

# Kravspekifikation

## RoboPlay

Semesterprojektgruppe: 10

Afleveringsdato: 19. december 2018

Version: 1.0.0

#	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

<b>1 Indholdsfortegnelse</b>	<b>1</b>
<b>2 Versionshistorik</b>	<b>3</b>
<b>3 Indledning</b>	<b>4</b>
<b>4 System beskrivelse</b>	<b>4</b>
<b>5 Funktionelle krav</b>	<b>8</b>
5.1 Aktør-kontekst diagram	8
5.2 Use case diagram	10
5.3 Brief use case beskrivelser	11
5.3.1 UC1: Start/stop bevægelse	11
5.3.2 UC2: Tænd robot	11
5.3.3 UC3: Lav bevægelsesmønster	11
5.3.4 UC4: Start/stop bevægelsesmønster	11
5.3.5 UC5: Rediger bevægelsesmønster	11
5.3.6 UC6: Sæt robot i startposition	12
5.3.7 UC7: Grib om objekt	12
5.4 Afgrænsning af funktionelle krav	12
5.5 Fully dressed use case beskrivelser	13
5.5.1 UC1: Start/stop bevægelse	13
5.5.2 UC2: Tænd robot	14
5.5.3 UC3: Lav bevægelsesmønster	14
5.5.4 UC4: Start/stop bevægelsesmønster	15
5.5.5 UC5: Rediger bevægelsesmønster	15
5.5.6 UC6: Sæt robot i startposition	15
5.5.7 UC7: Grib om objekt	16
<b>6 Ikke-funktionelle krav</b>	<b>17</b>

6.1 Smartphone applikation	17
6.2 Systemets ydeevne	18
6.3 Robottens bevægelser	19
6.4 Bevægelsernes hastigheder	20
6.5 Bevægelsernes præcision	21
6.6 Robottens løfteevne	21
6.7 Sikkerhed	22
6.8 Startposition	22
6.9 Rækkevidde	22

## 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

Tabel 2 - Versionshistorik

### 3 Indledning

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

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

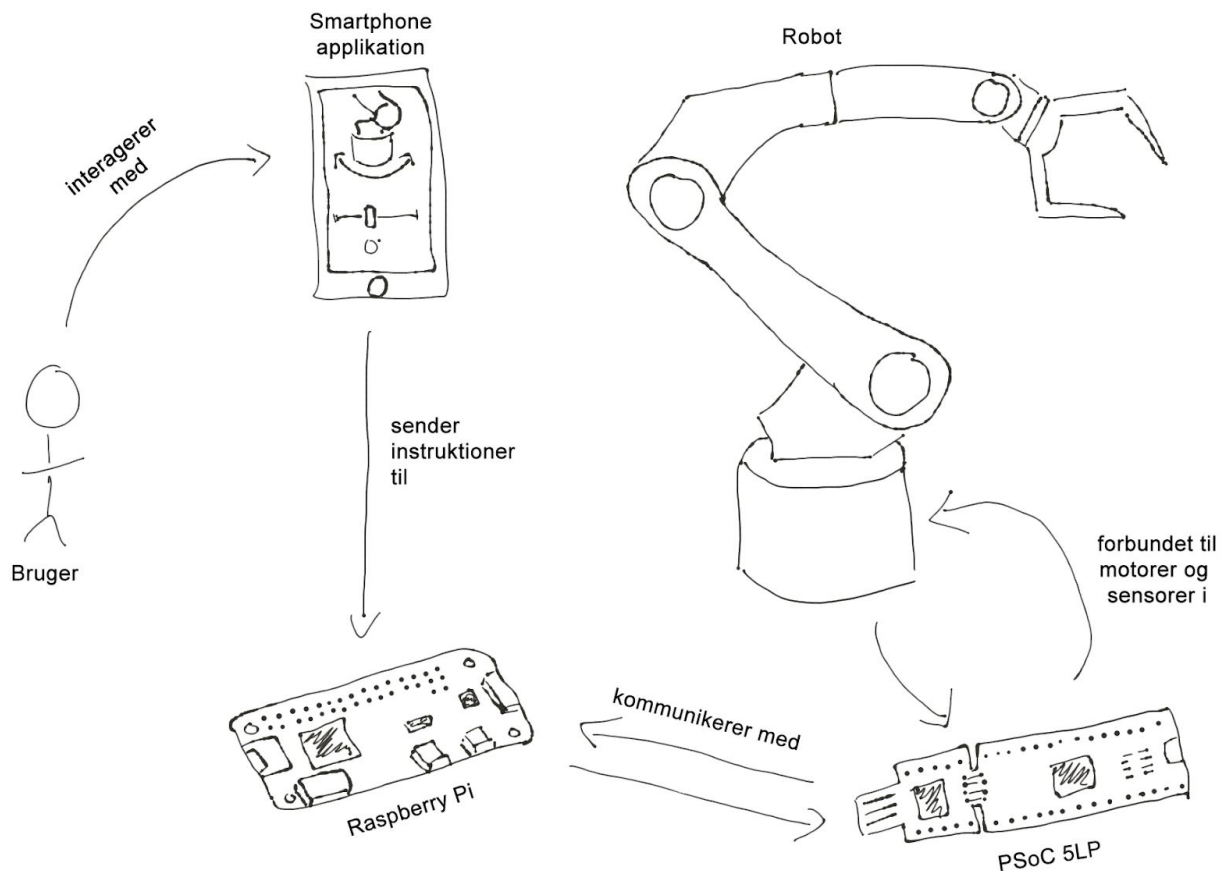
Dernæst specificeres kravene til systemet mere eksplicit under brug af use cases. De fundne use cases prioriteres med MoSCoW og den enkelte use case beskrives yderligere med "brief" og/eller "fully dressed" use case beskrivelser.

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

### 4 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 alt 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*.

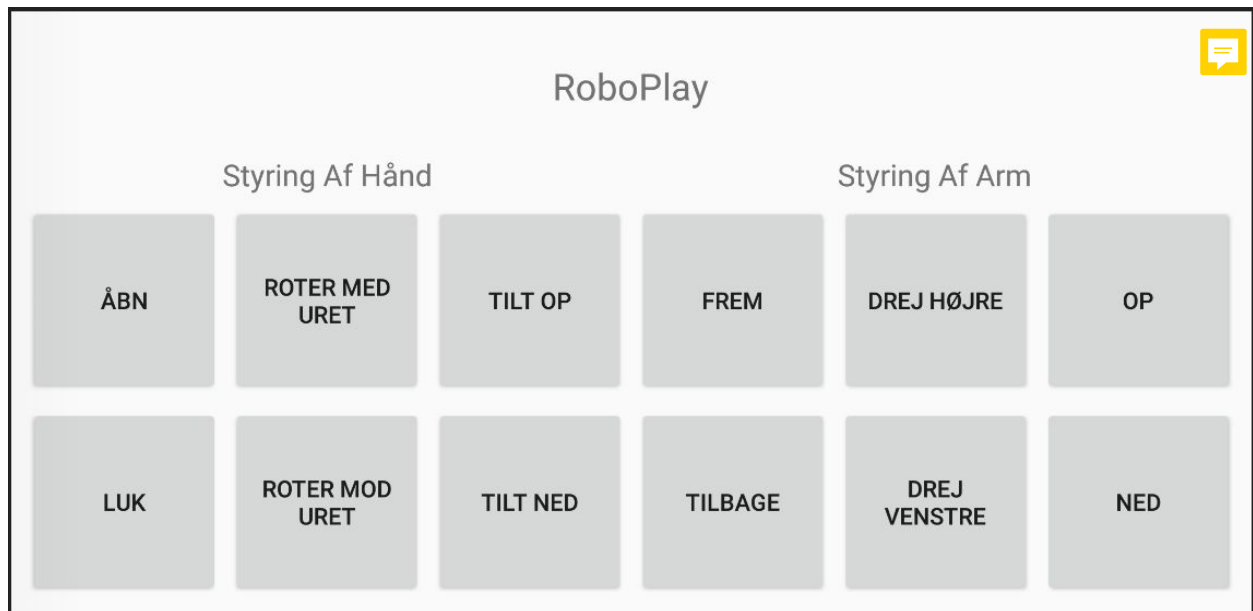


*Figur 1* - Rigtigt billede for RoboPlay der viser kommunikationen mellem brugeren og de forskellige delsystemer.

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 forøve nogen form for personskaade. Sensorer benyttes også til at indstille robottens led til en foruddefinerede startpositioner, når robotten startes op.

Input fra sensorer kommunikeres 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.

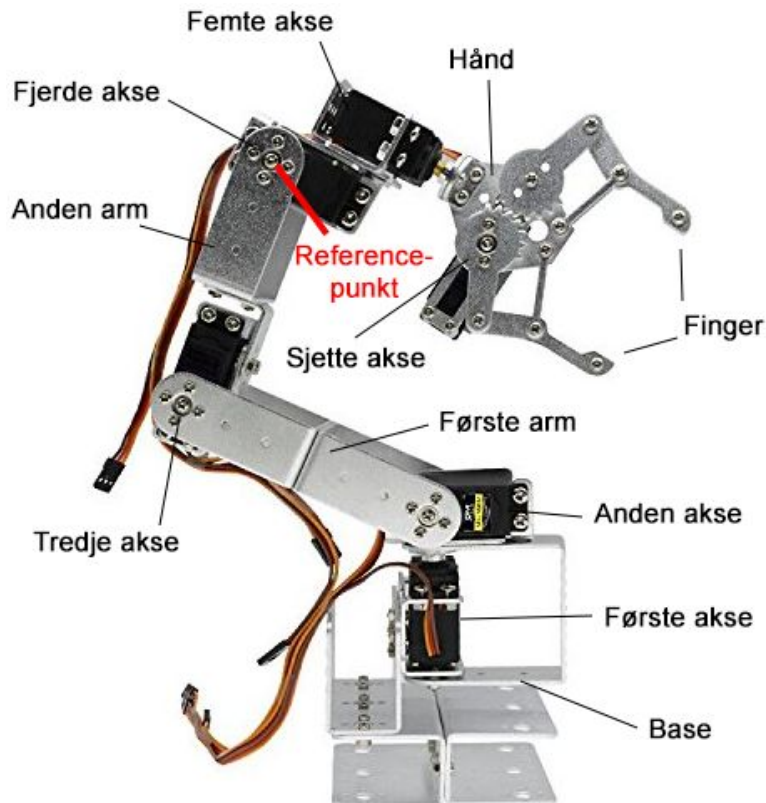


*Figur 2* - Mobilapplikationens design hvor det ses hvilke bevægelser brugeren kan bevæge robotten i. Mens en knap holdes nede udføres bevægelsen ellers er bevægelsen stoppet.

Brugeren kan altså bevæge robotten på i alt 12 forskellige måder, og disse kan inddeles i 2 kategorier:

- Bevægelser der styrer robottens hånd
  - Hånden kan åbnes/lukkes hvorved der kan gribes om genstande som kan slippes igen
  - Hånden kan roteres med/mod uret hvorved det fx er muligt at hælde væske ud af en flaske
  - Hånden kan tiltes op/ned hvilket svarer til at et menneskes håndled bukes op/ned
- Bevægelser der styrer robottens arm
  - Armen kan bevæges frem/tilbage
  - Armen kan bevæges op/ned
  - Armen kan roteres med/mod uret

I dette dokument vil blive henvist til robottens akser, arme, referencepunkt mv., som er defineret ved illustrationen vist på figur *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 MoSCoW analyses formål er alene at give læseren et overblik over hvilke dele af projektet, der prioriteres højest, inden kravene uddybes i de kommende afsnit. Resultatet ses i *tabel 3*.

Prioritet	Beskrivelse
Must	<ol style="list-style-type: none"> <li>1. Robottens hånd skal kunne bevæge sig lineært</li> <li>2. Robotten skal kunne fjernstyres via en grafisk brugergrænseflade</li> <li>3. Robotten skal kunne finde tilbage til en foruddefineret tilstand</li> <li>4. Robotten skal som minimum bestå af en PSoC, en Raspberry Pi, en sensor og en aktuator</li> </ol>



	<ul style="list-style-type: none"> <li>5. Systemet kan kommunikere imellem PSoC, en Raspberry Pi og en mobil applikation</li> <li>6. Robotarmen skal kunne gribe om og løfte en genstand</li> </ul>
Should	<ul style="list-style-type: none"> <li>1. Robotten bør have en anordning til begrænsning af mekanisk kraft</li> </ul>
Could	<ul style="list-style-type: none"> <li>1. Robotten kan have en mulighed for indstilling til begrænsning af mekanisk styrke</li> <li>2. Robotten kan indstilles til at udføre sekvenser af bevægelser via en grafisk brugergrænseflade</li> </ul>
Won't	

Tabel 3 - Overordnet MoSCoW analyse.

## 5 Funktionelle krav

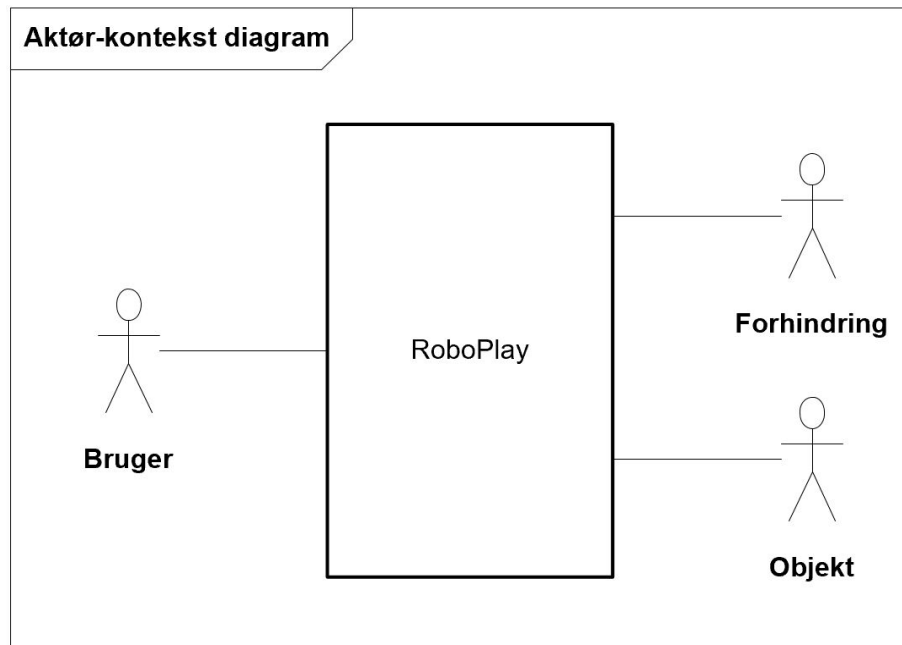
I dette afsnit beskrives først de aktører, der interagerer med RoboPlay. De funktionelle krav, som beskrives med use cases, vises derpå i et use case diagram og beskrives efterfølgende tekstuel i form af "brief" use case beskrivelser.

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

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

### 5.1 Aktør-kontekst diagram

På figur 4 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 4 - Aktør-kontekst diagram hvoraf det ses, at brugeren er den eneste aktør.

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

<b>Aktør navn</b>	Bruger
<b>Type</b>	Primær
<b>Beskrivelse</b>	Brugeren er den person der styrer robotarmen via RoboPlay's smartphone applikation, som er installeret på brugerens mobiltelefon.

Tabel 4 - Aktørbeskrivelse for aktøren "Bruger"

<b>Aktør navn</b>	Forhindring
<b>Type</b>	Sekundær
<b>Beskrivelse</b>	En forhindring er en genstand, som står i vejen for udførelsen af en eller flere af robottens bevægelser.

Tabel 5 - Aktørbeskrivelse for aktøren "Forhindring"

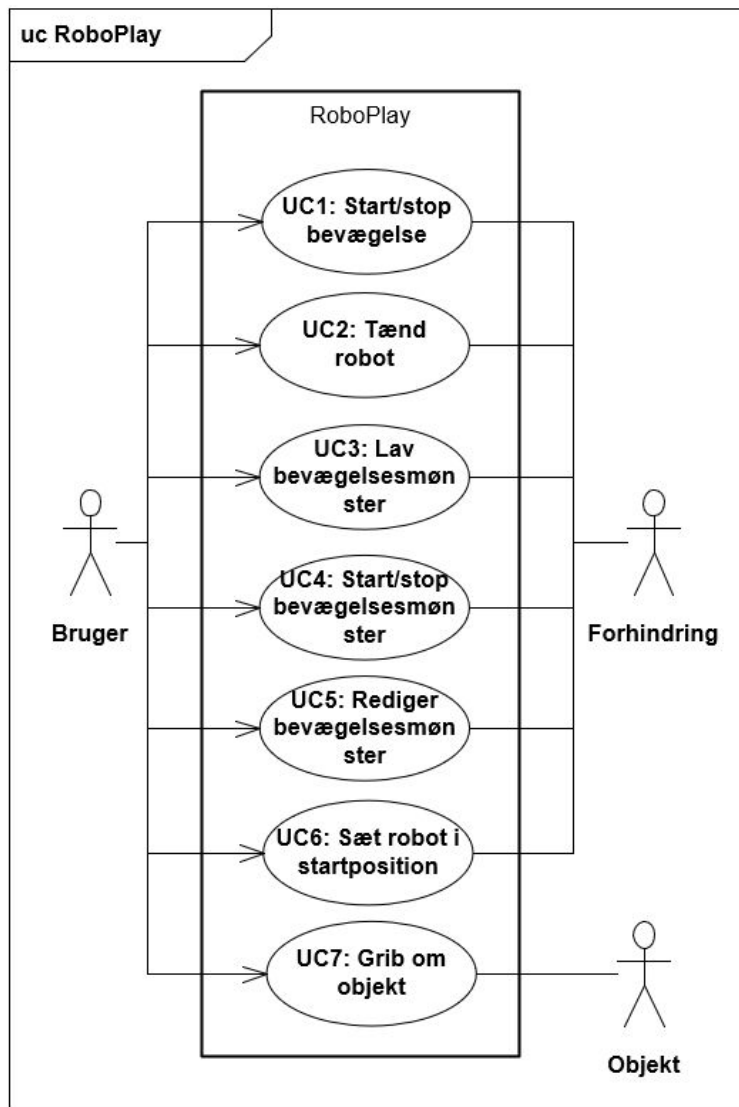
<b>Aktør navn</b>	Objekt
<b>Type</b>	Sekundær

<b>Beskrivelse</b>	Et objekt er en genstand, som robotten kan gribe om med dens hånd.
--------------------	--

Tabel 6 - Aktørbeskrivelse for aktøren "Objekt"

## 5.2 Use case diagram

På figur 5 ses samtlige identificerede use cases i et use case diagram.



Figur 5 - Use case diagram som viser samtlige identificerede use cases.

## 5.3 Brief use case beskrivelser

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

### 5.3.1 UC1: Start/stop 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.

### 5.3.2 UC2: Tænd robot

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

### 5.3.3 UC3: Lav bevægelsesmønster

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

### 5.3.4 UC4: Start/stop bevægelsesmønster

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

### 5.3.5 UC5: Rediger bevægelsesmønster

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

### 5.3.6 UC6: Sæt robot i startposition

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

### 5.3.7 UC7: Grib om objekt

UC7 omhandler den brugssituation, hvor brugeren benytter RoboPlay's smartphone applikation 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.

## 5.4 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 7*.

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	Start/stop 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 7* - Prioritering af de funktionelle krav under brug af MoSCoW

## 5.5 Fully dressed use case beskrivelser

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

### 5.5.1 UC1: Start/stop bevægelse

<b>Navn</b>	UC1: Start/stop bevægelse
<b>Mål</b>	Brugeren starter/stopper en bevægelse
<b>Initiering</b>	Bruger
<b>Aktører</b>	Primær: Bruger Sekundær: Forhindring
<b>Referencer</b>	Ingen
<b>Antal samtidige forekomster</b>	11 (Den 12. bevægelse, "Luk" behandles i UC7)
<b>Prækondition</b>	Systemet er operationelt
<b>Postkondition</b>	Den ønskede bevægelse er startet/stoppet
<b>Hovedscenarie</b>	<ol style="list-style-type: none"> <li>1. Brugeren holder en knap på brugergrænsefladen nede for at starte en <u>bevægelse</u>. [Ext 1: Bruger stopper bevægelse]</li> <li>2. Systemet starter bevægelsen, som vedrører knappen der er trykket på i punkt 1.</li> <li>3. Brugeren slipper knappen, som blev holdt nede i punkt 1. [Ext 2: Yderpunkt nået] [Ext 3: Forhindring detekteret]</li> <li>4. Systemet stopper bevægelsen, som blev startet i punkt 1.</li> </ol>
<b>Udvidelser eller undtagelser</b>	<p>[Ext 1: Bruger stopper bevægelse]</p> <ol style="list-style-type: none"> <li>1. Brugeren slipper en knap på brugergrænsefladen for at stoppe en <u>bevægelse</u>.</li> <li>2. Systemet stopper bevægelsen nævnt i Ext 1's punkt 1. </li> <li>3. UC afsluttes.</li> </ol> <p>[Ext 2: Yderpunkt nået]</p> <ol style="list-style-type: none"> <li>1. Systemet detekterer at bevægelsen, som blev startet i hovedscenariets punkt 1, har nået sit yderpunkt.</li> <li>2. Systemet stopper bevægelsen, som blev startet i hovedscenariets punkt 1.</li> </ol>

	<p>3. UC afsluttes.</p> <p>[Ext 3: Forhindring detekteret]</p> <ol style="list-style-type: none"> <li>1. Systemet detekterer en forhindring og standser alle igangværende bevægelser.</li> <li>2. UC afsluttes.</li> </ol>
<b>Data variationsliste</b>	<p>Bevægelse</p> <ul style="list-style-type: none"> <li>• Mulige bevægelser er de bevægelser, der er noteret under det ikke funktionelle krav "K1.4" dog undtaget "Luk" som er en del af UC7.</li> </ul>

Tabel 8 - "Fully dressed" use case beskrivelse for UC1: Start/stop bevægelse

### 5.5.2 UC2: Tænd robot

<b>Navn</b>	UC2: Tænd robot
<b>Mål</b>	At starte systemet op og klargøre det til brug
<b>Initiering</b>	Bruger
<b>Aktører</b>	Primær: Bruger
<b>Referencer</b>	UC6: Sæt robot i startposition
<b>Antal samtidige forekomster</b>	1
<b>Prækondition</b>	Robotter er slukket men tilsluttet strøm.
<b>Postkondition</b>	Robotten står i startposition og er klar til brug
<b>Hovedscenarie</b>	<ol style="list-style-type: none"> <li>1. Bruger tænder for robot.</li> <li>2. Gå til UC6 punkt 2.</li> </ol>
<b>Udvidelser eller undtagelser</b>	
<b>Data variationsliste</b>	

Tabel 9 - "Fully dressed" use case beskrivelse for UC2: Tænd robot

### 5.5.3 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.

#### 5.5.4 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.

#### 5.5.5 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.

#### 5.5.6 UC6: Sæt robot i startposition

<b>Navn</b>	UC6: Sæt robot i startposition
<b>Mål</b>	At indstille robottens akser til deres startpositioner
<b>Initiering</b>	Bruger
<b>Aktører</b>	Primær: Bruger Sekundær: Forhindring
<b>Referencer</b>	Ingen
<b>Antal samtidige forekomster</b>	1
<b>Prækondition</b>	Robotten er tilsluttet strøm og tændt
<b>Postkondition</b>	Alle akser er indstillet til deres startpositioner
<b>Hovedscenarie</b>	<ol style="list-style-type: none"><li>1. Bruger vælger "Startposition" på mobilapplikation.</li><li>2. Robotter starter nederste vertikale led i en opadgående bevægelse og kører til den registrere endepunktet for ledet. [Ext. 1: Robotten møder en forhindring]</li><li>3. Robotter starter nederste vertikale led i en nedadgående bevægelse og kører et i koden foruddefineret antal grader til den har nået midterpunktet for leddet.</li><li>4. Robotter starter midterste vertikale led i en opadgående bevægelse og kører til den registrere endepunktet for ledet. [Ext. 1: Robotten møder en forhindring]</li><li>5. Robotter starter midterste vertikale led i en nedadgående bevægelse og kører et i koden foruddefineret antal grader til den</li></ol>





	<p>har nået midterpunktet for leddet.</p> <ol style="list-style-type: none"> <li>6. Robotter starter øverste vertikale led i en opadgående bevægelse og kører til den registrere endepunktet for leddet. [Ext. 1: Robotten møder en forhindring]</li> <li>7. Robotter starter øverste vertikale led i en nedadgående bevægelse og kører et i koden foruddefineret antal grader til den har nået midterpunktet for leddet.</li> <li>8. Robotten åbner hånden til den registrere modstand. </li> <li>9. Robotten lukker hånden til den registrere modstand.</li> <li>10. Robotten roterer sin base med uret til den registrere endepunktet i retning med uret. [Ext. 1: Robotten møder en forhindring]</li> <li>11. Robotten roterer sin base mod uret til den når sit midtpunkt. </li> <li>12. Robotten er klar til brug.</li> </ol>
<b>Udvidelser eller undtagelser</b>	<p>Ext. 1: Robotten møder en forhindring</p> <ol style="list-style-type: none"> <li>1. Robot stopper alle bevægelser.</li> <li>2. Use case afbrydes.</li> </ol>
<b>Data variationsliste</b>	

Tabel 10 - "Fully dressed" use case beskrivelse for UC6: Sæt robot i startposition.

### 5.5.7 UC7: Grib om objekt

<b>Navn</b>	UC7: Grib om objekt
<b>Mål</b>	Robottens hånd griber fat om et objekt
<b>Initiering</b>	Bruger
<b>Aktører</b>	Primær: Bruger Sekundær: Objekt
<b>Referencer</b>	Ingen
<b>Antal samtidige forekomster</b>	1

<b>Prækondition</b>	Systemet er operationelt. Brugeren har indstillet robotten således, at et objekt er placeret mellem robottens fingre.
<b>Postkondition</b>	Robottens hånd har grebet fast om det ønskede objekt.
<b>Hovedscenarie</b>	<ol style="list-style-type: none"> <li>1. Brugeren holder brugergrænsefladens knap benævnt "Luk" nede.</li> <li>2. Systemet iværksætter bevægelsen "Luk".</li> <li>3. Systemet detekterer at hånden har grebet fast om objektet. [Ext 1: Brugeren stopper bevægelsen]</li> <li>4. Systemet stopper bevægelsen "Luk".</li> </ol>
<b>Udvidelser eller undtagelser</b>	[Ext 1: Brugeren stopper bevægelsen] <ol style="list-style-type: none"> <li>1. Brugeren slipper brugergrænsefladens knap benævnt "Luk".</li> <li>2. Systemet stopper bevægelsen "Luk".</li> <li>3. UC afsluttes.</li> </ol>
<b>Data variationsliste</b>	

Tabel 11 - "Fully dressed" use case beskrivelse for UC7: Grib om objekt



## 6 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
- Grader +/- 3 grader

### 6.1 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 er nødvendigt.	M
K1.4	<p>Applikationen skal have 12 knapper, som følger:</p> <ul style="list-style-type: none"> <li>• "Åbn"</li> <li>• "Luk"</li> <li>• "Roter med uret"</li> <li>• "Roter mod uret"</li> <li>• "Tilt op"</li> <li>• "Tilt ned"</li> <li>• "Frem"</li> <li>• "Tilbage"</li> <li>• "Drej højre"</li> <li>• "Drej venstre"</li> <li>• "Op"</li> <li>• "Ned"</li> </ul> <p>Hver knap henfører til en bevægelse og disse er defineret i <a href="#">afsnit 6.3</a>.</p>	
K1.5	Via applikationen skal det være muligt at udføre op til 3 forskellige bevægelser på én gang.	S

Tabel 12 - De ikke-funktionelle krav til smartphone applikationen.

## 6.2 Systemets ydeevne



ID	Navn	Prioritet
K2.1	Fra en bevægelse initieres/stoppes på smartphone applikationen til robotten påbegynder udførelsen af kommandoen må der maksimalt gå 1 sekund.	M
K2.2	Fra en bevægelse initieres/stoppes på smartphone applikationen til robotten påbegynder udførelsen af kommandoen må der maksimalt gå 0.5 sekund.	S
K2.3	Fra en bevægelse initieres/stoppes på smartphone applikationen til robotten påbegynder udførelsen af kommandoen må der maksimalt gå 0.1 sekund.	C

K2.4	Robotten skal være klar til brug indenfor 1 minut efter opstart.	M
------	--	---

Tabel 13 - De ikke-funktionelle krav til systemets ydeevne.



## 6.3 Robottens bevægelser

ID	Navn	Prioritet
K3.1	Bevægelsen "Åbn" er defineret ved, at robottens sjette akse bevæges så afstanden mellem robottens fingre øges.	M
K3.2	Bevægelsen "Luk" er defineret ved, at robottens sjette akse bevæges så afstanden mellem robottens fingre mindskes.	M
K3.3	Bevægelsen "Roter med uret" er defineret ved, at robottens femte akse bevæges modsat "Roter hånd mod uret".	M
K3.4	Bevægelsen "Roter mod uret" er defineret ved, at robottens femte akse bevæges modsat af bevægelsen "Roter med uret". 	M
K3.5	Bevægelsen "Tilt op" er defineret ved, at robottens fjerde akse bevæges modsat af bevægelsen "Tilt hånd ned".	M
K3.6	Bevægelsen "Tilt ned" er defineret ved, at robottens fjerde akse bevæges modsat af bevægelsen "Tilt hånd op".	M
K3.7	Bevægelsen "Frem" er defineret ved, at robottens anden, tredje og fjerde akse bevæges samtidigt således, at robottens referencepunkt bevæger sig væk fra basen i en vandret bevægelse.	M
K3.8	Bevægelsen "Tilbage" er defineret ved, at robottens anden, tredje og fjerde akse bevæges samtidigt således, at robottens referencepunkt bevæger sig mod basen i en vandret bevægelse.	M
K3.9	Bevægelsen "Drej højre" er defineret ved, at robottens første akse bevæges modsat af bevægelsen "Drej venstre".	M
K3.10	Bevægelsen "Drej venstre" er defineret ved, at robottens første akse bevæges modsat af bevægelsen "Drej højre".	M
K3.11	Bevægelsen "Op" er defineret ved, at robottens anden, tredje og fjerde akse bevæges samtidigt således, at robottens referencepunkt bevæger sig væk fra underlaget basen står på i en lodret bevægelse. 	M
K3.12	Bevægelsen "Ned" er defineret ved, at robottens anden, tredje og fjerde akse bevæges samtidigt således, at robottens referencepunkt	M

	bevæger mod underlaget basen står på i en lodret bevægelse.	
K3.13	Første akse har en bevægelsesfrihed fra yderpunkt til yderpunkt på 180 grader.	M
K3.14	Anden akse har en bevægelsesfrihed fra yderpunkt til yderpunkt på 180 grader.	M
K3.15	Tredje akse har en bevægelsesfrihed fra yderpunkt til yderpunkt på 180 grader.	M
K3.16	Fjerde akse har en bevægelsesfrihed fra yderpunkt til yderpunkt på 180 grader.	M
K3.17	Femte akse har en bevægelsesfrihed fra yderpunkt til yderpunkt på 180 grader.	M
K3.18	Sjette akse kan åbne hånden op med et mellemrum på X cm.	M

Tabel 14 - De ikke-funktionelle krav til robotens bevægelser.

## 6.4 Bevægelsernes hastigheder

ID	Navn	Prioritet
K4.1	Første akse skal bevæge sig med en hastighed på minimum 20 grader pr. sekund og maksimum 30 grader pr. sekund.	S
K4.2	Fjerde akse skal bevæge sig med en hastighed på minimum 20 grader pr. sekund og maksimum 30 grader pr. sekund.	S
K4.3	Femte akse skal bevæge sig med en hastighed på minimum 20 grader pr. sekund og maksimum 30 grader pr. sekund.	S
K4.4	Sjette akse skal bevæge sig med en hastighed på minimum 20 grader pr. sekund og maksimum 30 grader pr. sekund.	S
K4.5	Bevægelsen "Bevæg arm frem" skal flytte robotens referencepunkt med minimum 0.5 cm/s og maksimum 3 cm/s.	S
K4.6	Bevægelsen "Tilbage" skal flytte robotens referencepunkt med minimum 0.5 cm/s og maksimum 3 cm/s.	S
K4.7	Bevægelsen "Op" skal flytte robotens referencepunkt med minimum 0.5 cm/s og maksimum 3 cm/s.	S
K4.8	Bevægelsen "Ned" skal flytte robotens referencepunkt med minimum	S

	0.5 cm/s og maksimum 3 cm/s.	
--	------------------------------	--

Tabel 15 - De ikke-funktionelle krav til de hastigheder, som robottens bevægelser kan udføres med.

## 6.5 Bevægelsernes præcision

ID	Navn	Prioritet
K5.1	Fra samme udgangsposition for robotten skal ens input til bevægelsen "Frem" resultere i samme position for robottens referencepunkt +/- 2 cm.	S
K5.2	Fra samme udgangsposition for robotten skal ens input til bevægelsen "Tilbage" resultere i samme position for robottens referencepunkt +/- 2 cm.	S
K5.3	Fra samme udgangsposition for robotten skal ens input til bevægelsen "Op" resultere i samme position for robottens referencepunkt +/- 2 cm.	S
K5.4	Fra samme udgangsposition for robotten skal ens input til bevægelsen "Ned" resultere i samme position for robottens referencepunkt +/- 2 cm.	S

Tabel 16 - De ikke-funktionelle krav til den præcision som robottens bevægelser kan udføres med.

## 6.6 Robottens løfteevne

ID	Navn	Prioritet
K6.1	Robothånden skal kunne gribe om objekter der måler minimum 1x1x1 cm.	M
K6.2	Robothånden skal kunne gribe om objekter der måler maksimum 10x10x10 cm.	M
K6.3	Robotten skal kunne løfte objekter der vejer maksimum 500 gr.	M

Tabel 17 - De ikke-funktionelle krav vedrørende robottens løfteevne.

## 6.7 Sikkerhed

ID	Navn	Prioritet
K7.1	Sjette akse skal standse sin bevægelse hvis robottens fingre presses sammen med en kraft på 10 newton eller mere.	M
K7.2	Robotten skal standse alle aksers bevægelse hvis en forhindring påvirker robotten med 10 newton eller mere.	M

Tabel 18 - De ikke-funktionelle krav vedrørende sikkerheden under brug af robotten.

## 6.8 Startposition

ID	Navn	Prioritet
K8.1	Første akse startposition er midten mellem aksens yderpunkter.	S
K8.2	Anden akse startposition er midten mellem aksens yderpunkter.	S
K8.3	Tredje akse startposition er midten mellem aksens yderpunkter.	S
K8.4	Fjerde akse startposition er midten mellem aksens yderpunkter.	S
K8.5	Femte akse startposition er midten mellem aksens yderpunkter.	S
K8.6	Sjette akse startposition er midten mellem aksens yderpunkter.	S

Tabel 19 - De ikke-funktionelle krav til de positioner robottens akser skal indstilles i for at robotten indtager sin "Startposition".

## 6.9 Rækkevidde

ID	Navn	Prioritet
K9.1	Robotten skal kunne gribe om og løfte et objekt, som befinder sig i en radius på mindst 20 cm fra basens centrum.	S
K9.2	Robotten skal kunne gribe om og løfte et objekt, som befinder sig i en radius på maksimalt 60 cm fra basens centrum.	S

Tabel 20 - De ikke-funktionelle krav til robottens rækkevidde.