

WhatNext Vision Motors

Shaping the Future of Mobility with Innovation and Excellence

Author: Jerico G. Maningas

Date: 11/27/2025

Project Overview:

- So WhatNext Vision Motors runs in the automotive industry and focuses on improving mobility through modern technology and customer solutions. The project focuses on implementing a Salesforce CRM system that can supports the business in managing customers in the way of easily, tracking interactions, and running automated workflows to improve and smooth daily the operations. It also aims to simplify key processes through features like contact and lead management, booking and appointment tracking, automated task creation, approval workflows, and customized dashboards for a better reporting.

To support a comfortable and easy way in ordering experience, the CRM includes smart dealer recommendations, a real time inventory checks, and automatic updates for each stage of the order. and also, the Customers can choose the nearest dealer and place orders only for vehicles that are currently available for that time, making the process well organized and not confusing. With these way, the platform can helps reduce manual work, limit errors, and enhance both operational efficiency and customer satisfaction.

Objectives:

- The main goal of this project is to build a WhatNext Vision Motors CRM that can be simplifies customer interactions and internal processes. By implementing this system, the organization can achieve a better customer relationship management, reduce manual work through automation, and gain a better insights via reports and dashboards. The CRM also supports business growth by providing a expandable solution for tracking leads, bookings, and other critical operations, and also it was important when it comes saving time and improving decision-making.

Phase 1: Requirement Analysis & Planning

Understanding and Manage the Business Requirements:

Through to this discussions with stakeholders, it was identified that some of the organization needed a central system to manage customer data, streamline booking processes, and automate repetitive tasks. The following Challenges including the data redundancy, delayed response times, and lack of reporting capabilities.

Defining Project Scope and Objectives:

- Implementing a fully functional Salesforce CRM for managing leads, contacts, and bookings, because this is the main important part of it.
- Automates the certain task assignments, approvals, and notifications.
- Being enabled the data-driven decision-making through dashboards and reports.
- This can Ensuring that a proper data security and access controls for all users.

Design Data Model and Security Model:

- Customized objects were planned for bookings, feedback, and customer interactions.
- Fields, relationships, and validation rules were designed to maintain data integrity.
- Security model includes profiles, roles, and permission sets to manage access.

Stakeholder Mapping:

- Users: Sales staff, support staff, managers
- Admins: Salesforce administrators responsible for system configuration
- Executives: Monitor reports and dashboards for strategic decisions

This is the Execution RoadMap:

Phase 1: Requirement Analysis & Planning

Phase 2: Backend development and configuration

Phase 3: UI/UX customization

Phase 4: Data migration, testing, and security setup

Phase 5: Deployment and maintenance

Phase 2: Salesforce Development - Backend & Configurations

Setup Environment & DevOps Workflow:

- So the Salesforce sandbox environment was set up for development and testing.
- Deployment and version control workflows were defined using change sets.

Customization:

- Custom objects, fields, and validation rules were created to fit the business process.
- The Workflow rules, Process Builder, and Flows were implemented to automating the repetitive tasks.
- Apex Classes and Triggers were developed where automation required more complex logic.

CUSTOMIZATION OF OBJECTS

Vehicle Customer
Jeric

Related Details

Vehicle Customer Name
Jeric

Owner
Jerico Maningas

Email
jericoguilas1@gmail.com

Phone
123456789

Address
Philippines

Preferred Vehicle Type
Sedan

Created By
Jerico Maningas, 11/25/2025, 8:18 AM

Last Modified By
Jerico Maningas, 11/25/2025, 8:18 AM

Vehicle Dealer
Je

Related Details

Vehicle Dealer Name
Je

Owner
Jerico Maningas

Dealer Location
Philippines

Dealer Code
DC-0001

Phone
123456789

Email
abc@gmail.com

Created By
Jerico Maningas, 11/25/2025, 8:19 AM

Last Modified By
Jerico Maningas, 11/25/2025, 8:19 AM

https://orgfarm-e2976c9ff7-dev-ed.develop.lightning.force.com/lightning/r/Vehicle_Order__c/a03gl00000114BVQAZ/view

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

Vehicle Order O-0002

New Contact Edit New Opportunity

Related Details

Vehicle Order Number
O-0002

Vehicle Customer
[Jeric](#)

Vehicle
[Honda](#)

Order date
11/27/2025

Status
Confirmed

Assigned Dealer

Created By
[Jerico Maningas](#), 11/25/2025, 9:05 AM

Last Modified By
[Jerico Maningas](#), 11/25/2025, 9:05 AM

28°C Mostly cloudy 8:00 pm 26/11/2025

https://orgfarm-e2976c9ff7-dev-ed.develop.lightning.force.com/lightning/r/Vehicle_Service_Request__c/a05gl00000Avdh7QAB/view

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

Vehicle Service Request abc

New Contact Edit New Opportunity

Related Details

Vehicle Service Request Name
abc

Vehicle Customer
[Jeric](#)

Vehicle
[Honda](#)

Service Date
11/27/2025

Issue Description

Status
Requested

Created By
[Jerico Maningas](#), 11/26/2025, 4:07 AM

Last Modified By
[Jerico Maningas](#), 11/26/2025, 4:07 AM

28°C Mostly cloudy 8:00 pm 26/11/2025

https://orgfarm-e2976c9ff7-dev-ed.develop.lightning.force.com/lightning/r/Vehicle_Test_Drive__c/a04g100000AQN01QAH/view

WhatNext Vision Motors

Vehicle Test Drive abc

Related Details

Vehicle Test Drive Name abc Owner Jerico Maningas

Vehicle Customer Jeric

Vehicle Honda

Test Drive Date 11/27/2025

Status Scheduled

Created By Jerico Maningas, 11/25/2025, 8:56 AM

Last Modified By Jerico Maningas, 11/26/2025, 2:11 AM

28°C Mostly cloudy 8:00 pm 26/11/2025

https://orgfarm-e2976c9ff7-dev-ed.develop.lightning.force.com/lightning/r/Vehicle__c/a00g100000QQ8OQAX/view

WhatNext Vision Motors

Vehicles Honda

Related Details

Vehicle Name Honda Owner Jerico Maningas

Vehicle Model EV

Stock Quantity 0

Price \$0.000

Vehicle Dealer WW

Status Available

Created By Jerico Maningas, 11/25/2025, 8:21 AM

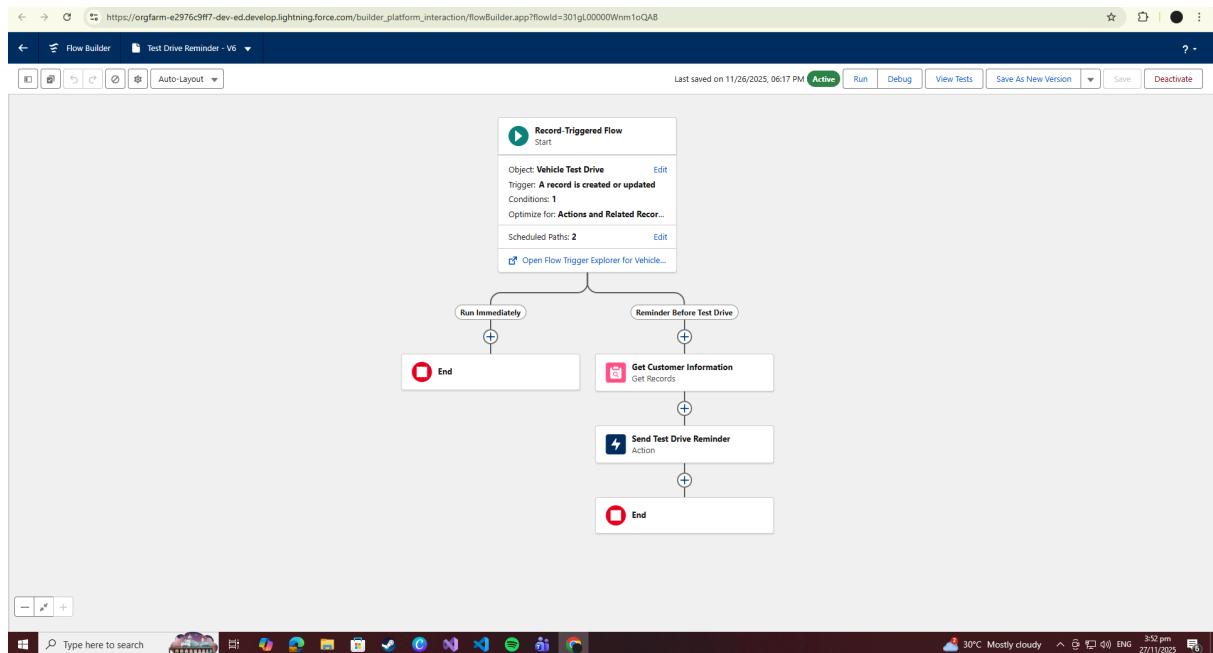
Last Modified By Jerico Maningas, 11/26/2025, 4:08 AM

28°C Mostly cloudy 8:00 pm 26/11/2025

FIELDS

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)		✓

FLOW



APEX CLASSES

- For the WhatNext Vision Motors project, I used the Apex Classes and Triggers to handle parts of the system that standard Salesforce tools that couldn't cover. The Apex Classes contain the easy way logic needed for tasks like checking data, updating records, and supporting some of the custom processes in the CRM. Meanwhile, the Triggers will run automatically when the certain records are created or being

updated, so when it was in need to prevent errors and creating follow up actions when needed. These two are important and worked together to fill the gaps that Flows and automation couldn't handle, making the system more reliable and consistent.

<https://orgfarm-e2975c9f7-dev-ed.develop.my.salesforce-setup.com/lightning/setup/ApexClasses/page?address=%2F01pgl000007NKf>

Setup Home Object Manager

Search Setup

Apex Classes

VehicleOrderBatch

Apex Class Detail

Name: VehicleOrderBatch

Namespace Prefix:

Created By: Jérôme Maninas 11/25/2025, 9:07 AM

Status: Active

Code Coverage: 0% (0/2)

Last Modified By: Jérôme Maninas 11/25/2025, 9:08 AM

Class Body

```

1 global class VehicleOrderBatch implements Database.Batchable<Object> {
2     global Database.QueryLocator start(Database.BatchableContext bc) {
3         return Database.getQueryLocator(
4             'SELECT Id, Status__c, Vehicle__c FROM Vehicle_Order__c WHERE Status__c = \'Pending\''
5         );
6     }
7
8     global void execute(Database.BatchableContext bc, List<Vehicle_Order__c> orderList) {
9         Set<Id> vehicleIds = new Set<Id>();
10        for (Vehicle_Order__c order : orderList) {
11            if (order.Vehicle__c != null) {
12                vehicleIds.add(order.Vehicle__c);
13            }
14        }
15
16        if (vehicleIds.isEmpty()) {
17            Map<Id, Vehicle__c> vehicleStockMap = new Map<Id, Vehicle__c>(
18                [SELECT Id, Stock_Quantity__c FROM Vehicle__c WHERE Id IN :vehicleIds]
19            );
20
21        List<Vehicle_Order__c> ordersToUpdate = new List<Vehicle_Order__c>();
22        List<Vehicle__c> vehiclesToUpdate = new List<Vehicle__c>();
23
24        for (Vehicle_Order__c order : orderList) {
25            Vehicle__c vehicle = vehicleStockMap.get(order.Vehicle__c);
26            if (vehicle != null && vehicle.getStockQuantity() > 0) {
27                if (order.Status__c != 'Confirmed' && vehicle.getStockQuantity() > 0) {
28                    order.setStatus('Confirmed');
29                    ordersToUpdate.add(order);
30                    vehiclesToUpdate.add(vehicle);
31                }
32            }
33        }
34
35        if (ordersToUpdate.isEmpty()) update ordersToUpdate;
36        if (vehiclesToUpdate.isEmpty()) update vehiclesToUpdate;
37    }
38
39    global void finish(Database.BatchableContext bc) {
40        System.debug('Vehicle order batch job completed');
41    }
42 }
43

```

Class Summary Version Settings Trace Flags

<https://orgfarm-e2975c9f7-dev-ed.develop.my.salesforce-setup.com/lightning/setup/ApexClasses/page?address=%2F01pgl000007NKw>

Setup Home Object Manager

Search Setup

Apex Classes

VehicleOrderBatchScheduler

Apex Class Detail

Name: VehicleOrderBatchScheduler

Namespace Prefix:

Created By: Jérôme Maninas 11/25/2025, 9:08 AM

Status: Active

Code Coverage: 0% (0/3)

Last Modified By: Jérôme Maninas 11/25/2025, 9:09 AM

Class Body

```

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 }

```

Class Summary Version Settings Trace Flags

Help for this Page

```

1  public class VehicleOrderManagement {
2      public static void handleTriggerList(Vehicle_Order__c newOrders, Map<Id, Vehicle_Order__c> oldOrders, Boolean isBefore, Boolean isAfter, Boolean insert, Boolean isUpdate) {
3          if (isBefore && !insert) {
4              preventDeleteOldStock(newOrders);
5          }
6          if (isAfter && !isUpdate) {
7              updateStockOnOrderPlacement(newOrders);
8          }
9      }
10     // ✎ Prevent placing an order if stock is zero
11     private static void preventDeleteOldStock(List<Vehicle_Order__c> orders) {
12         Set<Vehicle_Order__c> vehicles = Set<Vehicle_Order__c> orders;
13         for (Vehicle_Order__c order : orders) {
14             vehicles.add(order.Vehicle__c);
15         }
16     }
17     // ✎ Decrease stock when an order is confirmed
18     private static void updateStockOnOrderPlacement(List<Vehicle_Order__c> orders) {
19         Set<Vehicle__c> vehicles = Set<Vehicle__c> vehicles;
20         for (Vehicle_Order__c order : orders) {
21             if (order.Vehicle__c != null && order.Status__c == 'Confirmed') {
22                 vehicles.add(order.Vehicle__c);
23             }
24         }
25         if (vehicles.isEmpty()) {
26             Map<Id, Vehicle__c> vehicleStockMap = new Map<Id, Vehicle__c>();
27             [SELECT id, Stock_Quantity__c FROM Vehicle__c WHERE id IN :vehicles];
28             for (Vehicle__c vehicle : vehicleStockMap) {
29                 if (vehicle != null && vehicle.Stock_Quantity__c > 0) {
30                     vehicle.setStockQuantity(vehicle.Stock_Quantity__c - 1);
31                     vehicle.setLastUpdatedDate();
32                 }
33             }
34         }
35         if (vehicles.isEmpty()) {
36             List<Vehicle__c> vehiclesToUpdate = new List<Vehicle__c>;
37             for (Vehicle_Order__c order : orders) {
38                 if (order.Vehicle__c != null && order.Status__c == 'Confirmed') {
39                     vehiclesToUpdate.add(order.Vehicle__c);
40                 }
41             }
42             vehicles.add(vehiclesToUpdate);
43         }
44         if (vehicles.isEmpty()) {
45             Map<Id, Vehicle__c> vehicleStockMap = new Map<Id, Vehicle__c>();
46             [SELECT id, Stock_Quantity__c FROM Vehicle__c WHERE id IN :vehicles];
47             for (Vehicle__c vehicle : vehicleStockMap) {
48                 if (vehicle != null && vehicle.Stock_Quantity__c > 0) {
49                     vehicle.setStockQuantity(vehicle.Stock_Quantity__c - 1);
50                     vehicle.setLastUpdatedDate();
51                 }
52             }
53             if (vehiclesToUpdate.isEmpty()) {
54                 vehicles.add(vehiclesToUpdate);
55             }
56         }
57     }
58     if (!vehiclesToUpdate.isEmpty()) {
59         updateStockOnOrderPlacement(vehiclesToUpdate);
60     }
61 }
62

```

Phase 3: UI/UX Development & Customization

Lightning App Setup & Page Customization:

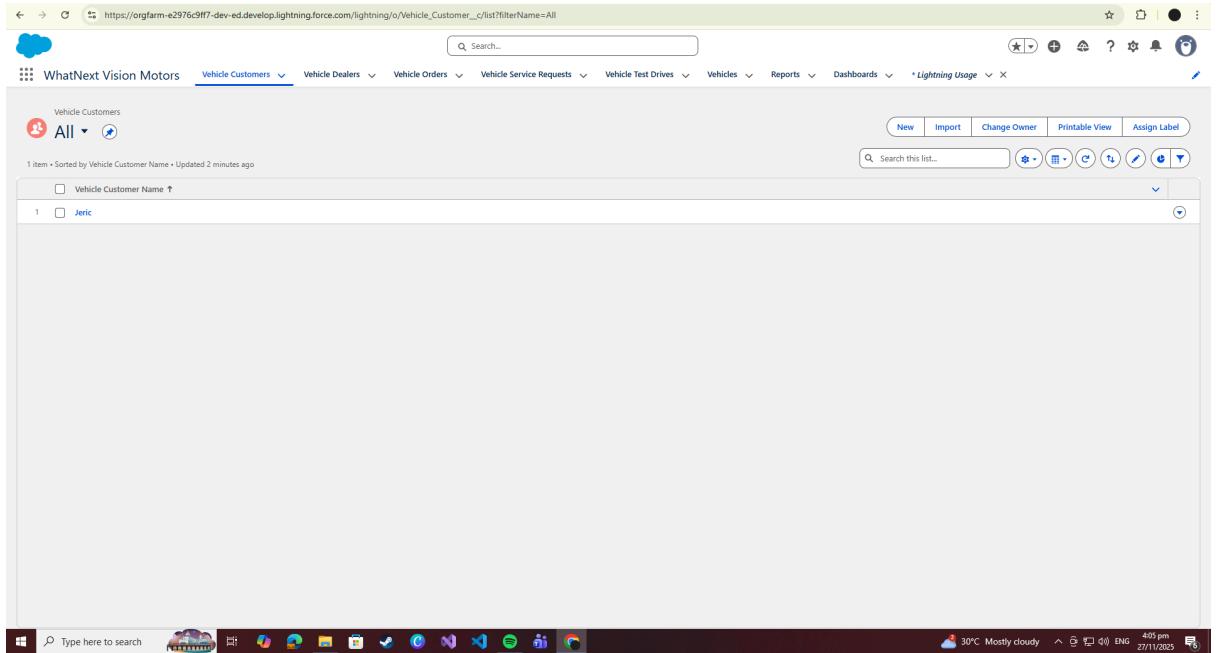
- A Lightning App was created using the App Manager to organize the tabs and components that belong together.
- Page layouts and dynamic forms are designed for an easy to navigate and better for user experience.
- In the user management was implemented to be ensure if the access levels are correct for each profile.

Reports and Dashboards:

- Custom reports are created to monitor such as leads, bookings, and completed tasks.
- The Dashboards are built to provide quick insights to managers and executives or even who have access on it.

LWC Development (Optional):

- Lightning Web Components are also developed for enhanced functionality in certain areas and to do tasks that relate to it.



Phase 4: Data Migration, Testing & Security

Data Loading:

- Customer and booking data were imported using Data Import Wizard and Data Loader.

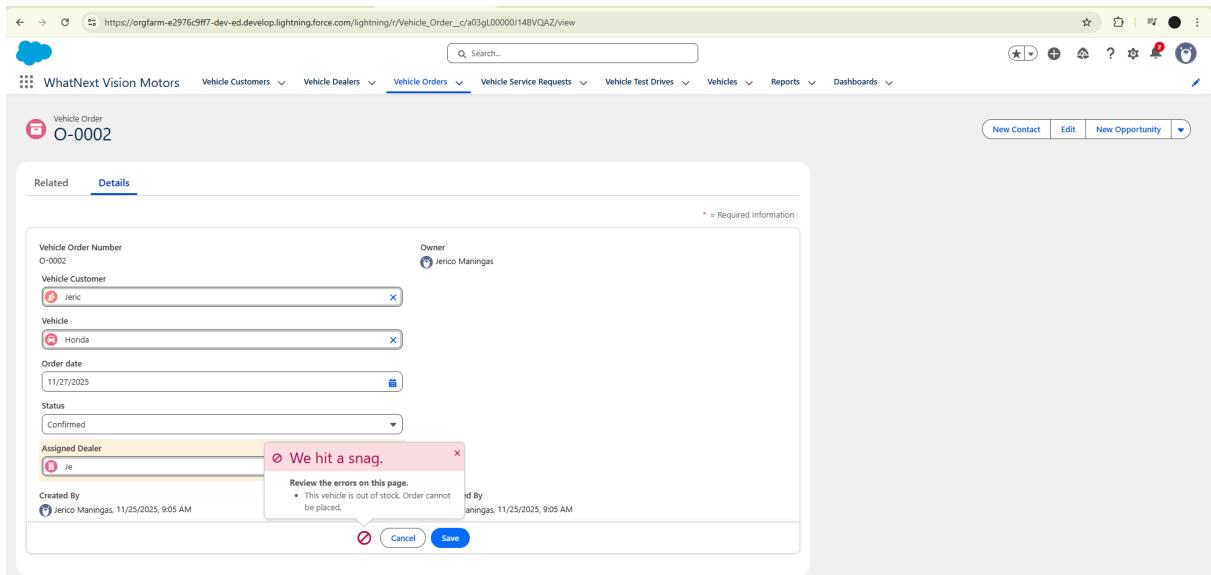
Security Setup:

- Field history tracking, duplicate rules, and matching rules were configured.
- The Profiles, roles, role hierarchy, permission sets, and sharing rules confirmed the proper access control.

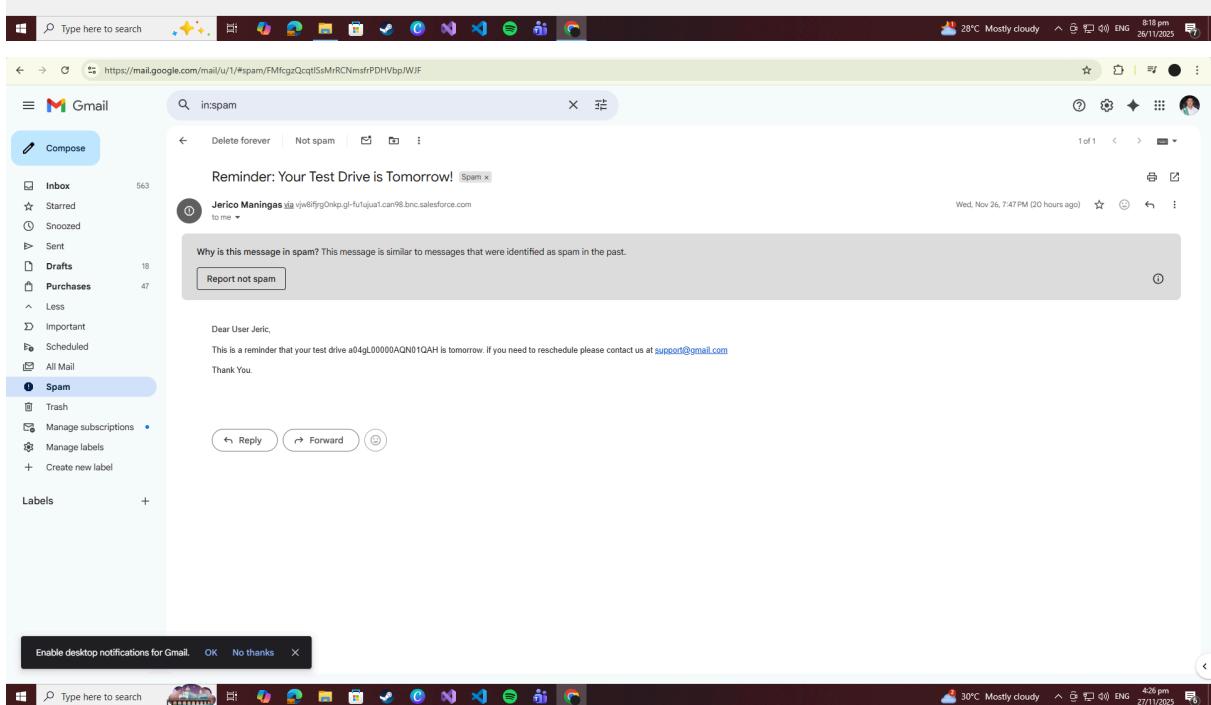
Testing:

- Testing classes are a better option to create validate Apex code functionality.
- Test cases for each feature, such as booking creation, approval processes, task automation, and flows, were documented with screenshots.

Order Creation: out of stock



The screenshot shows a Salesforce Lightning page for a vehicle order. The URL is https://orgfarm-e2976c9ff7-dev-ed.lightning.force.com/lightning/r/Vehicle_Order__c/a03gl0000014BVQAZ/view. The page title is "Vehicle Order O-0002". The "Details" tab is selected. A modal window titled "We hit a snag." displays an error message: "Review the errors on this page. • This vehicle is out of stock. Order cannot be placed." The modal has "Cancel" and "Save" buttons. The status field is set to "Confirmed". The assigned dealer is listed as "Je". The created by field shows "Jerico Maningas, 11/25/2025, 9:05 AM". The owner is "Jerico Maningas". The vehicle is listed as "Honda". The customer is "Jeric". The order date is "11/27/2025".



The screenshot shows a Gmail inbox with the URL https://mail.google.com/mail/u/1/#spam/FMfcgxQcqtlSshMrCNmsfrPDHlVbpjW/f. The subject of the email is "Reminder: Your Test Drive is Tomorrow!". The email is from "Jerico Maningas <view8f9gOnkip-gl-fuJuqsa1can98.bnc.salesforce.com>" and is marked as spam. The body of the email reads: "Dear User Jeric,
This is a reminder that your test drive a04g0.00000AQN01QAH is tomorrow. If you need to reschedule please contact us at support@gmail.com
Thank You." The email was sent on "Wed, Nov 26, 7:47 PM (20 hours ago)". The inbox also shows other items like "Compose", "Inbox" (563), "Starred", "Snoozed", "Sent", "Drafts" (18), "Purchases" (47), "Less", "Important", "Scheduled", "All Mail", and "Spam". A sidebar shows "Labels" and "Manage subscriptions". A notification at the bottom says "Enable desktop notifications for Gmail. Ok No thanks".

TEST DRIVE REMINDER EMAIL

Phase 5: Deployment, Documentation & Maintenance

Deployment Strategy:

- So based on my understanding, the Deployment was properly being executed using change of sets to transfer configurations from the sandbox to the production environment.

System Maintenance:

- Implementing a Regular monitoring of user activity, data integrity, and system logs will ensure smooth operation and prevent errors.
- Troubleshooting guides are being created for common errors and issues, when creating tasks.

Conclusion

So based on the understanding, The WhatNext Vision Motors project brings together a modern technology and practical solutions to improve how the company handles customer orders and daily operations. By building a Salesforce CRM it can supports smart dealer selection, real time stock checking, and automated order updates, also the system can helps reduce manual work and avoid common errors. It can gives both customers and staff a smoother, faster, well organized experience, in making each step of the process was easy to manage.

This CRM also strengthens internal efficiency by offering a way better and clearer reporting, automated tasks, and better tracking of bookings and interactions. With these tools, it can help company can respond faster, make more informed decisions, and maintain a higher level of service quality. Lastly, the project supports WhatNext Vision Motors' goal of delivering a more innovative, accurate, and customer-focused approach to mobility.