# Phase 2: Org Setup & Configuration

AI-Enabled Hospital & Pharmacy Management System **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Goal:** To configure the Salesforce org with users, roles, profiles, permissions, and security settings to create a strong foundation for the Hospital & Pharmacy Management System**.**

**Tasks in Phase 2:**

* Salesforce Editions
* Company Profile Setup
* Business Hours & Holidays
* Fiscal Year Settings
* User Setup & Licenses
* Profiles
* Roles
* Permission Sets
* OWD
* Sharing Rules
* Login Access Policies
* Dev Org Setup
* Sandbox Usage
* Deployment Basics

**Salesforce Edition**

I used the Developer Edition org for this project. It provided all the features I needed to build and test the Hospital & Pharmacy application.

**A screenshot of a computer

AI-generated content may be incorrect.**

**Company Profile Setup**

I updated the company profile with the project name, set the time zone to IST, local to English (India), and currency to INR. This ensured appointments and billing records follow Indian settings.

**A screenshot of a computer

AI-generated content may be incorrect.**

**Business Hours & Holidays**

I created Hospital Standard Hours and added holidays like Independence Day. This will help in future for workflows and appointment scheduling.

Hospital Standard Hours

**A screenshot of a computer

AI-generated content may be incorrect.**

Business Holidays

**A screenshot of a computer

AI-generated content may be incorrect.**

**Fiscal Year**

I kept the Standard Fiscal Year (April–March) since it is enough for hospital billing and reporting.

**A screenshot of a computer

AI-generated content may be incorrect.**

**User Setup & Licenses**

I created test users for Doctor, Nurse, Admin, Patient and Pharmacist and assigned them Salesforce Platform licenses. These users helped me test role-based access.

**A screenshot of a computer

AI-generated content may be incorrect.**

**User license for admin user (**similar for all the users**)**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Profiles**

I cloned the Standard Platform User profile and created custom profiles like Doctor Profile, Admin Profile, Nurse Profile, Patient Profile and Pharmacist Profile. For example, the Doctor Profile had access to Patients and Appointments, while the Pharmacist Profile had access to Inventory and Orders.

**A screenshot of a computer

AI-generated content may be incorrect.**

**Roles**

I created a Hospital Admin role under CEO, and child roles for Doctor, Nurse, and Pharmacist. I assigned my test users to these roles. This way, Hospital Admin could see records created by doctors or nurses**.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Permission Sets**

I made a Hospital App Access permission set and gave CRUD access to all custom objects (Patients, Doctors, Appointments, Pharmacy Inventory, Billing, Medicine Orders).

**A screenshot of a computer

AI-generated content may be incorrect.**

**OWD (Organization-Wide Defaults)**

I set OWD as:

* Patients → Private
* Appointments → Private
* Pharmacy Inventory → Public Read Only
* Billing → Private
* Medicine Orders → Private

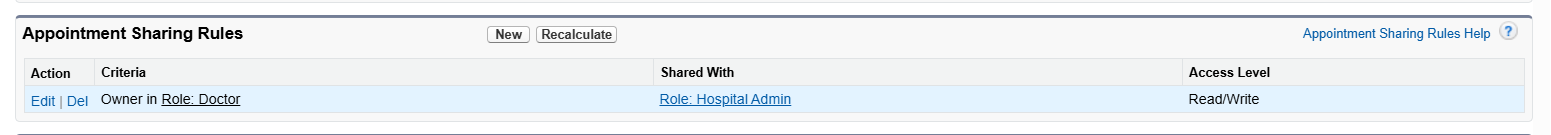
This kept sensitive data restricted while keeping inventory visible.

**A screenshot of a computer

AI-generated content may be incorrect.**

**Sharing Rules**

Since OWD was strict, I created Sharing Rules to give Hospital Admin access to all Patients and Appointments owned by doctors. This ensured admins could see everything.

****

**A screenshot of a computer

AI-generated content may be incorrect.**

**Login Access Policies**

I enabled Administrators Can Log in as Any User, which made it easier to test by logging in as Doctor, Nurse, or Pharmacist without managing separate logins.

**A screenshot of a computer

AI-generated content may be incorrect.**

**Dev Org Setup**

I set up My Domain(**caretrack-dev-ed.develop.my.salesforce.com)** and changed Email Deliverability to All Email, so login credentials and notifications could be sent to users.

My Domain

**A screenshot of a computer

AI-generated content may be incorrect.**

Deliverability Settings

**A screenshot of a computer

AI-generated content may be incorrect.**

**Sandbox**

Developer Edition doesn’t allow creating real sandboxes. In an Enterprise org, I would have used a Developer Sandbox for testing. To simulate this, I planned to use a second Developer Org as a sandbox for practice.

This ensures I don’t accidentally break my main org while experimenting.

**Deployment**

In this project, I explored the different ways Salesforce allows us to move changes from one org to another. Since I am working on a Developer Edition org, I don’t have access to Change Sets (they are only available in Enterprise orgs with Sandboxes). So, for my implementation, I used the Unmanaged Package approach to deploy my custom objects and fields.

How It Helps My Project?

By creating and installing this package, I proved that my Hospital & Pharmacy Management System is not locked to one org. I can package it and deploy it to any other Salesforce org if required. This also shows how the project could be moved from a test org to a real production org in future.

**Conclusion**

In Phase 2, I successfully set up and configured my Salesforce org for the Hospital & Pharmacy Management System. I updated the company details, defined business hours, and created separate users for Doctor, Nurse, Admin, Patient and Pharmacist with proper roles and profiles. I also built permission sets, set OWD rules, and added sharing rules to secure sensitive data like patients and billing, while still giving the Hospital Admin full visibility. Along the way, I configured My Domain, enabled email deliverability, and practiced deployment using unmanaged packages. These steps gave my project a solid and realistic foundation, making it well-prepared for the advanced automation and reporting I’ll be working on in Phase 3 and so on.