Titel: Beschreibung

Thema:

Key Words: Use Case, IPA, TsNet, Test, Open Loop

Revision:

Änderungsdatum: 2015-10-03

Dokument Status:

Autor: Dominik Zgraggen

Abteilung:

Verantwortliche Stelle:

Firma:

Basierend auf Vorlage:

# Use Cases

|  |  |
| --- | --- |
| **Neues Test-Projekt eröffnen (UC1)** | |
| Kurzbeschreibung: | Im Projekt werden die Dateien gesammelt. Der Namen des Projekts ist im Unterordner der Dateistruktur wieder zu finden. Mit der Eingabe einer Bezeichnung und dem Betätigen eines Buttons wird das Projekt eröffnet. |
| Akteur(e): | Applikationsentwickler |
| Auslöser: | Klick auf Button „neues Projekt anlegen“ |
| Vorbedingung: | Name ist eingegeben |
| Ergebnis, Nachbedingung: | Ordner wurden mit korrekten Namen angelegt. Das Programm weiss nun, in welchem Ordner der Test ausgeführt werden soll.  Falls der Ordner schon existiert, wird eine Zahl angehängt. |
| Benutzte Use Cases: | - |

|  |  |
| --- | --- |
| **Bestehendes Test-Projekt öffnen (UC2)** | |
| Kurzbeschreibung: | Der Projektordner wurde früher einmal angelegt und kann wieder verwendet werden. |
| Akteur(e): | Applikationsentwickler |
| Auslöser: | Klick auf Button, welcher Windows-Verzeichnis anwählen lässt |
| Vorbedingung: |  |
| Ergebnis, Nachbedingung: | Ordner lässt sich nur anwählen, wenn er die Richtige Dateistruktur beinhaltet. |
| Benutzte Use Cases: | UC2a |

|  |  |
| --- | --- |
| **Vorhandene Dateien laden (UC2a)** | |
| Kurzbeschreibung: | Bei einem bestehenden Test-Projekt kann es sein, dass in der Ordnerstruktur abegelegte Dateien verwendet werden können. Alle diese sollten im Programm angezeigt werden. |
| Akteur(e): | Applikationsentwickler |
| Auslöser: | Anwählen vom bestehenden Test-Projekt (UC2) |
| Vorbedingung: | Test-Projekt ist korrekt vorhanden |
| Ergebnis, Nachbedingung: | Vorhandene Dateien sind dem Programm bekannt und werden geladen. |
| Benutzte Use Cases: | UC2 |

|  |  |
| --- | --- |
| **Test-Projekt schliessen (UC3)** | |
| Kurzbeschreibung: | Durch schliessen des GUIs gehen keine Daten verloren, denn das Projekt ist jederzeit gespeichert und lässt sich wieder öffnen. |
| Akteur(e): | Applikationsentwickler |
| Auslöser: | Schliessen des Programms |
| Vorbedingung: | - |
| Ergebnis, Nachbedingung: | Programm ist geschlossen, Dateien unverändert. |
| Benutzte Use Cases: | - |

|  |  |
| --- | --- |
| **Applikations-Import (UC4)** | |
| Kurzbeschreibung: | Der ABT-Export liegt als Zip-Datei vor und kann Importiert werden. Das Ziel ist das Applikations-Model, welches nachher im Control-Model weiterverwendet wird.  Der Import ist nur notwendig, wenn kein bestehendes Control-Model vorhanden ist. |
| Akteur(e): | Applikationsentwickler |
| Auslöser: | Button „Import starten“ wird gedrückt. |
| Vorbedingung: | Die Zip-Datei ist angewählt (UC4a) und das Test-Projekt definiert. |
| Ergebnis, Nachbedingung: | Vollständiges Applikations-Model ist in Ordner korrekt abgelegt. |
| Benutzte Use Cases: | UC4a, UC4b |

|  |  |
| --- | --- |
| **Applikation (Zip) anwählen (UC4a)** | |
| Kurzbeschreibung: | Der ABT-Export muss vor dem Import im Windows angewählt werden. |
| Akteur(e): | Applikationsentwickler |
| Auslöser: | Button „Applikation auswählen“ wird gedrückt. |
| Vorbedingung: | Zip-Datei / ABT-Export ist vorhanden. |
| Ergebnis, Nachbedingung: | Das Zip wird in die Ordnerstruktur kopiert. Das Programm kennt das Zip. |
| Benutzte Use Cases: | UC4 |

|  |  |
| --- | --- |
| **Control-Model generieren (UC4b)** | |
| Kurzbeschreibung: | Nach dem Import steht das Applikations-Model zur Verfügung, welches nun im Control-Model integriert wird. |
| Akteur(e): | Applikationsentwickler |
| Auslöser: | Import der Applikation (ZIP) ist abgeschlossen (UC4) |
| Vorbedingung: | Import ist korrekt abgelaufen |
| Ergebnis, Nachbedingung: | Control-Model ist korrekt generiert und in Ordnerstruktur abgelegt. |
| Benutzte Use Cases: | UC4 |

|  |  |
| --- | --- |
| **Bestehendes Control-Model auswählen (UC5)** | |
| Kurzbeschreibung: | Wenn bereits ein Control-Model vorhanden ist, kann ohne ein Import durchzuführen, mit jenem Control-Model der Test durchgeführt werden. |
| Akteur(e): | Applikationsentwickler |
| Auslöser: | Auswahl durch Applikationsentwickler |
| Vorbedingung: | Control-Model ist (von früher) vorhanden |
| Ergebnis, Nachbedingung: | Das Zip wird in die Ordnerstruktur kopiert. Das Programm kennt das Zip. |
| Benutzte Use Cases: | UC6, UC4b |

|  |  |
| --- | --- |
| **Control-Model öffnen (UC6)** | |
| Kurzbeschreibung: | Das Control-Model kann im Simulink betrachtet werden. |
| Akteur(e): | Applikationsentwickler |
| Auslöser: | Klick auf Button „Control-Model öffnen“ |
| Vorbedingung: | Control-Model generiert (UC4b) oder Control-Model (von früher) ausgewählt (UC5), sonst ist der Button nicht verfügbar |
| Ergebnis, Nachbedingung: | Das Control-Model wird korrekt dargestellt und kann auch bearbeitet werden. |
| Benutzte Use Cases: | UC5, UC4b |

|  |  |
| --- | --- |
| **TsNet-Datei auswählen (UC7)** | |
| Kurzbeschreibung: | Im TsNet ist der Test spezifiziert. Durch die Anwahl der Excel-Dateien, weiss die Simulation, was und wie getestet wird. |
| Akteur(e): | Applikationsentwickler |
| Auslöser: | Klick auf Button „TsNet-Datei auswählen“ |
| Vorbedingung: | TsNet (Excel) -Datei ist vorhanden |
| Ergebnis, Nachbedingung: | Die TsNet-Datei wird in die Ordnerstruktur kopiert. Das Programm kennt nun das die Datei. |
| Benutzte Use Cases: | UC7a, UC7b |

|  |  |
| --- | --- |
| **Ausgewählte TsNet-Datei öffnen (UC7a)** | |
| Kurzbeschreibung: | DieTsNet-Datei kann im Excel betrachtet werden. |
| Akteur(e): | Applikationsentwickler |
| Auslöser: | Klick auf Button „TsNet-Datei öffnen“ |
| Vorbedingung: | TsNet-Datei ist ausgewählt (UC7) und befindet sich in Ordnerstruktur. |
| Ergebnis, Nachbedingung: | Die TsNet-Datei ist in Excel geöffnet und kann bearbeitet werden. |
| Benutzte Use Cases: | UC7 |

|  |  |
| --- | --- |
| **TsNet-Sheet auswählen (UC7b)** | |
| Kurzbeschreibung: | Das Programm muss wissen, welches Tabellenblatt die passenden Testinformationen enthält. |
| Akteur(e): | Applikationsentwickler |
| Auslöser: | Auswahl des Blattes in der Liste |
| Vorbedingung: | Die TsNet-Datei ist ausgewählt (UC7). |
| Ergebnis, Nachbedingung: | Dem Programm ist bekannt, welches Tabellenblatt ausgewählt worden ist. |
| Benutzte Use Cases: | UC7 |

|  |  |
| --- | --- |
| **Simulation Starten (UC8)** | |
| Kurzbeschreibung: | Die Simulation benötigt ein Control-Model und eine TsNet-Datei. Nach dem Fertigstellen der Simulation wird ein Report erstellt. |
| Akteur(e): | Applikationsentwickler |
| Auslöser: | Klick auf Button „Simulation starten“ |
| Vorbedingung: | Control-Model und TsNet-Datei sind (UC4 oder 5 und UC7) sind im Test-Projekt abgelegt. |
| Ergebnis, Nachbedingung: | Report ist vollständig und in Ordnerstruktur abgelegt. |
| Benutzte Use Cases: | - |

--

|  |  |
| --- | --- |
| **Hilfeinformationen anschauen (UC9)** | |
| Kurzbeschreibung: | Es kann jeder Zeit Hilfe angefordert werden. Es wird dann einen Text gezeigt, der die Eingaben usw. erklärt. |
| Akteur(e): | Applikationsentwickler |
| Auslöser: | Klick auf Button „Hilfe“ |
| Vorbedingung: | - |
| Ergebnis, Nachbedingung: | Der Applikationsentwickler sieht die Hilfe und die Unklarheit wird beseitigt. |
| Benutzte Use Cases: | - |