

---

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

**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>	27
<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
3.1.4 Use Case 4: Manuelt styre.....	13
3.1.4 Use case 9: Have GUI.....	15
3.1.3 Use Case 11: Indlæse/køre systemet DSL-filer direkte.....	15
3.2.1 Use Case 12: Se robot/simulator position.....	16
3.2.2 Use Case 13: Simulere koden.....	17
3.2.4 Use Case 15: Have grafisk visning.....	18
3.3.1 Use Case 17: Eksekvere DSL kode. ....	20
3.3.3 Use Case 19: Se DSL manual.....	21
3.3.9 Use Case 25: Åbne/lukke flere filer i flere tabs.....	22
3.3.10 Use Case 26: Gemme filer.....	23
3.3.21 Use case 37: Have DSL arbejdsområde.....	24
3.2 Test af øvrige specifikke krav.....	25
3.2.1 Test case: Krav 1:.....	25
3.2.2 Test case: Krav 2:.....	26
4. Godkendelse.....	27

# 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

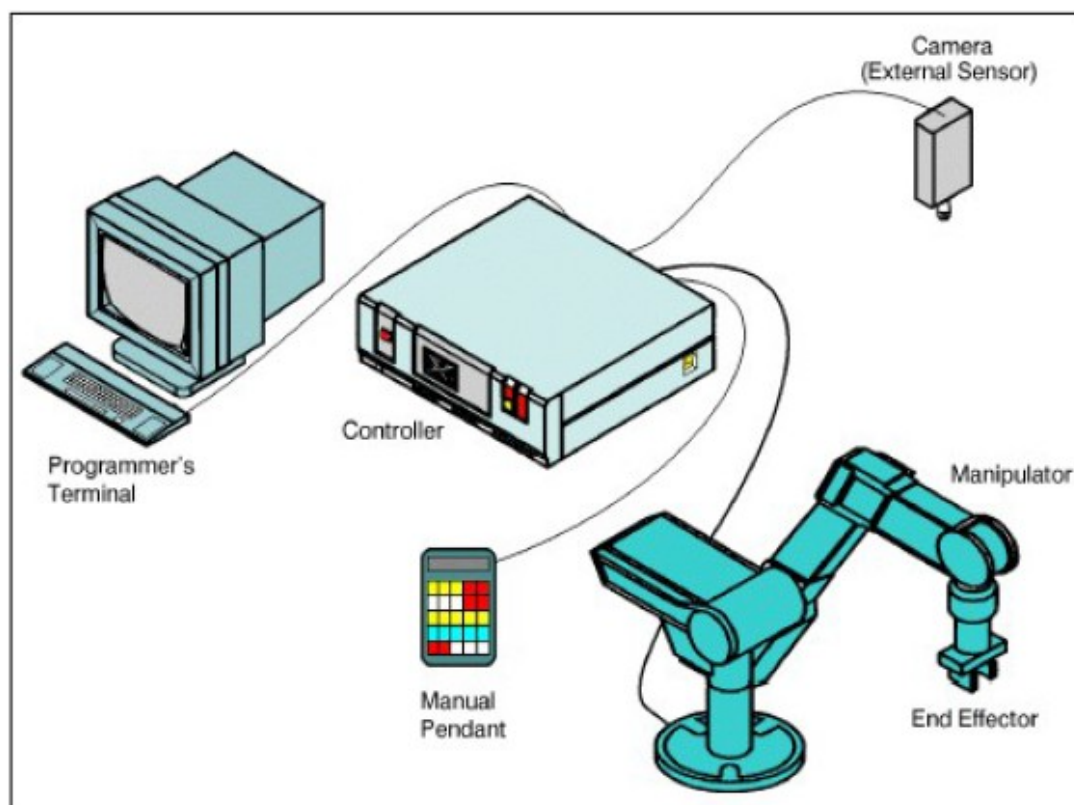
Software der skal tests:

Software	Version	Udgivelses dato	Bemærkning
SIR	1.0		
IDE	1.0		
Simulation	1.0		

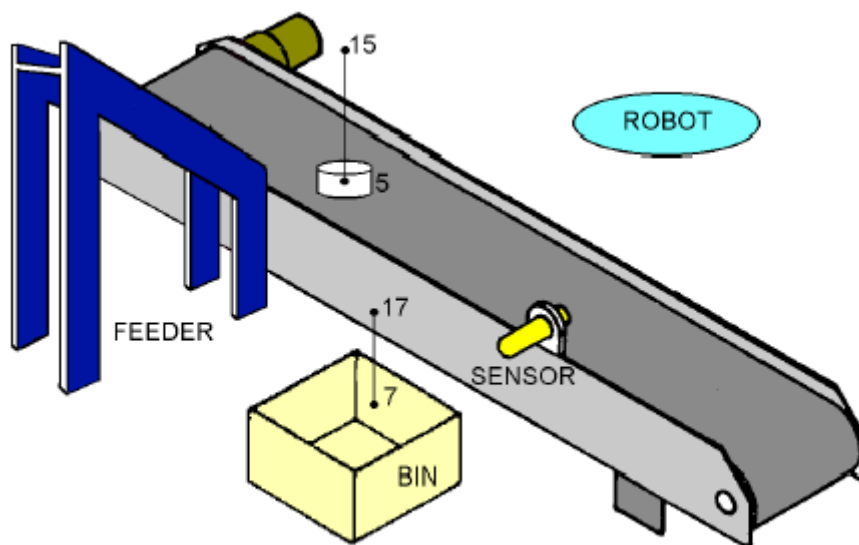
Hardware der skal testes:

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

### 2.2 Testsystemet



Figur 1: Visuel illustration af forbindelser i systemet.



*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 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

##### 3.1.1 Use Case 1: Starte/stoppe systemet.

###### Test case: normalforløb

Forberedelse: Computeren er tændt.  
Beskrivelse: Systemet er startet med visuel visning.  
Systemet er stoppet, og brugergrænsefladen lukkes.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren tænder for systemet.	Systemet startes op og vises visuelt.		
2.	Brugeren stopper systemet.	Systemet stoppes og lukkes ned.		

**3.1.2 Use Case 2: Styre klodsplacering.****Test case: normalforløb**

Forberedelse: Styresystem skal være startet op, samt skal der være tilgængelige klodser til stede.

Beskrivelse: Brugeren styrer klodsplacering.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren interagerer med brugergrænsefladen.	Brugergrænsefladen responderer.		
2.	Robotten tager en rød klods og tager målinger af denne.	Robotarmen tager en rød klods, måler dens dimensioner og dens masse.		
3.	Brugeren vælger placering af klodsen.	Robotarmen bevæger sig mod brugerens ønskede placering af klodsen og placerer klodsen med en usikkerhed på $\pm 1$ cm.		

**Test case: Afvigelses forløb 4.1**

Forberedelse: Styresystem skal være startet op, samt skal der være tilgængelige klodser til stede.

Beskrivelse: Brugeren styrer klodsplacering, men er ikke præcis.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Klodsens placeres ikke på det korrekte sted.	Der udskrives en fejl, at denne ikke er placeret korrekt.		

**Test case: Afvigelses forløb 4.2**

Forberedelse: Styresystemet er oppe at køre, og computeren er forbundet med SCORBOT systemet, samt er manuel styringsdialogen oppe at køre.  
(En af akserne har bevæget sig maksimalt i én retning.)

Beskrivelse: Robotten kan ikke bevæge sig yderligere, hvis dens akse har nået den maksimale rotation.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren trykker på brugergræns efladen eller på en tastatur genvej. (I en retning som der ikke kan bevæges sig mere i.)	Robotten vil ikke bevæge sig i retningen.		

**3.1.4 Use Case 4: Manuelt styre.****Test case: normalforløb**

Forberedelse: Styresystemet er oppe at køre, og computeren er forbundet med SCORBOT systemet.

Beskrivelse: Der startes manuel styring.

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

**Test case: Afvigelses forløb 3.1**

Forberedelse: Styresystemet er oppe at køre, og computeren er forbundet med SCORBOT systemet.  
(Strømmen for USB-Controlleren er ikke tændt)

Beskrivelse: (Robotten og ikke Simulatoren benyttes.)  
Problemer med at forbinde til SCORBOTTEN.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren interagerer med main programmet.	Brugergrænseflade for manuel styring vil blive vist efterfulgt af en besked om fejlen.		

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
2.	Brugeren lukker beskeden.	Brugergrænseflade for manuel styring vil lukke ned. Brugeren ser main programmet for styresystemet.		

**Test case: Afvigelses forløb 4.1**

Forberedelse: Styresystemet er oppe at køre, og computeren er forbundet med SCORBOT systemet, samt er manuel brugergrænseflade oppe at køre.

(En af akserne har nået sin rotationsgrænse.)

(Robotten og ikke Simulatoren benyttes.)

Beskrivelse: Problemer med at forbinde til SCORBOTTEN.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren interagerer med brugergrænsefladen for manuel styring eller med en genvej. (Med en akse som har nået sin rotationsgrænse.)	Robotten vil ikke bevæge sig i retningen.		

**3.1.4 Use case 9: Have GUI****Test case: normalforløb**

Forberedelse: Computeren er tændt.  
 Beskrivelse: Brugeren starter programmet op, som efterfølgende logger ind og har adgang til programmets funktioner.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren starter programmet op.	Et programvindue åbnes. Hvor brugeren kan indtaste brugernavn og kode.		
2.	Brugeren indtaster brugernavn og kodeord og trykker OK.	Brugeren er logget ind og valgmulighederne kan nu ses.		

**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 meldes en fejl i brugergrænsefladen.		

**3.2.1 Use Case 12: Se robot/simulator position.****Test case: normalforløb**

Forberedelse: Styresystemet er oppe at køre, og computeren er forbundet med SCORBOT systemet.  
 Beskrivelse: Der startes enten manuel- eller automatisk styring, hvorefter brugeren kan se robotens/simulatoren position.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren interagerer med brugergrænsefladen.	En dialog vil åbne op, hvor der vises muligheder for manuel- styring eller automatisk styring.		
2.	Brugeren vælger styringstype og kører robotten/simulatoren.	Robotten vil bevæge sig alt efter brugerens input, hvorefter positionen vises i brugergrænsefladen.		



**Test case: Afvigelses forløb 2.1**

Forberedelse: Styresystemet er oppe at køre, og computeren er forbundet med SCORBOT systemet.

(Strømmen for USB-Controlleren er ikke tændt)

Beskrivelse: Problemer med at forbinde til SCORBOTTEN.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren interagerer med brugergræns efladen.	En dialog vil åbne op, hvor der vises muligheder for manuel styring efterfulgt af en besked om fejl.		
2.	Brugeren 'accepterer' fejlen og vil blive taget tilbage til main programmet.	Brugeren ser main programmet for styresystemet.		

**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 interagerer med brugergrænsefladen.	Interaktionen bliver 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 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.	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.	Der gives besked, om at koden ikke kunne køres.		

**3.2.4 Use Case 15: Have grafisk visning.****Test case: normalforløb**

Forberedelse: Brugergænsefladen og main programmet er oppe og køre, samt befinder brugeren sig i simulator-fanen.

Beskrivelse: Brugeren vil få en grafisk visning af robotten i simulatoren.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren interagerer med brugergrens efladen.	Brugergænsefladen viser simulator-vinduet og giver brugeren mulighed for at		
2.	Brugeren kører en kommando fra en åbnet fil, således simulationen igangsættes.	Robotten vil bevæge sig alt efter brugerens fil, samt opdateres tiden og positionen i brugergænsefladen.		

### Test case: Afvigelses forløb 2.1

Forberedelse: Brugergænsefladen og main programmet er oppe og køre, samt befinder brugeren sig i simulator-fanen.

Beskrivelse: Brugeren får lavet en ny fil indeholdende en kommando, som laves i IDE-fanen, men køres i simulator-fanen, da filen ikke eksisterer.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren går til IDE-fanen og skriver en kommando i editoren.	Der forekommer tekst i editoren.		

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
2.	Brugeren gemmer filen med navnet "SimulatorTest".	Der forekommer et vindue med at gemme filen, hvorved bruger gemmer filen under filnavnet.		
3.	Brugeren går tilbage til simulator-fanen og åbner filen "SimulatorTest"	Filens indhold bliver indlæst i simulatoren.		
4.	Brugeren kører filen, således simulationen igangsættes.	Robotten i simulatoren vil bevæge sig, og brugeren kan følge med i, hvad den gør.		

### 3.3.1 Use Case 17: Eksekvere DSL kode.

#### Test case: normalforløb

Forberedelse: Styresystemet er oppe og køre, og brugeren befinder sig i IDE-fanen.

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 og kører denne.	Koden kører.		

**Test case: Afvigelses forløb 1.1**

Forberedelse: Styresystemet er oppe og køre, og brugeren befinder sig i IDE-fanen, samt skal der eksistere en fil med 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 DSL kode.	Filen åbner med succes, og indholdet fremvises i editoren.		

**Test case: Afvigelses forløb 3.1**

Forberedelse: Styresystemet er oppe og køre, og brugeren befinder sig i IDE-fanen.

Beskrivelse: Koden kunne ikke køre.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	GUI'en informerer om fejl.	Fejlen vil blive fremvist.		
2.	Brugeren kan tjekke kodens tilstand.	Brugeren kan vælge forskellige variable og se deres nuværende værdi.		

**3.3.3 Use Case 19: Se DSL manual.****Test case: normalforløb**

Forberedelse: IDE editoren er åbent via SIR programmet.

Beskrivelse: Der åbnes en webbrowser, hvor kommandoerne for brugeren vises.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren vælger "Help"	Et dropdown menu ses ved klik af "Help".		
2.	Brugeren vælger "View commands"	Der vises et webbrowser dokument med kommandoer.		

### 3.3.9 Use Case 25: Åbne/lukke flere filer i flere tabs.

#### Test case: normalforløb

Forberedelse: Styresystemet skal være startet op.  
 Beskrivelse: Brugeren får åbnet en flere filer. Brugeren får lukket en fil.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren vælger "Open file".	Man vil kunne se filer, der kan åbnes.		
2.	Brugeren vælger en eller flere filer.	Filen eller filerne bliver åbnet.		
3.	Brugeren vælger en fil, han vil lukke og vælger "Close file"	Filen bliver lukket.		

#### Test case: Afbvlgelses forløb 2.1

Forberedelse: Styresystemet skal være startet op.  
 Beskrivelse: Brugeren har ingen gemte filer.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren vælger "New file".	En ny fil vises i en ny fane.		

**Test case: Afvigelse forløb 2.2**

Forberedelse: Styresystemet skal være startet op.  
 Beskrivelse: Brugeren bruger en genvejstast.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Bruger bruger en genvejstast for at åbne ny tab	En ny blank tab vil blive vist.		

**3.3.10 Use Case 26: Gemme filer****Test case: normalforløb**

Forberedelse: Styresystemet er oppe og køre, og brugeren befinder sig i IDE-fanen, samt der er skrevet noget tekst/kode i editoren.  
 Beskrivelse: Brugeren får gemt teksten/koden.

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
1.	Brugeren interagerer med brugergrænsefladen og åbner for IDE-fanen.	IDE-fanen fremkommer og brugeren får mulighed for at skrive kode i editoren.		
2.	Brugeren skriver kode i editoren.	Der fremkommer i editoren.		

TRIN	Aktion/Input	Forventet resultat	CHK	Kommentarer
3.	Brugeren gemmer filen med filnavnet "PER.txt" ved brug af menuen.	Filen gemmes og vinduet returneres til editor vinduet.		

**Test case: Afvigelses forløb 3.1**

Forberedelse: Styresystemet er oppe og køre, og 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.txt" og trykker gem.	Filen gemmes og vinduet returneres til editor vinduet.		

**3.3.21 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 "File"	Et dropdown menu ses.		
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.		

### 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.		

## 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