

Antti Ahola

PoC: Pilvipalveluun tallennettavan terveystiedon parsiminen Pythonilla

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Hyvinvointiteknologia

Insinöörityö

16.02.2016

Tekijä(t) Otsikko Sivumäärä Aika	Antti Ahola PoC: Pilvipalveluun tallennettavan terveystiedon parsiminen Pythonilla 45 sivua + 2 liitettä 16.2.2016
Tutkinto	Insinööri (AMK)
Koulutusohjelma	Hyvinvointiteknologia
Suuntautumisvaihtoehto	Terveysteknologia
Ohjaaja(t)	Yliopettaja Mikael Soini Projektipäällikkö Tero Nurminen
<p>Opinnäytetyön aiheena oli Pilvipalveluun tallennettavan terveystiedon parsiminen Pythonilla. Työssä tutustutaan mHealthiin, pilvipalveluihin sekä proof of conceptin toteuttamiseen vaadittavaan teknologiaan.</p> <p>Työn tavoitteena oli suorittaa proof of concept, jossa BCG-anturista (Ballistocardiography) saatu terveystieto parsitaan Python-ohjelmointikielellä ja siirretään MySQL-tietokantaan. BCG-anturi on erittäin herkkä kiihtyvyyssanturi, joka mittaa sydämen sykkimisestä aiheutuvia kiihtyvyyksiä. Suoritettu työ on osa isompaa kokonaisuutta, jossa anturista saatava informaatio saatetaan pilveen asti ja visualisoidaan. Työ toteutettiin osana Metropolian IoT-projektia.</p> <p>Koko projektissa on Metropolian lisäksi Aalto-yliopisto, Murata Electronics, HUS (Helsingin ja Uudenmaan sairaanhoitopiiri) sekä NurseBuddy. Koko projektin tavoitteena on luoda terveydenhuollon henkilökunnan käyttöön mobiilia, non-invasiivista teknologiaa, joka auttaa potilaiden terveydentilan seurannassa.</p> <p>Työn teoriaosuus toteutettiin käyttämällä Metropolian kirjaston e-kirjoja, Google Scholaria sekä projektin mukana tullutta materiaalia. Työn käytännön osuus suoritettiin projektista saadulla kannettavalla tietokoneella, johon asennettiin tarvittavat ohjelmat, jotka olivat MySQL-tietokanta, Python-ohjelmointikieli sekä Muratan anturin ohjelmisto.</p> <p>Tuloksena saatiin toimiva Python-ohjelma, jonka avulla terveystietoa voidaan parsia ja tallentaa tietokantaan. Saatuja tuloksia voidaan hyödyntää projektin edetessä ja kirjoitettua ohjelmaa voidaan käyttää tulevien ohjelmaversioiden runkona. Työn teoriaosuus hyödyttää IoT-projektiin tulevaa uutta projektityöntekijää tai asiakasta hahmottamaan, mistä IoT-projektissa on kokonaisuudessaan kyse sekä siitä, mitkä ovat projektin tavoitteet.</p>	
Avainsanat	mHealth, Pilvipalvelu, PoC, Terveystieto

Author(s) Title Number of Pages Date	Antti Ahola PoC: Parsing mHealth Sensor's Health Data with Python and Storing Data into the Cloud 45 pages + 2 appendices 16 February 2016
Degree	Bachelor of Engineering
Degree Programme	Health Informatics
Specialisation option	Health Technology
Instructor(s)	Principal Lecturer Mikael Soini Project Manager Tero Nurminen
<p>The subject of this thesis focuses on creating a Proof of Concept for parsing mHealth sensor's health data with Python and storing this data into the cloud. This thesis explores mHealth, cloud computing and technology for creating the proof of concept.</p> <p>The objective of this study was to perform a proof of concept in which health data gathered from BCG (Ballistocardiography) bed sensor is parsed with Python programming language and then sent to the MySQL database. A BCG-sensor is a very sensitive accelerometer, which measures the signals gathered from the pulsation of the heart. The work is part of a bigger entity in which the objective is to gather health data from a bed sensor and send it into the cloud which is then visualized. The thesis was conducted as one part of Metropolia's IoT project.</p> <p>The whole project consists of five organizations including Metropolia UAS, Aalto University, HUS (Helsinki's and Espoo's health care district), Murata Electronics and NurseBuddy. The object of the whole project was to create technology to health care personnel which utilizes mobile and noninvasive technology that helps to monitor patients' state of health.</p> <p>The theoretical part of the thesis was put together by using Metropolia library's e-books, Google Scholar and material from the project. The practical part of the thesis was implemented with a laptop that had the necessary applications installed. The necessary applications were MySQL database, Python object oriented code language and Murata's software for the health sensor.</p> <p>Based on the findings of this thesis, a working Python program, which parses the health data and saves it into the database, was created. The program can be implemented in the upcoming versions of code needed to build the framework for the project in getting the health data into the cloud. The theoretical part can be utilized by both new project workers and customers. It is especially helpful for understanding the bigger picture and the objectives of the project.</p>	
Keywords	mHealth, Cloud Computing, PoC, Health data

Sisällys

1	Johdanto	1
2	Mobiilit mittaukset terveydenhuollossa	2
2.1	mHealth-hankkeiden nousu	5
2.2	Kroonisten sairauksien hoito etämonitoroinnin ja mHealthin avulla	6
2.3	mHealthin haasteet ja mahdollisuudet	8
2.4	mHealthin tulevaisuudennäkymät	10
3	Teknologiaympäristön määrittely ja läpikäynti	12
3.1	Pilvipalvelu	12
3.1.1	Software-, Platform- ja Infrastructure as a Service	15
3.1.2	Pilvipalvelumallit	17
3.1.3	Pilven olennaiset ominaisuudet	18
3.2	Matkapuhelinverkot	19
3.3	Tietokanta	21
3.4	Python	24
3.5	XML-tiedonsiirtomuoto	25
4	Opinnäytetyön tarkoitus ja tutkimusongelmat	27
5	Työn toteutus, Proof of concept	29
5.1	Murata Electronics ja Muratan BCG-anturi	29
5.2	Terveystieto pilveen	32
5.3	Projektin kokonaisarkkitehtuuri	33
5.4	PoC	36
6	Yhteenveto ja johtopäätökset	41
	Lähteet	43

Lyhenteet

1G	First Generation. Ensimmäisen sukupolven matkapuhelinverkkoteknologia.
2G	Second Generation. Toisen sukupolven matkapuhelinverkkoteknologia.
3G	Third Generation. Kolmannen sukupolven matkapuhelinverkkoteknologia.
4G	Fourth Generation. Neljännen sukupolven matkapuhelinverkkoteknologia.
ABS	Anti-lock braking system. Järjestelmä ajoneuvon pyörien lukkiutumattomiseen.
AHIMA	American Health Information Management Association. Yhdysvaltalainen potilasasiakirjojen hallinnointiin keskittyvä ammattijärjestö.
AIDS	Acquired Immune Deficiency Syndrome. Kädellisten eläinten immuunikatotauti.
ART	Anti-retroviral therapy. Hoitokeino HIV/AIDS -sairauksiin.
AWS	Amazon Web Services. Kokoelma Amazonin tarjoamia pilvipalveluita.
B2B	Beat to Beat. Sydämen kahden vierekkäisen sykäyksen välinen aika.
BCG	Ballistocardiography. Sydämen liikkeestä aiheutuva mekaaninen liike, josta muodostettu signaali voidaan tallentaa ja esittää graafisessa muodossa.
CRM	Customer relationship management. Prosessi, jolla hallitaan yrityksen ja asiakkaan suhteita.
DBMS	Database management system. Tietokoneohjelma, jota käytetään kommunikoinnissa tietokantojen, muiden sovellusten ja käyttäjien kanssa.

E2E	End to End. Klassinen tietokoneverkkojen suunnitteluperiaate, jossa informaatio kulkee verkossa aloituspisteestä haluttuun päätepisteeseen asti.
EC2	Amazon Elastic Compute Cloud. Kaupallinen verkkopalvelu tietokoneohjelmien ylläpitämiseen.
EKG	Elektrokardiogrammi. Sydämen toimintaan liittyviä sähköimpulsseja ja sydämen toimintaa kuvaava käyrä.
ESP	Electronic Stability Program. Tiettyjen autonvalmistajien käyttämä nimitys auton ajovirheitä korjaavasta järjestelmästä.
FDA	Food and Drug Administration. Yhdysvaltain elintarvike- ja lääkevirasto.
FiHTA	Finnish Healthtech Association. Suomalaisten terveysteknologiayritysten liitto.
GHz	Gigahertsi. Giga on SI-järjestelmän kerrannaisyksikön etuliite tarkoittaa miljardikertaista. Hz eli hertsi on taajuuden yksikkö. Yhdessä ne tarkoittavat miljardikertaa sekunnissa tapahtuvaa värähdysjaksoa.
HIV	Human immunodeficiency virus. Retrovirus, joka tarttuu ihmisen immuunijärjestelmän tärkeisiin osiin.
HR	Heart Rate. Sydämen lyöntimäärä minuutissa.
HRV	Heart Rate Variability. Sykevälivaihtelu, kuvaa sykkeiden välissä olevan ajan pituutta.
HUS	Helsingin ja Uudenmaan Sairaanhoidopiiri
IaaS	Infrastructure as a Service. Pilvipalvelussa tarjottu palvelun muoto.
IMT-2000	International Mobile Telecommunications-2000. Mobiilien laitteiden ja niiden välisen kommunikoinnin määrittävä standardi.

IoT	Internet of Things. Fyysisten objektien verkko, johon on sulautettu elektroniikkaa, antureita ja internetyhteys, mikä mahdollistaa objektien keskinäisen tiedon vaihdon.
ISP	Internet Service Provider. Internetiin yhteyden tarjoava organisaatio.
IT	Information Technology. Tietokoneiden avulla tehtävä tietojen muokkaus, siirto, tallennus ja haku.
JSON	JavaScript Object Notation. Tiedonsiirtomuoto, jota tietokone sekä ihminen voi lukea.
kbit/s	Kilobit per second. Kilo on SI-järjestelmän kerrannaisyksikön etuliite, tarkoittaa tuhatkertaista. Bitti on tietotekniikan perusyksikkö. Kbit/s kuvaa tiedonsiirtonopeutta.
KEMSA	Kenya Medical Supplies Agency. Kenian valtiollinen lääkintatarvikkeiden logistiikkaan keskittynyt yritys.
MD	Doctor of Medicine. Lääkäreiden ja kirurgien saama titteli, joka saadaan lääketieteellisestä koulusta valmistuessa.
MEMS	Microelectromechanical systems. Hyvin pienen kokoluokan laitteiden teknologia.
mHealth	Mobile Health. Lääketieteellisiä toimenpiteitä, jossa käytetään mobiilia teknologiaa.
NIST	National Institute of Standards and Technology. Yhdysvaltalainen virasto, joka kehittää mittaustekniikoita ja standardeja.
PaaS	Platform as a Service. Pilvipalvelussa tarjottu palvelun muoto.
PoC	Proof of Concept. Uuden tuotteen, palvelun tai konseptin testaus.
RAID	Redundant Array of Inexpensive Disks. Tiedon tallennukseen liittyvä tekniikka.

RR	Respiration Rate. Hengitystaajuus on henkilön minuutissa ottamien henkäysten määrä.
SaaS	Software as a Service. Pilvipalvelussa tarjottu palvelun muoto.
SGML	Standard Generalized Markup Language. Dokumenteille laadittu yleistetty merkintäkieli.
SMS	Short Message Service. Matkapuhelinverkossa lähetettävä lyhyt viesti.
SV	Stroke Volume. Vasemman kammion yhdellä sykäyksellä pumpattu verimäärä.
WAN	Wide Area Network. Tiedonsiirtoverkko, joka peittää laajoja maantieteellisiä alueita.
WHO	World Health Organization. Ihmisten terveyteen keskittyvä Yhdistyneiden kansakuntien järjestö.
WLAN	Wireless Local Area Network. Langaton lähiverkkotekniikka verkkolaitteiden yhdistämiseen ilman kaapeleita.
VM	Virtual Machine. Virtuaalinen kopio tietystä tietokonejärjestelmästä.
VPN	Virtual Private Network. Mahdollistaa käyttäjien datan välityksen julkisen verkon yli, mistä saadaan yksityisverkon turvallisuushyödyt.
WWW	World Wide Web. Avoimen lähdekoodin informaatiopankki, johon saadaan yhteys internetin avulla.
XML	Extensible Markup Language. Merkintäkieli, joka on sekä ihmiselle, että tietokoneelle ymmärrettävä.

1 Johdanto

Tämä opinnäytetyö suoritetaan osana Metropolian IoT-projektia (Internet of Things), johon on saatu teknologiateollisuuden myöntämä Health Capital & Data Given Healthcare -apuraha. Projektissa työskentelee insinööriöntekijöitä, projektiin palkattuja henkilöitä sekä Metropolian lehtoreita. Koko projektin johtajana on Metropolian tietoverkkojen osaamisaluepäällikkö Tero Nurminen. Projektissa on Metropolian lisäksi Aalto-yliopisto, Helsingin ja Uudenmaan Sairaanhoidopiiri, Murata, FiHTA (Finnish Healthtech Association) sekä NurseBuddy.

IoT-projektin tarkoituksena on kehittää hoitohenkilökunnalle teknologiaa, joka parantaa ja helpottaa potilaan elintoimintojen seurantaa. Teknologia on non-invasiivista, verkko- ja pilvipalveluita hyödyntävää. Metropolian tehtävä projektissa on huolehtia tiedon kulemisesta mittaavasta anturista pilveen asti, tiedon tallettamisesta pilvessä olevaan tietokantaan sekä tiedon visualisoinnista. Aalto-yliopisto sekä NurseBuddy huolehtivat tiedon analysoimisesta. Tämän lisäksi Metropolian on tarkoitus toimia pilvipalvelun tarjoajana projektissa. Tämä tarkoittaa sitä, että potilastiedot eivät mene julkiseen pilveen, kuten Googlen tai Microsoftin pilveen, vaan ne menevät Metropolian ylläpitämään ja hallinnoimaan pilveen, missä voidaan taata potilastietojen turvallisuus.

Opinnäytetyön tarkoituksena on perehtyä pilvipalveluihin ja mHealthiin, suorittaa PoC (proof of concept) terveystiedon välityksestä pilveen sekä pohtia PoC:in toteutusta, miksi päädyttiin juuri kyseiseen ratkaisuun. Opinnäytetyö tuo esiin mHealthin teknologian ja liiketoiminnan näkökulmia. Työ kertoo erilaisten esimerkkien avulla mahdollisuuksista, kuinka mHealthin teknologia voi auttaa terveydenhuollon henkilökuntaa terveydenhuollossa. Tämän lisäksi opinnäytetyö tuo esiin IoT-projektille tärkeitä teknologioita, joita tarvitaan terveystiedon kuljettamisessa mittaavasta anturista pilveen.

Opinnäytetyön teoriaosuudessa perehdytään kahteen suureen kokonaisuuteen, jotka molemmat tukevat työn käytännön osuutta. Ensimmäinen suuri kokonaisuus perehtyy luvussa kaksi mobiileihin terveystmittauksiin terveydenhuollossa. Siinä pohditaan mobiilien mittausten tulevaisuuden suuntia, käydään läpi kroonisiin sairauksiin jo käytössä olevia mobiileja hoitomenetelmiä ja -teknologioita sekä tutustutaan erilaisiin esimerkkita-pauksiin jo olemassa oleviin mobiiliin terveydenhuollon palveluihin ja -laitteisiin. Tämä kokonaisuus käydään läpi, jotta lukija saisi tietämystä myös markkinoilla olevista muista

mobiiliin terveydenhuollon teknologioista. Tämä antaa myös tieto- ja tutkimuspohjaa mHealthiin, joka on IoT-projektin pääteemana.

Luvussa kolme tutustutaan projektissa käytettyihin teknologioihin, jotka tulevat esiin myös työn toteutusvaiheessa. Luvussa kolme käyty teoria auttaa lukijaa ymmärtämään opinnäytetyön toteutusvaihetta. Teknologiaympäristön määrittely ja läpikäynti -luvussa perehdytään tämän lisäksi laajasti pilvipalveluihin ja pilvipalveluiden olennaisiin ominaisuuksiin, sillä opinnäytetyö suoritetaan Metropolian IoT-projektin yhteydessä, mikä on Metropolian johtava pilvipalveluiden kehityslaboratorio.

Kokonaisuudessaan opinnäytetyö muodostuu kuudesta luvusta, jossa perehdytetään lukija opinnäytetyöhön ja työssä vallitsevaan skenaarioon. Luvut kaksi ja kolme ovat opinnäytetyön teoriaosuutta. Luvuissa neljä ja viisi perehdytään opinnäytetyön tarkoitukseen, tutkimusongelmaan ja toteutukseen. Luku kuusi kokoaa käydyn asian yhteen ja tuo esille, kuinka saatuja tuloksia voidaan jatkossa hyödyntää.

2 Mobiilit mittaukset terveydenhuollossa

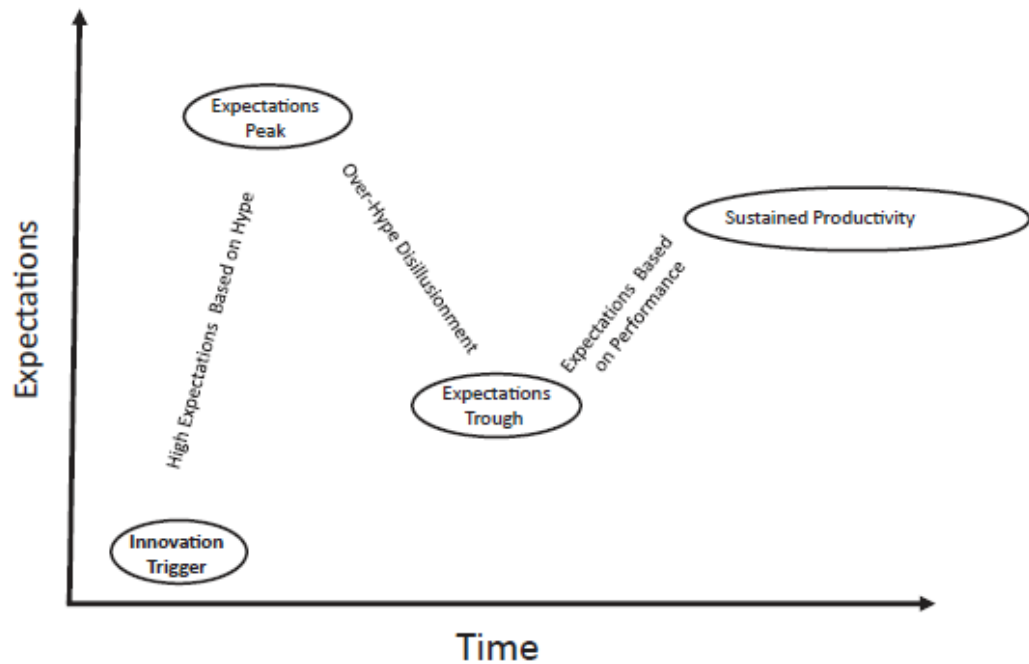
”mHealth on aikamme suurin teknologinen läpimurto koskien suurinta kansallista haastettamme”, sanoi Yhdysvaltojen sosiaali- ja terveysministeri Kathleen Sibelius vuonna 2011 pidetyssä mHealth-huippukokouksessa Washington DC:ssä. mHealth, kirjoitetaan myös m-health, tulee englannin kielen sanoista mobile (engl. mobile - saavutettava, liikkuva) ja health (engl. health - terveys). mHealthilla ei ole standardoitua tai hyväksyttyä määritelmää, mutta yleensä sen ajatellaan parantavan terveydenhuollon saavutettavuutta, mikä tekee terveydenhuollosta myös nopeampaa, parempaa ja halvempaa. [1.]

mHealthin on määritellyt myös American Health Information Management Association (AHIMA), jonka kyseistä määritelmää aion käyttää opinnäytetyössäni:

“the use of devices such as smartphones or tablets in the practice of medicine, and the downloading of health-related applications or ‘apps’ ... [to] help with the flow of information over a mobile network and ... improve communication,” specifically between individuals and clinicians. (Lähde: AHIMA Guide 2013)

mHealth on muuttunut vuosikymmenessä mielenkiintoisista terveyssovelluksista terveydenhuollossa potilailla käytettyihin sovelluksiin. mHealthin avulla terveysalan on odotettu siirtyvän lähitulevaisuudessa enemmän kohti personalisoitua, osallistavaa, ennaltaehkäisevää ja halvempaa terveydenhuoltoa. Tällä muodonmuutoksella on arvioitu olevan myös globaaleja vaikutuksia. Terveysalan raporteissa arvioidaan, että mHealthin markkinat ovat kasvamassa räjähdysmäisesti lähitulevaisuudessa. [1.]

Vaikka mHealth ei ole vielä saavuttanutkaan kestävän tuottavuuden (engl. sustained productivity) statusta niin sanotulla hypekäyrällä (kuva 1), on sen kuitenkin odotettu saavuttavan kyseinen taso lähitulevaisuudessa. Esimerkki kestävän tuottavuuden saavuttaneesta ilmiöstä on internet. 1990-luvun alussa vähän käytetty teknologia on nykyään vallalla kaikkialla ympäri maailman. Uusista teknologioista puhuttaessa käytetään usein teknologiahype-käyrää, joka ilmaisee uuden teknologian evoluution tai elämäncyklin ajan funktiona. Hetki jona innovaatio on saanut kestävän tuottavuuden statuksen, vaihtelee laajasti. Kaikilla kestävän tuottavuuden statuksen saaneilla tapauksilla innovaatiolla tulee olla hyvin suuri käyttäjäkunta. [1.]



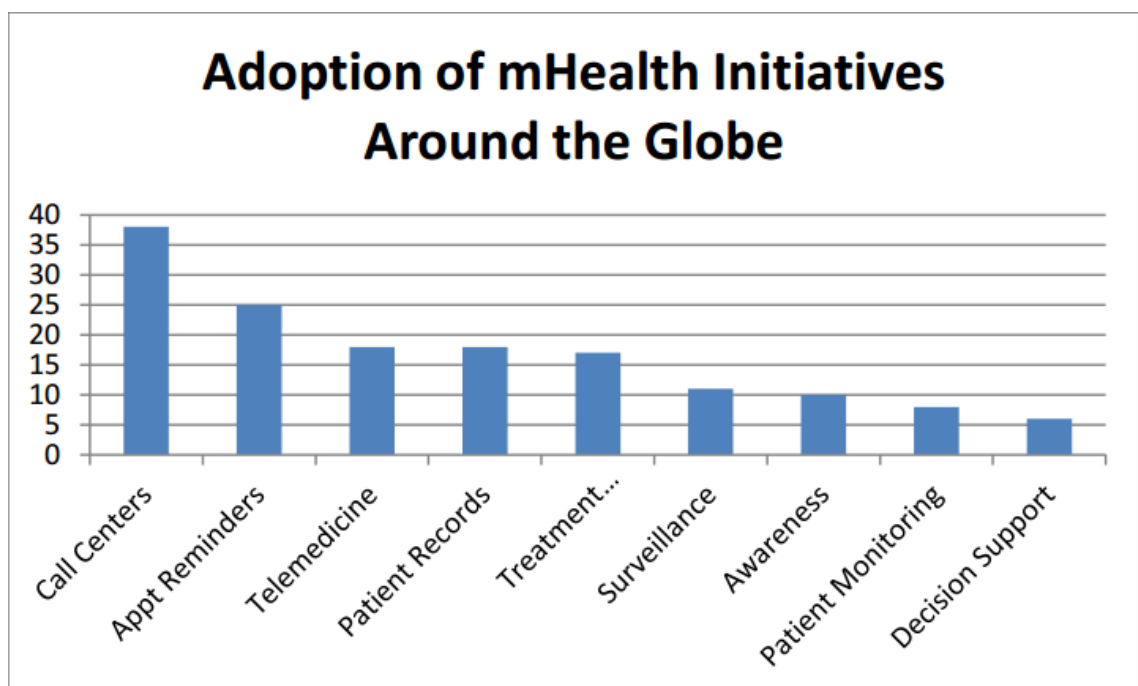
Kuva 1. Syntyvien teknologioiden hypevaikutus, "hypekäyrä" [1.]

Miksi mHealthia ylipäätän halutaan käyttää? On arvioitu, että mHealth laskee kustannuksia terveyssektorilla ja parantaa terveydenhuoltoa erityisesti kehittyvissä maissa. Suurimpana syynä mHealthin kasvuun voidaan pitää ikärakenteen muuttumista. Yhä suurempi osa väestöstä elää vanhaksi, ja vanhusten suhteellinen määrä koko väestöön kasvaa kaikissa kehittyneissä maissa huomattavaa vauhtia. Esimerkkejä maista, joissa väestön huoltosuhde on lähellä yhtä, ovat Suomi, Saksa, Japani sekä Yhdysvallat. Huoltosuhde tarkoittaa työssäkäyvien suhdetta työssä käymättömiin. Esimerkiksi jos huoltosuhde on kaksi tarkoittaa se sitä, että kaksi ihmistä käy töissä yhtä työssä käymättöntä kohden. Vuoden 2009 väestöennusteen mukaan vuonna 2030 Suomessa yli 65 vuotta täyttäneitä arvioidaan olevan 26 prosenttia väestöstä, kun vuoden 2010 lopussa se oli 17,5 prosenttia [2]. Ikärakenteen muutoksella on monia vaikutuksia. Muun muassa kasvaneet kroonisten sairauksien määrät vaativat jatkuvaa lääkinällistä hoitoa. Mobiilit teknologiat mahdollistavat lääkinällisen hoidon ja seurannan kotona, mikä vähentää sairaalakäyntien tarvetta. Vähentyneet sairaalakäynnit alentavat hoidon kustannuksia jatkuvan hoidon parantaessa potilaan elämänlaatua. [1.]

mHealth voi muuttaa terveydenhuollon segmenttejä, erityisesti kroonisten sairauksien hoidossa sekä yleislääkärikäyntejä. Se parantaa potilaan kokemusta saadusta hoidosta, sillä mHealth mahdollistaa uudenlaisen itsehoidon.

2.1 mHealth-hankkeiden nousu

WHO:n (World Health Organization) vuonna 2011 järjestämän kyselyn perusteella mHealth-hankkeita on järjestetty runsaasti. Kyselyyn vastasi 114 maata. Yleisin perustettu hanke oli terveystalvelukeskus, jossa vastattiin potilaiden kysymyksiin koskien heitä vaivaavia terveydellisiä ongelmia. Kuvassa 2 näkyvät vuoden 2011 toteutettujen terveyshankkeiden lukumäärät eniten toteutetusta vähiten toteutettuun. Terveystalvelukeskus-hanketta seurasi usein myös muita mHealth-hankkeita, kuten SMS-pohjaiset (Short Message Service) muistutukset tulevista lääkärikäynneistä, omien potilasrekistereiden seuranta internetistä sekä potilaalle määrättyjen hoitomenetelmien valvonta.



Kuva 2. mHealthin käyttö eri hankkeissa, N = 114 osallistujamaata.

Ei yllättänyt suuresti, että eri maiden välillä havaittiin suuria eroja eri hankkeiden omak-suttavuudella ja käyttöönotolla. Erot olivat suuria varsinkin kehitysmaissa, jossa

mHealthin käyttöönotto oli vähäisintä, ja länsimaissa, jossa mHealth otettiin käyttöön varsin laajasti. Osallistujamaiden suurimpina haasteena olivat liian niukka budjetointi mHealthin toteutukselle sekä liian pieni henkilökunnan määrä. Myös huolenaiheet yksityisyydenturvasta sekä tiedon turvallisuudesta nousivat esille. [3.]

2.2 Kroonisten sairauksien hoito etämonitoroinnin ja mHealthin avulla

Krooniset sairaudet ovat suurin haaste monien maiden terveydenhuollossa, erityisesti Yhdysvalloissa. Etämonitorointilaitteiden avulla kroonisesti sairaat potilaat pystyvät tallentamaan heidän terveydentilansa sekä terveydentilan muutokset ja lähettämään sähköisesti heidän lääkärilleen. Tämä vähentää turhia lääkärikäyntejä ja vähentää terveydenhuollon kustannuksia sekä tehostaa lääkäreiden toimintaa. [3.]

Brookings-instituutin tilaaman analyysin mukaan etämonitorointi ja mHealth voisivat säästää 197 miljardia dollaria seuraavan 25 vuoden aikana pelkästään Yhdysvalloissa. Analyysissä todettiin, että erityisesti kroonisten sairauksien, kuten sepelvaltimotautien, keuhkosairauksien, diabeteksen sekä haavaumien hoidossa voitaisiin tehdä merkittäviä säästöjä. Ympäri vuorokautisella monitoroinnilla ja sähköisellä tiedonsiirrolla pystytään tehokkaasti karsimaan potilaat, jotka vaativat välitöntä lääkinnällistä hoitoa sekä huomaamaan potilaan tilan muutos ja reagoida siihen jo ennen vakavia komplikaatioita. [3.]

Esimerkiksi sokeritautipotilailla on erittäin tärkeää monitoroida heidän veren glukoosipitoisuuttaan ja annostella insuliini oikein veren sokeriarvojen perusteella. Ennen diabetespotilaiden tuli käydä lääkärissä säännöllisin väliajoin, missä heiltä mitattiin sokeriarvot. Tämän jälkeen jouduttiin odottamaan arvojen tuloksia, mikä kaikki kulutti aikaa ja oli hankalaa kaikille prosessiin osallistuneille. Nykyään on kuitenkin mahdollista monitoroida glukoositasoja kotoa käsin. Mitatut arvot siirtyvät välittömästi sähköisessä muodossa henkilökohtaiselle lääkärille sekä halutuille muille osapuolille, kuten puolisolle. Potilaat käyttävät ”glukopuhelimia” (kuva 3), jotka monitoroivat ja lähettävät tiedon lääkärille samalla muistuttaen heitä seuraamaan sokeriarvojaan säännöllisin väliajoin. Näin potilaat itse ovat vastuussa sokeritasapainostaan, ja he voivat olla pois lääkärin vastaanotolta, mikä parantaa heidän elämänlaatuaan sekä vähentää turhaa työkuormaa lääkäriltä. Yhdysvalloissa on noin 24 miljoonaa diabeetikkoa, joista noin puolet, 11 miljoonaa, käyttää glukoosin etämonitorointilaitteita. [3.]



Kuva 3. Glukopuhelin. Sokeriarvojen mittaaminen glukopuhelimella. [4.]

Esimerkki käytössä olevasta etämonitoroinnista kroonisten sairauksien hoidossa on Kiinasta. Qualcomm Wireless Reach™ on laukaissut yhdessä Life Care Networksin sekä Community Health Association of Chinan kanssa EKG:tä (Elektrokardiogrammi) havaitsevan luurin tai puhelimen (kuva 4), joka tallentaa yhtäjaksoisesti 30 sekuntia sydämestä tulevaa tietoa, ja lähettää sen sähköisesti eteenpäin ympäri vuorokauden auki olevaan Life Care Networks Call Centeriin Pekingiin. EKG:llä tarkoitetaan sydänsähkökäyrää. Kyseisellä toimijalla on yli 40 lääkäriä työsuhteessa. Langattomat monitorit mahdollistavat etädiagnoosin potilaille, joiden asuinalueella on liian vähän lääkäreitä tarjoten reaaliaikaisen palautteen henkilöille, joilla on sydänperäisiä sairauksia. Sydänsairaudet tappavat pelkästään Kiinassa vuosittain 3 miljoonaa ihmistä. Erityisesti suurien kaupunkien ulkopuolella asuvien ihmisten on vaikeaa päästä hoitoon, suurista etäisyyksistä johtuen. Tämä ohjelma on saanut Computerworld Honors Laureate -palkinnon vuonna 2012. [3.]



Kuva 4. Qualcomm:in EKG-puhelin [9].

HIV (Human immunodeficiency virus) ja AIDS (Acquired Immune Deficiency Syndrome) ovat suuri kansallinen haaste Kenialle, jonka väestöstä 6,3 prosentilla on diagnosoitu HIV [3]. Kenian valtion tärkein keino HIV:iä ja AIDS:ia vastaan on antiretroviraalit hoidot (ART), jotka koostuvat antiretroviraaleista-lääkkeistä. Niiden tarkoituksena on pysäyttää HIV:n eteneminen ja heikentää jo päällä olevan viruksen tilaa. Lääke on toimiva erityisesti heti HIV-tartunnan saannin jälkeen [5]. 3G-tekniikan avulla terveydenhuollon työntekijöiden hallinnollinen taakka on keventynyt ART:n farmaseuttisen hallintoyksikön parantuessa, mikä antaa heille enemmän aikaa hoitaa potilaita. Farmaseuttisen hallintoyksikön parantumisella tarkoitetaan sitä, että terveydenhuollon henkilökunnan päätelaitteiden käyttö on helpottunut ja tehostunut, mikä mahdollistaa pidemmän ajan potilaiden hoitoon. Konkreettisia tuloksia mHealth-prosessin tuomista parannuksista oli 70 prosentin nousu ajallaan vastattuihin HIV/AIDS -raportteihin sekä keskimääräinen raportin palautus KEMSA:an (the Kenyan Medical Supplies Agency) laski kahdeksasta tunnista viiteen minuuttiin. [3.]

2.3 mHealthin haasteet ja mahdollisuudet

Jotta mHealthin tulevaisuus näyttäisi paremmalta, on sillä edessään vielä monia haasteita. Yksi suurimmista huolenaiheista on tiedon yksityisyyden säilyttäminen ja tiedonsiirron turvallisuuden takaaminen. Tiedon yksityisyyden sekä tiedonsiirron turvallisuuden tulee säilyä sekä lääkärin että potilaan käyttäessä mobiilisovelluksia. Vaikkakin kuluttajat ovat hiljalleen uskaltaneet käyttämään älypuhelimiaan, esimerkiksi laskujen maksa-

misessa ja verkkopankissa asioimisessa, ei se tarkoita, että he olisivat turvassa tietomurroilta ja -varkauksilta. Päinvastoin, se kohdentaa rikollisten toimintaa mobiililaitteisiin kohdistuviin tietovarkauksiin. Viime vuosina mediassa esille nousseet tietomurrot sekä vakoilutapaukset, kuten Edward Snowdenin tapaus, ovat kuitenkin nostaneet kuluttajien tietosuutta tietoturvan tärkeydestä. [3.]

Toinen suuri haaste, jotta mHealthin aseman voimistuisi, on keskittyminen loppukäyttäjiiin mobiileja terveyssovelluksia suunniteltaessa. Sovellukset, joita on vaikeita käyttää, jäävät mitä todennäköisemmin käyttämättä. Saksalaisen yrityksen Research2Guidancen teettämän tutkimuksen mukaan markkinoilla on tällä hetkellä yli 1100 diabetekseen liittyvää sovellusta. Suurimmalla osalla sovelluksista käyttäjiä on vain kourallinen, sillä sovellusten käyttö on vaivalloista. Tietoa täytyy syöttää käsin, eivätkä ne toimi potilaan aikaisemman verensokerimittarin kanssa, tai ne eivät yksinkertaisesti osaa lukea veren glukoosipitoisuuksia tarpeeksi tarkkaan. Käyttäjälähtöinen tuotekehitys sovelluksia suunniteltaessa on erittäin tärkeää, ja se määrittääkin, mitkä sovelluksista nousevat myyntimenestyksiksi ja mitkä jäävät tusinatavaraksi muiden keuhkojen sovellusten kanssa. [3.]

Myös sääntelyn epävarmuuden eliminoiminen on haasteena mHealthin kasvulle, erityisesti Yhdysvalloissa. Teknologia kehittyy suurin harppauksin joka vuosi ja tässä kehityksen tahdissa on FDA:n (Food and Drug Administration) ollut vaikeaa pysyä mukana. Vaikka FDA on yrittänyt erottaa mobiililaitteissa olevien terveyttä- ja sairautta hoitavien sovellusten välisiä eroja, on sääntelyssä kuitenkin vielä huomattavia epäselvyyksiä sekä epävarmuuksia. Epävarmuus nousee esiin, koska lääkinnälliselle laitteelle olevalle lisälaitteelle tarvitaan tarkat määritelmät, samoin kun tarvitaan sovelluksille, jotka tukevat lääkinnällisessä päätöksenteossa. [3.]

mHealthista ei ole vielä olemassa aineistoa, mikä olisi tieteellisesti todistanut, että mHealthilla on vaikutus yksilön terveyteen. Tähän asti kaikki, mukaan lukien sijoittajat, palveluntarjoajat, kuluttajat sekä valtiolliset toimijat ovat pelkästään uskoneet, että mHealth tuo hyötyjä. Tästä ei siis ole vielä konkreettisia todisteita. Jotta päästäisiin mHealthia koskevissa asioissa eteenpäin, tarvitsevat edellä mainitut sidosryhmät todisteita siitä, että mHealth saavuttaa sille määrättyt tavoitteet. [3.]

2.4 mHealthin tulevaisuudennäkymät

Vaikkei todisteita mHealthin hyödyistä ole vielä todennettu, on sillä mitä todennäköisimmin suuri vaikutus tulevaisuuden terveydenhuoltoon ja terveystalouden käytettävyyteen. Asiaa arvioi vuonna 2012 Harvardin julkisen terveyden osaston tohtori Dean Julio Frenk, joka arvioi positiivisesti mHealthin tulevaisuutta. [3.]

”The use of cellular phones for health care and public health is one of the most promising developments in the quest to achieve universal health care coverage worldwide because mobile phones are rapidly becoming the communication technology of choice—and increasingly so among the poor.” (HSPH News 2012)

Jo yli 20 vuotta sitten Lontoossa Kingston-yliopiston tietoliikenteiden professori, tohtori Robert Istepanian, määritteli termin mHealth, joten mHealthista puhuminen aivan uutena ilmiönä ei ole totuudenmukaista. Vieläkään emme pysty täysin sanomaan, mihin suuntaan ja kuinka kauan mHealthin kehitys tulee kestämään. [3.]

mHealth on mielletty perinteisiä keinoja hajottavana innovaationa vaihtaen totut menetelmät uusiin menetelmiin, jotka voidaan tehdä paljon halvempaan hintaan. mHealthin tulevaisuudelle teknologia tulee olemaan helpompia osa-alueita. Toisin sanoen teknologian kehitys ja sen implementointi mHealthiin ei tule olemaan ongelma, vaan suurimmat haasteet tullaan kohtaamaan länsimaissa terveydenhuollon järjestelmässä, josta esimerkkinä Yhdysvallat. Heidän terveydenhuolto on pirstaloitunut ja terveydenhuollon yksiköt ovat irrallisia kokonaisuuksia. Tämän korjaamiseen vaaditaan terveydenhuollon rakenteellisia muutoksia. Tällä tarkoitettaisiin Digital Health Groupin M.D. (Medical Doctor) Eric Dishmanin mukaan kustomoitua hoitoa, hoivan verkottumista tarkoittaen, että potilasta hoitavat henkilöt kommunikoisivat paremmin keskenään sekä hoidon skaalautuvuuden helpottamista. [3.]

mHealthin arvioidaan tuovan tasa-arvoa hoidon saantiin. Asiaa on tutkittu ja on saatu tutkimustuloksia, joiden valossa afroamerikkalainen sekä latinalaisamerikkalainen väestö käyttävät puhelimiin enemmän terveyteensä liittyvissä asioissa, kuten tiedon etsinnässä, kuin valkoihoiset amerikkalaiset. Tämä tieto on erityisen hyödyllistä suunniteltaessa tulevaisuuden terveystalouksia, jotka olisivat myös vähemmistöryhmien saavutettavissa. [3; 7.]

Tällä hetkellä nousevia trendejä ympäri maailman, joilla on vaikutuksia tulevaisuuden terveydenhuoltoon, ovat muun muassa itsehoidon kasvu, vähempi tarve sairaaloille sekä lääkinälliselle henkilökunnalle, jotka suorittavat testejä potilailla. Myös kasvanut valmius jakaa omia terveystietoja, voi johtaa markkinoiden enempään vapauttamiseen samalla nostaen tiedonsiirron ja tallennuksen kustannuksia. Edellä mainitut oletukset on voitu tehdä globaalien tutkimusten perusteella. Esimerkiksi Intelin tekemä kansainvälinen tutkimus, johon osallistui yli 12 000 aikuista eri puolilla maailmaa, paljasti, että 80 prosenttia vastanneista olivat optimistisia mHealthin käytön suhteen. Tämän lisäksi osallistujat osoittivat itsevarmuutta kyvyissään suorittaa lääkinällisiä testejä, kuten verenpaineen mittausta itseltään. Monet olivat valmiita jopa käyttämään nieltäviä monitorointilaitteita. [3; 6; 7.]

Intelin tutkimus paljasti myös, että erityisesti Yhdysvaltojen ulkopuolella asuvat eivät kokeneet sairaaloiden hyödyntämistä tärkeäksi. Erityisesti syrjäisillä paikoilla asuvat olivat tätä mieltä. Peräti 57 prosenttia tutkimukseen vastanneista ajattelivat, että sairaalat tulevat olemaan turhia tulevaisuudessa. Kokonaisuudessaan vastaukset mHealthia koskien olivat erittäin optimistisia ja mielipiteet mHealthia tukevia. Kohokohtia Intelin kansainvälisestä terveydenhuollon teknologioiden tutkimuksesta: [3; 6; 7.]

- 84 % vastanneita olivat myöntyväisiä jakamaan henkilökohtaiset terveystiedot.
- Yli 70 % ihmisistä olivat avoimia vessassa käytettäviin sensoreihin sekä nieltäviin etäseurantalaitteisiin.
- Yli puolet (53 %) luottaisivat heidän itsensä tekemiin testeihin yhtä paljon kuin lääkärin suorittamaan samaan testiin.
- Puolet kyselyyn vastanneista luottaisivat lääkärin tekemään diagnoosiin, joka oltaisiin saatu aikaan videokonferenssissa. [6.]

3 Teknologiaympäristön määrittely ja läpikäynti

Nykyään elämme maailmassa, missä mobiiliteknologia on kaikkialla. ”Mobiili” yksinkertaisesti tarkoittaa, ettemme ole enää riippuvaisia langallisista yhteyksistä käyttäessämme tietokoneita, kommunikointilaitteita tai tiedonlähteitä. Voimme käyttää älypuhelimia, tabletteja ja muita taskuun mahtuvia laitteita missä tahansa. Näin saadaan yhteys kehen tahansa mihin vuorokauden aikaan tahansa. Voimme katsoa elokuvan, tilata kahvin tai saada ajo-ohjeet haluamaamme paikkaan, samanaikaisesti. Monet lapset ja nuoret eivät olekaan nähneet lankapuhelimia saati pysty kuvittelemaan elämää ilman langatonta yhteyttä ja ympärivuorokautista pääsyä internetiin. Heille historia alkaa ensimmäisen sukupolven iPhone-puhelimista. Koska teknologia mahdollistaa, että olemme yhdistettynä kaikkiin verkossa oleviin huolimatta missä he sijaitsevat, voi henkilö, johon olemme yhteydessä olla naapurissa tai maailman toisella puolen. Maailma on kirjaimellisesti kätemme ulottuvissa. [3.]

Tässä luvussa määritellään esimerkkiteutuksessa käytetyt teknologiat, eli teknologiat, jotka on olennaista ymmärtää terveystiedon siirtämisessä käyttäjältä pilveen. Kyseisiä teknologioita ovat internet, mobiiliverkot, tietokannat, python-engine sekä tiedonlähetyksimuodot. Esimerkkiteutus esitetään luvussa viisi. Tämän lisäksi perehdytään syvällisemmin pilvipalveluihin.

3.1 Pilvipalvelu

Amerikkalaisen National Institute of Standards and Technology (NIST) on määritellyt vuonna 2011 pilvipalvelun tai pilvilaskennan käsitteen (Engl. Cloud Computing):

”Pilvilaskenta on malli, joka mahdollistaa kaikkialla olevan, kätevän, heti käytettävän internet-yhteyden muokattaviin tietokonelaskentaresursseihin (esimerkiksi verkot, palvelimet, tallennustilat, sovellukset, sekä palvelut). Näitä resursseja voidaan jakaa nopeasti pienellä hallinnoinnilla sekä vähäisellä kanssakäymisellä palveluntarjoajan kanssa. [10.]

Pilvipalvelu on globaali teknologia, joka on hyvin pitkälle automatisoitu, ja mahdollistaa kaikenkokoisille yrityksille vaihtoehdon yrityksen informaatiojärjestelmään. Nykypäivän äärimmilleen kilpaillussa yritysmaailmassa jokainen yritys tarvitsee luotettavan tieto- ja laskentajärjestelmän. Perinteisesti jokaisen yrityksen täytyi kehittää yritykselle oma tietotekninen osaaminen ja järjestelmä. Tämä aiheutti suuria ongelmia erityisesti pienissä ja keskisuurissa yrityksissä, joilla ei ollut varaa tai aikaa keskittyä liikeideansa lisäksi muuhun toimintaan. Vuonna 2006 Amazon toi markkinoille ensimmäisenä pilvipalvelun Elastic Compute Cloud (EC2). Elastic Compute Cloud mahdollisti pienyrityksille tietotekniikan ulkoistamisen, jättäen resursseja yrityksen toiminnan kehittämiseen. Yrittäjät vuokrasivat Amazonilta tietokoneita tuntihintaan, missä he ajoivat ohjelmistojaan. Amazonin oltua pioneerina pilvipalvelualalla, sai se suuren etumatkan muihin alan yrityksiin. Se onkin edelleen suurin, monipuolisin ja käytetyin pilvipalveluiden tarjoaja niin yksityiskäytössä kuin yrityksilläkin. [11.]

Yrityksiä pilvipalveluissa eniten houkuttaa mahdollisuus saada täysin toimiva tietojärjestelmä muutamassa tunnissa tai korkeintaan muutamassa päivässä, jos haluttu järjestelmä on monimutkainen. Tietojärjestelmällä tarkoitetaan järjestelmää, jossa tietokoneet ovat yhdistettyinä rypäkseen muita tietokoneita, jotka jakavat tiedon tallennusjärjestelmän ja lukuisten päätelaitteiden kuten printtereiden, skannereiden tai reitittimien kanssa. Jokainen järjestelmään yhdistetty tietokone voi toimia itsenäisesti, mutta se voi kommunikoida toisten laitteiden ja tietokoneiden kanssa [12]. Pilvipalvelualusta antaa yrityksille mahdollisuuden valita erittäin laajasta valikoimasta ominaisuuksia. He voivat valita esimerkiksi tarvitsemansa laitteiston, palvelun tyyppin, johon palataan tarkemmin myöhemmin luvussa, tarvittavat sovellukset sekä tallennustilan määrän.

Pilvipalveluiden asiakkaat pääsevät ostamaansa järjestelmään internetyhteyden avulla. Asiakkaat, jotka puolestaan vaativat suurempaa turvatasoa kommunikoinnilleen pilvessä, voivat käyttää Virtual Private Network (VPN) -yhteyttä, jonka tarjoaa Internet Service Provider (ISP). VPN, jota voi esimerkiksi käyttää tietty yritys, opetusinstituutti tai valtion virasto, on verkkoteknologia, joka tekee turvallisen internetympäristön julkisen internetin jatkoksi [13]. ISP on organisaatio, joka tarjoaa internetyhteyden. Esimerkiksi Elisa on internet service provider [14]. Palveluntarjoajan yhteysnopeus määrittää yrityksen ja pilven välisen kommunikointinopeuden. Esimerkiksi 10 -100 megabit/s internetyhteydet ovat nykyään edullisia ja mahdollinen vaihtoehto monille yksityiskäyttäjille sekä yrityksille yhteyden muodostamiseksi pilveen. [14.]

Muita pilvipalvelun tuomia hyötyjä ovat muun muassa mahdollisuus skaalata tarvittavien tietokoneressurssien määrää erittäin pienellä varoitusajalla, monien sovellusten käyttö ilman lisenssivaatimuksia, maksu ainoastaan käytetystä palvelusta sekä mahdollisuus käyttää erittäin suuria laskentakapasiteetteja. Eniten pilvipalvelut hyödyttävät yrityksiä siinä, että palvelut ovat saatavilla ilman minkäänlaista ylläpitoa tai hallinnointia, mikä kulluttaisi resursseja ja veisi huomiota yritykseltä. [11.]

Valittaessa pilvipalvelun tarjoajaa asiakkaalla on monta hyvää vaihtoehtoa valita. Suurimpia pilvipalveluiden tarjoajia ovat Amazon Web Services (AWS), Microsoft Office 365 ja Windows Azure, Google Apps, Rackspace sekä Salesforce. Amazonin ollessa suurin Microsoft 365 tarjoaa kaikki Office-pakettiin kuuluvat tuotteet kuten Wordin, Excelin, PowerPointin sekä Outlook-emailin. Azure puolestaan tarjoaa huippuluokan tietokanta-, hakukone- sekä tiedontallennuspalveluita yksilöille ja yrityksille. Google Apps tarjoaa tunnetut sovellukset kuten Gmailin, Google Docsin sekä Google Driven. Rackspace tarjoaa kaikki tavalliset pilvipalvelut keskittyen asiakastukeen, Fanatical Supporttiin. Salesforce on markkinajohtaja asiakkuudenhallintaohjelmissa, Customer Relations Management (CRM) software. CRM:llä tarkoitetaan toimintamallia ja liiketoimintastrategiaa, jolla pyritään vahvistamaan yrityksen ja asiakkaan välisiä suhteita [15]. Kuvassa 5 on suurimpien pilvipalveluiden tarjoajien logot.



Kuva 5. Markkinoiden suurimmat pilvipalveluiden tarjoajat.

Suurimmat pilvipalveluiden tarjoajat ovat investoineet miljardeja dollareita infrastruktuuriinsa. Tähän kuuluvat huippunopeat internetyhteydet sekä jalkapallokenttien kokoiset

konesalit, joissa sijaitsee kymmeniä, peräti satojatuhansia tietokoneita. Jokaista konesalia ympäri vuorokauden huoltaa enintään muutamasta kymmenestä insinööristä koostuva ryhmä. Esimerkki Suomessa olevasta konesalista on Googlen Haminassa sijaitseva konesali. Konesalit ovat yrityksille korvaamattoman arvokasta omaisuutta ja niitä valvotaankin erittäin tarkkaan. Vain yrityksen korkeimmat johtohenkilöt pääsevät vierailemaan datakeskuksissa siellä työskentelevien lisäksi. Kuvassa 6 on Amazonin konesalista otettu kuva.



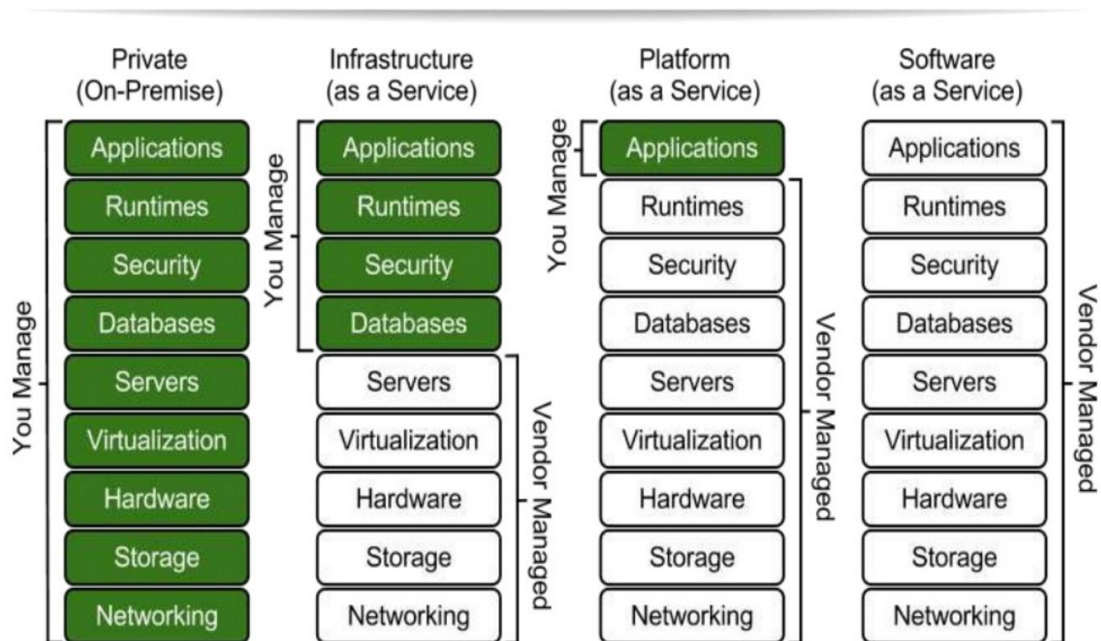
Kuva 6. Saksassa sijaitseva Amazon Web Servicesin datakeskus [16].

3.1.1 Software-, Platform- ja Infrastructure as a Service

Pilvipalveluita on kolmea perustyyppiä: Software as a Service (SaaS), Platform as a Service (PaaS) ja Infrastructure as a Service (IaaS), jotka ovat kuvassa 7. SaaS eli ohjelma tai sovellus palveluna tarjoaa ostavalle asiakkaalle sekä palvelimen laitteiston että ohjelmat. Asiakkaan ei tarvitse hallinnoida ajettavaa pilviarkkitehtuuria kuten verkkoa, palvelimia, käyttöjärjestelmää, tallennustilaa tai edes yksittäisen ohjelman muutoksia. Tästä kaikesta huolehtii pilvipalvelun tarjoaja. Yksinkertainen esimerkki SaaS:ista on yrityksen sähköposti kuten Hotmail tai Gmail. Toisin sanoen Software as a Servicella tarkoitetaan mahdollisuutta käyttää palveluntarjoajan ohjelmia, joita ajetaan pilvessä ja joita käytetään internetin kautta. SaaS-palveluiden mahdollisia käyttäjiä ovat yksittäiset kuluttajat sekä yritykset. [10; 11.]

Platform as a Service eli alusta palveluna on pilvipalvelu, joka antaa tilaajalleen suuremman vapauden valita, mitä kehitysalustaa käyttää kuin SaaS. Platform as a Servicen

käyttäjällä tulee olla riittävät IT-taidot pystyäkseen hallinnoimaan käytettyä kehitysalustaa ja sen päälle asennettuja sovelluksia. PaaS antaa asiakkaalleen ylläpidetyn ympäristön kehittää ja testata ohjelmiaan. Siksi PaaS onkin yleisimmin käytettynä suurissa yrityksissä kehitys- ja testausprosesseissa sekä sovellusten julkistamisprosesseissa.



Kuva 7. Pilvipalveluiden eri palvelutyyppit ja niiden hallinnoimissuhteet. [35.]

Palveluntarjoaja tarjoaa asiakkaalle prosessointikapasiteetin sekä tallennukseen tarvittavan laitteiston, verkkoyhteydet ja muut olennaiset tietokoneressit, joiden avulla asiakas voi ajaa haluamiaan ohjelmistoja, kuten käyttöjärjestelmiä tai sovelluksia. Asiakas ei siis hallinnoi alla olevaa pilviarkkitehtuuria, mutta voi hallinnoida käyttöjärjestelmää, tallennustilaa, ajettuja ohjelmia ja joitain verkon komponentteja kuten palomureja ja virusturvaa. Kuten kuvasta 7 ilmenee, kaikki kolme perustyyppiä rakentuvat samaan muottiin. Eroavaisuutena on vain tarjoajan ja käyttäjän välinen hallinnoimissuhde. SaaS:issa asiakas käyttää tiettyä sovellusta, ilman huolta ohjelman toimivuudesta. PaaS:issa asiakas käyttää tarjottua alustaa huolehtien sovellusten käytöstä, palveluntar-

joajan huolehtiessa laitteistosta ja käyttöjärjestelmästä. IaaS:issa asiakas huolehtii käyttöjärjestelmästä, turvallisuudesta ja sovelluksista. Palveluntarjoaja tarjoaa ainoastaan laitteiston, verkkoyhteyden laitteistoon sekä laitteiston ylläpidon. [11.]

3.1.2 Pilvipalvelumallit

Aiemmin mainitut pilvipalveluiden palvelutyypit (SaaS, PaaS ja IaaS) pyrkivät vastaamaan asiakkaan tarpeisiin riippuen tasosta, jolla hän haluaa hallinnoida laitteistoa sekä ohjelmia. Kukin malli korreloi organisaation suuruuden kanssa. Pienet yritykset valitsevat yleensä SaaS:in, kun taas suuret kansainväliset yritykset haluavat itse pystyä hallinnoimaan tietokonettaan valiten IaaS-tyyppisen palvelun. Näihin palveluihin liittyy pilvipalvelumallit, jotka kuvaavat käytetyn pilvipalvelun tyyppiä. Pilvipalvelumallit voidaan luokitella neljään eri luokkaan, jotka ovat Public Cloud, Private Cloud, Hybrid Cloud ja Community Cloud.

Puhuttaessa Public Cloudista eli julkisesta pilvestä tarkoitetaan sillä pilvipalvelua, joka on suunnattu suurelle yleisölle. Sen voi omistaa, hallinnoida ja ylläpitää yritys, hallinnollinen- tai akateeminen järjestö [10]. Esimerkiksi Google Drive tai Dropbox ovat Public Cloudia. Public Cloudin etuna on sen helppokäyttöisyys. Jos asiakas tarvitsee laskentakapasiteettia, tarvitsee heidän vain kirjautua julkisen pilvipalveluntarjoajan nettisivuille ja alkaa käyttämään palvelua. Public Cloudia käyttävät yleensä pienet ja keskisuuret yritykset. [11.]

Private Cloud tai yksityinen pilvi viittaa pilven omistukseen. Private Cloudissa yritys omistaa pilven ja täten pystyy kontrolloimaan pilveä huomattavasti enemmän kuin Public Cloudia. Private Cloud on kuitenkin erittäin kallis ja sitä käyttävätkin vain suuret yritykset, joilla on varaa ylläpitää pilviarkkitehtuuria, suuria konesaleja ja osaavaa henkilökuntaa. Yksityisessä pilvessä tiedot ovat paremmassa tallessa kuin julkisessa, mutta sen ylläpito on lähes mahdotonta kaikille muille paitsi suurille yrityksille. [11.]

Hybrid Cloud eli Hybridipilvellä tarkoitetaan nimensä mukaisesti hybridiä, missä on yhdistelmä kahta eri pilvipalvelumallia kuten julkista sekä yksityistä. Esimerkki hybridipilvestä voi olla yrityksestä, joka ylläpitää yksityistä pilveä, mutta laskentakapasiteettiin loppuessa yksityinen pilvi tukeutuu julkisen pilven laskentakapasiteettiin saaden vaaditun kokonaiskapasiteetin. Konkreettinen esimerkki hybridipilvestä voisi olla koulun internet-

sivut, jotka toimivat koulun omassa pilvessä. Internetsivut ovat yleensä vähäisellä käytöllä, mutta uusien hakijoiden hyväksymispäivänä yksityisen pilven laskentatehot loppuvat, koska kaikki joulukuun hakeneet oppilaat haluavat tarkastaa, pääsivätkö he kouluun opiskelemaan. Ilman hybridipilveä sivusto kaatuisi laskentakapasiteetin loppuessa. Hybridipilvi turvautuu suuren kuorman tullessa julkiseen pilveen, mistä se saa vaadittavan laskentakapasiteetin nettisivujen ylläpitoon. [11.]

Community Cloud eli yhteisöpilven käsite syntyi, kun yritykset tietyillä työmarkkinan osilla kuten energia-, talous- tai terveydenhuollon sektoreilla huomasivat tarvitsevansa erikoistuneita ohjelmia, joita ei voitu hyödyntää muilla sektoreilla. Community Cloud tarjoaa Public Cloudin hyödyt, jotka on rajoitettu tietyille työmarkkinoiden sektorille. [11.]

3.1.3 Pilven olennaiset ominaisuudet

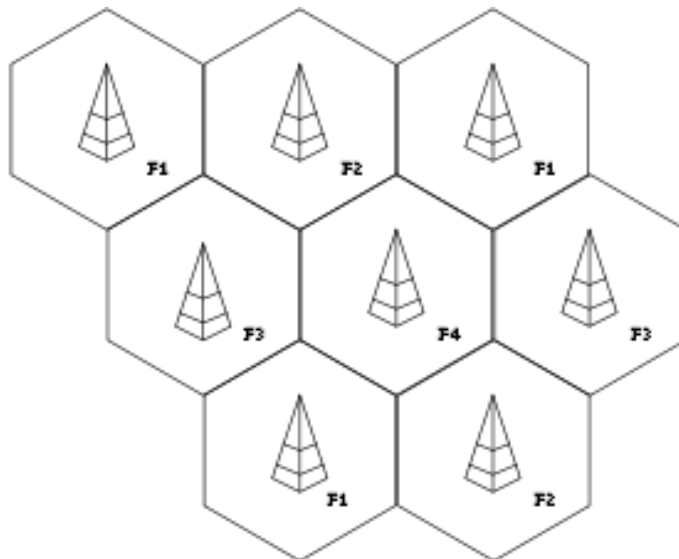
Pilvipalvelulle ominaisia piirteitä on yhteensä viisi, joita ovat:

- Skaalautuva ja automatisoitu itsepalvelu, jossa asiakas voi käyttää pilvipalveluita, kuten verkon tiedontallennuspalvelua, automaattisesti ilman kanssakäyntiä toisen ihmisen kanssa.
- Laiteriippumaton käyttö, jolla tarkoitetaan, että palvelut sijaitsevat verkossa ja niitä voi käyttää monin eri laittein. Tällaisia laitteita ovat esimerkiksi, älypuhelimet, tabletit, kannettavat tietokoneet ja pöytäkoneet.
- Resurssien jakaminen ja skaalautuvuus. Tällä tarkoitetaan, että pilvi jakaa laskentakapasiteettia asiakkaan tarpeen mukaan. Saatujen resurssien alkuperä ei ole yleensä asiakkaalla tiedossa. Esimerkiksi yhdellä käyttökerralla pilvipalvelu antaa laskentakapasiteettia Saksassa sijaitsevasta konesalista, kun taas toisella kerralla pilvi käyttää Irlannissa sijaitsevan konesalin tietokoneita. Resursseilla tarkoitetaan muun muassa tallennustilaa, muistia, prosessointikapasiteettia sekä internetyhteyden kaistan suuruutta.
- Suuri elastisuus. Tällä tarkoitetaan automaattista resurssien jakoa tai niiden vähentämistä riippuen asiakkaan sen hetkisistä tarpeista.

- Mitattu palvelu. Mitatulla palvelulla viitataan pilvijärjestelmän automaattiseen resurssien optimointiin. Eli jos asiakas ei enää tarvitse suurta laskentakapasiteettia, pienentää pilven järjestelmä asiakkaalle annetun laskentakapasiteetin tarvitulle tasolle, vähentäen asiakkaalle tulevia kustannuksia. Laskentakapasiteetti sisältää muun muassa muistin määrän ja käytettyjen prosessorien lukumäärän. [10.]

3.2 Matkapuhelinverkot

Matkapuhelinverkko tai soluverkko on kommunikaatioverkko, missä viimeinen kommunikaatioyhteys on langaton. Verkko saa nimensä solumaisesta rakenteesta, missä radiotornit kattavat verkotetun alueen muodostaen soluja (kuva 8). Radiotornit lähettävät eri taajuuksia (kuvassa F1-F4), joita jokaista voi käyttää erikseen verkkoon pääsemiseksi. Tämä mahdollistaa yhdellä alueella olevien henkilöiden saman taajuuden käytön heidän kummankaan datan sekoittumatta. Esimerkiksi kaksi eri asiakasta samalla mobiiliverkon peitealueella, jotka ovat eri soluissa voivat soittaa puhelun käyttäen taajuutta F1, ilman että puhelut häiriintyvät. Tässä on kuitenkin edellytyksenä, se ettei samaa taajuutta sisältävät solut saa olla vierekkäin. Muuten aiheutuisi liian suurta häiriötä, mikä estäisi samojen taajuuksien käytön. Äänen lisäksi matkapuhelinverkoissa voi siirtää myös dataa. [17.]

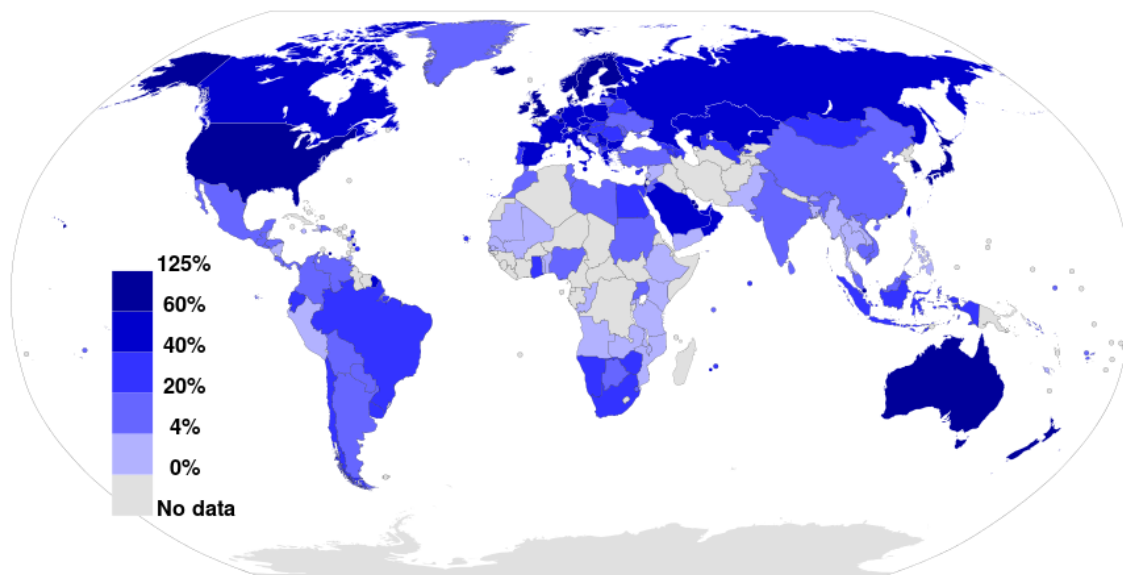


Kuva 8. Matkapuhelinverkon solumainen rakenne. [17].

Nykyään matkapuhelinverkoista puhuttaessa puhutaan 3G:stä ja 4G:stä. Näillä viitataan kolmannen tai neljännen sukupolven matkapuhelinverkkoteknologioihin. Nämä perustuvat International Telecommunication Unionin (ITU) asettamiin standardeihin. 3G-verkolle on määritetty International Mobile Telecommunications-2000 -standardi (IMT-2000), joka muun muassa määrittää vaaditut tiedonsiirtonopeudet ja saavutettavuudet, jotka 3G-verkon tulee saavuttaa. Saavuttaakseen IMT-2000-standardin, systeemin tulee pystyä tarjoamaan ainakin 200 kbit/s nopeudet ruuhka-aikaan. Kuitenkin teknologian kehittyttyä monet palveluntarjoajat pystyvät tarjoamaan jo huomattavasti nopeampia tiedonsiirtoyhteyksiä kuin vaaditun 200 kbit/s. [18.]

Uuden sukupolven matkapuhelinverkkostandardi on julkaistu noin kymmenen vuoden välein, alkaen vuodesta 1981, kun 1G-verkko julkaistiin. Jokaisella sukupolvella on oma taajuusalueensa, nopeammat tiedonsiirtoyhteydet eivätkä ne ole yhteensopivia aikaisempien sukupolvien kanssa. Esimerkiksi 3G:n taajuusalue on 1,8 – 2,5 GHz ja 4G:llä se on 2 – 8 GHz. Ensimmäiset 3G-verkot julkaistiin vuonna 1998 ja ensimmäiset neljännen sukupolven verkot esiteltiin kymmen vuotta myöhemmin vuonna 2008. [18.]

Matkapuhelinverkkojen peitealueet ovat kasvaneet ympäri maailman viimeisen vuosikymmenen aikana lisäten asiakkaiden mahdollisuuksia käyttää internetiä mobiililaitteillaan syrjäisemmälläkin alueilla. Tämä puolestaan parantaa muun muassa mHealthin tilannetta maailmalla ja tasa-arvoistaa terveydenhuollon palveluiden saatavuutta myös syrjäisillä alueilla asuville ihmisille. Vuonna 2011 90 % maailman asukkaista asui alueella, jossa oli 2G-verkko, kun taas 45 % maailman väestöstä asui alueilla, jossa oli sekä 2G että 3G saatavilla. Vain 5 % asui 4G-peitealueen piirissä. 2017 mennessä on arvioitu, että yli 90 % maailman väkiluvusta on 2G-verkon piirissä, 85 % 3G:n peitealueella ja 50 % 4G:n peitealueella. Kuvassa 9 näkyy vuonna 2012 tilattujen mobiililaajakaistojen tilausmäärät suhteutettuna maan väkilukuun. Kuvasta voidaan huomata, että köyhemmät maat, joissa terveydenhuolto on heikompaa kuin länsimaissa on myös mobiiliverkkojen käyttö väestöä kohden paljon pienempää. Köyhät maat hyötyisivätkin suuresti, jos matkapuhelinverkkojen peitealueet saataisiin koko maailman kattavaksi tuoden myös mHealth-palveluita jokaisen saatavaksi. [20.]

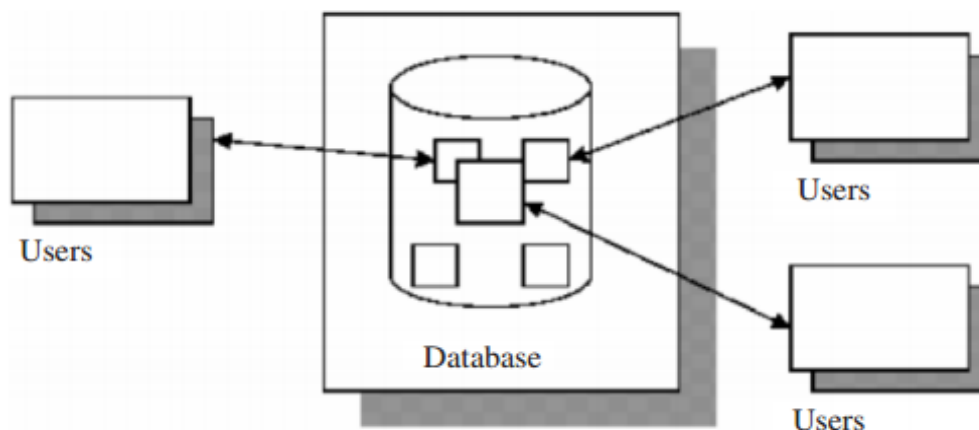


Kuva 9. Mobiililaajakaistojen määrä suhteutettuna maan väkilukuun vuonna 2012. [20.]

3.3 Tietokanta

PoC:ssa käytettiin MySQL-tietokantaa ja ennen esimerkin läpikäyntiä on hyvä kertoa, mitä tarkoitetaan tietokannalla, milloin tietokannat syntyivät, mitkä ovat tietokannan tärkeimmät roolit sekä mitä tunnettuja tietokannanhallintajärjestelmiä markkinoilla ylipääntänsä on.

Tietokanta on hyvin organisoitu kokoelma dataa, jotka liittyvät toisiinsa jollain merkittävällä tavalla. Tietokannassa oleva data on representaatio faktoista, konsepteista tai ohjeista, jotka ovat sopivia kommunikointiin, tulkintaan tai prosessointiin joko ihmisen tai automaation avulla. Tietokantojen hallintajärjestelmissä on kahdenlaista dataa: organisaatioille olennaista informaatiota sisältävää dataa tai metadataa, joka kuvaa tietokannassa olevaa dataa. Esimerkiksi lista erimerkkisistä autoista olisi dataa ja dataa kuvaava metadata voisi olla autonmerkki. Kuvassa 10 on yksinkertaistettu kuva tietokannasta. [22.]



Kuva 10. Yksinkertaistettu näkymä tietokannasta. [22.]

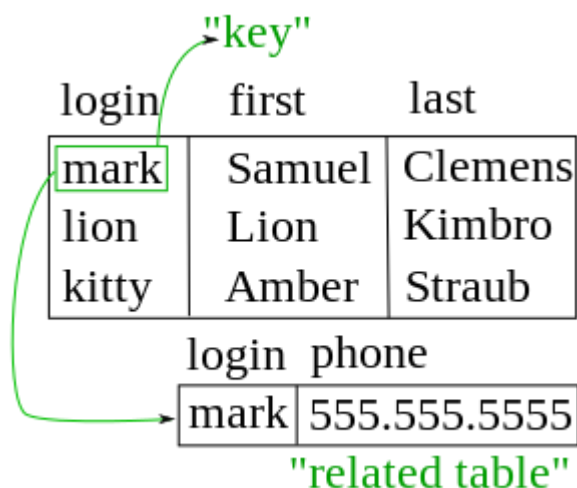
Tietokantojen hallintajärjestelmällä, database management systemillä (DBMS), tarkoitetaan useista tietokonesovelluksista koostuvaa ohjelmaa, joka mahdollistaa käyttäjälle yhden tai useamman tietokannan hallinnan sekä pääsyn itse dataan. DBMS tarjoaa monia vaihtoehtoja, joilla informaatioon pääsee käsittelemään, tallentamaan ja palauttamaan tietokannasta. Tietokanta ja tietokannanhallintajärjestelmä ovat kaksi eri asiaa, mutta monesti arkikielessä tietokannalla viitataan myös tietokannanhallintajärjestelmään DBMS:ään.

Tietokannanhallintajärjestelmät antavat käyttäjälle monia tietokannan hallintaan liittyviä työkaluja. Tiedon hallinnoinnin tyypit voidaan jakaa karkeasti neljään eri luokkaan.

- Datan organisointi – Tarkoitetaan sitä kuinka data on organisoitu ja kuinka sitä voidaan muokata, poistaa ja luoda uusia organisoimismääritelmiä.
- Päivitys – Dataa voi lisätä, muokata ja poistaa.
- Haku – Dataa voi noutaa tietokannasta, jota muut ohjelmat pystyvät hyödyntämään. Noudettu data voi olla samassa formaatissa kuin tietokannasta haettaessa tai muokattuna uuteen formaattiin riippuen datan käyttötarkoituksesta.
- Hallinta – Käyttäjien rekisteröinti ja monitorointi, tiedon turvallisuuden valvominen, tiedon eheyden säilyttäminen sekä tiedon palauttaminen, mikäli järjestelmässä on tapahtunut häiriö.

Tietokantojen historia alkaa 1960-luvun alkupuolelta. Tietokantateknologioiden kehitys voidaan jakaa kolmeen osaan: navigaatio-, relaatio- sekä postrelaatiotietokantaan. En-

simmäisenä markkinoille tulivat navigaatiotietokannat, jossa tieto oli suurina kokonaisuuksina jotka oli linkitetty toisiinsa. Kyseisessä mallissa oli suuria ongelmia tiedon hakemisessa, jos tiedot olivat kompleksisia sekä tämän mallin tiedon tallennus vei turhaa tilaa, sillä myös tyhjät tiedot täytyi sisällyttää kokonaisuuksiin. Tähän tietokantamalliin ei ollut tyytyväinen IBM:n työntekijä Edgarr Codd, joka kehitti relaatiotietokannan. Ensimmäiset relaatiotietokannat julkistettiin 1970-luvun alussa. Relaatiotietokannat ovat vieläkin hallitsevia tietokantavaihtoehtoja markkinoilla. Mullistavaa relaatiotietokannoissa oli tiedon tallentaminen normalisoituihin tauluihin. Postrelaatiotietokannoilla tarkoitetaan yleensä objektorientoitunutta tietokannanhallintajärjestelmää, missä tietokantaan tallennetaan datan sijasta (kuten kokonaisluku tai desimaaliluku) kompleksisia objekteja, joita voidaan monistaa ja hakea helposti [23]. Esimerkiksi objektina voi olla kokonainen videoklippii tai kuva, ei siis pelkästään pilkottua dataa. [21.]



Kuva 11. Esimerkki relaatiotietokannasta. [21.]

Relaatiomallissa olennaista taulujen lisäksi on se, että taulut voidaan linkittää toisiinsa avaimilla. Kun käyttäjästä halutaan saada tietoa, löytyvät myös henkilöä koskevat muut tiedot avaimen perusteella. Eli kuvassa 11 tietokantaan tehdään kysely, jossa halutaan saada tiedot markista, etsitään avaimella mark ja vastauksena tulevat kaikki tiedot, jotka ovat yhdistyneenä mark-nimiseen avaimeen.

Tietokantoihin voidaan esimerkiksi tallentaa informaatiota käyttäjästä: kuten heidän nimensä, kirjautumistunnus, puhelinnumero sekä osoite. Navigaatiomallissa kaikki nämä tiedot olisivat sisällytetty yhteen isoon kokonaisuuteen. Relaatiomallissa tiedot on tallennettu kuvan 10 mukaisesti tauluihin.

Fyysisesti tietokantapalvelimet ovat tietokoneita, joissa sijaitsee tietokannat sekä tietokannanhallintajärjestelmä sekä siihen kuuluvat oheissovellukset. Yleensä kyseiset palvelimet ajavat vain edellä mainittuja ohjelmia, eivät mitään muita. Eli ne ovat käytössä pelkästään tietokantana. Näissä tietokantapalvelimissa on yleensä moniydinprosessorit, paljon muistia sekä RAID-kiintolevytekniikkaa. RAID-tekniikkaa (Redundant Array of Independent Disks) käytetään parantamaan tietokoneen vikasietoisuutta ja nopeutta. Siinä tallennetaan sama tieto kahteen eri kiintolevyyn, ja jos toinen levyistä rikkoutuu, voidaan tieto yhä palauttaa tiedon kahdentamisen ansiosta. Tietokannanhallintajärjestelmät voidaan nykyään asentaa tavallisen käyttöjärjestelmän päälle. Ennen DBMS:ät vaativat kustomoidun käyttöjärjestelmän, mutta nykyään se onnistuu standardoidulla käyttöjärjestelmällä. Suurimpia kaupallisia relaatiotietokannanhallintajärjestelmien myyjiä ovat IBM, Oracle sekä Microsoft. Yritysten DBMS:ät ovat IBM:än DB2, Oraclen MySQL ja Microsoftin SQL Server.

3.4 Python

Python on monipuolinen, objektioitoitunut, korkean tason ohjelmointikieli. Pythonissa on moneen muuhun ohjelmointikieleen verrattuna yksinkertainen syntaksi. Python-enginellä tarkoitetaan laitetta esimerkiksi tietokonetta, johon on asennettu Python-ohjelma, ja se pystyy suorittamaan Pythonilla kirjoitettua koodia. Muita Pythonia vastaavia korkean tason tietorakenteiden ohjelmointikieliä on muun muassa Perl, Java sekä PHP. Pythonia voi käyttää komentoriviltä ajettavissa ohjelmissa ja sillä voi suorittaa haastavaakin tieteellistä laskentaa. [24.]

Pythonin perustavanlaatuisena ideana on suurien olemassa olevien kirjastojen hyödyntäminen. Perusteiden oppimisen jälkeen Python-ohjelmoija voi kirjoittaa yksinkertaista, mutta tehokasta koodia kirjastoja hyödyntäen. Kirjastolla tarkoitetaan valmiiksi tehtyä avoimen lähdekoodin tiedostoa, jonka voi ladata omaan Python-ohjelmaansa. Latauksen jälkeen tarvitsee ladattua kirjastoa vain kutsua sen nimellä, mikä antaa kirjaston ominaisuudet ohjelmoijan käyttöön. Tämän takia Python-ohjelmoijien sanotaan olevan erittäin tuotteliaita verrattuna esimerkiksi C-ohjelmointikielen kirjoittajiin, missä vastaavanlaisia kirjastoja ei ole käytössä ja kaikki tarvitsee koodata itse.

Muita Pythonin ominaisuuksia ovat muun muassa sen monialustainen ajettavuus. Se toimii muun muassa Windows-, Linux/Unix- sekä Mac OS X -käyttöjärjestelmällä. Python

on myös helposti integroitava ja sillä on rajapintoja tietokantojen kanssa. Python toimii esimerkiksi MySQL:n ja SQLite:n -DBMS:ien kanssa. Pythonia käytetään laajasti verkkosivujen tekoon ja tietomassojen parsimiseen. [24; 25.]

Python-kielen loi 1990-luvun alussa hollantilainen Guido Van Rossum työskennellessään Sticthingin matemaattisella laitoksella. Alun perin Pythonin piti olla jatkoa ABC-ohjelmointikielelle, missä korjattiin ABC:n puutteita. Nimensä Python saa brittiläisen 1960 - 1970 -luvulla vaikuttaneen brittiläisen komediaryhmä Monty Pythonin mukaan. [24; 25.]

Alkuperäisestä Python-kielestä julkaistiin vuonna 2000 jalostettu Python 2.0 -versio, joka on yhä tuettu versio. Vuonna 2008 julkaistiin Python 3.0, joka ei ollut enää yhteensopiva aikaisempien Python-versioiden kanssa. Siirtymisen helpottamiseksi versioista 2.x versioon 3.x onkin tehty käännösohjelma, joka kääntää automaattisesti osan koodista uudempaan versioon. [24.]

3.5 XML-tiedonsiirtomuoto

XML (Extensible Markup Language) on tietynlaisten merkintäkielten yläkäsite tai standardi, jolla tiedon merkitys on kuvattuna tiedon sekaan. XML on sekä ihmiselle että koneelle ymmärrettävää, mikä tekee kielestä erittäin hyödyllisen. Toisin sanoen XML kuvaa luokan dataobjekteja, joita nimitetään XML-dokumentiksi. XML on alun perin eristetty SGML-merkintäkielestä (Standard Generalized Markup Language) [ISO 8879]. [26; 28.]

XML on suunniteltu nettisovelluksille tiedonvaihtoon. Eri ohjelmat pystyvät vaihtamaan tietoa yhteisen kielen avulla. XML-kieli on helppo oppia, käyttää ja hyödyntää. XML suunniteltiin kuvaamaan dataa ja sen onkin määritellyt WWW-konsortio (World Wide Web), ”metodina jossa laitetaan järjestetty data tekstitiedostoon.” Kuvassa 12 on esimerkki XML-datan rakenteesta.

```

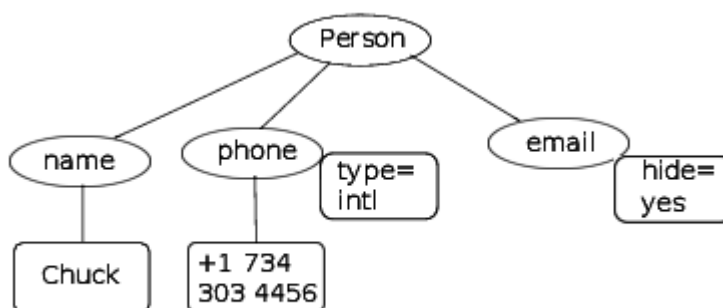
<person>
  <name>Chuck</name>
  <phone type="intl">
    +1 734 303 4456
  </phone>
  <email hide="yes"/>
</person>

```

Kuva 12. Esimerkki XML-dokumentin rakenteesta. [27.]

XML-dokumentin ulkoasussa elementtien ja attribuuttien lisäksi on myös prologi sekä dokumentin tyyppin määrittely. Prologi sisältää käytetyn XML:n version ja mahdollisen dokumentin koodauksen ja tiedon, voidaanko prologin jälkeen tuleva dokumentin tyyppi jättää lukematta. Kuvassa 12 on elementtejä ja attribuutteja. Elementit antavat XML-dokumentille merkityksen ja elementtejä tarkentavat attribuutit. Elementtejä voi olla sisäkkäin kuinka monta tahansa, mutta kaikkien elementtien tarvitsee sijaita yhden juurielementin alla. Kuvan 12 esimerkissä juurielementtinä on person. Muut elementit ovat name, phone ja email. Attribuutteja eli elementtejä tarkentavia asioita on kuvassa 12 esimerkiksi Chuck tai emailia tarkentava vaihtoehto hide = yes, mikä määrittää, ettei emailia haluta näkyviin.

Monesti on hyödyllistä ajatella XML-dokumentti puurakenteena tai tiedostorakenteena, missä sisennykset tekstissä kuvaavat, mikä on tiedoston ylätunniste ja mitkä ovat ylätunnisteeseen liittyviä alatunnisteita. Kuvassa 13 on esitetty XML-dokumentin puurakenne. [27.]



Kuva 13. XML-dokumentin puu- tai tiedostorakenne. [27.]

Vuonna 1996 XML-merkintäkielen kehittäneellä työryhmällä, jonka projektin johtajana oli Sun Microsystemsin Jon Bosak, oli tavoitteena kehittää kieli, joka:

1. Olisi mutkaton käyttää internetissä
2. tukisi laajaa sovelluskantaa
3. olisi yhteensopiva SGML:än kanssa
4. tekisi XML:ää käyttävien tietokoneohjelmien luonnin helpoksi
5. ei antaisi XML:lle valinnaisia ominaisuuksia
6. olisi ihmiselle luettavaa ja ymmärrettävää
7. mahdollistaisi XML:än rakenteen nopean suunnittelun
8. olisi rakenteeltaan muodollinen ja jämäkkä
9. mahdollistaisi helpon dokumentin luonnin
10. ei kuitenkaan liikaa keskittyisi kielen ytimekkyyteen. [22.]

Tavoitteiden saavuttamisen ansioista XML:stä onkin tullut erittäin laajasti käytetty internetin tiedonsiirtoformaatti, joka yhä säilyy käytössä muiden, kuten JSON-tiedonvälitysformaatin (JavaScript Object Notation) kanssa.

4 Opinnäytetyön tarkoitus ja tutkimusongelmat

Opinnäytetyössä perehdytään mHealthiin, pilvipalveluihin, erilaisiin pilvipalvelumalleihin ja teknologiaan, jota käytetään terveystiedon saattamisessa mittaavasta anturista pilveen asti. Tämän lisäksi suoritetaan proof of concept terveystiedon välityksestä pilveen sekä pohditaan, miksi PoC suoritettiin ja mitä sillä saavutettiin.

Henkilökohtainen osallistumiseni projektiin oli heinäkuusta 2015 maaliskuuhun 2016. Heinäkuussa suoritin projektijohtajan suosittamia verkkokursseja Courserassa. Käydyt

kurssit perehdyttivät Python-ohjelmointikieleen, ja erityisesti datan muokkaukseen Pythonin avulla.

Syyskuusta alkaen IoT-projektiryhmä aloitti toimintansa. Projektissa jokaiselle opinnäytetyöntekijälle annettiin oma lähestymiskulma projektissa käytetyn Muratan anturin datan tutustumiseen sekä anturilta tulevan datan parsimiseen, eli turhan datan poistamiseen. Lähtöasetelmat projektissa olivat haastavat. Projektiryhmällä oli käytössä muutama syketietoja mittaava anturi sekä kasa tiedostoja, joissa oli alustavanlaatuinen ohjelma sykedatan näyttämiseksi. Ohjeita sovelluksen käyttöön tai tiedostojen sisältöön oli niukanlaisesti.

Tehtäväni projektissa ja opinnäytetyöni tarkoituksena oli tutustua Muratan BCG-anturin (Ballistocardiography) toimintaan sekä kirjoittaa Python-ohjelma, joka parsisi eli poistaisi turhan informaation ja lähettäisi olennaiset anturilta saatavat tiedot eteenpäin haluttuun tietokantaan. BCG-anturi on kiihtyvyyksianturi, joka mittaa erittäin tarkasti pieniäkin kiihtyvyyksiä. Näitä kiihtyvyyksiä mitataan henkilön sydämen alueelta aortan ja muiden suurien valtimoverisuonien veren virtauksesta [29]. Olin osallisena myös puheluneuvotte luissa NurseBuddyn kanssa, jossa valmistelimme mahdollista anturin käyttöönottoa ensin Metropolian laboratorio-olosuhteissa ja sen jälkeen NurseBuddyn asiakkaille.

NurseBuddy on yksi projektiin osallistuvista yrityksistä ja sen suurimpana tavoitteena projektissa on BCG-anturin testaaminen ja käyttöönotto oikeassa asiakasympäristössä. NurseBuddy on hoiva-alan startup-yritys, joka tuo asiakkaalle hoiva-alan hallintajärjestelmän helpottaen vanhuksen, vanhuksen hoitajan, omaisten sekä hoitajan esimiehen välistä kommunikaatiota. NurseBuddyn hallintajärjestelmää voi käyttää älypuhelimella, tabletilla tai tietokoneella henkilön tarpeiden mukaisesti. Yhtenä myöhemmän ajankohdan tavoitteena IoT-projektissa on saada BCG-anturi integroitua NurseBuddyn hallintajärjestelmään, mikä antaisi hoitajalle ja omaiselle enemmän tietoa vanhuksen senhetkisestä terveydentilasta.

5 Työn toteutus, Proof of concept

Opinnäytetyön keskeisenä ideana oli suorittaa proof of concept, joka toteutettiin Python-ohjelmointikielellä ja MySQL-tietokannalla. Tärkeimpänä syynä toteutettuun PoC:iin oli sen tuoma hyöty kokonaisprojektille. Suoritettu PoC auttaisi toteuttamaan projektissa haluttua kokonaisarkkitehtuuria, josta puhutaan tarkemmin myöhemmin tässä luvussa. Muita syitä, miksi päädyttiin suoritettuun PoC:iin, on muun muassa se, että IoT-projekti toteutetaan Googlen App enginellä, jossa eräs mahdollisista ohjelmointirajapinnoista on Python. Toinen syy, miksi valittiin Python-ohjelmointikieli PoC:in toteuttamiseen, on Pythonin erinomainen soveltuvuus tiedon etsintään ja parsintaan. Valittuna tietokantana oli Oraclen MySQL, koska se on yksi suurin markkinoilla oleva tietokantojenhallintajärjestelmien valmistaja ja sitä on ilmaista käyttää.

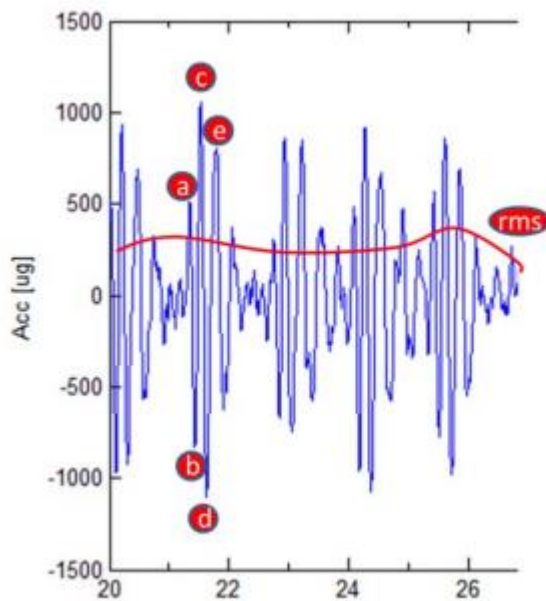
Opinnäytetyön keskeisenä sisältönä on tiedon parsinta, ja siksi on olennaista avata syitä, miksi tietoa parsitaan sekä mitä tiedon parsimisella voidaan saavuttaa. Projektissa mitaava anturi kerää tietoa ja lähettää sitä eteenpäin langattomaan välityspalvelimeen ja siitä eteenpäin lopulta pilveen asti. Tärkein syy tiedon parsintaan on saada tieto pienempään kokoon. Tiedon parsiminen on olennaista, erityisesti suurilla anturien lukumäärillä, sillä siirrettävän tiedon määrä kasvaa nopeasti suureksi. Jotta voidaan taata toimivuus myös suurilla anturien lukumäärillä, halutaan verkossa siirretty tietomäärä mahdollisimman pieneksi. Tämä edellyttää anturista lähteneen tiedon parsintaa, joka poistaa turhan datan jättäen vain olennaisen informaation jäljelle. Tiedon parsiminen vähentää myös tarvittavan tallennuskapasiteetin suuruutta.

5.1 Murata Electronics ja Muratan BCG-anturi

Projektissa mukana oleva yritys Murata Electronics on vuonna 1991 perustettu suomalainen yritys, entiseltä nimeltään VTI Technologies OY. Murata Electronics on erikoistunut matalan kiihtyvyyden, kallistuksen, kulmanopeuden ja paineen tarkkaan mittaamiseen. Tehdyt mittaukset perustuvat piikiekolle valmistettuihin MEMS-antureihin (Micro Electrical Mechanical Sensor). Henkilökuntaa yrityksellä on noin 700. Murata Electronics on osa japanilaista Murata-konsernia. [29.]

Murata Electronics on tunnettu autojen sähköisissä järjestelmissä olevista ajonvakautus (ESP) ja lukkiutumattomien jarrujen (ABS) järjestelmien valmistuksesta. Murata Electronics haluaa laajentaa tarkkojen anturiensa käyttöä myös terveyssektorille, johon se on kehittänyt BCG-anturin. Kyseinen anturi on käytössä projektissamme, ja käytän siitä nimitystä BCG-anturi. [29.]

Ballistokardiograafinen signaali on veren virtauksen aiheuttama rekyyli, kun veri virtaa aortassa ja sieltä eteenpäin valtimoihin. Tämä mitattu signaali saadaan mittaamalla veren liikkeen kiihtyvyyksiä kehon pituusakselin suuntaisesti. Tyypillinen BCG-signaali näkyy kuvassa 14, missä kuvaaja on kiihtyvyyden ajan funktio. Kohdassa a->b aortan läppä on auennut, ja veren virtaus kiihtyy kohti henkilön päätä, aiheuttaen vastakkaisuuntaisen rekyylikihtyvyyden. Pian veri tulee kohtaan, jossa aortta kääntyy alaspäin, jolloin rekyylikihtyvyyden suunta on toisen suuntainen b->c. Veren hajaantuessa eri valtimoihin, kiihtyvyyden pienenee ja vaihtaa merkkiä suunnilleen samaan aikaan aortan läpän sulkeutuessa, c->d. Anturin sijainnista ja vaimennuksesta riippuu paljon, kuinka suuri on saadun signaalin amplitudi, d->e. [34.]



Kuva 14. Tyypillinen BCG-signaali. [34.]

BCG-anturi raportoi kerran sekunnissa ajan, sykkeen (HR, heart rate), hengitystaajuuden (RR, respiration rate), sydämen suhteellisen lyöntivoimakkuuden (SV, relative stroke volume), sykevälivaihtelun (heart rate variability, HRV), signaalin vahvuuden (signal strength), anturin tilan, joka määräytyy signaalin vahvuuden perusteella sekä sydämen

lyönnistä lyöntiin ajat B2B, B2B' ja B2B". B2B-parametreja on kolme, sillä on arvioitu, että potilaan syke voi lyödä jopa kolmesti sekunnissa. [34.]

Markkinoilla on myös muita BCG-anturia hyödyntäviä yrityksiä, kuten Beddit sekä Emfit. Bedditin BCG-anturi asennetaan sänkyyn patjan alle ja sieltä se mittaa sydämen sykkimisestä aiheutuvia kiihtyvyyksiä. Tästä signaalista Bedditin anturi pystyy laskemaan muun muassa unenlaadun, sykkeen, hengitystaajuuden, kuorsauksen, kokonaisunen pituuden sekä sängystä poistumisten lukumäärän. Kaikista näistä tiedoista Bedditin mobiilisovellus laskee niin sanotun "sleep scoren", joka kuvaa kokonaisuudessaan nukutun unen laatua (kuva 15). [36.]



Kuva 15. Bedditin mobiilisovelluksen sleep score (vasen kuva) ja tarkempaa tietoa unesta (oikea kuva). [36.]



Kuva 16. Projektissa käytetty Muratan BCG-anturi.

5.2 Terveystieto pilveen

Projektissa käytettiin Muratan BCG-anturia, joka on kuvassa 16. Se on asennettu käyttäjän sängyn alle siten, että anturi on yhdensuuntainen käyttäjän lepoasennon kanssa. Kuvassa 17 on anturi asennettuna sänkyyn. Havainnointisyistä anturi on jätetty sängyn päälle, vaikka se asennetaankin sängyn alle metallirunkoon. Kun asennus on suoritettu ja anturi konfiguroitu sänkyyn sopivaksi, rekisteröi anturi potilaan sydämen sykkimissignaalin, jonka jälkeen anturissa oleva ohjelma laskee fysiologiset parametrit kuten sykkeen, sykevälivaihtelun, sydämen suhteellisen lyöntivoimakkuuden ja hengitystaajuu-

den. Ohjelman laskemat arvot anturi lähettää WLAN:n avulla anturin kanssa konfiguroituun puhelimeen tai tietokoneeseen. Tässä esimerkissä käytettiin kannettavaa tietokonetta. WLAN eli Wireless Local Area Network on langaton lähiverkkotekniikka, jolla erilaiset verkkolaitteet voidaan yhdistää ilman kaapeleita [30].

On hyvä huomata, ettei itse anturi ole yhdistettynä internetiin vaan se on vain yhdistetty langattomalla lähiverkkotekniikalla toiseen laitteeseen langattomaan välityspalvelimeen, tässä tapauksessa kannettavaan tietokoneeseen. Langaton välityspalvelin toimii porttina internetiin. Sillä on WAN-yhteys internetiin. WAN eli laajaverkko tulee englannin kielen sanoista Wide Area Network [31]. Laajaverkolla tarkoitetaan tiedonsiirtoverkkoa, joka peittää laajoja maantieteellisiä alueita.



Kuva 17. Anturi asennetaan yhdensuuntaisesti henkilön makuuasennon kanssa.

5.3 Projektin kokonaisarkkitehtuuri

Projektin kokonaiskuvan hahmottamiseksi tehtiin havainnollistava kuva 18, jossa näkyy vaihe vaiheelta, kuinka terveystieto saadaan eri vaiheiden jälkeen pilveen asti. Kuva on jaettu kuuteen eri kohtaan A – F, jotka käydään seuraavaksi läpi.

Kuvassa A on mittaava BCG-anturi, joka on asennettuna potilaan sänkyyn oikealla tavalla. Anturi kerää erittäin herkän kiihtyvyyssanturin avulla potilaasta saatavat sydämen

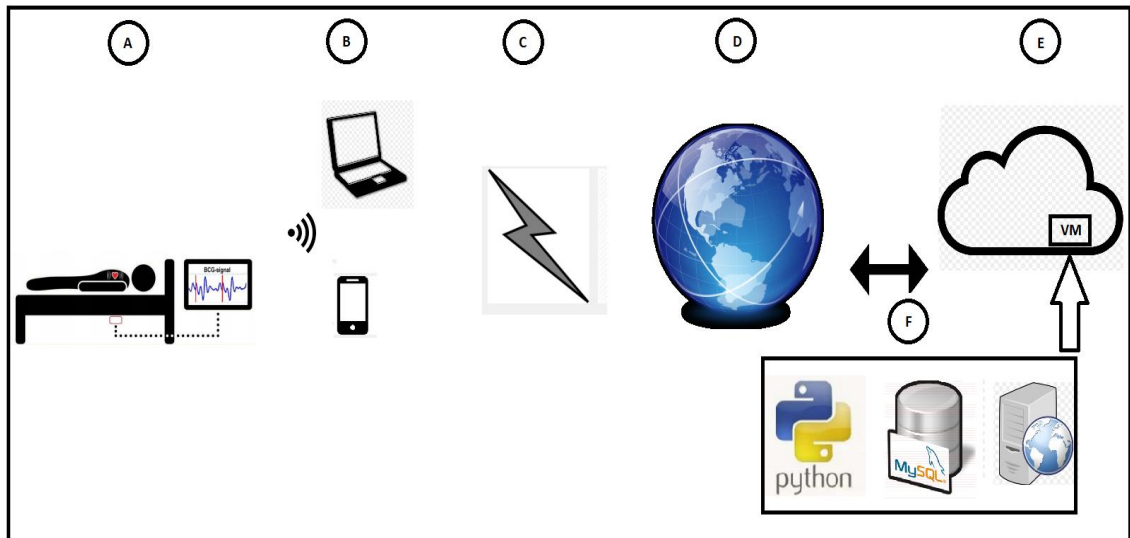
sykkimissignaalit, jotka anturiin asennettujen matemaattisten mallien avulla laskevat neljä eri parametria: sykkeen, hengitystaajuuden, sykevälivaihtelun sekä suhteellisen sydämen lyöntivoimakkuuden. Saadut arvot anturi lähettää WLAN:in (Wireless Local Area Network) avulla eteenpäin. Anturi ei siis ole kytkettynä internetiin, vaan ainoastaan lähiverkkoon vastaanottavan laitteen kanssa.

Kuvassa B on anturin lähettämän datan vastaanottava laite. Laite voi olla esimerkiksi kannettava tietokone tai älypuhelin. Yksi hyvä ja edullinen ratkaisu anturin lähettämän tiedon vastaanottajaksi on Raspberry Pi, joka on noin 40 euroa maksava kämmenen kokoinen tietokone. Vastaanottava laite saa anturin lähettämän datan XML-muodossa ja lähettää sen eteenpäin internetiin. Olennaista on, että kuvassa B olevan laitteen täytyy pystyä vastaanottamaan data ja lähettämään se internetyhteyttä pitkin eteenpäin. Se toimii myös anturin datan väliaikaisena varastona, sillä anturi pystyy säilyttämään saamansa tiedon ainoastaan 90 sekuntia, jonka jälkeen tiedot anturista katoavat ja uudet arvot tulevat tilalle.

Kuva C kuvaa WAN-yhteyttä. Laajaverkon (Wide Area Network) ääripäänä on internet, joka on kuvassa D.

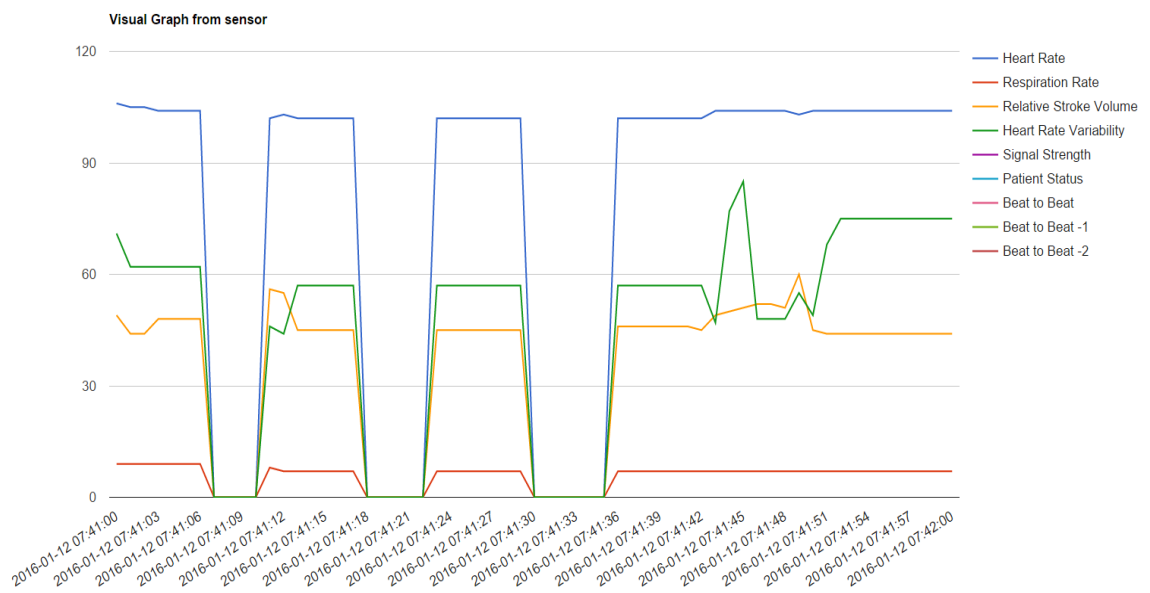
Kuvassa E on pilvipalvelu, johon on pääsy internetin kautta. Tieto kulkee sekä pilvestä internetiin että internetistä pilveen. Tämän takia internetin ja pilvipalvelun välillä on kaksisuuntainen nuoli. Pilvessä on virtuaalikoneita, jotka on merkitty kuvassa VM (engl. Virtual Machine). Virtuaalikone tarkoittaa nimensä mukaisesti virtuaalista tietokonetta, eli sillä on tietty määrä muistia, laskentatehoa, verkon kaistan leveyttä sekä joku käyttöjärjestelmä. Kaikki nämä sijaitsevat ohjelman sisällä, ja virtuaalikone ”luulee” olevansa fyysinen tietokone, vaikkei sitä olekaan.

Kuvassa F on näkymä virtuaalikoneen sisältä välttämättömistä ohjelmista. Vasemmalta oikealle tarvittavia ohjelmia on Python-tulkki, tietokanta sekä Web-palvelin. Jokaista näistä tarvitaan tiedon tallentamiseen pilveen. Web-palvelin kuuntelee palvelimelle saapuvaa tietoa. Data, joka palvelimelle lähetetään, on anturin lähettämää potilaan terveysdataa. Tätä dataa varten on kirjoitettu Python-ohjelmointikielellä koodia, joka käyttää Web-palvelimen vastaanottamaa terveysdataa, parsii ylimääräisen datan ja siirtää parsitun tiedon virtuaalikoneessa sijaitsevaan tietokantaan. Parsiminen on ohjelmoinnissa käytetty termi, joka tarkoittaa turhan tiedon karsimista, jättäen jäljelle vain halutun datakokonaisuuden.



Kuva 18. Kokonaiskuva terveystiedon kulkemisesta mittaavasta anturista pilveen.

Terveystiedon pilveen saannin jälkeen on projektiryhmämme tehtävä visualisoida pilvessä olevaa dataa. Visualisoinnista on saatu jo alustavia tuloksia, joita voidaan tarkastella nettiselaimen kautta. Visualisointi on toteutettu Google Chartsin avulla (kuva 19). Kuvassa x-akselilla on aika ja y-akselilla on signaalien suuruus. Esimerkiksi sininen signaali kuvastaa sydämen sykettä tietyillä ajanjaksoilla.



Kuva 19. Terveysdatan visualisointi Google Charts -ohjelmalla.

5.4 PoC

Opinnäytetyön tarkoituksena oli suorittaa proof of concept eli PoC joka voidaan käsittää monella eri tavalla asiayhteydestä riippuen. Yleisesti proof of concept on demonstraatio, jonka tarkoituksena on varmistaa, että tietyillä konsepteilla tai teorioilla on potentiaalia myös käytännön maailmassa. PoC on siis prototyyppi, joka on suunniteltu määrittämään demonstroidun tuotteen käytettävyyttä. Proof of conceptista voidaan käyttää myös ilmaisu proof of principle. [32.]

Tässä projektissa proof of conceptin tavoite oli löytää ratkaisu tekniseen ongelmaan. Opinnäytetyön tekninen ongelma oli BCG-anturin lähettämän datan parsiminen ja siirtäminen tietokantaan. Suoritettu proof of concept voidaan jakaa neljään vaiheeseen. Ne ovat määrittely, suunnittelu, toteutus ja testaus. Nämä neljä vaihetta kuuluvat miltei aina ohjelmistotuotantoprosessiin. Ohjelmistotuotanto tarkoittaa laajasti ymmärrettynä kaikki erilaiset tietokoneohjelmien valmistamisen menetelmät ja ohjelmistotuotantoprosessilla tarkoitetaan tietokoneohjelmien systemaattista tekemistä, eli niissä on ennalta määrätty vaiheet, joita noudatetaan. Tämä tehdään siksi, että päästään haluttuun lopputulokseen helpommin ja, kun on määritelty tarkasti, mitä halutaan tietokoneohjelmalla saavuttaa, on sen toteutus myös yksinkertaisempaa. [33.]

Työn määrittelyvaihe alkoi viime syys- ja lokakuussa. Kokoonnuimme projektiryhmämme kanssa viikoittain, jossa projektipäällikkö antoi jokaiselle oman lähestymistavan projektiin. Työkseni määriteltiin silloin Python-ohjelman tekeminen, joka parsisi BCG-anturilta tulevan datan ja lähettäisi parsitun datan tietokantaan.

Työn suunnitteluvaiheessa perehdyin MySQL-tietokantoihin, siinä käytettävään SQL-kieleen ja Windows-käyttöjärjestelmällä toimivaan MySQL-tietokantaan. Tämän lisäksi opiskelin Python-ohjelmointikielessä sitä, kuinka tuleva XML-formaatissa oleva data (liite 2) saadaan käsiteltyä Pythonilla ja kuinka se voidaan lähettää ylläpidettyyn tietokantaan. Suunnitteluvaiheessa jaoin ohjelman kuuteen eri osaan, jota sen tulisi pystyä tekemään.

Toteutusvaiheessa toteutin ohjelman, joka sisälsi suunnitellut vaiheet. Selostuksien alla Python-komennot kyseiseen vaiheeseen. Kirjoitettu Python-koodi löytyy liitteistä (liite 1). Suunnitellut vaiheet olivat:

1. Kirjaston importtaus. Tämä tarkoittaa sitä, että Pythonissa on ohjelman sisällä valmiita ohjelmia, joita voi kutsua tarvittaessa oman ohjelmansa apuun.

```
import mysql.connector
```

2. MySQL-yhteyden avaaminen. Eli Python-ohjelma ottaa yhteyden samassa tietokoneessa sijaitsevaan tietokantaan.

```
cnx = mysql.connector.connect(user='root', password='[REDACTED]',  
                              host='localhost',  
                              database='anturi')  
  
cnx.connect()
```

3. Tietojen keräys. Tässä kohdassa Python-ohjelma parsii BCG-anturilta saadun murata.xml tiedoston. Koodi tähän kohtaan löytyy liitteestä 1.
4. Tallentaa kerätyt tiedot MySQL-tietokantaan.

```
cnx.commit()
```

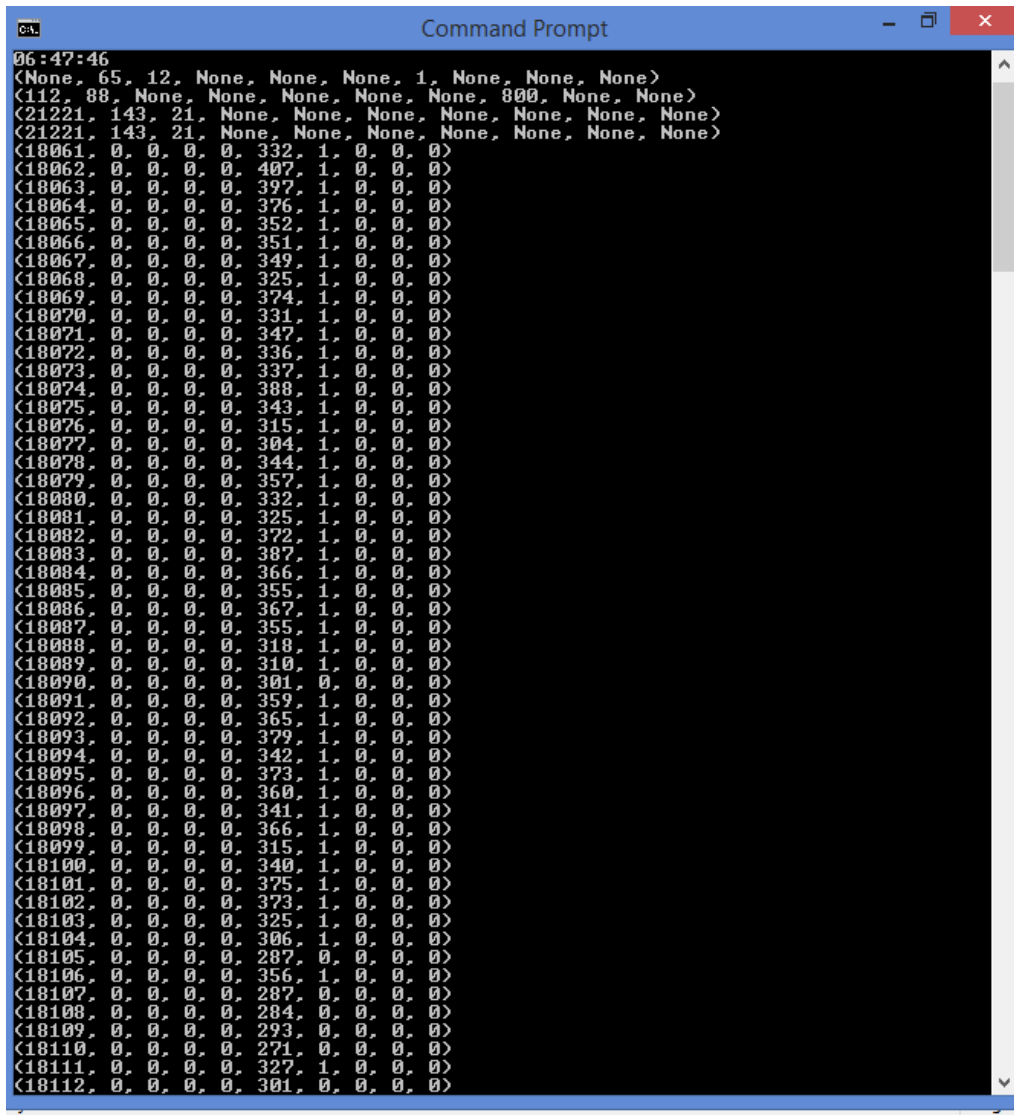
5. Valitsee sensori-nimisen taulun. Taululla tarkoitetaan tiettyä kokonaisuutta, jossa tiedot sijaitsevat. Yhden tietokannan sisällä voi olla monta eri taulua, jotka ovat linkittyneinä toisiinsa. Tauluja voi ajatella samalla tavalla kuin Excel-ohjelmassa auki olevia tauluja.

```
c.execute("SELECT * FROM sensor")
```

6. Tulostaa sensori-taulun sisällön.

```
for eachRow in rows:  
    print eachRow
```

Toteutusvaiheen jälkeen testasin, että ohjelma toimii halutulla tavalla ja suorittaa suunnitellut toiminnot. Ajettuani ohjelman Windowsin komentorivi tulosti BCG-anturin tallennetut arvot, jotka sijaitsivat MySQL-tietokannassa. (Kuva 20.)



```

06:47:46
(None, 65, 12, None, None, None, 1, None, None, None)
(112, 88, None, None, None, None, None, 800, None, None)
(21221, 143, 21, None, None, None, None, None, None, None)
(21221, 143, 21, None, None, None, None, None, None, None)
(18061, 0, 0, 0, 0, 332, 1, 0, 0, 0)
(18062, 0, 0, 0, 0, 407, 1, 0, 0, 0)
(18063, 0, 0, 0, 0, 397, 1, 0, 0, 0)
(18064, 0, 0, 0, 0, 376, 1, 0, 0, 0)
(18065, 0, 0, 0, 0, 352, 1, 0, 0, 0)
(18066, 0, 0, 0, 0, 351, 1, 0, 0, 0)
(18067, 0, 0, 0, 0, 349, 1, 0, 0, 0)
(18068, 0, 0, 0, 0, 325, 1, 0, 0, 0)
(18069, 0, 0, 0, 0, 374, 1, 0, 0, 0)
(18070, 0, 0, 0, 0, 331, 1, 0, 0, 0)
(18071, 0, 0, 0, 0, 347, 1, 0, 0, 0)
(18072, 0, 0, 0, 0, 336, 1, 0, 0, 0)
(18073, 0, 0, 0, 0, 337, 1, 0, 0, 0)
(18074, 0, 0, 0, 0, 388, 1, 0, 0, 0)
(18075, 0, 0, 0, 0, 343, 1, 0, 0, 0)
(18076, 0, 0, 0, 0, 315, 1, 0, 0, 0)
(18077, 0, 0, 0, 0, 304, 1, 0, 0, 0)
(18078, 0, 0, 0, 0, 344, 1, 0, 0, 0)
(18079, 0, 0, 0, 0, 357, 1, 0, 0, 0)
(18080, 0, 0, 0, 0, 332, 1, 0, 0, 0)
(18081, 0, 0, 0, 0, 325, 1, 0, 0, 0)
(18082, 0, 0, 0, 0, 372, 1, 0, 0, 0)
(18083, 0, 0, 0, 0, 387, 1, 0, 0, 0)
(18084, 0, 0, 0, 0, 366, 1, 0, 0, 0)
(18085, 0, 0, 0, 0, 355, 1, 0, 0, 0)
(18086, 0, 0, 0, 0, 367, 1, 0, 0, 0)
(18087, 0, 0, 0, 0, 355, 1, 0, 0, 0)
(18088, 0, 0, 0, 0, 318, 1, 0, 0, 0)
(18089, 0, 0, 0, 0, 310, 1, 0, 0, 0)
(18090, 0, 0, 0, 0, 301, 0, 0, 0, 0)
(18091, 0, 0, 0, 0, 359, 1, 0, 0, 0)
(18092, 0, 0, 0, 0, 365, 1, 0, 0, 0)
(18093, 0, 0, 0, 0, 379, 1, 0, 0, 0)
(18094, 0, 0, 0, 0, 342, 1, 0, 0, 0)
(18095, 0, 0, 0, 0, 373, 1, 0, 0, 0)
(18096, 0, 0, 0, 0, 360, 1, 0, 0, 0)
(18097, 0, 0, 0, 0, 341, 1, 0, 0, 0)
(18098, 0, 0, 0, 0, 366, 1, 0, 0, 0)
(18099, 0, 0, 0, 0, 315, 1, 0, 0, 0)
(18100, 0, 0, 0, 0, 340, 1, 0, 0, 0)
(18101, 0, 0, 0, 0, 375, 1, 0, 0, 0)
(18102, 0, 0, 0, 0, 373, 1, 0, 0, 0)
(18103, 0, 0, 0, 0, 325, 1, 0, 0, 0)
(18104, 0, 0, 0, 0, 306, 1, 0, 0, 0)
(18105, 0, 0, 0, 0, 287, 0, 0, 0, 0)
(18106, 0, 0, 0, 0, 356, 1, 0, 0, 0)
(18107, 0, 0, 0, 0, 287, 0, 0, 0, 0)
(18108, 0, 0, 0, 0, 284, 0, 0, 0, 0)
(18109, 0, 0, 0, 0, 293, 0, 0, 0, 0)
(18110, 0, 0, 0, 0, 271, 0, 0, 0, 0)
(18111, 0, 0, 0, 0, 327, 1, 0, 0, 0)
(18112, 0, 0, 0, 0, 301, 0, 0, 0, 0)

```

Kuva 20. Python-ohjelman tulostamat MYSQL-tietokannassa sijaitsevat BCG-anturin arvot.

Tallennetut BCG-anturin arvot sijaitsevat MySQL:ssä tietokannassa nimeltä anturi. Anturi tietokannassa on yksi taulu, johon BCG-anturilta saadut tiedot on tallennettu. Taulun nimi on sensor (kuva 21).

```

Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 26
Server version: 5.6.26-log MySQL Community Server (GPL)

Copyright (c) 2000, 2015, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| anturi |
| mysql |
| performance_schema |
| sakila |
| test |
| world |
+-----+
7 rows in set (0.02 sec)

mysql> use anturi;
Database changed
mysql> show tables;
+-----+
| Tables_in_anturi |
+-----+
| sensor |
+-----+
1 row in set (0.00 sec)

```

Kuva 21. MySQL:ssä olevat eri tietokannat sekä anturi-tietokannan sensor-taulu.

Jotta saatiin varmistus Python-ohjelman toimivuudesta, oli hyvä tarkastaa, mitä sensor-taulun sisälle oleviin sarakkeisiin saatiin arvoiksi. Kuvassa 22 sensor-taulussa olevat sarakkeet ja kuvassa 23 taulussa sijaitsevaa dataa.

```

mysql> describe sensor;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| Tick | int(11) | YES | | NULL | |
| HeartRate | int(11) | YES | | NULL | |
| RespirationRate | int(11) | YES | | NULL | |
| RelativeStrokeVolume | int(11) | YES | | NULL | |
| HeartRateVariability | int(11) | YES | | NULL | |
| MeasuredSignalStrength | int(11) | YES | | NULL | |
| Status | int(11) | YES | | NULL | |
| BeatToBeatTime | int(11) | YES | | NULL | |
| BeatToBeatTime1 | int(11) | YES | | NULL | |
| BeatToBeatTime2 | int(11) | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
10 rows in set (0.01 sec)

```

Kuva 22. Sensor-taulun sarakkeet ja sarakkeessa olevan datan tyypit.

```

Command Prompt - mysql.exe -p -u root

select * from sensor' at line 1
mysql> select * from sensor;
+-----+-----+-----+-----+-----+-----+-----+
| Tick | HeartRate | RespirationRate | RelativeStrokeVolume | HeartRateVariabil |
ity | MeasuredSignalStrength | Status | BeatToBeatTime | BeatToBeatTime1 | Beat |
oBeatTime2 |
+-----+-----+-----+-----+-----+-----+-----+
| NULL | 65 | NULL | 12 | NULL | NULL | NULL | N
ULL | NULL | 112 | NULL | NULL | 800 | NULL | N
ULL | 21221 | 143 | NULL | 21 | NULL | NULL | N
ULL | 21221 | 143 | NULL | 21 | NULL | NULL | N
ULL | NULL | 18061 | 0 | 332 | 0 | 0 | 0 |
0 | 0 | 18062 | 0 | 407 | 0 | 0 | 0 |
0 | 0 | 18063 | 0 | 397 | 0 | 0 | 0 |
0 | 0 | 18064 | 0 | 376 | 0 | 0 | 0 |
0 | 0 | 18065 | 0 | 352 | 0 | 0 | 0 |
0 | 0 | 18066 | 0 | 351 | 0 | 0 | 0 |
0 | 0 | 18067 | 0 | 349 | 0 | 0 | 0 |
0 | 0 | 18068 | 0 | 325 | 0 | 0 | 0 |
0 | 0 | 18069 | 0 | 374 | 0 | 0 | 0 |
0 | 0 | 18070 | 0 | 331 | 0 | 0 | 0 |
0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

```

Kuva 23. Anturi-tietokannassa olevan sensor-taulun arvoja.

Jos vertaa Python-ohjelman tulostamia arvoja kuvassa 20 ja MySQL:ssä sijaitsevia arvoja kuvassa 23, huomataan, että ne ovat samat. Tästä voidaan päätellä, että ohjelma toimi halutulla tavalla, tallentaen BCG-anturista saadun datan tietokantaan.

6 Yhteenveto ja johtopäätökset

Opinnäytetyössä tutustuttiin mHealthiin, pilvipalveluihin sekä terveystiedon kulkuun aina mittaavasta laitteesta pilveen asti. Työn käytännön osuudessa toteutettiin Proof of concept, missä mittaavan BCG-anturin keräämää dataa parsittiin Python-ohjelmointikielen avulla ja siirrettiin parsittu data MySQL-tietokantaan. Suoritettu proof of concept on yksi osa tiedon siirtämisessä pilveen ja auttaa projektiryhmäämme toteuttamaan IoT-projektin tavoitetta, joka on tiedon talletus pilveen ja pilvessä olevan terveystiedon visualisointi.

Opinnäytetyö oli osa Metropolian IoT-projektia. Osallistumiseni projektiin kesti syyskuun alusta 2015 maaliskuuhun 2016. Henkilökohtainen panokseni projektiin oli tuoda mHealthin tuomia mahdollisuuksia esiin teknologian ja liiketoiminnan näkökulmista valmiiksi kovin verkkopainotteisessa projektissa.

Opinnäytetyössä tuodaan esiin mHealthia teknologian- ja liiketoiminnan näkökulmasta ja perehdytään projektin teknisen toteutuksen lisäksi BCG-anturin fysiologisiin perusteisiin ja siihen, kuinka anturi pystyy mittaamaan potilaan sydämen sykkimissignaaleja ja tekemään signaalista tulkintoja. Tämän lisäksi opinnäytetyö tuo projektia esiin yleisemmällä tasolla, mikä helpottaa asiaan perehtymätöntä omaksumaan projektin kokonaiskuvaa sekä antamaan aiheeseen liittyvää taustatietoa. Uskon, että opinnäytetyötä voidaan hyödyntää mahdollisten asiakkaiden, rahoittajien ja projektiin liittyvien uusien jäsentien perehdytykseen ja yleiskuvan saamiseen.

Opinnäytetyön käytännön osuus suoritettiin ensin, mihin kuului projektiin ja annettuun materiaaliin sekä tehtävään perehtymistä. Perehtymisen jälkeen asensin projektin ajaksi saamaani kannettavaan tietokoneeseen MySQL:n, Pythonin sekä Muratan sovelluksen, joka näytti anturista saatavaa tietoa graafisessa muodossa. Tämän jälkeen kirjoitin Python-ohjelmaa, joka parsii terveystiedon ja siirsi MySQL-tietokantaan. Ohjelmointivaihe saatiin valmiiksi marraskuussa 2015.

Opinnäytetyö oli osa isompaa kokonaisuutta, jossa tavoitteena on tuoda teknologisia, verkko- ja pilvipalveluita hyödyntäviä ratkaisuja terveydenhuollon henkilökunnan käyttöön. Eli tarkoituksena IoT-projektissa on saada BCG-anturit asennettua potilaiden sän-

kyihin, joista terveydenhuollon henkilökunta voi seurata potilaiden terveydentilaa pilvipalvelun avulla. Saadut tiedot välitetään pilveen, ja pilvestä ne ovat käytettävissä terveydenhuollon henkilökunnalla jonkun verkossa olevan mobiililaitteen avulla.

Opinnäytetyössä saatuja tuloksia voidaan hyödyntää IoT-projektin edetessä. Kirjoitettu Python-ohjelma suorittaa yhden osan kokonaisarkkitehtuurista siirrettäessä terveystietoa pilveen. Tätä osuutta voidaan hyödyntää muodostettaessa ”end to end” -yhteyttä. Toisin sanoen, opinnäytetyössä tehty Python-ohjelma auttaa saavuttamaan IoT-projektin kokonaistavoitetta, missä terveydenhuollon henkilökunta voi lukea potilaasta saatavat terveystiedot pilven kautta. Tämän lisäksi kirjoitettua ohjelmaa voidaan käyttää runkona projektissa tulevien uudempien Python-ohjelmaversioiden tekemiseen.

Mahdollinen jatkokehittelyn aihe koko projektille voisi olla tietojen syvällisempi analyysi, missä yhdistettäisiin potilaan tilasta saatavia tietoja muihin lääketieteellisiin tietoihin. Niin sanotun strukturoidun ja strukturoimattoman datan yhdistäminen ja siitä analyysin tekeminen automaattisesti tietokoneen avulla. Esimerkiksi voitaisiin saada tarkempaa tietoa potilaan kokonaistilasta ja voinnista, jos analyysiin mukaan saataisiin tieto siitä, että potilas on esimerkiksi aloittanut uuden lääkityksen, joka huomataan sydämen lyöntitiheyden kasvuna. Tästä voitaisiin päätellä, ettei annettu uusi lääke ole henkilölle ehkä paras mahdollinen ja olisi syytä kokeilla jotakin muuta vaihtoehtolääkitystä juuri kyseisellä potilaalla.

Lähteet

1. Malvey, Slovensky .2014. mHealth –Tranforming Healthcare. Springer
2. Wikipedia. 2015. Suomen väestö. < https://fi.wikipedia.org/wiki/Suomen_väestö>. Päivitetty 29.11.2015. Luettu 13.1.2016.
3. Darrel. 2012. How mobile devices are transforming BCGthcare. Issues in technology innovation.
4. Diabetes. 2016. Verkkosivut. <<http://diabetesnet.com>>. Luettu 13.1.2016.
5. WHO. 2016. Verkkosivut. < <http://www.who.int/hiv/topics/treatment/en/>>. Luettu 13.1.2016.
6. Intel. 2013. The World Agrees: Technology Inspires Optimism for Healthcare Verkkosivut. http://newsroom.intel.com/community/intel_newsroom/blog/2013/12/09/the-world-agrees-technology-inspires-optimism-for-healthcare>. Luettu 15.1.2016.
7. Sarasohn-Kahn. 2010. Mobile health search is on the rise -but not yet at the tipping point. Verkkootikkeli. <<http://healthpopuli.com/2010/10/19/mobile-health-search-is-ontherise-but-not-yet-at-the-tipping-point/>>. Luettu 15.1.2016.
8. Fischer. 2013. Global study finds majority believe traditional hospitals will be obsolete in the near future. Verkkootikkeli. <<http://www.forbes.com/sites/theapothecary/2013/12/09/global-study-finds-majority-believe-traditional-hospitals-will-be-obsolete-in-the-near-future/>>. Luettu 15.1.2016.
9. Qualcomm. 2014. Wireless Heart Health: Using 3G to Assist Underserved Patients with Cardiovascular Disease. Verkkosivut. <www.qualcomm.com>. Luettu 15.1.2016.
10. Mell, Grance. 2011. The NIST Definition of Cloud Computing.
11. Srinivasan. 2014 Cloud Computing Basics. Springer
12. Business Dictionary. 2016. Computer System. Verkkosivut. < <http://www.business-dictionary.com/definition/computer-system.html>>. Luettu 19.1.2016.
13. Wikipedia. 2016. Virtual Private Network. Verkkosivut. < https://en.wikipedia.org/wiki/Virtual_private_network>. Luettu 19.1.2016
14. Wikipedia. 2016. Internet Service Provider. Verkkosivut. < https://en.wikipedia.org/wiki/Internet_service_provider>. Luettu 19.1.2016

15. Wikipedia. 2015. Asiakkuudenhallinta. Verkkosivut. <<https://fi.wikipedia.org/wiki/Asiakkuudenhallinta>>. Luettu 20.1.2016
16. The New York Times. 2014. Amazon Web Service to Open German Center. Verkkootikkeli. <http://bits.blogs.nytimes.com/2014/10/23/amazon-web-services-to-open-german-facility/?_r=0>. Luettu 20.1.2016
17. Wikipedia. 2015. Cellular Network. Verkkosivut. <https://en.wikipedia.org/wiki/Cellular_network>. Luettu 26.1.2016.
18. Wikipedia. 2016. 3G. Verkkosivut. <<https://en.wikipedia.org/wiki/3G>>. Luettu 26.1.2016.
19. Wikipedia. 2016. 4G. Verkkosivut. <<https://en.wikipedia.org/wiki/4G>>. Luettu 26.1.2016
20. Wikipedia. 2016. Mobile Broadband. Verkkosivut. <https://en.wikipedia.org/wiki/Mobile_broadband>. Luettu 26.1.2016.
21. Wikipedia. 2016. Database. Verkkosivut. <<https://en.wikipedia.org/wiki/Database>>. Luettu 27.1.2016.
22. Sumathi, Esakkirajan. 2007. Fundamentals of Relational Database Management System. Springer
23. Ghongade, Pursani. 2014. Comparison of Relational Database and Object Oriented Database. International Journal of Modern Trends in Engineering and Research. Verkkootikkeli. <http://www.academia.edu/14183866/Comparison_of_Relational_Database_and_Object_Oriented_Database>. Luettu 27.1.2016
24. Wikipedia. 2015. Python. Verkkosivut. <[https://fi.wikipedia.org/wiki/Python_\(ohjelmointikieli\)](https://fi.wikipedia.org/wiki/Python_(ohjelmointikieli))>. Luettu 27.1.2016
25. Quintagroup. 2016. Python. Verkkosivut. <<http://quintagroup.com/cms/python>>. Luettu 27.1.2016.
26. W3C. 2008. XML. Verkkosivut. <<https://www.w3.org/TR/REC-xml/#sec-intro>>. Luettu 29.1.2016.
27. Severance. 2015. Python for Informatics. University of Michigan.
28. Wikipedia. 2015. XML. Verkkosivut. <<https://fi.wikipedia.org/wiki/XML>>. Luettu 29.1.2016.
29. Wikipedia. 2013. Murata Electronics. Verkkosivut. <https://fi.wikipedia.org/wiki/Murata_Electronics>. Luettu 3.2.2016.

30. Wikipedia. 2015. WLAN. Verkkosivut. < <https://fi.wikipedia.org/wiki/WLAN>>. Luettu 4.2.2016.
31. Wikipedia. 2015. WAN. Verkkosivut. < Wikipedia. 2015. WLAN. Verkkosivut. < <https://fi.wikipedia.org/wiki/Laajaverkko>>. Luettu 4.2.2016.
32. Techopedia. 2016. Proof of Concept. Verkkosivut. < <https://www.techopedia.com/definition/4066/proof-of-concept-poc>>. Luettu 14.2.2016.
33. Matti Luukkainen. Ohjelmistojen mallintaminen. Verkko-opetusmateriaali.< <http://www.cs.helsinki.fi/u/mluukkai/ohmas10/luentokalvot/luento1.pdf>>. Luettu 14.2.2016.
34. Metropolia IoT-projekti. 2016. BCG-signaali. Luettu 4.2.2016
35. Saurab Katiyar. 2012. RackSpave vs Amazon WebServices. Verkkoartikkeli. < <http://www.slideshare.net/katiyarsaurabh/cloud-conversation-amazon-vs-rack-space> >. Luettu 2.12.2015.
36. Beddit. 2016. Verkkosivut. < <http://www.beddit.com/features/>>. Luettu 26.2.2016.

Python-koodi BCG-anturin datan parsimiseen ja tietokantaan tallentamiseen

2/3/2016

dbconnectex.py

```
1.
2.
3. import mysql.connector
4.
5. #Avaa yhteyden mysql tietokantaan
6.
7. cnx = mysql.connector.connect(user='root', password='mysql1234',
8.                                host='localhost',
9.                                database='anturi')
10. cnx.connect()
11.
12. #Avaa cursorin
13.
14. c = cnx.cursor()
15.
16.
17. #Avaa muratan filun, jossa sensorin tiedot xml muodossa.
18. #Keraa filusta sensorin lahettamat tiedot rivi riviltä.
19. #suorittaa sql komennon, joka tallentaa rivin tiedot mysql:aan.
20.
21. fh = open('murata.xml')
22. for line in fh:
23.     if line.startswith("<Measurement"):
24.         print line[51:59]
25.
```

```
26. if not line.startswith("<"):
27.
28.     x = line.split(",")
29.
30.     tick = int(x[0])
31.
32.     heartbeat = int(x[1])
33.
34.     respiration = int(x[2])
35.
36.     relativestroke = int(x[3])
37.
38.     hrv = int(x[4])
39.
40.     signalstrength = int(x[5])
41.
42.     status = int(x[6])
43.
44.     btb = int(x[7])
45.
46.     btb1 = int(x[8])
47.
48.     btb2 = int(x[9])
49.
50.
51. c.execute("INSERT INTO sensor (Tick, HeartRate, RespirationRate, RelativestrokeVolume, HeartRateVariability, MeasuredSignalStrength, Status, BeatToBeatTime, BeatToBeatTime1, BeatToBeatTime2)
52. (tick, heartbeat, respiration, relativestroke, hrv, signalstrength, status, btb, btb1, btb2)")
53.
```

```
55.  
56.  
57.  
58.  
59. #Valitsee kaikki sensori-nimisessa tablessa olevat datat ja pastee komentoriville.  
60.  
61. c.execute("SELECT * FROM sensor")  
62.
```

2/3/2016

```
63. rows= c.fetchall()  
64. #cnx.commit() #TALLENTAA TIEDON TIETOKANTAAN  
65.  
66. for eachRow in rows:  
67.     print eachRow  
68.  
69. cnx.close()  
70.  
71. #Mysql sijainti koneella:  
72. #C:\Program Files\MySQL\MySQL Server 5.6\bin>
```

dbconne

Murata.xml-tiedosto

BCG-anturin lähettämää dataa xml-muodossa.

```
<?xml version="1.0"?>
<Data xmlns="urn:wsn-openspi:sidf" version="1.7">
  <Network id="murata">
    <Node id="murata">
      <Sensor id="0">
        <Measurement time="2015-09-15T06:47:46+00:00" quantity="BioSignal">
          <Component id="heart rate" unit="bpm"/>
          <Component id="respiration rate" unit="rpm"/>
          <Component id="relative stroke volume" unit="ul"/>
          <Component id="heart rate variability" unit="ms"/>
          <Component id="measured signal strength"/>
          <Component id="status"/>
          <Component id="beat-to-beat time" unit="ms"/>
          <Component id="beat-to-beat time -1" unit="ms"/>
          <Component id="beat-to-beat time -2" unit="ms"/>
          <Values tick="sec"> 18061,0,0,0,0,332,1,0,0,0 18062,0,0,0,0,407,1,0,0,0 18063,0,0,0,0,397,1,0,0,0 18064,0,0,0,0,376,1,0,0,0
18065,0,0,0,0,352,1,0,0,0 18066,0,0,0,0,351,1,0,0,0 18067,0,0,0,0,349,1,0,0,0 18068,0,0,0,0,325,1,0,0,0 18069,0,0,0,0,374,1,0,0,0
18070,0,0,0,0,331,1,0,0,0 18071,0,0,0,0,347,1,0,0,0 18072,0,0,0,0,336,1,0,0,0 18073,0,0,0,0,337,1,0,0,0 18074,0,0,0,0,388,1,0,0,0
18075,0,0,0,0,343,1,0,0,0 18076,0,0,0,0,315,1,0,0,0 18077,0,0,0,0,304,1,0,0,0 18078,0,0,0,0,344,1,0,0,0 18079,0,0,0,0,357,1,0,0,0
18080,0,0,0,0,332,1,0,0,0 18081,0,0,0,0,325,1,0,0,0 18082,0,0,0,0,372,1,0,0,0 18083,0,0,0,0,387,1,0,0,0 18084,0,0,0,0,366,1,0,0,0
18085,0,0,0,0,355,1,0,0,0 18086,0,0,0,0,367,1,0,0,0 18087,0,0,0,0,355,1,0,0,0 18088,0,0,0,0,318,1,0,0,0 18089,0,0,0,0,310,1,0,0,0
18090,0,0,0,0,301,0,0,0,0 18091,0,0,0,0,359,1,0,0,0 18092,0,0,0,0,365,1,0,0,0 18093,0,0,0,0,379,1,0,0,0 18094,0,0,0,0,342,1,0,0,0
18095,0,0,0,0,373,1,0,0,0 18096,0,0,0,0,360,1,0,0,0 18097,0,0,0,0,341,1,0,0,0 18098,0,0,0,0,366,1,0,0,0 18099,0,0,0,0,315,1,0,0,0
18100,0,0,0,0,340,1,0,0,0 18101,0,0,0,0,375,1,0,0,0 18102,0,0,0,0,373,1,0,0,0 18103,0,0,0,0,325,1,0,0,0 18104,0,0,0,0,306,1,0,0,0
18105,0,0,0,0,287,0,0,0,0 18106,0,0,0,0,356,1,0,0,0 18107,0,0,0,0,287,0,0,0,0 18108,0,0,0,0,284,0,0,0,0 18109,0,0,0,0,293,0,0,0,0
18110,0,0,0,0,271,0,0,0,0 18111,0,0,0,0,327,1,0,0,0 18112,0,0,0,0,301,0,0,0,0 18113,0,0,0,0,338,1,0,0,0 18114,0,0,0,0,368,1,0,0,0
18115,0,0,0,0,346,1,0,0,0 18116,0,0,0,0,306,1,0,0,0 18117,0,0,0,0,284,0,0,0,0 18118,0,0,0,0,334,1,0,0,0 18119,0,0,0,0,353,1,0,0,0
18120,0,0,0,0,382,1,0,0,0 </Values>
        </Measurement>
      </Sensor>
    </Node>
  </Network>
</Data>
```