Bulacan State University

Salesforce Administrator/Developer Certification Capstone Project

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

Submitted in partial fulfillment of the requirements for the

Salesforce Administrator/Developer Certification

Submitted by:

Seven T. Drew

Section:

BSIT 4AG2

Date of Submission:

November 2025

Project Overview

The WhatNext Vision Motors CRM is a Salesforce system designed to manage the end-to-end vehicle sales process. It centralizes all information about vehicles, customers, dealers, orders, test drives, and service requests in a single platform.

The CRM streamlines dealership operations by automatically assigning the nearest dealer to customer orders, preventing orders for out-of-stock vehicles, and sending reminders for scheduled test drives. The system improves customer experience, ensures accurate stock management, and reduces manual work for the sales team.

Objectives

The goal of the WhatNext Vision Motors CRM is to build a system that simplifies and automates the entire vehicle sales process. The CRM is designed to keep customer and vehicle data organized, ensure accurate stock tracking, and make dealer assignment faster through automation. By reducing manual steps and improving data accuracy, the system helps the business deliver better service, avoid order delays, and maintain smoother operations overall.

Specific Objectives (Linked to Business Value):

- Improve customer management by storing complete and organized customer records.
- Ensure accurate stock control through validations and automated updates.
- Streamline the ordering process by preventing orders when vehicles are out of stock.
- Automate dealer assignment to reduce manual work and speed up processing.
- Enhance customer experience through automated test-drive reminders.
- Increase operational efficiency using batch jobs and automated order updates.

Phase 1: Requirement Analysis & Planning

This initial phase focused on gathering business needs, defining the project's boundaries, and creating a blueprint for the system's structure and security before any development began.

1.1 Understanding Business Requirements

WhatNext Vision Motors required a centralized Customer Relationship Management (CRM) system to streamline its vehicle sales operations. The primary goal was to reduce the manual workload for sales teams and dealers by automating key processes. Specific user needs included maintaining precise vehicle inventory data, preventing orders for out-of-stock models, automatically assigning dealers based on customer location, and sending timely test-drive reminders. Users also needed a unified view of customer details, dealer information, and service requests within a single platform. The existing manual processes were prone to delays and data inconsistencies. The new CRM was designed to address these challenges by introducing a structured database, automated business rules, and efficient workflows.

1.2 Defining Project Scope and Objectives

The project scope encompassed the creation of six custom objects: Vehicles, Customers, Dealers, Orders, Service Requests, and Test Drives, along with the implementation of automation using Salesforce Flows, Email Alerts, Apex Triggers, and Batch Apex. The project also included developing a user-friendly interface via Lightning App Builder and custom page layouts. Key functional areas covered order validation, dealer assignment, stock management, and notification systems.

The project objectives focused on enhancing operational efficiency, minimizing human error, ensuring accurate real-time stock tracking, and improving customer engagement. The system was built to manage the complete sales cycle from order placement and dealer assignment to test-drive scheduling, while supporting the business's need for well-organized data and automated process management.

1.3 Designing the Data Model and Security Model

The data model was structured around six custom objects interconnected with lookup relationships to mirror the actual dealership workflow. Vehicles are linked to customers through orders and test drives, while dealers are assigned according to the customer's location. Each object contains carefully designed fields to ensure complete and accurate data capture.

The security model is based on user profiles for Administrators, Sales Agents, and Dealers. These profiles control data access, with field-level security protecting sensitive information. A role hierarchy grants managers visibility into their team's records, and sharing rules are configured to ensure dealers can only access orders assigned to them.

1.4 Stakeholder Mapping

The system was designed to support four key user roles:

- Admin: Manages system configuration, maintenance, and monitors automated processes.
- Sales Agent: Creates orders and test drive records, and serves as the primary point of contact for customers.
- Dealer: Receives and fulfills orders assigned to them.
- Sales Manager: Oversees team performance and ensures smooth operational flow across the sales process.

The CRM interface and permissions were tailored to meet the specific needs of each stakeholder group.

1.5 Execution Roadmap

A phased approach was adopted for project execution:

- Design Phase: Finalize requirements, design the data model, and plan the security architecture.
- Development Phase: Build custom objects, fields, automation flows, triggers, and batch jobs.
- UI/UX Setup: Configure the Lightning app, page layouts, and navigation.
- Testing Phase: Validate triggers, test automation flows, confirm email alerts, and execute batch jobs.
- Security Configuration: Assign profiles, define roles, and configure sharing settings.
- Deployment & Documentation: Package components using change sets and prepare final documentation and demonstrations.

Figure 1

Data Management - Objects

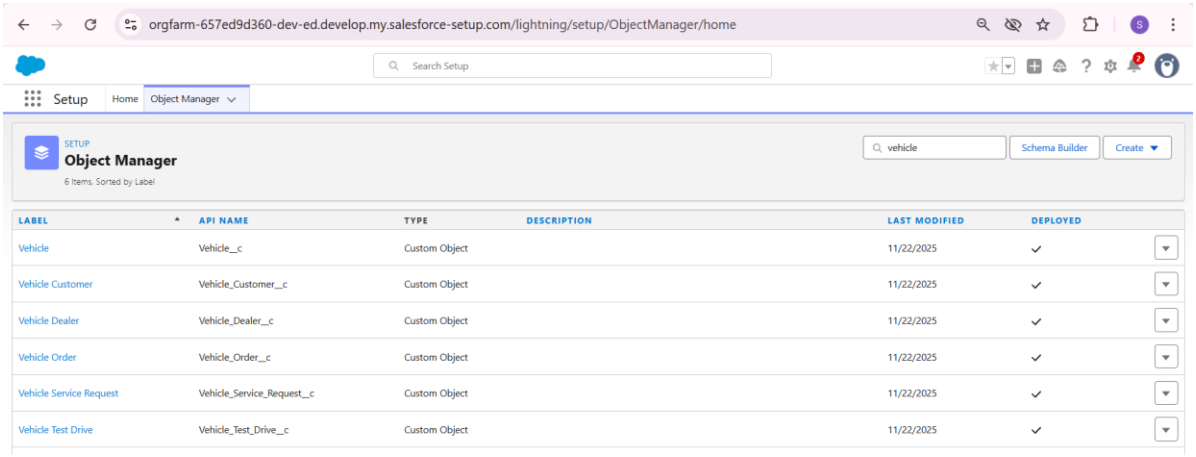

Figure 1 displays the Object Manager in Salesforce Setup. It lists the six main custom objects created for the WhatNext Vision Motors system: Vehicle, Vehicle Customer, Vehicle Dealer, Vehicle Order, Vehicle Service Request, and Vehicle Test Drive. These objects are the core containers for all data in the CRM. Each object holds information for a different part of the business process. This structure organizes all data in one place and connects the information for vehicles, customers, and orders.

Figure 2

Data Management - Tabs

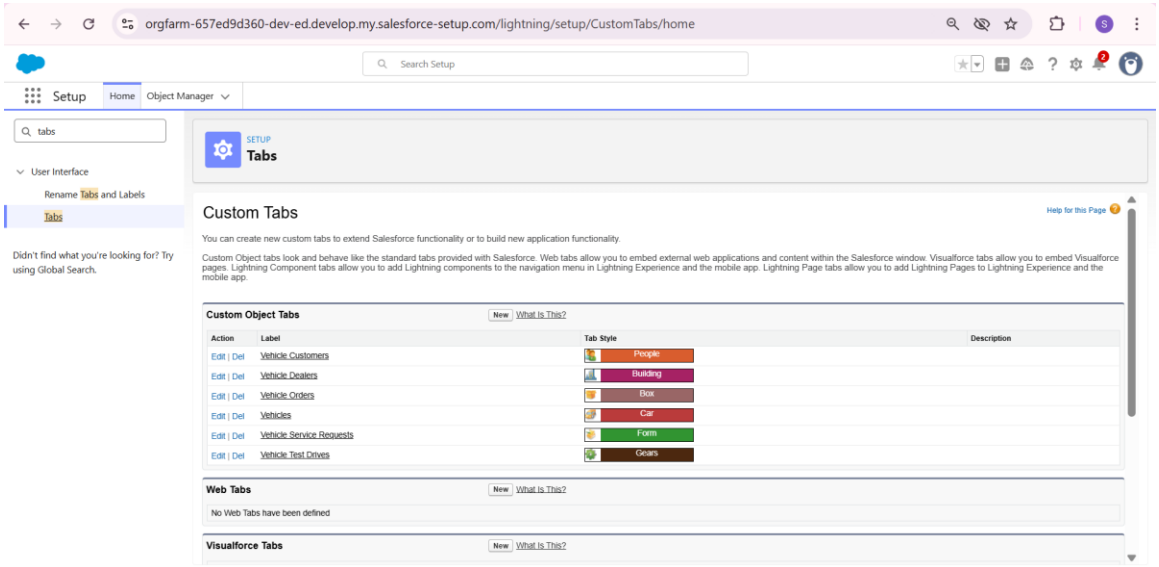

Figure 2 shows the setup page for custom tabs in Salesforce. This is where the navigation for the WhatNext Vision Motors app was built. A custom tab was created for each of the six main objects. These tabs allow users to easily view and manage all records for Vehicles, Customers, Orders, Dealers, Service Requests, and Test Drives. Placing these tabs in the app gives users a simple way to access all parts of the system.

Phase 2: Salesforce Development - Backend & Configurations

This phase involved building the core system logic, creating automation, and developing custom code to power the CRM's key functions.

2.1 Setup Environment

The complete CRM system was built within a single Salesforce Developer Edition organization. This environment served as the dedicated workspace for all development, testing, and configuration tasks. The entire system, including its data model, user interface, and automated processes, was constructed and validated here.

2.2 Customization of Objects, Fields, Validation Rules, and Automation

The application's foundation consists of six custom objects: Vehicles, Vehicle Orders, Vehicle Customers, Vehicle Dealers, Vehicle Test Drives, and Vehicle Service Requests. Each object was configured with specific fields to capture essential business data, such as customer contact information, vehicle stock levels, order status, and appointment schedules.

To streamline operations, two key automated flows were implemented:

- Auto-Assign Dealer Flow
    - This process automatically links a new Vehicle Order to the nearest dealer based on the customer's city, speeding up order assignment.
- Test Drive Reminder Flow
    - This automation sends an email reminder to customers 24 hours before their scheduled test drive, improving communication and reducing missed appointments.

Figure 3

Auto Assign Dealer Flow

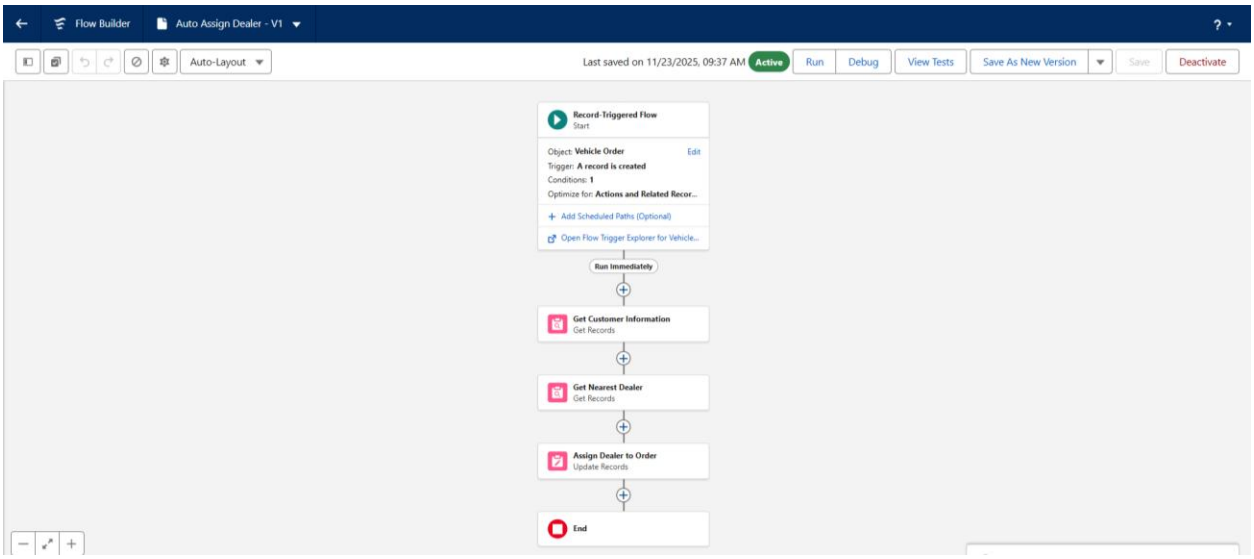

Figure 3 shows the automation flow that assigns a dealer to a new vehicle order. The flow starts when a customer creates a Vehicle Order record. It then gets the customer's location information and finds the nearest dealer based on their city. Finally, the flow automatically links the correct dealer to the order. This process removes the need for manual dealer assignment, saves time, and makes sure orders go to the right dealer quickly.

Figure 4

Test Drive Reminder Flow

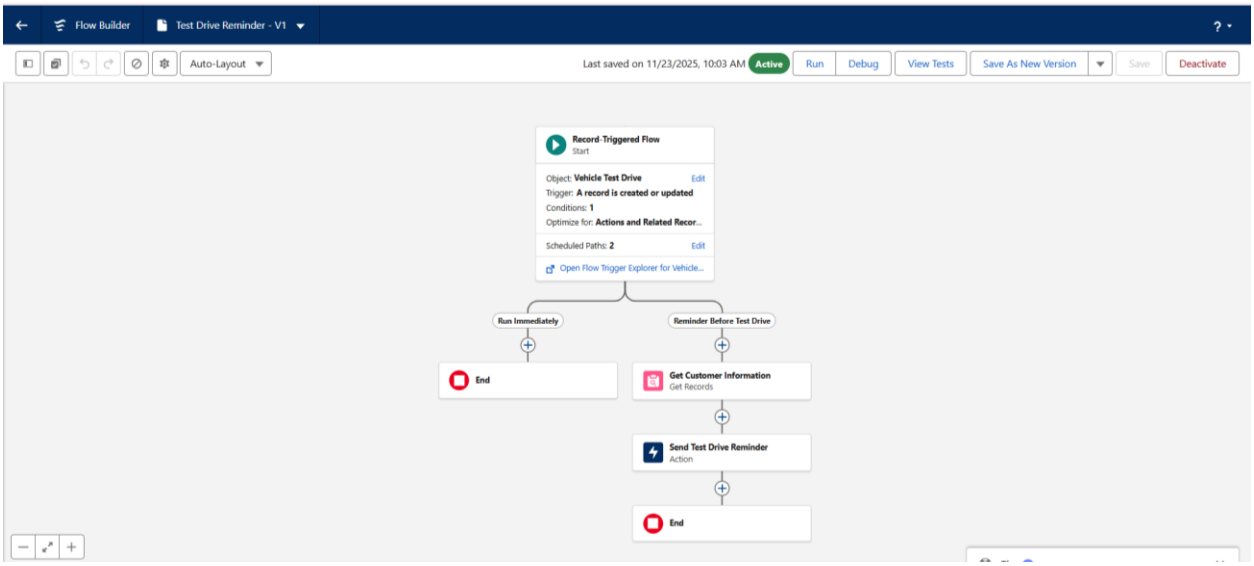

Figure 4 shows the automation that manages test drive reminders. This flow starts when a Vehicle Test Drive record is created or updated. It follows a set path to check if a reminder should be sent. The flow then collects customer information and automatically sends an email reminder about their scheduled test drive. This process ensures customers receive timely notifications, which helps reduce missed appointments and improves their overall experience.

Figure 5

Email Test Drive Reminder

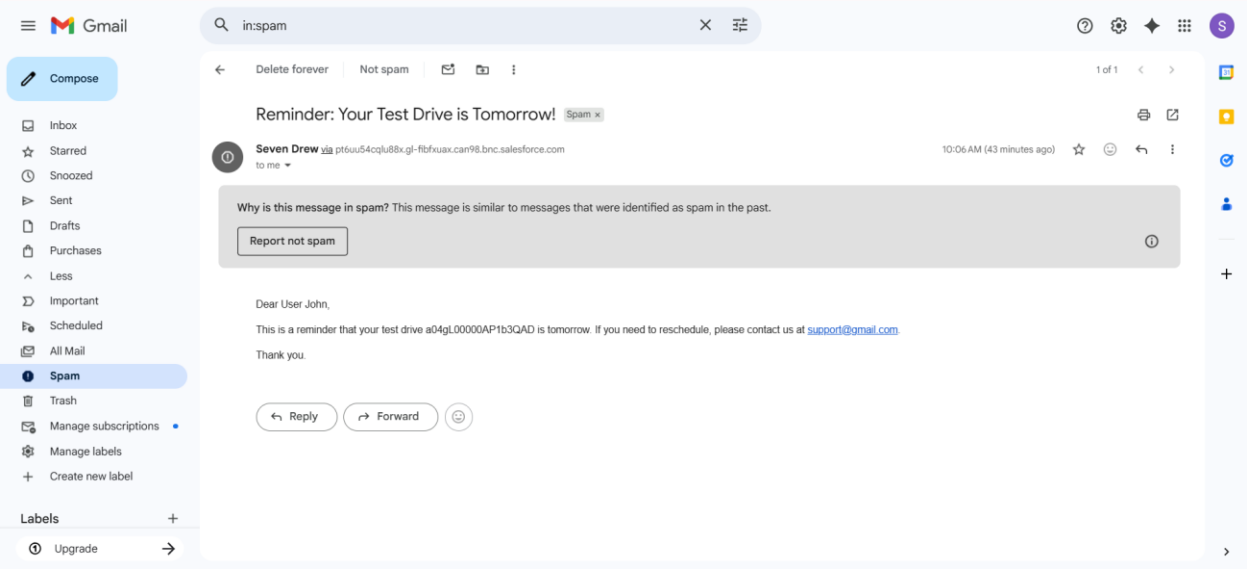

Figure 5 shows the automated email that customers receive before their scheduled test drive. The email serves as a reminder for the upcoming appointment and provides the test drive record number for reference. It also includes contact information in case the customer needs to reschedule. This automated communication helps ensure customers remember their appointments and know how to contact the dealership if needed, improving customer service and reducing missed test drives.

2.3 Apex Classes, Triggers, and Asynchronous Apex

Custom code was developed to handle complex business rules and bulk data processing.

a. VehicleOrderTrigger and Handler

A trigger was placed on the Vehicle Order object to enforce critical business rules. Its logic was managed by a separate handler class for better organization.

- Stock Validation: It prevents orders from being placed for vehicles that are out of stock.
- Real-Time Inventory Update: It automatically reduces a vehicle's stock count when an order is confirmed.

Figure 6

Vehicle Order Trigger

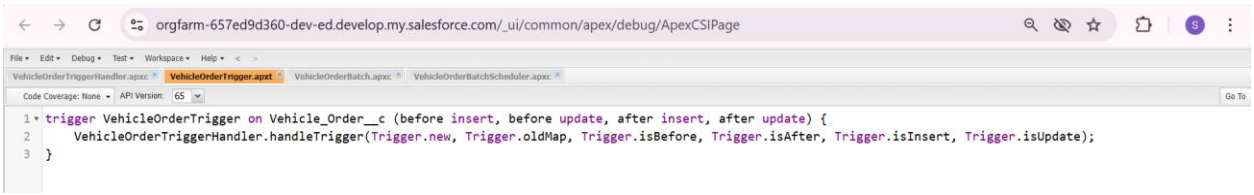

Figure 6 shows the Apex trigger code for the Vehicle Order object. This trigger acts as an automatic controller that runs whenever a Vehicle Order record is created or modified. It calls a separate handler class to manage all the business rules. The trigger is set to run both before and after records are saved to the system, allowing it to check conditions before saving data and to perform updates after data is saved. This structure keeps the code organized and makes it easier to maintain.

Figure 7

Vehicle Order Trigger Handler

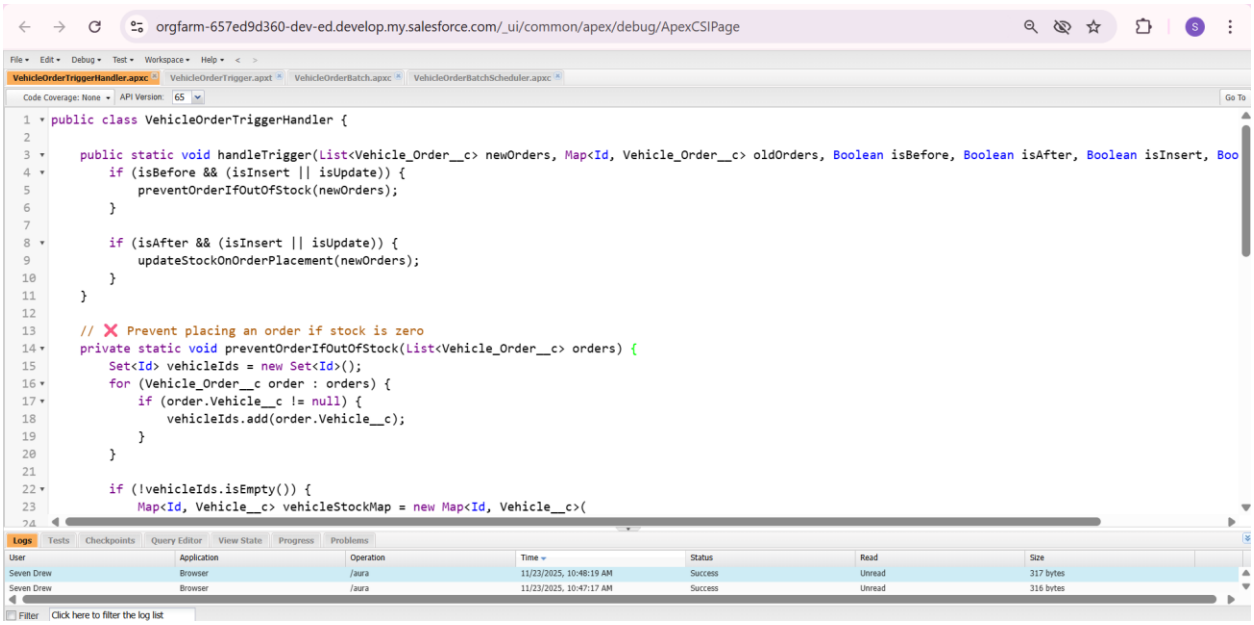

Figure 7 shows the main logic class that handles all business rules for vehicle orders. This class contains two important methods. The first method checks if a vehicle has available stock before an order can be placed. If the stock is zero, it blocks the order with an error message. The second method automatically reduces the vehicle's stock quantity when an order status changes to "Confirmed." This handler class keeps all the business logic organized in one place, separate from the trigger itself.

b. VehicleOrderBatch (Batch Apex)

For processing high volumes of data, a Batch Apex job was created. This job automatically finds all pending orders, confirms them if stock is available, and updates inventory levels in bulk, ensuring efficiency and data accuracy.

Figure 8

Vehicle Order Batch

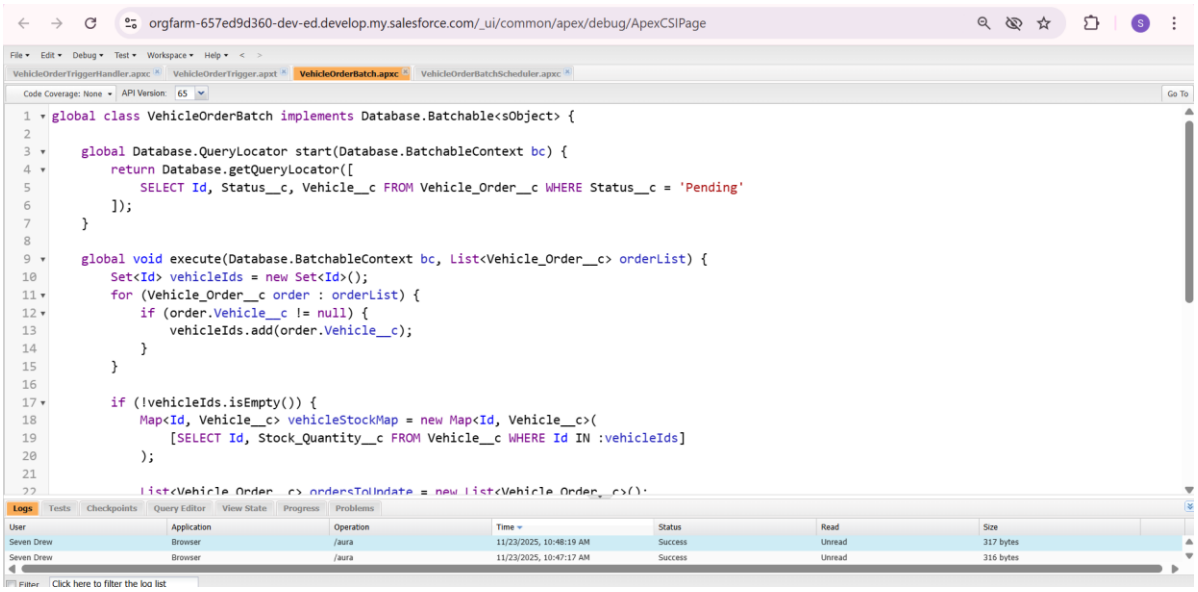

Figure 8 shows the batch processing class for handling multiple vehicle orders at once. This class is designed to process large numbers of pending orders automatically. It finds all orders with "Pending" status, checks if vehicles are in stock, and then confirms the orders while updating inventory levels. Batch processing allows the system to handle many records efficiently without manual work, making it ideal for processing high volumes of orders during busy sales periods.

c. VehicleOrderBatchScheduler

A scheduler was implemented to run the batch job automatically at regular intervals. This guarantees that pending orders are processed consistently without manual intervention, even during periods of high sales volume.

Figure 9

Vehicle Order Batch Scheduler

Figure 9 shows the scheduler class that automatically runs the batch job at regular intervals. This class tells Salesforce when to start the Vehicle Order Batch process. The number 50 means it will process 50 orders at a time for efficiency. By using this scheduler, the system can automatically check and confirm pending orders every hour, day, or week without any manual effort, ensuring orders are processed consistently.

Phase 3: UI/UX Development & Customization

This phase focused on creating a clear and efficient user interface, making the system easy to use for sales teams and dealers.

3.1 Lightning App Setup through App Manager

A dedicated application named "WhatNext Vision Motors" was built using the Lightning App Manager. This custom app brings together all the necessary components for vehicle sales in one place. It includes tabs for the six main objects while removing unused standard tabs, providing a clean and focused workspace for users.

Figure 10

Create Lightning App

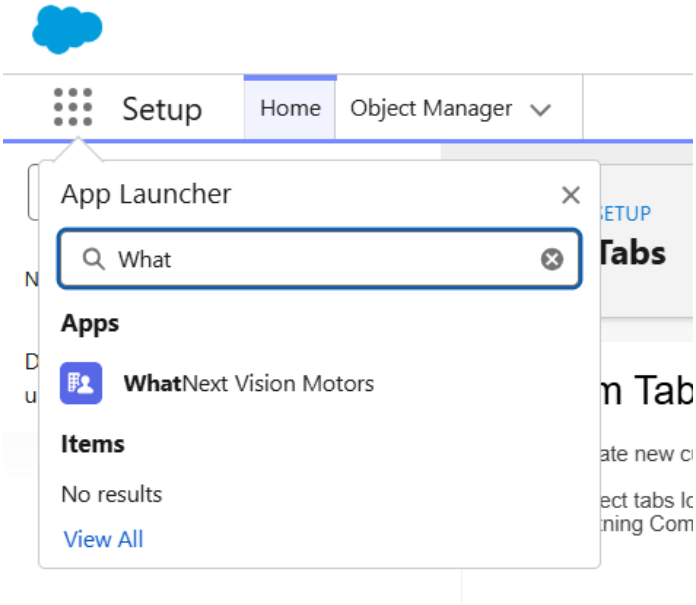

Figure 10 shows the first step in creating the custom WhatNext Vision Motors application. This setup screen is where the main navigation for the CRM was defined. The process involves selecting which tabs and features users will see when they open the application. This creates a dedicated workspace that includes only the tools needed for vehicle sales, making the system easier to learn and use.

Figure 11

Data Management - App Manager

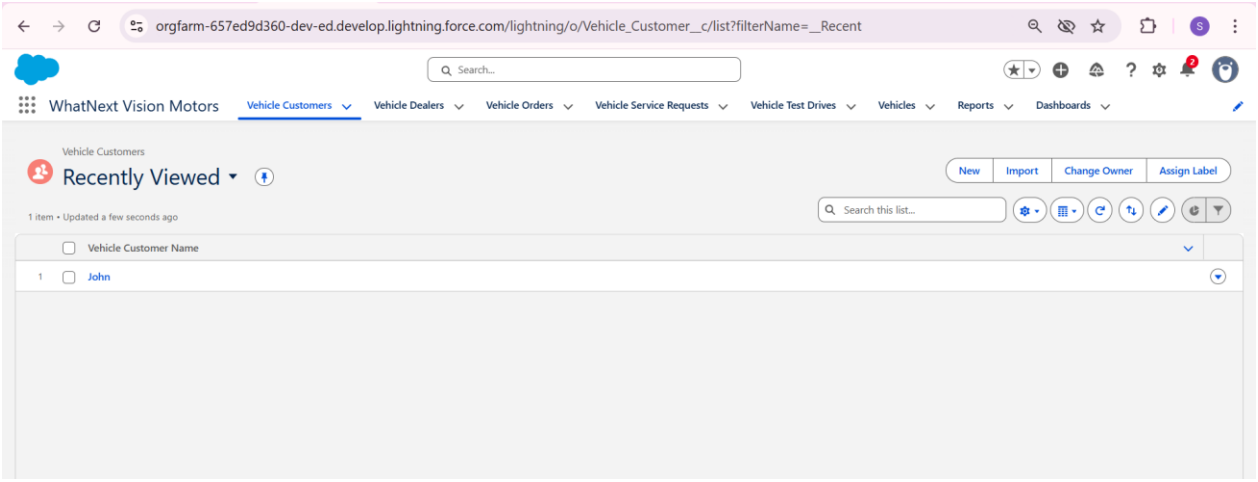

Figure 11 shows the completed WhatNext Vision Motors application with all its components. The App Manager displays the final navigation structure, including tabs for all six custom objects plus Reports and Dashboards. This organized layout provides users with direct access to vehicle customers, dealers, orders, service requests, test drives, and vehicle inventory from a single interface, creating a streamlined workspace for daily operations.

3.2 Page Layouts and Lightning Pages

Custom pages were designed for each object, such as Vehicles, Customers, and Orders. These pages organize information in a clear way, showing important details and related records together. For example, the Vehicle Order page displays customer details, the assigned dealer, and any connected test drives, allowing users to see everything they need on one screen.

Figure 12

Vehicle Customer Page

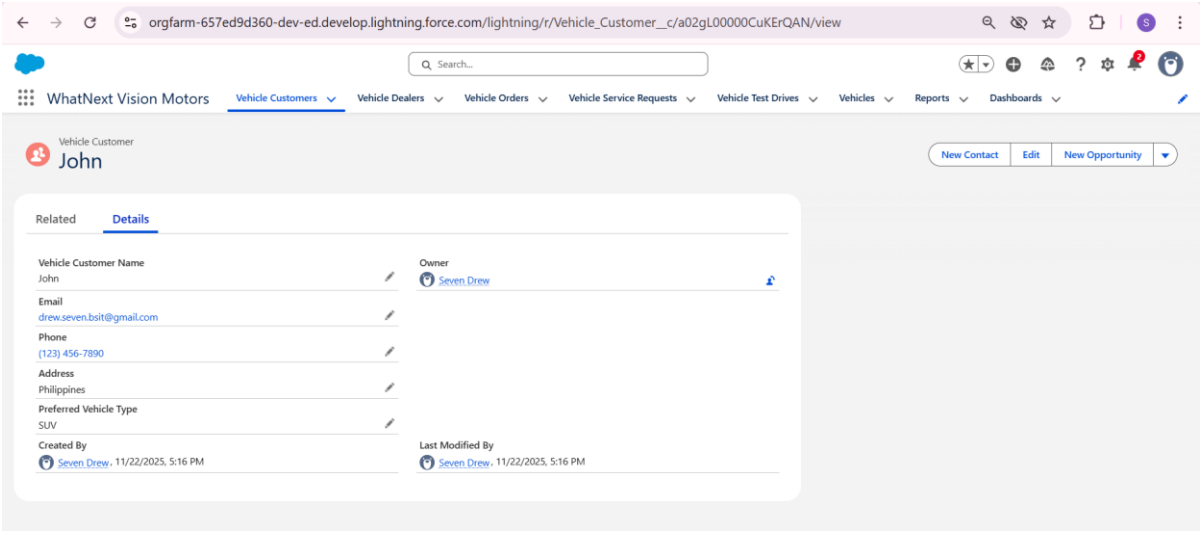

Figure 12 shows the detailed view page for a Vehicle Customer record. This page displays all essential customer information in one place, including contact details, address, and preferred vehicle type. The layout organizes the data clearly, making it easy for sales agents to quickly access customer information during sales conversations or when processing orders. This centralized view helps improve customer service by ensuring all relevant details are readily available.

Figure 13

Vehicle Dealers Page

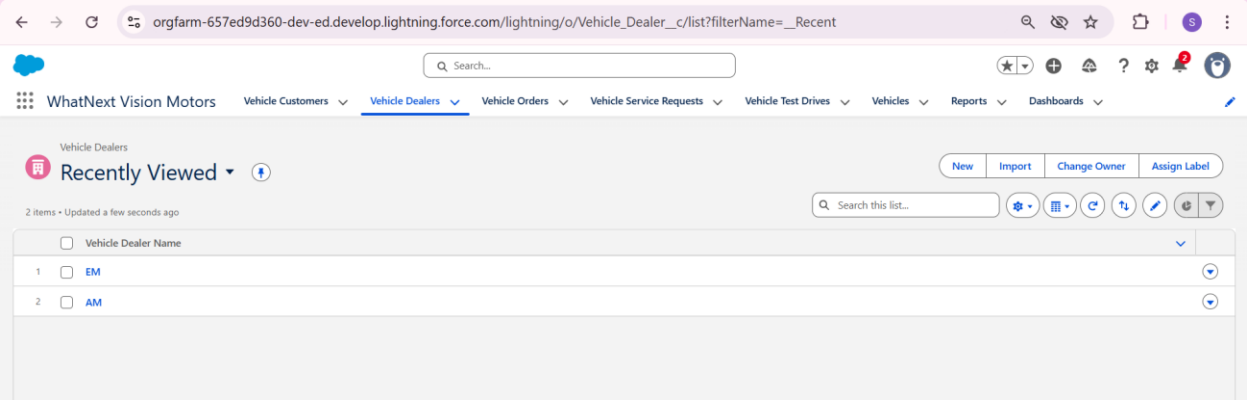

Figure 13 shows the list view of all vehicle dealers in the system. This page displays dealer records with their names, making it easy for sales teams to find and contact the right dealer. The list shows two dealers, GM and AM, who are available to fulfill customer orders. This centralized dealer directory helps ensure orders are properly routed to the correct fulfillment partners.

Figure 14

Vehicle Orders Page

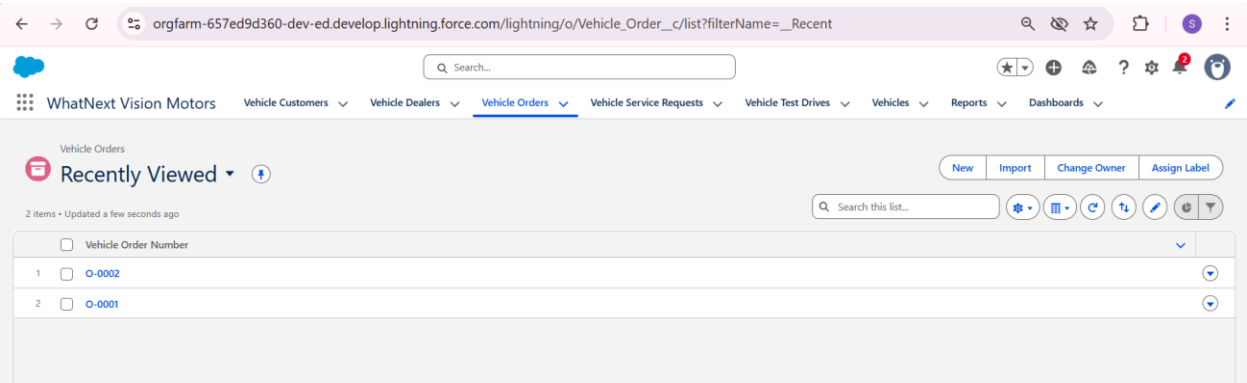

Figure 14 displays the list of all vehicle orders in the CRM. The page shows order numbers like O-0001 and O-0002, allowing users to quickly find and manage specific customer orders. This overview helps sales teams track order status, monitor sales progress, and ensure all customer requests are being processed in a timely manner.

Figure 15

Vehicle Service Requests Page

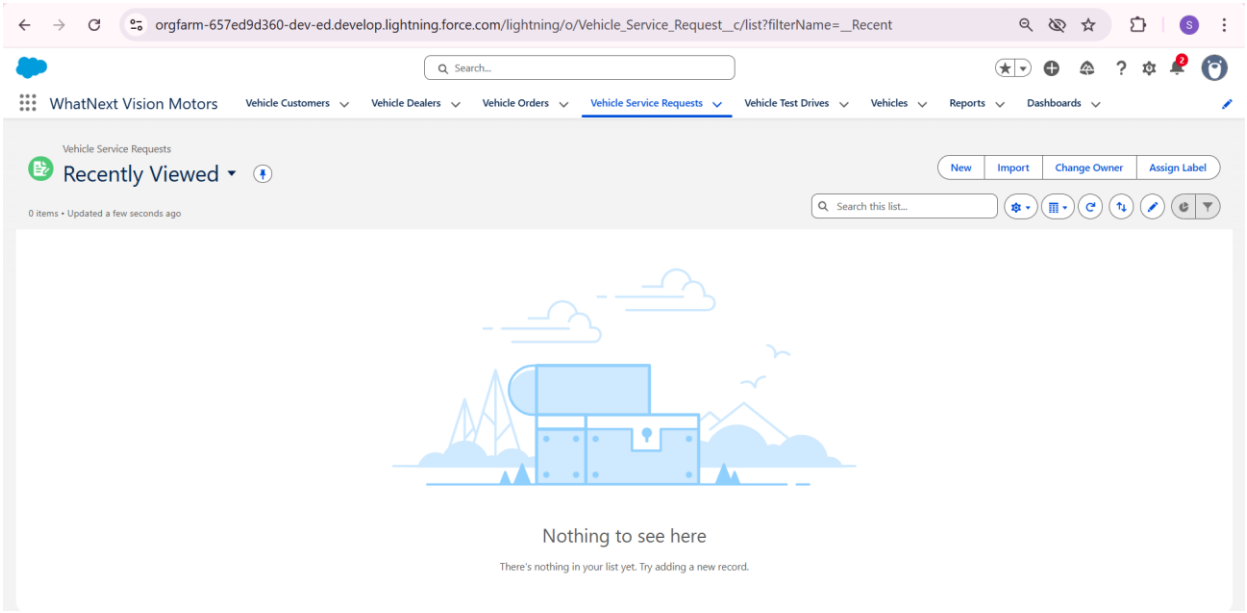

Figure 15 shows the service requests section of the CRM. This area is designed to track customer service needs and vehicle maintenance requests. While no service requests appear in this view, the page is ready to record issues like repairs, part replacements, or other after-sales services. This helps ensure customer vehicles receive proper maintenance and support after purchase.

Figure 16

Vehicle Test Drives Page

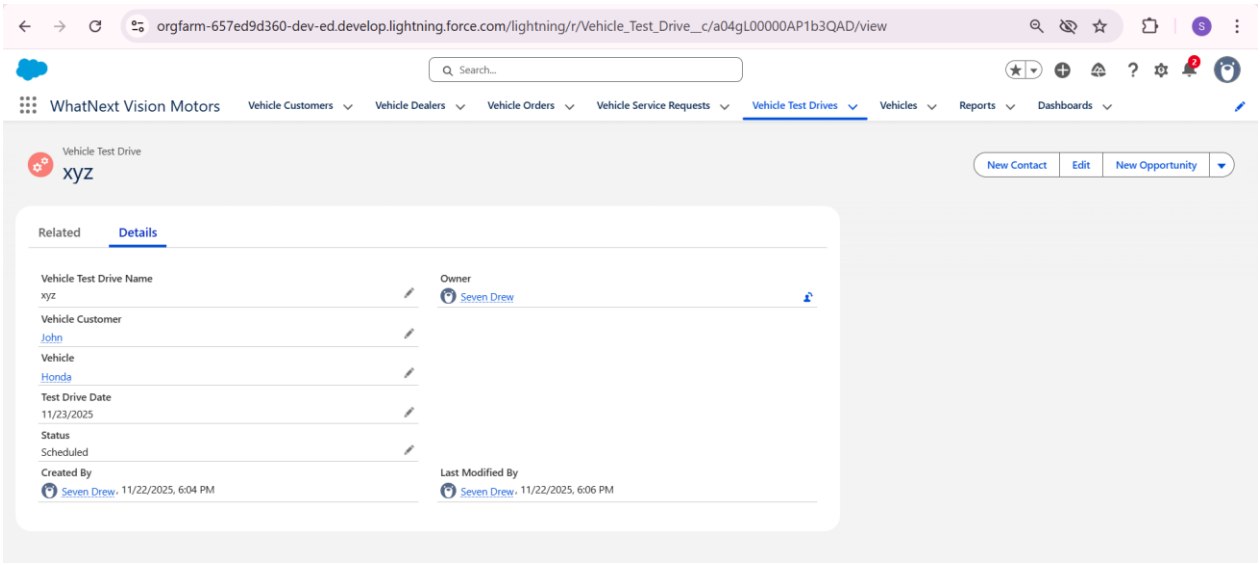

Figure 16 displays a detailed test drive record showing a scheduled appointment. The page captures important information including the customer, vehicle, test drive date, and current status. This organized view helps sales teams manage test drive schedules effectively, track customer interest in specific vehicles, and ensure all appointments are properly coordinated and followed up.

Figure 17

Vehicles Page

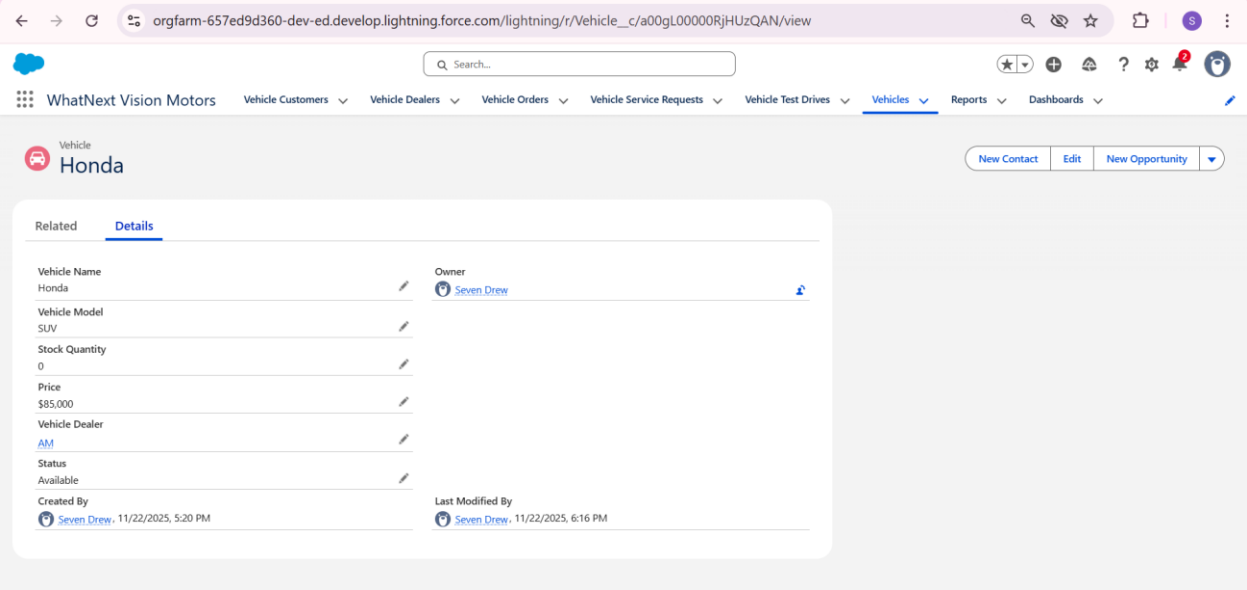

Figure 17 shows the detailed view of a vehicle record in the inventory. This page displays key information about the Honda SUV, including its model, price, stock quantity, and assigned dealer. The stock quantity shows zero, indicating this vehicle is currently out of stock. This detailed view helps sales teams quickly check vehicle availability, pricing, and specifications when assisting customers or processing orders.

Figure 18

Reports Page

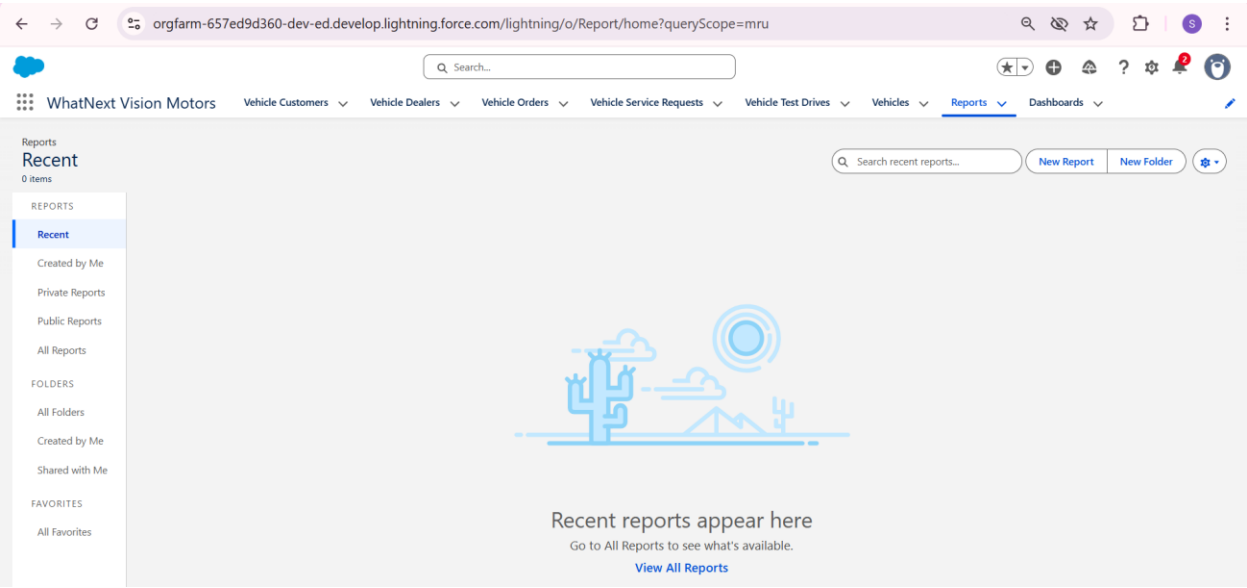

Figure 18 shows the Reports section of the CRM. This area provides access to various reports that can analyze sales data, customer information, and business performance. While specific reports are not yet created in this view, the page is ready to display important business metrics that help managers track sales performance and make informed decisions.

Figure 19

Dashboards Page

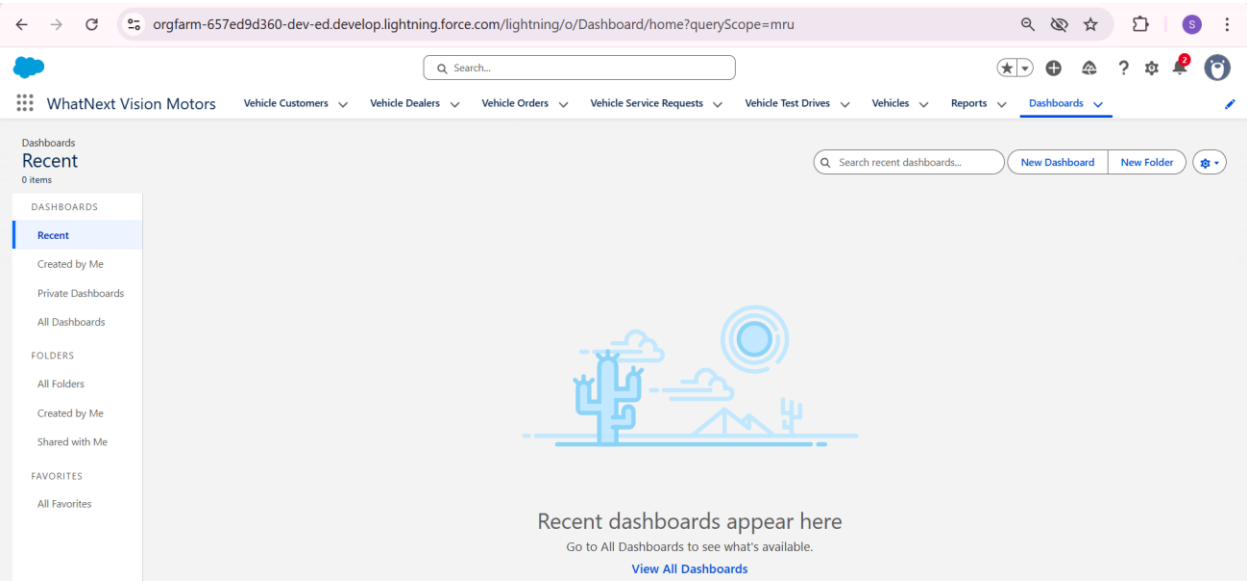

Figure 19 displays the Dashboards section where visual summaries of business data can be viewed. Dashboards provide at-a-glance views of key performance indicators through charts and graphs. This section is prepared to show important metrics like sales trends, inventory levels, and team performance once the dashboards are configured with relevant data.

3.3 User Management

The system uses standard Salesforce profiles to control what different users can see and do. While a detailed role hierarchy was not built for this version, the profile settings ensure that sales agents, dealers, and managers have the right level of access to perform their daily tasks within the application.

3.4 Reports and Dashboards

Formal reports and dashboards are planned for a future update. For now, users can track information using the record pages and related lists, which show connected data like orders linked to a customer or service requests for a vehicle.

Lightning Web Components (LWC)

The interface was built using Salesforce's built-in tools like the Lightning App Builder. Creating custom Lightning Web Components was noted as a valuable future improvement for adding more advanced features.

Phase 4: Data Migration, Testing & Security

This phase ensured the system works correctly, handles data properly, and maintains security standards.

4.1 Data Loading Process

Sample data was created directly in Salesforce to test all system functions. For moving to a live environment, the Data Import Wizard or Data Loader would transfer existing customer, vehicle, and dealer records using CSV files matching the custom object formats.

4.2 Field History Tracking, Duplicate Rules, and Matching Rules

The current system uses validation rules to maintain data quality. Features like Field History Tracking and Duplicate Management are planned for future updates when the business grows and needs more advanced data controls.

4.3 Profiles, Roles and Role Hierarchy, Permission Sets, Sharing Rules

Security was set up using standard Salesforce profiles that control what users can see and do. While detailed role structures were not created for this version, the profile settings properly limit access so users can only work with data relevant to their jobs.

Figure 20

Standard Profiles

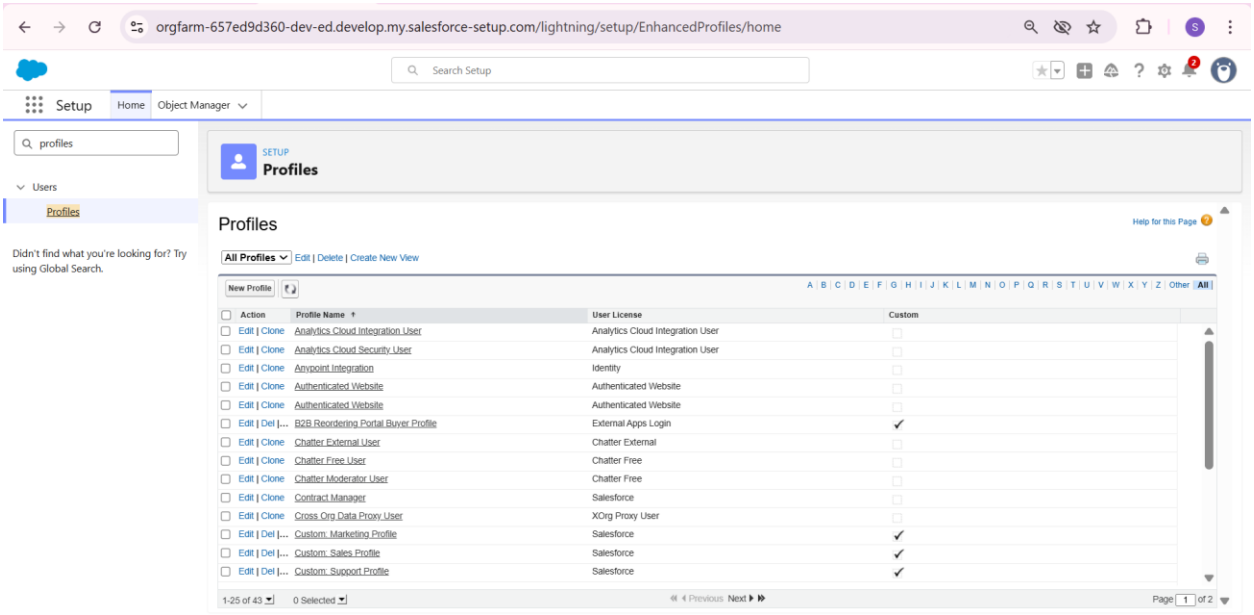

Figure 20 shows the Profiles setup page in Salesforce. This is where user access permissions are managed. The page displays the list of standard profiles available in the system, such as System Administrator and Standard User. These profiles control what different users can see and do within the application, ensuring each team member has the right level of access for their job responsibilities.

4.4 Creation of Test Classes

Thorough testing was done manually to verify all automated processes work as intended. Writing formal automated test classes is planned for future development to meet enterprise code quality standards.

4.5 Testing Approach and Test Cases

A detailed testing process checked all major system functions. Key tests included:

Figure 21

Test Case 1: Auto-Dealer Assignment Flow

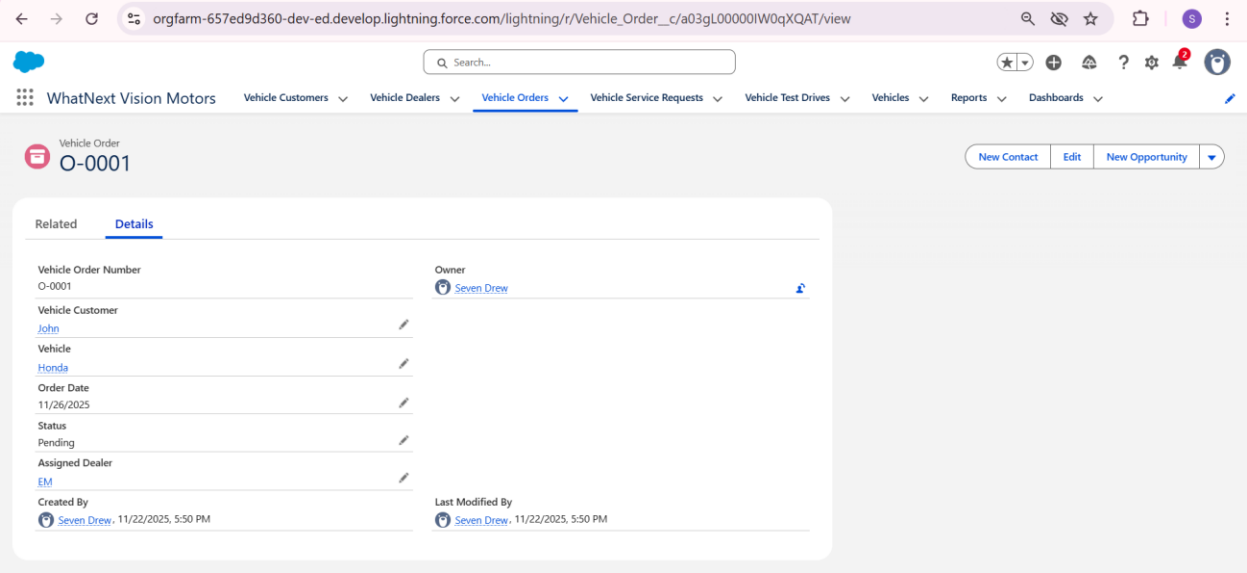

Figure 21 shows a successful test of the automatic dealer assignment feature. The Vehicle Order O-0001 displays that dealer "EM" was automatically assigned to the order based on the customer's location. This test confirms that the flow works correctly by matching customers with their nearest available dealer without manual intervention from sales staff.

Figure 22

Test Case 2: Vehicle Order Creation with Stock Validation

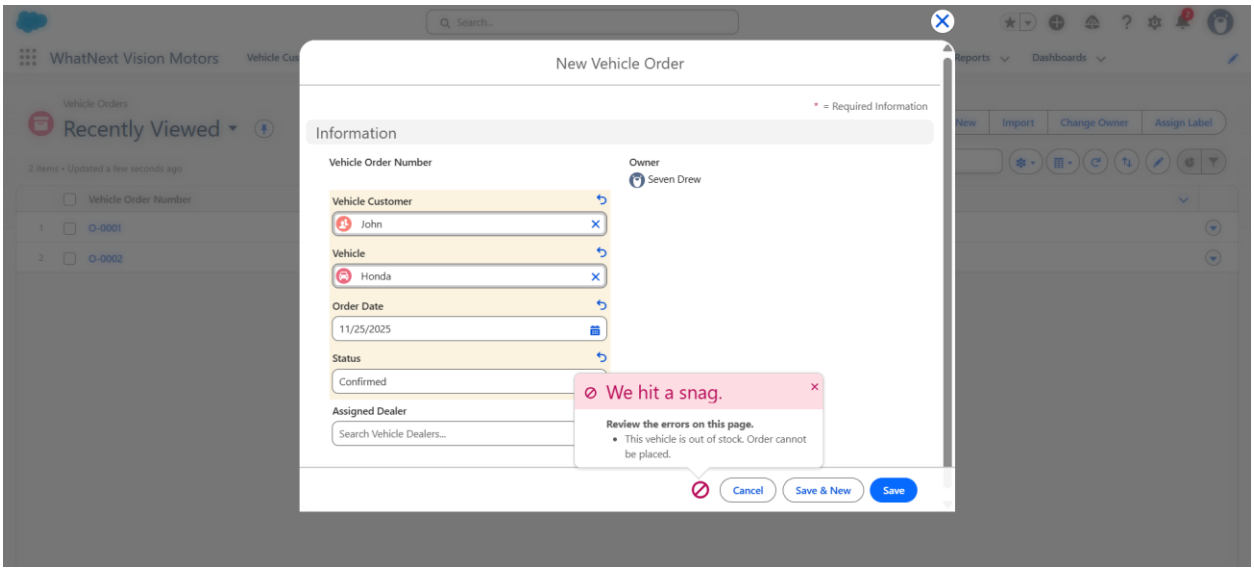

Figure 22 shows the system successfully preventing an order for an out-of-stock vehicle. When trying to create an order for the Honda vehicle, the system displays an error message: "This vehicle is out of stock. Order cannot be placed." This test confirms that the stock validation trigger is working correctly to prevent invalid orders and maintain accurate inventory management.

Figure 23

Test Case 3: Test Drive Reminder Automation

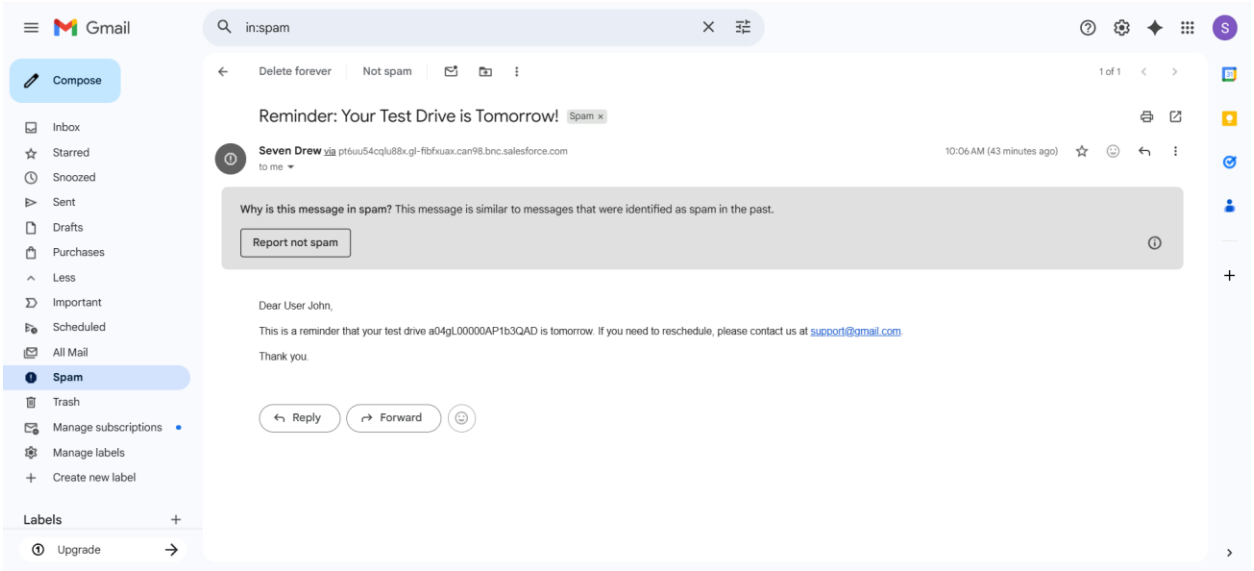

Figure 23 shows the automated email reminder sent to customers before their scheduled test drive. The email includes the test drive reference number and provides contact information for rescheduling. This test confirms that the reminder system works properly, helping to reduce missed appointments and improve customer communication.

Phase 5: Deployment, Documentation & Maintenance

This phase outlines the plan for launching the system to users, the ongoing support procedures, and the long-term maintenance strategy.

5.1 Deployment Strategy

The complete CRM solution was developed within a single Salesforce Developer Edition organization. For a future rollout to a production environment, the recommended method is to use Salesforce Change Sets. This approach allows for the controlled and auditable transfer of all custom components—including objects, fields, automation Flows, Apex code, and the Lightning app configuration—from a development space to a live business environmen.

5.2 System Maintenance and Monitoring

To ensure the system continues to operate effectively, regular maintenance checks will be necessary. This includes:

- Monitoring the execution logs of key automations, such as the Auto-Assign Dealer Flow and the Vehicle Order Batch job, to confirm they are running successfully.
- Checking email delivery logs to verify that test drive reminders are being sent to customers.
- Conducting periodic reviews of data quality and inventory levels to ensure accuracy.
- Staying updated with Salesforce's seasonal releases (Spring, Summer, Winter) to test for and address any impact on existing customizations.

5.3 Troubleshooting Approach

A structured method for resolving system issues is essential for maintenance. The recommended process is:

1. Identify: Clearly define the problem by replicating the issue and gathering any specific error messages.
2. Isolate: Use Salesforce's built-in diagnostic tools to find the root cause:

a. Debug Logs: For Apex Triggers and Batch Classes, use debug logs to trace the code's execution and check variable values.
b. Flow Debug Mode: For the Auto-Assign Dealer and Test Drive Reminder flows, use the debug functionality to test the flow logic and find errors.

3. Resolve: Once the cause is found, apply and test the fix in a sandbox environment before deploying it to the production system using Change Sets.

Conclusion

The WhatNext Vision Motors CRM project has successfully delivered a centralized and automated platform that streamlines the entire vehicle sales process. By implementing a custom data model, robust automation with Flows, and programmatic logic with Apex Triggers and Batch Jobs, the system meets its core objectives. It enhances customer management, ensures accurate real-time stock control, automates dealer assignment, and improves the overall customer experience.

The CRM effectively reduces manual tasks, prevents errors in the ordering process, and establishes a solid foundation for daily sales operations. This scalable architecture also allows for future enhancements, such as advanced analytics with Reports and Dashboards, AI-driven vehicle recommendations, or a customer self-service portal, to deliver continued value as WhatNext Vision Motors grows.