Field Service WorkOrder Optimization

Sahir Ahmed

Computer Science & Engineering, Amity University
Noida, Uttar Pradesh
thesahirahmad@gmail.com

Abstract—**The Field Service Work Order Optimization System streamlines operations for a company providing installations and repairs. Utilizing a robust database, the system efficiently matches work orders with skilled technicians based on technicians location, availibility, and skills. The system employs a prioritization algorithm, focusing on assigning tasks to technician. Automated communication keeps technicians informed, while analytics offer insights for continuous improvement. Overall, this solution maximizes efficiency, reduces operational costs, and improves customer satisfaction in the dynamic realm of field service operations.**

1. Developer Account Creation & Activation

In this section, we will create a Salesforce developer account and activate it by following a simple sign-up process.

1.1 Creating Developer Account:

1. Navigate to Salesforce Developer Signup.
2. Fill in the form with the required details:
    - First name & last name
    - Email (a valid email address)
    - Role: Developer
    - Company: College Name
    - Country: India
    - Postal Code
    - Username (in the format username@organization.com)
3. Submit the form by clicking **Sign Me Up**.

1.2 Account Activation:

1. Check the email inbox used during sign-up for a verification email from Salesforce.
2. Click on **Verify Account**.
3. Set a password and answer the security question.
4. Once the account is verified, you will be redirected to the Salesforce Setup page.

---

2. Creating Objects in Salesforce

In this section, we will create three custom objects—Technician, WorkOrder, and Assignment—to store and manage relevant data in Salesforce.

2.1 Create Technician Object:

1. Download and edit the Technician spreadsheet.
2. Navigate to Salesforce **Setup → Object Manager → Create → Custom Object from Spreadsheet**.
3. Upload the spreadsheet, set **Technician ID** as the Record Name, and validate field types.
4. Finalize and save the Technician object.

2.2 Create WorkOrder Object:

1. Repeat the process for the WorkOrder spreadsheet.
2. Ensure proper field mapping and finalize creation.

2.3 Create Assignment Object:

1. Navigate to **Object Manager → Create → Custom Object**.
2. Configure the following:
    - Label Name: **Assignment**
    - Record Name Label: **Assignment ID**
    - Auto Number Format: **A-{0000}**, Starting Number: **1**
3. Enable additional features (reports and search).
4. Save the object.

---

3. Custom Tabs in Salesforce

This section explains how to create a custom tab for the Assignment object in Salesforce. Tabs provide a user-friendly way to access and manage object data.

3.1 Creating a Custom Tab for Assignment:

1. Navigate to **Setup → Tabs**.
2. Click **New** under the **Custom Object Tabs** section.
3. Select **Assignment** as the object and choose a tab style.
4. On the **Add to Profiles** page, retain default settings and click **Next**.
5. On the **Add to Custom App** page, keep default settings and click **Next**.
6. Click **Save** to finalize the tab creation.

Note: Tabs for **WorkOrder** and **Technician** objects are created automatically when the objects are created.

---

4. The Lightning App

This section guides the creation of a Lightning App to manage and streamline navigation across custom objects and other relevant items for the Field Service WorkOrder Optimization system.

4.1 Steps to Create a Lightning App:

1. **Access App Manager:**
    - Navigate to **Setup → App Manager → New Lightning App**.
2. **Provide App Details:**
    - App Name: Field Service WorkOrder Optimization
    - Developer Name: Auto-populated
    - Description: "An app to optimize field service operations by streamlining work orders, technicians, and assignments."
    - Image: Optional
    - Primary Color Hex Value: Default
3. **Configure App Options:**
    - Keep default settings.
4. **Add Navigation Items:**
    - Items to include: **Home, WorkOrder, Technician, Assignment, Reports, Dashboard**.
    - Ensure proper mapping for custom objects (e.g., **Asset** for custom object).
5. **Add User Profiles:**
    - Assign the **System Administrator** profile to the app.
6. **Save and Finalize:**
    - Click **Save & Finish** to complete the app creation.

---

5. Fields & Relationships

In this section, we focus on defining and managing fields and relationships for the **Assignment** and **WorkOrder** objects to store various types of data. Below are the steps for each sub-section.

5.1: Creating a Lookup Field in the Assignment Object

1. Go to **Setup → Object Manager**.
2. Search for the **Assignment** object using the Quick Find bar and click on it.
3. Click on **Fields & Relationships → New**.
4. Select **Data Type** as **Lookup Relationship** and click **Next**.
5. For **Related To**, select the **WorkOrder** object.
    - **Note:** Ensure you select the custom **WorkOrder** object you created earlier, not a standard object.
6. Set the **Field Label** as **WorkOrder ID**.
7. Click **Next → Next → Save & New**.

5.2: Manage Picklist Values (1)

1. Navigate to **Setup → Object Manager**.
2. Search for the **WorkOrder** object and click on it.
3. Go to **Fields & Relationships**, select the **Location** field, scroll to the **Values** section, and click **New**.
4. Add the following values:
    - Nasik
    - Warangal
    - Nanded
5. Click **Save**.

5.3: Manage Picklist Values (2)

1. Repeat steps 1 and 2 from **5.2**.
2. Add values to the respective fields in the **WorkOrder** object:
    - **Field: Priority** → Add **High**.
    - **Field: Service Type** → Add the following values:
        - → Hardware repair
        - → Troubleshoot/Debugging
        - → Lane-Management
3. Save the changes for each field.

5.4: Creating a Formula Field in the WorkOrder Object

1. Go to **Setup → Object Manager → WorkOrder**.
2. Click on **Fields & Relationships → New**.
3. Select **Data Type** as **Formula** and click **Next**.
4. Set **Field Label** and **Field Name** as **Date**.
5. Choose **Formula Return Type** as **Date** and click **Next**.
6. Under **Advanced Formula**, enter the following formula: CreatedDate
7. Click **Check Syntax** to validate the formula.
8. Click **Next → Next → Save**.

5.5: Creating Remaining Fields for Respective Objects

For **Assignment Object**:

1. Create a **Lookup Field**:
    - **Field Name:** Technician ID
    - **Data Type:** Lookup (Technician)

2. Create a **Formula Field**:

    - **Field Name:** Assignment Date
    - **Formula Return Type:** Date
    - **Formula:**
    - WorkOrder_ID__r.Date__c

3. Create another **Formula Field**:

    - **Field Name:** Completion Date
    - **Formula Return Type:** Date
    - **Formula:**
    - IF(ISPICKVAL(WorkOrder_ID__r.Status __c, 'Resolved'), WorkOrder_ID__r.LastModifiedDate, NULL)

---

6. Profiles

Profiles control user permissions and access in Salesforce. This section describes the creation of the Technician profile to restrict access to specific objects and fields.

6.1 Creating the Technician Profile

1. **Create the Profile:**

    - Go to **Setup** → Search for **Profiles** in the Quick Find box → Click on **Profiles**.
    - Click **New Profile**.
    - In the **Existing Profile** dropdown, select **Standard Platform User**.
    - Set **Profile Name** to **Technician**.
    - Click **Save**.

2. **Edit the Profile:**

    - While on the **Technician Profile** page, click **Edit**.
    - Scroll down to the **Custom Object Permissions** section.
    - Set **Read-only Access** for the following objects:

        → Technician
        → WorkOrder
        → Assignment

3. **Set Field Access Permissions:**

    - While still in the **Technician Profile**, scroll down to **Custom Field-Level Security**.
    - Next to the **WorkOrder** object, click **View**.
    - Click **Edit**.
    - Enable the checkbox for the **Status** field.
    - Click **Save**.

---

7. Users

Users are employees or individuals associated with the organization who interact with the Salesforce platform. This section explains the steps to create a new user, assign relevant roles, licenses, and profiles, and grant access to system features.

7.1 Create User

- **User Details:**

    - **First Name:** Elina
    - **Last Name:** Gilbert
    - **Alias:** E.Gilbert
    - **Email:** Provided personal email ID
    - **Username:** elina.gilbert@company.com
    - **Nickname:** Elina
    - **User License:** Salesforce Platform
    - **Profile:** Technician

---

8. Apex Trigger

Utilize Apex classes and triggers to automate workflows, manage data, and send notifications in Salesforce.

8.1 Create an Apex Class (WorkOrderClass)

1. **Navigate to Developer Console:**

    - Go to **Setup** → Click on the gear icon → Select **Developer Console**.

2. **Create the Apex Class:**

    - In the Developer Console → Click **File** → **New** → **Apex Class**.
    - Name the class: **WorkOrderClass** → Click **OK**.

3. **Write Code Logic:**
   Copy and paste the provided source code for **WorkOrderClass** and click **Save**.

8.2 Create an Apex Trigger (WorkOrderTrigger)

1. **In Developer Console:**

    - Go to **File** → **New** → **Apex Trigger**.

2. **Trigger Details:**

    - Name: **WorkOrderTrigger**
    - sObject: **WorkOrder__c**
    - Click **Submit**.

3. **Write Code Logic:**
   Copy and paste the provided source code for the trigger and click **Save**.

8.3 Create an Apex Class (AssigningEmail)

Repeat the steps to create a new Apex class. Name the class **AssigningEmail**, write the provided source code, and save it.

8.4 Create an Apex Trigger (AssignmentTrigger)

Create a new trigger for **Assignment__c** with the name **AssignmentTrigger**, paste the source code, and save.

8.5 Create an Apex Class (CompletionMail)

1. Follow the standard steps to create an Apex class.
2. Name: **CompletionMail**
3. Write the provided source code and save it.

8.6 Update the WorkOrderTrigger Class

1. Open the existing **WorkOrderTrigger** class.
2. Update it to include logic for handling updates with the **CompletionMail** class.
3. Save the updated trigger.

8.7 Create an Asynchronous Apex Class (RecordDeletions)

1. Create a new Apex class named **RecordDeletions**.
2. Implement batchable logic to delete **WorkOrder** records meeting specified criteria.
3. Save the code.

8.8 Create a Schedule Class (ScheduleClass)

1. Create an Apex class named **ScheduleClass**.
2. Write the code to schedule the **RecordDeletions** batch job.
3. Save the class.

8.9 Schedule the Apex Job

1. **Schedule Apex Execution:**
   - From the Setup page → Search for **Apex Classes** → Click **Schedule Apex**.
2. **Enter Job Details:**
   - Job Name: **DeleteAssignmentSchedule**
   - Apex Class: **ScheduleClass**
   - Frequency: **Monthly**
   - Preferred Start Time: Choose any suitable time.
3. **Save the Job.**

---

9. Reports & Dashboards

Utilize Salesforce Reports and Dashboards to visualize and analyze data for better decision-making.

9.1 Report: Creating a New Report

1. **Access Reports Tab:**
   - Open the desired Salesforce app → Click on the **Reports** tab.
2. **Create a New Report:**
   - Click **New Report**.
   - Select the report type: From the **category**, **report type panel**, or **search panel** → Click **Start Report**.
3. **Customize the Report:**
   - Use the left pane to drag and drop fields.
   - Example: Group the report by **WorkOrder ID**.
4. **Save or Run the Report:**
   - Save the report with a meaningful name and run it if needed.
   - Note: The report content will vary based on your data.

9.2 Create Reports

WorkOrders Status Report

1. **Create a New Report:**
   - Repeat the steps above to create a new report.
2. **Select Report Type:**
   - Choose **WorkOrders Status Reports** as the type.
3. **Customize the Report:**
   - Add relevant fields such as WorkOrder ID, Status, and Completion Date.
   - Apply filters to show specific statuses if needed.
4. **Save and Run the Report.**

Technician and Assignment Details Report

1. **Create a New Report:**
    - Repeat the steps to create another report.
2. **Select Report Type:**
    - Choose **Technician and Assignment Details Reports** as the type.
3. **Customize the Report:**
    - Add fields like Technician Name, Assignment ID, and WorkOrder details.
4. **Save and Run the Report.**

9.3 Dashboard: Create a New Dashboard

1. **Access the Dashboards Tab:**
    - Open the desired Salesforce app → Click on the **Dashboards** tab.
2. **Create the Dashboard:**
    - Click **New Dashboard**.
    - Enter a name (e.g., **WorkOrder Dashboard**) → Click **Create**.
3. **Add a Component:**
    - Click **Add Component**.
    - Select a report created in the earlier steps (e.g., **WorkOrders Status Report**) → Click **Select**.
4. **Save and Finalize:**
    - Click **Add**, then **Save**, and finally **Done** to complete the dashboard.

9.4 Create Dashboards: Bar Graph for Completed WorkOrders

1. **Create a New Dashboard:**
    - Open the **Dashboards** tab → Click **New Dashboard**.
    - Name the dashboard (e.g., **Completed WorkOrders Dashboard**) → Click **Create**.
2. **Add a Component:**
    - Click **Add Component**.
    - Select the **WorkOrders Status Report**.
    - Choose a **Vertical Bar Graph** as the chart type.
3. **Customize and Save:**
    - Customize the graph to show completed WorkOrders grouped by status or technician.
    - Save the dashboard and click **Done**.

---

10. Conclusion

This project has successfully demonstrated how Salesforce can be utilized to design a robust and efficient field service management system. By addressing core challenges in workorder management and communication, the solution enhances operational efficiency, empowers technicians, and delivers value to both the organization and its customers.

The tools and processes implemented in this project can be extended to other use cases, making it a versatile and scalable solution for any service-oriented organization.