Daniel Alzona

Polytechnic University of the Philippines

WhatsNext Vision Motors: Shaping the Future of Mobility with Innovation and Excellence

ABSTRACT

This project showcases the development and deployment of a customized Salesforce CRM solution for WhatsNext Vision Motors, a forward-thinking automotive company focused on innovation and customer-centric mobility solutions. The main objective of the project was to streamline business processes, enhance sales and project management efficiency, and maintain accurate and centralized data across departments.

The solution includes the creation of multiple custom objects, including **Vehicle__c, Vehicle_Dealer__c, Vehicle_Customer__c, Vehicle_Order__c, Vehicle_Test_Drive__c, and Vehicle_Service_Request__c** to manage vehicle inventory, dealer assignments, customer orders, and service requests. Key business processes were automated using **Record-Triggered Flows, Scheduled Flows, Email Alerts, and Apex triggers**, enabling automatic notifications for new vehicle orders, test drive schedules, and service requests.

Validation rules were applied to ensure data accuracy, and a role-based security model was implemented for Admin, Sales, and Project Management teams. Scheduled Apex batch jobs were also configured to automatically update vehicle availability and track order status.

OBJECTIVE

The primary objective of this project is to design and implement a customized Salesforce CRM solution for WhatsNext Vision Motors that:

- Automates essential business processes including vehicle order assignment, stock validation, test drive scheduling, and service request tracking.
- Ensures accurate and reliable data entry through validation rules.
- Provides real-time visibility into vehicles, customer orders, test drives, and service requests.
- Strengthens cross-departmental coordination using structured role-based access control.

PROJECT PHASES

Phase 1: Architecture & Planning

Custom Object Creation

Six main custom objects were created to store business-critical data:

Object Name	Purpose	Relationships
Vehicle__c	Stores vehicle details	Related to Dealer & Orders

Object Name	**Purpose**	**Relationships**
Vehicle_Dealer__c	Stores authorized dealer info	Related to Orders
Vehicle_Customer__c	Stores customer details	Related to Orders & Test Drives
Vehicle_Order__c	Tracks vehicle purchases	Related to Customer & Vehicle
Vehicle_Test_Drive__c	Tracks test drive bookings	Related to Customer & Vehicle
Vehicle_Service_Request__c	Tracks vehicle servicing requests	Related to Customer & Vehicle

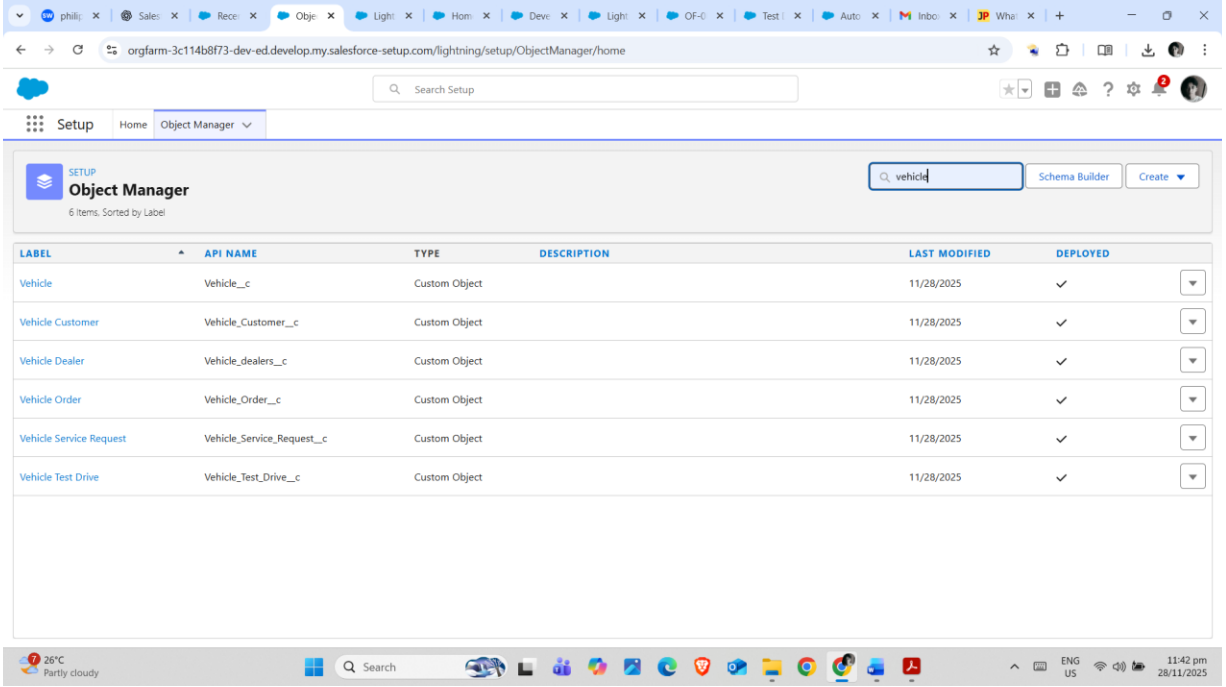
Steps Followed:

- Navigated to **Setup** → **Object Manager** → **Create** → **Custom Object**.
- Provided **Label, Object Name**, and enabled **Reports/Search**.
- Created **Tabs** for each object for easier navigation.

Data Management – Tabs

- Tab – Vehicle__c
- Tab – Vehicle_Dealer__c
- Tab – Vehicle_Customer__c
- Tab – Vehicle_Order__c
- Tab – Vehicle_Test_Drive__c
- Tab – Vehicle_Service_Request__c

Screenshot: Tabs created for all custom objects



SETUP

Object Manager

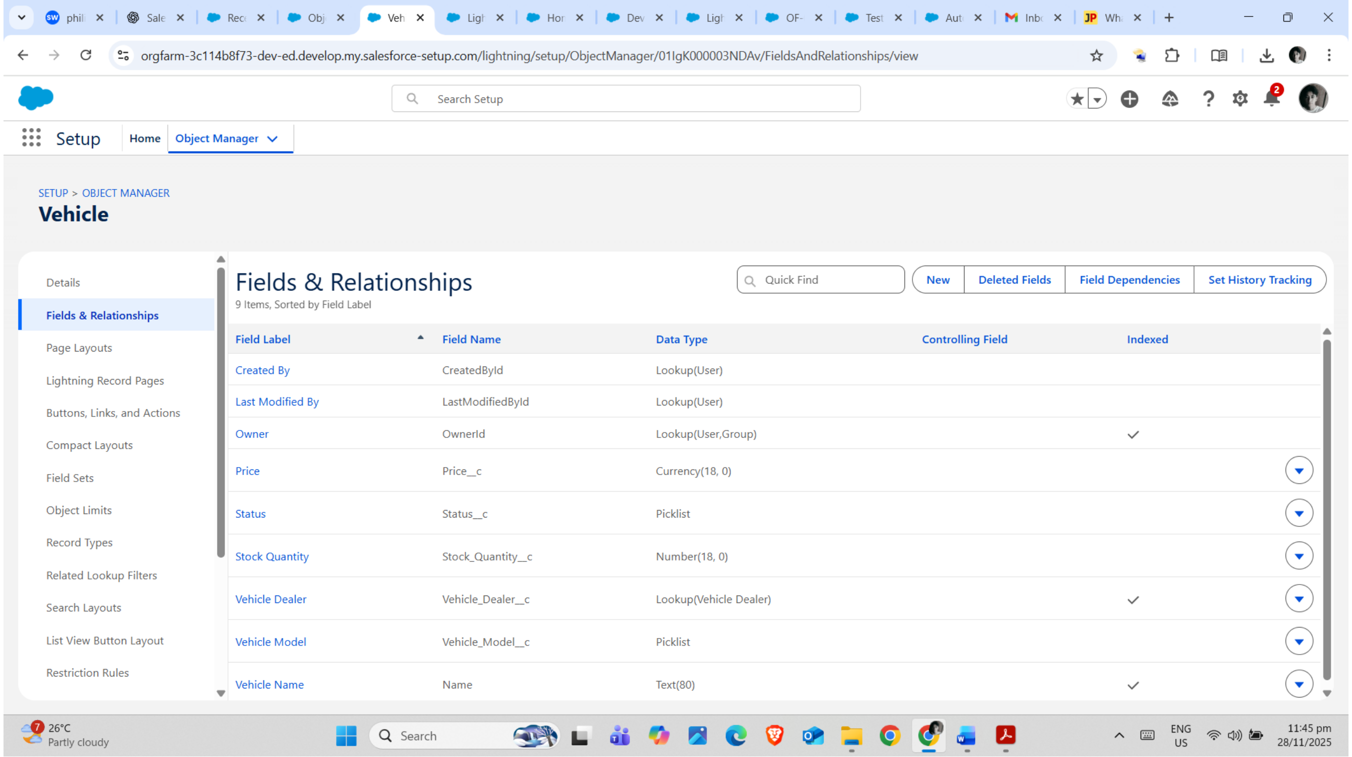
6 Items, Sorted by Label

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Vehicle	Vehicle__c	Custom Object		11/28/2025	✓
Vehicle Customer	Vehicle_Customer__c	Custom Object		11/28/2025	✓
Vehicle Dealer	Vehicle_dealers__c	Custom Object		11/28/2025	✓
Vehicle Order	Vehicle_Order__c	Custom Object		11/28/2025	✓
Vehicle Service Request	Vehicle_Service_Request__c	Custom Object		11/28/2025	✓
Vehicle Test Drive	Vehicle_Test_Drive__c	Custom Object		11/28/2025	✓

Data Management – Fields

Custom fields and lookup/master-detail relationships were created for each object:

- **Vehicle__c:** Vehicle Name, Model, Category, Price, Availability Status



SETUP > OBJECT MANAGER

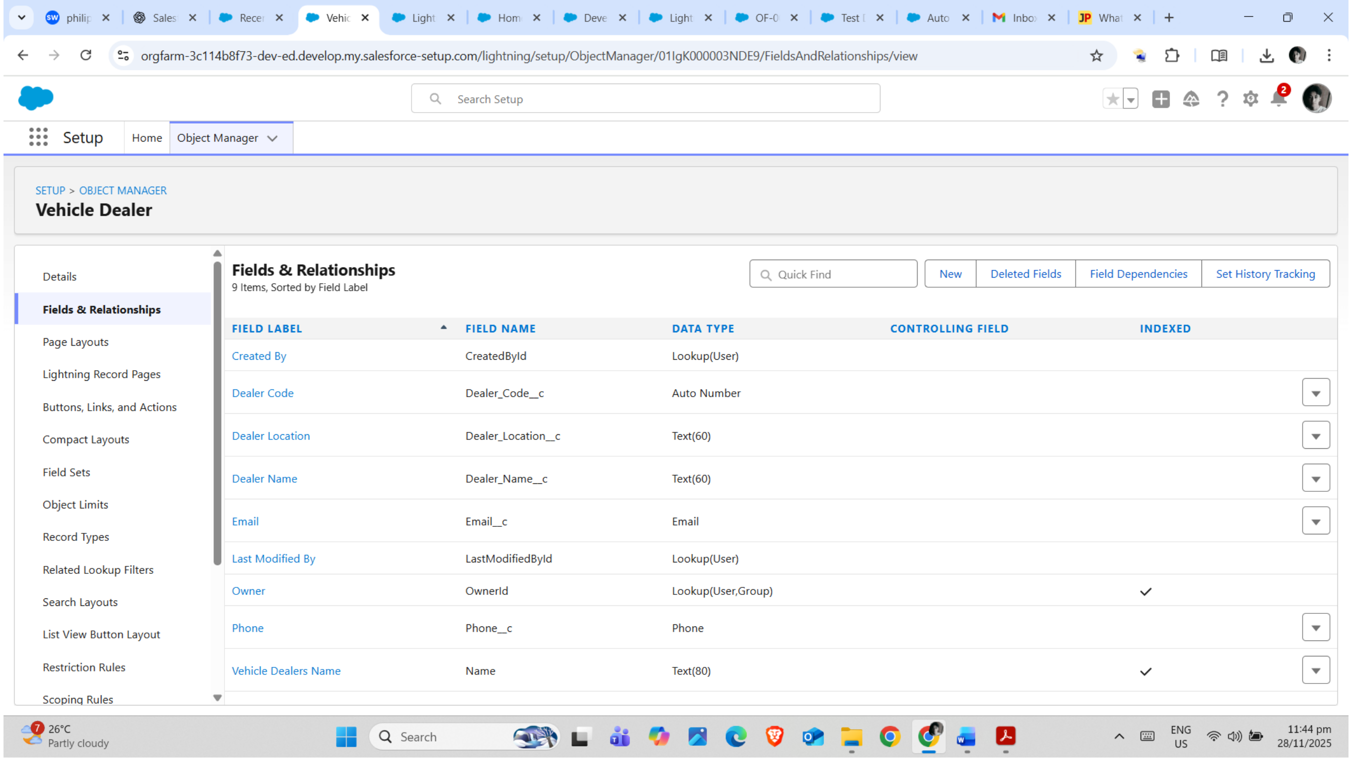
Vehicle

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)		✓

- **Vehicle_Dealer__c:** Dealer Name, Location, Contact Info



SETUP > OBJECT MANAGER

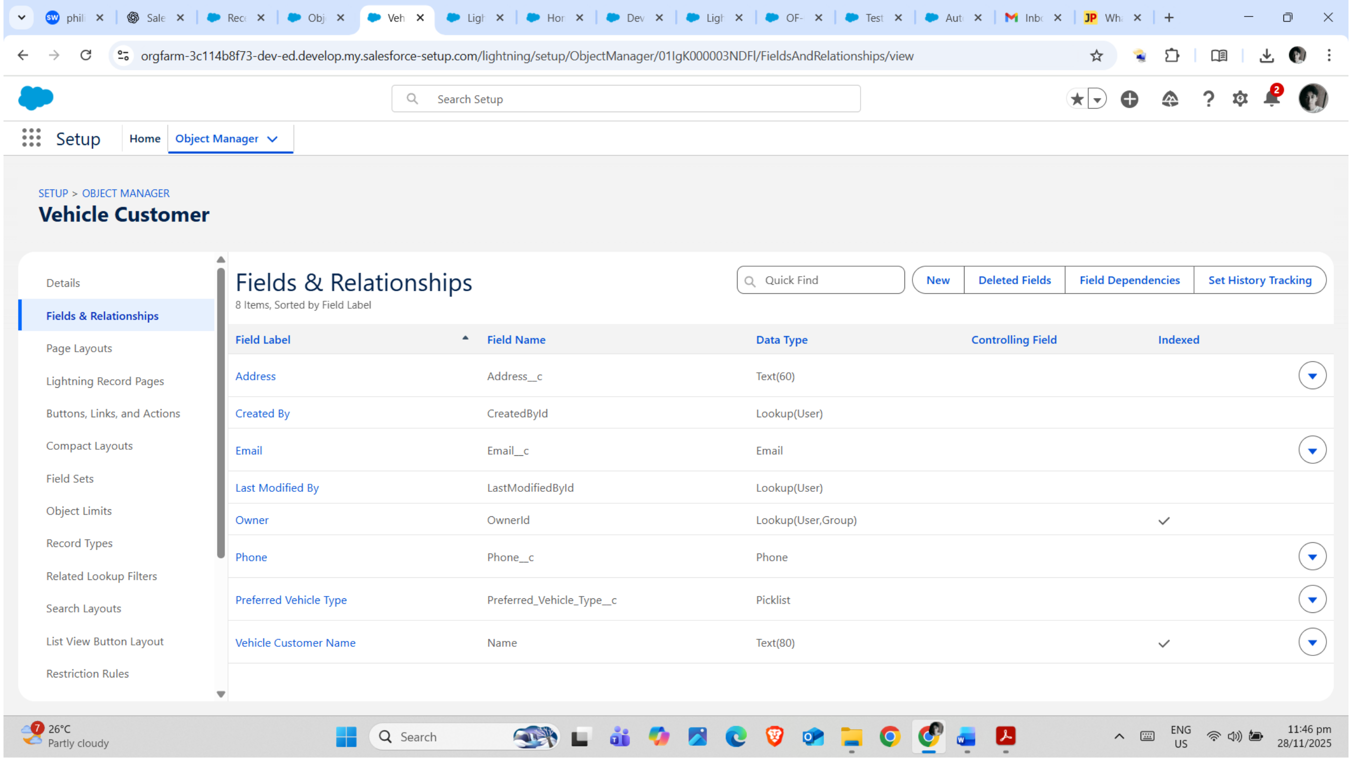
Vehicle Dealer

Fields & Relationships

9 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(60)		
Dealer Name	Dealer_Name__c	Text(60)		
Email	Email__c	Email		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Phone	Phone__c	Phone		
Vehicle Dealers Name	Name	Text(80)		✓

- **Vehicle_Customer__c:** Customer Name, Contact Info, Email



SETUP > OBJECT MANAGER

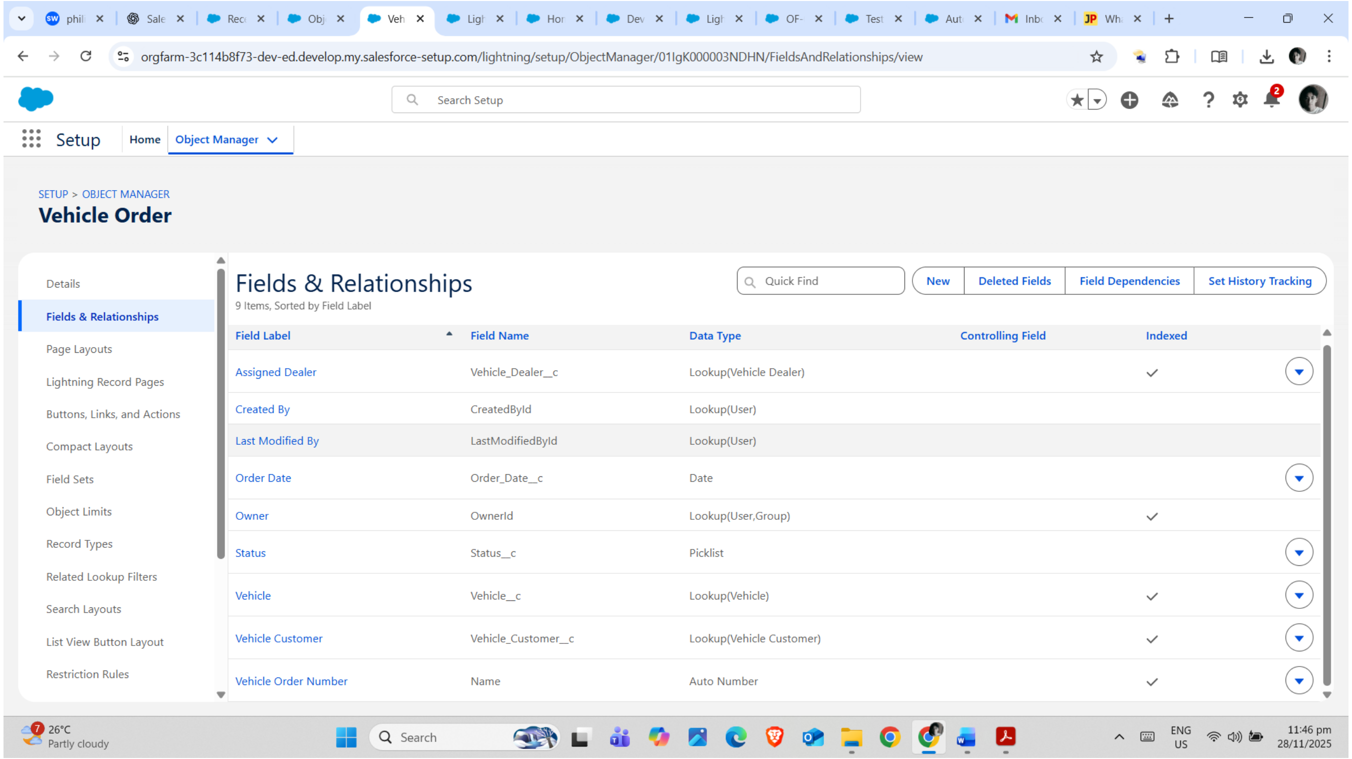
Vehicle Customer

Fields & Relationships

8 Items, Sorted by Field Label

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)		✓

- **Vehicle_Order__c:** Order Number, Order Date, Related Customer, Related Vehicle



SETUP > OBJECT MANAGER

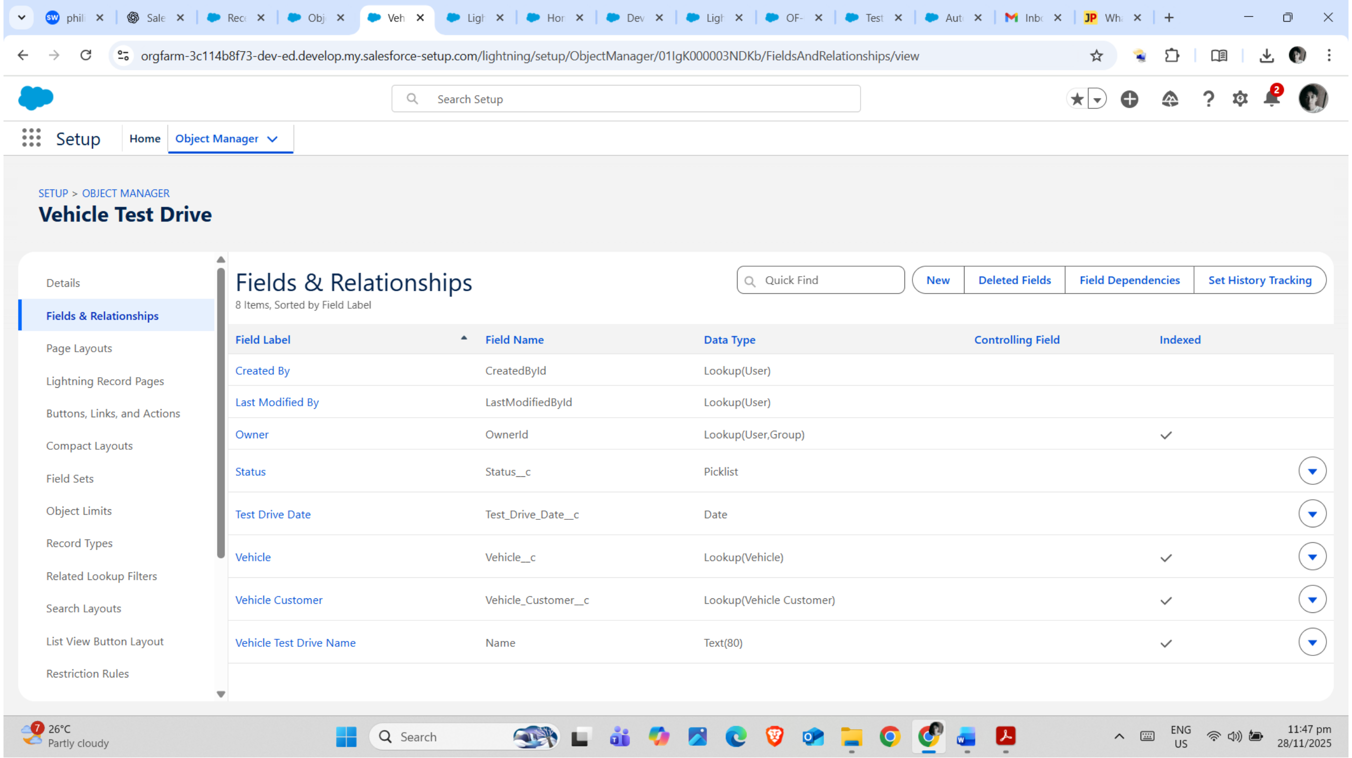
Vehicle Order

Fields & Relationships

9 Items, Sorted by Field Label

Field Label	Field Name	Data Type	Controlling Field	Indexed
Assigned Dealer	Vehicle_Dealer__c	Lookup(Vehicle Dealer)		✓
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		✓

- **Vehicle_Test_Drive__c:** Test Drive Date, Related Customer, Related Vehicle



SETUP > OBJECT MANAGER

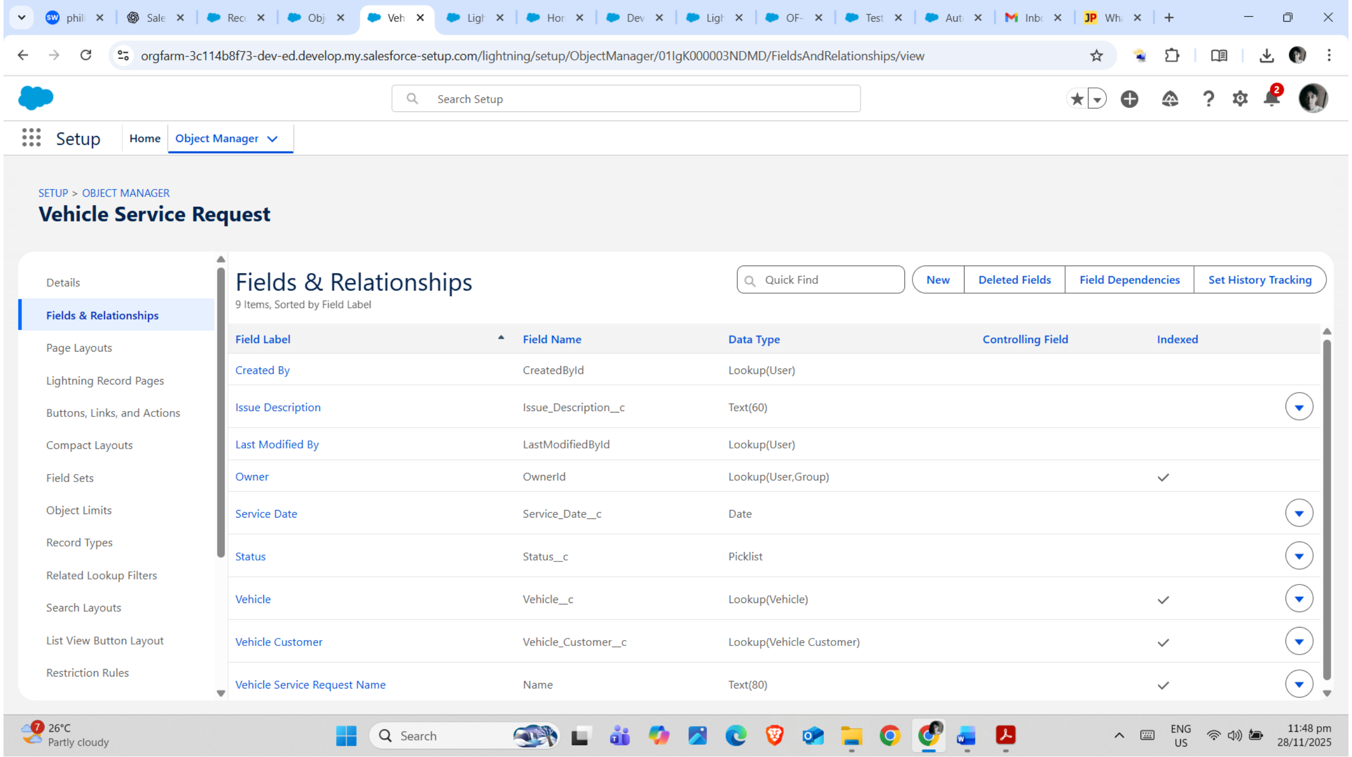
Vehicle Test Drive

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)		
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)		✓

- **Vehicle_Service_Request__c:** Service Type, Service Date, Related Customer, Related Vehicle



SETUP > OBJECT MANAGER

Vehicle Service Request

Fields & Relationships

9 Items, Sorted by Field Label

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)		✓

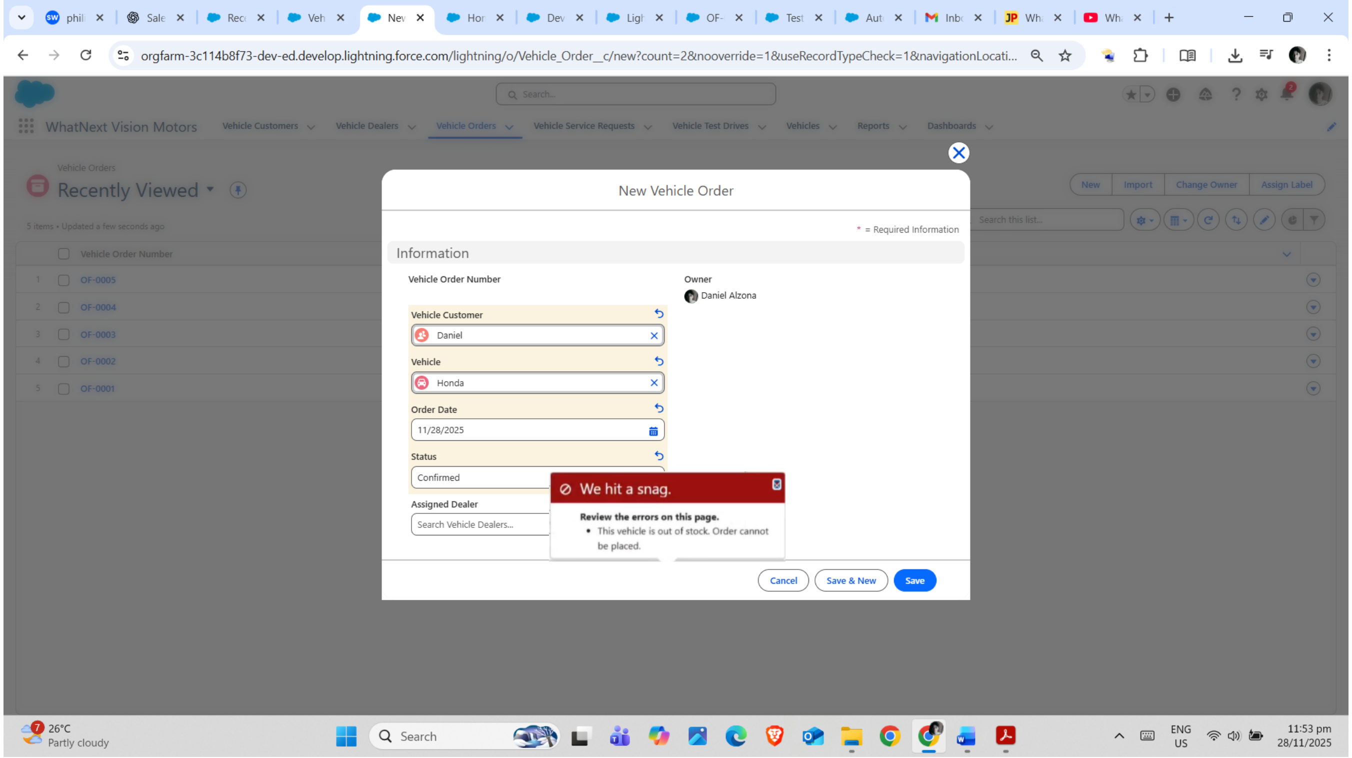
Data Configuration

Validation rules were implemented to ensure accurate data entry:

- **Vehicle__c:** Price must be greater than zero.
- **Vehicle_Order__c:** Cannot place order if Vehicle__c is out of stock.

- **Vehicle_Test_Drive__c:** Test drive date cannot be in the past.

Screenshot: Example of a validation rule applied to Vehicle_Order__c



New Vehicle Order

* = Required Information

Information

Vehicle Order Number

Owner: Daniel Alzona

Vehicle Customer: Daniel

Vehicle: Honda

Order Date: 11/28/2025

Status: Confirmed

Assigned Dealer: Search Vehicle Dealers...

We hit a snag.

Review the errors on this page.

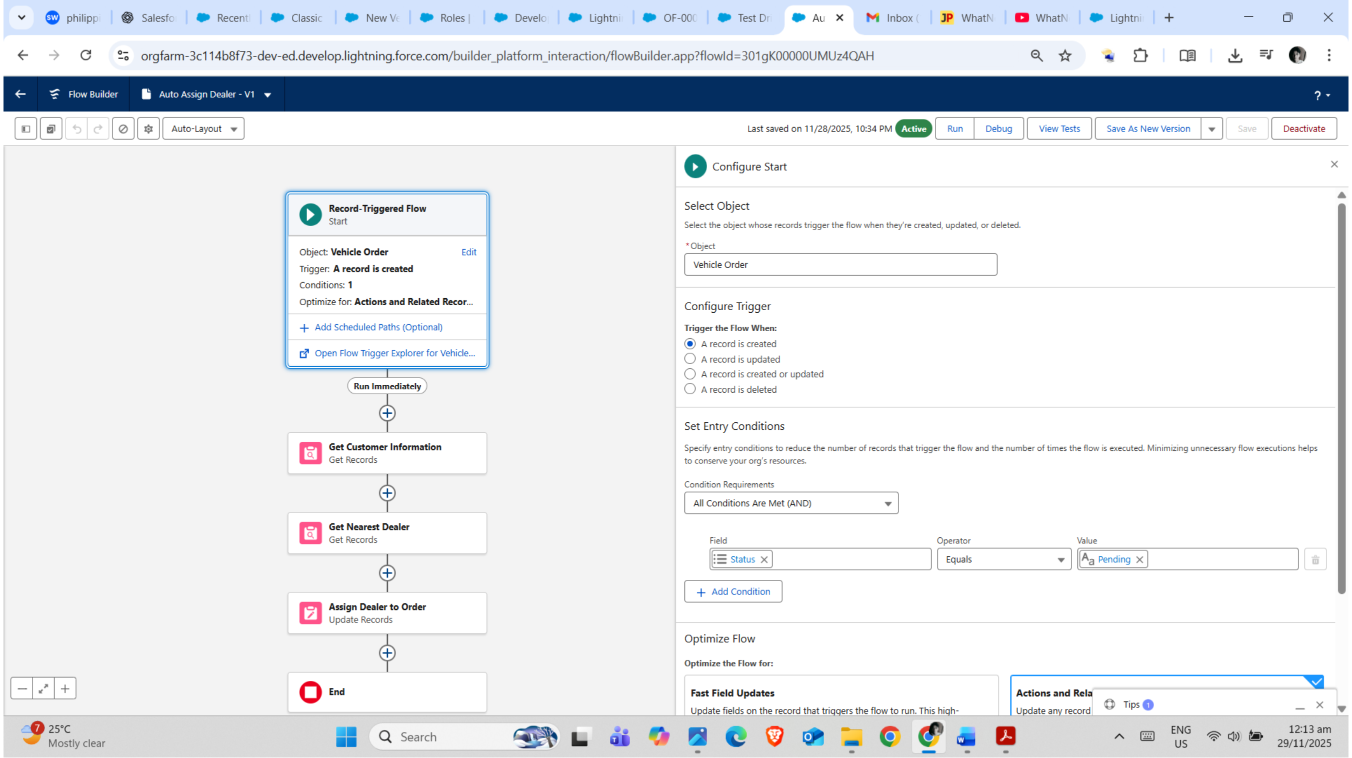
- This vehicle is out of stock. Order cannot be placed.

Cancel | Save & New | Save

Phase 2: Development

Email Templates & Alerts

- Created **Classic Email Templates:** Vehicle_Order_Confirmation, Test_Drive_Reminder, Service_Request_Notification.
- Configured **Email Alerts** linked to **Record-Triggered Flows**.


orgfarm-3c114b8f73-dev-ed.develop.lightning.force.com/builder_platform_interaction/flowBuilder.app?flowId=301gK00000UMUz4QAH
Flow Builder
Auto Assign Dealer - V1
Auto-Layout
Last saved on 11/28/2025, 10:34 PM
Active
Run
Debug
View Tests
Save As New Version
Save
Deactivate
25°C
Mostly clear
Search
12:13 am
29/11/2025

Record-Triggered Flow
Start

Object: **Vehicle Order** Edit
Trigger: **A record is created**
Conditions: **1**
Optimize for: **Actions and Related Recor...**

+ Add Scheduled Paths (Optional)
Open Flow Trigger Explorer for Vehicle...

Run Immediately

Get Customer Information
Get Records

Get Nearest Dealer
Get Records

Assign Dealer to Order
Update Records

End

Configure Start

Select Object

Select the object whose records trigger the flow when they're created, updated, or deleted.

* Object: Vehicle Order

Configure Trigger

Trigger the Flow When:

● A record is created
○ A record is updated
○ A record is created or updated
○ A record is deleted

Set Entry Conditions

Specify entry conditions to reduce the number of records that trigger the flow and the number of times the flow is executed. Minimizing unnecessary flow executions helps to conserve your org's resources.

Condition Requirements: All Conditions Are Met (AND)

Field	Operator	Value
Status	Equals	Pending

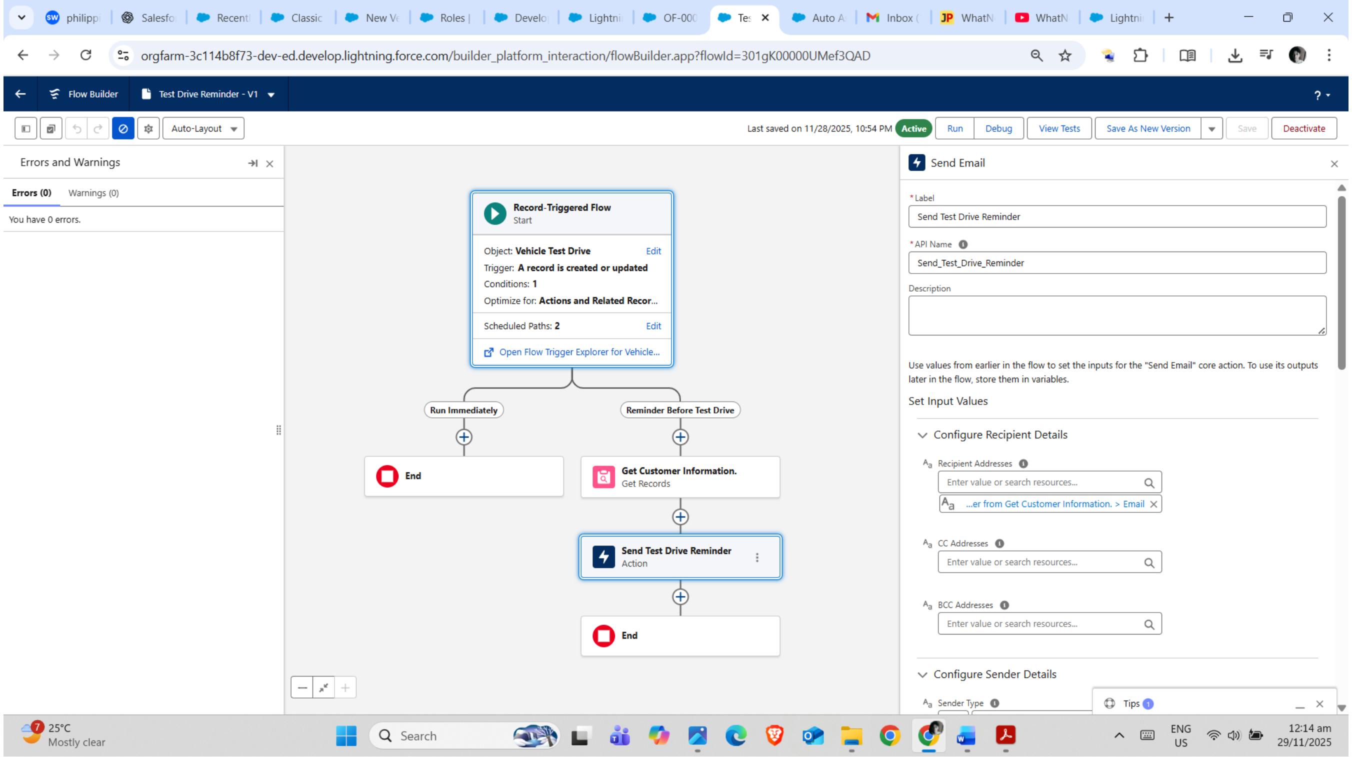
+ Add Condition

Optimize Flow

Optimize the Flow for:

Fast Field Updates
Update fields on the record that triggers the flow to run. This high-

Actions and Rela
Update any record


orgfarm-3c114b8f73-dev-ed.develop.lightning.force.com/builder_platform_interaction/flowBuilder.app?flowId=301gK00000UMef3QAD
Flow Builder
Test Drive Reminder - V1
Auto-Layout
Last saved on 11/28/2025, 10:54 PM
Active
Run
Debug
View Tests
Save As New Version
Save
Deactivate
25°C
Mostly clear
Search
12:14 am
29/11/2025

Errors and Warnings

Errors (0) Warnings (0)

You have 0 errors.

Record-Triggered Flow
Start

Object: **Vehicle Test Drive** Edit
Trigger: **A record is created or updated**
Conditions: **1**
Optimize for: **Actions and Related Recor...**

Scheduled Paths: **2** Edit

Open Flow Trigger Explorer for Vehicle...

Run Immediately

End

Reminder Before Test Drive

Get Customer Information.
Get Records

Send Test Drive Reminder
Action

End

Send Email

* Label: Send Test Drive Reminder

* API Name: Send_Test_Drive_Reminder

Description

Use values from earlier in the flow to set the inputs for the "Send Email" core action. To use its outputs later in the flow, store them in variables.

Set Input Values

Configure Recipient Details

Recipient Addresses: ...er from Get Customer Information. > Email

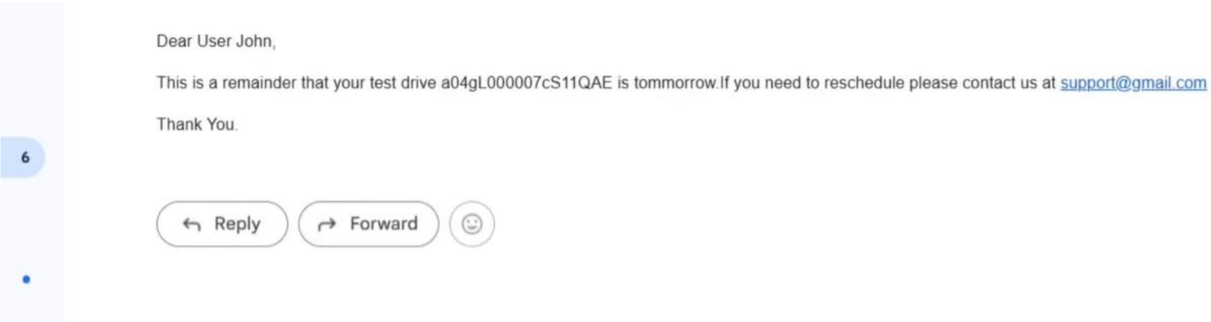
CC Addresses: Enter value or search resources...

BCC Addresses: Enter value or search resources...

Configure Sender Details

Sender Type

Screenshot: Email template example for Vehicle Order Confirmation



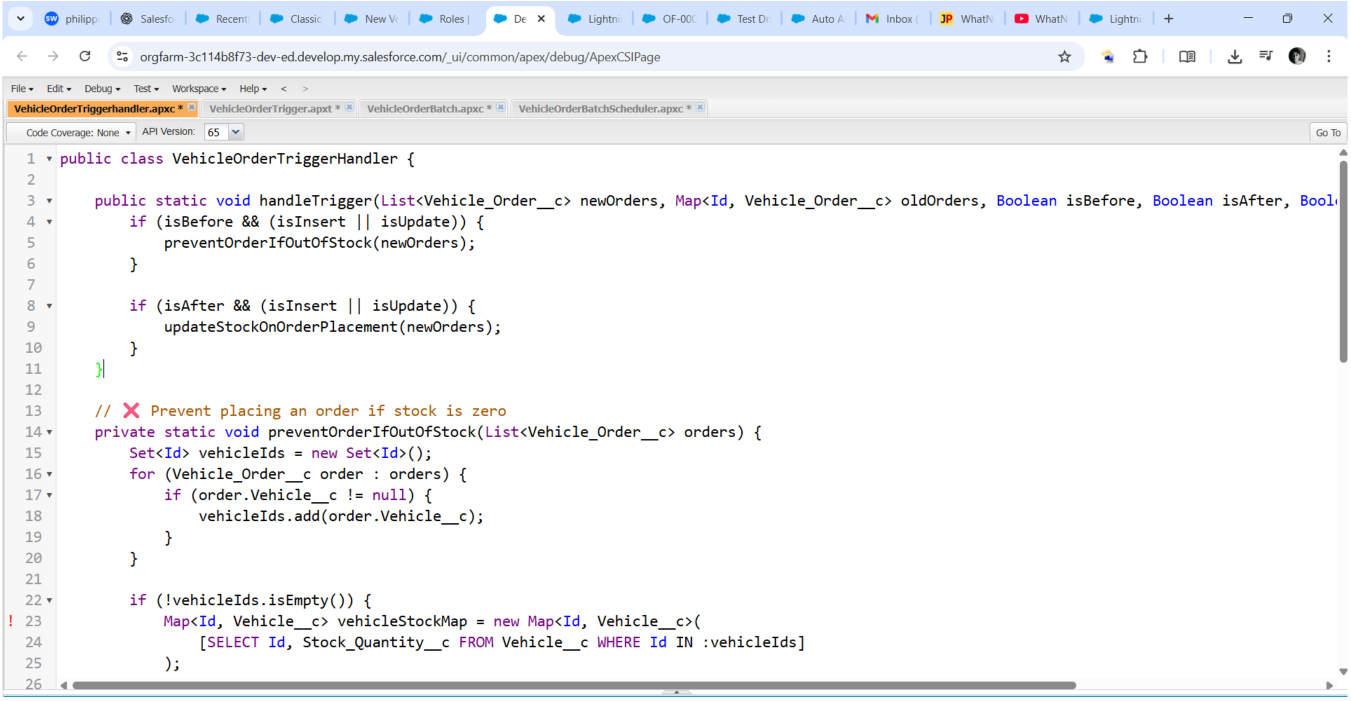
Dear User John,

This is a remainder that your test drive a04gL000007cS11QAE is tommorrow.If you need to reschedule please contact us at support@gmail.com

Thank You.

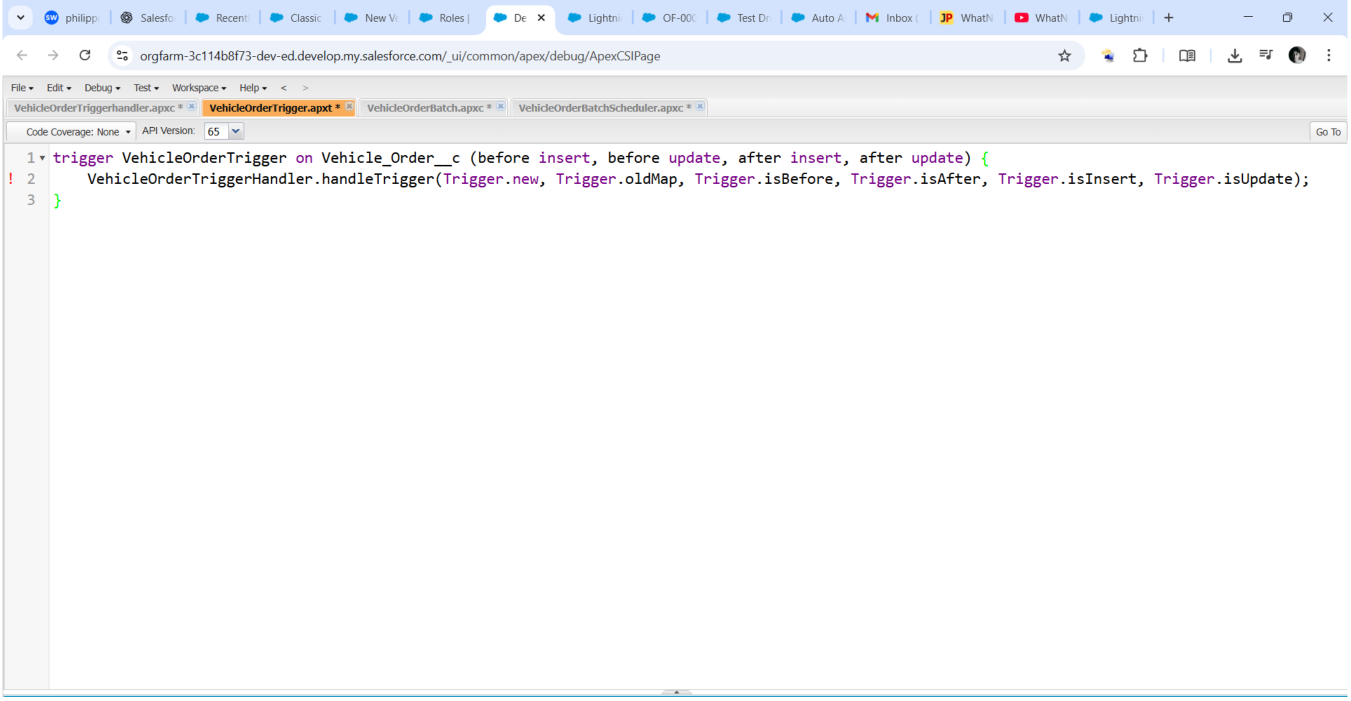
Automation Using Flows & Apex

- **Record-Triggered Flows** for sending order confirmations, test drive reminders, and service request notifications.
- **Apex Triggers:**
  - Apex class: VehicleOrderTriggerhandler


orgfarm-3c114b8f73-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
File Edit Debug Test Workspace Help
VehicleOrderTriggerhandler.apxc *
VehicleOrderTrigger.apxt *
VehicleOrderBatch.apxc *
VehicleOrderBatchScheduler.apxc *
Code Coverage: None
API Version: 65
Go To

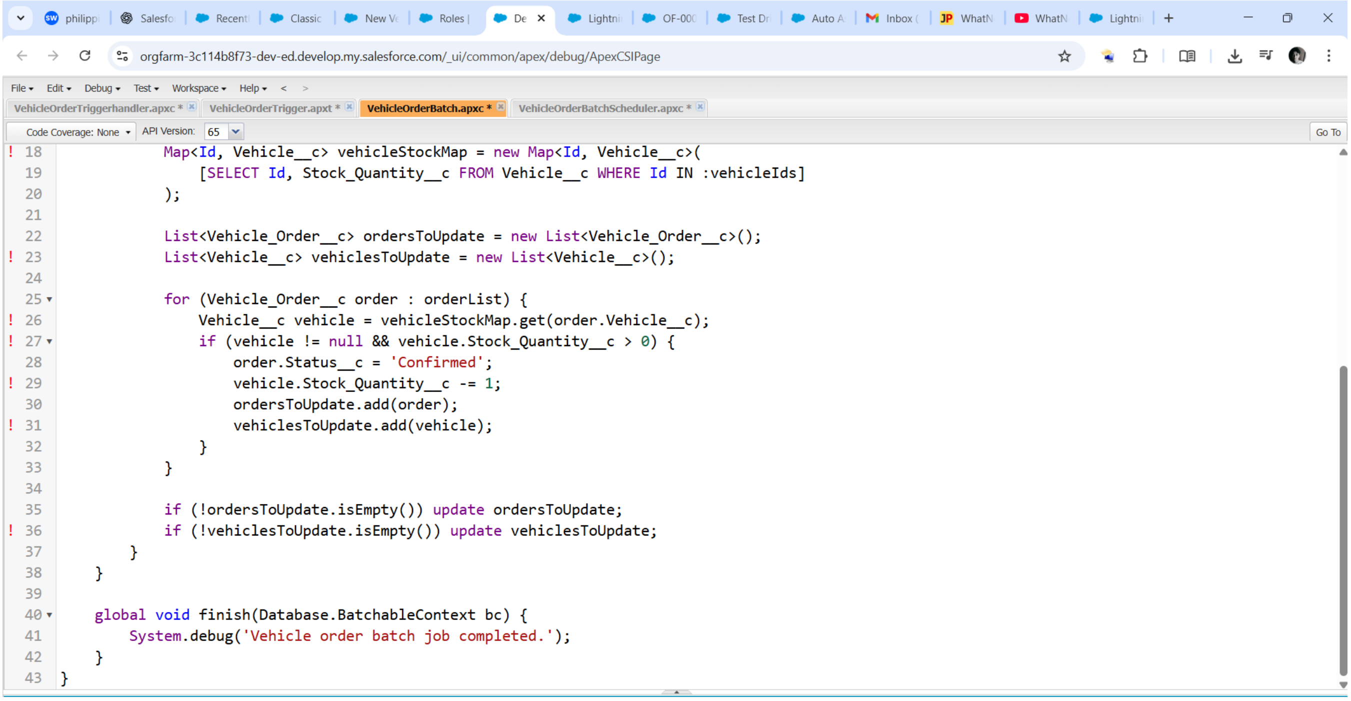
```
public class VehicleOrderTriggerHandler {

    public static void handleTrigger(List<Vehicle_Order__c> newOrders, Map<Id, Vehicle_Order__c> oldOrders, Boolean isBefore, Boolean isAfter, Bool
        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]
            );
```

  - Apex trigger: VehicleOrderTrigger


orgfarm-3c114b8f73-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
File Edit Debug Test Workspace Help
VehicleOrderTriggerhandler.apxc *
VehicleOrderTrigger.apxt *
VehicleOrderBatch.apxc *
VehicleOrderBatchScheduler.apxc *
Code Coverage: None
API Version: 65
Go To

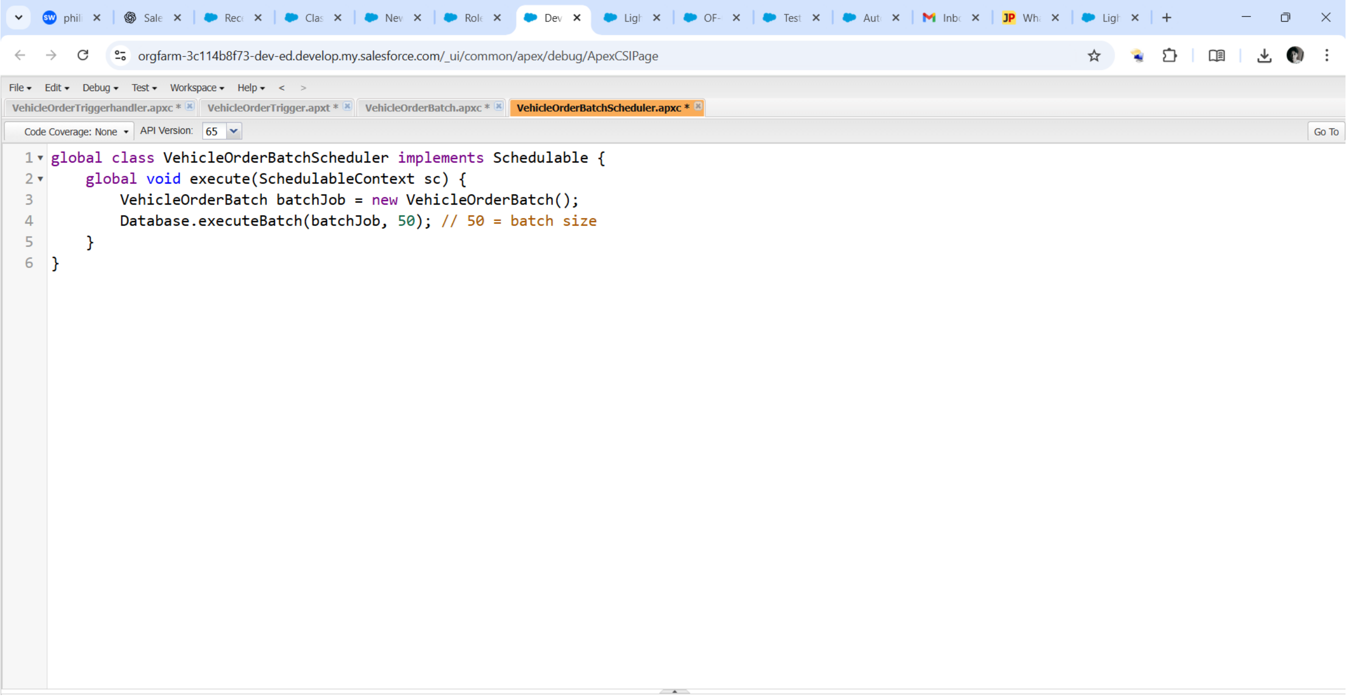
```
trigger VehicleOrderTrigger on Vehicle_Order__c (before insert, before update, after insert, after update) {
    VehicleOrderTriggerHandler.handleTrigger(Trigger.new, Trigger.oldMap, Trigger.isBefore, Trigger.isAfter, Trigger.isInsert, Trigger.isUpdate);
}
```

- **Batch Apex Class:** VehicleBatchJob for updating vehicle availability daily.
    - Apex class: VehicleOrderBatch


orgfarm-3c114b8f73-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
File Edit Debug Test Workspace Help
VehicleOrderTriggerhandler.apxc *
VehicleOrderTrigger.apxt *
VehicleOrderBatch.apxc *
VehicleOrderBatchScheduler.apxc *
Code Coverage: None
API Version: 65
Go To

```
            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.');
    }
}
```

    - Apex class: VehicleOrderBatchScheduler


phil
Sale
Rec
Clas
Nev
Role
Dev
Ligh
OF-
Test
Aut
Inbo
Wh
Ligh
orgfarm-3c114b8f73-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
File
Edit
Debug
Test
Workspace
Help
VehicleOrderTriggerhandler.apxc *
VehicleOrderTrigger.apxt *
VehicleOrderBatch.apxc *
VehicleOrderBatchScheduler.apxc *
Code Coverage: None
API Version: 65
Go To

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

CONCLUSION

The **WhatsNext Vision Motors CRM system** built on Salesforce successfully streamlines key operations such as vehicle management, order tracking, test drive scheduling, and service request handling. By leveraging Salesforce tools like **Custom Objects, Flows, Validation Rules, Email Alerts, and Apex**, the system ensures accurate data entry, real-time updates, and improved coordination across departments. Automation minimizes manual errors, enhances productivity, and provides actionable insights for better decision-making in vehicle sales and service execution.