



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

<b>Indhold</b>	<b>ii</b>
<b>1 Indledning</b>	<b>1</b>
<b>2 Projektformulering og afgrænsning</b>	<b>2</b>
2.1 Oversigt over projektdeltagere og hovedansvarsområder . . . . .	2
<b>3 Hjertet</b>	<b>3</b>
<b>4 Systembeskrivelse</b>	<b>5</b>
<b>5 Krav</b>	<b>6</b>
5.1 Aktørbeskrivelse . . . . .	7
5.2 Use case beskrivelse . . . . .	8
5.3 Ikke-funktionelle krav beskrivelse . . . . .	9
<b>6 Projektbeskrivelse</b>	<b>11</b>
6.1 Projektgennemførelse . . . . .	11
6.2 Metode . . . . .	13
6.3 Specifikation og analyse . . . . .	14
6.4 Arkitektur . . . . .	14
6.5 Design, implementering og test . . . . .	14
6.6 Resultater og diskussion . . . . .	16
6.7 Udviklingsværktøjer . . . . .	16
6.8 Opnåede erfaringer . . . . .	16
6.9 Fremtidigt arbejde . . . . .	16
<b>7 Konklusion</b>	<b>17</b>
<b>8 Referencer</b>	<b>18</b>
<b>Litteratur</b>	<b>19</b>
8.1 Referencer . . . . .	19

# Kapitel 1

## Indledning

Vi har i dette projekt arbejdet med design og udvikling af et blodtryks målesystem. Ideen bag vores system blev udtænkt og udarbejdet med viden om en blodtryksmåler på en operationsstue på Herning Sygehus. Vores opstillede krav samt design til systemet, blev derfor lavet ud fra de oplysninger vi fik af en sygeplejerske<sup>1</sup> på Herning Sygehus. Vi ønskede altså at lave et blodtryksmålersystem, som er så virkelighedsnært som muligt. Det design og de funktionaliteter vi efterstræbte at få opfyldt i vores projekt var derfor EKG, systoliske samt diastoliske blodtryk og iltmætning i blodet. Dette ønskede vi at få implementeret og hermed få grafer og værdier vist på en brugergrænseflade, som lever op til de standarder der ses på operationsstuer i dag.

Baggrunden for dette projekt er, at man i klinisk praksis har behov for kontinuert at observerer patientens blodtryk. Det er i særdeleshed på intensiv afdelinger og operationsstuer, at man ønsker at lave invasiv blodtryksmåling. Ved invasiv blodtryksmåling kan man monitorere patientens fysiske tilstand under eksempelvis operation. Blodtrykket kan give en identifikation af blødning, smerter og hvor dybt patienten sover under narkosen. Denne overvågning giver det sundhedsfaglige personale en ekstra tryghed og sikkerhed om patientens tilstand under operation.

Denne projekt rapport vil give en kort beskrivelse af vores samlede system samt krav hertil. Der vil være en udspecificeret projektbeskrivelse, som vil give et indblik i vores vej frem til det endelige resultat, og som også vil beskrive både de til- og fravalg vi har måttet tage, for at nå frem til netop det system vi ønskede at realiserer og præsenterer. Der har i den forbindelse været nogle ideer og ønsker, som ikke har været realistiske og tidsmæssigt mulige at opfylde og vil derfor være beskrevet i fremtidigt arbejde. Slutteligt vil derfor være en samlet konklusion på projektarbejdet.

---

<sup>1</sup>Anæstesi sygeplejerske Charlotte Høj, Herning Sygehus

## Kapitel 2

# Projektformulering og afgrænsning

### 2.1 Oversigt over projektdeltagere og hovedansvarsområder

Vi har i dette projekt valgt at dele projektets deltagerer op i to grupper med hver deres hovedansvarsområder. En som har beskæftiget sig med hardware og en som har beskæftiget sig med software. Derudover valgte vi i starten af forløbet at udvælge en projektleder, hvis opgaver bestod i at lave dagsorden for hvert møde, sørge for at deadlines blev overholdt, træffe endelige beslutninger og holde et overordnet overblik over projektet. Vi valgte også en procesleder hvis opgaver bestod i opgavestyring, godt arbejdsmiljø i gruppen og planlægning af projektførelsen. Herunder ses de forskellige gruppemedlemmers hovedområde.

Hardware	Software
Brian Hansen	Ida Mark Skovbjerg
Mohamed Hussein Mohamed	Mette Østergård Knudsen
Khaled Edwan	Line Skov Larsen

Under projektførelsen valgte vi at ændre denne rollefordeling, idet den generelle arbejdsfordeling samt arbejdsindsats i gruppen ikke fungerede optimalt. Vi valgte, at kun et gruppemedlem skulle varetage rollen som projektleder og procesleder, men at opgaver herunder kunne uddelegeres hvis nødvendigt. Denne leder blev i fællesskab valgt og baggrunden for dette valg var, at vi ønskede en leder som både havde viden og indflydelse på hardware og software, samt kunne træde i kræft hvis gruppearbejdet og deltagernes indsat ikke opfyldte den forventningsafstemning, som står skrevet i vores samarbejdskontrakt. Gruppens medlemmer med fulde navn og initialer ses herunder, det er initialerne benyttes som reference til gruppemedlemmerne herfra.

Fulde navn	Initialer
Ida Mark Skovbjerg	IMS
Line Skov Larsen	LL
Mette Østergård Knudsen	MK
Brian Hansen	BH
Mohamed Hussein Mohamed	MM
Khaled Edwan	KE

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

## Kapitel 3

# Hjertet

Hjertet er en muskel, der fungerer som en pumpe i kroppen. Hjertet pumper det iltede blod ud til resten af kroppen gennem arterie- og vene systemer. Når blodet passerer ud af hjertet igennem arterierne sker der et tryk mod arterievæggene, dette kaldes blodtrykket. Man benytter blodtrykket til at bedømme kraften og mængden af det blod, som pumpes ud af hjertet. Blodtrykket inddeles i to typer, systolisk og diastolisk.

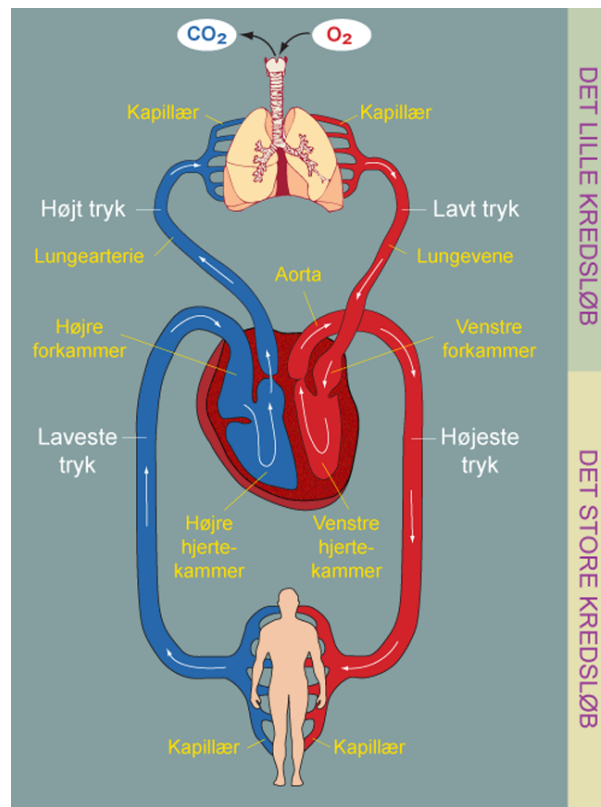
Systolen angives som den kraft, der skabes når der kommer pres på karvæggen i arterierne. Dette sker i hjertets sammentrækningsfase, hvor blodet pumpes ud i arterierne. Systolen i et normalt blodtryk i hvile vil ligge i intervallet 100-140 mm Hg og ved forhøjet blodtryk er værdien for trykket over 140 mm Hg. Diastolen angiver hjertets hvilefase og ses mellem to sammentrækninger. I denne fase fyldes ventriklerne med blod. Den normale værdi for diastolen ligger i intervallet 60-90 mm Hg. Hvis den overstiger 90 mm Hg anses blodtrykket for at være forhøjet. Udover systolen og diastolen kan også middeltrykket angives. Denne udregnes ved formlen: 
$$\text{middeltryk} = \frac{2}{3} \cdot \text{diastolisk} + \frac{1}{3} \cdot \text{systolisk}. [2]$$

Hypertension defineres som forhøjet systole, forhøjet diastole eller både forhøjet systole og diastole. Hypertension belaster hjertet, og medfører en øget risiko for udviklingen af apopleksi. Det er også forbundet med flere medicinske tilstande såsom arteriosklerose, hjerteinsufficiens og nyreskader. [3]

Ser man på anatomen bag blodtrykket og blodtryksændring, består dette af kredsløbssystemet, som omfatter blodets kredsløb, hjertet og lymfesystemet. Blodkredsløbet er et langt netværk af arterier og vener, som har til opgave at føre iltet blod ud til hver celle i kroppen, samt opfange affaldsprodukter fra celler og væv. [4] I lungekredsløbet udskilles affaldsprodukter og blodet iltes. Det er hjertet der pumper blodet ud i dette netværk, så blodet hele tiden er i bevægelse. Arterierne transportere blodet væk fra hjertet, og venernes opgave er at transportere blodet mod hjertet.

Kredsløbet kan funktionelt og anatomisk inddeles i to dele: Det pulmonale kredsløb og det systemiske kredsløb. De to kredsløb fungerer i symbiose mellem hjertets højre hjerte kamre og hjertets venstre hjertekamre. Højre siden pumper blodet gennem det pulmonale kredsløb, og venstre side pumper blodet rundt i det systemiske kredsløb. Hver pumpe består af to kamre: et forkammer, atrium og et hovedkammer, ventriklen. Det pulmonale kredsløbs primære funktion er at levere blodet til lungekredsløbet hvor blodet iltes. Hjertets højre side modtager iltfattigt blod og CO<sub>2</sub>-rigt blod via v. Cava superior og inferior fra det systemiske kredsløb og pumper

det videre ind i lungekredsløbet gennem det pulmonale kredsløb via pulmonal arterierne. Fra lungerne løber det nu ilttrige og CO<sub>2</sub>-fattige blod tilbage til venstre side af hjertet, som derfra pumper det videre til det systemiske kredsløb via aorta. [4]



Figur 3.1: Det lille og det store kredsløb. Motoren der driver det hele er hjertet. [1]

## Kapitel 4

# Systembeskrivelse



# Kapitel 5

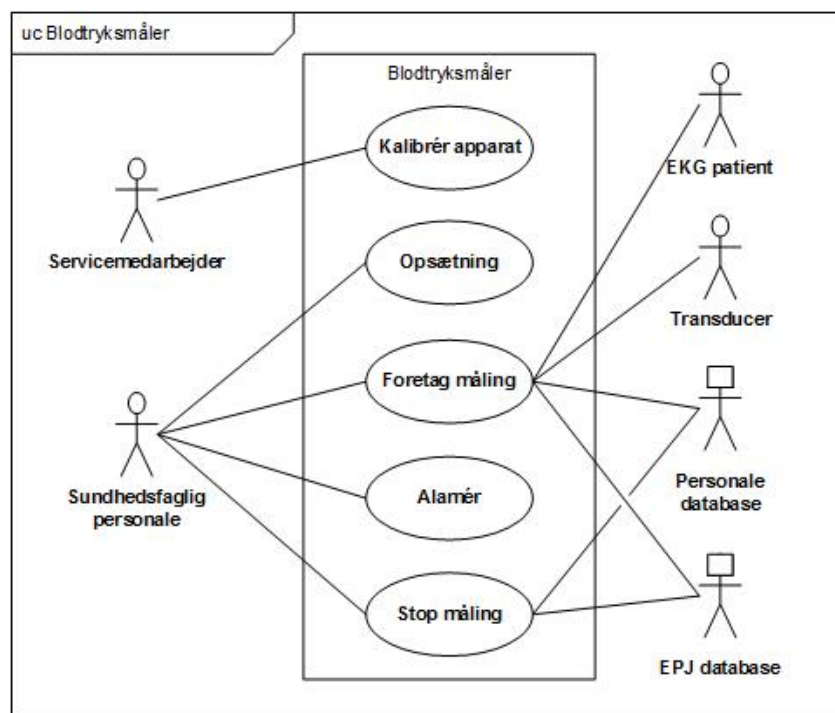
## Krav

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 leverer 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 under Kravspecifikation.



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 agerer 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**

Servicemedarbejderen skal i denne Use case foretage en kalibrering. Dette gøres ved at der er tre målepunkter på en væskesøjle, hvor en af disse vælges. På disse punkter måles spændingen (volt) og trykket (mmHg) aflæses fra væskesøjlen. Disse indtastes af Servicemedarbejderen ind i systemet igennem brugergrænsefladen; startskærmen. Herefter beregner systemet en kalibrerings værdi. Denne værdi bruges til at omskrive de indlæste værdier til en blodtrykskurve i mmHg.

### **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. [6]

**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

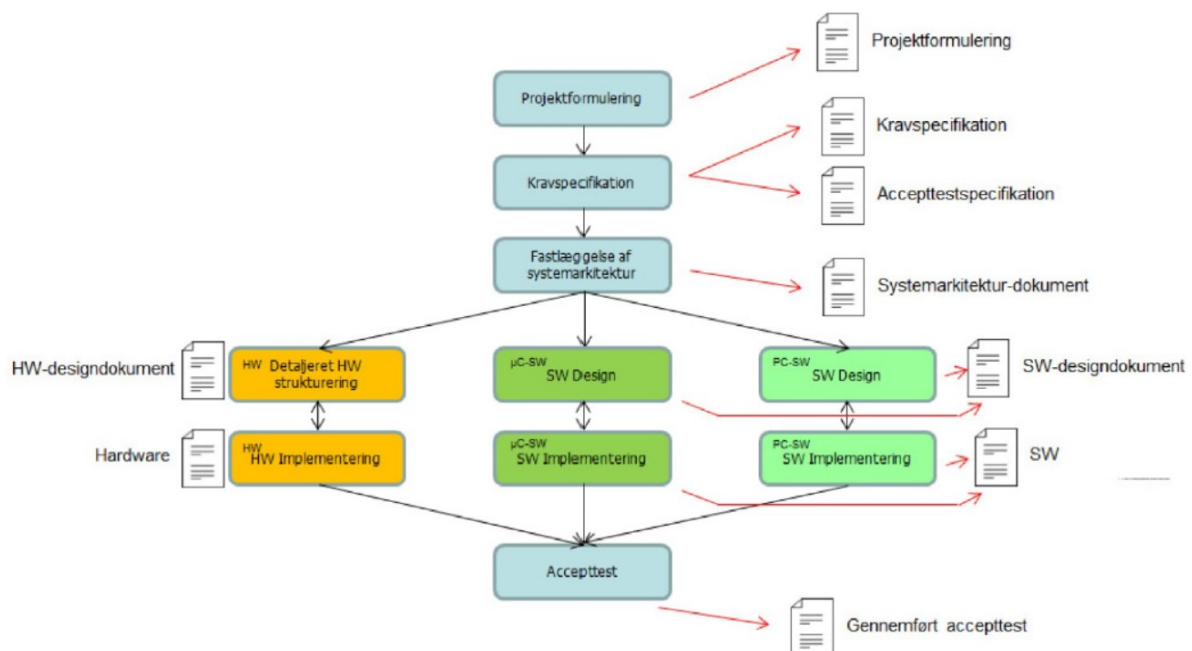
## Projektbeskrivelse

### 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.

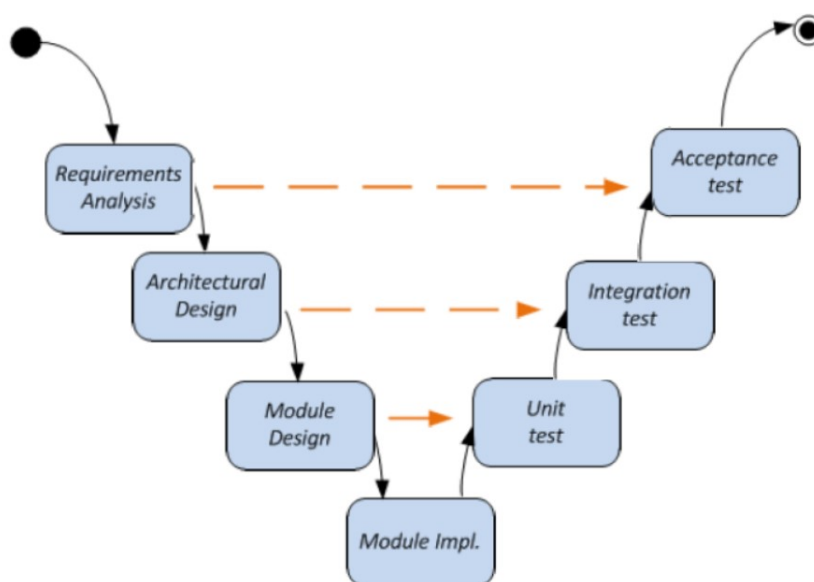


Figur 6.1: ASE-modellen.

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.

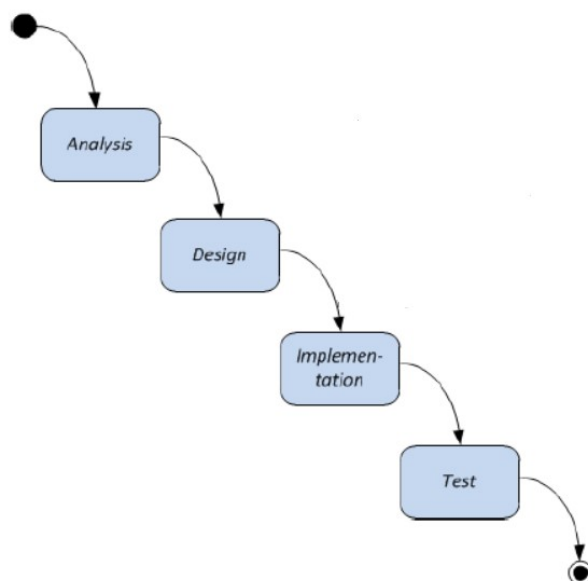


Figur 6.2: V-modellen.

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. Vandfalds-modellen er den proces, hvilken der er arbejdet ud fra i alle projektets facetter.

## Projektstyring

Ledelsen Tidsplan Opdeling Github SCRUM - Pivotal tracker

## 6.2 Metode

[7] For at beskrive den overordnede systemarkitektur og det detaljerede design for produktet, er der blevet benyttet SysML og UML. SysML er et grafisk modelleringssprog, hvilket kan hjælpe til at forstå og udvikle selv komplekse systemer. SysML udspringer af UML, dog benyttes SysML i højere grad ved systemer, der både indeholder software og hardware, end UML. UML benyttes til at beskrive softwaresystemets struktur og forløb. [5]

SysML's struktur-, funktionalitets og adfærdsdiagrammer er i projektet blevet benyttet til at specificere og dokumentere systemet og dets komponenter. Der anvendes to strukturdiagrammer; block definition diagram (BDD) og internal block definition diagram (IBD) for at vise systemstrukturen. Systemarkitekturen bliver delt op i blokke. BDD'et bruges til at vise forholdet mellem de forskellige blokke, deres komposition og forholdet mellem logiske- og fysiske enheder i systemet. IBD'et benyttes til at vise den forbindelse der er imellem de forskellige blokke der ses ud fra BDD'et. Dermed ses den vej signalet går igennem de forskellige enheder og hvordan disse enheder er koblet sammen.

For at kunne beskrive produktets funktionalitet er der blevet benyttet et Use case diagram. Dette diagram beskriver, hvordan systemet interagerer med de udefrakommende enheder (f.eks.



aktører) for at løse et antal af opsatte opgaver. Disse opgaver er beskrevet i forskellige Use cases. Et af adfærdsdiagrammerne er et sekvensdiagram, denne viser de handlinger der er mellem systemets dele (parts), hvilke er delt op i sekvenser.

For at beskrive overgangen fra de beskrevne Use cases til software, er applikationsmodellen benyttet. Applikationsmodellen tager udgangspunkt i Use cases og domænemodellen, hvor Use cases bruges til at identificere, hvad systemet skal kunne og hvordan det skal løses. Domænemodellen beskriver det overordnede systemdomæne. Systemdomænet beskrives ud fra de konceptuelle klasser, hvilke findes ud fra Use cases. Domænemodellen definere de klasser der skal være i softwaren og de interaktioner der er imellem disse, dette gøres med klassediagrammer og adfærdsdiagrammer.

En applikationsmodel med kontrolklasser, domæneklasser og boundaryklasser kan opbygges. Kontrolklasserne er funktionen at beskrive hvordan data behandles mellem domæne- og boundaryklasserne. Domæneklasserne indeholder funktionaliteten som det pågældende softwaremodul benytter, det er herfra klasserne der skal bruges i softwaren identificeres. Boundaryklasserne har den funktion at beskrive hvordan systemet kommunikerer med omverdenen og omvendt. Ved at benytte applikationsmodellen til design af softwarens opbygning gøres det lettere at udvikle et produkt med lav kobling samt høj samhörighed.

Systemets softwarestruktur og forløb beskrives ved at benytte UML klassediagrammer. Disse klassediagrammer viser den struktur der er mellem et objektorienteret systemklasse, og kan derfor bruges til at definere relationen mellem de forskellige klasser. Derfor dannes der et hurtigt overblik over sammenhængen i systemet ved at benytte klassediagrammet.

- 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/PULL mønster

PUSH/PULL er et software mønstre, der enten skubber eller trækker data op. PUSH betyder at skubbe, og betyder at data bliver skubbet op, når der sker en ændring, uden der skal gives besked om at hente data. PULL trækker data op imellem lagene. I PULL gives der besked til et lag om en ændring. Herefter skal laget der ønsker at kende til ændringen bede om ændringen,

før det sendes op.

I dette projekt er der valgt PUSH mønstret, da det er interessant at få målinger op til brugergrænsefalden hele tiden når, er indlæst nye målinger fra DAQ'en, og få sendte målingerne sammen med. Yderligere beskrivelse findes i projektdokumentationen under kapitlet Arkitektur og design under afsnittet Software design.

### Observer/Subject mønster

Observer/ subject mønstreret bruges til at sende data op kontinueret og på en fornuftig måde. Subject har en metode til at koble en observer til, og en anden metode til at fjerne observeren. Der er en tredje metode i subject, kaldet notify. Notify sørger for at give besked til den tilknyttet observer om opdatering. En observer er interesseret i at kende til ændring i et objekt. Beskeden om ændring kommer fra subjectet, der er tilknyttet observeren.

I dette projekt er der valgt at bruge observer/ subject mønstreret til at sende data fra datalag til logiklaget. Mønstreret er også brugt til at sende data videre fra logiklaget til præsentationslaget. Yderligere beskrivelse findes i projektdokumentationen under kapitlet Arkitektur og design under afsnittet Software design.

### Queue

For at sikre at data bliver sendt op i den rigtige rækkefølge og på en kontrolleret måde, er der i dette projekt valgt at benytte en kø. Til dette er der benyttet en indbygget funktion i Visual Studio, Queue class. Herunder ligger der to metoder, som kan bruges til at sende objekter af tal ind i den ene ende af køen og efter at køen er blevet fyldt op, er der en funktion til at fjerne det første objekt fra den anden ende af køen. Dette sørger for at der hele tiden er et flow i køen når der sendes data ind i denne.

En yderligere beskrivelse af hvordan Queue klassen benyttes i koden og hvordan denne er opbygget er beskrevet i dokumentationen under Arkitektur og design under afsnittet Implementering

### Trelagsmodel

Trelagsmodellen er en software model, til at inddele sin software i tre lag, præsentationslaget, logiklaget og datalaget. Hver lag har sit eget ansvar og funktionalitet. Præsentationslaget kan kun tilgå logiklaget, og må kun vise data på brugergrænseflade eller indlæse data fra brugergrænsefladen. Logiklaget må tilgå både datalaget og præsentationslaget. Logik er det midterste lag, og står for behandling af data fra præsentationslaget og datalaget. Datalaget er det nederste lag og sørge for at sende data fra et måle apparat op til logiklaget. Datalaget står også for at gemme data i en fil eller en database.

I dette projekt er koden opdelt efter trelagsmodellen, ved at have et præsentationslag med to brugergrænsefald, et logiklag med algoritmer til udregning af blodtryk, og et datalag der hente data fra DAQ'en og gemmer dem i en database. Yderligere beskrivelse og model over trelagsmodellens inddeling af software koden findes i projektdokumentationen under kapitlet Arkitektur og design under afsnittet Software design.

## 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 kompiller. 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 EKG 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**

# Litteratur

## 8.1 Referencer

### Billeder

- [1] Kredsløb og de vigtigste organer, fc.vucnordjylland.dk, læst d. 8. december 2015, <http://fc.vucnordjylland.dk/Flexundervisning/F0V1-000D35E9/F0V1-000948AA/Modul%204/Inspiration/I062B76D7>

### Bøger

### Internetsider

- [2] Blodtryk, Wikipedia.org, læst d. 7. december 2015, <https://da.wikipedia.org/wiki/Blodtryk>
- [3] Hypertension, Sundhed.dk, skrevet 4. august 2015, læst d. 7. december 2015, <https://www.sundhed.dk/sundhedsfaglig/laegehaandbogen/hjerte-kar/tilstande-og-sygdomme/oevrige-sygdomme/hypertension/>
- [4] Forskelle mellem systemiske & pulmonale cirkulation, Michael Kristensen, skrevet d. 24. maj 2015, læst d. 8. december 2015, <http://www.sundhedguide.com/2/2015/05/forskelle-mellem-systemiske-pulmonale-cirkulation.html>
- [5] UML, Wikipedia.org, Tim Ottinger og Jeff Langr, læst d. 9. december 2015, <http://da.wikipedia.org/wiki/UML>
- [6] FURPS+, læst d. 9. december 2015, <http://agileinaflash.blogspot.dk/2009/04/furps.html>

### Dokumenter

- [7] A practical guide to SysML, Sanford Friedenthal, Alan Moore og Rich Steiner, s. 29-60.