Airline Management System

Date	25-06-2025
Team ID	**LTVIP2025TMID31460**
Project Name	**AirLine Management System**
College Name	**Ideal Institute Of Technology**

TEAM MEMBERS

1.Team Leader : Tivari Sai Balaji Santosh Prasad

Mail:santutivari06@gmail.com

Role:Developer-flight

2. Team member : Vijay

Mail:kudipudivijay6@gmail.com

Role : Developer-booking

3. Team member : Siri Babu

Mail:siribabu678@gmail.com

Role : Tester

4.Team member : Thalatam Satya Harini

Mail:talatamsatyaharini@gmail.com

Role: Document Creator

AIRLINE
MANAGEMENT
SYSTEM

1.1 Introduction

The Airline Management System is a Salesforce-based project developed as part of the SmartInternz Virtual Internship Program. This project aims to provide a digital solution for managing key airline operations such as flight scheduling, passenger management, bookings, and crew assignments. Built entirely on the Salesforce platform, the system utilizes essential components like custom objects, Apex triggers, validation rules, reports, and dashboards to ensure streamlined workflows and accurate data handling.

1.2 Project Overview

Developed a cloud-based Airline Management System using the Salesforce platform.Manages key modules: Flights, Passengers, Bookings, and Crew.Implements automation using Apex triggers and validation rules.Provides real-time data insights through reports and dashboards.

1.3 Project Purpose

- To simplify and automate core airline operations in a centralized system.

To ensure **data accuracy** through validation and business logic.

IDEATION PHASE

2.1 Problem Statement

- Manual airline processes cause data errors and inefficiency.
- No centralized system for flights, bookings, passengers, and crew.
- Lack of real-time reports and user-friendly interface.
- Need for validation to ensure correct data entry.
- Requires an automated, cloud-based solution for smooth operations.

2.2 Brainstorming Ideas

Create objects for Flights, Passengers, Bookings, and Crew.

Add validation (e.g., phone number check using Apex Trigger)

Build dashboards to visualize bookings and flight data.

Assign crew to flights and manage roles.

Design a Lightning Home Page with a welcome message and key metrics.

Enable easy creation and tracking of bookings with success alerts.

2.3 Empathy Map

Thinks and Feels:"Is this the best flight option?"

Hear:Feedback from friends or agents about airline services.

Sees: Multiple flight options across different apps or websites

Gains: Fast and secure flight booking process.

3. Requirement Analysis

3.1 Customer Journey Map:

Open App → 2. Search Flights → 3. Select Flight → 4. Enter Details → 5. Book Ticket → 6. Receive Confirmation

3.2 Solution Requirement:

Functional Requirements

1. Flight Management
2. Passenger Management
3. Booking System
4. Crew Management
5. Dashboard and Reporting
6. Lightning App Home Page

❖ Non-Functional Requirements

- Data Accuracy: Enforced through validation logic and triggers.
  Ease of Use: User-friendly UI with a Lightning App Home Page.
- Scalability: Structured to support additional features in the future.
- Security: Role-based access to records and dashboards.

3.3 Data flow Diagram (DFD)

User Input → Booking System → Database → Confirmation Output

3.4 Technical Stack – Tools Used

- Salesforce Lightning Apex,
- Visualforce, SOQL, Custom Objects
- Validation Rules, Apex Test Classes, Data Loader, Workbench, Change Sets
- GitHub

4. Project Design

4.1 Problem Solution Fit:

The project addresses airline data and booking issues by offering an automated and centralized system on Salesforce.

4.2 Proposed Solution:

A cloud-based Airline Management System built on the Salesforce platform that manages Flights, Passengers, Bookings, and Crew with data validation, dashboards, and a user-friendly interface.

4.3 Solution Architecture:

The solution is designed using Salesforce components such as custom objects (Flight, Passenger, Booking, Crew), Apex triggers for data validation, Lightning App Builder for the home page, and reports/dashboards for data visualization. These elements interact to form a cloud-based architecture that ensures centralized management, real-time insights, and streamlined airline operations.

5. Project Planning & Scheduling

- Week 1: Requirement Analysis – Identified the core modules such as Flights, Passengers, Bookings, and Crew.
- Week 2: System Design – Designed custom objects, fields, and their relationships in Salesforce.
- Week 3: Development Phase I – Created custom objects and page layouts for each module.
- Week 4: Development Phase II – Implemented Apex triggers and validation rules (e.g., phone number check).

- Week 5: Testing – Conducted functional and performance testing using test classes and trigger validation.
- Week 6: UI Setup – Customized the Lightning App Home Page and added a welcome message.
- Week 7: Reports & Dashboards – Built visual reports and dashboards for booking and flight analytics.
- Week 8: Finalization – Compiled documentation, prepared output screenshots, and submitted the project.

6. Development Phase

- During the Development Phase, the core functionality of the Airline Management System was built out in two iterative sub-phases:

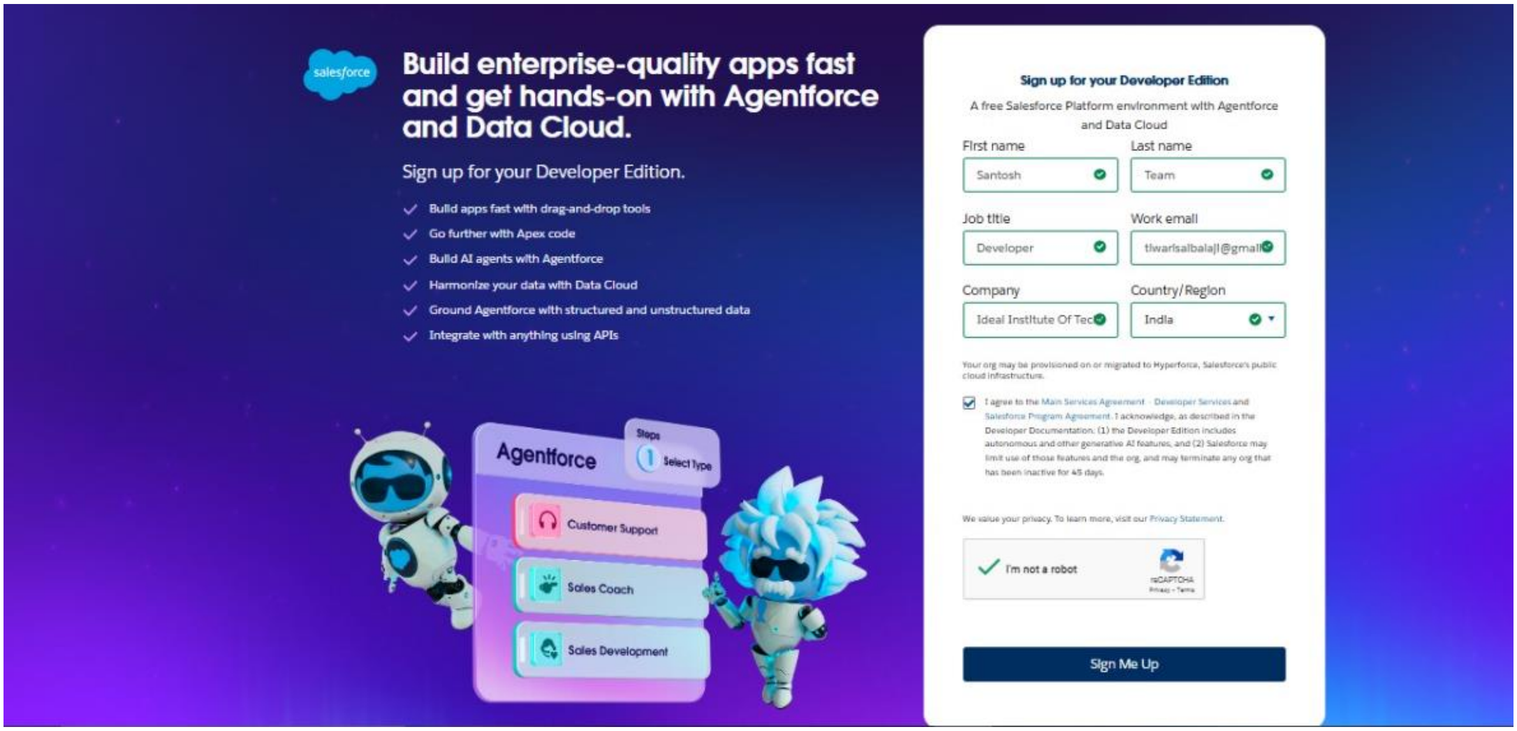
A Lightning App was configured with custom tabs and a dedicated Home Page, which later included a welcome message and dashboard view. Sample data was inserted to test the flow of records and to validate the business logic. Test classes were written to ensure proper code coverage and reliable performance.

This phase successfully laid the groundwork for testing, UI customization, and report/dashboard creation in the following stages.

PROJECT DEVELOPMENT PHASE

Date	**25-06-2025**
Team Id	**LTVIP2025TMID29760**
Project Name	**AirLine Management System**
College Name	**Ideal Institute Of Technology**

- Created developer org and explored platform features



Build enterprise-quality apps fast and get hands-on with Agentforce and Data Cloud.

Sign up for your Developer Edition.

- ✓ Build apps fast with drag-and-drop tools
- ✓ Go further with Apex code
- ✓ Build AI agents with Agentforce
- ✓ Harmonize your data with Data Cloud
- ✓ Ground Agentforce with structured and unstructured data
- ✓ Integrate with anything using APIs

Sign up for your Developer Edition

A free Salesforce Platform environment with Agentforce and Data Cloud

First name	Last name
Santosh	Team
Job title	Work email
Developer	tiwarisaibalaji@gmail
Company	Country/Region
Ideal Institute Of Tec	India

Your org may be provisioned on or migrated to Hyperforce, Salesforce's public cloud infrastructure.

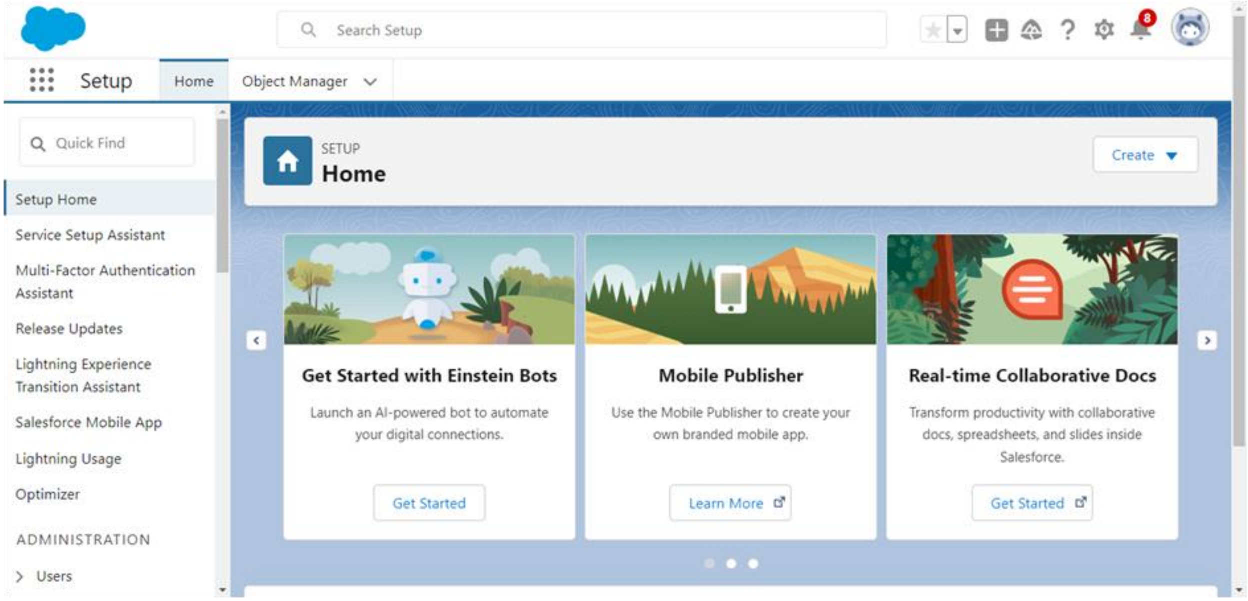
☑ I agree to the Main Services Agreement - Developer Services and Salesforce Program Agreement. I acknowledge, as described in the Developer Documentation: (1) the Developer Edition includes autonomous and other generative AI features, and (2) Salesforce may limit use of those features and the org, and may terminate any org that has been inactive for 45 days.

We value your privacy. To learn more, visit our Privacy Statement.

✓ I'm not a robot

reCAPTCHA
Privacy - Terms

Sign Me Up



Get Started with Einstein Bots

Launch an AI-powered bot to automate your digital connections.

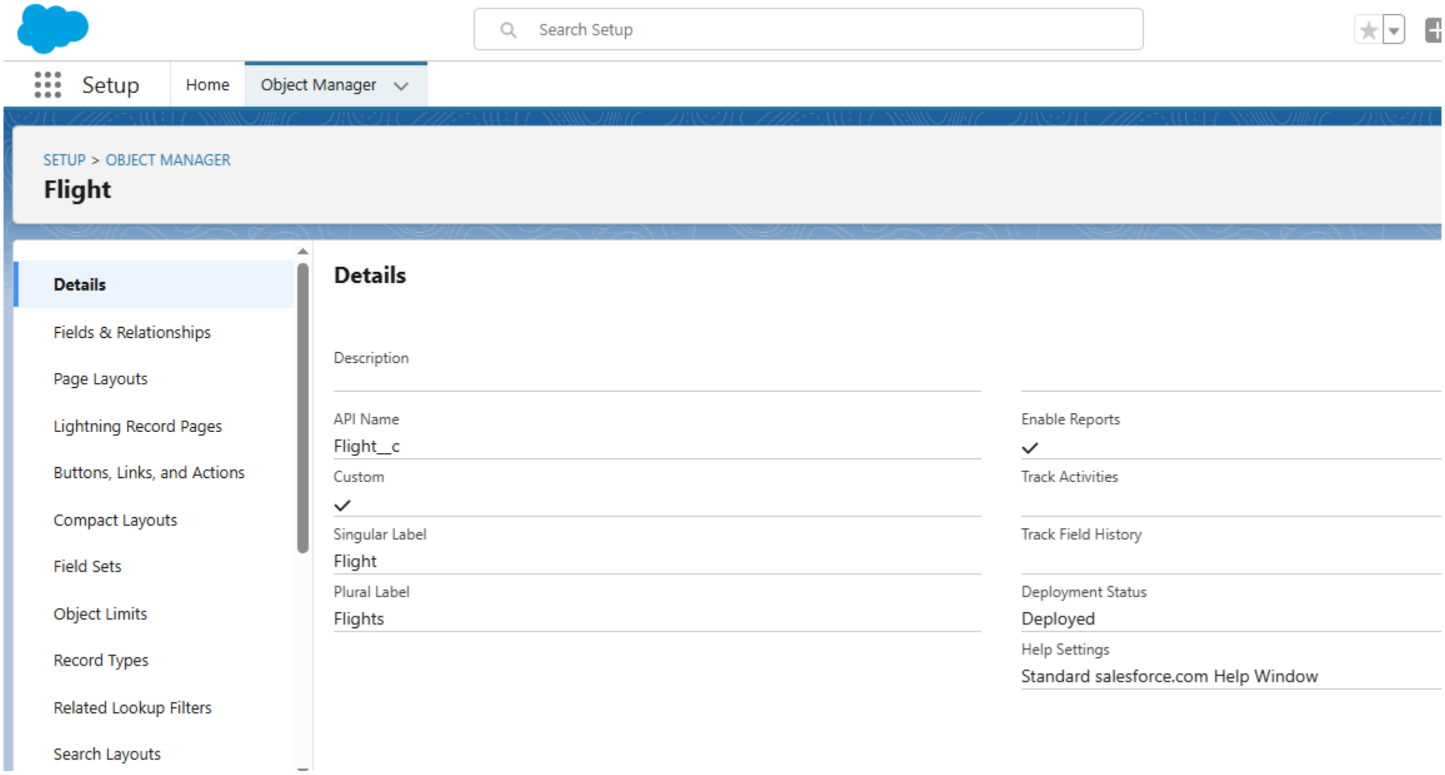
Mobile Publisher

Use the Mobile Publisher to create your own branded mobile app.

Real-time Collaborative Docs

Transform productivity with collaborative docs, spreadsheets, and slides inside Salesforce.

- Object and tab creation



Description

API Name
Flight__c

Custom
✓

Singular Label
Flight

Plural Label
Flights

Enable Reports
✓

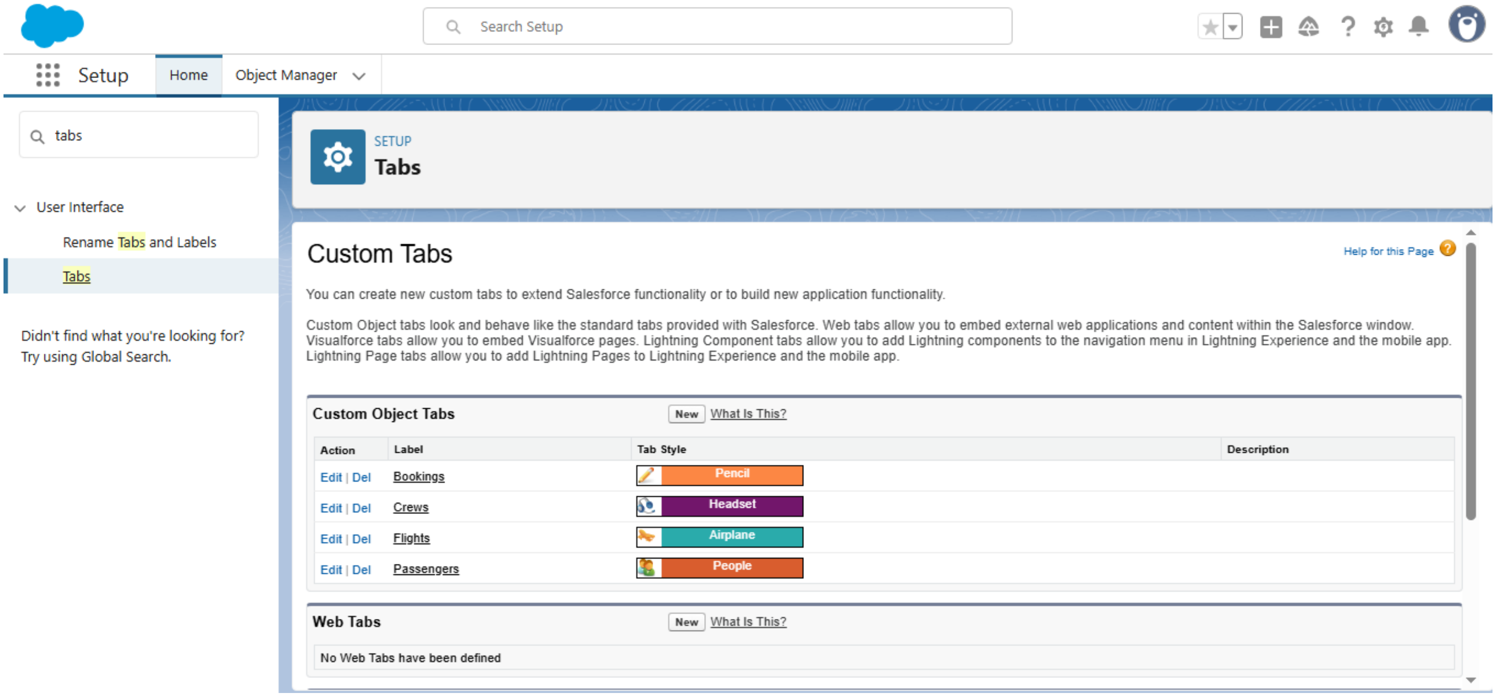
Track Activities

Track Field History

Deployment Status
Deployed

Help Settings
Standard salesforce.com Help Window

- Designed core objects: Passenger, Booking, Flight, Crew



Custom Tabs

Help for this Page

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 window. 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 the mobile app. Lightning Page tabs allow you to add Lightning Pages to Lightning Experience and the mobile app.

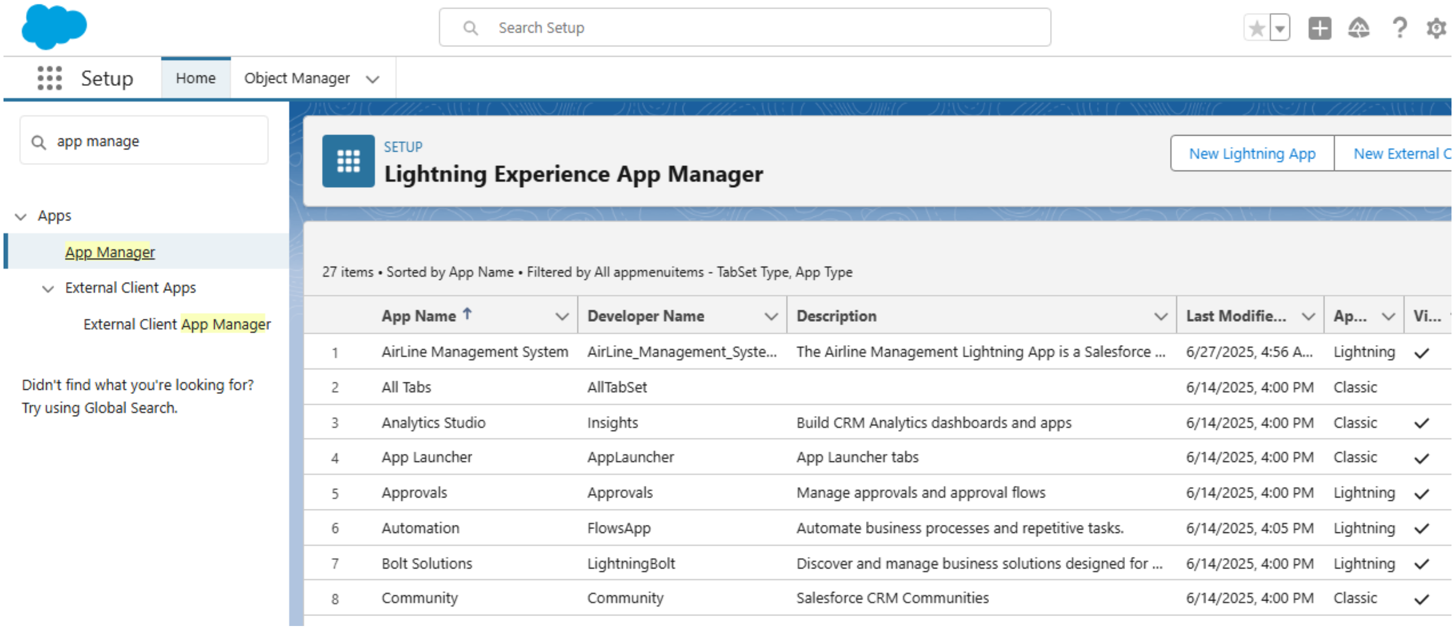
Custom Object Tabs New What Is This?

Action	Label	Tab Style	Description
Edit | Del	Bookings	Pencil	
Edit | Del	Crews	Headset	
Edit | Del	Flights	Airplane	
Edit | Del	Passengers	People	

Web Tabs New What Is This?

No Web Tabs have been defined

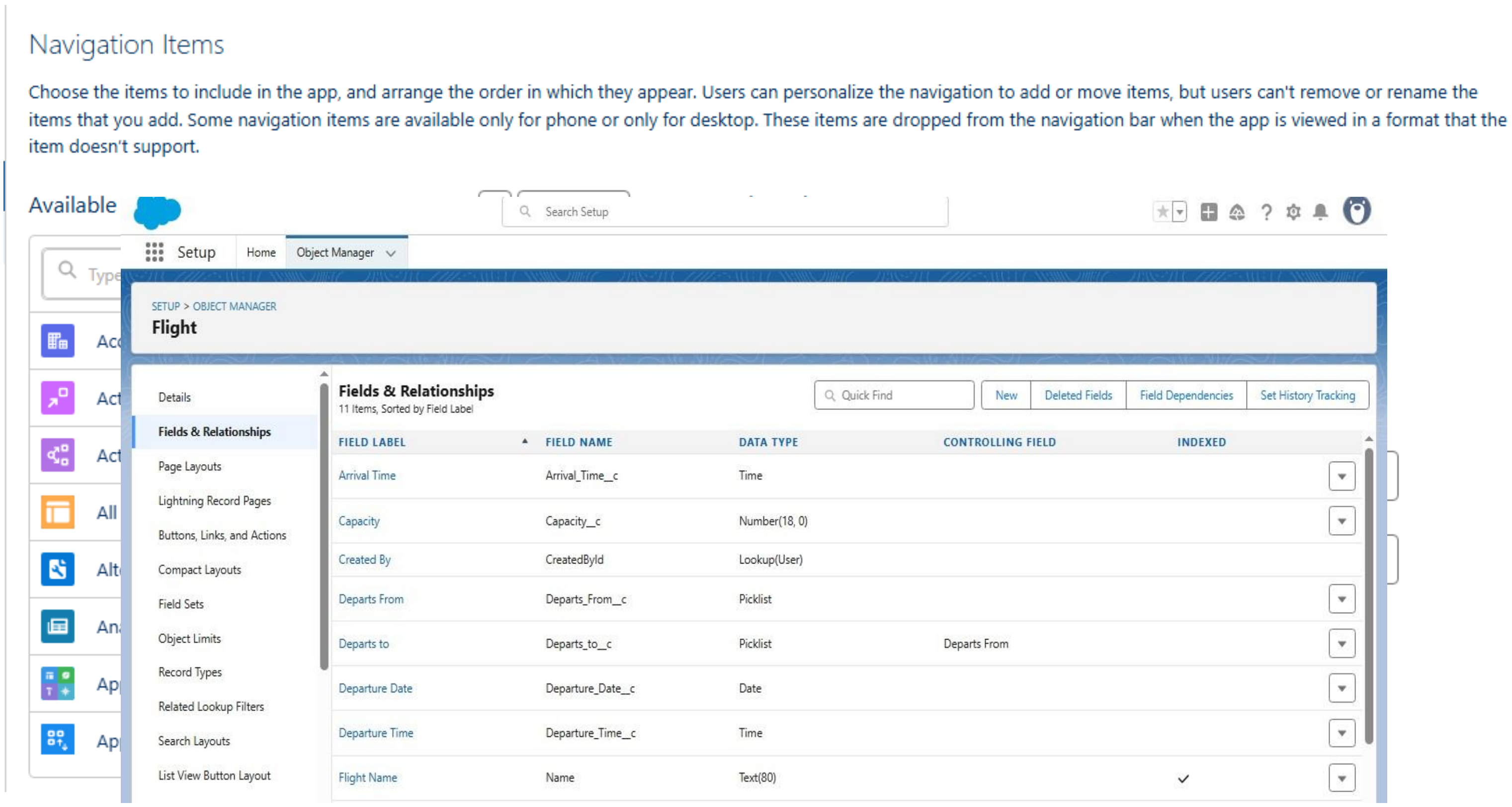
Lightning App Development


Search Setup
Setup
Home
Object Manager
app manage
Apps
App Manager
External Client Apps
External Client App Manager
Didn't find what you're looking for? Try using Global Search.
SETUP
Lightning Experience App Manager
New Lightning App
New External C

27 items • Sorted by App Name • Filtered by All appmenuitems - TabSet Type, App Type

	App Name	Developer Name	Description	Last Modifie...	Ap...	Vi...
1	AirLine Management System	AirLine_Management_Syste...	The Airline Management Lightning App is a Salesforce ...	6/27/2025, 4:56 A...	Lightning	✓
2	All Tabs	AllTabSet		6/14/2025, 4:00 PM	Classic	
3	Analytics Studio	Insights	Build CRM Analytics dashboards and apps	6/14/2025, 4:00 PM	Classic	✓
4	App Launcher	AppLauncher	App Launcher tabs	6/14/2025, 4:00 PM	Classic	✓
5	Approvals	Approvals	Manage approvals and approval flows	6/14/2025, 4:00 PM	Lightning	✓
6	Automation	FlowsApp	Automate business processes and repetitive tasks.	6/14/2025, 4:05 PM	Lightning	✓
7	Bolt Solutions	LightningBolt	Discover and manage business solutions designed for ...	6/14/2025, 4:00 PM	Lightning	✓
8	Community	Community	Salesforce CRM Communities	6/14/2025, 4:00 PM	Classic	✓

- Created “Airline Management System” app with navigation tabs



Navigation Items

Choose the items to include in the app, and arrange the order in which they appear. Users can personalize the navigation to add or move items, but users can't remove or rename the items that you add. Some navigation items are available only for phone or only for desktop. These items are dropped from the navigation bar when the app is viewed in a format that the item doesn't support.

SETUP > OBJECT MANAGER

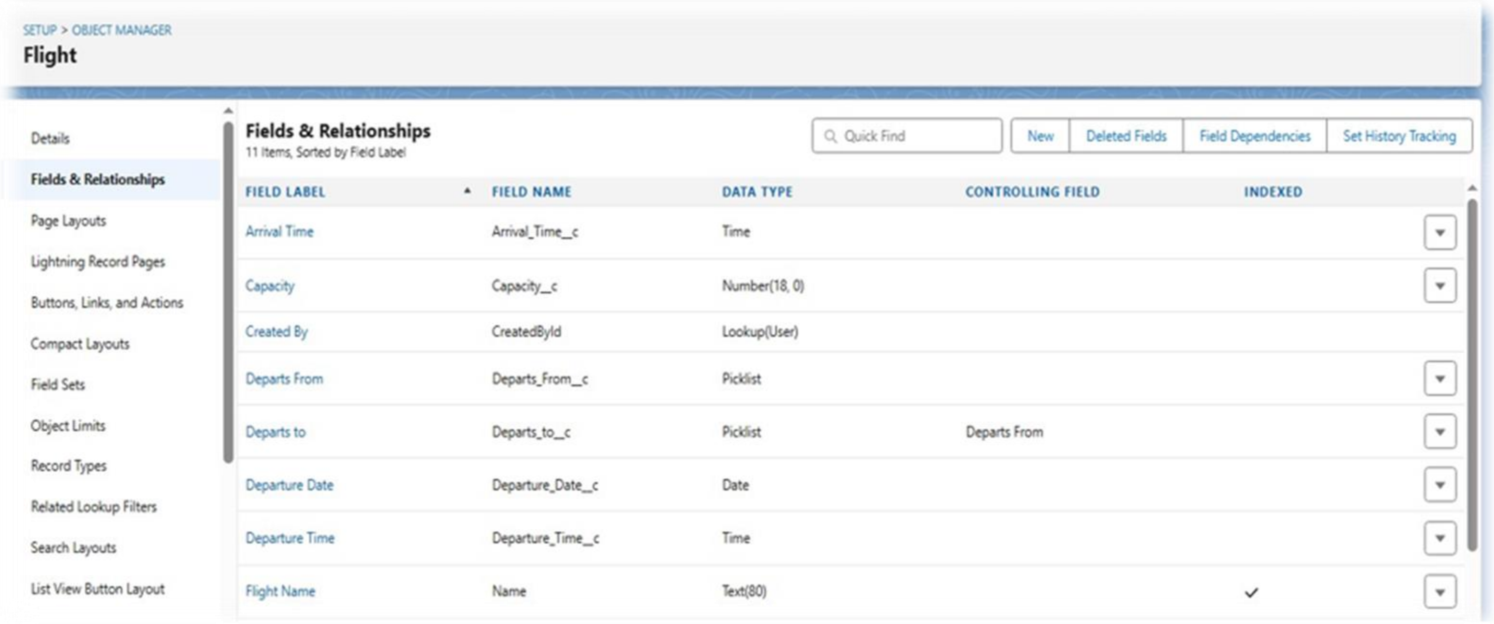
Flight

Fields & Relationships

11 Items, Sorted by Field Label

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Arrival Time	Arrival_Time__c	Time		
Capacity	Capacity__c	Number(18, 0)		
Created By	CreatedById	Lookup(User)		
Departs From	Departs_From__c	Picklist		
Departs to	Departs_to__c	Picklist	Departs From	
Departure Date	Departure_Date__c	Date		
Departure Time	Departure_Time__c	Time		
Flight Name	Name	Text(80)		✓

- Added custom fields, picklists, lookup relationships & field dependencies



SETUP > OBJECT MANAGER

Flight

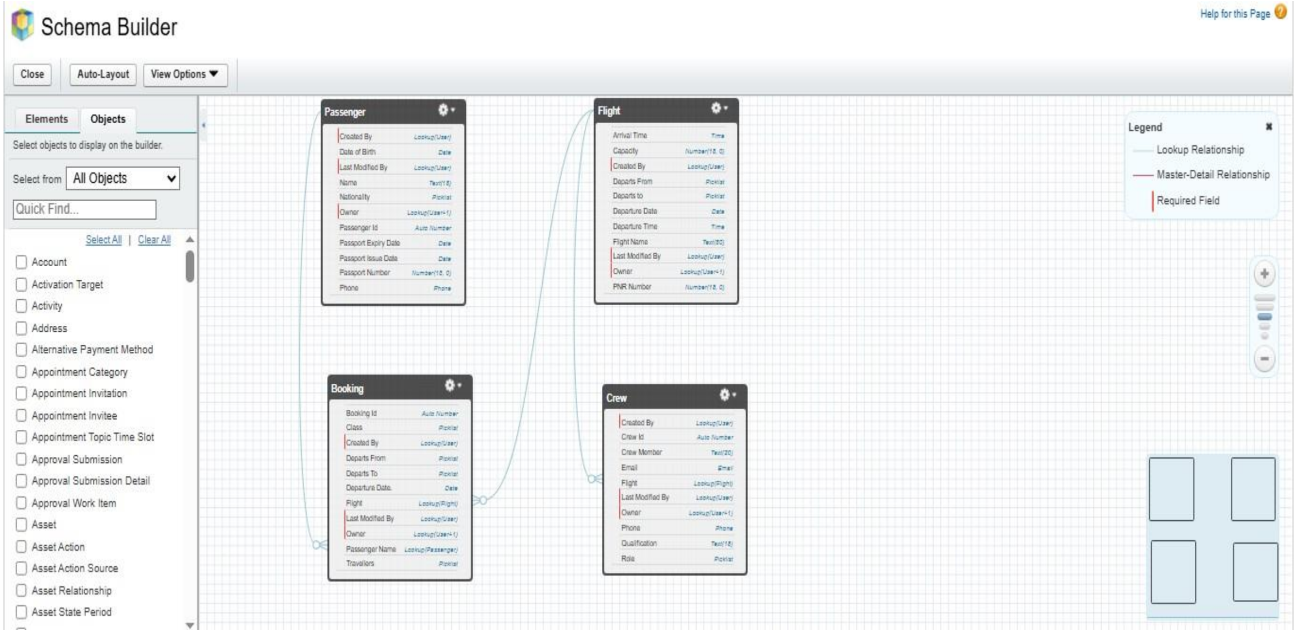
Fields & Relationships

11 Items, Sorted by Field Label

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Arrival Time	Arrival_Time__c	Time		
Capacity	Capacity__c	Number(18, 0)		
Created By	CreatedById	Lookup(User)		
Departs From	Departs_From__c	Picklist		
Departs to	Departs_to__c	Picklist	Departs From	
Departure Date	Departure_Date__c	Date		
Departure Time	Departure_Time__c	Time		
Flight Name	Name	Text(80)		✓

Schema Builder

Visualized object relationships using Salesforce Schema Builder



Passenger	
Created By	Lookup(User)
Date of Birth	Date
Last Modified By	Lookup(User)
Name	Text(18)
Nationality	Picklist
Owner	Lookup(User+1)
Passenger Id	Auto Number
Passport Expiry Date	Date
Passport Issue Date	Date
Passport Number	Number(18, 0)
Phone	Phone

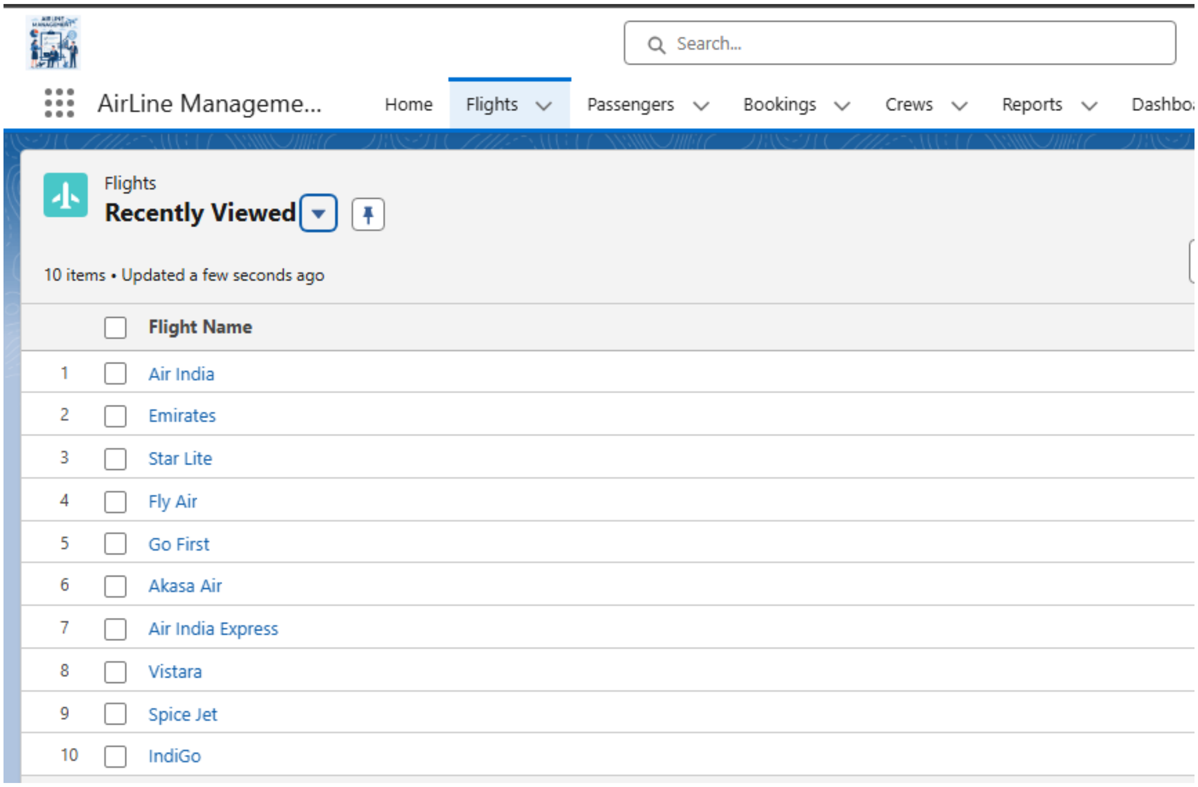
Flight	
Arrival Time	Time
Capacity	Number(18, 0)
Created By	Lookup(User)
Departs From	Picklist
Departs to	Picklist
Departure Date	Date
Departure Time	Time
Flight Name	Text(80)
Last Modified By	Lookup(User)
Owner	Lookup(User+1)
PNR Number	Number(18, 0)

Booking	
Booking Id	Auto Number
Class	Picklist
Created By	Lookup(User)
Departs From	Picklist
Departs To	Picklist
Departure Date	Date
Flight	Lookup(Flight)
Last Modified By	Lookup(User)
Owner	Lookup(User+1)
Passenger Name	Lookup(Passenger)
Travellers	Picklist

Crew	
Created By	Lookup(User)
Crew Id	Auto Number
Crew Member	Text(20)
Email	Email
Flight	Lookup(Flight)
Last Modified By	Lookup(User)
Owner	Lookup(User+1)
Phone	Phone
Qualification	Text(18)
Role	Picklist

Data Creation & Record Entry

Entered 10 records per object via Lightning interface

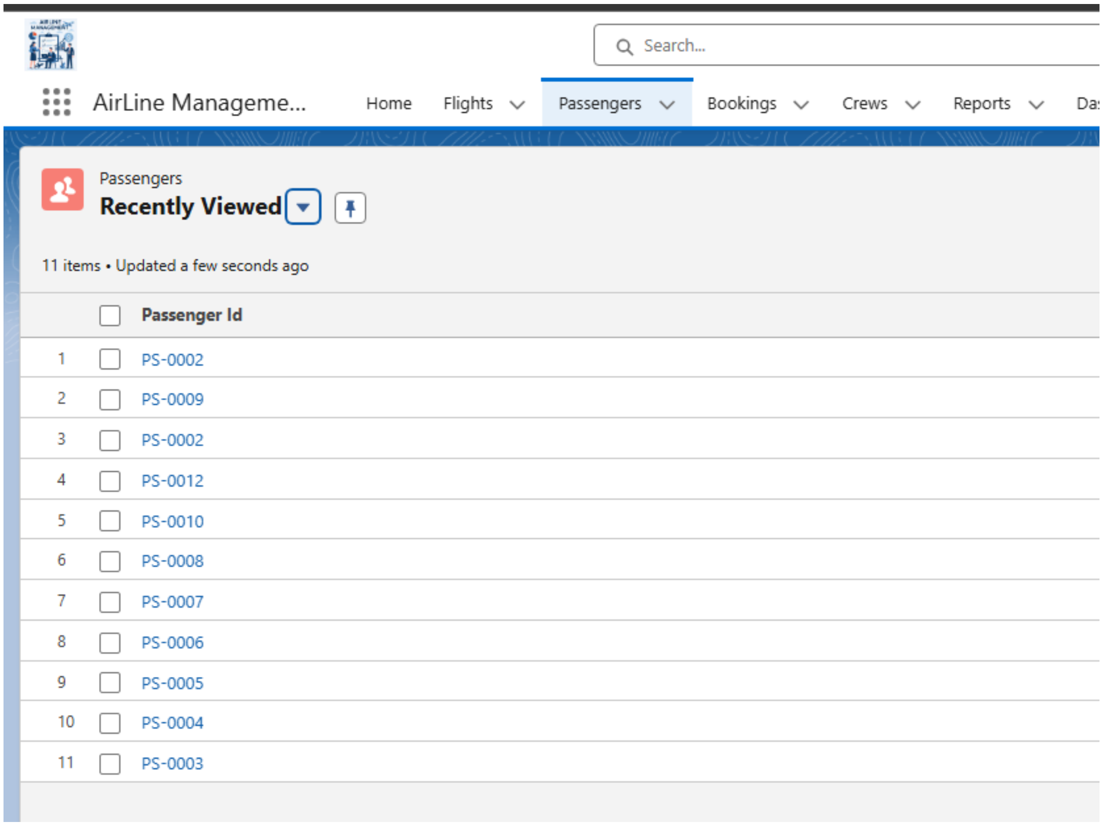


Flights

Recently Viewed

10 items • Updated a few seconds ago

	☐ Flight Name
1	☐ Air India
2	☐ Emirates
3	☐ Star Lite
4	☐ Fly Air
5	☐ Go First
6	☐ Akasa Air
7	☐ Air India Express
8	☐ Vistara
9	☐ Spice Jet
10	☐ IndiGo

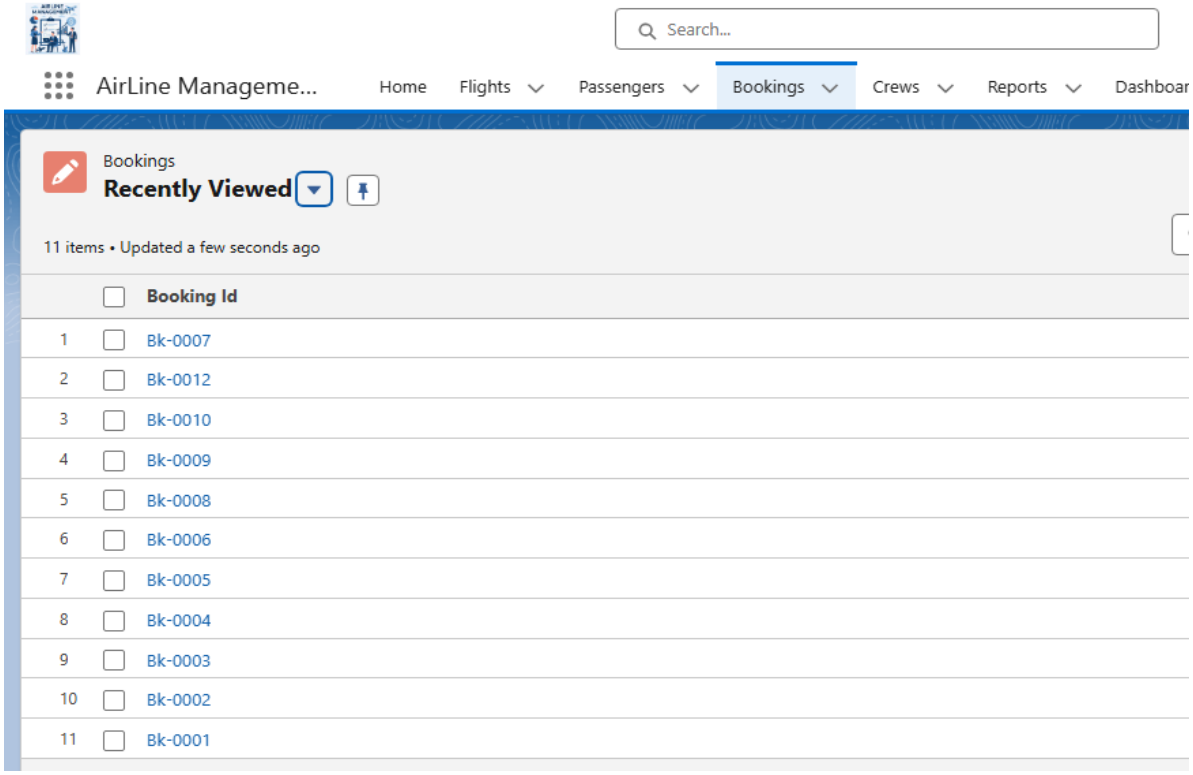


Passengers

Recently Viewed

11 items • Updated a few seconds ago

	Passenger Id
1	PS-0002
2	PS-0009
3	PS-0002
4	PS-0012
5	PS-0010
6	PS-0008
7	PS-0007
8	PS-0006
9	PS-0005
10	PS-0004
11	PS-0003

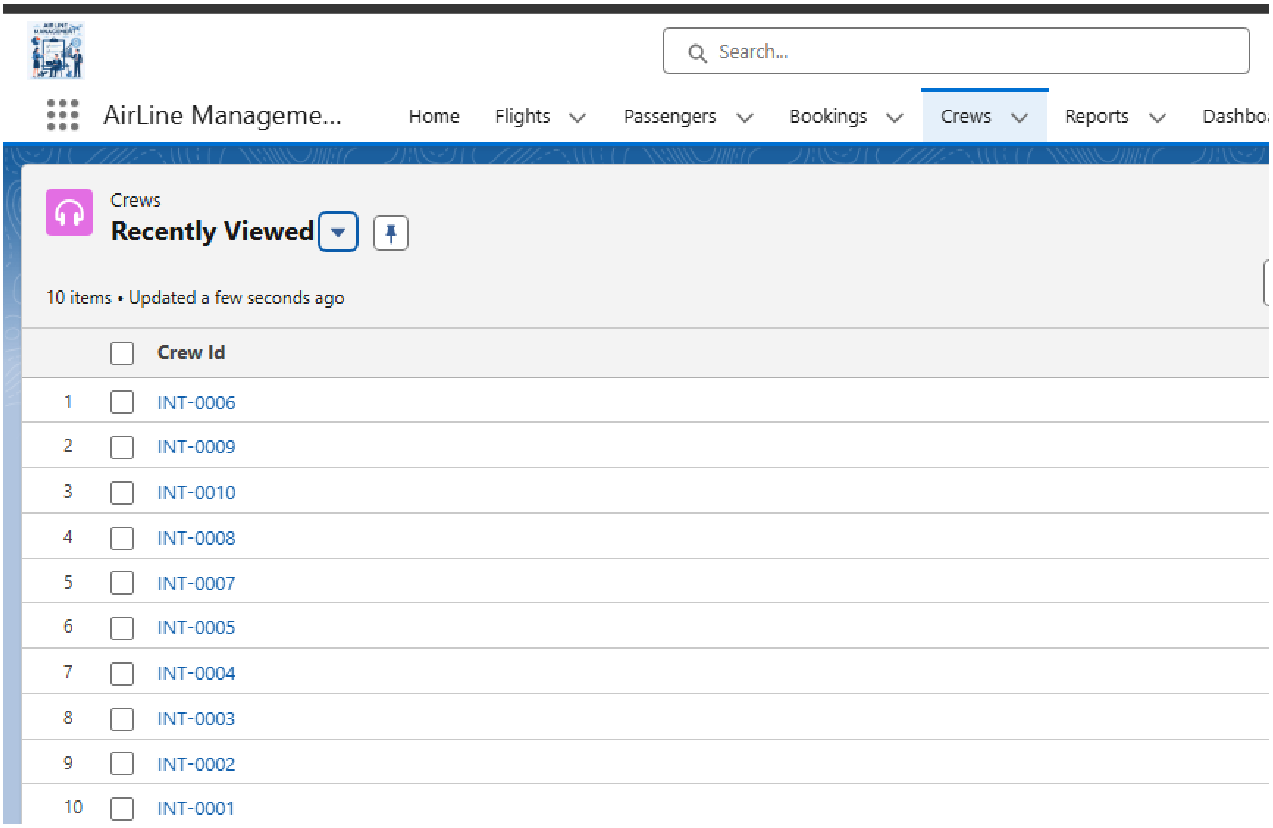


Bookings

Recently Viewed

11 items • Updated a few seconds ago

	Booking Id
1	Bk-0007
2	Bk-0012
3	Bk-0010
4	Bk-0009
5	Bk-0008
6	Bk-0006
7	Bk-0005
8	Bk-0004
9	Bk-0003
10	Bk-0002
11	Bk-0001



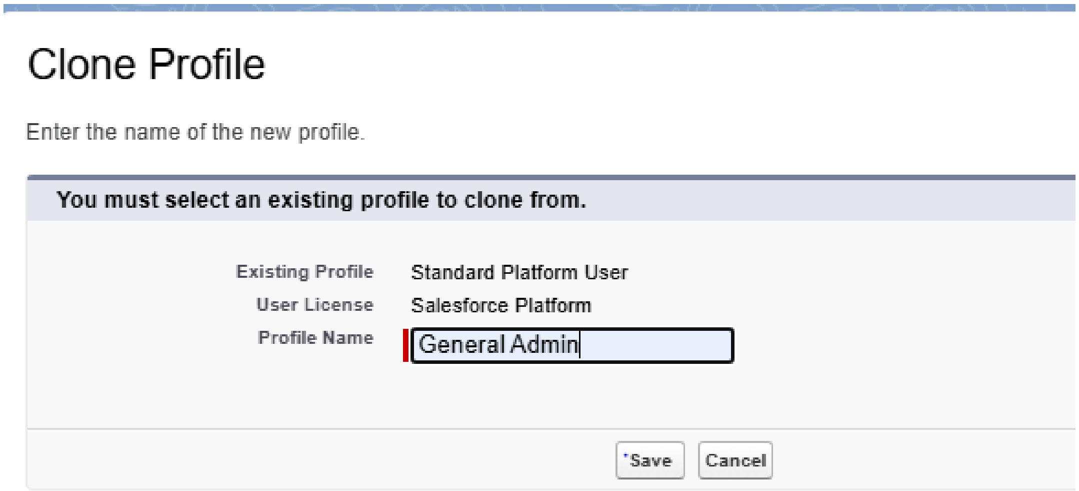
Crews

Recently Viewed

10 items • Updated a few seconds ago

	☐	Crew Id
1	☐	INT-0006
2	☐	INT-0009
3	☐	INT-0010
4	☐	INT-0008
5	☐	INT-0007
6	☐	INT-0005
7	☐	INT-0004
8	☐	INT-0003
9	☐	INT-0002
10	☐	INT-0001

Profile Management



Clone Profile

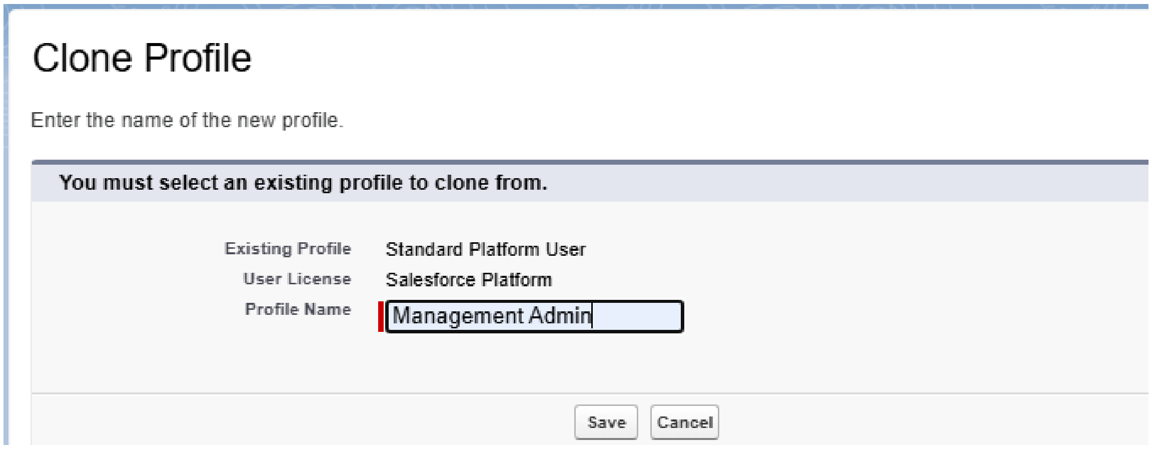
Enter the name of the new profile.

You must select an existing profile to clone from.

Existing Profile Standard Platform User

User License Salesforce Platform

Profile Name General Admin



Clone Profile

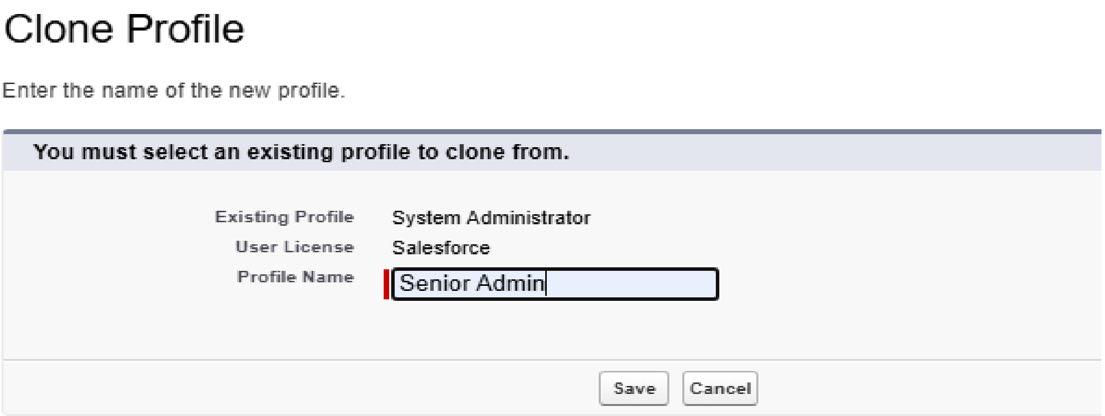
Enter the name of the new profile.

You must select an existing profile to clone from.

Existing Profile Standard Platform User

User License Salesforce Platform

Profile Name Management Admin



Clone Profile

Enter the name of the new profile.

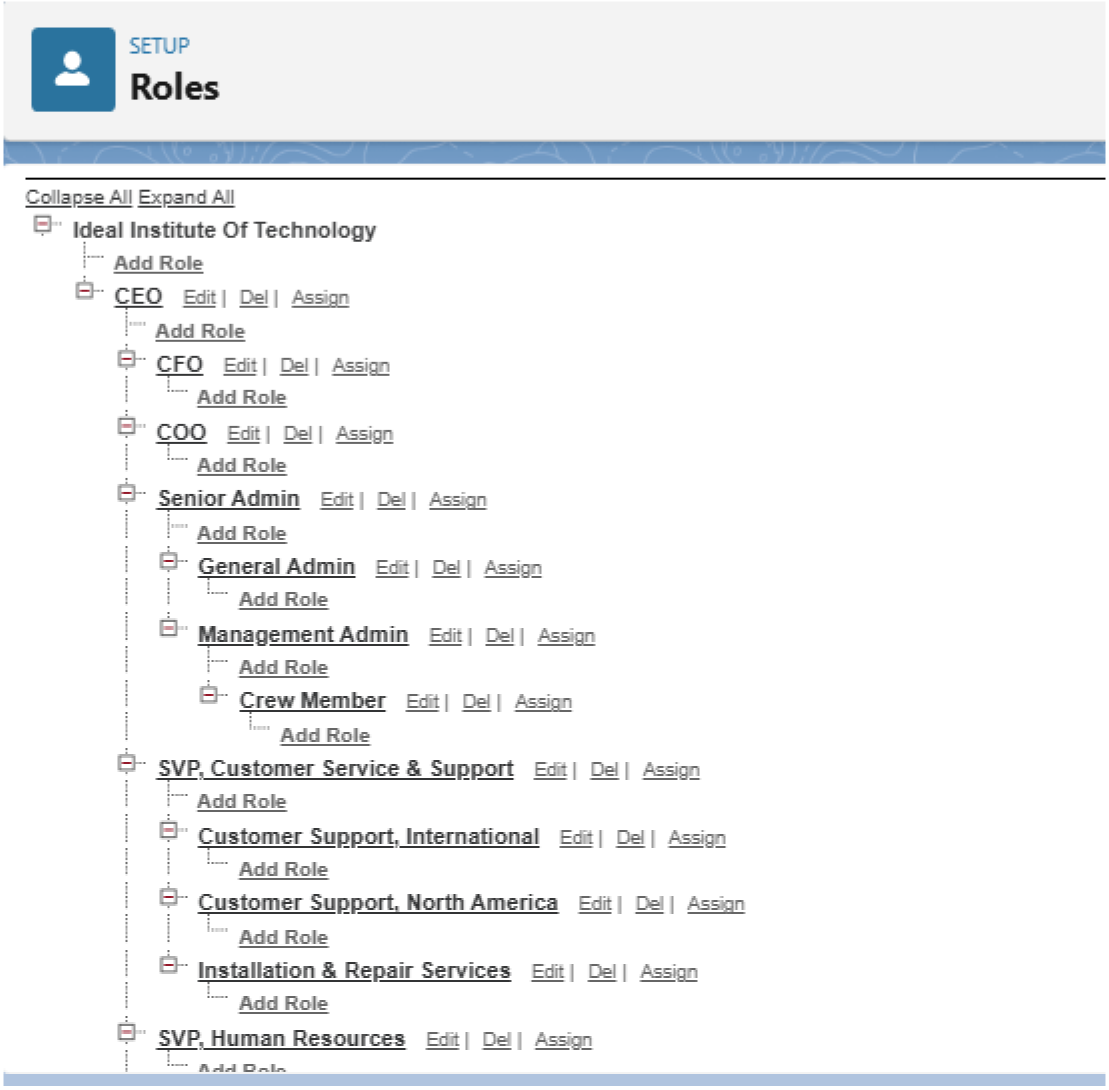
You must select an existing profile to clone from.

Existing Profile System Administrator

User License Salesforce

Profile Name Senior Admin

Role Hierarchy Setup

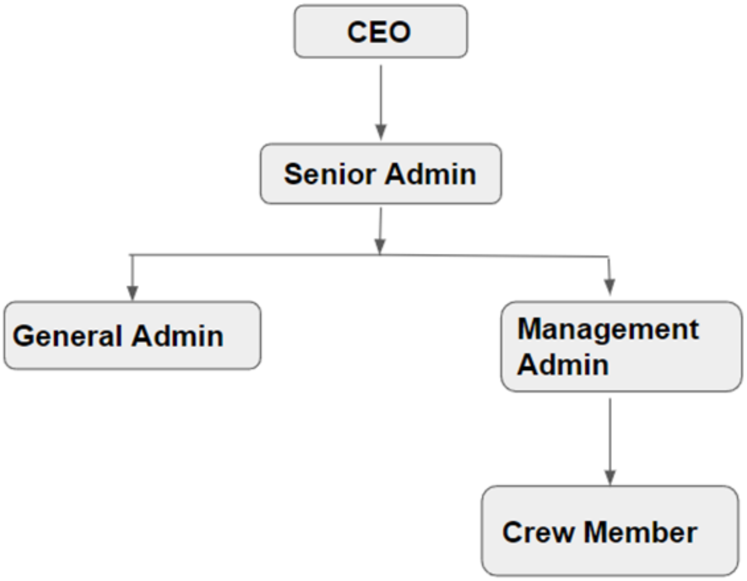


SETUP

Roles

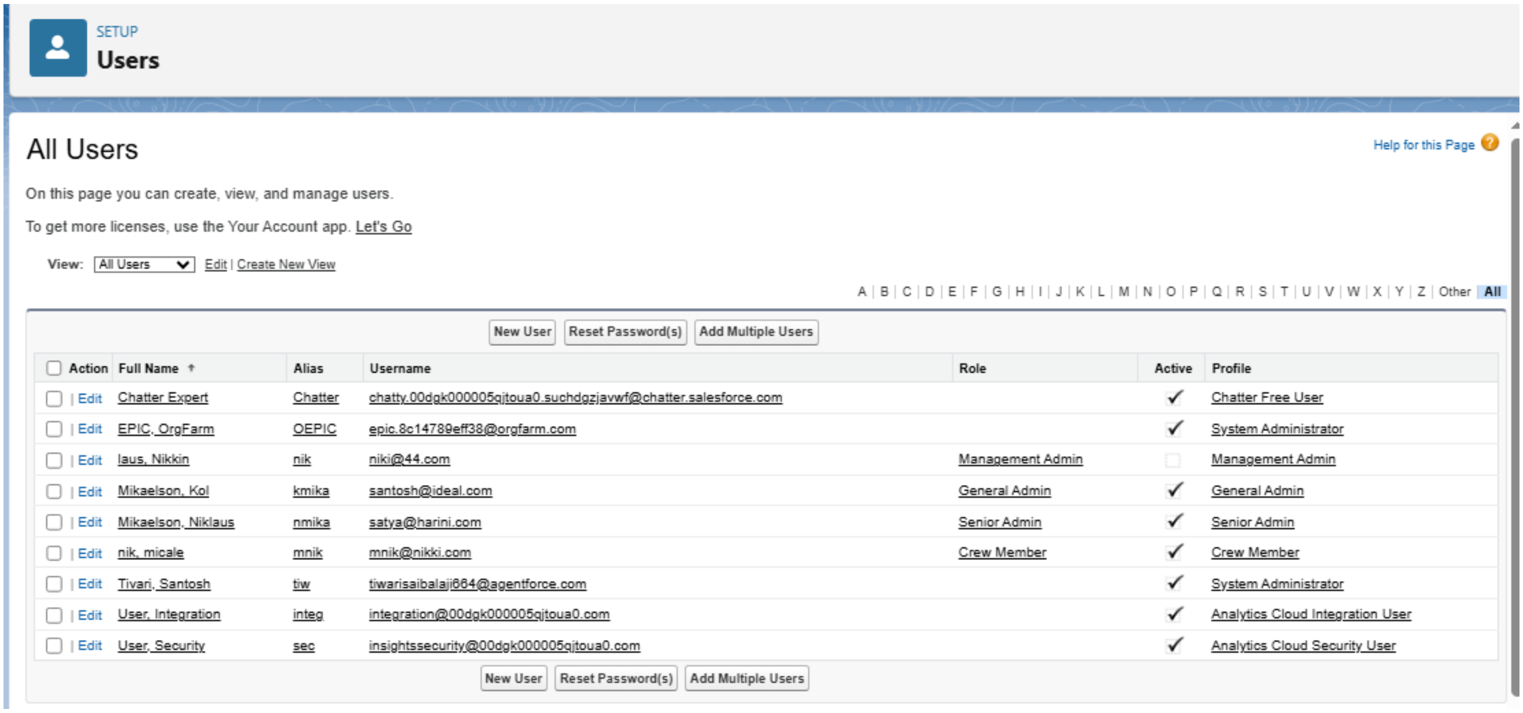
Collapse All Expand All

- Ideal Institute Of Technology
  - Add Role
  - CEO Edit | Del | Assign
    - Add Role
    - CFO Edit | Del | Assign
      - Add Role
    - COO Edit | Del | Assign
      - Add Role
    - Senior Admin Edit | Del | Assign
      - Add Role
      - General Admin Edit | Del | Assign
        - Add Role
      - Management Admin Edit | Del | Assign
        - Add Role
        - Crew Member Edit | Del | Assign
          - Add Role
    - SVP, Customer Service & Support Edit | Del | Assign
      - Add Role
      - Customer Support, International Edit | Del | Assign
        - Add Role
      - Customer Support, North America Edit | Del | Assign
        - Add Role
      - Installation & Repair Services Edit | Del | Assign
        - Add Role
    - SVP, Human Resources Edit | Del | Assign
      - Add Role



Role Hierarchy: The above diagram represents which role reports to which one.

User Management



All Users

Help for this Page

On this page you can create, view, and manage users.

To get more licenses, use the Your Account app. Let's Go

View: All Users | Edit | Create New View

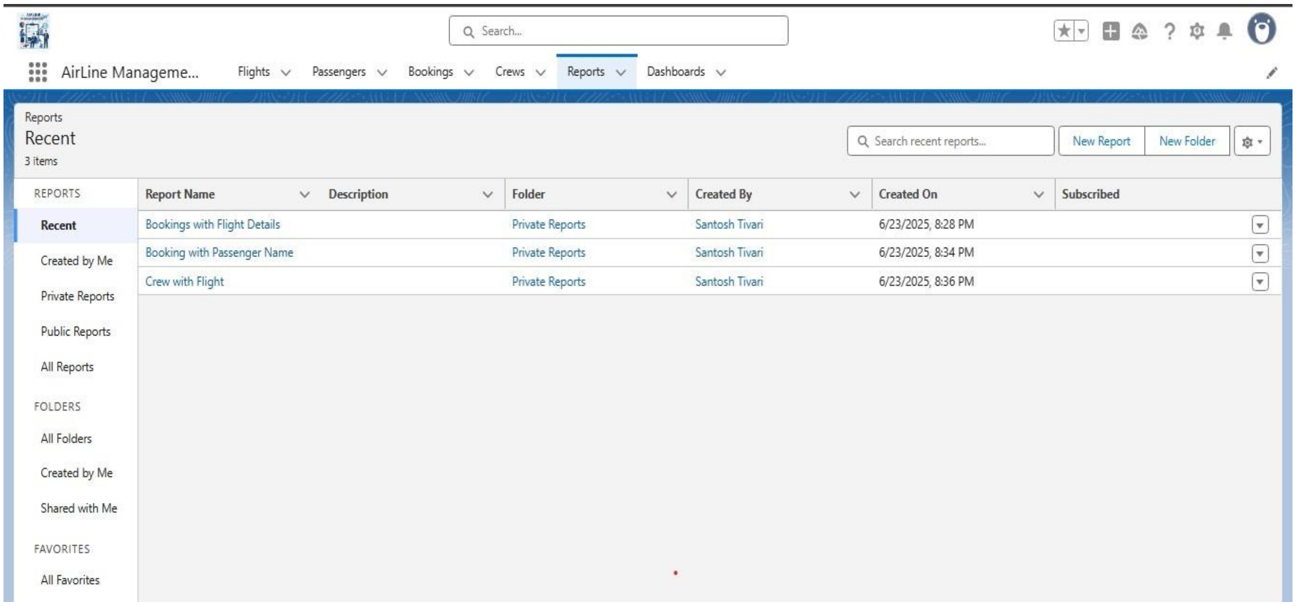
A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | Other | All

New User | Reset Password(s) | Add Multiple Users

□ Action	Full Name ↑	Alias	Username	Role	Active	Profile
□ | Edit	Chatter Expert	Chatter	chatty.00dgk000005qjtoua0.suchdgzjavwf@chatter.salesforce.com		✓	Chatter Free User
□ | Edit	EPIC, OrgFarm	OEPIC	epic.8c14789eff38@orgfarm.com		✓	System Administrator
□ | Edit	laus, Nikkin	nik	niki@44.com	Management Admin	□	Management Admin
□ | Edit	Mikaelson, Kol	kmika	santosh@ideal.com	General Admin	✓	General Admin
□ | Edit	Mikaelson, Niklaus	nmika	satya@harini.com	Senior Admin	✓	Senior Admin
□ | Edit	nik, micale	mnik	mnik@nikki.com	Crew Member	✓	Crew Member
□ | Edit	Tivari, Santosh	tiw	tiwarisaibalaji664@agentforce.com		✓	System Administrator
□ | Edit	User, Integration	integ	integration@00dgk000005qjtoua0.com		✓	Analytics Cloud Integration User
□ | Edit	User, Security	sec	insightssecurity@00dgk000005qjtoua0.com		✓	Analytics Cloud Security User

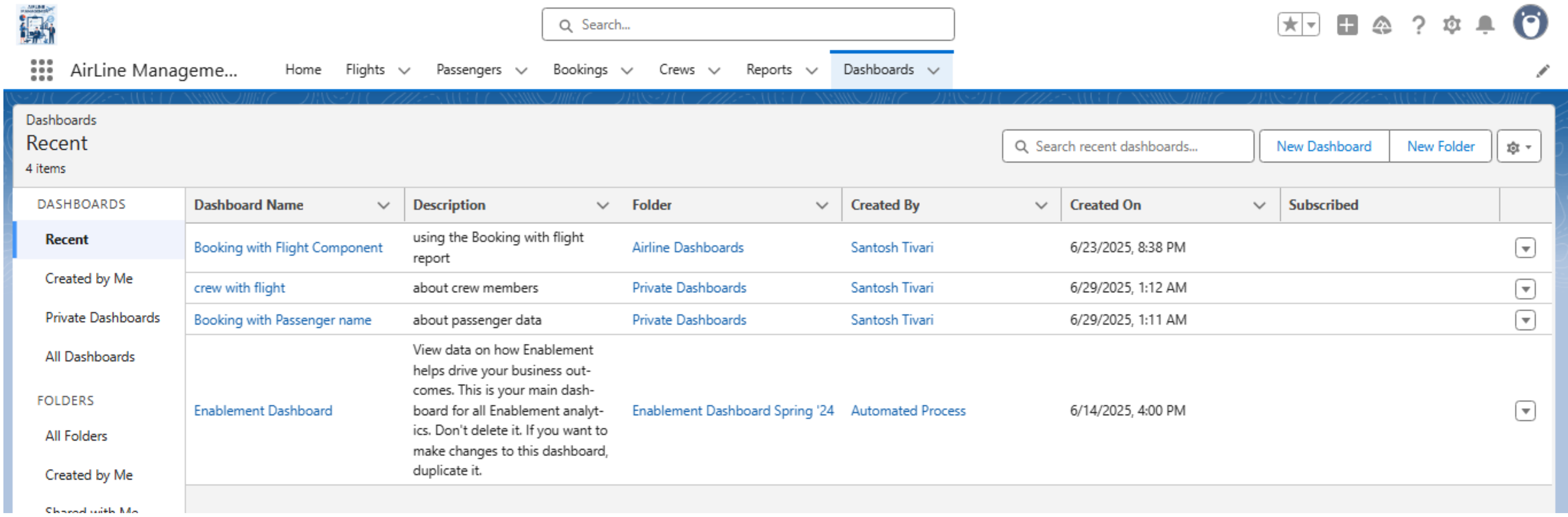
New User | Reset Password(s) | Add Multiple Users

Reports



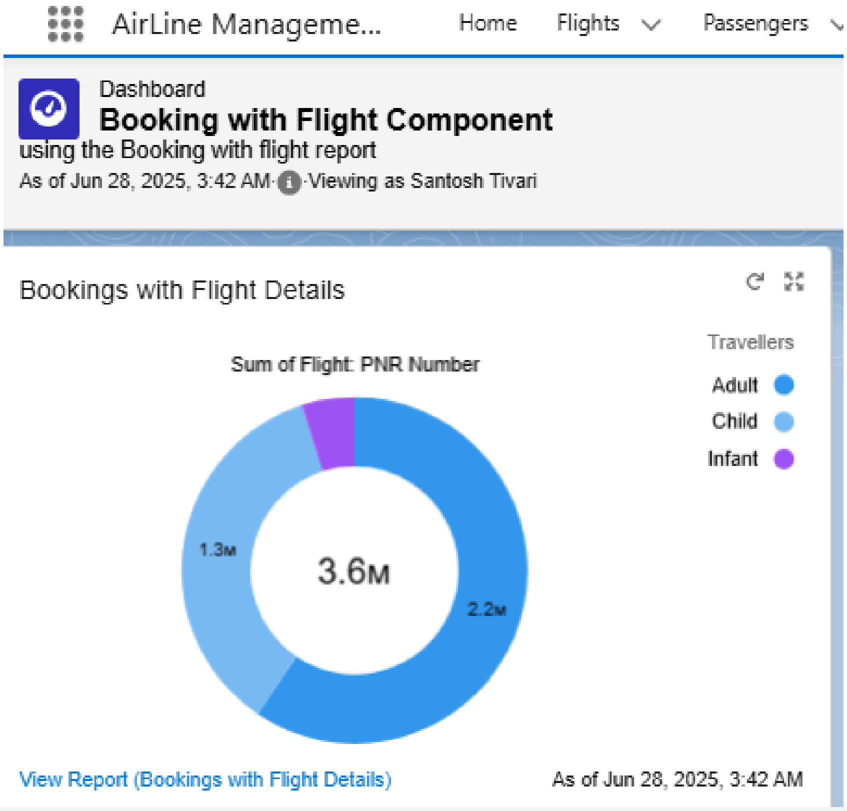
Report Name	Description	Folder	Created By	Created On	Subscribed
Bookings with Flight Details		Private Reports	Santosh Tivari	6/23/2025, 8:28 PM	
Booking with Passenger Name		Private Reports	Santosh Tivari	6/23/2025, 8:34 PM	
Crew with Flight		Private Reports	Santosh Tivari	6/23/2025, 8:36 PM	

Dashboards



Dashboard Name	Description	Folder	Created By	Created On	Subscribed
Booking with Flight Component	using the Booking with flight report	Airline Dashboards	Santosh Tivari	6/23/2025, 8:38 PM	
crew with flight	about crew members	Private Dashboards	Santosh Tivari	6/29/2025, 1:12 AM	
Booking with Passenger name	about passenger data	Private Dashboards	Santosh Tivari	6/29/2025, 1:11 AM	
Enablement Dashboard	View data on how Enablement helps drive your business outcomes. This is your main dashboard for all Enablement analytics. Don't delete it. If you want to make changes to this dashboard, duplicate it.	Enablement Dashboard Spring '24	Automated Process	6/14/2025, 4:00 PM	

Dashboards display as below:



Dashboard

Booking with Flight Component

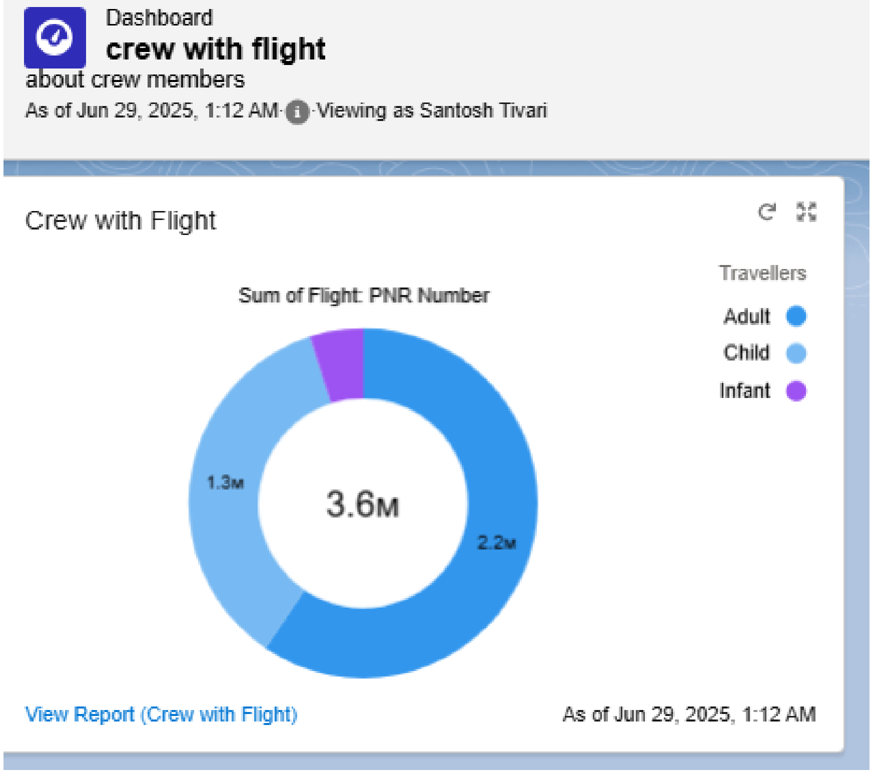
using the Booking with flight report

As of Jun 28, 2025, 3:42 AM · Viewing as Santosh Tivari

Bookings with Flight Details

View Report (Bookings with Flight Details)

As of Jun 28, 2025, 3:42 AM



Dashboard

crew with flight

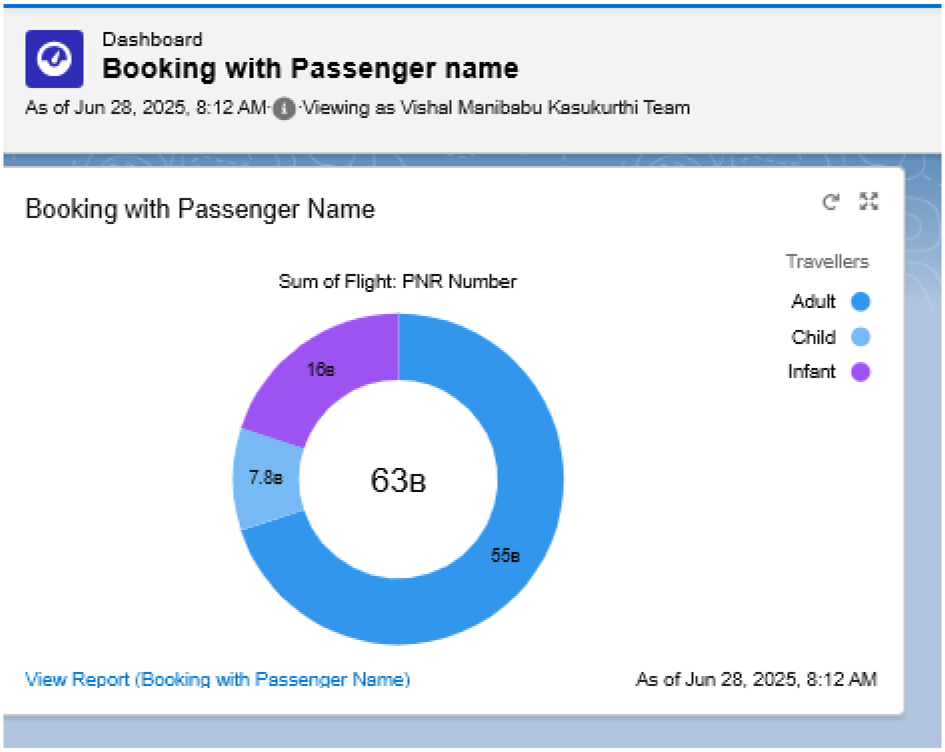
about crew members

As of Jun 29, 2025, 1:12 AM · Viewing as Santosh Tivari

Crew with Flight

View Report (Crew with Flight)

As of Jun 29, 2025, 1:12 AM



Dashboard

Booking with Passenger name

As of Jun 28, 2025, 8:12 AM·Viewing as Vishal Manibabu Kasukurthi Team

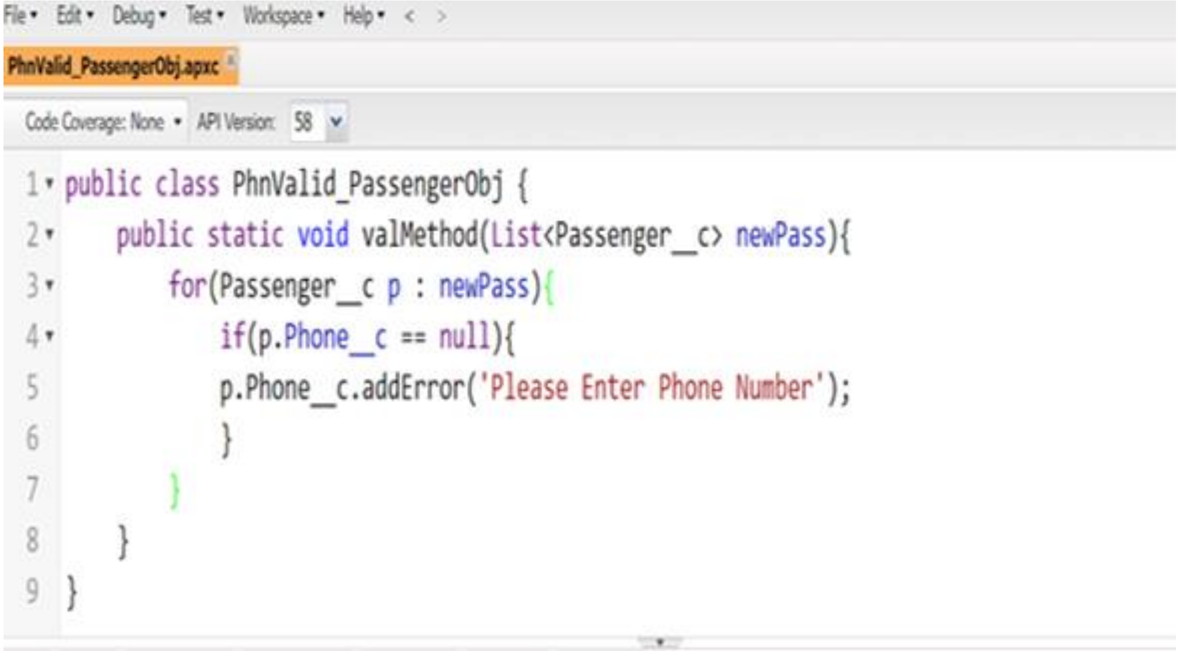
Booking with Passenger Name

View Report (Booking with Passenger Name)

As of Jun 28, 2025, 8:12 AM

- Apex Development

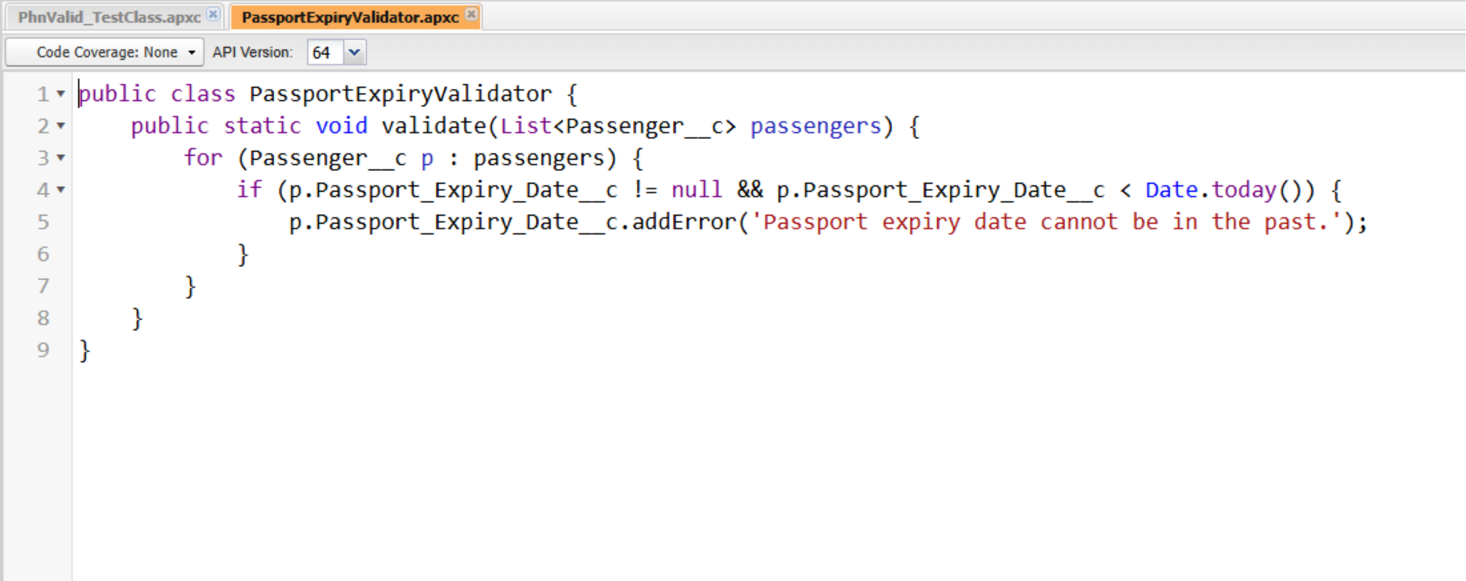
Wrote class ,trigger and test class to validate passenger phone input



File ▾ Edit ▾ Debug ▾ Test ▾ Workspace ▾ Help ▾ < >
PhnValid_PassengerObj.apxc

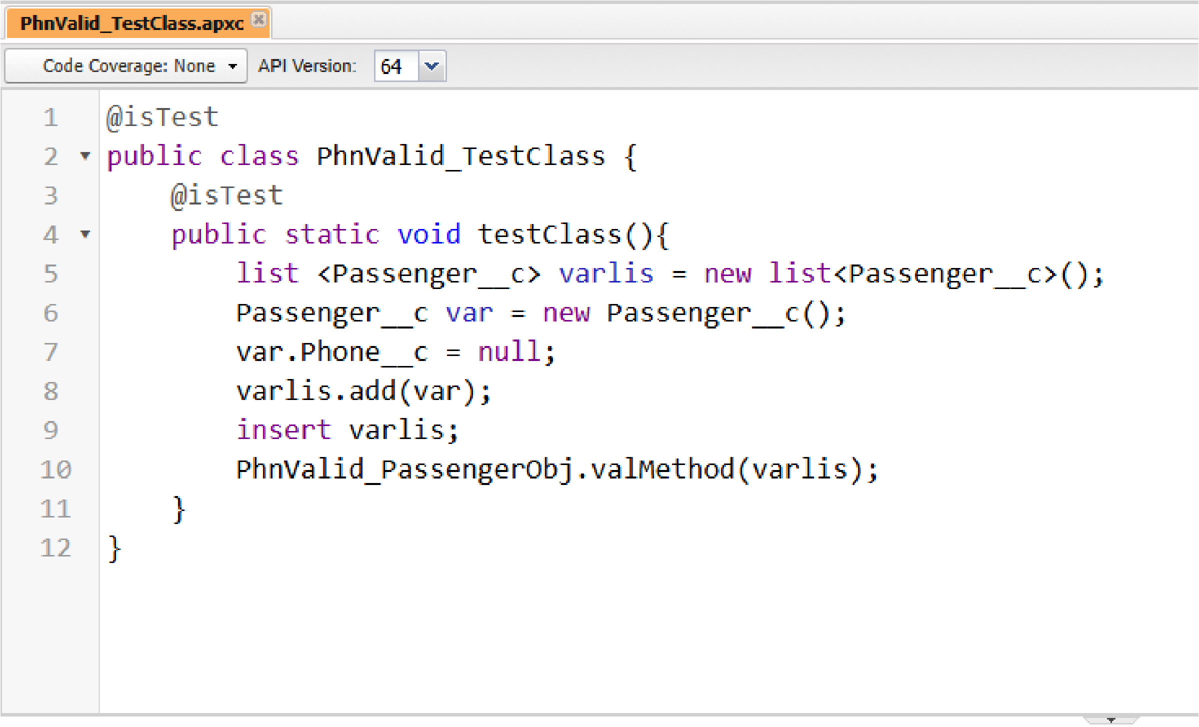
Code Coverage: None ▾ API Version: 58

```
public class PhnValid_PassengerObj {
    public static void valMethod(List<Passenger__c> newPass){
        for(Passenger__c p : newPass){
            if(p.Phone__c == null){
            p.Phone__c.addError('Please Enter Phone Number');
            }
        }
    }
}
```



Code Coverage: None API Version: 64

```
public class PassportExpiryValidator {
    public static void validate(List<Passenger__c> passengers) {
        for (Passenger__c p : passengers) {
            if (p.Passport_Expiry_Date__c != null && p.Passport_Expiry_Date__c < Date.today()) {
                p.Passport_Expiry_Date__c.addError('Passport expiry date cannot be in the past.');
            }
        }
    }
}
```

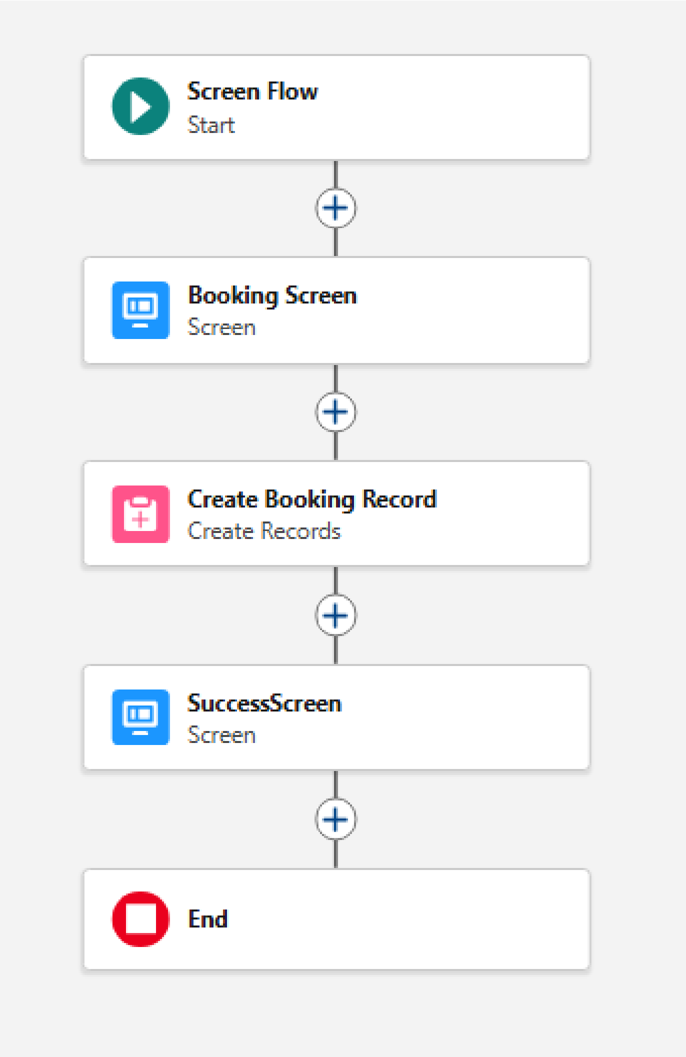


Code Coverage: None API Version: 64

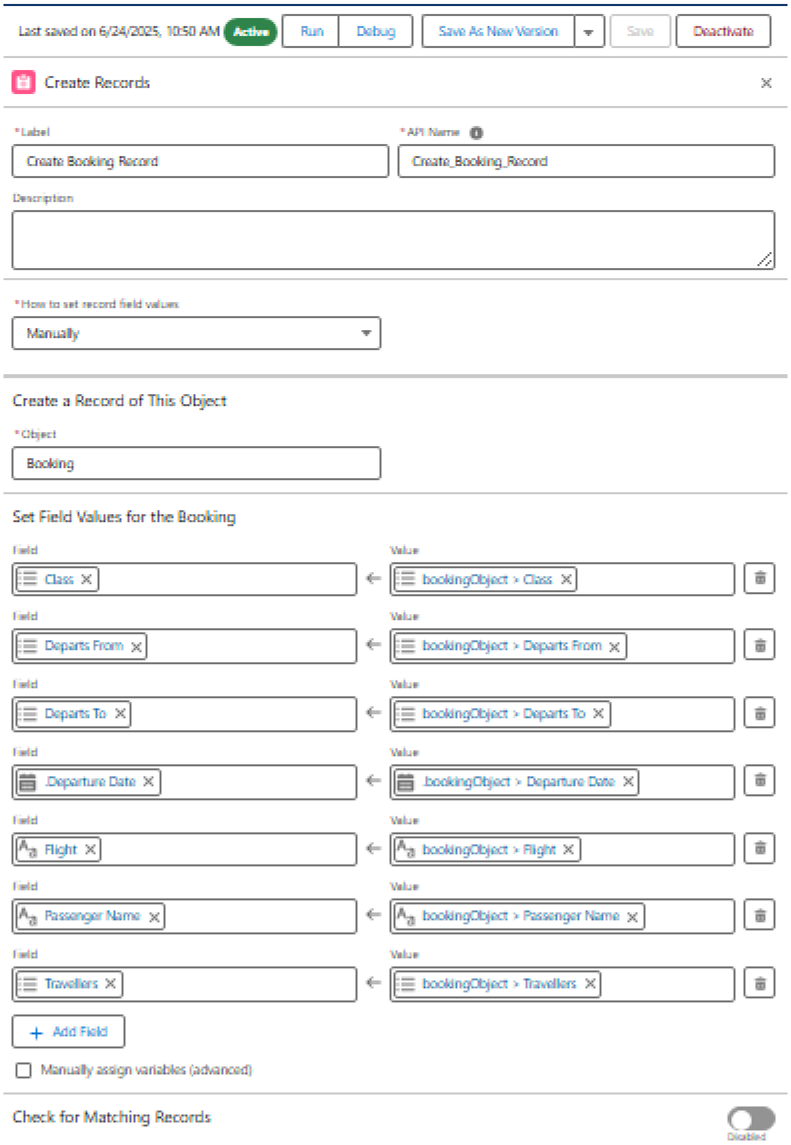
```
@isTest
public class PhnValid_TestClass {
    @isTest
    public static void testClass(){
        list <Passenger__c> varlis = new list<Passenger__c>();
        Passenger__c var = new Passenger__c();
        var.Phone__c = null;
        varlis.add(var);
        insert varlis;
        PhnValid_PassengerObj.valMethod(varlis);
    }
}
```

- Flow Automation

Built booking flow with success message confirmation screen



- **Built booking flow with success message confirmation screen**



Create Records

*Label: Create Booking Record

*API Name: Create_Booking_Record

Description:

*How to set record field values: Manually

Create a Record of This Object

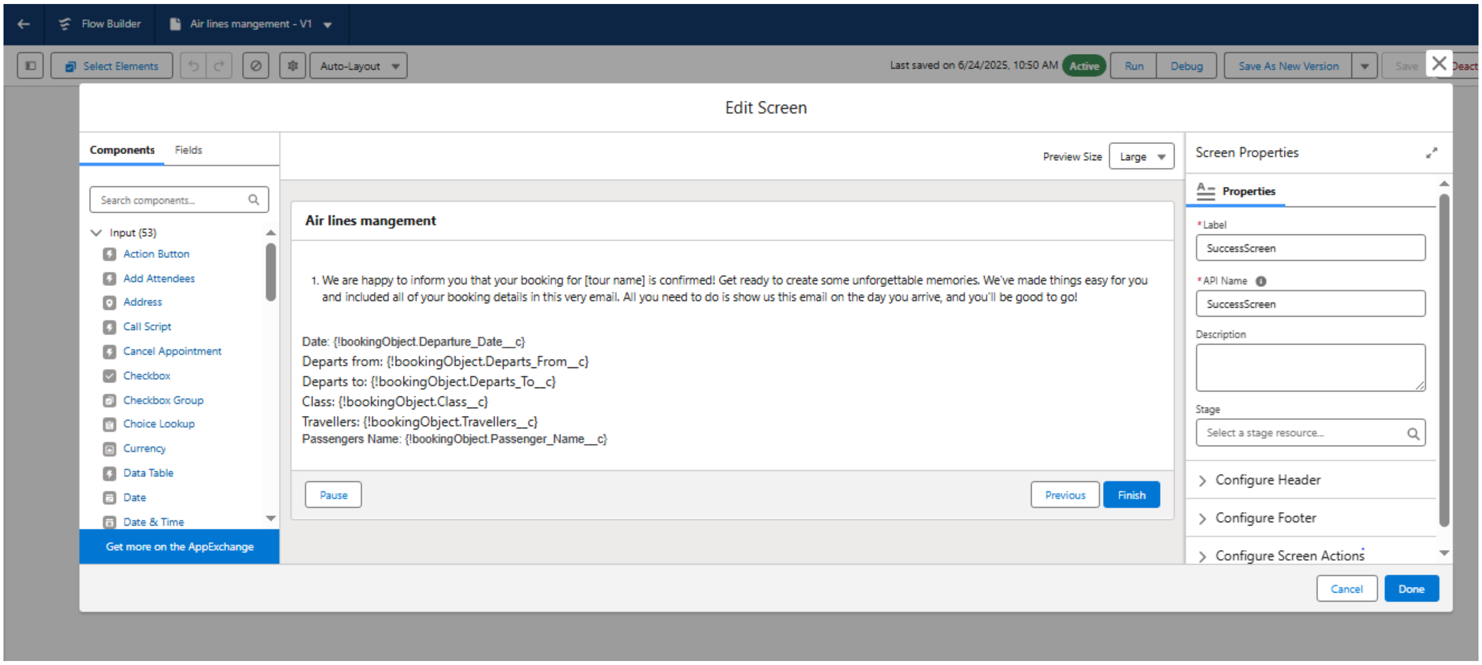
*Object: Booking

Set Field Values for the Booking

Field	Value
Class	bookingObject > Class
Departs From	bookingObject > Departs From
Departs To	bookingObject > Departs To
Departure Date	bookingObject > Departure Date
Flight	bookingObject > Flight
Passenger Name	bookingObject > Passenger Name
Travellers	bookingObject > Travellers

□ Manually assign variables (advanced)

Check for Matching Records: Disabled



Edit Screen

Air lines mangement

1. We are happy to inform you that your booking for [tour name] is confirmed! Get ready to create some unforgettable memories. We've made things easy for you and included all of your booking details in this very email. All you need to do is show us this email on the day you arrive, and you'll be good to go!

Date: {!bookingObject.Departure_Date__c}
Departs from: {!bookingObject.Departs_From__c}
Departs to: {!bookingObject.Departs_To__c}
Class: {!bookingObject.Class__c}
Travellers: {!bookingObject.Travellers__c}
Passengers Name: {!bookingObject.Passenger_Name__c}

Screen Properties

Properties

*Label: SuccessScreen

*API Name: SuccessScreen

Description:

Stage: Select a stage resource...

✓ Project Executable Files

In the development of the Airlines Management System, Salesforce's low-code platform was enhanced using Apex programming to implement custom validations and business logic. These executable files represent the backend logic that enforces data integrity and automates specific actions within the system.

The following components were developed as part of the executable logic:

Apex Class: Contains reusable methods that enforce custom validation rules on the Passenger object.

Apex Trigger: Automatically calls the validation method before a Passenger record is inserted, ensuring mandatory fields are checked.

Apex Test Class: Validates the logic through unit testing, ensuring that the trigger behaves correctly and achieves over 75% code coverage—a Salesforce deployment requirement.

Below are the Apex Class, Trigger, and Test Class used in the Airline Management System project:

1. Apex Class: Contains reusable methods that enforce custom validation rules on the Passenger object.

Apex Class: PhnValid_PassengerObj.cls

```
public class PhnValid_PassengerObj
        public static void valMethod(List<Passenger__c>
newPass)(
        for(Passenger___c p : newPass)(
        if(p.Phone__c == null)(
            p.Phone__c.addError('please Enter phone
Number');
```

2. Apex Trigger: Automatically calls the validation method before a Passenger record is inserted, ensuring mandatory fields are checked.

Apex Trigger: PhnValidTrigger.trigger

```
 trigger PhnValidTrigger on Passenger_pc
(before insert) {
        if(trigger.isBefore && trigger.isInsert)(

PhnValid_PassengerObj.valMethod(trigger.new);
```

3. Apex Test Class: Validates the logic through unit testing, ensuring that the trigger behaves correctly and achieves over 75% code coverage—a Salesforce deployment requirement.

Apex Test Class: PhnValid_TestClass.cls

```
 @isTest
 public class PhnValid_TestClass (
          @isTest
          public static void testClassb(
          List<Passenger__c> varlis = new
List<Passenger  _c>b;
          Passenger   _c var = new Passenger   _cb;
          var.Phone__c = null;
          varlis.add(var);
          insert varlis;
          PhnValid_PassengerObj.valMethod(varlis);
```

✓ Dataset

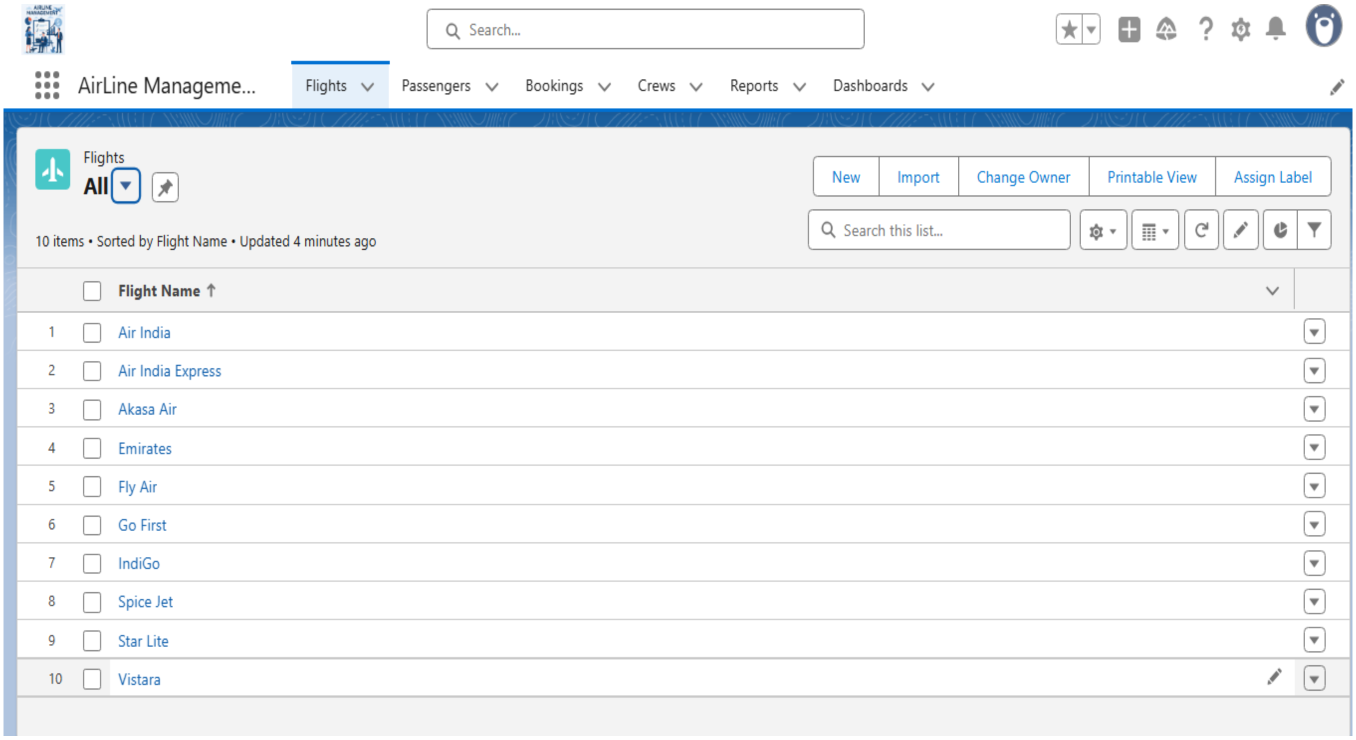
Sample records used for the Airline Management System are shown below. Each object includes a minimum of 10 records

Dataset Creation

As part of the system validation and testing process, datasets were manually created for each key object in the Airlines Management System. A minimum of 10 sample records were added to the following custom objects:

- ✈ Flight
- 🧍 Passenger
- 🧾 Booking
- 👨‍✈️ Crew

✓ FLIGHT

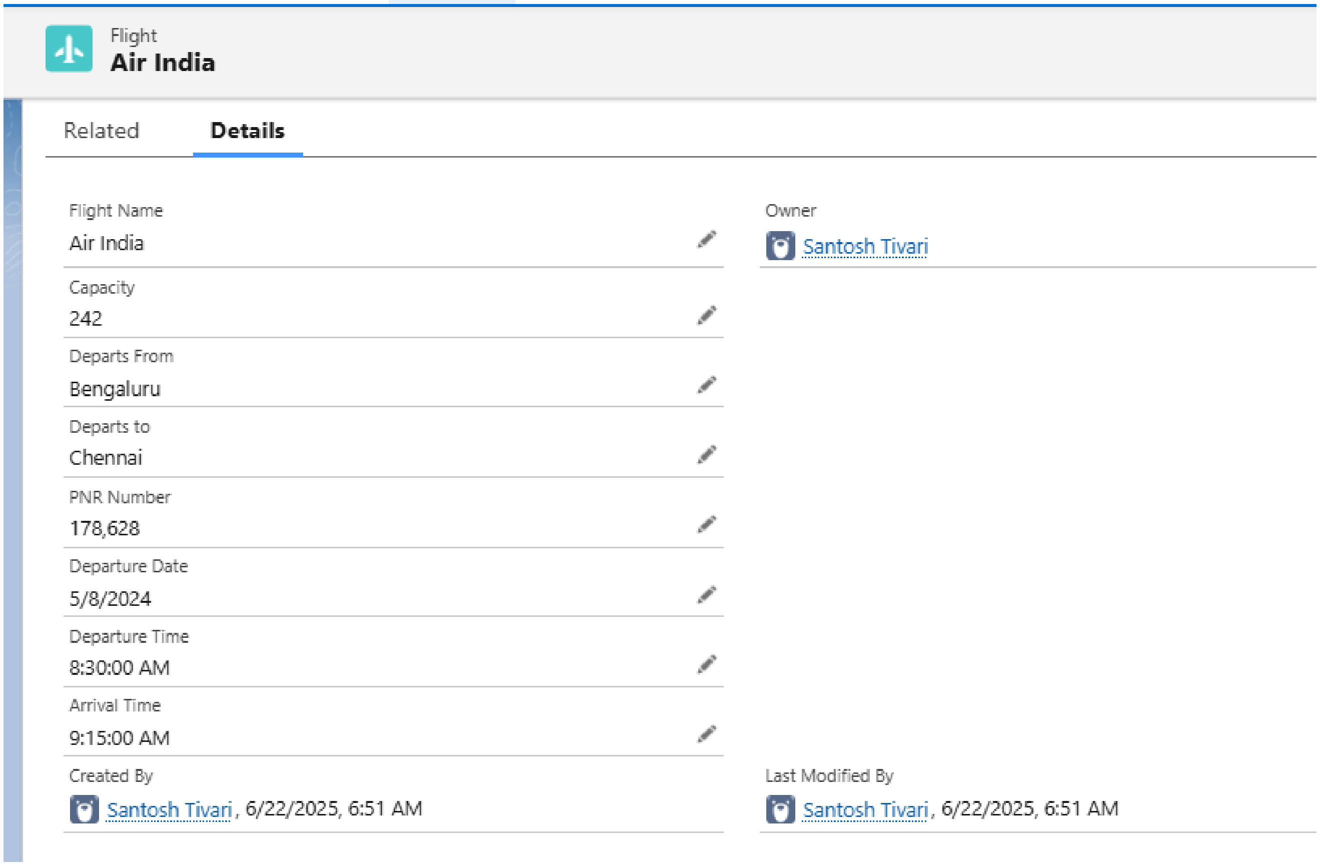


10 items • Sorted by Flight Name • Updated 4 minutes ago

	Flight Name ↑
1	Air India
2	Air India Express
3	Akasa Air
4	Emirates
5	Fly Air
6	Go First
7	IndiGo
8	Spice Jet
9	Star Lite
10	Vistara

To simulate real-world airline operations, 10 sample records were created for the Flight object within the Salesforce Lightning App. These records include essential details such as:

- Flight Name
- Departs From and Departs To locations
- Capacity
- PNR Number
- Departure & Arrival Times



Flight
Air India

Flight Name
Air India

Owner
Santosh Tivari

Capacity
242

Departs From
Bengaluru

Departs to
Chennai

PNR Number
178,628

Departure Date
5/8/2024

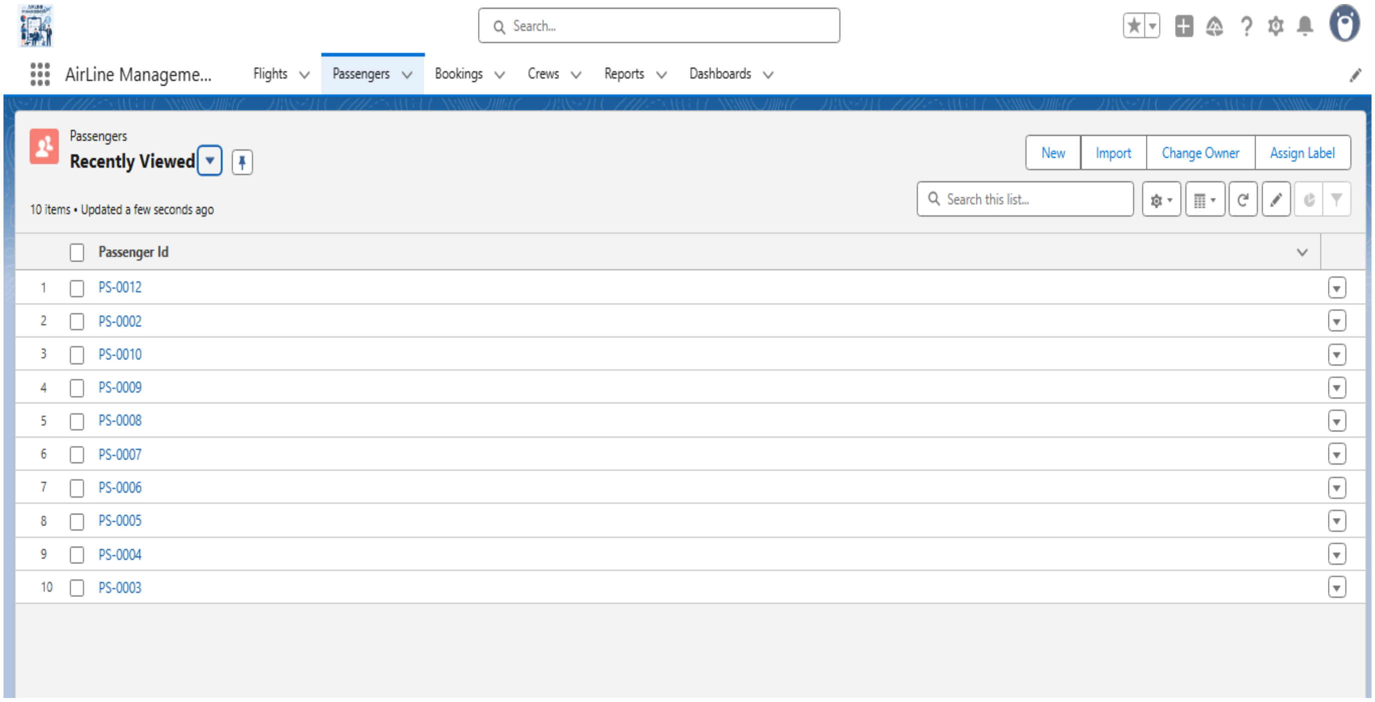
Departure Time
8:30:00 AM

Arrival Time
9:15:00 AM

Created By
Santosh Tivari, 6/22/2025, 6:51 AM

Last Modified By
Santosh Tivari, 6/22/2025, 6:51 AM

✓PASSENGER



Passengers

Recently Viewed

10 items • Updated a few seconds ago

	Passenger Id
1	PS-0012
2	PS-0002
3	PS-0010
4	PS-0009
5	PS-0008
6	PS-0007
7	PS-0006
8	PS-0005
9	PS-0004
10	PS-0003

Passenger Records Creation

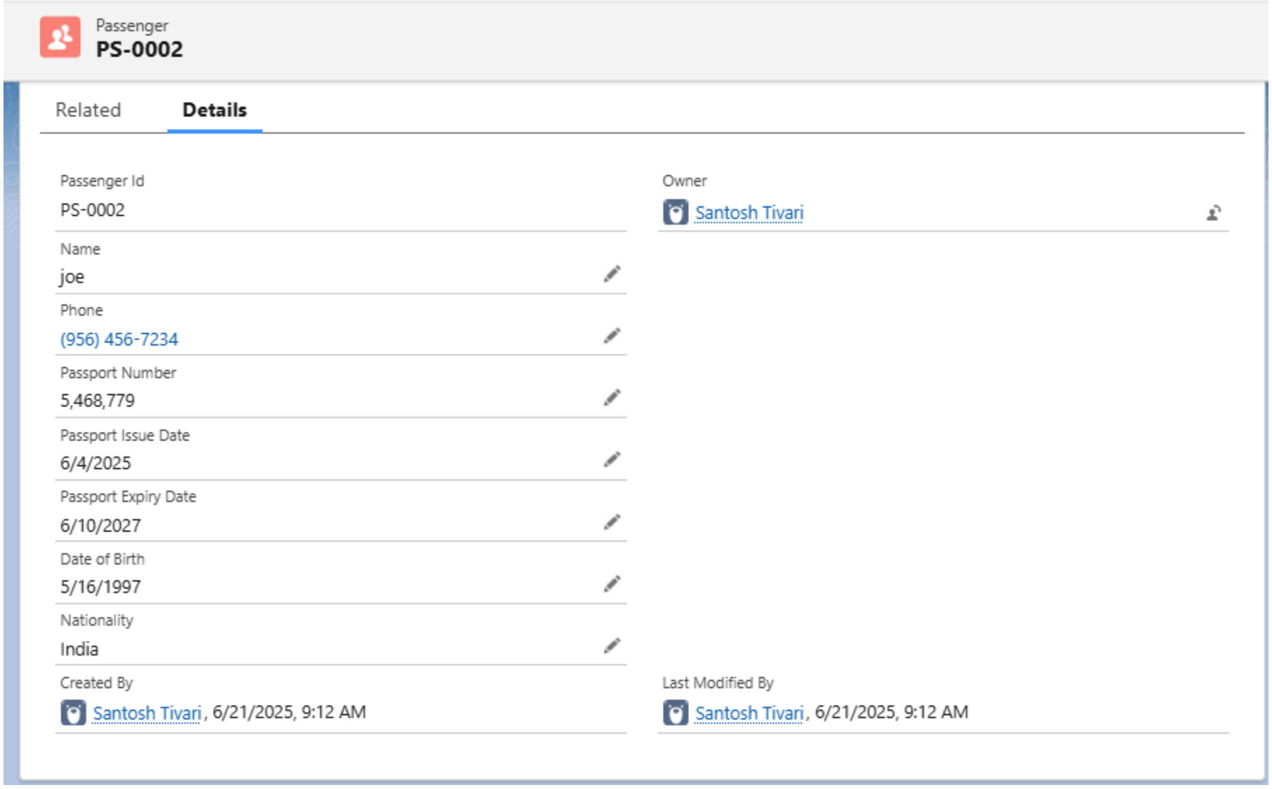
A total of 10 Passenger records were created within the Airlines Manageme represent various travelers booking flights through the platform. Each recor key personal and travel-related details such as:

- Passenger Name
- Phone Number
- Passport Details (Number, Issue Date, Expiry Date)
- Date of Birth

- Nationality (from a predefined picklist) Passenger Name (Lookup to Passenger)
- Flight (Lookup to Flight)
- Departure Date

  Departs From / Departs To
- Class (Economy, Business, First Class)
- Travellers (Adult, Child, Infant)

These entries were created using the Passenger tab in the Salesforce Lightning App.



Passenger
PS-0002

Passenger Id: PS-0002

Owner: Santosh Tivari

Name: joe

Phone: (956) 456-7234

Passport Number: 5,468,779

Passport Issue Date: 6/4/2025

Passport Expiry Date: 6/10/2027

Date of Birth: 5/16/1997

Nationality: India

Created By: Santosh Tivari, 6/21/2025, 9:12 AM

Last Modified By: Santosh Tivari, 6/21/2025, 9:12 AM

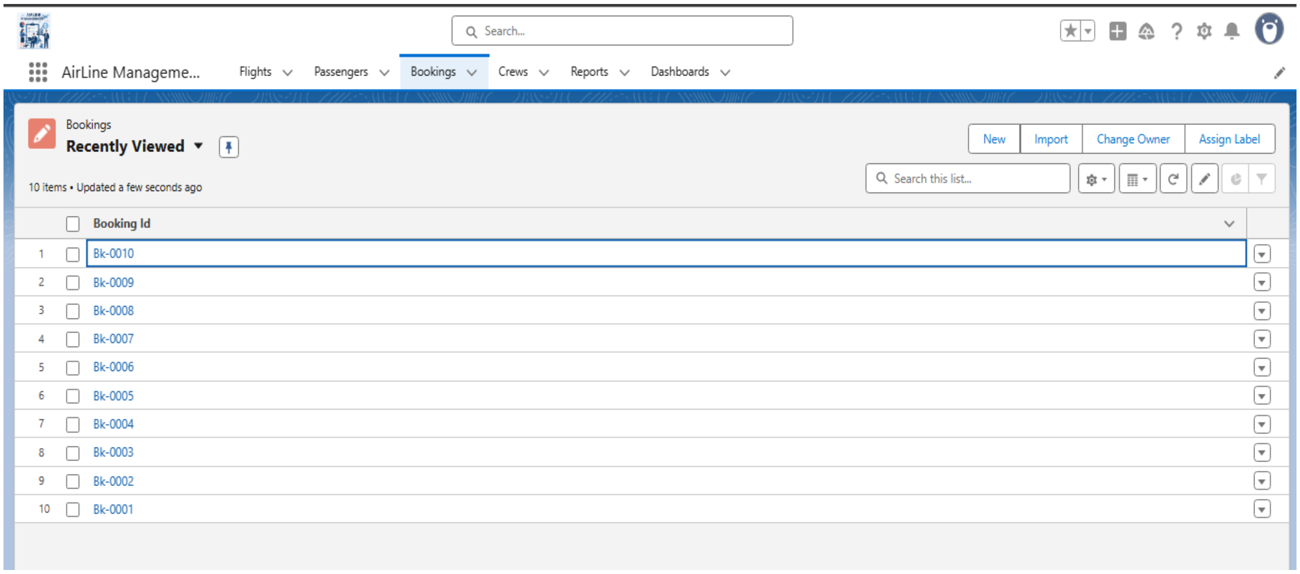
✓BOOKING

✈️ Booking Records Creation

A total of **10 Booking records** were created within the **Airlines Management System**, representing various flight reservations made by passengers through the platform. Each record captures essential booking-related details, including:

- **Booking Reference Number**
- **Booking Date**
- **Passenger (Lookup to Passenger Object)**
- **Flight Number (Lookup to Flight Object)**
- **Class of Travel** (Economy / Business / First – via picklist)
- **Number of Seats Booked**
- **Total Fare Amount**
- **Payment Status** (Paid / Pending / Cancelled – via picklist)

These records were created using the **Booking tab** within the **Salesforce Lightning App**, ensuring all bookings are tracked accurately and associated with the corresponding passenger and flight details.

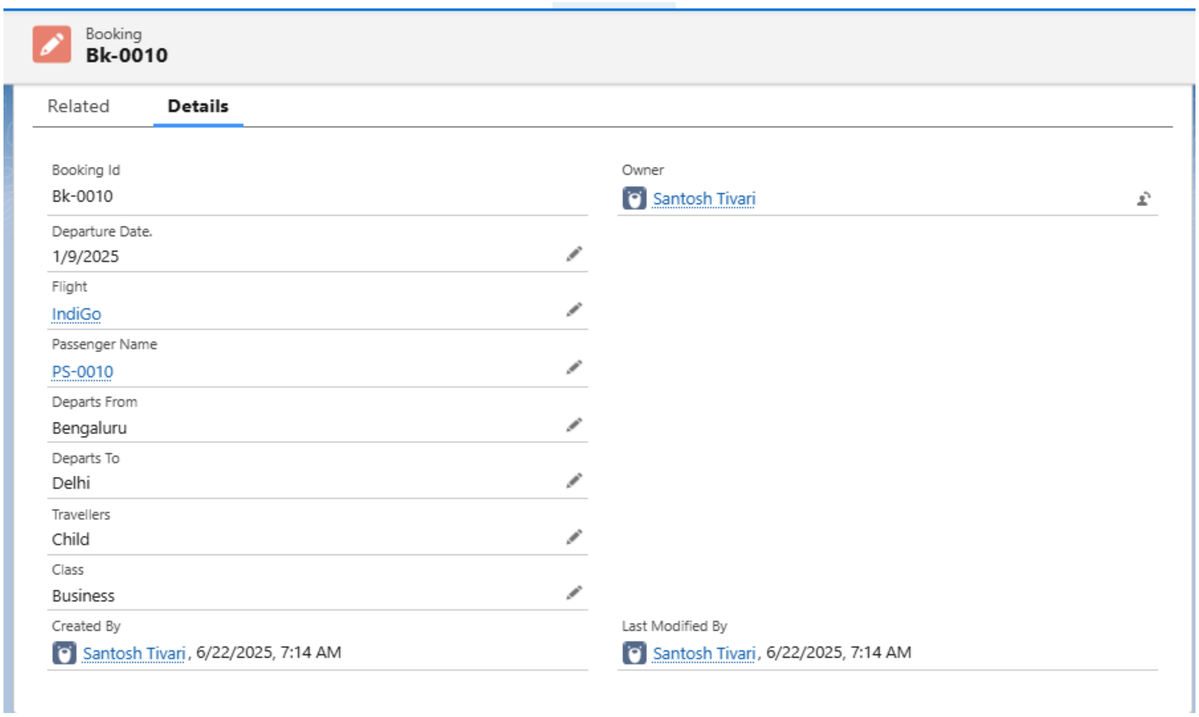


Bookings

Recently Viewed

10 items • Updated a few seconds ago

		Booking Id
1	□	Bk-0010
2	□	Bk-0009
3	□	Bk-0008
4	□	Bk-0007
5	□	Bk-0006
6	□	Bk-0005
7	□	Bk-0004
8	□	Bk-0003
9	□	Bk-0002
10	□	Bk-0001



Booking

Bk-0010

Booking Id
Bk-0010

Owner
Santosh Tivari

Departure Date.
1/9/2025

Flight
IndiGo

Passenger Name
PS-0010

Departs From
Bengaluru

Departs To
Delhi

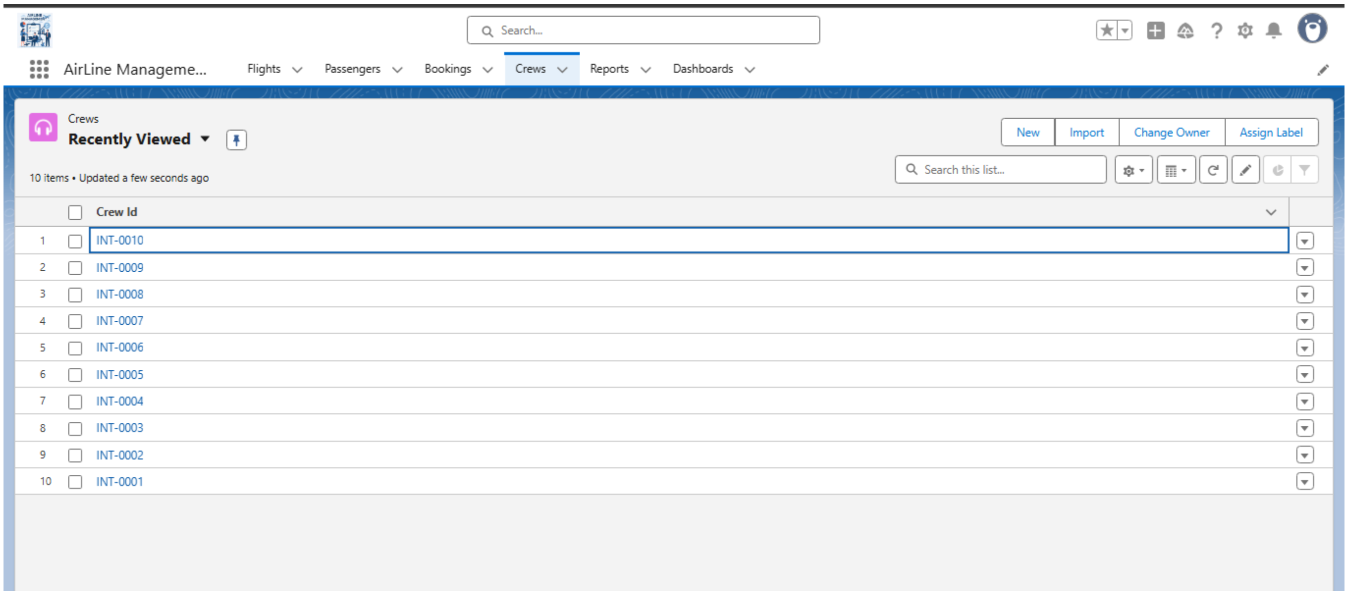
Travellers
Child

Class
Business

Created By
Santosh Tivari, 6/22/2025, 7:14 AM

Last Modified By
Santosh Tivari, 6/22/2025, 7:14 AM

✓CREW



Crews

Recently Viewed

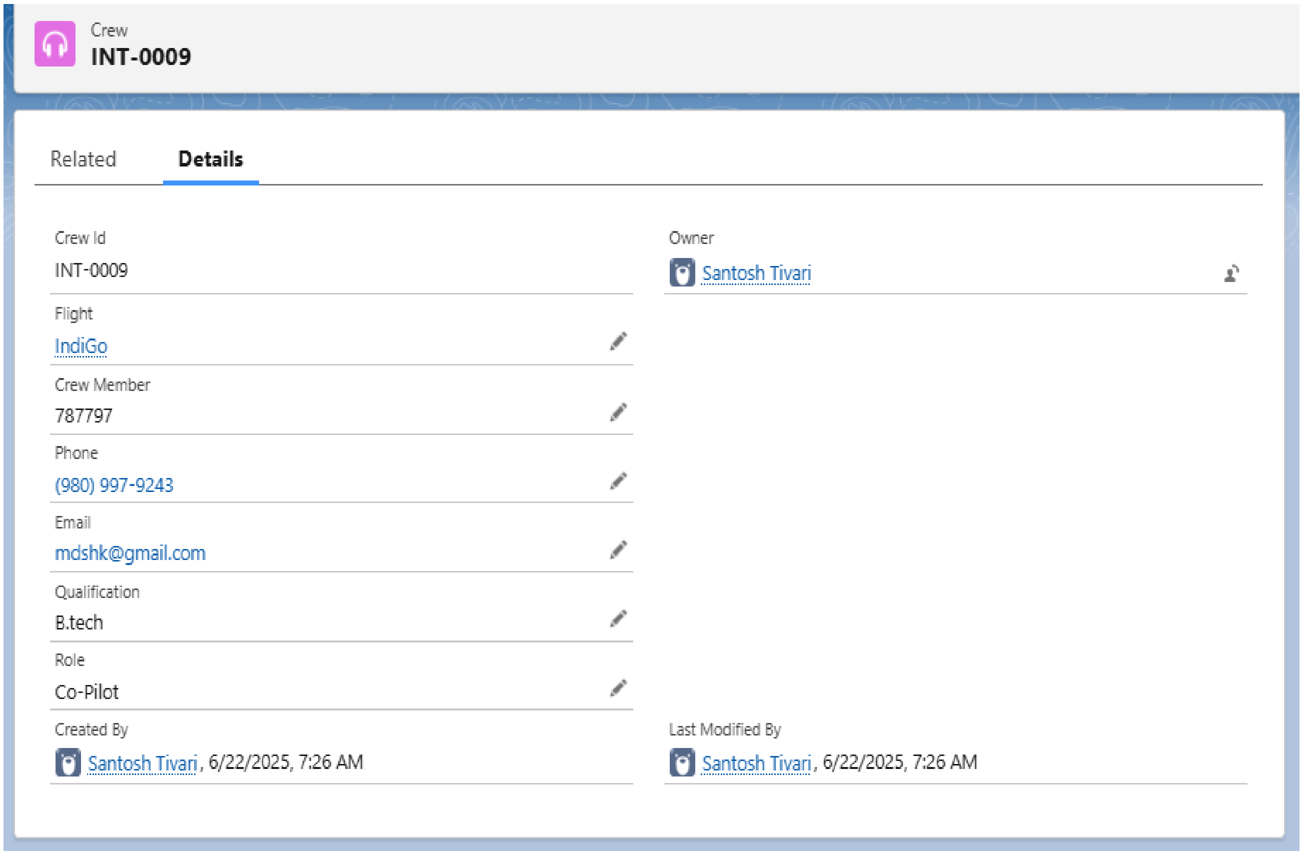
10 items • Updated a few seconds ago

	Crew Id
1	INT-0010
2	INT-0009
3	INT-0008
4	INT-0007
5	INT-0006
6	INT-0005
7	INT-0004
8	INT-0003
9	INT-0002
10	INT-0001

👨‍✈️ Crew Records Creation

To complete the airline operations
the Crew object in the Salesforc
real-life crew assignments for di
fields:

- Crew Member Name
- Phone Number
- Email Address
- Qualification (e.g., Comm
- Role (e.g., Pilot, Co-Pilot,
- Flight (Lookup to Flight o



INT-0009

Crew Id INT-0009	Owner Santosh Tivari
Flight IndiGo	
Crew Member 787797	
Phone (980) 997-9243	
Email mdshk@gmail.com	
Qualification B.tech	
Role Co-Pilot	
Created By Santosh Tivari, 6/22/2025, 7:26 AM	Last Modified By Santosh Tivari, 6/22/2025, 7:26 AM

Overview

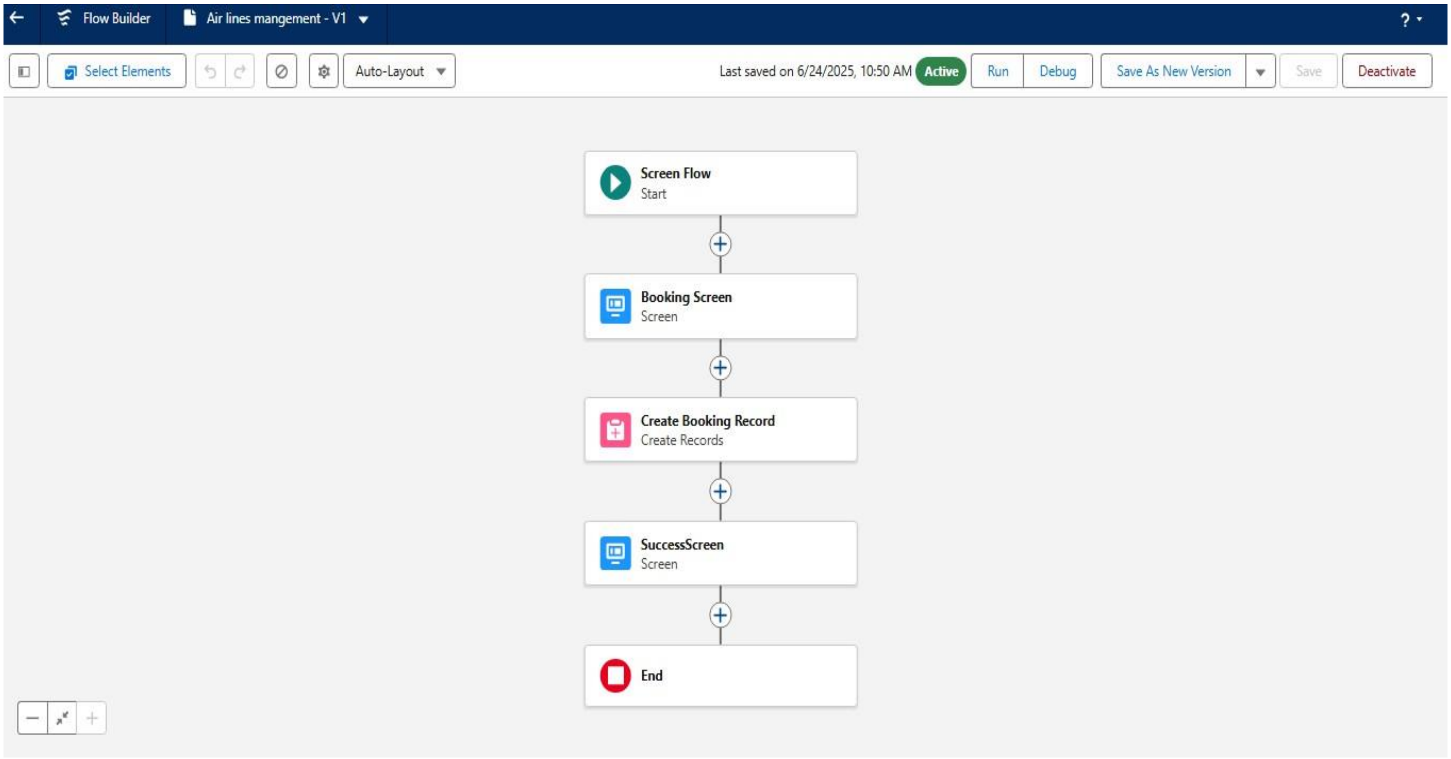
It includes Apex classes, triggers, and test classes used to implement core functionalities such as data validation. A sample dataset for major objects like Passenger, Flight, Booking, and Crew is provided to simulate real-world airline operations. The document also includes output screenshots from the Salesforce Org that demonstrate successful execution, error handling, dashboard analytics, and user interface features such as the Lightning App and report views. This project emphasizes the integration of business logic and visual reporting within the Salesforce platform for a streamlined airline management solution.

Functional & Performance Testing Outputs

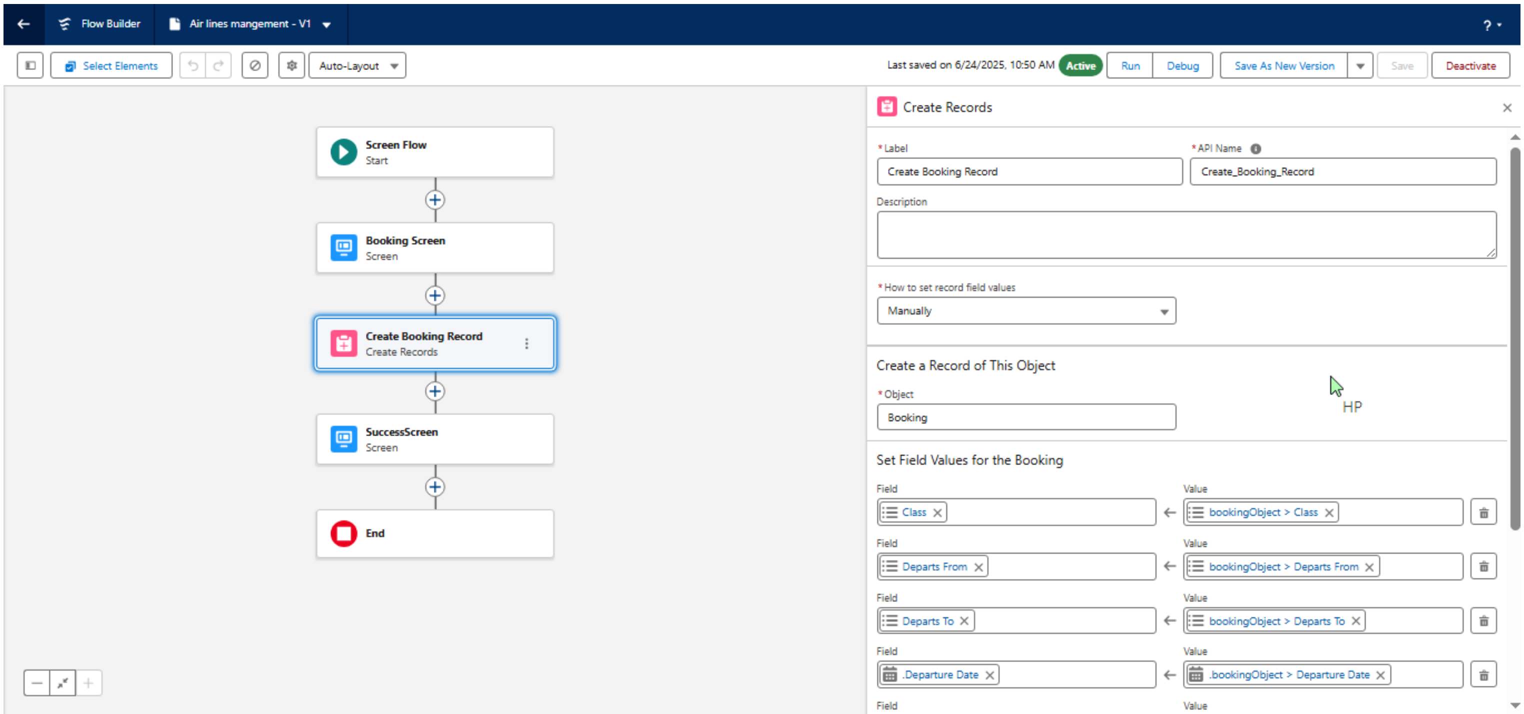
Date	**26-06-2025**
Team Id	**LTVIP2025TMID29760**
Project Name	**Air Line Management System**
College Name	**Ideal Institute Of Technology**

Output Screenshots

➢ Booking Flow (Screen & Success Message)



Last saved on 6/24/2025, 10:50 AM



Last saved on 6/24/2025, 10:50 AM

Create Records

*Label: Create Booking Record

*API Name: Create_Booking_Record

Description

*How to set record field values: Manually

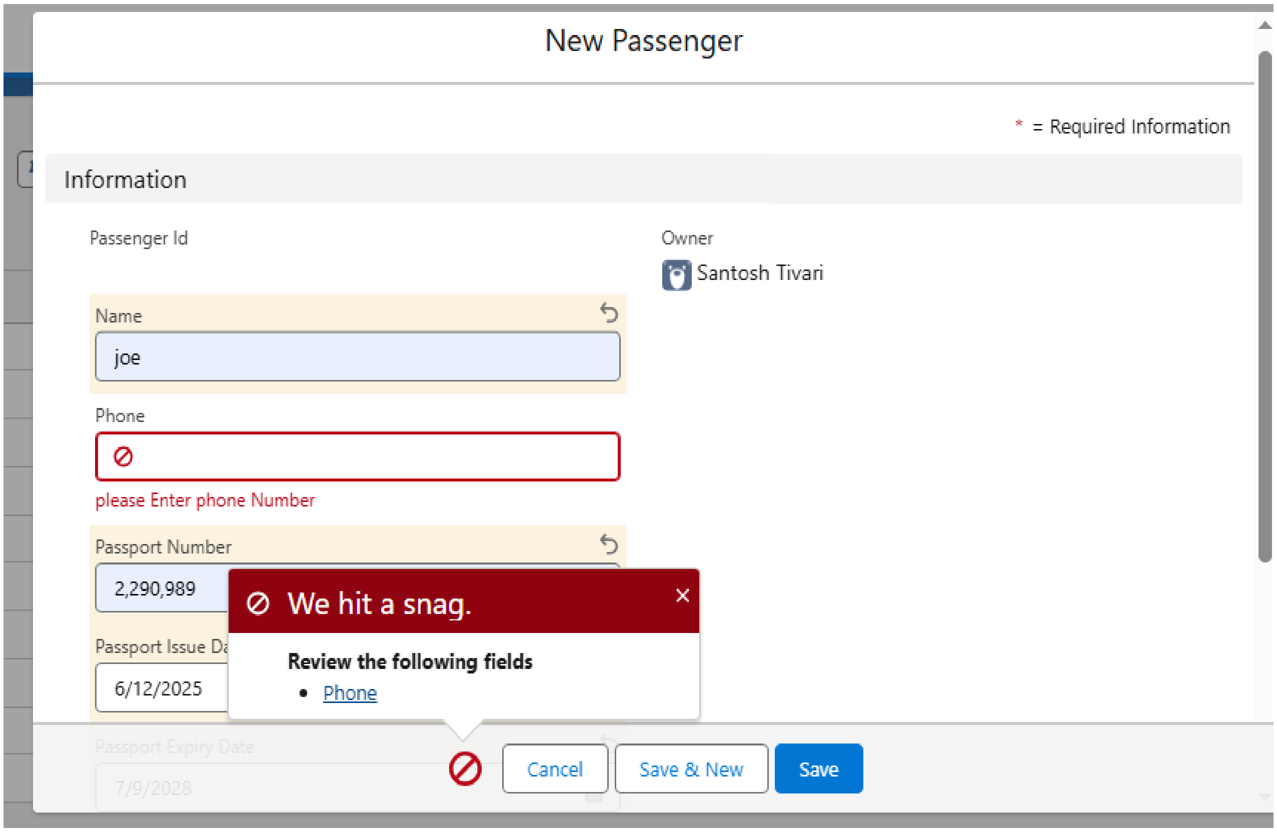
Create a Record of This Object

*Object: Booking

Set Field Values for the Booking

Field	Value
Class	bookingObject > Class
Departs From	bookingObject > Departs From
Departs To	bookingObject > Departs To
Departure Date	bookingObject > Departure Date
Field	Value

- ➢ Trigger Error on Passenger Phone



New Passenger

* = Required Information

Information

Passenger Id

Owner
Santosh Tivari

Name
joe

Phone

please Enter phone Number

Passport Number
2,290,989

Passport Issue Da
6/12/2025

Passport Expiry Date
7/9/2028

We hit a snag.

Review the following fields
- Phone

Trigger Validation Error When Mobile Number Is Missing

To ensure data accuracy, I implemented a **before-insert Apex trigger** that prevents a **Passenger** record from being created without a mobile number. This validation was handled using a custom **Apex class** and tested with both valid and invalid inputs.

Class Name: PhnValid_PassengerObj.cls

```
public class PhnValid_PassengerObj {

    public static void valMethod(List<Passenger__c> newPass){
        for(Passenger__c p : newPass){
            if(p.Phone__c == null){
                p.Phone__c.addError('Please enter phone number');
            }
        }
    }

}
```

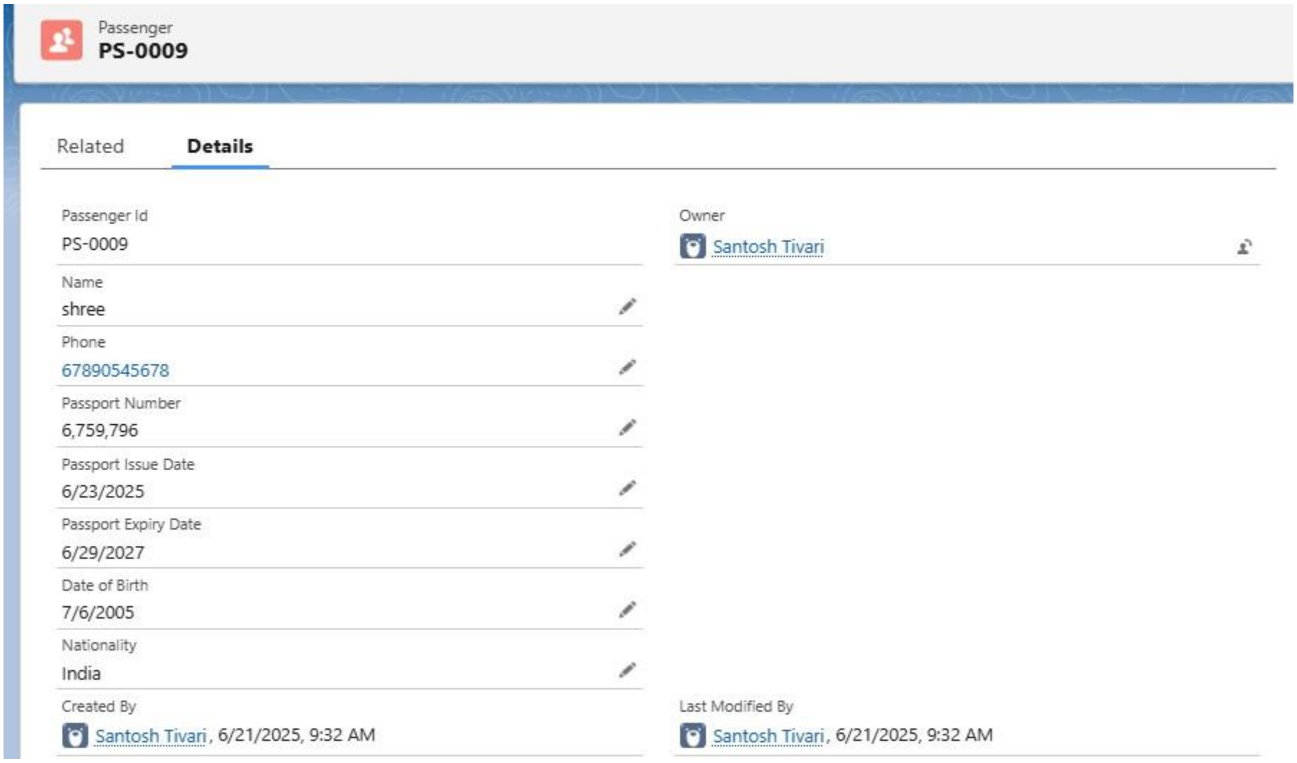
This class checks every passenger record and throws an error if the phone number is missing.

Step 2: Created an Apex Trigger

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

This trigger runs **before insert** and calls the validation method.

➢ Record Creation Example



Passenger
PS-0009

Passenger Id
PS-0009

Owner
Santosh Tivari

Name
shree

Phone
67890545678

Passport Number
6,759,796

Passport Issue Date
6/23/2025

Passport Expiry Date
6/29/2027

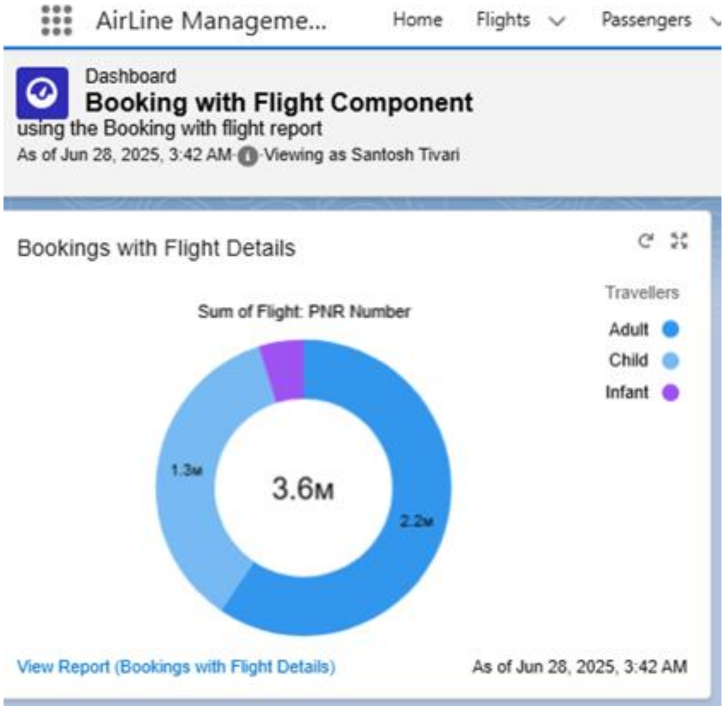
Date of Birth
7/6/2005

Nationality
India

Created By
Santosh Tivari, 6/21/2025, 9:32 AM

Last Modified By
Santosh Tivari, 6/21/2025, 9:32 AM

➢ Dashboard Display (Donut Chart)



Dashboard

Booking with Flight Component

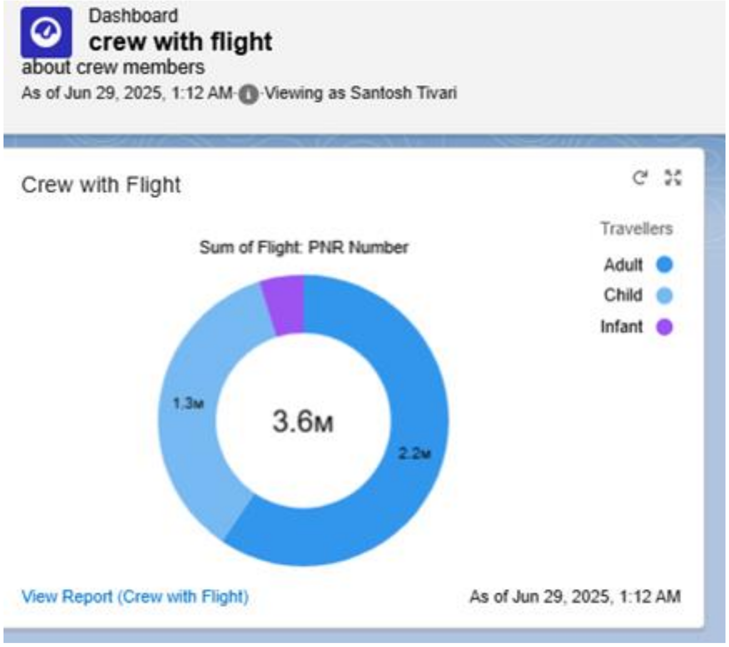
using the Booking with flight report

As of Jun 28, 2025, 3:42 AM · Viewing as Santosh Tivari

Bookings with Flight Details

View Report (Bookings with Flight Details)

As of Jun 28, 2025, 3:42 AM



Dashboard

crew with flight

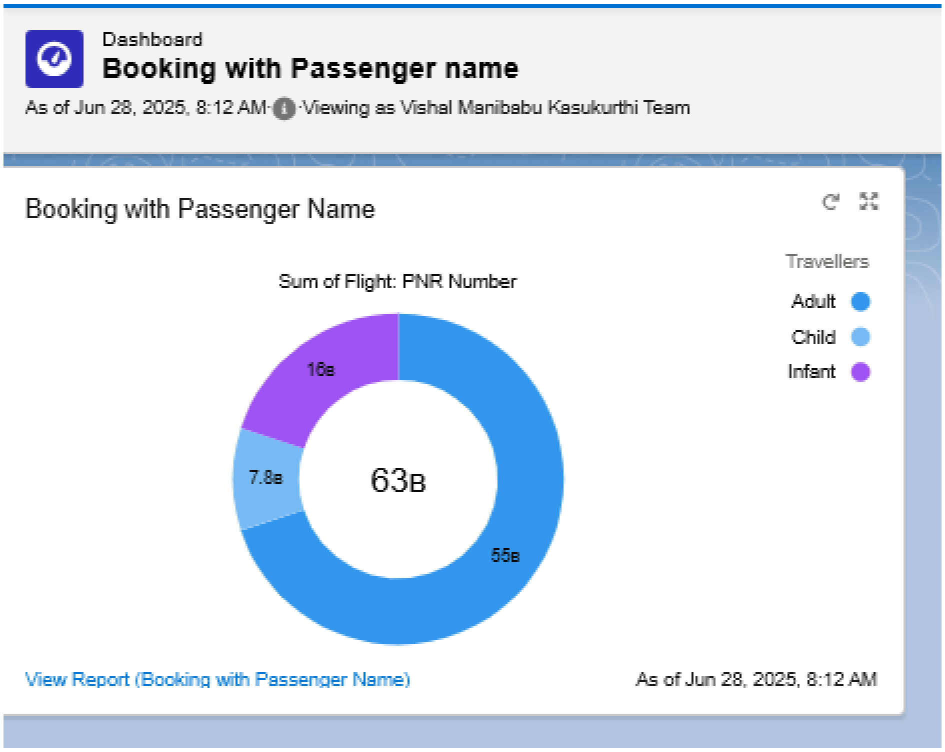
about crew members

As of Jun 29, 2025, 1:12 AM · Viewing as Santosh Tivari

Crew with Flight

View Report (Crew with Flight)

As of Jun 29, 2025, 1:12 AM



Dashboard

Booking with Passenger name

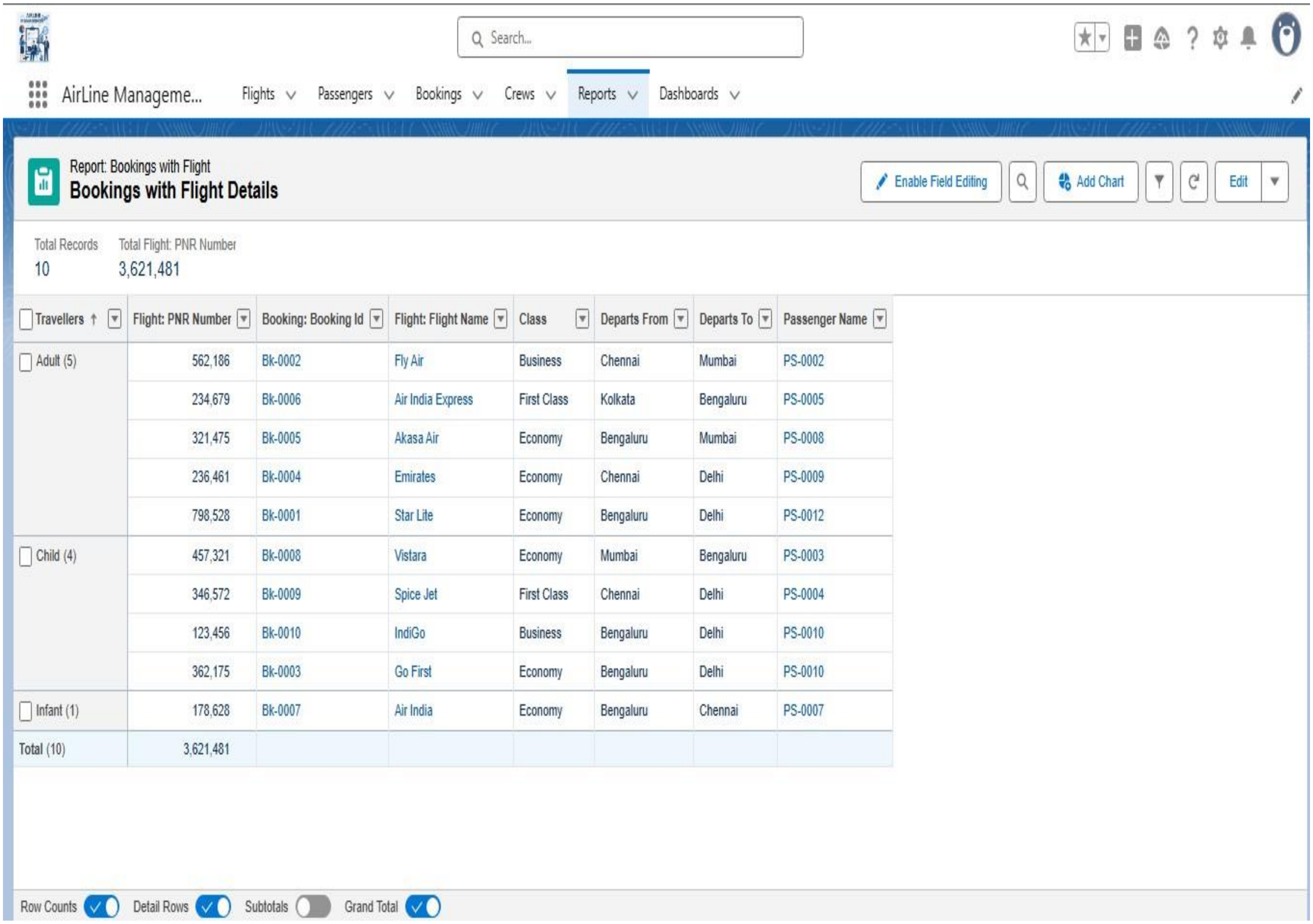
As of Jun 28, 2025, 8:12 AM · Viewing as Vishal Manibabu Kasukurthi Team

Booking with Passenger Name

View Report (Booking with Passenger Name)

As of Jun 28, 2025, 8:12 AM

➢ Report View (Grouped Bookings)



Report: Bookings with Flight

Bookings with Flight Details

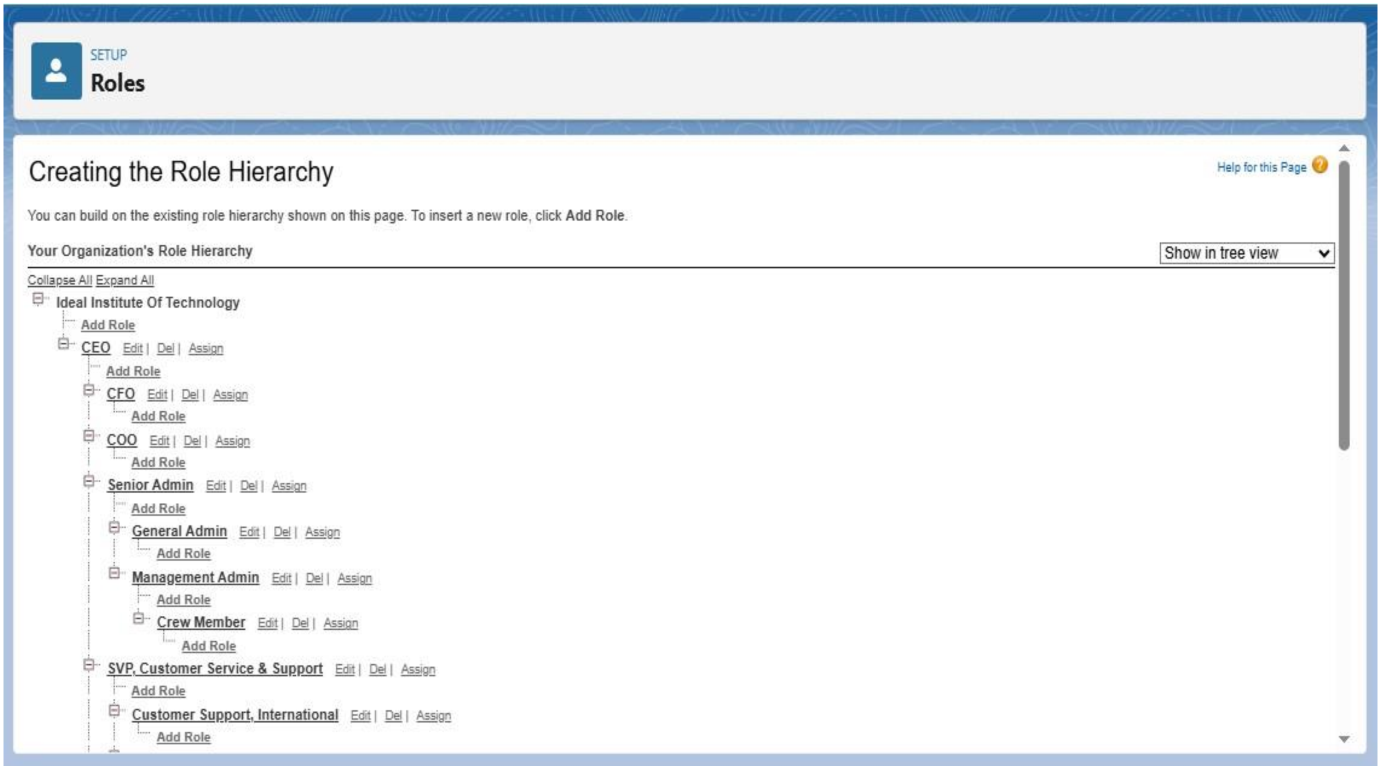
Total Records 10

Total Flight: PNR Number 3,621,481

Travellers ↑	Flight: PNR Number	Booking: Booking Id	Flight: Flight Name	Class	Departs From	Departs To	Passenger Name
□ Adult (5)	562,186	Bk-0002	Fly Air	Business	Chennai	Mumbai	PS-0002
	234,679	Bk-0006	Air India Express	First Class	Kolkata	Bengaluru	PS-0005
	321,475	Bk-0005	Akasa Air	Economy	Bengaluru	Mumbai	PS-0008
	236,461	Bk-0004	Emirates	Economy	Chennai	Delhi	PS-0009
	798,528	Bk-0001	Star Lite	Economy	Bengaluru	Delhi	PS-0012
□ Child (4)	457,321	Bk-0008	Vistara	Economy	Mumbai	Bengaluru	PS-0003
	346,572	Bk-0009	Spice Jet	First Class	Chennai	Delhi	PS-0004
	123,456	Bk-0010	IndiGo	Business	Bengaluru	Delhi	PS-0010
	362,175	Bk-0003	Go First	Economy	Bengaluru	Delhi	PS-0010
□ Infant (1)	178,628	Bk-0007	Air India	Economy	Bengaluru	Chennai	PS-0007
Total (10)	3,621,481						

Row Counts ✓ Detail Rows ✓ Subtotals ○ Grand Total ✓

➢Role Hierarchy View



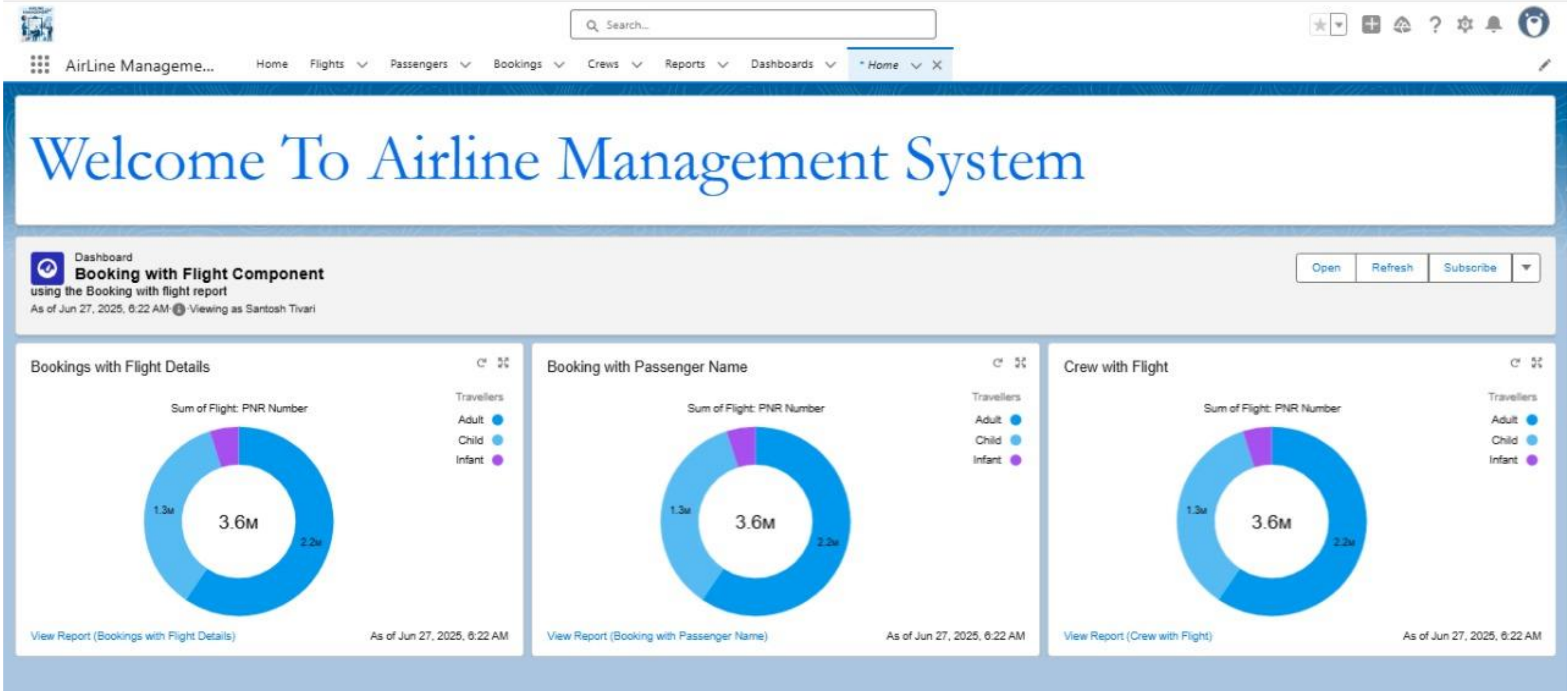
Creating the Role Hierarchy

You can build on the existing role hierarchy shown on this page. To insert a new role, click **Add Role**.

Your Organization's Role Hierarchy

- Ideal Institute Of Technology
  - Add Role
  - CEO Edit | Del | Assign
    - Add Role
    - CFO Edit | Del | Assign
      - Add Role
    - COO Edit | Del | Assign
      - Add Role
    - Senior Admin Edit | Del | Assign
      - Add Role
      - General Admin Edit | Del | Assign
        - Add Role
      - Management Admin Edit | Del | Assign
        - Add Role
        - Crew Member Edit | Del | Assign
          - Add Role
    - SVP, Customer Service & Support Edit | Del | Assign
      - Add Role
      - Customer Support, International Edit | Del | Assign
        - Add Role

- ➢ Lightning App Home Page preview



Search...
AirLine Manageme...
Home
Flights
Passengers
Bookings
Crews
Reports
Dashboards
* Home

Welcome To Airline Management System

Dashboard

Booking with Flight Component

using the Booking with flight report

As of Jun 27, 2025, 6:22 AM · Viewing as Santosh Tivari

Open
Refresh
Subscribe
Bookings with Flight Details
Sum of Flight: PNR Number
Travellers
Adult
Child
Infant
1.3M
3.6M
2.2M
View Report (Bookings with Flight Details)
As of Jun 27, 2025, 6:22 AM
Booking with Passenger Name
Sum of Flight: PNR Number
Travellers
Adult
Child
Infant
1.3M
3.6M
2.2M
View Report (Booking with Passenger Name)
As of Jun 27, 2025, 6:22 AM
Crew with Flight
Sum of Flight: PNR Number
Travellers
Adult
Child
Infant
1.3M
3.6M
2.2M
View Report (Crew with Flight)
As of Jun 27, 2025, 6:22 AM

8. ADVANTAGES & DISADVANTAGES

Advantages:

- Centralized Data Handling: All key records (Passengers, Flights, Bookings, Crew) are managed efficiently.
- Validation via Triggers: Apex triggers ensure data integrity (e.g., phone number is mandatory).
- Visual Dashboards: Booking statistics and trends are viewable in real-time via dashboards.
- Lightning Home Page: A user-friendly home page improves user experience.
- Modular & Scalable: seat selection, email
- Cloud-Based (Salesforce): user-based access con

Disadvantages:

- Manual Data Entry: for demo.
- No Real-Time Integration: or payment integration
- Limited User Interface: Only for internal use (no public-facing portal).
- Requires Salesforce License: Can't run independently without Salesforce platform access.

9. CONCLUSION

This Airline Management System project was successfully implemented using the Salesforce platform. It covers the core modules like Passenger, Flight, Booking, and Crew with real-time data validation and visualization. Triggers and test classes enhance the reliability of data entry, while dashboards provide insights for decision-making.

The system is scalable and can be extended with additional functionalities in future phases.

10. FUTURE SCOPE

- Flight Status Integration: Connect with live APIs for real-time flight tracking.
- Online Booking Portal: Develop a public web interface for passengers.
- Payment Gateway: Integrate online payment systems.
- Notifications: Add SMS/email alerts for ticket confirmations or cancellations.
- Dynamic Seat Selection: Allow passengers to choose seats during booking.
- AI Integration: Use AI for fare prediction or demand forecasting.

11. APPENDIX

Source Code (Apex Classes & Triggers):

Apex Class: PhnValid_PassengerObj.cls

```
public class PhnValid_PassengerObj {
    public static void valMethod(List<Passenger__c> newPass){
        for(Passenger__c p : newPass){
            if(p.Phone__c == null){
                p.Phone__c.addError('please Enter phone Number');
            }
        }
    }
}
```

Apex Trigger: PhnValidTrigger.trigger

```
trigger PhnValidTrigger on Passenger__c (before insert) {
    if(trigger.isBefore && trigger.isInsert){
        PhnValid_PassengerObj.valMethod(trigger.new);
    }
}
```

Apex Test Class: PhnValid_TestClass.cls

```
@isTest
public class PhnValid_TestClass {
    @isTest
    public static void testClass(){
        List<Passenger__c> varlis = new List<Passenger__c>();
        Passenger__c var = new Passenger__c();
        var.Phone__c = null;
        varlis.add(var);
        insert varlis;
        PhnValid_PassengerObj.valMethod(varlis);
    }
}
```

- **Dataset Link: Data entered through custom forms**

🔗 **GitHub Repository: https://github.com/Satyaharinitalatam/Airline-Management-System/upload/main/Document**

- 🎥 **Project Demo Video: https://drive.google.com/file/d/1H7pETEq61LKyBEfhShAGVCGPnEbMbFKd/view?usp=drive_link**