



AARHUS SCHOOL OF ENGINEERING

SUNDHEDSTEKNOLOGI
2. SEMESTERPROJEKT

Dokumentation

Gruppe 1

Lise Skytte Brodersen (201407432)
Mads Fryland Jørgensen (201403827)
Albert Jakob Fredshavn (201480425)
Malene Cecilie Mikkelsen (201405722)
Mohamed Hussein Mohamed (201370525)
Sara-Sofie Staub Kirkeby (201406211)
Martin Banasik (201408398)
Cecilie Ammizbøll Aarøe (201208778)

Vejleder

Studentervejleder
Lars Mortensen
Aarhus Universitet

30. april 2015

Gruppemedlemmer

| | |
|--------------------------------------|------|
| Lise Skytte Brodersen (201407432) | Dato |
| Mads Fryland Jørgensen (201403827) | Dato |
| Albert Jakob Fredshavn (201480425) | Dato |
| Malene Cecilie Mikkelsen (201405722) | Dato |
| Mohamed Hussein Mohamed (201370525) | Dato |
| Sara-sofie Staub Kirkeby (201406211) | Dato |
| Martin Banasik (201408398) | Dato |
| Cecilie Ammitzbøll Aarøe (201208778) | Dato |

Vejleder

| | |
|----------------|------|
| Lars Mortensen | Dato |
|----------------|------|

Ordliste

Indholdsfortegnelse

| | |
|--|------------|
| Ordliste | iii |
| Kapitel 1 Kravspecifikation | 1 |
| 1.1 Indledning | 1 |
| 1.2 Funktionelle krav | 1 |
| 1.2.1 Aktør-kontekstdiagram | 1 |
| 1.2.2 Aktørbeskrivelse | 2 |
| 1.2.3 Use case-diagram | 2 |
| 1.2.4 Use Cases | 3 |
| 1.3 Ikke-funktionelle krav | 6 |
| 1.3.1 (F)URPS+ | 6 |
| Kapitel 2 Design | 9 |
| 2.1 Indledning | 9 |
| 2.2 Hardware arkitektur | 9 |
| 2.2.1 Græseflader | 10 |
| 2.3 Software arkitektur | 10 |
| 2.3.1 GUI | 10 |
| 2.3.2 UML klassesdiagram | 13 |
| 2.3.3 Applikationsmodel | 13 |
| Kapitel 3 Accepttest | 19 |
| 3.1 Accepttest af Use Cases | 19 |
| 3.1.1 Use Case 1 | 19 |
| 3.1.2 Use Case 2 | 19 |
| 3.1.3 Use Case 3 | 20 |
| 3.1.4 Use Case 4 | 21 |
| 3.1.5 Use Case 5 | 21 |
| 3.2 Accepttest af ikke-funktionelle krav | 21 |

Kravspekifikation

1

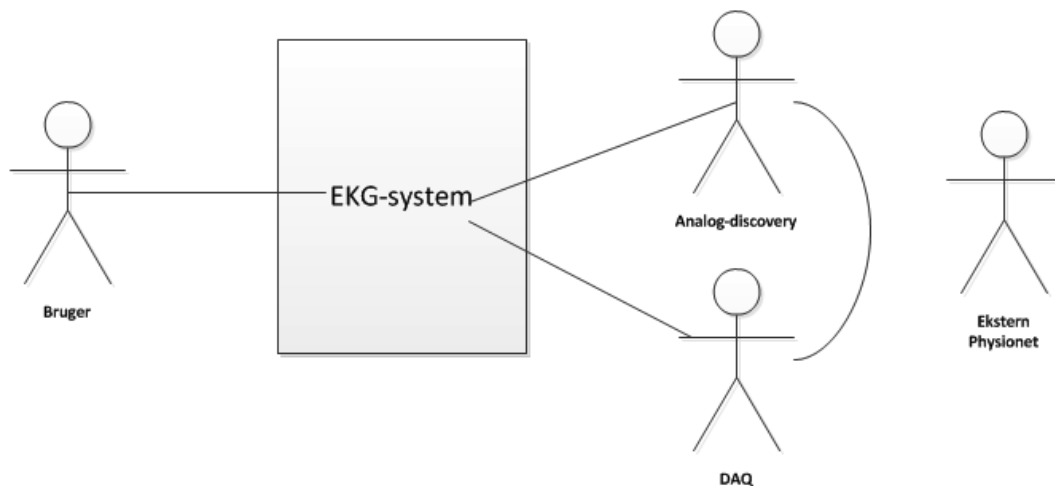
1.1 Indledning

Kravspekifikationen vil beskrive, ud fra en række modeller, hvordan EKG-systemet fungerer. Helt generelt er EKG-måling en simpel metode, til at måle hjertets elektriske aktivitet via elektroder, som registrerer elektriske impulser, placeret på huden. Ud fra disse impulser dannes en graf, som benyttes til at analysere hjertets funktionalitet ud fra P-, Q-, R-, S- og T-takkerne, og dermed konkludere om den pågældende patient har et raskt eller sygt hjerte, samt hvilken sygdom der er tale om. Helt specifikt for denne opgave er formålet, at identificere sygdommen atrieflimmer via et virtuelt EKG-signal.

1.2 Funktionelle krav

De funktionelle krav vil nedenstående beskrives ud fra Aktør-kontekstdiagram, aktørbeskrivelse, Use Cases samt Use Case diagram.

1.2.1 Aktør-kontekstdiagram



Figur 1.1: Aktør-kontekstdiagram

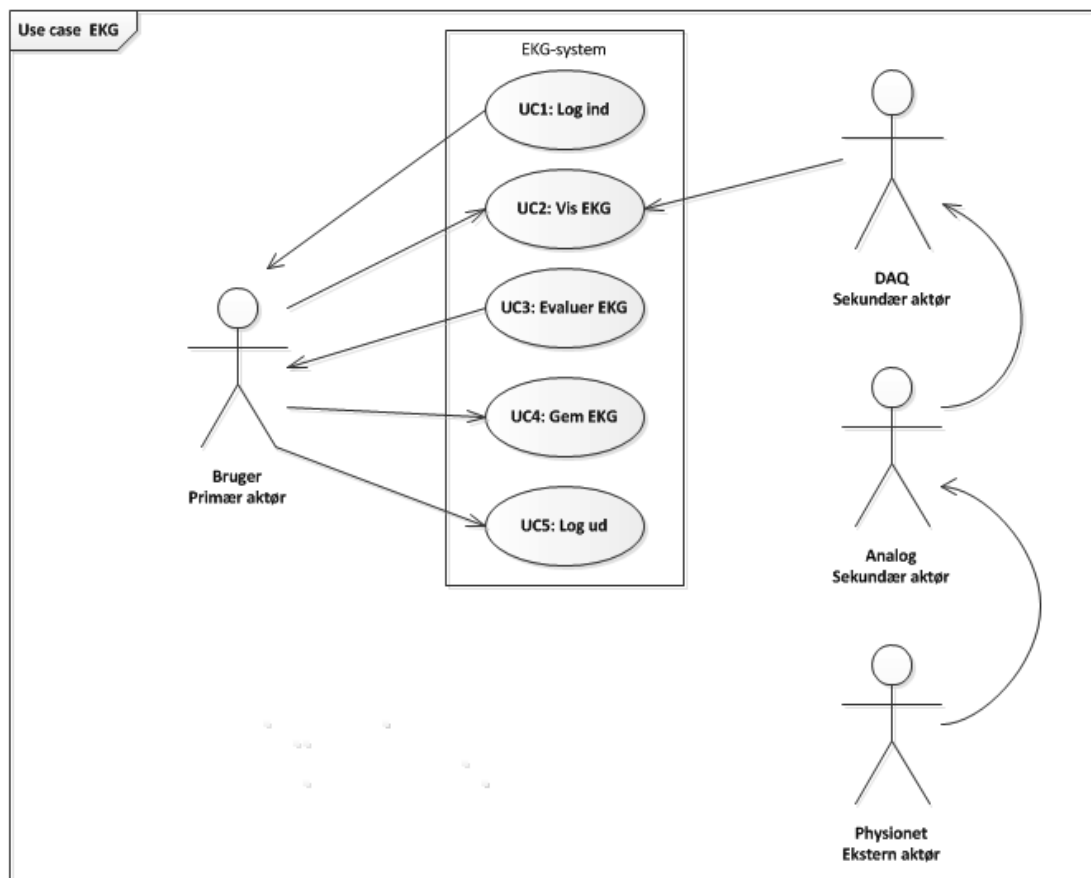
Data hentes ned fra den ekstern aktør, physionet, og via Analog-discovery omdannes csv-filens data til et analogt signal, der sendes til EKG-systemet. Ud fra disse data danner EKG-systemet en graf. Programmet detekterer markørudsving i EKG-grafen, som derefter valideres og analyseres af brugeren.

1.2.2 Aktørbeskrivelse

| Aktørnavn | Type | Beskrivelse |
|------------------|----------|---|
| Bruger | Primær | Brugeren er den aktør, der ønsker at foretage målinger, som omfatter EKG samt diagnosticering af ar-tieflimmer. Brugeren er en person, der har kendskab til EKG-systemet. Fx sundhedsfaglig personale |
| Analog-discovery | Sekundær | Analog-discovery omdanner data fra den eksterne aktør, physionet, til et analog signal |
| DAQ | Sekundær | DAQ'en omdanner det analoge signal fra analog-discovery til et digitalt signal, som EKG-systemet kan generere en graf ud fra |
| Physionet | Ekstern | Physionet er en database, hvor der ligger mange forskellige EKG-signaler. Det er ud fra disse EKG-signaler, virtuelle patienter skabes. |

Tabel 1.1: Aktørbeskrivelse

1.2.3 Use case-diagram



Figur 1.2: Use case-diagram

Brugeren, den primære aktør bliver bedt om sit log ind, inden EKG-vinduet vises. Brugeren vælger indstillinger og trykker på "start-knappen. EKG-dataerne fra den eksterne aktør, Physionet, behandles i Analog samt i DAQ'en, de sekundære aktør, hvor efter data vises som en EKG-graf i EKG-vinduet. Brugeren kan ud fra denne graf evaluere EKG-signalet i forhold til at diagnosticere atrieflimmer. Brugeren gemmer EKG-målingen i databasen og logger ud.

1.2.4 Use Cases

Use Case 1

| | |
|------------------|---|
| Navn | Log ind |
| Use case ID | 1 |
| Samtidige forløb | 1 |
| Primær aktør | Brugeren |
| Initialisere | Brugeren ønsker at logge ind |
| Forudsætninger | At der er logget ud efter en tidligere måling |
| Resultat | Brugeren bliver logget på og kan foretage en måling |
| Hovedforløb | <ol style="list-style-type: none"> 1. Brugeren indtaster username samt password 2. Brugeren trykker på "Login-knappen". Login-vinduet lukkes ned mens CPR-vinduet åbnes [2.a <i>Username eller password er forkert</i>] |
| Undtagelser | <ol style="list-style-type: none"> 2a. Besked vises på skærmen med tekst, der informerer om, at username eller password er forkert. Der forsættes i UC1 ved punkt 1 |

Tabel 1.2: Fully dressed Use Case 1.

Use Case 2

| | |
|------------------|----------|
| Navn | Vis EKG |
| Use case ID | 2 |
| Samtidige forløb | 1 |
| Primær aktør | Brugeren |
| Sekundær aktør | Analog |
| Sekunær aktør | DAQ |

| | |
|----------------|---|
| Ekstern aktør | Physionet |
| Initialisere | Brugeren ønsker at foretage en EKG-måling |
| Forudsætninger | Brugeren er logget ind og EKG-vinduet er vist samt Analog og DAQ'en er koblet til og data er hentet ned |
| Resultat | EKG-graf bliver vist |
| <hr/> | |
| Hovedforløb | <ol style="list-style-type: none"> 1. Brugeren indtaster virtuel patients CPR-nummer [1.a <i>CPR-nummeret findes ikke</i>] 2. Brugeren vælger indstillinger [2.a <i>Brugeren er tilfreds med default-indstillingerne</i>] 3. Målingen startes ved at trykke på "Start" 4. EKG-data illustreres på en graf |
| <hr/> | |
| Undtagelser | <ol style="list-style-type: none"> 1a. CPR-nummeret findes ikke. Besked vises på skærmen med tekst, der informerer om, at CPR- nummeret ikke er gyldigt. UC2 startes forfra med nyt CPR-nummer 2a. Der blev ikke ændret i default-indstillingerne. Der fortsættes ved punkt 2 i hovedforløbet med default indstillingerne |
| <hr/> | |

Tabel 1.3: Fully dressed Use Case 2.

Use Case 3

| | |
|------------------|---|
| Navn | Evaluer EKG |
| Use case ID | 3 |
| Samtidige forløb | 1 |
| Primær aktør | Brugeren |
| Initialisere | Use Case 2 er gennemført |
| Resultat | Brugeren kan ud fra EKG-graf diagnosticere sygdommen atrieflimmer |
| <hr/> | |
| Hovedforløb | <ol style="list-style-type: none"> 1. Brugeren validere programmets analyse af EKG-signalet 2. Brugeren stiller diagnosen atrieflimmer [2.a <i>Atriefrekvensen er ikke i intervallet 220-300 pr. minut</i>] |
| <hr/> | |
| Undtagelser | <ol style="list-style-type: none"> 2a. Det er ikke muligt at diagnosticere atrieflimmer ud fra grafen. Use case 3 afsluttes og Use case 2 gentages med evt. nye tidsindstillinger |
| <hr/> | |

Tabel 1.4: Fully dressed Use Case 3.

Use Case 4

| | |
|------------------|--|
| Navn | Gem EKG |
| Use case ID | 4 |
| Samtidige forløb | 1 |
| Primær aktør | Brugeren |
| Initialisere | Brugeren ønsker at gemme EKG i databasen |
| Forudsætninger | Use case 3 er gennemført |
| Resultat | EKG er gemt i databasen |
| Hovedforløb | <ol style="list-style-type: none"> 1. Brugeren trykker på "Gem-knappen". En messagebox kommer frem med besked om at data er gemt 2. Brugeren trykker på "Ok"-knappen for at lukke messageboxen og EKG-vinduet vises igen |
| Undtagelser | |

Tabel 1.5: Fully dressed Use Case 4.

Use Case 5

| | |
|------------------|---|
| Navn | Log ud |
| Use case ID | 5 |
| Samtidige forløb | 1 |
| Primær aktør | Brugeren |
| Initialisere | Brugeren ønsker at logge ud |
| Forudsætninger | Der skal være logget ind |
| Resultat | Brugeren bliver logget ud, og EKG-vinduet lukkes og login-vinduet fremkommer |
| Hovedforløb | <ol style="list-style-type: none"> 1. Brugeren trykker på "log ud-knappen"og EKG-vinduet lukkes, mens login-vinduet fremkommer |
| Undtagelser | |

Tabel 1.6: Fully dressed Use Case 5.

1.3 Ikke-funktionelle krav

De ikke-funktionelle krav er udarbejdet ved brug af (F)URPS+. De er alle prioriteret ved MoSCoW metoden - Must (skal være med), Should (bør være med, hvis muligt), Could (kunne have med, hvis det ikke influerer på andet), Won't/Would (ikke med nu, men med i fremtidige opdateringer).

1.3.1 (F)URPS+

MoSCoW er angivet i parentes med hhv. M, S, C eller W.

Usability

- (M) Brugeren skal kunne starte en default-måling maksimalt 20 sek. efter opstart af programmet
- (M) Brugeren skal have mulighed for at ændre tidsintervallet før målingerne foretages
- (M) Login-vinduet skal indholde en "login-knap til at logge på og få vist EKG-vinduet
- (M) EKG-vinduet skal indeholde en "start-knap til at igangsætte målingerne
- (M) EKG-vinduet skal indeholde en "stop-knap til at afslutte målingerne før den valgte tid
- (M) EKG-vinduet skal indeholde en "log ud-knap
- (M) EKG-vinduet skal indeholde en "gem-knap
- (M) Information-vinduet skal indeholde en "gem-knap
- (M) Målingen stopper automatisk efter det valgte tidsinterval

Reliability

- (M) Systemet skal have en effektiv MTBF (Mean Time Between Failure) på 20 minutter og en MTTR (Mean Time To Restore) på 1 minut.

$$Availability = \frac{MTBF}{MTBF + MTTR} = \frac{20}{20 + 1} = 0,952 = 95,2\% \quad (1.1)$$

Performance

- (M) Der skal vises en EKG-graf i EKG-vinduet, hvor spænding vises op af y-aksen (-1V til 1V) og tiden på x-aksen
- (M) Grafen skal være scrollbar på x-aksen, så brugeren selv ved brug af musen kan vælge det udsnit af grafen, der skal vises mere detaljeret

- (M) Skal tage en sample over et brugerbestemt interval, hvor frekvensen er tilpasset målingerne, således at grafen er analyserbar

Supportability

- (M) Softwaren er opbygget af trelagsmodellen

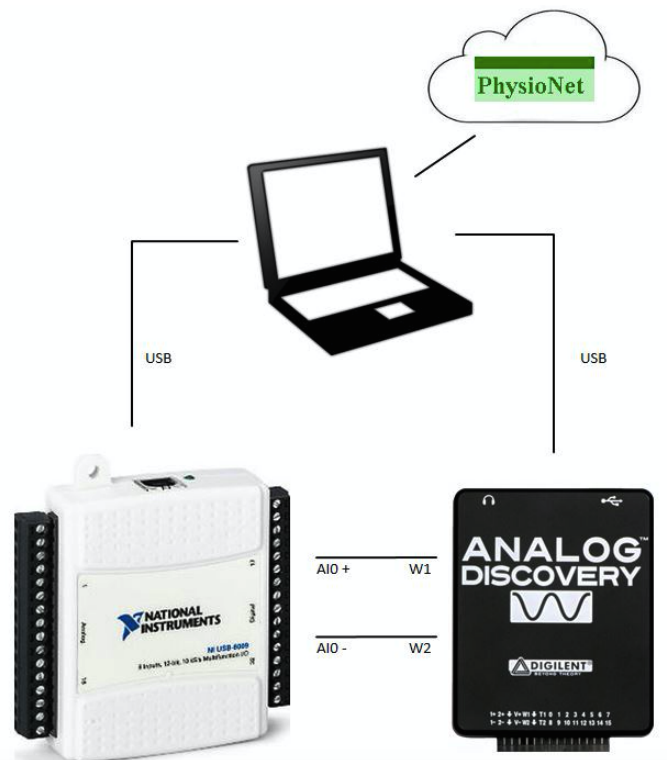
2.1 Indledning

Formålet med design afsnittet er at beskrive hardwaren og softwaren for EKG-systemet. Det beskrives med samtlige diagrammer og skitser for, at de forskellige klassers funktionalitet, og hvordan de snakker sammen klargøres.

2.2 Hardware arkitektur

I dette projekt er hardwaren, blevet udleveret, så beskrivelsen af dette bliver, derfor ikke beskrevet i yderste detalje. Hardwaren for systemet består af en National Instruments DAQ og en signalgenerator fra Analog Discovery. Disse er begge forbundet til en computer via USB-porte.

I dette projekt bliver EKG-signalet ikke fysisk målt fra en patient, men simuleret fra et signal, som er hentet ned fra websiden PhysioNet ¹.



Figur 2.1: Grafisk illustration af hardware opsætning

¹<http://www.physionet.org>

Som det ses på det overstående figur, er denne opstilling meget simpelt. Figur 2.1 viser, hvordan de forskellige komponenter relaterer til hinanden. Display hardwaren vil i dette projekt være en computer, da det kræver softwaren for at kunne vise EKG-signalet i form af en graf.

Som beskrevet ovenfor er det Analog Discovery sammen med information for et EKG-signal fra PhysioNet, der skaber vores fiktive patient. Før EKG-signalet vises på computerens skærm skal det igennem de forskellige komponenter i opstillingen. Analog Discovery modtager EKG-signals information som en CSV-fil, denne omdanner Analog Discovery til et analogt signal, som sendes videre til DAQ'en. DAQ'en digitaliserer det analoge signal, og tilpasser det, så det kan vises på computer skærmen.

2.2.1 Grænseflader

Grænseflader af forbindelserne imellem de forskellige dele af hardwaren.

| Forbindelse | Signaltype | Funktionalitet |
|-----------------------------|------------|---|
| DAQ - Computer | Digital | DAQ'en konverterer det analoge signal til digitalt og videresender det til computeren. Informationen sendes begge veje. |
| Computer - Analog Discovery | Digital | Computeren simulerer et EKG-signal og sender det til Analog Discovery. |
| Analog Discovery - DAQ | Analog | Analog Discovery konverterer signalet fra digitalt til analogt og videresender det til DAQ'en. |

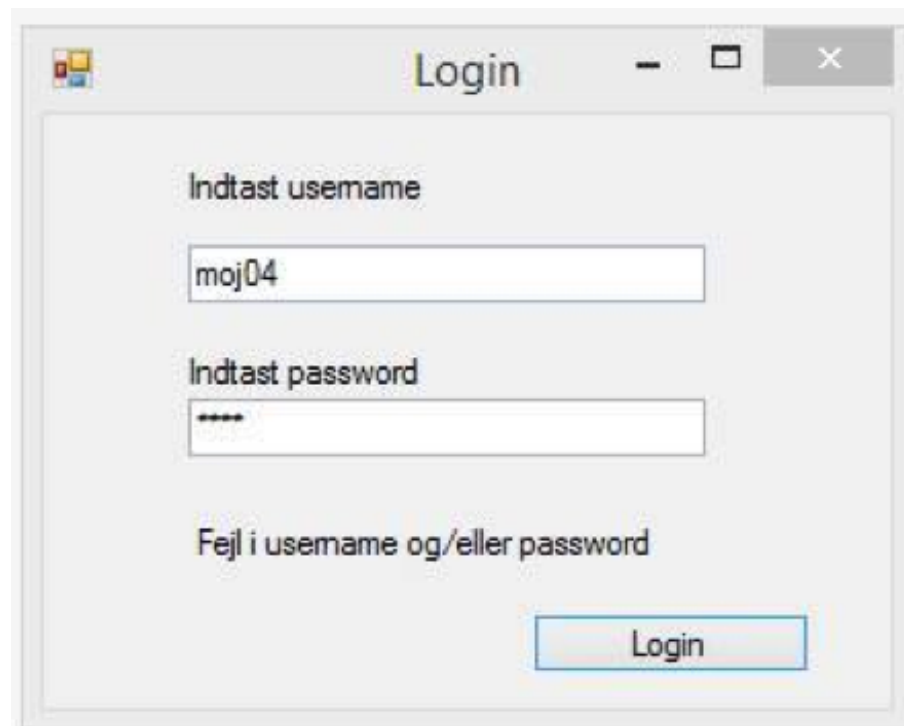
Tabel 2.1: Beskrivelse af grænseflader.

2.3 Software arkitektur

I dette projekt arbejdes der med objektorienteret programmering i programmeringssproget C#. Dette er opbygget i henhold til trelagsmodellen, som er beskrevet i Rapporten under afsnittet Baggrund. Grafiklaget består af 4 GUI'er, logiklaget af en enkelt klasse, og datalaget består af 3 klasser, hvoraf den ene er en blackbox.

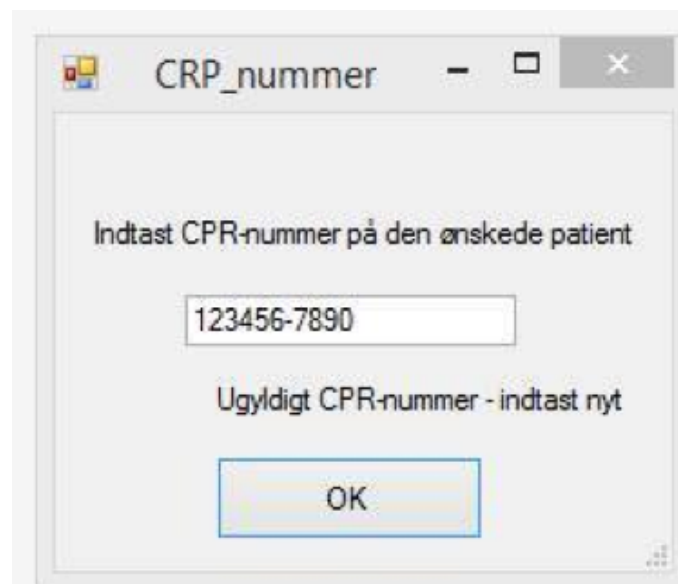
2.3.1 GUI

I dette afsnit vil GUI'erne nærmere beskrives. Grafiklaget består i alt af 4 GUI'er eller vinduer - Login-vindue, CPR-vindue, EKG-vindue og Gem_måling-vindue.



Figur 2.2: Login-vindue

Når programmet startes op, vil et Login-vindue fremkomme på skærmen. Her skal brugeren logge ind for at få adgang til programmet. Hvis password og username ikke bliver godkendt, returnerer vinduet en fejl, i form af en linje tekst nedenunder indtast-felterne. Bliver login godkendt åbner CPR-vinduet.

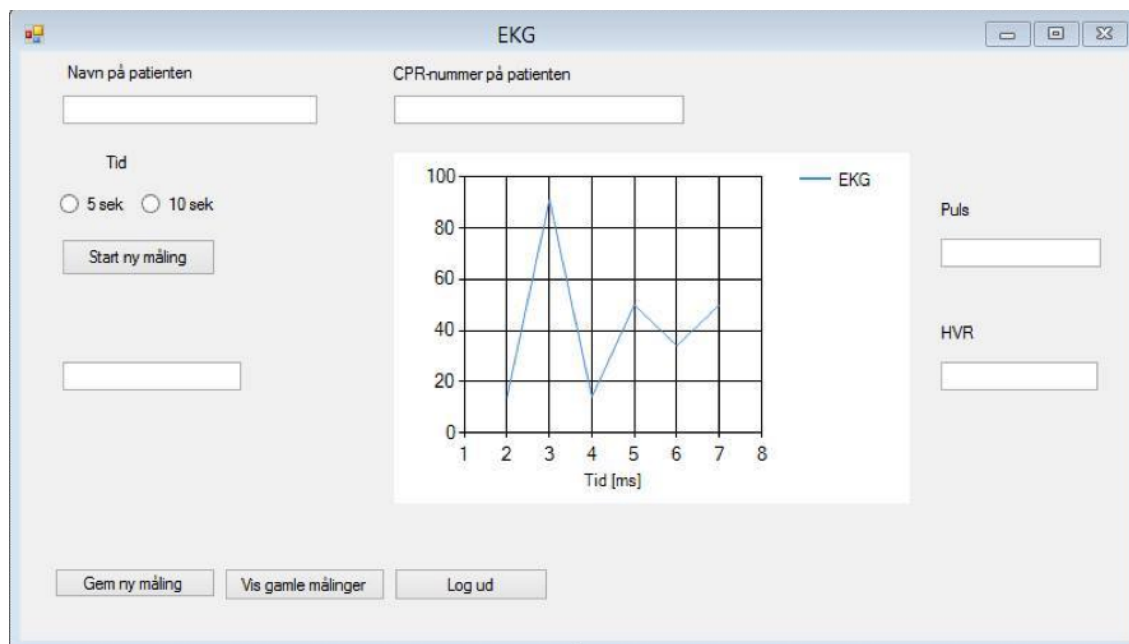


Figur 2.3: CPR-vindue

Her bliver brugeren bedt om at indskrive CPR-nummeret på den fiktive patient. CPR-nummeret bruges til at identificere personen i forbindelse med fremtidig måling samt

gemme målingerne. Hvis CPR-nummeret ikke bliver godkendt, vil brugerne blive gjort opmærksom på dette via en fejlmelding.

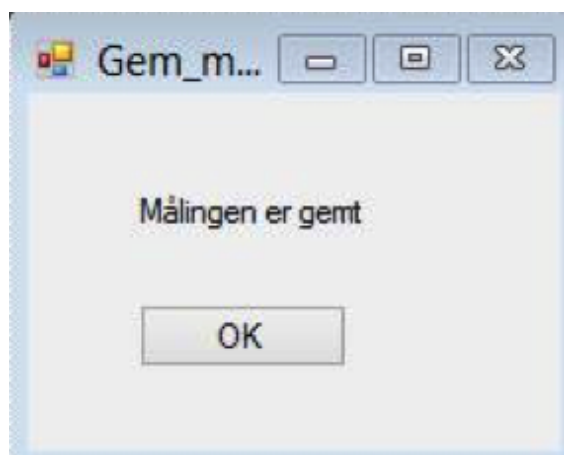
Når CPR-nummeret bliver godkendt, åbner det primære vindue, EKG-vinduet.



Figur 2.4: EKG-vindue

EKG-vinduet er, hvor man kan køre og se målinger fra. Der er knapper til at vælge, hvor lang tid man ønsker en måling skal køre over. Til højre kan grafen for EKG-signalet ses, efter der er trykket "Start ny måling". Yderligere til højre kan der ses puls og HVR for den pågældende graf. Nederst er der knapper til at gemme målingerne, samt at logge ud fra programmet. Hvis man logger ud, kommer man tilbage til Login-vinduet.

Det sidste vindue der er, forekommer når man trykker på "Gem ny måling". Her kommer der et pop-up vindue op, som bekræfter, at EKG-målingen er gemt.



Figur 2.5: Gem_måling-vindue

2.3.2 UML klassediagram

2.3.3 Applikationsmodel

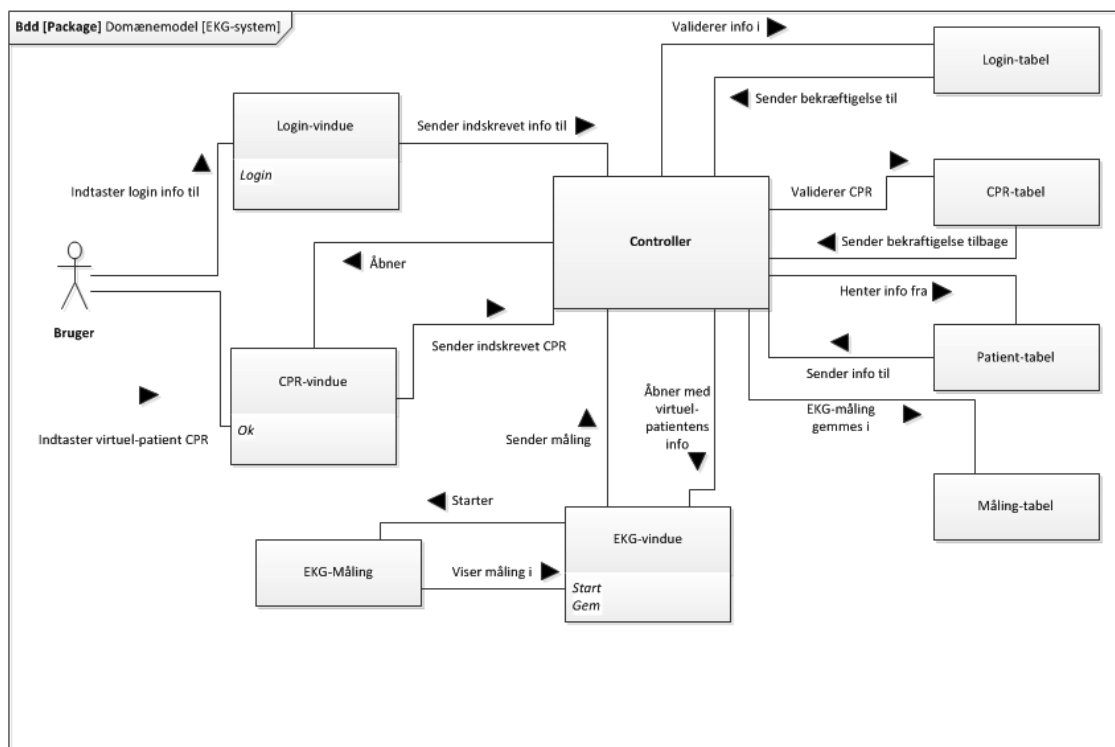
Applikationsmodel vil i almindelig forstand indeholde en overordnet domænemodel, herefter klassediagram samt sekvensdiagram og tilsidst et opdateret klassediagram, hvor metoderne fra sekvensdiagram er inkluderet.

Applikationsmodellen for dette projekt består af en overordnet domænemodel, herefter sekvensdiagram for de forskellige Use Cases med tilhørende opdateret klassediagram, som beskriver metodekald og kommunikation mellem klasserne. Klassediagrammerne er undladt, da de er irrelevante for dokumentationen for projektet. Applikationsmodellen er udarbejdet udfra Use Cases, hvilket medfører at metoderne er fiktive, altså ikke hentet direkte fra softwaren.

Domænemodel

Domænemodellen er skabt på baggrund af de fem Use Cases. Gennem navneordanalyse af Use Casene er de konceptuelle klasser fundet. I modellen beskrives, hvordan de konceptuelle klasser interagerer med hinanden. Controlleren er ikke en konceptuelle klasse, men er den, der sørger for at systemet fungerer optimalt.

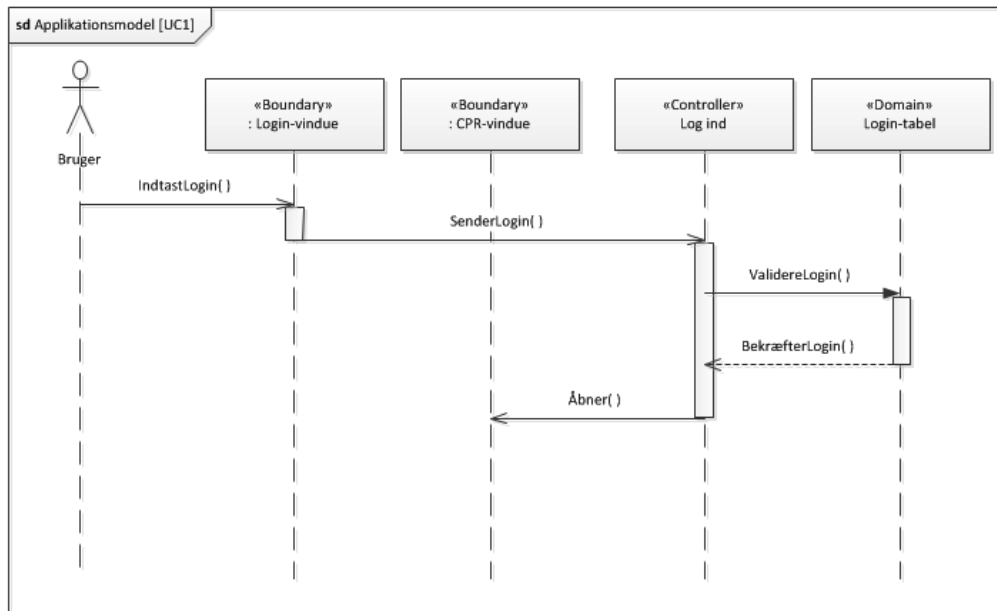
Der er ingen multiplicity indsat i modellen, da der kun arbejdes med et scenarie af gangen. !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! spørg Lars



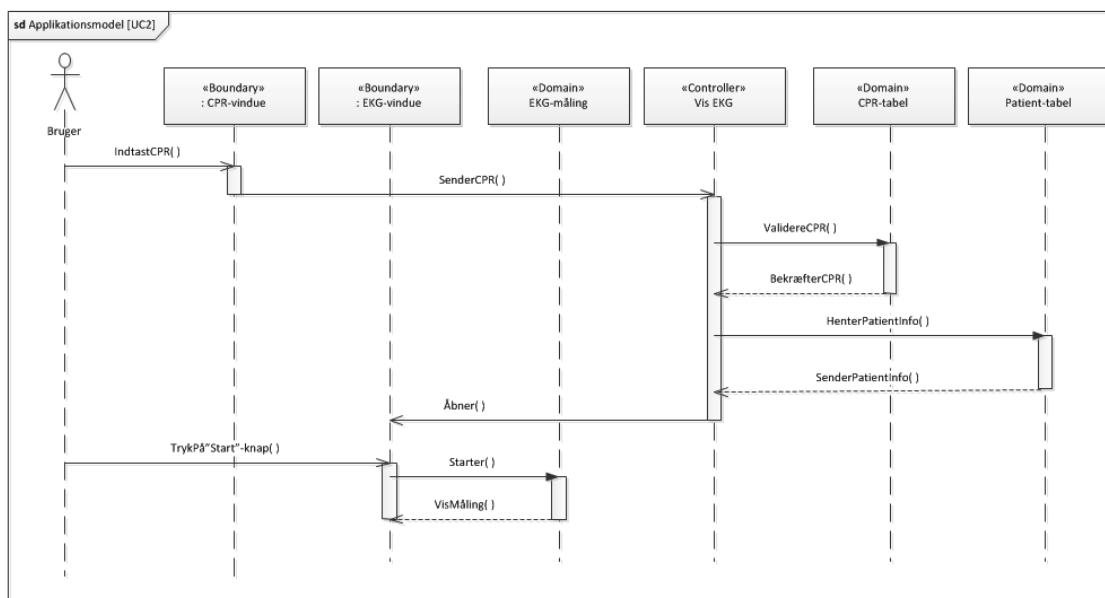
Figur 2.6: Domænemodel af EKG-systemet

Sekvensdiagram

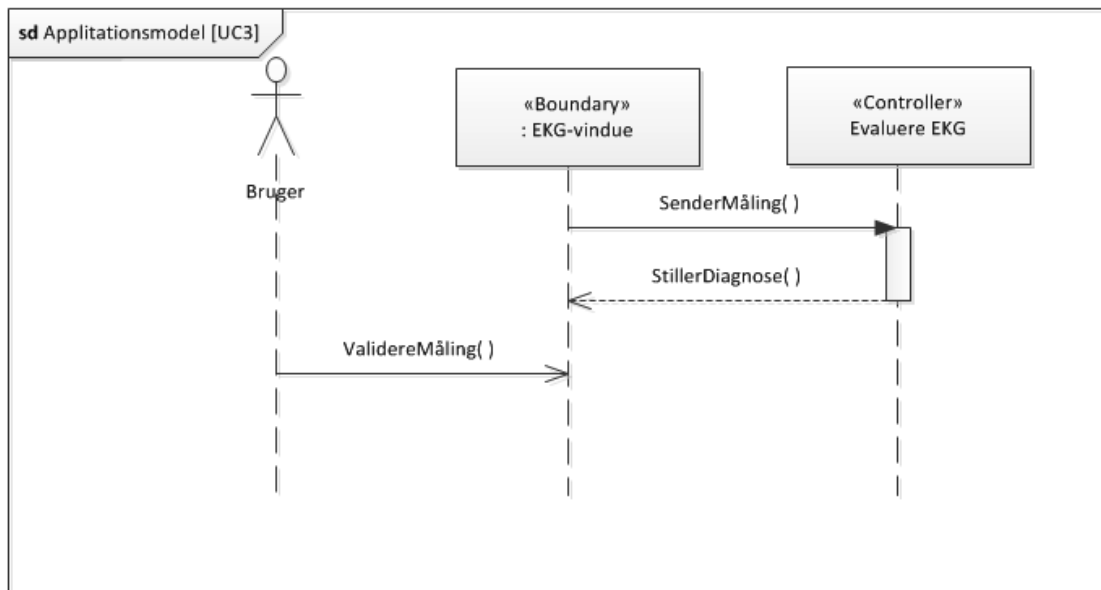
Sekvensdiagrammerne beskriver step-by-step via fiktive metoder forløbet i de forskellige Use Cases. Der er lavet et sekvensdiagram for hver Use Cases for at gøre systemet mere overskueligt. Et sekvensdiagram består af boundary-klasser og domain-klasser fra domænemodellen, samt en controller-klasse, som har navn efter den specifikke Use Case.



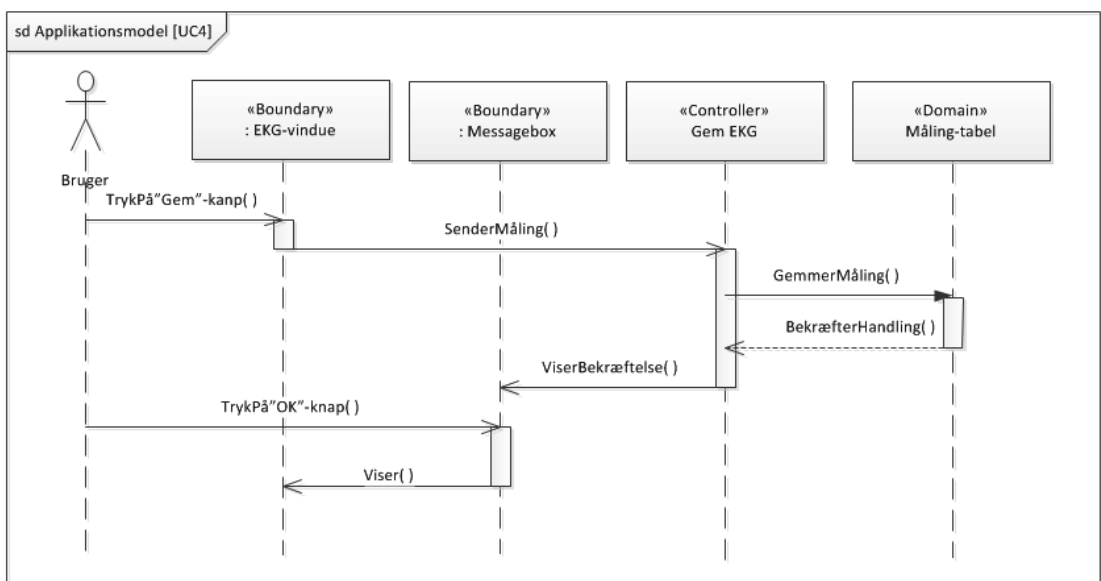
Figur 2.7: Sekvensdiagram for UC1



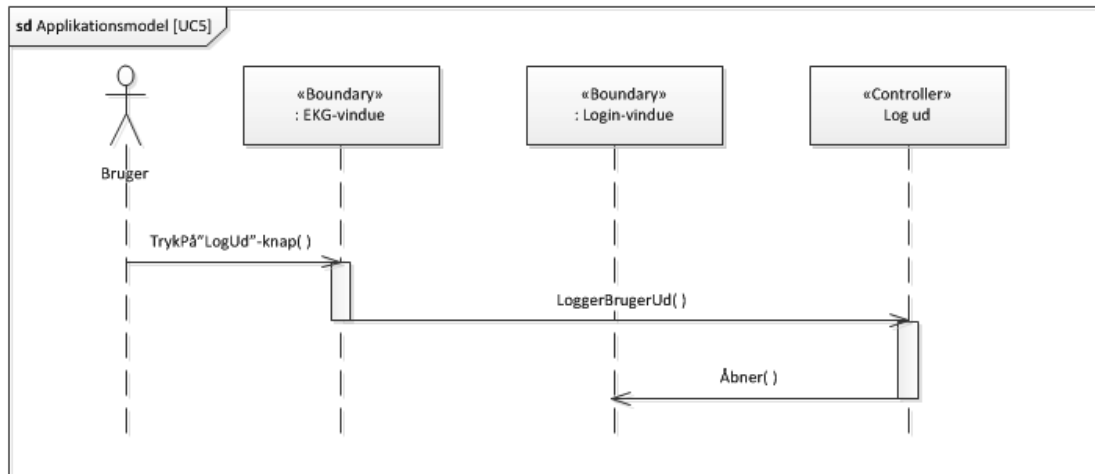
Figur 2.8: Sekvensdiagram for UC2



Figur 2.9: Sekvensdiagram for UC3



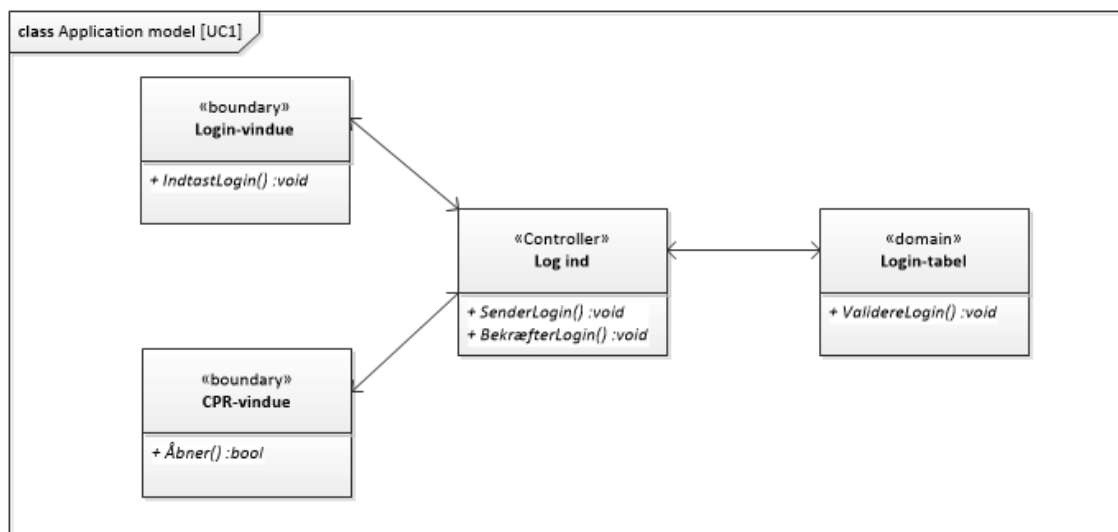
Figur 2.10: Sekvensdiagram for UC4



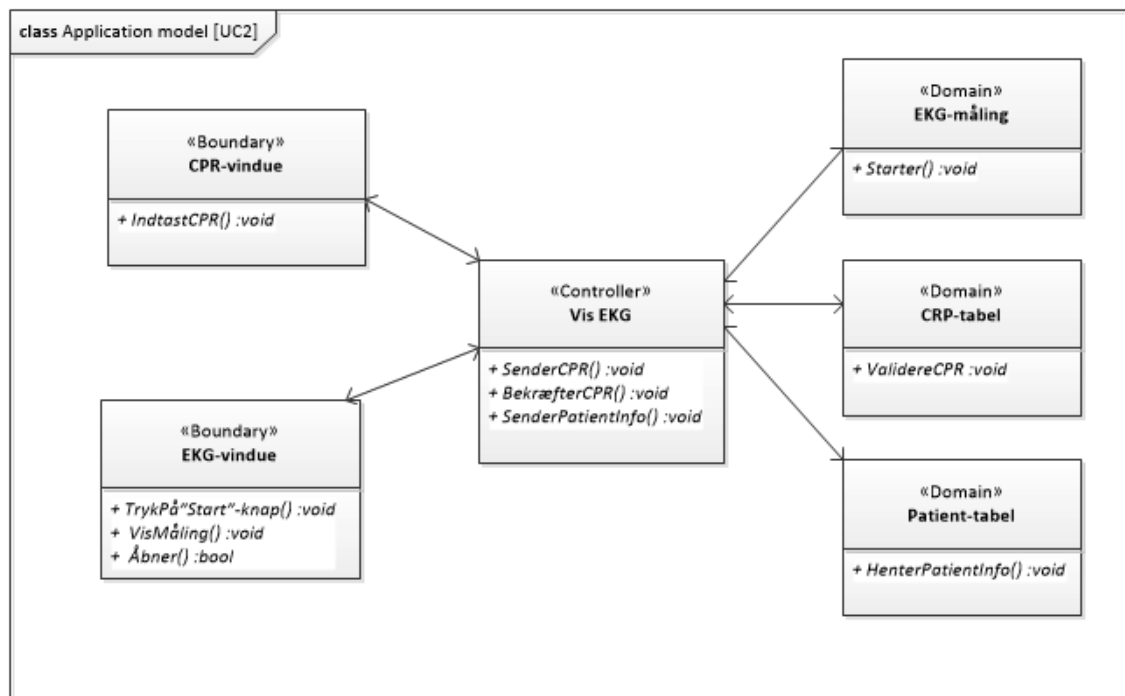
Figur 2.11: Sekvensdiagram for UC5

Opdateret Klassediagram

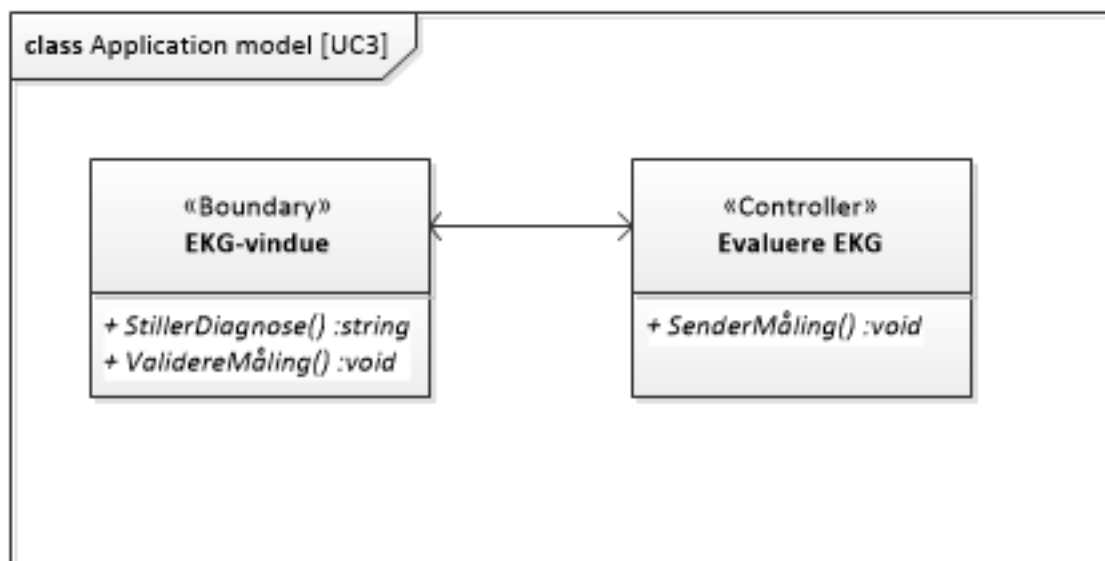
De opdateret klassediagrammer indeholder metoderne fra de dertilhørende sekvensdiagrammer - dette giver et overblik over, hvilke metoder de forskellige klasser består af.



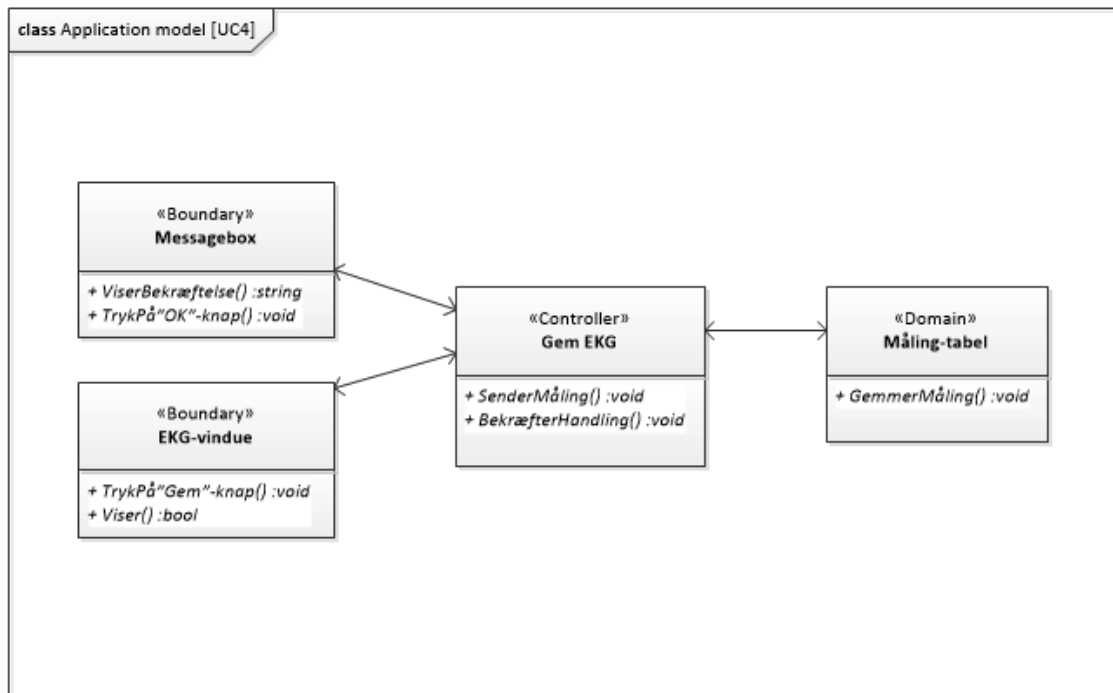
Figur 2.12: Klassediagram for UC1



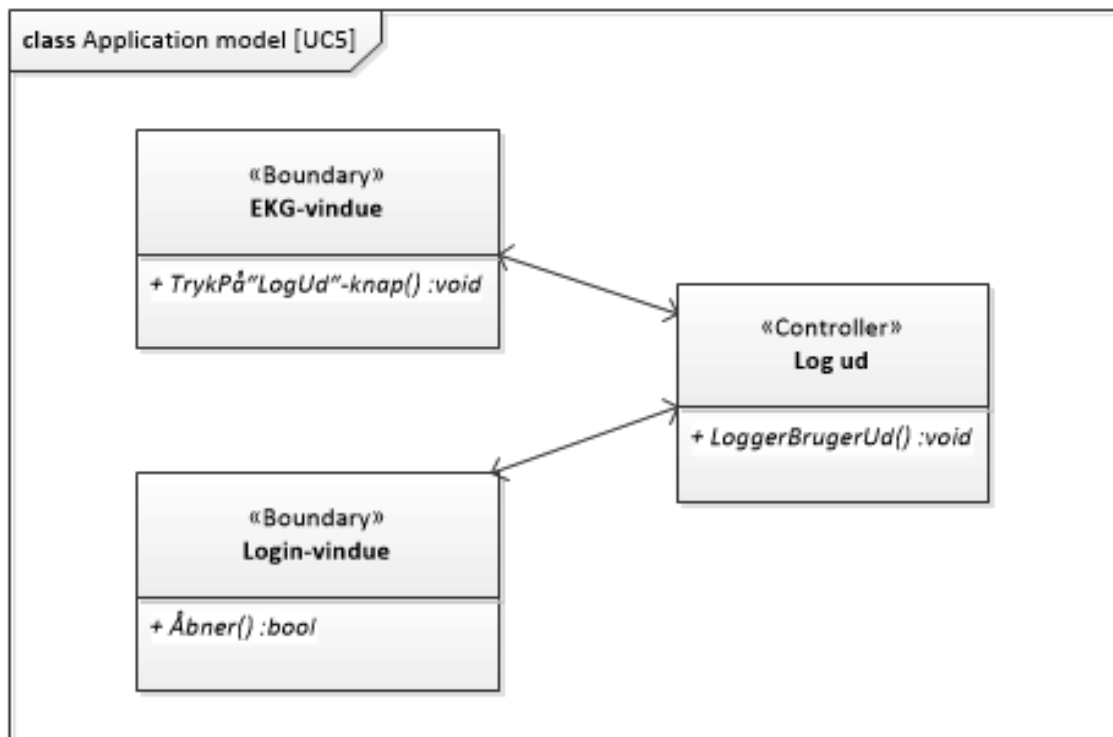
Figur 2.13: Klassesdiagram for UC2



Figur 2.14: Klassesdiagram for UC3



Figur 2.15: Klassediagram for UC4



Figur 2.16: Klassediagram for UC5

Accepttest 3

3.1 Accepttest af Use Cases

3.1.1 Use Case 1

Log ind

| Test | Forventet resultat | Faktiske observationer | Godkendt |
|--|---|------------------------|----------|
| <i>Hovedscenarie</i> | | | |
| 1. Indtast username samt password | Username- og passwordboks bliver udfyldt | | |
| 2. Tryk på "Login-knappen." | Login bliver godkendt. Login-vinduet lukkes ned mens CPR-vinduet åbnes | | |
| <i>Exentions</i> | | | |
| 2a. Username eller password er forkert | Besked vises på skærmen med tekst, der informerer om, at brugernavn eller password er forkert | | |

Tabel 3.1: Accepttest af Use Case 1.

3.1.2 Use Case 2

Vis EKG

| Test | Forventet resultat | Faktiske observationer | Godkendt |
|--|-------------------------------|------------------------|----------|
| <i>Hovedscenarie</i> | | | |
| 1. Indtast virtuel patients CPR-nummer | CPR-nummerboks bliver udfyldt | | |

| | | |
|-----|--|--|
| 2. | Vælg indstillinger | Indstillinger bliver valgt |
| 3. | Tryk på "Start ny måling" | Målingen startes i EKG-vinduet |
| 4. | EKG-data illustreres på en graf | En analyserebar graf fremvises i EKG-vinduet |
| 1.a | CPR-nummeret findes ikke. Besked vises med tekst, der informerer om, at CPR-nummeret ikke er gyldigt | Nyt CPR-nummer indtastes |
| 2.a | Ingen ændring i indstillinger | Målingen foretages med default-indstillingerne |

Tabel 3.2: Accepttest af Use Case 2.

3.1.3 Use Case 3

Evaluer EKG

| Test | Forventet resultat | Faktiske observationer | Godkendt |
|----------------------|---|--|----------|
| <i>Hovedscenarie</i> | | | |
| 1. | Validere program-mets analyse af EKG-signalet | Det er muligt at se små fluktuationer, som kan aflæses på EKG-grafen | |
| 2. | Stil diagnosen atrie-flimmer | Atrieflimmer kan aflæses ud fra EKG-grafen | |
| <i>Exentions</i> | | | |
| 2a. | Atriefrekvensen er ikke i intervallet 220-300 pr. minut | Det er ikke muligt at diagnosticere atrieflimmer ud fra EKG-grafen | |

Tabel 3.3: Accepttest af Use Case 3.

3.1.4 Use Case 4

Gem EKG

| Test | Forventet resultat | Faktiske observationer | Godkendt |
|-------------------------------|---|------------------------|----------|
| <i>Hovedscenarie</i> | | | |
| 1. Tryk på "Gem-knap- pen. | Messagebox kommer frem med besked om at målingen er gemt | | |
| 2. Tryk på "Ok-knappen | Målingen er gemt, vin- duet lukkes og EKG- vinduet vises igen | | |
| <i>Exentions</i> | | | |
| | | | |

Tabel 3.4: Accepttest af Use Case 4.

3.1.5 Use Case 5

Log ud

| Test | Forventet resultat | Faktiske observationer | Godkendt |
|---------------------------------|--|------------------------|----------|
| <i>Hovedscenarie</i> | | | |
| 1. Tryk på "log ud-knap- pen | EKG-vinduet lukkes ned, mens login- vinduet fremkommer | | |
| <i>Exentions</i> | | | |
| | | | |

Tabel 3.5: Accepttest af Use Case 5.

3.2 Accepttest af ikke-funktionelle krav

| Ikke-funktionelt krav | Test/handling | Forventet resultat | Faktiske observationer | Godkendt |
|---|--|---|------------------------|----------|
| <i>Usability</i> | | | | |
| Brugeren skal kunne starte en default-måling maksimalt 20 sekunder efter opstart af program | Start programmet, hvorefter der vha. stopur måles opstartstiden | At programmet er startet op indenfor 20 sekunder | | |
| Brugeren skal have mulighed for at ændre tidsintervallet før målingerne foretages | Start programmet og ændrer indstillingerne i toolbar | At der er mulighed for at ændre indstillinger | | |
| Login-vinduet skal indeholde en login-knap til at logge på og få vist EKG-vinduet | login-knappen er synlig i GUI, og ved tryk på knappen vises EKG-vinduet | At EKG-vinduet vises | | |
| EKG-vinduet skal indeholde en "start-knap" til at igangsætte målingerne | Startknappen er synlig i GUI, og ved tryk på knappen igangsættes måling | At målingen igangsættes | | |
| EKG-vinduet skal indeholde en "stop-knap" til at afslutte målingerne | Stopknappen er synlig i GUI, og ved tryk på knappen afsluttes måling | At målingen afsluttes | | |
| EKG-vinduet skal indeholde en "gem-knap" til at gemme målingerne | Gem-knappen er synlig i GUI, og ved tryk på knappen gemmes måling i database | Messageboks vises på skærmen med teksten "Måling er gemt" og kan findes i databasen | | |

| | | |
|---|--|--|
| EKG-vinduet skal indeholde en "log ud-knap til at logge ud | "log ud-knappen er synlig i GUI, og ved tryk på knap lukkes EKG-vinduet og login-vinduet vises | Login-vinduet vises |
| Målingen stopper automatisk efter det valgte tidsinterval | Der vælges et tidsinterval. Måling startes | Målingen stopper efter det valgte tidsinterval |
| <i>Reliability</i> | | |
| Systemet skal have en effektiv MTBF (Mean Time Between Failure) på 20 minutter og MT-TR (Mean Time To Restore) på 1 minut | Køre programmet i 20 minutter. Genstart derefter programmet, hvor der tages tid med et stopur | Programmet har kørt i 20 minutter og genstartes indenfor 1 minut |
| <i>Performance</i> | | |
| Der skal vises en EKG-graf i interfacet, hvor spænding vises op ad y-aksen (-1V til 1V) og tiden på x-aksen | Gennemfør en måling | At spændingen for EKG-signalet er op ad y-aksen, samt tiden hen ad x-aksen |
| Det skal være muligt at kunne scrolle igenem målingerne hen ad x-aksen | Der gennemføres en måling hvorefter der scrolles hen ad x-aksen | At der ved scrolling kan ses forskellige dele af EKG-signalet hen ad x-aksen |

| | | |
|--|---|--|
| Der skal kunne tages et sample over et bruger- bestemt interval, hvor frekvensen er tilpasses må- lingerne, således at grafen er ana- lyserbar | Gennemfør en måling, hvor et bestemt interval hen ad x-aksen er valgt | At det valgte in- terval synliggø- res |
|--|---|--|

Supportability

| | | |
|---|---|---|
| Software er op- bygget af tre- lagsmodellen | Kig i koden efter data-lag, logik-lag og GUI-lag | At koden inde- holder et data- lag, et logik-lag og et GUI-lag |
|---|---|---|

Tabel 3.6: Accepttest af Ikke-funktionelle krav