
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			

Godkendelsesformular

Forfatter(e):	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)
Godkendes af:	Poul Ejnar Røvsing
Projektnr:	1
Filnavn:	Accepttest.odt Accepttest.pdf
Antal sider:	15
Kunde:	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
3.2 Test af øvrige specifikke krav.....	10
3.3 Test af grænseflader.....	10
4. Godkendelse.....	11

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. Testspezifikation

Testspezifikation für SIR

2.1 Hardware og Software ID

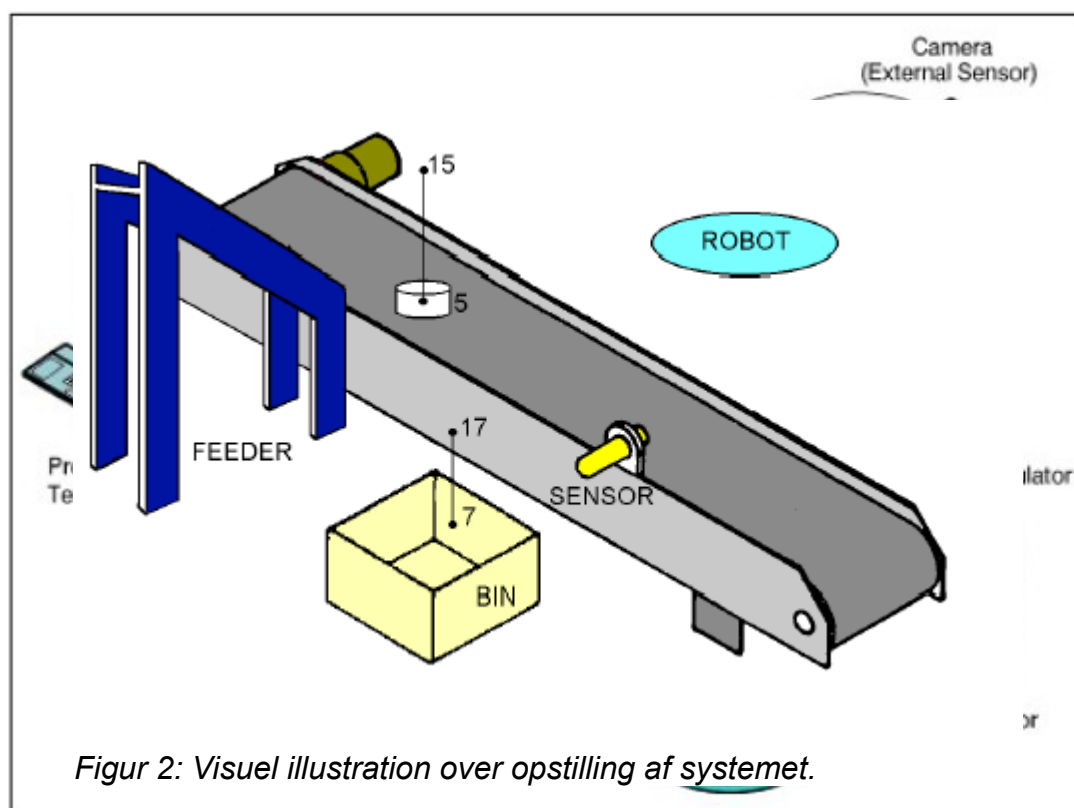
Software der skal tests:

Software	Version	Udgivelses dato	Bemærkning
SIR	1.0		
IDE	1.0		
Simulation	1.0		

Hardware der skal testes:

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

2.2 Testsystemet



Figur 1: Visuel illustration af forbindelser i 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
- EL del 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 forud defineret position.

2.5 Testmiljø

Der skal testes i et ikke-støvet og 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

3.1.2 Use case 37: Have DSL arbejdsområde

Test case: normalforløb

Forberedelse: IDE editoren er åbent via SIR programmet.

Beskrivelse: Der åbnes et nyt tomt dokument i editoren, hvor der kan skrives ny kode.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren trykker på "Filer"	Et dropdown menu ses ved klik af "Filer".		
2.	Brugeren trykker på "Ny Script"	Der vises et nyt tomt dokument.		
3.	Brugeren kan nu skrive på det nye dokument. Der skrives "Hello World"	Der ses Hello World på editoren.		

Test case: Afvigelses forløb 2

Forberedelse: IDE editoren er åbent via SIR programmet.

Beskrivelse: Der åbnes et nyt tomt dokument i editoren, hvor der kan skrives ny kode.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren trykker på "Filer"	Et dropdown menu ses ved klik af "Filer".		
2.	Brugeren trykker på "Ny Script"	Et advarsel vises: "Det nuværende dokument er ikke gemt. Vil du gemme?"		
3.	Brugeren har 3 valgmuligheder	Der kan vælges "Ja, Nej og annullere"		
4.	Der vælges "Ja"	Overtages af use case 29		
5.	Der vælges "Nej"	Et nyt tomt dokument åbnes		

3.1.3 Use Case 11: Indlæse/køre systemet DSL-filer direkte**Test case: normalforløb**

Forberedelse: Styresystemet skal være startet op.

Beskrivelse: Brugeren får læst sin valgte DSL fil, som robotten eller simulatoren kører.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren vælger en DSL fil, som skal indlæses.	DSL filen indlæses og indholdet fremkommer i editoren.		
2.	Robotten/simulatoren kører den indlæste fil.	Robotten/simulatoren vil udføre funktionaliteter efter DSL filens indhold.		

Test case: Afvigelses forløb 1

Forberedelse: Styresystemet skal være startet op.
 Beskrivelse: Brugeren får læst sin valgte DSL fil, som robotten eller simulatoren kører.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren vælger en filtype, der er ugyldig for styresystemet	Der popper en beskedboks op, som skriver, at der er indlæst en ugyldig filtype.		

3.2.2 Use Case 13: Simulere koden.**Test case: Normalforløb.**

Forberedelse: Styresystemet er oppe at køre og der er noget kode klar til at blive kørt.

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.	Brugeren	Interaktionen bliver		

	interagerer med brugergrænsefladen.	indikeret med besked i brugergrænsefladen.		
3.	Koden bliver kørt i simulatoren.	Simulatoren begynder at vise/skrive hvad der sker.		

Test case: Afvigelsesforløb 1.1.

Forberedelse: Styresystemet er oppe at køre og der er noget 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.	Koden bliver indlæst til editoren.	Koden bliver vist i editoren.		

Test case: Afvigelsesforløb 3.1.

Forberedelse: Styresystemet er oppe at køre og der er noget kode klar til at blive kørt(Enten i editoren eller på harddisken). Koden indeholder dog syntaks fejl.

Beskrivelse: DSL-koden prøver at blive kørt i simulatoren, men fejler på grund af ugyldig kode.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Kørsel af koden bliver ikke udført.	Simulatoren viser/skriver ikke noget.		
2.	Der vil blive givet besked om at koden ikke kunne køres.	En besked vil blive vist i brugergrænsefladen om fejlen.		

3.2 Test af øvrige specifikke krav

Krav 1: Responstiden mellem brugeren og brugergrænsefladen skal være mindre 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 med brugergrænsefladen.	Brugergrænsefladen viser at den har registreret det.(For eksempel 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.		

3.3 Test af grænseflader

Tekst.

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