

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

Project Overview

The WhatsNext Vision Motors CRM is a Salesforce-based customer relationship management system developed to streamline the vehicle ordering process and enhance overall customer experience in the automotive industry. The project focuses on automating dealer assignment, validating stock availability, and improving operational workflows to reduce manual work and service delays. Key features of the CRM include real-time tracking of vehicle inventory, prevention of orders for out-of-stock units, automatic dealer suggestions based on customer location, and scheduled updates that modify order statuses to either Pending or Confirmed depending on stock levels. By integrating automated processes, Apex development, and batch jobs, the system delivers a more efficient and transparent ordering flow—ensuring fast service, minimized errors, and higher customer satisfaction.

Objectives

The main objective of this project is to design and implement a customized Salesforce CRM solution for WhatsNext Vision Motors that enhances the overall ordering workflow, centralizes vehicle and dealer information, and improves the customer experience.

By creating a well-structured data model that organizes vehicle inventory, dealer location, and customer orders in a consistent and accessible format, the CRM aims to:

1. Improve data accuracy and reliability by implementing proper validation rules, enforcing stock availability checks, and maintaining consistent records across the platform.
2. Automate critical business processes such as dealer assignment based on customer location, status updates through scheduled jobs, and restrictions on orders for out-of-stock vehicles—reducing manual work and minimizing operational errors.
3. Enhance customer experience and engagement by ensuring faster order processing, real-time visibility of availability, and transparent communication on order status, promoting trust and satisfaction.
4. Increase operational efficiency by streamlining order management, reducing administrative workload, and allowing staff to focus on decision-making and strategic tasks rather than repetitive processes.

Phase 1: Requirement Analysis & Planning

Phase 1 involves gathering business requirements and establishing the overall direction for the development of the WhatsNext Vision Motors CRM. This stage focuses on identifying key operational issues in the current ordering process, such as lack of

automated dealer assignment, difficulty in monitoring stock availability, and potential errors caused by manual order handling. From these insights, the project scope, data structure, and system workflow were carefully planned to ensure that the solution addresses real business needs.

Key Deliverables in Phase 1 include:

- Understanding Business Requirements

The team analyzed how the organization handles vehicle inquiries, ordering, and inventory monitoring. The goal was to solve challenges like out-of-stock ordering conflicts, inefficient dealer assignment, and delayed updates on order status.

- Defining Project Scope and Objectives

The project was defined to focus on improving customer ordering experience, automating dealer allocation, maintaining accurate vehicle stock records, and implementing scheduled processing for order status updates. System limitations, timelines, and expected outputs were also documented.

- Design of Data Model and Security Model

The data model was structured to store and relate customer profiles, vehicle availability, dealer information, and order records. Security roles and access permissions were planned to ensure controlled visibility and data privacy within the organization.

- Stakeholders Mapping

Stakeholders such as system administrators, sales agents, dealers, and customers were identified along with their roles, responsibilities, and access needs within the CRM.

- Execution Roadmap

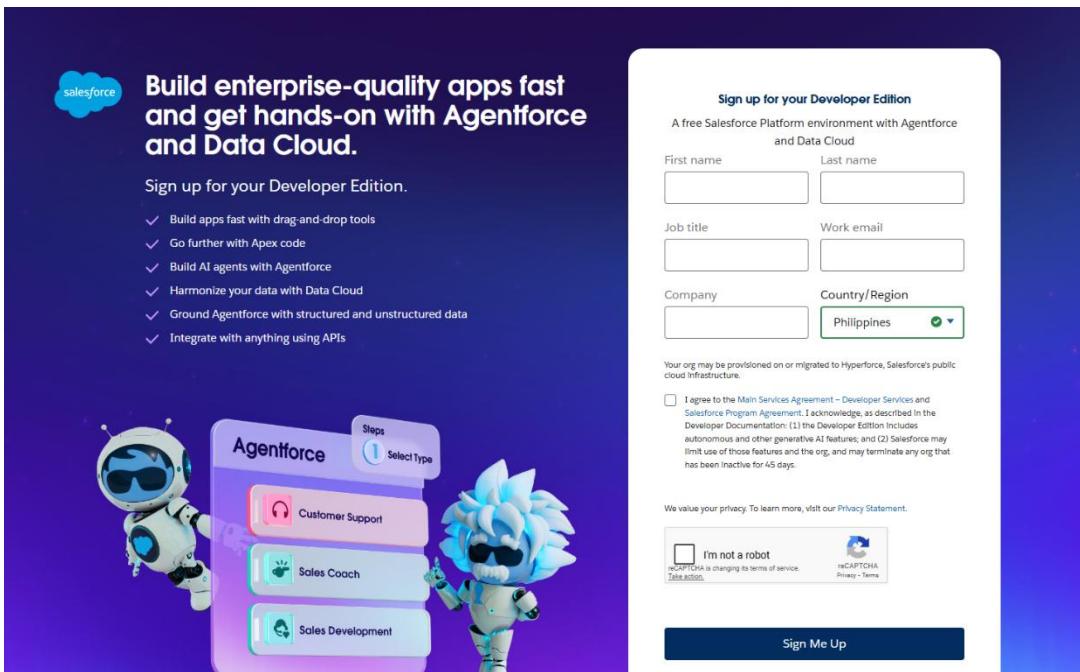
A development timeline was drafted, outlining milestones such as configuration setup, automation buildout, Apex trigger development, batch job implementation, testing phases, and deployment. This roadmap guided the step-by-step execution of the CRM.

Phase 2: Salesforce Development - Backend & Configurations

Environment Setup & DevOps Workflow

- Developer Org account was created using <https://developer.salesforce.com/signup>. The

account was created and verified. Password set and access was granted to the Salesforce Setup page.



Customization of Objects, Fields, Validation Rules, Automation

The following custom objects were created:

1. **Vehicle** - Stores vehicle details name, vehicle model, stock quantity, price, vehicle dealer and status.
2. **Vehicle Dealer** - Stores dealer info such as name, location, phone and email address.
3. **Vehicle Customer** - Stores customer details such as name, email address, phone, address, preferred vehicle type.
4. **Vehicle Order** - Stores vehicle purchases details such as customer, vehicle, order date, status and assigned dealer.
5. **Vehicle Test Drive** – Tracks and stores test drive bookings.
6. **Vehicle Service Request** – Tracks and stores vehicle servicing requests.

Table 1: Custom Object and their Key fields

Object Name	Key Fields
Vehicle	Name, Vehicle Model, Stock Quantity, Price, Vehicle Dealer Status.
Vehicle Dealer	Name, Location, Phone, Email Address
Vehicle Customer	Name, Email Address, Phone,

	Address, Preferred Vehicle Type
Vehicle Order	Customer Name, Vehicle, Order Date, Status, Assigned Dealer
Vehicle Test Drive	Test Drive Name, Customer Name, Vehicle, Test Drive Date, Status
Vehicle Service Request	Service Request Name, Customer. Service Date, Issue Description, Status

Validation Rules

- Out-of-Stock Order Validation – it blocks order creation if the chosen vehicle has zero stock.

Automation (Flows)

1. Auto Assign Dealer Flow - Automatically assign the nearest dealer to a customer's vehicle order based on the customer's address without manual input.

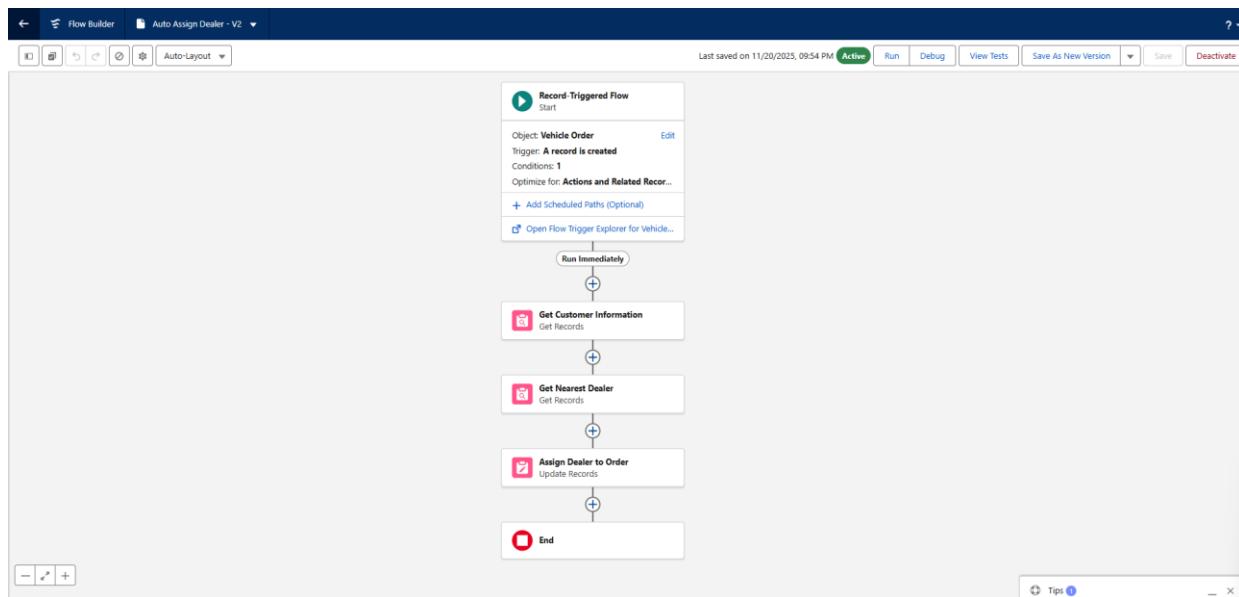


Fig.2. Auto Assign Dealer Flow

2. Test Drive Reminder Flow – it sends a reminder email to the customer about their scheduled test drive.

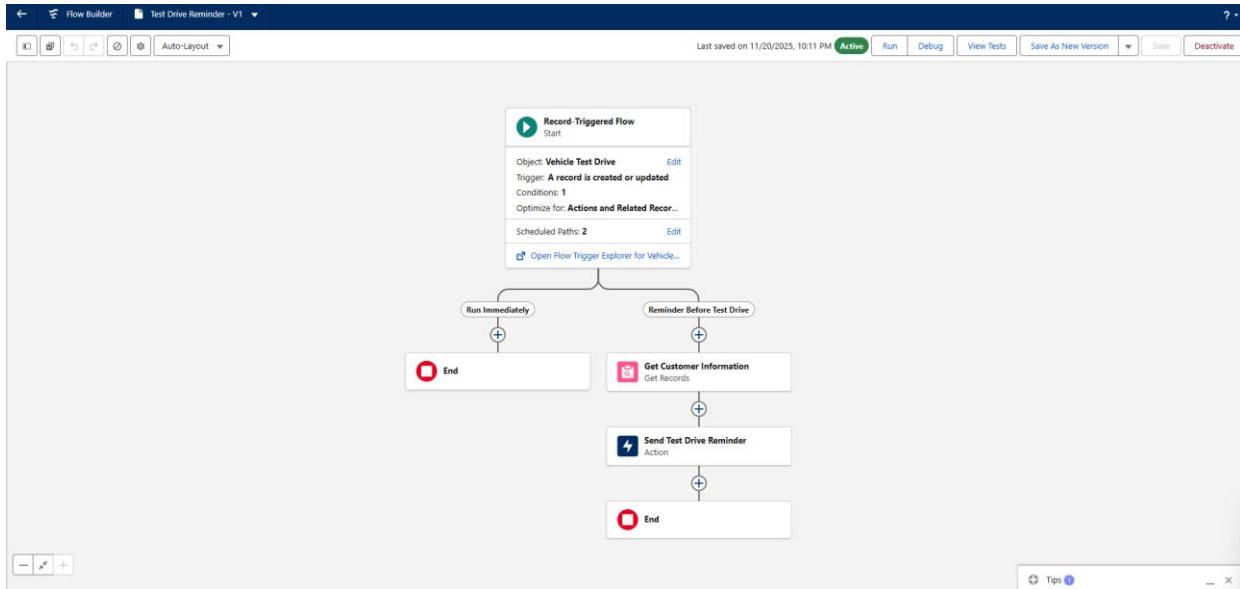


Fig.3. Test Drive Reminder Flow

Apex Classes and Triggers

Apex Classes:

- VehicleOrderBatch – this code checks if the stock is available for each pending order. If the stock is available, it will reduce the inventory count of the vehicle and updates both the order and vehicle records.
- VehicleOrderBatchScheduler - this code schedules the batch job to run daily at 12pm.
- VehicleOrderTriggerHandler - this code stops the order from continuing if the vehicle is out of stock and deducts inventory when the order is confirmed.

Action	Name	Namespace Prefix	Api Version	Status	Size Without Comments	Last Modified By	Has Trace Flags
Edit Del Security	VehicleOrderBatch		65.0	Active	2,010	Scharizze Barqueros	11/26/2025, 8:38 AM
Edit Del Security	VehicleOrderBatchScheduler		65.0	Active	230	Scharizze Barqueros	11/26/2025, 8:37 AM
Edit Del Security	VehicleOrderTriggerHandler		65.0	Active	2,775	Scharizze Barqueros	11/20/2025, 6:15 AM

Fig.4. Apex Classes

Apex Triggers:

- VehicleOrderTrigger - When a Vehicle Order record is inserted or updated, this trigger is triggered. It transfers all trigger context variables to the VehicleOrderTriggerHandler, which handles inventory updates and stock checking,

Action	Name	Namespace Prefix	sObject Type	Api Version	Status	Size Without Comments	Last Modified By	Has Trace Flags
Edit Del	VehicleOrderTrigger		Vehicle_Order	65.0	Active	255	Scharizze Barqueros	11/20/2025, 6:18 AM

rather than having logic of its own.

Fig.5.Apex Triggers

Phase 3: UI/UX Development & Customization

Lightning App

The WhatsNext Vision Motors App is created on the App Manager to provide a simplified and easy to use interface, enhancing the customer ordering experience and allowing users to efficiently manage vehicle orders, customer information, and dealer assignments.

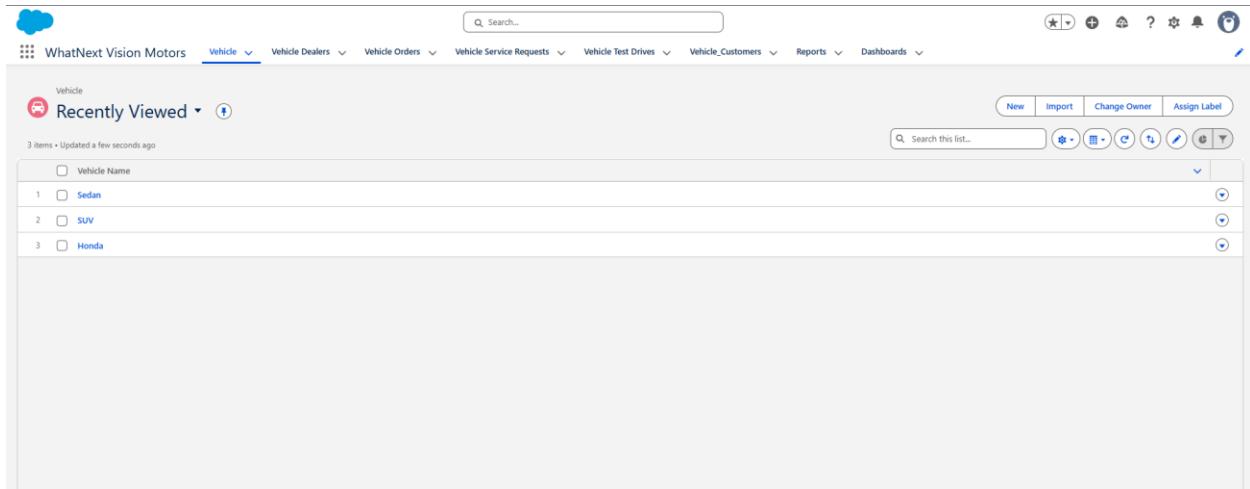


Fig.6. WhatsNext Vision Motors App

Page Layouts, Dynamic Forms

To improve usability and streamline data entry, the page layouts for Vehicle Orders, Customers, and Vehicles were customized. Dynamic Forms were implemented to display only relevant fields based on the context of each record, ensuring that users see the right information at the right time. The screenshots below illustrate the layout structure and how the dynamic fields appear during record creation and updates.

- **Vehicle**

The screenshot shows the 'Vehicle Detail' page layout. At the top, there are two sections: 'Standard Buttons' containing 'Edit', 'Delete', 'Clone', 'Change Owner', 'Change Record Type', 'Printable View', 'Sharing', and 'St' buttons; and 'Custom Buttons' which is currently empty. Below these are sections for 'Information' and 'Owner'. The 'Information' section contains fields for 'Vehicle Name' (marked with a red asterisk), 'Vehicle Model', 'Stock Quantity' (75,240), 'Price' (\$123.45), 'Vehicle Dealer' (Sample Text), and 'Status' (Sample Text). To the right of these fields is the 'Owner' section, which displays 'Owner' and 'Sample Text'.

Fig.7. Vehicle Detail Page Layout

New Vehicle

* = Required Information

Information

* Vehicle Name

Owner
 Scharizze Barqueros

Vehicle Model
 --None-- ▾

Stock Quantity

Price

Vehicle Dealer
 Search Vehicle Dealers... 

Status
 --None-- ▾

Fig.8. Form to add another vehicle

• Vehicle Dealers

Vehicle Dealer Detail

Standard Buttons

Information (Header visible on edit only)

 <input checked="" type="radio"/> Vehicle Dealer Name	Sample Text	Owner <input type="text" value="Sample Text"/>
Dealer Location_	Sample Text	
 Dealer Code	GEN-2004-001234	
Phone	1-415-555-1212	
Email	sarah.sample@company.com	

Fig.9. Vehicle Dealers Detail Page Layout

New Vehicle Dealer

* = Required Information

Information

* Vehicle Dealer Name

Owner

 Scharizze Barqueros

Dealer Location_

Dealer Code

Phone

Email

[Cancel](#)

[Save & New](#)

[Save](#)

Fig.10. Form to add new dealers

• Vehicle Orders

Vehicle Order Detail

Standard Buttons: [Edit](#) [Delete](#) [Clone](#) [Change Owner](#) [Change Record Type](#) [Printable View](#) [Sharing](#) [Sharing Hierarchy](#)

Information (Header visible on edit only)

Vehicle Order Number	GEN-2004-001234
Vehicle_Customer	Sample Text
Vehicle	Sample Text
Order Date	11/28/2025
Status	Sample Text
Assigned Dealer	Sample Text

Owner [Sample Text](#)

Fig.11. Vehicle Orders Detail Page Layout

New Vehicle Order

* = Required Information

Information

Vehicle Order Number

Owner

 Scharizze Barqueros

Vehicle_Customer



Vehicle



Order Date



Status

--None-- 

Assigned Dealer



[Cancel](#)

[Save & New](#)

[Save](#)

Fig.12. Form to add new vehicle order

- **Vehicle Service Requests**

Vehicle Service Request Detail

Standard Buttons: [Edit](#) [Delete](#) [Clone](#) [Change Owner](#) [Change Record Type](#) [Printable View](#) [Sharing](#) [Sharing Hierarchy](#)

Information (Header visible on edit only)	
★ <input checked="" type="radio"/> Vehicle Service Request Name	Sample Text
Vehicle_Customer	Sample Text
Vehicle	Sample Text
Service Date	11/28/2025
Issue Description	Sample Text
Status	Sample Text

Owner: [Sample Text](#)

Fig.13. Vehicle Service Requests Detail Page Layout

New Vehicle Service Request

* = Required Information

Information

* Vehicle Service Request Name	Owner  Scharizze Barqueros
Vehicle_Customer	Search Vehicle_Customers... 
Vehicle	Search Vehicle... 
Service Date	<input type="text"/> 
Issue Description	<input type="text"/>
Status	<input type="text"/> --None-- 

[Cancel](#) [Save & New](#) [Save](#)

Fig.14. Form to add new vehicle service request

- **Vehicle Test Drivers**

Vehicle Test Drive Detail

Standard Buttons: [Edit](#) [Delete](#) [Clone](#) [Change Owner](#) [Change Record Type](#) [Printable View](#) [Sharing](#) [Sharing Hierarchy](#)

Information (Header visible on edit only)	
★ <input checked="" type="radio"/> Vehicle Test_Drive Name	Sample Text
Vehicle_Customer	Sample Text
Vehicle	Sample Text
Test Drive Date	11/28/2025
Status	Sample Text

Owner: [Sample Text](#)

Fig.15. Vehicle Test Drivers Detail Page Layout

New Vehicle Test Drive

* = Required Information

Information

* Vehicle Test_Drive Name

Owner

 Scharizze Barqueros

Vehicle_Customer

🔍

Vehicle

🔍

Test Drive Date

📅

Status

--None--

Cancel
Save
Save & New

Fig.16. Form to add new vehicle test drive

• Vehicle Customers

Vehicle Customer Detail

Information (Header visible on edit only)

* <input checked="" type="radio"/> Vehicle_Customer	Sample Text
Name	Sample Text
Email	sarah.sample@company.com
Phone	1-415-555-1212
Address	Sample Text
Preferred Vehicle Type	Sample Text

Standard Buttons

Edit
Delete
Clone
Change Owner
Change Record Type
Printable View
Sharing
Sharing Hierarchy

Owner Sample Text

Fig.17. Vehicle Customers Detail Page Layout

New Vehicle_Customer

* = Required Information

Information

* Vehicle_Customer Name

Owner

 Scharizze Barqueros

Email

Phone

Address

Preferred Vehicle Type

--None--

Cancel
Save
Save & New

Fig.18. Form to add new customer

Reports & Dashboard

Reports and dashboards were created to provide real-time insights into vehicle availability, order status, and dealer performance. These visual tools help users monitor key metrics, track operational efficiency, and make informed decisions. The screenshots below show the generated reports and dashboard components used to support data-driven management within the system.

• Reports

The screenshot shows a report titled "Report: Vehicle Orders with Vehicle". The report displays 13 records of vehicle orders, each with a unique order number and the name of the vehicle. The columns are "Vehicle Order: Vehicle Order Number" and "Vehicle: Vehicle Name".

	Vehicle Order: Vehicle Order Number	Vehicle: Vehicle Name
1	O-0001	Honda
2	O-0002	Honda
3	O-0008	Sedan
4	O-0007	Sedan
5	O-0009	Sedan
6	O-0006	Sedan
7	O-0005	Sedan
8	O-0004	Sedan
9	O-0013	SUV
10	O-0011	SUV
11	O-0010	SUV
12	O-0012	SUV
13	O-0003	SUV

Fig.19. Vehicle Orders with Vehicle Report

• Dashboards

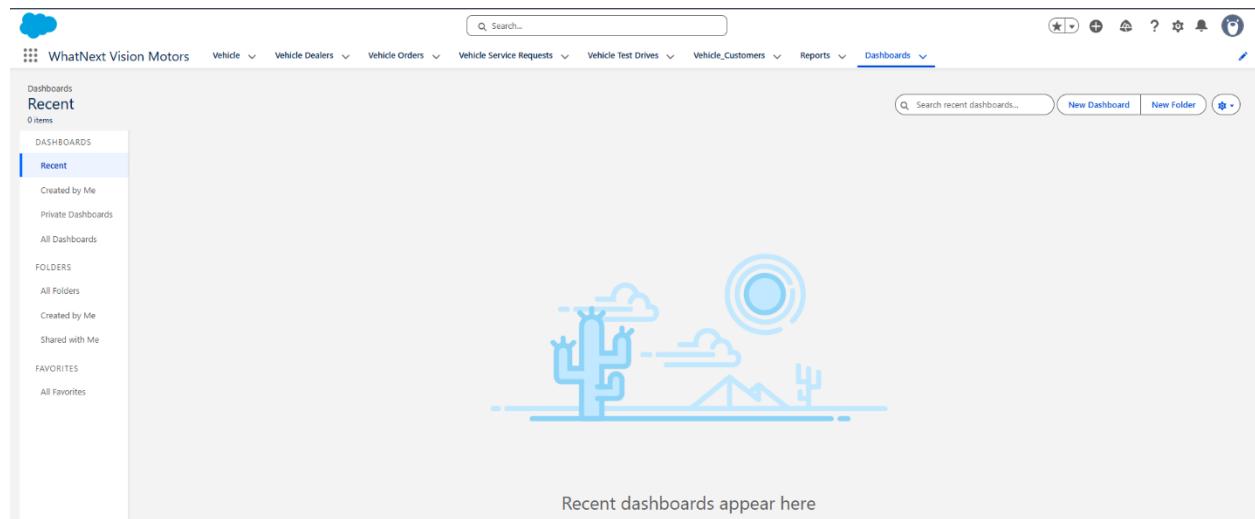
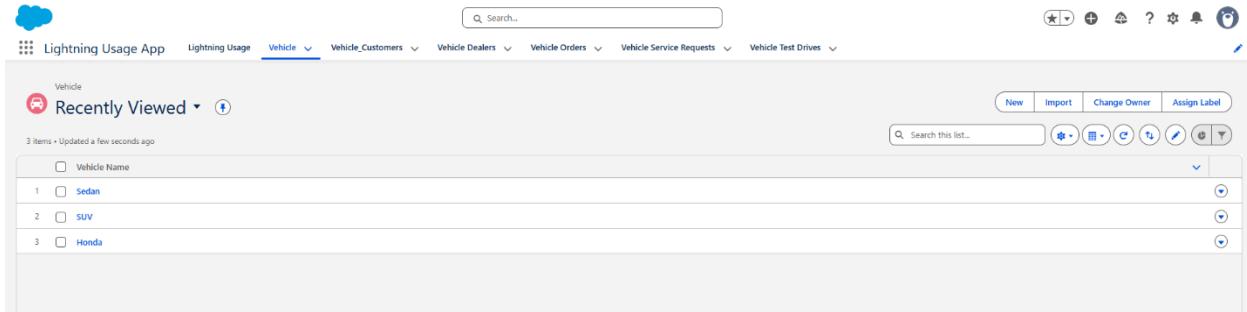


Fig.20. Dashboards

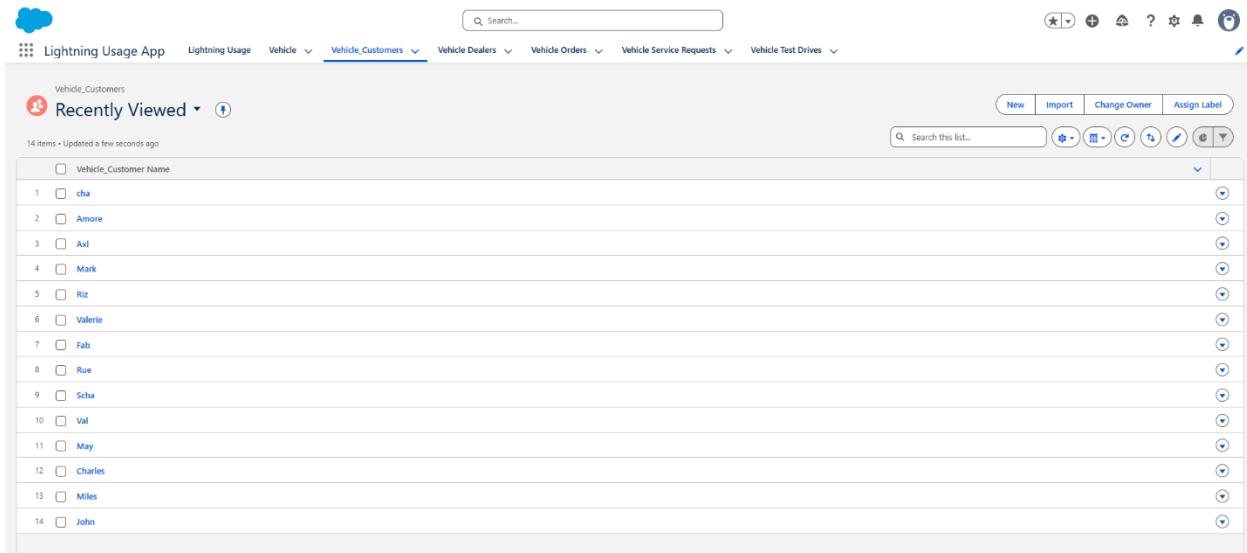
Lightning Pages

- Vehicle



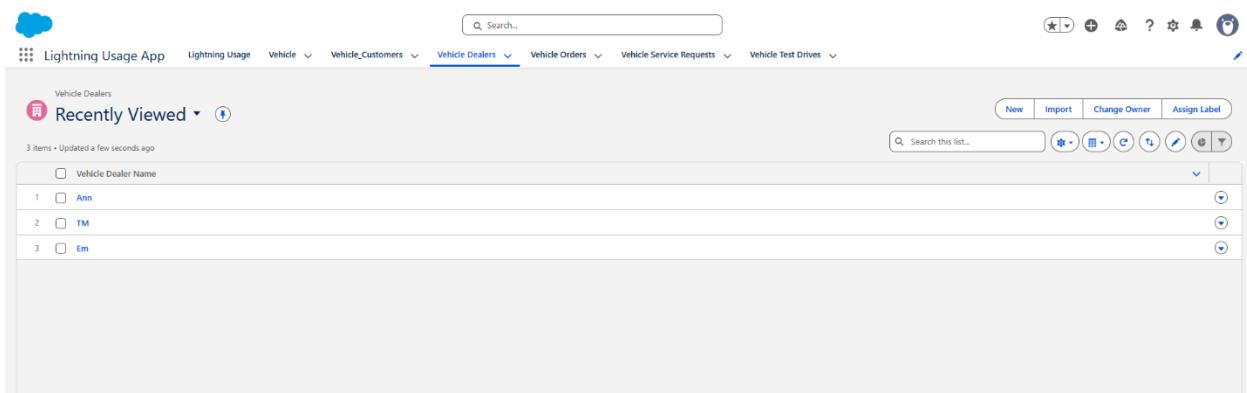
This screenshot shows the Lightning Page for the Vehicle object. The page title is "Vehicle". The navigation bar includes "Lightning Usage App", "Lightning Usage", "Vehicle", "Vehicle_Customers", "Vehicle Dealers", "Vehicle Orders", "Vehicle Service Requests", and "Vehicle Test Drives". The main content area displays a "Recently Viewed" list with 3 items: Sedan, SUV, and Honda. A search bar at the top right says "Search...". Below the list are standard Salesforce navigation icons.

- Vehicle Customer



This screenshot shows the Lightning Page for the Vehicle_Customer object. The page title is "Vehicle_Customers". The navigation bar includes "Lightning Usage App", "Lightning Usage", "Vehicle", "Vehicle_Customers", "Vehicle Dealers", "Vehicle Orders", "Vehicle Service Requests", and "Vehicle Test Drives". The main content area displays a "Recently Viewed" list with 14 items, each with a checkbox and a name: cha, Amore, Axl, Mark, Riz, Valerie, Fab, Rue, Scha, Val, May, Charles, Miles, and John. A search bar at the top right says "Search...". Below the list are standard Salesforce navigation icons.

- Vehicle Dealers



This screenshot shows the Lightning Page for the Vehicle_Dealer object. The page title is "Vehicle_Dealers". The navigation bar includes "Lightning Usage App", "Lightning Usage", "Vehicle", "Vehicle_Customers", "Vehicle_Dealers", "Vehicle Orders", "Vehicle Service Requests", and "Vehicle Test Drives". The main content area displays a "Recently Viewed" list with 3 items: Ann, TM, and Em. A search bar at the top right says "Search...". Below the list are standard Salesforce navigation icons.

- Vehicle Orders

• Vehicle Service Request

• Vehicle Test Drive

Phase 4: Data Migration, Testing & Security

Field History Tracking

Field History Tracking was configured to monitor important changes across key objects of the WhatNext Vision Motors system. This ensures transparency, auditability, and accountability for all business-critical updates.

Purpose:

To maintain a historical record of changes made to essential fields such as vehicle availability, order status, and customer details.

Objects with Tracking Enabled:

Vehicle__c – Stock Quantity, Vehicle Price

Customer__c – Email, Mobile Number

Order__c – Order Status, Assigned Vehicle, Dealer

Duplicate Rules & Matching Rules

To ensure clean, consistent, and reliable customer and dealer records, Salesforce duplicate detection was implemented through Matching and Duplicate Rules.

- **Matching Rules**

Matching Rules identify similarities between new and existing records.
A custom Matching Rule was created for:

Customer__c

Based on Email and Mobile Number

Dealer__c

Based on Dealer Name and Contact Email

- **Duplicate Rule**

Prevents duplicate customers from being created in the system

Profiles and Roles

Standard profiles such as *Standard User* and *System Administrator* were utilized to manage access levels within the system.

Role Hierarchy established:

- CEO
 - └ Sales Manager
 - └ Sales Rep

Permission Sets:

Created an Order Management Access permission set and assigned it to users who require access to create and manage Orders and Vehicles.

Sharing Rules

Public Read/Write for most custom objects

Manual Sharing allowed for sensitive customer records

Apex Test Classes

Apex Test Classes were created to validate system logic, ensure reliability, and meet Salesforce's minimum 75% code coverage requirement.

Objectives of Test Classes

- Validate triggers (VehicleOrderTrigger)
- Validate batch jobs (VehicleOrderBatch)
- Ensure no DML or logic errors occur
- Confirm automation works under real-world scenarios

Feature Testing & Test Cases

Comprehensive testing was done for every Salesforce automation implemented within the WhatNext Vision Motors ecosystem.

Features Tested

- Vehicle Booking Creation
- Approval Process for High-Value Orders
- Automatic Task Creation (e.g., customer follow-up tasks)
- Record-Triggered Flows
- Apex Triggers (Order validation, stock deduction)
- Batch Apex (processing pending orders)
-

Phase 5: Deployment, Documentation & Maintenance

The final phase of the WhatsNext Vision Motors CRM Implementation focused on deploying the developed Salesforce components into the production instance, finalizing documentation, and establishing long-term maintenance procedures to ensure system reliability and scalability.

• Deployment Strategy

Deployment was carried out using Salesforce Change Sets as the primary migration method. All required configurations—including custom objects, fields, flows, validation rules, Apex classes, triggers, profiles, and permission sets—were packaged in organized batches and deployed from Sandbox to Production.

Before deployment, User Acceptance Testing (UAT) was performed to ensure that automations executed correctly and business requirements were met. This strategy helped reduce deployment risks and ensured smooth transition with minimal production interruptions.

• Maintenance & Monitoring Approach

After the system went live, a structured maintenance plan was implemented to ensure continuous performance and operational stability. Routine monitoring activities include:

- Reviewing flow and automation logs for errors
- Maintaining objects, layouts, and fields based on evolving requirements
- Managing user access through profiles & permission sets
- Providing user support through reported tickets and guided troubleshooting

Enhancements and fixes will first be tested in Sandbox before deployment to Production to prevent data loss or system downtime.

- **Troubleshooting Documentation**

A troubleshooting reference was created to enable rapid response to system issues and reduce operational delays. It includes:

- A list of common user issues and recommended solutions
- Guidelines for modifying layouts, permissions, and validation rules
- Steps to debug Flow, Batch Jobs, and Apex Trigger errors using Debug Logs
- Verification procedures for new Change Sets prior to deployment

This ensures the system remains stable and allows administrators to resolve issues efficiently.

- **Documentation Overview**

All key system components were documented for easy reference and future development. The documentation includes:

- Custom objects and fields with their functional purposes
- Record-triggered flows and automation processes
- Validation rules implemented to maintain data integrity
- Profile and permission-set configuration for controlled system access

- **Testing Approach**

Multiple testing methods were performed prior to deployment:

Testing Type	Purpose
Apex Test Classes	Validate trigger logic & automation behavior
Flow Testing	Confirm correct dealer assignment & order validation
UAT (User Acceptance Testing)	Real-scenario order creation and approval testing
Reports & Dashboards Validation	Ensure accuracy of inventory and order visibility

Thorough testing ensured smooth functionality of automation, accurate order processing, and reliability of the stock monitoring system.

This ensured that all automation behaved as expected before production deployment.

Future Enhancements

To expand CRM capability and customer experience, the following improvements are proposed:

- AI-Based Vehicle Recommendation Engine for smarter customer matching
- Chatbot Support Integration for order tracking and inquiries
- Advanced BI Dashboards for real-time metrics and business analysis
- SMS/Email Alerts for order updates, stock changes, and schedule reminders

These enhancements would increase automation, user engagement, and operational intelligence.

Conclusion

The WhatsNext Vision Motors Salesforce CRM successfully delivered an integrated solution for managing customers, vehicle inventory, orders, and dealer assignments. Automation reduced manual workload, ensured stock accuracy, and enhanced overall customer experience.

This system lays a strong foundation for future expansion—scalable, efficient, and aligned with the company's vision of digital transformation. With planned enhancements and continuous monitoring, WhatsNext Vision Motors is now equipped for long-term growth and customer-centric innovation.