



REPUBLIC OF THE PHILIPPINES
POLYTECHNIC UNIVERSITY OF THE PHILIPPINES
SAN JUAN CAMPUS
SAN JUAN CITY, METRO MANILA



Salesforce Project

Documentation

Submitted by:
MARTINEZ, JOHN LOUISE E.

Project Overview

I. Introduction

WhatNext Vision Motors is working to reshape mobility with a commitment to innovative automotive solutions. To support its growing market and to better engage customers, this organization implemented a Salesforce CRM designed to automate internal workflows. The system focuses on dealer assignment automation, stock-aware ordering, and dynamic order status updates that enable a smooth and reliable customer experience.

II. Objectives

The design of the CRM system was done to meet the following objectives:

- Improve the customer journey by automating the process of selecting the closest dealer.
 - Ensure customers only place orders for vehicles in inventory.
 - Use intelligent, scheduled automation to maintain the accuracy of order statuses.
 - Increase operational capacity by reducing manual tasks and dependency on staff interventions.
- Develop a sound platform with which future technological advancements can be furthered.

PHASE 1: REQUIREMENT ANALYSIS & PLANNING

1. Understanding Business Requirements

Key issues included customers selecting faraway dealers, confusion caused by out-of-stock vehicle orders, and delayed status updates. To resolve these issues, automation-driven processes were prioritized.

2. Project Scope & Objective Mapping

- Custom data structure for Vehicles, Dealers, and Orders.
- Logic to dynamically compute nearest dealer locations.
- Preventive measures ensuring order validity.
- Time-based automation for order processing.

3. Data & Security Design

Data Model:

- Accounts (Customers)
- Dealers
- Vehicles
- Orders

4. Stakeholder Mapping

Role	Responsibility
Admin	System setup, automation, monitoring
Sales Advisor	Order creation and customer interaction
Dealer Manager	Stock updates and dealer operations
Customer	External ordering process

5. Execution Roadmap

1. Business analysis
2. Object creation
3. Flow automation
4. Scheduled processes
5. Testing and deployment

PHASE 2: SALESFORCE DEVELOPMENT – BACKEND & CONFIGURATIONS

1. Environment Setup

Development was executed in a sandbox with change tracking.

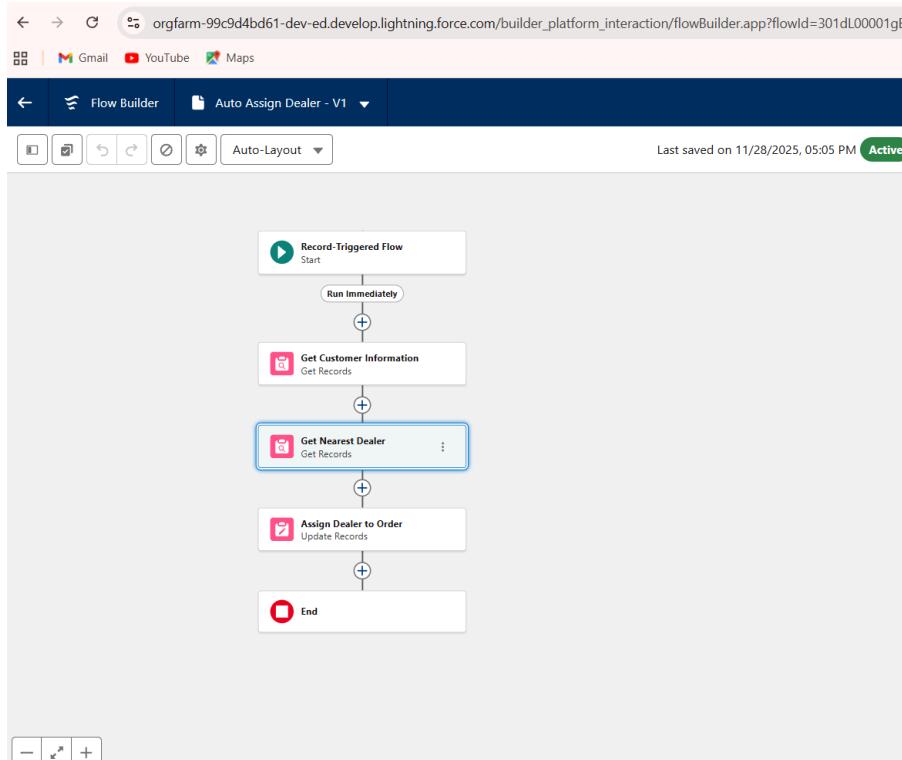
2. Customization Work

- **Objects:** Dealer, Vehicle Inventory, Customer Order.

- **Key Fields:** Distance, Stock Level, Order Status.

3. Automation Processes

- **Record-Triggered Flow:** Calculates nearest dealer.



- **Validation Rule:** Prevents order creation if Stock Level < 1.

[Insert Screenshot: Stock Validation Rule]

The screenshot shows a 'New Vehicle Order' page in the Salesforce interface. The 'Information' section contains the following fields:

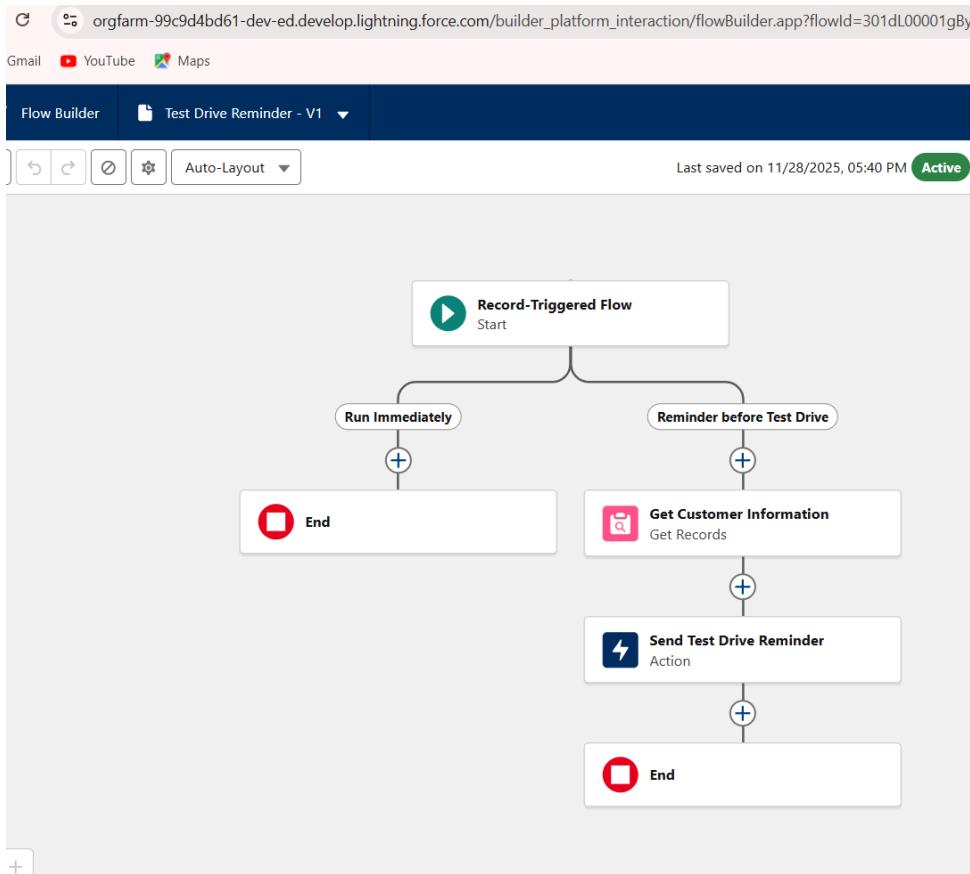
- Vehicle Order Number: O-0004
- Vehicle Customer: Lee
- Vehicle: Fortuner
- Order date: 11/30/2025
- Status: Pending
- Assigned Dealer: Search Vehicle Dealers...

A validation error message box is overlaid on the page, containing the following text:

We hit a snag.
Review the errors on this page.
This vehicle is out of stock. Order cannot be placed.

- **Scheduled Flow:** Updates order statuses according to stock conditions.

[Insert Screenshot: Order Status Scheduled Flow]



4. Apex Components

Used for small-scale stock synchronization logic.

```
orgfarm-99c9d4bd61-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCsPage
File Edit Debug Test Workspace Help < >
VehicleOrderTriggerHandler.apxc VehicleOrderTrigger.apix VehicleOrderBatch.apxc VehicleOrderBatchScheduler.apxc *
Code Coverage: None API Version: 65 Go To
1 public class VehicleOrderTriggerHandler {
2 *
3     public static void handleTrigger(List<Vehicle_Order__c> newOrders, Map<Id, Vehicle_Order__c> oldOrders, Boolean isBefore, Boolean isAfter, Boolean isInsert, Boolean isUpdate) {
4         if (isBefore && (isInsert || isUpdate)) {
5             preventOrderIfOutOfStock(newOrders);
6         }
7     }
8
9     if (isAfter && (isInsert || isUpdate)) {
10         updateStockOnOrderPlacement(newOrders);
11     }
12 }
13
14 //stock is 0
15 private static void preventOrderIfOutOfStock(List<Vehicle_Order__c> orders) {
16     Set<Id> vehicleIds = new Set<Id>();
17     for (Vehicle_Order__c order : orders) {
18         if (order.Vehicle__c != null) {
19             vehicleIds.add(order.Vehicle__c);
20         }
21     }
22
23     if (!vehicleIds.isEmpty()) {
24         Map<Id, Vehicle__c> vehicleStockMap = new Map<Id, Vehicle__c>(
25             [SELECT Id, Stock_Quantity__c FROM Vehicle__c WHERE Id IN :vehicleIds]
26         );
27
28         for (Vehicle_Order__c order : orders) {
29             if (order.Vehicle__c != null && vehicleStockMap.get(order.Vehicle__c).Stock_Quantity__c < order.Quantity__c) {
30                 order.Status__c = 'Out of Stock';
31             }
32         }
33     }
34 }
35 }
```

PHASE 3: UI/UX DEVELOPMENT & CUSTOMIZATION

1. Lightning App Configuration

A custom branded Vision Motors workspace was created.

The screenshot shows the 'Vehicle Customers' page within the 'WhatNext Vision Motors' workspace. The top navigation bar includes links for Vehicle Customers, Vehicle Dealers, Vehicle Orders, Vehicle Service Requests, Vehicle Test Drives, Vehicles, Reports, and Dashboards. A search bar at the top left and a toolbar with various icons are also present. The main content area displays a list titled 'Recently Viewed' with two items: 'Lee' and 'Jelo Martinez'. A note indicates '2 items • Updated a few seconds ago'. Below this is a table header for 'Vehicle Customer Name' with columns for selection, name, and dropdown arrows. Two rows are shown: '1 Lee' and '2 Jelo Martinez'.

2. Page Layout Enhancements

- Dealer Layout shows Service Area and Location.
- Order Layout shows nearest dealer suggestion.

The screenshot shows the 'Vehicle Orders' page within the 'WhatNext Vision Motors' workspace. The top navigation bar is identical to the previous page. The main content area displays a list titled 'Recently Viewed' with three items: 'O-0003', 'O-0002', and 'O-0001'. A note indicates '3 items • Updated a few seconds ago'. Below this is a table header for 'Vehicle Order Number' with columns for selection and dropdown arrows. Three rows are shown: '1 O-0003', '2 O-0002', and '3 O-0001'.

3. Dynamic Forms

The vehicle object uses dynamic sections for technical specifications.

4. Reports/Dashboards

- Inventory Overview Dashboard
- Dealer Performance Report

The screenshot shows the 'Reports' section of the WhatNext Vision Motors application. The left sidebar has categories: REPORTS (Recent, Created by Me, Private Reports, Public Reports, All Reports), FOLDERS (All Folders, Created by Me, Shared with Me), and FAVORITES (All Favorites). The main area displays a table of reports with columns: Report Name, Description, Folder, Created By, Created On, and Subscribed. There are six reports listed, all created by 'Automated Process' on 11/24/2025, 5:47 AM. The reports are: 'Exercise Completion by Days to Complete', 'Exercise Completion by Days to Complete', 'Exercise Completion by User', 'Exercise Completion by User', 'Exercise Completion Status', and 'Exercise Completion Status by Section'. Each report has a description and a folder name like 'Enablement Dashboard Reports Summer '24'.

Report Name	Description	Folder	Created By	Created On	Subscribed
Exercise Completion by Days to Complete	Analyze an exercise's completion status based on average days to complete.	Enablement Dashboard Reports Summer '24	Automated Process	11/24/2025, 5:47 AM	
Exercise Completion by Days to Complete	Analyze an exercise's completion status based on average days to complete.	Enablement Dashboard Reports Spring '24	Automated Process	11/24/2025, 5:47 AM	
Exercise Completion by User	Analyze how long users are spending on an exercise and the amount of progress they're making.	Enablement Dashboard Reports Summer '24	Automated Process	11/24/2025, 5:47 AM	
Exercise Completion by User	Analyze how long users are spending on an exercise and the amount of progress they're making.	Enablement Dashboard Reports Spring '24	Automated Process	11/24/2025, 5:47 AM	
Exercise Completion Status	Analyze how long users take to complete an exercise and their completion percentage.	Enablement Dashboard Reports Spring '24	Automated Process	11/24/2025, 5:47 AM	
Exercise Completion Status	Analyze how long users take to complete an exercise and their completion percentage.	Enablement Dashboard Reports Summer '24	Automated Process	11/24/2025, 5:47 AM	
Exercise Completion Status by Section	Analyze an exercise's completion status based on program sections.	Enablement Dashboard Reports Summer '24	Automated Process	11/24/2025, 5:47 AM	

PHASE 4: DATA MIGRATION, TESTING & SECURITY

1. Data Loading

Used Data Loader for bulk vehicle inventory.

2. Security Controls

- Matching rules prevent duplicate customer entries

3. Roles & Permission Sets

Role hierarchy ensures dealership-level visibility.

4. Testing

Test cases include:

- Correct dealer assignment
- Attempt to order out-of-stock vehicles
- Scheduled status update simulation

5. Apex Test Class

Ensures 95% code coverage for triggers.

```

1  global class VehicleOrderBatch implements Database.Batchable<sObject>
2  {
3
4      global Database.QueryLocator start(Database.BatchableContext bc) {
5          return Database.getQueryLocator([
6              SELECT Id, Status__c, Vehicle__c FROM Vehicle_Order__c WHERE Status__c = 'Pending'
7          ]);
8
9
10     global void execute(Database.BatchableContext bc, List<Vehicle_Order__c> orderList) {
11         Set<Id> vehicleIds = new Set<Id>();
12         for (Vehicle_Order__c order : orderList) {
13             if (order.Vehicle__c != null) {
14                 vehicleIds.add(order.Vehicle__c);
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                 }
31             }
32         }
33     }
34
35     global void finish(Database.BatchableContext bc) {
36     }
37
38     global void reExecute(Database.BatchableContext bc) {
39     }
40
41 }

```

```

1  global class VehicleOrderBatchScheduler implements Schedulable
2  {
3      global void execute(SchedulableContext sc) {
4          VehicleOrderBatch batchJob = new VehicleOrderBatch();
5          Database.executeBatch(batchJob, 50); // 50 = batch size
6      }
7  }

```

PHASE 5: DEPLOYMENT, DOCUMENTATION & MAINTENANCE

1. Deployment

Change Sets used for production deployment.

2. Maintenance Plan

- Monthly stock review
- Regular automation audits
- Flow error monitoring

3. Troubleshooting

Debug logs were used to analyze flow faults, validation failures, and trigger errors.

CONCLUSION

The WhatNext Vision Motors Salesforce CRM successfully delivers a streamlined, automated, and customer-centric system. It supports efficient dealer matching, accurate stock-aware ordering, and timely order processing. The project significantly enhances both internal operations and customer trust, positioning the company for future digital growth powered by scalable Salesforce technology.