

## PROJECT OVERVIEW

WhatsNext Vision Motors is a progressive automotive company committed to improving customer experience through digital solutions. To support this direction, a Salesforce CRM platform was developed to strengthen customer engagement, optimize dealer assignment, automate order handling, and maintain accurate stock monitoring. The system consolidates vehicle information, customer records, test drive schedules, orders, and service requests. It also automates dealer assignment and stock validation to ensure a smooth and reliable purchasing process. This implementation increases efficiency, reduces manual tasks, and enhances overall customer satisfaction.

## OBJECTIVES

The CRM aims to streamline the vehicle purchasing and service workflow by establishing an automated, precise, and customer-oriented system. It improves customer management, accelerates dealer assignment, simplifies booking processes, and provides real-time stock visibility. By automating routine tasks, applying business rules, improving communication, and ensuring transparency, the system supports better productivity and builds stronger customer confidence.

## **UNDERSTANDING BUSINESS REQUIREMENTS**

A review of the company's operational needs helped determine the essential functions the CRM must support. These requirements outline the key processes that need to be automated and monitored to ensure accuracy, efficiency, and a seamless experience for both customers and administrators.

The core business needs identified include:

- Assigning customers to the nearest dealer automatically.
- Preventing orders when vehicles are out of stock.
- Tracking vehicles, test drives, customers, orders, dealers, and service requests in a structured system.
- Sending automated reminders for scheduled test drives.
- Keeping order statuses accurate through scheduled and batch processing.
- Providing admins with dashboards and reports for business insights.

## **DEFINING PROJECT SCOPE AND OBJECTIVES**

The project scope was established to address the identified requirements through targeted development, configuration, and automation. It outlines the technical components, system features, and administrative setups required to implement a functional and scalable CRM solution.

Scope includes:

- Creation of 6 custom objects with defined relationships.
- Automation using Flows, Apex Triggers, Batch Apex, and Scheduled Apex.
- Implementation of validation rules and stock-check mechanisms.
- Development of Lightning App for navigation of modules.
- User management and security configuration.
- Testing and deployment activities.

## **SECURITY MODEL INCLUDES**

The security model was designed to ensure that users have the appropriate level of access based on their responsibilities. It defines how data is protected, how records are shared, and how key information is monitored to maintain system integrity and compliance.

- Profiles, Role Hierarchy, Permission Sets.
- Sharing settings for object-level and record-level access.
- Field history tracking for critical fields like Stock, Status, Email.

## **STAKEHOLDER MAPPING**

The stakeholder mapping identifies the groups involved in using or benefiting from the CRM. It outlines their roles within the system and clarifies how each group interacts with the platform to support business operations.

- System Admin: Manages entire CRM, objects, and automations.
- Sales Staff: Uses the CRM to create orders and track customers.
- Dealers: Receives assigned orders and manages vehicle delivery.
- Customers: Indirect stakeholders benefiting from efficient service.

## **EXECUTION ROADMAP**

The execution plan outlines the stages followed to build, verify, and deliver the CRM solution. Each phase focuses on a specific set of development and implementation activities to ensure a structured and reliable rollout.

1. Data Modeling → Object creation
2. Backend Automation → Triggers, batch jobs
3. UI Development → Lightning App, Page Layouts
4. Testing → Unit tests, flow tests, process validation
5. Deployment → Change Set deployment
6. Documentation & Maintenance

## **ENVIRONMENT SETUP & DEVOPS WORKFLOW**

A dedicated Salesforce Developer Org was configured to safely design, develop, and test custom objects, fields, flows, and Apex components. All changes were managed systematically using Change Sets to ensure controlled deployment between environments. The Developer Console was used for efficient writing, debugging, and testing of Apex classes and triggers. A structured DevOps workflow was established to support version control, prevent conflicts, and maintain organized testing and deployment processes.

- Configured a dedicated Salesforce Developer Org to safely create and test custom objects, fields, flows, and Apex components.
- Managed all changes systematically using Change Sets to ensure controlled deployment between environments.
- Utilized the Developer Console for efficient writing, debugging, and testing of Apex classes and triggers.
- Established a streamlined DevOps workflow, supporting version control, conflict prevention, and structured testing and deployment.

## **CUSTOMIZATION OF OBJECTS, FIELDS, VALIDATION RULES**

The CRM included the creation of custom objects to capture key business data and support automation:

- Vehicle\_\_c
- Vehicle\_Dealer\_\_c
- Vehicle\_Customer\_\_c
- Vehicle\_Order\_\_c
- Vehicle\_Test\_Drive\_\_c
- Vehicle\_Service\_Request\_\_c

## **KEY FIELDS IN CUSTOM OBJECTS**

Each custom object was designed with essential fields to capture critical business information and support automation, validation, and reporting. These fields ensure accurate tracking of inventory, customer preferences, order details, and scheduling.

- Stock\_Quantity\_\_c
- Status\_\_c
- Dealer\_Location\_\_c
- Preferred\_Vehicle\_Type\_\_c
- Order\_Date\_\_c
- Test\_Drive\_Date\_\_c

The screenshot shows the Salesforce Object Manager home page. At the top, there are tabs for Setup, Home, and Object Manager. A search bar contains the text "vehicle". Below the search bar is a table titled "Object Manager" with the following columns: LABEL, API NAME, TYPE, DESCRIPTION, LAST MODIFIED, and DEPLOYED. The table lists six custom objects:

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Vehicle	Vehicle_c	Custom Object		11/25/2025	✓
Vehicle Customer	Vehicle_Customer__c	Custom Object		11/25/2025	✓
Vehicle Dealer	Vehicle_Dealer__c	Custom Object		11/25/2025	✓
Vehicle Order	Vehicle_Order__c	Custom Object		11/25/2025	✓
Vehicle Service Request	Vehicle_Service_Request__c	Custom Object		11/25/2025	✓
Vehicle Test Drive	Vehicle_Test_Drive__c	Custom Object		11/25/2025	✓

The screenshot shows the details page for the Vehicle object in the Object Manager. 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, Scoping Rules, and Object Access. The main content area is titled "Vehicle" and shows the following details:

Details	Details
Description	
API Name	Vehicle_c
Custom	✓
Singular Label	Vehicle
Plural Label	Vehicles
Enable Reports	✓
Track Activities	
Track Field History	
Deployment Status	Deployed
Help Settings	Standard salesforce.com Help Window

At the bottom right of the main content area are "Edit" and "Delete" buttons.

philippines.myskillwalletai | Vehicle Customer | Salesforce | Recently Viewed | Vehicle | Auto Assign Dealer - V1 | Test Drive Reminder - V1 | Developer Console

orgfarm-813794a186-dev-ed.develop.my.salesforce-setup.com/lightning/setup/ObjectManager/01gL000003CPaP/Details/view

Setup Home Object Manager

SETUP > OBJECT MANAGER  
Vehicle Customer

**Details**

Description

API Name  
Vehicle\_Customer\_\_c  
Custom  
✓  
Singular Label  
Vehicle Customer  
Plural Label  
Vehicle Customers

Enable Reports  
✓  
Track Activities

Track Field History

Deployment Status  
Deployed  
Help Settings  
Standard salesforce.com Help Window

Edit Delete

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  
Scoping Rules  
Object Access

philippines.myskillwalletai | Vehicle Dealer | Salesforce | Recently Viewed | Vehicle | Auto Assign Dealer - V1 | Test Drive Reminder - V1 | Developer Console

orgfarm-813794a186-dev-ed.develop.my.salesforce-setup.com/lightning/setup/ObjectManager/01lgL000003CPYn/Details/view

Setup Home Object Manager

SETUP > OBJECT MANAGER  
Vehicle Dealer

**Details**

Description

API Name  
Vehicle\_Dealer\_\_c  
Custom  
✓  
Singular Label  
Vehicle Dealer  
Plural Label  
Vehicle Dealers

Enable Reports  
✓  
Track Activities

Track Field History

Deployment Status  
Deployed  
Help Settings  
Standard salesforce.com Help Window

Edit Delete

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  
Scoping Rules  
Object Access

philippines.myskillwalletai | Vehicle Order | Salesforce | Recently Viewed | Vehicle | Auto Assign Dealer - V1 | Test Drive Reminder - V1 | Developer Console

orgfarm-813794a186-dev-ed.develop.my.salesforce-setup.com/lightning/setup/ObjectManager/01gL000003CPdd/Details/view

Setup Home Object Manager

SETUP > OBJECT MANAGER

**Vehicle Order**

**Details**

Description

API Name  
Vehicle\_Order\_c

Custom

Singular Label  
Vehicle Order

Plural Label  
Vehicle Orders

Enable Reports  
✓

Track Activities

Track Field History

Deployment Status  
Deployed

Help Settings  
Standard salesforce.com Help Window

**Edit Delete**

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

Scoping Rules

Object Access

<https://orgfarm-813794a186-dev-ed.develop.my.salesforce-setup.com/one/one.app#/setup...>

philippines.myskillwalletai | Vehicle Service Request | Salesforce | Recently Viewed | Vehicle | Auto Assign Dealer - V1 | Test Drive Reminder - V1 | Developer Console

orgfarm-813794a186-dev-ed.develop.my.salesforce-setup.com/lightning/setup/ObjectManager/01lgL000003CPIT/Details/view

Setup Home Object Manager

SETUP > OBJECT MANAGER

**Vehicle Service Request**

**Details**

Description

API Name  
Vehicle\_Service\_Request\_c

Custom

Singular Label  
Vehicle Service Request

Plural Label  
Vehicle Service Requests

Enable Reports  
✓

Track Activities

Track Field History

Deployment Status  
Deployed

Help Settings  
Standard salesforce.com Help Window

**Edit Delete**

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

Scoping Rules

Object Access

The screenshot shows the Salesforce Setup interface with the following details:

- Setup** tab selected.
- Object Manager** dropdown selected.
- Vehicle Test Drive** object selected.
- Details** tab selected.
- Description**: API Name - Vehicle\_Test\_Drive\_c, Custom - ✓, Singular Label - Vehicle Test Drive, Plural Label - Vehicle Test Drives.
- Enable Reports**: ✓
- Track Activities**: ✓
- Track Field History**: ✓
- Deployment Status**: Deployed
- Help Settings**: Standard salesforce.com Help Window
- Buttons**: Edit, Delete.
- Left sidebar (Fields & Relationships)** includes: 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, Scoping Rules, Object Access.

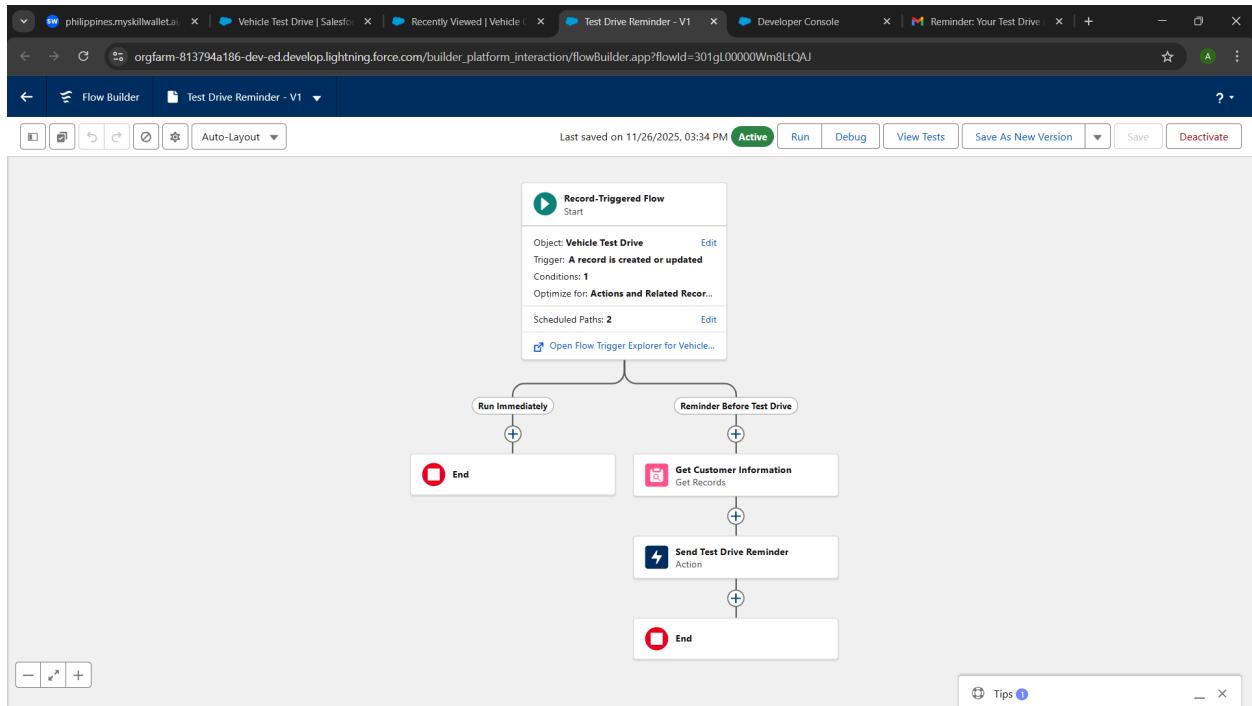
## AUTOMATION

### Record-Triggered Flows

The screenshot shows the Salesforce Flow Builder interface with the following details:

- Flow Builder** tab selected.
- Auto Assign Dealer - V1** flow selected.
- Last saved on**: 11/26/2025, 03:12 PM
- Status**: Active
- Run**, **Debug**, **View Tests**, **Save As New Version**, **Save**, **Deactivate** buttons.
- Flow Details**:
  - Object: Vehicle Order
  - Trigger: A record is created
  - Conditions: 1
  - Optimize for: Actions and Related Records
  - + Add Scheduled Paths (Optional)
  - Open Flow Trigger Explorer for Vehicle...
- Flow Steps**:
  - Start (Record-Triggered Flow)
  - Run Immediately
  - Get Customer Information (Get Records)
  - Get Nearest Dealer (Get Records)
  - Assign Dealer to Order (Update Records)
  - End
- Bottom Buttons**: -, x, +, Tips (1).

## Test Drive Email Reminder



The screenshot shows a Gmail inbox with a search bar set to "in:spam". A single email message is visible, labeled "Reminder: Your Test Drive is Tomorrow! Spam". The message is from "Anyl Febo Uriarte via If7pbq550tpvz3.gI-fyz8uuat.can98.bnc.salesforce.com" and was sent at "10:34 AM (1 hour ago)". The message content is as follows:

Why is this message in spam? This message is similar to messages that were identified as spam in the past.  
Report not spam

Dear User Viana,  
This is a reminder that your test drive a04gL00000AR6ErQAL is tomorrow. If you want a reschedule please contact us at support@gmail.com  
Thank You.

Reply Forward

## APEX CLASSES AND TRIGGERS

The CRM uses Apex components to manage complex logic that cannot be handled by declarative tools alone. These classes and triggers ensure that stock levels, order statuses, and automated processes remain accurate and consistent across the system.

### Apex Trigger Handler

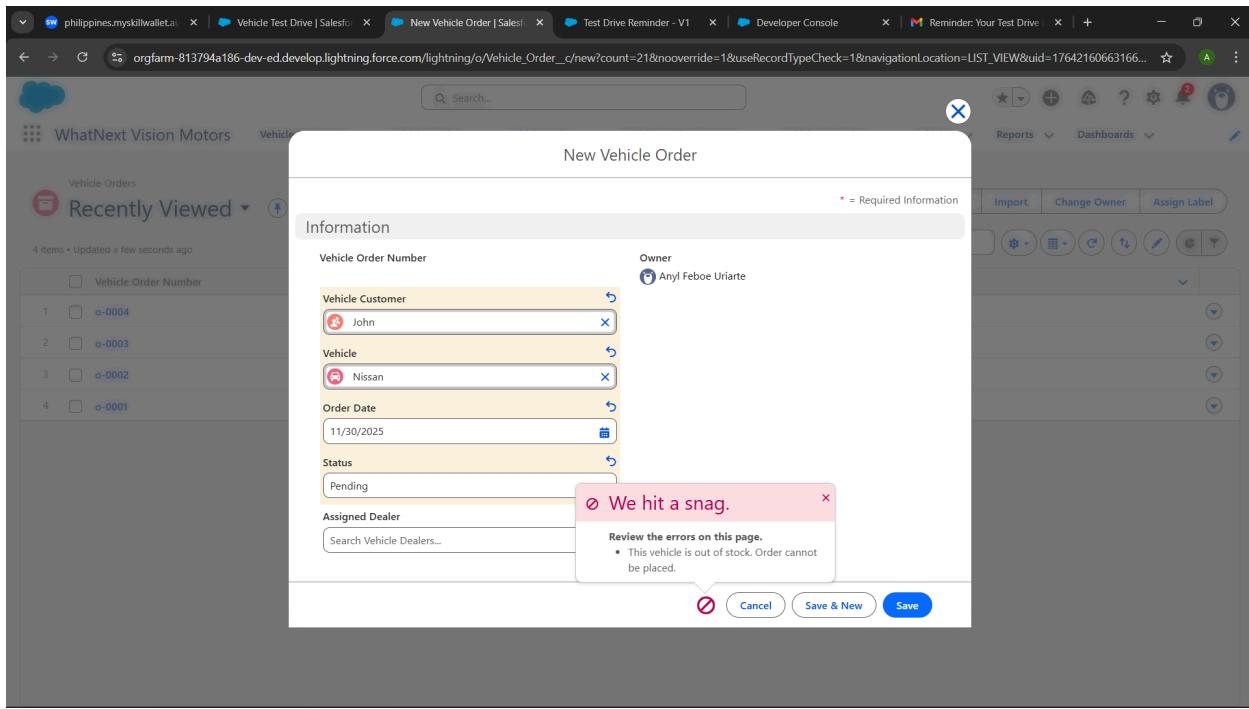
The trigger handler manages core order and stock logic, ensuring data accuracy during record creation and updates.

Handles:

- Preventing orders if the vehicle is out of stock.
- Reducing vehicle stock when order is confirmed.

### Trigger

Executed before and after insert and update events on **Vehicle\_Order\_\_c**, allowing the system to validate requests and update stock in real time.



## Asynchronous Apex

### Batch Apex

- Checks all pending orders.
- If vehicle stock is replenished, updates order to Confirmed.
- Reduces stock automatically.

### Scheduled Apex

- Schedules the batch process to run daily at a defined time.

```
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) {
            preventOrderFromStock(newOrders);
        }
        if (isAfter && isInsert && isUpdate) {
            updateStockOnOrderPlacement(newOrders);
        }
    }
}

// X Prevent placing an order if stock is zero
private void preventOrderFromStock(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 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> vehicleToUpdate = 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;
                vehicleToUpdate.add(vehicle);
            }
        }
        if (vehicleToUpdate.size() > 0) {
            updateVehicleStock(vehicleToUpdate);
        }
    }
}
```

The screenshot shows a browser window with several tabs open, likely from a Salesforce developer environment. The active tab is 'VehicleOrderBatch.apex', which contains the following Apex code:

```
1 global class VehicleOrderBatch implements Database.Batchable<sObject> {
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
17        if (!vehicleIds.isEmpty()) {
18            Map<Id, Vehicle__c> vehicleStockMap = new Map<Id, Vehicle__c>(
19                [SELECT Id, Stock_Quantity__c FROM Vehicle__c WHERE Id IN :vehicleIds]
20            );
21
22            List<Vehicle_Order__c> ordersToUpdate = new List<Vehicle_Order__c>();
23            List<Vehicle__c> vehiclesToUpdate = new List<Vehicle__c>();
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 }
```

The screenshot shows the Salesforce Developer Console interface. At the top, there are several tabs: 'philippines.myskillwallet...', 'Vehicle Test Drive | Salesfo...', 'New Vehicle Order | Salesfo...', 'Test Drive Reminder - V1', 'Developer Console', and 'Reminder: Your Test Drive...'. Below the tabs, the main area displays two Apex files:

```
VehicleOrderBatch.apxc VehicleOrderTriggerHandler.apxc
```

The code for `VehicleOrderBatch.apxc` is:

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

The code for `VehicleOrderTriggerHandler.apxc` is:

```
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);
}
```

At the bottom, there is a 'Logs, Tests, and Problems' section. The 'Logs' tab is selected, showing a table with columns: User, Application, Operation, Time, Status, and Read. There are no logs listed.

## **LIGHTNING APP SETUP**

The Lightning App was designed to provide users with organized navigation and easy access to all major CRM modules. The layout supports efficient workflows for administrators, sales staff, and dealers.

Created “WhatsNext Vision Motors” App:

- Navigation Items: Vehicles, Vehicle Dealers, Vehicle Customers, Vehicle Orders, Vehicle Test Drives, Vehicle Service Request, Reports, Dashboards

## **PAGE LAYOUTS AND DYNAMIC FORMS**

Page layouts and dynamic forms were configured to tailor data visibility and editing capabilities based on user responsibilities, improving usability and maintaining data security.

- Customized layouts for each object.
- Displayed fields based on user roles.
- Conditional visibility applied where required.

## **USER MANAGEMENT**

User access was managed through profile assignments and app permissions to ensure appropriate levels of system access.

- Assigned profiles (System Administrator).
- Configured app access for users.

## **REPORTS AND DASHBOARDS**

Reports and dashboards were created to provide insights into stock levels, dealer performance, and test drive activity.

- Vehicle Stock Report
- Dealer Assignment Report
- Test Drive Summary

## **LIGHTNING PAGES**

Custom Lightning record pages were designed to improve data organization and help users quickly view related records and actions.

- Custom record pages for Vehicle, Dealer, Customer.
- Related lists and quick actions added.

The screenshot shows a list of vehicle dealers. At the top, there is a search bar labeled "Search..." and a toolbar with icons for star, plus, cloud, question mark, settings, bell, and a profile icon. Below the toolbar, the navigation menu includes "Vehicle Customers", "Vehicle Dealers" (which is selected), "Vehicle Orders", "Vehicle Service Requests", "Vehicle Test Drives", "Vehicles", "Reports", and "Dashboards". The main content area is titled "Recently Viewed" and shows a list of 5 items, all updated a few seconds ago. The list includes:

	Vehicle Dealer Name
1	Lyna
2	josh
3	Juan
4	Jane
5	Bill

At the bottom right of the list, there are several icons for actions like New, Import, Change Owner, and Assign Label.

The screenshot shows a list of vehicle orders. The interface is similar to the dealer list, with a search bar, toolbar, and navigation menu. The "Vehicle Orders" tab is selected. The main content area is titled "Recently Viewed" and shows a list of 4 items, all updated a few seconds ago. The list includes:

	Vehicle Order Number
1	o-0004
2	o-0003
3	o-0002
4	o-0001

At the bottom right of the list, there are several icons for actions like New, Import, Change Owner, and Assign Label.

philippines.myskillwalletai/dash | Recently Viewed | Vehicle Service Requests | Test Drive Reminder - V1 | Reminder: Your Test Drive is Today

orgfarm-813794a186-dev-ed.develop.lightning.force.com/lightning/o/Vehicle\_Service\_Request\_\_c/list?filterName=\_Recent

WhatNext Vision Motors Vehicle Customers Vehicle Dealers Vehicle Orders Vehicle Service Requests Vehicle Test Drives Vehicles Reports Dashboards

Vehicle Service Requests

Recently Viewed

0 items • Updated a few seconds ago



Nothing to see here

There's nothing in your list yet. Try adding a new record.

philippines.myskillwalletai/dash | Recently Viewed | Vehicle Test Drives | Test Drive Reminder - V1 | Reminder: Your Test Drive is Today

orgfarm-813794a186-dev-ed.develop.lightning.force.com/lightning/o/Vehicle\_Test\_Drive\_\_c/list?filterName=\_Recent

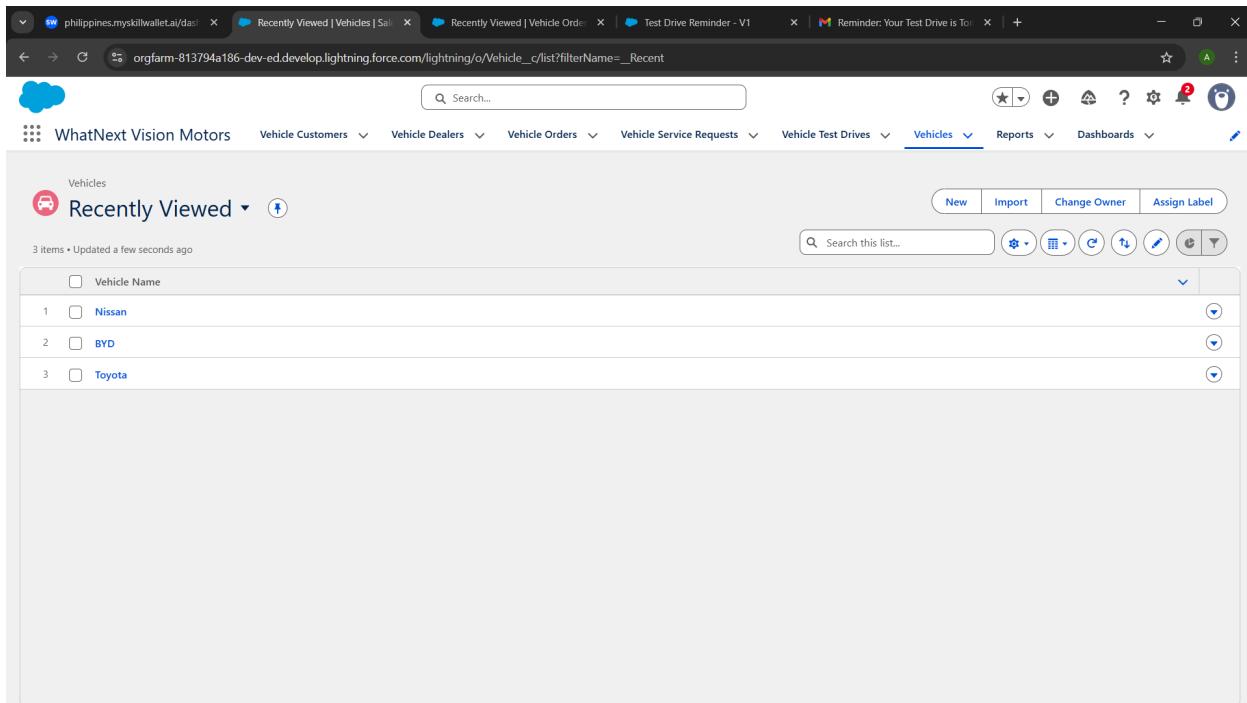
WhatNext Vision Motors Vehicle Customers Vehicle Dealers Vehicle Orders Vehicle Service Requests Vehicle Test Drives Vehicles Reports Dashboards

Vehicle Test Drives

Recently Viewed

2 items • Updated a few seconds ago

	Vehicle Test Name
1	<input type="checkbox"/> zz
2	<input type="checkbox"/> xyz



## DATA LOADING PROCESS

Sample data was added to support system testing and demonstrate functionality. For larger data imports, Salesforce tools such as the Data Import Wizard and Data Loader are also available.

Sample customer, dealer, and vehicle records were added to support testing. In this project, records were created manually through the Salesforce interface, though Salesforce also provides tools such as the Data Import Wizard and Data Loader for importing larger datasets if needed. These tools allow for guided uploads and bulk data entry without writing code.

## **SECURITY CONFIGURATIONS**

Security settings were implemented to ensure proper data protection and structured access across the organization.

The security model was configured to ensure proper data access across the organization. Profiles were set up to control CRUD permissions for each object, while the role hierarchy ensured that managers could automatically view dealer and customer records beneath them. Permission Sets were used to grant additional access to features such as test-drive reminders and reporting tools. Finally, Sharing Rules were implemented to provide controlled visibility for sales teams, ensuring they can only access the data relevant to their responsibilities.

## **FIELD HISTORY TRACKING**

Field history tracking was enabled to monitor changes to important business fields and support audit requirements.

Enabled for:

- Stock\_Quantity\_\_c
- Status\_\_c
- Order\_Date\_\_c

## TESTING PROCESS

Testing focused on validating automation, trigger behavior, and batch processing to ensure the CRM functions reliably across all workflows.

Flow tests were conducted to ensure all automations performed correctly. The dealer assignment flow was validated using before-and-after screenshots, and the test drive reminder flow was confirmed to work by adjusting the test drive dates. Trigger functionality was also verified: when a user attempted to order an out-of-stock vehicle, the correct error message appeared, and when an order was marked as Confirmed, the system automatically reduced the vehicle's stock. Batch Apex behavior was tested as well, confirming that pending orders were successfully auto-confirmed once the stock was updated.

The screenshot shows a Salesforce Lightning interface for a Vehicle Customer record. The top navigation bar includes tabs for Vehicle Customers, Vehicle Dealers, Vehicle Orders, Vehicle Service Requests, Vehicle Test Drives, Vehicles, Reports, and Dashboards. The main content area displays the details for a customer named 'Viana'. The 'Details' tab is selected, showing the following fields:

Field	Value
Vehicle Customer Name	Viana
Email	uriarteanylfelboe@gmail.com
Phone	(123) 456-7890
Address	Batangas
Preferred Vehicle Type	SUV
Created By	Anyl Feboe Uriarte, 11/26/2025, 6:24 PM
Owner	Anyl Feboe Uriarte
Last Modified By	Anyl Feboe Uriarte, 11/26/2025, 6:24 PM

Below the details, there are sections for 'Related' and 'Comments & Activities'. A 'New Contact' button is located at the bottom right of the main content area.

This screenshot shows the 'Vehicle Dealers' section of the Lyna application. The dealer record for 'Lyna' is displayed, owned by Anyl Feboe Uriarte. Key details include the dealer name, location (Batangas), code (DC-0004), phone number (123 456-7890), email (salesforce@gmail.com), and creation date (11/26/2025, 6:25 PM).

Field	Value
Vehicle Dealer Name	Lyna
Dealer Location	Batangas
Dealer Code	DC-0004
Phone	(123) 456-7890
Email	salesforce@gmail.com
Created By	Anyl Feboe Uriarte, 11/26/2025, 6:25 PM

This screenshot shows the 'Vehicle Orders' section of the Lyna application. The vehicle order record for 'o-0004' is displayed, owned by Anyl Feboe Uriarte. Key details include the order number, customer (Viana), vehicle (Nissan), order date (11/30/2025), status (Confirmed), and assigned dealer (Lyna). The creation date is listed as 11/26/2025, 6:28 PM.

Field	Value
Vehicle Order Number	o-0004
Vehicle Customer	Viana
Vehicle	Nissan
Order Date	11/30/2025
Status	Confirmed
Assigned Dealer	Lyna
Created By	Anyl Feboe Uriarte, 11/26/2025, 6:28 PM

philippines.myskillwalletai/dash Request Test Drive 1 | Vehicle Test Drives Recently Viewed | Vehicle Orders Test Drive Reminder - V1 Reminder: Your Test Drive is To... orgfarm-813794a186-dev-ed.develop.lightning.force.com/lightning/r/Vehicle\_Test\_Drive\_\_c/a04g1.00000AR6ErQAL/view

WhatNext Vision Motors Vehicle Customers Vehicle Dealers Vehicle Orders Vehicle Service Requests Vehicle Test Drives Vehicles Reports Dashboards

Vehicle Test Drive Request Test Drive 1

New Contact Edit New Opportunity

Related Details

Vehicle Test Name Request Test Drive 1

Vehicle Customer Viana

Vehicle Nissan

Test Drive Date 11/28/2025

Status Scheduled

Created By Anyl Feboe Uriarte, 11/26/2025, 6:33 PM

Owner Anyl Feboe Uriarte

Last Modified By Anyl Feboe Uriarte, 11/26/2025, 8:22 PM

philippines.myskillwalletai/dash Nissan | Vehicle | Salesforce Recently Viewed | Vehicle Orders Test Drive Reminder - V1 Reminder: Your Test Drive is To... orgfarm-813794a186-dev-ed.develop.lightning.force.com/lightning/r/Vehicle\_\_c/a00g1.00000SoJgsOAF/view

WhatNext Vision Motors Vehicle Customers Vehicle Dealers Vehicle Orders Vehicle Service Requests Vehicle Test Drives Vehicles Reports Dashboards

Vehicle Nissan

New Contact Edit New Opportunity

Related Details

Vehicle Name Nissan

Vehicle Model SUV

Stock Quantity 0

Price \$25,000

Vehicle Dealer Viana

Status Available

Created By Anyl Feboe Uriarte, 11/26/2025, 6:27 PM

Owner Anyl Feboe Uriarte

Last Modified By Anyl Feboe Uriarte, 11/26/2025, 6:29 PM

The screenshot shows a Gmail inbox with a search bar at the top set to "in:spam". There are 47 messages in the inbox, with 45 in the Spam folder. A message from "Anyl Febobe Uriarte" is selected, titled "Reminder: Your Test Drive is Tomorrow!". The message is marked as spam and contains a link to report it. The body of the email says:

Dear User Viana,  
This is a reminder that your test drive a04gL00000AR6ErQAL is tomorrow. If you want a reschedule please contact us at [support@gmail.com](mailto:support@gmail.com)  
Thank You.

Below the message are standard Gmail interaction buttons: Reply, Forward, and a smiley face icon.

The screenshot shows a "New Vehicle Order" page in the Salesforce Lightning interface. The page has fields for "Vehicle Order Number", "Vehicle Customer" (set to "Viana"), "Vehicle" (set to "Nissan"), "Order Date" (set to "11/30/2025"), "Status" (set to "--None--"), and "Assigned Dealer" (with a search bar). A modal window titled "New Vehicle Order" is open, displaying an error message: "We hit a snag." with a red exclamation mark icon. The message says: "Review the errors on this page." followed by a bullet point: "• This vehicle is out of stock. Order cannot be placed." At the bottom of the modal are three buttons: "Cancel", "Save & New", and "Save".

## **DEPLOYMENT STRATEGY**

Deployment followed a controlled and documented process using Change Sets to ensure accurate migration of system components.

- All components deployed using Change Sets:
  - Custom Objects
  - Flows
  - Apex Classes & Triggers
  - Page Layouts
  - Reports and Dashboards

## **MAINTENANCE PLAN**

Ongoing system maintenance ensures accuracy, optimal performance, and alignment with evolving business needs.

- Admin will monitor scheduled jobs.
  - Vehicle stock levels will be reviewed weekly.
  - Flows and triggers will be updated based on new business requirements.
  - Quarterly audits for data accuracy and sharing settings.

## **TROUBLESHOOTING APPROACH**

A structured troubleshooting procedure was established to diagnose issues with automation, Apex logic, and data integrity.

- Use Debug Logs to investigate flow or trigger issues.
- Use Setup → Paused & Failed Flow Interviews to diagnose flow failures.
- Apex exception logs reviewed for batch errors.
- Field history tracking supports data-related issues.

## **CONCLUSION**

The Salesforce CRM developed for WhatsNext Vision Motors delivers a comprehensive and modernized solution for managing the full vehicle purchasing and service lifecycle. The system streamlines operations by automating dealer assignment, enforcing stock validation, generating email reminders, and using batch processes to maintain accurate order statuses. These capabilities reduce manual effort, minimize errors, and increase operational transparency. The CRM also centralizes all vehicle, customer, dealer, order, and service data, enabling informed decision-making through structured reporting and dashboards. With its scalable architecture, secure data model, and optimized workflows, the platform is well-prepared for real-world deployment. It lays a strong foundation for advanced features, broader integrations, and business growth. As the organization evolves, the CRM can support additional channels, predictive analytics, and enhanced customer engagement strategies, ensuring long-term value and adaptability.

## FUTURE ENHANCEMENTS

- Chatbot integration for instant customer support: Implementing a Salesforce-powered chatbot would allow customers to get real-time assistance, ask questions about vehicles, check stock availability, or schedule test drives without waiting for a support agent.
- AI-powered suggestions for best dealer based on traffic or ratings: Using AI, the system could recommend the most suitable dealer for a customer by analyzing factors like dealer ratings, distance, and current traffic conditions, improving convenience and customer satisfaction.
- Integration with inventory management systems: Connecting Salesforce with external inventory systems would ensure that stock levels are always up-to-date and synchronized across all platforms, preventing stock discrepancies and improving order accuracy.
- Mobile App extension via Salesforce Mobile Publisher: Extending the CRM to a mobile application would enable sales reps and managers to access records, manage orders, and monitor stock on the go, providing flexibility and enhancing operational efficiency.