



Republic of the Philippines
BULACAN STATE UNIVERSITY
College of Information and Communications Technology



City of Malolos, Bulacan

Virtual Internship Program: Salesforce Developer

Capstone Project: WhatNext Vision Motors: Shaping the Future of Mobility with Innovation and Excellence

Maroma, Cielo P.

November 2025

Table of Contents

Project Overview.....	3
Objectives.....	3
Phase 1: Requirement Analysis & Planning.....	3
Understanding Business Requirements.....	3
Defining Project Scope and Objectives.....	3
Stakeholders Mapping.....	4
Phase 2: Salesforce Development - Backend & Configurations.....	4
Setup Environment.....	4
Custom Objects Created.....	4
Automation - Flows.....	8
Apex Classes & Triggers.....	10
Phase 3: UI/UX Development & Customization.....	12
App Launcher – WhatNext Vision Motors.....	12
Custom Tabs Created.....	13
Phase 4: Data Migration, Testing & Security.....	16
Dealer Assignment Success.....	17
Inventory Reduction Test (100 → 99).....	17
Stock Validation Error.....	19
Email Reminder – Test Drive.....	21
Testing Approach Summary.....	22
Phase 5: Deployment, Documentation & Maintenance.....	23
Deployment Strategy.....	23
Maintenance & Monitoring.....	24
Conclusion.....	24
Future Enhancements.....	24

Project Overview

WhatNext Vision Motors CRM automates vehicle order processing in Salesforce, assigning orders to nearest dealers based on customer location, preventing orders for out-of-stock vehicles, and sending test drive email reminders one day prior to scheduled dates. Key features include custom objects for vehicles, customers, dealers, orders, test drives, and service requests with automation via Flows and Apex triggers.

Business Needs Addressed:

- Streamlined order assignment to local dealers
- Inventory management preventing overselling
- Automated customer notifications for test drives

Objectives

The primary goals are to automate vehicle order processing, optimize dealer assignment based on geography, enforce stock availability logic, and enhance customer experience through timely test drive reminders. These objectives deliver business value by reducing manual processing time, minimizing order errors, improving inventory accuracy, and increasing customer satisfaction through proactive communication.

Phase 1: Requirement Analysis & Planning

Phase 1 focuses on building the structural foundation of the application.

Understanding Business Requirements

- Automate assignment of vehicle orders to geographically nearest dealers
- Prevent order creation for vehicles with zero stock quantity
- Send automated email reminders for test drives 1 day before scheduled date
- Track vehicle inventory levels and update on confirmed orders

Defining Project Scope and Objectives

In Scope:

- 6 Custom Objects: Vehicle, Vehicle Customer, Vehicle Dealer, Vehicle Order, Vehicle Test Drive, Vehicle Service Request
- Record-Triggered Flows for dealer assignment and email reminders
- Apex Triggers and Classes for inventory management
- Lightning App with custom tabs

Out of Scope: Production deployment, advanced reporting, mobile app

Stakeholders Mapping

- System Administrator - Full access to app and objects
- Dealers - View assigned orders
- Customers - Test drive notifications
- Sales Team - Order creation and management

Execution Roadmap

1. Developer Org Setup & Object Creation
2. Field Creation & Tab Configuration
3. Lightning App Development
4. Flow Automation Implementation
5. Apex Development & Testing

Phase 2: Salesforce Development - Backend & Configurations

This phase focuses on Salesforce automation: flows, triggers, and backend logic essential to dealership operations.

Setup Environment

- Salesforce Developer Edition Org created
- Custom Lightning App: WhatNext Vision Motors

Custom Objects Created

Figure 1. Object Manager showing all six custom objects

The screenshot shows the Salesforce Object Manager page. At the top, there's a navigation bar with links like Setup, Home, and Object Manager. Below that is a search bar with the placeholder "Search Setup" and a "vehicle" search term. A "Schema Builder" button and a "Create" button are also visible. The main area is titled "Object Manager" and displays a table of 6 items, sorted by Label. The columns are labeled: LABEL, API NAME, TYPE, DESCRIPTION, LAST MODIFIED, and DEPLOYED. Each row represents a custom object with its details.

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Vehicle	Vehicle__c	Custom Object		11/21/2025	✓
Vehicle Customer	Vehicle_Customer__c	Custom Object		11/21/2025	✓
Vehicle Dealer	Vehicle_Dealer__c	Custom Object		11/21/2025	✓
Vehicle Order	Vehicle_Order__c	Custom Object		11/21/2025	✓
Vehicle Service Request	Vehicle_Service_Request__c	Custom Object		11/21/2025	✓
Vehicle Test Drive	Vehicle_Test_Drive__c	Custom Object		11/21/2025	✓

This screenshot shows the complete list of custom objects created for the system. These objects represent the dealership's operational entities: Vehicles, Dealers, Vehicle Orders, Test Drives, Customer Profiles, and Inventory.

The Object Manager structure ensures scalability and centralized management as the system grows. Below are the detailed Fields & Relationships of the 6 custom objects:

Figure 2. Vehicle Fields & Relationships

The screenshot shows the Salesforce Object Manager interface for the 'Vehicle' object. The left sidebar has 'Fields & Relationships' selected. The main area displays a table titled 'Fields & Relationships' with 9 items. The columns are: FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED.

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)		

Figure 3. Vehicle Customer Fields & Relationships

The screenshot shows the Salesforce Object Manager interface for the 'Vehicle Customer' object. The left sidebar has 'Fields & Relationships' selected. The main area displays a table titled 'Fields & Relationships' with 8 items. The columns are: FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED.

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)		

Figure 4. Vehicle Dealer Fields & Relationships

Vehicle Dealer

Fields & Relationships

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)		
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)		

Figure 5 . Vehicle Order Fields & Relationships

Vehicle Order

Fields & Relationships

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Assigned Dealer	Assigned_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		

Figure 6 . Vehicle Service Request Fields & Relationships

Vehicle Service Request

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)		✓

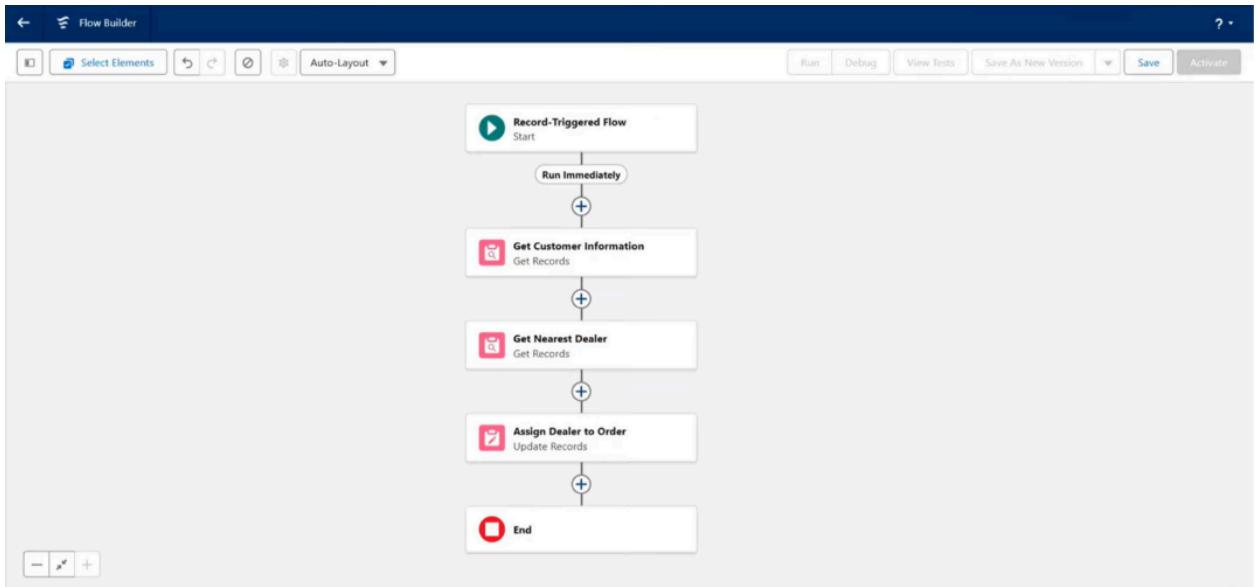
Figure 7 . Vehicle Test Drive Fields & Relationships

Vehicle Test Drive

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)		✓

Automation - Flows

Figure 8 . Auto Assign Dealer Flow

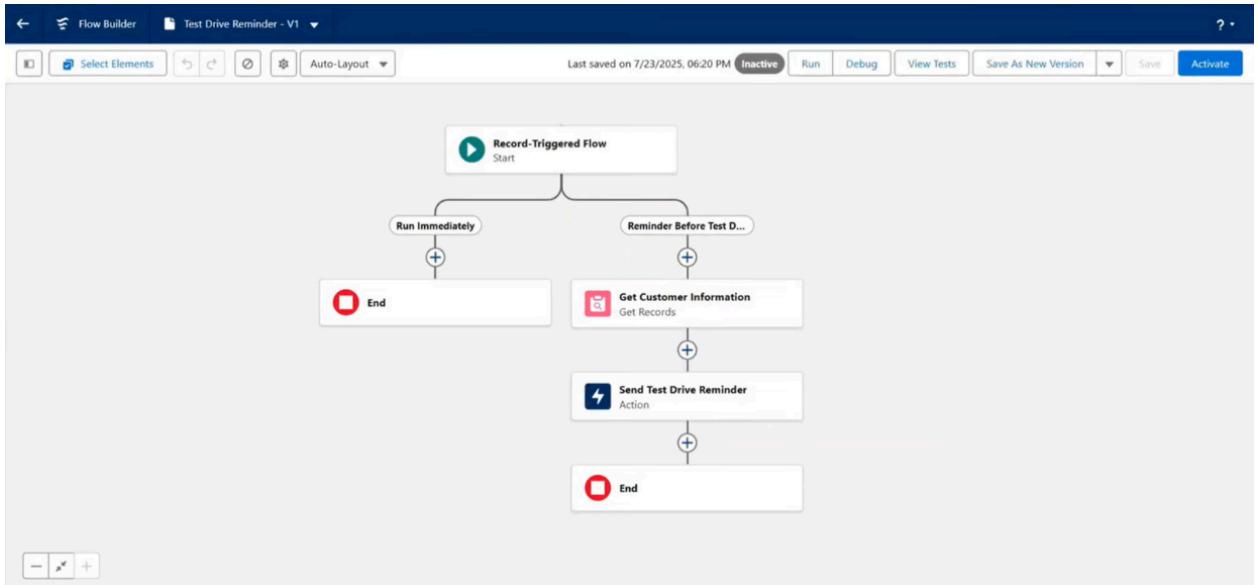


A Record-Triggered Flow automatically assigns a dealer based on the selected vehicle.

Flow Behavior

- Trigger: Vehicle Order creation
- Looks up the Dealer associated with the Vehicle__c
- Automatically populates Assigned_Dealer__c
- Removes manual workload from sales teams

Figure 9 . Test Drive Reminder Flow



A scheduled-path Flow ensures proper customer follow-up.

Flow Behavior

- Trigger: Test Drive record creation
- Scheduled path waits until before the Test Drive Date
- Sends customer reminder email
- Helps increase appointment attendance

Apex Classes & Triggers

To support instant inventory adjustments and reusable logic, an Apex Trigger and Handler were implemented.

Figure 10 . Vehicle Order Trigger Handler

The screenshot shows the Salesforce IDE interface with the code editor open. The file is named 'VehicleOrderTriggerHandler.apex'. The code implements a trigger handler for vehicle orders, specifically for updating vehicle stock quantities. It includes a query to select vehicles by ID, a loop to iterate over orders, and logic to update the vehicle's stock quantity if it is not null and greater than zero. The code follows best practices like bulkification and error handling.

```
File • Edit • Debug • Test • Workspace • Help • < >
VehicleOrderTriggerHandler.apex
Code Coverage: None • API Version: 64 • Go To
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 }
```

This Apex handler contains the business logic for vehicle stock updates. It follows best practices:

- **Separation of concerns** (logic stored in handler)
- **Bulkification** for mass updates
- **Error handling** using try-catch

The handler interacts directly with inventory data to update stock after orders.

Figure 11 . Vehicle Order Trigger

The screenshot shows the Salesforce IDE interface. The top part displays the trigger code:

```

1 * trigger VehicleOrderTrigger on Vehicle_Order__c (before insert, before update, after insert, after update) {
2     VehicleOrderTriggerHandler.handleTrigger(Trigger.new, Trigger.oldMap, Trigger.isBefore, Trigger.isAfter, Trigger.isInsert, Trigger.isUpdate);
3 }

```

The bottom part shows the Logs tab with the following data:

User	Application	Operation	Time	Status	Read	Size
Education Matters	Browser	/aura	7/23/2025, 6:50:57 PM	Success	Unread	7.98 KB
Education Matters	Unknown	common.api.soap.DirectSoap	7/23/2025, 6:50:56 PM	Success	Unread	522 bytes
Education Matters	Browser	/aura	7/23/2025, 6:48:06 PM	Success	Unread	16.63 KB
Education Matters	Unknown	common.api.soap.DirectSoap	7/23/2025, 6:48:06 PM	Success	Unread	522 bytes

This trigger activates when a Vehicle Order is inserted or updated.

Key operations:

- Validates order quantity
- Calls the trigger handler
- Updates Vehicle Inventory
- Ensures no negative stock occurs

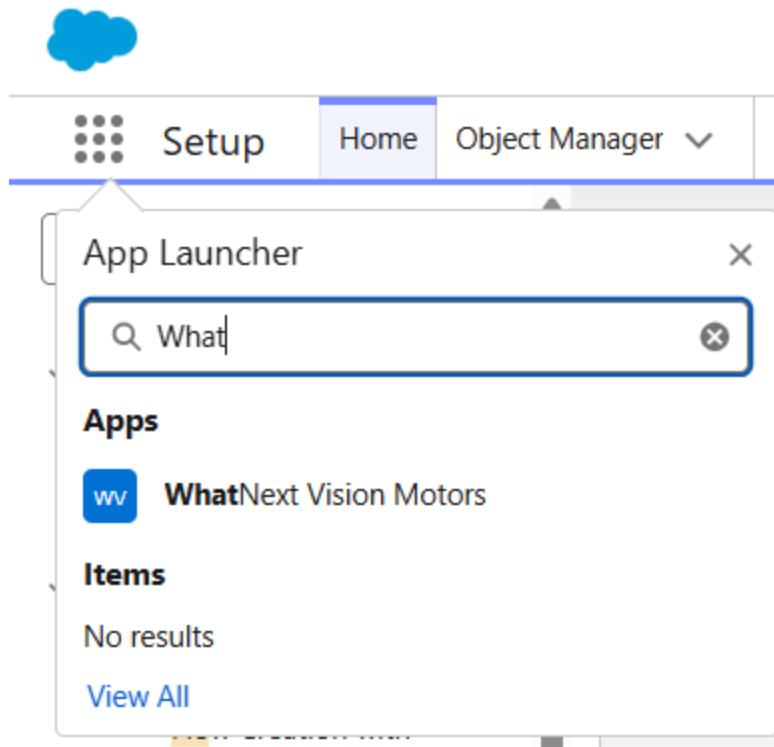
This ensures that dealership stock levels remain accurate.

Phase 3: UI/UX Development & Customization

Phase 3 focuses on creating a clean, accessible, and functional user interface for dealership employees.

App Launcher – WhatNext Vision Motors

Figure 12 . WhatNext Vision Motors App Launcher Entry



The custom Salesforce app consolidates all objects and tools into a dedicated workspace. It improves navigation and workflow speed for employees and managers.

Custom Tabs Created

All six custom objects had dedicated navigation tabs added. This improves usability and makes the CRM intuitive for all employees.

Figure 13. Vehicle Customers Tab

The screenshot shows a web browser window with the URL https://orgfarm-d4f339a47c-dev-ed.lightning.force.com/lightning/o/Vehicle_Customer_c/list?filterName=_Recent. The page title is "Recently Viewed | Vehicle Customers". The main content area displays a list titled "Recently Viewed" under "Vehicle Customers". The list contains two items: "Cielo Maroma" and "Cielo". Each item has a checkbox next to it. At the top right of the list area are buttons for "New", "Import", "Change Owner", and "Assign Label". Below the list is a search bar with the placeholder "Search this list..." and a set of filter icons.

Figure 14. Vehicle Dealers Tab

The screenshot shows a web browser window with the URL https://orgfarm-d4f339a47c-dev-ed.lightning.force.com/lightning/o/Vehicle_Dealer_c/list?filterName=_Recent. The page title is "Recently Viewed | Vehicle Dealers". The main content area displays a list titled "Recently Viewed" under "Vehicle Dealers". The list contains two items: "ZX" and "PH". Each item has a checkbox next to it. At the top right of the list area are buttons for "New", "Import", "Change Owner", and "Assign Label". Below the list is a search bar with the placeholder "Search this list..." and a set of filter icons.

Figure 15. Vehicle Orders Tab

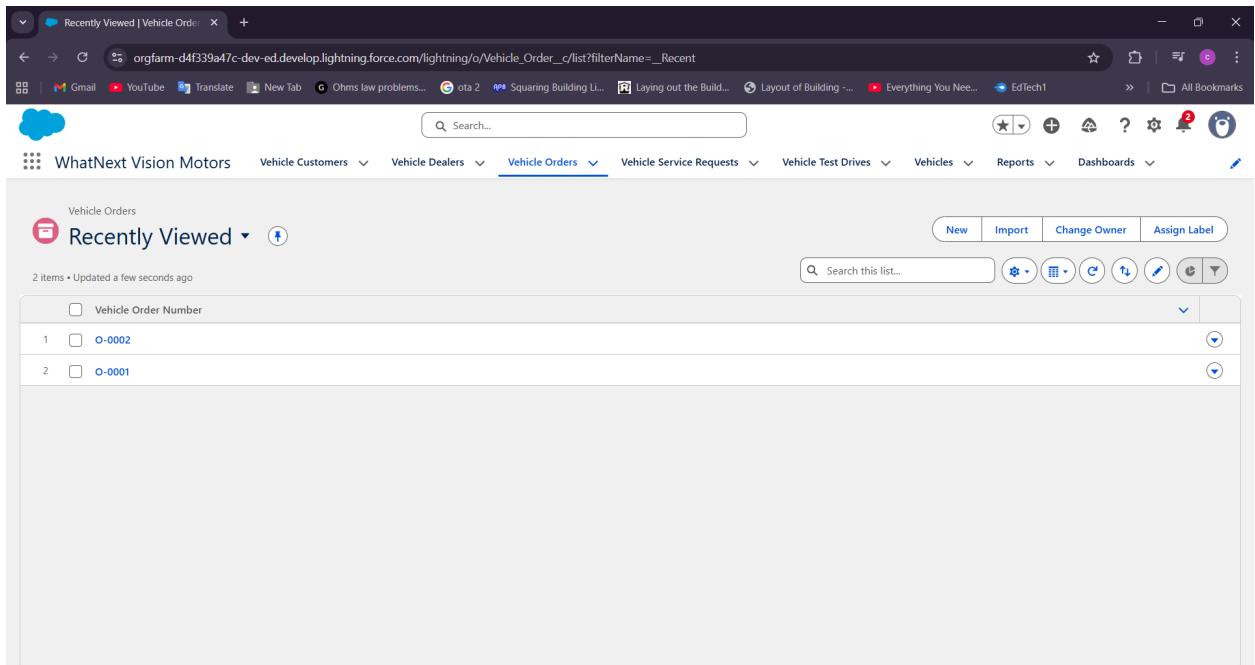


Figure 16. Vehicle Service Requests Tab

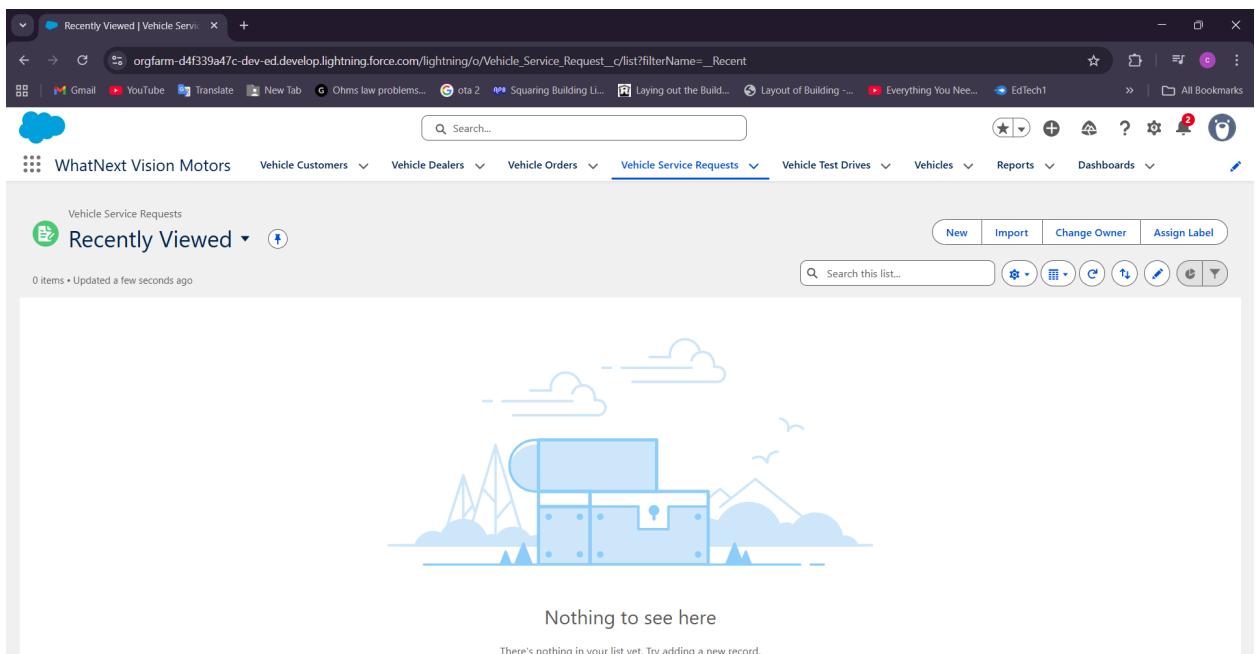


Figure 17. Vehicle Test Drives Tab

The screenshot shows the 'Vehicle Test Drives' tab selected in the top navigation bar. The main content area displays a list titled 'Recently Viewed' with two items: 'xyz' and 'abc'. The interface includes a search bar, filter options, and standard list management buttons.

Vehicle Test Drive Name
1 xyz
2 abc

Figure 18. Vehicles Tab

The screenshot shows the 'Vehicles' tab selected in the top navigation bar. The main content area displays a list titled 'Recently Viewed' with two items: 'Honda' and 'Mitsubishi'. The interface includes a search bar, filter options, and standard list management buttons.

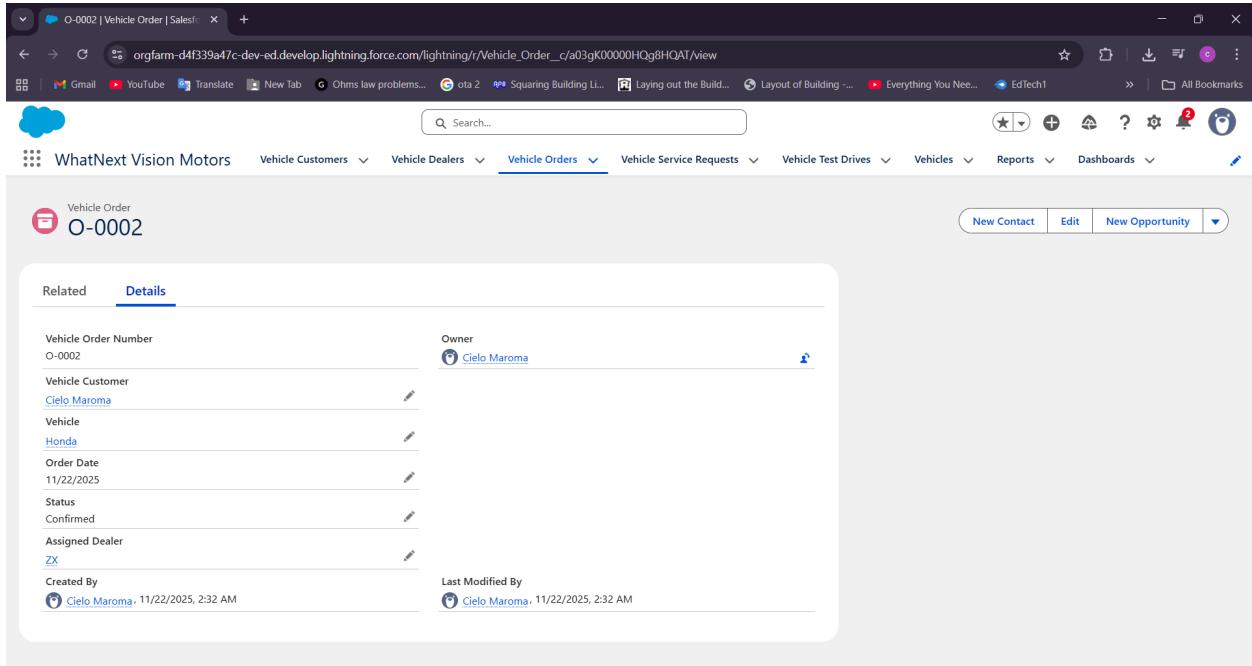
Vehicle Name
1 Honda
2 Mitsubishi

Phase 4: Data Migration, Testing & Security

Testing validates the system's functionality, automation integrity, and user experience

Dealer Assignment Success

Figure 19. Successful dealer assignment result



This figure shows the Auto-Assign Dealer Flow functioning correctly. After creating a vehicle order, the correct dealer was automatically populated.

Inventory Reduction Test (100 → 99)

Figure 20. 100 Stock Quantity Before Creating Vehicle Order

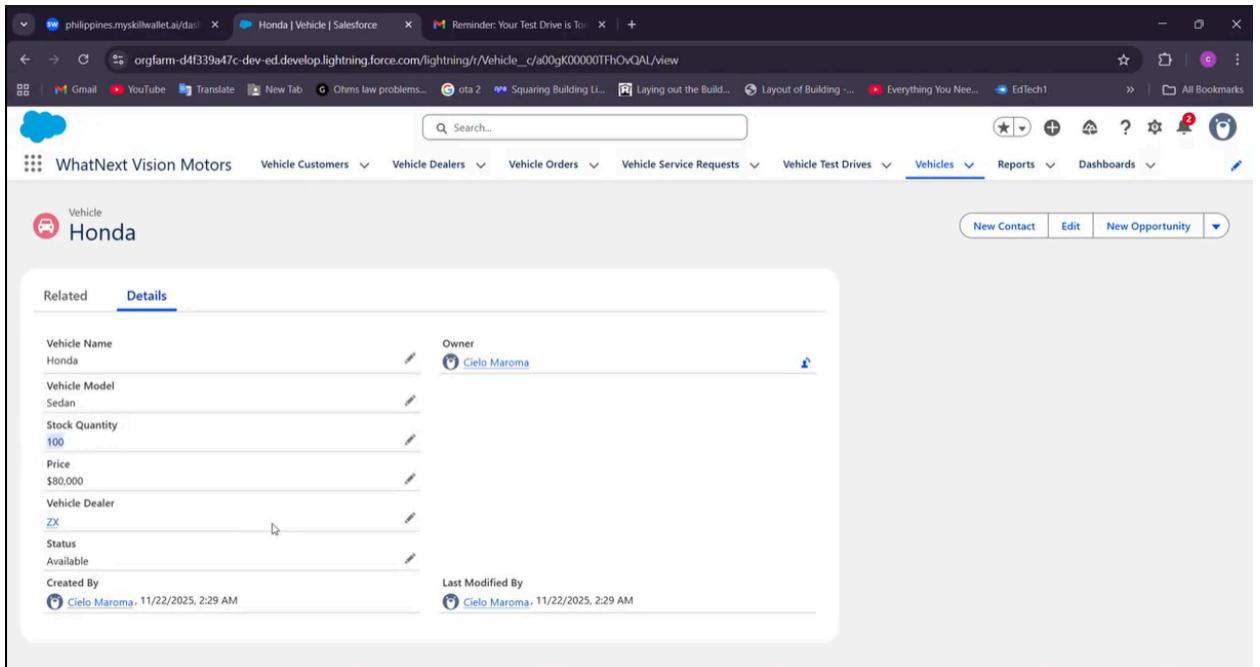


Figure 21. Creating Vehicle Order

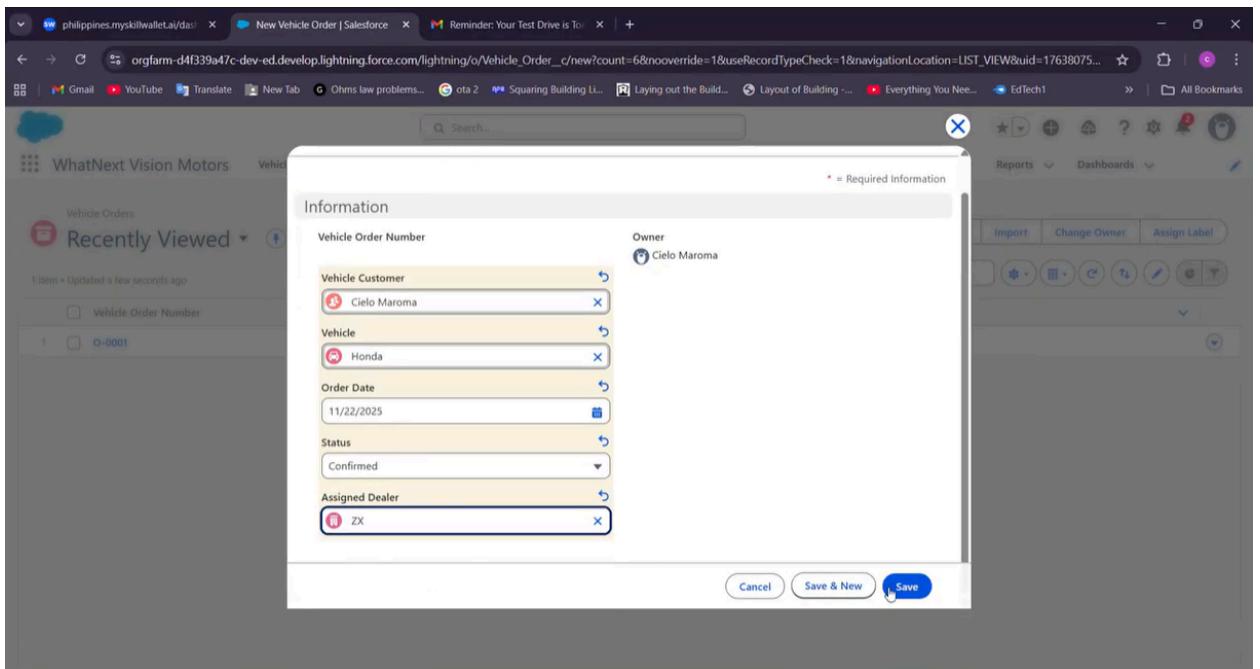
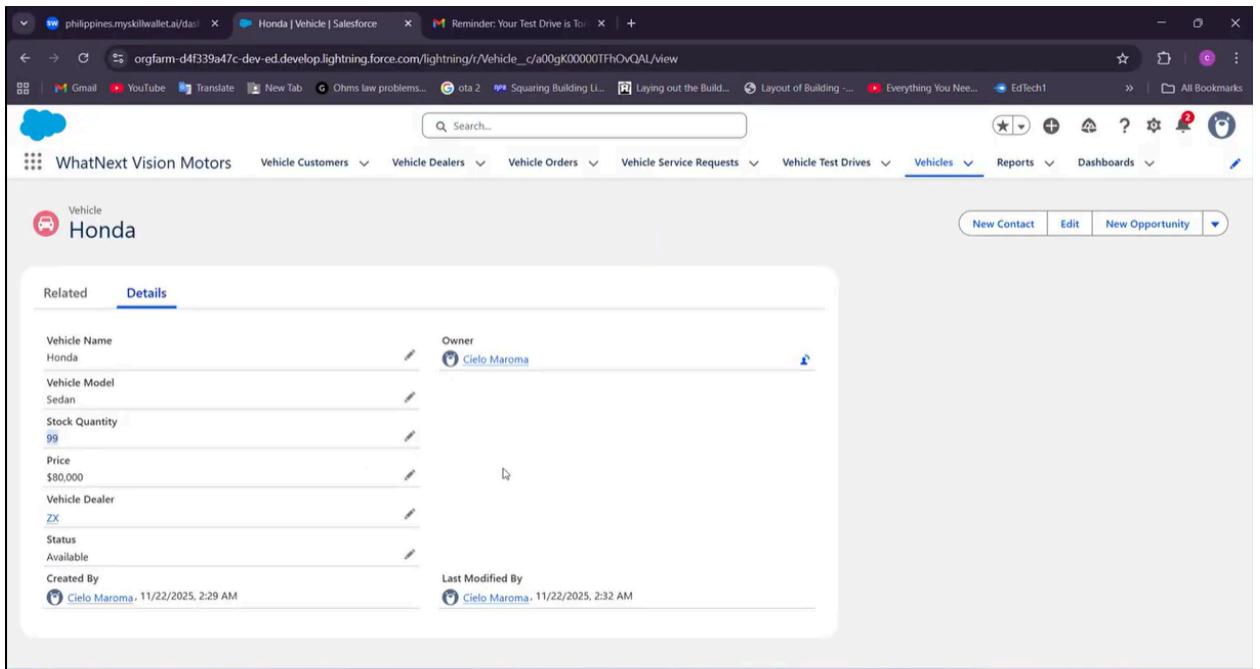


Figure 22. 99 Stock Quantity After Creating Vehicle Order



This test demonstrates the trigger's stock reduction logic. When an order of quantity 1 is placed, the inventory decreases from **100** to **99**. This confirms that inventory tracking is accurate and functioning.

Stock Validation Error

Figure 23. 0 Stock Quantity Stock to Test Validation Error Prompt

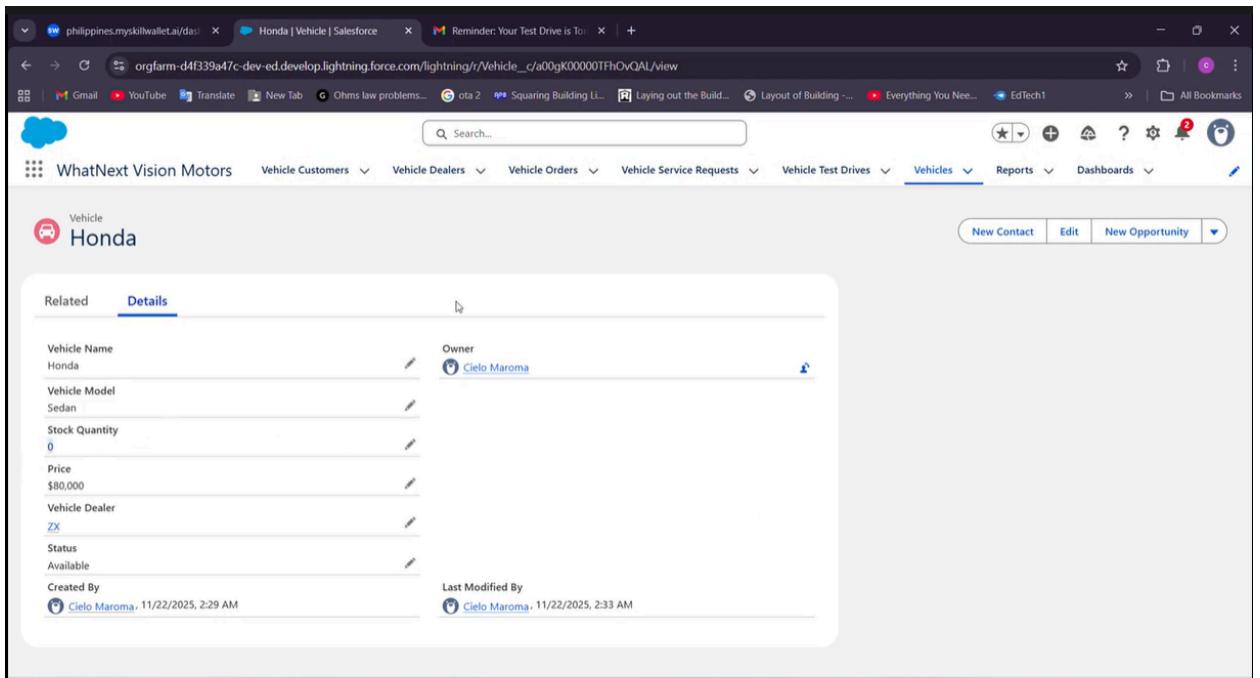
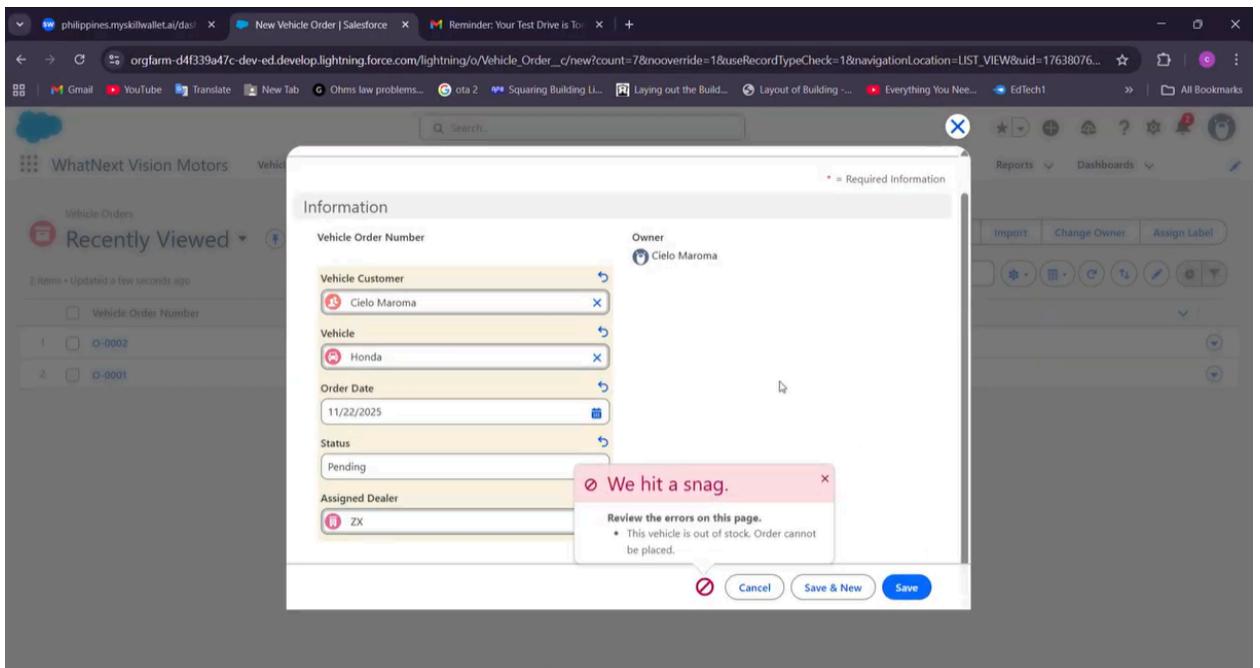


Figure 23. Stock Validation Error Prompt



This error message appears when:

- A user attempts to order more units than are available

- The trigger prevents negative stock scenarios

This enforces business rules and prevents data inconsistencies.

Email Reminder – Test Drive

Figure 24. Creating New Vehicle Test Drive

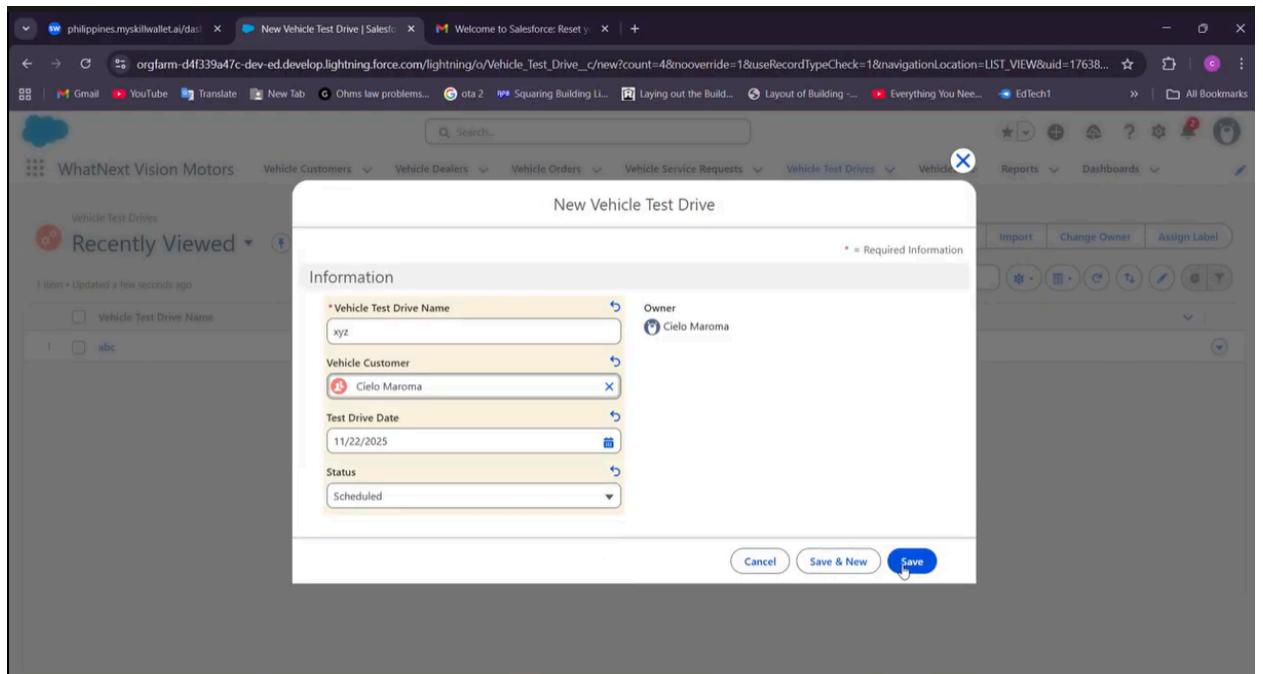
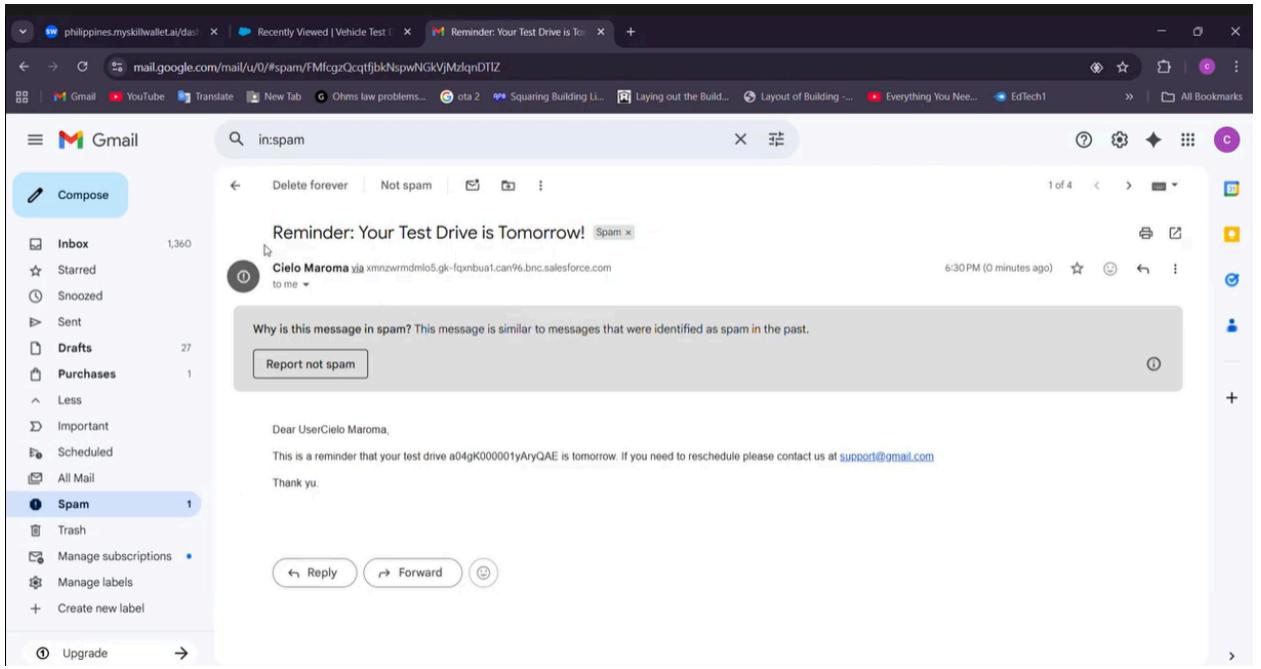


Figure 25. Email Reminder Received



The screenshot verifies that the Test Drive Reminder Flow successfully sends scheduled emails. This ensures customers are reminded promptly before their appointments.

Testing Approach Summary

Testing included:

Flow Testing

- Confirmed triggers fired on correct events
- Evaluated scheduled paths
- Verified all field updates and emails

Apex Logic Testing

- Trigger executed expected stock decrements
- No recursion or unintended updates occurred

Validation Rules Testing

- Confirmed improper data was blocked
- Prevented negative stock processes

UI Testing

- Page layouts displayed correct fields
- Tabs and apps navigated correctly

Functional Testing

- End-to-end order creation
- Dealer assignment
- Inventory update
- Test drive reminder automation

Phase 5: Deployment, Documentation & Maintenance

Phase 5 focuses on preparing the system for real-world use by finalizing deployment procedures, completing project documentation, and establishing maintenance practices. This phase ensures that all configurations, automation, and custom components are properly migrated to the production environment, supported by clear technical documentation for future reference. It also includes setting up monitoring, troubleshooting guidelines, and regular system checks to maintain performance, data integrity, and long-term usability of the WhatNext Vision Motors Salesforce application.

Deployment Strategy

- Change Sets recommended for sandbox-to-production deployment
- Sequential deployment: Objects → Fields → Flows → Apex Classes → App
- Post-deployment: Activate Flows, assign App permissions

Maintenance & Monitoring

Monitoring:

- Flow error emails and debug logs
- Apex exception handling in triggers
- Scheduled batch job monitoring

Troubleshooting Approach:

1. Check Flow debug logs for automation failures
2. Review Apex debug logs for trigger errors
3. Validate data quality in lookup relationships
4. Monitor email deliverability logs

Conclusion

The WhatNext Vision Motors CRM successfully automates core vehicle order processes, delivering geographically optimized dealer assignment, real-time inventory validation, and proactive customer notifications. Key achievements include 6 custom objects with comprehensive field structures, 2 record-triggered Flows, and Apex automation for stock management. The solution provides a scalable foundation for automotive CRM operations, demonstrating practical Salesforce automation capabilities.

Future Enhancements

- AI-Powered Dealer Matching using geolocation distance calculations
- Chatbot Integration for test drive scheduling
- Advanced Reporting with order fulfillment KPIs
- Mobile Lightning App for field sales team
- Einstein Predictions for demand forecasting