Courier Management System (CMS)

User Story:

Courier and logistics companies often struggle to efficiently manage shipments and deliveries due to reliance on manual processes and disconnected systems. The absence of a centralized shipment tracking mechanism makes it difficult to monitor parcel movement in real time, resulting in delayed delivery updates and poor visibility for customers. Delivery agent assignments are performed manually, increasing the chances of scheduling conflicts and operational delays. Furthermore, the lack of automated notifications prevents timely communication with customers, while the absence of performance analytics limits managerial insight into operational efficiency. Managing multiple branches becomes complex without an integrated platform, leading to coordination issues across locations. Collectively, these challenges result in customer dissatisfaction, reduced productivity, operational inefficiencies, and ultimately revenue loss for the organization.

1. INTRODUCTION

1.1 Project Overview :

Courier and logistics companies face challenges in efficiently managing shipments and deliveries due to manual operations and disconnected systems. These issues lead to delayed updates, poor customer visibility, and operational inefficiencies.

The Courier Management System (CMS) is a Salesforce-based cloud solution that centralizes shipment booking, tracking, delivery agent assignment, automated status updates, billing, and analytics. The system improves operational efficiency, enhances customer satisfaction, and provides real-time business insights.

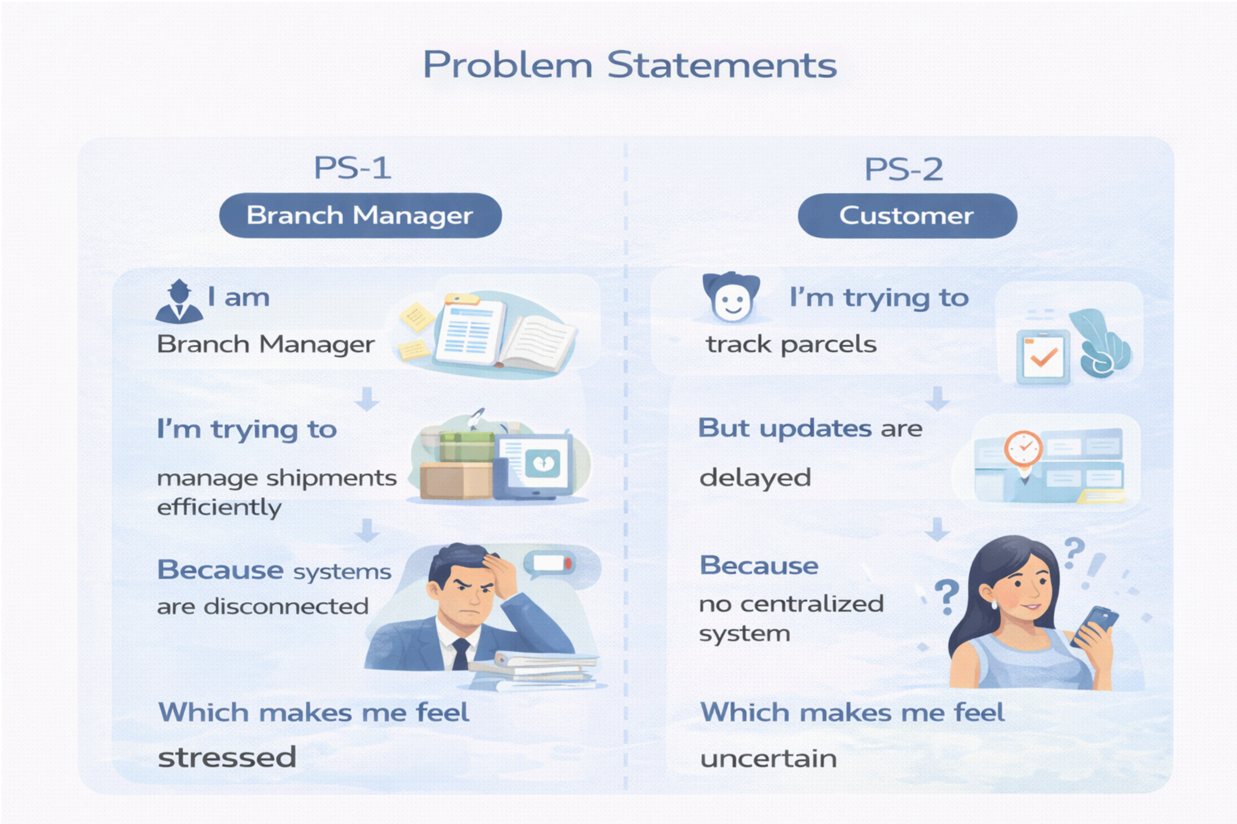
1.2 Purpose :

The purpose of the Courier Management System (CMS) is to automate and centralize courier and logistics operations using the Salesforce platform. The system aims to replace manual and fragmented processes with an integrated digital solution that enables efficient shipment booking, real-time tracking, delivery agent assignment, automated notifications, billing, and performance reporting. By providing a single, unified platform for managing shipments across multiple branches, CMS improves operational visibility, enhances customer communication, reduces delays and errors, and supports faster decision-making through analytics. Ultimately, the system is designed to increase service efficiency, customer satisfaction, and overall business profitability.

2. IDEATION PHASE

2.1 Problem Statement (Customer Problem Statement Template)

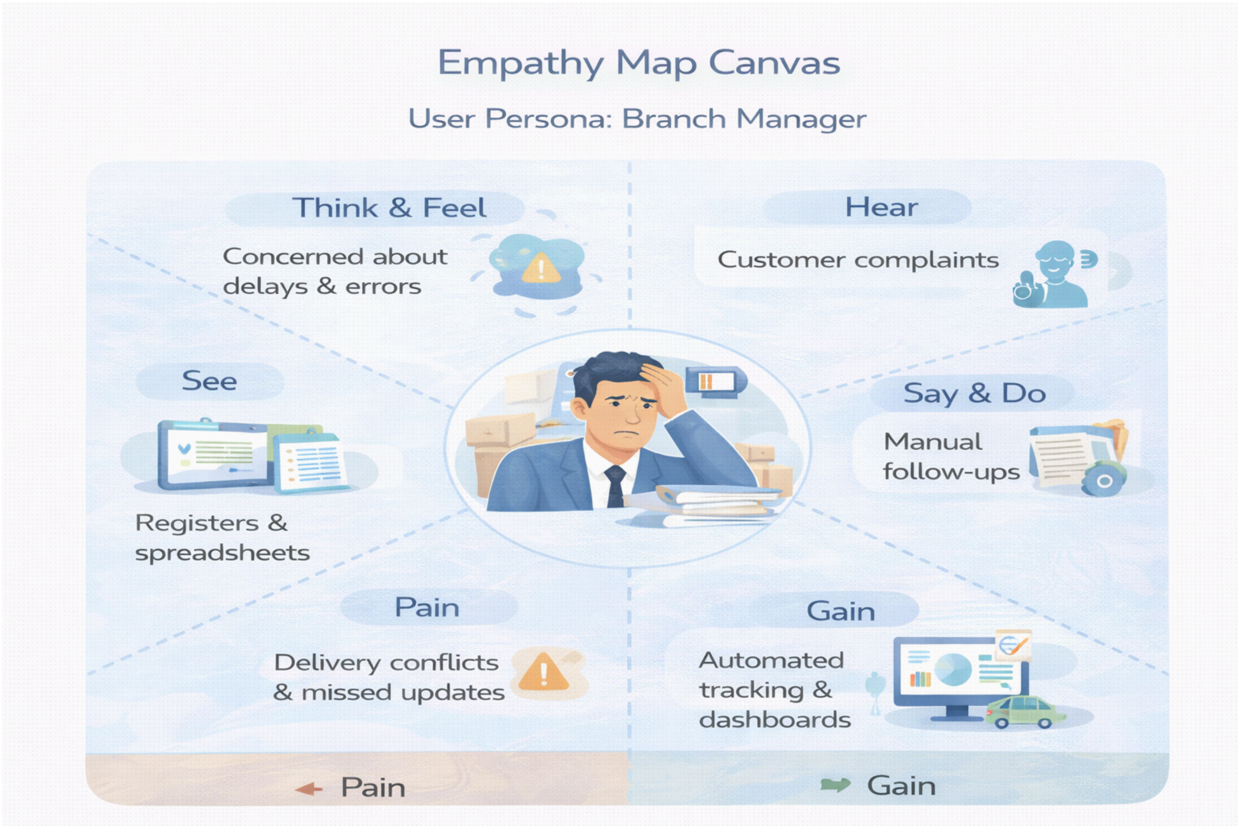
PS	I am (Customer)	I’m trying to	But	Because	Which makes me feel
PS-1	Branch Manager	manage shipments efficiently	tracking is manual	systems are disconnected	stressed
PS-2	Customer	track parcels	updates are delayed	no centralized system	uncertain



2.2 Empathy Map Canvas

User Persona: Branch Manager

- **Think & Feel:** Concerned about delays & errors
- **Hear:** Customer complaints
- **See:** Registers & spreadsheets
- **Say & Do:** Manual follow-ups
- **Pain:** Delivery conflicts & missed updates
- **Gain:** Automated tracking & dashboards



2.3 Brainstorming & Idea Prioritization

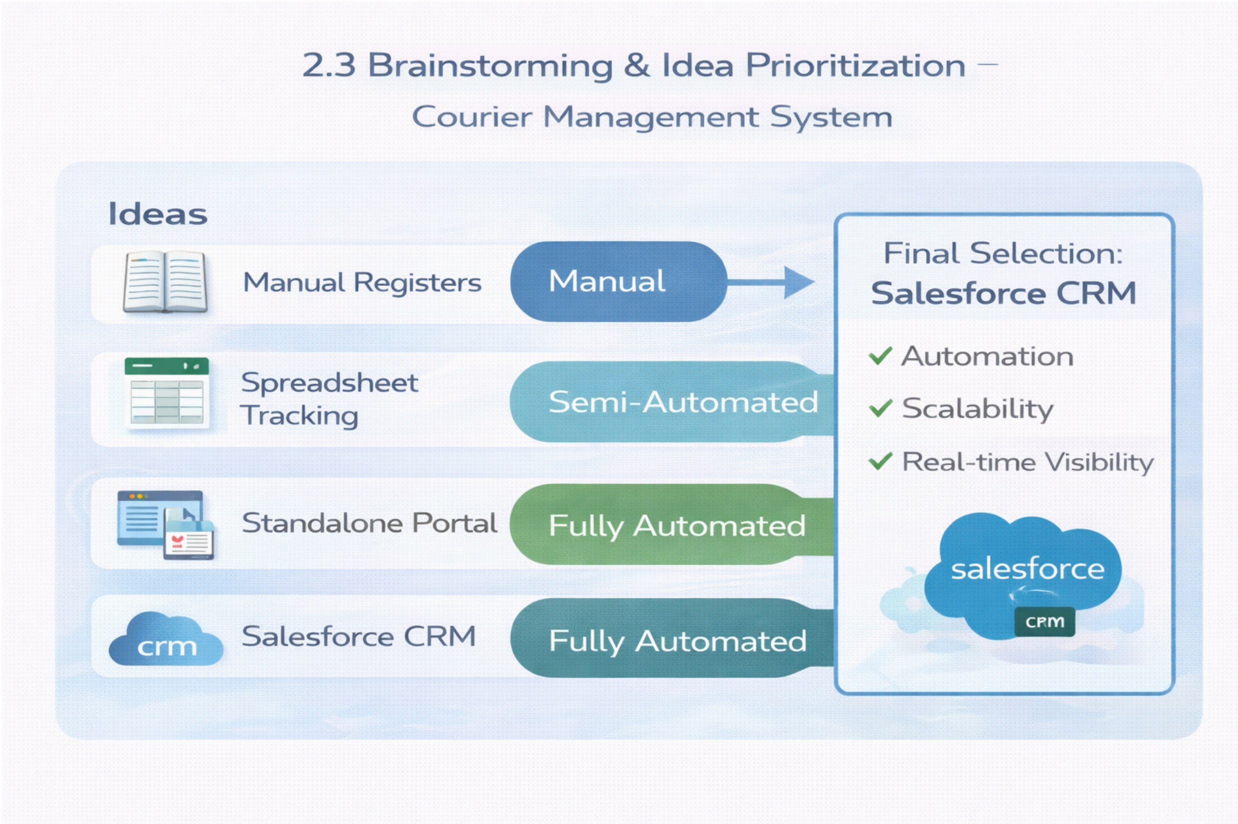
Ideas Considered

- Manual registers
- Spreadsheet tracking
- Standalone portal
- Salesforce CRM

Final Selection

Salesforce CRM selected due to:

- Automation
- Scalability
- Real-time visibility

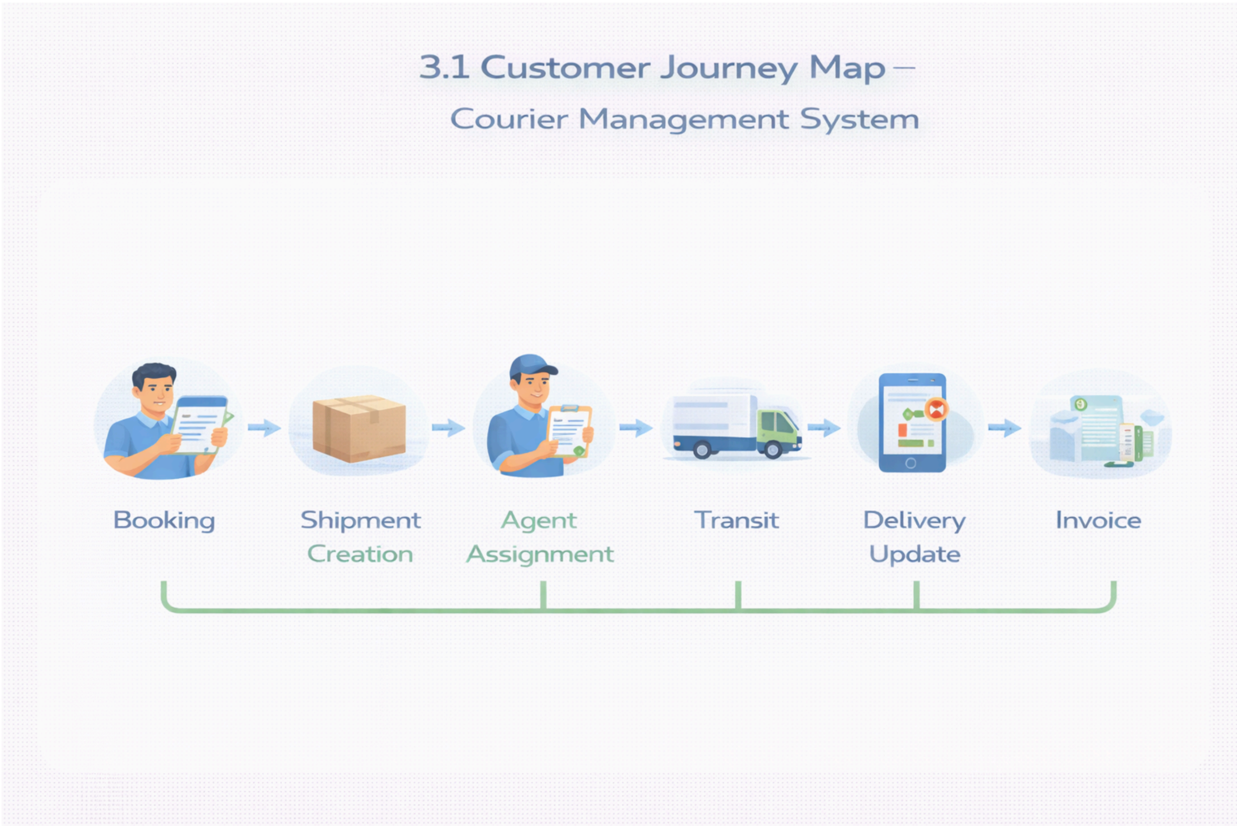


3.REQUIREMENT ANALYSIS PHASE

3.1 Customer Journey Map

Stages:

- Booking → Shipment Creation → Agent Assignment → Transit → Delivery Update → Invoice → Reporting



3.2 Solution Requirements

Functional Requirements

FR No	Requirement
FR-1	Customer management
FR-2	Shipment booking
FR-3	Delivery agent assignment
FR-4	Real-time status updates
FR-5	Invoice generation

FR-6	Reports & dashboards

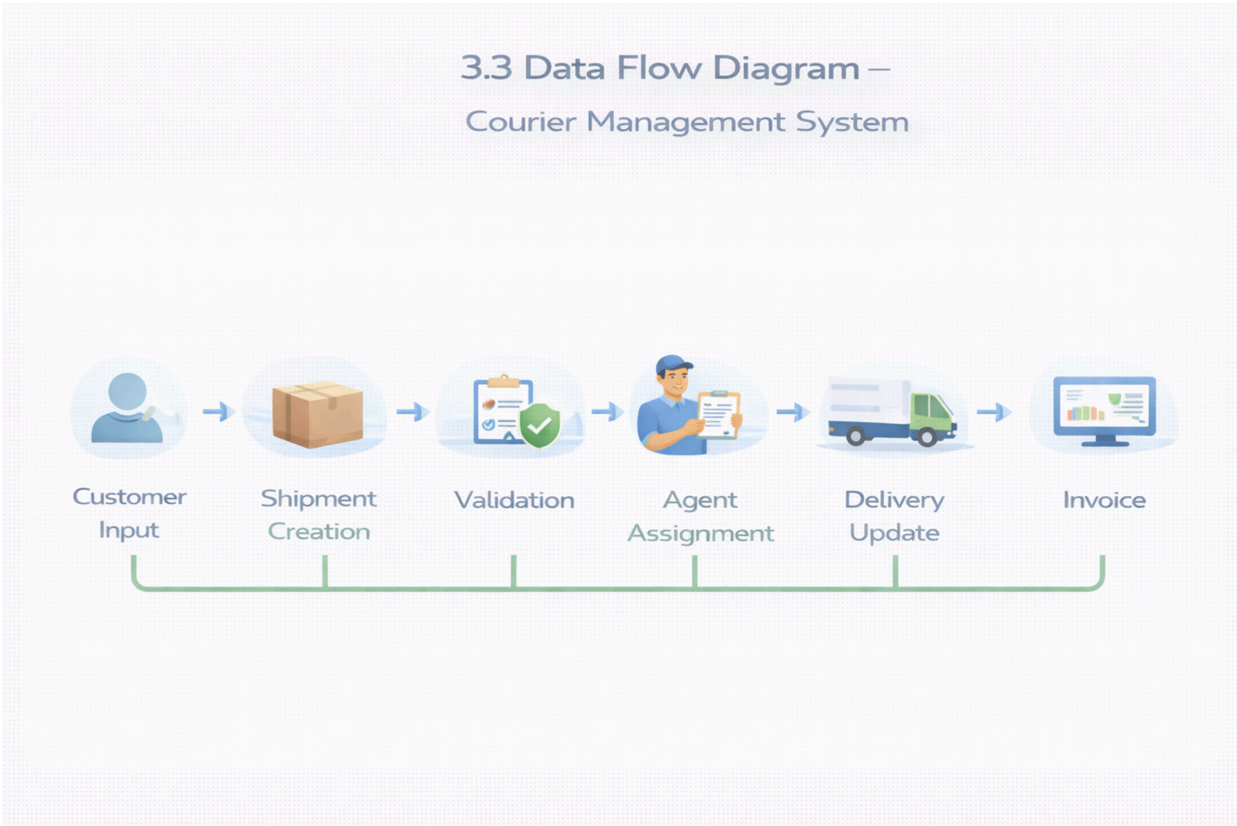
Non-Functional Requirements

NFR	Description
Usability	Easy Lightning interface
Security	Role-based access control
Reliability	Error-free automation
Performance	Fast tracking updates
Availability	24/7 cloud access
Scalability	Supports growing branches

3.3 Data Flow Diagram

Flow:

Customer Input → Shipment Creation → Validation → Agent Assignment → Delivery Update → Invoice → Reports



3.4 Technology Stack

Layer	**Technology**
UI	Salesforce Lightning
Logic	Salesforce Flows
Database	Salesforce Objects
Automation	Validation Rules, Flows
Security	Profiles, Roles
Reporting	Reports & Dashboards

4. PROJECT DESIGN PHASE

4.1 Problem–Solution Fit

CMS eliminates fragmented courier operations by automating shipment tracking, delivery assignment, billing, and reporting within a centralized Salesforce platform.

4.2 Proposed Solution

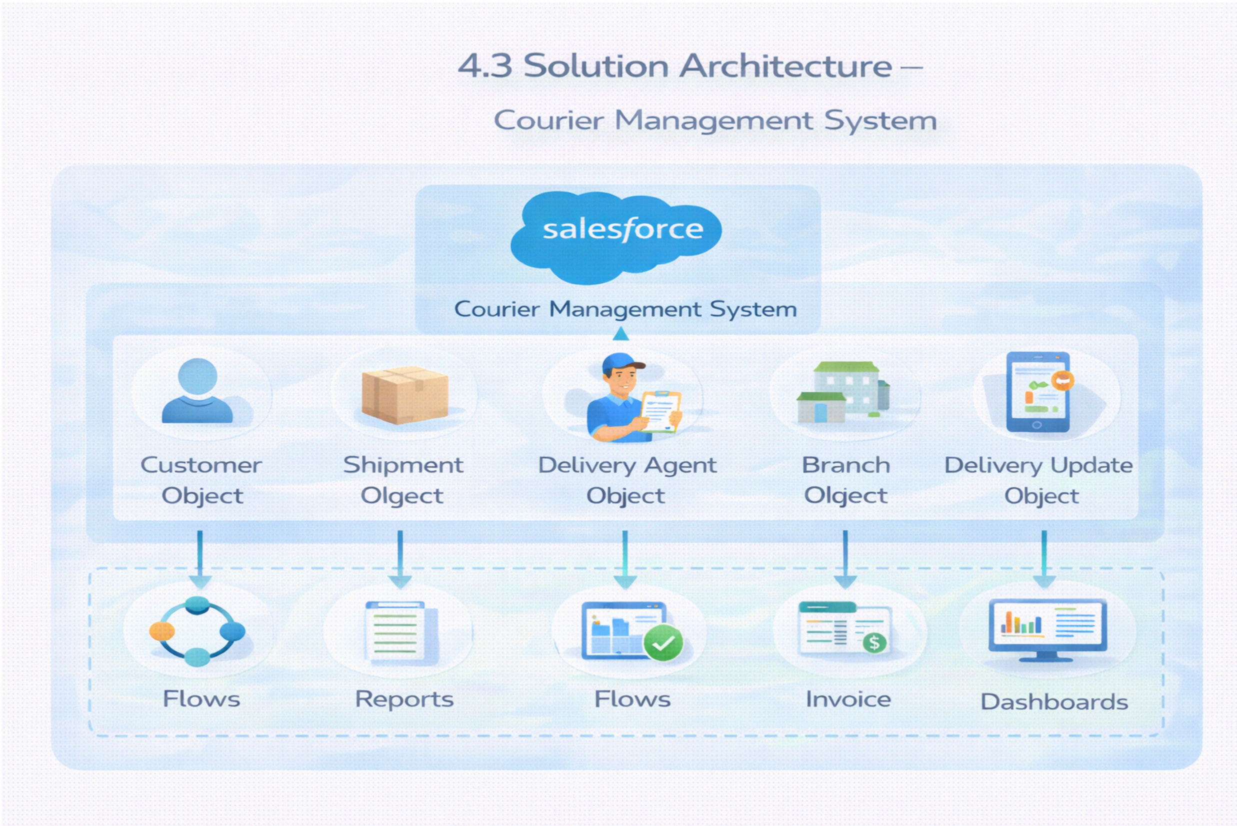
Functional Requirements

Parameter	Description
Problem	Manual shipment management
Solution	Salesforce CMS
Innovation	Automated tracking & billing
Impact	Faster deliveries
Business Model	Courier services
Scalability	Multi-branch ready

4.3 Solution Architecture

Components:

- Customer Object
- Shipment Object
- Delivery Agent Object
- Branch Object
- Delivery Update Object
- Invoice Object
- Flows
- Reports
- Dashboards



5. PROJECT PLANNING & SCHEDULING

5.1 Project Planning

- Agile methodology
- Epics → Stories → Story Points
- Sprint-based execution
- Velocity calculation used for estimation

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

Use the below template to create product backlog and sprint schedule

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Developer Setup	USN-1	As a system administrator, I want to create and configure a Salesforce developer account so that I can set up and deploy the Courier Management System.	3	High	Member1
Sprint-2	Data Modeling	USN-2	As an admin, I want to create Customer, Shipment, Delivery	5	High	Member2

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
			Agent, Branch, and Invoice objects with proper relationships so that courier data is stored and managed efficiently.			
Sprint-2	Data Modeling	USN-3	As a staff member, I want tabs and a Lightning App for all courier modules so that I can easily navigate and perform daily operations.	5	High	Member3
Sprint-3	Automation	USN-4	As a system, I want to validate shipment details such as delivery dates, addresses, and mandatory fields so that incorrect or incomplete records are prevented.	3	High	Member4
Sprint-3	Automation	USN-5	As a dispatcher, I want automated flows for shipment status updates and agent assignment so that deliveries are processed faster without manual intervention.	3	High	Member4
Sprint-4	Security	USN-6	As an administrator, I want role-based access control for managers, agents, and staff so that sensitive shipment and billing data is secure.	5	High	Member2
Sprint-5	Reports	USN-7	As a branch manager, I want reports showing delivery performance, shipment status, and revenue so that I can monitor operations and make informed decisions.	4	High	Member3
Sprint-6	Dashboards	USN-8	As a branch manager, I want dashboards showing delivery performance, shipment status, and revenue so that I can monitor operations and make informed decisions.	4	Medium	Member4

Project Tracker, Velocity & Burndown Chart: (4 Marks)

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022		
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022		
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022		

6. Project Development Phase:

What you'll learn

1. Real-time Salesforce project implementation
2. Designing custom objects and relationships
3. Creating page layouts & record types
4. Writing validation rules
5. Building automation with flows
6. Developing reports & dashboards
7. Setting up security with profiles, permission sets, and sharing
8. End-to-end application testing

Milestone 1: Salesforce Account

Introduction:

New to Salesforce and unsure where to start? Curious how real business applications—like loan processing and payment tracking—can be built on the platform? You're in the right place.

Welcome to the Loan Management System project!

Salesforce offers powerful tools that help automate processes, improve accuracy, and manage customer operations with ease. In this project, you'll learn how Salesforce can streamline lending workflows—from borrower onboarding to loan approval, EMI tracking, and document verification.

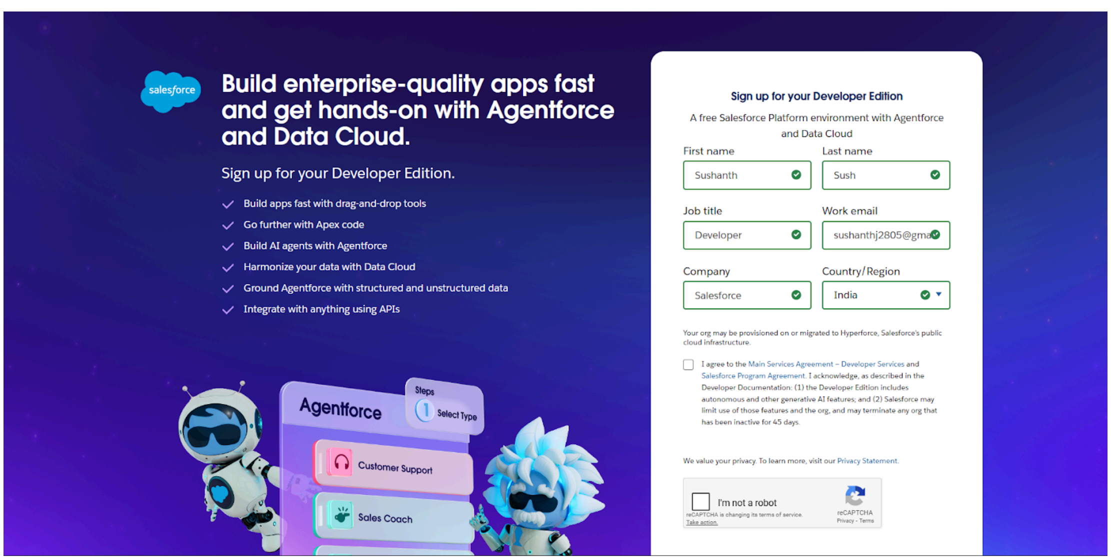
What Is Salesforce ?

Salesforce is your customer success platform, designed to help you sell, service, market, analyze, and connect with your customers. Salesforce has everything you need to run your business from anywhere. Using standard products and features, you can manage relationships with prospects and customers, collaborate and engage with employees and partners, and store your data securely in the cloud.

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 :
3. First name & Last name
4. Email



1) First name & Last name

2) Email

3) Job Title: Developer

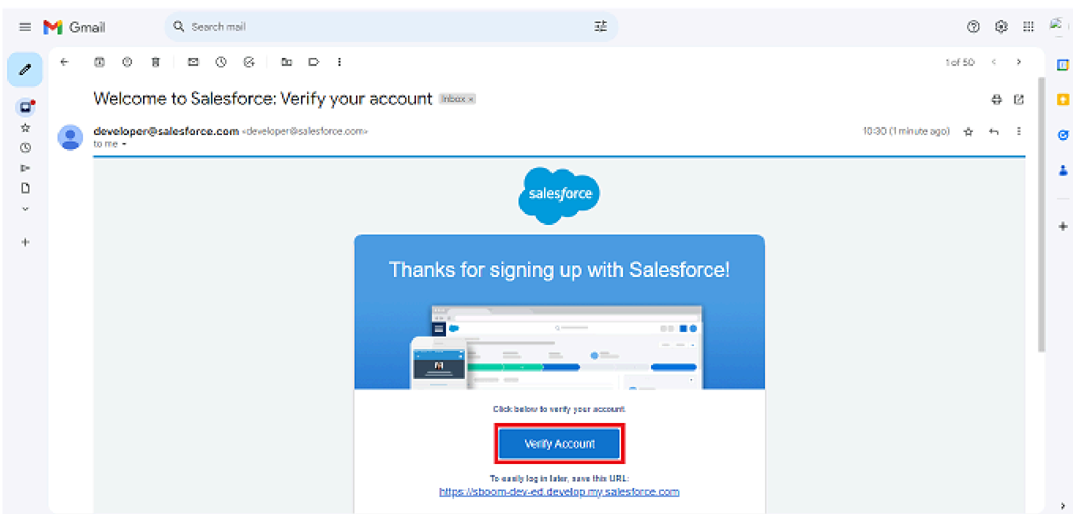
4) Company: College Name

5) Country: India

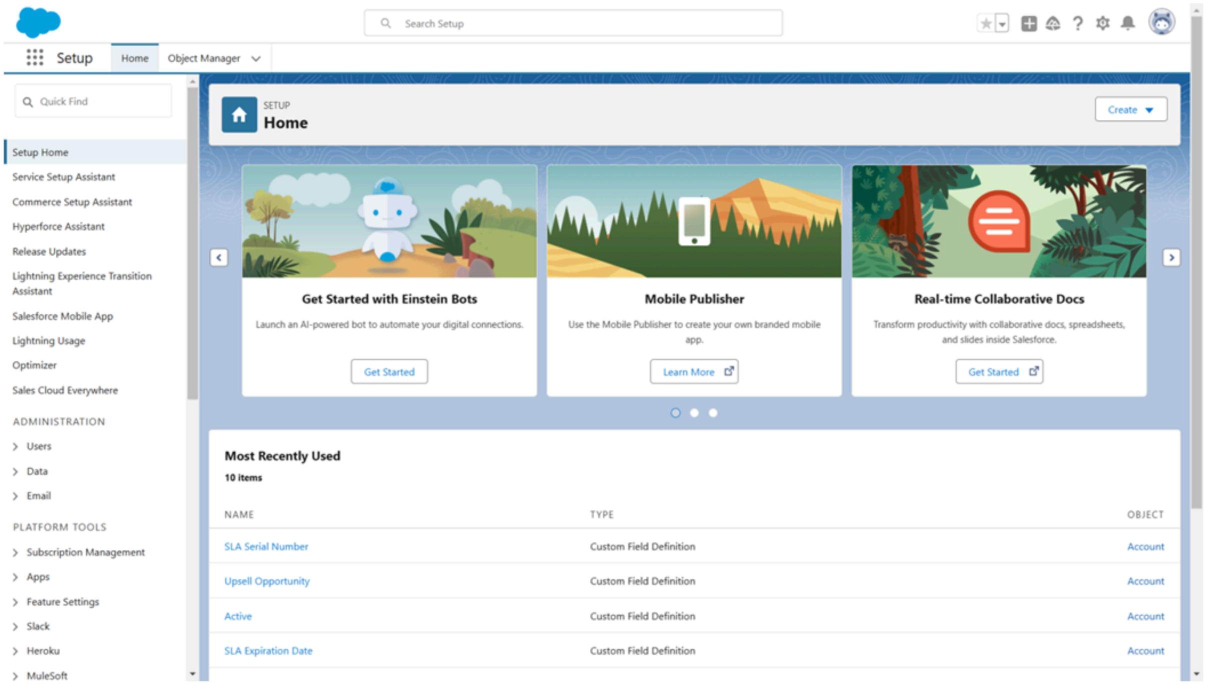
Click on sign me up after filling these.

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 10-30 mins and sometimes 2 hours.



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.



Milestone 2- Object:

What is an object?

Salesforce objects are database tables that permit you to store data that is specific to an organization. It consists of fields (columns) and records (rows).

Custom Objects Created:

1. **Customer__c** – stores customer personal and contact information
2. **Shipment__c** – manages parcel details, source, destination, and tracking information
3. **Delivery_Agent__c** – maintains delivery agent profiles and branch details
4. **Branch__c** – represents courier branch offices

5. **Delivery_Update__c** – stores status updates for each shipment

6. **Invoice__c** – manages billing details for delivered shipments

Relationships:

Customer → Shipment (Lookup)

Shipment → Delivery_Agent (Lookup)

Delivery_Agent → Branch (Lookup)

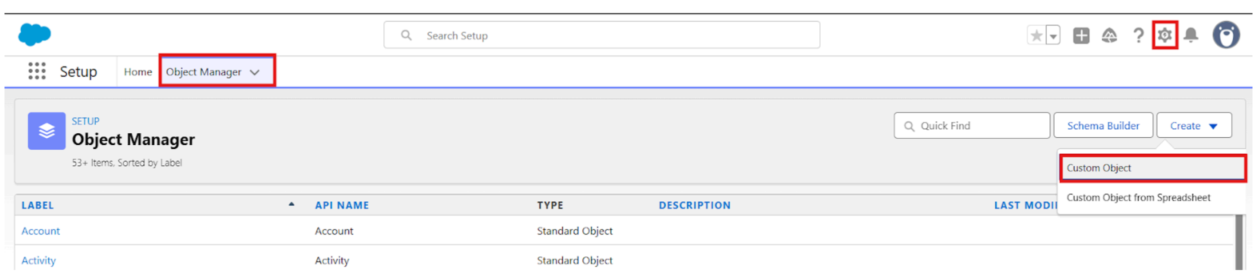
Delivery_Update → Shipment (Master-Detail)

Invoice → Shipment (Lookup)

Steps to Implement :

Activity 1: Create Customer Object

1. Click Setup → Object Manager → Create → Custom Object



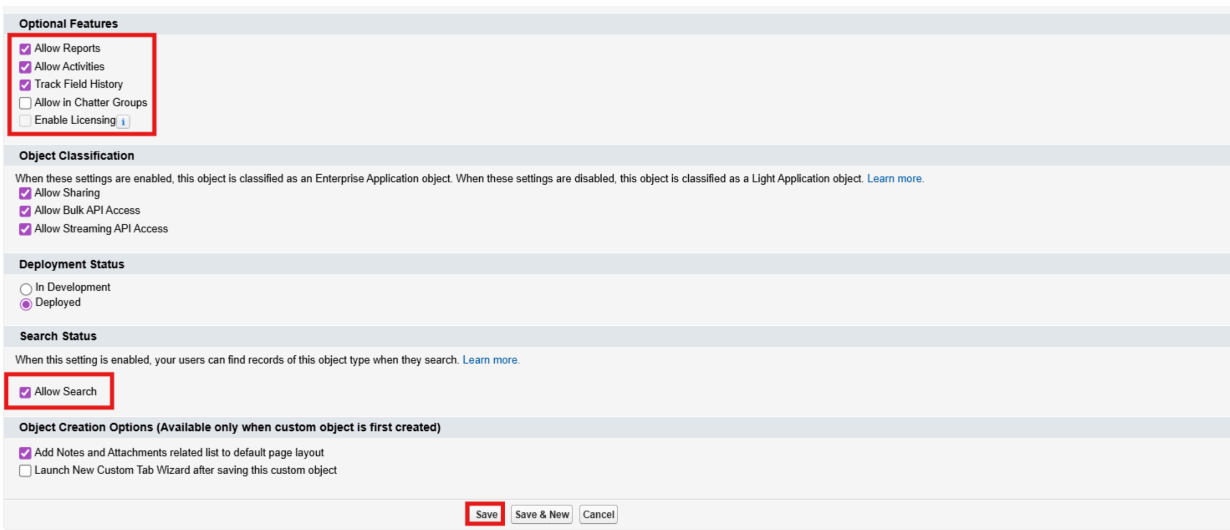
2. Enter the following details:

- Label: Customer
- Plural Label: Customers
- Record Name: Customer Name (Text)

3. Under Optional Features:

- Enable Allow Reports
- Enable Track Field History
- In Search Status check the box Allow Search
- In Object Creation Options check the box Add Notes and Attachments related list to default page layout

4. Click Save.



Activity 2: Create Shipment Object

1. Go to Setup → Object Manager → Create → Custom Object
2. Enter:

- Label: Shipment
- Plural Label: Shipments
- Record Name: Tracking Number (Auto Number)

3. Example Format: SHP-{00000}

   Starting Number: 0

4. Enable:

- Allow Reports
- Track Field History

5. Click Save

Activity 3: Create Delivery_Agent Object

1. Setup → Object Manager → Create → Custom Object
2. Enter:

- Label: Delivery Agent
- Plural Label: Delivery Agents
- Record Name: Agent Name (Text)

3. Enable:

- Allow Reports
- Track Field History

4. Click Save

Activity 4: Create Branch Object

1. Setup → Object Manager → Create → Custom Object
2. Enter:

- Label: Branch

- Plural Label: Branches

- Record Name: Branch Name (Text)

3. Enable:
- Allow Reports

4. Click Save.

Activity 5: Create Delivery_Update Object

1. Setup → Object Manager → Create → Custom Object

2. Enter:
- Label: Delivery Update

- Plural Label: Delivery Updates

- Record Name: Update Number (Auto Number)

3. Example Format: UPD-{0000}
 Enter Starting number : 0
4. Enable:
- Allow Reports

- Track Field History

5. Click Save.

Activity 6: Create Invoice Object

1. Setup → Object Manager → Create → Custom Object
2. Enter:

- Label: Invoice
- Plural Label: Invoices
- Record Name: Invoice Number (Auto Number)

3. Example Format: INV-{0000}
   Enter Starting number: 0
4. Enable:

- Allow Reports
- Track Field History

5. Click Save

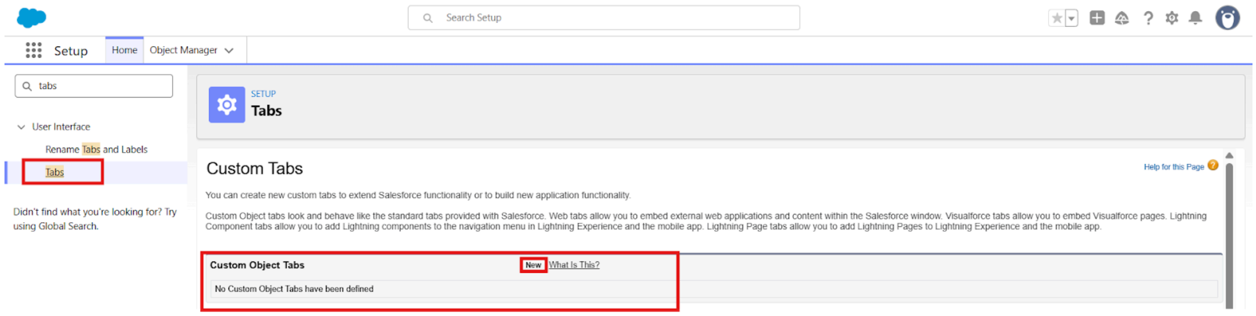
Milestone 3 - Tabs

Tab:

Tabs in Salesforce help users view the information at a glance. It displays the data of objects and other web content in the application.

Activity 1: Creating a Tab for Customer Object

1. Click Home, type Tabs in Quick Find, and select Tabs.
2. Under Custom Object Tabs, click New.



3. For Object, select Customer.

4. For Tab Style, choose any icon.

5. Leave all default settings as they are.

6. Click Next, Next, and Save.

Activity 2: Creating a Tab for Shipment Object

1. Click Home, type Tabs in Quick Find, and select Tabs.

2. Under Custom Object Tabs, click New.

3. For Object, select Shipment.

4. Choose any Tab Style.

5. Keep defaults unchanged.

6. Click Next, Next, and Save.

Activity 3: Creating a Tab for Delivery Agent Object

1. Click Home, type Tabs in Quick Find, and select Tabs.

2. Under Custom Object Tabs, click New.

3. For Object, select Delivery Agent.

4. Pick any Tab Style.
5. Leave default settings.
6. Click Next, Next, and Save.

Activity 4: Creating a Tab for Branch Object

1. Click Home, type Tabs in Quick Find, and select Tabs.
2. Under Custom Object Tabs, click New.
3. For Object, select Branch.
4. Choose any Tab Style.
5. Keep all defaults.
6. Click Next, Next, and Save.

Activity 5: Creating a Tab for Delivery Update Object

1. Click Home, type Tabs in Quick Find, and select Tabs.
2. Under Custom Object Tabs, click New.
3. For Object, select Delivery Update.
4. Select any Tab Style.
5. Keep default configurations.
6. Click Next, Next, and Save.

Activity 6: Creating a Tab for Invoice Object

1. Click Home, type Tabs in Quick Find, and select Tabs.
2. Under Custom Object Tabs, click New.
3. For Object, select Invoice.
4. Select any Tab Style.
5. Leave defaults unchanged.
6. Click Next, Next, and Save.

Milestone 4: Fields & Relationships

What are fields ?

Fields in Salesforce represent what the columns represent in relational databases. It can store data values which are required for a particular object in a record.

Activity 1: Creation of fields for the Customer Object

Field Name	API Name	Data Type	Required	Description
Customer Name	Name	Text	Yes	Full name of customer
Email	Email__c	Email	Yes	Customer email
Phone	Phone__c	Phone	Yes	Customer contact number
Address	Address__c	Text Area	Yes	Customer address

Customer Type	Customer_Type__c	Picklist	Yes	Regular / Business
City	City__c	Text	No	Customer city
State	State__c	Text	No	Customer state
Pincode	Pincode__c	Text	No	Postal code
Is Active	Is_Active__c	Checkbox	No	Active status

1. Email (Email Field)

- Go to Setup → Object Manager → Customer__c
- Click Fields & Relationships → New
- Select Email → Next
- Field Label: Email
- Check the Required box
- Click Next → Next → Save

2. Phone (Phone Field)

- Setup → Object Manager → Customer__c
- Fields & Relationships → New
- Select Phone → Next
- Field Label: Phone
- Check Required
- Click Next → Next → Save

3. Address (Text Area)

- Setup → Object Manager → Customer__c

- Fields & Relationships → New
- Select Textarea → Next
- Field Label: Address
- Check Required
- Save

4. Customer Type (Picklist)

- Setup → Customer__c → Fields → New
- Picklist → Next
- Field Label: Customer Type
- Enter values:
    - Regular
    - Business
- Check Required
- Save

5. City (Text)

- Setup → Customer__c → Fields → New
- Text → Next
- Field Label: **City**
- Save

6. State (Text)

- Setup → Customer__c → Fields → New
- Text → Next
- Field Label: **State**
- Save

7. Pincode (Text)

- Setup → Customer__c → Fields → New
- Text → Next
- Field Label: **Pincode**
- Save

8. Is Active (Checkbox)

- Setup → Customer__c → Fields → New
- Checkbox → Next
- Field Label: **Is Active**
- Save

Activity 2: Creation of fields for the Shipment Object

Field Name	API Name	Data Type	Required	Description
Tracking Number	Name	Auto Number	Yes	Unique ID

Customer	Customer__c	Lookup(Customer)	Yes	Related customer
Source Address	Source_Address__c	Text Area	Yes	Pickup location
Destination Address	Destination_Address__c	Text Area	Yes	Delivery location
Weight	Weight__c	Number	Yes	Parcel weight
Price	Price__c	Currency	Yes	Shipping charge
Status	Status__c	Picklist	Yes	Shipment status
Expected Date	Expected_Date__c	Date	Yes	Delivery estimate
Assigned Agent	Assigned_Agent__c	Lookup(Delivery_Agent)	No	Delivery agent
Shipment Type	Shipment_Type__c	Picklist	Yes	Domestic/International

1. Source Address (Text Area)

- Setup → Shipment__c → Fields → New
- Textarea → Next
- Field Label: **Source Address**
- Required → Save

2. Destination Address (Text Area)

- Setup → Shipment__c → Fields → New
- Textarea → Next

- Field Label: **Destination Address**
- Required → Save

3. Weight (Number)

- Setup → Shipment__c → Fields → New
- Number → Next
- Field Label: **Weight**
- Required → Save

4. Price (Currency)

- Setup → Shipment__c → Fields → New
- Currency → Next
- Field Label: **Price**
- Required → Save

5. Status (Picklist)

Values:

- Booked
- In Transit
- Out for Delivery
- Delivered

- Cancelled

Required → Save

6. Expected Date (Date)

- Setup → Shipment__c → Fields → New
- Date → Next
- Field Label: **Expected Date**
- Required → Save

7. Shipment Type (Picklist)

Values:

- Domestic
- International

Required → Save

Activity 3: Creation of fields for Delivery_Agent Object

Field Name	API Name	Data Type	Required
Agent Name	Name	Text	Yes
Email	Email__c	Email	Yes
Phone	Phone__c	Phone	Yes
Branch	Branch__c	Lookup	Yes

Status	Status__c	Picklist	Yes
Joining Date	Joining_Date__c	Date	No

1. Email

- Setup → Delivery_Agent__c → Fields → New
- Email → Next
- Field Label: Email
- Required → Save

2. Phone

- Setup → Delivery_Agent__c → Fields → New
- Phone → Next
- Field Label: Phone
- Required → Save

3. Status (Picklist)

Values:

- Active
- Inactive
- On Leave

Required → Save

4. Joining Date

- Setup → Delivery_Agent__c → Fields → New
- Date → Next
- Field Label: Joining Date
- Save

Activity 4: Creation of fields for Branch Object

Field Name	API Name	Data Type	Required
Branch Name	Name	Text	Yes
Branch Code	Branch_Code__c	Text	Yes
City	City__c	Text	Yes
State	State__c	Text	Yes
Manager Name	Manager_Name__c	Text	No

Create each field using:

- Setup → Branch__c → Fields → New
- Select respective Data Type
- Enter Field Label
- Save

(Repeat for all fields)

Activity 5: Creation of fields for Delivery_Update Object

Field Name	API Name	Data Type	Required

Update Number	Name	Auto Number	Yes
Shipment	Shipment__c	Master-Detail	Yes
Status	Status__c	Picklist	Yes
Update Date	Update_Date__c	DateTime	Yes
Remarks	Remarks__c	Text	No

Status Picklist Values:

- In Transit
- Out for Delivery
- Delivered
- Delayed

Required → Save

Activity 6: Creation of fields for Invoice Object

Field Name	API Name	Data Type	Required
Invoice Number	Name	Auto Number	Yes
Shipment	Shipment__c	Lookup	Yes
Amount	Amount__c	Currency	Yes
Invoice Date	Invoice_Date__c	Date	Yes
Payment Status	Payment_Status__c	Picklist	Yes

Payment Status Values

- Paid
- Unpaid
- Pending

Required → Save

Relationships :

Child Object	Field	Parent Object	Type
Shipment	Customer__c	Customer	Lookup
Shipment	Assigned_Agent__c	Delivery_Agent	Lookup
Delivery_Agent	Branch__c	Branch	Lookup
Delivery_Update	Shipment__c	Shipment	Master-Detail
Invoice	Shipment__c	Shipment	Lookup

Relationship Creation Steps :

Activity 1 – Shipment → Customer

- Setup → Shipment__c → Fields → New
- Lookup Relationship
- Related Object: **Customer__c**
- Field Label: **Customer**
- Save

Activity 2 – Shipment → Delivery Agent

- Setup → Shipment__c
- New → Lookup
- Object: **Delivery_Agent__c**
- Label: **Assigned Agent**
- Save

Activity 3 – Delivery Agent → Branch

- Setup → Delivery_Agent__c
- New → Lookup
- Related Object: **Branch__c**
- Label: **Branch**
- Save

Activity 4 – Delivery Update → Shipment

- Setup → Delivery_Update__c
- New → **Master-Detail**
- Related Object: **Shipment__c**
- Save

Activity 5 – Invoice → Shipment

- Setup → Invoice__c

- New → Lookup
- Object: **Shipment__c**
- Save

Milestone 5-Validation Rules :

Validation rules are applied when a user tries to save a record and are used to check if the data meets specified criteria. If the criteria are not met, the validation rule triggers an error message and prevents the user from saving the record until the issues are resolved.

Activity 1: Create Validation Rules on Customer__c

1. Validation Rule: Email Must Not Be Blank

Steps:

- Click the Gear icon → Setup
- Go to Object Manager
- Search and select Customer__c
- From the left menu, click Validation Rules
- Click New
- Rule Name: Email_Required
- Check Active

In Error Condition Formula, paste:

ISBLANK(Email__c)

- Click Check Syntax
- Error Message: Email is required.
- Error Location: Field → Email
- Click Save.

2. Validation Rule: Phone Must Be 10 Digits

Steps:

- Setup → Object Manager → Customer__c
- Validation Rules → New
- Rule Name: Validate_Phone_Number
- Check Active

Formula:

LEN(Phone__c) <> 10

- Check Syntax
- Error Message: **Phone number must be 10 digits.**
- Error Location: Field → Phone
- Save

3. Validation Rule: Pincode Must Be 6 Digits

Steps:

- Setup → Object Manager → Customer__c
- Validation Rules → New
- Rule Name: Validate_Pincode
- Active → Yes

Formula:

LEN(Pincode__c) <> 6

- Error Message: **Pincode must be 6 digits.**
- Location: Field → Pincode
- Save

Activity 2: Create Validation Rules on Shipment__c

1. Validation Rule: Weight Must Be Greater Than Zero

Steps:

- Setup → Object Manager → Shipment__c
- Click Validation Rules → New
- Rule Name: Validate_Weight
- Active → Checked

Formula:

Weight__c <= 0

- Check Syntax

- Error Message: **Weight must be greater than zero.**
- Location: Field → Weight
- Save

2. Validation Rule: Price Must Be Greater Than Zero

Steps:

- Setup → Object Manager → Shipment__c
- Validation Rules → New
- Rule Name: Validate_Price
- Active → Yes

Formula:

Price__c <= 0

- Error Message: **Price must be greater than zero.**
- Location: Field → Price
- Save

3. Validation Rule: Expected Date Cannot Be in the Past

Steps:

- Setup → Object Manager → Shipment__c
- Validation Rules → New
- Rule Name: Validate_Expected_Date

Formula:

Expected_Date__c < TODAY()

- Check Syntax
- Error: **Expected delivery date cannot be in the past.**
- Location: Field → Expected Date
- Save

Activity 3: Create Validation Rules on Delivery_Agent__c

1. Validation Rule: Email Must Be Valid

Steps:

- Setup → Object Manager → Delivery_Agent__c
- Validation Rules → New
- Rule Name: Validate_Agent_Email
- Active → Yes

Formula:

NOT(CONTAINS(Email__c, "@"))

- Error Message: **Enter a valid email address.**
- Location: Field → Email

- Save

Activity 4: Create Validation Rules on Branch__c

1. Validation Rule: Branch Code Must Not Be Blank

Steps:

- Setup → Object Manager → Branch__c
- Validation Rules → New
- Rule Name: Branch_Code_Required

Formula:

ISBLANK(Branch_Code__c)

- Error Message: **Branch Code is mandatory.**
- Location: Field → Branch Code
- Save

Activity 5: Create Validation Rules on Delivery_Update__c

1. Validation Rule: Remarks Required If Status is Delayed

Steps:

- Setup → Object Manager → Delivery_Update__c
- Validation Rules → New
- Rule Name: Remarks_Required_For_Delay

Formula:

```
AND(
 ISPICKVAL(Status__c, "Delayed"),
 ISBLANK(Remarks__c)
)
```

- Error Message: **Remarks are required when status is Delayed.**
- Location: Field → Remarks
- Save

Milestone 6 - Page Layouts :

Page Layouts in Salesforce control what a user sees on a record page and how information is organized.
They help structure fields, sections, buttons, related lists, and other components in a clean and user-friendly way.
By customizing page layouts, we ensure that users can easily enter, update, and view the required data.

Activity 1: Edit Page Layout for Customer__c

Steps:

- Setup → Object Manager → Customer__c
- Click Page Layouts
- Click the default layout (usually Customer Layout)

Sections to Add & Arrange:

- Open the default layout

- In the top palette, click “Section”
- Drag the Section and drop it anywhere on the layout

A pop-up appears → Fill details:

- Section Name: “Customer Information”
- Layout:
    - 1-Column or
    - 2-Column (recommended)
- Tab-Key Order: Top-Down
- Click OK

The new section will appear on the layout.

- Drop fields inside the section (it will highlight).
- Repeat the same for each field.

Section 1: Customer Information

- Customer Name
- Customer Type
- Email
- Phone
- Is Active

Section 2: Address Details

- Address
- City

- State
- Pincode

9. Edit the layout as needed
10. Click Save.

Activity 2: Edit Page Layout for Shipment__c

Steps:

- Setup → Object Manager → Shipment__c
- Click Page Layouts
- Open the default Shipment Layout

Sections to Add:

Section 1: Shipment Details

- Tracking Number
- Customer
- Shipment Type
- Weight
- Price
- Status

Section 2: Address Information

- Source Address
- Destination Address

Section 3: Delivery Information

- Expected Date
- Assigned Agent
- Arrange fields properly
- Click Save

Activity 3: Edit Page Layout for Delivery_Agent__c

Steps:

- Setup → Object Manager → Delivery_Agent__c
- Page Layouts
- Open default Delivery Agent Layout

Arrange Fields:

Agent Details

- Agent Name
- Email
- Phone
- Branch
- Status

- Joining Date

Additional Information

- Related Shipments
- Related Delivery Updates
- Edit → Save

Activity 4: Edit Page Layout for Branch__c

Steps:

- Setup → Object Manager → Branch__c
- Page Layouts
- Open default Branch Layout

Sections:

Branch Information

- Branch Name
- Branch Code
- City
- State

- Manager Name
- Arrange fields
- Click Save

Activity 5: Edit Page Layout for Delivery_Update__c

Steps:

- Setup → Object Manager → Delivery_Update__c
- Page Layouts
- Open default Delivery Update Layout

Sections:

Update Information

- Update Number
- Shipment
- Status
- Update Date
- Remarks
- Arrange properly
- Click **Save**

Activity 6: Edit Page Layout for Invoice__c

Steps:

- Setup → Object Manager → Invoice__c
- Page Layouts
- Open default Invoice Layout

Sections:

Invoice Details

- Invoice Number
- Shipment
- Amount
- Invoice Date
- Payment Status
- Arrange fields as needed
- Click Save

Milestone 7 - Flows

In Salesforce, a flow is a powerful tool that allows you to automate business processes, collect and update data, and guide users through a series of screens or steps. Flows are built using a visual interface and can be created without any coding knowledge.

Activity 1: Auto-Update Shipment Status When Delivery Update is Created

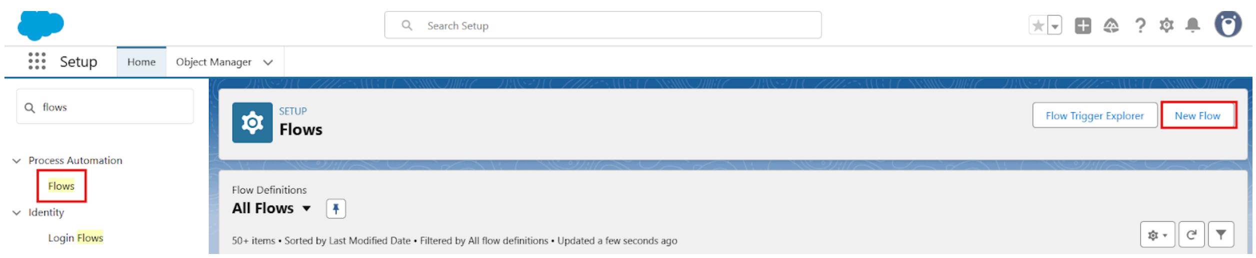
Scenario:

Whenever a Delivery_Update__c record is created, automatically update the related Shipment__c Status with the latest status from Delivery Update.

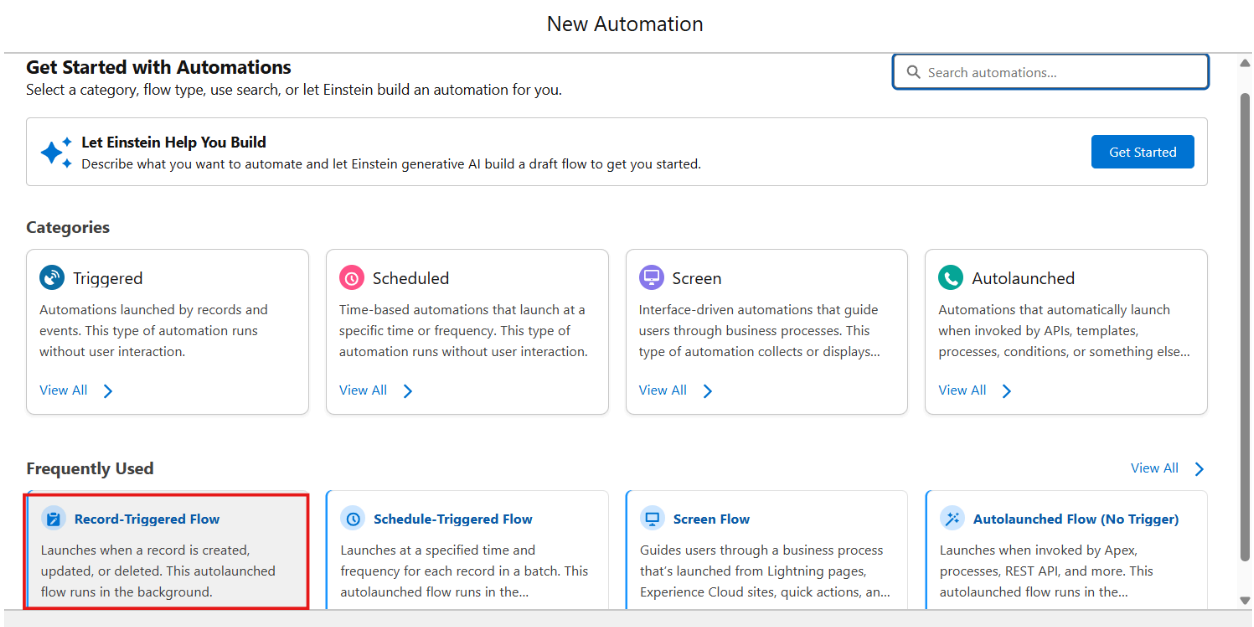
Part 1: Create the Record-Triggered Flow

Step 1: Open Flow Builder

- Click the Gear Icon → Click Setup
- In the Quick Find box (left side), type Flows
- Click Flows
- Click the New Flow button



- Select Record-Triggered Flow
- Click Create



Step 2: Configure the Start (Trigger)

On the Start panel on the right:

- Object: select Delivery_Update__c
- Trigger the Flow When:
    - Select A record is created
- Condition Requirements:
    - Choose: None (Always Run the Flow)
- Optimize the Flow For:
    - Select Actions and Related Records
- Click Done

Part 2: Update Related Shipment Record

Step 3: Add "Update Records" Element

- Click the + icon under the Start element
- Select Update Records

Configure the Update Records Element:

- Label: Update_Shipment_Status
- How to Find Records to Update:
    - Select Specify conditions to identify records

Set Filter Conditions:

- Object: Shipment__c

- Field: Id
- Operator: Equals
- Value: {!$Record.Shipment__c}

(This means update the Shipment linked to the Delivery Update)

Set Field Values to Update

Add field mapping:

Shipment Field	Value Source
Status__c	{!$Record.Status__c}

(This copies status from Delivery Update to Shipment)

- Click Done

Part 3: Save and Activate

Step 4: Save Flow

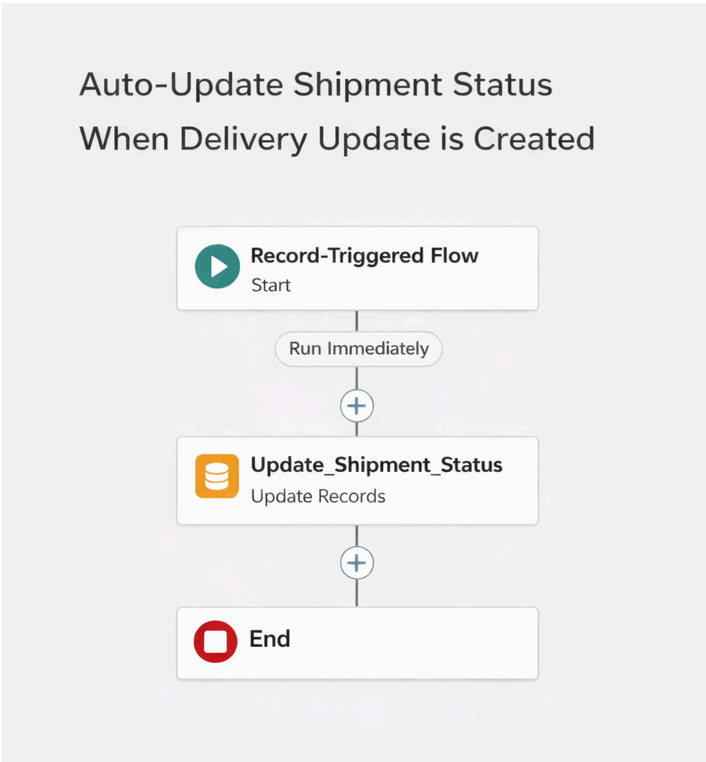
- Click Save (top-right)
- Flow Label: Auto_Update_Shipment_Status
- API Name will auto-fill
- Click Save

Step 5: Activate Flow

- Click Activate

- Flow status should show Active

Final Flow Layout:



Milestone 8 - The Lightning App :

What is an App ?

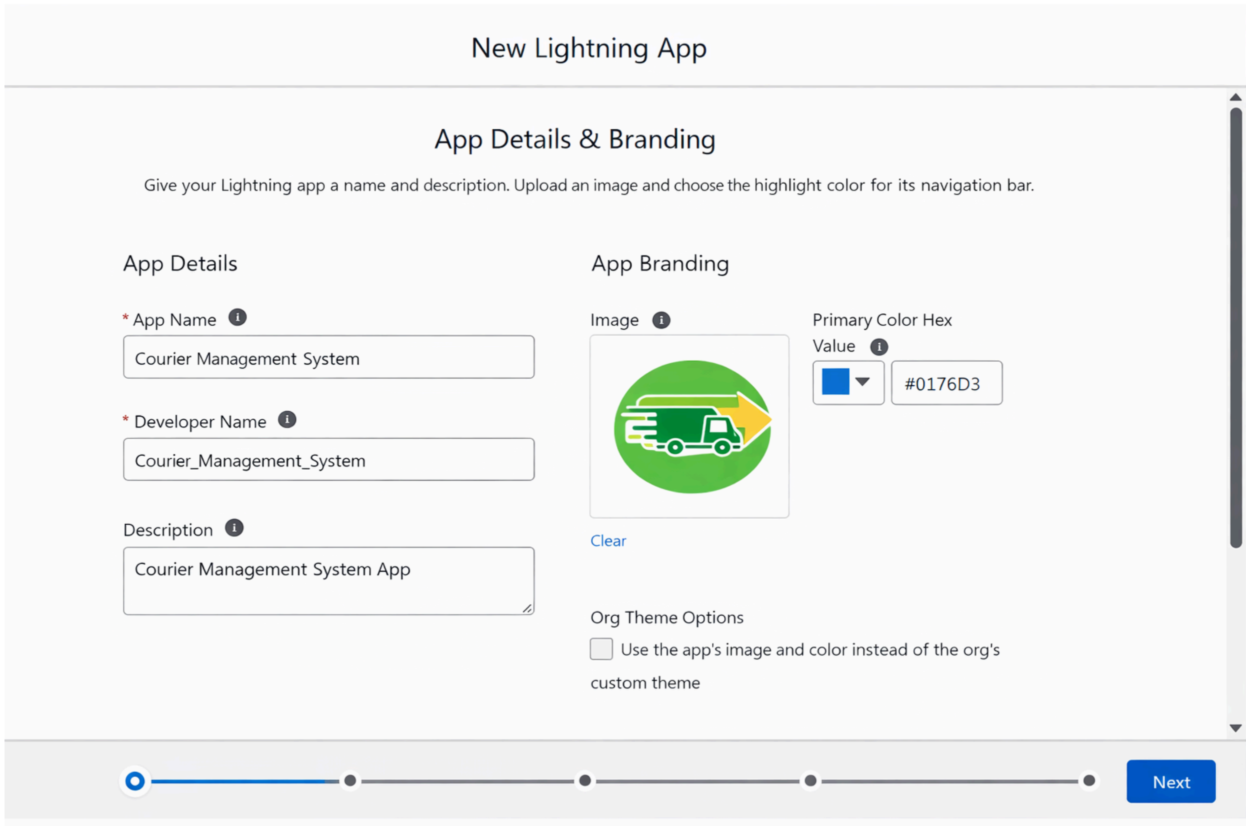
Apps in Salesforce are a group of tabs that help the application function by working together as a unit. It has a name, a logo, and a particular set of tabs. The simplest app usually has just two tabs.

Activity 1: Create a Lightning App

Create the Courier Management System App

1. From Setup, enter App Manager in the Quick Find and select App Manager.

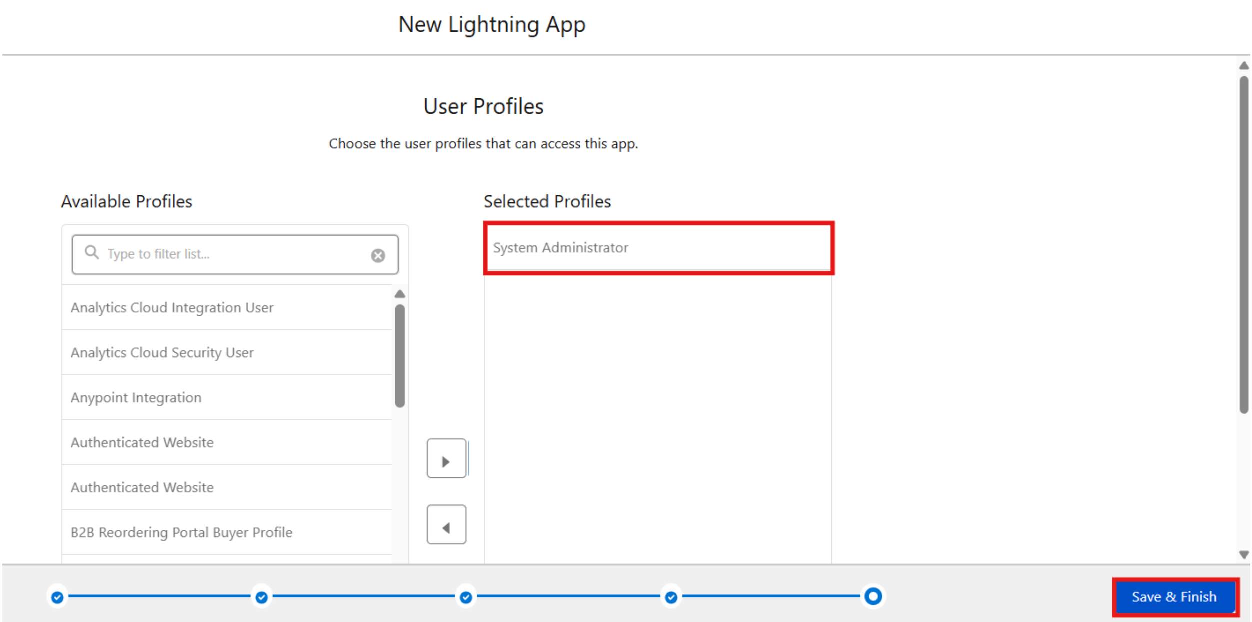
2. Click New Lightning App.
    - Enter the Courier Management System as the App Name.
3. Upload an image or icon for the app, then click Next.



4. Under App Options, leave the default selections and click Next.
5. Under Utility Items, leave as it is and click Next.
6. From Available Items, select the following and move them to Selected Items:
    - Customer
    - Shipment
    - Delivery Agent
    - Branch
    - Delivery Update

- Invoice
- Reports
- Dashboards

7. Click Next.
8. From Available Profiles, select System Administrator and move it to Selected Profiles.
9. Click Save & Finish.



Milestone 9- Profile:

What is a profile?

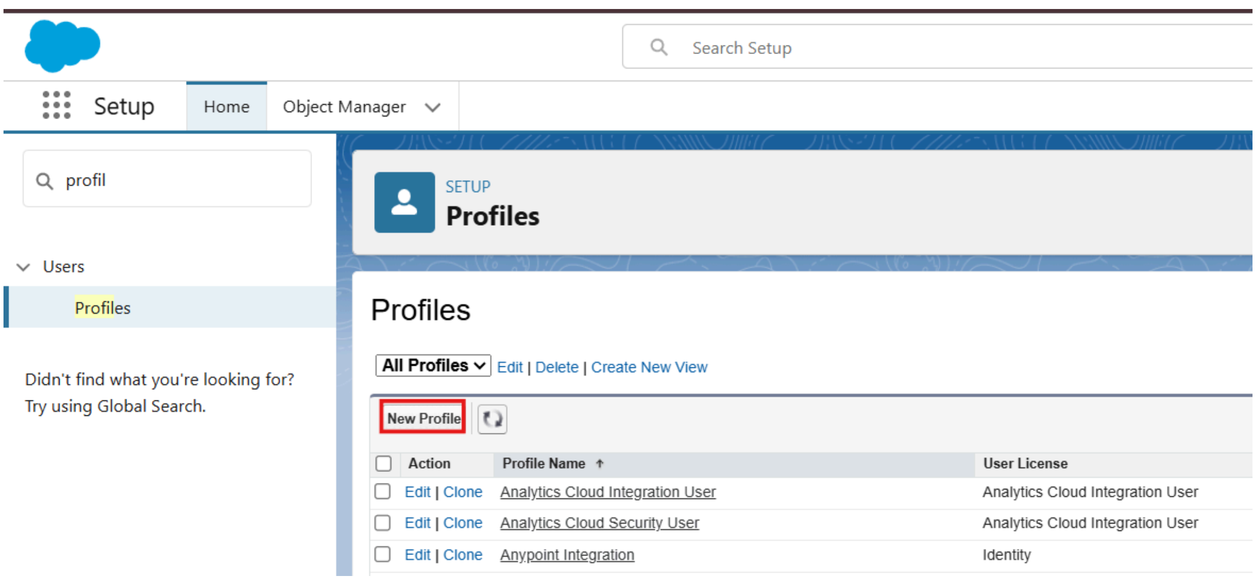
A profile is a group/collection of settings and permissions that define what a user can do in salesforce. A profile controls “Object permissions, Field permissions, User permissions, Tab settings, App settings, Apex class access, Visualforce page access, Page layouts, Record Types, Login hours & Login IP ranges.

A profile can be assigned to many users, but users can be assigned a single profile at a time.

Activity 1: Create a Custom Profile – Courier Admin

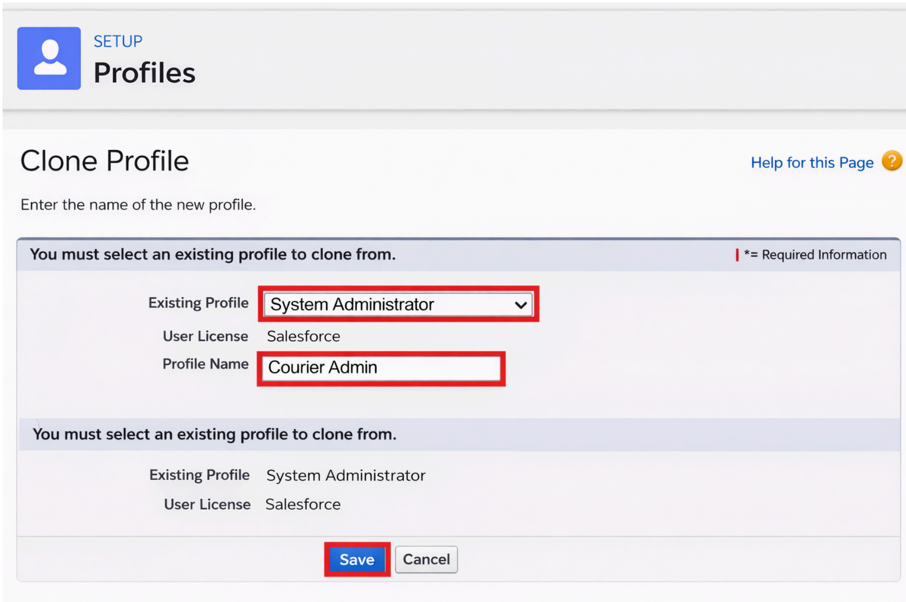
Create a Custom Profile:

1. From Setup, enter Profiles in the Quick Find box.
2. Click Profiles.
3. Click New Profile.



4. Select Existing Profile as: System Administrator.
5. Enter Profile Name: Courier Admin.

6. Click Save.



Why use System Administrator as Base?

It already has:

- Modify All Data
- Customize Application
- Full CRUD access on all objects

By cloning it, the **Courier Admin profile automatically inherits full access**.
You can later restrict permissions if required.

Activity 2: Create a Custom Profile – Branch Manager

Steps:

1. From Setup, enter Profiles in the Quick Find box.
2. Click Profiles.
3. Click New Profile.
4. Select Existing Profile as: Standard Platform User.

5. Enter Profile Name: Branch Manager.
6. Click Save.
7. Click Edit on the Branch Manager profile.

Set App & Tab Settings

Under Assigned Apps / Custom App Settings, set the default app to:

Courier Management System App

Ensure Visible Tabs Include:

Set these tabs as Default On or Visible:

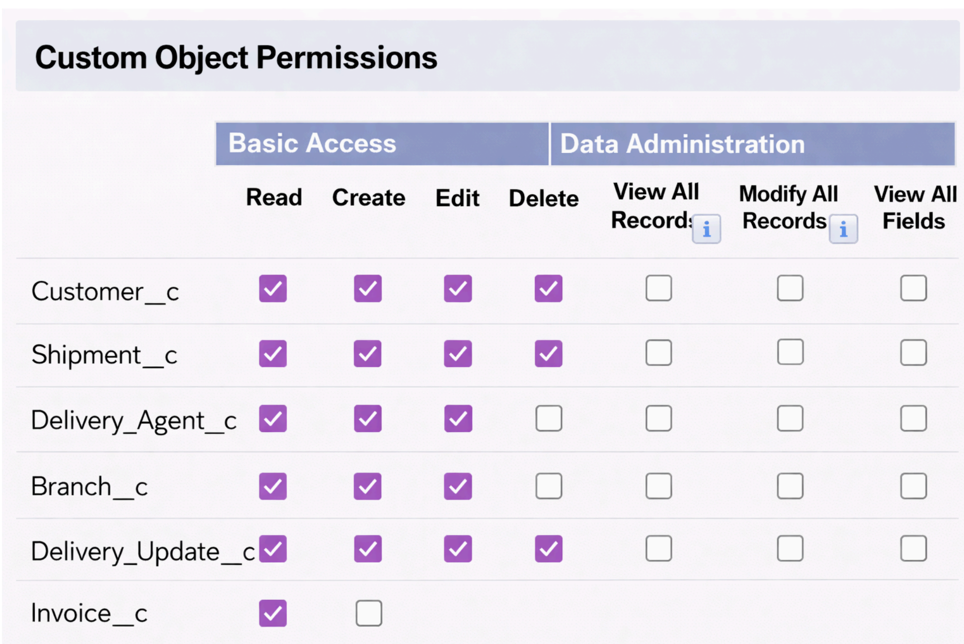
- Customers
- Shipments
- Delivery Agents
- Branches
- Delivery Updates
- Invoices
- Reports
- Dashboards

Custom Object Permissions

Scroll to Custom Object Permissions and set access as follows:

Object	Permissions to Grant
Customer__c	Create, Read, Edit

Shipment__c	Create, Read, Edit
Delivery_Agent__c	Read, Edit
Branch__c	Read, Edit
Delivery_Update__c	Create, Read, Edit
Invoice__c	Read



Custom Object Permissions

	Basic Access				Data Administration		
	Read	Create	Edit	Delete	View All Records	Modify All Records	View All Fields
Customer__c	☑	☑	☑	☑	☐	☐	☐
Shipment__c	☑	☑	☑	☑	☐	☐	☐
Delivery_Agent__c	☑	☑	☑	☐	☐	☐	☐
Branch__c	☑	☑	☑	☐	☐	☐	☐
Delivery_Update__c	☑	☑	☑	☑	☐	☐	☐
Invoice__c	☑	☐					

Session & Password Settings

Under Session Settings:

- Session Timeout: 2 Hours of Inactivity

Under Password Policies:

- User passwords expire in: Never Expires
- Minimum Password Length: 8

Click Save.

Activity 3: Create a Custom Profile – Delivery Agent

Steps:

1. Setup → Quick Find → Profiles
2. Click New Profile
3. Existing Profile: Standard Platform User
4. Profile Name: Delivery Agent
5. Click Save

App & Tab Access

Default App: Courier Management System

Visible Tabs:

- Shipments
- Delivery Updates

Object Permissions :

Object	Permissions
Shipment__c	Read, Edit
Delivery_Update__c	Create, Read, Edit
Customer__c	Read
Invoice__c	Read

Restrictions

- No access to Branch editing
- No delete permissions
- Limited reporting access

Click Save.

Activity 4: Create a Custom Profile – Customer Support

Steps:

- Setup → Profiles → New Profile
- Base Profile: Standard User
- Profile Name: Customer Support

Access Configuration

Tabs Visible:

- Customers
- Shipments
- Delivery Updates

Object Permissions :

Object	Access
Customer__c	Read, Edit
Shipment__c	Read, Edit
Delivery_Update__c	Read
Invoice__c	Read

Session Settings

- Session Timeout: 2 Hours
- Password Length: 8

Click Save

Final Profiles Created in Project :

Profile Name	Purpose

Courier Admin	Full system control
Branch Manager	Manage shipments and agents
Delivery Agent	Update delivery status
Customer Support	Handle customers and tracking

Milestone 10- Role:

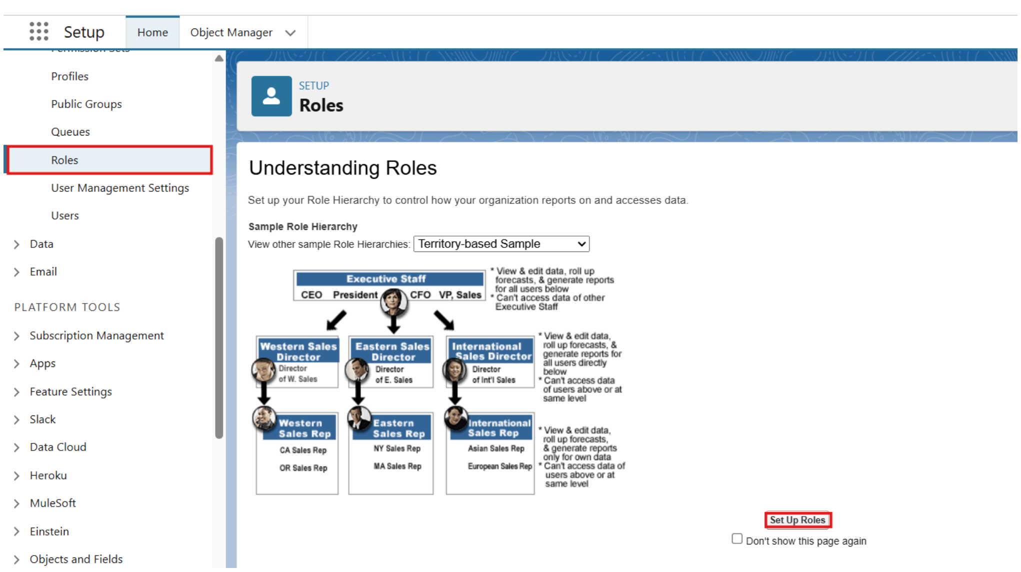
In Salesforce, roles are used to determine which users have access to certain data and functions within the system. They are also used to define the reporting hierarchy within an organization. Users with higher roles have greater access to data and more control over the system.

Activity 1: Creation of Courier Admin Role

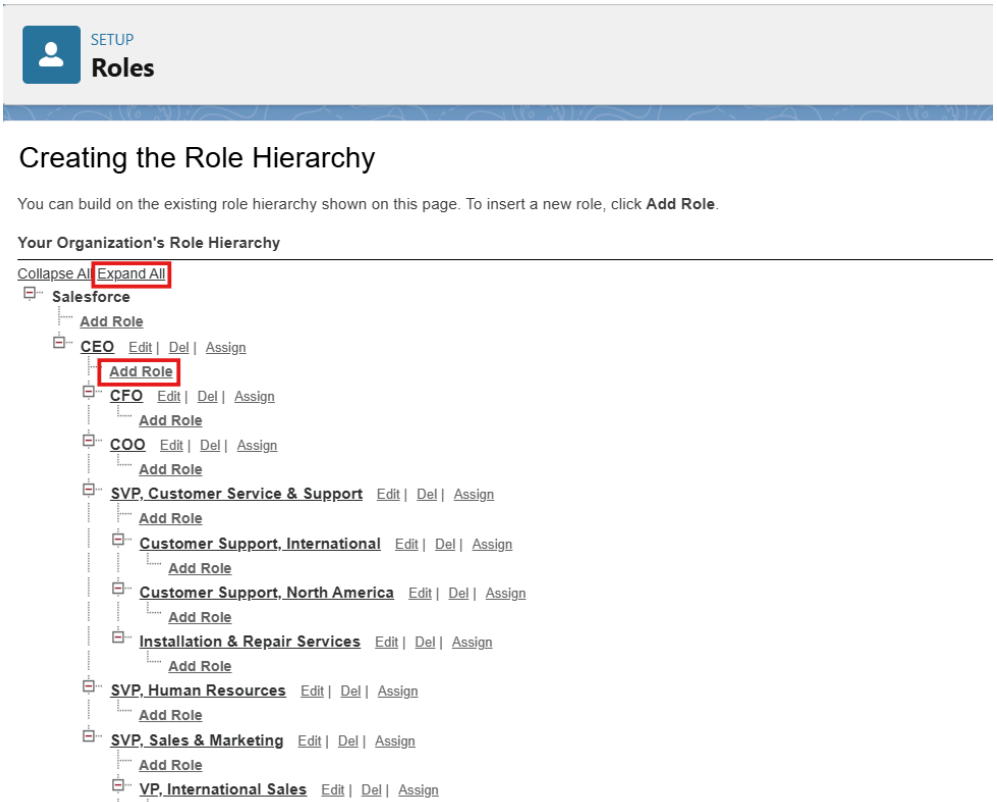
Creation of Role:

1. From the Quick Find box, search for Roles and click on the Roles option.

2. Select Set Up Roles.



3. Click on Expand All and click Add Role under CEO.



4. Give Label as “Courier Admin”.
   - The Role Name will auto-populate.
5. Click Save.

Activity 2: Creating Branch Manager Role Under Courier Admin Role

Create another role:

1. Create a role with the name “Branch Manager”.
2. Below the Courier Admin, click on Add Role.
3. Enter the label name as “Branch Manager” and the Role Name will automatically populate.

4. Click Save.

Activity 3: Creating Delivery Agent Role Under Branch Manager Role

Steps:

1. Below the Branch Manager role, click on Add Role.
2. Enter Label as “Delivery Agent”.
    - Role Name will auto-populate.
3. Click Save.

Activity 4: Creating Customer Support Role Under Branch Manager

Steps:

1. Under the Branch Manager role, click on Add Role.
2. Enter Label as “Customer Support”.
3. Role Name will automatically populate.
4. Click Save.

Purpose of This Hierarchy :

- **Courier Admin** – Highest level access to all data
- **Branch Manager** – Manages branch-level operations
- **Delivery Agent** – Access only to assigned shipments
- **Customer Support –** Limited access to customer and shipment tracking

This hierarchy ensures proper data sharing and security in the Courier Management System.

Milestone 11- User :

What is a user?

A user is anyone who logs in to Salesforce. Users are employees at your company, such as sales,reps, managers, and IT specialists, who need access to the company's records. Every user in Salesforce has a user account. The user account identifies the user, and the user account settings determine what features and records the user can access.

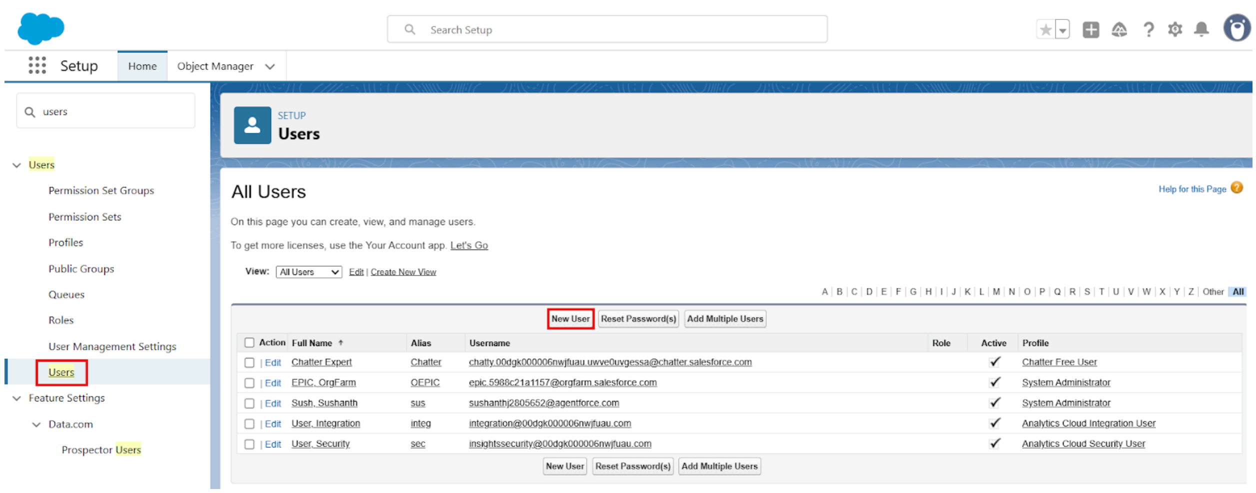
NOTE- As Salesforce license can only be used by 2 Users at a time in Dev Org, So If you don't find a salesforce license then deactivate a user who has a salesforce license Or change the license type from Salesforce to any other.

Activity 1: To Create a User

Steps:

1. From Setup, enter Users in the Quick Find box, then select Users.

2. Click New User.



Fill in the Fields:

Field Name	Value
First Name	Enter first name (Example: Ravi)
Last Name	Enter last name (Example: Kumar)

Alias	Give an alias name
Email ID	Enter your personal email ID
Username	Must be in the format: text@text.text and must be unique
Nickname	Give a nickname
Role	Select **Courier Admin**
User License	Salesforce
Profile	**Courier Admin**

3. Click Save.

Activity 2: Creating Other Users

Repeat the same steps as **Activity 1** for creating additional users.

Use the following details:

User Type	Role	User License	Profile
Branch Manager	Branch Manager	Salesforce	Branch Manager
Delivery Agent	Delivery Agent	Salesforce Platform	Delivery Agent
Customer Support	Customer Support	Salesforce Platform	Customer Support

Example:

To create a Branch Manager User:

- Go to Setup → Users → New User
- Fill details like:

Field	Example Value
First Name	Arjun
Last Name	Reddy
Email	arjun@gmail.com
Username	arjun.manager@cms.com
Role	Branch Manager
Profile	Branch Manager

- Click Save

\

To Create Delivery Agent User:

Follow same steps with:

- Role: Delivery Agent
- Profile: Delivery Agent

To Create Customer Support User:

- Role: Customer Support
- Profile: Customer Support

Important Points

- Every user must have:
    - Unique username
    - Valid email
    - Assigned Role

    - Assigned Profile

- Profiles control permissions
- Roles control data visibility

Milestone 15 : Reports & Dashboards

Reports and Dashboards help analyze business data and track courier operations efficiently.

Activity 1: Create Report Types

Step 1: Open Report Types

1. Click Gear Icon → Select Setup.
2. In Quick Find search Report Types.
3. Click New Custom Report Type.

Step 2: Create Shipment Report Type

- Primary Object: Shipment__c
- Label: Shipment Tracking Report
- Category: Reports
- Deployment Status: Deployed
- Click Next → Save

Step 3: Create Customer Report Type

- Primary Object: Customer__c
- Label: Customer Summary Report
- Category: Reports

- Deployment Status: Deployed
- Click Next → Save

Step 4: Create Agent Performance Report Type

- Primary Object: Delivery_Agent__c
- Label: Agent Performance Report
- Category: Reports
- Deployment Status: Deployed
- Click Next → Save

Activity 2: Create Reports

Report 1: Shipment Status Summary Report

Step 1: Create the Report

1. Click App Launcher (9 dots) → Search Reports
2. Click New Report
3. Choose Shipment Tracking Report
4. Click Continue

Step 2: Add Columns

Add these fields:

- Tracking Number
- Customer
- Status

- Expected Date
- Assigned Agent

Step 3: Add Filters

- Show Me: All Shipments
- Date Filter: This Month
- Status Filter: Select specific Status (Optional)

Step 4: Group & Summarize

- Group rows by Status__c
- Add summary on Price

Step 5: Save Report

- Click Save & Run
- Name: Shipment Status Summary
- Folder: CMS Reports

Report 2: Agent Performance Report

Step 1: Create Report

- New Report → Choose Agent Performance Report
- Click Continue

Step 2: Add Columns

- Agent Name
- Branch

- Number of Shipments
- Status

Step 3: Group Data

- Group by Delivery_Agent__c

Step 4: Save Report

- Name: Agent Performance Report
- Folder: CMS Reports
- Save & Run

Report 3: Revenue Report

Steps

- Create report using Shipment Tracking Report Type

Add fields:

- Tracking Number
- Customer
- Price
- Invoice Status

Group by:

- Branch
- Month

Save as: Revenue by Branch Report

Activity 3: Create Dashboard

Step 1: Open Dashboards

1. Click App Launcher (9 dots)
2. Search Dashboards
3. Click New Dashboard

Step 2: Enter Dashboard Details

- Name: Courier Management Dashboard
- Folder: CMS Reports
- Click Create

Activity 4: Add Dashboard Components

Component 1: Shipment Status Overview (Donut Chart)

1. Click + Component
2. Choose report: Shipment Status Summary
3. Visual Type: Donut Chart
4. Click Add

Component 2: Agent Performance (Bar Chart)

- Select report: Agent Performance Report

- Chart Type: Bar Chart
- Click Add

Component 3: Revenue by Branch (Column Chart)

- Choose report: Revenue by Branch Report
- Chart Type: Column Chart
- Click Add

Activity 5: Save & Activate Dashboard

1. Click Save
2. Click Done

Share Dashboard With:

- Courier Admin
- Branch Managers
- Customer Support Team

Final Layout :

7. FUNCTIONAL & PERFORMANCE TESTING

During testing, screenshots were captured from Salesforce to verify record creation, validation rule enforcement, delivery status updates, flow automation, invoice generation, and dashboard visualization. All scenarios were validated to ensure correct performance and reliability.

8. ADVANTAGES & DISADVANTAGES

Advantages

- Centralized data
- Automated workflows
- Real-time reporting

Disadvantages

- Salesforce dependency
- Requires initial configuration effort

9. Conclusion:

The Courier Management System successfully digitizes courier operations using Salesforce automation, improving efficiency, accuracy, and customer satisfaction.

10. FUTURE SCOPE

- Mobile tracking app
- GPS integration
- SMS notifications
- AI-based delivery optimization

11. APPENDIX

- Source Code: Salesforce Configuration
- Dataset: Salesforce Objects
- GitHub / Demo: Salesforce Developer Org

Thank You