
FINAL PROJECT REPORT

Project Title: A CRM Application for Public Transport Management System

Team Information

Team ID	LTVIP2025TMID29469	Email ID
Team Size	4	
Team Leader	N Saisree	22F21A3738@gatesit.ac.in
Team Member	T Muhammad Tahir	Mahammadtahir143@gmail.com
Team Member	Maniyar Ameesha Siddiqा	abdullabasha920@gmail.com
Team Member	Kayapati Gayathri	gayethrigaye03@gmail.com

1. INTRODUCTION

Building a Centralized CRM for Efficient Public Transport Management

1.1 Project Overview – Transforming RTC Operations through CRM Technology

In an era of growing urbanization and mobility needs, public transport systems are expected to deliver efficient, timely, and accountable services. However, many Regional Transport Corporations (RTCs) still rely on outdated, manual processes for managing their daily operations. These challenges include inconsistent employee management, error-prone fare calculation, and the lack of centralized reporting systems.

To address these gaps, this project introduces a custom-built CRM application using the Salesforce platform. Designed specifically for RTCs, this solution digitizes critical processes such as employee role assignment, trip creation, ticket fare calculation, and data reporting.

It harnesses Salesforce's automation tools, custom object modeling, and analytics capabilities to create an intelligent, scalable, and maintainable system.

Key Features of the Solution:

- Customized object model supporting buses, bus stations, employees, trips, and fares
- Automated logic for fare calculation and employee role validation using flows and Apex triggers
- Formula-based fields for real-time computation of trip fare, employee age, and more
- Comprehensive dashboards and reports for administrative decision-making
- Centralized control with accurate data relationships and validations

1.2 Purpose – Solving Operational Inefficiencies in RTCs

The purpose of this project is to address the real-world problems faced by RTCs due to fragmented workflows and manual data entry. By implementing a centralized CRM system using Salesforce, the project aims to:

- Streamline operations through automated data flows
- Improve data accuracy using built-in validation rules and relationships
- Enable informed decision-making through dynamic reporting tools
- Create a system that is flexible, user-friendly, and scalable for future integration

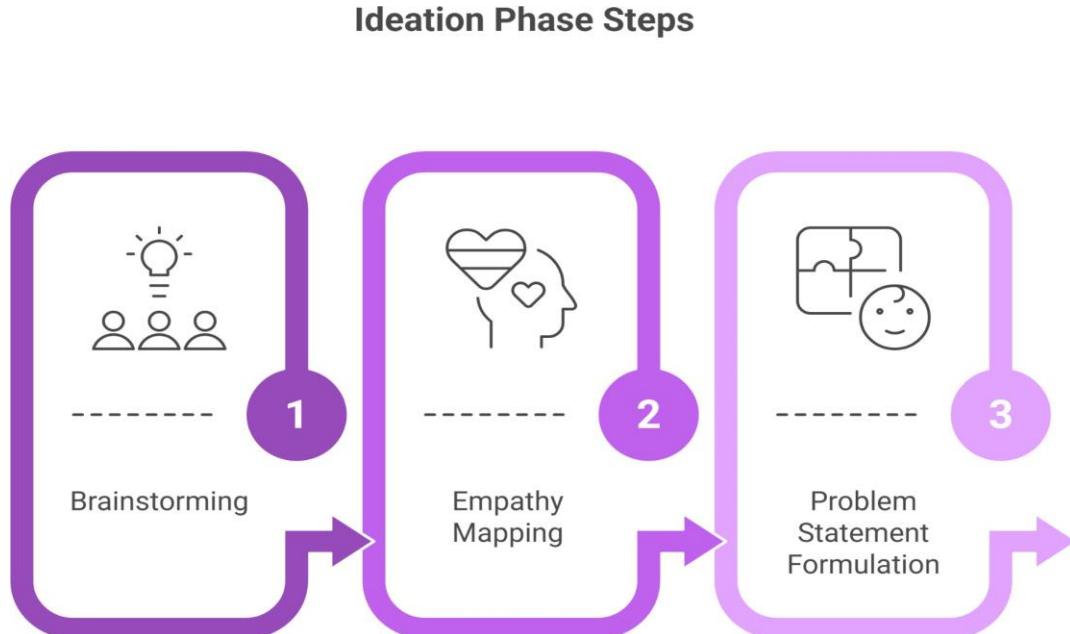
2. Ideation Phase

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 Public Transport Management System”. This phase was critical to ensure we built a system tailored to the actual challenges faced by RTC departments, bus station managers, conductors, and passengers.

The ideation phase included three main steps:

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



2.1 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 RTC systems and proposing a tech-driven solution using Salesforce. Through collaborative meetings, online whiteboards, and use-case discussions, we collectively explored pain points faced by public transport staff and administrative heads. We reviewed real-world operations and identified that most RTC systems rely heavily on manual workflows for managing:

- Bus trip schedules and fares
- Driver and conductor assignments
- Ticket fare collection
- Monthly performance reporting

After several discussions, we clearly defined the core issue:

Problem Statement:

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

This became the backbone of our project scope.

Step 2: Brainstorm, Idea Listing, and Grouping

We performed a team-wide brainstorming session using a digital board where everyone contributed raw ideas. The ideas were categorized into themes:

- **Data Management:** centralized employee, bus, and trip data
- **Automation:** real-time fare calculation, role-based triggers
- **Reporting:** monthly dashboards for trip count, passenger data, and revenue
- **Validation & Access Control:** rule-based data integrity and secure access

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

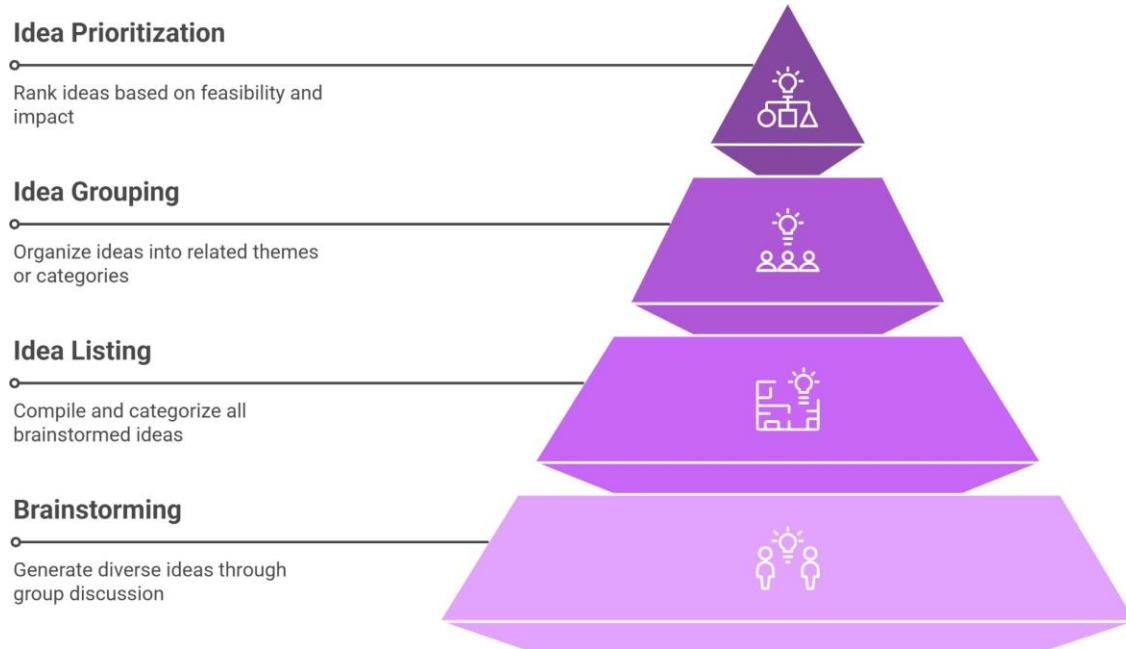
We created a decision matrix that helped us arrive at an **MVP** (Minimum Viable Product) plan:

- **Top Priority Features:**

- Automated fare and passenger data updates
- Role-based employee assignment (Driver/Conductor)
- Trigger-based alerts for invalid assignments
- Real-time summary dashboards (e.g., trips, passengers, revenue)
- Controlled and dependent picklists for bus and route management

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

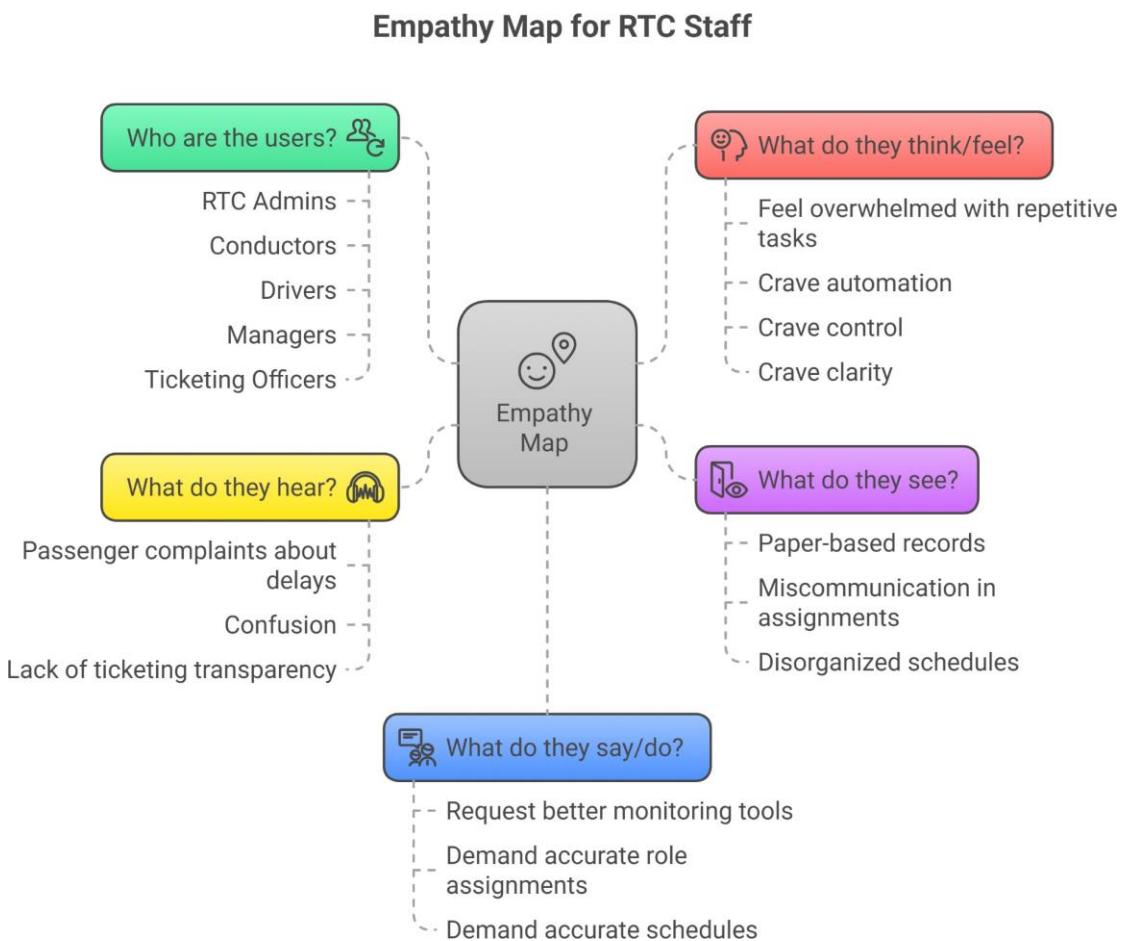
Idea Prioritization Pyramid



2.2 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 RTC staff, including administrators, bus station managers, and field employees.



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

2.3 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:

RTC departments manage public transportation 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.



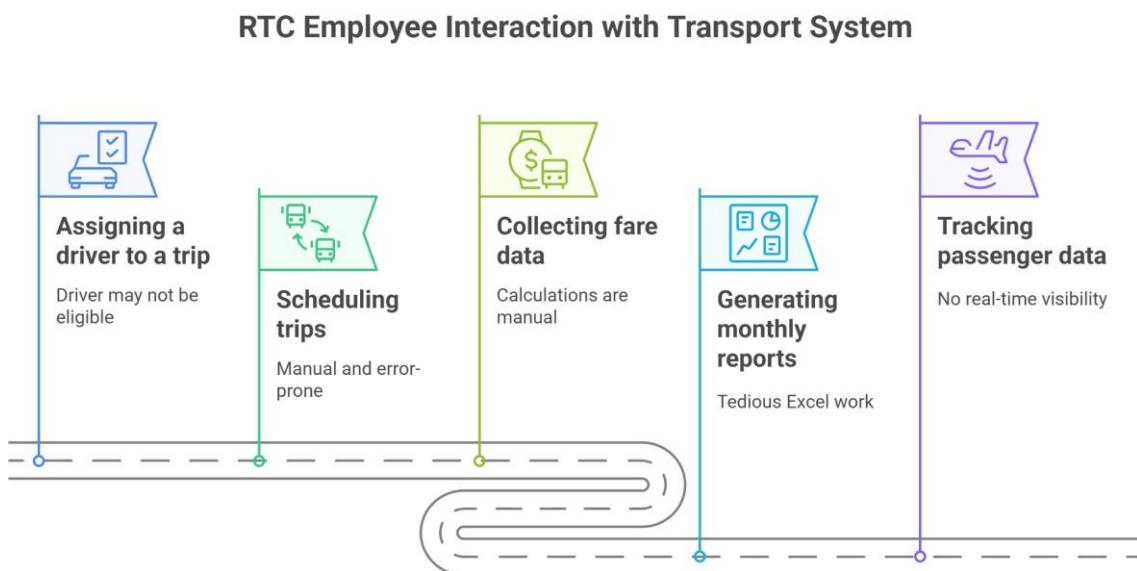
3. Requirement Analysis

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 (RTC departments, drivers, managers) truly need. In our project, "A CRM Application for Public Transport 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.

3.1 Customer Journey Map-Understanding User Experience Flow

Purpose: The **Customer Journey Map** visualises how RTC employees (e.g., Bus Station Managers, Admins, Drivers, Conductors) 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 RTC Admin):



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

3.2 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 CRM. It helped us structure relationships between:

- Bus Station, Bus, Trip, Employee, and Ticket Fare

Level 1 DFD Overview:

1. Admin Inputs:

- Adds bus, station, trip details
- Assigns drivers and conductors (from Employee object)

2. System Logic:

- Validates employee role (Driver or Conductor)
- Fetches fare via flow based on route + model
- Calculates total fare using formula (Passenger Count × Fare)

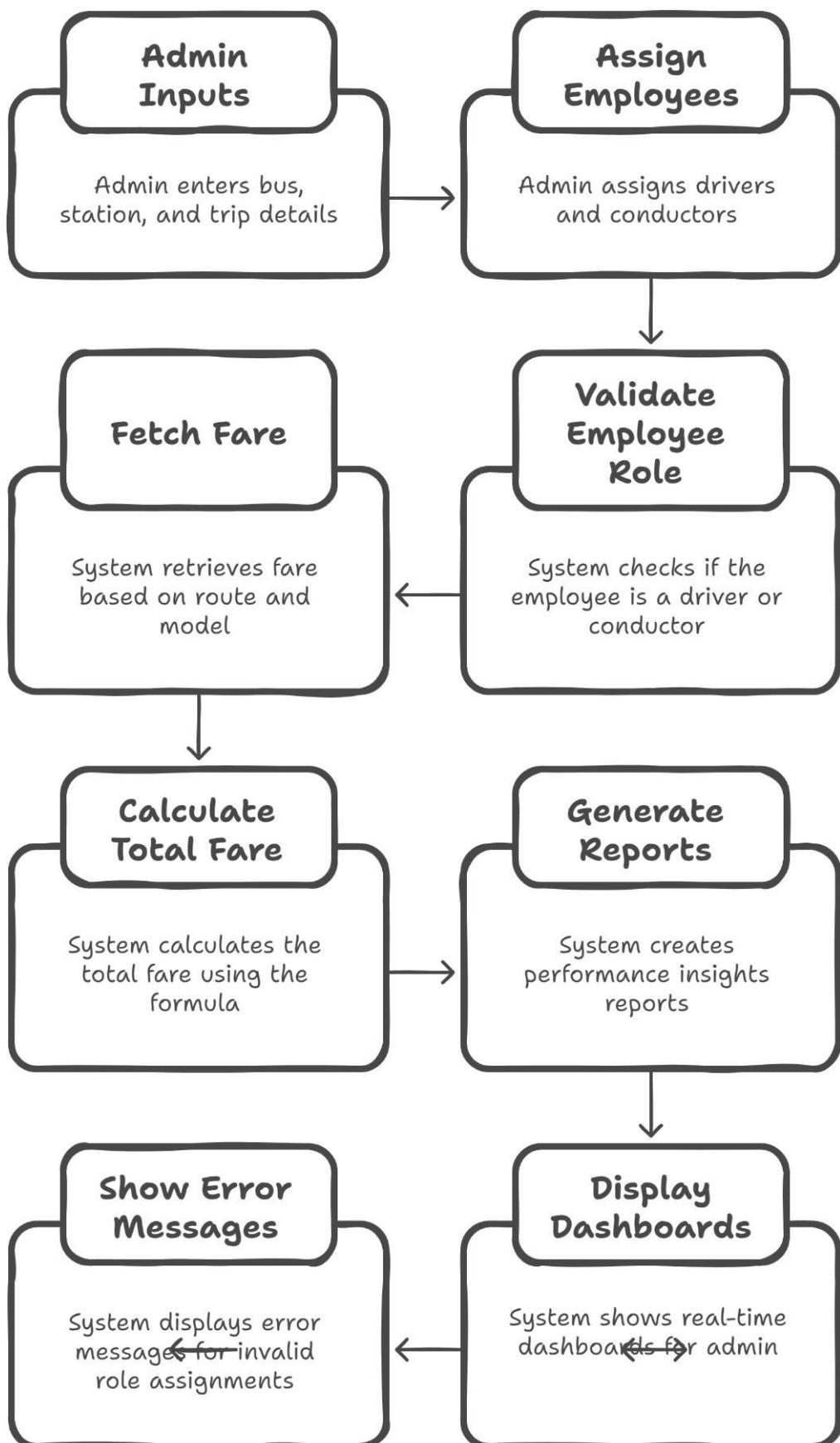
3. Outputs:

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

Objects Involved in Bus Transportation Process

Bus Station → Bus → Employee → Trip → Ticket Fare

Level 1 DFD Overview



3.3 Solution Requirements

What the CRM System Must Do

Functional Requirements:

- Create custom objects for: **Bus Station, Bus, Trip, Employee, 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

3.4 Technology Stack

Tools & Platforms Used for CRM Implementation

Category	Technology Used	Description
Platform	Salesforce Lightning	Used to build custom CRM using standard & custom objects
Automation	Flows & Validation Rules	Automate fare fetching and restrict invalid data
Custom Logic	Apex Triggers & Classes	Used to validate Driver/Conductor ID
Reports & Dashboards	Salesforce Reports	To analyze employee data, trips, revenue, etc.
UI/UX	Lightning App Builder	For creating a unified app view for Public Transport CRM

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.

4. 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, “A CRM Application for Public Transport Management System,” this phase bridges the gap between ideation and execution by converting insights from the previous requirement analysis into a well-structured CRM solution.

4.1 Problem–Solution Fit

Problem Recap:

RTC (Regional Transport Corporations) operate in a domain that requires real-time tracking of buses, trips, employees, and fare collection. However, operations are largely manual or siloed in spreadsheets, causing:

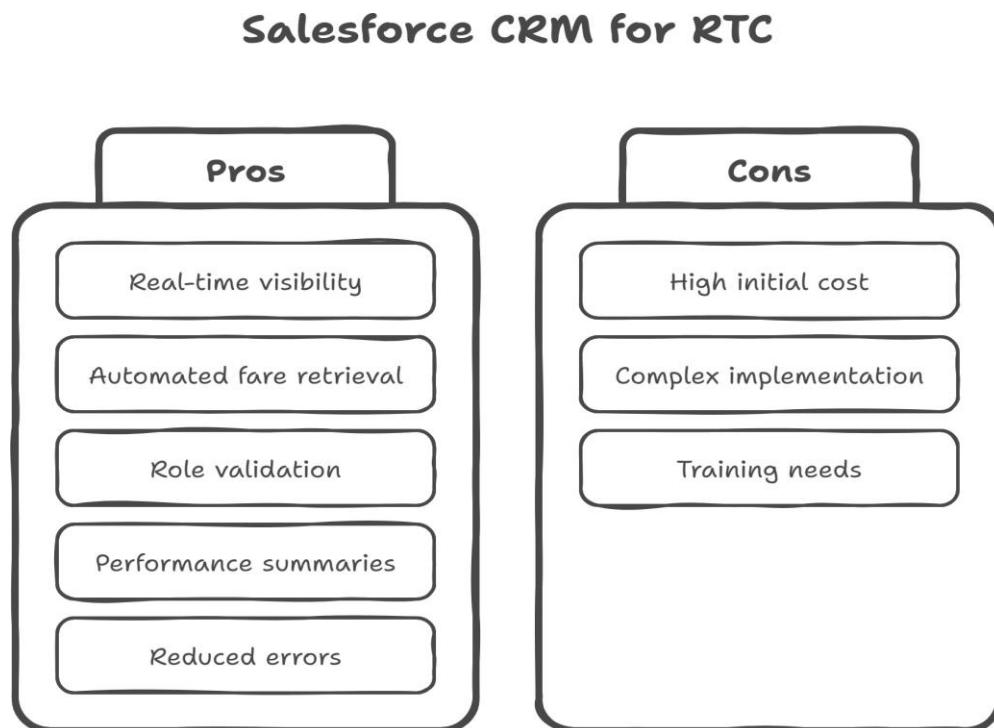
- Data entry errors and redundancy
 - No real-time visibility for decision-makers
 - Poor assignment of drivers/conductors
 - Difficulty in fare management and trip reporting
-

Does the Proposed Solution Fit?

Yes. The Salesforce CRM solution:

- Introduces **object-level control** for Buses, Stations, Employees, Trips, and Ticket Fares
- Enables **automated fare retrieval** through Flows
- Validates driver and conductor roles 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 RTC workflows.



4.2 Proposed Solution

How Our CRM Will Solve the Identified Problems

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

Key Functional Features:

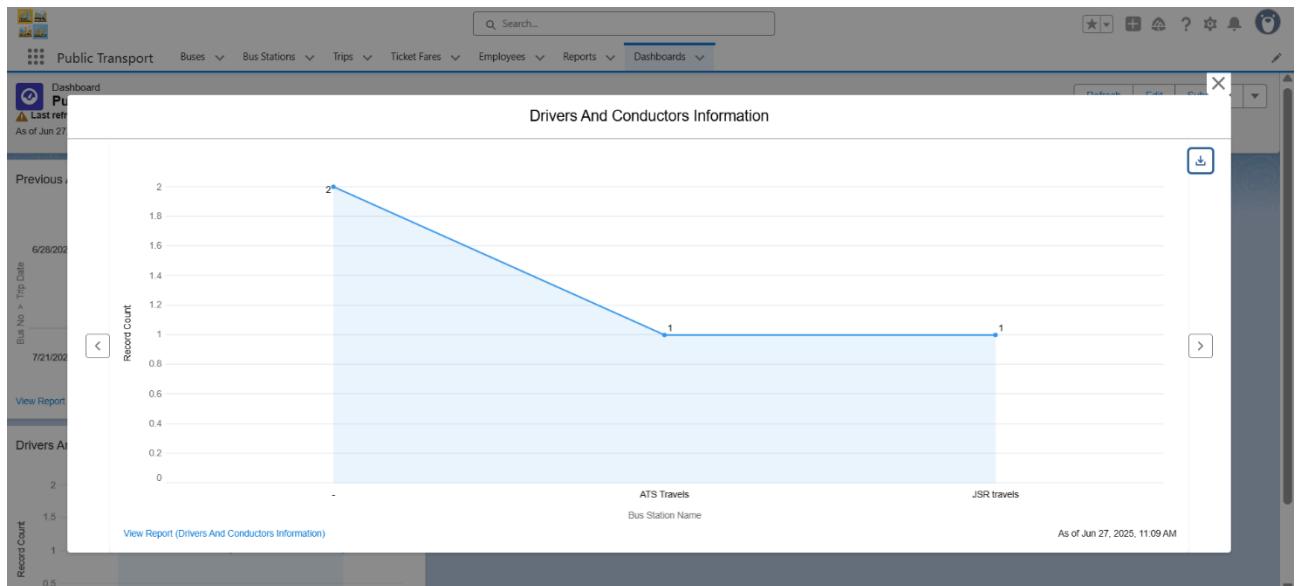
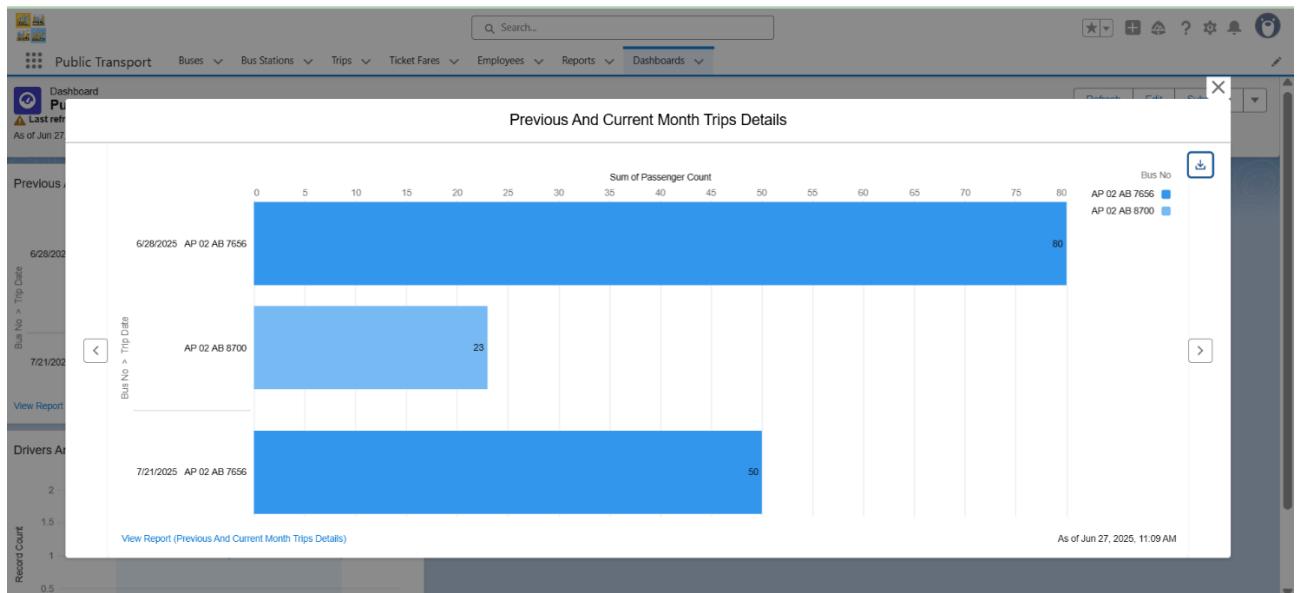
- **Custom Objects:**
 - Bus Station, Bus, Trip, Ticket Fare, Employee
- **Automation & Validation:**
 - Role verification (Driver/Conductor) via **Apex Triggers**
 - Fare calculation automation via **Flows**
 - Input control via **Validation Rules**
- **Formula Fields for Efficiency:**
 - Age, Experience,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

- 0 Page layouts designed by object and role

- **Reports and Dashboards:**

- 0 Trip Summary Reports
- 0 Driver/Conductor Assignment Reports
- 0 Monthly Revenue and Passenger Count Dashboards

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



4.3 Solution Architecture

Visualizing the Technical Structure and Object Relationships

Object Relationship Overview:

Object	Key Fields / Features
Bus Station	Name, Category, Amenities, Address
Bus	Linked to Bus Station, Category, Model, Capacity
Employee	Name, Role (Picklist), DOB, Experience, Lookup to Station
Trip	Linked to Bus, Driver, Conductor, Ticket Fare, Date, Passenger Count, Total Fare (Fx)
Ticket Fare	Route, Bus Model, Fare

Lookups:

- Employee → Bus Station
- Trip → Bus, Employee (Driver), Employee (Conductor), Ticket Fare
- Bus → Bus Station

Formulas:

- Total_Amount__c = Passenger_Count__c * Ticket_Fare__c
 - Driver_Name__c = Driver_Id__r.Employee_Name__c
 - Conductor_Name__c = Conductor_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 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.

5. 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. Structured task breakdown and time-bound execution planning ensures delivery efficiency.

5.1 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 – Bus, Bus Station, Employee, Trip, 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.

6. Project Executable Files

This phase outlines the actual Salesforce configurations, data, and outcomes used and generated during the execution of your project: "A CRM Application for Public Transport 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.

6.1 Project Files

Project Executable Files

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

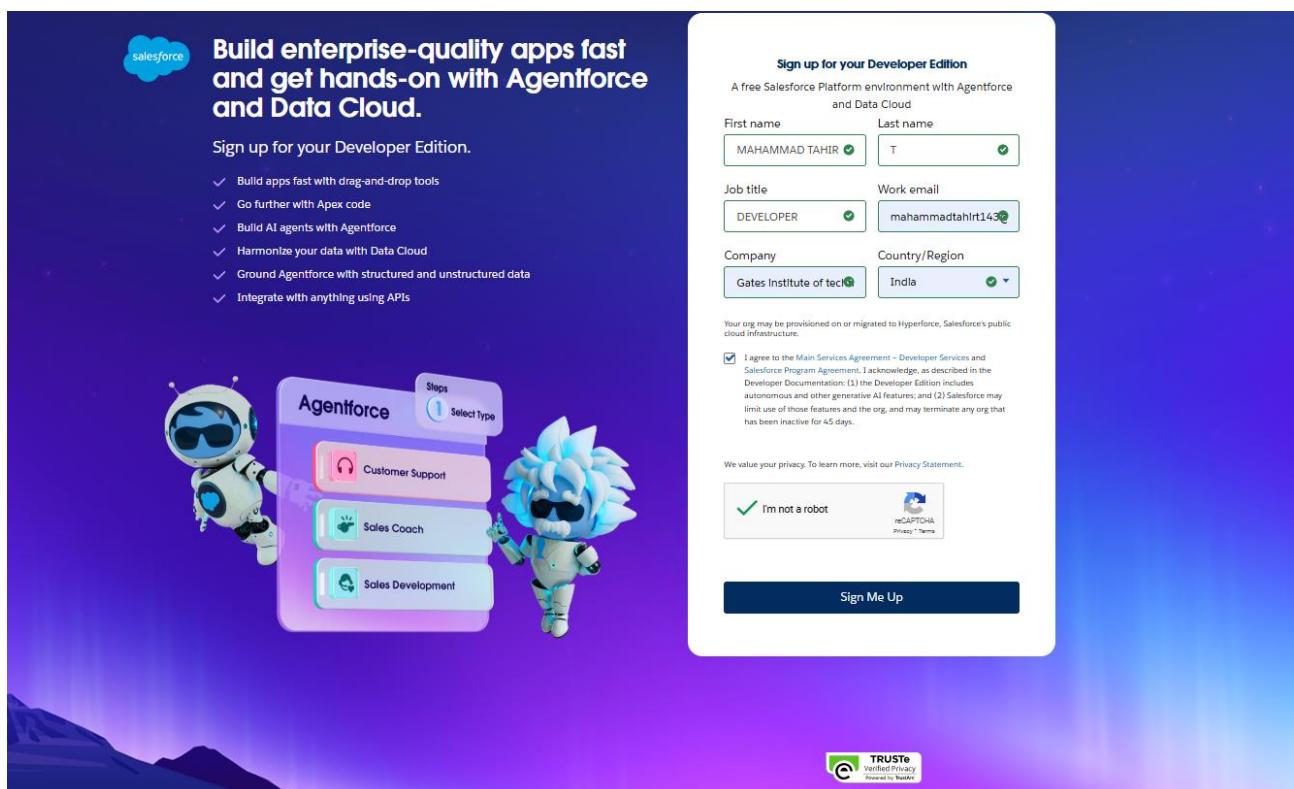
-  **Milestone 1: Developer Account Setup**
-  **Milestone 2: Object Creation**
-  **Milestone 3: Tab Creation**
-  **Milestone 4: Lightning App Setup**
-  **Milestone 5: Field Creation**
-  **Milestone 6: Page Layouts**
-  **Milestone 7: Validation Rules**
-  **Milestone 8: Flow Setup**
-  **Milestone 9: Apex Trigger**
-  **Milestone 10: Reports**
-  **Milestone 11: Dashboards**
-  **Milestone 12: Final Review**

List of Milestone Tasks with Supporting Screenshots and Descriptions

📁 Milestone 1: Developer Account Setup

- Created and activated a Salesforce Developer Org.
 - Link: <https://developer.salesforce.com/signup>
 - 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:
 - Bus Station, Bus, Employee, Trip, and Ticket Fare.
 - Established foundational schema for RTC data tracking.
 - Configured relationships using lookup fields.
-

OUTPUT SCREENSHOT

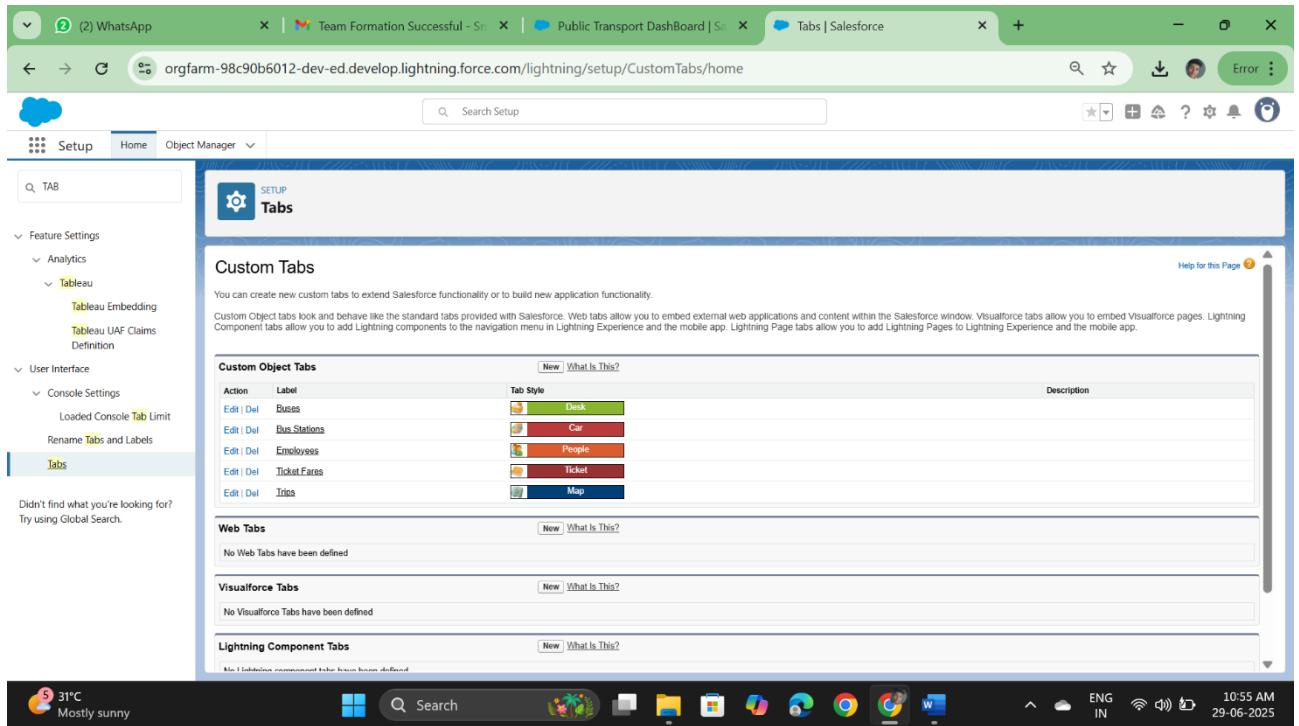
The screenshot shows the Salesforce Object Manager page. At the top, there are tabs for Setup, Home, and Object Manager. A search bar labeled "Search Setup" is present. On the right side of the header, there are icons for Quick Find, Schema Builder, and Create, along with a dropdown menu. The main area displays a table of objects:

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.

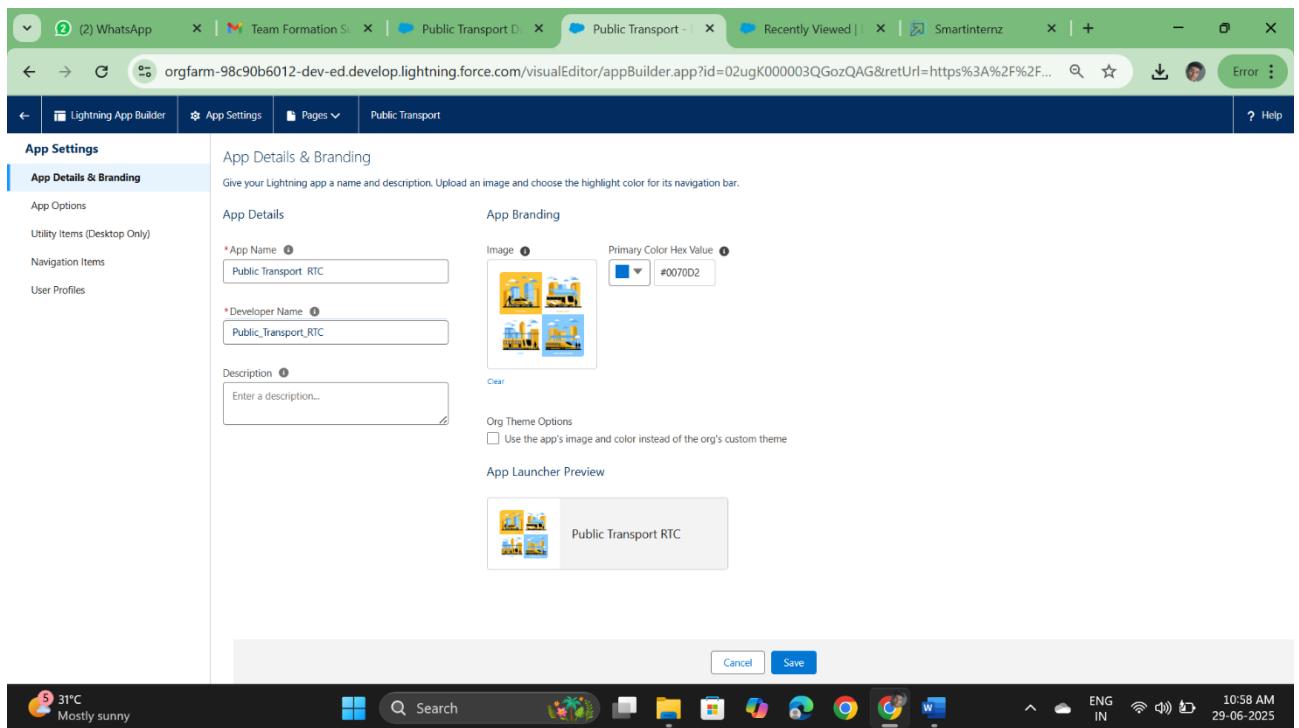
OUTPUT SCREENSHOT



Milestone 4: Lightning App Setup

- Built a custom Lightning App named "**Public Transport RTC**".
- Added relevant tabs to centralize operations.
- Simplified user workflow by grouping features.

OUTPUT SCREENSHOT

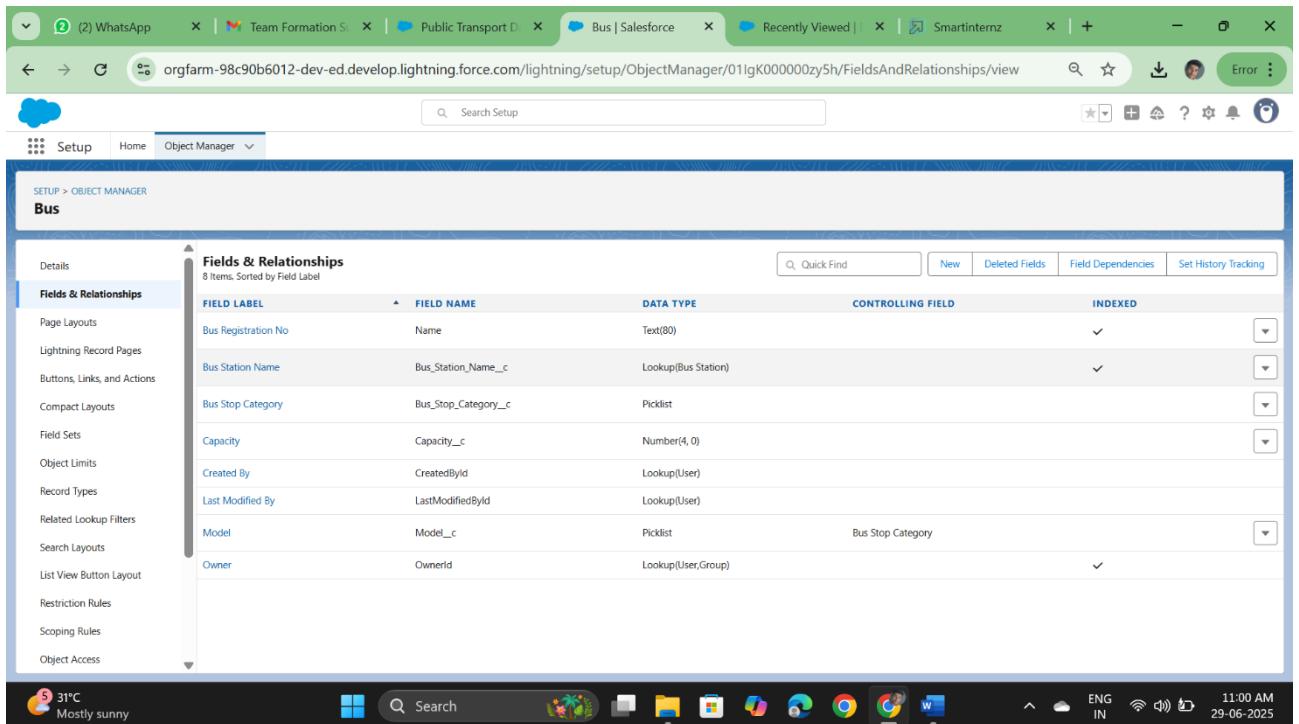


Milestone 5: Field Creation

- Added custom and formula fields like:
 - Role (Picklist), Age, Experience, Fare, Passenger Count.
 - Implemented: **Total Fare = Fare × Passenger Count** formula.
- Linked Driver and Conductor to Trip via lookup fields.

OUTPUT SCREENSHOT

Bus object Fields



The screenshot shows the Salesforce Object Manager interface for the 'Bus' object. The left sidebar lists various setup options like Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, etc. The main content area displays the 'Fields & Relationships' section for the 'Bus' object. It shows 8 items sorted by Field Label. The table includes columns for FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED status. The fields listed are:

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)		✓
Bus Stop Category	Bus_Stop_Category__c	Picklist		✓
Capacity	Capacity__c	Number(4, 0)		✓
Created By	CreatedById	Lookup(User)		✓
Last Modified By	LastModifiedById	Lookup(User)		✓
Model	Model__c	Picklist	Bus Stop Category	✓
Owner	OwnerId	Lookup(User,Group)		✓

Bus Station object Fields

The screenshot shows the Salesforce Object Manager interface for the 'Bus Station' object. The top navigation bar includes tabs for 'Setup', 'Home', and 'Object Manager'. The main content area displays the 'Fields & Relationships' section. A sidebar on the left lists various configuration categories such as Details, Fields & Relationships, Page Layouts, 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, Triggers, Flow Triggers, Validation Rules, and Conditional Field Formatting. The 'Fields & Relationships' table lists the following fields:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Amrities	Amrities__c	Picklist (Multi-Select)		
Bench	Bench__c	Checkbox		
Bus Station Name	Name	Text(80)		✓
Bus Stop Category	Bus_Stop_Catagory__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)		
Owner	Ownerid	Lookup(User/Group)		✓
Shared Available	Shared_available__c	Checkbox		
State/Province	State_Province__c	Text(25)		
Street	Street__c	Text Area(255)		
Zip/PostalCode	Zip_PostalCode__c	Text(10)		

The bottom of the screen shows the Windows taskbar with icons for File Explorer, Search, Task View, File, Settings, Control Panel, and Edge browser, along with system status indicators like battery level, signal strength, and date/time.

Employee object Fields

The screenshot shows the Salesforce Object Manager interface for the 'Employee' object. The top navigation bar includes tabs for 'Setup', 'Home', and 'Object Manager'. The main content area displays the 'Fields & Relationships' section. A sidebar on the left lists various configuration categories such as Details, Fields & Relationships, Page Layouts, 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, Triggers, Flow Triggers, Validation Rules, and Conditional Field Formatting. The 'Fields & Relationships' table lists the following fields:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Age	Age__c	Number		
Actual Time	Actual_Time__c	Text		
Bus Station Name	Bus_Station_Name__c	Lookup(Bus Station)		
City	City__c	Text(10)		
Country	Country__c	Text(10)		
Created By	CreatedById	Lookup(User)		
Date of Birth	Date_of_Birth__c	Date		
Date of Joining	Date_of_Joining__c	Date		
Date of Retirement	Date_of_Retirement__c	Date		
Employee ID	Name	Text(20)		
Employee Name	Employee_Name__c	Text(10)		
Experience	Experience__c	Number		
Last Modified By	LastModifiedById	Lookup(User)		
Manager	Managerid	Lookup(User/Group)		
Phone No	Phone_No__c	Phone		
Role	Role__c	Picklist		
Salary	Salary__c	Currency(10,2)		
State/Province	State_Province__c	Text(10)		
Street	Street__c	Text		
Work Phone	Work_Phone__c	Text(10)		
Zip/PostalCode	Zip_PostalCode__c	Text(10)		

Trip object Fields

The screenshot shows the Salesforce Object Manager interface for the 'Trip' object. The left sidebar lists various setup categories like Details, Fields & Relationships, Page Layouts, etc. The main content area displays the 'Fields & Relationships' section with 21 items. The table columns are FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED. Key fields include 'Arrival Time' (Field Name: Arrival_Time_c, Data Type: Picklist), 'Bus No.' (Field Name: Bus_No_c, Data Type: Lookup(Bus)), 'Bus Starting Terminal' (Field Name: Bus_Startng_Terminal_c, Data Type: Text(40)), 'Conductor Id' (Field Name: Conductor_Id_c, Data Type: Lookup(Employee)), 'Conductor Name' (Field Name: Conductor_Name_c, Data Type: Formula(Text)), 'Created By' (Field Name: CreatedByid, Data Type: Lookup(User)), 'Departure Time' (Field Name: Departure_Time_c, Data Type: Picklist), 'Destination Terminal' (Field Name: Destination_Terminal_c, Data Type: Text(30)), 'Driver' (Field Name: Driver_c, Data Type: Formula(Text)), 'Driver Id' (Field Name: Driver_Id_c, Data Type: Lookup(Employee)), 'Estimated Travel Time' (Field Name: Estimated_Travel_Time_c, Data Type: Number(18, 0)), 'Frequency Per Day' (Field Name: Frequency_Per_Day_c, Data Type: Number(2, 0)), 'Last Modified By' (Field Name: LastModifiedByid, Data Type: Lookup(User)), 'No. of Stops' (Field Name: No_of_Stops_c, Data Type: Number(2, 0)), 'Owner' (Field Name: Owner, Data Type: Lookup(User/Group)), 'Passenger Count' (Field Name: Passenger_Count_c, Data Type: Number(4, 0)), 'Route Name' (Field Name: Route_Name_c, Data Type: Lookup(Ticket Fare)), and 'Ticket Fare' (Field Name: Ticket_Fare_c, Data Type: Currency(10, 2)). The bottom status bar shows USD/INR exchange rate (-0.22%), system icons, and the date/time (11:05 AM, 29-06-2025).

Ticket fare object Fields

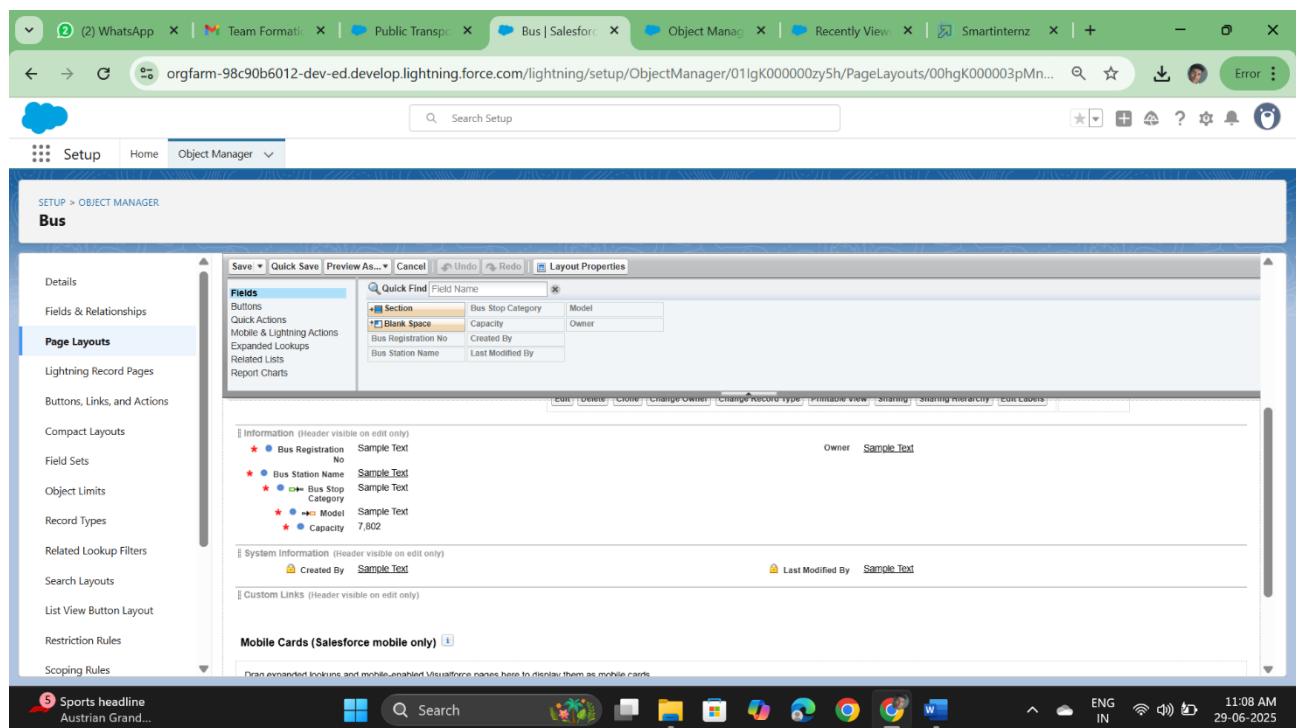
The screenshot shows the Salesforce Object Manager interface for the 'Ticket Fare' object. The left sidebar lists various setup categories like Details, Fields & Relationships, Page Layouts, etc. The main content area displays the 'Fields & Relationships' section with 6 items. The table columns are FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED. Key fields include 'Bus Model' (Field Name: Bus_Model_c, Data Type: Picklist), 'Created By' (Field Name: CreatedByid, Data Type: Lookup(User)), 'Last Modified By' (Field Name: LastModifiedByid, Data Type: Lookup(User)), 'Owner' (Field Name: OwnerId, Data Type: Lookup(User/Group)), 'Route Name' (Field Name: Name, Data Type: Text(80)), and 'Ticket Fare' (Field Name: Ticket_Fare_c, Data Type: Currency(10, 2)). The bottom status bar shows USD/INR exchange rate (-0.22%), system icons, and the date/time (11:06 AM, 29-06-2025).

Milestone 6: Page Layouts

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

OUTPUT SCREENSHOT

- Bus object Page Layouts



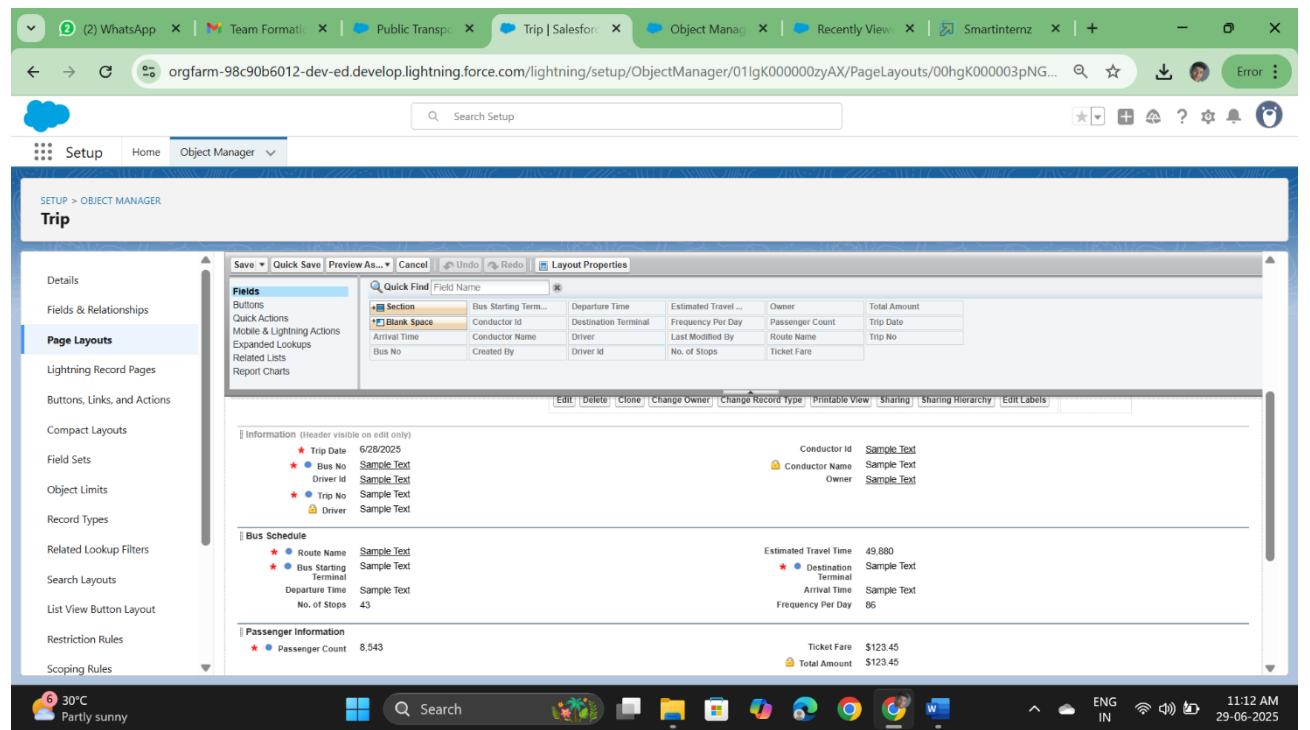
- Bus station object Page Layouts

The screenshot shows the Salesforce Object Manager interface for the 'Bus Station' object. The left sidebar has 'Page Layouts' selected. The main area displays the 'Bus Station Detail' page layout with various sections and fields. The 'Fields' section at the top lists fields like Bus Station Name, Last Modified By, State/Province, Bus Stop Category, Last Updated, Amenities, City, Owner, Zip/Postal/Code, Bench, and Created By. Below this is the 'Bus Station Detail' section with fields for Bus Station Name, Amenities, Bus Stop Category, Last Updated, Street, City, State/Province, Zip/Postal/Code, Owner, and Shelter available. There are also sections for Address Information, System Information, and Custom Links.

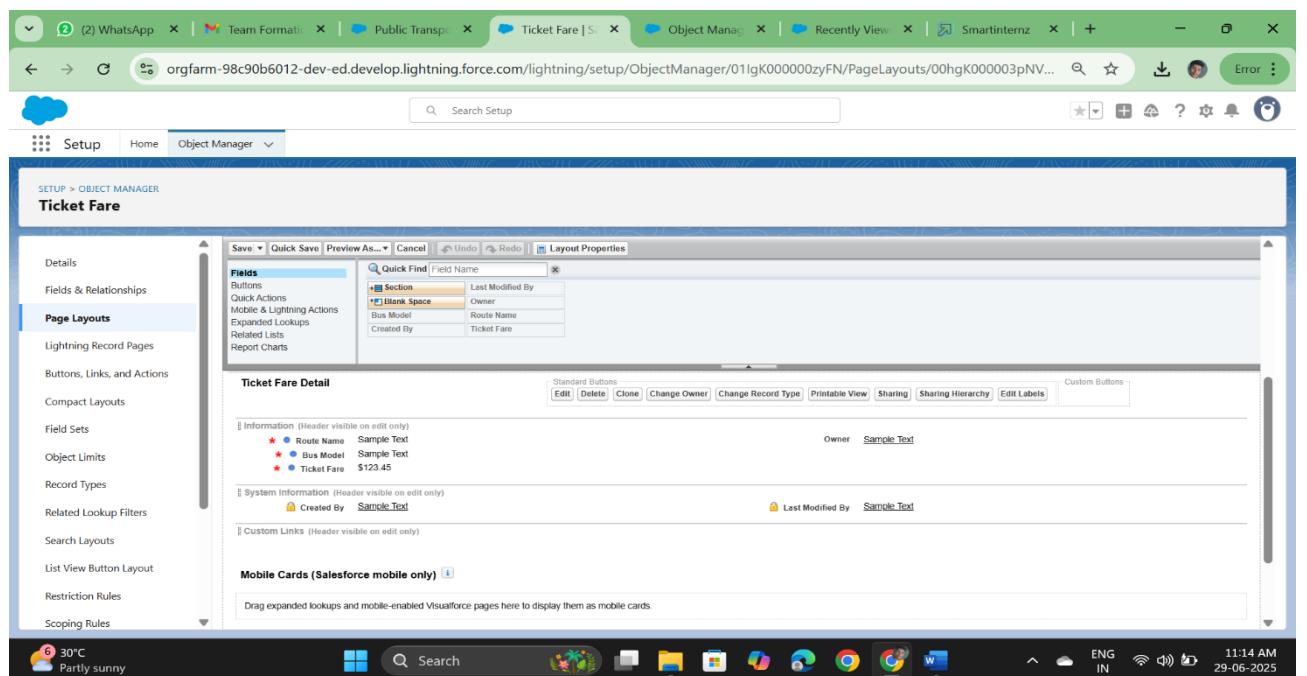
- Employee object Page Layouts

The screenshot shows the Salesforce Object Manager interface for the 'Employee' object. The left sidebar has 'Page Layouts' selected. The main area displays the 'Employee Detail' page layout with various sections and fields. The 'Fields' section at the top lists fields like Bus Station Name, Date of Birth, Employee Name, Phone No, Street, City, Date of Joining, Experience, Role, Work Place, Country, Zip/Postal/Code, Last Modified By, Employee Id, Owner, and State/Province. Below this is the 'Employee Detail' section with fields for Employee id, Employee Name, Bus Station Name, Salary, Experience, Date of Birth, Phone No, Work Place, Role, Date of Joining, Date of Retirement, Age, and Experience. There are also sections for Personal Details, Address, and System Information.

- Trip object Page Layouts



- Ticket Fare object Page Layouts

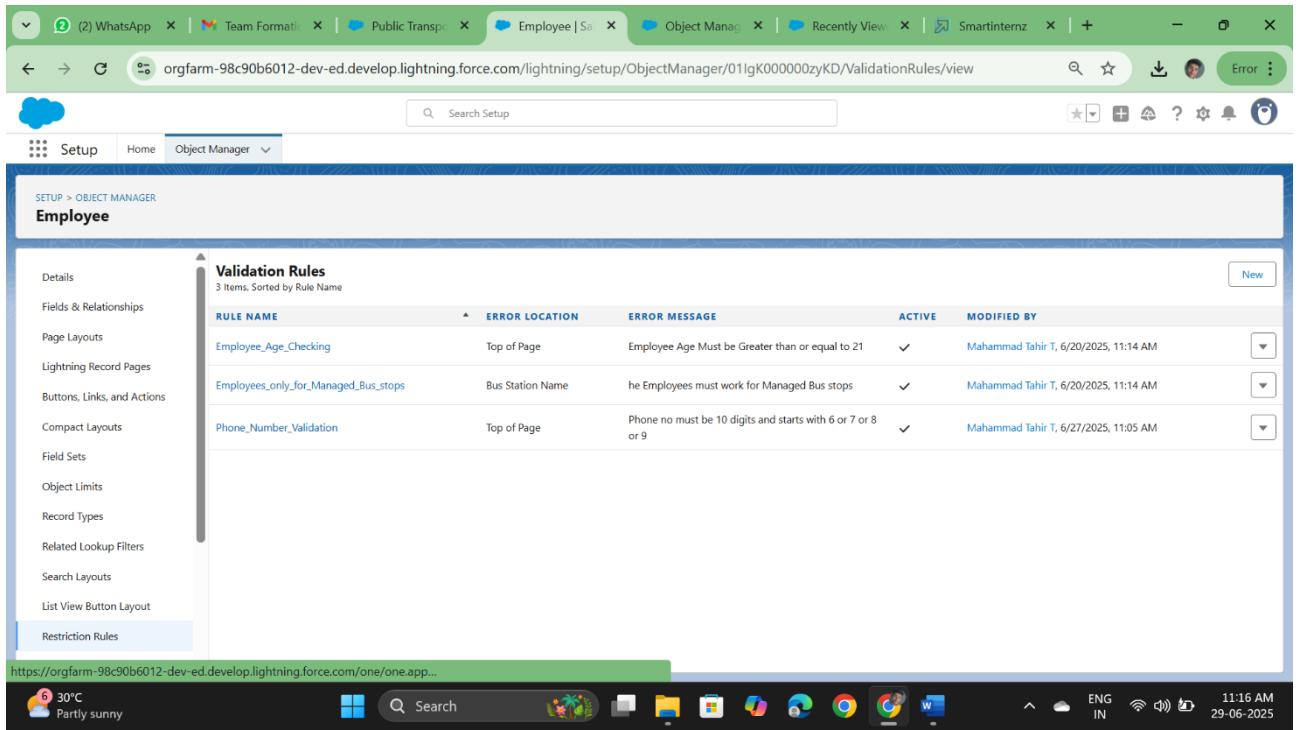


Milestone 7: Validation Rules

Enforced data quality with validations like:

- Age \geq 21, correct phone number format, required fields.
- Prevented invalid inputs and ensured system reliability.

OUTPUT SCREENSHOT



The screenshot shows the Salesforce Object Manager interface for the 'Employee' object. The left sidebar lists various setup options: 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, and Restriction Rules. The main content area displays the 'Validation Rules' section, which lists three rules:

Rule Name	Error Location	Error Message	Active	Modified By
Employee_Age_Checking	Top of Page	Employee Age Must be Greater than or equal to 21	✓	Mahammad Tahir T, 6/20/2025, 11:14 AM
Employees_only_for_Managed_Bus_stops	Bus Station Name	he Employees must work for Managed Bus stops	✓	Mahammad Tahir T, 6/20/2025, 11:14 AM
Phone_Number_Validation	Top of Page	Phone no must be 10 digits and starts with 6 or 7 or 8 or 9	✓	Mahammad Tahir T, 6/27/2025, 11:05 AM

The browser address bar shows the URL: orgfarm-98c90b6012-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/01lgK000000zyKD/ValidationRules/view. The system status bar at the bottom indicates a temperature of 30°C, weather as partly sunny, and a timestamp of 11:16 AM on 29-06-2025.

The screenshot shows the Salesforce Object Manager interface for the 'Bus' object. The left sidebar lists various setup options like Details, Fields & Relationships, Page Layouts, Lightning Record Pages, etc. The main content area displays a table titled 'Validation Rules' with one item:

RULE NAME	ERROR LOCATION	ERROR MESSAGE	ACTIVE	MODIFIED BY
Bus_Registration_Number_Validation	Top of Page	The bus Registration Number must be in the format of 2 Capital Letters(State Code), space, 2 Numbers (District Code), space 1 or 2 Capital Letters(Series), space and 4 Numbers (Number).	✓	Mahammad Tahir T, 6/23/2025, 11:50 PM

The status bar at the bottom shows it's 30°C, partly sunny, and the date is 29-06-2025.

The screenshot shows the Salesforce Object Manager interface for the 'Trip' object. The left sidebar lists various setup options. The main content area displays a table titled 'Validation Rules' with two items:

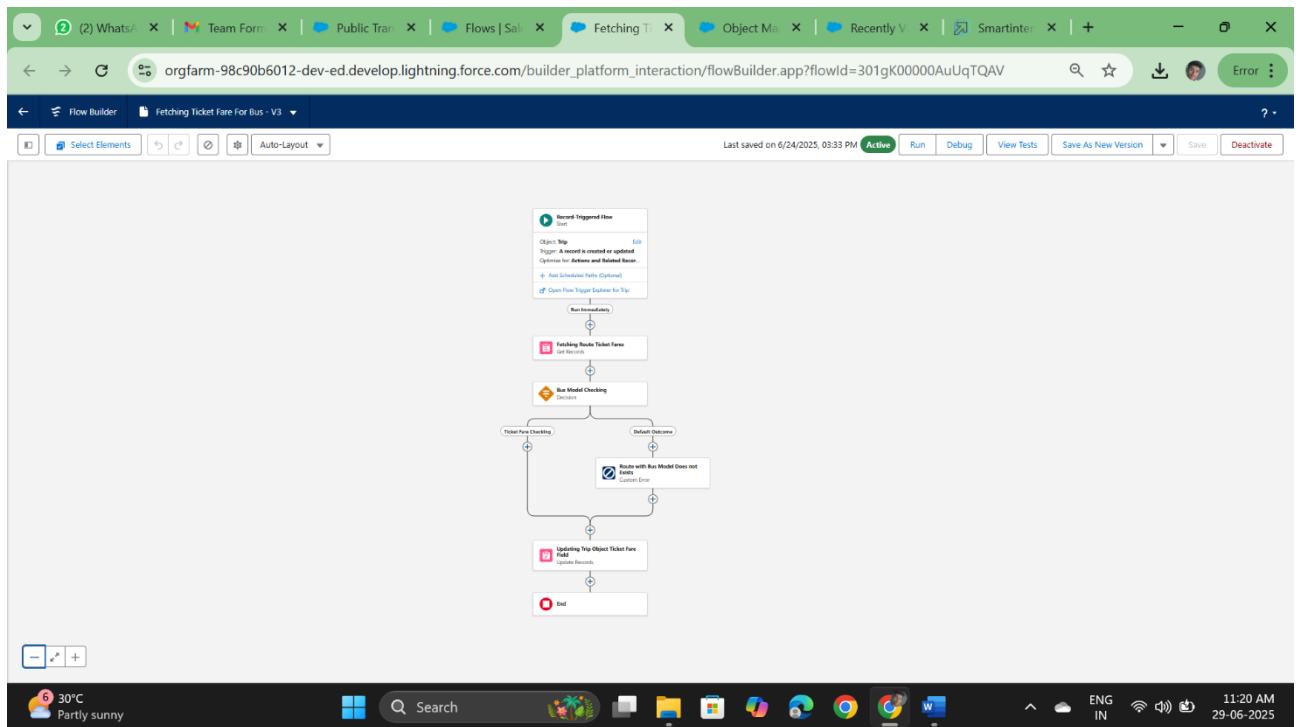
RULE NAME	ERROR LOCATION	ERROR MESSAGE	ACTIVE	MODIFIED BY
Departure_and_Arrival_Time_Checking	Top of Page	The Departure Time and Arrival Time Should not be the same	✓	Mahammad Tahir T, 6/20/2025, 11:09 AM
Passenger_Count_Checking_for_Few_Buses	Passenger Count	For Super Deluxe, Semi Sleeper and Sleeper Buses ,the Passenger Count must be less than or equal to the Capacity of the Bus	✓	Mahammad Tahir T, 6/23/2025, 2:53 AM

The status bar at the bottom shows it's 30°C, partly sunny, and the date is 29-06-2025.

📁 Milestone 8: Flow Setup

- Designed a **record-triggered flow**:
 - Automatically fetches fare based on route + model.
 - Improved automation and reduced manual input errors.
-

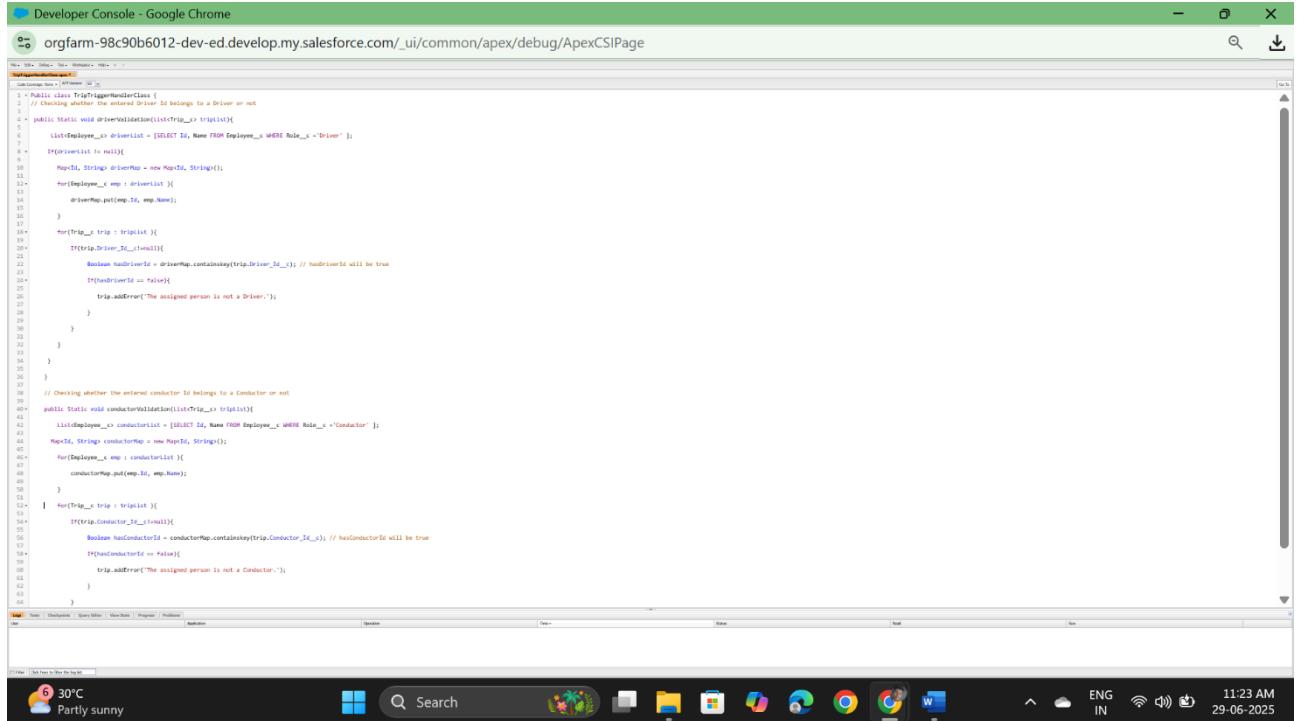
OUTPUT SCREENSHOT



Milestone 9: Apex Trigger

- Developed a trigger to validate:
 - Only users with role “Driver” can be added as Driver.
 - Only “Conductor” can be added as Conductor.
- Ensured business logic is enforced at data level.

OUTPUT SCREENSHOT



The screenshot shows the Salesforce Developer Console in Google Chrome. The URL is http://orgfarm-98c90b6012-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage. The code editor displays two Apex classes:

```
1 // Public class to check if Driver/Conductor Id belongs to a Driver or not
2
3 // Checks whether the entered Driver Id belongs to a Driver or not
4
5 public static void driverValidation(List<Trip> tripList){
6
7     List<Employee__c> driverList = [SELECT Id, Name FROM Employee__c WHERE Role__c = 'Driver'];
8
9     Map<String, String> driverMap = new Map<String, String>();
10
11    for(Employee__c emp : driverList) {
12
13        driverMap.put(emp.Id, emp.Name);
14
15    }
16
17    for(Trip__c trip : tripList) {
18
19        if(trip.Driver_Id__c != null) {
20
21            Boolean hasDriverId = driverMap.containsKey(trip.Driver_Id__c); // hasDriverId will be true
22
23            if(hasDriverId == false) {
24
25                trip.addError('The assigned person is not a Driver.');
26
27            }
28
29        }
30
31    }
32
33 }
34
35
36
37 // Checking whether the entered conductor Id belongs to a Conductor or not
38
39 public static void conductorValidation(List<Trip> tripList){
40
41     List<Employee__c> conductorList = [SELECT Id, Name FROM Employee__c WHERE Role__c = 'Conductor'];
42
43     Map<String, String> conductorMap = new Map<String, String>();
44
45    for(Employee__c emp : conductorList) {
46
47        conductorMap.put(emp.Id, emp.Name);
48
49    }
50
51
52    for(Trip__c trip : tripList) {
53
54        if(trip.Conductor_Id__c != null) {
55
56            Boolean hasConductorId = conductorMap.containsKey(trip.Conductor_Id__c); // hasConductorId will be true
57
58            if(hasConductorId == false) {
59
60                trip.addError('The assigned person is not a Conductor.');
61
62            }
63
64        }
65
66    }
67 }
```

The status bar at the bottom shows the date and time as 29-06-2025, 11:23 AM.

Developer Console - Google Chrome

orgfarm-98c90b6012-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage

TripTriggerHandlerClass.apcpx * TripTrigger.apxt

Code Coverage: None API Version: 64 Go To

```
1 * trigger TripTrigger on Trip__c (before insert, before update) {
2
3
4
5 *     if(trigger.isBefore){
6
7 *         if(trigger.isInsert || trigger.isUpdate){
8
9 *             // Validating the Conductor Id in Trip is really a Conductor or not
10
11             TripTriggerHandlerClass.driverValidation(trigger.new);
12
13
14
15             // Validating the Conductor Id in Trip is really a Conductor or not
16
17             TripTriggerHandlerClass.conductorValidation(trigger.new);
18
19     }
20
21 }
22 }
```

Logs Tests Checkpoints Query Editor View State Progress Problems

User Application Operation Time Status Read Size

Filter Click here to filter the log list.

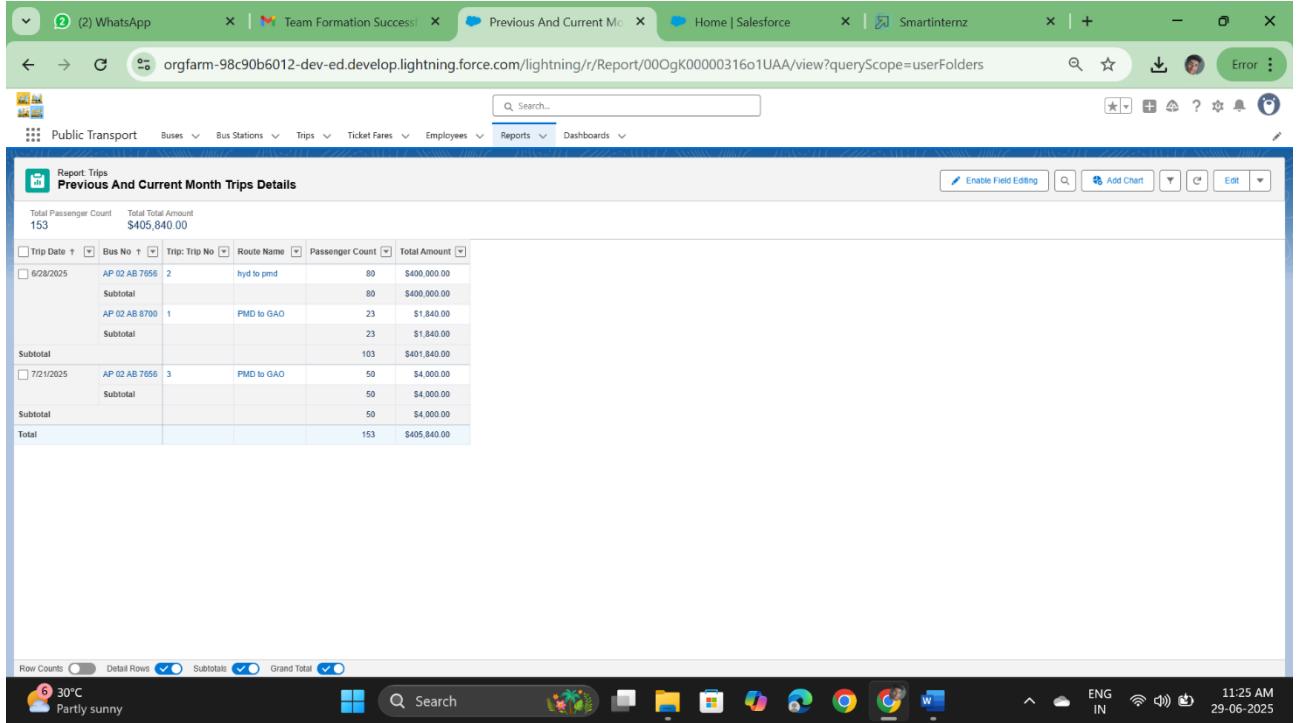
6 30°C Partly sunny Search

Cloud ENG IN 11:23 AM 29-06-2025

Milestone 10: Reports

-
- Created reports for:
 - Employees per station, monthly revenue, trip counts.
 - Used grouping, filters, and field summaries.
 - Enabled data-driven decision-making.
-

OUTPUT SCREENSHOT



The screenshot shows a Salesforce Lightning report titled "Report: Trips Previous And Current Month Trips Details". The report displays trip data for Public Transport, specifically for trips made on 6/28/2025 and 7/21/2025. The data includes trip date, bus number, trip ID, route name, passenger count, and total amount. The total passenger count is 153, and the total amount is \$405,840.00.

Trip Date	Bus No	Trip ID	Route Name	Passenger Count	Total Amount
6/28/2025	AP 02 AB 7056	2	hyd to pmrd	80	\$400,000.00
			Subtotal	80	\$400,000.00
	AP 02 AB 8700	1	PMD to GAO	23	\$1,840.00
			Subtotal	23	\$1,840.00
			Subtotal	103	\$401,840.00
7/21/2025	AP 02 AB 7056	3	PMD to GAO	50	\$4,000.00
			Subtotal	50	\$4,000.00
			Subtotal	50	\$4,000.00
			Total	153	\$405,840.00

At the bottom of the screenshot, the Windows taskbar shows the date as 29-06-2025 and the time as 11:25 AM.

Screenshot of a web browser showing a Salesforce report titled "Employees By Bus Station".

The report displays 4 total records:

Bus Station Name	Employee ID	Employee Name	Role
- (2)	a04gK000000VqD	T Muhammad Tahir	Conductor
	a04gK000000vKRZ	akash	Driver
Subtotal			
ATS Travels (1)	a04gK000000XGar	ANJI	Driver
Subtotal			
JSR travels (1)	a04gK000000XGCT	NOOR	Conductor
Subtotal			
Total (4)			

Report navigation and settings at the bottom:

- Row Counts: On
- Detail Rows: On
- Subtotals: On
- Grand Total: On

System status bar at the bottom:

- 30°C
- Partly sunny
- Search icon
- Cloud icon
- ENG IN
- Wi-Fi icon
- Battery icon
- 11:26 AM
- 29-06-2025

Screenshot of a web browser showing a Salesforce report titled "Drivers And Conductors Information".

The report displays 4 total records:

Bus Station Name	Role	Employee Name	Employee ID
JSR travels (1)	Conductor	NOOR	a04gK000000XGCT
Subtotal			
ATS Travels (1)	Driver	ANJI	a04gK000000XGar
Subtotal			
- (2)	Conductor	T Muhammad Tahir	a04gK000000VqD
	Driver	akash	a04gK000000vKRZ
Subtotal			
Total (4)			

Report navigation and settings at the bottom:

- Row Counts: On
- Detail Rows: On
- Subtotals: On
- Grand Total: On

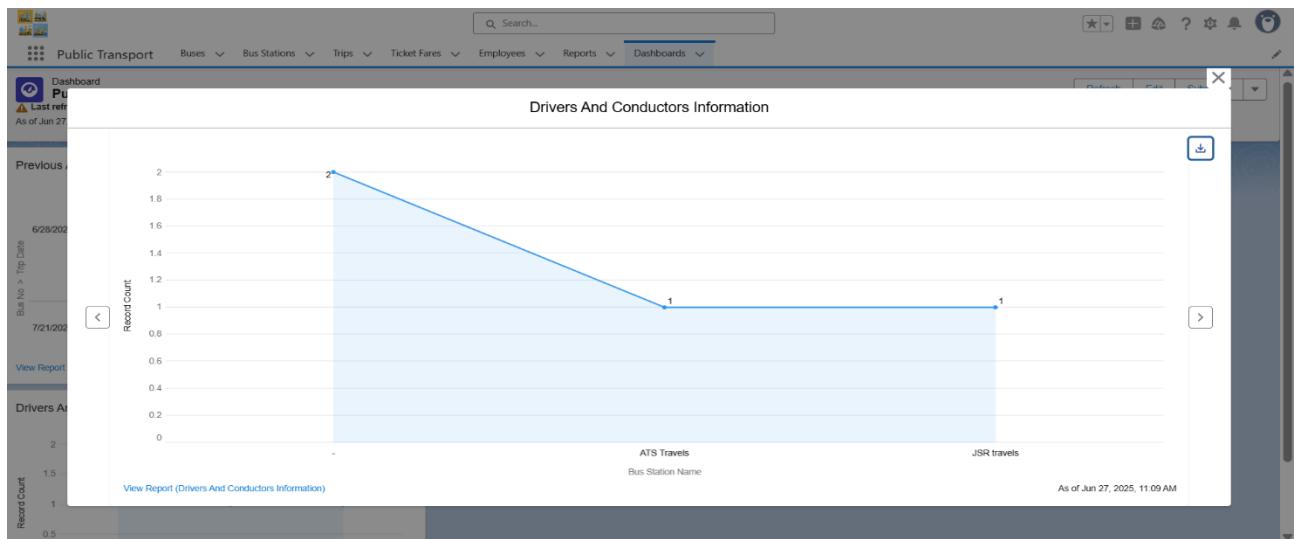
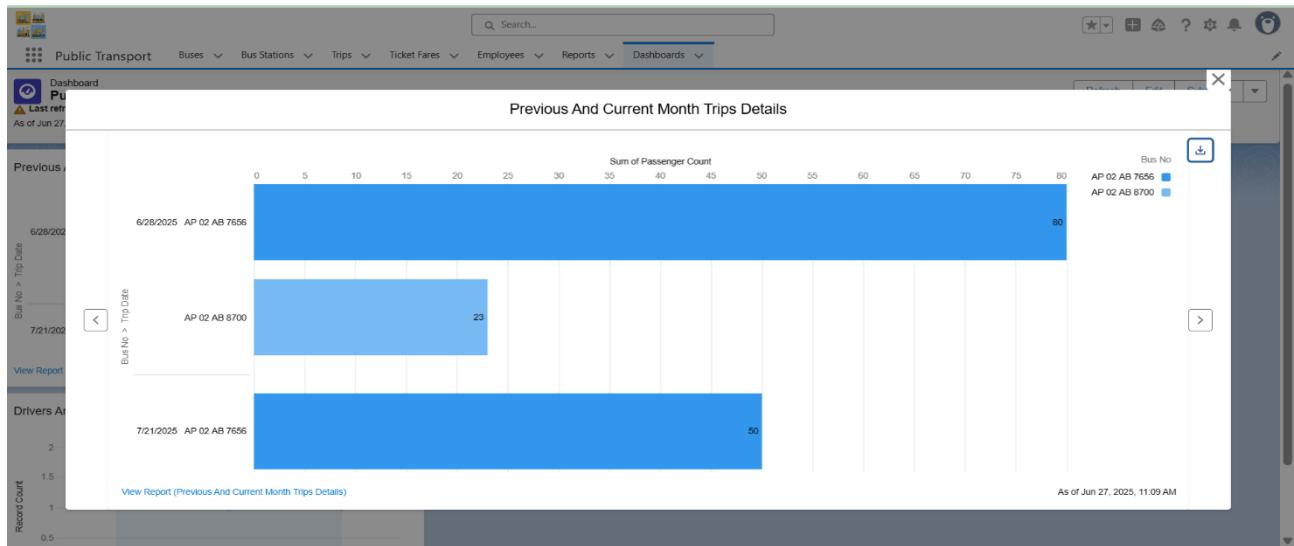
System status bar at the bottom:

- 30°C
- Partly sunny
- Search icon
- Cloud icon
- ENG IN
- Wi-Fi icon
- Battery icon
- 11:26 AM
- 29-06-2025

Milestone 11: Dashboards

- Designed dashboards showing:
 - Revenue trends, trip performance, and role summary.
- Used bar, pie, and summary widgets for visualization.

OUTPUT SCREENSHOT



Milestone 12: Final Review

- Conducted full system testing.
 - Checked all object links, flows, triggers, reports.
 - Confirmed the CRM is fully functional and stable.
-

6.2 Dataset

The system was tested with the following types of data:(Sample Records and Input Values Used During Testing)

Object	Sample Fields Used
Bus Station	Name: Visakhapatnam Central, Category: Managed
Bus	Model: Super Deluxe, Capacity: 45, Station: Visakhapatnam
Employee	Name: Rakesh, Role: Driver, Phone: 9876543210, DOB: 1980-06-14
Ticket Fare	Route: Hyderabad-Warangal (Express), Fare: ₹220
Trip	Route: Hyderabad-Warangal, Passengers: 34, Estimated Time: 4 hrs

 Note: All test records were created using Salesforce's UI and validated via flows and formula fields

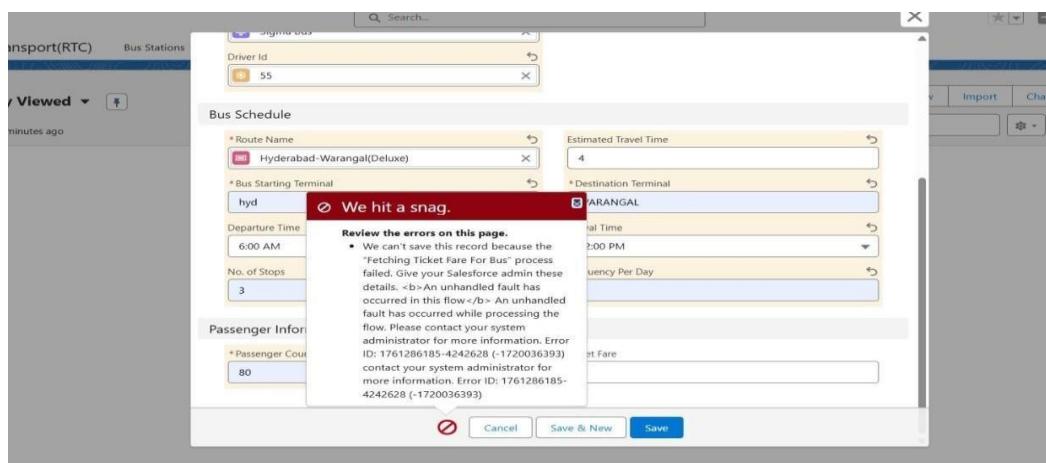
7. FUNCTIONAL AND PERFORMANCE TESTING

7.1 Performance Testing

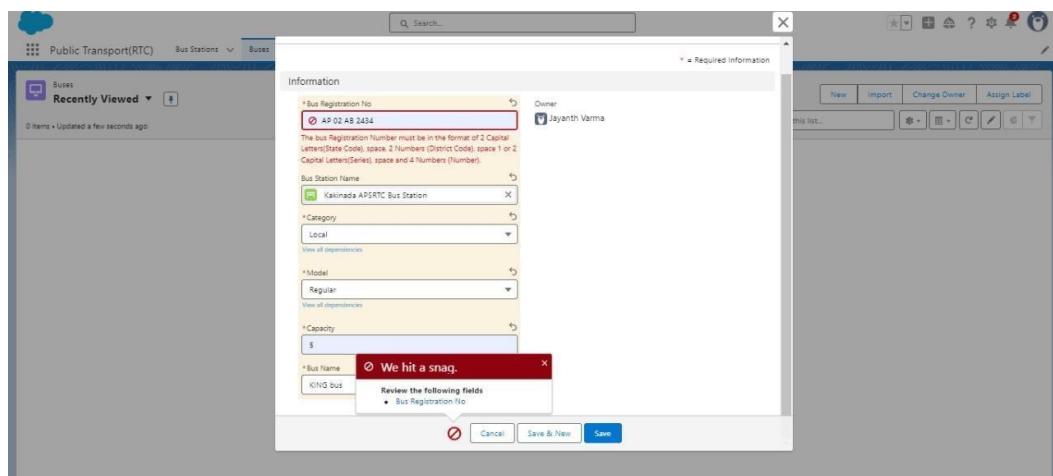
The system was tested using realistic data entries.

Key validations included:

- Trigger for role validation (Driver/Conductor)
- Flow for auto-fetching ticket fare
- Field validation for formats (e.g., registration number)
Screenshots demonstrating these were captured during testing phases.
- Flow Automation Error (Fare Fetch Failure)



- Validation Rule Error (Bus Registration Format)



8. RESULTS

8.1 Output Screenshots

Key functional screenshots include:

- Correctly filled record (Trip, Fare, etc.)

The screenshot shows a Salesforce Lightning interface for a 'Ticket Fare' record. The record details are as follows:

Field	Value
Route Name	hyd to pmd
Bus Model	Regular
Ticket Fare	\$5,000.00
Created By	Mohammad Tahir T, 6/24/2025, 5:10 AM
Owner	Mohammad Tahir T
Last Modified By	Mohammad Tahir T, 6/24/2025, 5:10 AM

The browser taskbar at the bottom shows various open tabs and system status.

- Final report previews with grouped data

The screenshot shows a Salesforce Lightning report titled 'Report: Employees Drivers And Conductors Information'. The report displays the following data:

Role	Employee Name	Employee ID
Conductor	NOOR	a04gK00000XGCT
Driver	ANJLI	a04gK00000XGAr
Conductor	T Mohammad Tahir	a04gK00000VYqD
Driver	akash	a04gK00000VkrZ

The browser taskbar at the bottom shows various open tabs and system status.

- Trigger error and fare auto-fill during Trip creation

New Trip

* = Required Information

Information

- * Trip No: 35
- Conductor Id: 16
- Owner: Bharath Adithya
- Driver Id: 3

Bus Schedule

- * Route Name: Hyderabad-Warangal(Deluxe)

Cancel Save & New Save

- Validation error on form submission

We hit a snag.

Review the errors on this page.

- We can't save this record because the "Fetching Ticket Fare For Bus" process failed. Give your Salesforce admin these details.

An unhandled fault has occurred in this flow

An unhandled fault has occurred while processing the flow. Please contact your system administrator for more information. Error ID: 1761286185-4242628 (-1720036393)

contact your system administrator for more information. Error ID: 1761286185-4242628 (-1720036393)

Cancel Save & New Save

9. ADVANTAGES & DISADVANTAGES

Advantages

- Real-time data validation and automation
- Modular structure allows easy customization
- Centralized and streamlined process for trip, fare, and employee management
- Visual reports and dashboards for decision-making

Disadvantages

- Requires knowledge of Salesforce for advanced use
- Limited functionality in offline mode due to cloud dependency
- Dependent on correctly established object relationships

10. CONCLUSION

The project successfully delivered a tailored Salesforce CRM application for RTC operations. It addressed key inefficiencies and brought structure to transport management workflows. The solution also served as a valuable learning experience for the team in both declarative and programmatic Salesforce development.

11. FUTURE SCOPE

The CRM can be extended further with features such as:

- Third-party ticketing and payment system integration
 - Automated alerts and communication features
 - Mobile-first user interface with Salesforce Experience Cloud
 - AI-powered forecasting using Salesforce Einstein
-

12. APPENDIX

- Source Code: Attach Apex Trigger, Flow logic files



A screenshot of the Salesforce Apex code editor. The title bar shows the URL: orgfarm-ed7725a0e4-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage. The tab bar has two tabs: 'TripTriggerHandlerClass.apsc' and 'TripTrigger.apxt'. The 'TripTrigger.apxt' tab is active. The code editor displays the following Apex trigger code:

```
1 trigger TripTrigger on Trip__c (before insert, before update) {
2
3
4
5 if(trigger.isBefore){
6
7 if(trigger.isInsert || trigger.isUpdate){
8
9     // Validating the Conductor Id in Trip is really a Conductor or not
10    TripTriggerHandlerClass.driverValidation(trigger.new);
11
12
13
14    // Validating the Conductor Id in Trip is really a Conductor or not
15    TripTriggerHandlerClass.conductorValidation(trigger.new);
16
17
18
19 }
20
21 }
22 }
```

- GitHub Link:

<https://github.com/MahammadTahir/CMR-Public-Transport-System-Management>
