
FINAL PROJECT REPORT

Project Title: Airlines Management System

Team Information

Team ID	LTVIP2025TMID29463	Email ID
Team Size	4	
Team Leader	Yelamkur Afaan Ahamed	Afaanahamed18@gmail.com
Team Member	Malle Bhavitha	Reddybhavitha007@gmail.com
Team Member	Chadulla Praveen	Praveenchadulla9999@gmail.com
Team Member	Mannala Ankitha	Mannalaankitha409@gmail.com

INTRODUCTION

Building a Centralized CRM for Efficient Airlines Management

Project Overview – Transforming Airline Operations through CRM technology

In today's fast-evolving aviation industry, airlines are expected to deliver highly efficient, timely, and customer-focused services. However, many airlines still depend on fragmented, manual systems for managing key operations such as flight scheduling, crew assignments, ticket fare management, and performance reporting.

These outdated processes lead to:

- **Inconsistent crew management**
- **Error-prone fare calculations**
- **Inefficient scheduling and data silos**
- **Limited real-time visibility for decision-makers**

To address these operational gaps, this project introduces a **custom-built CRM application using the Salesforce platform**.

Designed specifically for **airline operations**, the solution digitizes and streamlines:

- Crew and staff assignment
- Flight scheduling
- Automated ticket fare calculations
- Centralized reporting and real-time dashboards

This CRM solution aims to **transform airline processes from manual to automated, from fragmented to unified**, providing a reliable foundation for data-driven decision-making and operational excellence.

Key Features of the Solution:

- **Customized Object Model:**
Supports **airplanes, airports, airline employees (pilots, cabin crew), flights, and ticket fares** with well-defined data relationships.
- **Automated Logic:**
Automated **ticket fare calculation** and **employee role validation** using Salesforce **Flows** and **Apex Triggers** to ensure correct crew assignments and fare accuracy.
- **Formula-Based Fields:**
Real-time computations for **total flight fare, employee age, experience, and retirement dates** using formula fields.
- **Comprehensive Dashboards & Reports:**
Visual dashboards and detailed reports to support **administrative decision-making** across flight operations, employee assignments, and revenue tracking.

Purpose – Solving Operational Inefficiencies in Airline Management

By implementing a **centralized CRM system using Salesforce**, this project aims to:

- **Streamline airline operations** through automated processes and real-time data flows.
- **Improve data accuracy and integrity** using built-in validation rules and relationship-driven architecture.
- **Enable informed decision-making** using dynamic dashboards and reporting tools that provide real-time operational insights.
- **Develop a flexible, user-friendly, and scalable CRM platform** that can easily support future enhancements and integrations.

Ideation Phase

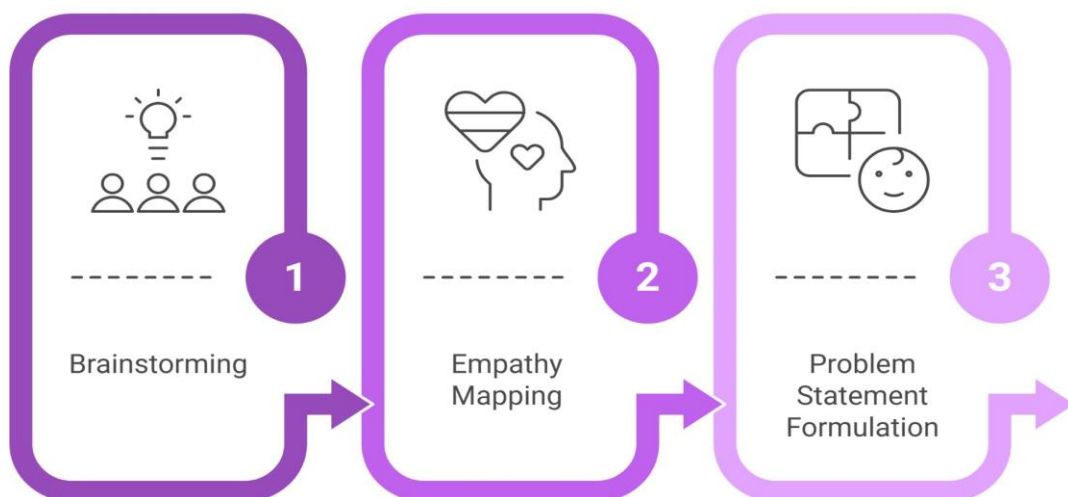
The Ideation Phase serves as the foundation of any successful project. It blends creativity, user empathy, and structured thinking to identify the core problem, generate meaningful ideas, and prioritize solutions that bring value to users. Where creativity and structured thinking combine to find meaningful and impactful documentation.

In our project titled: “A CRM Application for Airlines Management System”, this phase was critical to ensure we built a system tailored to the actual challenges faced by airline departments, ground staff, flight attendants, pilots, and passengers.

The ideation phase included three main steps:

1. Brainstorming
2. Empathy Mapping
3. Problem Statement Formulation

Ideation Phase Steps



1. ♦ Brainstorming & Idea Prioritization Template

Step 1: Team Gathering, Collaboration, and Selecting the Problem Statement

Our team convened with the goal of identifying inefficiencies in existing Airline operations and proposing a tech-driven solution using Salesforce. Through collaborative meetings, online whiteboards, and use-case discussions, we collectively explored pain points faced by airline staff and administrative personal. We reviewed real-world operations and identified that most airlines systems rely heavily on manual workflows for managing:

- Flight schedules and fare structures
- Pilot and cabin crew assignments
- Ticket booking and fare collection
- Monthly performance and safety reporting

After several discussions, we clearly defined the core issue:

Problem Statement:

"Airline departments lack a unified digital platform for managing flights, staff, ticketing, and operational metrics in real time. Existing manual processes are inefficient, error-prone, and restrict access to performance and safety insights."

This became the backbone of our project scope.

Step 2: Brainstorm, Idea Listing, and Grouping

We conducted a **team-wide brainstorming session** using a digital board where all team members contributed their raw ideas for improving airline operations. The collected ideas were categorized into the following key themes:

- **Data Management:** managing employee records, aircraft data, flight schedules, passenger information, and maintenance logs.
- **Automation:** real-time fare calculation, based demand and availability.
- **Reporting:** Automated monthly dashboards for flight counts, passenger traffic, on-time performance, baggage handling stats, and revenue.
- **Validation & Access Control:** rule-based data integrity to prevent scheduling and secure aircraft safely.

From around 25–30 ideas, we grouped and shortlisted the ones that aligned directly with operational efficiency.

Step 3: Idea Prioritization

Each grouped idea was evaluated on:

- **Feasibility:** How easily it could be implemented on Salesforce
- **Impact:** The significance of the feature on transport operations
- **Urgency:** Whether it solved a current, pressing problem

We created a decision matrix that helped us arrive at an **MVP** (Minimum Viable Product) plan:

- **Top Priority Features:**

- Automated fare and passenger data updates
- Role-based employee assignment (Pilot/Cabin Crew)
- Trigger-based alerts for invalid assignments
- Real-time summary dashboards (e.g., flights, passengers, revenue)
- Controlled and dependent picklists for aircraft and route management

These features formed the scope of our system design in the later phases.

Idea Prioritization Pyramid

Idea Prioritization

Rank ideas based on feasibility and impact

Idea Grouping

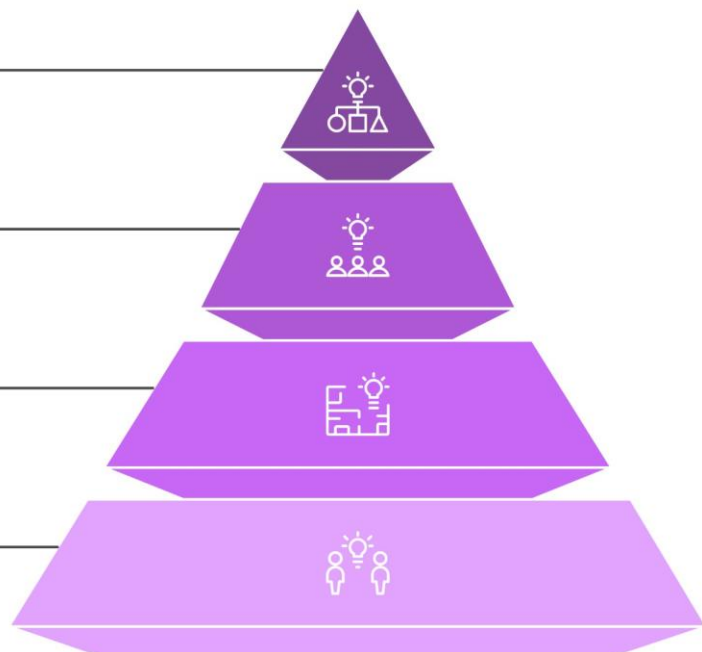
Organize ideas into related themes or categories

Idea Listing

Compile and categorize all brainstormed ideas

Brainstorming

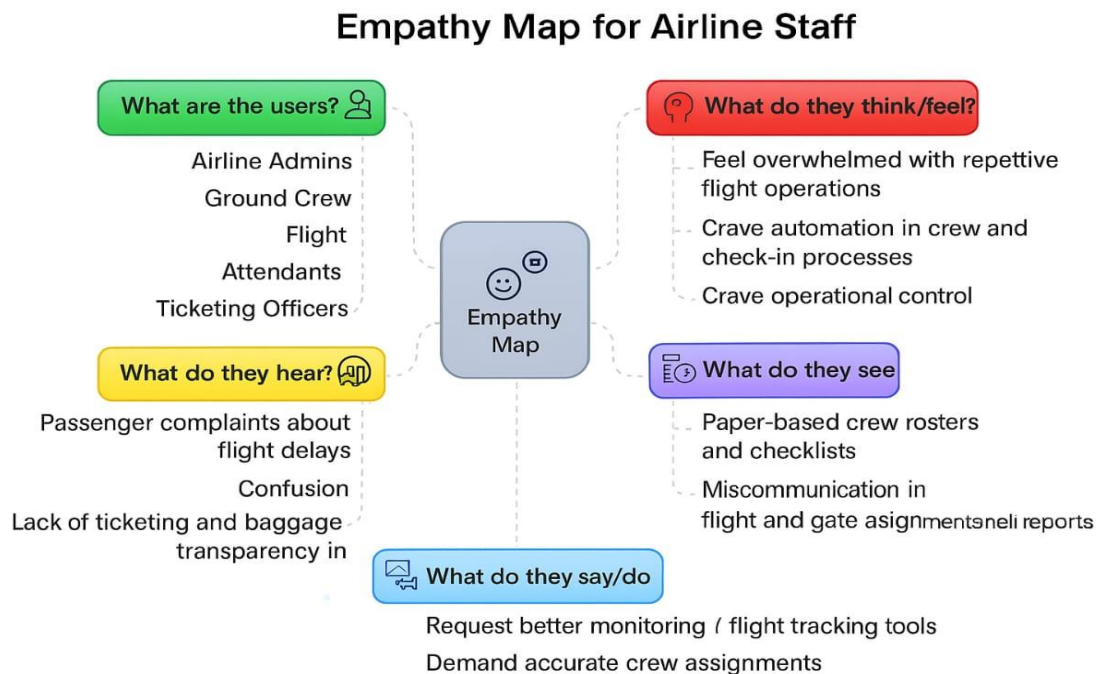
Generate diverse ideas through group discussion



◆ Empathy Mapping- Empathize & Discover

Empathy Map Canvas

An empathy map is a visual tool that helps teams deeply understand their users' experiences, pains, and expectations. We used it to map the daily journey of airline staff, including administrators, pilots and ticket officers.



By stepping into the user's shoes, we ensured that our Salesforce CRM features (formulas, flows, triggers, dashboards) directly addressed their key frustrations faced by airlines personnel.

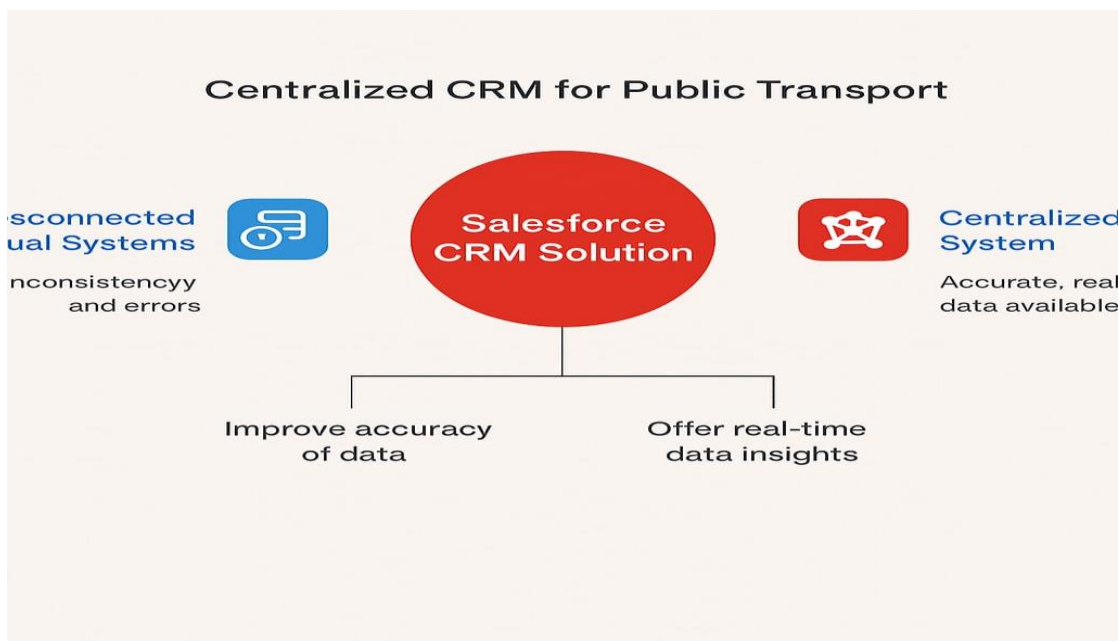
◆ Define the Problem Statements

Customer Problem Statement Template

To build a successful solution, it's essential to clearly define what the customer/user is struggling with. This helped us stay focused on delivering real-world impact instead of just implementing technical features.

Final Customer Problem Statement:

Airline departments manage critical operations manually using disconnected systems. This leads to data inconsistency, assignment errors, revenue loss, and lack of real-time performance visibility. A centralized Salesforce CRM system can digitize workflows, ensure accuracy, and provide actionable insights through dashboards, flows, and automation. This statement aligns with the end user's expectations and guided our object design, validations, formulas, triggers, and reports.



Requirement Analysis

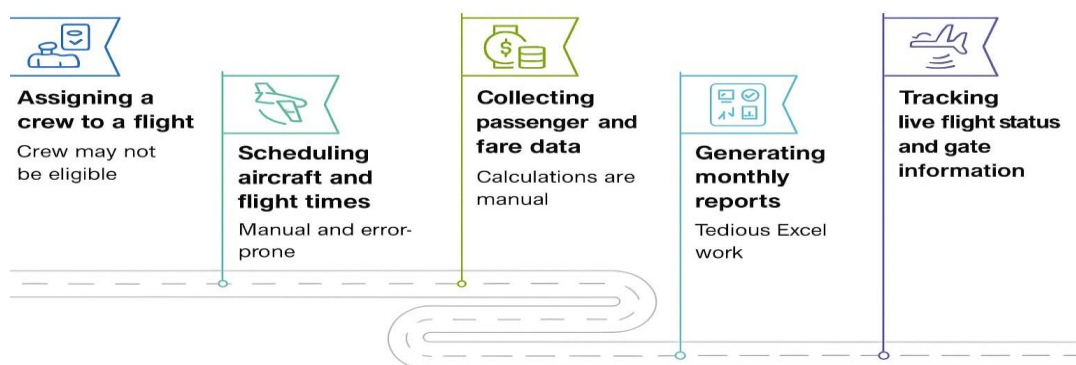
The Requirement Analysis Phase focuses on collecting, organizing, and validating everything your system needs to accomplish. It ensures that the solution you're building is not only technically sound but also directly aligned with what your stakeholders (airlines departments, pilots, ground crew and flight managers) truly need. In our project, "Airlines management system," this phase helped bridge the gap between problem understanding and system design, using real user journeys, data flows, and solution requirements. Where user-centric planning meets technical clarity to transform a problem into a buildable solution.

◆ Customer Journey Map-Understanding User Experience Flow

Purpose: The **Customer Journey** Map visualizes how airlines employees (e.g., Ground operation Managers, Admins, Pilots, Cabin crew) interact with the transport system daily. It highlights key actions, pain points, and opportunities for improvement, guiding CRM design from a real-world usage perspective.

Journey Steps (For an Airline Admin):

Journey Steps (For an Airline Admin):



This journey helped prioritize features like formula fields, validation rules, and real-time dashboards in our object and flow design.

◆ Data Flow Diagram

Purpose: Mapping Information Flow Between Objects

The Data Flow Diagram (DFD) models how information moves between Salesforce objects and components in the Airlines management system. It helped us structure relationships between:

- Airport, Airplane, Flight, Employee, and Ticket Fare

Level 1 DFD Overview:

1. Admin Inputs:

- Adds airplane, airport, and flight details
- Assigns pilots and cabin crew (from Employee object)

2. System Logic:

- Validates employee role (Pilot or Cabin crew)
- Fetches fare via flow based on flight route and airplane model
- Calculates total fare using formula (Passenger Count x Fare)

3. Outputs:

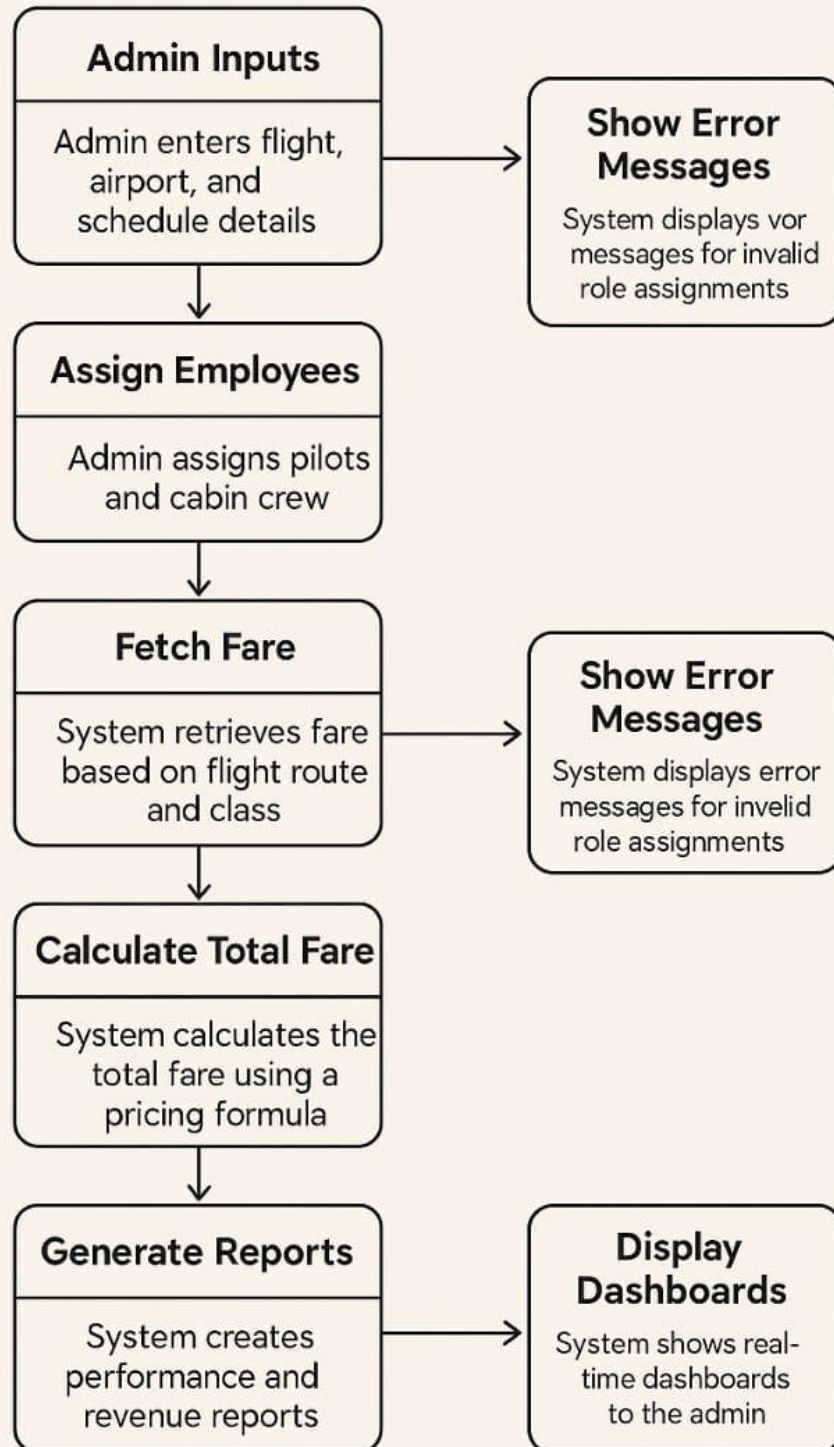
- Performance insights via reports and Real-time dashboards for admin
- Error messages for invalid role assignments

Objects Involved in Airline Operations

Airport → Aircraft → Employee → Flight → Ticket Fare

Airlines Management System

Level 1 DFD Overview



◆ Solution Requirements

What the Airlines management System Must Do

Functional Requirements:

- Create custom objects for: **Airport, Airplane, Flight, Airlines Employees** (Pilot, Cabin Crew, Ground Staff), **Ticket Fare**
- Establish relationships using lookup and formula fields
- Validate employee roles using Apex Triggers
- Automate fare fetching via Flows
- Support performance dashboards & summary reports

Non-Functional Requirements:

- User-friendly Lightning App Interface
- Real-time field-level validation
- Centralized database with accurate relationships

◆ Technology Stack

Tools & Platforms Used for Airlines management system Implementation

Category	Technology Used	Description
Platform	Salesforce Lightning	Used to build customer Airline CRM using standard & custom objects
Automation	Flows & Validation Rules	Automate fare fetching and restrict invalid data
Custom Logic	Apex Triggers & Classes	Used to validate Pilot/Cabin crew ID
Reports & Dashboards	Salesforce Reports	To analyze employee data, flight schedules, revenue, etc.
UI/UX	Lightning App Builder	Create a unified Airlines Management App for handling flights, employees, airports, and ticketing

Summary

This Requirement Analysis Phase helped ensure that all CRM features were grounded in user pain points, supported by data structures, and enabled by the right technologies. It directly shaped how we approached object modeling, data automation, UI design, and performance reporting in Salesforce.

Project Design Phase

The Project Design Phase defines the logical, technical, and functional foundation of the solution. It ensures that your proposed Salesforce CRM not only solves the right problems but is also scalable, maintainable, and aligned with Salesforce architecture principles. Where validated problems transform into structured, scalable, and implementable solutions.

In our project, “Airlines Management System,” this phase bridges the gap between ideation and execution by converting insights from the previous requirement analysis well-structured CRM solution tailored for airline operations.

◆ Problem–Solution Fit

Problem Recap:

Airlines operate in a complex environment requiring real-time tracking of flights, airplanes, crew members, airports, and fare management. However, many airline processes are still managed manually or isolated in separate systems, leading to:

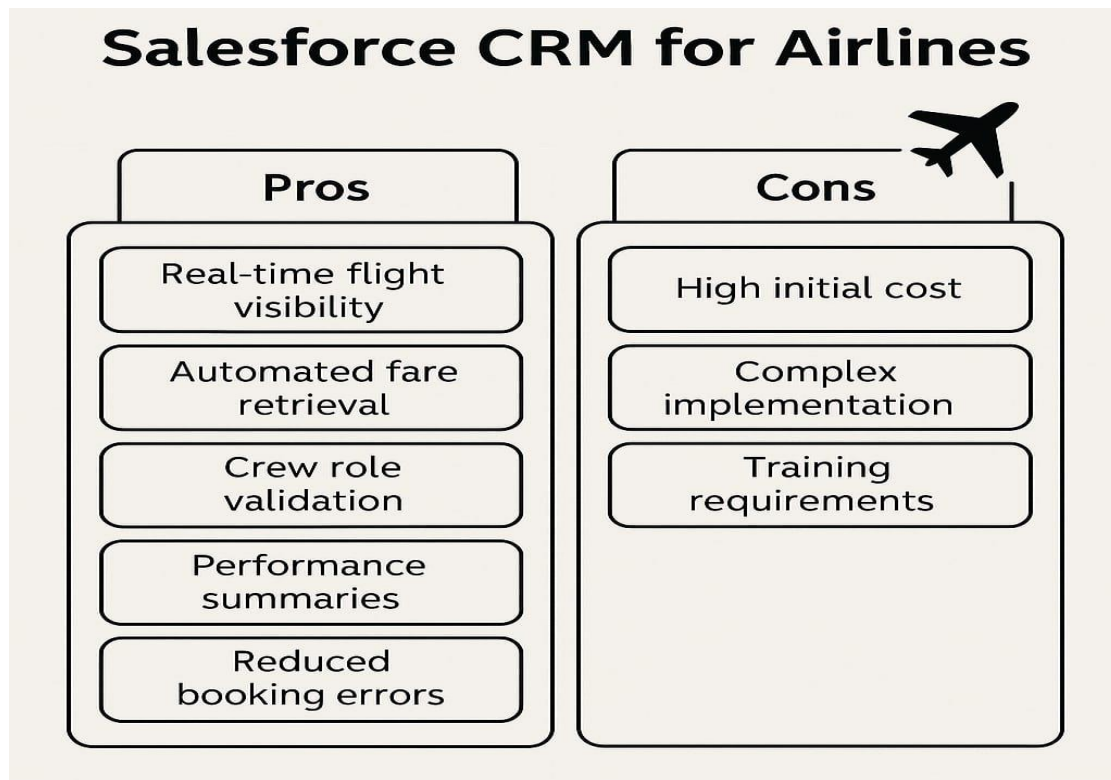
- Data entry errors and duplications
 - Lack of real-time visibility for flight scheduling and operations
 - Improper assignment of Pilots/Cabin crew
 - Challenges in fare management, flight reporting, and occupancy tracking
-

Does the Proposed Solution Fit?

Yes. The Salesforce CRM solution for Airlines Management directly addresses the critical pain points in airline operations. The solution:

- Introduces **object-level control** for Airplanes, Airports, Employees (Pilots, Cabin Crew), Flights, and Ticket Fares
- Enables **automated fare retrieval** through Flows
- Validates pilots and cabin crew assignments using **Apex Triggers**
- Summarizes performance using **Reports and Dashboards**
- Uses formula fields to reduce calculation errors
- Creates centralized views using **Lightning App Builder**

Thus, it directly fits the core operational pain points of Airlines workflows.



◆ Proposed Solution

How Our CRM Will Solve the Identified Problems

Our proposed Airlines CRM application is designed to digitize and streamline Airline operations using Salesforce's declarative and programmatic capabilities.

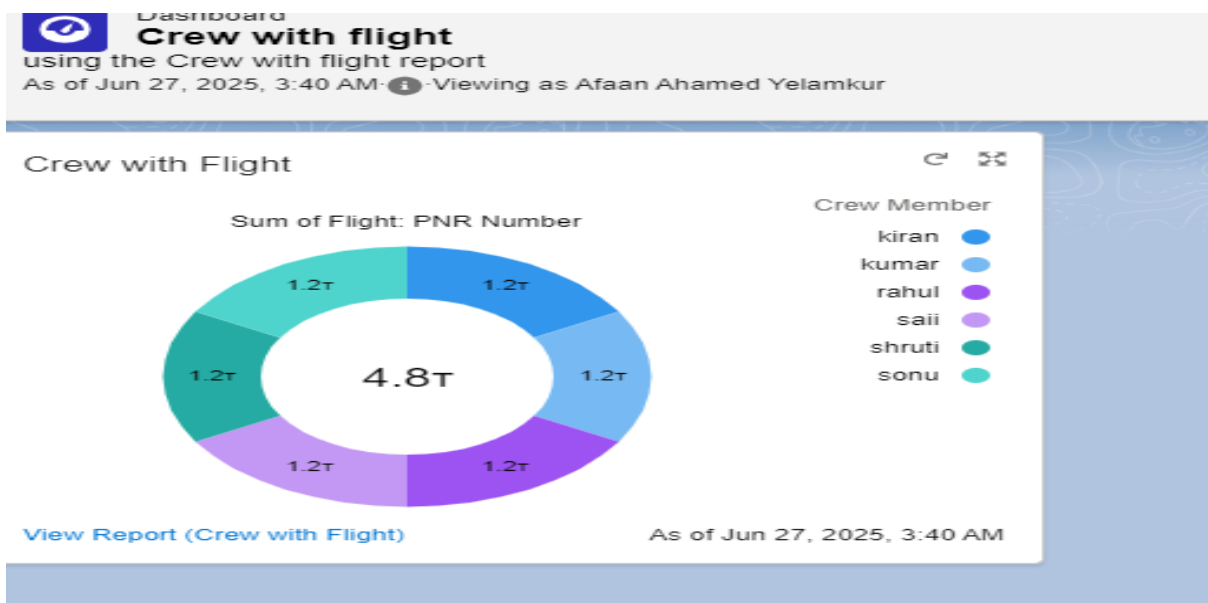
Key Functional Features:

- **Custom Objects:**
 - Airport, Airplane, Flight, Ticket Fare, Employee
- **Automation & Validation:**
 - Role verification (Pilot/Cabin crew) via **Apex Triggers**
 - Fare calculation automation via **Flows**
 - Input control via **Validation Rules**
- **Formula Fields for Efficiency:**
 - Age, Experience, Date of Retirement for employees
 - Total Fare = Passenger Count × Ticket Fare
 - Pilot & Cabin crew Names (auto-derived via lookups)
- **UI & Navigation:**
 - Public Transport App using **Lightning App Builder**
 - Tabs for all custom objects

- Page layouts designed by object and role
- **Reports and Dashboards:**
 - Trip Summary Reports
 - Cabin crew with Flight Assignment Reports
 - Monthly Revenue and Passenger Count Dashboards

Together, these components will offer a centralized, role-driven, and insight-ready system.

Report: Crews with Flight Crew with Flight								
Total Records 6		Total Flight: PNR Number 4,848,484,969,360						
<input type="checkbox"/> Crew Member ↑	Crew: Crew Id	Flight: Flight Name	Role	Flight: PNR Number	Flight: Departs From	Flight: Departs to	Flight: Arrival Time	Phone
<input type="checkbox"/> kiran (1)	Bk-0002	indigo	Pilot	1,212,121,242,325	Mumbai	Bengaluru	1:15 AM	9000431655
<input type="checkbox"/> kumar (1)	Bk-0004	emirates	Flight Attendants	1,212,121,242,355	Delhi	Kolkata	5:30 AM	9000431655
<input type="checkbox"/> rahul (1)	Bk-0003	indigo	Co-Pilot	1,212,121,242,325	Mumbai	Bengaluru	1:15 AM	9000431655
<input type="checkbox"/> sai (1)	Bk-0005	emirates	Air Hosts	1,212,121,242,355	Chennai	Kolkata	5:45 AM	9000431655
<input type="checkbox"/> shruti (1)	Bk-0006	air india	Air Hostesses	1,212,121,242,325	Mumbai	Chennai	3:15 PM	9000431655
<input type="checkbox"/> sonu (1)	Bk-0001	air india	Air Hosts	1,212,121,242,325	Mumbai	Chennai	3:15 PM	09000431655
Total (6)				4,848,484,969,360				



◆ Solution Architecture

Visualizing the Technical Structure and Object Relationships

Object Relationship Overview:

Object	Key Fields / Features
Airport	Name, Category, Amenities, Address
Airplane	Linked to Airport, Category, Model, Capacity
Airline Employee	Name, Role (Picklist), DOB, Experience, Lookup to Airport
Flight	Linked to Airplane, Pilot, Cabin Crew, Ticket Fare, Flight Date, Passenger Count, Total Fare
Ticket Fare	Route, Airplane Model, Fare

Lookups:

- Employee → Airport
- Flight → Airplane, Employee (Pilot), Employee (Cabin Crew), Ticket Fare
- Airplane → Airport

Formulas:

- $\text{Total_Amount_c} = \text{Passenger_Count_c} * \text{Ticket_Fare_c}$
 - $\text{Pilot_Name_c} = \text{Pilot_Id_r.Employee_Name_c}$
 - $\text{Cabin_crew_Name_c} = \text{Cabin_crew_Id_r.Employee_Name_c}$
-

Automation:

- Flows for fare fetch logic
 - Triggers for role validation
 - Reports & dashboards for output
-

Summary

The Project Design Phase ensured that our Airlines CRM not only met the users' needs but also followed Salesforce best practices in object modeling, validation, automation, and user experience. This clear blueprint guided our execution in upcoming development and configuration phases.

Project Planning Phase

The Project Planning Phase converts high-level milestones into actionable sprints aligned with the internship timeline. This helps streamline delivery and keeps all team members aligned with progress, ownership, and deadlines. Where structured task breakdown and time-bound execution planning ensures delivery efficiency.

◆ Project Planning Template

Sprint Schedule – Based on Project Milestones

Sprint	Functional Requirement (Epic)	Task (Mapped from Milestone)	Priority	Team Members
Sprint-1	Developer Setup & Basic Objects	Creating Developer Account & Activating Org	High	Member 1
Sprint-1	Custom Object Creation	Creating custom objects – Airport, Airplane, Airline Employee, Flight, Ticket Fare	High	Member 1, 2
Sprint-2	UI Tabs & App Creation	Creating Tabs & Lightning App	High	Member 3
Sprint-2	Field Configuration	Creating fields, formula fields, picklists, relationships	Medium	Member 1, 3
Sprint-3	Layouts & Validations	Page Layouts + Validation Rules	High	Member 2, 4
Sprint-3	Flows & Triggers	Automations using Flows and Apex Triggers	High	Member 2, 3

Sprint-4	Reports & Dashboards	Generate Reports and create Dashboards	High	Member 4
Sprint-4	Final Integration & Conclusion	Final Review, Testing, and Functional Summary	Medium	All Members

Project Tracker & Sprint Timeline

Duration: Each sprint is 6 days, aligned with your **June 2025 internship schedule**

Sprint	Duration	Sprint Start Date	Sprint End Date	Sprint Release Date
Sprint-1	6 Days	03 Jun 2025	08 Jun 2025	08 Jun 2025
Sprint-2	6 Days	09 Jun 2025	14 Jun 2025	14 Jun 2025
Sprint-3	6 Days	15 Jun 2025	20 Jun 2025	20 Jun 2025
Sprint-4	6 Days	21 Jun 2025	26 Jun 2025	26 Jun 2025

Summary

The **Project Planning Phase** allowed our team to convert 12 major milestones into 4 streamlined sprints with assigned priorities and contributors. By aligning sprints with real internship dates and breaking tasks down into functional chunks, we ensured steady progress and simplified execution.

Project Executable Files


This phase outlines the actual Salesforce configurations, data, and outcomes used and generated during the execution of your project: “Airlines Management System.” It ensures that all key project elements—objects, data, and output—are traceable and reusable for future reference or assessment. Where practical configurations and working modules of the project are documented for clarity, replication, and validation.


◆ 1. Project Files

Project Executable Files


The following project files were executed in the Salesforce Developer Org:


 **Milestone 1: Developer Account Setup**

 **Milestone 2: Object Creation**

 **Milestone 3: Tab Creation**

 **Milestone 4: Lightning App Setup**

 **Milestone 5: Field Creation**


 **Milestone 6: User adoption**

 **Milestone 7: Profiles**

 **Milestone 8: Role**

 **Milestone 9: User**

 **Milestone 10: Reports**

 **Milestone 11: Dashboards**

 **Milestone 12: Apex**

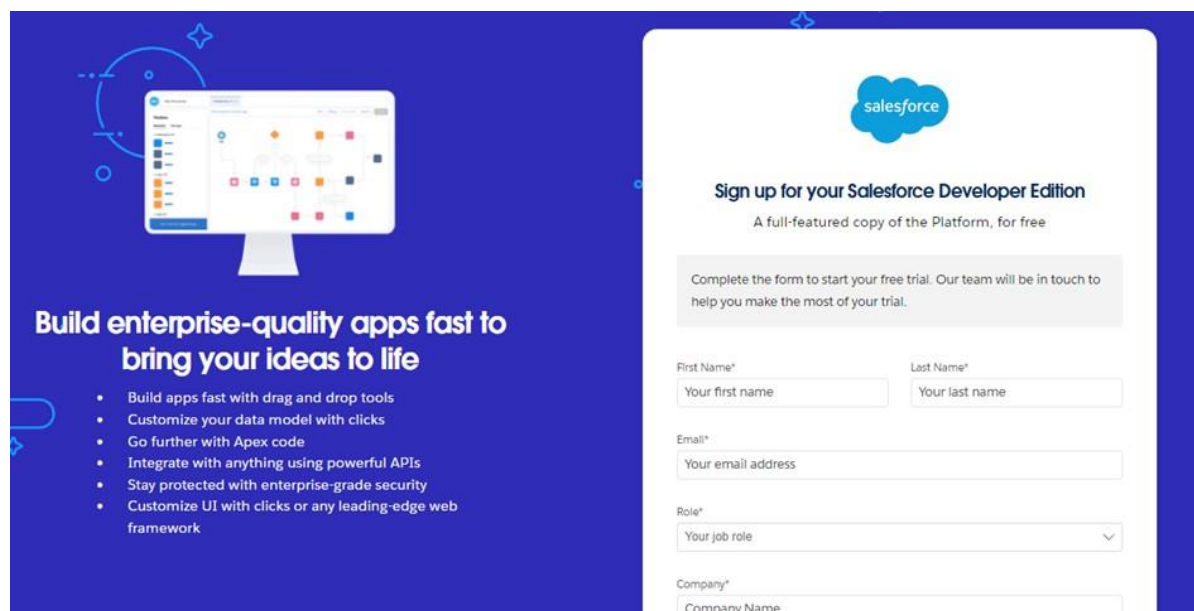
 **Milestone 13: Flows**

List of Milestone Tasks with Supporting Screenshots and Descriptions

Milestone 1: Developer Account Setup

- Created and activated a Salesforce Developer Org.
 - Setup the base environment for CRM development.
 - Verified access to Object Manager, Flow Builder, and App Builder.
-

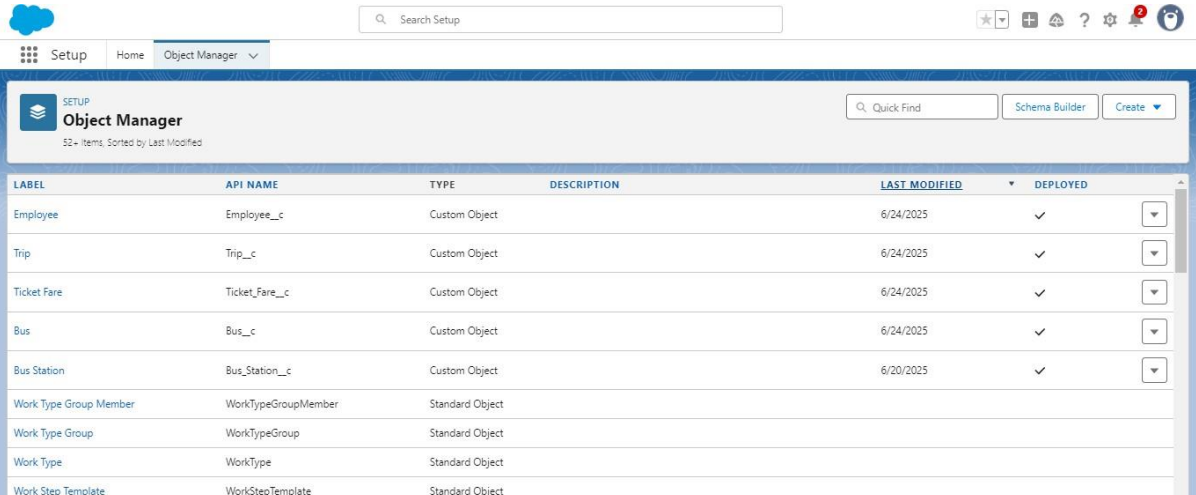
OUTPUT SCREENSHOT



Milestone 2: Object Creation

- Created 5 custom objects:
 - Airport, Airplane, Airline Employee, Flight and Ticket Fare.
 - Established foundational schema for Airlines data tracking.
- Configured relationships using lookup fields.

OUTPUT SCREENSHOT




The screenshot displays the Salesforce Object Manager interface. At the top, there is a navigation bar with 'Setup', 'Home', and 'Object Manager' (selected). A search bar labeled 'Search Setup' is also present. Below the navigation bar, the 'Object Manager' section is titled with a sub-header '52 Items, Sorted by Last Modified'. The main content area shows a table of objects with columns: LABEL, API NAME, TYPE, DESCRIPTION, LAST MODIFIED, and DEPLOYED. The table lists several custom objects (Employee, Trip, Ticket Fare, Bus, Bus Station) and standard objects (Work Type Group Member, Work Type Group, Work Type, Work Step Template).

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Employee	Employee_c	Custom Object		6/24/2025	✓
Trip	Trip_c	Custom Object		6/24/2025	✓
Ticket Fare	Ticket_Fare_c	Custom Object		6/24/2025	✓
Bus	Bus_c	Custom Object		6/24/2025	✓
Bus Station	Bus_Station_c	Custom Object		6/20/2025	✓
Work Type Group Member	WorkTypeGroupMember	Standard Object			
Work Type Group	WorkTypeGroup	Standard Object			
Work Type	WorkType	Standard Object			
Work Step Template	WorkStepTemplate	Standard Object			

Milestone 3: Tab Creation

- Created tabs for each custom object.
 - Enabled easy navigation and object access in the app.
 - Ensured users can create/view records from the UI.
-





OUTPUT SCREENSHOT

 **SETUP**
Tabs

Custom Tabs

You can create new custom tabs to extend Salesforce functionality or to build new application functionality.

Custom Object tabs look and behave like the standard tabs provided with Salesforce. Web tabs allow you to embed external web applications and content within the Salesforce Visualforce tabs allow you to embed Visualforce pages. Lightning Component tabs allow you to add Lightning components to the navigation menu in Lightning Experience and Lightning Page tabs allow you to add Lightning Pages to Lightning Experience and the mobile app.

Custom Object Tabs		New	What Is This?
Action	Label	Tab Style	Description
Edit Del	Bookings	 Bell	
Edit Del	Crews	 Presenter	
Edit Del	Flights	 Airplane	
Edit Del	Passengers	 Bank	

Milestone 4: Lightning App Setup

- Built a custom Lightning App named "**Airlines Management system**".
 - Added relevant tabs to centralize operations.
 - Simplified user workflow by grouping features.
-

OUTPUT SCREENSHOT

* App Name ⓘ

AirLine Management System


* Developer Name ⓘ

AirLine_Management_System


Description ⓘ

AirLines Management System

Image ⓘ



Primary Color Hex Value ⓘ


 ▼

#0070D2

Org Theme Options

☐ Use the app's image and color instead of the org's custom theme

App Launcher Preview



AirLine Management System
AirLines Management System

Milestone 5: Field Creation

- Added custom and formula fields like:
 - Capacity, Departure date, Departs from, Departs to, Fare, Passenger Count.
 - Implemented: **Total Fare = Fare × Passenger Count** formula.
- Linked Driver and Conductor to Trip via lookup fields.

Milestone 5- Fields

<u>Object</u>	<u>Field Name</u>	<u>Data Type</u>
<u>Airport</u>	<u>Airport Name(Standard)</u>	<u>Text</u>
	<u>Airport Category</u>	<u>Picklist(Unmanaged Bus Stop, Managed Bus Stop)</u>
	<u>Last Updated</u>	<u>Formula(Date)</u>
	<u>Amenities</u>	<u>Picklist(Multi-select)</u>
	<u>City</u>	<u>Text(40)</u>
	<u>Street</u>	<u>TextArea</u>
	<u>State/Province</u>	<u>Text(25)</u>
	<u>Zip/PostalCode</u>	<u>Text(10)</u>
	<u>lounge</u>	<u>Checkbox</u>
	<u>Bench</u>	<u>Checkbox</u>

<u>Airplane</u>	<u>Registration No</u> <u>(Standard)</u>	<u>Text</u>
	<u>Name</u>	<u>Lookup(Bus Station)</u>
	<u>Capacity</u>	<u>Number(4,0)</u>
	<u>Category</u>	<u>Picklist</u>
	<u>Model</u>	<u>Picklist(Dependent on Category)</u>
<u>Flight</u>	<u>Flight No(Standard)</u>	<u>Text</u>
	<u>Flight Date</u>	<u>Date</u>
	<u>Flight No</u>	<u>Lookup(Bus)</u>
	<u>Route Name</u>	<u>Lookup(Ticket Fare)</u>
	<u>Arrival Time</u>	<u>Picklist</u>
	<u>Departure Time</u>	<u>Picklist</u>
	<u>Starting</u> <u>Terminal</u>	<u>Text</u>
	<u>Destination</u> <u>Terminal</u>	<u>Text</u>
	<u>Pilot Id</u>	<u>Lookup(Employee)</u>
	<u>pilot</u>	<u>Formula</u>
	<u>Cabin crew Id</u>	<u>Lookup(Employee)</u>
	<u>Cabin crew</u>	<u>Formula</u>
	<u>Estimated Travel</u> <u>Time</u>	<u>Number</u>
	<u>Frequency Per Day</u>	<u>Number(2,0)</u>
	<u>No. of Stops</u>	<u>Number(2,0)</u>
	<u>Passenger Count</u>	<u>Number(4,0)</u>

	<u>Ticket Fare</u>	<u>Currency(16,2)</u>
	<u>Total Amount</u>	<u>Formula</u>
<u>Ticket Fare</u>	<u>Route Name(Standard)</u>	<u>Text</u>
	<u>Model</u>	<u>Picklist</u>
	<u>Ticket Fare</u>	<u>Currency(10,2)</u>
<u>Employee</u>	<u>Employee Id(Standard)</u>	<u>Text</u>
	<u>Name</u>	<u>Lookup(Bus Station)</u>
	<u>Employee Name</u>	<u>Text</u>
	<u>Role</u>	<u>Picklist</u>
	<u>Date of Birth</u>	<u>Date</u>
	<u>Age</u>	<u>Formula(Number)</u>
	<u>Work Place</u>	<u>Text</u>
	<u>Salary</u>	<u>Currency(18,0)</u>
	<u>Phone</u>	<u>Phone</u>
	<u>Date of Joining</u>	<u>Date</u>
	<u>Date of Retirement</u>	<u>Formula(Date)</u>
	<u>Experience</u>	<u>Formula(Number)</u>
	<u>Street</u>	<u>TextArea</u>
	<u>City</u>	<u>Text</u>
	<u>State/Province</u>	<u>Text</u>
	<u>Country</u>	<u>Text</u>
	<u>Zip/PostalCode</u>	<u>Text</u>

OUTPUT SCREENSHOT

Setup

Home

Object Manager

Search Setup

Star

Grid

Refresh

Help

Settings

Notifications

SETUP > OBJECT MANAGER

Bus

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

Fields & Relationships

8 Items, Sorted by Field Label

Quick Find

New

Deleted Fields

Field Dependencies

Set History Track

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Bus Registration No	Name	Text(80)		✓
Bus Station Name	Bus_Station_Name__c	Lookup(Bus Station)		✓
Capacity	Capacity__c	Number(4, 0)		
Category	Category__c	Picklist		
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Model	Model__c	Picklist	Category	
Owner	OwnerId	Lookup(User,Group)		✓

Setup

Home

Object Manager

Search Setup

Star

Grid

Refresh

Help

Settings

Notifications

SETUP > OBJECT MANAGER

Bus Station

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

Scoping Rules

Fields & Relationships

14 Items, Sorted by Field Label

Quick Find

New

Deleted Fields

Field Dependencies

Set History Tr

Bus Station Name	Name	Text(80)		✓
Bus Stop Category	Bus_Stop_Category__c	Picklist		
City	City__c	Text(40)		
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Last Updated	Last_Updated__c	Formula (Date)		
Location	Location__c	Text(40)		
Owner	OwnerId	Lookup(User,Group)		✓
Shelter available	Shelter_available__c	Checkbox		
State/Province	State_Province__c	Text(25)		

Setup

Home

Object Manager

Search Setup

Star

Share

Help

Settings

Notifications

SETUP > OBJECT MANAGER

Employee

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

20 Items, Sorted by Field Label

Quick Find

New

Deleted Fields

Field Dependencies

Set History Tr

Date of Retirement	Date_of_Retirement__c	Formula (Date)	
Employee Id	Name	Text(80)	✓
Employee Name	Employee_Name__c	Text(80)	
Experience	Experience__c	Formula (Number)	
Last Modified By	LastModifiedById	Lookup(User)	
Owner	OwnerId	Lookup(User,Group)	✓
Phone	Phone__c	Phone	
Role	Role__c	Picklist	
Salary	Salary__c	Currency(18, 0)	
State/Province	State_Province__c	Text(18)	

All DO

Setup

Home

Object Manager

Search Setup

Star

Share

Help

Settings

Notifications

SETUP > OBJECT MANAGER

Trip

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

Scoping Rules

Fields & Relationships

21 Items, Sorted by Field Label

Quick Find

New

Deleted Fields

Field Dependencies

Set History Tr

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Arrival Time	Arrival_Time__c	Picklist		
Bus No	Bus_No__c	Lookup(Bus)		✓
Bus Starting Terminal	Bus_Starting_Terminal__c	Text(18)		
Conductor Id	Conductor_id__c	Lookup(Employee)		✓
Conductor Name	Conductor_Name__c	Formula (Text)		
Created By	CreatedById	Lookup(User)		
Departure Time	Departure_Time__c	Picklist		
Destination Terminal	Destination_Terminal__c	Text(18)		
Driver Id	Driver_id__c	Lookup(Employee)		✓

Setup

Home

Object Manager

Search Setup

Star

Share

Help

Settings

Notifications

SETUP > OBJECT MANAGER

Ticket Fare

Details

Fields & Relationships

Page Layouts

Lightning Record Pages

Buttons, Links, and Actions

Compact Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

Fields & Relationships

6 Items, Sorted by Field Label

Quick Find

New

Deleted Fields

Field Dependencies

Set History Tr

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Bus Model	Bus_Model__c	Picklist		
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Route Name	Name	Text(80)		✓
Ticket Fare	Ticket_Fare__c	Currency(10, 2)		

Milestone 6: User Adoption

- Customized page layouts per object.
- Grouped fields logically for usability.
- Enhanced record readability and data entry experience.

OUTPUT SCREENSHOT

* App Name ⓘ

AirLine Management System

* Developer Name ⓘ

AirLine_Management_System


Description ⓘ

AirLines Management System

Image ⓘ

Upload


Primary Color Hex Value ⓘ

 #0070D2

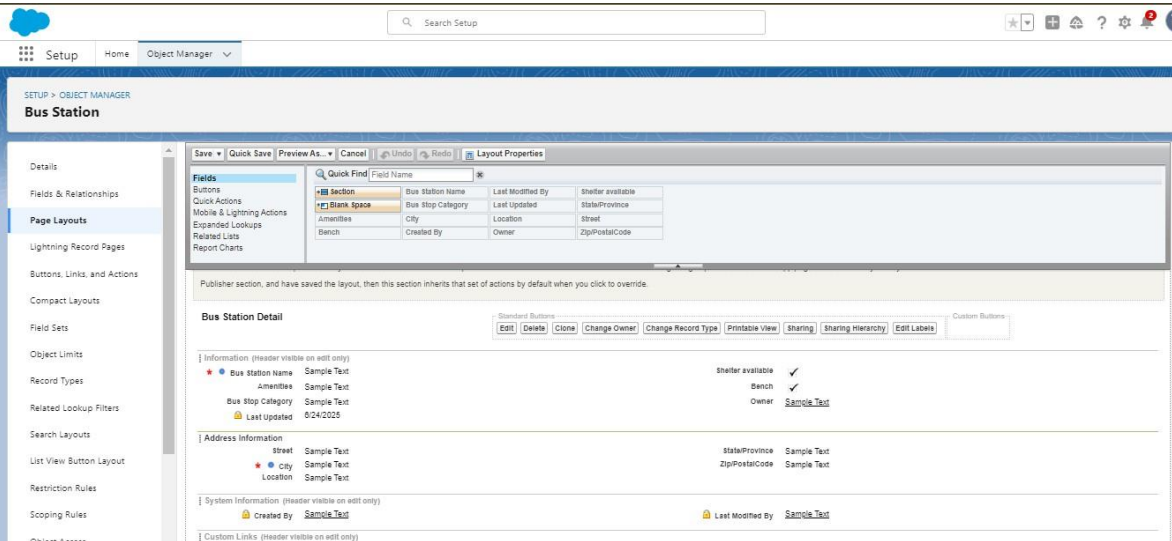
Org Theme Options

☐ Use the app's image and color instead of the org's custom theme

App Launcher Preview



AirLine Management System
AirLines Management System



The screenshot shows the Salesforce Setup interface for the 'Bus Station' object. The left sidebar contains navigation links: Setup, Home, Object Manager, Details, Fields & Relationships, Page Layouts (selected), 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, Scoping Rules, and Object Access. The main content area is titled 'Bus Station' and includes a 'Details' section with a 'Page Layouts' tab. The 'Page Layouts' section shows a 'Bus Station Detail' layout with a 'Fields' table and a 'Form' section. The 'Fields' table lists fields such as 'Bus Station Name', 'Bus Stop Category', 'City', 'Location', 'Created By', 'Last Updated', 'State/Province', 'Street', and 'ZipPostalCode'. The 'Form' section displays the layout of the 'Bus Station' record, including fields for 'Bus Station Name', 'Amenities', 'Bus Stop Category', 'Last Updated', 'Address Information', 'System Information', and 'Custom Links'. The 'Form' section also includes a 'Buttons' section with buttons like 'Edit', 'Delete', 'Clone', 'Change Owner', 'Change Record Type', 'Printable View', 'Sharing', 'Sharing Hierarchy', and 'Edit Labels'.

SetupHomeObject Manager

Search Setup

StarAddCloudHelpSettingsNotifications

SETUP > OBJECT MANAGER

Bus

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

SaveQuick SavePreview As...CancelUndoRedoLayout Properties

Fields

Quick FindField Name

Section

Blank Space

Bus Registration No

Bus Station Name

Capacity

Category

Created By

Last Modified By

Model

Owner

Bus Detail

Standard Buttons

EditDeleteCloneChange OwnerChange Record TypePrintable ViewSharingSharing HierarchyEdit Labels

Custom Buttons

Information (Header visible on edit only)

Bus RegistrationNoSample Text

Bus Station NameSample Text

CategorySample Text

ModelSample Text

Capacity1,007

OwnerSample Text

System Information (Header visible on edit only)

Created BySample Text

Last Modified BySample Text

Custom Links (Header visible on edit only)

SetupHomeObject Manager

Search Setup

StarAddCloudHelpSettingsNotifications

SETUP > OBJECT MANAGER

Trip

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

Scoping Rules

Object Access

SaveQuick SavePreview As...CancelUndoRedoLayout Properties

Fields

Quick FindField Name

Section

Blank Space

Bus Starting Term...

Arrival Time

Bus No

Conductor Id

Conductor Name

Created By

Driver Name

Departure Time

Destination Terminal

Driver Id

No. of Stops

Estimated Travel ...

Frequency Per Day

Last Modified By

No. of Stops

Owner

Passenger Count

Route Name

Ticket Fare

Total Amount

Trip Date

Trip No

Trip Detail

Standard Buttons

EditDeleteCloneChange OwnerChange Record TypePrintable ViewSharingSharing HierarchyEdit Labels

Custom Buttons

Information (Header visible on edit only)

Trip NoSample Text

Trip Date02/24/2025

Bus NoSample Text

Driver IdSample Text

Driver NameSample Text

Conductor IdSample Text

Conductor NameSample Text

OwnerSample Text

Bus Schedule

Route NameSample Text

Bus starting TerminalSample Text

Departure TimeSample Text

No. of Stops64

Estimated Travel Time53,353

Destination TerminalSample Text

Arrival TimeSample Text

Frequency Per Day64

Passenger Information

Passenger Count5,203

Ticket Fare\$123.45

Total Amount\$123.45

System Information (Header visible on edit only)

Passengers

Recently Viewed

NewImportChange OwnerAssign

6 items • Updated a few seconds ago

Search this list...

StarAddCloudHelpSettingsNotifications

Passenger Id

1

2

3

4

5

6

Bk-0006

Bk-0005

Bk-0004

Bk-0003

Bk-0002

Bk-0001

Milestone 7: Profiles

Types of profiles in salesforce


Standard profiles;

By default salesforce provides below standard profiles

Custom Profiles:

Custom ones defined by us.

OUTPUT SCREENSHOT



SETUP Profiles

Profile Edit
General Admin
Set the permissions and page layouts for this profile.

Profile Edit

Name: General Admin
User License: Salesforce Platform
Custom Profile: ☒
Description:

Custom App Settings

	Visible	Default		Visible	Default
AirLine Management System (AirLine_Management_System)	<input type="checkbox"/>	<input type="radio"/>	Platform (standard__Platform)	<input checked="" type="checkbox"/>	<input checked="" type="radio"/>
Analytics Studio (standard__Insights)	<input type="checkbox"/>	<input type="radio"/>	WDC (standard__Work)	<input type="checkbox"/>	<input type="radio"/>
App Launcher	<input type="checkbox"/>	<input type="radio"/>			

SETUP Profiles

Profile Edit
Senior Admin Profile
Set the permissions and page layouts for this profile.

Profile Edit

Name: Senior Admin Profile
User License: Salesforce
Custom Profile: ☒
Description:

Custom App Settings

	Visible	Default		Visible	Default
AirLine Management System (AirLine_Management_System)	<input checked="" type="checkbox"/>	<input type="radio"/>	My Service Journey (standard__MSJApp)	<input checked="" type="checkbox"/>	<input type="radio"/>
All Tabs (standard__AllTabSet)	<input checked="" type="checkbox"/>	<input type="radio"/>	Queue Management (standard__QueueManagement)	<input checked="" type="checkbox"/>	<input type="radio"/>
Analytics Studio (standard__Insights)	<input checked="" type="checkbox"/>	<input type="radio"/>	Sales (standard__LightningSales)	<input checked="" type="checkbox"/>	<input type="radio"/>
App Launcher (standard__AppLauncher)	<input checked="" type="checkbox"/>	<input type="radio"/>	Sales (standard__Sales)	<input checked="" type="checkbox"/>	<input checked="" type="radio"/>

Profiles

Profile Edit
Management Admin
Set the permissions and page layouts for this profile.

Profile Edit

Name: Management Admin
User License: Salesforce Platform
Custom Profile: ☒
Description:

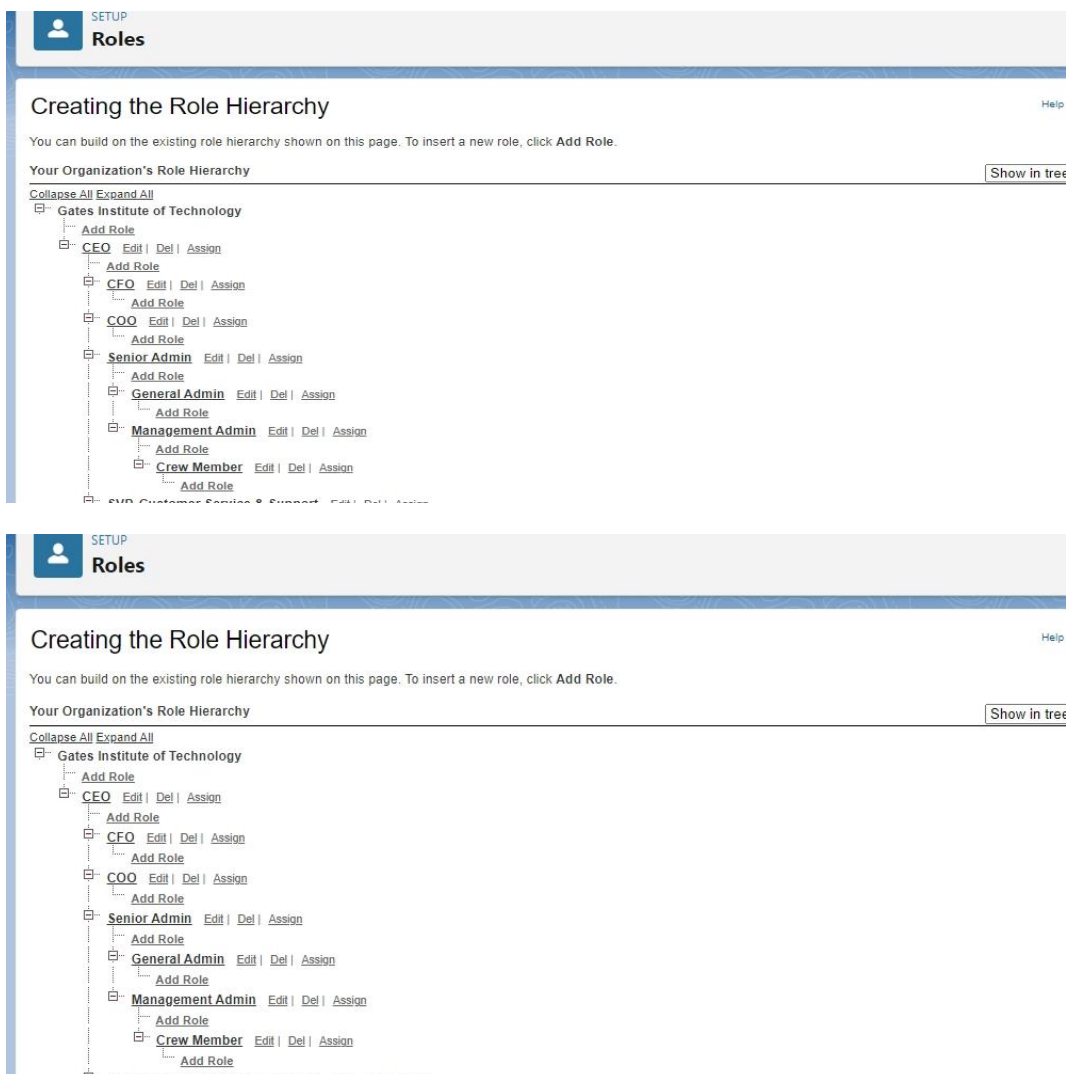
Custom App Settings

	Visible	Default		Visible	Default
AirLine Management System (AirLine_Management_System)	<input type="checkbox"/>	<input type="radio"/>	Platform (standard__Platform)	<input checked="" type="checkbox"/>	<input checked="" type="radio"/>
Analytics Studio (standard__Insights)	<input type="checkbox"/>	<input type="radio"/>	WDC (standard__Work)	<input type="checkbox"/>	<input type="radio"/>

Milestone 8: Role

A role in Salesforce defines a user's visibility access at the record level. Roles may be used to specify the types of access that people in your Salesforce organization can have to data.

OUTPUT SCREENSHOT



The image displays two screenshots of the Salesforce 'Roles' setup page, illustrating the 'Creating the Role Hierarchy' process. Both screenshots show a role hierarchy for the organization 'Gates Institute of Technology'.

Screenshot 1 (Top): Shows the role hierarchy with the following structure:

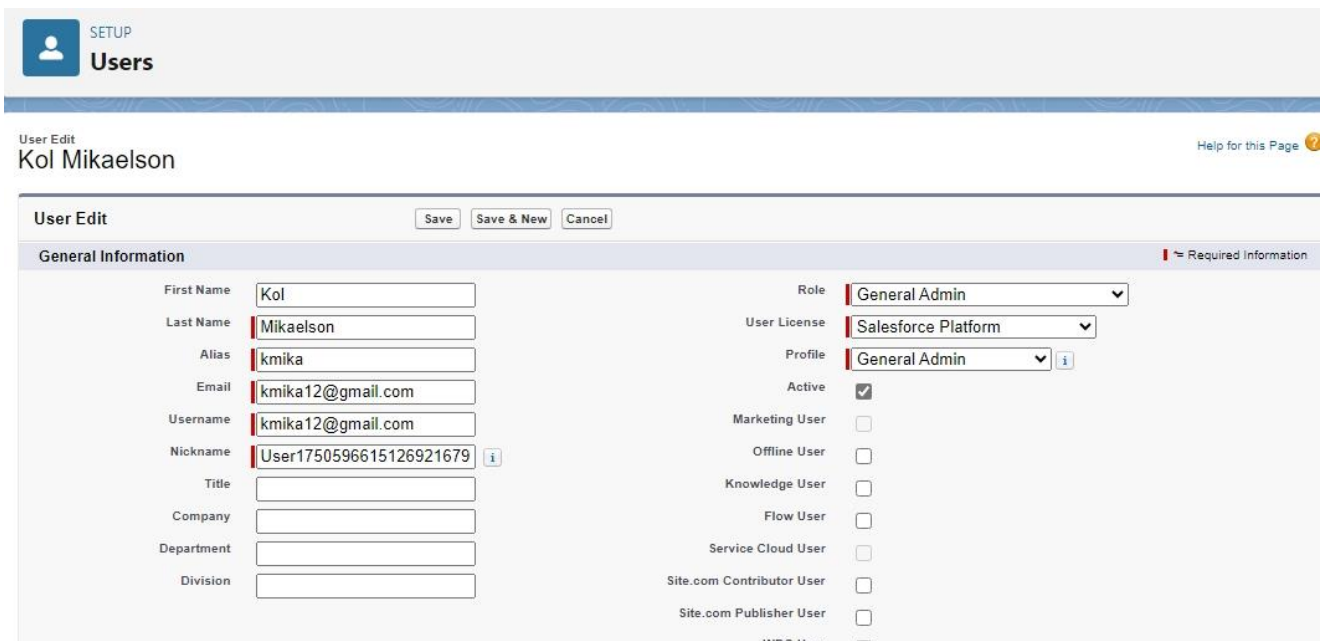
- Gates Institute of Technology** (Add Role)
 - CEO** (Edit | Del | Assign)
 - CFO** (Edit | Del | Assign)
 - COO** (Edit | Del | Assign)
 - Senior Admin** (Edit | Del | Assign)
 - General Admin** (Edit | Del | Assign)
 - Management Admin** (Edit | Del | Assign)
 - Crew Member** (Edit | Del | Assign)

Screenshot 2 (Bottom): Shows the same role hierarchy, but with the 'Add Role' button visible next to each role in the hierarchy, indicating the process of adding new roles.

Milestone 9: Users

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.

OUTPUT SCREENSHOT



The screenshot displays the 'User Edit' interface in Salesforce. At the top, there's a header with a 'SETUP' button and a 'Users' link. Below this, the user's name 'Kol Mikaelson' is shown. The main section is titled 'User Edit' and contains a 'General Information' tab. This tab is divided into two columns of fields. The left column includes 'First Name' (Kol), 'Last Name' (Mikaelson), 'Alias' (kmika), 'Email' (kmika12@gmail.com), 'Username' (kmika12@gmail.com), 'Nickname' (User1750596615126921679), 'Title', 'Company', 'Department', and 'Division'. The right column includes 'Role' (General Admin), 'User License' (Salesforce Platform), 'Profile' (General Admin), 'Active' (checked), 'Marketing User', 'Offline User', 'Knowledge User', 'Flow User', 'Service Cloud User', 'Site.com Contributor User', 'Site.com Publisher User', and 'WDC User'. A 'Required Information' icon is visible in the top right corner of the form area. Buttons for 'Save', 'Save & New', and 'Cancel' are located at the top of the form.

General Information	
First Name	Kol
Last Name	Mikaelson
Alias	kmika
Email	kmika12@gmail.com
Username	kmika12@gmail.com
Nickname	User1750596615126921679
Title	
Company	
Department	
Division	
Role	General Admin
User License	Salesforce Platform
Profile	General Admin
Active	<input checked="" type="checkbox"/>
Marketing User	<input type="checkbox"/>
Offline User	<input type="checkbox"/>
Knowledge User	<input type="checkbox"/>
Flow User	<input type="checkbox"/>
Service Cloud User	<input type="checkbox"/>
Site.com Contributor User	<input type="checkbox"/>
Site.com Publisher User	<input type="checkbox"/>
WDC User	<input type="checkbox"/>

SETUP

Users

User Edit

Niklaus Mikaelson

Help for this Page

User Edit

SaveSave & NewCancel

General Information

Required Information

First Name

Niklaus

Last Name

Mikaelson

Alias

nmika

Email

afaanahamed18@gmail.com

Username

afaanahamed18@gmail.com

Nickname

User1750596615126921679

Title

Company

Department

Division

Role

Senior Admin

User License

Salesforce

Profile

Senior Admin Profile

Active

☒

Marketing User

☐

Offline User

☐

Knowledge User

☐

Flow User

☐

Service Cloud User

☐

Site.com Contributor User

☐

Site.com Publisher User

☐

WDC User

☐

SETUP

Users

User Edit

Niklaus Mikaelson

Help for this Page

User Edit

SaveSave & NewCancel

General Information

Required Information

First Name

Niklaus

Last Name

Mikaelson

Alias

nmika

Email

afaanahamed18@gmail.com

Username

afaanahamed18@gmail.com

Nickname

User1750596615126921679

Title

Company

Department

Division

Role

Senior Admin

User License

Salesforce

Profile

Senior Admin Profile

Active

☒

Marketing User

☐

Offline User

☐

Knowledge User

☐

Flow User

☐

Service Cloud User

☐

Site.com Contributor User

☐

Site.com Publisher User

☐

WDC User

☐

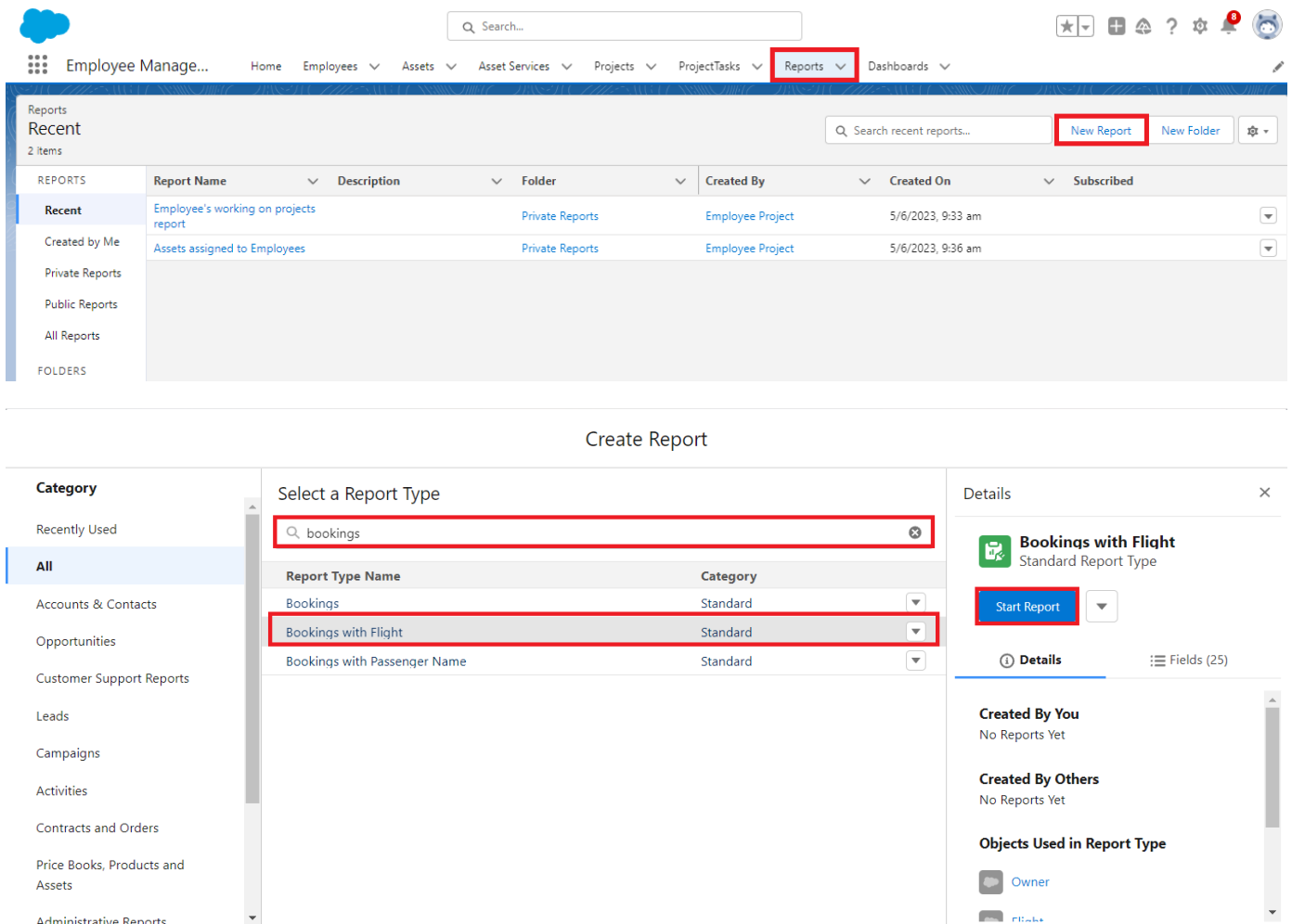
Milestone 10: Reports

- Created reports for:

Reports give you access to your Salesforce data. You can examine your Salesforce data in almost infinite combinations, display it in easy-to-understand formats, and share the resulting insights with others..

- Used grouping, filters, and field summaries.
- Enabled data-driven decision-making.

OUTPUT SCREENSHOT



The screenshot displays the Salesforce Reports interface. At the top, the navigation bar includes the Salesforce logo, a search bar, and various utility icons. The main navigation menu shows 'Employee Manage...' and a dropdown menu with 'Home', 'Employees', 'Assets', 'Asset Services', 'Projects', 'ProjectTasks', 'Reports' (highlighted with a red box), and 'Dashboards'.

The 'Reports' section is active, showing a 'Recent' list of 2 items. The list includes two reports: 'Employee's working on projects report' and 'Assets assigned to Employees', both created by 'Employee Project' on 5/6/2023. A 'New Report' button is highlighted with a red box.

Below the reports list, the 'Create Report' dialog is open. It features a 'Category' sidebar on the left with options like 'Recently Used', 'All', 'Accounts & Contacts', 'Opportunities', 'Customer Support Reports', 'Leads', 'Campaigns', 'Activities', 'Contracts and Orders', 'Price Books, Products and Assets', and 'Administrative Reports'. The main area is titled 'Select a Report Type' and contains a search bar with 'bookings' entered. A table lists three report types: 'Bookings', 'Bookings with Flight' (highlighted with a red box), and 'Bookings with Passenger Name', all categorized as 'Standard'. A 'Start Report' button is highlighted with a red box.

The right sidebar shows the 'Details' for the selected report type, 'Bookings with Flight'. It includes a 'Start Report' button, a 'Details' section with 'Created By You' and 'Created By Others' (both showing 'No Reports Yet'), and a section for 'Objects Used in Report Type' listing 'Owner' and 'FLIGHT'.



Report: Bookings with Passenger
Booking with Passenger Name

[Enable Field Editing](#)



[Add Chart](#)



Total Records
6

<input type="checkbox"/> Travellers ↑	Booking: Booking Id	Passenger: Passenger Id	Flight	Departs From	Departs To	Class	Passenger: Owner Name
<input type="checkbox"/> Adult (5)	Bk-0006	Bk-0006	air india	Bengaluru	Mumbai	First Class	Afaan Ahamed Yelamkur
	Bk-0005	Bk-0005	emirates	Mumbai	Delhi	Business	Afaan Ahamed Yelamkur
	Bk-0004	Bk-0006	emirates	Chennai	Mumbai	Business	Afaan Ahamed Yelamkur
	Bk-0003	Bk-0005	indigo	Bengaluru	Delhi	Economy	Afaan Ahamed Yelamkur
	Bk-0001	Bk-0002	emirates	Bengaluru	Delhi	Business	Afaan Ahamed Yelamkur
<input type="checkbox"/> Child (1)	Bk-0002	Bk-0004	air india	Kolkata	Delhi	First Class	Afaan Ahamed Yelamkur
Total (6)							



Report: Crews with Flight
Crew with Flight

[Enable Field Editing](#)



[Add Chart](#)



[Edit](#)



Total Records
6

Total Flight: PNR Number
4,848,484,969,360

<input type="checkbox"/> Crew Member ↑	Crew: Crew Id	Flight: Flight Name	Role	Flight: PNR Number	Flight: Departs From	Flight: Departs to	Flight: Arrival Time	Phone
<input type="checkbox"/> kiran (1)	Bk-0002	indigo	Pilot	1,212,121,242,325	Mumbai	Bengaluru	1:15 AM	📞 9000431655
<input type="checkbox"/> kumar (1)	Bk-0004	emirates	Flight Attendants	1,212,121,242,355	Delhi	Kolkata	5:30 AM	📞 9000431655
<input type="checkbox"/> rahul (1)	Bk-0003	indigo	Co-Pilot	1,212,121,242,325	Mumbai	Bengaluru	1:15 AM	📞 9000431655
<input type="checkbox"/> saii (1)	Bk-0005	emirates	Air Hosts	1,212,121,242,355	Chennai	Kolkata	5:45 AM	📞 9000431655
<input type="checkbox"/> shruti (1)	Bk-0006	air india	Air Hostesses	1,212,121,242,325	Mumbai	Chennai	3:15 PM	📞 9000431655
<input type="checkbox"/> sonu (1)	Bk-0001	air india	Air Hosts	1,212,121,242,325	Mumbai	Chennai	3:15 PM	📞 09000431655
Total (6)				4,848,484,969,360				

Milestone 11: Dashboards

- Designed dashboards showing:
 - Booking with flights, Booking with Passenger Name, Crew with Flight
- Used Donut, and summary widgets for visualization.

OUTPUT SCREENSHOT

Add Component

Report

Bookings with Flight Details

☐ Use chart settings from report

Display As

Sum of Flight: PNR Number

Sliced By

Travellers

Preview

Bookings with Flight Details

Sum of Flight: PNR Number

Travellers

- Adult
- Child
- Infant

457k

679k

1.3M

1.3M

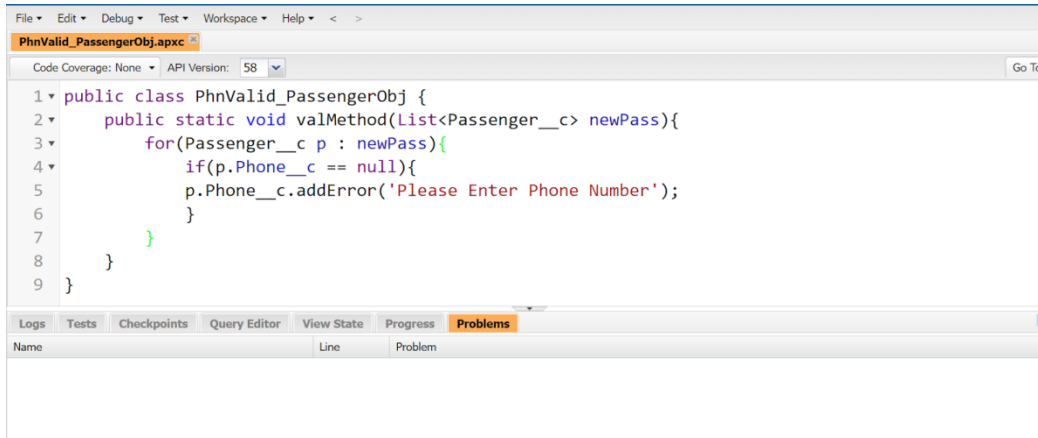
View Report (Bookings with Flight Details)

Cancel

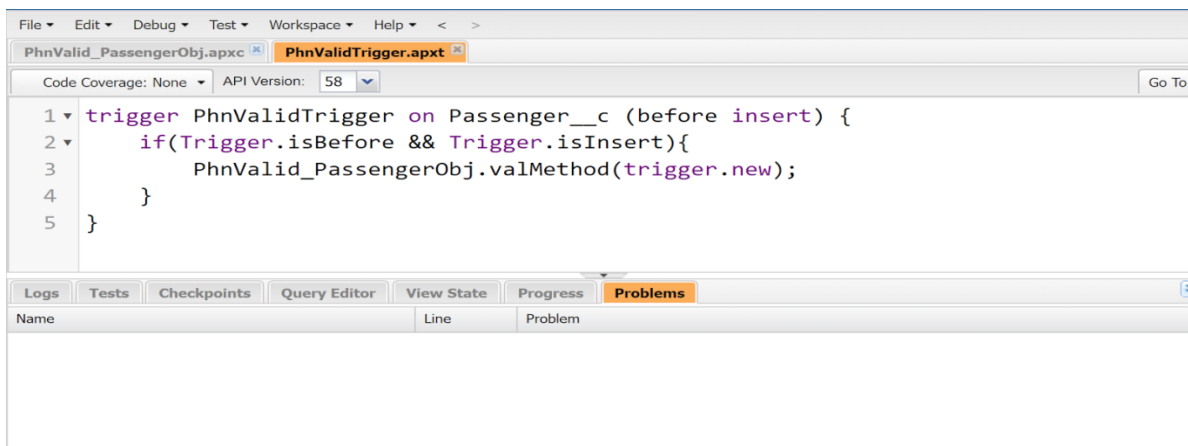
Add

Milestone 12: Apex

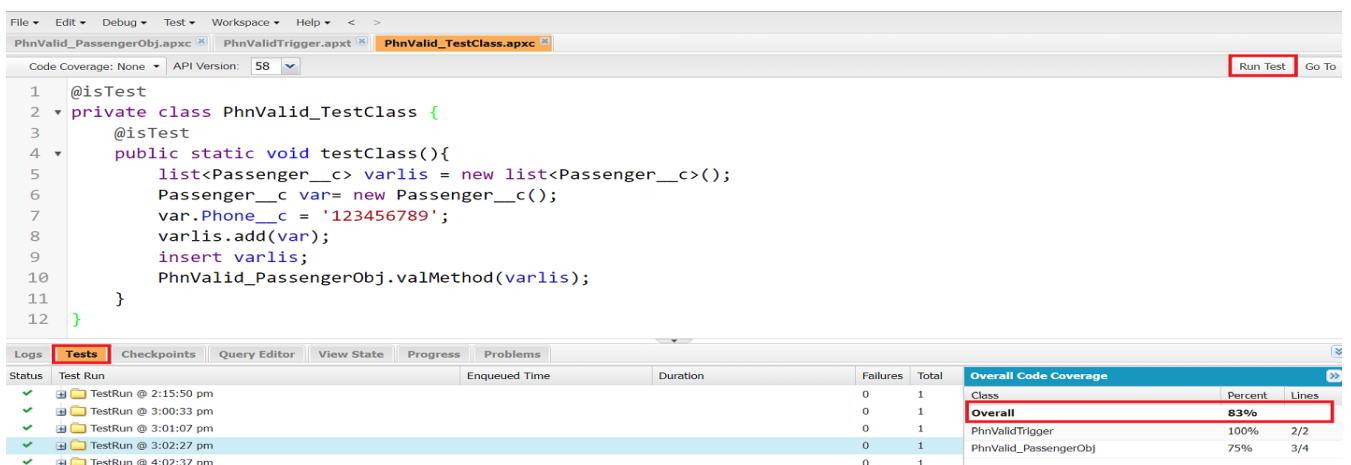
Apex is a strongly typed, object-oriented programming language that allows developers to execute flow and transaction control statements on the Lightning platform server in conjunction with calls to the Lightning Platform API. Using syntax that looks like Java and acts like database stored



```
1 public class PhnValid_PassengerObj {
2     public static void valMethod(List<Passenger__c> newPass){
3         for(Passenger__c p : newPass){
4             if(p.Phone__c == null){
5                 p.Phone__c.addError('Please Enter Phone Number');
6             }
7         }
8     }
9 }
```



```
1 trigger PhnValidTrigger on Passenger__c (before insert) {
2     if(Trigger.isBefore && Trigger.isInsert){
3         PhnValid_PassengerObj.valMethod(trigger.new);
4     }
5 }
```



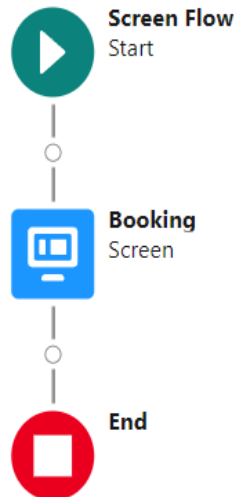
```
1 @isTest
2 private class PhnValid_TestClass {
3     @isTest
4     public static void testClass(){
5         list<Passenger__c> varlis = new list<Passenger__c>();
6         Passenger__c var= new Passenger__c();
7         var.Phone__c = '123456789';
8         varlis.add(var);
9         insert varlis;
10        PhnValid_PassengerObj.valMethod(varlis);
11    }
12 }
```

Status	Test Run	Enqueued Time	Duration	Failures	Total
✓	TestRun @ 2:15:50 pm			0	1
✓	TestRun @ 3:00:33 pm			0	1
✓	TestRun @ 3:01:07 pm			0	1
✓	TestRun @ 3:02:27 pm			0	1
✓	TestRun @ 4:02:37 pm			0	1

Overall Code Coverage		
Class	Percent	Lines
Overall	83%	
PhnValidTrigger	100%	2/2
PhnValid_PassengerObj	75%	3/4

Milestone 13: Flows

A sales representative needs to create a new booking for a customer who has expressed interest in a specific flight service. The representative wants to ensure that all relevant information is captured accurately and consistently.



Set Field Values for the Booking

Field	Value
Class__c	bookingObject > Class X
Departs_From__c	bookingObject > Departs From X
Departs_to__c	bookingObject > Departs to X
Departure_Date__c	bookingObject > Departure Date X
Flight__c	A bookingObject > Flight X
Passenger_Name__c	A bookingObject > Passenger Name X
Travellers__c	bookingObject > Travellers X

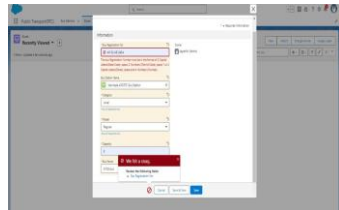
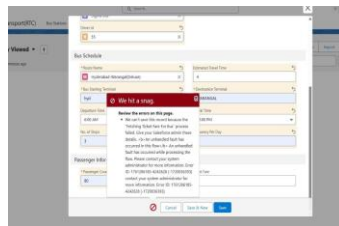
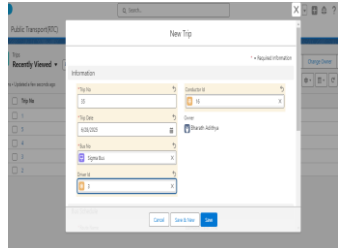
+ Add Field

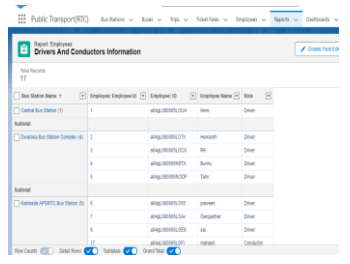
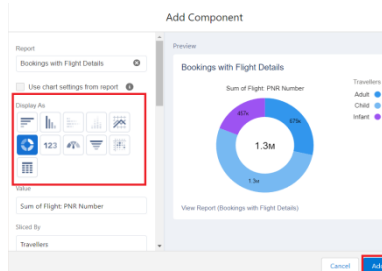
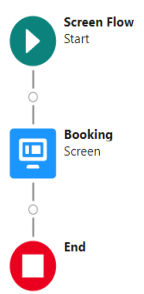
Cancel Done

Functional and Performance Testing

The Functional and Performance Testing Phase ensures that every feature implemented—such as object creation, field validation, automation, flows, triggers, reports, and dashboards—works exactly as intended. This phase guarantees stability, correctness, and user-readiness of your Airlines Management System. Where the system is validated for accuracy, behavior, and data correctness across all configured objects and logic.

◆ Performance Testing Summary Table

S. No	Parameter	Value/Observation	Screenshot Suggestion
1	Model Summary	Salesforce CRM setup for Airlines automation using Objects, Relationships, Reports, Flows, and Triggers. <i>Note:</i> Test data import only succeeds if object relationships and formats are correct. Mismatched values raise validation errors.	
2	Field Validations	Tested rules such as: - Age must be above 21 - Phone must be 10 digits - Mandatory fields cannot be empty. System blocks incorrect data.	
3	Automation Accuracy (Flow + Trigger)	- Flow: Auto-fetch ticket fare when Trip is created based on route & bus model. - Trigger: Checks if selected employee role matches “Pilot” or “Cabin crew” on Trip. - All logic executed successfully in test runs.	

4	Reports Testing	<ul style="list-style-type: none"> - Trip Summary Report shows correct grouping by route & date. - Fare amounts and passenger counts calculated accurately using formula fields. - Report download & export functions verified. 	 <p>The screenshot shows a web application interface for 'Public TransportPTG'. It displays a report titled 'Drivers And Conductors Information'. The report includes a table with columns for Employee ID, Employee Name, and Role. The table lists several employees, including 'Hem', 'Neha', 'Ravi', 'Surya', 'Tari', 'Jyoti', 'Jyoti', 'Jyoti', 'Jyoti', and 'Jyoti'. The interface also includes filters for 'Bus Station Name' and 'Employee ID'.</p>
5	Dashboard Verification	<ul style="list-style-type: none"> - Dashboards built using correct source reports. - Visuals reflect live Trip data, fare summaries, passenger totals. - Tested filter logic and component refresh. 	 <p>The screenshot shows a 'Add Component' dialog box. It features a 'Preview' section on the right displaying a donut chart titled 'Bookings with Flight Details'. The chart shows a total of '1.3M' and is segmented by 'Transit Mode' (Adult, Child, Infant). The left section shows the 'Report' settings, including 'Bookings with Flight Details' and 'Sum of Flight: PNR Number'.</p>
6	Data Accuracy (Manual + Automated)	<ul style="list-style-type: none"> - Manual test records were entered to test validation. - Flow logic verified for multiple ticket fare combinations. - Output matched expected values in all test cases. 	 <p>The screenshot shows a screen flow diagram. It starts with a green circle labeled 'Screen Flow Start', followed by a blue square labeled 'Booking Screen', and ends with a red circle labeled 'End'. The flow is indicated by arrows connecting the elements.</p>

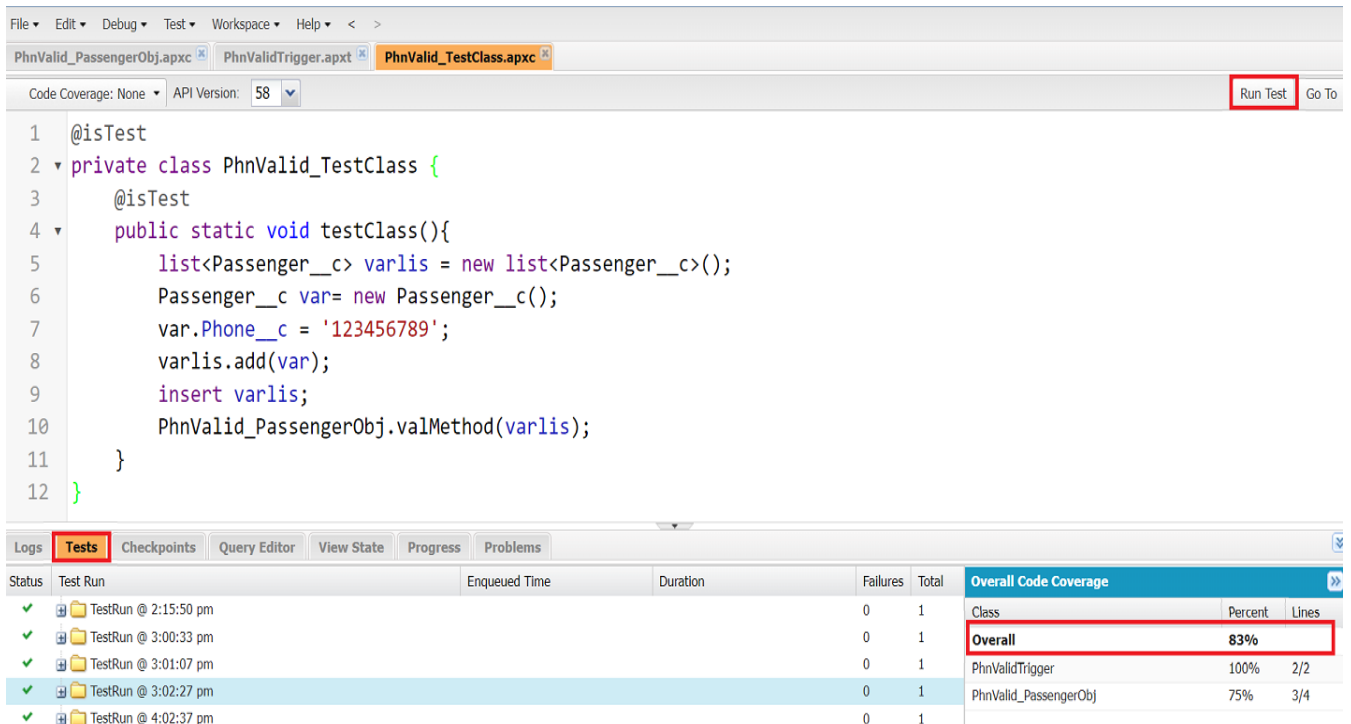
RESULTS

Output Screenshots



The screenshot shows the code editor for `PhnValidTrigger.apxt`. The code defines a trigger `PhnValidTrigger` on the `Passenger__c` object, triggered before insert. The trigger logic checks if the trigger is before and insert, then calls `PhnValid_PassengerObj.valMethod(trigger.new)`.

```
1 trigger PhnValidTrigger on Passenger__c (before insert) {  
2     if(Trigger.isBefore && Trigger.isInsert){  
3         PhnValid_PassengerObj.valMethod(trigger.new);  
4     }  
5 }
```



The screenshot shows the code editor for `PhnValid_TestClass.apxc` and the Test Results table. The code defines a test class `PhnValid_TestClass` with a static method `testClass()` that creates a list of `Passenger__c` objects, adds a new object with phone number '123456789', and calls `PhnValid_PassengerObj.valMethod(varlis)`.

```
1 @isTest  
2 private class PhnValid_TestClass {  
3     @isTest  
4     public static void testClass(){  
5         list<Passenger__c> varlis = new list<Passenger__c>();  
6         Passenger__c var= new Passenger__c();  
7         var.Phone__c = '123456789';  
8         varlis.add(var);  
9         insert varlis;  
10        PhnValid_PassengerObj.valMethod(varlis);  
11    }  
12 }
```

The Test Results table shows the following data:

Status	Test Run	Enqueued Time	Duration	Failures	Total	Overall Code Coverage
✓	TestRun @ 2:15:50 pm			0	1	
✓	TestRun @ 3:00:33 pm			0	1	
✓	TestRun @ 3:01:07 pm			0	1	
✓	TestRun @ 3:02:27 pm			0	1	
✓	TestRun @ 4:02:37 pm			0	1	

The Overall Code Coverage table shows the following data:

Class	Percent	Lines
Overall	83%	
PhnValidTrigger	100%	2/2
PhnValid_PassengerObj	75%	3/4

Add Component

Report

Bookings with Flight Details

☐ Use chart settings from report

Display As

123

Value

Sum of Flight: PNR Number

Sliced By

Travellers

Preview

Bookings with Flight Details

Sum of Flight: PNR Number

457k

679k

1.3M

1.3M

1.3M

Travellers

Adult

Child

Infant

View Report (Bookings with Flight Details)

Cancel

Add

Screen Flow

Start

Booking Screen

End

ADVANTAGES & DISADVANTAGES

Advantages:

- **Real-time Data Validation and Automation:**
Ensures flight schedules, crew assignments, and fare calculations are always accurate and up to date.
- **Modular Structure:**
Supports **easy customization** for different airlines, routes, airplane models, and employee roles.
- **Centralized and Streamlined Process:**
Provides an integrated system to manage **flights, fares, airports, airplanes, and employee assignments** from a single platform.
- **Visual Reports and Dashboards:**
Offers **intuitive reporting and real-time dashboards** for airline management to make faster, data-driven decisions.

Disadvantages:

- **Salesforce Skill Requirement:**
Advanced usage, troubleshooting, and customization require **Salesforce platform knowledge**.
- **Cloud Dependency:**
Limited offline functionality due to reliance on **Salesforce's cloud-based environment**.
- **Data Integrity Dependency:**
System heavily relies on **correctly configured object relationships**; misconfigurations can impact performance and accuracy.

CONCLUSION

The project successfully delivered a **custom Salesforce CRM solution for airline operations**. It addressed core inefficiencies by:

- Automating complex processes like **crew assignments and fare calculations**
- Centralizing flight management workflows
- Improving data accuracy and operational visibility

Overall, this CRM solution offers a **scalable, maintainable, and user-friendly platform** that can significantly improve airline management efficiency.

FUTURE SCOPE

The Airlines CRM can be further enhanced with advanced features and integrations to improve scalability, user engagement, and decision-making capabilities. Potential future developments include:

- Third-Party Ticketing and Payment System Integration
- Automated Alerts and Communication Features
- Mobile-First User Interface with Salesforce Experience Cloud
- AI-Powered Forecasting with Salesforce

APPENDIX:

The screenshot displays an IDE interface with an Apex class file named `PhnValid_TestClass.apxc` open. The code defines a test class `PhnValid_TestClass` with a static method `testClass()` that creates a list of `Passenger__c` objects, adds one with a specific phone number, and calls `valMethod` on a `PhnValid_PassengerObj` instance.

Below the code editor, the **Tests** tab is active, showing a table of test runs. The last test run at 4:02:37 pm is selected, showing a status of 'Success' and 0 failures. To the right, the **Overall Code Coverage** table shows an overall coverage of 83%.

Status	Test Run	Enqueued Time	Duration	Failures	Total
✓	TestRun @ 2:15:50 pm			0	1
✓	TestRun @ 3:00:33 pm			0	1
✓	TestRun @ 3:01:07 pm			0	1
✓	TestRun @ 3:02:27 pm			0	1
✓	TestRun @ 4:02:37 pm			0	1

Overall Code Coverage		
Class	Percent	Lines
Overall	83%	
PhnValidTrigger	100%	2/2
PhnValid_PassengerObj	75%	3/4

- GitHub Link:
<https://github.com/afaan10/Airlines-Management-System>