PROJECT REPORT

Salesforce Project Implementation Phases with Concepts (Admin + Developer)

Project Title: Manufacturing After-Sales & Service CRM

Name: Puneet Verma

Email Id: verma0404puneet@gmail.com

GitHubID: PuneetVerma04

Project Repository: Manufacturing After-Sales & Service CRM

Project Overview

The Manufacturing After-Sales & Service CRM is a Salesforce-based solution designed to streamline after-sales services, warranty management, and customer support for manufacturing businesses. This project leverages Salesforce's powerful platform to manage service cases, dispatch engineers, and collect customer feedback efficiently. Key features include service case management, automated engineer dispatch, warranty and AMC tracking, customer feedback collection, and custom reporting for SLA compliance and engineer performance.

Objectives

The main goals of building this CRM are to centralize product registration and warranty/AMC tracking, automate service case creation and engineer assignment, provide real-time visibility for all stakeholders, and enable customer self-service. These objectives are directly linked to improving operational efficiency, reducing costs, and increasing customer satisfaction and retention.

Phase 1: Problem Understanding & Industry Analysis

Requirement Gathering

Manufacturing companies face growing challenges in their after-sales service operations. Customers often encounter delayed responses, a lack of visibility into service requests, and frequent warranty or AMC disputes due to manual and error-prone tracking. This leads to customer dissatisfaction, increased operational costs, and SLA breaches.

Stakeholder Analysis

Stakeholder	Role	Needs
Customers	Register products, raise service requests, track warranties/AMCs	Quick service resolution, warranty clarity, real-time request visibility
Service Agents	Manage cases, assign engineers, handle warranty disputes	Automated case assignment, SLA tracking, reduced manual effort
Field Engineers	Perform on-site repairs and maintenance	Clear job assignments, optimized scheduling, ability to update job status

Stakeholder	Role	Needs
Product Managers	Track product failures, analyze service data	Reports on product defects, recurring issues, and spare parts demand
Management	Oversee SLA compliance, service costs, customer satisfaction	Performance dashboards, SLA breach alerts, operational insights

Business Process Mapping

The after-sales service lifecycle consists of the following steps:

1. Customer purchases a product and registers it.
2. Warranty/AMC details are validated.
3. Customer raises a service request.
4. The case is auto-assigned to a service agent.
5. The service agent dispatches the nearest skilled engineer.
6. The engineer completes the service and updates the case to closure.
7. Customer feedback is collected, and reports are generated for management.

Industry-specific Use Case Analysis

- **Warranty & AMC Lifecycle Management**: Automating warranty validation and AMC tracking to reduce disputes.
- **Case Management with SLAs**: Ensuring timely response and resolution to customer requests.
- **Field Engineer Scheduling & Tracking**: Assigning engineers based on skill and location to improve efficiency.
- **Predictive Maintenance (IoT Integration)**: Anticipating failures before they occur to provide proactive service.
- **Customer Satisfaction Tracking**: Collecting feedback for quality improvements and to identify areas for improvement.

Phase 2: Org Setup & Configuration

Salesforce Edition

The project is built on the **Salesforce Developer Edition**.

Company Profile Setup

The company profile for "Satya Enterprises" was configured with the following settings:

- **Default Locale**: English (India)
- **Default Time Zone**: GMT+05:30 (India Standard Time)
- **Currency**: INR

Business Hours & Holidays

Business hours are set from **9:00 AM to 6:00 PM, Monday through Saturday**, which is used for SLA monitoring.

User Setup & Licenses

The following user accounts were created with the appropriate licenses and profiles:

- **Manager User**: Salesforce License, Manager Profile
- **Service Agent User**: Salesforce Platform License, Service Agent Profile
- **Field Engineer User**: Salesforce Platform License, Field Engineer Profile
- **Customer User**: Customer Community Login License, Customer Profile

Phase 3: Data Modeling & Relationships

Standard & Custom Objects

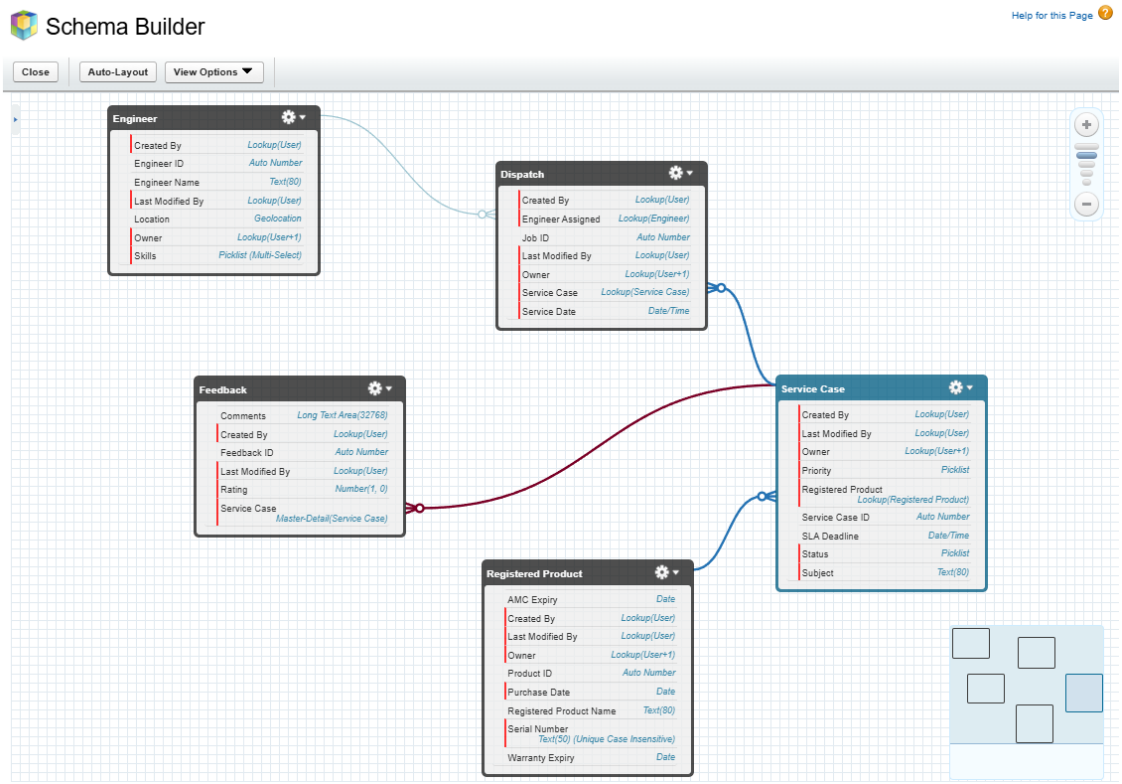
The data model includes the following custom objects to support the after-sales service process:

- **Registered Product (Registered_Product__c)**: Tracks products registered by customers for warranty and service.
- **Service Case (Service_Case__c)**: Manages customer service cases.
- **Dispatch (Dispatch__c)**: Handles engineer dispatch details.
- **Feedback (Feedback__c)**: Collects customer feedback on service cases.
- **Engineer (Engineer__c)**: Represents field engineers with their skills and availability.
- **Service Agent (Service_Agent__c)**: Represents service agents who manage customer cases.
- **Error Log (Error_Log__c)**: A custom object to store application error logs for debugging and monitoring purposes.

Schema Builder and Relationships:

The following relationships are defined between the objects:

- **Registered Product to Service Case**: One-to-Many lookup relationship.
- **Service Case to Dispatch**: One-to-One lookup relationship.
- **Engineer to Dispatch**: One-to-Many lookup relationship.
- **Service Case to Feedback**: One-to-One Master-Detail relationship.



Phase 4: Process Automation (Admin)

Validation Rules

- **Warranty_Expiry_after_Purchase**: Ensures that the warranty expiry date on a Registered_Product__c record is after the purchase date.
- **SLA_Deadline_Check**: Prevents the SLA deadline on a Service_Case__c record from being set in the past.

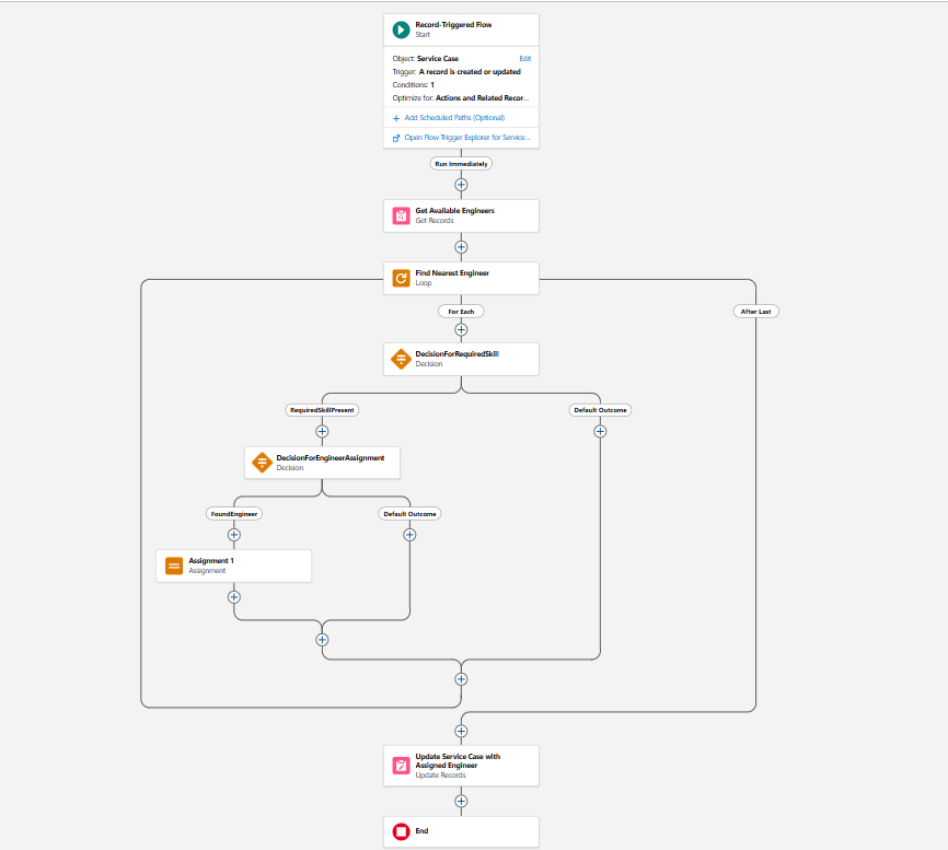
Flows

- **Auto-Assign Engineer Flow**: This flow is triggered when a new Service_Case__c is created. It automatically assigns an available engineer based on the required skills and geographic proximity to the customer.

- **Status Update Flow**: This flow automates the status updates of a Service_Case__c as it progresses through the service lifecycle, from "New" to "Closed".

Approval Processes

An approval process is configured for warranty claims on products that are more than one year old. When a Service_Case__c is created for a product that was purchased over a year ago, the case is automatically submitted for manager approval.

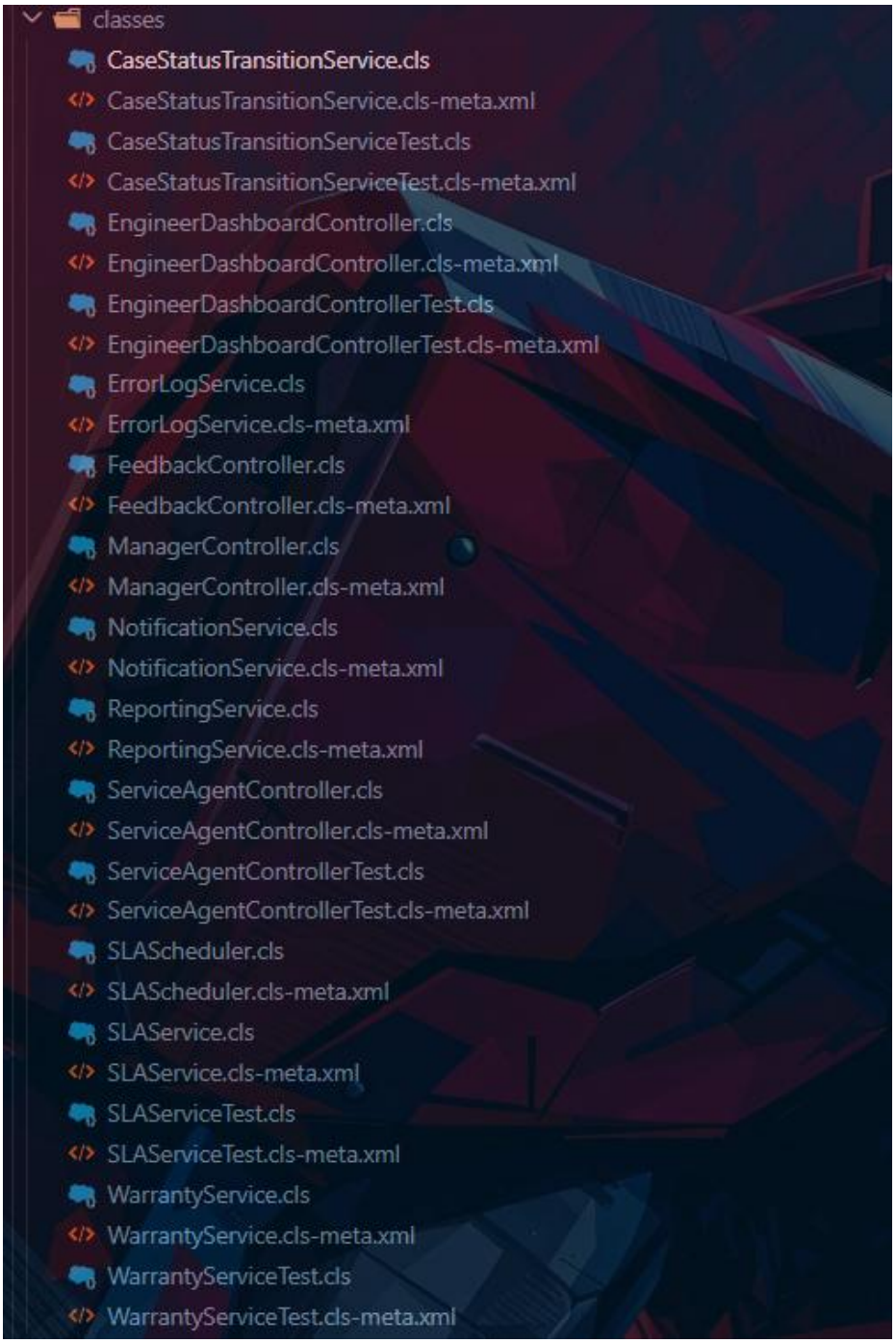


Phase 5: Apex Programming (Developer)

Apex Classes

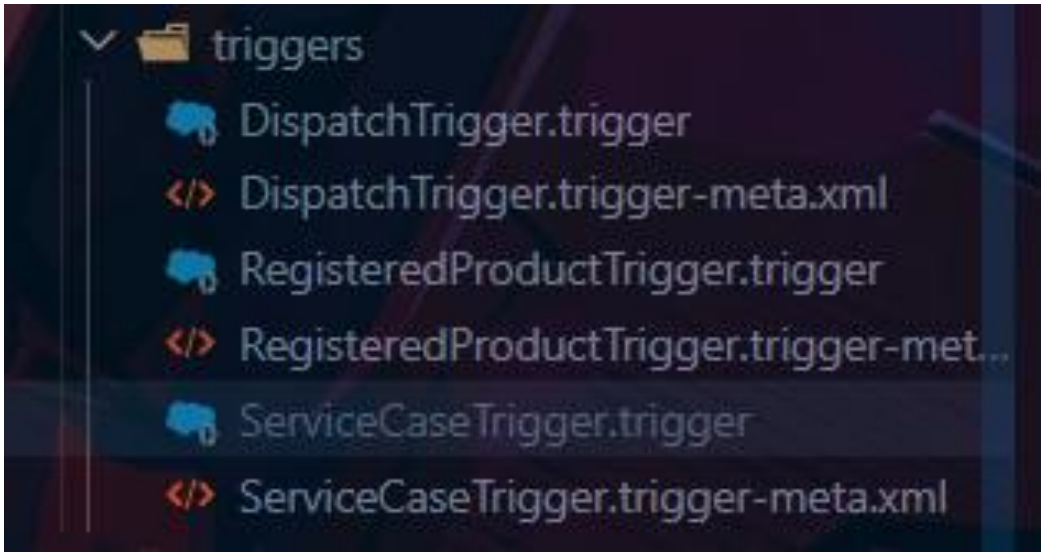
- **CaseStatusTransitionService**: Manages the valid transitions for Service_Case__c statuses using Custom Metadata Types.
- **EngineerDashboardController**: Provides data to the Engineer Dashboard LWC, including assigned jobs and engineer information.
- **ErrorLogService**: A centralized service for logging application errors to the Error_Log__c custom object.
- **FeedbackController**: Handles the logic for submitting customer feedback.
- **ManagerController**: Provides data to the Manager Dashboard LWC, including key statistics and performance metrics.

- **NotificationService**: Sends email notifications to engineers and service agents when they are assigned to a new case or dispatch.
- **ReportingService**: Provides data for custom reports, such as the top 5 products by service requests.
- **SLAService**: Contains logic for finding and notifying stakeholders about breached SLAs.
- **ServiceAgentController**: Provides data to the Service Agent Dashboard LWC, including recent cases and feedback.
- **WarrantyService**: A service class to evaluate the warranty and AMC status of a registered product.



Apex Triggers

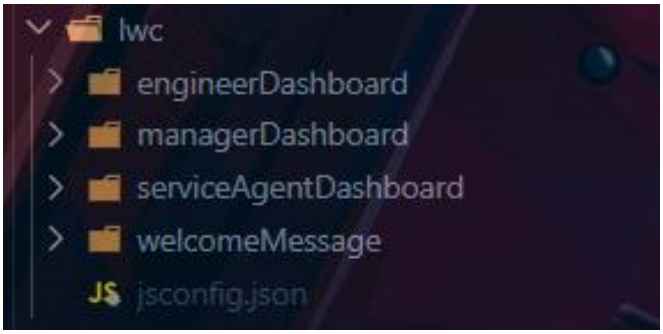
- **DispatchTrigger**: Fires after a Dispatch__c record is inserted or updated to send notifications to the assigned engineer and service agent.
- **RegisteredProductTrigger**: Fires after a Registered_Product__c record is updated. If the Defective__c checkbox is checked, it automatically creates a new Service_Case__c.
- **ServiceCaseTrigger**: Fires before and after a Service_Case__c record is updated. It sets the Closed_Date__c when the status changes to "Closed" and sends notifications when an engineer or service agent is assigned.



Phase 6: User Interface Development

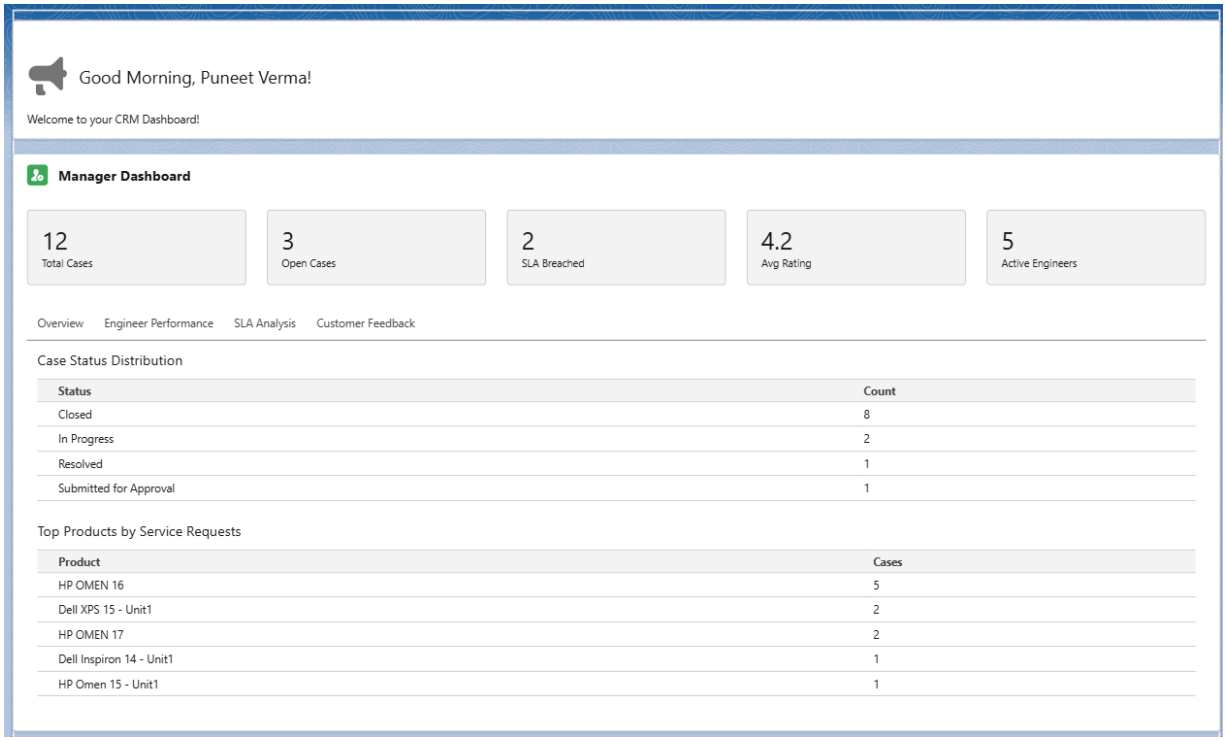
Lightning Web Components (LWCs)

- **engineerDashboard**: A dashboard for field engineers to view their assigned jobs, dispatch details, and update the status of their cases.
- **managerDashboard**: A dashboard for managers to monitor key performance indicators, such as total cases, open cases, SLA breaches, and engineer performance.
- **serviceAgentDashboard**: A dashboard for service agents to view and manage service cases, recent feedback, and product status.
- **welcomeMessage**: A simple component that displays a welcome message to the logged-in user.



Layouts

Custom page layouts have been created for all custom objects to provide a tailored user experience for different user profiles. This includes layouts for Dispatch__c, Engineer__c, Feedback__c, Registered_Product__c, and Service_Case__c.

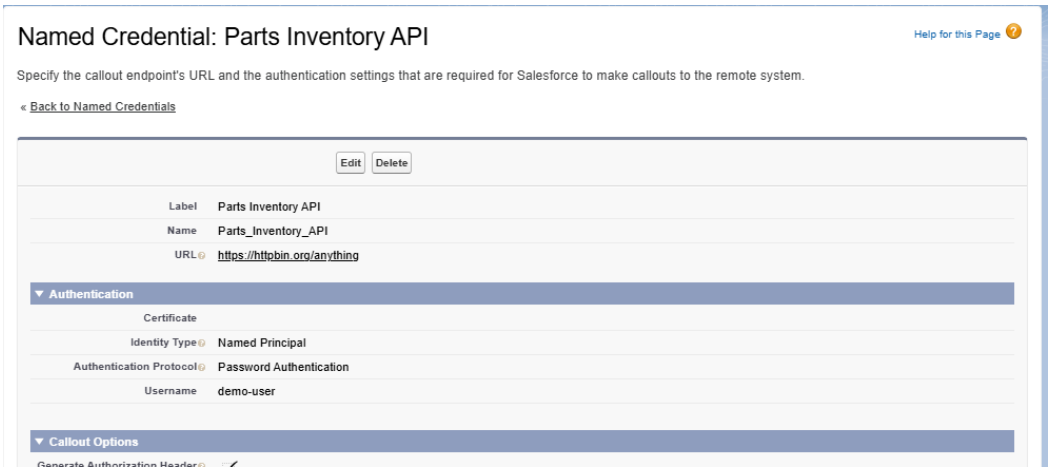


Phase 7: Integration & External Access

Named Credential – Mock API

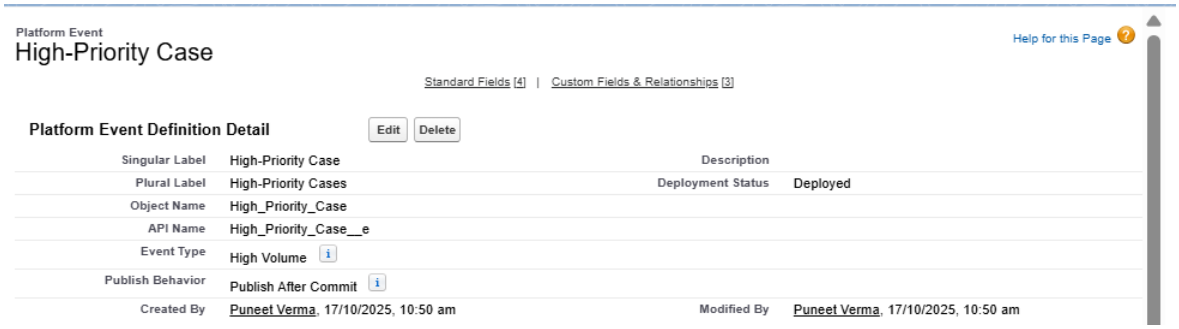
Used to securely call a public mock service for demo integration.

- **Label:** Mock API
- **URL:** https://jsonplaceholder.typicode.com
- **Auth:** None
  Use: For Apex callouts during testing.



Platform Event – High-Priority Case
Publishes events when high-priority cases are created for real-time external updates.

- **Fields:** Case_ID__c, Case_Subject__c, Priority__c
- **Trigger:** Record-triggered Flow or Apex trigger



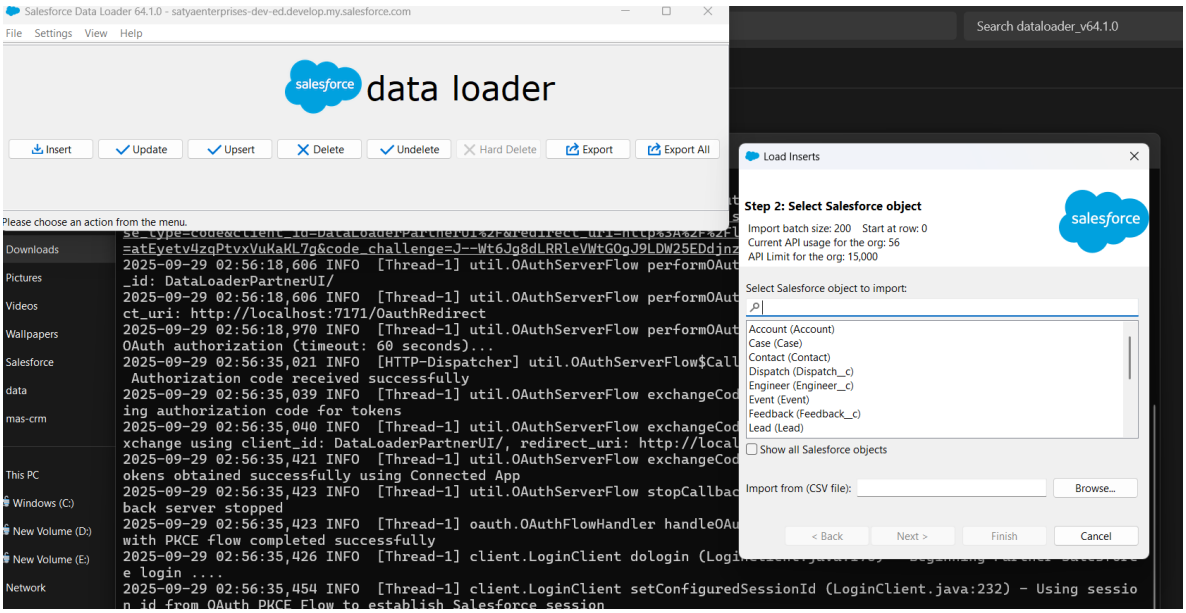
Phase 8: Data Management & Deployment

Data Management

Sample data has been provided in CSV format for all major custom objects, including Registered_Product__c, Service_Case__c, Dispatch__c, Feedback__c, and Engineer__c. This data can be imported using the Data Import Wizard or Data Loader for testing and demonstration purposes.

Deployment

The project is structured for deployment using Salesforce DX. The sfdx-project.json file and the manifest/package.xml file define the project structure and metadata for deployment to a scratch org or a sandbox.

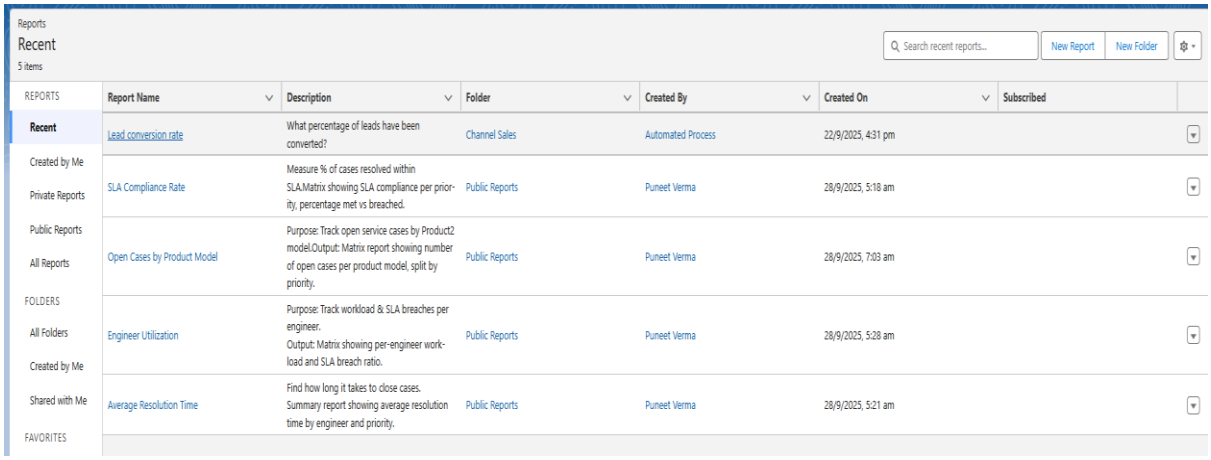


Phase 9: Reporting, Dashboards & Security Review

Reports and Dashboards

The data model is designed to support a variety of reports and dashboards to provide insights into the after-sales service process. Custom report types have been created to facilitate reporting on:

- Products with the most service cases
- SLA compliance and breach rates
- Engineer performance and workload
- Customer satisfaction and feedback trends



Security

The security model is designed to ensure that users only have access to the data they need to perform their jobs. This is achieved through a combination of:

- **Profiles**: Custom profiles have been created for Customers, Service Agents, Field Engineers, and Managers, with specific object and field-level security settings.
- **Roles**: A role hierarchy has been established to control record visibility.
- **Sharing Rules**: Sharing rules can be configured to grant additional access to records as needed.
- **Org-Wide Defaults (OWD)**: The OWD for Service_Case__c is set to "Private" for external users to ensure that customers can only see their own cases.

Dispatch	Public Read/Write	Private
Engineer	Public Read/Write	Private
Error Log	Public Read/Write	Private
Feedback	Controlled by Parent	Controlled by Parent
Registered Product	Public Read/Write	Public Read Only
Service Agent	Public Read/Write	Private
Service Case	Public Read Only	Private

Phase 10: Quality Assurance Testing

Test Classes

Apex test classes have been created for all Apex classes to ensure that the code is working as expected and to meet the code coverage requirements for deployment. This includes test classes for CaseStatusTransitionServiceTest, EngineerDashboardControllerTest, SLAServiceTest, ServiceAgentControllerTest, and WarrantyServiceTest

Use Case/Scenario	Test Steps	Expected Result
Engineer Assignment Flow	Create a new Service_Case__c with a specific location and required skill.	An available engineer with the matching skill and the closest proximity is automatically assigned to the case.
SLA Tracking	Create a Service_Case__c with a "High" priority.	The SLA_Deadline__c is automatically set to 24 hours from the creation time.
Warranty Claim Approval	Create a Service_Case__c for a product that was purchased more than one year ago.	The case is automatically submitted for manager approval, and the status is updated to "Submitted for Approval".
Defective Product Trigger	Update a Registered_Product__c record and check the Defective__c checkbox.	A new Service_Case__c with a "High" priority is automatically created for the product.

Conclusion

The Manufacturing After-Sales & Service CRM successfully addresses the key challenges of after-sales service management in the manufacturing industry. By automating key processes, providing real-time insights, and improving visibility for all stakeholders, the solution enhances operational efficiency, SLA compliance, and customer satisfaction.

Future Enhancements

- Integrate AI-based recommendations for engineer assignment to further optimize dispatching.
- Add chatbot support for customer self-service to handle common queries and service requests.
- Implement predictive analytics for SLA breach prevention to proactively identify and address at-risk cases.