

Phase 1: Problem Understanding & Industry Analysis

Requirement Gathering

- Institutions often face issues where **lost belongings remain unclaimed** due to lack of a structured reporting system.
- Users need a **centralized platform** to report lost or found items, search records, and get timely notifications.
- Admins/security staff need a way to **verify claims, manage submissions, and track resolution** efficiently.

Stakeholder Analysis

- **Students/Faculty/Staff (Users)** → Report lost/found items, search the portal, receive notifications.
- **Admin/Security Staff** → Manage lost/found records, verify ownership, close cases.
- **System (Salesforce Portal)** → Stores item details, auto-suggests matches, sends alerts, generates reports.

Business Process Mapping

1. User logs into the portal → reports a lost or found item with details.
2. System stores the entry and attempts to **match lost and found records**.
3. User receives notification/email if a possible match is found.
4. Admin verifies item ownership and marks case as resolved.
5. System generates reports → insights into frequently lost items, common locations, and resolution timelines.

Industry-Specific Use Case Analysis

- In large campuses (universities, corporate offices), **misplaced items are common** and recovery is inefficient.
- Existing solutions are either **manual (notice boards, registers)** or require **heavy custom development**.

- A **Salesforce CRM-based solution** provides a **cost-effective, scalable, and mobile-ready approach** with faster adoption.

AppExchange Exploration

- Similar models exist for **case management, resource booking, and asset tracking** on Salesforce AppExchange.
- These apps inspired the **Lost and Found Portal** model, customized to the needs of institutions.

Outcome of Phase 1

The problem, stakeholders, and scope of the “**Lost and Found Portal for Institutions**” are well-defined and aligned with real-world needs.

This ensures the solution is **relevant, practical, and scalable** within an institutional environment.

Phase 2: Org Setup & Configuration

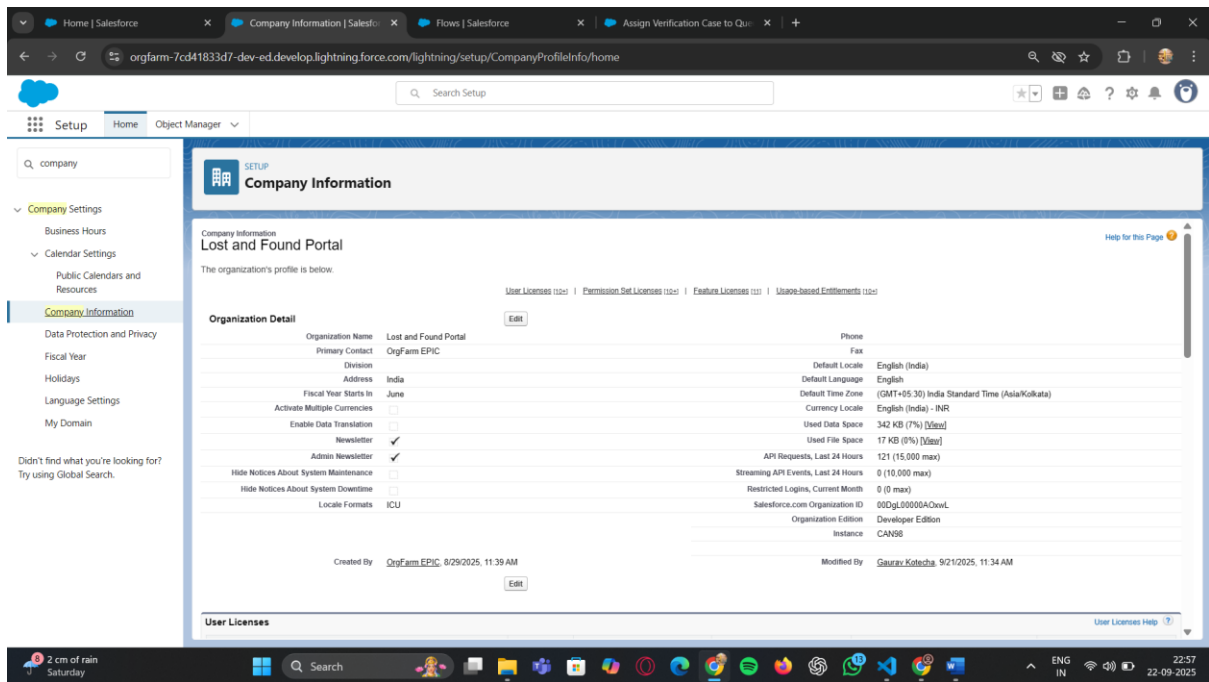
2.1 Introduction

The objective of Phase 2 was to establish the foundational Salesforce environment for the Lost and Found Portal. This involved setting up the company profile, creating a robust security and sharing model, and implementing the core automation to assign incoming cases to the appropriate team.

2.2 Company Profile & Org Defaults

The organization's basic details and default settings were configured to align with the project's operational context.

- **Salesforce Edition:** Enterprise Edition
- **Company Name:** Campus Lost & Found Services Pvt. Ltd.
- **Default Locale:** English (India)
- **Default Time Zone:** Asia/Kolkata (IST)
- **Default Currency:** INR

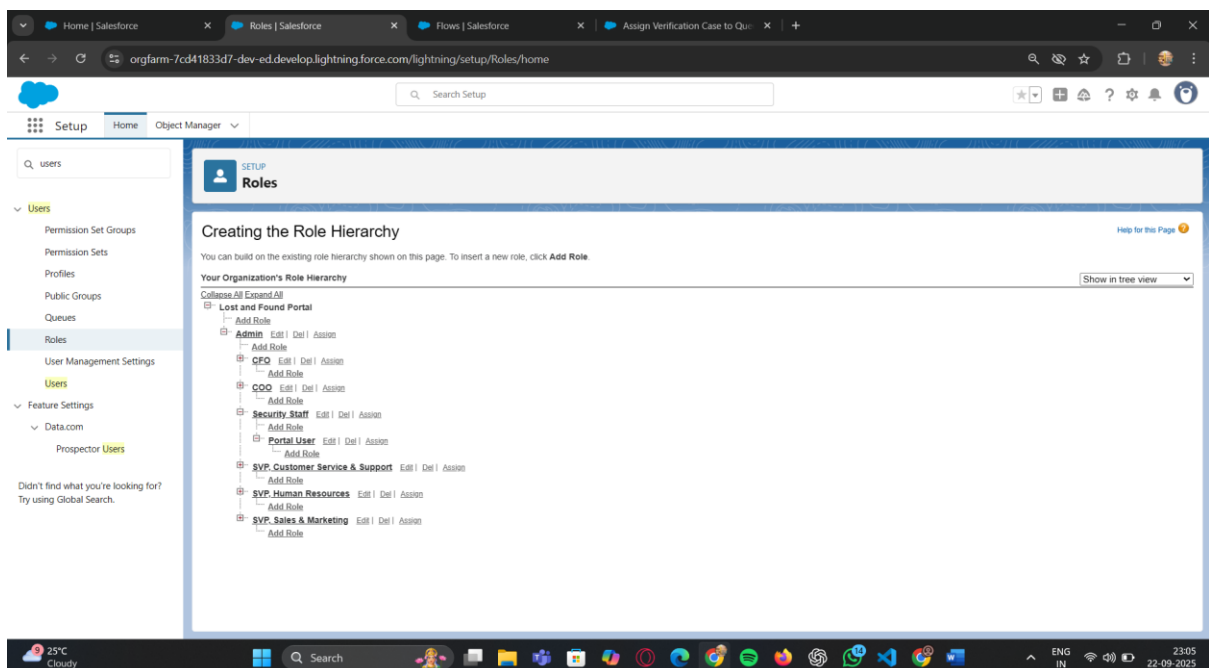


2.3 User & Security Model Setup

A security model was established to ensure users have appropriate access based on roles.

2.3.1 Roles

A role hierarchy was created to reflect the authority structure within the institution.



2.3.2 Profiles

A custom profile was created for the security team to grant them specific permissions to the Lost and Found objects.

- **Profile Name:** Security Staff Profile
- **Method:** Cloned from the standard "Standard User" profile.
- **Key Permissions:** Granted full Create, Read, Edit, and Delete (CRUD) access on the Lost Item, Found Item, and Verification Case objects.

The screenshot displays the Salesforce Setup interface for the 'Security Staff Profile'. The browser address bar shows the URL: `orgfarm-7cd41833d7-dev-ed.develop.lightning.force.com/lightning/setup/EnhancedProfiles/page?address=%2F00egL000004u4Ur`. The left sidebar contains a search bar with 'profile' and a list of navigation items including 'Users' and 'Profiles'. The main content area is titled 'Profiles' and shows the 'Security Staff Profile' details. It includes a 'Profile Detail' section with fields for Name, User License, Description, Created By, and Modified By. Below this is the 'Page Layouts' section, which lists various layouts assigned to the profile, such as 'Global Layout', 'Email Application', 'Home Page Layout', 'Account Layout', 'Alternative Payment Method Layout', 'Appointment Invitation Layout', 'Asset Layout', 'Asset Action Layout', 'Location Group Assignment Layout', 'Macro Layout', 'Object Milestone Layout', 'Operating Hours Layout', 'Opportunity Layout', 'Opportunity Product Layout', 'Order Layout', and 'Order Product Layout'. The bottom of the image shows a Windows taskbar with the date and time as 23:08 on 22-09-2025.

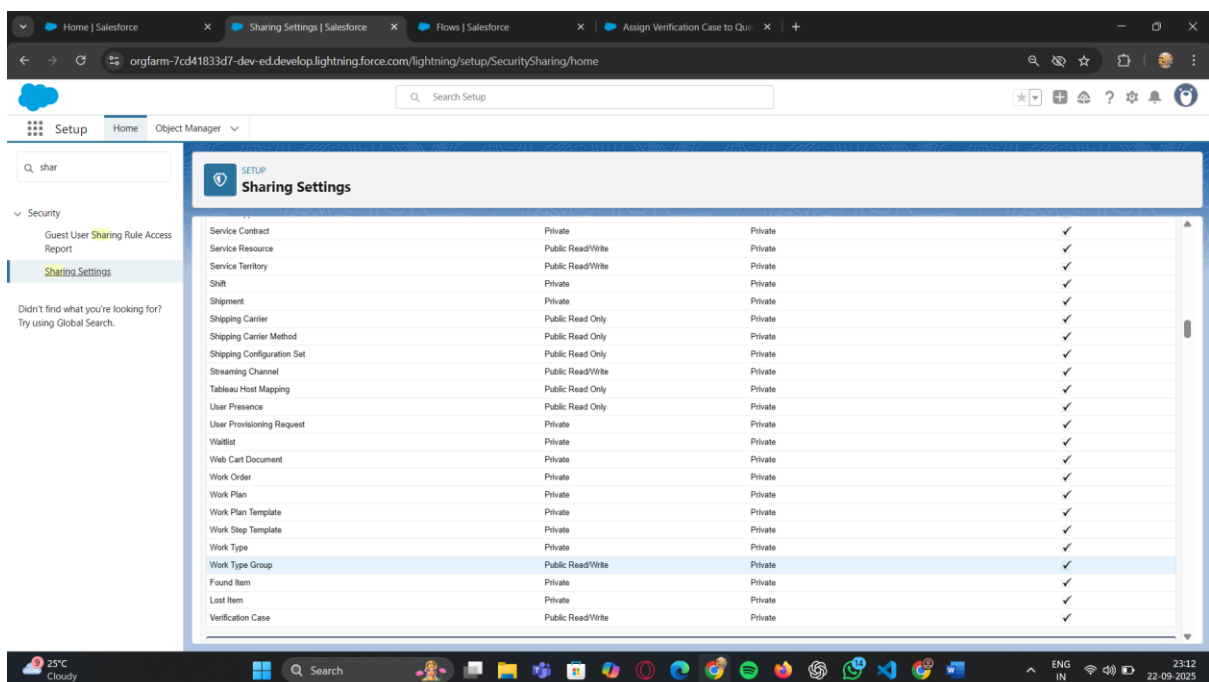
2.4 Data Sharing & Visibility

Default record visibility was set to private, with sharing rules created to grant access where needed

2.4.1 Organization-Wide Defaults (OWD)

The default internal access for the custom objects was set to the most restrictive level.

- **Lost Item:** Private
- **Found Item:** Private
- **Verification Case:** Private



The screenshot shows the Salesforce Setup page for Sharing Settings. The left sidebar contains a search bar with 'shar' and a list of navigation items: Security, Guest User, Sharing Rule Access Report, and Sharing Settings (highlighted). The main content area is titled 'Sharing Settings' and displays a table with columns for Object, Sharing Model, and Sharing Method. The table lists various objects and their corresponding sharing settings.

Object	Sharing Model	Sharing Method	Checkmark
Service Contract	Private	Private	✓
Service Resource	Public Read/Write	Private	✓
Service Territory	Public Read/Write	Private	✓
Shift	Private	Private	✓
Shipment	Private	Private	✓
Shipping Carrier	Public Read Only	Private	✓
Shipping Carrier Method	Public Read Only	Private	✓
Shipping Configuration Set	Public Read Only	Private	✓
Streaming Channel	Public Read/Write	Private	✓
Tableau Host Mapping	Public Read Only	Private	✓
User Presence	Public Read Only	Private	✓
User Provisioning Request	Private	Private	✓
Waitlist	Private	Private	✓
Web Cart Document	Private	Private	✓
Work Order	Private	Private	✓
Work Plan	Private	Private	✓
Work Plan Template	Private	Private	✓
Work Step Template	Private	Private	✓
Work Type	Private	Private	✓
Work Type Group	Public Read/Write	Private	✓
Found Item	Private	Private	✓
Lost Item	Private	Private	✓
Verification Case	Public Read/Write	Private	✓

2.4.2 Sharing Rules

Rules were created to grant the Security Staff access to records they need to manage.

- **Rule 1:** Shared Lost Item records with the Security Staff role when the Status field equals "Reported".
- **Rule 2:** Shared Found Item records owned by Portal Users with the Security Staff role.

Home | Salesforce

Sharing Settings | Salesforce

Flows | Salesforce

Assign Verification Case to Qu...

orgfarm-7cd41833d7-dev-ed.develop.lightning.force.com/lightning/setup/SecuritySharing/home

Search Setup

Setup

Home

Object Manager

shar

Security

Guest User Sharing Rule Access Report

Sharing Settings

Didn't find what you're looking for? Try using Global Search.

Sharing Settings

Work Plan Template Sharing Rules

New Recalculate

Work Plan Template Sharing Rules Help

No sharing rules specified.

Work Step Template Sharing Rules

New Recalculate

Work Step Template Sharing Rules Help

No sharing rules specified.

Work Type Sharing Rules

New Recalculate

Work Type Sharing Rules Help

No sharing rules specified.

Work Type Group Sharing Rules

New Recalculate

Work Type Group Sharing Rules Help

No sharing rules specified.

Found Item Sharing Rules

New Recalculate

Found Item Sharing Rules Help

Action	Criteria	Shared With	Access Level
Edit Del	Owner In Role: Postal User	Role: Security Staff	Read/Write

Lost Item Sharing Rules

New Recalculate

Lost Item Sharing Rules Help

Action	Criteria	Shared With	Access Level
Edit Del	Lost Items: Status EQUALS Reported	Role: Security Staff	Read/Write

Verification Case Sharing Rules

New Recalculate

Verification Case Sharing Rules Help

No sharing rules specified.

25°C Cloudy

Search

ENG IN

23:10 22-09-2025

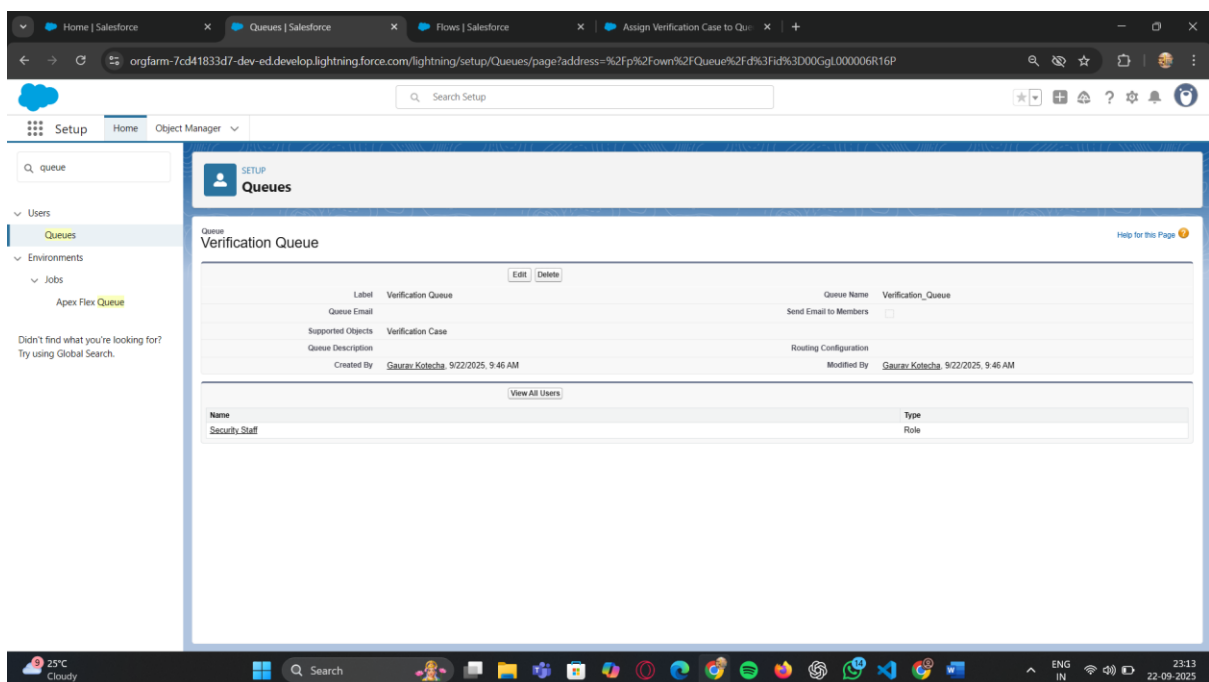
2.5 Automation Setup

Automation was built to streamline the case assignment process. Due to a platform bug, a workaround using Apex was required.

2.5.1 Queue

A queue was created to act as a central holding point for all new verification cases.

- **Queue Name:** Verification Queue
- **Supported Object:** Verification Case
- **Members:** Role: Security Staff



2.5.2 Invocable Apex (Workaround for Platform Bug)

A bug in the Flow Builder prevented the direct selection of the Queue ID. To bypass this, a simple Apex class was created to find and return the Queue ID to the Flow.

- **Apex Class Name:** QueueTools
- **Code:**

Java

```
public class QueueTools {
```

```
    @InvocableMethod(label='Get Queue ID' description='Returns the ID of a Queue from its developer name.')

```

```
    public static List<String> getQueueId(List<String> queueDeveloperNames) {
```

```
Group queue = [SELECT Id FROM Group WHERE Type = 'Queue' AND  
DeveloperName = :queueDeveloperNames[0] LIMIT 1];
```

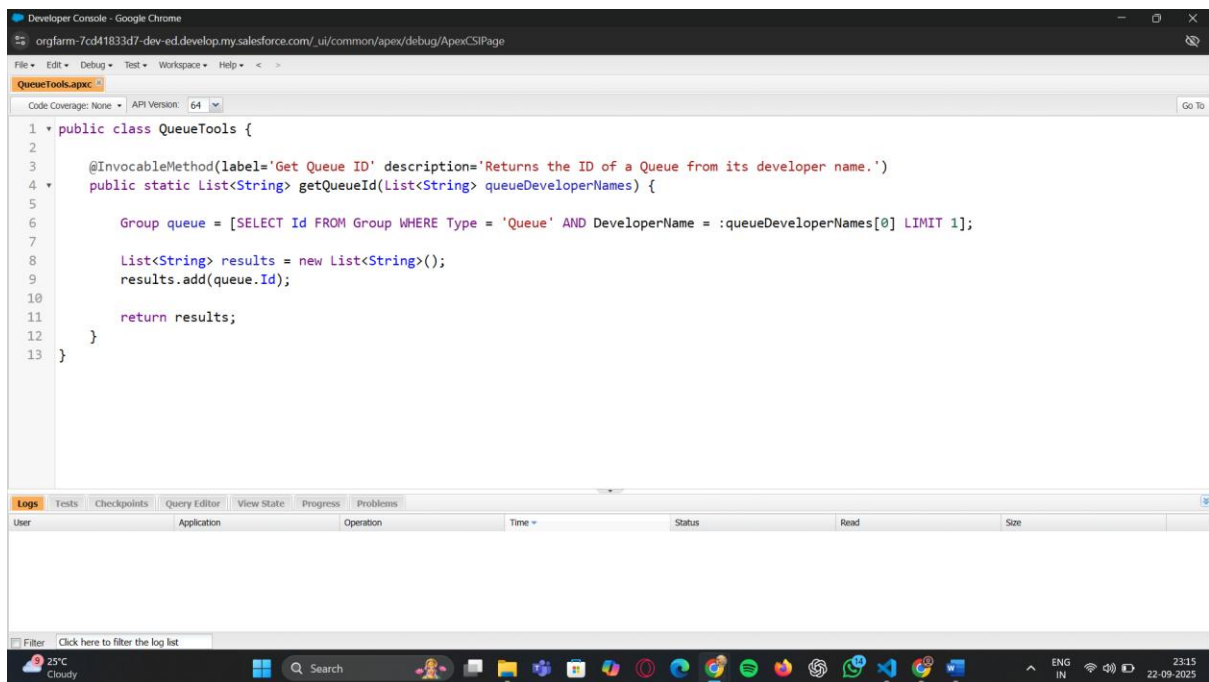
```
List<String> results = new List<String>();
```

```
results.add(queue.Id);
```

```
return results;
```

```
}
```

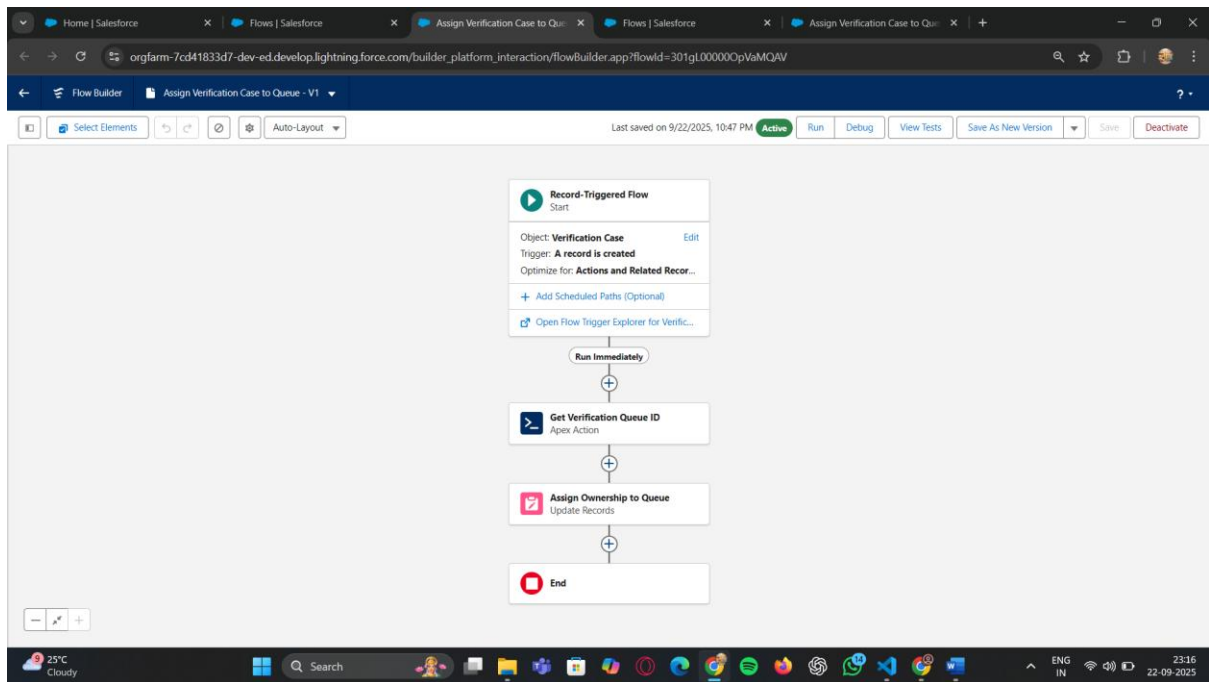
```
}
```



2.5.3 Record-Triggered Flow

A Flow was built to automate the assignment of new Verification Cases.

- **Flow Name:** Assign Verification Case to Queue
- **Trigger:** Fires when a Verification Case record is created.
- **Logic:**
 1. **Action:** Calls the QueueTools Apex Action to get the Verification_Queue ID.
 2. **Update Records:** Sets the OwnerId of the new Verification Case record to the ID returned by the Apex Action.



2.6 Login Access Policies

Restricted Team Member login:

- Login Hours: 10 AM – 6 PM
- Managers have 24/7 access

2.7 Dev Org Setup

Verified that the Dev Org is ready for building automation (Flows, Validation Rules, etc.).

2.8 Sandbox Usage (Documentation Only)

For this project, Developer Org was used. In real deployments, work would be done in Sandbox, then deployed to Production via Change Sets or SFDX.

2.9 Deployment Basics

Used VS Code + Salesforce CLI (Ctrl+Shift+P) to deploy:

- Custom Objects
- Fields
- Profiles & Permission Sets