



AARHUS
UNIVERSITET
INGENIØRHØJSKOLEN

Projektrapport

Semesterprojekt 3

Blodtryksmålersystem

ST3PRJ3-03

Gruppe 5

Sundhedsteknologi

Århus Universitet, IHA

Vejleder: Samuel Alberg Thyresøe

Dato: 16/12 2015

Mette Østergård Knudsen, 201404501

Ida Mark Skovbjerg, 201404669

Line Skov Larsen, 201405838

Brian Hansen, 201310502

Mohamed Hussein Mohamed, 201370525

Khaled Edwan, 200800899

Resumé
Abstract

Indhold

| | |
|--|-----------|
| Indhold | ii |
| 1 Indledning | 1 |
| 2 Projektformulering og afgrænsning | 2 |
| 3 Hjertet | 3 |
| 4 Systembeskrivelse | 4 |
| 5 Krav | 5 |
| 5.1 Aktørbeskrivelse | 6 |
| 5.2 Use case beskrivelse | 7 |
| 5.3 Ikke-funktionelle krav beskrivelse | 8 |
| 6 Projektbeskrivelse | 10 |
| 6.1 Projektgennemførelse | 10 |
| 6.2 Metode | 10 |
| 6.3 Specifikation og analyse | 10 |
| 6.4 Arkitektur | 10 |
| 6.5 Design, implementering og test | 11 |
| 6.6 Resultater og diskussion | 11 |
| 6.7 Udviklingsværktøjer | 11 |
| 6.8 Opnåede erfaringer | 12 |
| 6.9 Fremtidigt arbejde | 12 |
| 7 Konklusion | 13 |
| 8 Referencer | 14 |

Kapitel 1

Indledning

Årsag Formål Produktet Use cases

Kapitel 2

Projektformulering og afgrænsning

Hvad vil vi Hvad skal produktet kunne Løsning Fremtiden Deltager; Fulde navn og initialer

Kapitel 3

Hjertet

Beskrivelse af hjertet Blodtryk systole diastole middeltryk EKG puls Statistik?

Kapitel 4

Systembeskrivelse

Kapitel 5

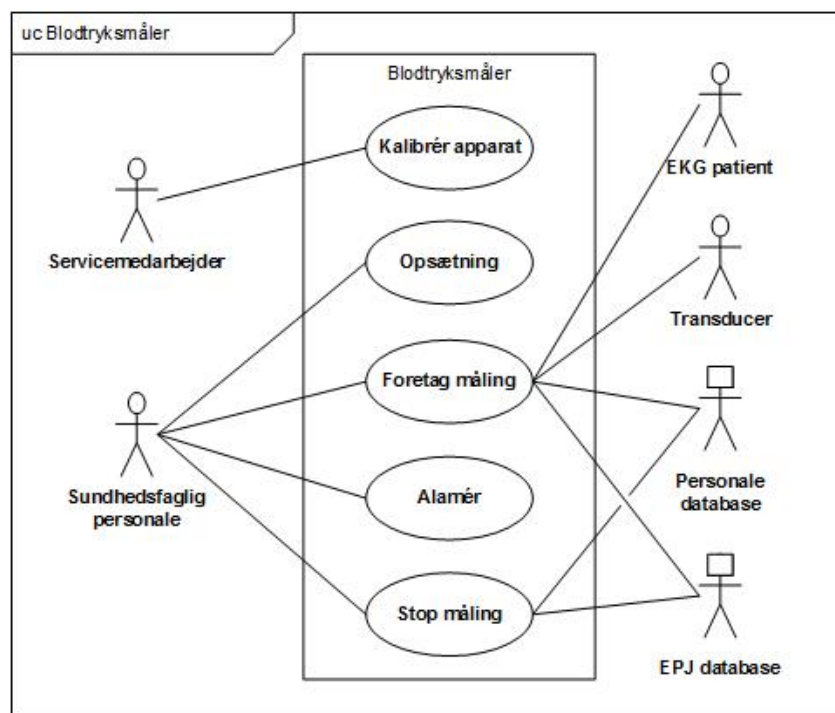
Krav

Hvilke krav der stilles til produktet udarbejdes i en kravspecifikation. Denne kravspecifikation består af en række Use cases. Disse Use cases beskriver interaktionen mellem aktørerne og systemet. Use casene har til formål at specificere, hvilke krav der stilles til produktet. Kravene opstilles ud fra hvad kunden ønsker samt hvad leverandøren finder muligt at realisere.

Der er nogle obligatoriske krav til produktet, der skal opfyldes. Disse krav er bl.a at systemet skal kunne kalibrere blodtrykssignalet og foretage en nulpunkts justering. Desuden skal blodtrykssignalet vises kontinuerligt, hvilket betyder, at signalet skal vises grafisk samtidigt med at det bliver hentet/indsendt fra hardwaren. Disse målte data skal efter de er blevet behandlet gemmes i en database.

Et af de obligatoriske krav er desuden at der skal være et digitalt filter der kan tilgås når programmet kører. Dette filter skal både kunne slås til og fra. Dette filter skal sørge for at signalet haves i to forskellige modes; et diagnose mode, hvor signalet er rå med alle udsving, og et monitor mode, hvor signalet er filtreret og afrundet.

Foruden disse obligatoriske krav er det valgt, at det skal være muligt starte og stoppe målingen. Når målingen er startet skal det være muligt at kunne justere grænseværdierne for systolen og diastolen, så disse værdier kan tilpasses. Når signalets værdier så kommer over grænseværdien, skal en alarmering begynde. Denne alarm skal kunne udsættes i et minut, dog kun alarmens lyd, så værdien blinker fortsat, så det stadig er tydeligt hvilken grænseværdi der er overskredet. Systemet består af en computer med en programkode, en NI-DAQ, et lavpas filter, en forstærker, en transducer og en væskesøjle. Systemet gør det muligt at få arterietrykket sendt ind i systemet igennem hardwaren. Arterietrykket dannes i denne prototype af en væskesøjle, som levere et tryk til en transducer, herefter sender transduceren signalet videre til hardwaren, hvor et lavpasfilter, filtrerer signalet, hvorefter en forstærker forstærker signalet. Dette signal sendes derefter igennem NI-DAQ ind i systemet, som derefter behandler og analyserer signalet. Den fulde beskrivelse af de udarbejdede Use cases (fully dressed Use case skema) findes i dokumentationen.



Figur 5.1: Use Case diagram. Dette diagram viser aktørernes interaktion med systemet.

5.1 Aktørbeskrivelse

Ud fra use case diagrammet ses de seks aktører; *Sundhedsfaglig personale*, *Transducer*, *EKG patient*, *EPJ database*, *Personale database* og *Servicemedarbejder*. Det er disse aktører der interagerer med systemet.

Sundhedsfaglig personale

Det er det sundhedsfaglige personale der er den aktør der påsætter måleudstyret på patienten. I dette tilfælde vil patienten være en transducer, hvilken er tilsluttet en væskesøjle. Det sundhedsfaglige personale er desuden den aktør der interagerer med systemet; logger på og foretager en måling. Det sundhedsfaglige personale har dermed tilgang til de viste målinger på brugergrænsefladerne; startskærm og hovedskærm

Transducer

Transduceren er i projektet den aktør der agere som patienten, denne består af en væskesøjle der er påsat en tryktransducer. Væskesøjlen leverer et tryk i mmHg videre til tryktransduceren, og dette signal, der virker som arteriestrykket, sendes videre i hardwaren, som behandler signalet. Derfor er denne aktør kilden til måleresultaterne for systole, diastole og middeltryk.

EKG-patient

EKG patienten er den aktør som er kilden til EKG-kurven, idet vi ikke har en rigtig patient, hvorfra både EKG og blodtryk kan fås fra. Idet det er fra denne aktør at EKG-kurven findes er

det fra denne aktør, pulsen kan bestemmes.
Værdierne for denne aktør hentes fra PhysioBank ATM.

Servicemedarbejder

Denne aktør, Servicemedarbejder, er den aktør, der står for at foretage kalibreringen. Dette gør Servicemedarbejderen ved at påsætte systemet til kalibrerings systemet, og indtaster de værdier for tryk (mmHg) og spænding (Volt), som måles ved tre forskellige målepunkter på en væskesøjle, hvorefter en kalibreringsværdi findes af systemet.

EPJ database

EPJ databasen, er databasen, hvori patientdata ligger samt den database hvori grafer og måleresultaterne, der bestemmes ved analysen, bliver gemt. Måleresultaterne er systole, diastole, middeltryk og pulsen. Graferne for EKG-signalet og arteriekurven gemmes som tallister. Denne EPJ database skal simulere den database der fungerer på sygehusene i virkeligheden. Denne database kobler patienterne i denne database sammen med en sundhedsfaglig fra Personale databasen.

Personale database

Det sundhedsfaglige personales login informationer ligger i Personale databasen. Det er derfor denne database der indeholder informationer om det Sundhedsfaglige personale og dermed de informationer der benyttes til at tilgå systemet.

5.2 Use case beskrivelse

Use case diagrammet viser ligeledes de 4 Use cases der er for systemet: Kalibrér apparat, Foretag måling, Alamér og Stop måling. Disse Use cases er en beskrivelse af hvad systmet skal kunne og dermed beskriver de interaktioner der sker mellem aktørerne og systemet.

Use case 1: Kalibrér apparat

Service medarbejderen skal i denne Use case foretage en kalibrering. Dette gøres ved at have tre målepunkter på en væskesøjle. På disse punkter måles spændingen (volt) og trykket (mmHg). Disse indtastes af Servicemedarbejderen ind i systemet igennem brugergrænsefladen; startskærmen. Herefter beregner systemet en kalibrerings værdi. Denne værdi bruges på arteriekurven (blodtrykskurven), for at få den kalibrede graf.

Use case 2: Foretag måling

Denne Use case styrer målingerne af graferne der indhentes. For at dette kan gøres skal det Sundhedsfaglige personale først logge på systemet. Dette gør denne aktør ved at indtaste sit brugernavn og sin adgangskode på brugergrænsefladen; startskærmen. Når den sundhedsfaglige på logget på henter systemet de tilknyttede patienter, hvorefter det Sundhedsfaglige personale kan vælge den ønskede patient. Når Patienten er blevet valgt startes hovedskærmen, hvilken repræsenterer en blodtryksmålers brugergrænseflade. Her kan den Sundhedsfaglige starte målingen. Når dette gøres henter systemet arteriekurven og EKG-signalet. Dette bruges derefter af

systemet til at beregne hhv. systole, diastole, middeltryk og puls. Systemet viser graferne kontinuerligt på hver sin graf og systole, diastole, middeltryk og puls vises som talværdier. Samtidig gemmer systemet automatisk kontinuerligt disse data i EPJ databasen.

Det vil være muligt for det Sundhedsfaglige personale at slå det digitale filter fra og til. Filteret er fra start slået til.

DEt er også muligt for det Sundhedsfaglige personale at justere grænseværdierne for systolen og diastolen, dette gøres for at indstille grænseværdierne mere individuelt for hver patient.

For at sikre at graferne der er hentet ligger rigtigt på graferne, kan det Sundhedsfaglige personale nulpunkts justere systemet, sådan at graferne kommer til at ligge rigtigt på akserne.

Use case 3: Alarmér

Alarmen startes af systemet, når en af grænseværdierne overskrides. Dette gør at grænseværdien der er overskredet blinker og at der starter en alarm lyd. Når alarmen kan endnu en grænse værdi overskrides, hvis dette sker begynder denne grænseværdi ligeledes at blinke, men der sker ikke noget med alarmlyden, denne fortsætter fra tidligere.

Det vil her være muligt for det Sundhedsfaglige af udsætte alarmen. Dette gøres ved at trykke på knappen, hvorefter at systemet stopper alarmens lyd i et minut, efter dette minut starter alarmens lyd igen. Grænseværdien/grænseværdierne der er overskredet fortsætter med at blinke indtil forholdene er normaliseret, altså indtil grænseværdien ikke længere er overskredet.

Use case 4: Stop måling

Det Sundhedsfaglige personale skal også stoppe målingen. Dette gøres ved at det Sundhedsfaglige personale interagere med hovedskærmen. Herefter vil det være muligt for det Sundhedsfaglige personale at logge

5.3 Ikke-funktionelle krav beskrivelse

Ikke funktionelle krav er opsat efter FURPS+ metoden. Kravene er herefter blevet prioriteret efter MoSCoW.

FURPS+

Functionality

Funktionalitet er det brugeren ønsker sig. Dette omfatter også sikkerhedsrelaterede behov. Dette er krav til hvad programmet skal kunne af funktionalitet, f.eks. at programmet skal have et digitalt filter.

Usability

Hvor effektiv er produktet, set fra forbrugers side, dette er det aspekt brugervenlighed ser på. Er produktet nemt at bruge. Hvordan bruges produktet; er der nogen brugergrænseflader. Det er herunder kravene til hvilke knapper der skal være på brugergrænsefladerne, og dermed også hvilke brugergrænseflader der skal være.

Reliability

Pålidelighed tager sig af aspekter som, hvor længe er det maksimalt at systemet må være nede. Er der fejl der kan forudses. Hvor præcist kan resultaterne vises. Er produktet let at vedligeholde; kan delene i produktet skiftes let.

Performance

Præstationen for produktet, handler om hvor hurtigt produktet er om at starte op og hvor hurtigt svartiden er. Hvor stor må svartiden maksimalt være. Så det er altså herunder, at det er beskrevet, hvor lang tid der går fra der er trykket på en knap, til at systemet svarer.

Suportability

Produktets support fortæller, om det er muligt at teste på produktet, om det er muligt at udvide produktet, installere og konfigurere produktet. Desuden om produktet er kompatibelt. Det er herunder programmets opbygnings model beskrives

+

Kunden kan have nogle yderligere behov, disse yderligere behov beskrives under +. Hvilke begrænsninger er der ved designet. Er der nogle krav for brugergrænsefladerne. Er der nogle fysiske eller implementerings krav. Er der dele der kan genbruges, herunder hele systemer eller dele af dem. Det er bl.a herunder at kravene til computeren der benyttes beskrives.

<http://agileinaflash.blogspot.dk/2009/04/furps.html>

MoSCoW**Must**

De krav der bliver markeret som et must er de krav som produktet skal have. Altså det produktet skal have/indeholde.

Should

De krav der markeres som et should krav, er de krav til produktet der burde være med. Altså er det hvad produktet bør indeholde

Could

Kravene kan også markeres som could. Disse krav er de krav der kunne være gode at have med, men som ikke bliver prioriteret. Så det er kun, hvis man kan nu at få det med, at de skal være der.

Would/Won't

Det er ikke alle krav der skrives, der skal være gældende for produktet, disse krav markeres som would/won't. Det er altså disse krav som man ikke tager med eller de krav som ville være sjove at have med, men ikke har en betydning for produktet, men er en tilføjelse eller udvidelse. Det er disse krav der danner rammen for fremtidigt og videregående arbejde med projektet.

Kapitel 6

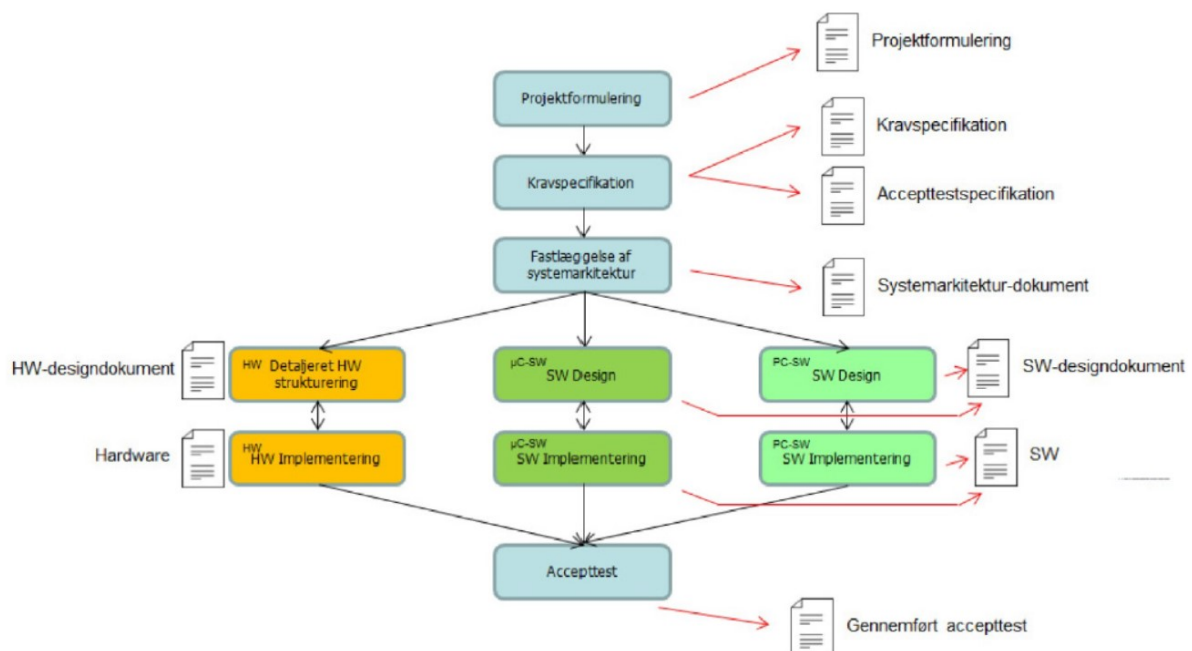
Projektbeskrivelse

6.1 Projektgennemførelse

Modellerne

ASE-modellen

Til udviklingen af dette produkt er der primært benyttet ASE-modellen. Modellen ses herunder. ASE-modellen er en udviklingsmodel, der er udarbejdet af Aarhus Ingeniørhøjskole. Denne model er en mellemvægtig semi-iterativ udviklingsproces, hvilken drives ud fra Use cases. Projektudviklerne fastlægger først en projektformulering, hvorfra en kravspecifikation kan udarbejdes, hvilke indeholder kravene fra kunden. Systemarkitekturen fastlægges ud fra kravspecifikationen, for derefter at designe og implementere de enkelte hardware og software dele hver for sig i iterationer. Til sidst fører dette til en accepttest, hvilken gennemføres, så der opstår en enighed mellem projektudviklerne og kunden.

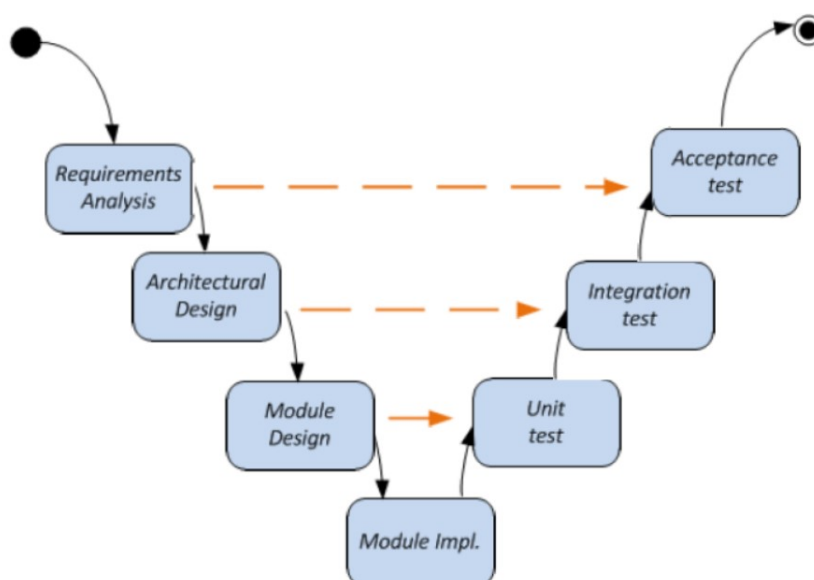


Figur 6.1: ASE-modellen.

Kravspecifikationen bliver specificeret som en række Use cases ud fra problemformuleringen. Use cases benyttes til at beskrive aktørernes interaktion med systemet. Sammen med de ikke-funktionelle krav opnås der med Use casene et overblik over hvilke krav der stilles til systemets funktionalitet. Systemets design bestemmes igennem hardware og software design og implementering, herefter kan en accepttest udarbejdes. Ved udførelsen af accepttesten tjekkes det at kravene er opnået.

V-modellen

V-modellen er en model som er faseopdelt udviklingsmodel. Denne model beskriver udviklingsfaserne og testfaserne i et projekt sideløbende. V-modellen er blevet benyttet sideløbende med ASE-modellen. Specifikationen af test foregår parallelt med udviklingen af selve systemet i V-modellen. Fordelen ved denne model er at testene sker på forskellige niveauer, hvilket sikrer at udviklede delsystemer virker som ønsket. Det vigtige her er at en fase er færdiggjort, inden den næste fase påbegyndes.

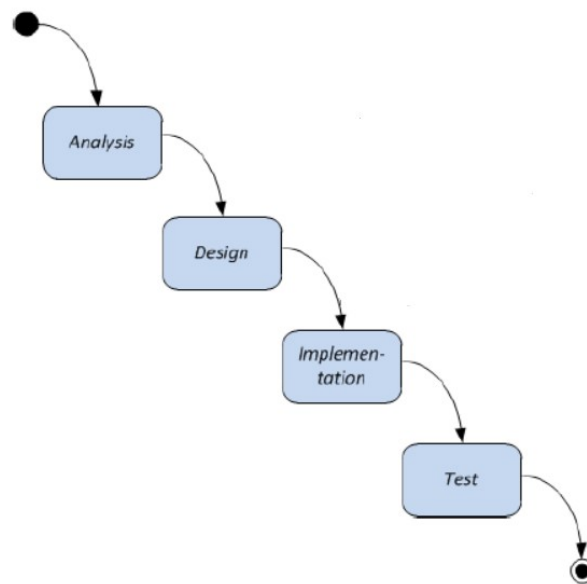


Figur 6.2: V-modellen.

Ud fra modellen ses det at den første fase er udviklingen af kravspecifikationen. Her udarbejdes der en tilhørende accepttest, denne test gør det muligt at tjekke om systemet til sidst lever op til de opstillede krav. Den næste fase er herefter systemarkitektur, her udvikles den tilhørende test, denne test undersøger integrationen mellem de implementerede moduler. De sidste to fase af udviklingsfasen er design og implementering af systemets enkelte moduler, her udføres enhedstesten løbende af de implementerede moduler.

Vandfaldsmodellen

Vandfaldsmodellen er en model der bruges til udvikling af software, hvor softwareudviklingen betragtes som værende konstant løbende nedad. Her kan udviklingen af et modul i en ny fase først påbegyndes når den foranliggende fase er afsluttet. Denne model er i projektet benyttet til udviklingen af softwarearkitekturen.



Figur 6.3: Vandfalds modellen.

Disse tre modeller; ASE-modellen, V-modellen og vandfalds-modellen arbejder alle i en kronologisk logisk rækkefølge. Modellerne er benyttet i et omfang, hvor de har kunnet supplere hinanden. ASE-modellen giver det store overblik, mens V-modellen sikrer udarbejdelsen af de nødvendige tests. Vandfaldsmodellen er den proces, hvilken der er arbejdet ud fra i alle projektets facetter.

Projektstyring

Tidsplan Opdeling Github SCRUM - Pivotal tracker

6.2 Metode

UML SysML - Beskrivelse af modellerne - BDD - IBD - Domæne - Sekvens - Applikation - Klasse diagrammer - Software (black box?)

6.3 Specifikation og analyse

?? Beregninger til hardware

6.4 Arkitektur

Hardware design

kort indledning Hardware beskrivelse Signalets vej BDD indsættes

Software design

Kort indledning

6.5 Design, implementering og test

PUSH Observer/subject QUEUE

6.6 Resultater og diskussion

Hvad er der kommet frem til

6.7 Udviklingsværktøjer

Til udviklingen af produktet, og dermed igennem hele projektarbejdet, er der blevet brugt en række udviklingsværktøjer. Følgende er disse udviklingsværktøjer beskrevet yderligere.

Microsoft Visio 2010

Tegneværktøjet Microsoft Visio, er blevet anvendt i forbindelse med design af både UML og SysML diagrammer. Microsoft Visio er det oplagte valg til at designe disse diagrammer, da programmet søger for at diagrammerne får et enkelt udseende og tydeligt kommunikerer til læseren, hvad diagrammerne vil vise.

Visual Studio 2013

Programmeringskoden til projektet er blevet skrevet i C#. Her er Visual Studio 2013 blevet anvendt som kompiler. Dette program er blevet valgt da Visual Studio gør det nemt at omskrive ideer og tekst til kode, som derefter kan omskrives til metoder. Desuden indeholder Visual Studio 2013 funktionen Windows Form Application. Denne funktion gør at de ønskede resultater fra programmets metoder visuelt kan fremstilles.

NI Multisim 13.0 med Ultiboard 13.0

NI-DAQmx

NI-DAQmx, også omtalt NI-DAQ, er et værktøj som er blevet udarbejdet af National Instrument. Dette værktøj anvendes til at forstærke det indkommende signal. Dette signal kommer fra hardwaren. NI-DAQ omdanner desuden signalet fra et digitalt signal til at være et signal der kan bruges i kode øjemed.

Analog Discovery fra Digilent and Analog Devices

Analog Discovery er i projektet den der benyttes om en waveform-generator. Det er derfor Analog Discovery der benyttes til at teste systemet. Dette sker ved at der sendes et sinus signal ind i systemet, som systemet derefter skal behandle.

6.8 Opnåede erfaringer

Hvad er erfaret - hardware - forskellige måder at lave det samme på. - programmering - observer subject - Push - tråde reviews

6.9 Fremtidigt arbejde

Iltmætning Få produktet til at virke på en patient Andet fremtidigt Både indenfor hardware og software

Kapitel 7

Konklusion

Afrunding konkludering Hvad lykkedes og hvad lykkedes ikke

Kapitel 8

Referencer