

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

ABSTRACT

WhatsNext Vision Motors is currently in the process of deploying a comprehensive Salesforce CRM system to revolutionize both the customer journey and the effectiveness of their internal processes within the automotive industry. A fundamental aspect of this solution involves a data framework that unifies car inventory details, dealership locations, and customer relationship management. This CRM system employs automated processes and Apex triggers to uphold crucial business guidelines, such as preventing orders for vehicles that are currently unavailable and automatically routing new orders to the dealership closest to the customer's location, determined through geolocation.

In addition to these real-time protective features, this project also incorporates scheduled routines; these include batch Apex processes that routinely update stock levels across large quantities of orders, along with timed Apex jobs that send out email notifications reminding customers about upcoming test drives and prompting for stock resupply. Working together, these elements create an integrated platform designed to provide clear and reliable tracking of orders, which are labeled with statuses like "Awaiting Confirmation" or "Confirmed" depending on the current stock situation.

The technical rollout features well-organized data structures for car and dealer information, the use of Lightning App Builder to create easy-to-use interfaces, event-driven flows that enable flexible logic processing, and reliable Apex programming for checks and automated functions. This project aims to greatly improve how happy customers are and how much the company can produce by lessening the need for staff involvement, smoothing out any difficulties that customers might face, and reacting quickly to shifts in stock availability. Ultimately, WhatsNext Vision Motors seeks to become a more flexible and customer-focused company.

OBJECTIVE

This initiative seeks to improve how customers place orders by using Salesforce automation and smart information management. The goal is to make dealer allocation easier, confirm inventory availability instantly, and deliver precise order status notifications. Through these steps, WhatsNext Vision Motors hopes to boost both customer happiness and how well its mobile services work.

- Better Customer Journey: Making ordering simpler by automatically suggesting dealers and providing current updates.
- Up-to-Date Inventory Control: Stopping orders for items not in stock by precisely checking what's available.
- Automatic Dealer Allocation: Sending orders to the closest dealer depending on where the customer is located.
- Smooth Order Handling: Utilizing planned procedures to keep order details current and speed up completion.
- Intelligent Salesforce Connection: Employing Apex, Flows, and Batch Jobs to grow and automate activities.

TECHNOLOGY DESCRIPTION

Salesforce:-

Salesforce is a CRM system operating in the cloud, allowing companies to handle their customer relationships, information, and workflows effectively. For this specific undertaking, it serves the purpose of holding and structuring data about vehicles, locations of dealerships, and purchase requests from customers. The system allows for automated tasks using Flows, Apex triggers, and scheduled processes, guaranteeing precise inventory checks and intelligent allocation of dealers. Utilizing Salesforce enables WhatsNext Vision Motors to improve how they run their business and offer a smooth, customer-focused process for placing orders.

Custom Objects:-

Objects in salesforce are like tables in a database. Custom Objects are created to store specific data.

Example:

- Vehicle_c - Stores vehicle details
- Vehicle_Dealer_c - Stores authorized dealer info
- Vehicle_Customer_c - Stores customer details
- Vehicle_Order_c - Tracks vehicle purchases
- Vehicle_Test_Drive_c - Tracks test drive bookings
- Vehicle_Service_Request_c - Tracks vehicle servicing requests

Tabs:-

Tabs are used to display data in the Salesforce UI

Example: A tab for vehicle_c allows users to easily view, create, and manage vehicle

Custom APP:-

An App in Salesforce is a collection of tabs grouped together for a specific business purpose, allowing users to access related data and functionality in one place.

Fields:-

A **fields** in Salesforce is a specific piece of data stored in an object, much like a column in a spreadsheet. Each field holds a particular type of information such as text, date, number, or picklist values.

Example: Vehicle_Model_c – A **Picklist** field in the Vehicle_c object with options like *Sedan, SUV, EV*, etc.

Flows:-

Flows enable the automation of business processes through a no-code approach. They have the ability to automatically generate or modify records, as well as dispatch alerts.

Example:-

- An email notification is automatically sent when a new order is created due to a flow. Apex:-

Apex serves as Salesforce's programming language, utilizing an object-oriented structure. It gives developers the capability to implement tailored logic.

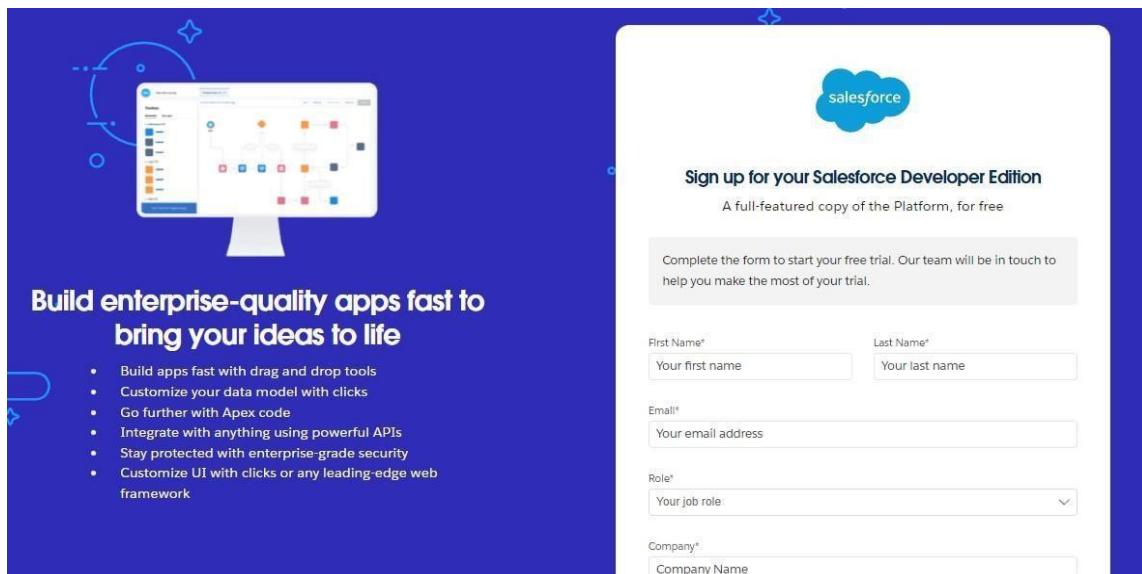
Example Triggers:

- Upon the creation of a new vehicle order, a trigger instantly verifies the stock levels.
- It stops the processing of orders if the specified vehicle is currently unavailable in the inventory.
- Upon the successful confirmation of an order, the trigger amends the stock count by decreasing it accordingly.

DETAILED EXECUTION OF PROJECT PHASES

1. Developer Org Setup

- A Salesforce Developer Org was Created Using
<https://developer.salesforce.com/signup>



- The account was verified, password set , and access was granted to the Salesforce Setup page.

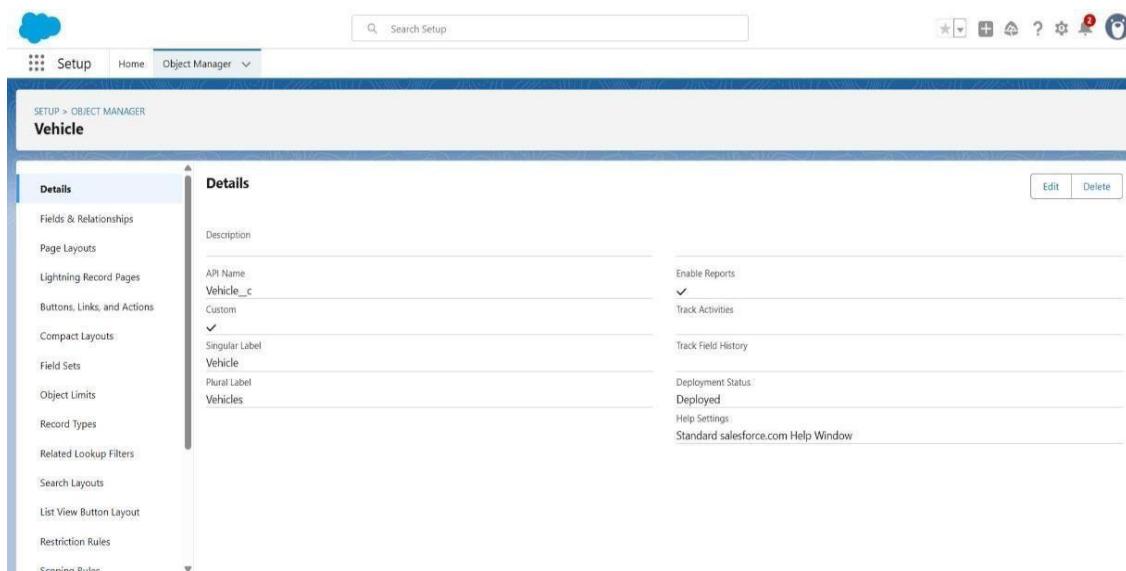
2. Custom Object Creation

Six Custom Objects were created to store business-critical data

- **Vehicle** – Stores vehicle details like model, stock, price, and status.
- **Vehicle Dealer** – Contains information about dealers such as location and contact details.
- **Vehicle Order** – Tracks customer orders, order dates, and order status.
- **Vehicle Customer** – Maintains customer details including contact and preferred vehicle type.
- **Vehicle Test Drive** – Records test drive schedules and status.
- **Vehicle Service Request** – Logs service requests, dates, issues, and progress status.

Steps followed:

- Navigated to Setup → Object Manager → Create → Custom Object.
- Provided Label, Name, and enabled options for Reports and Search.
- Saved and created Tabs for each object.



3. Creating a Custom Tab

The screenshot shows the Salesforce Setup interface with the 'Tabs' page selected. The top navigation bar includes 'Setup', 'Home', and 'Object Manager'. A search bar at the top right says 'Search Setup'. Below the navigation, there's a sidebar with 'User Interface' and 'Rename Tabs and Labels' sections, and a note about using Global Search.

The main content area is titled 'Custom Tabs' and contains three sections:

- Custom Object Tabs:** A table listing tabs for 'Vehicle Customers', 'Vehicle Dealers', 'Vehicle Orders', 'Vehicles', 'Vehicle Service Requests', and 'Vehicle Test Drives'. Each row includes an 'Edit | Del' link and a 'Tab Style' icon.
- Web Tabs:** A section stating 'No Web Tabs have been defined'.
- Visualforce Tabs:** A section stating 'No Visualforce Tabs have been defined'.

4. Creating the Lightning App

- A custom Lightning App named WhatNext Vision Motors was created.
- Included tabs: Vehicle, Vehicle Dealer, Vehicle Order, Vehicle Customer, Vehicle Test Drive, Vehicle Service
- Assigned to the System Administrator profile.

The screenshot shows the Lightning App Builder interface with the 'App Details & Branding' page selected. The top navigation bar includes 'Lightning App Builder', 'App Settings', 'Pages', and the app name 'WhatNext Vision Motors'. A help link is at the top right.

The left sidebar shows 'App Settings' and 'App Details & Branding' is selected. Under 'App Details & Branding', there are sections for 'App Details' and 'App Branding'.

App Details:

- * App Name: WhatNext Vision Motors
- * Developer Name: WhatNext_Vision_Motors
- Description: An app to manage vehicles, dealers, customers, orders, test drives, and service requests.

App Branding:

- Image: A placeholder box with an 'Upload' button.
- Primary Color Hex Value: #007002
- Org Theme Options: A checkbox for using the app's image and color instead of the org's custom theme.

App Launcher Preview: Shows a preview of the app icon (a blue square with 'WV') and the app name 'WhatNext Vision Motors' along with its description.

5. Creating Fields

Custom fields were created on each object to store specific business data.

For example, on the Vehicle_c object, a Picklist field named Vehicle_Model_c was created with values like *Sedan*, *SUV*, and *EV*.

Steps followed:

- Navigated to Setup → Object Manager → [Object Name] → Fields & Relationships → New.

The screenshot shows the Salesforce Object Manager interface. The top navigation bar includes 'Setup', 'Home', and 'Object Manager'. The main title is 'SETUP > OBJECT MANAGER' followed by 'Vehicle'. On the left, there's a sidebar with various configuration options like 'Details', 'Fields & Relationships', 'Page Layouts', etc. The 'Fields & Relationships' section is currently selected and displays a table titled 'Fields & Relationships' with 9 items. The columns are 'FIELD LABEL', 'FIELD NAME', 'DATA TYPE', 'CONTROLLING FIELD', and 'INDEXED'. The table lists fields such as Created By, Dealer, Last Modified By, Owner, Price, Status, Stock Quantity, Vehicle Model, and Vehicle Name, each with its corresponding data type and whether it's indexed or not.

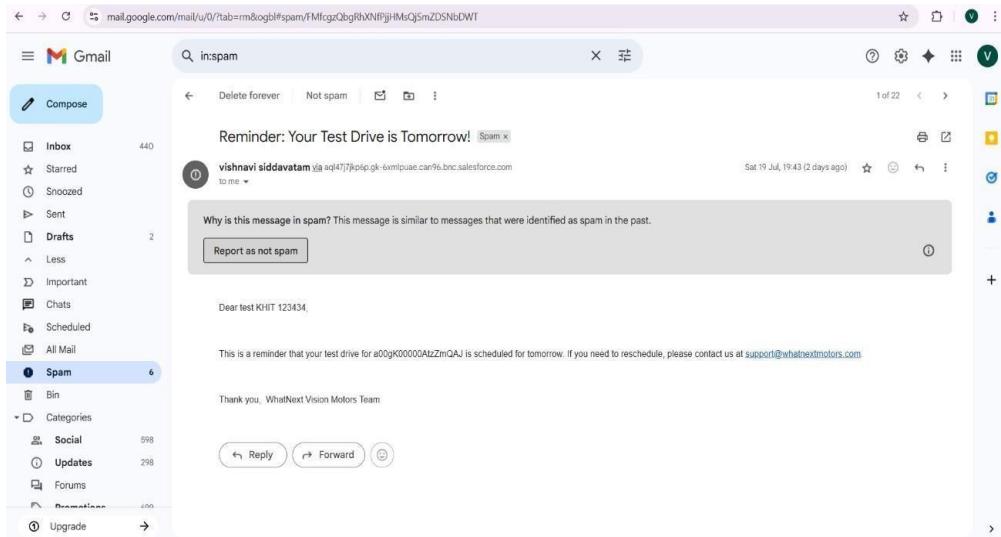
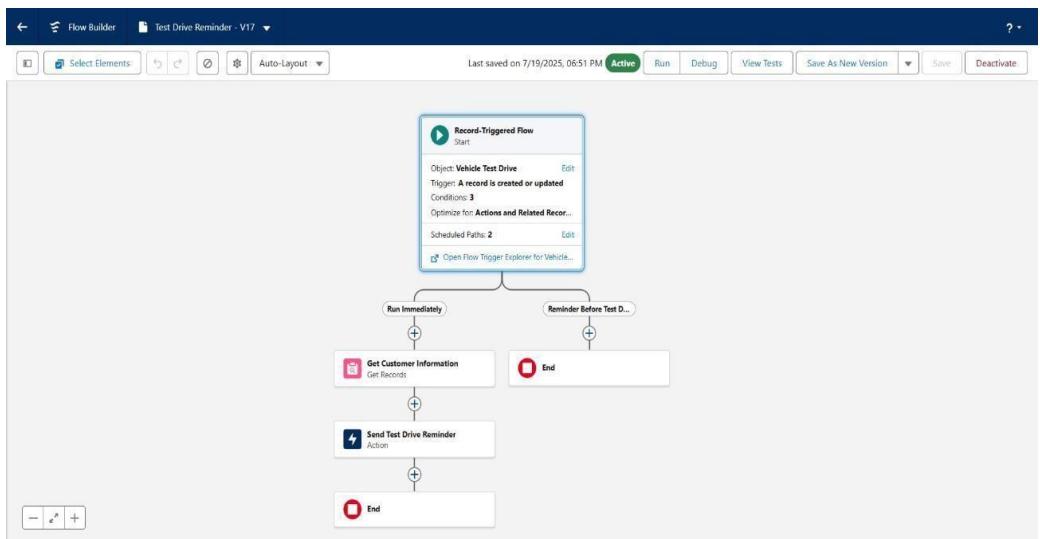
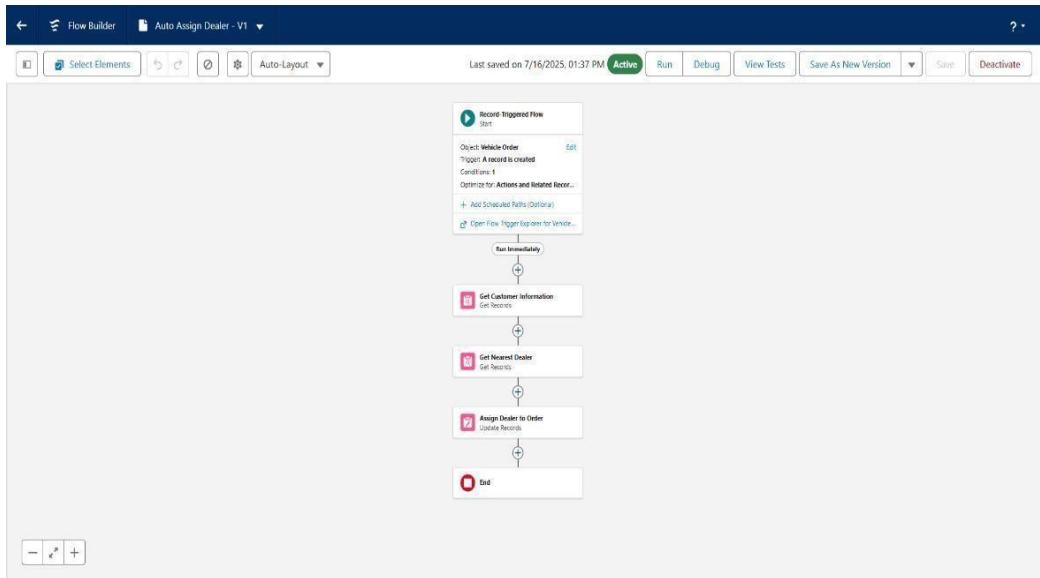
6. Creating Flows

Record-triggered flows were created to automate key business processes without code. For example, a flow was built to **auto-assign the nearest dealer** when a vehicle order is created with status *Pending*.

Another flow sends **email reminders** to customers **one day before** their scheduled test drive.

Steps followed:

- Setup → Flows → New Flow → Record-Triggered Flow.



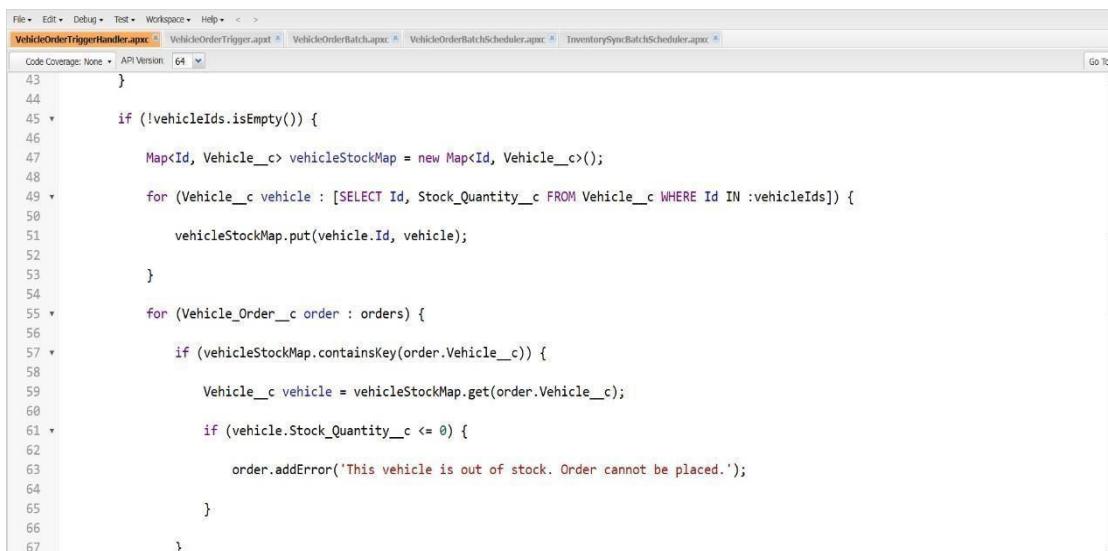
7. Create Apex and Trigger Batch Jobs:

Apex classes and triggers were created to enforce business logic during order processing.

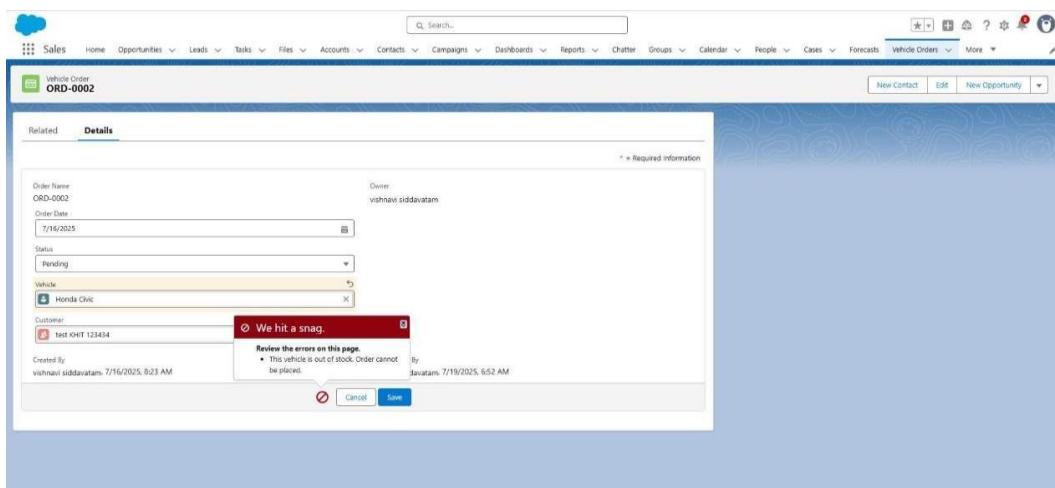
- A **trigger** checks vehicle stock when an order is placed and updates stock on confirmation.
- A **batch class** regularly scans pending orders and updates them to *Confirmed* if stock becomes available.
- A **scheduled class** runs the batch job daily to automate order updates.

Steps followed:

- Developer Console → File → New → Apex Class.



```
File Edit Debug Test Workspace Help < >
VehicleOrderTriggerHandler.apex VehicleOrderTrigger.apxt VehicleOrderBatch.apxc VehicleOrderBatchScheduler.apxc InventorySyncBatchScheduler.apxc
Code Coverage: None API Version: 64 Go To
43     }
44
45     if (!vehicleIds.isEmpty()) {
46
47         Map<Id, Vehicle__c> vehicleStockMap = new Map<Id, Vehicle__c>();
48
49         for (Vehicle__c vehicle : [SELECT Id, Stock_Quantity__c FROM Vehicle__c WHERE Id IN :vehicleIds]) {
50
51             vehicleStockMap.put(vehicle.Id, vehicle);
52
53         }
54
55         for (Vehicle_Order__c order : orders) {
56
57             if (vehicleStockMap.containsKey(order.Vehicle__c)) {
58
59                 Vehicle__c vehicle = vehicleStockMap.get(order.Vehicle__c);
60
61                 if (vehicle.Stock_Quantity__c <= 0) {
62
63                     order.addError('This vehicle is out of stock. Order cannot be placed.');
64
65                 }
66
67             }
68
69         }
70
71     }
72 }
```



8. Scheduled Jobs

Within Salesforce, the Scheduled Apex Jobs area lists all Apex jobs set up to execute on their own, including tasks like batch processes, classes designed to be scheduled, and jobs that can be queued. This section presents vital information such as the designation of the job, its classification, its current state (whether it's waiting, actively running, finished, or unsuccessful), the date and time it was submitted, when it's slated to run next, and the individual responsible for submitting it. From this location, those with administrative rights have the ability to oversee the advancement of jobs, look for any problems, and take control of jobs by stopping them, putting them on hold, or removing them if necessary. This functionality aids in guaranteeing that automated operations function without interruption and that any problems are found and taken care of without delay.

The screenshot shows the Salesforce Setup interface. The left sidebar has a 'Jobs' section with 'Scheduled Jobs' selected. The main content area is titled 'Scheduled Jobs' and displays a table of scheduled Apex jobs. The table columns are: Action, Job Name, Submitted By, Submitted, Started, Next Scheduled Run, Type, and Cron Trigger ID. The table contains the following data:

Action	Job Name	Submitted By	Submitted	Started	Next Scheduled Run	Type	Cron Trigger ID
Manage Del Pause Job	Daily Inventory Sync	siddavatam_vishnavi	7/16/2025, 4:48 AM	7/21/2025, 12:02 AM	7/22/2025, 12:00 AM	Scheduled Apex	08egK000007WP45
Manage Del Pause Job	Daily Vehicle Order Processing	siddavatam_vishnavi	7/16/2025, 4:13 AM	7/20/2025, 12:02 PM	7/21/2025, 12:00 PM	Scheduled Apex	08egK000007W15P
Del	Analytics Data Loader Job for Org: 00Dgk000006Mfp	User_Integration	7/7/2025, 11:28 AM	7/20/2025, 6:24 AM	7/21/2025, 6:24 AM	Autonomous Data Loader Job	08egK000006oxBE
	Program Milestone Computation Cron Job	Process, Automated	7/7/2025, 11:28 AM	7/21/2025, 12:00 AM	7/21/2025, 6:59 AM	Program Milestone Computation Cron Job	08egK000006oxBC
	Program Status Update Cron Job	Process, Automated	7/7/2025, 11:28 AM	7/20/2025, 8:01 PM	7/21/2025, 5:00 AM	Program Status Update Cron Job	08egK000006x8D

PROJECT EXPLANATION WITH REAL-WORLD EXAMPLE

1. Customer Registration

Real-Life Example:

A customer, **Elijah Mikaelson**, visits the automobile showroom or website.

In Salesforce:

- A new record is created in the **Vehicle_Customer__c** object with his details such as Name, Phone, Email, and Address.
- **Validation Rule:** Ensures the email is valid (e.g., must contain "@gmail.com"). If invalid, an error is shown.

2. Vehicle & Inventory Management

Real-Life Example:

The store admin adds **cars like BMW X5, Toyota Fortuner, etc.**, with stock quantity (e.g., 5 BMWs and 3 Fortuners).

In Salesforce:

- Each vehicle is stored in the **Vehicle__c** object with details like Price, Model, and Stock Quantity.
- **Validation Rule:** Prevents adding a vehicle with negative stock values.

3. Order Placement

Real-Life Example:

Elijah places an order for **1 BMW X5** (Price ₹50,00,000).

In Salesforce:

- A new record is created in the **Vehicle_Order__c** object.
- **Apex Trigger (VehicleOrderTrigger):**
 - Automatically calculates **Total Amount = Price × Quantity**.
 - Checks **stock availability**:
 - If stock > 0 → **Status__c = Confirmed**.
 - If stock = 0 → **Status__c = Pending**.

4. Inventory Update

Real-Life Example:

When Elijah's order is confirmed, the BMW stock decreases by 1 (from 5 → 4).

In Salesforce:

- **Apex Trigger (VehicleOrderTriggerHandler):**
 - Updates **Vehicle__c.Stock_Quantity__c** by subtracting the quantity ordered.
 - Prevents stock from going below zero.

5. Pending → Confirmed Orders (Stock Refill)

Real-Life Example:

If BMW stock is 0 and Elijah's order was pending, when the admin refills **5 BMWs**, the system **automatically confirms pending orders** (if stock is available).

In Salesforce:

- **VehicleTrigger (After Update):**
 - Detects stock refill.
 - Calls `confirmPendingOrders()` in the **VehicleOrderTriggerHandler**.
 - Updates all pending orders to **Confirmed** until stock runs out.

6. Scheduled Batch Processing

Real-Life Example:

At the end of each day, a **batch job** runs to update **any remaining pending orders** automatically.

In Salesforce:

- **VehicleOrderBatch & VehicleOrderBatchScheduler:**
 - Confirm pending orders if stock is available.
 - Sends admin a log of processed orders.

7. Email Notifications (Flow)

Real-Life Example:

When an order is **confirmed** or **pending**, Elijah gets an **email notification**. In

Salesforce:

- **Record-Triggered Flow:**

- Sends an email alert:
"Your BMW X5 order is confirmed!"
or
"Your order is pending due to low stock."

8. Users and Roles

Real-Life Example:

- **Niklaus Mikaelson** (Sales Manager) handles customer orders.
- **Kol Mikaelson** (Inventory Manager) manages vehicle stock.

In Salesforce:

- Roles and Profiles:

- **Sales Role:** Access to Vehicle_Order_c and Vehicle_Customer_c.
- **Inventory Role:** Access to Vehicle_c and stock management.

9. Real-Life Benefit

This system ensures:

- **No manual stock checks** (triggers handle it automatically).
- **Automatic order status updates** (Confirmed or Pending).
- **Smooth communication with customers** via email alerts.
- **Dealer assignment** (optional via Flow) ensures orders are assigned to the nearest dealer.

SCREENSHOTS

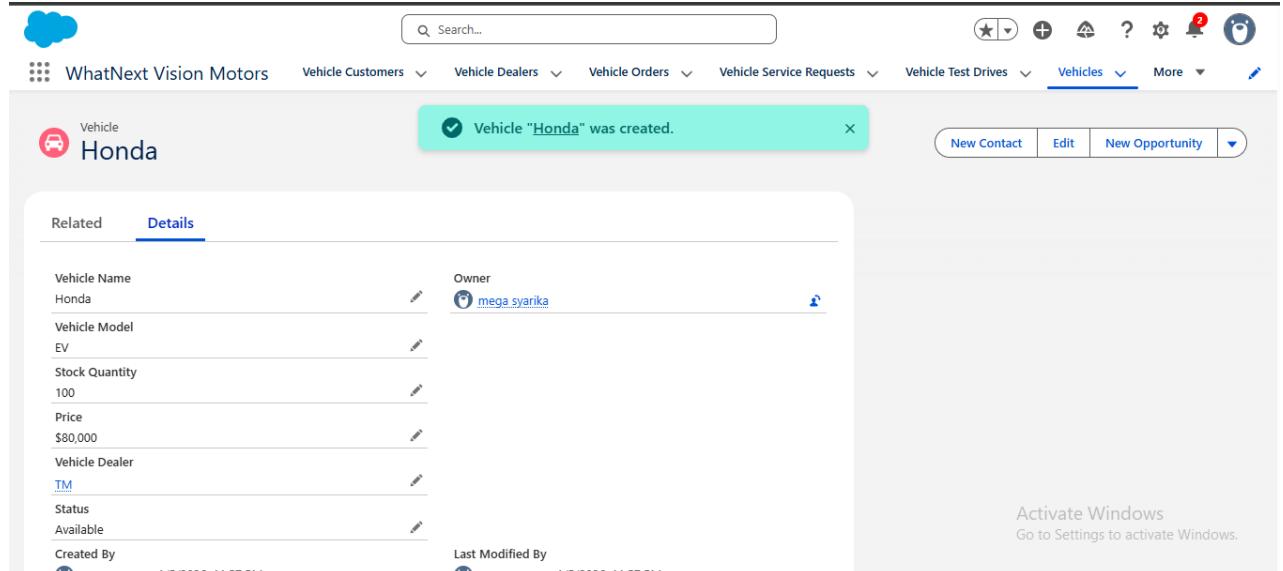


Fig: Vehicles in WhatNext Vision Motors

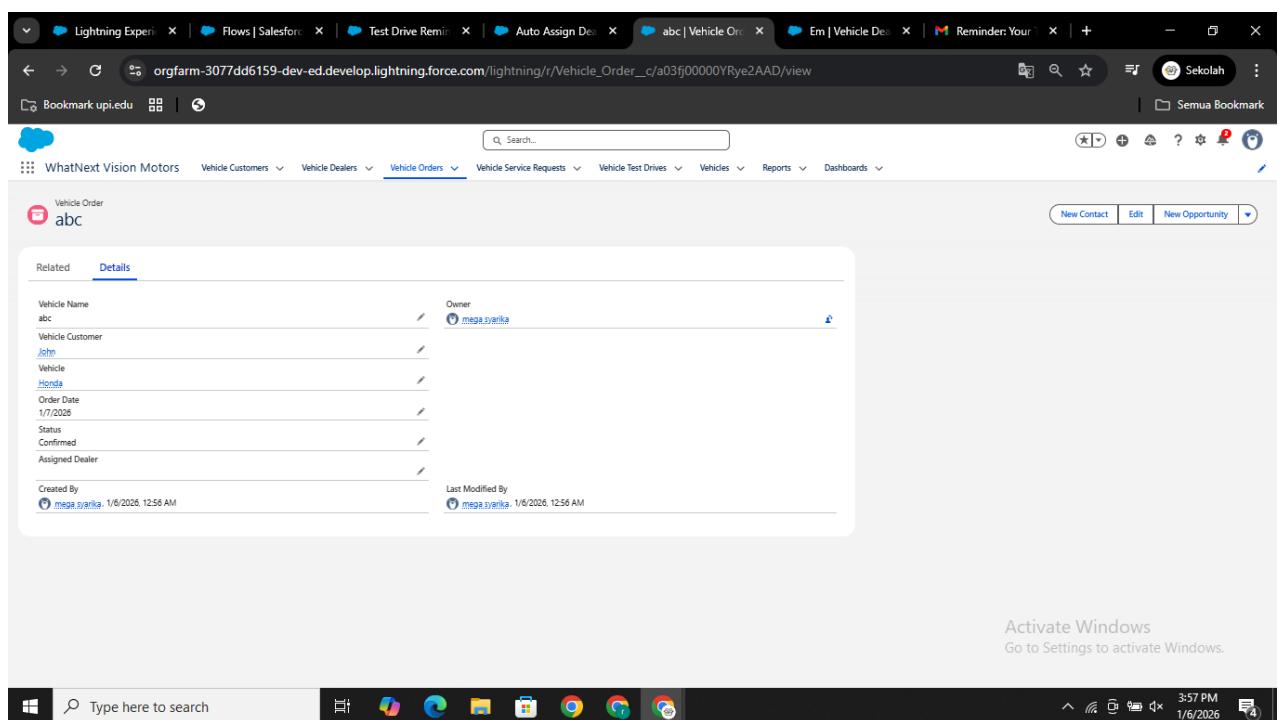


Fig: Customers in WhatNext Vision Motors

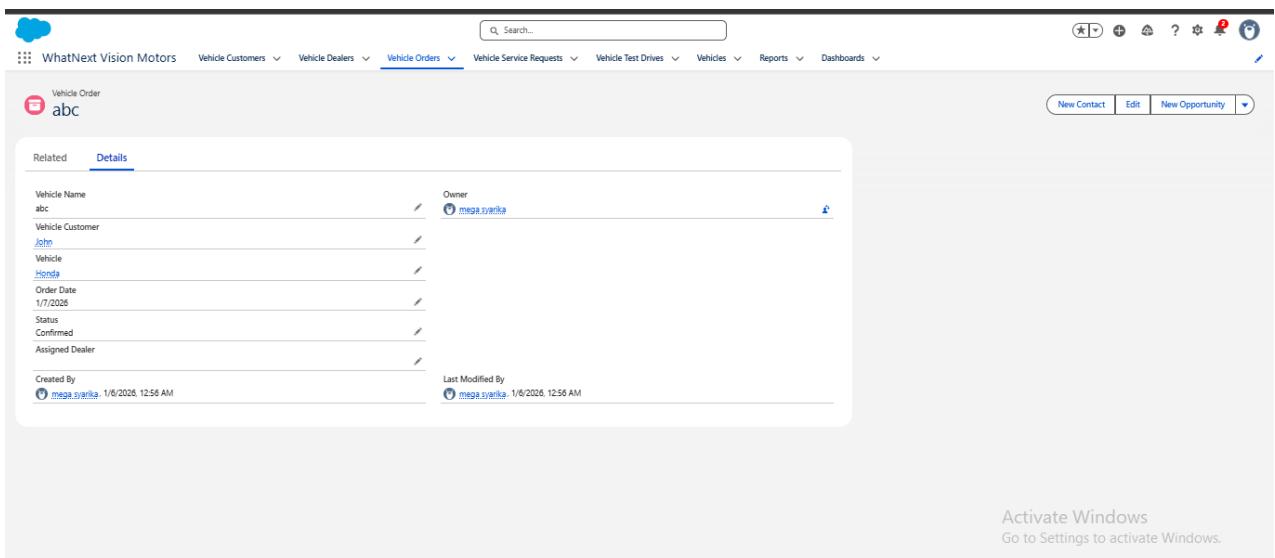


Fig: Vehicle Orders in WhatNext Vision Motors

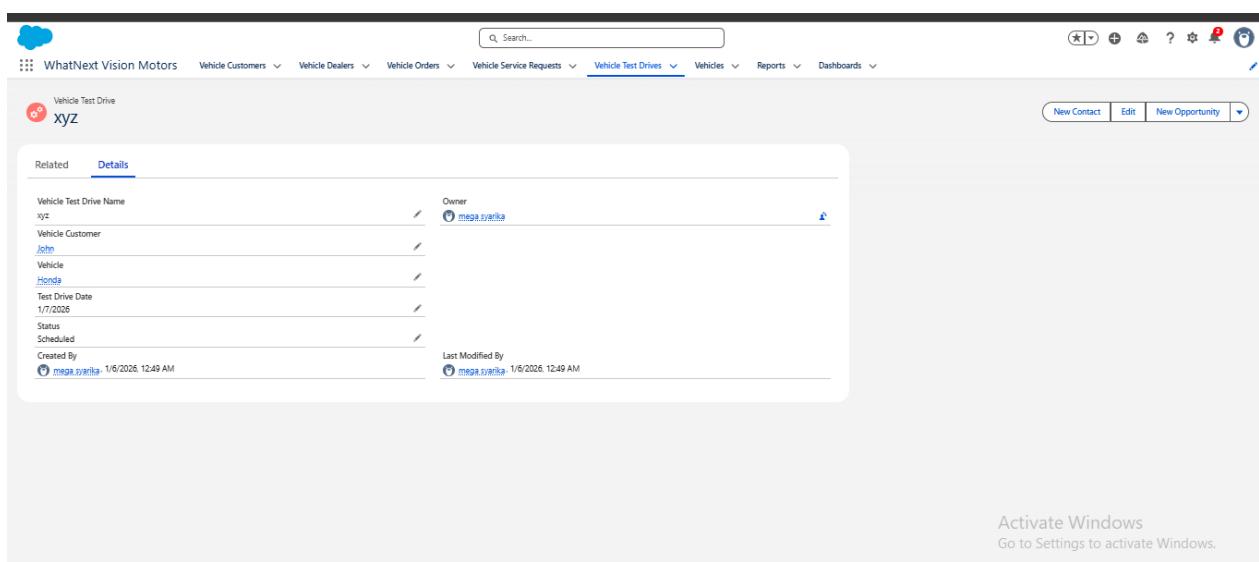


Fig: Test Drives in WhatNext Vision Motors

CONCULSION

WhatNext Vision Motors utilizes Salesforce to make its car sales and maintenance tasks more efficient. The business employs tailor-made items, automatic processes, and Apex triggers to guarantee smooth order handling, current inventory control, and better interaction with customers. This shift to digital methods boosts output, lowers mistakes, and offers a trouble-free experience for workers and buyers, setting up WhatNext Vision Motors for lasting expansion and accomplishments.

Future Scope

1. Customer Portal Integration

- Develop a **Salesforce Community Portal** where customers can log in to:
 - View their order history and current order status.
 - Check loyalty program points and rewards.
 - Raise queries or service requests directly.

2. Mobile App using Salesforce Mobile SDK

- Build a **mobile application** for store staff and managers to:
 - Manage inventory (add or update stock).
 - Process customer orders in real-time.
 - Access dashboards and order reports on the go.

3. Reports & Dashboard

- Create **dynamic dashboards** and reports in Salesforce for:
 - Real-time vehicle stock levels.
 - Sales trends and revenue analysis.
 - Monitoring loyalty program metrics.

4. AI-Powered Recommendations (Einstein AI)

- Integrate **Salesforce Einstein** to:
 - Analyze customer purchase history.
 - Provide **personalized vehicle recommendations** (e.g., suggest SUVs for customers buying family cars).

5. WhatsApp/SMS Integration

- Use Salesforce **Digital Engagement** or third-party tools (like Twilio) to:
 - Send **instant notifications** about order confirmation or pending status.
 - Update customers about loyalty status or promotional offers.