

**Title:**  
**Field Service WorkOrder Optimization**

**College:** Seshadri Rao Gudlavalleru Engineering College

**Team Details:**

**Team Leader :** Sarika Boddu

**Team member :** Velpuru Siddhartha

**Team member :** Nidamanuri Srija

**Team member :** Kollapu Sahithi

# User Story:

## 1. INTRODUCTION

### 1.1 Project Overview:

Field service operations—such as installations, repairs, and maintenance—are critical to many businesses that rely on mobile workforces. Managing these operations efficiently involves technician scheduling, skill-based task assignments, route optimization, and real-time status tracking. Traditionally, companies have relied on manual coordination through calls, spreadsheets, and siloed systems, leading to scheduling conflicts, technician idle time, and missed service-level agreements. This project focuses on building a Field Service WorkOrder Optimization System on the Salesforce platform, designed to automate and optimize the work order lifecycle. The system intelligently matches work orders to technicians based on location, availability, and skills. Salesforce features such as Lightning UI, Apex Triggers, Flows, and Dashboards are used to streamline communication, prevent assignment errors, and track technician performance.

### 1.2 Purpose:

The primary goal is to develop a centralized, intelligent, and scalable solution for managing field service work orders using Salesforce CRM. The system provides:

- Smart technician assignment based on availability, location, and expertise
- Automation for scheduling, updates, and reminders
- Conflict prevention through validation and rule-based scheduling
- Real-time dashboards for work order tracking and technician utilization
- Foundation for future integration with GPS routing, mobile access, and customer feedback

## 2. IDEATION PHASE

### 2.1 Problem Statement:

Field service companies often face the following operational inefficiencies:

- No centralized visibility into work orders and technician schedules
- Manual coordination causing delays and overlaps in assignments
- Lack of real-time updates for technicians or customers
- Inability to track performance or identify optimization opportunities

These issues lead to missed deadlines, low customer satisfaction, and increased operational costs. A CRM-driven system is needed to automate and centralize field operations.

### 2.2 Empathy Map Canvas

**Customer Persona: Dispatch Manager / Field Technician / Customer**

#### Category Input

**Says** "Is the technician available today?" / "When will the repair be done?"

**Thinks** "How do I ensure all jobs are covered without overloading anyone?"

**Does** Checks availability manually, assigns tasks, handles complaints

**Feels** Stressed due to schedule clashes and lack of transparency

#### Goals:

- Efficient technician dispatch
- Accurate job scheduling
- Real-time communication

#### Challenges:

- Manual errors and double bookings
- No centralized platform for updates
- Delayed task completion due to skill mismatch

## 2.3 Brainstorming

After ideation and team discussions, key ideas include:

- A CRM-based platform to unify work order data
- Logic-based automation for technician assignments
- Conflict prevention using validation rules
- Technician availability tracking and skill-matching
- Dashboards for dispatchers to monitor workloads
- Audit trail and analytics for optimization

## 3. REQUIREMENT ANALYSIS

### 3.1 Customer Journey Map

Stages:

1. Service Request – Customer calls or submits a request form
2. Work Order Creation – Dispatcher logs request into CRM
3. Technician Assignment – Auto-match based on criteria (location, skills, availability)
4. Task Execution – Technician completes the task and updates status
5. Review & Feedback – Dispatcher verifies completion, collects feedback

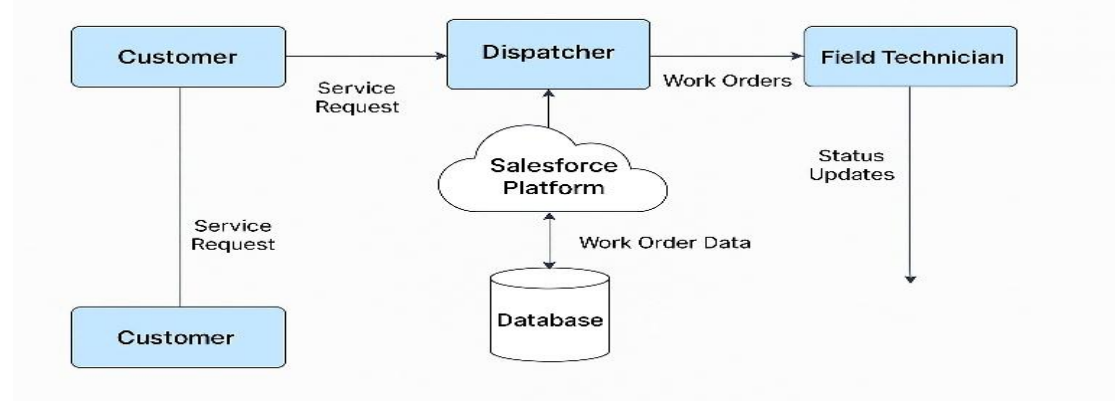
Pain Points:

- Overlapping assignments
- Technicians with wrong skill sets
- Delays due to poor communication

### 3.2 Solution Requirement

- Salesforce Developer Org
- Custom Objects:
  - Work\_Order\_\_c, Technician\_\_c, Skill\_\_c
- Standard Objects:
  - Account, Contact, Case
- Apex Triggers for assignment logic
- Flow automation for notifications and updates
- Validation Rules to avoid duplicate bookings
- Reports and Dashboards to monitor KPIs
- Lightning App for field technician interaction

### 3.3 Data Flow Diagram



### 3.4 Technology Stack

Component	Technology Used
Platform	Salesforce Developer Org
Backend Logic	Apex Triggers, Flows
Data Management	Custom & Standard Objects
User Interface	Salesforce Lightning UI
Reporting	Dashboards & Reports
Automation	Process Builder, Flow

## 4. PROJECT DESIGN

### 4.1 Problem-Solution Fit

The current field service domain lacks automation and centralization. The Salesforce solution bridges this gap by offering a cloud-based CRM with optimized logic for work order management. The system ensures timely, skill-appropriate task execution with complete visibility for dispatchers and real-time updates for technicians.

### 4.2 Proposed Solution

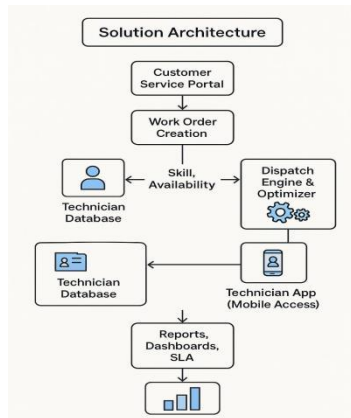
A smart Salesforce-based application that:

- Allows quick logging of service requests and work orders
- Assigns technicians automatically based on skill, location, and workload
- Sends notifications to relevant users via Flow

## 4.3 Solution Architecture

### Layers:

- **Presentation Layer:** Salesforce Lightning Experience (Desktop & Mobile)
- **Logic Layer:** Apex Triggers, Flows, Validation Rules
- **Data Layer:** Custom & Standard Objects (Work Order, Technician, Skill, etc.)



## 5. PROJECT PLANNING & SCHEDULING

### 5.1 Project Planning

Milestones with relevant screenshots provided: I.

### Salesforce Account Setup:

#### Activity 1: Creating Developer Account

Creating a developer org in salesforce.

1. Go to <https://developer.salesforce.com/signup>
2. On the sign up form, enter the following details :

**Build enterprise-quality apps fast to bring your ideas to life**

- Build apps fast with drag and drop tools
- Customize your data model with clicks
- Go further with Apex code
- Integrate with anything using powerful APIs
- Stay protected with enterprise-grade security
- Customize UI with clicks or any leading-edge web framework

**Sign up for your Salesforce Developer Edition**  
A full-featured copy of the Platform, for free

Complete the form to start your free trial. Our team will be in touch to help you make the most of your trial.

First Name\*  
Your first name

Last Name\*  
Your last name

Email\*  
Your email address

Role\*  
Your job role

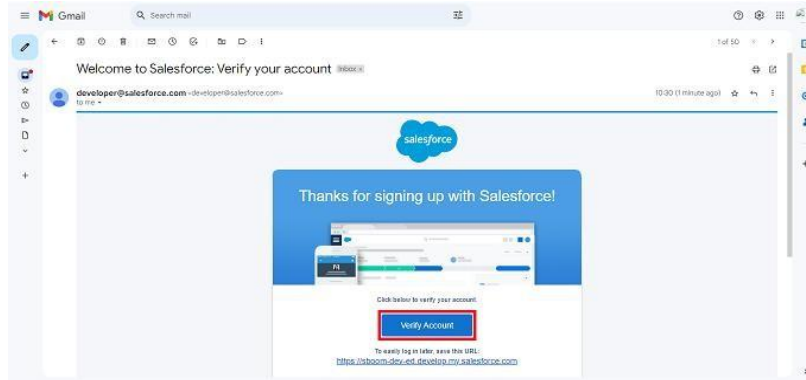
Company\*  
Company Name

1. First name & Last name
2. Email
3. Role : Developer
4. Company : College Name
5. Country : India

6. Postal Code : pin code
7. Username : should be a combination of your name and company

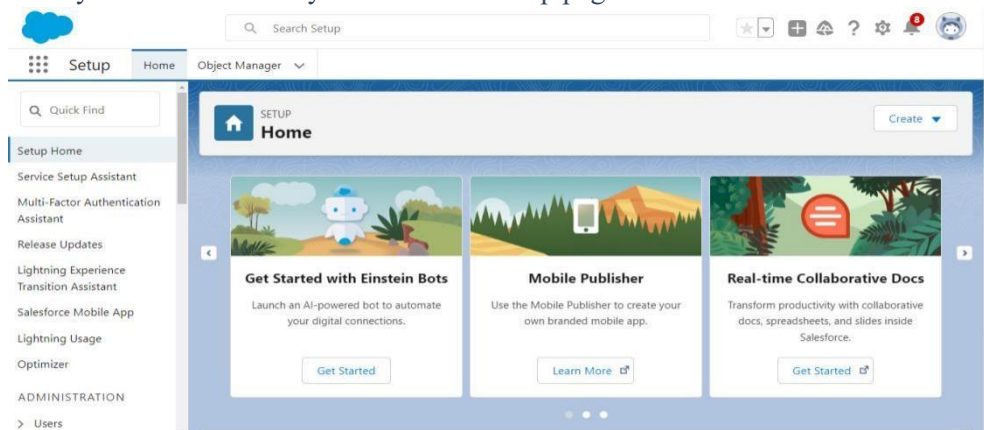
## Activity 2: Account Activation:

1. Go to the inbox of the email that you used while signing up. Click on the verify account to activate your account. The email may take 5-10mins.



2. Click on Verify Account
3. Give a password and answer a security question and click on change password.

4. Then you will redirect to your salesforce setup page.



## 2. Object Creation:

The purpose of creating object is it allows you to structure, store, and manage data specific to your application's workflow. For that we need to create 3 objects:

### i. Technician object

The screenshot shows the Salesforce Setup interface for the 'Technician' object. The left sidebar contains a navigation menu with options: Details, Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, and List View Button Layouts. The main content area is titled 'Technician' and includes a 'Details' section. The 'Details' section contains the following information:

Field	Value
Description	
API Name	Technician__c
Custom	✓
Singular Label	Technician
Plural Label	Technician
Enable Reports	✓
Track Activities	✓
Track Field History	
Deployment Status	Deployed
Help Settings	Standard salesforce.com Help Window

### ii. Workorder object

The screenshot shows the Salesforce Setup interface for the 'workOrder' object. The left sidebar contains a navigation menu with options: Details, Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, and List View Button Layouts. The main content area is titled 'workOrder' and includes a 'Details' section. The 'Details' section contains the following information:

Field	Value
Description	
API Name	workOrder__c
Custom	✓
Singular Label	workOrder
Plural Label	workOrder
Enable Reports	✓
Track Activities	✓
Track Field History	
Deployment Status	Deployed
Help Settings	Standard salesforce.com Help Window



### iii. Assignment object

The screenshot shows the Salesforce Setup interface for the Assignment object. The left sidebar contains a navigation menu with options like Details, Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, and List View Button Layout. The main content area is titled 'Assignment' and shows details for the object, including API Name (Assignment\_\_c), Custom, Singular Label (Assignment), Plural Label (Assignments), and various settings like Enable Reports, Track Activities, Track Field History, Deployment Status (Deployed), and Help Settings.

### 3. Tabs:

For this milestone we need to create 3 tabs for the CRM application Technician, Workorder and Assignment

The screenshot shows the Salesforce Setup interface for Custom Tabs. The left sidebar contains a navigation menu with options like User Interface, Rename Tabs and Labels, and Tabs. The main content area is titled 'Custom Tabs' and shows a table of Custom Object Tabs. The table has columns for Action, Label, Tab Style, and Description. The rows show three tabs: Assignments (Bookings), Technician (Box), and workOrder (Box). Below the table, there are sections for Web Tabs and Visualforce Tabs, both of which are currently empty.

### 4. Lightning App for :

Create a lightning app for the booking :

- From Setup, enter App Manager in the Quick Find and select App Manager.
- Click New Lightning App.
- Enter Field service workorder optimization as the App Name, then click next.
- Under App Options, leave the default selections and click next.
- Under Utility Items, leave as is and click Next.
- From Available Items, select Home, workorder, technician, assignment, Reports, and Dashboards and move them to Selected Item and Click Next.
- From Available Profiles, select System Administrator and move it to Selected Profiles.

viii. Click Save & Finish.

The screenshot shows the 'Lightning Experience App Manager' in the Salesforce Setup environment. The left sidebar contains navigation links for 'Setup', 'Home', and 'Object Manager'. The main content area displays a list of 26 items, sorted by App Name. The 'Field Service WorkOrder Optimization' app is highlighted. The table below lists the apps:

App Name	Developer Name	Description	Last Modified	App Type
6 Bolt Solutions	LightningBolt	Discover and manage business solutions designed for your industry.	17/06/2025, 2:46 pm	Lightning
7 Community	Community	Salesforce CRM Communities	17/06/2025, 2:46 pm	Classic
8 Content	Content	Salesforce CRM Content	17/06/2025, 2:46 pm	Classic
9 Data Manager	DataManager	Use Data Manager to view limits, monitor usage, and manage recipes.	17/06/2025, 2:46 pm	Lightning
10 Digital Experiences	SalesforceCMS	Manage content and media for all of your sites.	17/06/2025, 2:46 pm	Lightning
11 Field Service WorkOrder Optimization	Field_Service_WorkOrder_Optimization	this is app of Field Service WorkOrder Optimization	18/06/2025, 6:11 pm	Lightning
12 Lightning Usage App	LightningInstrumentation	View Adoption and Usage Metrics for Lightning Experience	17/06/2025, 2:46 pm	Lightning
13 Marketing CRM Classic	Marketing	Track sales and marketing efforts with CRM objects.	17/06/2025, 2:46 pm	Classic
14 My Service Journey	MSIAApp	Discover new customer service capabilities.	17/06/2025, 2:46 pm	Lightning
15 Platform	Platform	The fundamental Lightning Platform	17/06/2025, 2:46 pm	Classic
16 Queue Management	QueueManagement	Create and manage queues for your business.	17/06/2025, 2:46 pm	Lightning
17 Sales	Sales	The world's most popular sales force automation (SFA) solution	17/06/2025, 2:46 pm	Classic
18 Sales	LightningSales	Manage your sales process with accounts, leads, opportunities, and ...	17/06/2025, 2:46 pm	Lightning

## 5. Fields:

Create respective fields for the objects

### i. Assignment:

The screenshot shows the 'Assignment' page in the Salesforce Setup environment. The left sidebar contains navigation links for 'Setup', 'Home', and 'Object Manager'. The main content area displays the 'Fields & Relationships' section for the 'Assignment' object. The table below lists the fields:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Assignment Date	Assignment_Date__c	Formula (Date)		
Assignment ID	Name	Auto Number		✓
Completion Date	Completion_Date__c	Formula (Date)		
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Technician ID	Technician_ID__c	Lookup(Technician)		✓
WorkOrder ID	WorkOrder_ID__c	Lookup(workOrder)		✓

ii. Workorder:

Setup

Home

Object Manager

Q Search Setup

★

+

🔄

?

🔔

SETUP > OBJECT MANAGER

workOrder

Details

Fields & Relationships

Page Layouts

Lightning Record Pages

Buttons, Links, and Actions

Compact Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

Search Layouts

List View Button Layout

Restriction Rules

Fields & Relationships

11 Items, Sorted by Field Label

Q Quick Find

New

Deleted Fields

Field Dependencies

Set History Tracking

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Date	Date_c	Formula (Date)		
Description	Description_c	Long Text Area(131072)		
Email	Email_c	Email		
Last Modified By	LastModifiedById	Lookup(User)		
Location	Location_c	Picklist		
Owner	OwnerId	Lookup(User,Group)		✓
Priority	Priority_c	Picklist		
Service Type	Service_Type_c	Picklist		

iii. Technician:

Setup

Home

Object Manager

Q Search Setup

★

+

🔄

?

🔔

SETUP > OBJECT MANAGER

Technician

Details

Fields & Relationships

Page Layouts

Lightning Record Pages

Buttons, Links, and Actions

Compact Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

Search Layouts

Fields & Relationships

11 Items, Sorted by Field Label

Q Quick Find

New

Deleted Fields

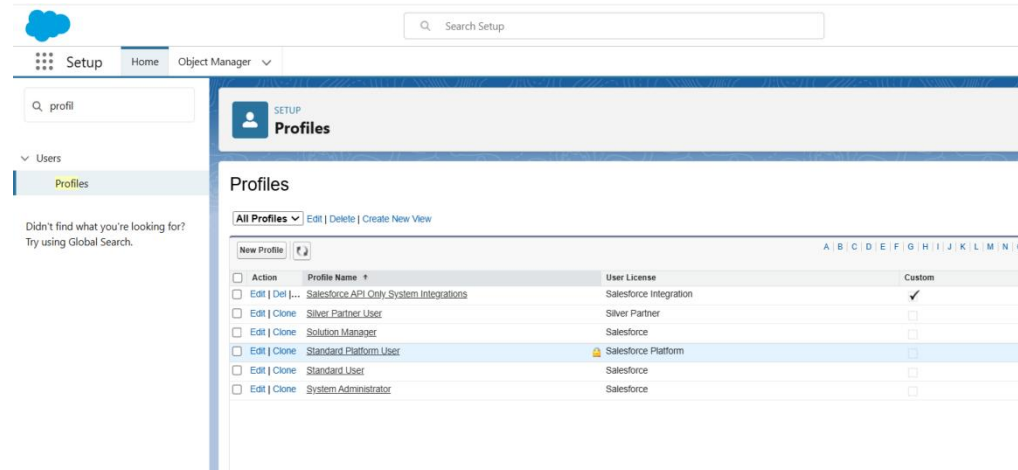
Field Dependencies

Set History Tracking

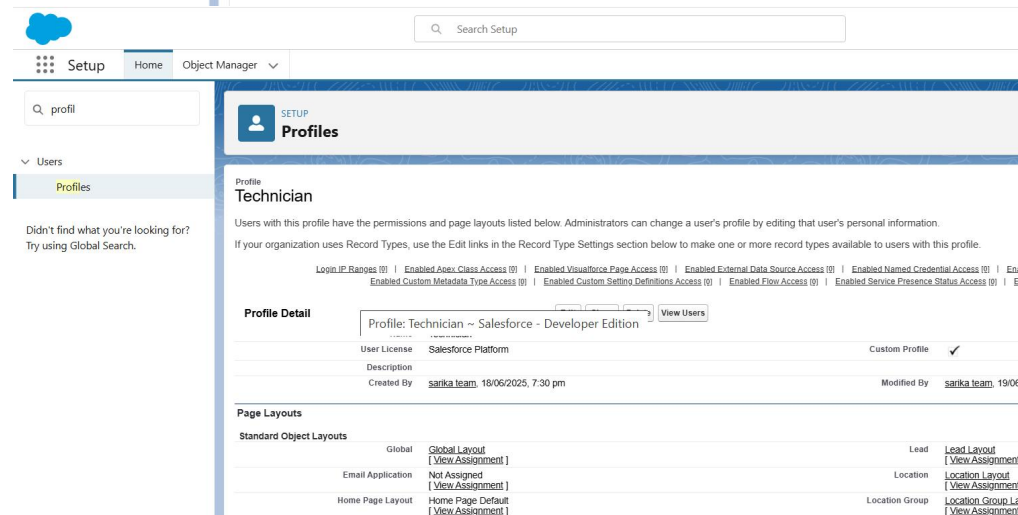
FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Availability	Availability_c	Picklist		
Created By	CreatedById	Lookup(User)		
Email	Email_c	Email		
Last Modified By	LastModifiedById	Lookup(User)		
Location	Location_c	Picklist		
Name	Name_c	Text(255)		
Owner	OwnerId	Lookup(User,Group)		

6. creation of profiles:

Here we have to create Technician profile.

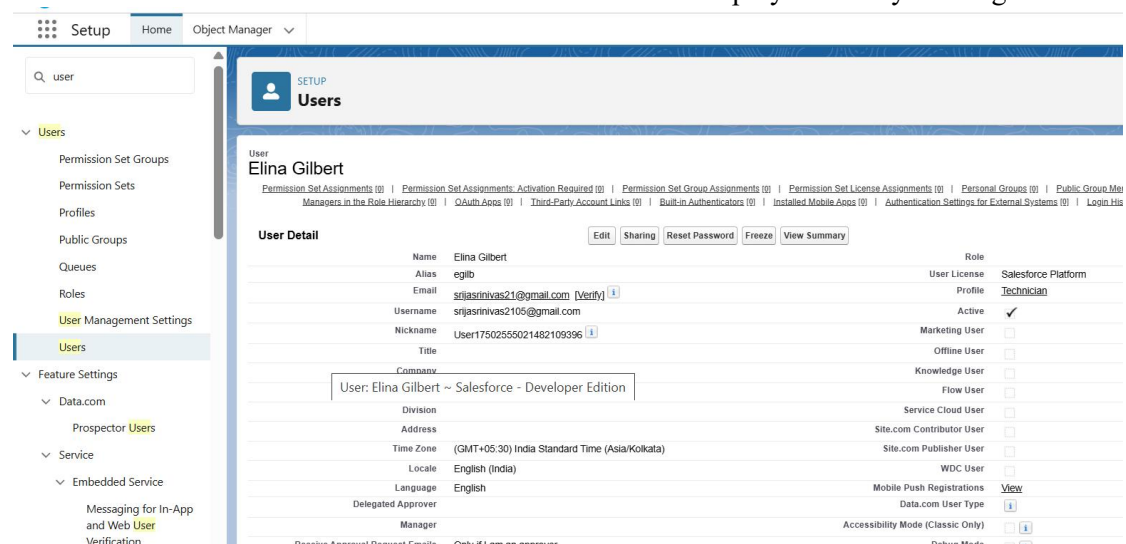


The screenshot shows the Salesforce Setup interface. The left sidebar has a search bar with 'profil' and a list of navigation items: Users, Profiles, and a search result 'Didn't find what you're looking for? Try using Global Search.' The main content area is titled 'SETUP Profiles' and shows a list of profiles. The 'All Profiles' dropdown is selected. The list includes profiles like 'Salesforce API Only System Integrations', 'Silver Partner User', 'Solution Manager', 'Standard Platform User', 'Standard User', and 'System Administrator'. The 'Standard Platform User' profile is highlighted.



The screenshot shows the 'Profile: Technician' page in Salesforce Setup. The left sidebar is the same as the previous screenshot. The main content area is titled 'SETUP Profiles' and shows the details for the 'Technician' profile. The 'Profile Detail' section shows the profile name 'Technician ~ Salesforce - Developer Edition', user license 'Salesforce Platform', and a description 'Created By: sarika team, 18/06/2025, 7:30 pm'. The 'Page Layouts' section shows the standard object layouts for the profile, including Global, Email Application, Home Page Layout, Lead, Location, and Location Group.

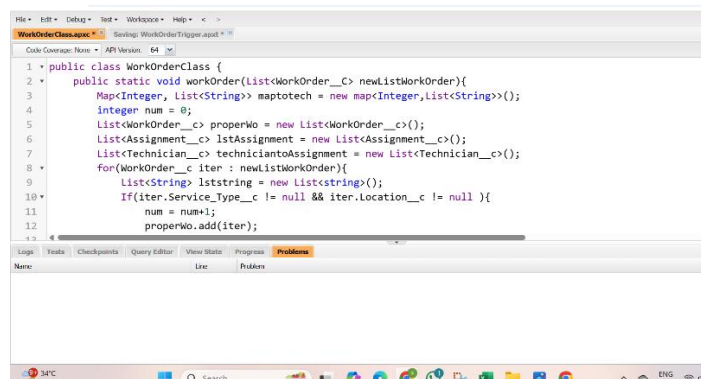
7. Creation of users: Users are defined as the employees of your organization.



The screenshot shows the Salesforce Setup interface for the 'Users' section. The left sidebar has a search bar with 'user' and a list of navigation items: Users, Permission Set Groups, Permission Sets, Profiles, Public Groups, Queues, Roles, User Management Settings, Users, Feature Settings, Data.com, Prospectors, Service, Embedded Service, Messaging for In-App and Web User Verification. The main content area is titled 'SETUP Users' and shows the details for a user named 'Elina Gilbert'. The 'User Detail' section shows the user's name, alias, email, username, nickname, title, company, division, address, time zone, locale, language, delegated approver, manager, and role. The 'User License' is 'Salesforce Platform' and the 'Profile' is 'Technician'. The 'Active' checkbox is checked. The 'Marketing User' checkbox is unchecked. The 'Offline User' checkbox is unchecked. The 'Knowledge User' checkbox is unchecked. The 'Flow User' checkbox is unchecked. The 'Service Cloud User' checkbox is unchecked. The 'Site.com Contributor User' checkbox is unchecked. The 'Site.com Publisher User' checkbox is unchecked. The 'WDC User' checkbox is unchecked. The 'Mobile Push Registrations' checkbox is checked. The 'Data.com User Type' is 'View'. The 'Accessibility Mode (Classic Only)' checkbox is checked. The 'Debug Mode' checkbox is checked.

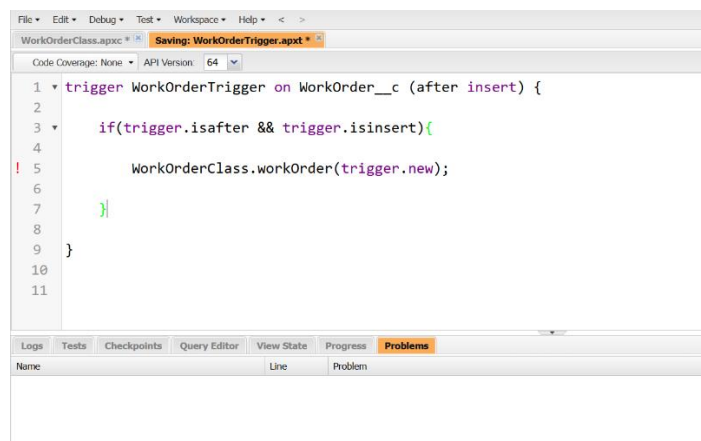
## 8. Apex triggers:

### I. Workorder class:



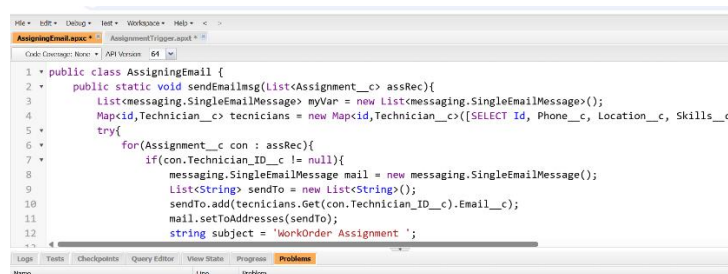
```
1 public class WorkOrderClass {
2     public static void workOrder(List<WorkOrder__c> newListWorkOrder){
3         Map<Integer, List<String>> maptech = new Map<Integer, List<String>>();
4         Integer num = 0;
5         List<WorkOrder__c> properWo = new List<WorkOrder__c>();
6         List<Assignment__c> lstAssignment = new List<Assignment__c>();
7         List<Technician__c> technicianToAssignment = new List<Technician__c>();
8         for(WorkOrder__c iter : newListWorkOrder){
9             List<String> lststring = new List<String>();
10            If(iter.Service_Type__c != null && iter.Location__c != null ){
11                num = num+1;
12                properWo.add(iter);
13            }
14        }
15    }
```

### II. Workorder trigger:



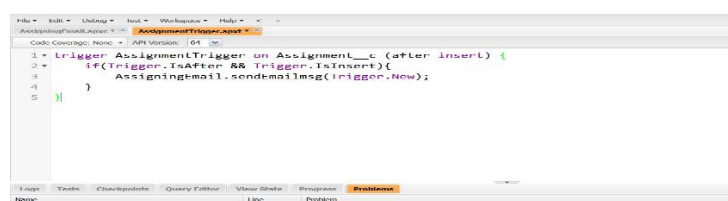
```
1 trigger WorkOrderTrigger on WorkOrder__c (after insert) {
2
3     if(trigger.isafter && trigger.isinsert){
4
5         WorkOrderClass.workOrder(trigger.new);
6
7     }
8
9 }
10
11
```

### III. Assignment class:



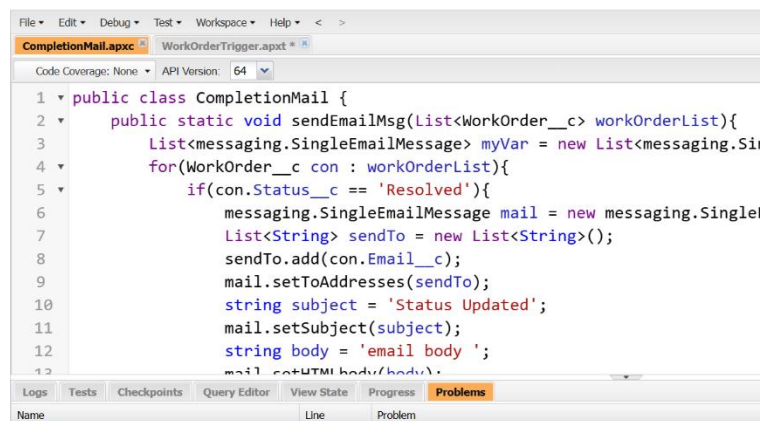
```
1 public class AssigningEmail {
2     public static void sendEmailmsg(List<Assignment__c> assRec){
3         List<messaging.SingleEmailMessage> myVar = new List<messaging.SingleEmailMessage>();
4         Map<Id, Technician__c> technicians = new Map<Id, Technician__c>([SELECT Id, Phone__c, Location__c, Skills__c,
5
6         for(Assignment__c con : assRec){
7             if(con.Technician_ID__c != null){
8                 messaging.SingleEmailMessage mail = new messaging.SingleEmailMessage();
9                 List<String> sendTo = new List<String>();
10                sendTo.add(tecnicians.Get(con.Technician_ID__c).Email__c);
11                mail.setToAddresses(sendTo);
12                string subject = 'WorkOrder Assignment ';
13            }
14        }
15    }
```

### IV. Assignment trigger:



```
1 trigger AssignmentTrigger on Assignment__c (after insert) {
2     if(trigger.isafter && trigger.isinsert){
3         AssigningEmail.sendEmailmsg(trigger.new);
4     }
5 }
```

## V. Completion mail class:

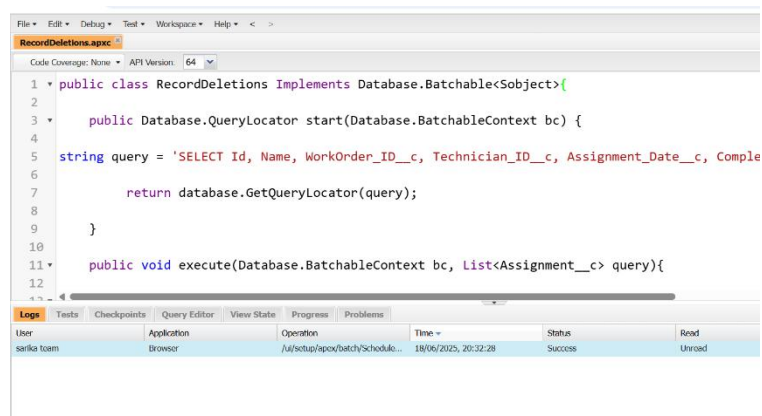


```
File Edit Debug Test Workspace Help < >
CompletionMail.aspxc WorkOrderTrigger.aspxc
Code Coverage: None API Version: 64
1 public class CompletionMail {
2     public static void sendEmailMsg(List<WorkOrder__c> workOrderList){
3         List<messaging.SingleEmailMessage> myVar = new List<messaging.Si
4         for(WorkOrder__c con : workOrderList){
5             if(con.Status__c == 'Resolved'){
6                 messaging.SingleEmailMessage mail = new messaging.Single
7                 List<String> sendTo = new List<String>();
8                 sendTo.add(con.Email__c);
9                 mail.setToAddresses(sendTo);
10                string subject = 'Status Updated';
11                mail.setSubject(subject);
12                string body = 'email body ';
13                mail.setHtmlBody(body);
14            }
15        }
16    }
17 }
```

Logs Tests Checkpoints Query Editor View State Progress Problems

Name	Line	Problem
------	------	---------

## VI. Record deletion class:

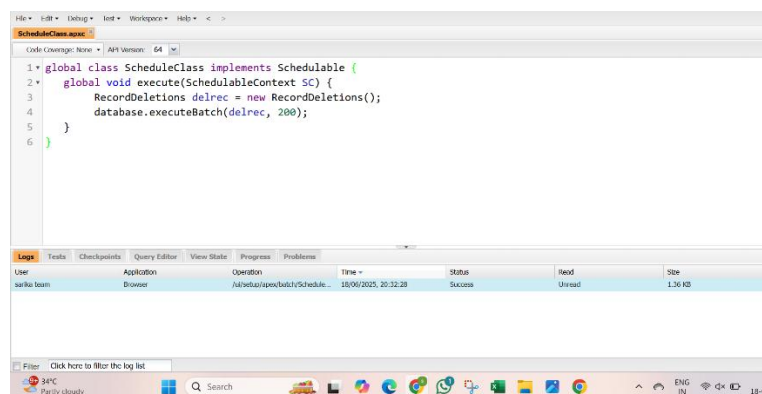


```
File Edit Debug Test Workspace Help < >
RecordDeletions.aspxc
Code Coverage: None API Version: 64
1 public class RecordDeletions implements Database.Batchable<Subject>{
2
3     public Database.QueryLocator start(Database.BatchableContext bc) {
4
5         string query = 'SELECT Id, Name, WorkOrder_ID__c, Technician_ID__c, Assignment_Date__c, Comple
6
7         return database.GetQueryLocator(query);
8
9     }
10
11     public void execute(Database.BatchableContext bc, List<Assignment__c> query){
12
13     }
14 }
```

Logs Tests Checkpoints Query Editor View State Progress Problems

User	Application	Operation	Time	Status	Read
sarika team	Browser	/ui/setup/apex/batch/Schedule...	18/06/2025, 20:32:28	Success	Unread

## VII. Schedule class:



```
File Edit Debug Test Workspace Help < >
ScheduleClass.aspxc
Code Coverage: None API Version: 64
1 global class ScheduleClass implements Schedulable {
2     global void execute(SchedulableContext sc) {
3         RecordDeletions delrec = new RecordDeletions();
4         database.executeBatch(delrec, 200);
5     }
6 }
7 }
```

Logs Tests Checkpoints Query Editor View State Progress Problems

User	Application	Operation	Time	Status	Read	Size
sarika team	Browser	/ui/setup/apex/batch/Schedule...	18/06/2025, 20:32:28	Success	Unread	1.36 KB

Filter Click here to filter the log list

24°C Parity cloudy

## 9.Reports and dashboards:

Field Service WorkO... Home workOrder Assignments Technician Reports Dashboards

workOrder Recently Viewed

2 Items • Updated a few seconds ago

New Change Owner Import Assign Label

Search this list...

	WorkOrder ID	Email	Service Type	Description	Location	Priority	status
1	002	hello@123.com	Line-Management	Lane repair	Warangal	High	New
2	001	abc@123.com	Hardware repair	Hardware Repair	Nasik	High	In Progress

Field Service WorkO... Home workOrder Assignments Technician Reports Dashboards

Report: WorkOrders Status Reports WorkOrders Status Report

Enable Field Editing Add Chart Edit

Total Records: 2

WorkOrder ID	status	Location	Priority	Service Type
001 (1)	In Progress	Nasik	High	Hardware repair
Subtotal				
002 (1)	New	Warangal	High	Line-Management
Subtotal				
Total (2)				

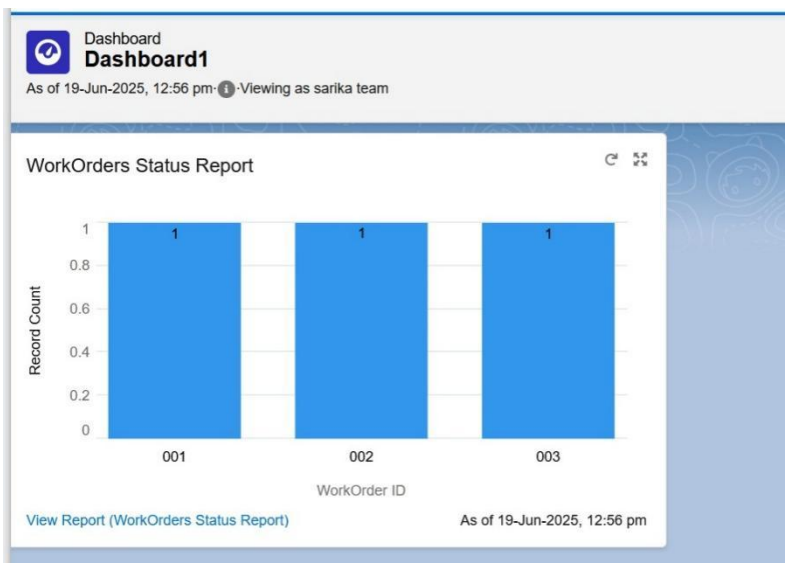
Reports Recent 3 Items

Search recent reports... New Report New Folder

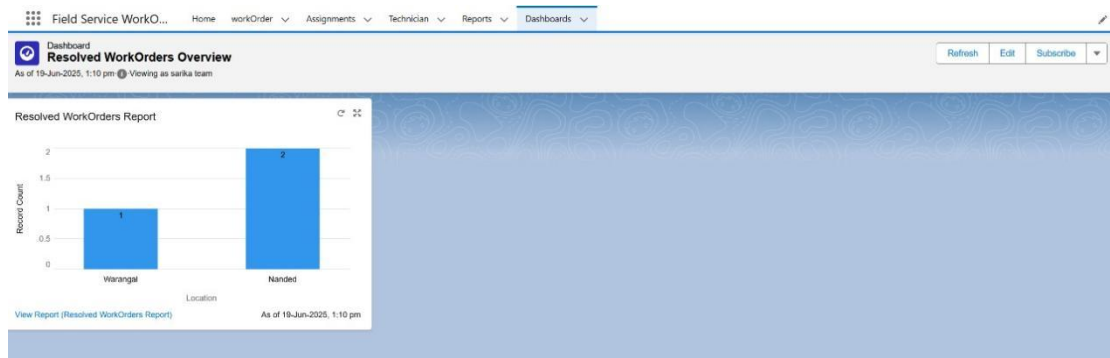
REPORTS	Report Name	Description	Folder	Created By	Created On	Subscribed
Recent	Resolved WorkOrders Report		Private Reports	sarika team	19/6/2025, 1:06 pm	
Created by Me	WorkOrders Status Report		Private Reports	sarika team	19/6/2025, 12:46 pm	
Private Reports	Technician and Assignment Details Report		Private Reports	sarika team	19/6/2025, 12:50 pm	
Public Reports						
All Reports						
FOLDERS						

### Dashboards:

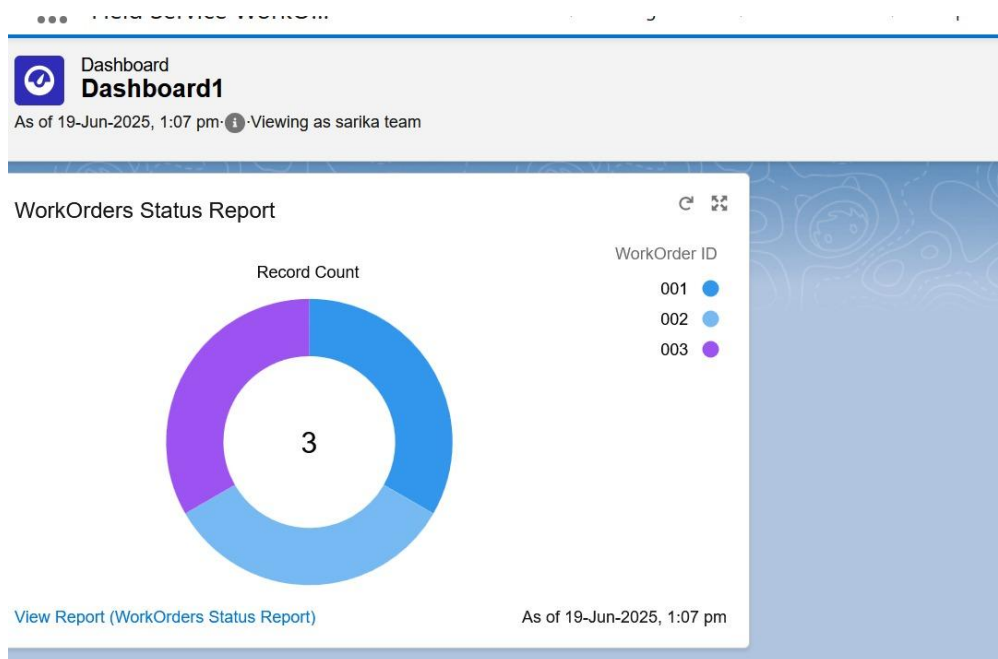
In Salesforce, dashboards are powerful tools for visualizing and analyzing data from reports, providing users with at-a-glance insights into key metrics and performance indicators.







## Preview:



## 10. Conclusion:

By implementing the enhanced **Field Service WorkOrder Optimization System** in Salesforce, service-based businesses can deliver a highly efficient and customer-centric experience for managing installations, repairs, and maintenance tasks. This intelligent CRM solution streamlines the assignment of work orders based on technician availability, location, and skill set—reducing scheduling conflicts and enhancing operational accuracy.

The project showcases the power of Salesforce in automating field operations and managing real-world technician dispatch scenarios. It minimizes manual coordination, reduces human error, and ensures timely service delivery through automated workflows, real-time updates, and centralized data management. With dashboards and reports, managers gain insights into workforce performance and job trends, enabling data-driven decisions.

This scalable and flexible application lays a solid foundation for future enhancements like mobile technician apps, route optimization using geolocation, and customer feedback integration—making it a strategic asset for organizations focused on service excellence and operational efficiency.



## 6. FUNCTIONAL AND PERFORMANCE TESTING

As the sole developer and tester, the application was thoroughly validated manually across a variety of field service use cases to ensure accurate functionality, automation behavior, and reliability.

Functional Checks:

- Creation, editing, and closure of Work Orders
- Automatic technician assignment using Flows and Apex Triggers
- Record-type specific page layouts for different service types (Installation, Repair, Maintenance)
- Lookup relationships between Work Orders, Technicians, and Accounts
- Validation rules for service date, skill match, and duplicate prevention

Performance Validation:

- Simulated 30+ Work Orders with varied complexity and technician profiles
- Dashboards loaded quickly with real-time metrics on technician performance and work order status
- No system crashes, automation delays, or data duplication encountered
- Flows triggered on creation/update performed within expected response times

## 7. RESULTS

The final Salesforce Field Service CRM application successfully delivers:

- End-to-end management of field service requests
- Automated technician allocation based on defined logic
- Real-time dashboards for task monitoring and technician availability
- Data validation to prevent errors and ensure scheduling integrity

All critical features were verified, and the system is ready for demonstration, user acceptance testing, or pilot deployment in a real-world scenario.

## 8. ADVANTAGES & DISADVANTAGES

**Advantages:**

- Cloud-based access for dispatchers and field agents from any location
- Automation-driven technician scheduling and service delivery
- Accurate validation to reduce human error in assignments
- Interactive dashboards and reports for management and operational insight

**Disadvantages:**

- Requires familiarity with Salesforce for optimal usage
- Dependence on consistent internet connectivity for real-time updates

- Limited offline functionality without additional mobile customization
- Custom development needed to integrate with external GPS or routing tools

## 10. FUTURE SCOPE

To further enhance the system, the following future enhancements are proposed:

- Mobile Application for field technicians via Salesforce Mobile Publisher or LWC-based mobile UI
- SMS/Email Notifications using Salesforce Flow and Messaging APIs for real-time communication
- Customer Feedback Collection integrated into post-service flow using surveys
- Integration with Payment Systems to allow invoicing and online payments post service
- Calendar Integration for dispatchers and staff to manage work schedules visually
- AI-based Assignment Optimization using Salesforce Einstein or external logic for high-efficiency dispatching

## 11. APPENDIX

- **Source Code / Configurations:** Configured within Salesforce Developer Org
- **Dataset:** Dummy entries created for testing purposes
- **Demo video link:**  
<https://drive.google.com/file/d/1Cogv91zpuqWihMeg0sDpwP6xLP6nlrfM/view?usp=drivesdk>