

Skybaserte løsninger for avstandsoppfølging av kroniske pasienter:

Design og evaluering av et pulsoksymeter

Andreas Drivenes

Master i datateknologi

Innlevert: juni 2017

Hovedveileder: Dag Svanæs, IDI

Norges teknisk-naturvitenskapelige universitet
Institutt for datateknologi og informatikk

Sammendrag

Kroniske sykdommer er den mest vanlige dødsårsaken på verdensbasis. Ved å følge opp pasienter i sitt eget hjem med egenrapportering av dagsform og innrapportering av sensordata, ønsker man å redusere sykehusinnleggelses- og øke trygghets- og mestringsfølelsen. Avstandsoppfølging av kronisk syke (avstandsoppfølging) er et satsingsområde for Norge som har bevilget 30 millioner til et prosjekt for å teste ut dette i fire kommuner: Sarpsborg, Oslo, Stavanger og Trondheim.

Skybasert tingenes internett (IoT) blir presentert som en teknologi som skal løse en mengde problemer knyttet til innsamling av sensordata i stor skala. Nye plattformer som AWS IoT har blitt lansert for å støtte denne utviklingen med ferdig infrastruktur og komponenter. I denne masteroppgaven ble en frittstående prototype av et pulsoksymeter koblet til AWS IoT, evaluert for bruk i avstandsoppfølging. Hensikten med dette var å finne en arkitektur for avstandsoppfølging basert på moderne IoT-skyløsninger, og finne ut hva som vil bli viktige aspekter ved utviklingen av en slik teknologisk plattform fremover. Prototypen var basert på en Raspberry Pi Zero W, en fingeravtrykkssensor for autentisering av pasienten, en knapp og tre lysdioder.

Det ble gjennomført en case-studie for å se hvordan Trondheim kommune jobber med avstandsoppfølging, og hva som er status på det prosjektet i første halvdel av 2017. Et bakgrunnsintervju med Trondheim kommune ble gjennomført. På slutten av prosjektet ble prototypen vist fram til kommunen og SINTEF i to ulike intervjuer.

Resultatene viste at en mulig arkitektur for avstandsoppfølging kan realiseres med skyplattformen AWS IoT som gir god innebygd sikkerhet. Prosjektet identifiserte fire implikasjoner for utviklingen av skybasert IoT som teknologisk plattform for avstandsoppfølging: sikkerhet og autentisering, personvern, utviklingsomgivelser og pris og administrasjon. Med tanke på sikkerhet og autentisering er det viktig å benytte ende-til-ende-kryptering og tofaktorautentisering. Benytt helst Bluetooth-versjon 4.2 eller nyere. For personvern gjelder det å sørge for at brukeren kan kontrollere egne data og tenke på hvordan de nye personvernreglene i 2018 vil påvirke systemer. AWS IoT var enkelt å benytte for utviklere og gjorde at mer tid kunne brukes til å jobbe med selve prototypen. Kommunen måtte bruke mye ressurser på å eie hele verdikjeden selv. Det kan være billigere og enklere kun å dele ut sensorer.

Trondheim kommune påpekte at det vanskelige med avstandsoppfølging ikke nødvendigvis er teknologien, men å få løsningen til å fungere i storskala i en kommune med flere hundre brukere. De ville kanskje heller ha en løsning med flere sensorer rundt en sentral hub (som godt kunne være frittstående), enn å ha alt integrert i sensoren. De så for seg en fremtid der brukeren selv skal kunne velge hva slags løsning de vil bruke og at kommunen kan være til stede på de plattformene brukeren er fra før av. Forskere ved SINTEF mente at rapportering av dagsform burde vært med i prototypen. Helst ville de også ha inn egenbehandlingsplanen til brukeren inn slik at den ikke bare er på papir. De var teknologioptimistiske når det gjaldt bruken av sensordata i fremtiden og så på automatisk analyse av data og bedre toveiskommunikasjon som noe positivt å jobbe med videre.

Abstract

Chronic diseases are the most common cause of death worldwide. By following up on patients in their own homes by self-reporting their daily form and reporting sensor data, the goals are to reduce hospitalizations and increase the sense of security and achievement. Remote monitoring of patients with chronic diseases (remote monitoring) is a priority area for Norway, which has allocated 30 million NOK to a project to test this in four municipalities: Sarpsborg, Oslo, Stavanger and Trondheim.

Cloud-based Internet of Things (IoT) is presented as a technology that will solve problems with collecting data from sensors in a large scale. New platforms such as AWS IoT has been released to support this development with infrastructure and components. In this master's thesis, a prototype of a stand-alone pulse oximeter connected to AWS IoT was evaluated for use in remote monitoring. The purpose of this was to find an architecture for remote monitoring based on modern IoT solutions in the cloud released during the last couple of years, and what the important aspects will be with the development of such a technological platform in the future. The solution was based on a Raspberry Pi Zero W, a fingerprint sensor for authentication of the patient, one button and three LEDs.

A case study was conducted to see how the municipality of Trondheim is working with remote monitoring, and what the current situation is in that project in the beginning of 2017. One background interview with the municipality of Trondheim was conducted. In the end of the project, the prototype was showed to the municipality of Trondheim and SINTEF in two separate interviews.

The results showed that a possible architecture for remote monitoring can be realized with the cloud platform AWS IoT, which gives good built-in safety. The project identified four implications for the development of cloud-based IoT as a technological platform for remote monitoring: security and authentication, protection of privacy, developer experience and price/administration. In regards to security and authentication, it is important to use end-to-end encryption and two-factor authentication. Use Bluetooth version 4.2 or newer if you can. For personal privacy, make sure that user maintains control over their own data and think about how the new regulations (GDPR) from 2018 affect systems. AWS IoT was simple to use from a developer's point of view, and this saved time to work on the prototype itself. The municipality had to use a lot of resources to own the entire value chain. It could be simpler and cheaper to only give out sensors.

The municipality of Trondheim pointed out that the difficulty in remote monitoring is not necessarily the technology, but making the solution work in large scale in a municipality with hundreds of users. They would perhaps rather have a solution with sensors around a central hub (which could have been stand-alone) than having everything be integrated in the sensor. They projected a future where the user can choose the solution they want to use and that the municipality can be present on the platforms the users already are on. The researchers from SINTEF thought that self-reporting the daily form should have been in the prototype. Preferably, they wanted the self-treatment plan in the solution, and not only on a piece of paper. They were optimistic about the technology when it came to reporting of sensor data, and they viewed automatic analysis of data and better two-way communication as something positive to work on in the future.

Forord

Takk til veileder professor Dag Svanæs for kjempegod hjelp og tilbakemeldinger gjennom hele prosjektet. Jeg har satt stor pris på ditt gode humør og engasjement for faget.

Takk til Terje Røsand ved instituttet for hyggelige samtaler og uvurderlig hjelp med 3D-printing, lodding og elektronikk. IDI er heldige som har en så stor ressurs som deg.

En stor takk rettes til Trondheim kommune for godt samarbeid og utlån av pulsoksymeter. Ingar og N.N. – til tross for en veldig travel periode satte dere av tid til meg, takk for det! Takk til SINTEF for hjelpen med å komme i gang med prosjektet, og for at dere stilte på evaluerings- og oppsummeringsintervju.

Takk til Kristoffer Larsen for teknisk hjelp med AWS.

Andreas Drivenes

Trondheim, 19. juni 2017

Innhold

Sammendrag	i
Abstract	iii
Forord	v
Innhold	vii
Tabeller	xi
Figurer	xiii
Forkortelser	xv
1 Introduksjon	1
1.1 Bakgrunn og motivasjon	1
1.2 Forskningsspørsmål	3
1.3 Forskningsmetoder og forskningsdesign	4
1.4 Avgrensning av forskningen	4
1.5 Deltakere i prosjektet	5
1.6 Disposisjon av oppgaven	5
2 Bakgrunn	7
2.1 Velferdsteknologi	7
2.2 Avstandsoppfølging av kronisk syke	9
2.3 Eldres bruk av teknologi	9
2.4 Personvern i helse- og velferdsteknologi	10
2.5 Tingenes internett (IoT)	11
2.5.1 Sikkerhet i tingenes internett	11
2.6 Tingenes internett i velferdsteknologi	13
3 Forskningsmetoder	15
3.1 Forskningsstrategier	15
3.1.1 Design og kreasjon	15
3.1.2 Case-studie	16
3.1.3 Prototyping	17
3.1.4 Brukersentrert utvikling	17
3.2 Datagenereringsmetoder	17
3.2.1 Semistrukturert intervju	17

3.2.2	Dokumenter	18
4	Forskningsdesign	19
4.1	Forskningsspørsmål	19
4.1.1	Forskningsspørsmål 1	19
4.1.2	Forskningsspørsmål 2	20
4.1.3	Forskningsspørsmål 3	21
4.1.4	Forskningsspørsmål 4	21
4.2	Forskningsstrategier	21
4.2.1	Design og kreasjon av et pulsoksymeter	22
4.2.2	Case-studie: Avstandsoppfølging i Trondheim kommune	22
4.3	Forskningsparadigme og forskningskvalitet	23
5	Case-studie: Avstandsoppfølging i Trondheim kommune	25
5.1	Innsamling av data	25
5.2	HelsaMi+: Bakgrunn og tjenesteforløp	26
5.3	Motivasjon for HelsaMi+	27
5.4	Politikk og strategi i HelsaMi+	28
5.5	Demonstrasjon av HelsaMi+	29
5.5.1	Rapportere dagsform	29
5.5.2	Utføre en måling med pulsoksimeter	30
5.5.3	Se på innrapport data	30
5.6	Utfordringer og refleksjoner fra prosjektledelsen	34
5.6.1	Samhandling	34
5.6.2	Juridisk	34
5.6.3	Sensorer, måledata og egenrapportering	35
5.6.4	Brukeropplevelse, sikkerhet og teknologi	36
5.7	HelsaMi+ i fremtiden	37
6	Kvalitetskrav til løsning	39
6.1	Sikkerhet	39
6.2	Personvern	40
6.3	Interoperabilitet	40
6.4	Tilgjengelighet	40
6.5	Ytelse	40
6.6	Brukervennlighet	41
7	Teknologi	43
7.1	Protokoller og kommunikasjon	43
7.1.1	MQTT	43
7.1.2	Bluetooth Low Energy	44
7.1.3	Seriell kommunikasjon og GPIO	45
7.2	Skyteknologi	46
7.2.1	Skybasert tingenes internett	46
7.2.2	AWS IoT	46
7.2.3	Microsoft Azure IoT Hub	48
7.3	Prototypeplattform: Raspberry Pi Zero W	49
7.4	Sensorer og aktuatorer	50

7.4.1	Pulsoksymeter	50
7.4.2	Fingeravtrykksensor	50
7.4.3	Trykknapp og lysdioder	50
8	Design og implementasjon av et skytilkoblet pulsoksymeter	53
8.1	Designkrav til prototype	53
8.2	Bruksscenario	54
8.3	Beskrivelse av prototype	55
8.4	Implementasjon av klientkomponenter	56
8.4.1	Trykknapp og lysdioder	56
8.4.2	GT-511C3	59
8.4.3	Nonin 3230	59
8.4.4	3D-printet hus	60
8.5	Implementasjon av skytilkobling	61
8.5.1	Registrering og oppkobling til AWS IoT	61
8.5.2	Oppsett av businessregler og visning av data	64
9	Evaluering av prototype og oppsummering	67
9.1	Planlegging av evaluering	67
9.2	Intervju med Trondheim kommune	67
9.2.1	Tilbakemeldinger på prototypen	68
9.2.2	Brukeropplevelse	68
9.2.3	Pris	69
9.2.4	Sikkerhet og autentisering	69
9.2.5	Fremtidige løsninger i avstandsoppfølging og HelsaMi+	69
9.3	Intervju med SINTEF	70
9.3.1	Forbedringspotensiale i HelsaMi+	71
9.3.2	Bruken av sensorer og måledata	71
9.3.3	Tilbakemeldinger på prototypen	72
9.3.4	Oppsummering og veien videre	73
10	Diskusjon	75
10.1	Metodediskusjon	76
10.2	Validitet	77
10.2.1	Bekreftbarhet og pålitelighet	77
10.2.2	Troverdighet	77
10.2.3	Overførbarhet	77
10.3	Avgrensning av forskningen	78
11	Konklusjon	79
11.1	Forskningsspørsmål 1	79
11.2	Forskningsspørsmål 2	79
11.3	Forskningsspørsmål 3	80
11.4	Forskningsspørsmål 4	80
11.4.1	Sikkerhet og autentisering	80
11.4.2	Personvern	81
11.4.3	Utviklingsomgivelser	81
11.4.4	Pris og administrasjon av utstyr	81

11.5 Refleksjoner rundt overordnet forskningsspørsmål	82
11.6 Videre arbeid	82
Referanser	85
A Kode	89
A.1 Hovedapplikasjon (smart-pulse-oximeter)	89
A.2 pigpio-components	92
A.3 nonin-3230-ble	95
B Formell henvendelse til Trondheim kommune	97
C Henvendelse til Nonin om pulsoksymeter	99
D Invitasjoner til intervju	101
D.1 Intervjuhenvendelse til Trondheim kommune (case-studie)	101
D.2 Intervjuhenvendelse til Trondheim kommune og SINTEF (evaluering og oppsummering)	101
D.3 Intervjuhenvendelse til teknologileverandøren Imatis	102
E Forberedelse til intervju med Trondheim kommune	103
F Samtykkeerklæringer	107
F1 Anonym	107
F2 Ikke-anonym	108

Tabeller

8.1	Nonin 3230: Format på datapakke (Nonin Medical, Inc., 2016, s. 20)	60
8.2	Nonin 3230: Format på statusfeltet til datapakke (Nonin Medical, Inc., 2016, s. 20)	61

Figurer

1.1	Bruk av nettbrett i HelsaMi+	2
1.2	Bruk av digital penn (Solberg, 2009)	3
2.1	Den nye arkitekturen (Porter & Heppelmann, 2014).	12
2.2	Egenskapene til smarte produkter (Porter & Heppelmann, 2014).	12
2.3	Oversikt over Continua-rammeverket	13
3.1	Modell av forskningsprosessen fra Oates (2006).	16
3.2	Gjensidige avhengigheter innenfor menneskeorientert designaktiviteter (International Organization for Standardization [ISO], 2010).	18
4.1	Forskningsstrategien til FS1	20
4.2	Forskningsstrategien til FS2	20
4.3	Forskningsstrategien til FS3	21
4.4	Forskningsstrategien til FS4	21
5.1	HelsaMi+: Hovedskjerm	29
5.2	HelsaMi+: Tilbakemelding KOLS	30
5.3	HelsaMi+: Spørsmål 1	31
5.4	HelsaMi+: Sammendrag	31
5.5	HelsaMi+: Pulsoksimeter-oversikt	32
5.6	HelsaMi+: Pulsoksimeter-veiledning	32
5.7	HelsaMi+: Pulsoksimeter-måleskjerm	33
5.8	HelsaMi+: Oversikt for administrator 1/2	33
7.1	Hierarki for GATT-profiler (Bluetooth.com, udatert)	44
7.2	Seriell databuss (sparkfun.com, udatert)	46
7.3	AWS IoT: Hvordan det virker (Amazon Web Services, 2016b)	47
7.4	Microsoft Azure IoT Hub: Løsningsarkitektur (Azure, 2017)	48
7.5	Raspberry Pi Zero W. Foto: http://www.raspberrypi-spy.co.uk/	49
7.6	Pi Zero W: GPIO header. Foto: http://www.raspberrypi-spy.co.uk	50
7.7	Nonin 3230	51
7.8	GT-511C3	51
7.9	RGB-lysdiode, motstander og trykknapp	52
8.1	Stegene i brukerinteraksjonen (Drivenes, 2017)	54
8.2	Prototypen ved fungerende måling	55
8.3	Arkitekturoversikt over løsningen	56
8.4	Oppkobling av trykknapp og lysdioder	57

8.5	Oppkobling av trykknapp og lysdioder på brett	58
8.6	Koblingskjema av trykknapp og lysdioder	58
8.7	3D-modell av prototype: Topp	62
8.8	3D-modell av prototype: Topp, sett fra bunn	62
8.9	3D-modell av prototype: Bunn	63
8.10	Grafana-dashboard av pulsoksymeterdataen	66
8.11	Oppsett av Grafana-dashboard	66

Forkortelser

AWS Amazon Web Services. 1, 3, 40, 46

BLE *Bluetooth Low Energy*. 43, 44, 50, 55, 60

GPIO *general-purpose input/output*. 43, 45, 49

IoT tingenes internett. 1, 2, 3, 4, 5, 7, 11, 19, 20, 21, 43, 45, 46, 77, 79, 80, 82

M2M Machine to Machine. 43

MITM *man-in-the-middle-angrep*. 39, 44, 45

MQTT *MQ Telemetry Transport*. 43, 44

NPM *Node Package Manager*. 59

TCP *Transmission Control Protocol*. 43

TLS *Transport Layer Security*. 44, 46, 79

UUID Universally unique identifier. 59

Let us change our traditional attitude to the construction of programs: Instead of imagining that our main task is to instruct a computer what to do, let us concentrate rather on explaining to human beings what we want a computer to do.

Donald Knuth, *Literate Programming*

Kapittel 1

Introduksjon

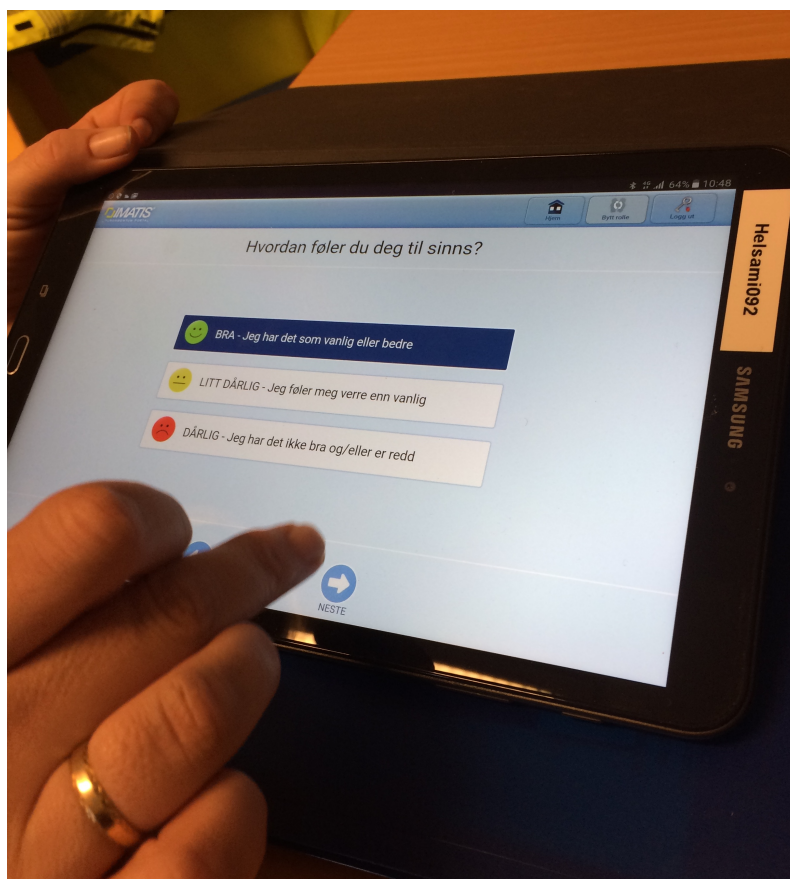
Hensikten med dette forskningsprosjektet er å evaluere ny teknologi for avstandsoppfølging av kronisk syke i form av en første prototype av et skytilkoblet pulsoksymeter. Prototypen vil vises fram for prosjektledere og forskere innen velferdsteknologi for å avdekke hva som vil bli viktige aspekter ved utviklingen av helseprodukter basert på tingenes internett (IoT)-skyløsninger i fremtiden. Det overordnede forskningsspørsmålet er: «Hvor egnet er skybasert IoT som teknologisk plattform for avstandsoppfølging av kronisk syke?» Velferdsteknologiprogrammet i Trondheim kommune og SINTEF bistår prosjektet. Trondheim brukes som eksempelkommune på hvordan avstandsoppfølging prøves ut i starten av 2017.

Dette kapittelet beskriver bakgrunnen for forskningsprosjektet: hva som motiverer det, forskningsspørsmålene som skal besvares og forskningsdesignet som understøtter det, hvilke begrensninger og avveininger som er gjort og til slutt en disposisjon av oppgaven.

1.1 Bakgrunn og motivasjon

«Tingenes internett (IoT) er en global infrastruktur for informasjonssamfunnet som muliggjør avanserte tjenester ved å knytte ting sammen fysisk og virtuelt, basert på eksisterende og kommende kompatible informasjon- og kommunikasjonsteknologier» (ITU, 2012). IoT gir nye og spennende muligheter for kreative løsninger på problemer på forskjellige områder, alt fra husholdningsapparater og fjernoppdatering av programvare i biler, til oppfølging av helseproblemer innen velferdsteknologi. Datakraft og lagring har blitt mye billigere, noe som gjør at man i større grad enn tidligere kan samle inn informasjon om verden rundt oss med sensorer. De to største leverandørene av skytjenester, Amazon Web Services (AWS) og Microsoft Azure, har i løpet av det siste halvannet året lansert nye tjenester for å få sensorenheter tilkoblet til Internett på en enkel og sikker måte (Amazon Web Services, 2015) (Microsoft Azure, 2015).

Velferdsteknologi har vært høyt prioritert av helsemyndighetene i Norge de siste årene. Trondheim kommune prøver ut et prosjekt med avstandsoppfølging av kronisk syke kalt HelsaMi+ på oppdrag fra Helsedirektoratet. Avstandsoppfølging kan øke trygghets-

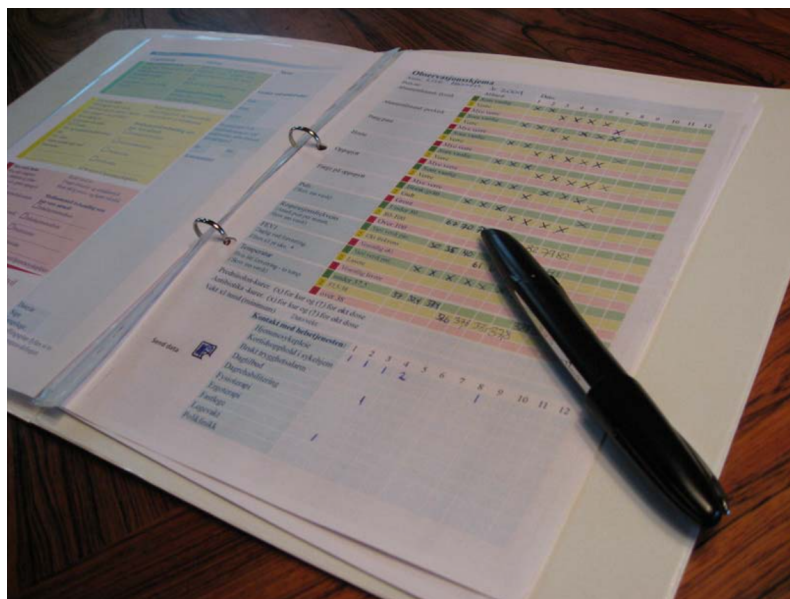


Figur 1.1: Bruk av nettbrett i HelsaMi+

følelsen for innbyggerne og føre til lavere kostnader og færre sykehusinnleggelses. Løsningen til Trondheim kommune innebærer bruk av et nettbrett der man daglig svarer på spørsmål om hvordan formen er (figur 1.1). Noen av pasientene kan måle vekt, blodtrykk, pulsfrekvens og oksygenmetning ved hjelp av sensorer koblet til nettbrettet.

Tidligere prosjekter som KOLS Heim har også undersøkt hvordan en bedre kan behandle kroniske pasienter hjemme, og få til økt grad av samarbeid mellom sykehusene og den kommunale helsetjenesten. KOLS Heim gikk fra 2008 til 2011, og målet var at «(...) bedre samhandling skal kunne bedre pasientens behandlingstilbud, helsetilstand og livskvalitet, øke kunnskapen om KOLS innen hjemmetjenesten og forhindre hyppige sykehusinnleggelses» (Hukkelås, 2008). Masteroppgaven «Alternativer for registrering av pasientopplysninger i hjemmetjenesten» sammenlignet tre ulike teknologier for KOLS Heim: digital penn og papir, PDA/mobiltelefon og laptop (figur 1.2). Der ble digital penn og papir foretrukket (Solberg, 2009). Mye har skjedd i den digitale utviklingen siden den gang, avstandsoppfølging blir dermed en naturlig etterfølger til arbeidet og forskningen som er gjort tidligere.

Med avstandsoppfølging følger det med seg en rekke utfordringer knyttet til sikkerhet, autentisering, samhandling, teknologi og brukervennlighet. Hvordan ivaretar man sikkerheten og personvernet til brukeren på en brukervennlig måte? Dette er krav som skybasert IoT allerede delvis løser, eller må løse. Mulighetene er store for denne teknologien med en ny infrastruktur for ting koblet sammen istedenfor infrastruktur for



Figur 1.2: Bruk av digital penn (Solberg, 2009)

datamaskiner. Prosjektet vil derfor gjøre en utforsking av tinginfrastrukturen rettet mot helseområdet.

Motivasjonen for forskningsprosjektet er å ta i bruk den nye skyteknologien fra AWS i en frittstående pulsoksymeterprototype for avstandsoppfølging av kronisk syke som et alternativ til den løsningen Trondheim kommune tester ut i dag. Prosjektet vil undersøke hvordan teknologien kan hjelpe til med å understøtte tjenesten som testes ut, og om den kan avlaste noen av utfordringene kommunen opplever i dag.

Utviklingen av en ny og frittstående prototype gjør at dette prosjektet kan sammenligne de to løsningene og undersøke hvordan ny teknologi kan brukes i avstandsoppfølging av kronisk syke. Et pulsoksymeter er en sensor som kan måle pulsfrekvens og oksygenmetning i blodet.

1.2 Forskningsspørsmål

Skybasert IoT blir presentert som en teknologi som skal løse en mengde problemer knyttet til innsamling av sensordata i stor skala. Hvis dette er sant, så vil det forenkle utviklingen av systemer fordi man kan nyttiggjøre seg ferdig infrastruktur og komponenter. Allikevel er det ennå få systemer som baserer seg på den nye teknologien. Det er et åpent forskningsspørsmål om teknologien kan brukes i helsedomenet som har mange komplekse problemstillinger. Avstandsoppfølging passer godt inn som et praktisk eksempel, og utifra lovnadene har skybasert IoT potensiale til å forenkle utviklingen i stor grad. Om dette fungerer i praksis er et åpent spørsmål. Det motiverer følgende forskningsspørsmål:

- FS1** Hvordan gjøres avstandsoppfølging av kronisk syke i prosjektet HelsaMi+ i Trondheim, og hvilke utfordringer ser prosjektledelsen når det gjelder denne type teknologi/tjenester?

FS2 Hva er en mulig arkitektur for realisering av skybasert IoT for avstandsoppfølging av kronisk syke som adresserer utfordringene fra FS1?

FS3 Hvordan vurderer domeneeksperter innen velferdsteknologi frittstående skybaserte IoT-løsninger for avstandsoppfølging av kronisk syke?

FS4 Basert på funnene fra FS1, FS2 og FS3, hva er implikasjonene for utviklingen av skybasert IoT som teknologisk plattform for avstandsoppfølging av kronisk syke?

Heretter vil «avstandsoppfølging av kronisk syke» bli betegnet som «avstandsoppfølging». Brukere av tjenesten vil som oftest bli betegnet som «brukere», men i noen tilfeller kan det bety det samme som «pasient».

1.3 Forskningsmetoder og forskningsdesign

Rammeverket for å beskrive forskningsprosjektet er hentet fra Oates (2006). Hovedstrategien for å besvare på **FS1** i delkapittel 1.2 er å gjøre en liten case-studie på hvordan avstandsoppfølging foregår i Trondheim kommune. Datagenereringsmetoder er intervjuer og dokumenter.

For å svare på **FS2** og **FS3** vil den primære strategien være *design og kreasjon* (prototyping/*proof of concept*) med intervjuer og dokumenter som datagenereringsmetode. Case-studien vil også hjelpe til med å besvare disse spørsmålene. **FS4** vil drøftes kvalitativt utifra svarene på de andre forskningsspørsmålene. Det vil kun gjøres kvalitative dataanalyser.

Forskningsmetodene og forskningsdesignet er utdypet i kapittel 3 og 4.

1.4 Avgrensning av forskningen

Om man kan gjøre kliniske vurderinger basert på sensordataene, kommer ikke til å være i søkelyset for dette prosjektet, men vil drøftes kort i kapittel 5 med kommentarer fra en fastlege. Det er noe som kunne vært ytterligere problematisert i en annen oppgave. Denne forskningen kommer til å anta at det er nyttig å samle inn helsedata fra sensorer for å kartlegge helsetilstanden til kroniske pasienter.

Prototypen gjennomgår ikke brukbarhetstesting. En annen tilnærming til oppgaven kunne vært å gjennomføre brukertester på pasienter eller andre brukere, men det fører med seg krav om behandling av sensitive helseopplysninger.

Det vil også være umulig å ta hensyn til alle krav og regelverk i helsesektoren ved implementasjonen av prototypen og skyløsningen. Målet vil heller være å se hvordan teknologien kan understøtte de utfordringene som blir observert innenfor avstandsoppfølging og hvilke muligheter ny teknologi kan gi.

1.5 Deltakere i prosjektet

Fra Trondheim kommune sin side er Ingar Børre Sandvik (prosjektleder, HelsaMi+, program for velferdsteknologi) og en som jobber med velferdsteknologi involert (N.N.). Kommunen låner ut et pulsoksymeter.

SINTEF bidrar til forskningsprosjektet. SINTEF gjør mye forskning på avstandsoppfølging, og bidro også til å få i gang et samarbeid med Trondheim kommune. To forskere fra SINTEF var med på evaluerings- og oppsummeringsintervju der prototypen ble vist fram.

Terje Røsand (overingeniør, institutt for datateknologi og informatikk) bistår med teknisk hjelp til prototyping og 3D-printing.

Alle aktørene er gjort kjent med formålet for forskningsprosjektet og hva den innsamlede dataen skal brukes til. De er informert om at intervjuene tas opp på telefon. De har fått tilbud om å lese gjennom transkriberingen av intervjuene og signert samtykkeerklæringer.

1.6 Disposisjon av oppgaven

Neste kapittel gir en teoretisk bakgrunn innenfor velferdsteknologifeltet og avstandsoppfølging av kronisk syke spesielt. Det inneholde også bakgrunn om eldres bruk av teknologi, personvern, IoT og bruken av IoT i velferdsteknologi.

Kapittel 3 redegjør for forskningsmetodene som brukes i prosjektet, og 4 beskriver hvilket forskningsdesign som brukes i prosjektet for å besvare forskningsspørsmålene.

Kapittel 5 handler om avstandsoppfølging i Trondheim kommune som en case-studie. Det baserer seg på intervjuer gjort med prosjektleder og en ansatt i Trondheim kommune og skjermbilder av løsningen.

I kapittel 6, drøftes kvalitetskrav til en frittstående IoT-løsning til bruk i avstandsoppfølging basert på de tidligere kapitlene.

Kapittel 7 går igjennom teknologi som er relevant for implementasjonen av et frittstående og skytilkoblet pulsoksymeter, før kapittel 8 går i detalj på hvordan prototypen ble laget og hvordan den ble sendt ut.

Evalueringen av prototypen skjer i kapittel 9, der resultatene av intervjuene gjort hos SINTEF og Trondheim kommune oppsummeres.

Avslutningsvis omhandler kapittel 10 diskusjon og drøfting av resultatene fra evalueringene, og konklusjonen i kapittel 11 oppsummerer funnene i oppgaven med bakgrunn i forskningsspørsmålene, med en del om videre arbeid.

Kapittel 2

Bakgrunn

Første delkapittel gir en innføring i velferdsteknologi, og tegner opp linjene for hvordan veien har gått fra Hagen-utvalget til utprøving av avstandsoppfølging i kommunene (delkapittel 2.2). Videre presenteres resultatene av en undersøkelse av eldres bruk av teknologi i Trondheim, og personvern i helse- og velferdsteknologi blir kort drøftet med utgangspunkt i Datatilsynet sine retningslinjer og den nye personvernsforordningen til EU. De to siste delkapitlene handler om IoT med vekt på hvordan denne plattformen legger til rette for en ny arkitektur, og til slutt kort om bruken av IoT i velferdsteknologi.

2.1 Velferdsteknologi

NOU 2011: 11 (Hagen-utvalget, 2011) er en norsk offentlig utredning (NOU) om velferdsteknologi med tittelen «Innovasjon i omsorg». Utredningen benytter følgende definisjon av velferdsteknologi:

Med velferdsteknologi menes først og fremst teknologisk assistanse som bidrar til økt trygghet, sikkerhet, sosial deltakelse, mobilitet og fysisk og kulturell aktivitet, og styrker den enkeltes evne til å klare seg selv i hverdagen til tross for sykdom og sosial, psykisk eller fysisk nedsatt funksjons- evne. Velferdsteknologi kan også fungere som teknologisk støtte til pårørende og ellers bidra til å forbedre tilgjengelighet, ressursutnyttelse og kvalitet på tjenestetilbudet. Velferdsteknologiske løsninger kan i mange tilfeller forebygge behov for tjenester eller innleggelse i institusjon.

Hagen-utvalget (2011) anbefaler fem punkter for myndighetene å fokusere på i fremtiden:

1. «Næromsorg» – den andre samhandlingsreformen: mobiliser ressurser i samarbeid med lokalsamfunnet, det sosiale nettverket til pasienten og familien.
2. «Teknoplan 2015» – teknologistøtte til omsorg: bruk ny og eksisterende teknologi for å gi brukere bedre trygghet og muligheten til å bo hjemme og motta støtte.
3. «Nye rom» – fremtidens boligløsninger og nærmiljø: boliger og leiligheter må tilpasses eldre.

4. Et nasjonalt program for kommunal innovasjon i omsorg: kommunene har hovedansvaret for omsorgstjenester, og myndighetene kan støtte kommunene med incentiver til nye løsninger.
5. Omsorgsfeltet som næring: Norge kan være en ledende nasjon i utviklingen av nye omsorgsprodukter, og eksportere disse innovasjonene. Omsorgsfeltet åpnes opp for importering og eksportering av varer og tjenester.

«Teknoplan 2015» legges fram som en trinnvis plan på tre punkter der punkt én handler om å «videreutvikle trygghetsalarmen til en trygghetspakke som inkluderer tilrettelegging for smarthus», punkt to handler om å ta i bruk moderne kommunikasjonsteknologi og sosiale medier for å redusere ensomhet og punkt tre handler om å «ta i bruk teknologi som stimulerer, underholder, aktiviserer og strukturerer hverdagen» (Hagenutvalget, 2011, s. 17).

Myndighetenes rolle blir å tilrettelegge for kommunal innovasjon i omsorgssektoren med et nasjonalt program som kan gi støtte i form av penger og opplæring. Hagenutvalget (2011, s. 19) mener at det må etableres en kommunal innovasjonsskole i samarbeid med KS for ledere og andre omsorgspersoner, og at minst én prosent av budsjettet i omsorgssektoren skal gå til forskning, utvikling og innovasjon.

Stortingsmeldingen «Morgendagens omsorg» bygger på arbeidet til Hagenutvalget (Helse- og omsorgsdepartementet, 2013). I denne meldingen legger regjeringen frem «Omsorgsplan 2020» som inkluderer et program for velferdsteknologi de neste årene. Ett av initiativene i programmet er å bygge og etablere åpne velferdsteknologistandarder. Dette vil lette integreringen av nye løsninger i privat og offentlig sektor. Andre initiativer er å utvikle og teste velferdsteknologiløsninger i kommunene, bygge og dele kunnskap og lage modeller og rammeverk som andre kan bruke.

I revidert statsbudsjett for 2014 bevilget Stortinget penger til et nasjonalt velferdsteknologiprogram. De fem prioriterte initiativene i programmet er:

1. Trygghet og mestring hjemme
2. Avstandsoppfølging av personer med kroniske sykdommer
3. mHelse
4. Sosiale nettverk – motvirke og redusere ensomhet blant eldre
5. Bidra til økt aktivitet (inkl. fritidsaktiviteter) for barn og unge med nedsatt funksjonsevne

«mHelse, eller personlige mobile helseløsninger, er å benytte mobilbaserte verktøy og helseapplikasjoner (helse-apper) til helseformål» (Direktoratet for e-helse, 2016). De ulike initiativene går litt inn i hverandre. Avstandsoppfølging kan føre til mer trygghet og mestring hjemme og ta i bruk mHelse. Avstandsoppfølging er dekket nærmere i delkapittel 2.2.

2.2 Avstandsoppfølging av kronisk syke

Rojahn mfl. (2016) definerer avstandsoppfølging som «overvåking av en poliklinisk pasient med en enhet som overfører data». I avstandsoppfølging følges pasienter opp i sitt eget hjem, og målet med avstandsoppfølging er å unngå sykehusinnleggelse ved å oppdage forverringer tidlig og la pasientene behandle sin egen sykdom. Innleggelse på sykehus koster gjennomsnittlig 40000 kr per døgn og er en stor utgift for helsemyndighetene (Regjeringen, 2014b).

Kroniske sykdommer er den mest vanlige dødsårsaken på verdensbasis (World Health Organization [WHO], udatert). I følge Austad, Liverud, Strisland, Ausen og Reitan (2016), kan kroniske sykdommer som KOLS, hjerte- og karsykdommer og diabetes følges opp klinisk med noen få sensorer: vekt, blodtryksmåler, pulsoksymeter og spirometer. Typiske tegn på hjertesvikt kan være tungpusthet, vektforandring, hoste, hjertebank, hevelser i beina og tretthet/tiltaksløshet (Direktoratet for e-helse, 2015) (Nasjonalforeningen for folkehelsen, 2015).

Fire kommuner ble valgt ut til å ta del i et nasjonalt prosjekt for å legge til rette for økt mestring og trygghet hjemme med nye digitale behandlingsmetoder: Trondheim kommune, Stavanger kommune, Oslo kommune og Sarpsborg kommune. Stortinget bevilget 30 millioner som en del av budsjettavtalen for 2015 til prosjektet som omfatter rundt 4-500 brukere (Regjeringen, 2015). Her er de første forskningsresultatene ventet i midten av 2017. Foreløpige resultater fra Oslo kommune viser at en klar reduksjon i bruken av sykehustjenester (færre konsultasjoner og innleggelser), samt nedgang i bruk av tjenester fra hjemmesykepleien. Den viser også besparelser på 73000 kr i gjennomsnitt hvert år (reduksjon på 32%) (Intro International og Arkitektur-og designhøgskolen i Oslo, 2011).

2.3 Eldres bruk av teknologi

En undersøkelse av 430 åttiåringers teknologibruk i Trondheim viser at majoriteten av benytter en eller annen form for teknologi (Sødal, Klungerbo & Oldervoll, 2016). Sødal mfl. (2016) skriver i oppsummeringen at:

97 prosent benytter mobiltelefon, 55 prosent har brukt pc eller nettbrett for å hente informasjon fra nettet, 47 prosent benytter e-post og 49 prosent nettbank. 54 prosent benytter datamaskin (pc eller mac) og 37 prosent nettbrett. 29 prosent ønsker opplæring i bruk av nettbrett. Menn benytter teknologi mer enn kvinner, men kvinner benytter sosiale medier mest.

Infosenteret for seniorer i Trondheim er et kommunalt tilbud for åttiåringers som ikke bruker noen kommunale tjenester i dag. De gjør forebyggende hjemmebesøk slik at eldre kan leve trygge og selvstendige liv. Samtalen på hjemmebesøket kan handle om blant annet «meningsfylt aktivitet, sosial deltakelse, samt trygghet og sikkerhet i hjemmet. Det vektlegges spesielt hva den enkelte selv mener er viktig for å ivareta egen helse» (Trondheim kommune, 2014).

2.4 Personvern i helse- og velferdsteknologi

For velferdsteknologi spesielt, oppgir Datatilsynet (2014) ti viktige generelle retningslinjer for å sikre personvernet i utviklingen av ny velferdsteknologi:

1. Velg den minst inngripende løsningen
2. Begrens mengden data som lagres
3. Velg sanntidsløsning hvis mulig
4. Lagre lokalt hvis mulig
5. La brukeren ha kontroll over løsningen
6. Slett data etter bruk
7. Begrens tilgangen til informasjon
8. Innsyn i egne data
9. Dataene bør krypteres
10. Anonymisering av data

En ny personsvernsforordning ble vedtatt av EU 27.april i år. Denne overtar etter personverndirektivet som er implementert i norsk lovgivning gjennom personopplysningsloven. Forordningen trer i kraft 25. mai 2018, og medfører strenge sanksjoner til selskaper som bryter reglene. Siden Norge er et EØS-medlem, må hele forordningen tas inn i det norske lovverket.

Datatilsynet (2015) skriver at *innebygd personvern* betyr at det tas hensyn til personvernet i alle utviklingsfaser av et system eller en løsning. Det er både kostnadsbesparende og mer effektivt enn å endre et ferdig system. Innebygget personvern er tatt med i den nye personvernforordningen til EU. Artikkel 25 sier at standarden til et system skal være at man kun prosesserer og lagrer de personopplysningene som skal til for å løse de oppgavene systemet har (Council of European Union, 2016).

Artikkel 7 i forordningen har tittelen Vilkår for samtykke. Selskapet må innhente eksplisitt samtykke, samt vise at det har hentet inn et slikt samtykke. Samtykket må vise hva slags informasjon som samles inn, og hva informasjonen skal brukes til. Et viktig moment her er at dette samtykket skal kunne trekkes tilbake når som helst på en enkel og brukervennlig måte (Council of European Union, 2016).

Det siste viktige punktet i forordningen er den korte fristen fra man oppdager et datainnbrudd til man varsler den relevante myndigheten. Denne fristen er satt til 72 timer. Merk at man ikke nødvendigvis må varsle offentligheten. Om myndigheten tror at innbruddet vil gå ut over individuelle brukere må disse bli varslet av selskapet (Berntsen, 2016).

Personvernsforordningen inneholder hjemler for å straffe selskap som bryter reglene. I verste fall kan et selskap få 20 millioner euro i bot, eller opptil 4 % av fjorårets globale omsetning. Mildere sanksjoner er skriftlige advarsler, og periodiske granskninger av datasikringen (Wikipedia, 2016).

2.5 Tingenes internett (IoT)

Weber og Weber (2010) definerer tingenes internett som en «fremvoksende global Internett-basert informasjonsarkitektur som fasiliterer utvekslingen av varer og tjenester». I denne definisjonen ligger det en visjon om en verden knyttet sammen av objekter som kommuniserer med hverandre.

Andre definisjoner av IoT eksisterer avhengig av bransje. Porter og Heppelmann (2014) skriver at tingenes internett ikke er en veldig god beskrivelse av den nye trenden med sammenkoblede enheter: «Det som gjør smarte, tilkoblede produkter fundamentalt annerledes er ikke Internett, men at «tingenes» natur endrer seg. Det er de nye mulighetene til smarte, tilkoblede produkter og dataen de genererer som fører til en ny periode med konkurranse».

Derfor introduserer de begrepet «smarte produkter» som består av tre elementer:

- Fysiske komponenter: de elektriske og mekaniske delene til produktet.
- «Smarte» komponenter: sensorer, prosessorer, programvare, operativsystem, lagring, brukergrenesnitt.
- Tilkoblingskomponenter: porter, antenner, protokoller som muliggjør kablede/trådløse tilkoblinger én-til-én, én-til-mange, mange-til-mange.

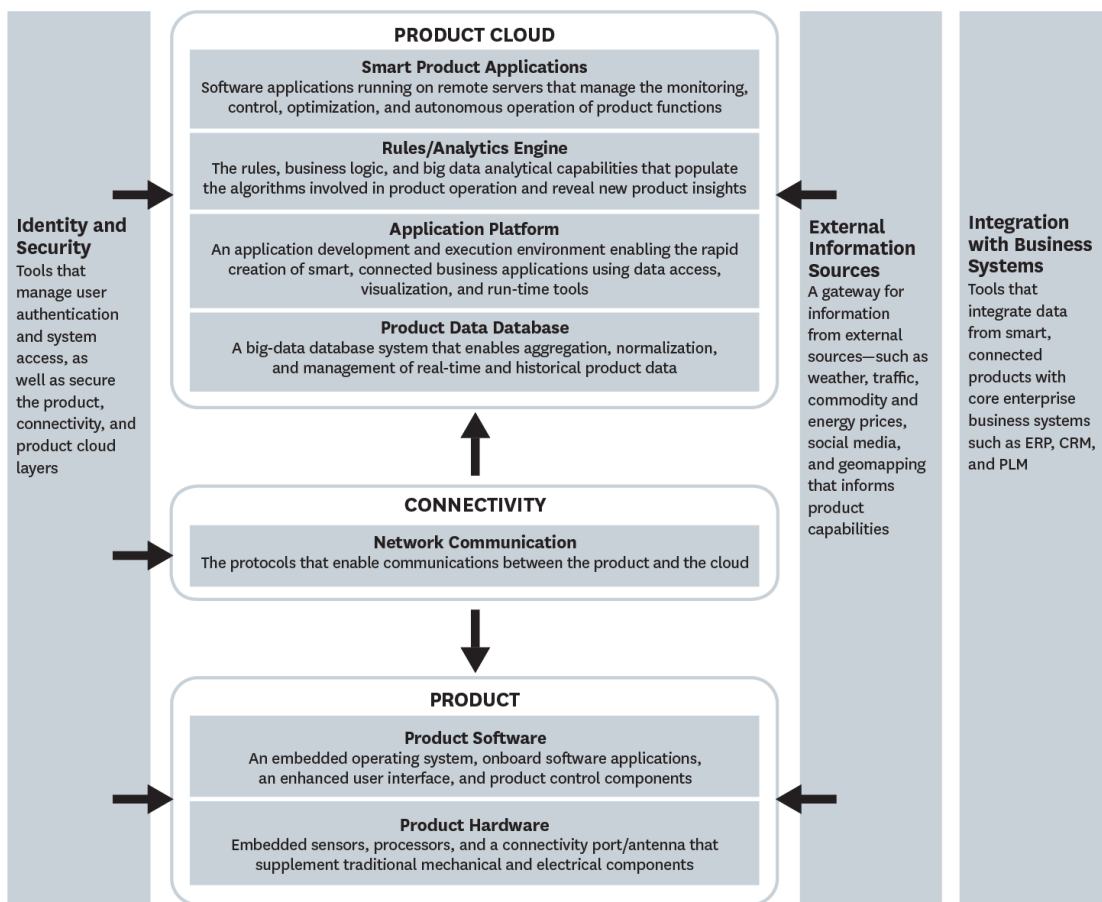
Smarte produkter introduserer en ny arkitektur (figur 2.1) med produkter koblet til en produktsky. Produktskyen inkluderer et «Big Data»-databasesystem og regel- og analysemotor for å håndtere forretningslogikk på en effektiv måte, og utnytte og analysere all informasjonen fra produktene. Disse smarte, tilkoblede produktene har fire egenskaper som bygger på hverandre: Overvåking, kontroll, optimalisering og autonomi. Tesla er et eksempel på et smart produkt som kombinerer overvåking, kontroll og optimalisering for å oppnå autonomi. Bilen diagnostiserer seg selv, og oppdaterer seg selv automatisk uten at den må på verksted.

Porter og Heppelmann (2015) antyder at smarte, tilkoblede produkter endrer selskaper også, ved å endre hvert steg i verdikjeden. Utnyttelse av data blir mer viktig, og produktdesign er ikke bare noe mekanisk, men et tverrfaglig samarbeid der programvareutvikling vil spille en større rolle.

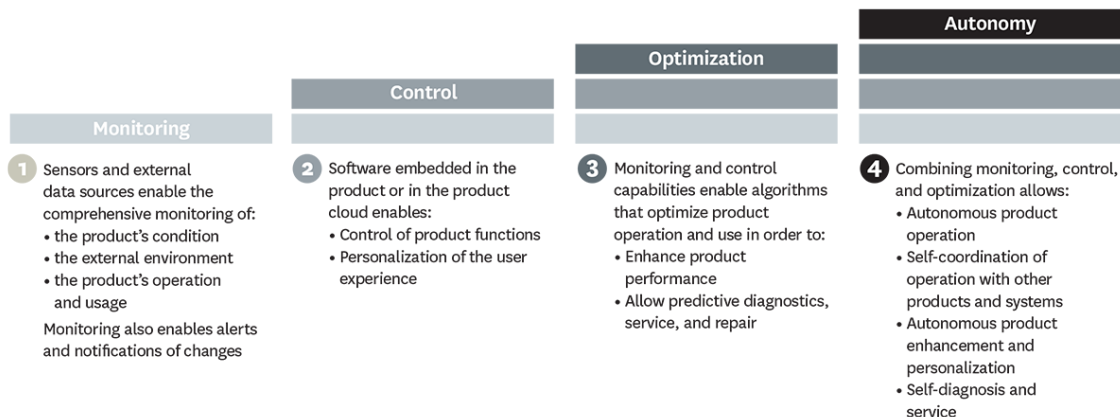
2.5.1 Sikkerhet i tingenes internett

Ekspertene har kommet med innvendinger når det gjelder sikkerhetsmodellen til IoT. 21. oktober 2017 ble Internett-kjerneinfrastrukturselskapet *Dyn* angrepet via flere millioner små enheter (hovedsaklig rutere og kameraer) med lav grad av sikkerhet (Peterson, 2016). Dette påvirket flere store nettsider som Twitter, Spotify og PayPal. Rutere og webkameraer leveres ofte med standardpassord som brukere ikke endrer selv. I tillegg til dette kan flere vanlige porter som 22 (SSH), 80 (HTTP) og 23 (Telnet) være helt åpne mot omverdenen (Zeifman, Bekerman & Herzberg, 2016).

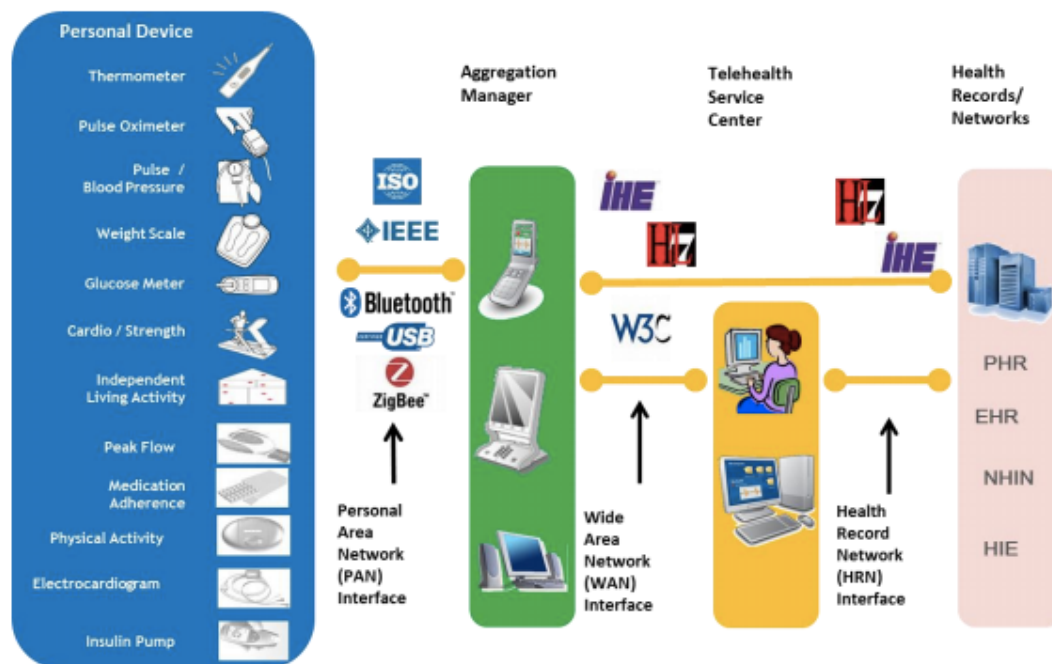
Schneier (2016) oppfordrer myndighetene til å pålegge restriksjoner på IoT. Han argumenterer for at markedet selv ikke klarer å beskytte sikkerheten til forbrukerne. For-



Figur 2.1: Den nye arkitekturen (Porter & Heppelmann, 2014).



Figur 2.2: Egenskaperne til smarte produkter (Porter & Heppelmann, 2014).



Illustrasjon: Slik fungerer Continua: Continua Health Alliance

Figur 2.3: Oversikt over Continua-rammeverket

brukerne bryr seg ikke, produsentene bryr seg ikke og produktene kan aldri patches etter de er solgt og levert. Denne diskusjonen om reguleringer må gjøres før det skjer en IoT-katastrofe, mener Schneier.

2.6 Tingenes internett i velferdsteknologi

Personal Connected Health Alliance (PCHA) publiserer Continua-designretningslinjene hvert år, et åpent rammeverk for ende-til-ende-kompabilitet i personlige, tilkoblede helseenheter og helsesystemer (PCHA, 2016). En oversikt av Continua-rammeverket kan finnes i figur 2.3. Personlige enheter kommuniserer med protokoller som Bluetooth eller Zigbee til en hub (Personal Area Network), og denne huben sender informasjonen videre til et telehelse-servicesenter. Derfra kan dataen overføres til helseregistret.

Som en del av «Omsorgsplan 2020» og nasjonalt program for velferdsteknologi, bestemte regjeringen i slutten av 2014 at Continua-rammeverket skal være grunnlaget for alle velferdsteknologiløsninger i Norge. Dette var også anbefalingen til Helsedirektoratet – å standardisere på ett rammeverk sikrer at ulike løsninger virker godt sammen (Regjeringen, 2014a).

Kapittel 3

Forskningsmetoder

Dette kapitlet redegjør for forskningsmetodene til prosjektet. Forskningsstrategiene og datagenereringsmetodene tar utgangspunkt i Oates (2006). Se figur 3.1 for en oversikt over forskningsrammeverket.

3.1 Forskningsstrategier

En forskningsstrategi er en overordnet tilnærming for å svare på et forskningsspørsmål (Oates, 2006, s. 35).

3.1.1 Design og kreasjon

«Forskningsmetoden design og kreasjon fokuserer på utvikling av nye IT-produkter, også kalt *artefakter*. Typer IT-artifakter inkluderer: begreper, modeller, metoder og implementasjoner» (Oates, 2006). Hovedpoenget er å lære gjennom å lage, og Oates (2006) identifiserer fem steg for denne prosessen:

Bevisstgjøring Hva er problemet? Hva er rammene for problemet?

Forslag Hvordan kan problemet løses? Hva er ideen?

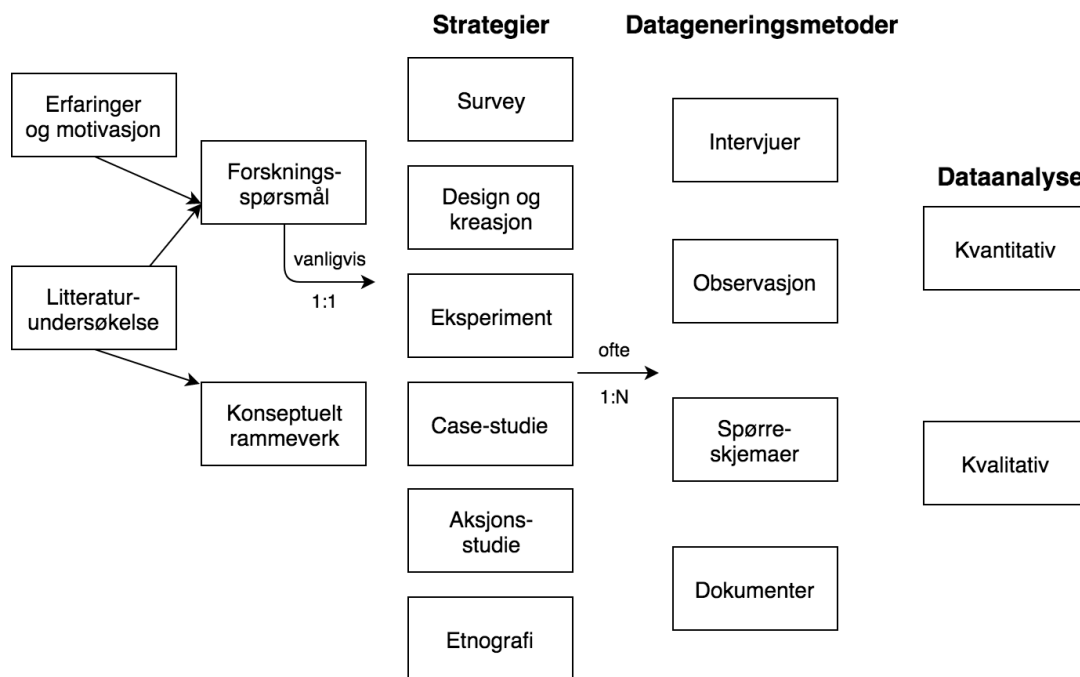
Utvikling Problemløsningsfasen. Implementasjon av løsning.

Evaluering Hvordan gikk dette? Hva var forventningene?

Konklusjon Hva kan vi trekke ut av dette? Hva må man se mer på?

I følge Oates (2006, s.121-122) er fordelene til design og kreasjon-strategien at det er enklere å relatere til noe som eksisterer i virkeligheten enn en idé, en tanke eller et konsept. Den raske teknologiutviklingen gjør også at det er mulig å foreslå og utvikle nye artefakter, og at man dermed bidrar til ny kunnskap.

Ulempen med å velge denne strategien er at det er fort gjort at en kun demonstrerer teknisk kompetanse uten at det kvalifiserer som god forskning. I tillegg til det nye



Figur 3.1: Modell av forskningsprosessen fra Oates (2006).

produktet må ny kunnskap utvikles, bygget på analyse, argumenter og kritiske evalueringer (Oates, 2006, s. 109). En ting er at det lages et nytt produkt i et domene – viktigere momenter er hvilken ny informasjon den fører med seg og konteksten den blir satt inn i.

3.1.2 Case-studie

Yin (2003) definerer i følge Oates (2006) case-studie som «en empirisk undersøkelse som utforsker et samtidfenomen i kontekst til virkeligheten, spesielt når grensene mellom fenomenet og konteksten ikke er helt klart». «En case-studie fokuserer på én instans av det som skal undersøkes: en organisasjon, en avdeling, et informasjonssystem, et diskusjonsforum (...). Denne ene instansen, eller tilfellet, studeres i detalj med forskjellige datagenereringsmetoder som intervju, observasjon, dokumenter (...)» (Oates, 2006).

Det som kjennetegner en case-studie er at det mer fokus på dybde snarere enn bredde, at casen undersøkes i en naturlig ramme på en helhetlig måte, med flere kilder og metoder (Oates, 2006, s. 142). En undersøkende case-studie er nyttig for å kunne forstå forskningsområdet bedre og definere gode forskningsspørsmål, spesielt dersom det eksisterer lite forskning allerede, mens en beskrivende studie vil rette seg mer mot å belyse én sak fra flere sider i en helhetlig historie. Den siste typen case-studie er forklarende, og ønsker å finne ut hvorfor noe spesielt skjedde og hva som forårsaket det (Oates, 2006, s. 143).

Ulemper med denne strategien er, i følge Oates (2006), at mangelen på streng krav kan føre til generaliseringer med lav kredibilitet, at det kan være vanskelig å få tilgang til

riktige personer og riktig informasjon, samt at forskeren i seg selv kan påvirke mye av det som blir gjort fordi det er så få regler.

3.1.3 Prototyping

En prototype er en implementasjon av en artefakt som ikke er produksjonsklar, eller som ikke nødvendigvis er ment for å bli helt ferdig i fremtiden. Ved prototyping vil man ofte lage flere ulike modeller som er delvis ferdige for å finne ut hva som egner seg, og for å få ny kunnskap om det som lages. Det er lettere å endre på noe iterativt på prototypestadiet enn på produksjonsstadiet.

Man skiller vanligvis mellom prototyper som er lavnivå og høynivå. Lavnivå prototyper er enkle og kan lett endres. Eksempler er for eksempel papirprototyper der knapper og bokser kan flyttes på for å endre designet og skisser. Høynivå prototyper vil derimot ligne mer på sluttproduktet med bruk av ekte materialer, og kanskje programvare på alfa- eller betanivå som delvis fungerer.

3.1.4 Brukersentrert utvikling

Brukersentrert utvikling er en prosess der brukeren er involvert i hvert steg. Stegene er å forstå brukskonteksten, etablere krav, implementere artefakt og evaluere artefakt. Dette er en syklus som kan gjentas flere ganger. ISO (2010) standardiserer brukersentrert utvikling. Se figur 3.2 for de ulike stegene i prosessen. Brukersentrert utvikling passer godt inn i de fem stegene som inngår i «design og kreasjon»-forskningsmetoden. Prosessen fungerer best hvis den gjentas flere ganger, gjerne med en lavnivå prototype som gradvis nærmer seg mer og mer produksjonsklar.

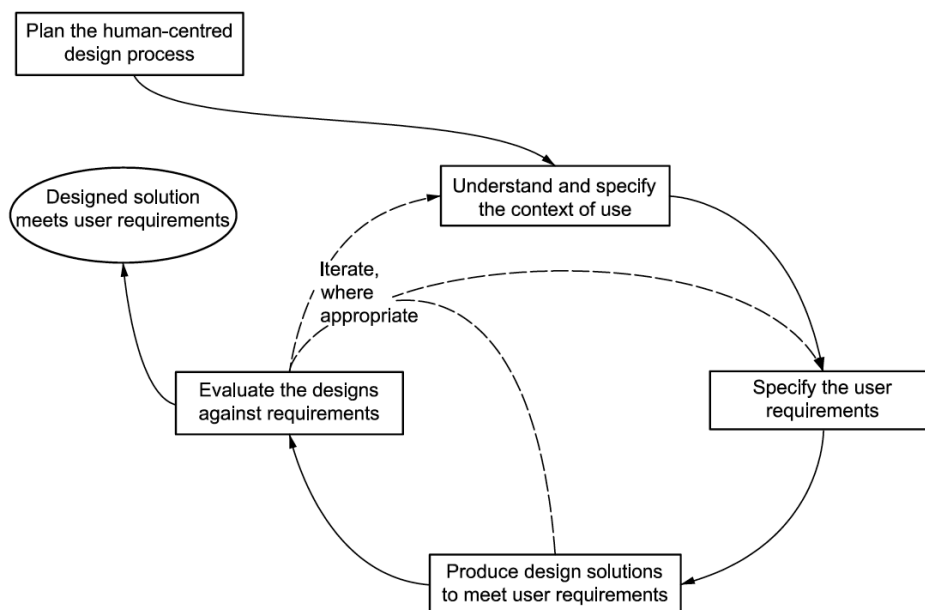
3.2 Datagenereringsmetoder

En datagenereringsmetode er måten man innhenter empirisk materiale og data i kvalitativ eller kvantitativ form (Oates, 2006, s. 36).

3.2.1 Semistrukturert intervju

Det finnes flere forskjellige typer intervju. Semistrukturert intervju er en datagenereringsmetode der noen temaer og spørsmål er forberedt på forhånd, men rekkefølgen spørsmålene stilles i kan endre seg og nye temaer og spørsmål kan komme opp basert på samtalen man har med intervjuobjektet. Det er et kompromiss mellom et strukturert intervju og et helt ustrukturert intervju (Oates, 2006, s. 188).

Det er en mye brukt metode i case-studier for å få detaljert informasjon ut av intervjuobjektet. I tillegg kan metoden brukes til å få tilbakemeldinger fra brukere på en kravspesifikasjon og en endelige prototype (Oates, 2006, s. 187).



Figur 3.2: Gjensidige avhengigheter innenfor menneskeorientert designaktiviteter (ISO, 2010).

Det er vanlig at intervjuer transkriberes og analyseres som et dokument i etterkant, spesielt hvis det er snakk om semi-eller ustrukturerte intervjuer. Her kan kildene få mulighet til å gå over svarene i intervjuet dersom de ønsker det, siden det aldri vil være mulig å transkribere noe helt likt det muntlige som ble sagt i samtalen.

3.2.2 Dokumenter

Dokumenter kan deles inn i to kategorier: dokumenter som eksisterer før forskningen og dokumenter laget underveis i forskningen. Eksempler på førstnevnte er offentlig informasjon en kommune har lagt ut på Internett, rapporter om velferdsteknologi og råd fra myndigheter. Dokumenter som lages underveis i forskningen kan for eksempel være bilder tatt i en case-studie, transkribering av intervjuer og modeller og diagrammer knyttet til en spesiell implementasjon av en artefakt (Oates, 2006, s. 233-234).

En bredere definisjon av dokumenter kan også med fordel benyttes. Det er ikke bare snakk om side på side med tekst på papir, men multimediaelementer som nettsider, spill, videoer, applikasjoner og nettforum (Oates, 2006, s. 235).

Det er to ulike tilnærminger til analysering av dokumenter: dokumenter som beholder for data og dokumenter som objekter i seg selv. Førstnevnte metode der dokumenter er entiteter som inneholder data er relevant for dette forskningsprosjektet. Med denne metoden kan en enten for eksempel telle hvor mange ganger et ord opptrer i et dokument (kvantitativt), eller så kan en kvalitativt analysere hvilke temaer som var til stede i dokumentet (kvalitativt) med tekstmarkering.

Kapittel 4

Forskningsdesign

I dette kapitlet presenteres og drøftes forskningsdesignet til studien. Som nevnt tidligere, bygger rammeverket til forskningsprosessen på boken *Researching Information Systems and Computing* (Oates, 2006).

4.1 Forskningsspørsmål

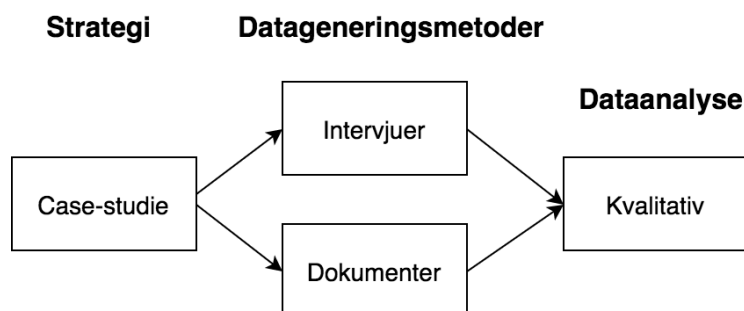
Som skrevet i kapittel 1, er formålet med denne forskningen å undersøke hvordan ny teknologi basert på IoT-skyløsninger kan brukes til avstandsoppfølging med prosjektet til Trondheim kommune som ramme: «**Hvor egnet er skybasert IoT som teknologisk plattform for avstandsoppfølging av kronisk syke?** »

Forskningsspørsmålene som ligger til grunn for denne masteroppgaven har gått igjennom en iterativ prosess i løpet av arbeidet med oppgaven. Det vil si at intervjuene gjort med Trondheim kommune og arbeidet med den tekniske løsningen har påvirket det prosjektet ønsker å undersøke nærmere. Alle forskningsstrategiene utdypes nærmere i delkapittel 4.2.

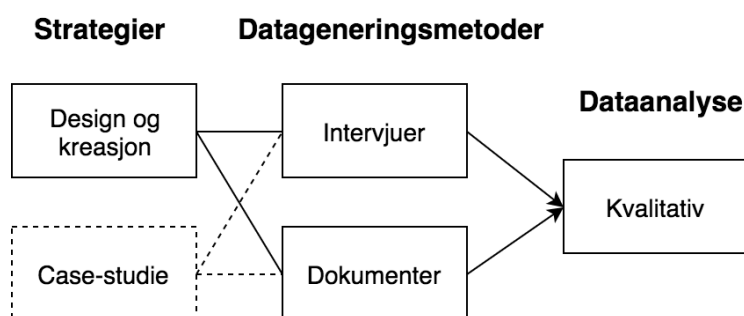
4.1.1 Forskningsspørsmål 1

Hvordan gjøres avstandsoppfølging av kronisk syke i prosjektet HelsaMi+ i Trondheim, og hvilke utfordringer ser prosjektledelsen når det gjelder denne type teknologi/tjenester?

Avstandsoppfølging er ganske nytt i Norge, og det foreligger ikke så mye informasjon om hvordan det gjøres ettersom det i skrivende stund fortsatt er i testfasen i mange kommuner. Trondheim kommune vil bli brukt som et eksempel på hvordan avstandsoppfølging gjøres i dag og hvilke planer de har for ny teknologi i tiden som kommer. Forskningsprosjektet vil finne ut hvilke utfordringer kommunen har med avstandsoppfølging i dag og hvilke erfaringer de har gjort seg hittil.



Figur 4.1: Forskningsstrategien til FS1



Figur 4.2: Forskningsstrategien til FS2

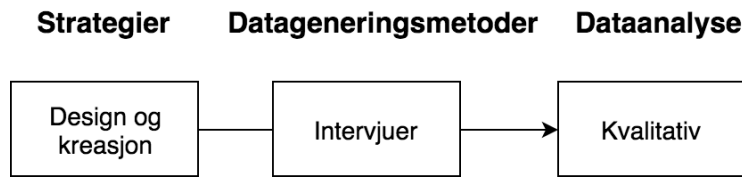
For å belyse spørsmålet er forskningsløpet som i figur 4.1. Case-studie er hovedstrategien, og intervjuer og dokumenter er datagenereringsmetodene. Intervjuene transkriberes der det er relevant og analyseres basert på temaer.

4.1.2 Forskningsspørsmål 2

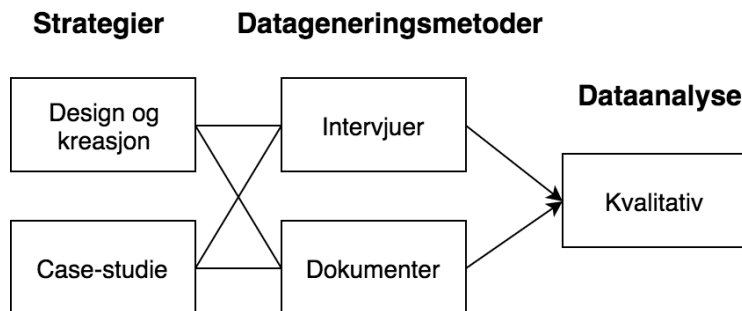
Hva er en mulig arkitektur for realisering av skybasert IoT for avstandsoppfølging av kronisk syke som adresserer utfordringene fra FS1?

Hovedstrategien er design og kreasjon, med intervjuer og dokumenter som datagenereringsmetode (figur 4.2). Ved å lage et *proof of concept* basert på eksisterende skyteknologi, er målet å finne en mulig arkitektur for teknologien. Case-studien av Trondheim kommune hjelper til med å forstå brukskonteksten og kravene til en slik arkitektur og om denne arkitekturen kan hjelpe kommunen med de utfordringene som blir belyst i case-studien.

Arkitektur er definert som strukturen til klientsiden av applikasjonen, hvordan denne fungerer og snakker med sensorene rundt – og på hvilken måte sensordataene sendes fra klienten og behandles av en bakenforliggende serverløsning. Hele dette samspillet danner en mulig arkitektur for avstandsoppfølging.



Figur 4.3: Forskningsstrategien til FS3



Figur 4.4: Forskningsstrategien til FS4

4.1.3 Forskningsspørsmål 3

Hvordan vurderer domeneekspertene innen velferdsteknologi frittstående skybaserte IoT-løsninger for avstandsoppfølging av kronisk syke?

Forskningsstrategien er å lage en frittstående løsning i form av et skytilkoblet pulsoksymeter og evaluere denne løsningen med semistrukturerte intervjuer (figur 4.3).

Frittstående betyr at løsningen er helt plattformuavhengig og ikke basert på eksisterende plattformer som Android og iOS. Domeneekspertene er definert som forskere, prosjektledere og ansatte innenfor velferdsteknologi og avstandsoppfølging.

4.1.4 Forskningsspørsmål 4

Basert på funnene fra FS1, FS2 og FS3, hva er implikasjonene for utviklingen av skybasert IoT som teknologisk plattform for avstandsoppfølging av kronisk syke?

Dette forskningsspørsmålet sammenstiller all data fra de andre forskningsspørsmålene. Det søker å finne ut hva implikasjonene er for utviklingen av skybasert IoT for avstandsoppfølging gjennom en kvalitativ analyse og drøfting. Figur 4.4 oppsummerer hele forskningsløpet.

4.2 Forskningsstrategier

Dette delkapittelet drøfter og beskriver kort hva som skal skje i de forskningsstrategiene som er valgt.

4.2.1 Design og kreasjon av et pulsoksymeter

Artefakten i dette prosjektet er en høynivå prototype i form av et proof of concept, både på hardware- og softwaresiden. Siden teknologiutforskning er et mål, vil prototypen implementeres på et ganske høyt nivå, og ikke være en lavnivå papirprototype. Prototypen vil sammenlignes med den eksisterende løsningen og vises fram til prosjektledere, ansatte og forskere innen velferdsteknologi.

Den industrielle utformingen av prototypen vektlegges ikke. Det er ikke meningen at prototypen skal ut i produksjon slik som den er designet. En annen tilnærming kunne vært å vektlegge brukerperspektivet enda mer, og gjennomføre flere runder med brukertester med helsepersonell og pasienter. Brukertester med pasienter ville medført krav om riktig behandling av sensitive helseopplysninger og en godkjent søknad til Norsk senter for forskningsdata (NSD).

Brukersentrert utvikling er metodikken som brukes for design og kreasjon av prototypen. Filosofien bak å inkludere brukeren i hvert aspekt av utviklingen er å fortere finne ut hva som ikke fungerer slik at en ikke maler seg inn i et hjørne. Case-studien av avstandsoppfølging i Trondheim kommune hjelper til med å forstå konteksten pulsoksymeteret skal brukes i og hvordan det skal brukes, noe som fører til at utviklingen av en prototype blir enklere.

Dette er design og kreasjon-strategien oppsummert:

Bevistgjøring Semistrukturert intervju med Trondheim kommune om velferdsteknologi og avstandsoppfølging. Kort bakgrunnkapittel om velferdsteknologi og avstandsoppfølging (kapittel 2).

Forslag En frittstående og skytilkoblet velferdsteknologiløsning der sensorene er innebygget vil fungere bedre eller like bra som eksisterende løsninger med nettbrett. Å basere seg på AWS IoT som skyplattform vil gjøre utviklingen enklere.

Utvikling En prototype av et skytilkoblet pulsoksymeter med fingeravtrykksensor som autentiseringsmetode.

Evaluering Semistrukturerte intervjuer med Trondheim kommune og SINTEF der prototypen vises fram.

Konklusjon Baseres på alle de foregående stegene.

Hvordan evalueringen av prototypen med semistrukturerte intervjuer ble gjennomført er beskrevet i kapittel 9.

4.2.2 Case-studie: Avstandsoppfølging i Trondheim kommune

Det er flere tilnærminger til avstandsoppfølging, og det er flere kommuner enn Trondheim som har pågående forsøk, blant annet Sarpsborg og Oslo. Fordelen med å velge ut én kommune er at det belyser hvordan de har tilnærmet seg avstandsoppfølging som helhet og forenkler datainnhenting – det er naturlig å velge ut Trondheim kommune siden det gjør at tilgangen på personer er bedre.

Målet med case-studien er å skjønne hvilke problemstillinger Trondheim kommune prøver å løse, hvordan de løser dem og hvilke erfaringer de har gjort seg hittil. Dette vil

påvirke den nye teknologien som bygges i dette forskningsprosjektet. Case-studien vil gi en oversikt over tjenesten fra A til Å, fra hvordan brukerne blir rekruttert til hvordan de bruker løsningen hjemme.

Datagenereringsmetoder i dette case-studiet er semistrukturerte intervjuer og dokumenter. Observasjon av hvordan pasientene bruker Trondheim kommunes løsning kunne vært en god metode, men for å snevre inn omfanget og slippe å innhente tillatelser velges ikke denne metoden. Det hadde vært mer aktuelt om case-studiet hadde vært den eneste hovedstrategien i prosjektet. Eksempler på dokumenter er all den informasjonen som finnes om avstandsoppfølging på Internett og det Trondheim kommune har lagt ut selv. Helsedirektoratet og Datatilsynet har også en rekke retningslinjer og råd som kommunene må og kan forholde seg til.

Hvordan de ulike intervjuene ble gjennomført og analysert er beskrevet i starten av kapittel 5.

4.3 Forskningsparadigme og forskningskvalitet

Forskningsprosjektet inneholder kun kvalitative datagenereringsmetoder, og er dermed en form for kvalitativ forskning. Det tar en interpretivistisk tilnærming til det som skal undersøkes. Det er flere måter å gjøre avstandsoppfølging på, og det er ikke sikkert at en metode er bedre enn en annen. Siden kun kvalitativ data analyseres, vil nødvendigvis forskerens subjektive holdninger påvirke drøftingen av forskningsspørsmålene.

Validitet og objektivitet må forstås på en annen måte i dette prosjektet enn om dette for eksempel hadde vært et kvantitativt eksperiment. Lincoln og Guba foreslår, i følge Oates (2006), at kriteriene for interpretivisme heller skal være hvor stor tillit man kan ha til forskningen, gjennom begrepene bekreftbarhet, pålitelighet, troverdighet og overførbarhet. Her handler bekreftbarhet og pålitelighet om hvorvidt studiet er godt nok beskrevet og dokumentert til at man skjønner hvordan det har foregått og kan stole på forskningsløpet. Forskning blir mer troverdig med flere forskjellige kilder til data og om man sjekker med kildene om man har tolket data riktig. Dette diskuteres i delkapittel 10.2.

Kapittel 5

Case-studie: Avstandsoppfølging i Trondheim kommune

5.1 Innsamling av data

Forskningsprosjektet utformet en formell henvendelse om samarbeid til velferdsteknologiprogrammet i Trondheim kommune. Se tillegg B for hele henvendelsen. I henvendelsen spurte prosjektet om å få et intervju med en som jobber med velferdsteknologi i Trondheim (heretter kalt N.N.), i mars (maks to timer), og et intervju i april med tilbakemeldinger på den tekniske løsningen (maks to timer). Hvis mulig ble det også ytret et ønske om å få til en fokusgruppe med fire-fem frikjøpte helsearbeidere. Kontakten med Trondheim kommune ble initiert av SINTEF i Trondheim.

Materialet i dette kapitlet er hovedsaklig basert på et gruppeintervju gjort med N.N. og Ingar Børre Sandvik med besøk av en fastlege som har pasienter med i prosjektet. Dette intervjuet ble gjennomført 24. mars 2017 fra kl. 09.00 til kl. 11.20 i et lokale Trondheim kommune disponerer. Intervjuhendelsen er vedlagt i tillegg D. Temaene og de forhåndsskrevne spørsmålene til intervjuet ligger i vedlegg E. Noe materiale fra evalueringintervjuene gjort med SINTEF og Trondheim kommune i mai er også med. Hvordan disse intervjuene ble gjennomført er nøyere beskrevet i kapittel 9.

Trondheim kommune ga forskningsprosjektet en bruker til testmiljøet til løsningen de kjører. De viste også fram nettbrettet som brukerne får utdelt. Dette, og informasjonen som allerede ligger tilgjengelig på nettet er en del av dokumentdelen som analyseres. Teknologileverandøren Imatis ble invitert til et videointervju for å snakke om den tekniske løsningen og velferdsteknologi, men de svarte ikke på henvendelsen. Henvendelsen er i vedlegg D.

De viktigste og mest relevante delene av intervjuet gjort 24. mars ble transkribert og analysert. Åtte hovedtemaer ble identifisert:

Bakgrunn for HelsaMi+

Hvorfor Trondheim kommune? Hvem finansierer prosjektet? Hvordan jobber Trondheim kommune med velferdsteknologi?

Om HelsaMi+ per i dag	Hvor mange brukere? Pris på løsningen? Hvor mange bruker sensorer?
Tjenesteforløp i HelsaMi+	Hvordan får brukerne tilbud om HelsaMi+ og hva skjer da? Hvordan blir HelsaMi+ brukt?
Motivasjon for HelsaMi+	Hvorfor avstandsoppfølging? Hvilke fordeler kan det gi for kommunen?
Politikk og samhandling	Hva er strategien til Trondheim kommune på velferdsteknologiområdet? Hvordan påvirker denne de valgene som er tatt i HelsaMi+? Hvilke utfordringer har kommunen i samarbeidet med andre aktører som fastleger og sykehusene?
Juridiske utfordringer	Hvordan blir dataene lagret? Har man nytte av dataen? Hvordan ser man på den nye personvernsforordningen som kommer neste år? Hvilke juridiske utfordringer har man generelt i velferdsteknologi?
Tekniske utfordringer	Hvordan fungerer nettbrett og sensorer? Hvilke problemer har man hatt der? Hva ønsker man eventuelt å endre på?
Brukeropplevelse og sikkerhet	Er kommunen fornøyd med brukergrensesnittet? Hvilke tanker har kommunen om sikkerhet og tofaktorautentisering med de brukerne de har?
Tilbakemeldinger og planer	Hva viser de første resultatene? Hva mener fastleger om sensorer og sensordata? Hva ønsker kommunen å endre på? Hva vil skje etter prosjektperioden? Er det noe kommunen ser at de mangler i dagens løsning? Hvordan ser de på velferdsteknologiområdet fremover?

Hver av disse fikk en fargekode, og relevante deler av teksten ble markert med riktig fargekode. De neste delkapitlene følger til en viss grad denne tematikken.

5.2 HelsaMi+: Bakgrunn og tjenesteforløp

Trondheim Kommune har fått i oppdrag fra Helsedirektoratet å delta i nasjonal utprøving av medisinsk avstandsoppfølging av personer med kroniske sykdommer (kols og hjertesvikt). Prosjektet gjennomføres i samarbeid med Klæbu, Malvik, Melhus og Midtre Gauldal, St. Olavs hospital, Helse Midt-Norge, HEMIT og teknologileverandøren Imatis.

Hensikten med prosjektet er å utvikle og evaluere tjenester med tilhørende velferdsteknologi for behandling og oppfølging av personer med kronisk sykdom mens de bor hjemme (Trondheim kommune, 2017).

I følge I. B. Sandvik (personlig kommunikasjon, 24. mars, 2017), ble Trondheim kommune valgt ut til å delta i avstandsoppfølgingsprosjektet hovedsaklig fordi de hadde et

prosjekt tidligere som het *HelsaMi* der ti KOLS-pasienter ble fulgt opp hjemme. I dette prosjektet var det ikke noen sensorer involvert, kun mulighet til å rapportere dagsform med et nettbrett. Nåværende avstandsoppfølgingprosjekt er en fortsettelse av det arbeidet og kalles for *HelsaMi+*. Prosjektet startet i 2015 og går fram til 2018. I dette prosjektet har noen av brukerne også sensorer for vekt, blodtrykk og pulsoksymetri.

Kriteriene for å få tilbud om uttestingen er å ha KOLS eller hjertesvikt, være bosatt i en av kommunene som har tilbudet og ikke være så kognitivt svekket at man ikke har nytte av teknologien. Det starter som oftest med en henvendelse fra fastlegen eller fra legespesialist på St. Olavs til Helse- og velferdskontoret i kommunen som deretter henviser videre til *HelsaMi+*. Noen henvisninger kommer også direkte til *HelsaMi+* eller fra samarbeidskommuner. *HelsaMi+* vurderer om det er en aktuell bruker og tar en telefonsamtale til brukeren. Om brukeren fortsatt er interessert, avtaler *HelsaMi+* et møte hos fastlegen der det blir utarbeidet en egenbehandlingsplan i samråd med bruker, en representant fra *HelsaMi+* og fastlegen. I etterkant av møtet avtales det et nytt tidspunkt for hjemmebesøk der man får utstyr og opplæring i bruk av utstyret (N.N., 24. mars, personlig kommunikasjon).

Brukeren sender inn dagsformen ved å svare på spørsmål i nettbrettet. «Ved innrapportering av en forverring vil brukerne bli kontaktet av helsepersonell i kommunens vaktentral, som vil bistå brukeren i å gjøre de mest korrekte valgene basert på egenbehandlingsplanen» (Trondheim kommune, 2017). Brukeren får ikke beskjed i nettbrettet om at det er på tide å melde inn dagsform eller foreta en sensormåling, men N.N. sier at mange har det som en del av morgenrutinen. Selv om brukerne med sensorer ikke trenger å gjøre målinger hver dag, er det flere av brukerne som gjør det hver dag allikevel slik at de ikke skal glemme det. Minstekravet i tjenesten er to ganger i uken.

Per 24. mars er det 65 brukere i løsningen og seks av disse har utplassert sensorer. Trondheim kommune har fått et ønsket måltall fra ehelse-direktoratet på 150 til 200 brukere i løpet av året. Per 2. mai var det 70 brukere, og kommunen kommer ikke til å øke antall brukere med sensorer. Prisen på nettbrettet brukerne får utlevert er rundt 3000 kr. Videre koster pulsoksymeter rundt 2000 kr, mens vekt og blodtrykksapparat koster rundt 1000 kr hver. Det gir i utgangspunktet en total sum på ca. 7000 kr for et nettbrett med tre sensorer. Kommunen eier SIM-kortet som står i nettbrettet og betaler for all datatrafikken. I. B. Sandvik sa at det for kommunen er en ganske stor administrativ byrde å eie masse SIM-kort, og at det er et moment som må være med i kostnadsvurderingen av denne type tjenester. Det er Direktoratet for e-helse for som anbefaler at kommunen bør eie hele verdikjeden, i dette tilfellet nettbrett og SIM-kort.

5.3 Motivasjon for *HelsaMi+*

Et moment er det målbare med tjenesten, nemlig i hvor stor grad man kan foregripe og redusere innleggelser på sykehus. Et annet moment er om man kan øke trygghet- og mestringsfølelsen til brukerne. N.N. sa at de ønsker at *HelsaMi+* skal være et lavthengende kontaktpunkt. Mange tar kontakt med legen altfor sent og det fører ofte til en innleggelse. Erfaringene viser at brukerne ikke forbinder et nettbrett med en helsetjeneste: «De føler at kommunen er i kulissen, men det er lite inngripende i hverdagen».

Det kan gjøre at de mestrer hverdagen bedre gjennom den følte tryggheten. Om fastlegene sier N.N. dette:

Fastlegen sier litt forskjellig om hva de tenker og mener om det her. Men de aller fleste sier at de ser de kroniske syke for sjeldent. De er ikke flinke nok til å bestille time – det er en ting som er spilt inn veldig. Vi driver og skal videreutvikle den egenbehandlingsplanen nå. Og et av innspillene der er at legene ønsker å ha inn 'husk å ta de årlige vaksinene og bestill time til det'.

5.4 Politikk og strategi i HelsaMi+

Trondheim kommune drifter de fleste tjenestene som for eksempel hjemmebesøk, helsevakta, avstandsoppfølging og responscenter med egne ressurser: «Trondheim kommune har jo en veldig tydelig politisk ledelse og forankring på å ha mesteparten av ansvaret for sin egne tjenester selv, altså inhouse» (I. B. Sandvik, 24. mars, personlig kommunikasjon). Dette skiller seg fra Oslo og Sarpsborg som har større grad av outsourcing av tjenestene. Han påpeker også at Trondheim kommune til dels skiller seg fra andre kommuner ved å tenke på robuste tjenester på en helhetlig måte, og ikke bare på teknologien:

For det som Trondheim kommune er veldig tydelige på, og kanskje til dels skiller oss fra andre kommuner når det gjelder velferdsteknologi – det er at det er ikke teknikken og dingsene som på en måte skal løse alt. Det er tjenesten vi skal bygge som understøttes av teknologien. Vi ser jo det at det å kunne pilotere i småskala, med ny teknologi, det greier de fleste. Men det å bygge en robust tjeneste som skal implementeres i en stor kommune – det er noe helt annet. Og som skal da kunne rulles ut i storskala med mange hundre brukere av ulike tjenester.

(...) Det er kanskje noe av det som er gjennomgangsmelodien av de kommunene som har vært med i det nasjonale programmet gjennom år nå, på å teste ut og pilotert ulike type tjenester innenfor velferdsteknologi. Det og så ta det fra en pilot der du har noen prosjektressurser som kan sitte og følge opp og ta kontrollen og greie å holde styr på et gitt antall brukere, til å faktisk få det inn som en tjeneste som er implementert i organisasjonen som kan rulles ut og stå på egne bein. Det er det som er jobben. Og der har Trondheim kommune hatt en veldig klar filosofi at, til dels, hva skal vi si, frustrasjon for mange andre som synes det går seint før det kommer i gang og får opp antall brukere og alt dette. Så det er en veldig viktig dimensjon å ha med seg at det er ikke bare å bestille utstyr og så er man i gang. (I. B. Sandvik, 24. mars, personlig kommunikasjon)



Figur 5.1: HelsaMi+: Hovedskjerm

5.5 Demonstrasjon av HelsaMi+

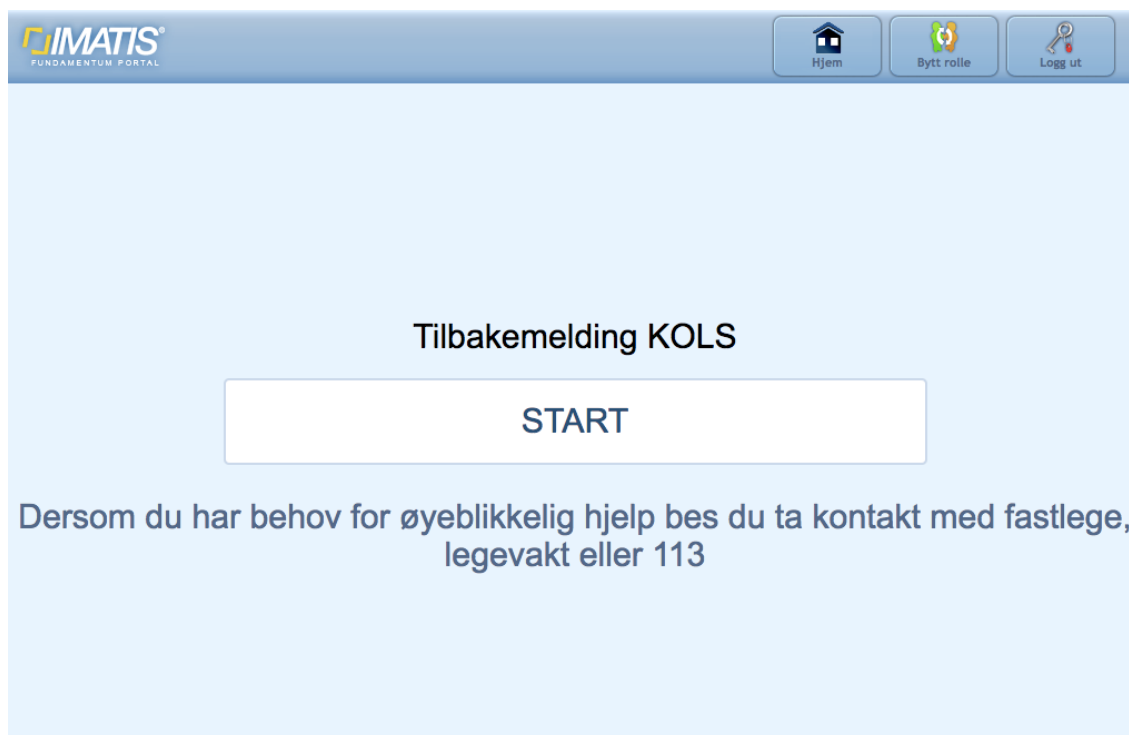
Brukeren har et nettbrett som kjører en androidapplikasjon. Denne applikasjonen pakker inn et webgrensesnitt. I tillegg til hovedapplikasjonen er det en sensorhub-applikasjon som kjører i bakgrunnen og har ansvaret for tilkoblingen til alle sensorene. Alle skjermbildene i dette delkapittelet er tatt i testmiljøet til Trondheim kommune i en nettleser.

Beskrivelsen av løsningen og skjermbildene tar utgangspunkt i to scenarioer for sluttbrukere og ett scenario for helsepersonell: for brukere er det å rapportere dagsform og utføre en måling med pulsoksymeter, og for helsepersonell er det å gå inn for å se på hvilken data som er kommet inn.

Det første som møter brukeren er hovedskjermen i figur 5.1. I testmodus er alle de ulike alternativene vist. Herfra kan brukeren trykke seg inn for å rapportere dagsform for *KOLS* og *HJERTESVIKT*, eller utføre en sensormåling med enten vekt, blodtrykk eller pulsoksimeter. Det er også mulig å gå på *Brukerveiledninger* for å få instruksjoner og hjelp. For en bruker som har *KOLS* og en sensor utplassert hjemme, vil skjermen for eksempel vise *KOLS* og *SpO2*.

5.5.1 Rapportere dagsform

Scenariet er at en bruker med *KOLS* skal rapportere dagsform. Brukeren trykker på *KOLS* på hovedskjermen, og kommer til figur 5.2. Et klikk på *Start* tar brukeren til 5.3.



Figur 5.2: HelsaMi+: Tilbakemelding KOLS

Det er fire spørsmål brukeren svarer på. Alle svaralternativer som er subjektive spør brukeren om å sammenligne med hvordan det er til vanlig.

- Hvordan er pusten din?
- Hvordan er hosten?
- Hvordan er fargen på oppspyttet ditt?
- Hvordan føler du deg til sinns?

Etter spørsmålene får brukeren mulighet til å legge inn en skriftlig kommentar og se over at det er greit før alt sendes inn (figur 5.4). Brukeren får beskjed om at tilbakemeldingen er sendt, og må selv aktivt klikke seg tilbake til menyen etterpå.

5.5.2 Utføre en måling med pulsoksimeter

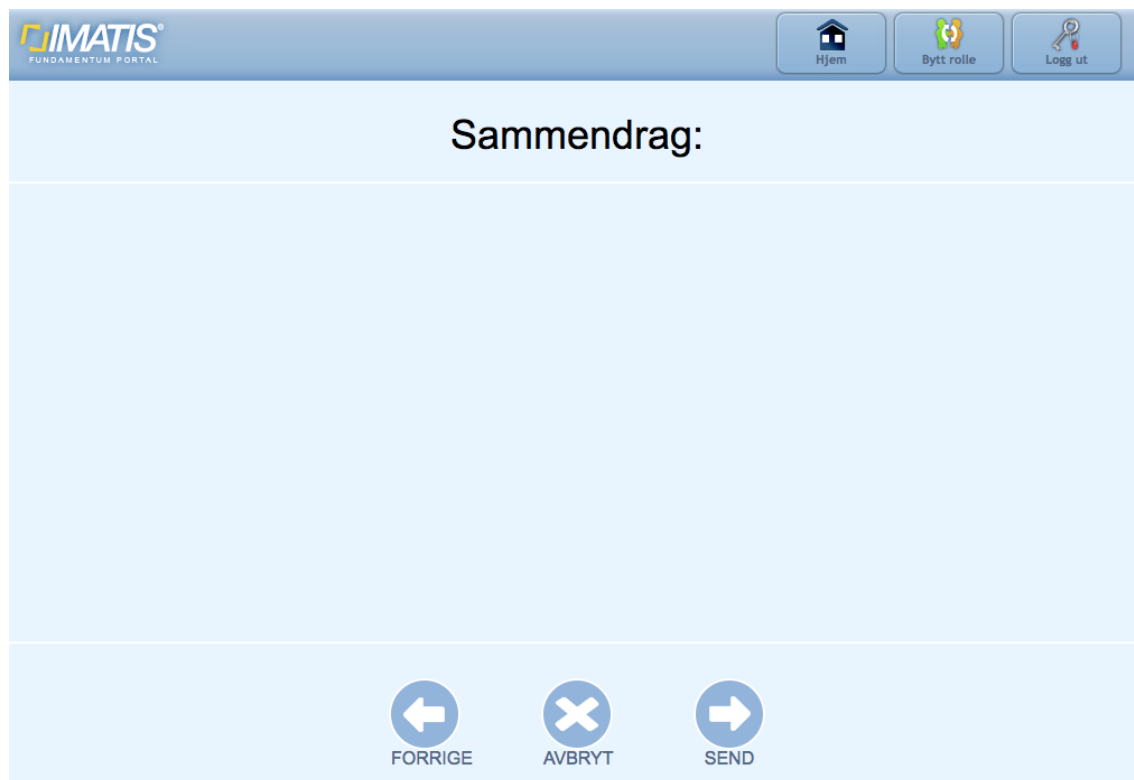
Et klikk på *Test-SpO2* tar brukeren til en mellomskjerm med informasjon om at brukeren ikke har utført en måling ennå med et bilde av sensoren. Brukeren klikker seg videre til en skjerm med to valg, enten *Trykk her for å måle oksygenmetning og puls* eller *Trykk her for brukerveiledning* (figur 5.5). Det første valget tar brukeren til figur 5.7 der brukeren klikke på en knapp og setter måleren på fingeren. Brukeren må ha en gyldig sensorhub installert for å få lov til å gjennomføre en måling.

5.5.3 Se på innrapportert data

Trykk på *HelsaMi*. Figur 5.8 viser hvordan oversikten over innrapportert data ser ut.



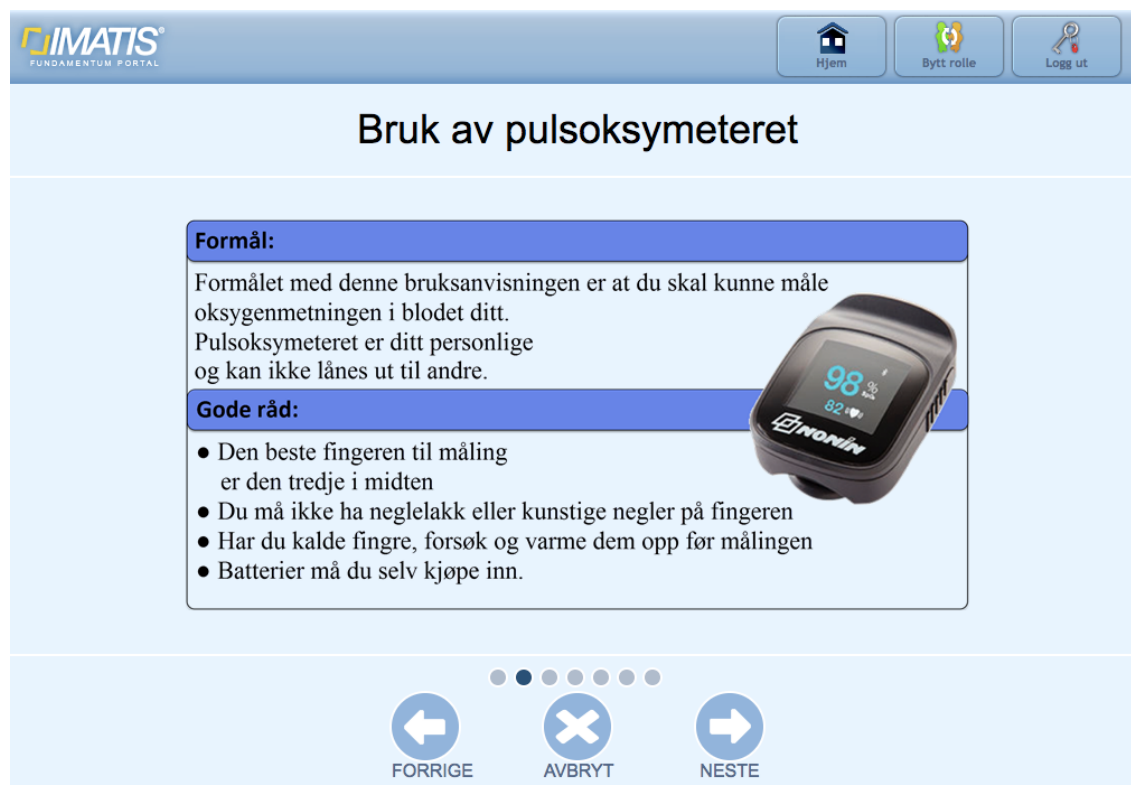
Figur 5.3: HelseMi+: Spørsmål 1



Figur 5.4: HelseMi+: Sammendrag



Figur 5.5: HelsaMi+: Pulsoksimeter-oversikt



Figur 5.6: HelsaMi+: Pulsoksimeter-veiledning



Figur 5.7: HelseMi+: Pulsoksimeter-måleskjerm

HI...	B...	ALDER	ALARM	A...	SENSOR	REG	MOTTATT	DAGSFORM	PUST	HOSTE	SP...	M...	KOMM...	HJERTEMOTTATT	P...	T...	F...
	...	58															
	...	20					12.12 10:05										
	I...	90					22.12 15:34							23.12 10:29			
	...	85					22.12 08:50							21.12 22:53			
	...	39					30.01 14:10							30.01 12:57			
	...	70					04.01 09:15							04.01 09:29			
	...	37		...			27.04 08:12				BL...		Test	26.04 10:58			
	...	24					14.03 07:58				GU...			14.03 07:58			
	...	4+6					09.12 10:45							17.01 14:24			

Andreas (Andreas) 14 av 14 synlige rader. Data sist endret noen sekunder siden

Figur 5.8: HelseMi+: Oversikt for administrator 1/2

5.6 Utfordringer og refleksjoner fra prosjektledelsen

5.6.1 Samhandling

I. B. Sandvik (personlig kommunikasjon, 23. mars, 2017) påpekte at den største utfordringen med prosjektet ikke er teknologien, men samhandlingen mellom de ulike helseinstansene:

Det som er interessant med denne uttestingen av tjenesten er jo å se hvilke gevinster, for det er jo gevinster man ønsker å kartlegge, hvilke gevinster man tror man kan få med å drive med den type ny helsetjeneste med oppfølging av brukere mens de bor hjemme. Og det som gjør prosjektet her så relativt kompleks – en ting er jo den teknologiske delen – det er jo stort sett håndterbart. Dette er jo et prosjekt som går på tvers av kommune, fastlege og spesialisthelsetjeneste. Det vil si midt inne i samhandlingsreformdomenet.

De ulike helseinstansene har forskjellige former for måloppnåelse og egne budsjetter. De statlige helseforetakene har ansvaret for sykehusene, ikke kommunen. I. B. Sandvik forklarte hvorfor dette kan være en utfordring:

Det er jo en utfordring av mange grunner. Både på grunn av kultur, fokus. Spesialister er jo diagnoserettet, kommunen har jo det hele mennesket som fokus. Og fastlegen sitter jo der i mellom og skal egentlig ha ansvaret og kontroll over alt som foregår over sine pasienter mens de er utenfor sykehuset sitt domene. Så det er mange interessante problemstillinger som har blitt avdekket. Blant annet det vi kan nevne et par eksempler det med det juridiske: hvem skal ha ansvaret for hva, hvordan skal opplysninger deles mellom partene – ikke rett frem. Det andre er dette med finansieringsmodellene. Det er jo ikke noe tvil om at den største investeringen i denne type tjeneste gjøres av kommunen, når det gjelder å rigge tjeneste, oppfølging, utstyr, alt dette her. Mens gevinstene kanskje i større grad tas ut på sykehuset, med færre innleggelseser.

5.6.2 Juridisk

I. B. Sandvik reflekterte også rundt den juridiske problematikken med avstandsoppfølging relatert til personvernet:

(...) og en annen dimensjon også der som går på det juridiske for at – et spørsmål rundt dette med egenmålinger og rapportering er 'hva trenger på en måte kommunen å vite'. I teorien så trenger vi egentlig bare å vite det som gir et grunnlag til en vurdering og tiltak. Og der er jo den juridiske vurderingen 'hvem skal eie og sitte på dataen' og per i dag så får vi jo alt innrapportert og har tilgang på all informasjon i den her løsningen.

Konklusjonen per i dag fra Trondheim kommune sin side er at så lenge fastlegen vurderer at dataen som kommer inn er relevant, så er det innenfor regelverket. Utgangs-

punktet nå er at alle som får egenbehandlingsplan av fastlegen er vurdert som relevante for å sende inn måledata. I samtalen med I. B. Sandvik kom man inn på at det kun er målinger som blir identifisert som i faresonen (røde målinger) som skal føre til et tiltak fra kommunen sin side, mens grønne målinger er greie. Skal man da lagre de grønne målingene? Spørsmålet er om de grønne målingene kan være nyttige i en trendanalyse av dataene. I dag blir alle målinger lagret av kommunen i løsningen.

I. B. Sandvik sa at det er teknologisk forbedringspotensiale på at brukeren skal finne ut hvilken data som er samlet inn og styre sin egen data på en bedre måte. Dette er ikke implementert i løsningen til Trondheim kommune i dag, men blir enda mer relevant med ny personvernslov.

5.6.3 Sensorer, måledata og egenrapportering

N.N. påpekte at sensorer ikke passer for alle brukere. For noen kan ta det helt over hverdagen og bli usunt:

Sensorer kan skape veldig mye angst og utrygghet. Mange kan la sensorene bestemme hvordan dagen min er. Det er oksygenmetningen, eller vekta, eller blodtrykket som bestemmer om den dagen er bra eller ikke. Og de luker vi ut og tar fra sensorene. (...) det gagnar ingen da. Det er (...) du selv som skal sette hverdagen din og ikke data fra en sensor. (...) de kan lett sykeliggjøre seg selv da. (...) få en sånn avhengighet. (...) det er ikke alle som har godt av og tåler å motta å ha all den informasjonen om seg selv. Det kan bli for mye.

Fastlegene har i dag ikke automatisk tilgang på målingene fra sensorene. Det er også spørsmål knyttet til hva man kan trekke ut av den måledataen som kommer inn. En fastlege sa at det er usikkert hva det betyr når måleinstrumentene brukes hjemme. På sykehusene har man kontrollerte former og muligheten til å gjøre randomiserte forsøk. Hjemme blir det ikke det samme:

Det er klart at er det (måledataene red.anm.) forskrekkelig lavt så ser vi nå at personen er døende. Men ellers så vet vi jo strengt tatt ikke hvordan variasjonene på enkelte ting skjer i hverdagen. Vi vet hvordan det er når folk kommer syke i ambulansen, eller når de ligger under observasjon på sykehus. Men vi vet ikke hvordan det er når de vanligvis fra kjøkkenet til toalettet setter seg ned, tar seg en pause.

Legen sa at det ikke var så lenge siden at kontinuerlig måling av blodtrykk over 24 timer hjemme ble akseptert som klinisk relevant for blodtrykksdiagnose. Når legen ikke kan se pasienten i virkeligheten, mister en også flere aspekter som kan brukes til å vurdere pasienten:

(...) når de er på sykehus, så er de jo per definisjon syke. Og da har målingene en annen betydning. Vi går jo glipp av ansiktsfarge, vi går glipp av angstuttrykk, vi går glipp av stemmeleie – alle de her indikatorene som intuitivt sier oss noen ting om hvordan ting er, og forholde det til målingene.

Det viser seg også at brukerne tolker den subjektive vurderingen av egen helsetilstand forskjellig. Noen svarer konsekvent at de er dårlige selv om de har gitt inntrykk til fastlegen at de klarer seg greit. Fastlegen sa at dette er signaler som må tolkes forskjellig utifra hvilken bruker det er snakk om. Det er noen indikatorer som man må lære seg å kjenne. N.N. sa at det spørsmålene egentlig handler om, er om brukeren føler seg verre enn normalt. Det normale kan være at man er dårlig hele tiden for en rekke brukere.

5.6.4 Brukeropplevelse, sikkerhet og teknologi

I. B. Sandvik sa at programvaren kommunen har kjøpt inn til HelsaMi+ er basert på modifisert hylleware fra teknologileverandøren Imatis. Den opprinnelige programvaren var ikke beregnet på avstandsoppfølging, men på sykehus og tavlevisning. «De har på en måte gitt oss en verktøyskasse da. Og så er det vi selv som har programmert opp og bygd opp driftsskjerm da».

Hva hadde du endret på i den tekniske løsningen i dag? Hva ønsker dere (...) å utvikle i framtiden på (...) den tekniske siden?

Det er jo ikke tvil om det at det er brukergrensesnittet mot sluttbrukeren og at sluttbrukeren i større grad kan få tilgang og kontroll og velge egen grafisk visning og data. Det er det som må forbedres heftig. Og så er det helt sikkert forbedringer som kan gjøres på den delen som vi sitter med (...) på vaktcentralen da.

Videre sa han at det merkes godt i nettbrettsløsningen at det ikke er en native app. Det vil si at den ikke kjører direkte på operativsystemet til enheten, men at en nettside er pakket sammen som en applikasjon. Det kan føre til at applikasjonen føles treig. Brukergrensesnittet er litt låst til hvordan Imatis har valgt å bygge opp applikasjonen.

Kommunen jobbet med å få til sertifikater knyttet til brukeren: «(...) vi skal nå ut med sertifikater. Unike sertifikater på hvert nettbrett som er kobla med brukeren. Slik at ikke et nettbrett kommer på avveie, og så kan noen tilfeldigvis greie å logge seg på, sant». Dette var ikke til stede i løsningen per mars 2017. Brukeren må logge inn med brukernavn og passord som lagres på enheten slik at de slipper å skrive det inn på nytt hver gang de skal logge inn. Passordet lagres ikke i klartekst. Kommunen arbeidet med denne problematikken parallelt med å få applikasjonen inn i Norsk helsenett, et eget lukket nett for helsevesenet. En annen del av oppdraget var å finne løsninger som kan øke sikkerheten, men som ikke er for komplisert for brukeren:

(...) vi kunne sagt at 'ja vi skal ha en tofaktorautentisering for alle brukerne som skal logge seg på'. Teknisk så hadde det vært helt problemløst å fikse det. utfordringen er at da kan man sannsynligvis si at 60-70 % brukerne – det blir for komplisert, sant. Og da vil de på en måte ikke kunne bruke den tjenesten. Og det har vi fått aksept for, men da må vi introdusere alternative sikkerhetsmekanismer som er god nok da.

Andre problemer var blant annet knyttet til valget av Android som teknologiplattform. Det ble problemer når det kom ut en ny Android-versjon – automatiske oppdateringer til operativsystemet ble dermed slått av. Ikke forstyr-funksjonalitet ble slått på for at

ikke andre elementer på nettbrettet skulle ta fokuset vekk fra HelsaMi+. Applikasjonen ligger lett tilgjengelig på startskjermen til nettbrettet. I starten var det også problemer med sensortilkoblingen, sa I. B. Sandvik, sensoren fikk tidsavbrudd. Trondheim kommune måtte jobbe tett med leverandør for å få løsningen stabil nok, og det førte til utviklingen av sensorhuben som har ansvaret for Bluetooth-tilkoblingen. Kommunen slet også med at applikasjonen åpnet masse faner i og med at det var en innpakket webapplikasjon. I mai 2017 var det fortsatt noen problemer med at sensorhuben sendte inn den eldste målingen når den først fikk kontakt og at det var noe kluss med tidsstempling av målinger.

5.7 HelsaMi+ i fremtiden

«Prosjektet vil i løpet av 2017 videreutvikle tjenesten til å rette et større fokus på forebyggende tiltak som bruker kan gjennomføre selv med bistand fra teknologien» (Trondheim kommune, 2017). Hva betyr dette?

N.N. svarte at kommunen ser på tjenesten som bokstavene A-B-C. Per i dag har kommunen bare B-en. Brukeren har avstandsoppfølging, og det blir gjennomført tiltak ved forverring. For HelsaMi+ skal være en god tjeneste fra start til slutt, og for at tjenesten skal implementeres som en del av tilbudet til kommunen, må kommunen gjøre mer på de dagene som er bra, og tenke mer helhetlig. N.N. sa at A-en handler om gode ikke-medisinske tiltak: mestringsstrategier, trening, ernæring, oppmuntring, klapp på skuldrene og motivasjon for å gjøre livstilsendringer. Dette er ikke kommunen flinke nok på i dag. C-en er å koble inn andre profesjoner som fysioterapeuter og ergoterapeuter. De kan jobbe med hverdagsrehabilitering inn mot brukeren.

A Bruker: «Hva kan jeg gjøre selv for at helsen min skal bli bedre?»

B Responssenterløsningen: Medisinsk avstandsoppfølging, HelsaMi+, sykepleiere.

C Andre instanser: hverdagsrehabilitering, fysioterapeuter og ergoterapeuter, bidra inn mot A.

Dette er noe som Oslo kommune i større grad har tilbud om i dag. Som N.N. sa, de har «mange flere ingredienser i suppa si», mens Trondheim har «veldig begrenset butikkutvalg». Vis-rapporten fra Oslo kommune viser, i følge N.N., at det er store besparelser ved tiltak som for eksempel å ha intensiv fysioterapi hjemme over en tre-fire ukers periode og utlevere medisindispensere som varsler om når det er tid for å ta en tablett. Det kan føre til at man kan ta vekk vedtaksfestet tid på hjemmesykepleie og redusere antall hjemmebesøk.

«Hva tenker dere om en løsning som på en måte ikke går via tablett? (...) hvis man skulle lage et eget system – hva tenker dere om det?»

N.N. svarte:

Ja, altså i utgangspunktet så får jeg jo litt lett angst når jeg har et nytt system. Men det er for at hverdagen til ansatte i drift i dag – det handler om enormt mange systemer, inn og ut. De må ha – sist jeg telte så var det åtte

eller ni passord daglig for å gjøre jobben sin – bare innenfor EPJ-systemet vårt, så må jeg ha elleve roller for å gjøre jobben min.

(...) og vi er veldig opptatt av at alt det nye må integreres i det vi har . I Trondheim kommune så er det vedtatt at alle systemer i forhold til oppfølging og trygghetsalarmer, lokaliseringsteknologi – alt skal integreres for TransMed 8, for det er det som er vedtatt som driftssystem (...) . Så alt skal på en måte integreres inn dit, for å sikre arbeidsflyten til ansatte oppi her.

N.N. trakk fram at for innbyggerne sin del så kan det være vel så greit med en egen boks. Men N.N. sa at nyere undersøkelser (se delkapittel 2.3) viser at eldre har god styring på teknologi og at mange bruker PC og nettbrett hjemme.

Kapittel 6

Kvalitetskrav til løsning

Kvalitetskrav er test- og målbare ikke-funksjonelle krav til et system. I følge Bass, Clements og Kazman (2013) er sikkerhet, ytelse og tilgjengelighet de viktigste kvalitetskravene for skyteknologi. Personvern er viktig for kritiske helsedata, og er noe som Datatilsynet legger stor vekt på. I tillegg til disse kvalitetskravene er interoperabilitet og brukervennlighet trukket fram som kvalitetskrav. Basert på bakgrunns materialet og case-studien, ble kvalitetskravene rangert for en frittstående skybasert velferdsteknologiløsning:

1. Sikkerhet
2. Personvern
3. Interoperabilitet
4. Tilgjengelighet
5. Ytelse
6. Brukervennlighet

Disse kvalitetskravene er beskrevet nøyere i de neste delkapitlene.

6.1 Sikkerhet

Systemet skal være beskyttet mot *man-in-the-middle-angrep* (MITM) og tyvlytting, og skal ha tilgangskontroll på alle ressurser. Alle kommuniserende enheter skal være autentisert for å unngå at data kan bli endret. Det skal være intern tilgangskontroll for utviklere, helsepersonell og tredjepartsløsninger. Når en applikasjon kjører som en tjeneste på andre sin infrastruktur, betyr det i praksis at den fysiske sikkerheten og tilgangen til data er satt bort til en tredjepartsleverandør. Det er et viktig moment å vurdere for noen som tenker på å kjøpe infrastruktur.

6.2 Personvern

Datatilsynet foreslår at alle nye velferdsteknologisystemer lages med innebygd personvern (*privacy by design*), noe som vil si at det tas hensyn til personvern i alle fasene av utviklingsløpet (Datatilsynet, 2015). Det er mye billigere enn å endre systemet for øke personvernshensynet i etterkant. Det er lov til å overføre personopplysninger til land utenfor Norge så lenge disse sikrer en forsvarlig behandling av opplysningene (Datatilsynet, 2016). I teorien gjør dette det mulig å kjøre helseapplikasjoner i AWS-regioner som er innenfor EU, for eksempel Frankfurt og Irland. Norge har allikevel pleid å sette som krav at helseopplysninger skal lagres i Norge, og det har vært en debatt i midten av 2017 om utfordringene ved å ha opplysninger lagret i Norge som det er mulig å nå fra andre land som drifter tjenesten.

6.3 Interoperabilitet

Det er forventet at andre og tidligere utviklede systemer må integreres i løsningen. Derfor trenger løsningen flere forskjellige grensesnitt som gjør det mulig å kommunisere med andre plattformer. Norge har bestemt at Continua-rammeverket skal brukes i nye velferdsteknologiløsninger.

6.4 Tilgjengelighet

I velferdsteknologi kan tilgjengelighet være et spørsmål om liv og død. Dersom den nødvendige informasjonen ikke blir gitt til den riktige personen kan liv gå tapt. En leverandør av skytjenester lover typisk en oppetid på 99,95 % i løpet av et år, noe som betyr at tjenesten er forventet å ikke være tilgjengelig rundt fire og en halv time i gjennomsnitt i løpet av et år. I noen systemer kan en sånn nedetid være helt ødeleggende. Netflix kjører hele sin platform (unntatt videostreaming som er i forskjellige *content delivery networks*) på AWS (Izrailevsky, 2016). Netflix får til en oppetid på nesten 99,99 % ved å legge til redundans og mykfeil (*graceful degradation*) på toppen av upålitelige komponenter.

6.5 Ytelse

Løsningen må kunne skalere fra noen få brukere til flere tusen brukere. Hvis én bruker sender noen meldinger hver dag, kan det bety opp mot en million meldinger i døgnet. Løsningen må håndtere mange meldinger uanstrengt og uten avbrytelse.

6.6 Brukervennlighet

Brukervennlighet i denne konteksten betyr brukervennlighet for sluttbruker, utvikler og systemadministrator. Sluttbrukeren må føle seg trygg når løsningen benyttes, og forstå hvordan sensormålingene utføres. For utvikler og systemadministrator er spørsmålene hvor god dokumentasjonen er, om det er mulig å automatisere utvikling og vedlikehold og hvordan man vedlikeholder og overvåker løsningen.

Kapittel 7

Teknologi

Dette kapitlet redegjør kort for de ulike teknologiene som er relevante for avstandsoppfølging i dette prosjektet: protokoller og metoder for kommunikasjon, skyteknologi og skyløsninger, plattformer for elektronikkprototyping og sensorer og aktuatorer.

7.1 Protokoller og kommunikasjon

Delkapitlet tar for seg *MQ Telemetry Transport* (MQTT) som kjører på *Transmission Control Protocol* (TCP), trådløsstandarden *Bluetooth Low Energy* (BLE) og sikkerhetsutfordringer med BLE og seriell kommunikasjon og *general-purpose input/output* (GPIO) for kommunikasjonsgrensesnitt mellom elektronikk.

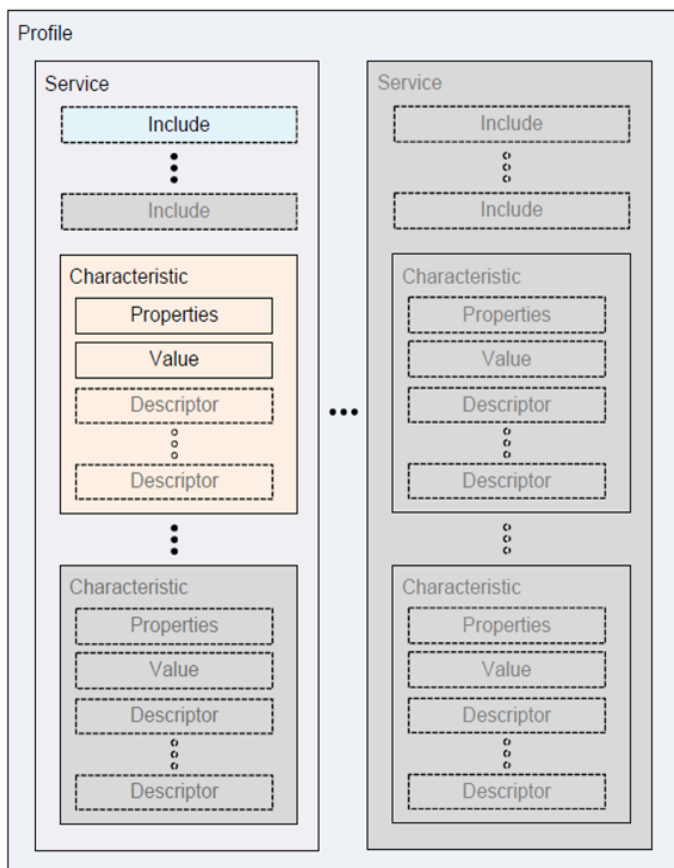
7.1.1 MQTT

MQTT er en lettvekt og meldingbasert klient-tjener-protokoll bygget for IoT-kommunikasjon og Machine to Machine (M2M) (Banks & Gupta, 2014). Den er designet for å bruke lite data og fungere bra på dårlige nettverksforbindelser. MQTT kjører direkte på TCP/IP eller over WebSockets.

Protokollen er basert på *publish-subscribe*-meldingsmønsteret. En klient kan publisere hva som helst til et emne (streng, eller liste av strenger). En annen klient som abonnerer på det samme emne vil motta datapakken fra den andre klienten. Dette gir de følgende klientmetodesignaturene for MQTT i kodesnutt 7.1:

```
1 connect(mqtt://address:port)
2 subscribe(topic: String | list of topics: String)
3 unsubscribe(topic: String | list of topics: String)
4 publish(topic: String, payload: String/binary)
5 onMessageReceived(topic: String, payload: String/binary)
6 close()
```

Kodesnutt 7.1: Klient-side-API for MQTT



Figur 7.1: Hierarki for GATT-profiler (Bluetooth.com, udatert)

MQTT-*brokieren* tar i mot og behandler innkommende klienttilkoblinger og stopp av abonnement. I tillegg mottar den meldinger og sender meldinger videre til klienter som abonnerer på et emne. MQTT kjører på port 1883, mens MQTT over *Transport Layer Security* (TLS) kjører på port 8883.

7.1.2 Bluetooth Low Energy

BLE, tidligere markedsført som Bluetooth Smart, ble en del av Bluetooth-standarden fra versjon 4. BLE er egnet for trådløs direktekommunikasjon mellom små enheter med lavt strømforbruk.

Bluetooth-standarden opererer med applikasjonsprofiler som beskriver hvordan man skal samhandle med en Bluetooth-enhet. Profilene er bygget på *generic attribute profile* (GATT), en spesifikasjon for å sende og motta små databiter (attributter) over en datalink (Bluetooth.com, udatert). En profil har typisk flere ulike tjenester, som igjen har flere ulike karakteristikk (se figur 7.1). Karakteristikkene har ulike egenskaper, attributtverdi og en databeskriver. Egenskapene definerer hvilke operasjoner som er lov til å gjøre på en attributtverdi, for eksempel lese, skrive og lytte på. Bluetooth-standarden definerer en rekke standardiserte profiler på bluetooth.com, blant annet *Pulse Oximeter Service* og *Battery Profile*.

Sikkerhet i BLE

De største sikkerhetsproblemene til BLE generelt er passiv tyvlytting, MITM og identitetssporing. Bluetooth 4.0 ble annonsert i 2010, og er nå delvis utdatert når det gjelder sikkerhet. Alle paringsmetodene til 4.0 og 4.1 kalles for *LE Legacy Pairing*.

For å unngå passiv tyvlytting, krypterer BLE dataen som sendes mellom to enheter med en sikker blokkchiffer (AES-CCM). Problemet oppstår i nøkkelutvekslingen mellom de to enhetene. I den vanligste paringsmetoden i *LE Legacy Pairing, Just Works™*, brukes en svak midlertidig nøkkel (nøkkelen er tallet 0). Dermed er det enkelt for en angriper å finne ut hva korttidsnøkkelen blir med rå datakraft og gjetting. Andre paringsmetoder eksisterer, men disse trenger brukerinteraksjon eller andre trådløse protokoller (NFC). De er imidlertid rimelig sikre mot MITM (Bon, 2015).

4.2 introduserer *LE Secure Connections*. I denne utgaven av Just Works™-paring blir det kun generert én langtidsnøkkel ved hjelp av Elliptic Curve Diffie Hellman asymmetrisk kryptering. Dette løser problemet med passiv tyvlytting, men forbindelsen er fortsatt sårbar for MITM siden autentisering mangler. Et lite tillegg til Just Works™ der man sammenligner om to verdier er like gjør denne metoden sikker mot MITM (Bon, 2015).

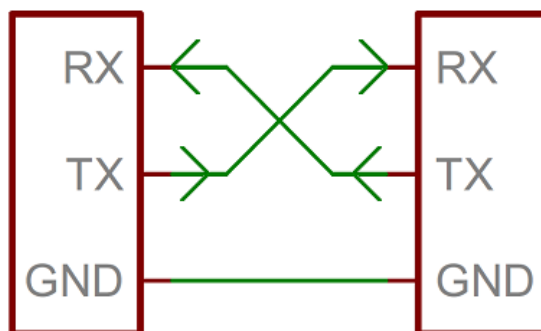
Etter at enhetene er parete, er det mulig å lagre nøklene på hver enhet slik at de kan gjenbrukes uten å gå igjennom hele paringsprosessen på nytt. Dette kalles *bonding*.

Bluetooth 5 som begynner å komme i produkter våren 2017, lover lengre rekkevidde og økt hastighet.

7.1.3 Seriell kommunikasjon og GPIO

Det er flere ulike kommunikasjonsprotokoller for å snakke med annen elektronikk og andre integrerte kretser. Seriell kommunikasjon består, i sin enkleste form, av å sende og motta binærdata over en asynkron seriell datalink. Figur 7.2 viser oppsettet mellom to enheter. En seriell enhet må ha en port for å motta data (RX) og en port for å sende data (TX) (sparkfun.com, utdatert).

GPIO er et generisk tilkoblingspunkt på en datamaskin eller integrert krets. Det er et grensesnitt mellom enheten og omverdenen som gjør det mulig koble til for eksempel knapper og lys. Tilkoblingspunktene kan være konfigurert som input eller output. I output-modus kan tilstanden til et punkt være enten høy (typisk 5V eller 3.3V) eller lav (0V). I input-modus leses også signalet som høyt eller lavt med en trekk opp- eller trekk ned-motstand, og kan dermed brukes til *interrupts* (raspberrypi.org, utdatert).



Figur 7.2: Seriell databuss (sparkfun.com, udatert)

7.2 Skyteknologi

7.2.1 Skybasert tingenes internett

Det er flere skyløsninger på markedet med støtte for IoT og det er et marked i stor vekst. Microsoft Azure IoT Hub og AWS IoT er allerede nevnt. IBM har en løsning kalt Watson Internet of Things Platform og Google har Google Cloud IoT som prøver å løse lignende problemer som Microsoft og Amazon. Disse fire leverandørene dominerer markedet totalt. Data fra Synergy Research Group (2017) viser at AWS holder seg på 40 % markedsandel i slutten av 2016, mens Microsoft, Google og IBM har 23 % markedsandel kombinert. De tre sistnevnte leverandørene vokser raskere enn AWS.

Dette prosjektet vil ta i fokusere på AWS IoT siden AWS er mest brukt i markedet i dag og har et veldig modent økosystem. Azure IoT Hub nevnes kort. De fleste skysystemene tilbyr veldig like løsninger.

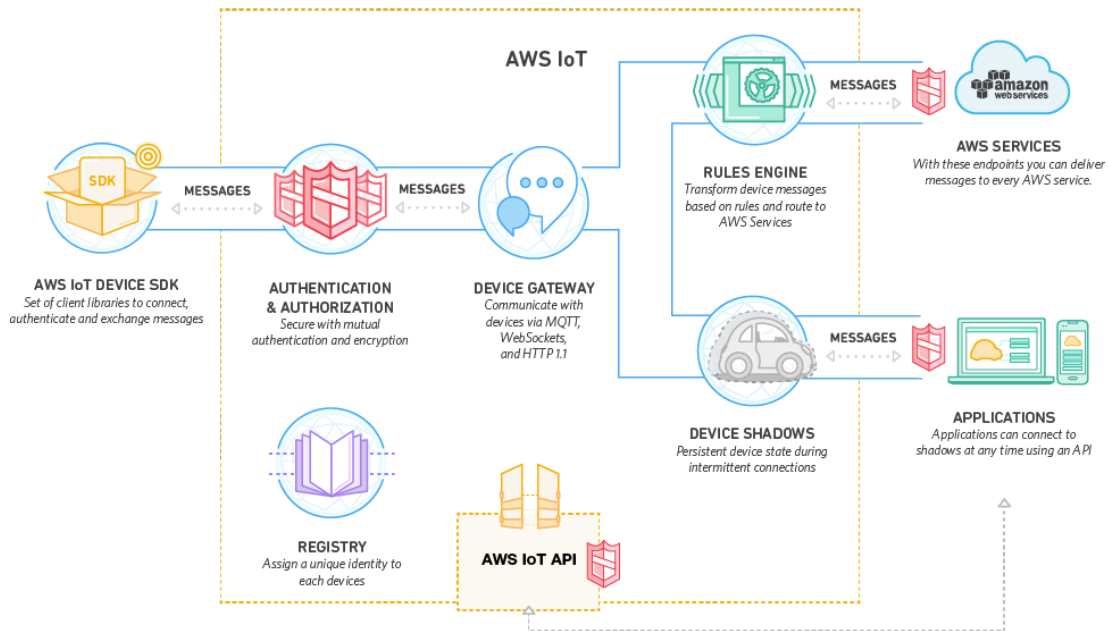
7.2.2 AWS IoT

Amazon sin IoT-løsning i skyen (AWS IoT) ble annonsert 9. oktober 2015. Viktige aspekter av denne løsningen er *AWS IoT Device SDKs* som tilkobler mot en *device gateway*, og en *rules engine* som tillater løsningen å integrere mot andre AWS-tjenester. Se figur 7.3 for arkitekturoversikt. Denne arkitekturen likner veldig på den Porter og Heppelmann (2014) foreslo i kapittel 2.

Thing registry er en liste av alle enheter tilkoblet tjenesten, og en *device shadow* holder styr på tilstanden til en enhet som kan hentes eller modifiseres fra andre applikasjoner.

AWS IoT Device SDKs

SDK-er er åpen kildekode på Github, og er tilgjengelige for embedded C, JavaScript (Node.js og nettleter), Arduino Yún, Java, Python, iOS og Android (Amazon Web Services, 2016a).



Figur 7.3: AWS IoT: Hvordan det virker (Amazon Web Services, 2016b)

Sikkerhet: Autentisering og tilgangskontroll

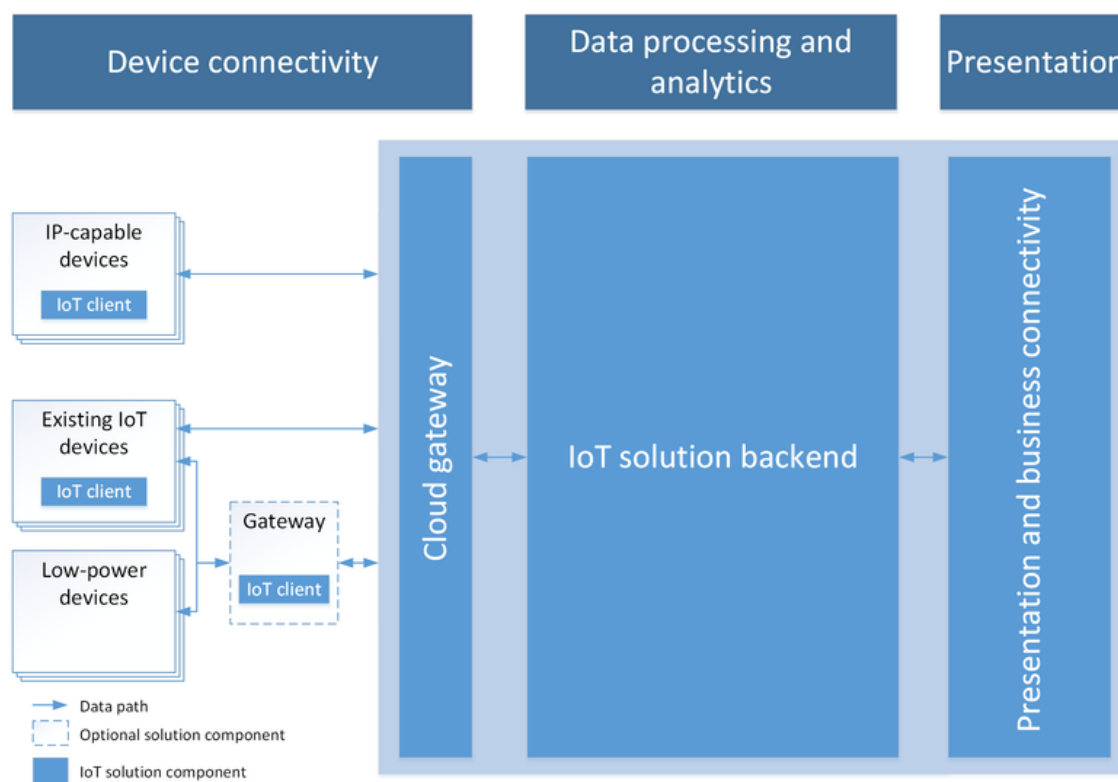
AWS IoT tilbyr ende-til-ende-kryptering og gjensidig autentisering av alle meldinger med TLS og klientside-X.509-sertifikater. Andre autentiseringsmetoder som Amazon sin egen SigV4-protokoll er tilgjengelig også.

Device gateway

Device gateway fungerer som en skalerbar *meldingsbroker* basert på publish-subscribe-mønsteret, med støtte for MQTT (publish/subscribe), MQTT over WebSockets (publish/subscribe) og HTTP (publish).

Rules engine

Rules engine evaluerer innkommende meldinger etter at de passerer igjennom device gateway, og videresender disse meldingene til resten av AWS-økosystemet avhengig av hvilke regler som er satt opp. Dette betyr at meldinger for eksempel kan bli endret og sendt videre, lagret i en database, aggregert i en dataanalyseløsning og brakt videre til andre skytjenester. Amazon Web Services (2016b) nevner AWS Lambda, Amazon Kinesis, Amazon S3, Amazon Machine Learning, Amazon DynamoDB, Amazon CloudWatch og Amazon Elasticsearch Service som mulige integrasjoner.



Figur 7.4: Microsoft Azure IoT Hub: Løsningsarkitektur (Azure, 2017)

Device shadows

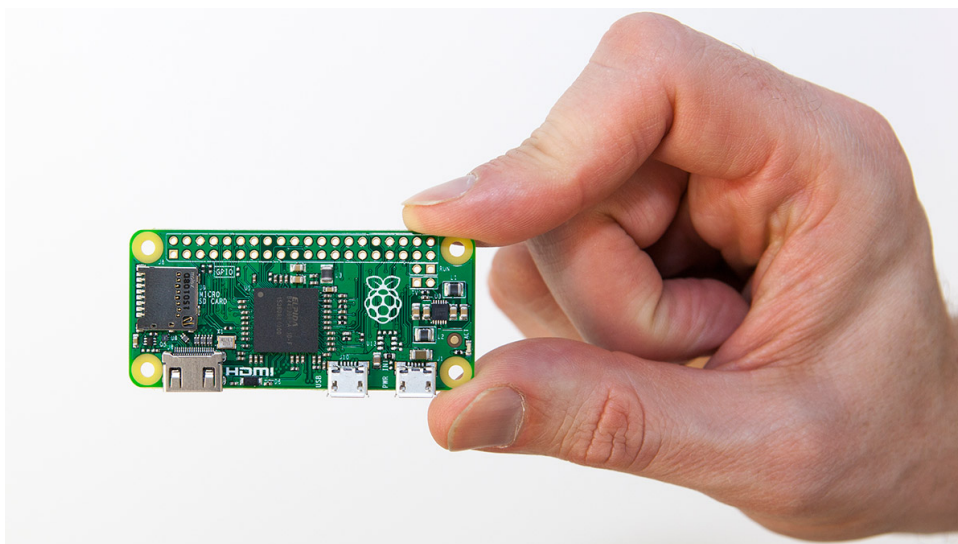
Alle enheter koblet til device gateway har en thing shadow som er et JSON-dokument med nåværende tilstand og informasjon om enheten. En thing shadow til en ting er alltid tilgjengelig, selv om enheten er koblet fra Internett. Den kan alltid bli hentet og endret over HTTP eller MQTT. En thing shadow er identifisert med sitt unike navn. AWS IoT reserverer emnenavnene *get*, *update* og *delete* for å kommunisere med device shadows.

Thing registry

Et thing registry er en JSON-liste av alle tingene som er tilkoblet til tjenesten. Hver ting har et navn, og kan valgfritt ha attributter i nøkkel-verdi-par, for eksempel modellnavn og watt for en lyspære. Det er mulig å lage forskjellige typer ting og assosiere disse typene med en ting.

7.2.3 Microsoft Azure IoT Hub

Microsoft Azure IoT Hub (figur 7.4) har mange av de samme egenskapene som AWS IoT med andre navn: *cloud gateway*, *device twins* og *identity registry*. Autentisering gjøres enten med et unikt token per enhet eller med X.509-sertifikater. I praksis tilbyr de to skyteknologileverandørene veldig mye av det samme.



Figur 7.5: Raspberry Pi Zero W. Foto: <http://www.raspberrypi-spy.co.uk/>

7.3 Prototypeplattform: Raspberry Pi Zero W

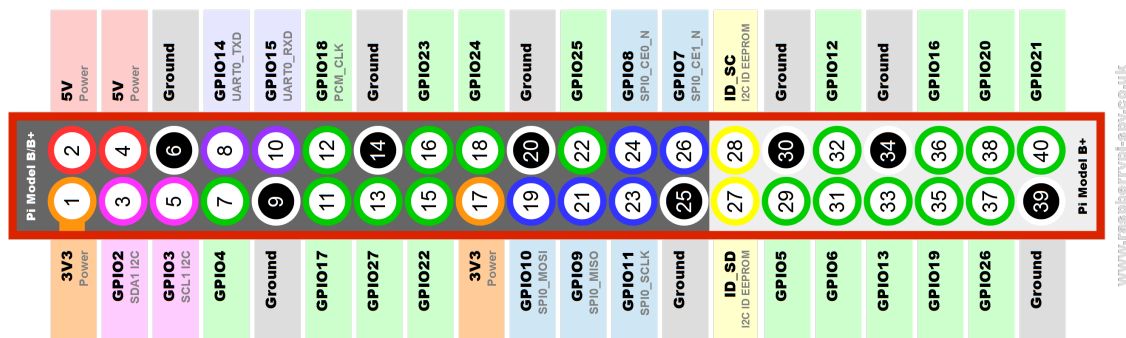
Raspberry Pi Zero W^{1 2} er en liten og billig datamaskin med lavt strømforbruk lansert i februar 2017 (figur 7.5). Den har følgende spesifikasjoner:

- 1GHz, énkjerne-CPU
- 512MB RAM
- 802.11 b/g/n trådløs LAN
- Bluetooth 4.1 (BLE)
- Mini HDMI og USB On-The-Go-porter
- Micro SD-kortmodul
- HAT-kompatibel 40-pin header
- **Strømforbruk:** typisk 100mA i hvilemodus og maks 350mA under stress
- **Størrelse:** 65 mm × 30 mm × 5 mm

Figur 7.6 viser de ulike portene til GPIO-headeren. Det er flere kontakter for jording, 5V, 3V, og noen porter kan brukes til seriell kommunikasjon og I2C i tillegg til GPIO. Raspberry Pi Zero W kjører på Raspbian, et Debian-basert operativsystem laget spesielt for de ulike utgavene av Raspberry Pi med Linux-kjernen i bunn..

¹<https://www.raspberrypi.org/products/pi-zero-w/>

²<https://www.raspberrypi.org/help/faqs>



Figur 7.6: Pi Zero W: GPIO header. Foto: <http://www.raspberrypi-spy.co.uk>

7.4 Sensorer og aktuatorer

En sensor er en innretning som gir signal om en tilstand i den fysiske omverdenen. Eksempler på sensorer kan være termometer, bevegelsessensor, ultralyd og trykknapp. Aktuatorer er innretninger som påvirker den fysiske omverdenen på en eller annen måte, for eksempel lysdioder, buzzere og LCD-skjermer.

7.4.1 Pulsoksymeter

Et pulsoksymeter er en sensor som måler oksygenmetning i blodet og pulsfrekvens.

Trondheim kommune lånte ut en Nonin 3230 (se figur 7.7) til bruk i denne masteropp-gaven. Nonin 3230 er et pulsoksymeter med støtte for BLE 4.0. Dette pulsoksymeteret ble anbefalt av Austad mfl. (2016) i rapporten *Sensorer til støtte for avstandsoppfølging*. De trakk fram at alle Nonins produkter er klinisk validerte og at det er støtte for BLE.

Nonin 3230 måler oksygenmetning fra 0 til 100% og pulsfrekvens fra 18 til 321 slag per minutt. Den går på to AAA-batterier, og er spesifisert for 2200 spotmålinger (25 sekunder per måling).

7.4.2 Fingeravtrykksensor

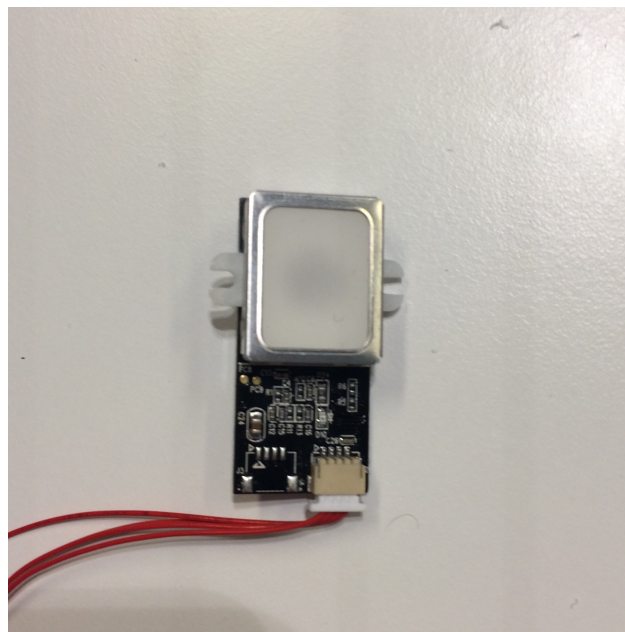
Fingeravtrykksensor er en sensor som kan prosessere et fingeravtrykk og verifisere og matche et avtrykk mot et tidligere lagret bilde av avtrykket. I dette prosjektet er det blitt kjøpt inn sensor som heter GT-511C3 (figur 7.8). Prisen på denne sensoren er litt over 400 kr. Den tar i mot enkle kommandoer for å styre sensoren over seriell kommunikasjon.

7.4.3 Trykknapp og lysdioder

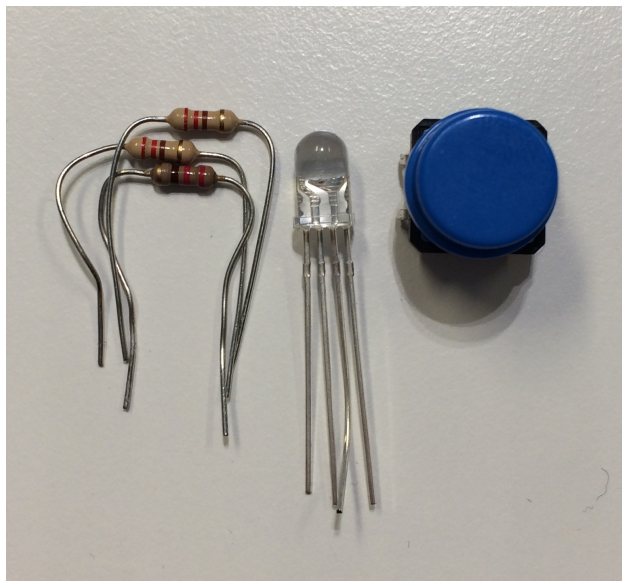
Lysdioder er enkle aktuatorer som gir fra seg lys og kan styres programmatisk. I dette prosjektet brukes tre RGB-lysdioder og en hvit lysdiode. RGB-lysdiodene trenger tre



Figur 7.7: Nonin 3230



Figur 7.8: GT-511C3



Figur 7.9: RGB-lysdiode, motstander og trykknapp

tilkoblingspunkter for rødt, grønt og blått lys i tillegg til jording. RGB-lysdiodene må ha en motstand foran seg. Lysdiodene styres med pulsbreddemodulasjon i software. Det brukes også en helt enkel trykknapp som er konfigurert til pullup, det vil si at verdien er 0 (LOW) når knappen trykkes ned og 1 (HIGH) når knappen er åpen. Figur 7.9 viser noe som tilsvarer de ulike komponentene.

Kapittel 8

Design og implementasjon av et skytilkoblet pulsoksymeter

Kapittelet beskriver hvordan en prototype av et skytilkoblet pulsoksymeter med innebygget autentisering ble utviklet. Første del av kapittelet handler om designkrav, brukerscenario og beskriver hvordan prototypen ble laget og hvordan den ble sendt ut. Siste del av kapittelet går mer detaljert inn i utviklingen av prototypen, med delkapitlene *Implementasjon av klientkomponenter* og *Implementasjon av skytilkobling*.

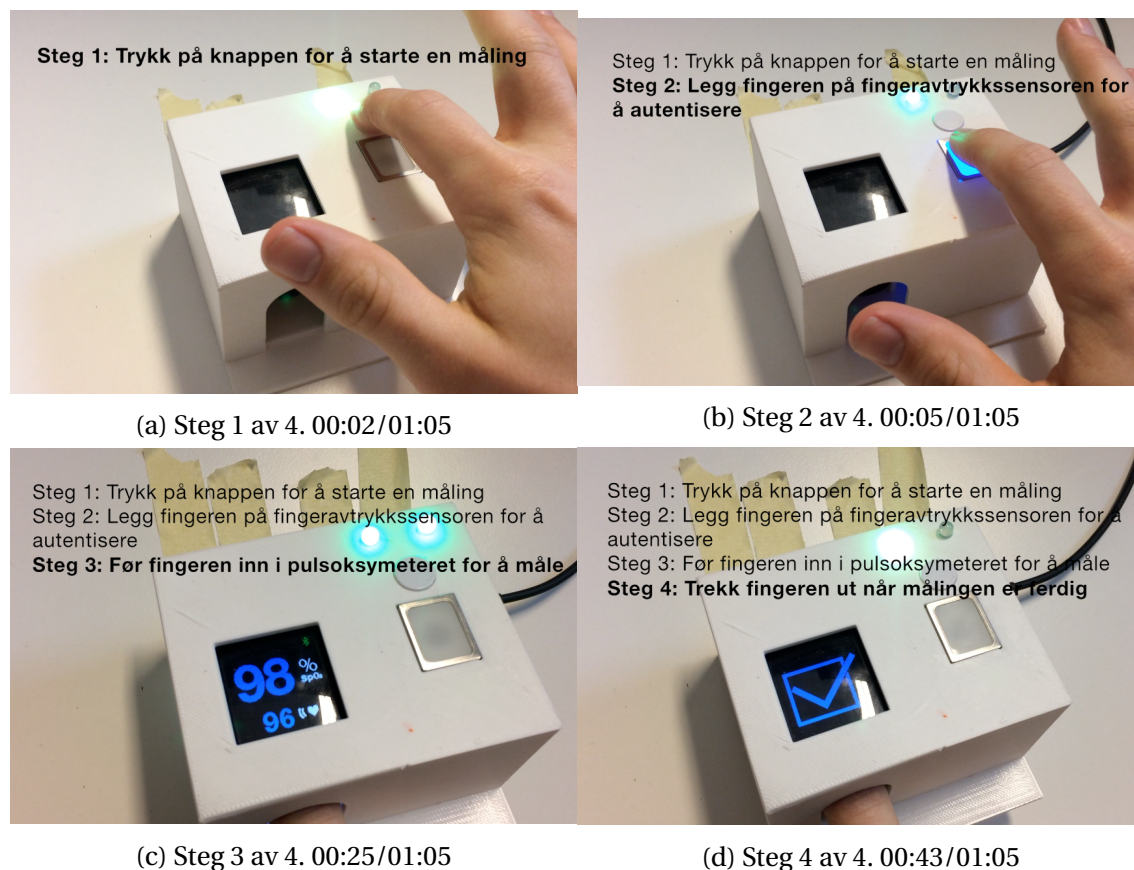
8.1 Designkrav til prototype

FK1 Når brukeren trykker på knappen skal systemet sette i gang en måling

- FK1.1** Brukeren må legge fingeren på fingeravtrykksensoren for å autentisere innen 10 sekunder.
- FK1.2** Ved vellykket autentisering skal brukeren få 120 sekunder til å gjennomføre en måling.
- FK1.3** Brukeren starter målingen ved å stikke fingeren inn i pulsoksymeteret. En måling er definert som kontinuerlig datainnhenting i 20 sekunder.
- FK1.4** Systemet skal sende data fra målingen kontinuerlig til en skytjeneste.
- FK1.5** Når en måling er ferdig, skal skjermen på pulsoksymeteret og lysdiodene indikere til brukeren at det er godkjent.
- FK1.6** Systemet skal vise rødt lys og sende brukeren tilbake til sovende modus dersom et av de foregående kravene ikke oppfylles.

FK2 Når brukeren holder inne knappen i over fire sekunder og slipper i utviklingsmodus, skal prosessen for å registrere et fingeravtrykk settes i gang.

- FK2.1** Registreringen blir gjort sammen med helsepersonell som forklarer hva som skal gjøres. Brukeren legger fingeren på sensoren, venter til det blir et blått blink, og gjentar dette to ganger til med samme finger.



Figur 8.1: Stegene i brukerinteraksjonen (Drivenes, 2017)

FK2.2 Dersom noe går galt under registreringen, skal systemet vise rødt lys og sende brukeren tilbake til sovende modus igjen.

8.2 Bruksscenario

Geir (70) har KOLS og foretar en måling med pulsoksimeter hver dag som en del av morgenrutinen. I systemet har han lagt inn høyre pekefinger. Han ser at pulsoksimeteret er oppe og kjører og blinker gult. Han trykker på den hvite knappen som er øverst og fingeravtrykksensoren begynner å lyse blått (**steg 1, figur 8.1a**). Han vet at det betyr at han må legge pekefingeren sin på sensoren. Når han legger fingeren på sensoren blinker lyset blått og slår seg av. Den høyre lysdioden er grønn, og han skjønner at fingeren ble registrert riktig (**steg 2, figur 8.1b**). En hvit lysdiode indikerer at det er på tide å stikke fingeren inn i måleren. Geir stikker fingeren inn i måleren som slår seg på, og observerer at Bluetooth-indikatoren blir grønn og at alt er greit (**steg 3, figur 8.1c**). Han har fingeren i måleren helt til det kommer opp et checkmark som indikerer at målingen er ferdig. Systemet går tilbake til å blinke gult (**steg 4, figur 8.1d**).

På sentralen oppdager sykepleier Lise (42) at sensordataene er kommet inn. Hun ser at batteriet til sensoren er lavt, og merker seg at hun må si i fra til Geir at han må huske å bytte batterier neste gang de har kontakt.



Figur 8.2: Prototypen ved fungerende måling

8.3 Beskrivelse av prototype

Prototypen bestod av følgende komponenter på klientsiden:

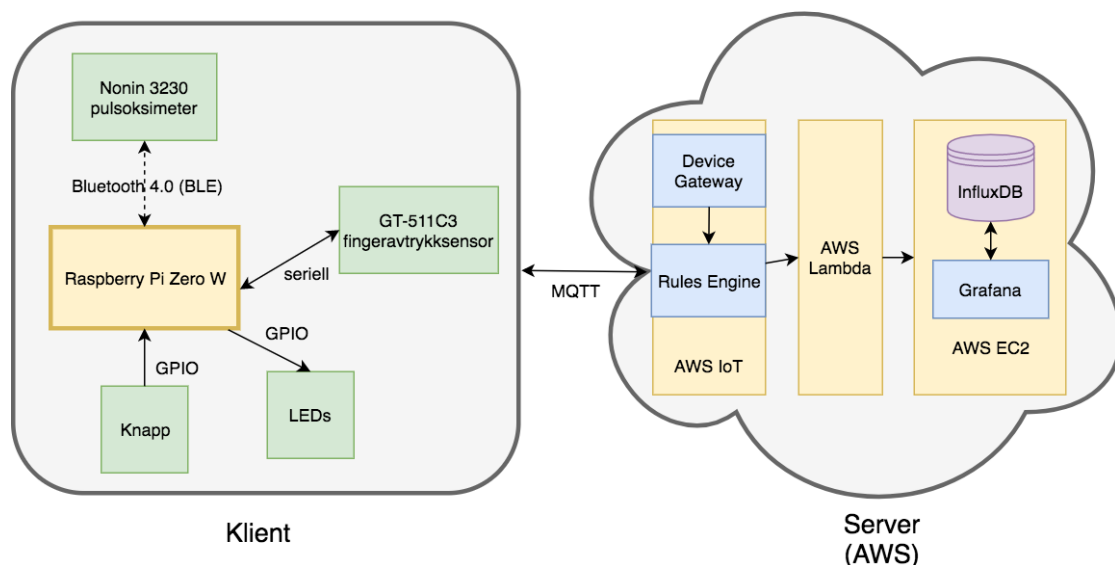
- Raspberry Pi Zero W
- GT-511C3 (fingeravtrykksensor)
- Nonin 3230 (pulsoksymeter med BLE)
- Standard trykknapp
- Tre RGB-lysdioder og én hvit lysdiode
- 3D-printet hus

Figur 8.2 viser prototypen når det ble gjennomført en vellykket måling med overføring. Drivenes (2017) viser en video av en vellykket måling og et forsøk på å autentisere med feil fingeravtrykk etterpå¹.

Komponentene ble presentert i forrige kapittel. Raspberry Pi Zero W ble valgt som prototypeplattform istedenfor alternativer som Tessel 2 og Arduino Yún. Det var et ønske om å bruke JavaScript og Node.js som utviklingsmiljø, og Arduino bruker C/C++ som utviklingspråk. Tessel 2 er en prototypeplattform basert på Node.js, men har få konfigurasjonsmuligheter og BLE er ikke innebygget.

Prototypen bestod av to deler, en klientsideløsning og en serverløsning. Arkitekturoversikt over løsningen med hvordan komponentene kommuniserte med Raspberry Pi, og hvordan Pi-en kommuniserte med serverløsningen basert på AWS er i figur 8.3.

¹<https://www.youtube.com/watch?v=ALizPkFko9U>



Figur 8.3: Arkitekturoversikt over løsningen

Klientsideløsningen ble utviklet rundt et Nonin 3230-pulsoksymeter utlånt av Trondheim kommune. Den kjørte på Node.js, en kryssplattform JavaScript-runtime som bruker den samme JavaScript-motoren som Chromium (V8). Løsningen bestod av én hovedapplikasjon, og én tilhørende modul hver for henholdsvis fingeravtrykksensoren, pulsoksymeteret og knapp/lysdioder. Disse modulene er utdypt nærmere i neste delkapittel, *Implementasjon av klientkomponenter*. Koden til hovedmodulen er i vedlegg A.1.

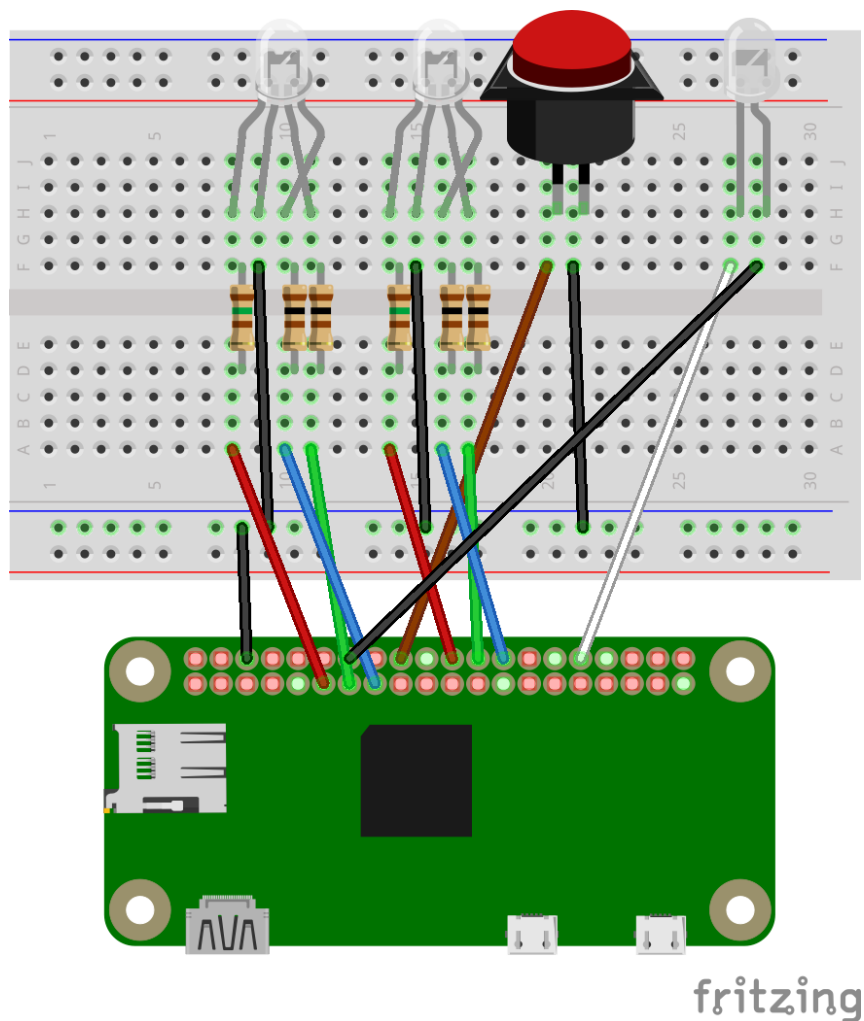
Applikasjonen startet automatisk opp på Raspberry Pi, og loggførte alle hendelser til en tekstfil. Prototypen sendte IP-adressen sin ved oppstart slik at det var mulig å nå den utenfra. Klientsiden snakket med serveren over MQTT, og ble autentisert med et sertifikat. Sensordataen fra pulsoksymeteret ble publisert til AWS IoT dersom brukeren var autentisert med fingeravtrykksensor. Virkemåten til prototypen var den samme som designkravene fra starten av dette kapitlet.

Serveren, som var basert på AWS IoT, hadde en regel for å prosessere dataen som kom inn. Denne regelen sendte dataen videre til en lambdafunksjon som puttet dataen i InfluxDB, en tidsseriedatabase. Denne databasen ble knyttet opp mot Grafana, et åpent kildekode-verktøy som integrerer med InfluxDB og viste fram grafer og metrikker i ulike konfigurerbare *dashboards*. Serverløsningen er videre utdypt i delkapittel 8.5.

8.4 Implementasjon av klientkomponenter

8.4.1 Trykknapp og lysdioder

Figur 8.4 viser hvordan trykknappen og lysdiodene ble koblet på portene til Raspberry Pi med et *breadboard*. Oppkoblingen er logisk ekvivalent med prototypen. I selve



Figur 8.4: Oppkobling av trykknapp og lysdioder

prototypen ble to RGB-lysdioder og en knapp loddet fast med motstander på et lite Brett av Terje Røsand (se figur 8.5). Koblingskjemaet er i figur 8.6. Alle komponentene delte felles jord.

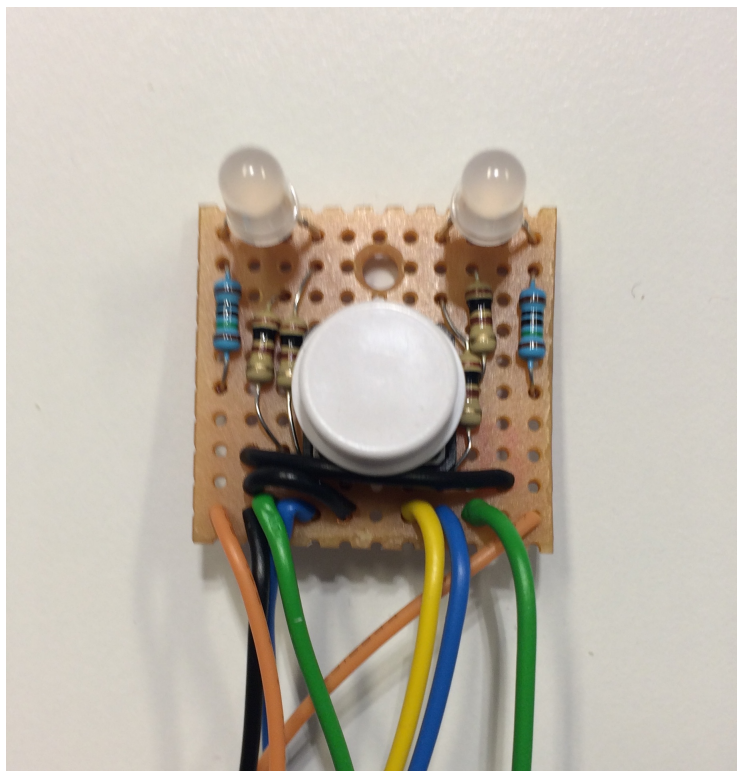
For å gjøre håndteringen av knapp og lysdioder enklere, ble det utviklet et lite JavaScript-bibliotek. Koden til dette biblioteket er lagt ved i tillegg A.2. Kodesnutt 8.1 er et eksempel på bruk av dette biblioteket med den samme oppkoblingen som i figur 8.4.

```

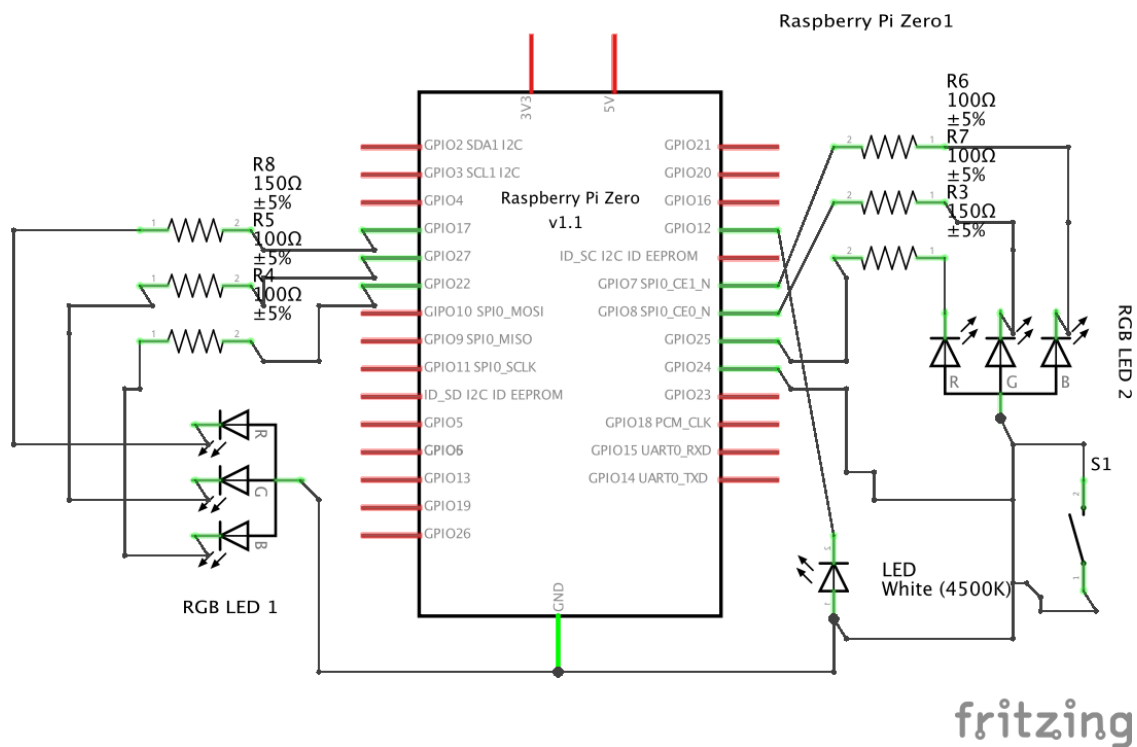
1  const button = new Button({ gpio: 24, isPullup: true });
2  const rgbLed1 = new RGBLed({ red: 25, green: 8, blue: 7 });
3  const rgbLed2 = new RGBLed({ red: 17, green: 27, blue: 22 });
4  const whiteLed = new Led(12);
5
6  button.on('click', () =>
7    console.log('press and release within 500 ms'));
8  whiteLed.on();
9  rgbLed1.color('blue').on();
10 rgbLed2.color('#DC143C').strobe();

```

Kodesnutt 8.1: Bruk av pigpio-components



Figur 8.5: Oppkobling av trykknapp og lysdioder på brett



Figur 8.6: Koblingskjema av trykknapp og lysdioder

8.4.2 GT-511C3

Koden til fingeravtrykksensoren ble utviklet som en tynn abstraksjon over *Node Package Manager* (NPM)-biblioteket *GT-511C3*², utgitt under BSD-3-lisens. Implementasjonen ligger i filen *index.js* i modulen *fingerprint*³. Denne modulen eksporterer to metoder: *enrollFingerAndRetrieveTemplate* og *setFingerprintTemplateAndVerify*.

Den første metoden går igjennom prosessen for å legge til et fingeravtrykk i databasen til sensoren, og henter ut et avtrykk på 498 bytes. Dette avtrykket kan og bør lagres i en skytjeneste, men dette ble ikke implementert i prototypen. Den andre metoden har støtte for å ta inn et slikt avtrykk, og legge det inn i databasen til sensoren før avtrykket verifiseres ved å legge riktig finger på sensoren. I prototypen ble det antatt at avtrykket lagres på samme sted i databasen hver gang.

Fingeravtrykksensoren ble koblet til GPIO14 (TX) og GPIO15 (RX) på Raspberry Pi som har seriell kommunikasjon. Sensoren ble også koblet til jord og 5V for strøm. Ut av boksen støtter ikke Raspberry Pi Zero W seriell kommunikasjon fordi Bluetooth opptar den serielle porten. Derfor ble *mini-uart* slått på. Dette er en seriell port med mindre hastighet knyttet til CPU-frekvensen.

8.4.3 Nonin 3230

Nonin ble kontaktet for å tilgang på integrasjonsguiden og kommunikasjonsprotokollen til pulsoksymeteret. Se vedlegg C. Utgaven av Nonin 3230 Trondheim kommune har kjøpt inn bruker en proprietær Bluetooth-tjeneste med 128-bit Universally unique identifier (UUID). I nyere utgaver kan man også bruke den åpne spesifikasjonen til *Bluetooth SIG Pulse Oximeter Service*. Den proprietære tjenesten heter *Nonin Oximetry Service* og har UUID *46a970e00d5f11e28b5e0002a5d5c51b*. Tjenesten har to karakteristikk – *Nonin Oximetry Measurement* som sender data hvert sekund som spesifisert i tabell 8.1 og *Nonin Control Point* som kan brukes til å synkronisere skjermen, indikere at en måling er ferdig, eller sette sikkerhetsmodus.

Et NPM-bibliotek⁴ kalt *nonin-3230-ble* ble utviklet for å gjøre integrasjonen mot sensoren enklere. Biblioteket baserer seg på *noble-device*⁵, og inneholder metoder for å oppdage et pulsoksymeter, koble til, lytte etter ny sensordata og indikere at en måling er ferdig. Sensordata kommer som et JavaScript-objekt og inneholder teller, pulsfrekvens, SpO2 og et statusobjekt (se tabell 8.2). Koden er lagt ved i appendiks A.4. Eksempel på bruk kan finnes i kodesnutt 8.2.

Koden slik den er i dag er lagt opp til å utføre kontinuerlige målinger hvor sensordata mottas hvert sekund. Pulsoksymeteret har også støtte for å utføre en spot-måling, det vil si én god måling som er kvalitetssikret av sensoren. Trondheim kommune bruker kun spot-målinger i sin løsning i dag.

²<https://github.com/the-AjK/GT-511C3>

³<https://github.com/andybb/smart-pulse-oximeter/blob/master/fingerprint/index.js>

⁴<https://github.com/andybb/nonin-3230-ble>

⁵<https://github.com/sandeepmistry/noble-device>

Datapakke til pulsoksymeter		
Byte	Felt	Beskrivelse
1	Lengde	Antallet bytes inkludert denne.
2	Status	Indikerer nåværende enhetsstatus (tabell 8.2).
3	Batterispennning	Spenningsnivået til batteriene som brukes.
4-5	Perfusjonsindeks (PI)	$PI = AC/DC * 100\%$
6-7	Teller	Verdien økes hvert sekund (0-65535). Kan bli brukt til å sjekke at det ikke er noe datatap.
8	SpO2	SpO2-prosent, 0-100 (gjennomsnitt av fire slag).
9-10	Pulsfrekvens	Pulsfrekvens i slag per minutt, 0-321 (gjennomsnitt av fire slag).
>11	Reservert	Reservert til fremtidig bruk

Tabell 8.1: Nonin 3230: Format på datapakke (Nonin Medical, Inc., 2016, s. 20)

```

1  const Nonin3230 = require('nonin-3230-ble');
2
3  Nonin3230.discover((pulseOximeter) => {
4    pulseOximeter.connectAndSetup((error) => {
5      if (error) {
6        console.error(error);
7      }
8      let counter = 0;
9      // receive a new measurement every second
10     pulseOximeter.on('data', (data) => {
11       // data: { counter: int, pulseRate: int, oxygenSaturation: int,
12         status: object }
13       counter++;
14       if (counter > 15) {
15         pulseOximeter.stopMeasurement(() => console.log('Stopped'));
16       }
17     });
18   });

```

Kodesnutt 8.2: Bruk av nonin-3230-ble

Kildekoden er åpen og kan kjøres fra alle enheter som har støtte for Node og BLE.

8.4.4 3D-printet hus

Det 3D-printede huset til prototypen ble designet og printet ut av Terje Røsand basert på alle komponentstørrelsene i samarbeid med Andreas Drivenes. Å integrere pulsoksymeteret i huset var den mest utfordrende delen av designet, og det gjorde at det ikke var mulig å få plass til noen andre komponenter over denne sensoren. Alle de andre sensorene måtte plasseres til høyre for pulsoksymeteret. Huset bestod av to ulike deler

Statusfelt (aktiv høy)		
Bit	Felt	Beskrivelse
7	Reservert	Reservert til fremtidig bruk.
6	Kryptering	1 = tilkoblingen er kryptert 2 = tilkoblingen er ikke kryptert
5	Lavt batteri	Batterinivået er lavt. Bytt batteri.
4	CorrectCheck	1 = OK 2 = Plasser fingeren lengre inn i enheten
3	Søker	Oksymeteret søker etter etterfølgende pulssignaler.
2	SmartPoint	Brukt til å indikere at dataen har bestått SmartPoint-algoritmen.
1	Lavt/svakt signal	Styrken til pulssignalet er 0,3 % modulasjonsgrad eller mindre.
0	Indikering av skjermssynkronisering	Indikerer at skjermen er synkronisert med innsamleren av data.

Tabell 8.2: Nonin 3230: Format på statusfeltet til datapakke (Nonin Medical, Inc., 2016, s. 20)

som klikkes sammen. Figur 8.7 og figur 8.8 viser toppdelen, og figur 8.9 viser bunndelen.

Pulsoksymeteret og brettet med lysdioder og knapp ble skrudd fast med tre skruer til toppdelen. Pulsoksymeteret ble presset mot toppdelen med et innlagt, mykt materiale. Raspberry Pi Zero W ble skrudd fast med fire skruer til bunndelen.

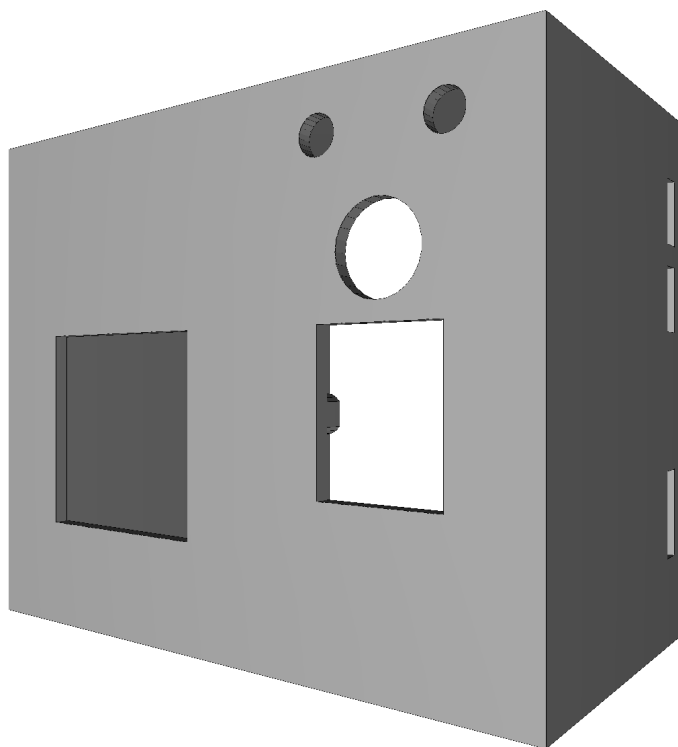
8.5 Implementasjon av skytilkobling

8.5.1 Registrering og oppkobling til AWS IoT

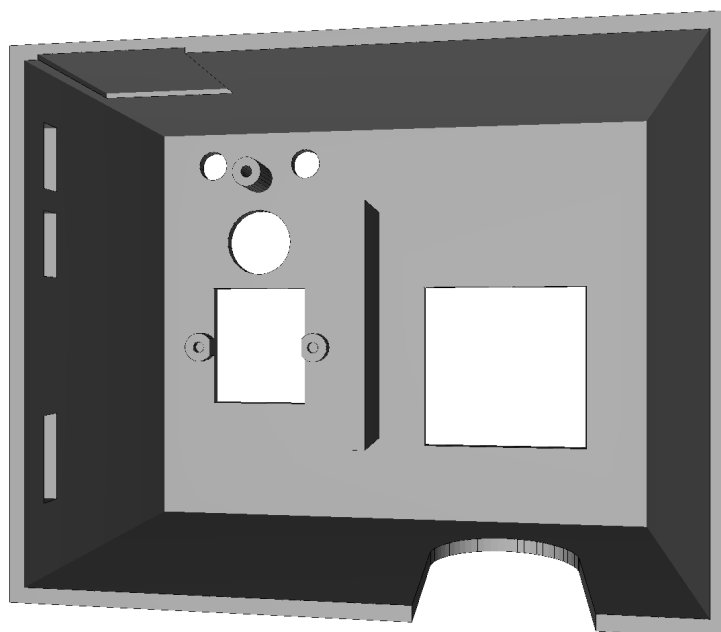
Som beskrevet i delkapittel 7.2.2, må en enhet registreres i thing registry i AWS IoT og få tilknyttet et sertifikat for å fungere. Kodesnutt 8.3 viser hvordan prototypen ble registrert hos AWS IoT med kommandolinjeverktøyet de tilbyr. Andre muligheter hadde vært å bruke webgrensesnittet eller gjort det programmatisk med REST API-et.

Kommandoen på linje 2 i kodesnutt 8.3 genererte et X.509-sertifikat og et 2048-bit RSA nøkkelpar. Den printet ut et Amazon Resource Number (ARN) og en id som hører til sertifikatet, sertifikatsdataen og det genererte nøkkelparet (assymetrisk). Sertifikatet ble satt til å være aktivt. Det er også mulig å registrere egne sertifikater fra en egen sertifikatutsteder.

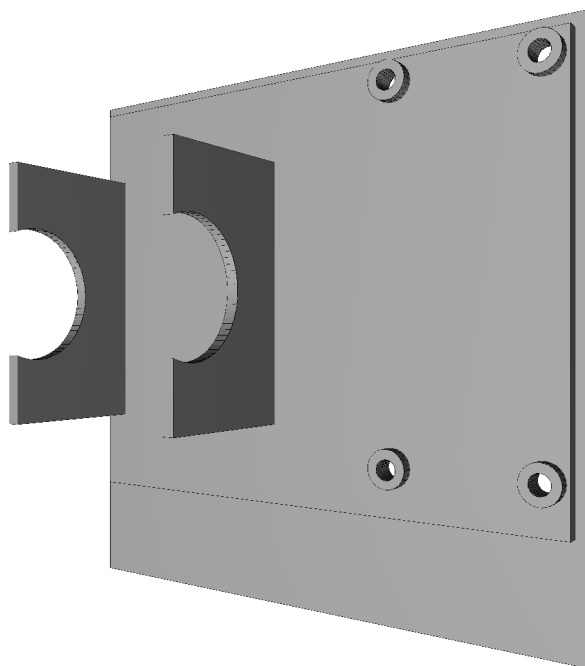
En policy trengs for at prototypen skal godkjennes av AWS IoT. Policy er et JSON-dokument med flere *statements* som inkluderer feltene *effect*, *action* og *resource*. Policyen til prototypen ble laget på linje 3 i kodesnutt 8.3, og kan sees i kodesnutt 8.4. Denne policyen var veldig lite restriktiv og tillot alle actions (connect, publish, subscribe, get/modify thing shadow) på hvilken som helst resource. En mer restriktiv policy hadde vært mer



Figur 8.7: 3D-modell av prototype: Topp



Figur 8.8: 3D-modell av prototype: Topp, sett fra bunn



Figur 8.9: 3D-modell av prototype: Bunn

relevant om prototypen skulle blitt satt i produksjon.

De to siste kommandoene knyttet henholdsvis sertifikatet til tingen, og policyen til sertifikatet. Dette gjorde at man kunne autentisere prototypen basert på sertifikatet.

```

1 aws iot create-thing --thing-name "smart-pulse-oximeter"
2 aws iot create-keys-and-certificate --set-as-active
3 aws iot create-policy --policy-name "smart-pulse-oximeter-Policy" --
  policy-document "file://policy.json"
4 aws iot attach-thing-principal --thing-name "smart-pulse-oximeter" --
  principal "[sertifikats-ARN generert paa linje 2]"
5 aws iot attach-principal-policy --principal "[sertifikats-ARN generert
  paa linje to]" --policy-name "smart-pulse-oximeter-Policy"
    
```

Kodesnutt 8.3: Registrere en ting i AWS IoT

```

1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": [
7         "iot:*"
8       ],
9       "Resource": [
10        "*"
11      ]
    }
  ]
}
    
```

```

12 |     }
13 |   ]
14 | }

```

Kodesnutt 8.4: Policy-dokument (policy.json)

For å koble klientsiden av prototypen til AWS IoT, ble nøklene og sertifikatet generert over brukt. AWS IoT autentiserer prototypen basert på tingnavnet, nøklene og sertifikatet. Se kodesnutt 8.5. Skyløsningen kjørte på AWS sin region i Frankfurt, *eu-central-1*.

```

1  const awsIot = require('aws-iot-device-sdk');
2
3  const device = awsIot.device({
4    keyPath: process.env.AWS_KEY_PATH,
5    certPath: process.env.AWS_CERT_PATH,
6    caPath: process.env.AWS_CA_PATH,
7    clientId: 'smart-pulse-oximeter',
8    region: 'eu-central-1'
9  });
10
11 device.on('message', (topic, payload) => {
12   // do something with payload
13 });
14 device.subscribe('oximetry');
15
16 // Pretend like we have got an object called 'oximetry' from the pulse
   oximeter
17 device.publish('oximetry', JSON.stringify({ oximetry }));

```

Kodesnutt 8.5: Koble prototype til AWS IoT

8.5.2 Oppsett av businessregler og visning av data

Kommandoen i kodesnutt 8.6 satte opp en businessregel i rules engine. Denne regelen sørget for å kjøre en AWS Lambda-funksjon med all dataen fra emnet *oximetry*. Lambda-funksjonen er i kodesnutt 8.7. Den ble basert på eksisterende kode for å sette opp og rulle ut en AWS Lambda-funksjon⁶. Under panseret tok den i bruk kommandoene *aws lambda create-function*, *aws lambda update-function-code* og *aws lambda update-function-configuration*.

Lambda-funksjonen satte inn data fra *oximetry* i InfluxDB som ble koblet sammen med Grafana for å vise fram dataene. InfluxDB og Grafana kjørte på Amazon EC2 (én instans av en vanlig virtuell server i skyen) i Docker. Grafana ble satt opp med tre paneller: ett panel for puls, ett panel for oksygenmetning og ett panel hvor alle var samlet. Se figur 8.10 og 8.11. Grafana bruker et SQL-lignende spørrespråk for å hente ut og aggregere data. InfluxDB er en åpen kildekode-tidsseriedatabase som er spesielt egnet for monitorering, metrikk og sensordata.

⁶<https://github.com/andybb/smart-pulse-oximeter-grafana>


```

1 aws iot create-topic-rule --rule-name "smart-pulse-oximeter-Rule" --
  topic-rule-payload "file://payload.json"
2
3 #payload.json file:
4 {
5   "sql": "SELECT * from 'oximetry'",
6   "description": "Send data from pulse oximeter to a database and
  viewing tool",
7   "actions": [
8     {
9       "lambda": {
10        "functionArn": "[ARN of lambda function]"
11      }
12    }
13  ],
14  "ruleDisabled": false,
15  "awsIotSqlVersion": "2016-03-23"
16 }

```

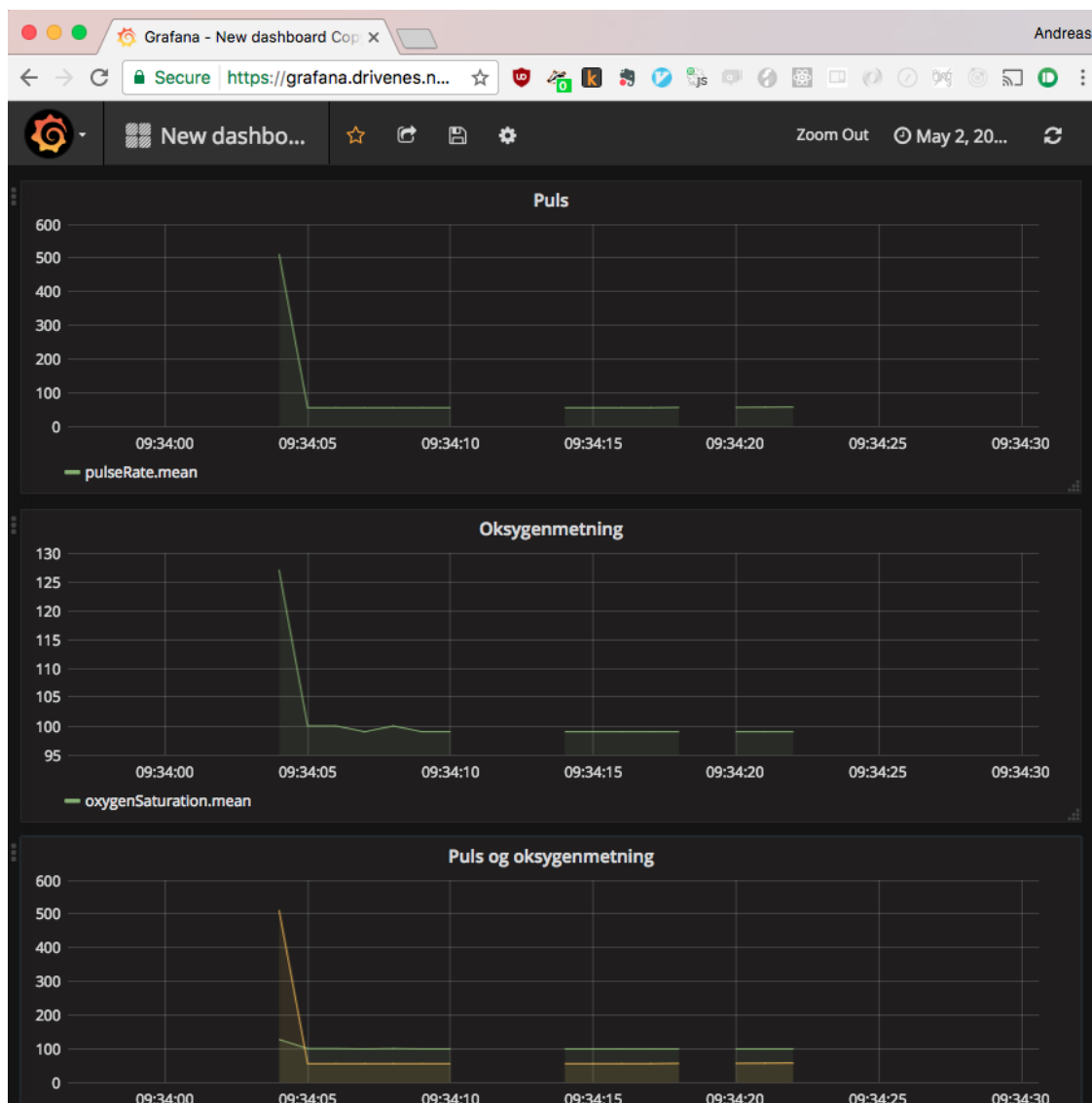
Kodesnutt 8.6: Sette opp en businessregel i AWS IoT

```

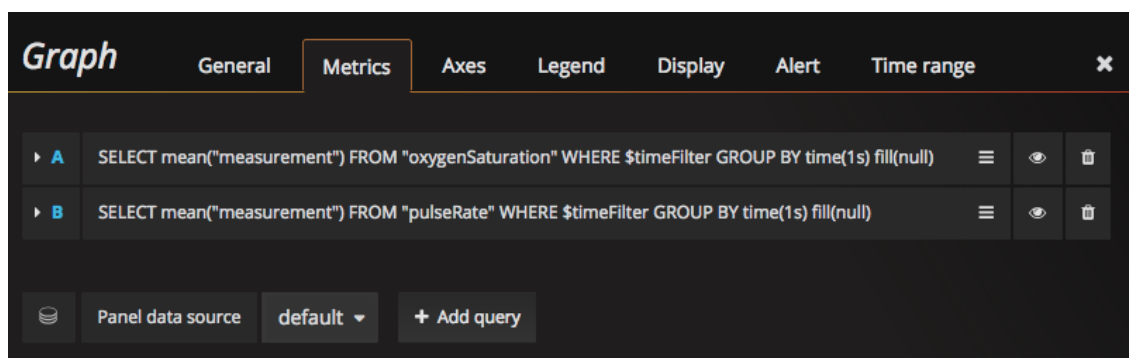
1 import 'babel-polyfill';
2 import λ from 'apex.js';
3 import Influx from 'influx';
4 import config from './config';
5
6 const influx = new Influx.InfluxDB({
7   host: config.influx,
8   database: config.influxDatabase
9 });
10
11 export default λ((event, ctx) => influx.writePoints([
12   {
13     measurement: 'oxygenSaturation',
14     fields: { measurement: Number(event.oxygenSaturation) }
15   },
16   {
17     measurement: 'pulseRate',
18     fields: { measurement: Number(event.pulseRate) }
19   }
20 ])
21 .then(() => {
22   ctx.succeed();
23 })
24 .catch((err) => {
25   console.log(err);
26   ctx.fail();
27 })
28 ));

```

Kodesnutt 8.7: AWS Lambda-funksjon for å sette dataen inn i en database



Figur 8.10: Grafana-dashboard av pulsoksymeterdataen



Figur 8.11: Oppsett av Grafana-dashboard

Kapittel 9

Evaluering av prototype og oppsummering

Kapittelet beskriver resultatet av evaluerings- og oppsummeringsintervjuene gjort med SINTEF og Trondheim kommune.

9.1 Planlegging av evaluering

Kapittel 4 redegjør for og drøfter valgene av evalueringsmetoder. Evalueringsmetoden er semistrukturerte intervjuer med innlagt demonstrasjon av prototype. Delkapittel 5.1 beskriver hvordan kontakten med Trondheim kommune ble opprettet.

Intervjuforespørselen til evaluering- og oppsummeringsintervju ligger vedlagt i tillegg D, og ble sendt ut til Trondheim kommune og SINTEF. Alle intervjuobjekter fikk beskjed om at intervjuet ble tatt opp med telefon og at lydopptaket blir slettet. I etterkant ble det informert om at alle deltakerne kunne lese igjennom transkriberingen fra intervjuet, og trekke seg fra prosjektet når som helst. De signerte anonyme og ikke-anonyme samtykkeerklæringer (vedlegg F).

Hensikten med intervjuene var å få tilbakemeldinger på den utviklede prototypen og ha en åpen samtale om hvilke løsninger som kan egne seg for avstandsoppfølging og hvordan utviklingen blir på dette feltet i fremtiden. Det var ikke en brukertest, men Trondheim kommune fikk prøve løsningen selv også. Invitasjonen resulterte i to separate gruppeintervjuer, ett intervju med Trondheim kommune representert ved N.N. og I. B. Sandvik og ett intervju med to forskere fra SINTEF. Selv om ikke alle resultatene her er direkte relevante for forskningsspørsmålene, er det tatt med for å få et helhetsbilde av avstandsoppfølging og hva kommunen og SINTEF jobber med.

9.2 Intervju med Trondheim kommune

Intervjuet ble gjennomført tirsdag 2. mai 2017 kl. 09.30 til rundt kl. 10.20 i lokaler som Trondheim kommune disponerer. I. B. Sandvik måtte gå kl. 10.00. Intervjuet startet

med en gjennomgang og forklaring av prototypen, før prototypen ble demonstrert. I. B. Sandvik la inn sitt eget fingeravtrykk og fikk prøve prototypen selv. Deretter dreide samtalen inn på tilbakemeldinger, nåværende status på HelsaMi+ og utviklingen på velferdsteknologifeltet og hvilke løsninger de så for seg i fremtiden. Intervjuet ble transkribert, og relevante deler ble fargekodet for temaene tilbakemeldinger, brukeropplevelse, pris, sikkerhet og autentisering og fremtidige løsninger. Alle disse temaene ble relatert til den løsningen som Trondheim kommune har i dag.

9.2.1 Tilbakemeldinger på prototypen

De generelle tilbakemeldingene var at prototypen var enkel og at de var imponert over hvor kort tid det hadde tatt å utvikle den. «Jeg synes konseptet er veldig bra. Det virka. Jeg synes jo at den prototypen er brukervennlig» (N.N., 2. mai, personlig kommunikasjon).

Nei, så konseptet her er jo veldig interessant da. Og det viser jo at det er relativt enkelt å få til da. Du har jo sikkert brukt en del timer på det da. (...) Det var jo veldig enkelt å forstå konseptet her, sant. Du brukte jo bare ett og to minutt å forklare (...) (I. B. Sandvik 2. mai, personlig kommunikasjon)

Senere i samtalen:

N.N.: «Ja den der ville ha fungert hjemme hos veldig mange av våre brukere.»

I. B. Sandvik:

Sant. Da tenker jeg at – la oss si da – tenkt deg brukerscenario da at noen brukere kan være – det er liksom SpO2-måler som er interessant, trenger ikke å rapportere noe annet, ikke sant. Så sier vi at du tar dette her tre ganger i uka. Og så bare tar du målinga og så er dataen sendt. That's it. Du trenger ikke å slå på noen nettbrett – du trenger ikke å slå på noen pc, du bare, hvis du ikke vil se grafen din som du kan, men, det er det du gjør, ikke sant. Tar målingen. Så får du kvittert ferdig (...).

I. B. Sandvik: «Jeg må si at innenfor de rammene du har så syntes jeg det her ble bra jeg.» (...) «Så er det litt artig å se ting som fungerer og ikke bare – det er nok av leverandører som prater om hvor bra alt skal bli.»

Kommunen sa at Nonin kunne tenkt på et lignende produkt: «Det første som slår meg det er jo dette med at sånn som Nonin som lager denne her, de kunne jo på en måte ha laget alt sammen i den enheten da, sant. Som kanskje hadde blitt litt større.» (I. B. Sandvik)

9.2.2 Brukeropplevelse

En workshop med brukere av HelsaMi+ viste, i følge I. B. Sandvik, at de fleste brukerne syntes nåværende løsning var enkel. Dette var noe som overrasket kommunen litt, i og med at de ikke syntes at nåværende brukerapplikasjon var så imponerende. «Og vi

hadde jo ønsket oss at leverandøren kunne ha vært litt mer framoverlent og litt mer i forkant av ting da. Men det er nå på en måte det vi har da. Men det interessante det var at brukerne synes at det var enkelt og greit.»

Aldersspennet på brukerne som var med på workshopen var fra 58 opp til rundt 82. N.N. sa at den yngste brukeren som har HelsaMi+ er 52 år.

At brukeren får et checkmark opp på skjermen til pulsoksymeteret når målingen er ferdig, var noe kommunen ikke hadde i deres løsning. Det var de interessert i å få med, og de tok bilde av det. I.B Sandvik sa: «Det der burde vi ha fått fiksa på. Det på en måte – det gir en endelig beskjed til brukeren at den er ferdig.»

9.2.3 Pris

I. B. Sandvik spurte om prisen på systemet. Han syntes løsningen var billig. Merk at svaret om prisen på fingeravtrykksensoren var feil. Det ble oppgitt at den kostet 25 dollar når den egentlige prisen er 50 dollar. Prisen på Raspberry Pi Zero W er 10 dollar, men det kan være vanskelig å få tak i kun enheten uten å kjøpe en startpakke i tillegg.

Raspberry Pi Zero W: ca. 100 kr (thepihut.com, 6. juni 2017)

GT511-C3: ca. 420 kr (sparkfun.com, 6. juni 2017)

Knapp, lysdioder, annet: ca. 30 kr (høyt estimat)

Totalsum: ca. 550 kr

9.2.4 Sikkerhet og autentisering

Kommunen var i gang med å få på plass sertifikater på nettbrettene, men I. B. Sandvik sa at de hadde problemer med å finne en god metode for å distribuere og håndtere sertifikatene:

(...) vi skal prøve å få implementert sertifikat på nettbrettene, på devicene. Så har vi en sak gående med Evry, som er kommunen sin part, leverandør. Men vi har ikke fått noen gode løsninger på hvordan vi skal gjøre det her. Hvordan skal du distribuere et sånt sertifikat ut til et nettbrett som er levert ut til en privatperson.

(...) for det vi holder på med og så se på sammen med e-helse og Norsk Helsenett, det er nå og så teste ut at vi kan legge sertifikatsdistribusjonen som en komponent i kall det – helseapplikasjonen da. Slik at du kan pushe ut sertifikater i fra applikasjonen til devicen.

9.2.5 Fremtidige løsninger i avstandsoppfølging og HelsaMi+

Kommunen har erfart at brukerne er veldig forskjellige. Noen er veldig komfortable med teknologi og føler seg hjemme, mens andre kun trykker på det de har fått opplæ-

ring i. I. B. Sandvik sa:

(...) man gjøre en individuell vurdering da – okei, for deg som er aktuell for tjenesten – okei, har du nettbrett i dag? Ja. Da kan vi kanskje bare pushe ut en applikasjon som du får tilgang til med sikkerheten. Har du nettbrett eller er du familiær med teknologi? Nei aldri brukt det før – okei, da har vi utstyr som du kan få låne av kommunen og du får opplæring, ikke sant.

I. B. Sandvik sin konklusjon var at kommunen må tilpasse tjenesten til brukeren. Det må gjøres en individuell vurdering i hvert enkelt tilfelle – dette passer også bra inn i kommunens strategi ellers. Å være til stede på de plattformene brukeren er på gjør også at kommunen kan bruke færre ressurser på å eie og vedlikeholde masse utstyr.

Brukerne var forskjellige når det gjaldt å ha tilgang på egne målinger: «(...) noen tenkte jo at 'ja, sikkert ikke så dumt'. mens andre 'nei, ikke interessert'. Det handla også sikkert om en del andre ting også. Diagnosen og ja, vil helst ikke forholde seg til det» (I. B. Sandvik). Det er ikke et krav at brukeren må se sine egne data om brukeren ikke har lyst: «(...) det trenger ikke nødvendigvis å være noe som du må ha tilgang til som bruker da, men at det – du skal kunne ha tilgang til det» (I. B. Sandvik).

Kommunen fikk forespeilet en løsning hvor sensoren var utenfor huben. Det vil si at produktet ikke er fullintegrert, men består av en løsning med flere sensorer rundt en hub som kan ligne på den som ble utviklet i dette prosjektet. I. B Sandvik sa at en slik løsning kanskje hadde vært bedre. Den løsningen hadde gitt mulighet til å koble på blodtryksmåler og vekt i tillegg til pulsoksymeter.

Kommunen ville holde seg på seks brukere med sensorer ut året fordi det var nok å administrere alt med de stabilitetsproblemene som som var i løsningen i dag.

(...) og så har vi for dårlig grafer og sånn til at vi kan gjøre oss noe god faglig nytte av det per i dag da. Så vi er veldig i utvikling med leverandøren. Men de som har sensorer er jo veldig fornøyd. Det er de. De synes jo at det er en ekstra trygghet å få vurdert sensordataen sine. (N.N.)

Fastlegen skal kobles på for å sette grenseverdier på sensordataen, men det har ikke skjedd ennå i påvente på at løsningen skal bli litt mer stabil.

9.3 Intervju med SINTEF

Intervjuet med to forskere fra SINTEF ble gjennomført i SINTEF sine lokaler kl. 10.30 til kl. 11.30 mandag 8. mai. SINTEF er en tjenesteutviklingspartner med Trondheim kommune på HelsaMi+. SINTEF har gjort en del intervjuer med brukere av HelsaMi+ for å være en uavhengig tredjepart slik at brukerne kan føle at de kan snakke fritt. Forskerne fikk et innblikk i bakgrunnen til prosjektet og hensikten med utviklingen av prototypen, før de presenterte seg selv og hva de jobbet med. Deretter handlet samtalen om HelsaMi+ og avstandsoppfølging. Prototypen ble vist fram, men ikke prøvd av forskerne. Samtaleemner var forberedt på forhånd, men intervjuobjektene fikk styre samtalen inn på emner de ville snakke om. Intervjuet ble markert utifra temaene forbedringspotensialene i HelsaMi+, bruken av sensorer og måledata, tilbakemeldinger på prototypen

og oppsummering og veien videre. Forskerne blir betegnet som henholdsvis F1 og F2 i intervjuet.

9.3.1 Forbedringspotensiale i HelsaMi+

Innrapportering av dagsform og egenbehandlingsplan

At egenbehandlingsplanen kun er på papir og ikke er en del av applikasjonen i HelsaMi+, gjør at brukerne ikke får et forhold til den. Egentlig skal brukerne selv sette inn tiltak ved en forverring ved å se på egenbehandlingsplanen:

(...) men det gjør de jo ikke i dag. I Trondheim da så går løsningen mest på at de bare snakker med helsepersonell som ringer dem opp hvis de er dårlig. Og det fører jo til at egenmestringen som vi ønsker at skal styrkes, ikke har noe særlig effekt akkurat nå da (F1, personlig kommunikasjon, 8. mai)

I tillegg sa F1 at brukerne skjønner at de blir ringt opp hvis de rapporterer inn noe som gult:

(...) fordi de vurderer om de ønsker hjelp eller ikke. Hvis de markerer gult så vet de at de blir ringt opp. Så det vil jo si at hvis de føler seg litt dårlig, men allikevel tenker at de mestrer selv, så scorer de grønt.

F2 fortsatte litt senere:

(...) tjenesten er lagt opp sånn nå at det er grønt, gult, og rødt. Og hvis du trykker på gul eller rødt så blir du alltid ringt opp. Og det gjennomskuer jo folk. Så hvis de vil være i fred, så trykker de grønt. (F2, personlig kommunikasjon, 8. mai)

Konsekvensen av dette kan bli at brukeren får grønn, grønn, grønn og så plutselig en rød måling. Systemet fanger ikke opp at brukerne holder på å bli dårlige fordi de venter for lenge.

9.3.2 Bruken av sensorer og måledata

«Har dere (...) troen på at man ved hjelp av sensorer kan (...) automatisere den prosessen i fremtiden?»

F1 svarte:

Jeg tror det blir mer et veldig viktig beslutningsstøtteverktøy for helsepersonell. Ja, det tror jeg. Og det å ha noen objektive data i tillegg – for den her scoringa rød gul grønn, den er jo subjektiv, utifra hvordan de føler seg (...)

F1 sa videre at brukerne kan oppleve økt egenmestring gjennom å få tilgang på egne måledata. Brukeren kan selv skjønne hvilke data som gjør at det er lurt å ta kontakt med legen. F1 viste til et eksempel i en annen kommune enn Trondheim der en bruker var hos fastlegen og viste fram kurvene, som fastlegen deretter brukte til å regulere

blodtrykksmedisinen. Andre kommuner enn Trondheim har mer bruk av sensorer og mulighet fra brukerne sin side til å se på dataen. Disse kommunene har en annen teknologileverandør enn Trondheim kommune.

Selv om forskerne tror at sensorer er veien å gå, påpekte forskerne også at det er viktig å få med nyansene her. Telefonsamtalen fører til økt trygghet og mestring for veldig mange og er mer lavterskel enn å vente i telefonkø hos fastlegen.

(...) så det er liksom sånne nyanser da som er ganske interessante. Men jeg tror at det med sensorer er absolutt veien å gå da. Så lenge man har – for alle vil jo ha ulike terskelverdier for hva som er riktig for dem. Og det er jo viktig at man tar inn da i en sånn tjeneste. (F1)

I følge F1 har fastlegene sagt at de kan synes det er vanskelig å ha ansvaret for å sette grenseverdier for sensormålingene – hva om de setter feil grenseverdier og brukeren blir syk? Det oppleves som et stort ansvar.

9.3.3 Tilbakemeldinger på prototypen

Forskerne mente at tidsvinduet for å gjennomføre en måling etter autentisering burde økes. Her kan man legge seg på industristandarder som for eksempel nettbanker bruker for hver økt etter at brukeren logger inn med BankID – 10-15 minutter. Om det korte tidsvinduet:

F1: «Og så fører det til så mye unødvendig stress, som igjen fører til at det tar lenger tid (...)».

F2: «(...) som igjen kan føre til en feilmåling. Hvis de blir stressa om de må rekke det (...)»

De sa at fingeravtrykksensor ikke er optimalt for alle brukere. Noen av diabetespasientene har nesten ikke fingeravtrykk igjen etter alle stikkene i fingeren. Andre autentiseringsmetoder må også undersøkes.

(...) hva tenker dere om frittstående løsninger sammenlignet med at det er tablet for eksempel? (...)

F2 svarte: «Jeg at for de brukerne som ikke har behov for den der rapporteringen med tablet daglig, så kan det være greit å ha kun den her, som en sånn dings du kan ha på nattbordet eller –» F1 sa videre: «(...) de som kanskje ikke trenger de her subjektive målingene, men de kunne hatt en sånn der "jeg har det i bra i dag". Ferdig med det.»

Vil du heller ha sensorene for seg selv (...) og så huben som en sånn liten boks (...) – eller ha sensoren innbakt? (...) Eller er det nødvendigvis ikke en motsetning

F1: «Det siste du sa. Jeg ser for meg, kall det et modulsystem, som er avtagbart. At du kan gjøre som du vil.»

F2: «Se for deg sånn legobrikker. At en legobrikke er pålogging (...)»

Forskerne så for seg at det beste kanskje kunne være å ha en hovedmodul som hadde autentisering slik at en kunne logge seg på der og ta i bruk alle de andre modulene rundt. Det må være greit å samle alt sammen og ta det med seg på en enkel måte.

Om brukervennlighet i prototypen:

F1: «(...) det ser jo enkelt og lekkert ut med lysende dioder og diskret knapp. (...) men samtidig ikke så veldig lett å skjønne hva som er hva. (...) Hvorfor blinker den nå?»

Som skrevet tidligere savnet forskerne en måte for brukeren å gjøre en subjektiv vurdering av egen helsetilstand i prototypen. Det er noe som må legges inn dersom noe lignende skal produksjonssettes.

9.3.4 Oppsummering og veien videre

Hvordan blir veien videre for Trondheim kommune nå? Hva tror dere blir resultatene av det prosjektet? (...) hva er fremtiden for avstandsoppfølging? Sånn i praksis.

F2 svarte at det er et stort ønske om å bygge alt ferdig. Det er målet både fra direktoratet og kommunen sitt ståsted. Samtidig må det noen endringer til for at det skal være nyttig og bærekraftig overtid med tanke på økonomi og andre ting.

F1 sa:

Jeg innbiller meg at noe av det som gjøres i Sarpsborg og for såvidt Oslo også, med at de har mer bruk av sensorer og også det at – selv om det varierer hvor mye brukerne selv ser på kurvene sine og sånn, men at de har mulighet til det og kan lære av det. Men også det å få inn, sånn som vi snakket om i sted, få egenbehandlingsplanen inn i systemet. Det vil jo ikke det her svare på i utgangspunktet. Så det er jo et minus vil jeg absolutt si. Sånn sett. Så det må løses på en eller annen måte. Trenger ikke å være mer komplisert enn at man ved en type måling som tilsier at man bør følge noen instruksjoner får beskjed om det. For det er jo det som mangler nå at når du kommer i gul sone, så skal du egentlig gjøre noe. Men du får ikke beskjed om det, annet enn på telefon. Og det er ikke nødvendigvis slik at du trenger det når det skal være en egenbehandling.

Brukerne må altså få mer informasjon tilbake fra systemet, gjerne umiddelbart. Brukeren vet ikke hva som skjer med det som sendes inn. F2 sa:

Det har vi også tilbakemeldinger om (...) i Trondheim så har du mulighet til å sende inn en fritekstkommentar. Og der er det jo en del som skriver noe inn, men så vet de jo ikke hva som skjer, fordi at de får ikke noe respons. De vet ikke liksom 'er det datamaskin som tar i mot det her, eller leser de det her eller' (...) så det ønsker de en respons på da (...)

F1: «for å ta den der dioden videre, så kunne det her vært tre dioder som lyste grønt, oransje og rødt. (...) men som viser at 'her har jeg faktisk tatt en måling som ikke var helt innenfor'.»

F2: «Men da må brukeren også vite hva man skal gjøre da hvis man ikke er innenfor – det må følge med en tjeneste.»

Kapittel 10

Diskusjon

Det er arkitekturen og basisteknologien som er det viktige i dette forskningsprosjektet. Basisteknologien ble satt inn i et praktisk problem i et komplekst domene. Sensorer til støtte for avstandsoppfølging er et godt eksempel på noe som kan kobles til den nye arkitekturen for tingenes internett, som beskrevet av Porter og Heppelmann (2014).

Prototypen ble laget for å si noe om denne basisteknologien med forskningsstrategien design og kreasjon. Case-studien av avstandsoppfølging i Trondheim kommune ble gjennomført for å øke kunnskapen om domenet og finne ut hvilke utfordringer kommunen sliter med. Det viste seg at kommunen hadde problemer med å få til tofaktorautentisering eller noe med tilsvarende grad av sikkerhet. Kommunen jobbet også med å få til sertifikater og distribusjon av sertifikater. Eksisterende skyteknologi har allerede noe av dette innebygget. AWS IoT har infrastruktur for å håndtere sertifikater, selv om distribusjonen av sertifikatene til enhetene må ordnes på egenhånd.

Prosjektet viste at det er enkelt å sette opp et standard grafsystem med målinger tilbake i tid på toppen av åpne kildekode-løsninger på veldig kort tid. Kommunen var imponert over hvor langt prosjektet hadde kommet på en måned med utvikling. Under evalueringen pekte kommunen på at de i dag har for dårlige verktøy til å se på måledata, og dermed ikke kan nyttiggjøre seg sensorteknologien. Leverandørene av tjenester til bruk i velferdsteknologi har et ansvar for å levere gode løsninger og ta i bruk ny teknologi, spesielt dersom de får være med på forsknings- og utviklingsprosjekt (FOU). Kommunen har selv sagt i intervju at de ønsker å være fremoverlente på teknologi og til stede på de plattformene brukerne er. Det er tydelig at de løsningene kommunen bruker i dag ikke ser så moderne ut og bruker modifisert hyllevare fra leverandøren.

Det hadde vært ønskelig om prosjektet hadde kommet enda lengre med implementasjonen av en større skyløsning med toveiskommunikasjon og mer tilbakemeldinger til brukeren for å løse flere av utfordringene kommunen og SINTEF pekte på. Dette kunne dannet et enda større bilde av egnetheten til skybasert IoT som teknologisk plattform for avstandsoppfølging. Allikevel er erfaringene at arkitekturen for sensoren fungerte veldig bra og at det var enkelt å komme i gang med. Dokumentasjonen til skyteknologien var god. Prosjektet sitter igjen med at plattformen var til å stole på og har høy tillit til at det fungerer.

MQTT ble brukt som kommunikasjonsprotokoll mellom prototype og skyløsning. På

en annen side kunne kanskje en mer tradisjonell løsning også fungert. MQTT-protokollen egner seg kanskje enda bedre for sensorer som sender data kontinuerlig. Brukeren sender 20 sekunder med data hver dag i bruksscenarioet som ble skissert i dette prosjektet. Det er ikke så veldig mye data. Den samme dataen kunne blitt sendt over HTTPS og en standard arkitektur som REST istedenfor. Istedenfor sertifikater for autentisering kunne en brukt tokens med så og så lang gyldighet som bevis på at enheten er logget inn.

I fremtiden kan det hende at det blir mer aktuelt å gå med sensorer over lengre tidsperioder dersom det blir vurdert som klinisk relevant for en bruker. I så fall vil AWS IoT være veldig skalerbar og pålitelig for å håndtere en økt mengde meldinger. Verdien av automatisk skalerbarhet, en lang rekke med mulige integrasjoner og at man slipper å ha egen infrastruktur må ikke undervurderes.

Selv om AWS IoT har god nytteverdi mellom klient og server, har ikke skyplattformen noe med kommunikasjonen fra sensor til klient. Her er det det Bluetooth-teknologien som råder og kommunen er prisgitt at leverandørene er tidlig ute med ny teknologi. I praksis vil nok ikke sikkerhetsrisikoen ved bruk av Bluetooth være så stor. Angriperne må være helt i nærheten for at det skal være teoretisk mulig å sette i gang et angrep.

10.1 Metodediskusjon

Starten av diskusjonskapittelet trakk mest fram det positive med valgene av metodikk i dette prosjektet. Det hadde vært nyttig med mer teknisk informasjon om hvordan HelsaMi+ fungerer fra leverandøren for å sammenligne, se hvor langt de har kommet sikkerhetsmessig og hva de tenker om fremtiden. Leverandøren Imatis ble kontaktet for et intervju, men de svarte ikke på henvendelsen. Kanskje mer burde blitt gjort for å snakke med en leverandør av tekniske løsninger til dette prosjektet. Det er en viktig komponent som mangler litt. Det er en annen leverandør i Oslo og Sarpsborg. De to leverandørene kunne blitt sammenlignet med tanke på hva de tilbyr av tjenester og hvordan de har bygget opp sin løsning mot avstandsoppfølging. Det kunne vært interessant å høre hva leverandørene tenker om denne arkitekturen for avstandsoppfølging, hva de identifiserer som de største utfordringene for en slik arkitektur og om de planlegger å gå over på en lignende arkitektur selv. Dette er noe som kan jobbes videre med i andre forskningsprosjekt. Mulighetene er absolutt til stede for å sette opp en lignende arkitektur fra bunnen av selv med egen infrastruktur.

Hele temaet er veldig stort, og det hadde kanskje vært lurere å snevret inn hvilke kvalitetskrav man skulle sette ekstra søkelys på, og hvordan disse kravene skulle vært testet slik at en kunne gjort et bedre dypdykk og snakket med eksperter innenfor det området. På den måten kunne en for eksempel sett nærmere på problematikken rundt tofaktorautentisering og snakket med eksperter på det.

10.2 Validitet

10.2.1 Bekreftbarhet og pålitelighet

Tidligere i kapitlet ble fremgangsmåten til studiet presentert. All kode ligger åpent tilgjengelig på Internett slik at hvem som helst kan tittle på den. Transkriberinger av intervjuene er tilgjengelige ved forespørsel. Tidligere ble implementasjonen av prototypen beskrevet i så god detalj at det er mulig å lage noe tilsvarende kun ved å lese i teksten og se på koden.

10.2.2 Troverdighet

Alle intervjuene med Trondheim kommune må tolkes med det faktum at prosjektledere og ansatte har en tilknytning som gjør at de ønsker at avstandsoppfølgingprosjektet skal lykkes i bakhodet. Aktører direkte involvert i et prosjekt er mottakelige for bekreftelestendenser der man søker etter å bekrefte noe man allerede tror. Selv om SINTEF er en tredjepart som bistår Trondheim kommune, er de også mye involvert i prosjektet. Det er mulig at forskningen kunne blitt enda mer troverdig med en kritisk tredjepart uten direkte tilknytning til velferdsteknologiprojektet i Trondheim – gjerne en stemme som ikke er udelt positiv til avstandsoppfølging som teknologi for velferd. Fordelen med å snakke med personer som jobber med avstandsoppfølging i Trondheim kommune, er at de har veldig god kunnskap om temaet og muligheten til å sammenligne løsningen som presenteres i denne forskningen med noe som eksisterer i praksis i dag. De er en troverdig stemme i avstandsoppfølgingdebatten.

I fokusgrupper, brukbarhetstester og intervjuer har deltakerne en tendens til å ville blidgjøre den som har laget produktet (i dette tilfellet utvikleren), dersom det er samme person som gjennomfører testene. Dette er en veldig menneskelig egenskap. SINTEF var en uavhengig tredjepart i intervjuer med brukere av HelsaMi+ for å få brukerne til å snakke mer fritt. I dette prosjektet har samme person utviklet og gjennomført intervjuene, men det virker som intervjudeltakerne har snakket veldig fritt og vært åpne i samtalen.

Transkriberingen av store deler av de tre gjennomførte intervjuene ble gjort for å øke troverdigheten og motvirke egen tendens til å tolke alt på en spesiell måte. Det gjorde at man fikk konteksten til hele intervjuet. Et spørsmål er uansett om man med utviklerblikket som er perspektivet i dette prosjektet er i stand til å tolke alt som blir sagt. Noe av det må nødvendigvis bli subjektivt.

10.2.3 Overførbarhet

Spørsmålet som må stilles er om funnene og arbeidet gjort i dette prosjektet kan overføres til avstandsoppfølging som generelt tilfelle. Trondheim kommune brukes som eksempel, og konklusjonen vil generalisere med kun dette tilfellet som bakgrunn. Er

det faktisk mulig å finne noen implikasjoner for utviklingen av skybasert IoT som teknologisk plattform for avstandsoppfølging? Hvilke resonnementer og argumenter er i så fall disse resonnementene fundert i?

Trondheim kommune har ikke så mange brukere med sensorer, og SINTEF trakk fram at andre kommuner har kommet lengre på det området. Uansett vil det være rimelig å anta at andre kommuner og leverandører vil ha noe av den samme problematikken som Trondheim med sikkerhet, sertifikater, autentisering, personvern og skalering.

10.3 Avgrensning av forskningen

Flere av utfordringene som ble identifisert i dette prosjektet kunne vært tema for andre og mer detaljerte undersøkelser. Det ble allikevel vurdert som nyttig å ta med helheten i avstandsoppfølgingsprosjektet, selv om kanskje ikke alt var direkte relevant for forskningsspørsmålene. I virkeligheten er alt komplekst, og mange ting spiller inn på arkitekturen til en løsning. Det er krav fra flere forskjellige hold som en må ta hensyn til og forholde seg til.

Det har vært et bevisst valg å distansere seg fra flere av disse kravene for å ikke drukne i informasjon, slik at søkelyset kunne ligge på IoT som teknologisk plattform med arkitektur og programvareutvikling. Flere ting kommunen og SINTEF nevner kunne blitt nøyere undersøkt, men her har bare disse instansene vært sannhetsvitner for problematikk som dukker opp i praksis når dette skal ut i produksjon. Selv om ikke brukbarhet har vært viktig for prosjektet, er tilbakemeldingene på prototypen når det gjelder brukskvalitet tatt med allikevel for å få med det store bildet.

Slik det ble forklart i delkapittel 1.4, har ikke de kliniske vurderingene man kan gjøre med sensordataene vært drøftet i prosjektet.

Kapittel 11

Konklusjon

11.1 Forskningsspørsmål 1

Hvordan gjøres avstandsoppfølging av kronisk syke i prosjektet HelsaMi+ i Trondheim, og hvilke utfordringer ser prosjektledelsen når det gjelder denne type teknologi/tjenester?

I Trondheim kommune har ca. 70 brukere fått avstandsoppfølging hjemme. Det foregår med et Android-nettbrett brukerne får utlevert. Dette nettbrettet har en applikasjon der brukerne kan svare på hvordan dagsformen er. Spørsmålene er tilpasset diagnosen til brukeren. I samråd med fastlegen har brukeren en egenbehandlingsplan på papir som iverksettes på bakgrunn av helsetilstanden. Fastlegen har ikke direkte tilgang til avstandsoppfølgingssystemet per i dag. Koordinasjon skjer via helsevakta i Trondheim kommune. Seks brukere har fått sensorer (pulsoksymeter, vekt, blodtryksmåler) de kan bruke til målinger. Dette gjøres også via applikasjonen. Helsevakta får tilgang på spotmålingen og svarene på spørsmålene i en adminapplikasjon. Tilbakemeldingene på prosjektet i seg selv er veldig positive.

Prosjektledelsen har hatt utfordringer med stabiliteten til sensorløsningen. De ser også utfordringer med personvern og å la brukeren få muligheten til å administrere og se mer av sin egen data. Kommunen jobber med å få til sertifikater på nettbrettene, men det er ikke så lett. En annen erfaring kommunen har gjort seg, er at brukerne er veldig forskjellige. Kommunen ønsker å være tidlig ute med ny teknologi, men det viser seg å ikke være så enkelt å få til i praksis.

11.2 Forskningsspørsmål 2

Hva er en mulig arkitektur for realisering av skybasert IoT for avstandsoppfølging av kronisk syke som adresserer utfordringene fra FS1?

En mulig arkitektur for realisering av skybasert IoT for avstandsoppfølging er en klient-server-arkitektur basert på arbeidet som er gjort i dette prosjektet med noen små modifikasjoner. Arkitekturen har en hub som alle sensorene kobler seg på. Sensorene kom-

muniserer med huben med Bluetooth 4.2 eller nyere. Alle sensorer skal bondes med huben slik at sensorene ikke må pares på nytt hver gang det skal sendes data. En mulig strategi for autentisering er å bruke et klientsidesertifikat, og at dette sertifikatet er unikt for hver bruker av løsningen. Dette er mest egnet for en backendløsning som håndterer sertifikater, for eksempel AWS IoT, der MQTT over TLS med egne sertifikater støttes. Det er helt klart mulig å benytte andre metoder for autentisering slik at brukeren slipper å skrive inn passord hele tiden. Transport av data skal uansett alltid skje med ende-til-ende-kryptering (TLS).

Arkitekturen har en regelbasert backend der dataen lagres fortløpende og prosesseres i sanntid. Denne arkitekturen løser ikke problemer knyttet til personvern og datalagring. Det er noe som må implementeres på toppen av skyløsningen. AWS gir allikevel mange gode verktøy ut av boksen for å få til dette. Det kan understøtte mange av de tjenesteutfordringene som HelsaMi+ opplever i dag.

11.3 Forskningsspørsmål 3

Hvordan vurderer domeneeksperter innen velferdsteknologi frittstående skybaserte IoT-løsninger for avstandsoppfølging av kronisk syke?

Prosjektledere i Trondheim kommune mente at prototypen hadde tilstrekkelig brukskvalitet til å ruller ut til brukere. De mente imidlertid at det kanskje var bedre at sensor og hub var adskilt og ikke koblet sammen til ett produkt. Prisen på prototypen ble vurdert som rimelig. Indikering av at en måling er ferdig på skjermen til pulsoksymeteret var noe kommunen ikke hadde med i sin løsning.

Forskere ved SINTEF var enige i at sensor og hub heller kunne vært adskilt, og trakk fram at prototypen manglet muligheten til å svare på hvordan dagsformen er. De så for seg at man kunne utvide prototypen med flere moduler, for eksempel en modul med muligheten til å svare på spørsmål.

11.4 Forskningsspørsmål 4

Basert på funnene fra FS1, FS2 og FS3, hva er implikasjonene for utviklingen av skybasert IoT som teknologisk plattform for avstandsoppfølging av kronisk syke?

Konklusjonen til dette forskningsspørsmålet deles opp i avsnittene sikkerhet og autentisering, personvern, utviklingsomgivelser og pris.

11.4.1 Sikkerhet og autentisering

- Benytt ende-til-ende-kryptering (TLS).
- Autentiser brukeren med sertifikater eller tokens og benytt tofaktorautentisering.

- En moderne skytjeneste som AWS med sin løsning AWS IoT gir god innebygd sikkerhet som standard med TLS og sertifikater. Dette gjelder fra klient til skyen, men i en del tilfeller vil man ha et sikkerhetsproblem mellom klient og sensor.
- Bruk den nyeste utgaven av Bluetooth-standarder mellom klient og sensor som sensoren din støtter. Helst utgave 4.2 eller nyere. Hvis du må bruke en eldre versjon må du være observant på hvilke angrepsskemaer som finnes og minimere risikoen for sikkerhetsbrudd. Les nøye hva beste praksis er.

11.4.2 Personvern

- Helsedata lagres primært i Norge i dag. Dette utelukker skyløsninger som har mange nødvendige sikkerhetsprimitiver innebygd som minstestandard, selv om personopplysningsloven åpner for å flytte data til utlandet dersom visse kriterier er oppfylt. I skrivende stund tilbyr verken AWS eller Microsoft skyløsninger med lagring i Norge, men dette er et rent praktisk problem. Det har ikke noe med teknologien å gjøre.
- Med EUs nye personvernsforordning får brukeren enda mer rett over sin egen data. Brukeren må kunne se hva som er lagret, slette det og tilbyderen må kunne argumentere med hvorfor dataen må lagres. Dette kompliserer utviklingen av tjenester til bruk i avstandsoppfølging både juridisk og teknisk. Gode løsninger for brukere og administratorer må utvikles for å håndtere dette på en god måte. En moderne skytjeneste kan avlaste disse utfordringene ved å gi utviklere og administratorer gode verktøy og et helhetlig økosystem for integrasjon slik at implementasjonen blir enklere.

11.4.3 Utviklingsomgivelser

- AWS IoT var enkelt å komme i gang med for en utvikler og har mange gode verktøy som egner seg for avstandsoppfølging av kronisk syke.
- Plattformen har flere ulike grensesnitt for tilkobling og integreringen mot andre tjenester i økosystemet er veldig god.
- Kommunen var imponert over at prototypen fungerte og hvor kort tid det hadde tatt å utvikle den. Skyplattformen gjorde at det var mulig å komme raskt i gang slik at det var mulig å bruke mer tid på prototypen på klientsiden.

11.4.4 Pris og administrasjon av utstyr

- Det er en stor kostnad for kommunen å administrere og vedlikeholde nettbrett.
- Direktoratet for e-helse anbefaler at kommunen skal eie hele verdikjeden, og kommunen eier dermed SIM-kort og abonnement for tilkobling til nettet. Det er potensiale for å se på effektivisering her dersom brukeren kan ha utstyret selv,

eller bruke 3G som ekstrametode for tilkobling hvis det ikke går an å koble på WIFI.

- Det kan være billigere for kommunen å være til stede på de plattformene brukeren er fra før av. Det kan føre til lavere administrasjonskostnader. Kommunen trenger i så fall kun å dele ut sensorer.

11.5 Refleksjoner rundt overordnet forskningsspørsmål

Hvor egnet er skybasert IoT som teknologisk plattform for avstandsoppfølging av kronisk syke?

Som utvikler var det enkelt å jobbe med AWS IoT som teknologisk plattform for avstandsoppfølging av kronisk syke. Denne plattformen bygger på det som har blitt en ganske standard arkitektur for IoT med innebygget infrastruktur for å håndtere ting og hvordan tingene kobles til skyløsningen på en enkel og sikker måte. Prosjektet hadde stor tillit til plattformen og det var behagelig at sikkerhet var standard og med fra starten av. Det førte til at prosjektet kunne konsentere seg om å jobbe med prototypen som tok mesteparten av utviklingstiden. Det tok svært kort tid å sette opp AWS IoT.

AWS IoT er veldig godt dokumentert og kan konfigureres og brukes med nettside, kommandolinjeverktøy og flere forskjellige programmeringsspråk. For utviklere er det behagelig å ha mange verktøy i verktøyskassen, uten at man må lage alle verktøy fra bunnen av selv.

AWS IoT er en proprietær og kommersiell tjeneste som baserer seg på ganske standard arkitektur. Ulempen med å velge en slik leverandør er at man kan bli ganske låst til leverandøren og økosystemet til leverandøren. Det er ikke helt klart om det er enkelt å bytte mellom ulike leverandører. Kanskje er det mulig å skrive abstraksjoner som gjør at den underliggende plattformen kan byttes ut. Fordelen er at man aldri trenger å tenke på skaleringen til tjenesten og fysisk driftssikkerhet. Det er en styrke for plattformen at selv om den er proprietær, så baserer den seg på åpne protokoller (MQTT) der det er mulig. AWS IoT vil være svært godt egnet dersom kommuner vil samle inn enda mer data i fremtiden, for eksempel hvis en bruker går med en sensor over lengre perioder hver dag. Hovedkonklusjonen basert på erfaringene i dette forskningsprosjektet, er det mange gode grunner til å bygge framtidige løsninger på lignende plattformer. Ingen store ulemper har blitt identifisert.

11.6 Videre arbeid

Det er mye som kan jobbes med videre innenfor avstandsoppfølging. Spesielt lovende virker det å få til en høyere grad av toveiskommunikasjon. Brukeren kan for eksempel få et varsel når det er på tide å gjøre en måling. Det er mulig å bake inn kommunikasjon med Helsevakta inne i applikasjonen, enten med chat eller med videosamtale. Sanntidsanalyse av data basert på grenseverdier hadde vært teknologisk mulig å få til

uten for mye styr. Helsevakta kunne fått opp et varsel i sin adminapplikasjon. Om brukeren selv skulle fått opp et slikt varsel måtte i så fall blitt individuelt tilpasset. Som erfaringene til Trondheim kommune viser har ikke alle så godt av å vite for mye om sin egen helsetilstand. SINTEF sa at det kunne være mye positivt i at en bruker blir kjent med sine egne målinger. Analyse av data tilbake i tid kunne også vært gjort. Her peker fastlegen på at alt ikke er så klart og at det trengs mer forskning og flere erfaringer.

Egenbehandlingsplanen til brukeren må bakes inn i alle løsninger slik at brukeren får et bedre forhold til den. Basert på hva brukeren rapporterer inn av dagsform og sensordata må det komme opp hvilke tiltak brukeren skal iverksette. Dette vil øke egenmestringen til brukeren.

Det er også potensiale i overvåke alle sensorene på en bedre måte, slik at en kan følge med på hvor mye batteri som er igjen og hva kvaliteten på sensordataen er. Her må en benytte alle de mulighetene sensoren gir for integrasjon, som for eksempel checkmark når brukeren er ferdig med en måling. Teknologi som device shadow og device twins (delkapittel 7.2.2 og 7.2.3) kan brukes til å holde styr på tilstanden og manipulere tilstanden fra ulike applikasjoner. Kommunen ønsker å være tidlig ute med ny teknologi, men det viser seg å ikke være så enkelt å få til i praksis.

Mye kan gjøres i utviklingen av applikasjoner på eksisterende plattformer (Android, iOS, web). Disse applikasjonene bør helst være native for best mulig ytelse og brukeropplevelse. Her kan en med fordel nyttiggjøre seg av ny teknologi og nye løsninger laget fra bunnen av for å få til dette på en bedre måte.

Referanser

- Amazon Web Services. (2015, 8. oktober). Announcing AWS IoT, now in beta. Hentet 12. mai 2017, fra <https://aws.amazon.com/about-aws/whats-new/2015/10/announcing-aws-iot-now-in-beta/>
- Amazon Web Services. (2016a). AWS IoT Device SDK. Hentet 6. november 2016, fra <https://aws.amazon.com/iot/sdk/>
- Amazon Web Services. (2016b). How the AWS IoT Platform Works. Hentet 6. november 2016, fra <https://aws.amazon.com/iot/how-it-works/>
- Austad, H. O., Liverud, A. E., Strisland, F., Ausen, D. & Reitan, J. (2016). Sensorer til støtte for avstandsoppfølging av kronisk syke. *SINTEF Rapport*.
- Azure. (2017, 24. mars). Azure and Internet of Things. Hentet 18. juni 2017, fra <https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-what-is-azure-iot>
- MQTT Version 3.1.1. (2014, 29. oktober). Hentet 4. desember 2016, fra <http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.html>
- Bass, L., Clements, P. & Kazman, R. (2013). *Software Architecture in Practice* (3rd). Addison-Wesley.
- Berntsen, J. (2016). COMPLIANCE: GENERAL DATA PROTECTION REGULATION. Hentet 17. desember 2016, fra <http://docplayer.me/24846608-Compliance-general-data-protection-regulation-johnny-berntsen-iso-iec-master-knowit.html>
- Bluetooth.com. (udatert). Generic Attributes (GATT) and the Generic Attribute Profile. Hentet 17. juni 2016, fra <https://www.bluetooth.com/specifications/generic-attributes-overview>
- Bon, M. (2015, 25. oktober). A Basic Introduction to BLE Security. Hentet 12. mai 2017, fra <https://eewiki.net/display/Wireless/A+Basic+Introduction+to+BLE+Security>
- Council of European Union. (2016, 27. april). Council regulation (EU) 2016/679. Hentet 17. desember 2016, fra <http://eur-lex.europa.eu/legal-content/en/TXT/?uri=CELEX:32016R0679>
- Datatilsynet. (2014, 19. august). Velferdsteknologi. Hentet 3. november 2016, fra <https://www.datatilsynet.no/Sektor/Helse-og-omsorg/Velferdsteknologi/>
- Datatilsynet. (2015, 15. mars). Innebygd personvern. Hentet 3. november 2016, fra <https://www.datatilsynet.no/Teknologi/Innebygd-personvern/>
- Datatilsynet. (2016, 13. juli). Overføring av personopplysninger til utlandet. Hentet 12. juni 2017, fra <https://www.datatilsynet.no/Regelverk/Internasjonalt/Overfoering/>
- Direktoratet for e-helse. (2015). Hjertesvikt - symptomer og behandling. Hentet 3. november 2016, fra <https://helsenorge.no/sykdom/hjerte-og-kar/hjertesvikt#Symptomer-p%C3%A5-hjertesvikt>

- Direktoratet for e-helse. (2016, 31. mai). m-Helse. Hentet 30. mai 2017, fra <https://ehelse.no/m-helse>
- Drivenes, A. (2017, 13. juli). Video av pulsoksymeter til masteroppgave på NTNU. Hentet 13. juli 2017, fra <https://www.youtube.com/watch?v=ALizPkFko9U>
- Helse- og omsorgsdepartementet. (2013, 19. april). *Morgendagens omsorg* (St.meld. Nr. 29 2012-2013). Hentet 12. desember 2016, fra <https://www.regjeringen.no/no/dokumenter/meld-st-29-20122013/id723252/>
- Hukkelås, I. M. (2008). *Brukersentrert utvikling av IT-støtte for oppfølging av KOLS pasienter – en metodesammenligning* (Masteroppgave, Norges teknisk-naturvitenskapelige universitet, Trondheim, Norge).
- International Organization for Standardization. (2010). *Ergonomi for samhandling mellom menneske og system - Del 210: Menneskeorientert design for interaktive systemer* (ISO/DIS Nr. 9421-210:2010). Hentet 7. mai 2016, fra <http://www.standard.no/no/Nettbutikk/produktkatalogen/Produktpresentasjon/?ProductID=466417>
- Intro International og Arkitektur-og designhøgskolen i Oslo. (2011). *Velferdsteknologi i sentrum – Innføring av velferdsteknologi i sentrumsbydelene i Oslo, delleveranse 2 av 2 (Vis-rapporten)*. Oslo. Hentet 3. mai 2017, fra <https://helsedirektoratet.no/velferdsteknologi#-rapporter-og-utredninger>
- ITU. (2012, 15. juni). ITU-T Y.4000/Y.2060 (06/2012). Hentet 15. mai 2017, fra <http://www.itu.int/ITU-T/recommendations/rec.aspx?rec=y.2060>
- Izrailevsky, Y. (2016, 11. februar). Completing the Netflix Cloud Migration. Hentet 12. desember 2016, fra <https://media.netflix.com/en/company-blog/completing-the-netflix-cloud-migration>
- Microsoft Azure. (2015, 9. november). Microsoft Azure IoT Suite – Connecting Your Things to the Cloud. Hentet 19. mai 2017, fra <https://azure.microsoft.com/en-us/blog/microsoft-azure-iot-suite-connecting-your-things-to-the-cloud/>
- Nasjonalforeningen for folkehelsen. (2015, 18. desember). Hjertesvikt. Hentet 22. mai 2017, fra <http://www.who.int/chp/en/>
- Nonin Medical, Inc. (2016). *Model 3230 Bluetooth Smart Pulse Oximeter Integration Guide and Specifications*.
- NOU 2011: 11. (2011). *Innovasjon i omsorg*. Oslo: Departementenes servicesenter, Informasjonsforvaltning. Hentet 3. november 2016, fra <https://www.regjeringen.no/no/no/dokumenter/nou-2011-11/id646812/>
- Oates, B. J. (2006). *Researching Information Systems and Computing*. Sage Publications.
- PCHA. (2016). CONTINUA DESIGN GUIDELINES: FLEXIBLE IMPLEMENTATION FRAMEWORK FOR AUTHENTIC INTEROPERABILITY. Hentet 12. desember 2016, fra <http://www.pchalliance.org/continua/products/design-guidelines>
- Peterson, A. (2016, 21. oktober). Internet of Things compounded Friday's hack of major websites. *The Washington Post*. Hentet 14. november 2016, fra <https://www.washingtonpost.com/news/the-switch/wp/2016/10/21/someone-attacked-a-major-part-of-the-internets-infrastructure/>
- Porter, M. E. & Heppelmann, J. E. (2014). How Smart, Connected Products Are Transforming Competition. *Harvard Business Review*. Hentet 15. november 2016, fra <https://hbr.org/2014/11/how-smart-connected-products-are-transforming-competition>
- Porter, M. E. & Heppelmann, J. E. (2015). How Smart, Connected Products Are Transforming Companies. *Harvard Business Review*. Hentet 15. november 2016, fra

- <https://hbr.org/2015/10/how-smart-connected-products-are-transforming-companies>
- raspberrypi.org. (udatert). GPIO: RASPBERRY PI MODELS A AND B. Hentet 18. juni 2017, fra <https://www.raspberrypi.org/documentation/usage/gpio/>
- Regjeringen. (2014a, 18. desember). Standardisering av velferdsteknologi med Continua. Hentet 13. desember 2016, fra <https://www.regjeringen.no/no/aktuelt/standardisering-av-velferdsteknologi-med-continua/id2356200/>
- Regjeringen. (2014b, 24. november). Tema: Sykehus. Ofte stilte spørsmål. Hentet 25. mai 2017, fra <https://www.regjeringen.no/no/tema/helse-og-omsorg/sykehus/innsikt/nokkeltall-og-fakta---ny/ofte-stilte-sporsmal/id534086/>
- Regjeringen. (2015, 4. juni). Nye digitale behandlingsmetoder for personer med kroniske sykdommer. Hentet 9. juni 2017, fra <https://www.regjeringen.no/no/aktuelt/nye-digitale-behandlingsmetoder-for-personer-med-kroniske-sykdommer/id2414937/>
- Rojahn, K., Laplante, S., Sloand, J., Main, C., Ibrahim, A., Wild, J., ... Johnson, K. I. (2016). Remote monitoring of chronic diseases: a landscape assessment of policies in four European countries. *PloS one*, 11(5), e0155738. Hentet 13. mai 2017, fra <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4873167/>
- Schneier, B. (2016, 10. november). Regulation of the Internet of Things. Hentet 14. november 2016, fra https://www.schneier.com/blog/archives/2016/11/regulation_of_t.html
- Sødal, A.-B., Klungerbo, B. & Oldervoll, L. (2016). Åttiåringenes teknologibruk i Trondheim 2015. *Ergoterapeuten*, (4). Hentet 6. juni 2017, fra <http://www.ergoterapeuten.no/fag-og-vitenskap/fagartikler>
- Solberg, Ø. (2009). *Alternativer for registrering av pasientopplysninger i hjemmetjenesten – En sammenligning av tre teknologier* (Masteroppgave, Norges teknisk-naturvitenskapelige universitet, Trondheim, Norge).
- sparkfun.com. (udatert). Serial Communication. Hentet 19. juni 2017, fra <https://learn.sparkfun.com/tutorials/serial-communication>
- Synergy Research Group. (2017, 2. februar). Amazon Leads; Microsoft, IBM & Google Chase; Others Trail. Hentet 1. juni 2017, fra <https://www.srgresearch.com/articles/microsoft-google-and-ibm-charge-public-cloud-expense-smaller-providers>
- Trondheim kommune. (2014, 17. oktober). Helsefremmende og forebyggende hjemmebesøk ved 80 år. Hentet 6. juni 2017, fra <https://www.trondheim.kommune.no/content/1117726214/Helsefremmende-og-forebyggende-hjemmebesok-ved-80-ar>
- Trondheim kommune. (2017, 22. mars). Avstandsoppfølging av personer med kronisk sykdom: HelsaMi+. Hentet 10. juni 2017, fra <https://www.trondheim.kommune.no/content/1117753344/Avstandsoppfolgning-av-personer-med-kronisk-sykdom:-HelsaMi+>
- Weber, R. H. & Weber, R. (2010). *Internet of Things: Legal Perspectives* (2010 edition). Springer.
- Wikipedia. (2016). General Data Protection Regulation — Wikipedia, The Free Encyclopedia. Hentet 17. desember 2016, fra https://en.wikipedia.org/w/index.php?title=General_Data_Protection_Regulation&oldid=754575179

World Health Organization. (udatert). Chronic diseases and health promotion. Hentet 25. mai 2017, fra <http://www.who.int/chp/en/>

Zeifman, I., Bekerman, D. & Herzberg, B. (2016, 26. oktober). Breaking Down Mirai: An IoT DDoS Botnet Analysis. Hentet 15. november 2016, fra <https://www.incapsula.com/blog/malware-analysis-mirai-ddos-botnet.html>

Tillegg A

Kode

A.1 Hovedapplikasjon (smart-pulse-oximeter)

Link til Github-repo med hele applikasjonen: <https://github.com/andybb/smart-pulse-oximeter>.

```
1  const PulseOximeter = require('nonin-3230-ble');
2  const awsIot = require('aws-iot-device-sdk');
3  const { Button, RGBLed, Led } = require('pigpio-components');
4  const { enrollFingerAndRetrieveTemplate,
5    setFingerprintTemplateAndVerify,
6    blinkingFingerprintLed } = require('./fingerprint');
7  const { awsConfig } = require('./config');
8  const { delay } = require('./utils');
9  const logger = require('./logger');
10
11 const NUMBER_OF_SENSOR_READINGS = 20;
12
13 /* Noble attaches a listener that interrupts the GPIO cleanup.
14  * The Noble listener does not do anything else than
15  * invoking process.exit(1) if it is the last listener, */
16 process.removeListener('SIGINT', process.listeners('SIGINT')[0]);
17
18 const device = awsIot.device(awsConfig);
19 const button = new Button({ gpio: 24, isPullup: true });
20 const leftLed = new RGBLed({ red: 25, green: 8, blue: 7 });
21 const rightLed = new RGBLed({ red: 17, green: 27, blue: 22 });
22 const oximeterLed = new Led(12);
23
24 device
25   .on('connect', () => {
26     logger.log('debug', 'Connected to AWS');
27     device.subscribe('oximetry');
28     leftLed.color('yellow').strobe(2000);
29     rightLed.stop();
30   })
31   .on('error', (err) => logger.error(err))
32   .on('message', (topic, payload) => logger.log('debug', 'Received
33     message', topic, payload));
```

```
32
33
34 const showErrorLeds = () => {
35   leftLed.stop().color('red').on();
36   rightLed.stop().color('red').on();
37   delay(4000).then(() => {
38     rightLed.stop();
39     leftLed.stop().color('orange').strobe(2000);
40   });
41 };
42
43 const startMeasurement = () => {
44   logger.log('debug', 'Start measurement');
45   leftLed.color('green').stop().on();
46   setFingerprintTemplateAndVerify()
47     .then(() => {
48     logger.log('debug', 'Fingerprint verified');
49     rightLed.stop().color('green').on();
50     oximeterLed.setValue(255);
51     let timer;
52     let counter = 0;
53
54     const onDiscover = ((pulseOximeter) => {
55       logger.log('debug', 'Pulse oximeter discovered');
56       pulseOximeter.connectAndSetup((error) => {
57         if (error) {
58           logger.error(error);
59         }
60       });
61
62       pulseOximeter
63         .on('data', (data) => {
64           logger.log('info', 'Pulse oximeter data', data);
65           counter += 1;
66           device.publish('oximetry', JSON.stringify(data));
67           if (counter === NUMBER_OF_SENSOR_READINGS) {
68             logger.log('debug', 'Finished measurement. Cleaning up');
69             leftLed.stop().color('yellow').strobe(2000);
70             rightLed.stop();
71             oximeterLed.off();
72             clearTimeout(timer);
73             pulseOximeter.completeMeasurement();
74           }
75         })
76         .on('disconnect', () => {
77           logger.log('debug', 'Pulse oximeter disconnect');
78           if (counter < NUMBER_OF_SENSOR_READINGS) {
79             logger.log('debug', 'Pulse oximeter disconnect before
80               completion');
81             oximeterLed.off();
82             showErrorLeds();
83             clearTimeout(timer);
84           }
85         });
86     });
```

```
87 |
88 |     PulseOximeter.discover(onDiscover);
89 |
90 |     timer = setTimeout(() => {
91 |         if (counter < NUMBER_OF_SENSOR_READINGS) {
92 |             logger.log('debug', 'Pulse oximeter disconnect before
93 |                 completion');
94 |             oximeterLed.off();
95 |             rightLed.off();
96 |             leftLed.color('yellow').strobe(2000);
97 |         }
98 |         logger.log('debug', 'Pulse oximeter: Stopping discover');
99 |         PulseOximeter.stopDiscover(onDiscover);
100 |     }, 120000);
101 |
102 |     },
103 |     (error) => {
104 |         logger.error(error);
105 |         showErrorLeds()
106 |     });
107 |
108 | button.on('click', () => {
109 |     startMeasurement();
110 | });
111 |
112 | if (process.env.NODE_ENV === "development") {
113 |     logger.log('debug', 'Environment: development');
114 |
115 |     button.on('long press', () => {
116 |         logger.log('debug', 'Button: Long press');
117 |         leftLed.color('green').on();
118 |         enrollFingerAndRetrieveTemplate()
119 |             .then(() => {
120 |                 logger.log('debug', 'Fingerprint stored');
121 |                 rightLed.stop().color('green').on();
122 |                 delay(4000).then(() => {
123 |                     leftLed.color('yellow').strobe(2000);
124 |                     rightLed.stop();
125 |                 });
126 |             }, (error) => {
127 |                 logger.error(error);
128 |                 showErrorLeds()
129 |             }
130 |         );
131 |     });
132 | }
```

A.2 pigpio-components

Link til Github-repo med kode: <https://github.com/andybb/pigpio-components>.

```

1  const EventEmitter = require('events');
2  const { Gpio } = require('pigpio');
3
4  class Button extends EventEmitter {
5    constructor(options) {
6      super();
7      this._button = new Gpio(options.gpio, {
8        mode: Gpio.INPUT,
9        pullUpDown: options.isPullup ? Gpio.PUD_UP : Gpio.PUD_DOWN,
10       edge: Gpio.EITHER_EDGE
11     });
12     this._timer = null;
13     this._onInterrupt();
14   }
15
16   _onInterrupt() {
17     let intervals = 0;
18     this._button.on('interrupt', (value) => {
19       if (value === 0) {
20         this._timer = setInterval(() => {
21           intervals++;
22         }, 500);
23       }
24       else if (value === 1 && intervals === 0) {
25         clearInterval(this._timer);
26         intervals = 0;
27         this.emit('click');
28       }
29       else if (value === 1 && intervals >= 4) {
30         clearInterval(this._timer);
31         intervals = 0;
32         this.emit('long press');
33       }
34       else {
35         if (this._timer) {
36           clearInterval(this._timer);
37           intervals = 0;
38         }
39       }
40     });
41   }
42 }
43 };
44
45 module.exports = Button;

```

Kodesnutt A.2: pigpio-components: Button class

```

1  const { Gpio } = require('pigpio');

```

```
2 const Color = require('color');
3
4 class Led {
5   constructor(gpio) {
6     this._led = new Gpio(gpio, { mode: Gpio.OUTPUT });
7     process.on('SIGINT', () => {
8       this.off();
9       setTimeout(() => process.exit(1), 100);
10    });
11  }
12
13  setValue(val) {
14    if (val >= 0 && val < 256) {
15      this._led.pwmWrite(val);
16    }
17    else {
18      throw new Error('Value must be between 0 and 255');
19    }
20  }
21
22  on() {
23    this.setValue(255);
24  }
25
26  off() {
27    this._led.digitalWrite(0);
28  }
29 };
30
31 class RGBLed {
32   constructor(ports) {
33     this._leds =
34       [new Led(ports.red), new Led(ports.green), new Led(ports.blue)];
35     this._timer = null;
36     this._rgb = Color.rgb(255, 255, 255).rgb().array();
37     process.on('SIGINT', () => {
38       this.stop();
39       // Wait until other LEDs are cleaned up
40       setTimeout(() => process.exit(1), 100);
41     });
42   }
43
44   color(color) {
45     this._rgb = Color(color).rgb().array();
46     return this;
47   }
48
49   on() {
50     this._leds.forEach((led, index) => led.setValue(this._rgb[index]));
51     return this;
52   }
53
54   off() {
55     this._leds.forEach((led) => led.off());
56     return this;
57   }
}
```

```
58
59 stop() {
60     this.off();
61     if (this._timer) {
62         clearInterval(this._timer);
63     }
64     return this;
65 }
66
67 strobe(interval = 1000) {
68     if (this._timer) {
69         clearInterval(this._timer);
70     }
71
72     let on = true;
73     this.on();
74     this._timer = setInterval(() => {
75         if (on) {
76             this.off();
77             on = false;
78         }
79         else {
80             this.on();
81             on = true;
82         }
83     }, interval);
84     return this;
85 }
86
87 pulse(speed = 100) {
88     if (this._timer) {
89         clearInterval(this._timer);
90     }
91
92     let r = 0, g = 0, b = 0;
93     this._timer = setInterval(() => {
94         this._leds[0].setValue(r);
95         this._leds[1].setValue(g);
96         this._leds[2].setValue(b);
97         r += 5;
98         g += 7;
99         b += 9;
100     if (r > 255) {
101         r -= 255;
102     }
103     if (g > 255) {
104         g -= 255;
105     }
106     if (b > 255) {
107         b -= 255;
108     }
109     }, speed);
110     return this;
111 }
112
113 rainbow(speed = 500) {
```

```

114   if (this._timer) {
115     clearInterval(this._timer);
116   }
117   const colors = ['red', 'orange', 'yellow', 'green', 'blue', 'indigo',
118                 ', 'violet'];
119   let index = 0;
120   this._timer = setInterval(() => {
121     this.color(colors[index]);
122     this.on();
123     index++;
124     if (index > colors.length - 1) {
125       index = 0;
126     }
127   }, speed);
128   return this;
129 }
130 }
131
132 module.exports = { RGBLed, Led };

```

Kodesnutt A.3: pigpio-components: Led classes

A.3 nonin-3230-ble

Link til Github-repo med kode: <https://github.com/andybb/nonin-3230-ble>.

```

1  const NobleDevice = require('noble-device');
2
3  const SERVICE_UUID = '46a970e00d5f11e28b5e0002a5d5c51b';
4  const NOTIFY_CHAR = '0aad7ea00d6011e28e3c0002a5d5c51b';
5
6  const Nonin3230 = function(peripheral) {
7    NobleDevice.call(this, peripheral);
8  };
9
10 Nonin3230.SCAN_UUIDS = [SERVICE_UUID];
11
12 Nonin3230.is = (peripheral) => (
13   peripheral.advertisement.localName.indexOf('Nonin3230_') > -1
14 );
15
16 NobleDevice.Util.inherits(Nonin3230, NobleDevice);
17
18 NobleDevice.Util.mixin(Nonin3230, NobleDevice.BatteryService);
19 NobleDevice.Util.mixin(Nonin3230, NobleDevice.DeviceInformationService)
20   ;
21
22 Nonin3230.prototype.onMeasurement = function(data) {
23   const b = data[1];
24   const status = {
25     encryption: !(b & 0x40),

```

```
25     lowBattery: !(b & 0x20),
26     correctCheck: !(b & 0x10),
27     searching: !(b & 0x8),
28     smartPoint: !(b & 0x4)
29   }
30   const counter = data.readInt16BE(5);
31   const oxygenSaturation = data.readInt8(7);
32   const pulseRate = data.readInt16BE(8);
33   this.emit('data', { counter, oxygenSaturation, pulseRate, status });
34 };
35
36 Nonin3230.prototype.completeMeasurement = function(done) {
37   this.writeDataCharacteristic(SERVICE_UUID, '1447
38     af800d6011e288b60002a5d5c51b',
39     new Buffer([0x62, 0x4E, 0x4D, 0x49]), done);
40 };
41 Nonin3230.prototype.connectAndSetup = function(callback) {
42   NobleDevice.prototype.connectAndSetup.call(this, function(error) {
43     this.notifyCharacteristic(SERVICE_UUID, NOTIFY_CHAR, true,
44     this.onMeasurement.bind(this), function(err) {
45       callback(err);
46     });
47   }).bind(this));
48 };
49
50 module.exports = Nonin3230;
```

Kodesnutt A.4: Nonin 3230 noble-device library

Tillegg B

Formell henvendelse til Trondheim kommune

Merk: Henvendelsen er omskrevet tre steder for å anonymisere

Til Prosjekt- og programledelsen Velferdsteknologi, Trondheim kommune

Vi søker med dette om samarbeid for en forskningsprosjekt for en masterstudent (Andreas Drivenes) ved Institutt for Datateknikk og Informatikk ved NTNU innen området veldferdsteknologi med undertegnede som veileder. Prosjektet omhandler forbedring av brukervennlighet for pulsoksimeter til avstandsoppfølging av kronisk syke. Bakgrunnen for prosjektet er et mangeårig samarbeid med SINTEF Helse (omskrevet) rundt brukervennlighet og design av veldferdsteknologi.

Studenten har koblet opp et Nonin Pulsoksimeter mot en skytjeneste via Bluetooth (omskrevet). Bakgrunnen for prosjektet er en mistanke om at en del kronisk syke vil ha problemer med tekniske løsninger for avstandsoppfølging dersom de selv alene skal bruke måleutstyr som krever teknisk kunnskap, f.eks. oppkobling av bluetooth til nettbrett. Den nye s.k. Internet-of-Things teknologien med direkte kobling til skytjenester gjør det mulig å ha måleutstyret direkte på internett, slik at man slipper å gå veien om f.eks. et nettbrett. Vi tror dette vil kunne bedre brukervennligheten for kronisk syke, og gi en bedre tjenestekvalitet. Det vil muligens også kunne redusere behovet for nettbrett for noen brukere.

Rent konkret så ser vi for oss følgende samarbeid med kommunen:

1. Samarbeid med C.C. i form av (omskrevet):
 - Ett intervju a maks to timer (mars)
 - Feedback på teknisk løsning, maks to timer (april).

Hvis mulig ønsker vi også å få tilbakemelding fra helsearbeidere som er nærmere feltet:

2. Fokusgruppe med 4-5 helsearbeidere som har erfaring med avstandsoppfølging av kronisk syke.
 - Hjelp til rekruttering av fokusgruppedeltagere.
 - Frikjøp av helsearbeidere, maks en time. (april).

Med håp om konstruktivt samarbeid.

mvh Dag Svanæs

Dag Svanæs, Professor, PhD
Department of Computer Science (IDI)
Norwegian University of Science and Technology (NTNU)
Trondheim, Norway

Tillegg C

Henvendelse til Nonin om pulsoksymeter

Question about Nonin 3230 Bluetooth integration (Sendt til oem@nonin.com)

Hi!

I am currently writing a master's thesis about welfare technology at the Norwegian University of Science and Technology.

This is done in collaboration with the municipality of Trondheim which uses Nonin 3230 in their solution for remote monitoring of patients with chronic diseases.

As a part of the thesis, I am going to implement a simple prototype of how a smart and cloud connected pulse oximeter could work.

Would it be possible to receive the technical documentation for Nonin 3230? I am especially interested in hearing about how the GATT-based proprietary profile works, so that I can make sense of the data received from the sensor.

Best regards,
Andreas Drivenes

Tillegg D

Invitasjoner til intervju

D.1 Intervjuhenvendelse til Trondheim kommune (case-studie)

Hei!

Har dere mulighet til å ta et intervju torsdag eller fredag i neste uke? Når som helst på dagen passer. Ellers så passer det når som helst på dagtid uka etter det igjen (27. mars - 31. mars).

Det hadde vært flott om jeg fikk lov til å ta opp intervjuet. Jeg kommer til å forberede noen skriftlige spørsmål (som jeg kan sende på forhånd om dere ønsker), men vi kan gjerne bevege oss litt utover det med oppfølgingsspørsmål om det passer seg sånn.

Det hadde vært fint å titte kort på løsningen dere har nå etter intervjuet, og ta noen bilder av den.

Andreas

D.2 Intervjuhenvendelse til Trondheim kommune og SINTEF (evaluering og oppsummering)

Hei!

Jeg har utviklet en prototype dere kan se på og teste, og som jeg gjerne vil ha tilbakemeldinger på.

Etter testen tenkte jeg at vi kunne snakke litt om hvordan brukere har opplevd løsningen som Trondheim kommune prøver nå, hva dere tenker om smarte og integrerte sensorer som alltid er på nett, og hva som vil være viktig når det skal utvikles smarte velferdsteknologiprodukter for oppfølging av kronisk syke i fremtiden.

Har dere mulighet til å møte meg i slutten av neste uke, eller tidlig uken etter det igjen (tidsrommene 26. - 28. april og 2. - 5. mai)? Jeg anslår at det kommer til å ta en time til

halvannen time, og jeg er veldig fleksibel på tidspunkt på dagtid.

Vi kan godt møtes hos dere, Z.Z., hvis det er enklest for dere.

Andreas

D.3 Intervjuhenvendelse til teknologileverandøren Imatis

Hei!

Mitt navn er Andreas Drivenes, og jeg går siste året på datateknologi på NTNU. Jeg skriver for tiden en masteroppgave om avstandsoppfølging av kronisk syke i samarbeid med Trondheim kommune og SINTEF. Veilederen min er professor Dag Svanæs.

Jeg har fått låne et Nonin 3230-pulsoksimeter av Trondheim kommune som jeg har integrert med en Raspberry Pi Zero. Den har en fingeravtrykkssensor, en knapp og to leds. Den publiserer sensordataen til AWS IoT, og jeg har satt opp en helt enkel grafisk applikasjon som viser målingene. Koden til oppgaven kommer til å ligge åpent på Github.

Det ville være til stor hjelp for oppgaven om dere kunne snakke litt om hva slags teknologi dere bruker i løsningen deres i dag.

Rent konkret lurer jeg på om du (eller noen andre) kunne være med på et kort intervju på appear.in eller lignende? Jeg ser for meg at det kommer til å ta maks 30-45 minutter. Temaer for samtalen kommer til å være hvilken teknologi og arkitektur dere bruker i løsningen for Trondheim kommune i dag, sikkerhet, tanker om prototypen og teknologien jeg har brukt, og den digitale utviklingen i velferdsteknologi og avstandsoppfølging av kronisk syke i årene som kommer.

Jeg er ledig på dagtid det meste av den neste uken.

Jeg sender selvsagt en kopi av masteroppgaven når den blir ferdig i midten av juni.

--

Med vennlig hilsen
Andreas Drivenes

Tillegg E

Forberedelse til intervju med Trondheim kommune

Info før intervjuet: Tar opp samtalen på mobilen, sletter opptaket etterpå når jeg er ferdig.

Jeg tenkte jeg skulle starte med å snakke litt kort om meg og hva jeg holder på med og hva planene er for masteroppgaven. Jeg heter Andreas Drivenes og går i femte klasse datateknologi på NTNU. Jeg skriver en masteroppgave om avstandsoppfølging av kronisk syke med Dag Svanæs som veileder. Opprinnelig var temaet for oppgaven min å undersøke ulike skybaserte tingenes internett-løsninger, og casen vi hadde sett på var deling av veidata for rullatorer. Imidlertid hadde jeg et møte med X.X. og Y.Y på Sintef som pekte meg i retningen av avstandsoppfølging av kronisk syke siden det var noe som faktisk var brukt i praksis og som det var tester på i Trondheim. Så i fjor høst skrev jeg en prosjektoppgave om avstandsoppfølging og hvordan det kunne bli brukt sammen med moderne skyløsninger. Jeg skal snakke litt mer om hva vi tenker å lage som prototype etterpå

Velkommen til intervju, det var veldig flott at dere kunne sette av tid til det. Først snakke litt om dere og hva dere jobber med. Så litt kort om bakgrunnen til velferdsteknologi-programmet til Trondheim, deretter om løsningen dere har for avstandsoppfølging av kronisk syke i dag - hvordan den fungerer fra A-Å, det som er bra med den dårlig med den etc, hvordan dere tror det vil være i fremtiden, hvordan brukere ser på den osv. Etter det tenkte jeg å presentere litt av min egen løsning og spørre rundt den. Tilslutt tenkte jeg at vi kunne se litt på løsningen deres om jeg får lov til det

BAKGRUNN Kan dere starte med å si hva dere heter, og hva dere jobber med i Trondheim Kommune?

Jeg har inntrykk av at Trondheim kommune er veldig flinke på å ta i bruk ny teknologi i helsetjenester. Kan dere si litt om bakgrunnen for det og hvilke prosjekter dere har nå?

Hvordan foregår samarbeidet med Helsedirektoratet? Utveksler dere kunnskap med andre kommuner?

HELSAMI+ Spørsmål om avstandsoppfølging av kronisk syke generelt: Hvor god klinisk vurdering kan man gjøre av helsetilstand med måling av blodtrykk, vekt, puls og

oksygenmetning, samt spørsmål om egen dagsform?

Hvorfor er det så viktig at pasienter følges opp hjemme? Hva er fordelene med det? (trygghets- og mestringsfølelse, færre reinleggelser?)

Hva er den historiske bakgrunnen for avstandsoppfølging-prosjektet (altså HelsaMi+) til Trondheim kommune?

Kan dere beskrive hvordan løsningen i HelsaMi+ fungerer for en typisk pasient fra A til Å? Fra installering av utstyr, til måling, til individuell oppfølgingsavtale?

Hva gjøres med målingen som kommer inn? Har helsepersonell tilgang på den? Hva er det som eventuelt stopper at helsepersonell kan se direkte på målingene?

Kan pasienten selv se målingene sine i nettbrettet? Sammenlignet med historiske data.

Hva slags type spørsmål er det pasienten må svare på i løsningen? (Det som kalles for fastlagt spørreskjema)

Er det pasienten selv som må sørge for at alt har batteri og strøm og koblet opp på Internett? Har det vært noen utfordringer der?

Hvordan bestemte dere hvordan løsningen skulle se ut? Hvem var med på å utforme kravene til løsningen?

Hva er tilbakemeldingen fra pasientene på løsningen? Har dere gjennomført noen brukertester med eldre pasienter? Har pasientene vært med på utviklingen av tjenesten?

Hva er tilbakemeldingen fra helsepersonell, altså folk i kommunen hos dere, fastleger osv. på løsningen?

Er det noe regelverk eller sikkerhetsutfordringer som har stoppet dere i fra å gjøre løsningen på en annen måte?

Hva slags type nettbrett er det brukerne får?

Hvor mye koster utplassering av systemet deres hos en pasient? (ca.-pris, hvis jeg kan spørre om det)

Hvordan har samarbeidet deres vært med Imatis? Har dere kommet med innspill som har gjort at de har endret på ting?

“Da prosjektet følges av en forskningspartner så vil du for å delta måtte signere et samtykkeskjema for at du kan bli intervjuet og at anonymiserte data kan hentes ut for å se på om denne tjenesten gir en bedre effekt på for eksempelvis færre reinnleggelser eller en økt mestring og trygghetsfølelse for brukerne.” Vil denne forskningen bli publisert offentlig? Er det allerede nå blitt publisert noe? Når tenker man å publisere? Kan jeg kontakte noen for å få tilgang til noe materiale?

HELAMI+ i fremtiden Sitat: “Prosjektet vil i løpet av 2017 videreutvikle tjenesten til å rette et større fokus på forebyggende tiltak som bruker kan gjennomføre selv med bistand fra teknologien.” Hva betyr dette?

Hvis dere kunne endre på noe i løsningen selv, hva ville dere endret på da?

Hva er planene for HelsaMi+ framover? Hva vil skje etter 2018?

Kunne det vært et dataprogram som automatisk oppdaget avvik basert på pasientenes historiske data?

Integrering med andre løsninger?

MIN LØSNING smart pulsoksymeter med fingerprintsensor som autentiseringsmetode (slipper styret med passord) trykke på knapp -> blinker gult -> autentiserer med finger -> grønt lys, blinker gult lengre borte -> måler oksygenmetning og puls -> blinker grønt når ferdig.

Integrert løsning med autentisering, toveiskommunikasjon til skyen, lager prototype som skal testes, brukeren forholder seg til én dings

Hva tenker dere rundt en sånn løsning?

Lurt med fingerprintsensor som autentisering, eller potensielle fallgruver?

Kan det være bra å få info om når målingene skal gjøres?

TIL SLUTT Tusen takk for at dere stilte. Er det noe jeg burde spurt om som jeg har glemt?

Tillegg F

Samtykkeerklæringer

F.1 Anonym

*Tingenes internett for avstandsoppfølging av kronisk syke
Deltakelse på gruppeintervju*

Samtykkeerklæring

Jeg har mottatt muntlig informasjon om studien, og fått anledning til å stille spørsmål. Jeg er klar over at det er frivillig å delta, og at jeg kan trekke meg fra studien når som helst uten å oppgi noen grunn. Jeg samtykker i å delta i studien.

Det vil bli tatt lydopptak. Dette gjøres for at vi skal kunne vurdere intervjuene i etterkant og for å sikre oss at vi har forstått utsagn og handlinger riktig. Vi vil sørge for at materiale vil bli anonymisert slik at det ikke vil være mulig å føre opplysningene tilbake til enkeltpersonene som deltar i prosjektet. Dette innebærer at informasjon som blir formidlet til offentligheten ikke vil kunne settes i sammenheng med den enkelte. Det er kun de involverte i prosjektet som vil kunne se opptakene i ettertid.

Trondheim, _____

Navn

Underskrift

Masterstudent ansvarlig for studien:

Andreas Drivenes, e-mail: andreas.drivenes@gmail.com

Faglig ansvarlig:

Professor Dag Svanæs, e-mail: dags@idi.ntnu.no

Ansvarlig avdeling:

Institutt for Datateknologi og Informatikk (IDI), NTNU

F.2 Ikke-anonym

*Tingenes internett for avstandsoppfølging av kronisk syke
Deltakelse på gruppeintervju*

Samtykkeerklæring

Jeg har mottatt muntlig informasjon om studien, og fått anledning til å stille spørsmål. Jeg er klar over at det er frivillig å delta, og at jeg kan trekke meg fra studien når som helst uten å oppgi noen grunn. Jeg samtykker i å delta i studien.

Det vil bli tatt lydopptak. Dette gjøres for at vi skal kunne vurdere intervjuene i etterkant og for å sikre oss at vi har forstått utsagn og handlinger riktig. Jeg samtykker i at mitt navn kan bli brukt i masteroppgaven og at det jeg sier kobles til mine utsagn. Jeg blitt informert om at jeg har fått mulighet til å lese igjennom referatene fra intervjuene og kommentere på eventuelle feil i transkribering.

Det er kun de involverte i prosjektet som vil kunne høre opptakene i ettertid.

Trondheim, _____

Navn

Underskrift

Masterstudent ansvarlig for studien:

Andreas Drivenes, e-mail: andreas.drivenes@gmail.com

Faglig ansvarlig:

Professor Dag Svanæs, e-mail: dags@idi.ntnu.no

Ansvarlig avdeling:

Institutt for Datateknologi og Informatikk (IDI), NTNU