

---

Titel:

**Accepttestrapport  
for  
Sorting Industrial Robot  
(SIR)**

**Versionshistorik**

Ver.	Dato	Initialer	Beskrivelse
0.1	11-02-12	RHT	Første udkast.
0.2	16-02-12	RHT	Indsat firmanavn og produktnavn.
0.3	07-03-12	NIQ	Rettet format fejl
1.0	08-03-12	Alle	Første version før Construction.
1.1	02-04-12	Alle	Revideret version efter iteration 1.
1.2	23-04-12	CTD	Merge med alle branches
1.3	21-05-12	SLT	Rettet til efter review-mødet.
1.4	23-05-12	SLT	Rettet til med endelige use cases.
1.5	01-06-12	SLT, RHT	Endelig version.

**Godkendelsesformular**

<b>Forfatter(e):</b>	Søren Howe Gersager(10430) Cong Thanh Dao(10517) Yusuf Tezel(10568) Nicolaj Quottrup(10754) René Høgh Thomsen(10778) Michael Batz Hansen(10791) Sam Luu Tong(10898)
<b>Godkendes af:</b>	Poul Ejnar Røvsing
<b>Projektnr:</b>	1
<b>Filnavn:</b>	Accepttest.odt Accepttest.pdf
<b>Antal sider:</b>	22
<b>Kunde:</b>	Robotic Global Organization(RoboGO)

**Ved underskrivelse af dette dokument accepteres det af begge parter, som værende kravene til udviklingen af det ønskede system.**

**Sted og dato:**

---

René H. Thomsen

---

Poul Ejnar Røvsing

## Indholdsfortegnelse

1. Indledning.....	4
1.1 Formål.....	4
1.2 Referencer.....	4
1.3 Omfang.....	5
1.4 Godkendelses kriterier.....	5
1.5 Definitioner.....	6
2. Testspecifikation.....	7
2.1 Hardware og Software ID.....	7
2.2 Testsystemet.....	7
2.3 Identifikation af testobjekter.....	8
2.4 Testobjekt forberedelse.....	9
2.5 Testmiljø.....	9
3. Testprocedure.....	10
3.1 Test af funktionelle krav.....	10
3.1.1 Use Case 1: Starte/stoppe systemet.....	10
3.1.2 Use Case 2: Styre klodsplacering.....	11
3.1.3 Use Case 3: Tjekke loggen. ....	12
3.1.4 Use Case 4: Manuelt styre.....	13
3.1.5 Use Case 5: Skifte mellem robot og simulator. ....	14
3.1.6 Use Case 6: Se klodser lagret.....	15
3.1.7 Use Case 7: Login. ....	15
3.1.8 Use Case 8: Indlæse/køre systemet DSL-filer direkte.....	16
3.2.1 Use Case 9: Se simulator position.....	17
3.2.2 Use Case 10: Simulere koden.....	18
3.3.1 Use Case 11: Se DSL manual.....	19
3.3.2 Use Case 12: Åbne/lukke filer i faner.....	19
3.3.3 Use Case 13: Gemme filer.....	21
3.3.4 Use Case 14: Eksekvere DSL kode. ....	22
3.4.1 Use Case 15: Måle vægten af objekter. ....	23
3.2 Test af øvrige specifikke krav.....	24
3.2.1 Test case: Krav 1:.....	24
3.2.2 Test case: Krav 2:.....	24
4. Godkendelse.....	25

# 1. Indledning

## 1.1 Formål

Dette dokument indeholder en accepttestspecifikation for systemet kaldet SIR, Sorting Industrial Robot. Dokumentet specificerer accepttesten og vil i udfyldt stand senere udgøre accepttestrapporten.

For at teste SIR udføres tre niveauer af test:

### 1. Enhedstest:

Dette omfatter test af de enkelte funktioner implementeret i klasserne, (modulerne) som softwaren sammenstykket af.

### 2. Integrationstest:

Dette omfatter test af grænseflader mellem klasser (moduler), samt test af hele systemets funktionalitet.

### 3. Accepttest:

Dette omfatter en test af funktionelle krav fra kravspecifikationen. Endvidere vil usikkerheder fra kravspecifikationen være gældende i accepttesten.

Dette dokument omhandler punkt tre i accepttesten.

Ændringer i accepttestspecifikationen beskrives i dokumentets versionshistorie, således der hurtigt kan spores tilbage på betydende ændringer.

## 1.2 Referencer

I dette punkt indgår referencer til relevante projektdokumenter, som indgår i SIR:

- Kravspecifikation
- Produktoplæg (findes på <http://kurser.iha.dk/eit/i4prj4/> )

### 1.3 Omfang

Denne accepttest vil dække over hele systemet, hvorved al funktionalitet vil blive eftertestet. Dette gør sig gældende for alle use cases og deres undtagelser. Dernæst vil det blive taget op, om produktet godkendes eller ej ud fra kundens side.

### 1.4 Godkendelses kriterier

Godkendelsen af accepttesten består af to trin:

1. Godkendelse af accepttestspecifikationen. Dette gøres på side to af dokumentet i "godkendes af" feltet.
2. Godkendelse af selve accepttesten. Dette gøres i 'Godkendelse'.

Accepttesten er afsluttet, når de testspecificerede cases er gennemført.

Hvis der under accepttesten opstår fejl, der umuliggør fortsat udførsel af de efterfølgende test cases, afbrydes accepttesten.

Hvis der opstår fejl i enkelte test cases; men fortsat accepttest er mulig, underkendes den enkelte test, og accepttesten forsættes med efterfølgende test cases.

Såfremt en test afbrydes, eller et test case underkendes, skal der udfærdiges en problemrapport, der beskriver årsagen til underkendelse.

## 1.5 Definitioner

Accepttestspecifikation	Dokument som specificerer test af funktionelle krav fra kravspecifikationen.
Accepttestrapport	I udfyldt stand vil accepttesten udgøre en rapport. Rapporten godkendes i afsnit 4.
Internt testobjekt Eksternt testobjekt	De objekter/testemner der er omfattet af denne accepttest Objekt der anvendes for at kunne udføre testen men som ikke er omfattet af godkendelse af accepttesten. En defekt fundet i et testobjekt vil således ikke umiddelbart kunne medføre underkendelse af accepttesten.

## 1.6 Dokumentets opbygning

- Afsnit 1. Introducerer dokumentet og forklarer reglerne for anvendelse af dokumentet.
- Afsnit 2. Specifikation af testen; identifikation af hardware og software, testoversigt, testobjekter, testforberedelse og miljø (omgivelser).
- Afsnit 3. Beskrivelse af testprocedurer og deres tilknyttede test cases.
- Afsnit 4. Godkendelse efter udført test.

## 2. Testspecifikation

Testspecifikation for SIR

### 2.1 Hardware og Software ID

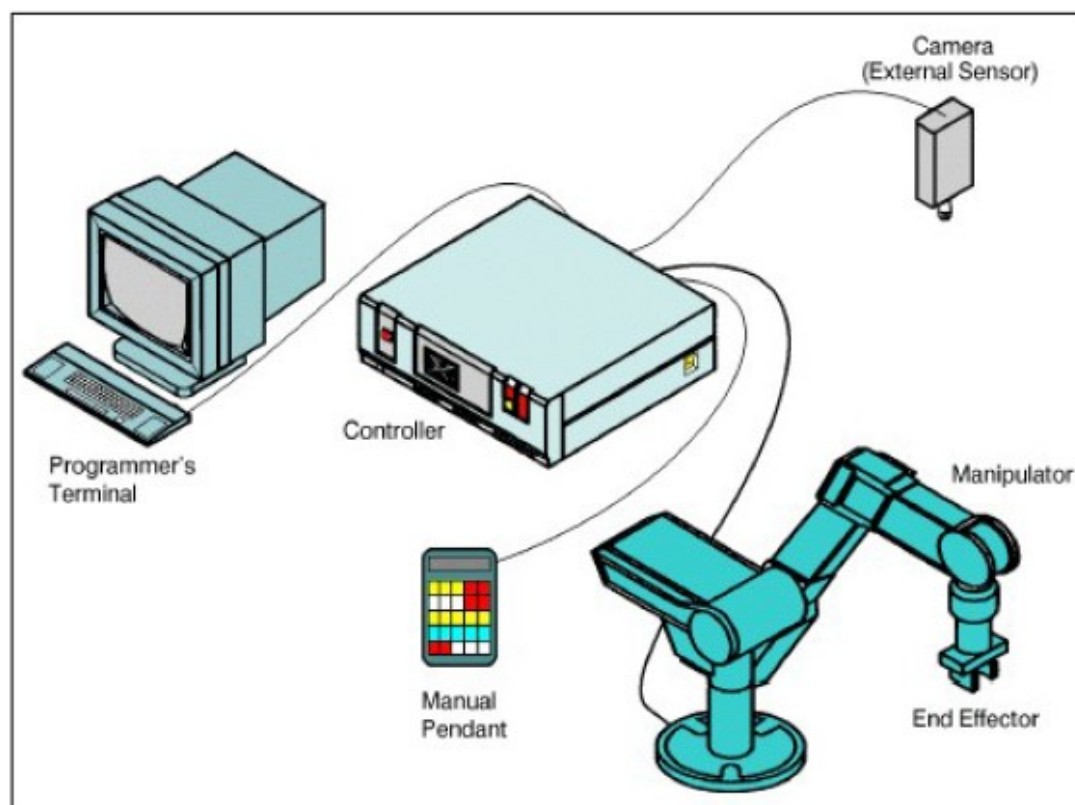
Software der skal tests:

Software	Version	Udgivelses dato	Bemærkning
SIR	1.0	01-06-2012	
IDE	1.0	01-06-2012	
Simulation	1.0	01-06-2012	

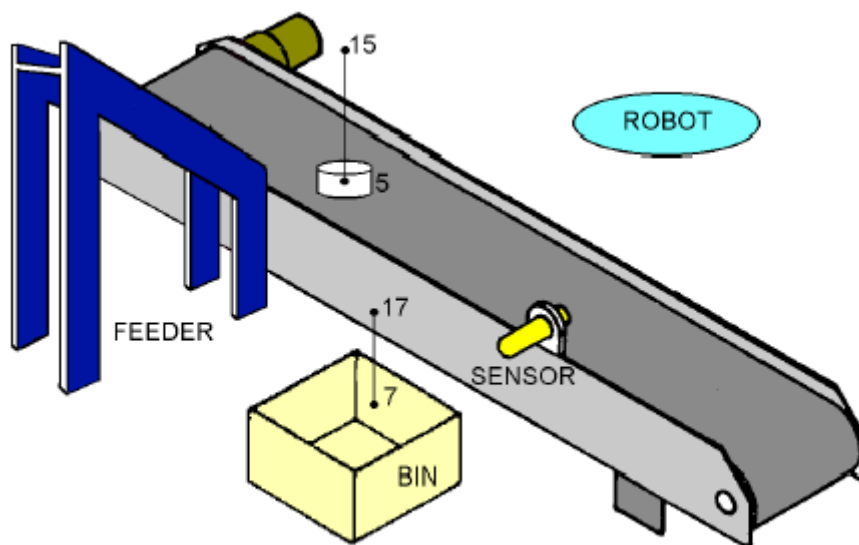
Hardware der skal testes:

Hardware	Version	Udgivelses dato	Bemærkning
Scorbot ER 4-U	-	01-06-2012	
Vægt	1.0	01-06-2012	

### 2.2 Testsystemet



Figur 1: Visuel illustration af forbindelser i systemet.



*Figur 2: Visuel illustration over opstilling af systemet.*

SIR main programmet, IDE og simulations programmet eksekveres på en PC, som er forbundet til en USB Controller. Via USB controlleren styres SCORBOT armen, vægten og transportbåndet.

## 2.3 Identifikation af testobjekter

Interne testobjekter:

- SIR main program
- IDE
- Simulations program
- Elektronikdel til vægt

Eksterne testobjekter

- Scrobot ER 4-U
- Vejecellen
- Transportbåndet
- USB-Controller



## **2.4 Testobjekt forberedelse**

Der skal som minimum være forbindelse til IHA's Intranet, hvor der kan forbindes til IHA's SQL database. SCORBOT, transportbåndet, sensorerne skal være i de faste positioner, da de blev udleveret. Vægten skal placeres i den foruddefineret position. Desuden skal styresystemet være startet op.

## **2.5 Testmiljø**

Der skal testes i et ikke-støvet og ikke-fugtigt lokale. Temperaturen i lokalet skal være mellem 15– 30 graders celsius. Der må ikke være direkte sollys på testobjekterne. Al nydelse af mad og drikkevarer må ikke foregå ved siden af testobjekterne.

### 3. Testprocedure

Dette punkt bliver skrevet, efterhånden som user stories fra Product Backlog bliver valgt til sprintene.

#### 3.1 Test af funktionelle krav

##### 3.1.1 Use Case 1: Styre klodsplacering.

###### Test case: normalforløb

Forberedelse:	Serveren er oppe og køre.
Beskrivelse:	Brugeren får ændret værdierne i databasen og gemmer disse.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Bruger går ind i databasen og ændrer værdier for positionerne med X = 3, Y = 5, Z = 10.	Værdierne i tabellen ændres.		
2.	Brugeren gemmer værdierne ved tryk af 'Save'.	Værdierne gemmes.		

**3.1.2 Use Case 2: Tjekke loggen.****Test case: normalforløb**

Forberedelse: Programmet har været oppe og køre.  
 Beskrivelse: Der laves en log, som logger en liste af events, der er forekommet.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren tjekker loggen.	Brugeren kan se loggen over events, der er foretaget siden systemets start.		

**3.1.3 Use Case 3: Foretage manuel styring.****Test case: normalforløb**

Forberedelse: Ingen.  
 Beskrivelse: Robotten bevæger sig alt efter, hvad brugeren beder den om i runtime.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren vælger manuel styring.	Brugergrænseflade for manuel styring vil blive vist, hvor der vises muligheder for manuel styring.		
2.	Brugeren interagerer med brugergrænsefladen med musen.	Robotten vil bevæge sig alt efter brugerens input.		

**Test case: Undtagelsesforløb 3.1**

Forberedelse: En af akserne har nået sin rotationsgrænse ved  
 benyttelse af robotten.  
 Beskrivelse: Robotten bevæger sig ikke.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren får robotten til at bevæge sig med en akse som har nået sin rotationsgrænse.	Robotten vil ikke bevæge sig i retningen.		

**3.1.4 Use Case 4: Skifte mellem robot og simulator.****Test case: normalforløb**

Forberedelse: Ingen.  
 Beskrivelse: Der skiftes til robotten fra simulatoren.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren tjekker forbindelsen mellem GUI og simulator ved knappen "Check Connectivity"	Der indikeres en forbindelse ved at lyse grøn ved progressbar.		

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
2.	Brugeren skifter til robotten og trykker på 'Select'.	Der står i combo-boksen, at robotten er valgt.		
3.	Brugeren tjekker forbindelsen mellem GUI og robotten ved knappen "Check Connectivity"	Der indikeres en forbindelse ved at lyse grøn ved progressbar.		

**Test case: Undtagelsesforløb 4.1**

Forberedelse:

Ingen.

Beskrivelse:

Brugeren kan ikke få forbindelse til robotten.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren tjekker forbindelsen mellem GUI og robotten ved knappen "Check Connectivity"	Der indikeres, at der ikke er en forbindelse ved at lyse rødt ved progressbar.		

**3.1.5 Use Case 5: Se klodser lagret.****Test case: normalforløb**

Forberedelse: Ingen.  
Beskrivelse: Brugeren får en liste over klodser, der er placeret.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren vælger Info tab, herefter vælges en tabel.	Brugeren får en liste af klodserne, der er placeret.		

**3.1.6 Use Case 6: Logge ind.****Test case: normalforløb**

Forberedelse: Ingen.  
Beskrivelse: Brugeren logger ind.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren skriver brugernavnet "TDC2K" og kodeordet "TDC2K".	Brugeren logger ind og kan nu interagere med styresystemet.		

**Test case: Undtagelsesforløb 2.1**

Forberedelse: Ingen.  
 Beskrivelse: Brugeren får en fejlmeddelelse om, at kodeord og brugernavn ikke er gyldigt.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren skriver ugyldigt brugernavn "Poul" og kodeord "Poul".	Der fremkommer besked om ugyldigt kodeord og brugernavn.		

**3.2.1 Use Case 7: Eksekvere koden i simulatoren.****Test case: Normalforløb.**

Forberedelse: Editoren indeholder kørbare kode, og syntaksen for koden er overholdt.  
 Beskrivelse: DSL-koden bliver kørt via simulatoren, hvor der vil blive vist/skrevet, hvad robotten(virtuelt) gør ud fra koden.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren skriver sin kode.	Koden bliver vist i editoren.		
2.	Koden bliver eksekveret i simulatoren.	Simulatoren begynder at vise/skrive, hvad der sker under simulationen.		

**Test case: Extensionforløb 1.1.**

Forberedelse: Der er noget eksisterende kørbare kode klar i forvejen, som er persisteret på harddisken.

Beskrivelse: DSL-koden bliver kørt via simulatoren, hvor der vil blive vist/skrevet, hvad robotten(virtuelt) gør ud fra koden.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren åbner en allerede eksisterende fil med kørbare kode.	Koden bliver vist i editoren.		

**3.3.1 Use Case 8: Se DSL manual.****Test case: normalforløb**

Forberedelse: Ingen.

Beskrivelse: En liste med kommandoer åbnes til rådighed.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren vælger "Help"	Et dropdown menu ses ved klik af "Help".		
2.	Brugeren vælger "View commands"	En liste med kommandoer kommer frem.		



**3.3.2 Use Case 9: Have flere filer åbne.****Test case: normalforløb**

Forberedelse: Der er filer tilgængelige til at blive åbnet og indlæst.  
 Beskrivelse: Flere filer vil blive åbnet.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren vælger "Open file".	Man vil kunne se filer, der kan åbnes.		
2.	Brugeren vælger filerne "Poul1.py" og "Poul2.py"	Filerne bliver åbnet.		

**Test case: Extensionforløb 1.1**

Forberedelse: Der er filer tilgængelige til at blive åbnet og indlæst.  
 Beskrivelse: Brugeren anvender en genvejstast for at åbne flere filer.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren anvender genvejstasten "CTRL + O" for at åbne filer.	Man vil kunne se filer, der kan åbnes.		
2.	Brugeren vælger filerne "Poul1.py" og "Poul2.py"	Filerne bliver åbnet.		

**3.3.3 Use Case 10: Gemme filer****Test case: normalforløb**

Forberedelse: Ingen.  
 Beskrivelse: Brugeren får gemt koden.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren skriver kode i editoren.	Der fremkommer tekst i editoren.		
2.	Brugeren gemmer filen med filnavnet "PER.py" ved brug af menuen.	Filen gemmes og vinduet returneres til editor vinduet.		

**Test case: Extensionforløb 2.1**

Forberedelse: Brugeren befinder sig i IDE-fanen, samt der er skrevet noget tekst/kode i editoren.  
 Beskrivelse: Brugeren anvender genvejstast til at gemme med i stedet for menuen.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Filen gemmes via genvejstasten 'CTRL + S'.	Bruger får mulighed for at gemme sin fil.		
2.	Bruger angiver filnavnet "PER.py" og gemmer.	Filen gemmes og vinduet returneres til editor vinduet.		

**3.3.4 Use Case 11: Eksekvere DSL kode.****Test case: normalforløb**

Forberedelse: Den skrevne kode overholder syntaksen.  
Beskrivelse: Brugeren vil kunne eksekvere kode.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren skriver kode i editoren.	Der fremkommer kode i editoren.		
2.	Brugeren eksekverer koden.	Koden eksekverer, og forløbet fremskrives i GUI'en.		

**Test case: Extensionforløb 1.1**

Forberedelse: Der eksisterer en fil med kørbare DSL-kode.  
Beskrivelse: Brugeren åbner en fil med DSL kode i stedet for at skrive den i editoren.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren åbner en fil med kørbare DSL kode.	Der fremkommer kode i editoren.		
2.	Brugeren eksekverer koden.	Koden kører.		

**3.4.1 Use Case 12: Måle vægten af objekter.****Test case: normalforløb**

Forberedelse: Der er tilgængelige objekter til stede.  
 Beskrivelse: Vægten af objektet måles.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Klodsens lægges på vægten ved brug af robotten.	Der foretages målinger.		
2.	Brugeren skriver <code>getWeight()</code> i IDE'.	Brugeren får værdien fra målingen.		

**3.2 Test af øvrige specifikke krav**

**Krav 1:** Responstiden mellem brugeren og brugergrænsefladen skal være mindre end ét sekund.

**Krav 2:** I IDE komponenten skal responstiden være mindre end ét halvt sekund ved redigering af DSL-filer.

**3.2.1 Test case: Krav 1:**

Forberedelse: Styresystemet er oppe at køre.  
 Beskrivelse: Der bliver interageret med brugergrænsefladen og der tjekkes om det bliver registreret hurtigt nok.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Der bliver interageret	Brugergrænsefladen viser at den har registreret det.(For eksempel		

	med brugergrænsef laden.	ved knap highlight)		
--	--------------------------------	---------------------	--	--

### 3.2.2 Test case: Krav 2:

Forberedelse: Styresystemet er oppe at køre og IDE'en kører.

Beskrivelse: Der bliver skrevet DSL-kode og der tjekkes for  
forsinkelser.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Der bliver kontinuert skrevet kode i IDE'en.	Hvert tastetryk bliver skrevet ind i IDE'en på under et halvt sekund.		

## 4. Godkendelse

Ved afslutningen af accepttesten skrives under på at alle verifikationer er gennemført som aftalt og at accepttesten er succesfuldt gennemført. Eksisterer der problemrapporter efter testen, så noteres antallet og disse vedlægges sammen med rapporten.

Alle verifikationer er gennemført tilfredsstillende \_\_\_\_\_

I modsat fald beskrives hvilken aktion, der aftales :

Problemrapporter (antal): \_\_\_\_\_

\_\_\_\_\_  
IHA - testansvarlig

\_\_\_\_\_  
dato

\_\_\_\_\_  
Kunde - ansvarlig

\_\_\_\_\_  
dato