

Kravspekifikation

RoboPlay

Semesterprojektgruppe: 10

Afleveringsdato: 19. december 2018

Version: 2.1.0

#	Stud.nr.	Navn	Initialer
1	201704441	Frank Andersen	FA
2	201705186	Kristian Bang Nielsen	KN
3	201509800	Christian Lundtoft Trebbien	CT
4	201707807	Frederik Munch-Hansen	FM
5	06923	Niels Pallisgaard Thøgersen	NT
6	201505470	Mads Skytte Nielsen	MN
7	201704714	Michael Møller-Hansen	MM
Vejleder: Martin Ansbjerg Kjær			

Tabel 1 - Medlemmer i gruppe 10

1 Indholdsfortegnelse

1 Indholdsfortegnelse	1
2 Versionshistorik	3
3 Ordbog	3
4 Indledning	6
5 System beskrivelse	6
6 Definition af robottens bevægelser	10
7 Funktionelle krav	12
7.1 Aktør-kontekst diagram	12
7.2 Use case diagram	13
7.3 Brief use case beskrivelser	14
7.3.1 UC1: Udfør bevægelse	14
7.3.2 UC2: Tænd robot	15
7.3.3 UC3: Lav bevægelsesmønster	15
7.3.4 UC4: Start/stop bevægelsesmønster	15
7.3.5 UC5: Rediger bevægelsesmønster	15
7.3.6 UC6: Sæt robot i startposition	15
7.3.7 UC7: Grib om objekt	15
7.4 Afgrænsning af funktionelle krav	16
7.5 Fully dressed use case beskrivelser	16
7.5.1 UC1: Udfør bevægelse	16
7.5.2 UC2: Tænd robot	18
7.5.3 UC3: Lav bevægelsesmønster	18
7.5.4 UC4: Start/stop bevægelsesmønster	18
7.5.5 UC5: Rediger bevægelsesmønster	19
7.5.6 UC6: Sæt robot i startposition	19

7.5.7 UC7: Grib om objekt	19
8 Ikke-funktionelle krav	20
8.1 Smartphone applikation	21
8.2 Systemets ydeevne	21
8.3 Bevægelsernes hastigheder	22
8.4 Bevægelsernes præcision	23
8.5 Robottens løfteevne	23
8.6 Sikkerhed	23
8.7 Startposition	24
8.8 Rækkevidde	24

2 Versionshistorik

Vers	Dato (Y-M-D)	Navn	Ændringer
v0.1.0	2018-09-17	NT	Dokument oprettet
v0.2.0	2018-09-27	MM	Udkast til hele afsnittet omhandlende funktionelle krav lavet med brief use case beskrivelser og tilhørende diagrammer og prioritering. Opstart på ikke-funktionelle krav uden det dog på nogen måde er færdigt.
v0.3.0	2018-09-30	MM	Nogle ikke-funktionelle krav tilføjet.
v0.4.0	2018-10-03	MM & FA	Sekundær aktør "Objekt" tilføjet på aktør-kontekst diagram.
v0.5.0	2018-10-04	MM	UC7 tilføjet. Diagrammer opdateret og beskrivelser af den nye UC lavet. Indledning + systembeskrivelse omskrevet.
v0.6.0	2018-10-08	Alle -MN	Gennemlæst og rettet i fællesskab
v1.0.0	2018-10-08	Alle	Sidste ændringer - klar til review #1
v1.1.0	2018-10-14	MM, FA	Ændringer som følge af review foretaget.
v1.1.1	2018-10-16	NT	Ordbog tilføjet
v1.1.3	2018-10-26	MM	MoSCoW
v1.1.4	2018-10-28	NT	Ordbog opdateret. Smårettelser.
v2.0.0	2018-10-29	MM	Rettelser som følge af gruppemøde den 29/10/18 foretaget.
v2.0.1	2018-11-06	MM	Prækondition tilføjet til UC2's fully dressed beskrivelse.
v2.0.2	2018-11-19	FA	Billeder af robotten tilføjet
v2.1.0	2018-11-26	MM	Denne version er sendt til reviewgruppe i forbindelse med review #2.

Tabel 2 - Versionshistorik

3 Ordbog

I *tabel 3* herunder ses en oversigt over de begreber, der er benyttet i dette projekt samt forklaring af deres betydning. Denne ordbog kan benyttes som et opslagsværk under læsningen af dette dokument.

Ord	Betydning
RoboPlay	Det system, der udvikles, inklusive hard- og software
Robot	Den fysiske robotarm, inklusive motorer, sensore, mfl.
RPi	Raspberry Pi Zero W med Linux installeret
RPiApp	Software, der udvikles til RPi, inklusive eventuelle kernemoduler
PSoC	Cypress PSoC 5LP development board
PSoCApp	Software, der udvikles til PSoC
Mobiltelefon	Smartphone med Android operativsystem
MobilApp	Software, der udvikles til Mobiltelefon
Invers kinematisk bevægelse	En bevægelse mod et fastlagt punkt, hvor det skal beregnes, hvordan man kommer derhen.
Xamarin	En cross-platform teknologi udviklet af Microsoft. Xamarin giver mulighed for en høj grad af kodegenbrug mellem forskellige platforme såsom iOS og Android.
MVVM	Forkortelse for "Model-View-ViewModel. Den foretrukne arkitektur når Xamarin benyttes.
Yderpunkt	Det yderste punkt en given akse kan komme i.
Kontakt	Kontakten som tænder og slukker robotten
Servomotor	Servomotor af typen MG996R. Begreberne <i>motor</i> og <i>servomotor</i> ses som synonyme.
Hjul	<i>Encoder disc</i> til optocoupler.
Mellemrum	Hul i hjul, der tillader lys at passere igennem.
CRC	Cyclic Redundancy Check. En metode til at detektere fejl i data-transmission.
Motor	Servomotor af typen MG996R. Begreberne <i>motor</i> og <i>servomotor</i> ses som synonyme.



Tabel 3 - Ordbog

4 Indledning

I dette dokument ses den fulde kravspecifikation for RoboPlay. Kravspecifikationen er en viderebygning af projektformuleringen, og den danner fundamentet for det videre projektarbejde, hvor en prototype for RoboPlay udvikles.

Først beskrives systemets overordnede funktionalitet og en helt overordnet MoSCoW analyse sætter rammen for, hvilke dele projektgruppen prioriterer højest.

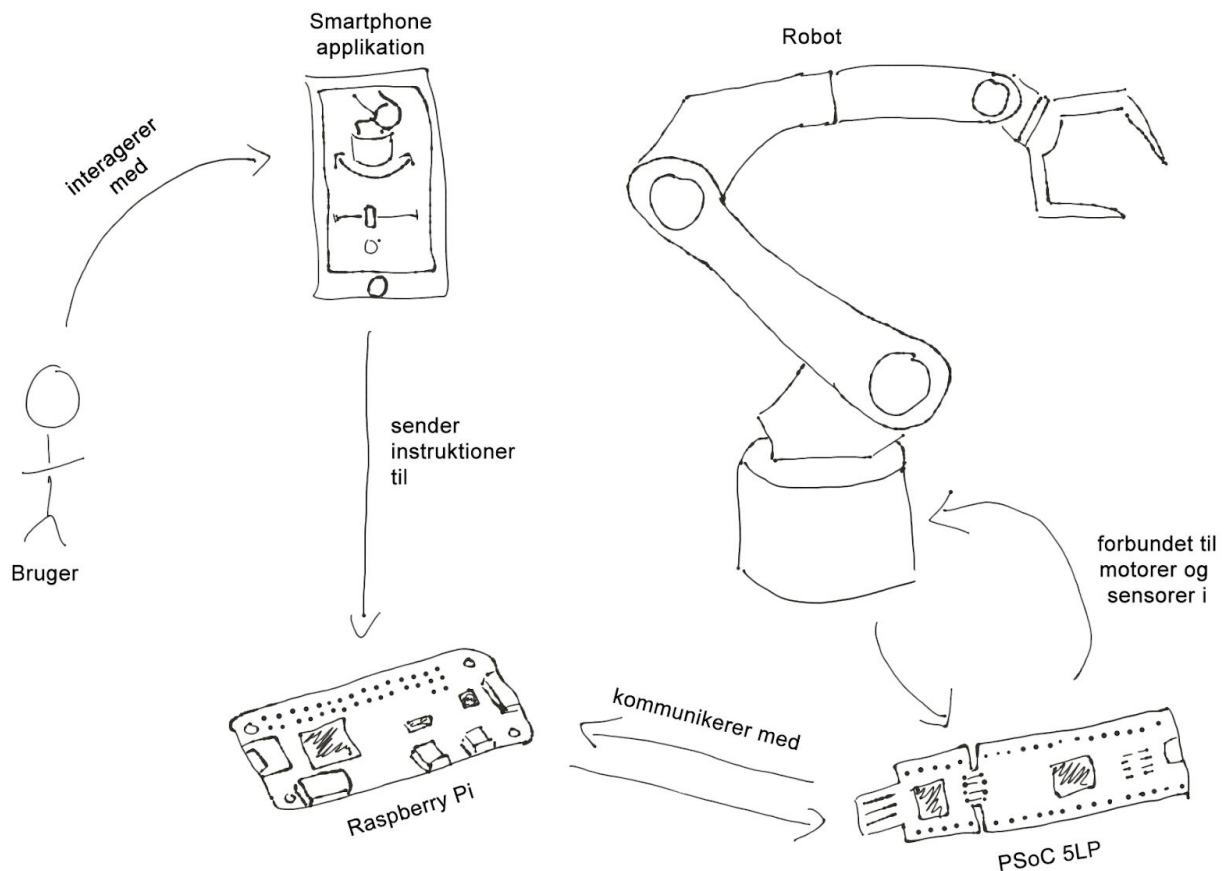
Dernæst specificeres kravene til systemet mere eksplicit ved brug af use cases. De fundne use cases prioriteres med MoSCoW og den enkelte use case beskrives yderligere med "brief" og/eller "fully dressed" use case beskrivelser.

Til sidst specificeres de ikke-funktionelle krav til systemet, som ligeledes prioriteres med MoSCoW.

5 System beskrivelse

RoboPlay er et stykke legetøj, der udvikles til børn i alderen 12+ år. Formålet med RoboPlay er at nedbryde barrieren mellem barn og teknologi, hvorved barnet motiveres og opmuntres til at lære gennem leg.

RoboPlay er i sin enkelthed en robot, der styres via en applikation, som er installeret på brugerens mobiltelefon. Mobiltelefonen kommunikerer trådløst med en Raspberry Pi, som er hjernen i systemet. Her foretages de nødvendige beregninger, hvorefter en PSoC benyttes til at omsætte beregningerne til bevægelser i robotten. Et rigt billede, der illustrerer ovenstående, ses på *figur 1*.

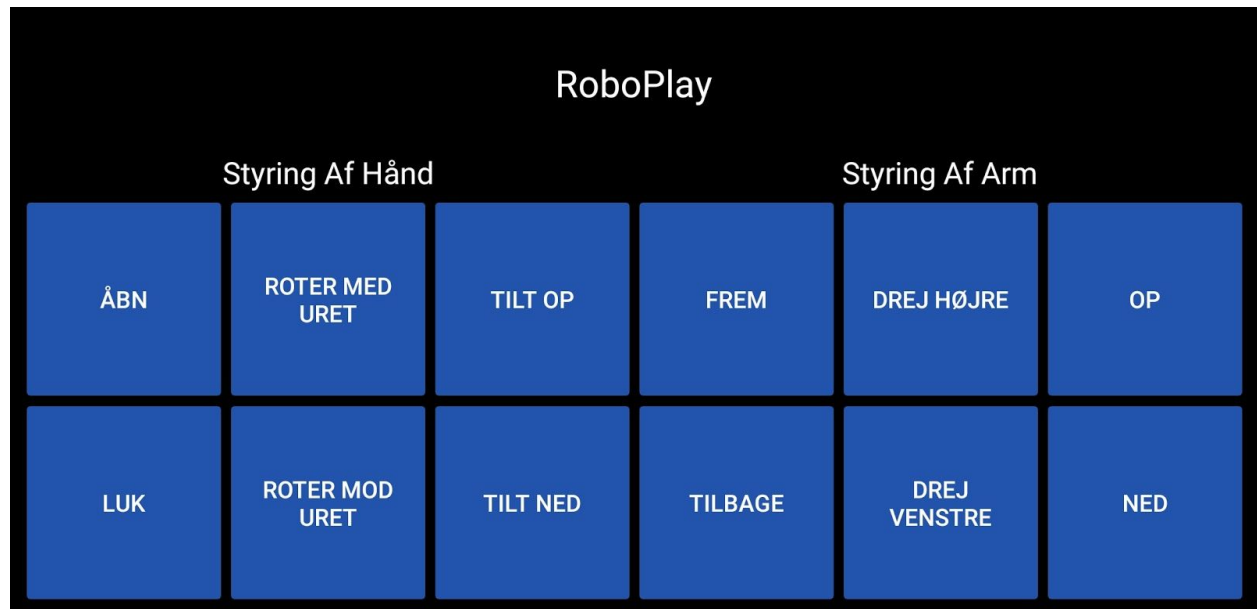


Figur 1 - Rigtigt billede for RoboPlay der viser kommunikationen mellem brugeren og de forskellige delsystemer.

En række sensorer, som er tilsluttet PSoC'en, sikrer at robottens bevægelser begrænses således, at robotten ikke kan gå i stykker som følge af brugerens input. På samme vis sikres det, at robotten forhindres i at udføre bevægelser, der kan forvolde nogen form for personskaade. Sensorer benyttes også til at indstille robottens led til foruddefinerede startpositioner, når robotten startes op.

Input fra sensorer kommunikerer tilbage til Raspberry Pi'en, hvorfor der på det rige billede også ses, at der er to-vejskommunikation mellem Raspberry Pi'en og PSoC'en.

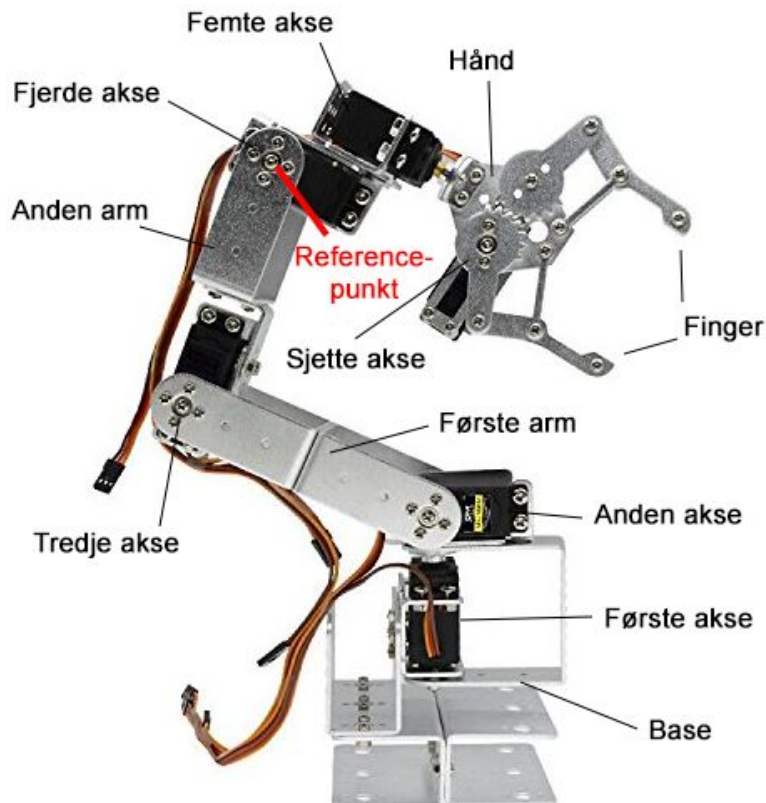
På *figur 2* ses en skitse af mobilapplikationens design.



Figur 2 - Mobilapplikationens design hvor det ses hvilke bevægelser brugeren kan bevæge robotten. Mens en knap holdes nede udføres bevægelsen ellers er bevægelsen stoppet.

Brugeren kan altså bevæge robotten på i alt 12 forskellige måder, og disse er defineret nærmere i afsnit [6 Definition af robottens bevægelser](#).

I dette dokument vil blive henvist til robottens akser, arme, referencepunkt mv., som er defineret ved illustrationen vist på *figur 3* nedenfor.



Figur 3 - Oversigt over robottens akser, arme og øvrige elementer. Referencepunktet er angivet som centrum af midterste skrue for fjerde akse.

Med baggrund i ovenstående kan en helt overordnet MoSCoW-analyse for RoboPlay laves. Denne analyses formål er alene at give læseren et overblik over hvilke dele af projektet, der prioriteres højest, inden kravene uddybes i de kommende afsnit. Idet der netop er tale om udviklingen af en prototype forventes kun krav prioriteret med “must” og “should” implementeret. Resultatet ses i *tabel 4*.

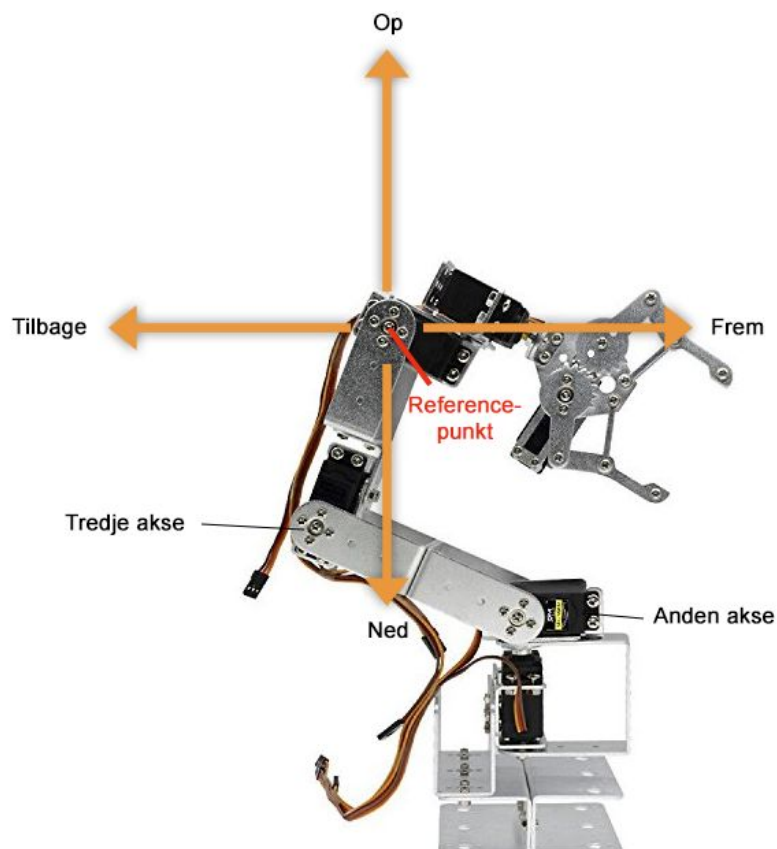
Prioritet	Beskrivelse
Must	<ol style="list-style-type: none"> 1. Mindst en af robottens bevægelser skal gøre brug af invers kinematik 2. Robotten skal kunne fjernstyres via en grafisk brugergrænseflade 3. Robotten skal som minimum bestå af en PSoC, en Raspberry Pi, en sensor og en aktuator 4. Systemet kan kommunikere imellem PSoC, en Raspberry Pi og en mobil applikation 5. Robotarmen skal kunne gribe om og løfte en genstand
Should	<ol style="list-style-type: none"> 1. Robotten bør have en anordning til begrænsning af mekanisk kraft

	2. Robotten bør kunne finde tilbage til en foruddefineret startposition
Could	1. Robotten kan indstilles til at udføre sekvenser af bevægelser via en grafisk brugergrænseflade
Won't	1. Robotten vil ikke have en anordning til begrænsning af mekanisk kraft på alle akser.

Tabel 4 - Overordnet MoSCoW analyse.

6 Definition af robottens bevægelser

Robottens bevægelser er kort beskrevet i det foregående afsnit, men her defineres de enkelte bevægelser entydigt. Visse bevægelser er kinematiske og for disses vedkommende tages der udgangspunkt i *figur 4*. Beskrivelsen af de forskellige bevægelser ses i *tabel 5*.



Figur 4 - Oversigt over robottens kinematiske bevægelsesretninger "Frem", "Tilbage", "Op" og "Ned"

Navn	Beskrivelse
Åbn	Bevægelsen "Åbn" er defineret ved, at robottens sjette akse bevæges så afstanden mellem robottens fingre øges.
Luk	Bevægelsen "Luk" er defineret ved, at robottens sjette akse bevæges så afstanden mellem robottens fingre mindskes.
Roter med uret	Bevægelsen "Roter med uret" er defineret ved, at robottens femte akse bevæges modsat "Roter mod uret".
Roter mod uret	Bevægelsen "Roter mod uret" er defineret ved, at robottens femte akse bevæges modsat af bevægelsen "Roter med uret".
Tilt op	Bevægelsen "Tilt op" er defineret ved, at robottens fjerde akse bevæges modsat af bevægelsen "Tilt ned".
Tilt ned	Bevægelsen "Tilt ned" er defineret ved, at robottens fjerde akse bevæges modsat af bevægelsen "Tilt op".
Frem	Bevægelsen "Frem" er defineret ved, at robottens anden og tredje akse bevæges samtidigt således, at robottens referencepunkt forskydes som vist på figur 4.
Tilbage	Bevægelsen "Tilbage" er defineret ved, at robottens anden og tredje akse bevæges samtidigt således, at robottens referencepunkt forskydes som vist på figur 4.
Drej højre	Bevægelsen "Drej højre" er defineret ved, at robottens første akse bevæges modsat af bevægelsen "Drej venstre".
Drej venstre	Bevægelsen "Drej venstre" er defineret ved, at robottens første akse bevæges modsat af bevægelsen "Drej højre".
Op	Bevægelsen "Op" er defineret ved, at robottens anden og tredje akse bevæges samtidigt således, at robottens referencepunkt forskydes som vist på figur 4
Ned	Bevægelsen "Ned" er defineret ved, at robottens anden og tredje akse bevæges samtidigt således, at robottens referencepunkt forskydes som vist på figur 4.

Tabel 5 - De ikke-funktionelle krav til robottens bevægelser.

7 Funktionelle krav

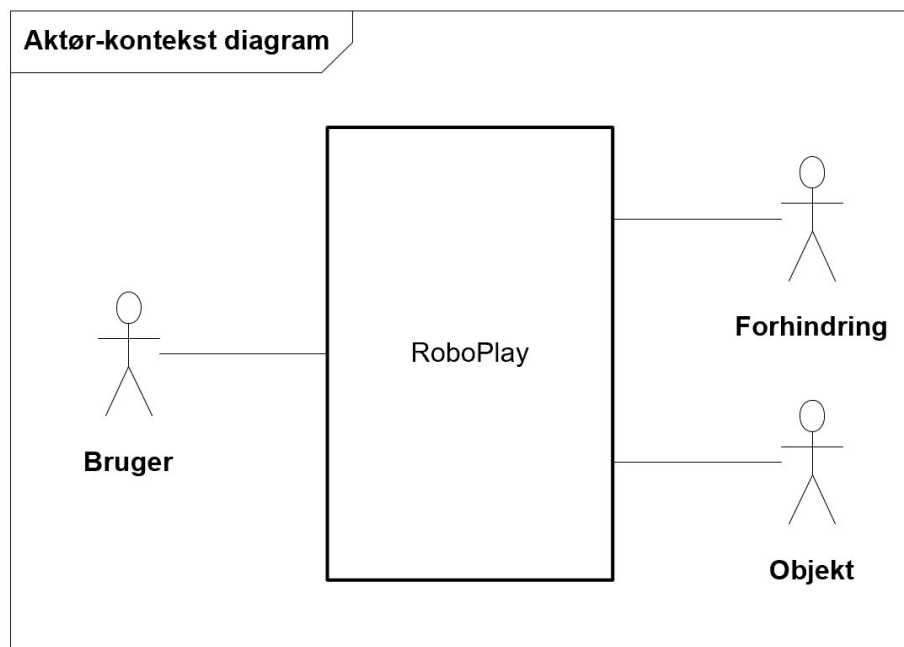
I dette afsnit beskrives først de aktører, der interagerer med RoboPlay. De funktionelle krav, som beskrives med use cases, vises derpå i et use case diagram og beskrives efterfølgende tekstuel i form af "brief" use case beskrivelser.

Deraf følger en prioritering af de fundne krav under brug af MoSCoW mhp. at afgrænse projektets omfang.

De use cases, der er prioriteret højest i MoSCoW-analysen, beskrives afslutningsvist i detaljer med "fully dressed" use case beskrivelser.

7.1 Aktør-kontekst diagram

På *figur 5* ses et aktør-kontekst diagram for RoboPlay. Selve robotten, dens sensorer samt mobiltelefonen, som brugeren benytter, betragtes altså som værende en del af systemet.



Figur 5 - Aktør-kontekst diagram hvoraf det ses, at brugeren er den eneste aktør.

I tabellerne nedenfor ses aktørbeskrivelserne for aktørerne *Bruger*, *Forhindring* og *Objekt*.

Aktør navn	Bruger
-------------------	--------



Type	Primær
Beskrivelse	Brugeren er den person der styrer robotarmen via RoboPlay's smartphoneapplikation, som er installeret på brugerens mobiltelefon.

Tabel 6 - Aktørbeskrivelse for aktøren "Bruger"

Aktør navn	Forhindring
Type	Sekundær
Beskrivelse	En forhindring er en genstand, som står i vejen for udførelsen af en eller flere af robotens bevægelser.

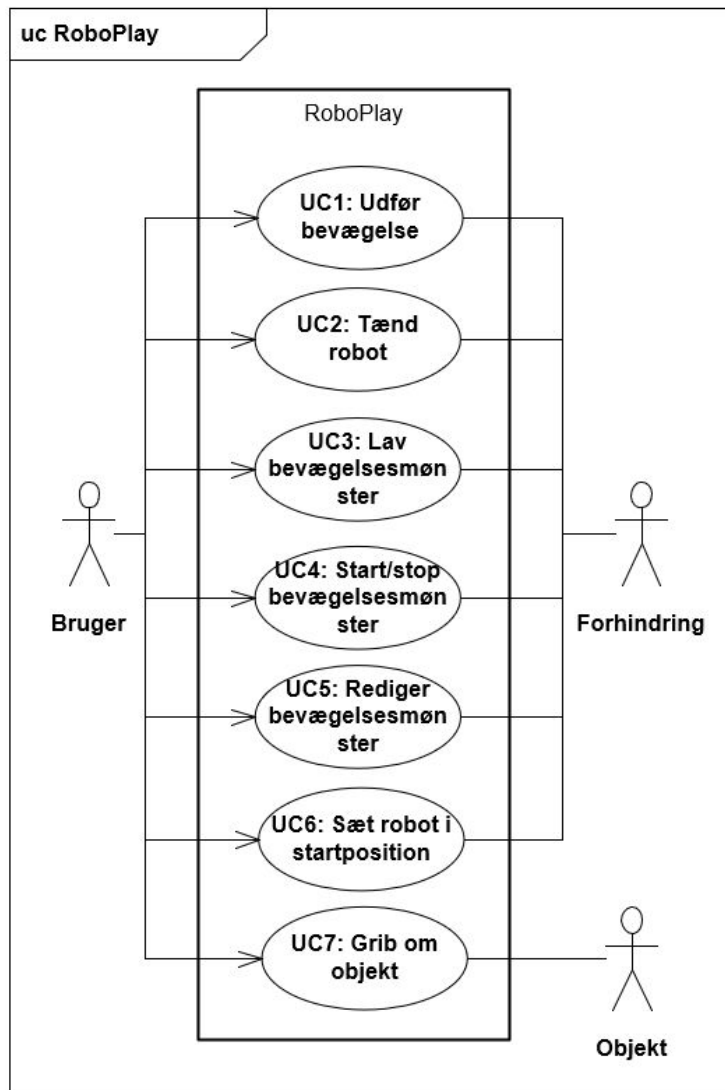
Tabel 7 - Aktørbeskrivelse for aktøren "Forhindring"

Aktør navn	Objekt
Type	Sekundær
Beskrivelse	Et objekt er en genstand, som roboten kan gribe om med dens hånd.

Tabel 8 - Aktørbeskrivelse for aktøren "Objekt"

7.2 Use case diagram

På figur 6 ses samtlige identificerede use cases i et use case diagram.



Figur 6 - Use case diagram som viser samtlige identificerede use cases.

7.3 Brief use case beskrivelser

I dette afsnit beskrives hver enkelt use case med "brief" use case beskrivelser.

7.3.1 UC1: Udfør bevægelse

UC1 omhandler den brugssituation, hvor brugeren benytter RoboPlay's smartphone- applikation til at styre robotten manuelt i en eller flere bevægelser. Use casen indeholder altså samtlige bevægelser, som robotten understøtter - dog undtaget bevægelsen "Luk", som behandles i

UC7. Eksempler på sådanne bevægelser kunne være at bevæge robotarmen fremad eller åbne robothånden.

7.3.2 UC2: Tænd robot

UC2 omhandler den brugssituation, hvor brugeren tænder robotten. Under opstarten indstilles robottens led automatisk til deres startpositioner.

7.3.3 UC3: Lav bevægelsesmønster

UC3 omhandler den brugssituation, hvor brugeren benytter RoboPlay's smartphoneapplikation til at kombinere og gemme en eller flere bevægelser i et såkaldt bevægelsesmønster, der kan eksekveres på et senere tidspunkt.

7.3.4 UC4: Start/stop bevægelsesmønster

UC4 omhandler den brugssituation, hvor brugeren benytter RoboPlay's smartphoneapplikation til at få robotten til at eksekvere et gemt bevægelsesmønster. Use casen omhandler også situationen, hvor et påbegyndt bevægelsesmønster afbrydes før tid.

7.3.5 UC5: Rediger bevægelsesmønster

UC5 omhandler den brugssituation, hvor brugeren benytter RoboPlay's smartphoneapplikation til at redigere eller slette et eksisterende bevægelsesmønster.

7.3.6 UC6: Sæt robot i startposition

UC6 omhandler den brugssituation, hvor brugeren manuelt benytter RoboPlay's smartphoneapplikation til at indstille robottens led til de startpositioner, som robotten også indstilles til under udførelse af UC2.

7.3.7 UC7: Grib om objekt

UC7 omhandler den brugssituation, hvor brugeren benytter RoboPlay's smartphoneapplikation til at lukke robottens hånd mhp. at gribe om et objekt. Dette kunne fx være i forbindelse med, at brugeren ønsker at benytte robotten til at samle et objekt op.

7.4 Afgrænsning af funktionelle krav

For at begrænse arbejdet med funktionelle krav, som potentielt ikke når at blive implementeret, afgrænses kravene under brug af MoSCoW. Resultatet ses i *tabel 9*.

Kun de krav, der har prioriteten “must have” og “should have”, uddybes indledningsvist med fyldestgørende “fully dressed” use case beskrivelser. Accepttests specificeres endvidere kun til krav, der er lavet “fully dressed” use case beskrivelser af.

Såfremt de lavere prioriterede krav bliver mulige at implementere i løbet af projektperioden, udarbejdes “fully dressed” use case beskrivelser og tilhørende accepttests løbende.

ID	Navn	Prioritet
UC1	Udfør bevægelse	M
UC2	Tænd robot	S
UC3	Lav bevægelsesmønster	C
UC4	Start/stop bevægelsesmønster	C
UC5	Rediger bevægelsesmønster	W
UC6	Indstil robot til startposition	S
UC7	Grib om objekt	M

Tabel 9 - Prioritering af de funktionelle krav under brug af MoSCoW

7.5 Fully dressed use case beskrivelser

I dette afsnit uddybes beskrivelsen af de højest prioriterede krav med “fully dressed” use case beskrivelser.

7.5.1 UC1: Udfør bevægelse

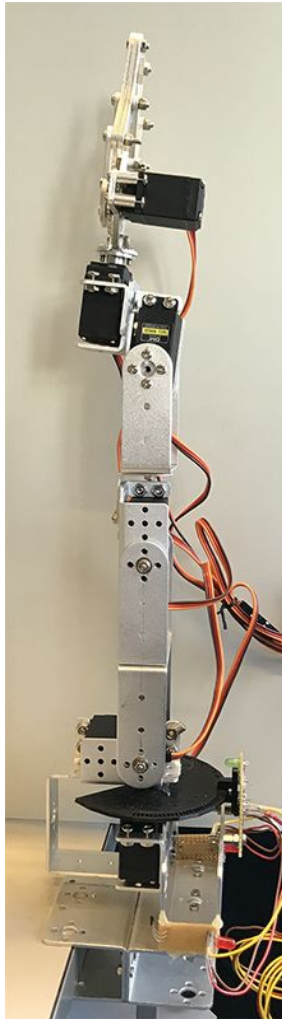
Navn	UC1: Udfør bevægelse
Mål	Brugeren starter en bevægelse som stoppes igen
Initiering	Bruger

Aktører	Primær: Bruger Sekundær: Forhindring
Referencer	Ingen
Antal samtidige forekomster	4 såfremt UC7 ikke er under afvikling. Er UC7 under afvikling så kan der være 3 samtidige forekomster af UC1. Visse bevægelser ophæver hinanden - f.eks. bevægelserne "Frem" og "Tilbage", som vil resultere i, at ingen af bevægelserne udføres.
Prækondition	Systemet er operationelt
Postkondition	Den ønskede bevægelse er udført
Hovedscenarie	<ol style="list-style-type: none"> 1. Brugeren holder en knap på brugergrænsefladen nede for at starte en bevægelse (se data variationsliste). 2. Systemet starter bevægelsen, som vedrører knappen der er trykket på i punkt 1. 3. Brugeren slipper knappen, som blev holdt nede i punkt 1. [Ext 1: Yderpunkt nået] [Ext 2: Forhindring detekteret] 4. Systemet stopper bevægelsen, som blev startet i punkt 1.
Udvidelser eller undtagelser	<p>[Ext 1: Yderpunkt nået]</p> <ol style="list-style-type: none"> 1. Systemet detekterer at bevægelsen, som blev startet i hovedscenariets punkt 1, har nået sit yderpunkt. 2. Systemet stopper bevægelsen, som blev startet i hovedscenariets punkt 1. 3. UC afsluttes. <p>[Ext 2: Forhindring detekteret]</p> <ol style="list-style-type: none"> 1. Systemet detekterer en forhindring og standser alle igangværende bevægelser. 2. UC afsluttes.
Data variationsliste	<p>Bevægelse</p> <ul style="list-style-type: none"> • Mulige bevægelser er de bevægelser, der er beskrevet i afsnit "6 Definition af robotens bevægelser" med undtagelsen af bevægelsen "Luk", som er håndteret i UC7.

Tabel 10 - "Fully dressed" use case beskrivelse for UC1: Udfør bevægelse

7.5.2 UC2: Tænd robot

En prækondition til denne use case er, at robotten skal placeres som vist på *figur 7* inden use casen udføres.



Figur 7 - Robottens led skal placeres som vist på dette billede før use casen udføres.

Navn	UC2: Tænd robot
Mål	At starte systemet op og klargøre det til brug
Initiering	Bruger
Aktører	Primær: Bruger
Referencer	UC6: Sæt robot i startposition

Antal samtidige forekomster	1
Prækondition	Robotter er slukket men tilsluttet strøm. Robottens led er placeret som vist på <i>figur 7</i> .
Postkondition	Robotten står i startposition og er klar til brug
Hovedscenarie	<ol style="list-style-type: none"> 1. Brugeren tænder for systemet ved at påvirke systemets kontakt. 2. Gå til UC6 punkt 2.
Udvidelser eller undtagelser	
Data variationsliste	

Tabel 11 - "Fully dressed" use case beskrivelse for UC2: Tænd robot

7.5.3 UC3: Lav bevægelsesmønster

Fully dressed use case beskrivelse er undladt for denne use case indtil implementering af den er realistisk. Se i stedet den korte beskrivelse af use casen i tidligere afsnit.

7.5.4 UC4: Start/stop bevægelsesmønster

Fully dressed use case beskrivelse er undladt for denne use case indtil implementering af den er realistisk. Se i stedet den korte beskrivelse af use casen i tidligere afsnit.

7.5.5 UC5: Rediger bevægelsesmønster

Fully dressed use case beskrivelse er undladt for denne use case indtil implementering af den er realistisk. Se i stedet den korte beskrivelse af use casen i tidligere afsnit.

7.5.6 UC6: Sæt robot i startposition

Navn	UC6: Sæt robot i startposition
Mål	At indstille robottens akser til deres startpositioner
Initiering	Bruger
Aktører	Primær: Bruger Sekundær: Forhindring
Referencer	Ingen

Antal samtidige forekomster	1
Prækondition	Robotten er tilsluttet strøm og tændt
Postkondition	Alle robotens akser er indstillet til deres startpositioner
Hovedscenarie	<ol style="list-style-type: none"> 1. Brugeren trykker på brugergrænsefladens knap benævnt "Sæt robot i startposition". 2. Systemet indstiller alle robotens akser til deres startpositioner. [Ext 1: Robotten møder en forhindring]
Udvidelser eller undtagelser	[Ext 1: Robotten møder en forhindring] <ol style="list-style-type: none"> 1. Systemet stopper indstilling af akserne. 2. UC afsluttes.
Data variationsliste	

Tabel 12 - "Fully dressed" use case beskrivelse for UC6: Sæt robot i startposition.

7.5.7 UC7: Grib om objekt

Navn	UC7: Grib om objekt
Mål	Robotens hånd griber fat om et objekt
Initiering	Bruger
Aktører	Primær: Bruger Sekundær: Objekt
Referencer	Ingen
Antal samtidige forekomster	1
Prækondition	Systemet er operationelt. Brugeren har indstillet roboten således, at et objekt er placeret mellem robothåndens fingre.
Postkondition	Robotens hånd har grebet fast om det ønskede objekt.
Hovedscenarie	<ol style="list-style-type: none"> 1. Brugeren holder brugergrænsefladens knap benævnt "Luk" nede. 2. Systemet iværksætter bevægelsen "Luk". 3. Systemet detekterer at hånden har grebet fast om objektet. [Ext 1: Bruger stopper bevægelsen]

	4. Systemet stopper bevægelsen "Luk".
Udvidelser eller undtagelser	[Ext 1: Bruger stopper bevægelsen] 1. Brugeren slipper brugergrænsefladens knap benævnt "Luk". 2. Systemet stopper bevægelsen "Luk". 3. UC afsluttes.
Data variationsliste	

Tabel 13 - "Fully dressed" use case beskrivelse for UC7: Grib om objekt

8 Ikke-funktionelle krav

I de følgende underafsnit er de ikke-funktionelle krav grupperet og opskrevet på kravlister, som er prioriterede under brug af MoSCoW.

I de definerede ikke-funktionelle krav gælder følgende usikkerheds-intervaller medmindre andet særskilt angivet:

- Afstand: +/- 0,1 cm
- Tid: +/- 0,5 sekunder
- Vinkel: +/- 3 grader
- Vægt: +/- 1 gram

8.1 Smartphone applikation

ID	Navn	Prioritet
K1.1	Applikationen skal kunne eksekveres på mobiltelefoner, der kører styresystemet Android Oreo.	M
K1.2	Al tekst der vises for brugeren i applikationen skal være på sproget dansk.	M
K1.3	Ved opstart af applikationen skal der skabes trådløs forbindelse til robotten således denne kan styres uden yderligere opsætning.	M
K1.4	Applikationen skal have to skærm billeder benævnt "Manuel styring" og "Indstillinger".	M

K1.5	<p>Applikationen skal have 12 knapper på skærbilledet "Manuel styring". Knapperne benævnes:</p> <ul style="list-style-type: none"> • "Åbn" • "Luk" • "Roter med uret" • "Roter mod uret" • "Tilt op" • "Tilt ned" • "Frem" • "Tilbage" • "Drej højre" • "Drej venstre" • "Op" • "Ned" <p>Hver knap henfører til en bevægelse og disse er defineret i afsnit "6 Definition af robotens bevægelser".</p>	M
K1.6	<p>Applikationen skal have en knap på skærbilledet "Indstillinger" benævnt "Sæt robot til startposition", som indstiller robotens akser til deres startpositioner, som anført i det ikke-funktionelle krav K7.1</p>	M

Tabel 14 - De ikke-funktionelle krav til smartphone applikationen.

8.2 Systemets ydeevne

ID	Navn	Prioritet
K2.1	Fra en bevægelse initieres/stoppes på smartphone applikationen til roboten påbegynder udførelsen af kommandoen må der maksimalt gå 1 sekund.	M
K2.2	Systemet skal have en kontakt, som kan sættes i to tilstande. En hvor systemet tændes og en hvor systemet slukkes.	M
K2.3	Robotten skal være klar til brug maksimalt 60 sekunder efter systemet er tændt ved påvirkning af kontakten nævnt i K2.2.	S

Tabel 15 - De ikke-funktionelle krav til systemets ydeevne.

8.3 Bevægelsernes hastigheder

ID	Navn	Prioritet
----	------	-----------

K3.1	Første akse skal bevæge sig med en hastighed på minimum 20 grader pr. sekund og maksimum 30 grader pr. sekund.	S
K3.2	Fjerde akse skal bevæge sig med en hastighed på minimum 20 grader pr. sekund og maksimum 30 grader pr. sekund.	S
K3.3	Femte akse skal bevæge sig med en hastighed på minimum 20 grader pr. sekund og maksimum 30 grader pr. sekund.	S
K3.4	Sjette akse skal bevæge sig med en hastighed på minimum 0.5 cm/s og maksimum 3 cm/s.	S
K3.5	Bevægelsen "Frem" skal flytte robottens referencepunkt med minimum 0.5 cm/s og maksimum 3 cm/s.	S
K3.6	Bevægelsen "Tilbage" skal flytte robottens referencepunkt med minimum 0.5 cm/s og maksimum 3 cm/s.	S
K3.7	Bevægelsen "Op" skal flytte robottens referencepunkt med minimum 0.5 cm/s og maksimum 3 cm/s.	S
K3.8	Bevægelsen "Ned" skal flytte robottens referencepunkt med minimum 0.5 cm/s og maksimum 3 cm/s.	S

Tabel 16 - De ikke-funktionelle krav til de hastigheder, som robottens bevægelser kan udføres med.

8.4 Bevægelsernes præcision

ID	Navn	Prioritet
K4.1	Fra samme udgangsposition for robotten skal ens input til bevægelsen "Frem" resultere i samme position for robottens referencepunkt +/- 2 cm.	S
K4.2	Fra samme udgangsposition for robotten skal ens input til bevægelsen "Tilbage" resultere i samme position for robottens referencepunkt +/- 2 cm.	S
K4.3	Fra samme udgangsposition for robotten skal ens input til bevægelsen "Op" resultere i samme position for robottens referencepunkt +/- 2 cm.	S
K4.4	Fra samme udgangsposition for robotten skal ens input til bevægelsen "Ned" resultere i samme position for robottens referencepunkt +/- 2 cm.	S

Tabel 17 - De ikke-funktionelle krav til den præcision som robottens bevægelser kan udføres med.

8.5 Robottens løfteevne

ID	Navn	Prioritet
K5.1	Robothånden skal kunne gribe om objekter der måler minimum 2x2x2 cm.	M
K5.2	Robothånden skal kunne gribe om objekter der måler maksimalt 5x5x5 cm.	M
K5.3	Robotten skal kunne løfte objekter der vejer maksimum 50 g.	M

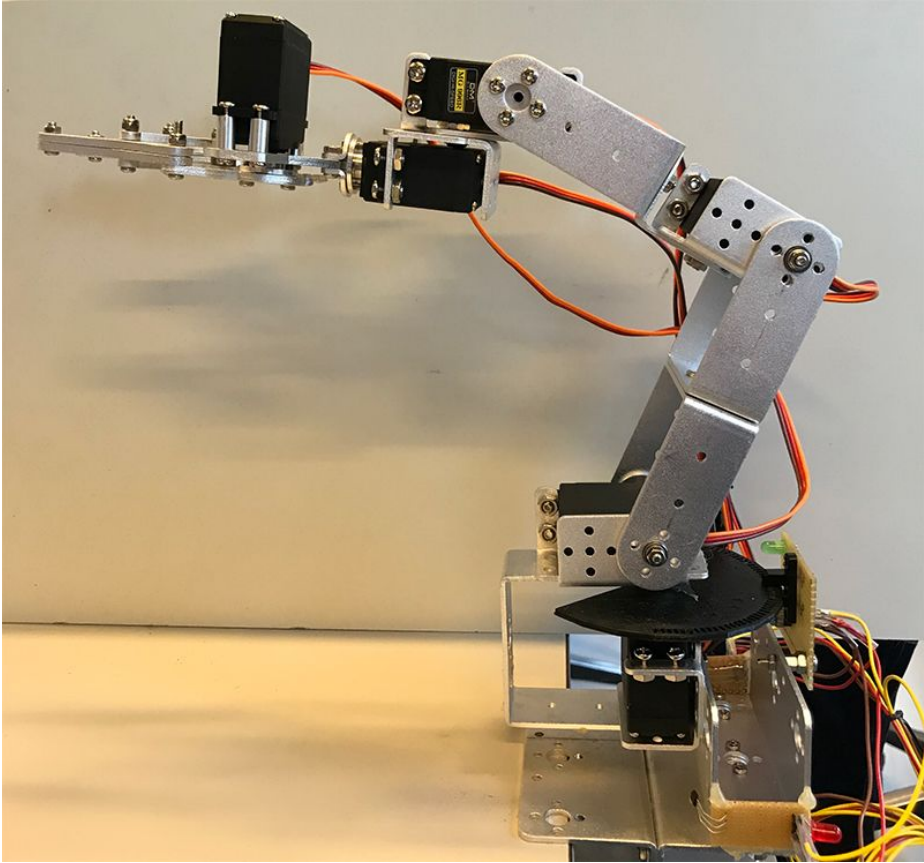
Tabel 18 - De ikke-funktionelle krav vedrørende robottens løfteevne.

8.6 Sikkerhed

ID	Navn	Prioritet
K6.1	Robotten skal standse alle aksers igangværende bevægelser hvis en forhindring påvirker robottens første akse med en kraft på 10 newton eller mere.	M
K6.2	Robotten skal standse alle aksers igangværende bevægelser hvis en forhindring påvirker robottens anden akse med en kraft på 10 newton eller mere.	C
K6.3	Robotten skal standse alle aksers igangværende bevægelser hvis en forhindring påvirker robottens tredje akse med en kraft på 10 newton eller mere.	C
K6.4	Robotten skal standse alle aksers igangværende bevægelser hvis en forhindring påvirker robottens fjerde akse med en kraft på 10 newton eller mere.	C
K6.5	Robotten skal standse alle aksers igangværende bevægelser hvis en forhindring påvirker robottens femte akse med en kraft på 10 newton eller mere.	C
K6.6	Robotten skal standse sjette akse igangværende bevægelse hvis robottens fingre presses mod hinanden eller mod et objekt med en kraft på 10 newton eller mere.	M

Tabel 19 - De ikke-funktionelle krav vedrørende sikkerheden under brug af robotten.

8.7 Startposition

ID	Navn	Prioritet
K7.1	<p>Robottens startposition skal være som vist på billedet nedenfor:</p> 	S

Tabel 20 - De ikke-funktionelle krav til de positioner robottens akser skal indstilles i for at robotten indtager sin "Startposition".

8.8 Rækkevidde

ID	Navn	Prioritet
K8.1	Robotten skal kunne gribe om og løfte et objekt, som befinder sig i en afstand på mindst 6 cm fra basens centrum.	S

K8.2	Robotten skal kunne gribe om og løfte et objekt, som befinder sig i en afstand på maksimalt 30 cm fra basens centrum.	S
------	---	---

Tabel 21 - De ikke-funktionelle krav til robottens rækkevidde.