

Smart Automotive Order Management System

Project Overview:

The **Smart Automotive Order Management System** is a Salesforce CRM-based solution designed to enhance the customer experience and operational efficiency in the automotive sector. It integrates key business components such as vehicle inventory, dealer management, and customer ordering into a unified Salesforce ecosystem.

This project automates critical workflows — such as **dealer assignment**, **stock validation**, and **order tracking** — using **Apex Triggers**, **Flows**, and **Batch Apex**. Orders for unavailable vehicles are automatically restricted, while valid orders are routed to the nearest dealer based on customer location.

Additionally, the system includes **scheduled Apex jobs** to update order statuses and send **email notifications** for test drives and stock updates. Together, these automations ensure data accuracy, minimize manual intervention, and improve end-to-end transparency in the order process.

Objective:

The primary goal of this project is to optimize and digitize the vehicle ordering process using salesforce CRM tools.

Key objectives include:

- **Enhanced Customer Experience** – Simplify the vehicle ordering process through intelligent dealer suggestions and real-time notifications.
- **Real-Time Stock Management** – Prevent out-of-stock orders by implementing automated inventory validation.
- **Automated Dealer Assignment** – Assign orders dynamically to the nearest dealer using geolocation data
- **Efficient Order Processing** – Automate order updates using Batch Apex and Scheduled Apex jobs.
- **Scalable Automation** – Leverage Apex, Flows, and Process Builder for modular and reusable automation components.

DETAILED EXECUTION OF PROJECT PHASES

Phase 1: Problem Understanding & Industry Analysis:

Problem Understanding:

The automotive retail sector often struggles with disconnected systems, manual order tracking, and limited visibility into stock availability. Customers frequently face delays and communication gaps when ordering vehicles. To address these issues, We implemented a **Salesforce-based Smart Automotion Order Management System** to:

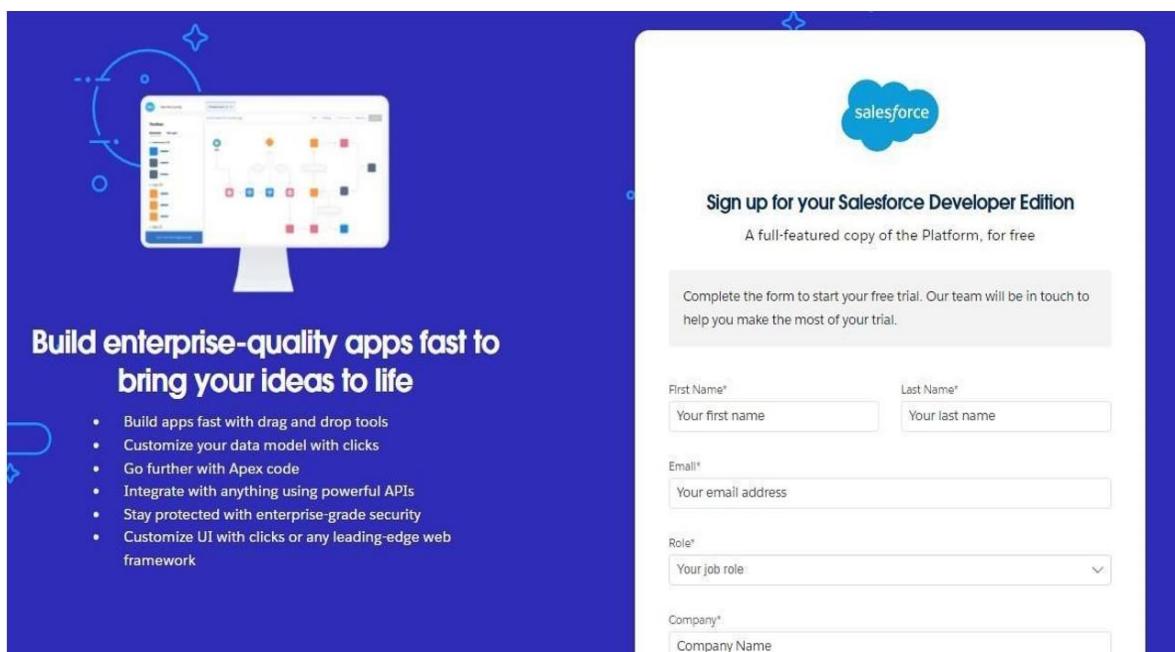
- Automate order creation and stock validation.
- Provide real-time dealer assignment.
- Enable automated notifications and process transparency.

Industry analysis:

Industry analysis revealed that automation in CRM-driven vehicle order management reduces errors, accelerates sales cycles, and increases customer retention.

Phase 2: Org Setup & Configuration:

A Salesforce Developer Org was Created Using <https://developer.salesforce.com/signup>.



The account was verified, password set , and access was granted to the Salesforce Setup page.

Phase 3: Data Modeling & Relationships:

Custom Object Creation

Six Custom Objects were created to store business-critical data

- **Vehicle** – Stores vehicle details like model, stock, price, and status.
- **Vehicle Dealer** – Contains information about dealers such as location and contact details.
- **Vehicle Order** – Tracks customer orders, order dates, and order status.
- **Vehicle Customer** – Maintains customer details including contact and preferred vehicle type.
- **Vehicle Test Drive** – Records test drive schedules and status.
- **Vehicle Service Request** – Logs service requests, dates, issues, and progress status.

This screenshot shows the 'Object Manager' section of the Salesforce Setup interface for the 'Vehicle' object. The left sidebar lists various configuration options: Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, List View Button Layout, Restriction Rules, and Scoping Rules. The main 'Details' tab is selected. The 'Description' field contains the text 'API Name: Vehicle'. Other fields include 'Custom' (checkbox checked), 'Singular Label' (Vehicle), and 'Plural Label' (Vehicles). To the right, there are sections for 'Enable Reports', 'Track Activities', 'Track Field History', and deployment status ('Deployed'). The bottom status bar shows 'Thunderstorm w...' and the date '10/26/2025'.

This screenshot shows the 'Object Manager' section of the Salesforce Setup interface for the 'Vehicle Dealer' object. The left sidebar lists the same configuration options as the previous screenshot. The main 'Details' tab is selected. The 'Description' field contains the text 'API Name: Vehicle_Dealer__c'. Other fields include 'Custom' (checkbox checked), 'Singular Label' (Vehicle Dealer), and 'Plural Label' (Vehicle Dealers). To the right, there are sections for 'Enable Reports', 'Track Activities', 'Track Field History', and deployment status ('Deployed'). The bottom status bar shows 'NZ Bnd' and the date '10/26/2025'.

This screenshot shows the 'Object Manager' section of the Salesforce Setup interface for the 'Vehicle Order' object. The left sidebar lists the same configuration options. The main 'Details' tab is selected. The 'Description' field contains the text 'API Name: Vehicle_Order__c'. Other fields include 'Custom' (checkbox checked), 'Singular Label' (Vehicle Order), and 'Plural Label' (Vehicle Orders). To the right, there are sections for 'Enable Reports', 'Track Activities', 'Track Field History', and deployment status ('Deployed'). The bottom status bar shows 'Air Maritime' and the date '10/26/2025'.

Setup > OBJECT MANAGER

Vehicle Customer

Details

Fields & Relationships	Description
Page Layouts	API Name Vehicle_Customer_c Custom ✓ Singular Label Vehicle Customer Plural Label Vehicle Customers
Lightning Record Pages	Enable Reports ✓ Track Activities
Buttons, Links, and Actions	Track Field History
Compact Layouts	Deployment Status Deployed
Field Sets	Help Settings
Object Limits	Standard salesforce.com Help Window
Record Types	
Related Lookup Filters	
Search Layouts	
List View Button Layout	
Restriction Rules	
Scoping Rules	

https://orgfarm-f023a29e09-dev-ed.my.salesforce.com/one/one.app#/setup/ObjectManager/01lg500000CH39Y/PageLayouts/view

HNZ - INC Live

0 44 AM 10/26/2025

Setup > OBJECT MANAGER

Vehicle Test Drive

Details

Fields & Relationships	Description
Page Layouts	API Name Vehicle_Test_Drive_c Custom ✓ Singular Label Vehicle Test Drive Plural Label Vehicle Test Drives
Lightning Record Pages	Enable Reports ✓ Track Activities
Buttons, Links, and Actions	Track Field History
Compact Layouts	Deployment Status Deployed
Field Sets	Help Settings
Object Limits	Standard salesforce.com Help Window
Record Types	
Related Lookup Filters	
Search Layouts	
List View Button Layout	
Restriction Rules	
Scoping Rules	

Very humid Now

0 845 AM 10/26/2025

Setup > OBJECT MANAGER

vehicle Service Request

Details

Fields & Relationships	Description
Page Layouts	API Name vehicle_Service_Request_c Custom ✓ Singular Label vehicle Service Request Plural Label vehicle Service Requests
Lightning Record Pages	Enable Reports ✓ Track Activities
Buttons, Links, and Actions	Track Field History
Compact Layouts	Deployment Status Deployed
Field Sets	Help Settings
Object Limits	Standard salesforce.com Help Window
Record Types	
Related Lookup Filters	
Search Layouts	
List View Button Layout	
Restriction Rules	
Scoping Rules	

https://orgfarm-f023a29e09-dev-ed.my.salesforce.com/one/one.app#/setup/...

Air Moderate Tomorrow

0 845 AM 10/26/2025

Fields & Relationship

Setup > Object Manager > Vehicle

Fields & Relationships				
9 items, Sorted by Field Label				
FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User/Group)		
Price	Price__c	Currency(18, 0)		
Status	Status__c	Picklist		
Stock Quantity	Stock_Quantity__c	Number(18, 0)		
Vehicle Dealer	Vehicle_Dealer__c	Lookup(Vehicle Dealer)		
Vehicle Model	Vehicle_Model__c	Picklist		
Vehicle Name	Name	Text(80)		

Setup > Object Manager > Vehicle Dealer

Fields & Relationships				
8 items, Sorted by Field Label				
FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Dealer Code	Dealer_Code__c	Auto Number		
Dealer Location	Dealer_Location__c	Text(50)		
Email	Email__c	Email		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User/Group)		
Phone	Phone__c	Phone		
Vehicle Dealer Name	Name	Text(80)		

Setup > Object Manager > Vehicle Order

Fields & Relationships				
8 items, Sorted by Field Label				
FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Order Date	Order_Date__c	Date		
Owner	OwnerId	Lookup(User/Group)		
Status	Status__c	Picklist		
Vehicle	Vehicle__c	Lookup(Vehicle)		
Vehicle Customer	Vehicle_Customer__c	Lookup(Vehicle Customer)		
Vehicle Order Number	Name	Auto Number		

https://orgfarm-f823a29e09-dev-ed.develop.my.salesforce.com/lightning/setup/ObjectManager/01g500000CH3V/FieldsAndRelationships/view

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Address	Address__c	Text(60)		
Created By	CreatedById	Lookup(User)		
Email	Email__c	Email		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Phone	Phone__c	Phone		
Preferred Vehicle Type	Preferred_Vehicle_Type__c	Picklist		
Vehicle Customer Name	Name	Text(80)		✓

https://orgfarm-f823a29e09-dev-ed.develop.my.salesforce.com/lightning/setup/ObjectManager/01g500000CH8L/FieldsAndRelationships/view

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Status	Status__c	Picklist		
Test Drive Date	Test_Drive_Date__c	Date		
Vehicle	Vehicle__c	Lookup(Vehicle)		✓
Vehicle Customer	Vehicle_Customer__c	Lookup(Vehicle Customer)		✓
Vehicle Test Drive Name	Name	Text(80)		✓

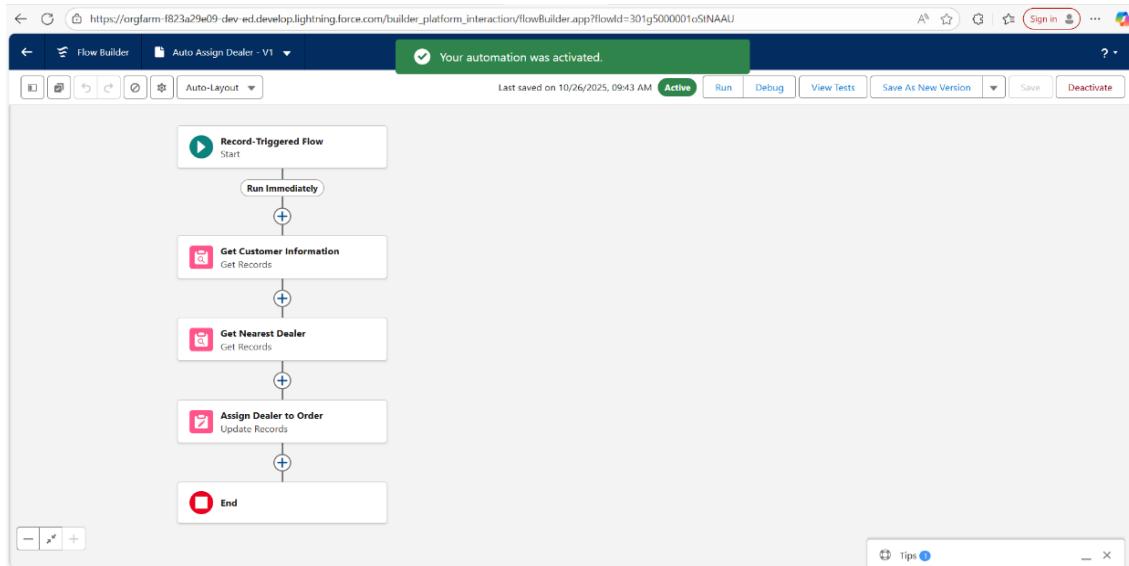
https://orgfarm-f823a29e09-dev-ed.develop.my.salesforce.com/lightning/setup/ObjectManager/01g500000CH9J/FieldsAndRelationships/view

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Issue Description	Issue_Description__c	Text(60)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Service Date	Service_Date__c	Date		
Status	Status__c	Picklist		
Vehicle	Vehicle__c	Lookup(Vehicle)		✓
Vehicle Customer	Vehicle_Customer__c	Lookup(Vehicle Customer)		✓
vehicle Service Request Name	Name	Text(80)		✓

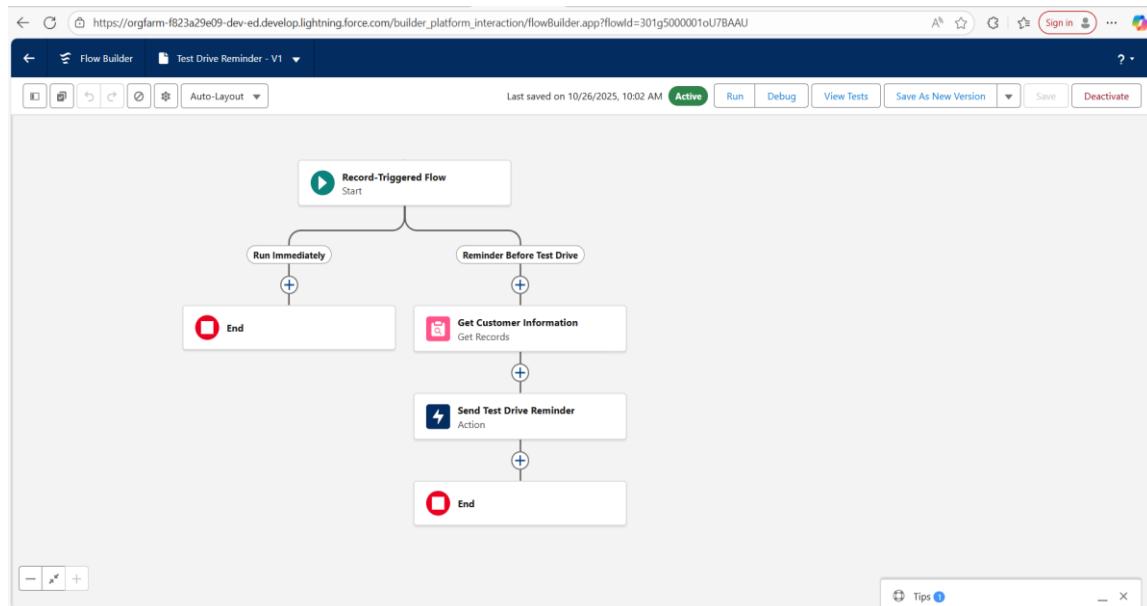
Phase 4: Process Automation (Admin)

Creating Flows

Record-triggered flows were created to automate key business processes without code. For example, a flow was built to **auto-assign the nearest dealer** when a vehicle order is created with status *Pending*.



Another flow sends **email reminders** to customers **one day before** their scheduled test drive. Send automated email notifications before scheduled test drives.



Phase 5: Apex Programming (Developer)

Apex Triggers:

- Validate stock availability on new orders.
- Update order status dynamically (*Pending / Confirmed*).
- Reduce stock quantity automatically after confirmed orders.

Batch Apex:

- Periodically checks stock updates and processes pending orders.

Scheduled Apex:

- Executes the batch process daily to maintain data accuracy.

VehicleOrderTrigger:-

```
trigger VehicleOrderTrigger on Vehicle_Order__c (before insert, before update, after insert, after update) {
```

```
    VehicleOrderTriggerHandler.handleTrigger(Trigger.new, Trigger.oldMap,
    Trigger.isBefore, Trigger.isAfter, Trigger.isInsert, Trigger.isUpdate);
}
```

The screenshot shows the Salesforce Developer Console interface. The top part displays the apex class code for `VehicleOrderTrigger`. The bottom part shows the log tab, which lists two entries from the user 'Garvika Vyshnavi'. The log table has columns for User, Application, Operation, Time, Status, Read, and Size.

User	Application	Operation	Time	Status	Read	Size
Garvika Vyshnavi	Browser	/aura	10/26/2025, 10:13:47 AM	Success	Unread	16.33 KB
Garvika Vyshnavi	Unknown	common.api.soap.DirectSoap	10/26/2025, 10:13:47 AM	Success	Unread	523 bytes

VehicleOrderTriggerHandler:-

```
public class VehicleOrderTriggerHandler {  
    public static void handleTrigger(List<Vehicle_Order_c> newOrders, Map<Id, Vehicle_Order_c> oldOrders, Boolean isBefore, Boolean isAfter, Boolean isInsert, Boolean isUpdate) {  
        if (isBefore && (isInsert || isUpdate)) {  
            preventOrderIfOutOfStock(newOrders);  
        }  
        if (isAfter && (isInsert || isUpdate)) {  
            updateStockOnOrderPlacement(newOrders);  
        }  
    }  
    // ✗ Prevent placing an order if stock is zero  
    private static void preventOrderIfOutOfStock(List<Vehicle_Order_c> orders) {  
        Set<Id> vehicleIds = new Set<Id>();  
        for (Vehicle_Order_c order : orders) {  
            if (order.Vehicle_c != null) {  
                vehicleIds.add(order.Vehicle_c);  
            }  
        }  
        if (!vehicleIds.isEmpty()) {  
            Map<Id, Vehicle_c> vehicleStockMap = new Map<Id, Vehicle_c>(  
                [SELECT Id, Stock_Quantity_c FROM Vehicle_c WHERE Id IN :vehicleIds]  
            );  
  
            for (Vehicle_Order_c order : orders) {  
                Vehicle_c vehicle = vehicleStockMap.get(order.Vehicle_c);  
                if (vehicle != null && vehicle.Stock_Quantity_c <= 0) {  
                    order.addError('This vehicle is out of stock. Order cannot be placed.');//  
                }  
            }  
        }  
    }  
    // ☑ Decrease stock when an order is confirmed  
    private static void updateStockOnOrderPlacement(List<Vehicle_Order_c> orders) {  
        Set<Id> vehicleIds = new Set<Id>();  
        for (Vehicle_Order_c order : orders) {  
            if (order.Vehicle_c != null && order.Status_c == 'Confirmed') {  
                vehicleIds.add(order.Vehicle_c);  
            }  
        }  
        if (!vehicleIds.isEmpty()) {  
            Map<Id, Vehicle_c> vehicleStockMap = new Map<Id, Vehicle_c>(  
                [SELECT Id, Stock_Quantity_c FROM Vehicle_c WHERE Id IN :vehicleIds]  
            );  
            List<Vehicle_c> vehiclesToUpdate = new List<Vehicle_c>();  
            for (Vehicle_Order_c order : orders) {  
                Vehicle_c vehicle = vehicleStockMap.get(order.Vehicle_c);  
                if (vehicle != null && vehicle.Stock_Quantity_c > 0) {  
                    vehicle.Stock_Quantity_c -= 1;  
                    vehiclesToUpdate.add(vehicle);  
                }  
            }  
            if (!vehiclesToUpdate.isEmpty()) {  
                update vehiclesToUpdate;  
            }  
        }  
    }  
}
```

```

47     [SELECT Id, Stock_Quantity__c FROM Vehicle__c WHERE Id IN :vehicleIds]
48 );
49
50     List<Vehicle__c> vehiclesToUpdate = new List<Vehicle__c>();
51     for (Vehicle_Order__c order : orders) {
52         Vehicle__c vehicle = vehicleStockMap.get(order.Vehicle__c);
53         if (vehicle != null && vehicle.Stock_Quantity__c > 0) {
54             vehicle.Stock_Quantity__c -= 1;
55             vehiclesToUpdate.add(vehicle);
56         }
57     }
58
59     if (!vehiclesToUpdate.isEmpty()) {
60         update vehiclesToUpdate;
61     }
62 }
63
64 }
65

```

User	Application	Operation	Time	Status	Read	Size
Gavvala Vyshnavi	Browser	/aura	10/26/2025, 10:13:47 AM	Success	Unread	16.33 KB
Gavvala Vyshnavi	Unknown	common.api.soap.DirectSoap	10/26/2025, 10:13:47 AM	Success	Unread	523 bytes

VehicleOrderBatchScheduler:-

```

global class VehicleOrderBatchScheduler implements Schedulable {
    global void execute(SchedulableContext sc) {
        VehicleOrderBatch batchJob = new VehicleOrderBatch();
        Database.executeBatch(batchJob, 50); // 50 = batch size
    }
}

```

```

1 * global class VehicleOrderBatchScheduler implements Schedulable {
2 *     global void execute(SchedulableContext sc) {
3         VehicleOrderBatch batchJob = new VehicleOrderBatch();
4         Database.executeBatch(batchJob, 50); // 50 = batch size
5     }
6

```

User	Application	Operation	Time	Status	Read	Size
Gavvala Vyshnavi	Browser	/aura	10/26/2025, 10:16:13 AM	Success	Unread	7.81 KB
Gavvala Vyshnavi	Unknown	common.api.soap.DirectSoap	10/26/2025, 10:16:13 AM	Success	Unread	523 bytes
Gavvala Vyshnavi	Browser	/aura	10/26/2025, 10:16:05 AM	Success	Unread	7.82 KB
Gavvala Vyshnavi	Unknown	common.api.soap.DirectSoap	10/26/2025, 10:16:05 AM	Success	Unread	523 bytes
Gavvala Vyshnavi	Browser	/aura	10/26/2025, 10:13:47 AM	Success	Unread	16.33 KB
Gavvala Vyshnavi	Unknown	common.api.soap.DirectSoap	10/26/2025, 10:13:47 AM	Success	Unread	523 bytes

VehicleOrderBatch:-

```
global class VehicleOrderBatch implements Database.Batchable<sObject> {

    global Database.QueryLocator start(Database.BatchableContext bc) {
        return Database.getQueryLocator([
            SELECT Id, Status__c, Vehicle__c FROM Vehicle_Order__c WHERE Status__c
= 'Pending'
        ]);
    }

    global void execute(Database.BatchableContext bc, List<Vehicle_Order__c>
orderList) {
        Set<Id> vehicleIds = new Set<Id>();
        for (Vehicle_Order__c order : orderList) {
            if (order.Vehicle__c != null) {
                vehicleIds.add(order.Vehicle__c);
            }
        }

        if (!vehicleIds.isEmpty()) {
            Map<Id, Vehicle__c> vehicleStockMap = new Map<Id, Vehicle__c>(
                [SELECT Id, Stock_Quantity__c FROM Vehicle__c WHERE Id IN :vehicleIds]
            );

            List<Vehicle_Order__c> ordersToUpdate = new List<Vehicle_Order__c>();
            List<Vehicle__c> vehiclesToUpdate = new List<Vehicle__c>();

            for (Vehicle_Order__c order : orderList) {
                Vehicle__c vehicle = vehicleStockMap.get(order.Vehicle__c);
                if (vehicle != null && vehicle.Stock_Quantity__c > 0) {
                    order.Status__c = 'Confirmed';
                    vehicle.Stock_Quantity__c -= 1;
                    ordersToUpdate.add(order);
                    vehiclesToUpdate.add(vehicle);
                }
            }

            if (!ordersToUpdate.isEmpty()) update ordersToUpdate;
            if (!vehiclesToUpdate.isEmpty()) update vehiclesToUpdate;
        }
    }

    global void finish(Database.BatchableContext bc) {
        System.debug('Vehicle order batch job completed.');
    }
}
```

```

24
25 *         for (Vehicle_Order__c order : orderList) {
26             Vehicle__c vehicle = vehicleStockMap.get(order.Vehicle__c);
27             if (vehicle != null && vehicle.Stock_Quantity__c > 0) {
28                 order.Status__c = 'Confirmed';
29                 vehicle.Stock_Quantity__c -= 1;
30                 ordersToUpdate.add(order);
31                 vehiclesToUpdate.add(vehicle);
32             }
33         }
34
35         if (!ordersToUpdate.isEmpty()) update ordersToUpdate;
36         if (!vehiclesToUpdate.isEmpty()) update vehiclesToUpdate;
37     }
38 }
39
40 *     global void finish(Database.BatchableContext bc) {
41     System.debug('Vehicle order batch job completed.');
42 }
43

```

Log tab showing log entries:

User	Application	Operation	Time	Status	Read	Size
Gavvala Vysnavi	Browser	/aura	10/26/2025, 10:18:13 AM	Success	Unread	7.81 kB
Gavvala Vysnavi	Unknown	common.api.soap.DirectSoap	10/26/2025, 10:18:13 AM	Success	Unread	523 bytes
Gavvala Vysnavi	Browser	/aura	10/26/2025, 10:18:05 AM	Success	Unread	7.82 kB
Gavvala Vysnavi	Unknown	common.api.soap.DirectSoap	10/26/2025, 10:18:05 AM	Success	Unread	523 bytes
Gavvala Vysnavi	Browser	/aura	10/26/2025, 10:13:47 AM	Success	Unread	16.33 kB
Gavvala Vysnavi	Unknown	common.api.soap.DirectSoap	10/26/2025, 10:13:47 AM	Success	Unread	523 bytes

Phase 6: User Interface Development

Creating a Custom Tab

Creating Custom Object tabs for

- Vehicle
- Vehicle Customers
- Vehicle Dealer
- Vehicle Order
- Vehicle Test Drive
- Vehicle Service Request

Custom Tabs

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

Custom Object Tabs

Action	Label	Tab Style	Description
Edit Del	Vehicle Customers	People	
Edit Del	Vehicle Dealers	Building	
Edit Del	Vehicle Orders	Box	
Edit Del	Vehicles	Car	
Edit Del	Vehicle Service Requests	Form	
Edit Del	Vehicle Test Drives	Gears	

Web Tabs

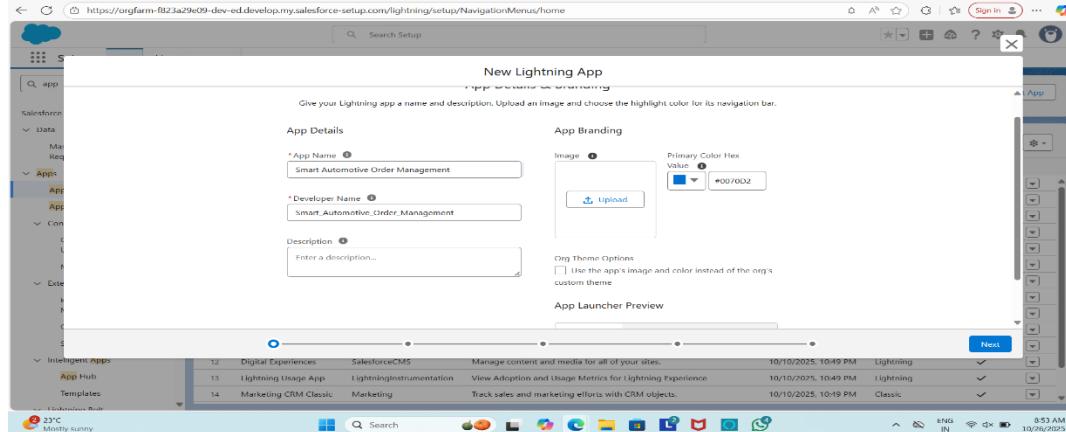
No Web Tabs have been defined.

Visualforce Tabs

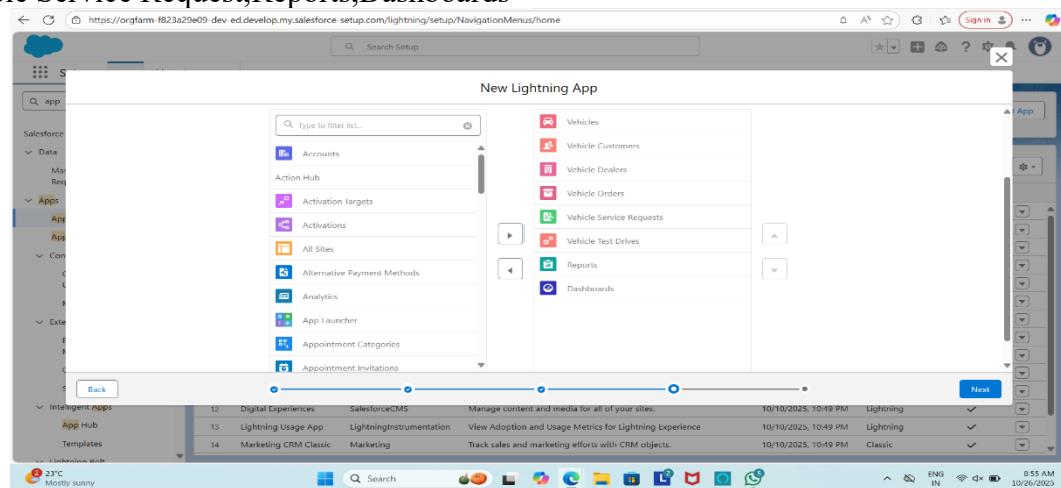
No visualforce Tabs have been defined.

Creating the Lighting App

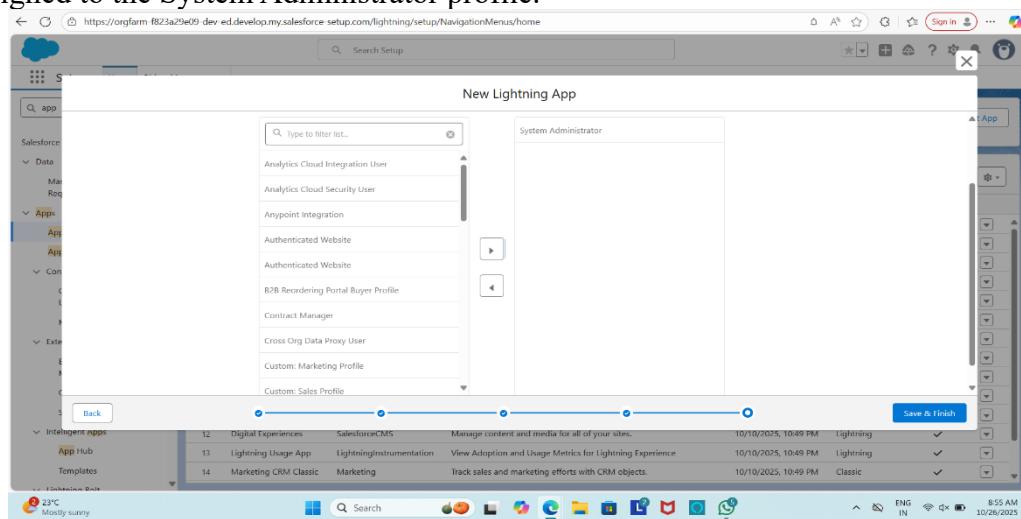
A custom Lightning App named **Smart Automotive Order Management System** was created.



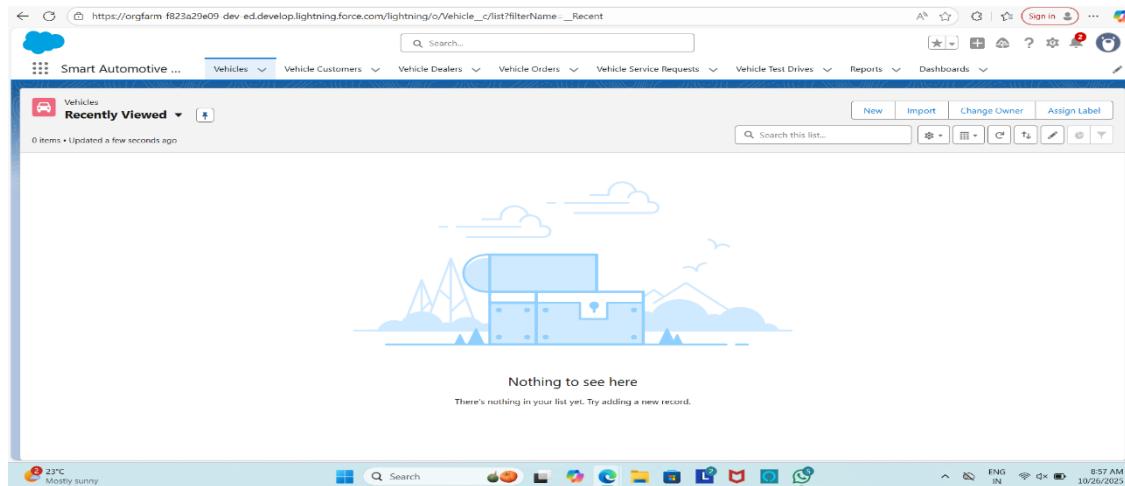
Included tabs: Vehicle, Vehicle Dealer, Vehicle Order, Vehicle Customer, Vehicle Test Drive, Vehicle Service Request, Reports, Dashboards



Assigned to the System Administrator profile.

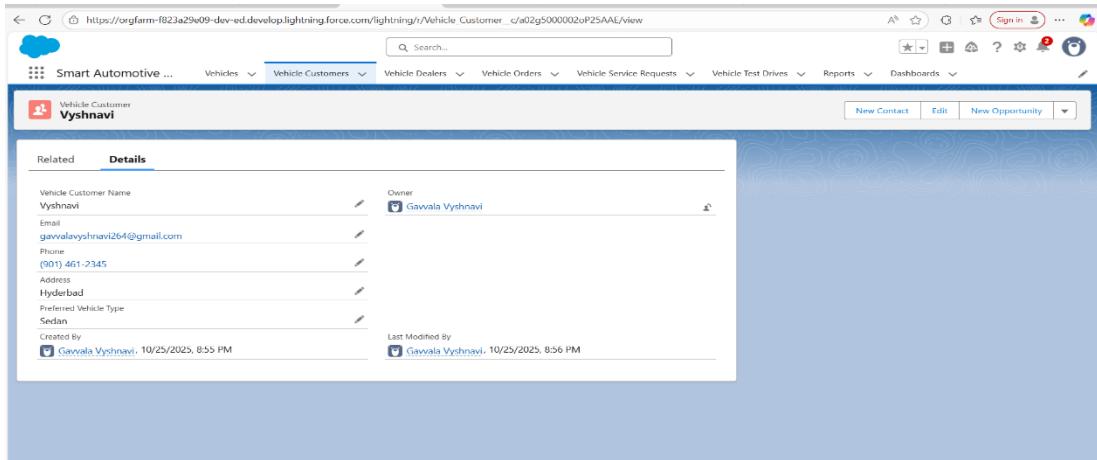
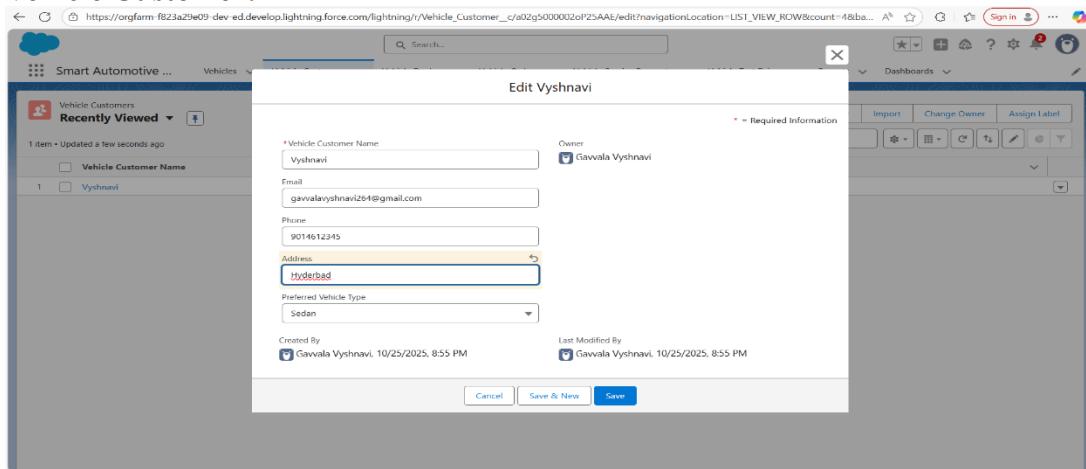


Smart Automotive Order Management App:



Phase 7&8: Integration & External Access & Data Management & Deployment

Vehicle Customer:



Vehicle Dealer:

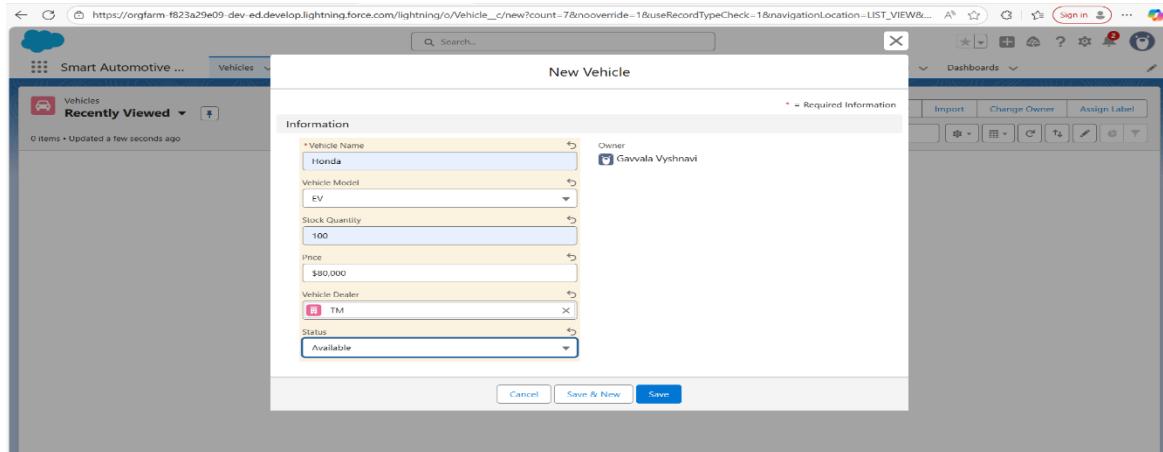
The screenshot shows the 'New Vehicle Dealer' form in the Smart Automotive application. The form includes fields for Vehicle Dealer Name (EM), Dealer Location (Hyderabad), Dealer Code (DC-0001), Phone (9014612345), and Email (gavvalavyshnavi264@gmail.com). The owner is listed as Gavvala Vyshnavi. Buttons at the bottom include 'Cancel', 'Save & New', and 'Save'. A message at the bottom says, 'There's nothing in your list yet. Try adding a new record.'

The screenshot shows the details page for the Vehicle Dealer 'EM'. It displays the same information as the new form: Vehicle Dealer Name (EM), Dealer Location (Hyderabad), Dealer Code (DC-0001), Phone (9014612345), and Email (gavvalavyshnavi264@gmail.com). The owner is listed as Gavvala Vyshnavi. The page also shows the creation date (10/25/2025, 8:58 PM) and the last modified date (10/25/2025, 8:58 PM).

Vehicle Order

The screenshot shows the details page for the Vehicle Order 'O-0001'. It displays the following information: Vehicle Order Number (O-0001), Vehicle Customer (Vyshnavi), Vehicle (Honda), Order Date (10/26/2025), Status (Pending), Assigned Dealer (EM), and Creation Date (10/25/2025, 9:15 PM). The owner is listed as Gavvala Vyshnavi.

Vehicle



Phase 9: Reporting, Dashboards

Dashboards: Real-time vehicle stock, order volume, and revenue insights.

Reports:

- Vehicle Sales by Dealer
- Pending vs Confirmed Orders
- Test Drive Statistics

The screenshot shows the 'New Vehicles Report' page. The report preview displays the following data:

Vehicle	Vehicle Name	Vehicle Dealer	Vehicle Model	Stock Quantity	Price	Status
1	Honda	TM	EV	100	\$80,000	Available

The report configuration sidebar on the left includes sections for 'Fields' (Outline, Iterations, Groups, Add group), 'Columns' (Add column, Vehicle Name, Vehicle Dealer, Vehicle Model, Stock Quantity, Price, Status), and 'Groups' (group rows).

Phase 10: Quality Assurance Testing

Smart Automotive Order Management System was conducted to ensure the reliability of all automated and manual processes. When a new vehicle order was added while stock was available, the system successfully confirmed the order. Attempts to create an order when stock was zero were correctly blocked. The daily batch job reliably updated pending orders to confirmed status. Orders created from a customer's location were automatically assigned to the nearest dealer. Additionally, scheduling a test drive triggered email reminders to be sent before the scheduled time.

Real time Stock Management:

Automatic stock decrease when we confirm the order:

Smart Automotive ... Vehicles Vehicle Customers Vehicle Dealers Vehicle Orders Vehicle Service Requests Vehicle Test Drives Reports Dashboards

Vehicle Honda

Vehicle "Honda" was created.

Related Details

Vehicle Name: Honda Owner: Gavala Vyshnavi

Vehicle Model: EV

Stock Quantity: 100

Price: \$80,000

Vehicle Dealer: TM

Status: Available

Created By: Gavala Vyshnavi, 10/25/2025, 9:01 PM

Last Modified By: Gavala Vyshnavi, 10/25/2025, 9:01 PM

Smart Automotive ... Vehicles

New Vehicle Order

Information

Vehicle Order Number: O-0001 Owner: Gavala Vyshnavi

Vehicle Customer: Vyshnavi

Vehicle: Honda

Order Date: 10/27/2025

Status: Confirmed

Assigned Dealer: Search Vehicle Dealers...

Cancel Save & New Save

Smart Automotive ... Vehicles Vehicle Customers Vehicle Dealers Vehicle Orders Vehicle Service Requests Vehicle Test Drives Reports Dashboards

Vehicle Honda

Vehicle "Honda" was created.

Related Details

Vehicle Name: Honda Owner: Gavala Vyshnavi

Vehicle Model: EV

Stock Quantity: 99

Price: \$80,000

Vehicle Dealer: TM

Status: Available

Created By: Gavala Vyshnavi, 10/25/2025, 9:01 PM

Last Modified By: Gavala Vyshnavi, 10/25/2025, 9:43 PM

Auto Assign Dealer:

In the vehicle order the Assigned Dealer is Automatically assigned with its nearest customer location.

New Vehicle Order

Information

Vehicle Order Number

Owner
Gavala Vyshnavi

Vehicle Customer
Vyshnavi

Vehicle
Honda

Order Date
10/26/2025

Status
Pending

Assigned Dealer
Search Vehicle Dealers...

Cancel Save & New Save

Vehicle Order
O-0001

Related Details

Vehicle Order Number
O-0001

Owner
Gavala Vyshnavi

Vehicle Customer
Vyshnavi

Vehicle
Honda

Order Date
10/26/2025

Status
Pending

Assigned Dealer
EM

Created By
Gavala Vyshnavi - 10/25/2025, 9:15 PM

Last Modified By
Gavala Vyshnavi - 10/25/2025, 9:15 PM

New Contact Edit New Opportunity

Out-of-Stock :

The screenshot shows a vehicle record for a Honda. Key details include:

- Vehicle Name: Honda
- Vehicle Model: EV
- Stock Quantity: 0
- Price: \$80,000
- Vehicle Dealer: TM
- Status: Available

The record was created by Gavvala Vyshnavi on 10/25/2025, 9:01 PM, and last modified by her on 10/25/2025, 9:47 PM.

The screenshot shows a 'New Vehicle Order' page. The vehicle customer is listed as Vyshnavi, and the vehicle is listed as Honda. The order date is set to 10/27/2025. The status is Pending. A red error message box appears, stating: "We hit a snag. Review the errors on this page. • This vehicle is out of stock. Order cannot be placed." The page includes standard CRM navigation and save buttons.

Test Drive Remainder:

The screenshot shows an email from Gavvala Vyshnavi in the inbox. The subject is "Reminder: Your Test Drive is Tomorrow!" The email body contains:

Reminder: Your Test Drive is Tomorrow!

Dear User Vyshnavi,

This is a reminder that your test drive for a04g50000066AnAA is scheduled for tomorrow. If you need to reschedule, contact us at Support@gmail.com.

Thank you, Smart Automotive Order Management Team

The email was sent at 10:06 AM (0 minutes ago).

CONCULSION:

Smart Automotive Order Management System leverages Salesforce to streamline its vehicle sales and service operations. By creating custom objects, automated flows, and Apex triggers, the company ensures efficient order processing, real-time stock management, and improved customer engagement. This digital transformation enhances productivity, reduces errors, and delivers a seamless experience for both staff and customers, positioning WhatNext Vision Motors for sustained growth and success.

Future Scope:

1. Customer Portal Integration

- Develop a **Salesforce Community Portal** where customers can log in to:
 - View their order history and current order status.
 - Check loyalty program points and rewards.
 - Raise queries or service requests directly.

2. Mobile App using Salesforce Mobile SDK

- Build a **mobile application** for store staff and managers to:
 - Manage inventory (add or update stock).
 - Process customer orders in real-time.
 - Access dashboards and order reports on the go.

3. Reports & Dashboard

- Create **dynamic dashboards** and reports in Salesforce for:
 - Real-time vehicle stock levels.
 - Sales trends and revenue analysis.
 - Monitoring loyalty program metrics.

4. AI-Powered Recommendations (Einstein AI)

- Integrate Salesforce **Einstein** to:
 - Analyze customer purchase history.
 - Provide **personalized vehicle recommendations** (e.g., suggest SUVs for customers buying family cars).

5. WhatsApp/SMS Integration

- Use Salesforce **Digital Engagement** or third-party tools (like Twilio) to:
 - Send **instant notifications** about order confirmation or pending status.
 - Update customers about loyalty status or promotional offers.