fi/fi/>
- Tohtori.fi. Verenpaineen mittaus. Luettu 27.12.2016 http://www.tohtori.fi/files/Verenpaineen_mittaus_potilasohje.pdf
- Uniapnea.fi. Tietoa uniapneasta. Luettu 20.1.2017. <http://www.uniapnea.fi/hoito/diagnosointi/suppea-unitutkimus>
- Vehmaanpera, J. 2016. Wapice IoT-ticket valmomoalusta kiinteistöautomaatiokäytössä. Talotekniikan koulutus. Tampereen ammattikorkeakoulu. Opinnäytetyö. Luettu 28.12.2016.

Vesterlund, N. 2012. Selvitystyö verensokerimittareiden toimintaperiaatteista ja virhelähteistä. Hyvinvointiteknologian koulutusohjelma. Oulun seudun ammattikorkeakoulu. Opinnäytetyö. Luettu 15.1.2017

Vuononvirta, T. 2011. Etäterveydenhuollon käyttöönotto terveydenhuollon verkostoissa. Oulu: Oulun yliopisto. <http://jultika.oulu.fi/files/isbn9789514297175.pdf>

Wapice. 15.12.2016. Wapice rakentaa LoRa-verkon seitsemään kaupunkiin. Luettu 19.1.2017. <https://www.iot-ticket.com/fi/uutiset/60-wapice-rakentaa-lora-verkon-seitsemään-kaupunkiin>

Wapice. 2016. IoT-ticket. Luettu 15.11.2016. <https://www.iot-ticket.com/>

Wapice. 2016. Dashboard luonti. Tutustuttu 20.11.2016
<https://my.iot-ticket.com/Dashboard/main/root/#desktop>
<https://www.youtube.com/watch?v=Y2sNDSOspIk&feature=youtu.be>

Warburg, O.H. Happivajaus saattaa olla seurausta esimerkiksi kroonisesta keuhkosairaudesta, sydänsairaudesta, dekompensoituksesta tai astmakohtauksesta. Luettu 15.12.2016. <http://graviolateam.blogspot.fi/p/blog-page.html>

Wiio, A. Käyttäjätavallisen sovelluksen suunnittelu, Luettu 18.10.2016,
<http://www.technologos.fi/kirja.htm>
<http://www.technologos.fi/Luku-13.pdf>

Windows 10 Iot Core, Luettu 21.2.2017.
<https://developer.microsoft.com/en-us/windows/iot>

Wiring Pi, USB Serial connection to Raspberry Pi, Luettu 18.1.2017.
<http://wiringpi.com/reference/serial-library/>

Wiring Pi Library, USB Serial connection to Raspberry Pi , Luettu 19.1.2013.
<https://projects.drogon.net/raspberry-pi/wiringpi/serial-library/>

Withings. 2017. Luettu 4.2.2017. <http://www.withings.com/eu/en/products>

Libcurl, Luettu 1.2.2017. <https://curl.haxx.se/libcurl/c/libcurl.html>

w3schools, JSON. https://www.w3schools.com/js/js_json_intro.asp

LIITTEET

Liite 1. etäk_arduino.ino

(Perälä, T. TAMK 2016)

1 (4)

```

1  /*
2  TAMK etäkuntoutus
3
4  <PinChangeInt.h> ja <eHealth.h> eHealth websivulta, Wire.h arduinon kirjastoissa valmiina
5
6  */
7
8  #include <PinChangeInt.h>
9  #include <eHealth.h>
10 #include <Wire.h> // Used for I2C
11
12
13 // The breakout board defaults to 1, set to 0 if SA0 jumper on the bottom of the board is set
14 #define MMA8452_ADDRESS 0x1D // 0x1D if SA0 is high, 0x1C if low
15
16 //Define a few of the registers that we will be accessing on the MMA8452
17 #define OUT_X_MSB 0x01
18 #define XYZ_DATA_CFG 0x0E
19 #define WHO_AM_I 0x0D
20 #define CTRL_REG1 0x2A
21
22 #define GSCALE 2 // Sets full-scale range to +/-2, 4, or 8g. Used to calc real g values.
23
24 int laskuri = 0; // laskuri 10ms*i=aika
25 int cont = 0; // pulssin mittauksen laskuri
26
27 void setup()
28 {
29     Serial.begin(115200);
30
31     Wire.begin(); //liittyminen i2c master
32
33     eHealth.initPulsioximeter(); //Test and init pulsioximeter
34     initMMA8452(); //Test and init asentoanturi
35
36     PCintPort::attachInterrupt(6, readPulsioximeter, RISING); //nappia painettaessa keskeytys
37 }
38

```

(jatkuu)

```

39 void loop()
40 {
41   int accelCount[3]; // Stores the 12-bit signed value
42
43
44   if (laskuri == 100) // alla olevat tapahtuu 100*10ms välein (n.1000ms)
45   {
46
47     //ASENTOANTURI
48     readAccelData(accelCount); // Read the x/y/z adc values
49
50     // Now we'll calculate the acceleration value into actual g's
51     float accelG[3]; // Stores the real accel value in g's
52     for (int i = 0 ; i < 3 ; i++)
53     {
54       accelG[i] = (float) accelCount[i] / ((1<<12)/(2*GSCALE)); // get actual g value, this depends on scale being set
55       if(accelG[i] < -4) // alle -4 arvot -0 -1 väärille
56         accelG[i] = accelG[i] + 4;
57     }
58
59
60     // for (int i = 0 ; i < 3 ; i++)
61     // {
62     //   if(i==0){
63     //     Serial.print("x:"); // accelG[0]
64     //   }
65     //   else if(i==1){
66     //     Serial.print("y:"); // accelG[1]
67     //   }
68     //   else{
69     //     Serial.print("z:"); // accelG[2]
70     //   }
71     //   Serial.print(accelG[i], 2); // Printtaa g arvot, 2 desimaalin tarkkuudella
72     // }
73     Serial.print("x:");
74     Serial.print(accelG[0], 2);
75     Serial.print(",");
76
77     Serial.print("y:");
78     Serial.print(accelG[1], 2);
79     Serial.print(",");
80
81     Serial.print("z:");
82     Serial.print(accelG[2], 2);
83     Serial.print(";"); //datatyyppin loppu
84
85
86     //PULSIOXIMETER
87
88     Serial.print("bpm:");
89     Serial.print(eHealth.getBPM());
90     Serial.print(";"); //datatyyppin loppu
91
92     Serial.print("spo:");
93     Serial.print(eHealth.getOxygenSaturation());
94     Serial.print(";"); //datatyyppin loppu
95
96     Serial.println(" "); //?
97
98     laskuri=0; //laskurin nollaus
99   }
100
101   laskuri++; //laskuri+1
102   delay(10); //10ms viive
103 }
104
105
106
107
108 void readAccelData(int *destination)
109 {
110   byte rawData[6]; // x/y/z accel register data stored here
111
112   readRegisters(OUT_X_MSB, 6, rawData); // Read the six raw data registers into data array
113

```

(jatkuu)

3 (4)

```

111
112 // Loop to calculate 12-bit ADC and g value for each axis
113 for(int i = 0; i < 3 ; i++)
114 {
115     int gCount = (rawData[i*2] << 8) | rawData[(i*2)+1]; //Combine the two 8 bit registers into one 12-bit number
116     gCount >>= 4; //The registers are left align, here we right align the 12-bit integer
117
118     // If the number is negative, we have to make it so manually (no 12-bit data type)
119     if (rawData[i*2] > 0x7F)
120     {
121         gCount -= 0x1000;
122     }
123
124     destination[i] = gCount; //Record this gCount into the 3 int array
125 }
126 }
127
128 // Initialize the MMA8452 registers
129 // See the many application notes for more info on setting all of these registers:
130 // http://www.freescale.com/webapp/sps/site/prod\_summary.jsp?code=MMA8452Q
131 void initMMA8452()
132 {
133     // byte c = readRegister(WHO_AM_I); // Read WHO_AM_I register
134     // if (c == 0x2A) // WHO_AM_I should always be 0x2A
135     // {
136     //     Serial.println("Asentoanturi havaittu...");
137     // }
138     // else
139     // {
140     //     Serial.print("Ei yhteyttä", WHO_AM_I rekisterin arvo: 0x");
141     //     Serial.println(c, HEX);
142     //     while(1) ; // Loop forever if communication doesn't happen
143     // }
144

```

```

145 MMA8452Standby(); // Must be in standby to change registers
146
147 // Set up the full scale range to 2, 4, or 8g.
148 byte fsr = GSCALE;
149 if(fsr > 8) fsr = 8; //Easy error check
150 fsr >>= 2; // Neat trick, see page 22. 00 = 2G, 01 = 4A, 10 = 8G
151 writeRegister(XYZ_DATA_CFG, fsr);
152
153 //The default data rate is 800Hz and we don't modify it in this example code
154
155 MMA8452Active(); // Set to active to start reading
156 }
157
158 // Sets the MMA8452 to standby mode. It must be in standby to change most register settings
159 void MMA8452Standby()
160 {
161     byte c = readRegister(CTRL_REG1);
162     writeRegister(CTRL_REG1, c & ~(0x01)); //Clear the active bit to go into standby
163 }
164
165 // Sets the MMA8452 to active mode. Needs to be in this mode to output data
166 void MMA8452Active()
167 {
168     byte c = readRegister(CTRL_REG1);
169     writeRegister(CTRL_REG1, c | 0x01); //Set the active bit to begin detection
170 }
171
172
173
174 // Read bytesToRead sequentially, starting at addressToRead into the dest byte array
175 void readRegisters(byte addressToRead, int bytesToRead, byte * dest)
176 {
177     Wire.beginTransmission(MMA8452_ADDRESS);
178     Wire.write(addressToRead);
179     Wire.endTransmission(false); //endTransmission but keep the connection active
180
181     Wire.requestFrom(MMA8452_ADDRESS, bytesToRead); //Ask for bytes, once done, bus is released by default
182

```

(jatkuu)

4 (4)

```

183     while(Wire.available() < bytesToRead); //Hang out until we get the # of bytes we expect
184
185     for(int x = 0 ; x < bytesToRead ; x++)
186         dest[x] = Wire.read();
187 }
188
189 // Read a single byte from addressToRead and return it as a byte
190 byte readRegister(byte addressToRead)
191 {
192     Wire.beginTransmission(MMA8452_ADDRESS);
193     Wire.write(addressToRead);
194     Wire.endTransmission(false); //endTransmission but keep the connection active
195
196     Wire.requestFrom(MMA8452_ADDRESS, 1); //Ask for 1 byte, once done, bus is released by default
197
198     while(!Wire.available()) ; //Wait for the data to come back
199     return Wire.read(); //Return this one byte
200 }
201

```

```

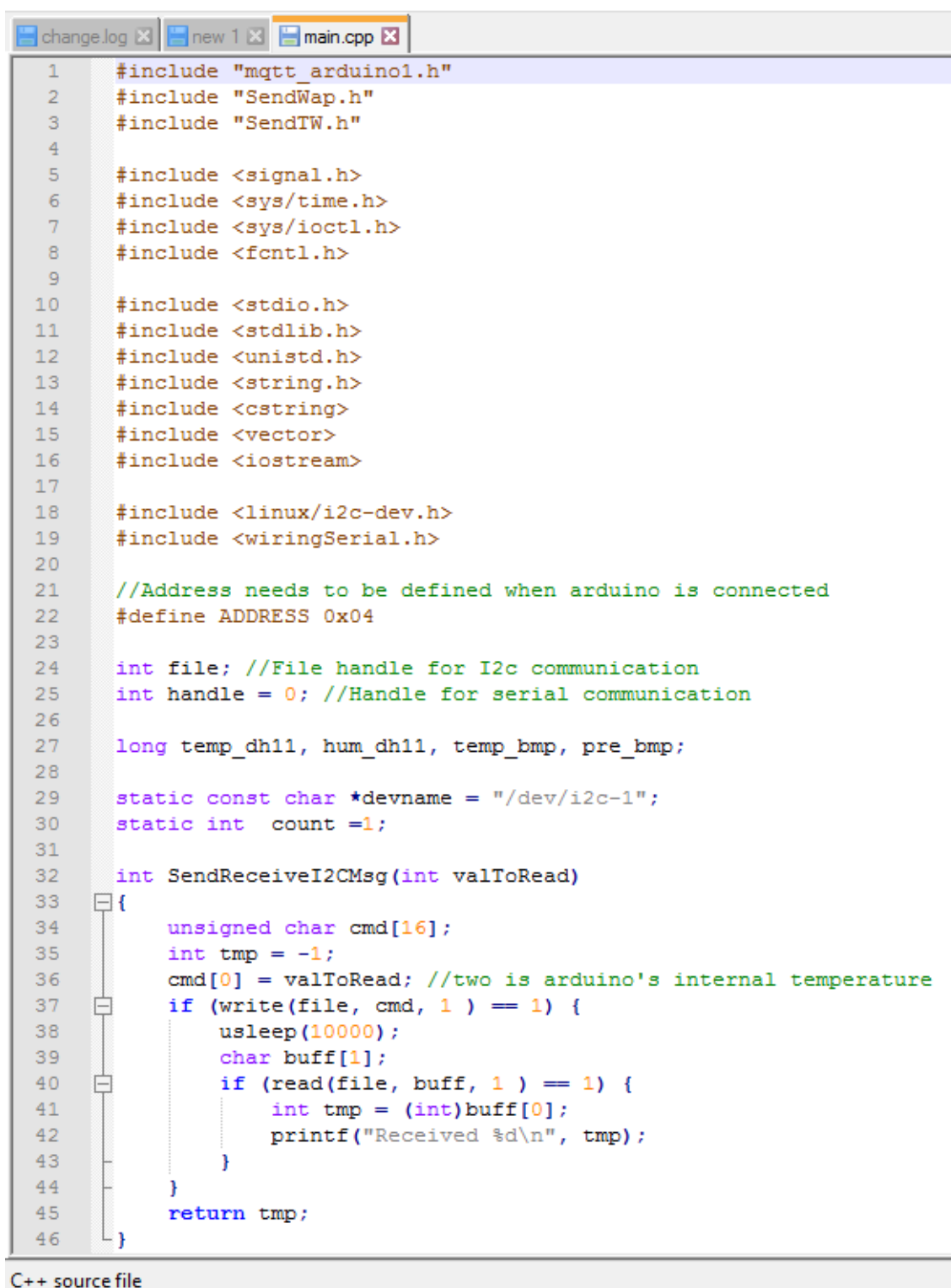
202
203 // Writes a single byte (dataToWrite) into addressToWrite
204 void writeRegister(byte addressToWrite, byte dataToWrite)
205 {
206     Wire.beginTransmission(MMA8452_ADDRESS);
207     Wire.write(addressToWrite);
208     Wire.write(dataToWrite);
209     Wire.endTransmission(); //Stop transmitting
210 }
211
212
213 void readPulsioximeter()
214 {
215
216     cont ++;
217
218     if (cont == 50) { //Get only of one 50 measures to reduce the latency
219         eHealth.readPulsioximeter();
220         cont = 0;
221     }
222 }
223

```

Lite 2. main.cpp

(Salminen, T. TAMK 2017)

1 (5)



```

1  #include "mqtt_arduino1.h"
2  #include "SendWap.h"
3  #include "SendTW.h"
4
5  #include <signal.h>
6  #include <sys/time.h>
7  #include <sys/ioctl.h>
8  #include <fcntl.h>
9
10 #include <stdio.h>
11 #include <stdlib.h>
12 #include <unistd.h>
13 #include <string.h>
14 #include <cstring>
15 #include <vector>
16 #include <iostream>
17
18 #include <linux/i2c-dev.h>
19 #include <wiringSerial.h>
20
21 //Address needs to be defined when arduino is connected
22 #define ADDRESS 0x04
23
24 int file; //File handle for I2c communication
25 int handle = 0; //Handle for serial communication
26
27 long temp_dh11, hum_dh11, temp_bmp, pre_bmp;
28
29 static const char *devname = "/dev/i2c-1";
30 static int count =1;
31
32 int SendReceiveI2CMsg(int valToRead)
33 {
34     unsigned char cmd[16];
35     int tmp = -1;
36     cmd[0] = valToRead; //two is arduino's internal temperature
37     if (write(file, cmd, 1 ) == 1) {
38         usleep(10000);
39         char buff[1];
40         if (read(file, buff, 1 ) == 1) {
41             int tmp = (int)buff[0];
42             printf("Received %d\n", tmp);
43         }
44     }
45     return tmp;
46 }

```

C++ source file

(jatkuu)

2 (5)

```
47
48 std::vector<std::string> GetData()
49 {
50     std::vector<std::string> testi;
51
52     int chr = 0;
53     char* content = (char*)malloc(100);
54     char c = '0';
55     int i = 0;
56     while(chr != '\r'){
57         char c;
58         chr = serialGetchar(handle);
59         //printf ("%d", chr);
60         //printf ("DataAvail: %d\n", chr);
61         if (chr != -1) {
62             c = (char)chr;
63             printf ("%c", c);
64
65             if (content == NULL)
66                 printf("NULL\n");
67             else
68                 content[i] = c;
69         }
70         i++;
71     }
72
73     //printf("\n");
74
75     //Contains all then necessary data that should be sent to IoT ticket
76     char* strs = (char*)malloc(100);
77     strs = strtok(content, ",:;");
78
79     while (strs != NULL)
80     {
81         printf("%s\n", strs);
82         testi.push_back(strs);
83         strs = strtok(NULL, ",:;");
84     }
85     delete content;
86     delete strs;
87     return testi;
88 }
89
```

(jatkuu)

3 (5)

```

90 void timer_handler (int signum)
91 {
92
93     //int ret = SendReceiveI2CMsg(2);
94
95     std::vector<std::string> ret = GetData();
96
97     printf ("Sending to IoT  %d count\n", ++count);
98     printf ("Arduino len:  %d\n", ret.size());
99     for (int i=0; i<ret.size(); i++) {
100         //if (ret[i].c_str() == " ")
101         // printf("Empty\n");
102         //else
103         // printf("%s\n", ret[i].c_str());
104         //send_to_Wap(ret[0].c_str(),"Main/level1",atof(ret[1].c_str()),"%");
105         //printf("%s", ret[0].c_str());
106     }
107
108     printf("%s : %s\n", ret[0].c_str(), ret[1].c_str());
109     if (ret.size() >= 10) {
110         send_to_Wap(ret[0].c_str(), "HVVNT",atof(ret[1].c_str()),"v");
111         send_to_Wap(ret[2].c_str(), "HVVNT",atof(ret[3].c_str()),"v");
112         send_to_Wap(ret[4].c_str(), "HVVNT",atof(ret[5].c_str()),"v");
113         send_to_Wap(ret[6].c_str(), "HVVNT",atof(ret[7].c_str()),"v");
114         send_to_Wap(ret[8].c_str(), "HVVNT",atof(ret[9].c_str()),"v");
115     }
116     //send_to_Wap("hum_test", "Main/level2",12,"%");
117
118     //send_to_Wap("temp_dh32", "Main/level1",temp_dh11/10, "C");
119
120     //send_to_Wap("pre_bmp", "Main/level1",pre_bmp/10, "mbar");
121
122     //send_to_Wap("temp_bmp", "Main/level1",temp_bmp/10, "C");
123
124     //send_to_TW("Temperature",temp_dh11/100);
125
126     //send_to_TW("Humidity",hum_dh11/100);
127
128 }
129
130

```

(jatkuu)

4 (5)

```
131 int InitI2C()
132 {
133
134     printf("I2C: connecting\n");
135     if ((file = open(devname, O_RDWR)) < 0)
136     {
137         fprintf(stderr, "I2C: Failed to access %d\n", devname);
138         return 1;
139     }
140
141     printf("I2C: acquiring buss to 0x%x\n", ADDRESS);
142
143     if(ioctl(file, I2C_SLAVE, ADDRESS) < 0) {
144         fprintf(stderr, "I2C: Failed to acquire bus access/talk to slave 0x%x\n", ADDRESS);
145         return 1;
146     }
147
148     printf("I2C Initalized. Ready to talk");
149     return 0;
150 }
151
152 int OpenSerial()
153 {
154
155
156     handle = serialOpen("/dev/ttyAMA0", 115200);
157     if (handle == -1)
158     printf("Serial open failed\n");
159     else {
160         printf("Serial open Succeed\n");
161         handle = serialOpen("/dev/ttyUSB0", 115200);
162         if (handle == -1)
163         printf("Serial2 open failed\n");
164     }
165     printf("Serial2 hadle: %d\n", handle);
166     return handle;
167 }
168
```

(jatkuu)

5 (5)

```
169 int main(int argc, char *argv[])
170 {
171     class mqtt_tempconv *tempconv;
172     int rc;
173
174     temp_dh11=0; hum_dh11=0; temp_bmp=0; pre_bmp=0;
175
176     mosqpp::lib_init();
177     //InitI2C();
178     OpenSerial();
179
180     tempconv = new mqtt_tempconv("outTopic", "localhost", 1883);
181
182     struct sigaction sa;
183     struct itimerval timer;
184
185     /* Install timer_handler as the signal handler for SIGVTALRM. */
186     memset (&sa, 0, sizeof (sa));
187     sa.sa_handler = &timer_handler;
188     sigaction (SIGALRM, &sa, NULL);
189
190     /* Configure the timer to expire after 250 msec... */
191     timer.it_value.tv_sec = 0;
192     timer.it_value.tv_usec = 100000;
193
194     /* ... and every 3 000 msec after that. */
195     timer.it_interval.tv_sec = 1;
196     timer.it_interval.tv_usec = 0;
197
198     /* Start a virtual timer. It counts down whenever this process is executing. */
199     setitimer (ITIMER_REAL, &timer, NULL);
200
201
202     while(1){
203         printf("Luuppaako\n");
204         rc = tempconv->loop();
205         if(rc){
206             tempconv->reconnect();
207         }
208     }
209
210     mosqpp::lib_cleanup();
211
212     return 0;
213 }
```

Liite 3. SendWap.c

(Salminen, T. TAMK 2017)

1 (2)

```
change.log x new 1 x main.cpp x SendWap.c x
1 //
2 // Wapice rest api demo 11.10.2016 KN
3 //
4
5
6 #define devID "14917c9f10a34b769353097507864061"
7
8 #define user_name "KN_tamk"
9
10 #define passwd "Tamk123"
11
12
13
14 #include "SendTW.h"
15
16
17 /////////////////////////////////////////////////// CURL //////////////////////////////////////
18 void send_to_Wap(const char sensor1[], const char path1[],float num1,const char quant1[])
19
20 {
21     CURLcode ret;
22     CURL *hnd;
23     struct curl_slist *slist1;
24
25     std::string wap_url = "https://my.iot-ticket.com/api/v1/process/write/";
26
27     std::string url_id = wap_url + devID;
28
29     slist1 = NULL;
30     slist1 = curl_slist_append(slist1, "Content-Type: application/json");
31
32     int num=int(num1);
33     //float num = num1;
34
35     char sensor[20]=" ";
36     strcpy(sensor,sensor1);
37
38     char path[30]=" ";
39     strcpy(path,path1);
40
41
42     char quant [10]=" ";
43     strcpy(quant,quant1);
44
```

(jatkuu)

2 (2)

```
45
46     char pdata[200] ;
47
48     sprintf(pdata, "[{\"name\": \"%s\", \"path\": \"%s\", \"v\": %d, \"unit\": \"%s\"}]", sensor, path, num, quant) ;
49
50
51     std::string jsonstr = "{ }";
52
53     // jsonstr = "[{\"name\": \"airflow\", \"path\": \"Engine/Primary\", \"v\": 15, \"unit\": \"ls\"}]";
54
55     jsonstr=pdata;
56
57     //printf( "\n");
58
59     //printf(st);
60
61     //printf( "\n");
62
63     hnd = curl_easy_init();
64
65     curl_easy_setopt(hnd, CURLOPT_URL, url_id.c_str());
66
67     curl_easy_setopt(hnd, CURLOPT_NOPROGRESS, 1L);
68
69     curl_easy_setopt(hnd, CURLOPT_USERNAME, user_name);
70     curl_easy_setopt(hnd, CURLOPT_PASSWORD, passwd);
71
72     curl_easy_setopt(hnd, CURLOPT_POSTFIELDS, jsonstr.c_str());
73
74     curl_easy_setopt(hnd, CURLOPT_POSTFIELDSIZE_LARGE, (curl_off_t)71);
75     curl_easy_setopt(hnd, CURLOPT_USERAGENT, "curl/7.38.0");
76     curl_easy_setopt(hnd, CURLOPT_HTTPHEADER, slist1);
77     curl_easy_setopt(hnd, CURLOPT_MAXREDIRS, 50L);
78     curl_easy_setopt(hnd, CURLOPT_SSH_KNOWNHOSTS, "/home/pi/.ssh/known_hosts");
79     curl_easy_setopt(hnd, CURLOPT_CUSTOMREQUEST, "POST");
80     curl_easy_setopt(hnd, CURLOPT_TCP_KEEPALIVE, 1L);
81
82     ret = curl_easy_perform(hnd);
83
84     curl_easy_cleanup(hnd);
85     hnd = NULL;
86     curl_slist_free_all(slist1);
87     slist1 = NULL;
88
```