Salesforce Project Implementation Phases with Concepts (Admin + Developer)

Project Title: Manufacturing After-Sales & Service CRM

Phase 10: Quality Assurance Testing

This phase outlines the quality assurance (QA) testing strategy and test cases designed to validate the functionality, reliability, and usability of the Manufacturing After-Sales & Service CRM. The testing approach ensures that all features, from data validation to complex automation, are working as expected before deployment.

Testing Approach

The testing approach for this project combines both automated and manual testing to ensure comprehensive coverage:

- **Automated Testing (Apex Unit Tests):**
    - All Apex classes and triggers are accompanied by dedicated test classes (e.g., WarrantyServiceTest.cls, SLAServiceTest.cls).
    - These tests are written to validate the business logic in the Apex code, including service classes, controllers, and triggers.
    - The tests cover positive and negative scenarios, bulk data processing, and user permissions to ensure the code is robust and scalable.
    - All Apex code meets the minimum 75% code coverage requirement for deployment to production.
- **Manual Testing (User Acceptance Testing - UAT):**
    - Manual test cases are executed by QA testers or end-users to validate the end-to-end functionality of the system from a user's perspective.
    - This includes testing the user interface (UI) of the Lightning Web Components, the behavior of automation rules (Flows, Validation Rules, Approval Processes), and the overall user experience.
    - The test cases below represent the manual testing plan for this project.

Test Cases:

The following test cases are designed to validate the key features of the CRM:

- **Validation Rule: Warranty Expiry:**
  - **Test Steps (with input):**
    - 1. Navigate to the Registered_Product__c object and create a new record.
    - 2. Enter a Purchase_Date__c of today.
    - 3. Enter a Warranty_Expiry__c of yesterday.
    - 4. Click "Save."
  - **Expected Result:** The record fails to save, and an error message "Warranty Expiry must be after Purchase Date" is displayed.
  - **Actual Result (with Screenshot):**

- **Flow: Auto-Assign Engineer**
  - **Test Steps (with input):**
    - 1. Create a new Service_Case__c record.
    - 2. Set the Required_Skill__c to "Electronics."
    - 3. Set the Customer_Location__c to a specific location (e.g., latitude/longitude).
    - 4. Ensure there is an available Engineer__c with the "Electronics" skill and a location close to the customer.
    - 5. Save the Service_Case__c record.
  - **Expected Result:** The Engineer_Assigned__c field on the Service_Case__c record is automatically populated with the name of the correct engineer.

- o **Actual Result (with Screenshot):**

- **Approval Process: Warranty Claim**
  - o **Test Steps (with input):**
    - 1. Create a Registered_Product__c record with a Purchase_Date__c of two years ago.
    - 2. Create a new Service_Case__c and link it to the product created in step 1.
    - 3. Save the Service_Case__c record.
  - o **Expected Result:** The case status is automatically set to "Submitted for Approval," and an approval request is sent to the appropriate manager. The record is locked from editing.
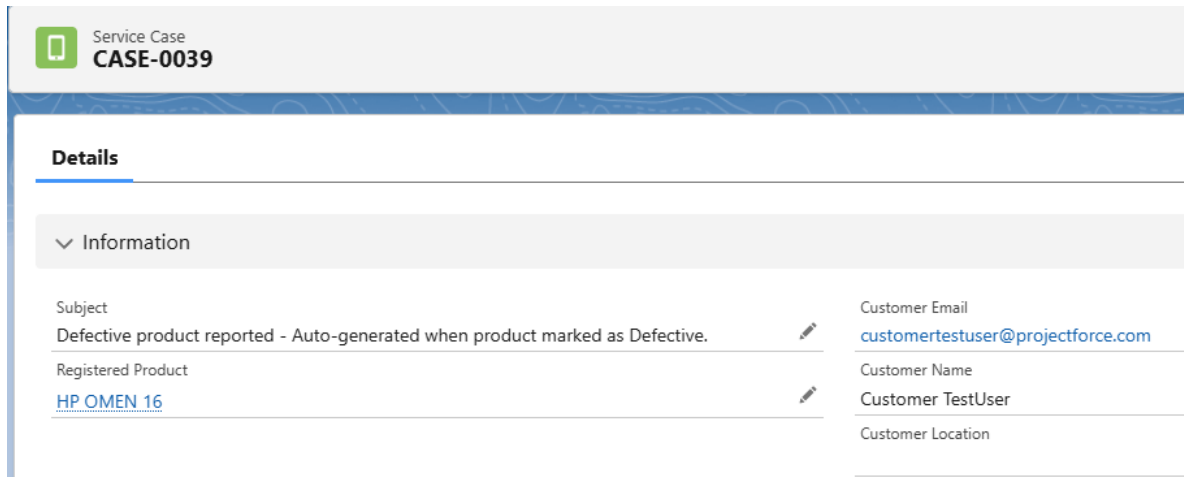 - o **Actual Result (with Screenshot):**

- **Trigger: Defective Product**
  - **Test Steps (with input):**
    - 1. Navigate to an existing Registered_Product__c record.
    - 2. Check the Defective__c checkbox and save the record.
  - **Expected Result:** A new Service_Case__c record is automatically created with a "High" priority and is linked to the defective product.
  - **Actual Result (with Screenshot):**
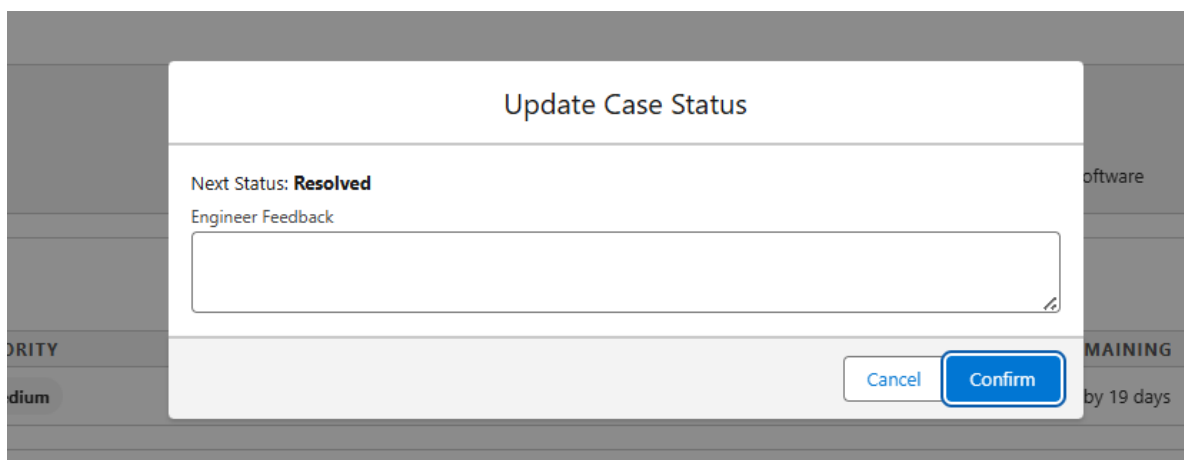
- **LWC: Engineer Dashboard Status Update**
  - **Test Steps (with input):**
    - 1. Log in as a Field Engineer user.
    - 2. Navigate to the Engineer Dashboard.
    - 3. For an assigned job with the status "In Progress," click the "Update Status" button.
    - 4. In the modal, confirm the next status is "Resolved" and click "Confirm."2. Check the Defective__c checkbox and save the record.
  - **Expected Result:** The case status is updated to "Resolved," and the change is reflected in the dashboard and on the Service_Case__c record.
  - **Actual Result (with Screenshot):**

- **Apex: Warranty Service Evaluation**
  - **Test Steps (with input):**
    - 1. Create a Registered_Product__c with a Warranty_Expiry__c in the future and an AMC_Expiry__c in the past.
    - 2. Use the Developer Console to run WarrantyService.evaluate(recordId).
  - **Expected Result:** The WarrantyStatus result from the Apex method returns inWarranty = true and inAmc = false.
  - **Actual Result (with Screenshot):**

Class	Method	Duration	Result	Errors	Stack Trace
TestRun @ 1:24:24 pm					
WarrantyServiceTest	testEvaluateWithIdNullCov...	0:00	Fail	System.AssertException: Assertion Failed: ...	Class.WarrantyServiceTest.testEvaluateWithIdNullCoverage: line 140, column 1
WarrantyServiceTest	testEvaluateWithNullId	0:00	Fail	System.AssertException: Assertion Failed: ...	Class.WarrantyServiceTest.testEvaluateWithNullId: line 172, column 1
WarrantyServiceTest	testEvaluateWithNullRecord	0:00	Fail	System.AssertException: Assertion Failed: ...	Class.WarrantyServiceTest.testEvaluateWithNullRecord: line 206, column 1
WarrantyServiceTest	testWarrantyStatusWrapper	0:00	Fail	System.AssertException: Assertion Failed: ...	Class.WarrantyServiceTest.testWarrantyStatusWrapper: line 220, column 1
WarrantyServiceTest	testEvaluateWithIdActive...	0:00	Comp...		
WarrantyServiceTest	testEvaluateWithIdBothEx...	0:00	Comp...		
WarrantyServiceTest	testEvaluateWithIdExpired...	0:00	Comp...		
WarrantyServiceTest	testEvaluateWithIdWarran...	0:00	Comp...		
WarrantyServiceTest	testEvaluateWithRecordAc...	0:00	Comp...		
WarrantyServiceTest	testInvalidProductId	0:00	Comp...		
WarrantyServiceTest	testWarrantyExpiringToday	0:00	Comp...		

Logs	Tests	Checkpoints	Query Editor	View State	Progress	Problems

Status	Test Run	Enqueued Time	Duration
✖	TestRun @ 1:24:24 pm		
✖	WarrantyServiceTest		
✖	testEvaluateWithIdNullCoverage		0:00
✖	testEvaluateWithNullId		0:00
✖	testEvaluateWithNullRecord		0:00
✖	testWarrantyStatusWrapper		0:00
✔	testEvaluateWithIdActiveWarrantyAndAMC		0:00
✔	testEvaluateWithIdBothExpired		0:00
✔	testEvaluateWithIdExpiredWarrantyActiveAMC		0:00
✔	testEvaluateWithIdWarrantyOnly		0:00
✔	testEvaluateWithRecordActiveCoverage		0:00
✔	testInvalidProductId		0:00
✔	testWarrantyExpiringToday		0:00