

What next vision motors shaping the future of mobility with innovation and excellence

Abstract:

This project explores how What Next Vision Motors leverages Salesforce to drive innovation, enhance customer experiences, and lead the transformation of the mobility sector. As the automotive industry shifts toward electrification, digitalization, and sustainability, What Next Vision Motors integrates Salesforce solutions—such as Sales Cloud, Service Cloud, and Experience Cloud—to streamline operations, personalize customer engagement, and enable data-driven decision-making. The company uses Salesforce CRM to manage leads, track service interactions, and build long-term customer loyalty, while automation tools improve efficiency across sales and service channels. Through this implementation, the project demonstrates how cloud-based technologies empower automotive companies to stay agile, customer-focused, and innovative—ultimately shaping the future of mobility with excellence.

Objective:

- To explore how Salesforce CRM can support digital transformation in the automotive industry.
- To demonstrate the use of Salesforce tools (e.g., Sales Cloud, Service Cloud, Flow) in streamlining operations at What Next Vision Motors.
- To improve sales and service efficiency using automation and real-time analytics.
- To align technology solutions with the company's mission of innovation, sustainability, and mobility excellence.

- To develop practical knowledge of Salesforce platform capabilities through hands-on project implementation.

Technology description:

1. Custom Objects & Data Modelling

Custom objects were created to represent different business entities.

- Vehicle__c: Vehicle details like model, quantity, price.
- Vehicle_Dealer__c: Dealer information including contact details.
- Vehicle_Customer__c: Customer profiles.
- Vehicle_Order__c: Tracks orders with relationships to both Vehicle and Customer.
- Vehicle_Test_Drive__c: Manages test drive bookings.
- Vehicle_Service_Request__c: Handles servicing requests.

Each object includes key fields such as picklists, currency, auto-number, lookup fields, and text entries to capture relevant business data.

2. Lightning App Development

- A custom Lightning app named WhatNext Vision Motors was built using:
- App Manager to create and configure the app.
- Navigation Items to include all relevant custom tabs (Vehicle, Order, Dealer, Customer, etc.).
- User Profiles assigned to control visibility and access.

3. Object Tab Creation

- Custom object tabs were added for each major entity:
- Created via Setup → Tabs → New Custom Object Tab
- Enabled direct access to object records within the app.

4. Flow Automation (No-Code Logic)

- A Record-Triggered Flow was built to automate dealer assignment:
- Trigger: New order creation.
- Actions: Get Vehicle Customer record. Find nearest dealer based on location.
- Update the order record with the assigned dealer.
- Activated via Salesforce Flow Builder.

5. Apex Batch Class (Advanced Processing)

- An Apex Batch Class was proposed for high-volume updates:
- Example Use Case: Auto-update Vehicle_Service_Request__c status to “Completed” if overdue.
- Allows scheduled background execution and processing large datasets asynchronously.

6. Email Notifications

- Emails can be triggered through either:
- Flow Email Alerts: Declaratively send notifications post-dealer assignment.
- Apex Email Class: For conditional or scheduled emailing logic.

Benefits of the Solution:

- Real-time dealer assignment using automation.
- Centralized vehicle and customer data model.
- Seamless UI using Lightning App framework.
- Scalable backend via Apex batch processing.
- Ready-to-use solution for automotive CRM use cases.

Detailed execution and project phase:

Environment Setup – Salesforce Developer Account:

Objective:

To set up a personal Salesforce Developer Org for building and testing the WhatNext Motors application.

Steps Followed:

1. Go to <https://developer.salesforce.com/signup>
2. Filled the registration form with the following:

First Name & Last Name

Email Address

Role: Developer

Company Name / College Name

Country: India

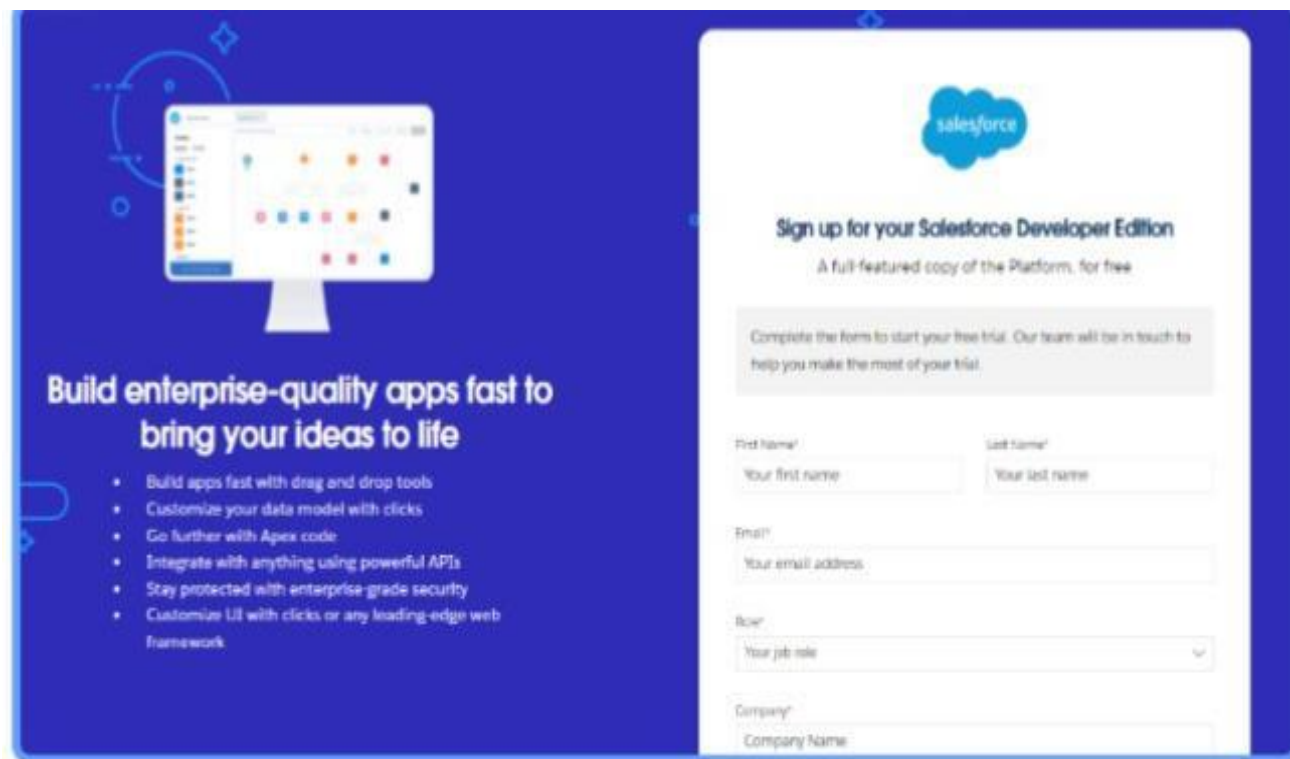
Postal Code

Username (unique email-style format, e.g., yourname@whatnextvisionmotors.com)

3. Clicked Sign Me Up.
4. Verified email via confirmation link to activate the Developer Org.

Result:

Successfully created a Salesforce Developer Org with access to all necessary tools: App Builder, Object Manager, Flow Builder, Apex Class Editor, etc.



Phase 1: Planning & Requirement Analysis

Objective: To build a CRM-based application for WhatNext Vision Motors that digitizes and automates vehicle sales, test drives, and service operations using Salesforce.

Key Activities:

Understanding business use cases: vehicle inventory, customer records, service tracking.

Identifying the required custom objects and relationships.

Planning automation flow logic for dealer assignment.

Phase 2: Data Modeling – Objects & Relationships

Goal:

Define the data structure for the app using custom objects and fields.

Key Custom Objects Created:

Object Name Description

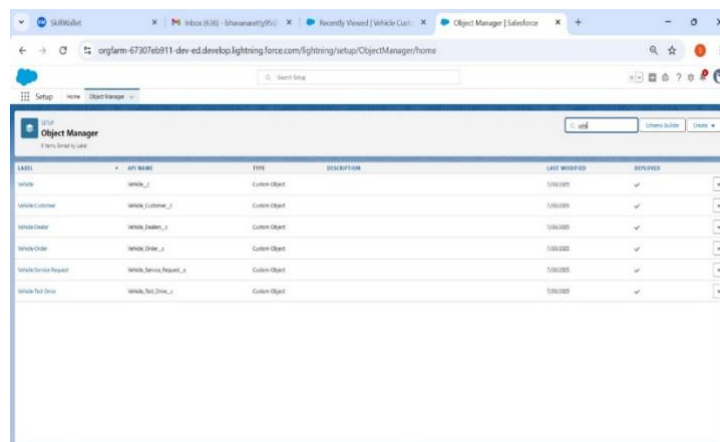
- Vehicle__c Stores vehicle details
- Vehicle_Dealer__c Stores dealer information
- Vehicle_Customer__c Stores customer profiles
- Vehicle_Order__c Tracks customer orders
- Vehicle_Test_Drive__c Tracks test drive bookings
- Vehicle_Service_Request__c Tracks vehicle servicing

Relationships:

Lookup relationships were established:

Vehicle_Order__c links to both Customer__c and Vehicle__c

Vehicle_Service_Request__c and Vehicle_Test_Drive__c link to both Vehicle__c and Customer__c



The screenshot shows the Salesforce Object Manager interface. The table lists the following custom objects:

NAME	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DELETED
Vehicle	Vehicle__c	Custom Object		7/26/2023	✓
Vehicle Customer	Vehicle_Customer__c	Custom Object		7/26/2023	✓
Vehicle Dealer	Vehicle_Dealer__c	Custom Object		7/26/2023	✓
Vehicle Order	Vehicle_Order__c	Custom Object		7/26/2023	✓
Vehicle Service Request	Vehicle_Service_Request__c	Custom Object		7/26/2023	✓
Vehicle Test Drive	Vehicle_Test_Drive__c	Custom Object		7/26/2023	✓

Phase 3: App Development – Lightning App & Tabs

Goal:

Create a custom Lightning App to access and manage all objects.

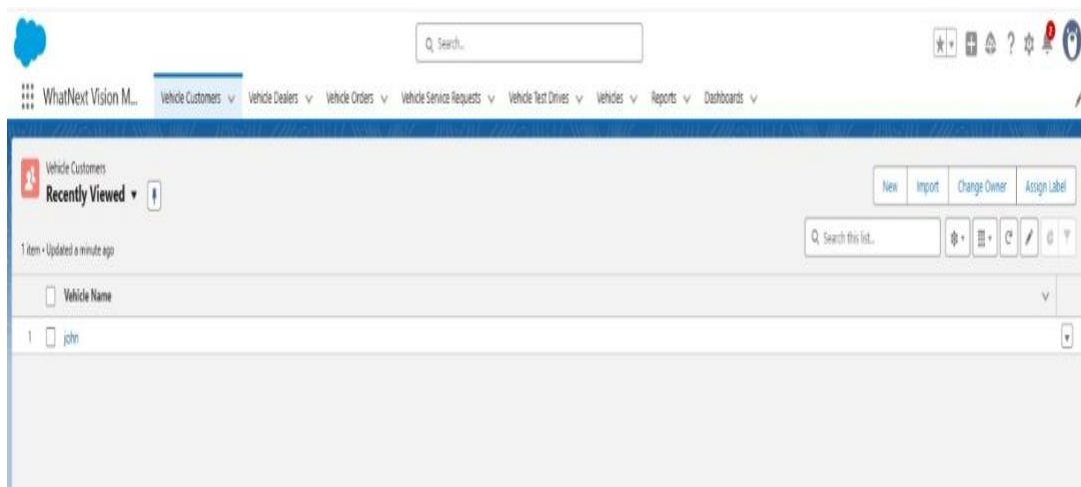
Steps Executed:

Navigate to App Manager → Create new app WhatNext Vision Motors

Add navigation items: Vehicle, Dealer, Customer, Order, Test Drive, Service Request

Configure branding, app icon, and profiles

Create Custom Tabs for all objects via Setup → Tabs → Custom Object Tabs



Phase 4: Flow Automation – Record Triggered Flow

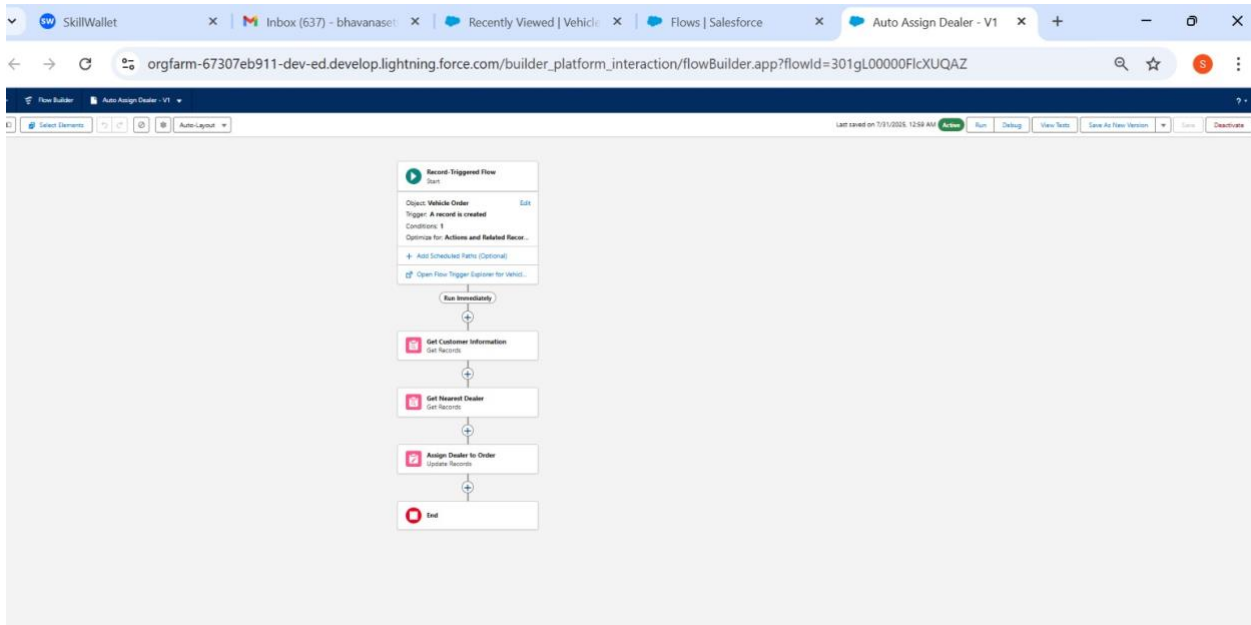
Goal:

Automatically assign the nearest dealer to the customer when an order is placed.

Steps in Flow:

1. Trigger: On creation of a Vehicle_Order__c record.
2. Get Records: Fetch customer details from Vehicle_Customer__c
3. Get Nearest Dealer: (Using logic or distance criteria; here assumed static demo logic)
4. Update Records: Assign the dealer to the order.
5. Save & Activate the flow.

Tools Used: Flow Builder (Low-code automation)



Phase 5: Email Notification (Optional)

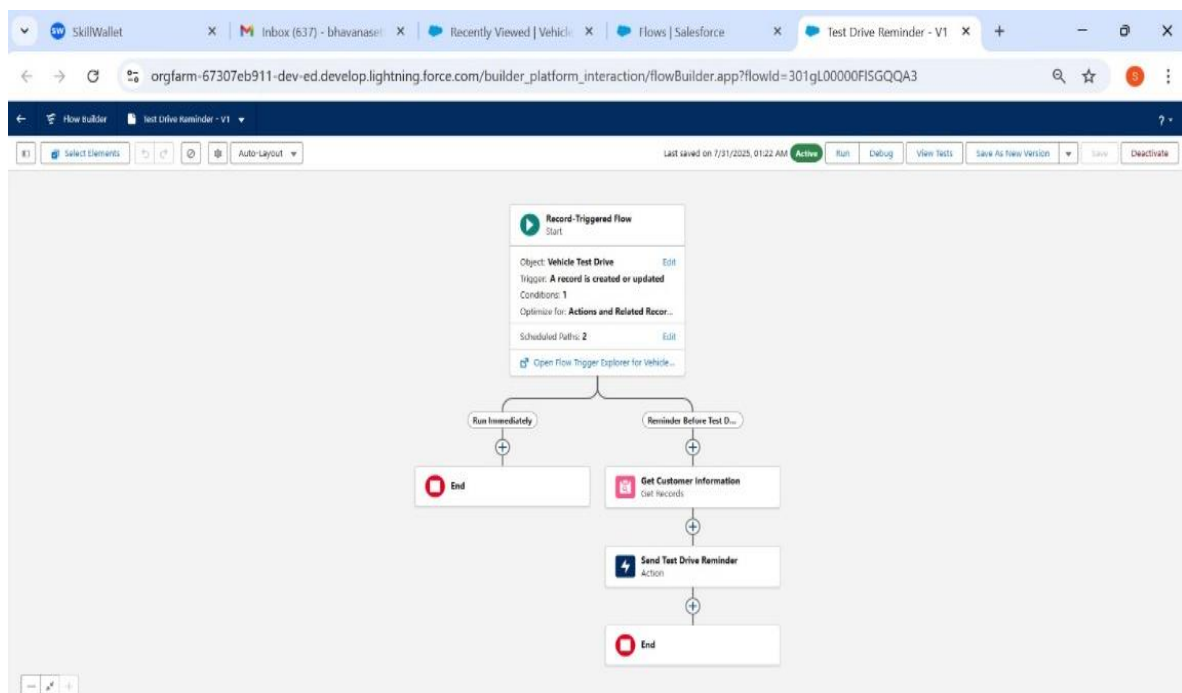
Goal:

Send an email to customers confirming test drive/service or order placement.

Two options implemented:

Using Flow Email Alerts: Configure an Email Alert in Flow.

Using Apex: Use `Messaging.sendEmail()` method (if conditional logic is needed).

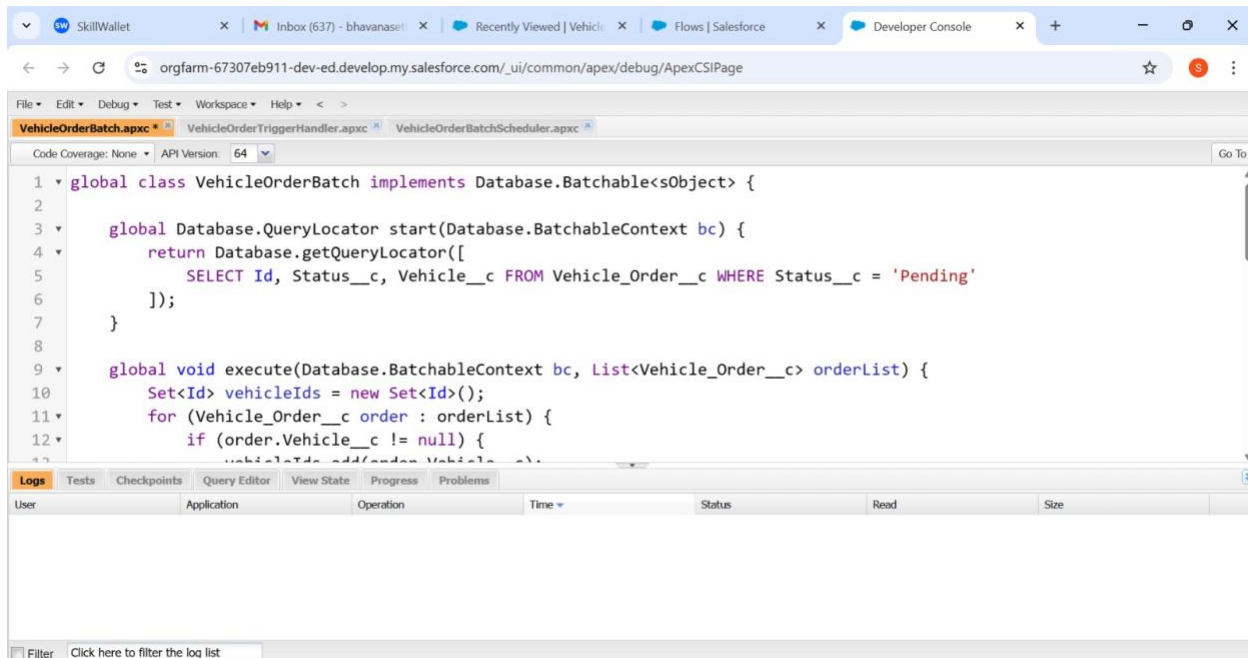


Phase 6: Apex Batch Class (Advanced / Optional)

Goal:

Handle large-volume data updates automatically

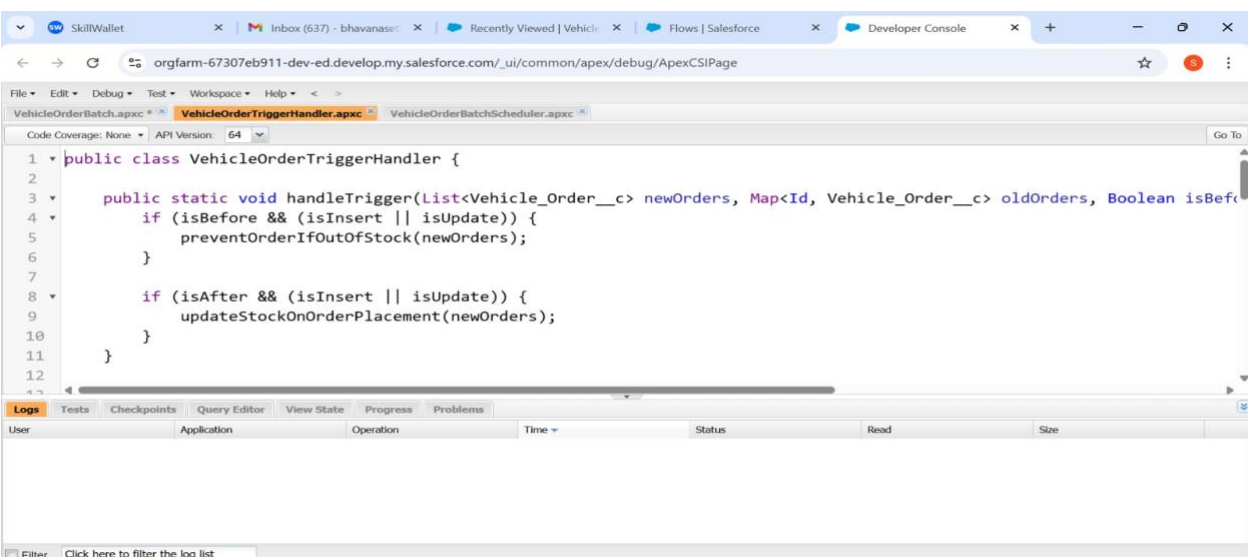
Example Use Case: Automatically mark overdue service requests as “Completed”



The screenshot shows the Salesforce Developer Console with the Apex class `VehicleOrderBatch.apxc` open. The class implements the `Database.Batchable<Object>` interface. The `start` method returns a `Database.QueryLocator` for a query that selects records from `Vehicle_Order__c` where `Status__c` is 'Pending'. The `execute` method iterates through a list of `Vehicle_Order__c` records and updates their status to 'Completed'.

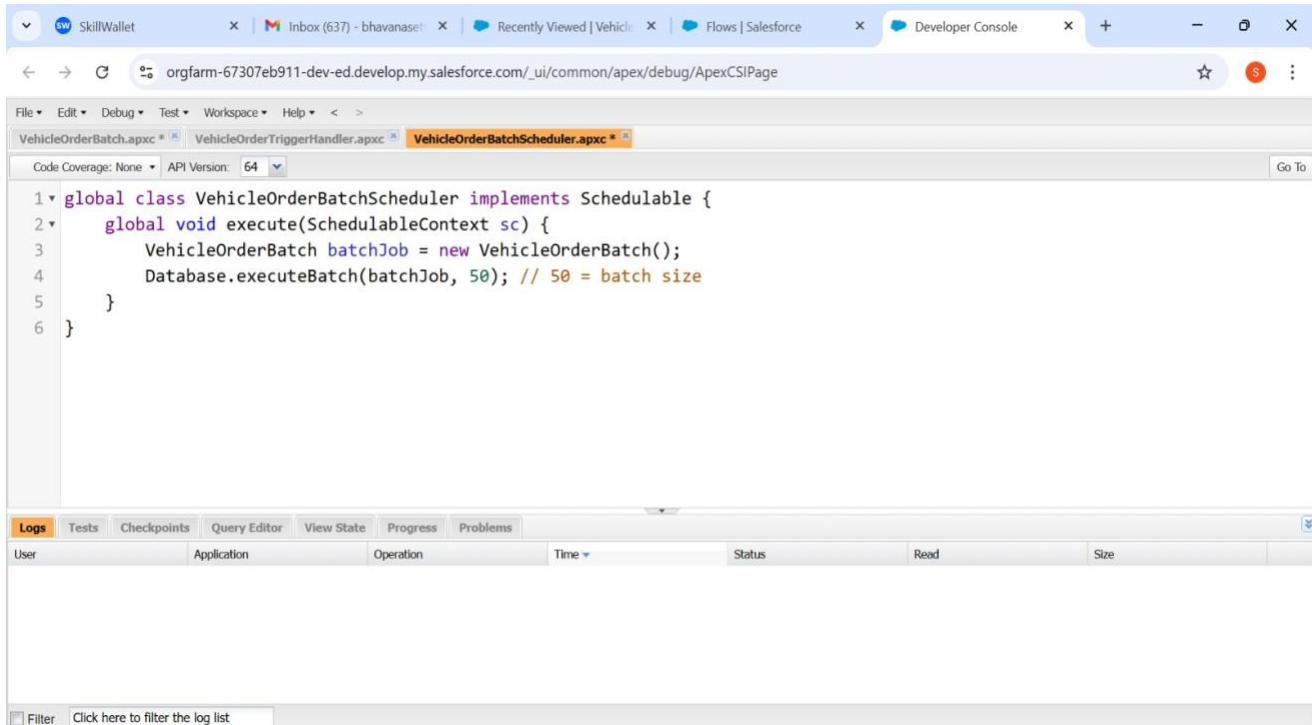
```
1 global class VehicleOrderBatch implements Database.Batchable<Object> {
2
3     global Database.QueryLocator start(Database.BatchableContext bc) {
4         return Database.getQueryLocator([
5             SELECT Id, Status__c, Vehicle__c FROM Vehicle_Order__c WHERE Status__c = 'Pending'
6         ]);
7     }
8
9     global void execute(Database.BatchableContext bc, List<Vehicle_Order__c> orderList) {
10         Set<Id> vehicleIds = new Set<Id>();
11         for (Vehicle_Order__c order : orderList) {
12             if (order.Vehicle__c != null) {
13                 vehicleIds.add(order.Vehicle__c);
14             }
15         }
16         update Vehicle__c where Id in :vehicleIds;
17     }
18 }
```

Apex Batch Class Created:



The screenshot shows the Salesforce Developer Console with the Apex class `VehicleOrderTriggerHandler.apxc` open. The class implements the `TriggerHandler` interface. The `handleTrigger` method is a static method that takes a list of `Vehicle_Order__c` records, a map of `Id` to `Vehicle_Order__c` records, and a boolean `isBefore` as input. It calls `preventOrderIfOutOfStock` if `isBefore` is true, and `updateStockOnOrderPlacement` if `isBefore` is false.

```
1 public class VehicleOrderTriggerHandler {
2
3     public static void handleTrigger(List<Vehicle_Order__c> newOrders, Map<Id, Vehicle_Order__c> oldOrders, Boolean isBefore) {
4         if (isBefore && (isInsert || isUpdate)) {
5             preventOrderIfOutOfStock(newOrders);
6         }
7
8         if (isAfter && (isInsert || isUpdate)) {
9             updateStockOnOrderPlacement(newOrders);
10        }
11    }
12 }
```

Phase 7: Testing & Validation

Goal:

Ensure end-to-end functionality of the app, automation, and backend logic.

Testing Activities:

Created test data for vehicles, customers, orders, and service requests.

Verified:

Record-triggered flow assigns dealers correctly.

Email notifications are sent (if configured). Apex batch updates records in bulk without errors.

Phase 8: Deployment & Presentation

Goal:

Submit and present the working Salesforce app and flow logic.

Final Deliverables:

Lightning App with custom navigation

All custom objects and relationships

Active automation flow

Optional Apex batch and email logic

Screenshots for each configuration



Reminder: Your Test Drive is Tomorrow

Spam



setty bhav... 12:31 pm
to me ▾



Why is this message in spam?

It is similar to messages that were identified as spam in the past.

Report not spam



Dear userjohn,

This is a remainder that your test drive
a04gL000008DmqIQAS is tommorrow. if you need to
reschedule please contact us at support@gmail.com

Thank you



Reply



Reply
all



Forwa
rd



99+



Project explanation with real world example:

To design a Salesforce-based CRM application that helps an automotive company (WhatNext Vision Motors) manage customers, vehicle inventory, dealers, test drives, orders, and service requests—just like real companies such as Tata Motors, Hyundai, or Tesla.

Real-World Example: Tata Motors CRM System:

- Let's say Tata Motors launches a new electric vehicle (EV), and they want to:
- Track each vehicle and its specifications
- Record every dealership that sells the vehicle
- Let customers book test drives and place orders online
- Assign the nearest dealership automatically
- Schedule service and track customer satisfaction
- Your project simulates this exact setup using Salesforce CRM tools.

Modules Implemented and Real-World Mapping:

Module Your Salesforce ObjectReal-World Example

- Vehicle Inventory **Vehicle__c** Like Tata stores Nexon EV specs, price, and features
- Dealers **Vehicle_Dealer__c** Like Tata showrooms in Hyderabad, Delhi, etc.
- Customers **Vehicle_Customer__c** Tata customer info: name, address, contact
- Orders **Vehicle_Order__c** When a customer books a Nexon EV, order record created
- Test Drives **Vehicle_Test_Drive__c** Booking test drives via Tata's official website
- Service Requests **Vehicle_Service_Request__c** Customer raises service tickets for issues
- Dealer Auto-Assignment **Salesforce Flow** Like system assigning nearest showroom when customer books
- Bulk Record Updates **Apex Batch Class** Automatically marking overdue service tickets as complete
- Email Notification **Flow Email Alert** Customer gets email after test drive booking or order confirmation

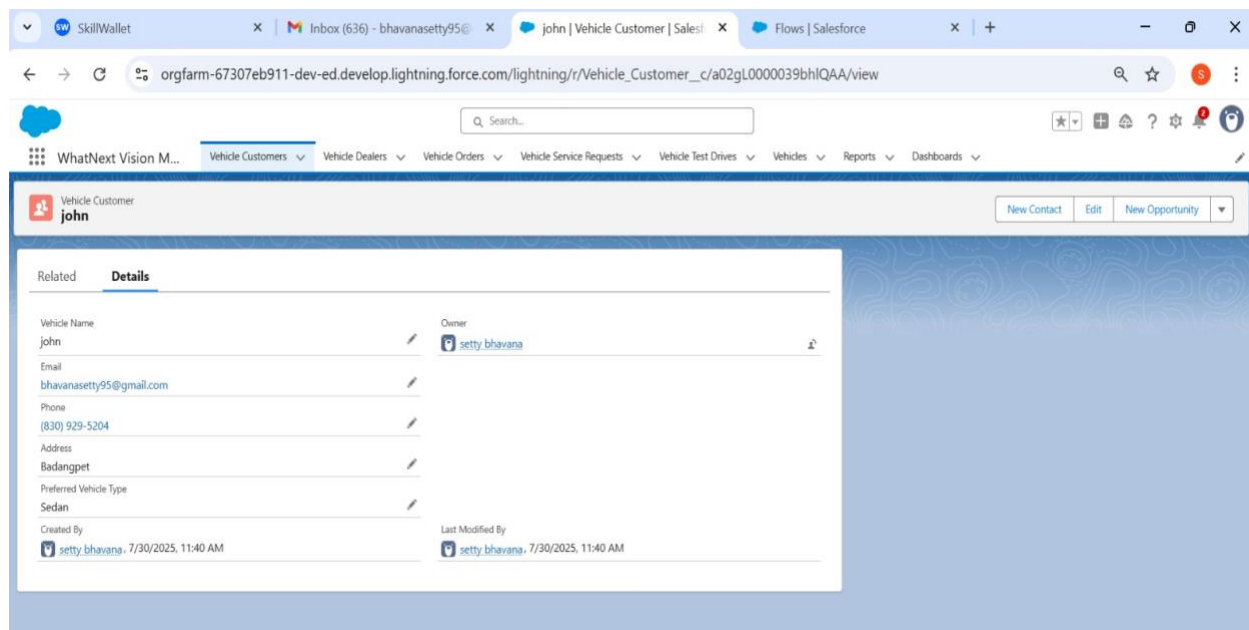
Real-World Impact and Use:

- Without Salesforce:
- Manual data entry
- Delayed dealer follow-ups

- No automatic test drive scheduling
- No centralized vehicle/service tracking
- With Salesforce (as built in your project):
- All information stored in one platform
- Test drives/orders assigned automatically
- Dealers get notified instantly
- Customers receive timely emails
- Admins can manage everything efficiently

Why This Matters in the Industry:

- Major companies like Hyundai, Ford, and Mahindra use CRM tools like Salesforce to:
- Boost sales efficiency
- Track customer satisfaction
- Automate servicing
- Analyze data for better decision-making
- Your project mimics these real-world processes in a simulated business environment, giving you practical CRM and automation experience.



Browser tabs: SkillWallet, Inbox (636) - bhavanasetty95@, Recently Viewed | Vehicle Order, Flows | Salesforce

Address bar: orgfarm-67307eb911-dev-ed.develop.lightning.force.com/lightning/o/Vehicle_Order_c/list?filterName=__Recent

Search bar: Search...

Navigation menu: WhatNext Vision M..., Vehicle Customers, Vehicle Dealers, Vehicle Orders, Vehicle Service Requests, Vehicle Test Drives, Vehicles, Reports, Dashboards

Vehicle Orders Recently Viewed

2 items • Updated a few seconds ago

Buttons: New, Import, Change Owner, Assign Label

Search this list...

	Vehicle Order Number	
1	O-0002	
2	O-0001	

Browser tabs: SkillWallet, Inbox (636) - bhavanasetty95@, Recently Viewed | Vehicle Test, Flows | Salesforce

Address bar: orgfarm-67307eb911-dev-ed.develop.lightning.force.com/lightning/o/Vehicle_Test_Drive_c/list?filterName=__Recent

Search bar: Search...

Navigation menu: WhatNext Vision M..., Vehicle Customers, Vehicle Dealers, Vehicle Orders, Vehicle Service Requests, Vehicle Test Drives, Vehicles, Reports, Dashboards

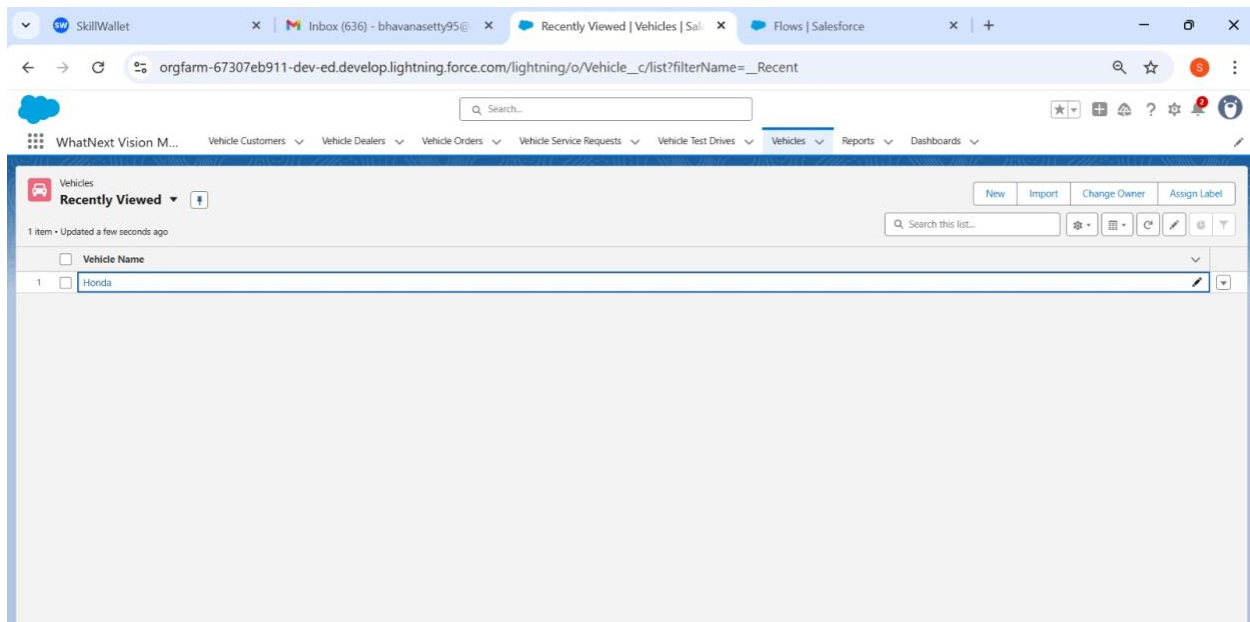
Vehicle Test Drives Recently Viewed

2 items • Updated a few seconds ago

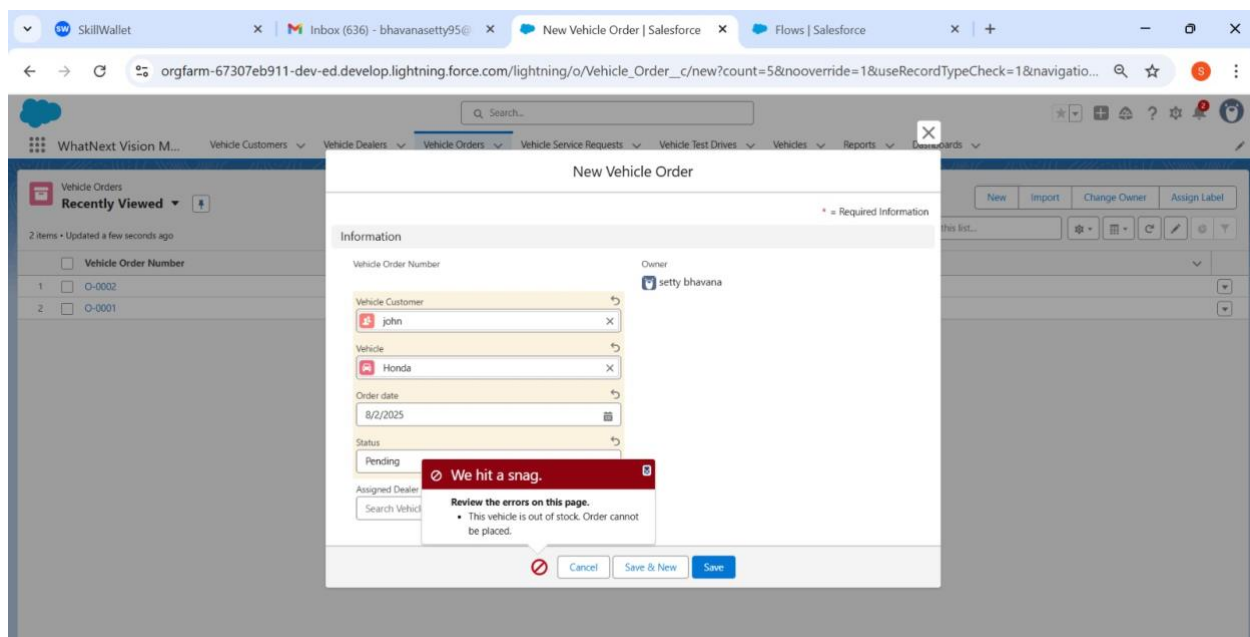
Buttons: New, Import, Change Owner, Assign Label

Search this list...

	Vehicle Name	
1	ABC	
2	XYZ	



If it's out of stock:



Conclusion:

The WhatNext Vision Motors project successfully demonstrates how Salesforce can be used to digitally transform automotive business operations by centralizing customer data, automating key processes, and enhancing customer experiences.

Through the implementation of custom objects, Lightning App, automated flows, and Apex logic, this project replicates the real-world functioning of a modern automotive CRM system—similar to how companies like Tata Motors, Tesla, or Hyundai manage vehicle sales, service, and customer relationships.

Key outcomes of the project include:

- Seamless vehicle, dealer, and customer data management
- Automated dealer assignment and service tracking
- Real-time email notifications for better communication
- Scalable logic using Apex batch processing

This hands-on project not only enhanced understanding of Salesforce tools like Flow Builder, Object Manager, and Apex Classes, but also built real-world skills in CRM, automation, and business application development.

In conclusion, the project reflects the future of mobility, where innovation, technology, and customer-centric solutions drive excellence in the automotive sector.