Use Case: AssemblyLogin

Actors:
1. Assembler

Pre-Conditions:
1. The Assembler is not already logged in.

Primary Flow:
1. The use case starts when an Assembler visits the Assembly Page.
2. The Assembler is presented with a login dialog, where he is requested to enter his username and password.
3. The Assembler enters and submits his credentials.
4. The system redirects the Assembler to the Assembly Dashboard.
5. Scenarios:
6. AssemblyLoginInvalidCredentials
7. AssemblyLoginMissingUsername
8. AssemblyLoginMissingPassword

Post-Conditions:
1. The Assembler is now logged in.

Use Case: AssemblyLogin
Secondary Scenario: AssemblyLoginInvalidCredentials

Actors:
1. Assembler

Primary Flow:
1. The use case begins in step 3 of the AssemblyLogin use case, when the Assembler provides invalid credentials.
2. The system presents the Assembler with an Error Dialog, informing him that the specified credentials were not recognized.
3. The system re-prompts the Assembler for his authentication credentials (Use Case: AssemblyLogin).

Use Case: AssemblyLogin
Secondary Scenario: AssemblyLoginMissingUsername

Actors:
1. Assembler

Primary Flow:
1. The use case begins in step 3 of the AssemblyLogin use case, when the Assembler does not provide a username.
2. The system presents the Assembler with an Error Notification, informing him that he did not enter his username.
3. The system re-prompts the Assembler for his authentication credentials (Use Case: AssemblyLogin).

Use Case: AssemblyLogin
Secondary Scenario: AssemblyLoginMissingPassword

Actors:
1. Assembler

Primary Flow:
1. The use case begins in step 3 of the AssemblyLogin use case, when the Assembler does not provide a password.
2. The system presents the Assembler with an Error Notification, informing him that he did not enter his password.
3. The system re-prompts the Assembler for his authentication credentials (Use Case: AssemblyLogin).

Use Case: AssemblyDashboard

Actors:
1. Assembler

Pre-Conditions:
1. The Assembler is Logged in.
2. The Assembler is on the Assembly Dashboard.

Primary Flow:
1. The use case begins when an Assembler visits the Assembly Dashboard.
2. The system presents the Assembler with a Tabbed View, which consists of the following tabs.
   a. Pending Orders Tab (UC: AssemblyViewOrdersPendingTab).
   b. In-Progress Orders Tab (UC: AssemblyViewOrdersInProgressTab).
3. The system automatically selects and displays the Pending Orders Tab.

Secondary Scenarios:
1. AssemblyViewOrdersPendingTab
2. AssemblyViewOrdersInProgressTab

Secondary Scenarios:
1. At any point, the Assembler may logout by pressing the Logout Button.

Use Case: AssemblyDashboard
Secondary Scenario: AssemblyViewOrdersPendingTab

Actors:
1. Assembler

Primary Flow:
1. The use case begins in step 2 of the AssemblyDashboard use case, when the Assembler selects the Pending Orders Tab.
2. If the system finds any Pending Orders, then
   a. For each Order found
      i. The system displays a row containing information about the Order, such as its Controller Parts, its Submission Date, etc.
      ii. The system displays a Button which allows the Assembler to start the Assembly of the Order (UC: AssemblyViewOrdersStartOrderAssembly).
3. If the system does not find any Pending Orders, then
   a. The system notifies the Assembler that there are no Pending Orders.

Secondary Scenarios:

Use Case: AssemblyViewOrdersPendingTab
Secondary Scenario:
AssemblyViewOrdersStartOrderAssembly

Actors:
1. Assembler

Primary Flow:
1. The use case begins in step 2.a.ii of the AssemblyViewOrdersPendingTab, when the Assembler presses the Start Assembly Button.
2. The system presents the Assembler with a Confirmation Dialog, prompting him to confirm his choice.
3. If the Assembler confirms his choice
   a. The system marks the specified order as In-Progress.
   b. The system redirects the Assembler to the previous page.

Use Case: AssemblyDashboard
Secondary Scenario: AssemblyViewOrdersInProgressTab

1. AssemblyViewOrdersStartOrderAssembly

Notes and Remarks:
1. Pending Orders are Orders that the Assembler has not start Assembling yet.

Actors:
1. Assembler

Primary Flow:
1. The use case begins in step 2 of the AssemblyDashboard use case, when the Assembler selects the In-Progress Orders Tab.
2. If the system finds any In-Progress Orders, then
   a. For each Order found
      i. The system displays a row containing information about the Order, such as its Controller Parts, its Submission Date, Assembly Start Date, etc.
      ii. The system displays a Button which allows the Assembler to end the Assembly of the Order (UC: AssemblyViewOrdersEndOrderAssembly).
3. If the system does not find any In-Progress Orders, then
   a. The system notifies the Assembler that there are no In-Progress Orders.

Secondary Scenarios:
1. AssemblyViewOrdersEndOrderAssembly

Notes and Remarks:
1. Pending Orders are Orders that the Assembler has not start Assembling yet.

Use Case: AssemblyViewOrdersInProgressTab Secondary Scenario: AssemblyViewOrdersEndOrderAssembly

Actors:
1. Assembler

Primary Flow:
1. The use case begins in step 2.a.ii of the AssemblyViewOrdersInProgressTab, when the Assembler presses the End Assembly Button.
2. The system presents the Assembler with a Confirmation Dialog, prompting him to confirm his choice.
3. If the Assembler confirms his choice
   a. The system marks the specified order as Assembled.
   b. The system adds the Order to the Undelivered Orders Tab of the Sales Manager's Order Page.
   c. The system redirects the Assembler to the previous page.