

SALESFORCE-SUPPORTED VIRTUAL INTERNSHIP PROGRAM 2025

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

NAME:K.BALADITHYA

BRANCH:B.TECH CSE

YEAR:4TH YEAR

COLLEGE NAME:GITAM UNIVERSITY, HYDERABAD

PERSONAL MAIL:baladithyakinthada@gmail.com

INTERN: SALESFORCE DEVELOPER WITH AGENTBLAZER

TRAILBLAZER:<https://www.salesforce.com/trailblazer/v5c9fsu42nhjzbplhn>

DEMO VIDEO LINK:

https://drive.google.com/file/d/19OUwkFn9fAvi0k4xeoNc3nwa1H0erFmk/view?usp=share_link

Date:17/07/2025

Project Overview:

The WhatsNext Vision Motors Salesforce CRM Project is aimed at transforming customer experience and operational efficiency through a solid CRM solution. The CRM provides automated operational solutions to streamline the ordering process for vehicles through automatic assignment of orders to the closest dealership based on customer location, stopping orders from going through if the vehicle is out of stock, and managing vehicle order status updates with reminders for test drives sent via email according to a schedule. Features of the solution include stock validation through Apex triggers, batch jobs to run stock updates in bulk, and using scheduled Apex to process vehicle orders whenever a Vehicle Order processing record is submitted. The business needs addressed were increased customer satisfaction, increased accuracy in vehicle orders, and increased operational efficiency in a competitive automotive market.

Objectives:

The main objectives for developing the WhatsNext Vision Motors CRM are to improve customer management, automate order processes, and connect dealers easily using real-time data. By automatically assigning orders to the nearest dealer, successfully managing inventory with stock validation, and improving communication through automated reminders, the aim is to create an efficient booking system with minimal manual user effort. These objectives provide real business value by increasing customer satisfaction through timely service, reducing error rates with accurate order fulfilment, and increasing revenue from a more efficient operation & improved dealer performance.

Phase 1: Requirement Analysis & Planning

Defining Business:

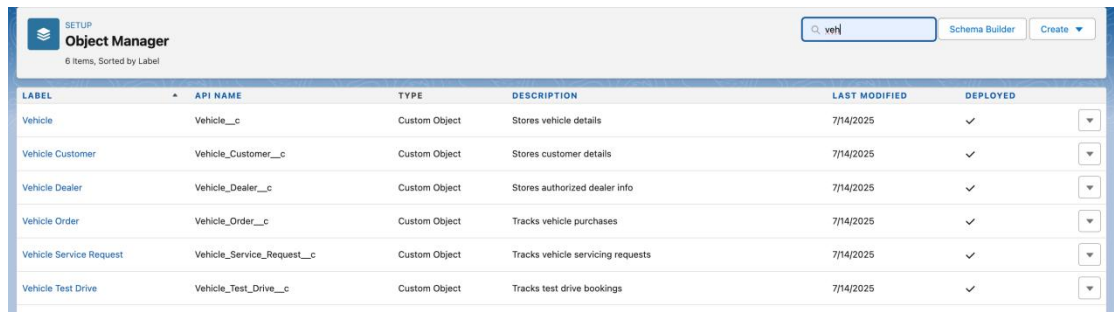
Needs What is helpful for our customers, is that WhatsNext Vision Motors CRM is resolving clear user necessities while solving important operational problems. Customers want a simple and uncomplicated order process that provides accurate stock information and dealer assignments based on travel distance - small things like not having to worry about getting a delayed or the wrong order. Dealers also need effective tools to manage inventory and customer information, as issues regularly arise when they rely on personal, tangible, and manually updated files to communicate with customers or fulfill customer orders. The CRM solves the customer and dealer challenges presented by providing automated dealer recommendations, order suggestions that don't get flagged as out of stock and sending reminders to customers when it is time to test a drive. All of these factors contribute to an improved operational process while increasing internal and external user satisfaction.

Determine Project Scope / Deliverable Objectives:

- Develop a CRM to manage for vehicle inventory, customer orders and dealer assignments.
- Implement automated notifications of order status and like an email notification to customer - Quick order status updates communicate with team faster.
- Real-time validation of stock and proximity weighting for accurate dealer selection.
- Define scalable capacity for addition future benefits and new features such as AI driven recommendations and forecasts.
- An intuitive, simple to use interface to serve both customers (B2C) and dealer customers (B2B).

Design Data Model and Security Model:

- **Data Model** - Includes custom objects such as -



The screenshot shows the Salesforce Object Manager interface. At the top, there's a search bar with 'veh' entered and buttons for 'Schema Builder' and 'Create'. Below the header, a table lists 6 items, sorted by label. The table has columns for LABEL, API NAME, TYPE, DESCRIPTION, LAST MODIFIED, and DEPLOYED. Each row represents a custom object with a dropdown arrow on the right.

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Vehicle	Vehicle__c	Custom Object	Stores vehicle details	7/14/2025	✓
Vehicle Customer	Vehicle_Customer__c	Custom Object	Stores customer details	7/14/2025	✓
Vehicle Dealer	Vehicle_Dealer__c	Custom Object	Stores authorized dealer info	7/14/2025	✓
Vehicle Order	Vehicle_Order__c	Custom Object	Tracks vehicle purchases	7/14/2025	✓
Vehicle Service Request	Vehicle_Service_Request__c	Custom Object	Tracks vehicle servicing requests	7/14/2025	✓
Vehicle Test Drive	Vehicle_Test_Drive__c	Custom Object	Tracks test drive bookings	7/14/2025	✓

- **Security Model -**

The security model implements role-based access with profiles for administrators or users, role hierarchy for dealers and customers, and sharing rules to select what each roles can see. All dealers and customers have permission sets that ensure limited access to stock levels or customer information.

Phase 2:Salesforce Development - Backend & Configurations

Setup environment:

Using the URL: <https://developer.salesforce.com/signup>, I created a developer organization in Salesforce with my first and last names, email address, role as developer, company name (college name), country (India), and a custom username (name@collegename.com, for example: baladithyakinthada436@agentforce.com of mine). Verifying email, creating a password, and going to the Salesforce setup page activates the account. used a sandbox setting for testing and development.

Customization of Objects, Fields, Validation Rules, Automation (Workflow Rules, Process Builder, Flows, Approval Process):

➤ Objects & Relationships -

- **Vehicle__c:** Stores vehicle details, related to Dealer__c and Vehicle_Order__c.
- **Vehicle_Dealer__c:** Stores authorized dealer info, related to Vehicle_Order__c.
- **Vehicle_Customer__c:** Stores customer details, related to Vehicle_Order__c and Vehicle_Test_Drive__c.
- **Vehicle_Order__c:** Tracks vehicle purchases, related to Vehicle_Customer__c and Vehicle__c.
- **Vehicle_Test_Drive__c:** Tracks test drive bookings, related to Vehicle_Customer__c and Vehicle__c.

- **Vehicle_Service_Request__c**: Tracks vehicle servicing requests, related to **Vehicle_Customer__c** and **Vehicle__c**.

➤ **Key Fields for Each Object-**

SETUP > OBJECT MANAGER					
Vehicle					
Details	Fields & Relationships 9 Items, Sorted by Field Label <input type="text" value="Quick Find"/> <input type="button" value="New"/> <input type="button" value="Deleted Fields"/> <input type="button" value="Field Dependencies"/> <input type="button" value="Set History Tracking"/>				
Fields & Relationships	FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Page Layouts	Created By	CreatedById	Lookup(User)		
Lightning Record Pages	Dealer	Dealer__c	Lookup(Vehicle Dealer)		✓
Buttons, Links, and Actions	Last Modified By	LastModifiedById	Lookup(User)		
Compact Layouts	Owner	OwnerId	Lookup(User,Group)		✓
Field Sets	Price	Price__c	Currency(18, 0)		
Object Limits	Status	Status__c	Picklist		
Record Types	Stock Quantity	Stock_Quantity__c	Number(18, 0)		
Related Lookup Filters	Vehicle Model	Vehicle_Model__c	Picklist		
Search Layouts	Vehicle Name	Name	Text(80)		✓
List View Button Layout					
Restriction Rules					
Scoping Rules					
Object Access					

SETUP > OBJECT MANAGER					
Vehicle Dealer					
Details	Fields & Relationships 8 Items, Sorted by Field Label <input type="text" value="Quick Find"/> <input type="button" value="New"/> <input type="button" value="Deleted Fields"/> <input type="button" value="Field Dependencies"/> <input type="button" value="Set History Tracking"/>				
Fields & Relationships	FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Page Layouts	Created By	CreatedById	Lookup(User)		
Lightning Record Pages	Dealer Code	Dealer_Code__c	Auto Number		
Buttons, Links, and Actions	Dealer Location	Dealer_Location__c	Text(100)		
Compact Layouts	Dealer Name	Name	Text(80)		✓
Field Sets	Email	Email__c	Email		
Object Limits	Last Modified By	LastModifiedById	Lookup(User)		
Record Types	Owner	OwnerId	Lookup(User,Group)		✓
Related Lookup Filters	Phone	Phone__c	Phone		
Search Layouts					
List View Button Layout					
Restriction Rules					
Scoping Rules					
Object Access					

SETUP > OBJECT MANAGER

Vehicle Order

Details	<div><div>Fields & Relationships</div><div>8 Items, Sorted by Field Label</div></div> <div><div>Quick Find</div><div>NewDeleted FieldsField DependenciesSet History Tracking</div></div>																																													
Fields & Relationships	<table><tr><th>FIELD LABEL</th><th>FIELD NAME</th><th>DATA TYPE</th><th>CONTROLLING FIELD</th><th>INDEXED</th></tr><tr><td>Created By</td><td>CreatedById</td><td>Lookup(User)</td><td></td><td></td></tr><tr><td>Customer</td><td>Customer__c</td><td>Lookup(Vehicle Customer)</td><td></td><td>✓</td></tr><tr><td>Last Modified By</td><td>LastModifiedById</td><td>Lookup(User)</td><td></td><td></td></tr><tr><td>Order Date</td><td>Order_Date__c</td><td>Date</td><td></td><td></td></tr><tr><td>Order Name</td><td>Name</td><td>Text(80)</td><td></td><td>✓</td></tr><tr><td>Owner</td><td>OwnerId</td><td>Lookup(User,Group)</td><td></td><td>✓</td></tr><tr><td>Status</td><td>Status__c</td><td>Picklist</td><td></td><td></td></tr><tr><td>Vehicle</td><td>Vehicle__c</td><td>Lookup(Vehicle)</td><td></td><td>✓</td></tr></table>	FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED	Created By	CreatedById	Lookup(User)			Customer	Customer__c	Lookup(Vehicle Customer)		✓	Last Modified By	LastModifiedById	Lookup(User)			Order Date	Order_Date__c	Date			Order Name	Name	Text(80)		✓	Owner	OwnerId	Lookup(User,Group)		✓	Status	Status__c	Picklist			Vehicle	Vehicle__c	Lookup(Vehicle)		✓
FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED																																										
Created By	CreatedById	Lookup(User)																																												
Customer	Customer__c	Lookup(Vehicle Customer)		✓																																										
Last Modified By	LastModifiedById	Lookup(User)																																												
Order Date	Order_Date__c	Date																																												
Order Name	Name	Text(80)		✓																																										
Owner	OwnerId	Lookup(User,Group)		✓																																										
Status	Status__c	Picklist																																												
Vehicle	Vehicle__c	Lookup(Vehicle)		✓																																										
Page Layouts																																														
Lightning Record Pages																																														
Buttons, Links, and Actions																																														
Compact Layouts																																														
Field Sets																																														
Object Limits																																														
Record Types																																														
Related Lookup Filters																																														
Search Layouts																																														
List View Button Layout																																														
Restriction Rules																																														
Scoping Rules																																														
Object Access																																														

SETUP > OBJECT MANAGER

Vehicle Test Drive

Details	<div><div>Fields & Relationships</div><div>8 Items, Sorted by Field Label</div></div> <div><div>Quick Find</div><div>NewDeleted FieldsField DependenciesSet History Tracking</div></div>																																													
Fields & Relationships	<table><tr><th>FIELD LABEL</th><th>FIELD NAME</th><th>DATA TYPE</th><th>CONTROLLING FIELD</th><th>INDEXED</th></tr><tr><td>Created By</td><td>CreatedById</td><td>Lookup(User)</td><td></td><td></td></tr><tr><td>Customer</td><td>Customer__c</td><td>Lookup(Vehicle Customer)</td><td></td><td>✓</td></tr><tr><td>Last Modified By</td><td>LastModifiedById</td><td>Lookup(User)</td><td></td><td></td></tr><tr><td>Owner</td><td>OwnerId</td><td>Lookup(User,Group)</td><td></td><td>✓</td></tr><tr><td>Status</td><td>Status__c</td><td>Picklist</td><td></td><td></td></tr><tr><td>Test Drive Date</td><td>Test_Drive_Date__c</td><td>Date</td><td></td><td></td></tr><tr><td>Test Drive Name</td><td>Name</td><td>Text(80)</td><td></td><td>✓</td></tr><tr><td>Vehicle</td><td>Vehicle__c</td><td>Lookup(Vehicle)</td><td></td><td>✓</td></tr></table>	FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED	Created By	CreatedById	Lookup(User)			Customer	Customer__c	Lookup(Vehicle Customer)		✓	Last Modified By	LastModifiedById	Lookup(User)			Owner	OwnerId	Lookup(User,Group)		✓	Status	Status__c	Picklist			Test Drive Date	Test_Drive_Date__c	Date			Test Drive Name	Name	Text(80)		✓	Vehicle	Vehicle__c	Lookup(Vehicle)		✓
FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED																																										
Created By	CreatedById	Lookup(User)																																												
Customer	Customer__c	Lookup(Vehicle Customer)		✓																																										
Last Modified By	LastModifiedById	Lookup(User)																																												
Owner	OwnerId	Lookup(User,Group)		✓																																										
Status	Status__c	Picklist																																												
Test Drive Date	Test_Drive_Date__c	Date																																												
Test Drive Name	Name	Text(80)		✓																																										
Vehicle	Vehicle__c	Lookup(Vehicle)		✓																																										
Page Layouts																																														
Lightning Record Pages																																														
Buttons, Links, and Actions																																														
Compact Layouts																																														
Field Sets																																														
Object Limits																																														
Record Types																																														
Related Lookup Filters																																														
Search Layouts																																														
List View Button Layout																																														
Restriction Rules																																														
Scoping Rules																																														
Object Access																																														

SETUP > OBJECT MANAGER

Vehicle Service Request

Details

Fields & Relationships

Page Layouts

Lightning Record Pages

Buttons, Links, and Actions

Compact Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

Search Layouts

List View Button Layout

Restriction Rules

Scoping Rules

Object Access

Fields & Relationships

9 Items, Sorted by Field Label

Quick Find

New

Deleted Fields

Field Dependencies

Set History Tracking

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Customer	Customer__c	Lookup(Vehicle Customer)		✓
Issue Description	Issue_Description__c	Text(255)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Service Date	Service_Date__c	Date		
Service Request Name	Name	Text(80)		✓
Status	Status__c	Picklist		
Vehicle	Vehicle__c	Lookup(Vehicle)		✓

➤ Validation Rules-

Vehicle

Mahendra

New Contact

Edit

New Opportunity

Related

Details

Vehicle Name

Mahendra

Vehicle Model

SUV

Stock Quantity

5

Dealer

Price

Status

Available

Created By

Kinthada Baladithya, 7/15/2025, 7:52 AM

Last Modified By

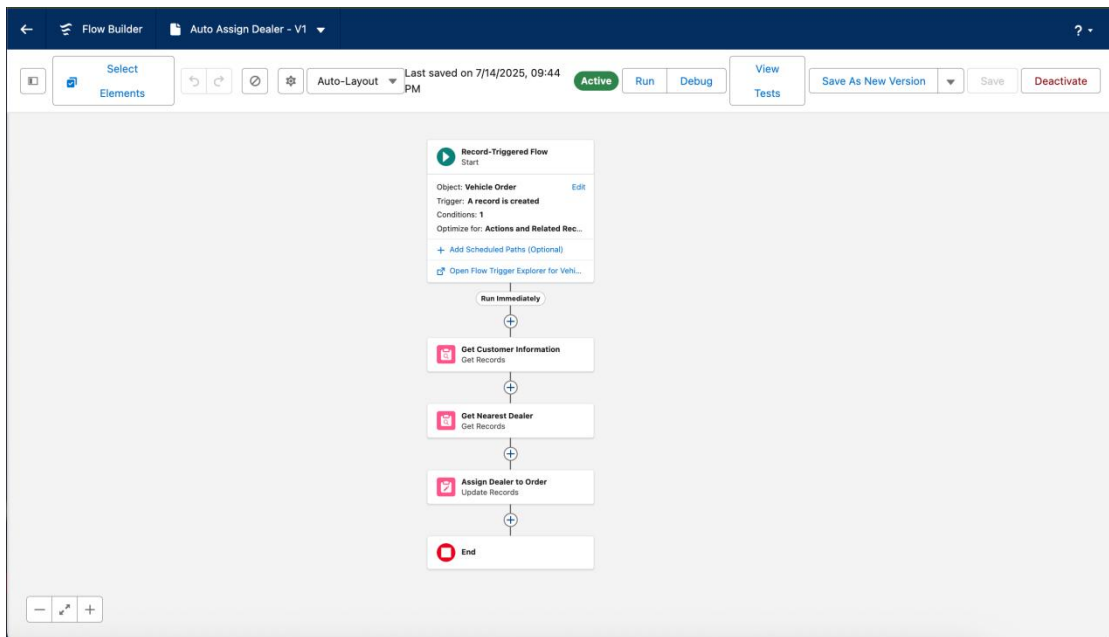
Kinthada Baladithya, 7/15/2025, 7:52 AM

Owner

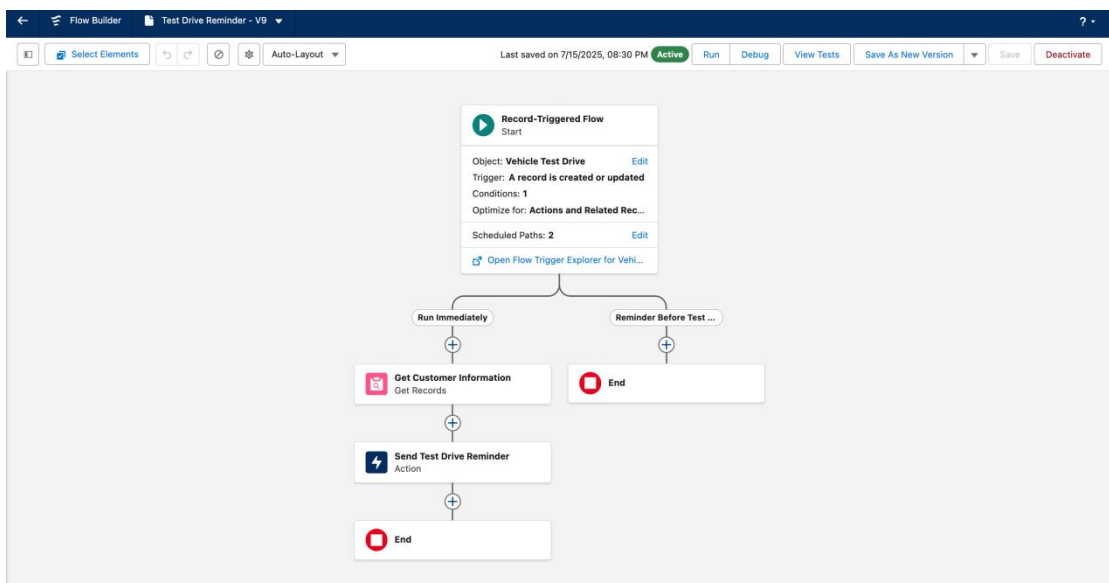
Kinthada Baladithya

➤ Automation-

1. Flow - Auto Assign Dealer:-



2. Flow - Test Drive Reminder:-



Apex Classes, Triggers, Asynchronous Apex Classes-

- **Apex Class - VehicleOrderTriggerHandler:**

```
public class VehicleOrderTriggerHandler {
    public static void handleTrigger(List<Vehicle_Order__c>
newOrders, Map<Id, Vehicle_Order__c> oldOrders, Boolean isBefore,
Boolean isAfter, Boolean isInsert, Boolean isUpdate) {
        if (isBefore) {
            if (isInsert || isUpdate) {
                preventOrderIfOutOfStock(newOrders);
            }
        }
    }
}
```



```

    }
    if (isAfter) {
        if (isInsert || isUpdate) {
            updateStockOnOrderPlacement(newOrders);
        }
    }
}
// Method to prevent orders when the vehicle is out of stock

private static void
preventOrderIfOutOfStock(List<Vehicle_Order__c> orders) {

    Set<Id> vehicleIds = new Set<Id>();

    for (Vehicle_Order__c order : orders) {

        if (order.Vehicle__c != null) {

            vehicleIds.add(order.Vehicle__c);

        }

    }

    if (!vehicleIds.isEmpty()) {

        Map<Id, Vehicle__c> vehicleStockMap = new Map<Id,
Vehicle__c>();

        for (Vehicle__c vehicle : [SELECT Id, Stock_Quantity__c
FROM Vehicle__c WHERE Id IN :vehicleIds]) {

            vehicleStockMap.put(vehicle.Id, vehicle);

        }

        for (Vehicle_Order__c order : orders) {

            if (vehicleStockMap.containsKey(order.Vehicle__c)) {

                Vehicle__c vehicle =
vehicleStockMap.get(order.Vehicle__c);

                if (vehicle.Stock_Quantity__c <= 0) {

                    order.addError('This vehicle is out of stock. Order
cannot be placed.');
```

```

    }

    }

    }

}

// Method to update vehicle stock when an order is placed

private static void
updateStockOnOrderPlacement(List<Vehicle_Order__c> orders) {

    Set<Id> vehicleIds = new Set<Id>();

    for (Vehicle_Order__c order : orders) {

        if (order.Vehicle__c != null && order.Status__c ==
'Confirmed') {

            vehicleIds.add(order.Vehicle__c);

        }

    }

    if (!vehicleIds.isEmpty()) {

        Map<Id, Vehicle__c> vehicleStockMap = new Map<Id,
Vehicle__c>();

        for (Vehicle__c vehicle : [SELECT Id, Stock_Quantity__c
FROM Vehicle__c WHERE Id IN :vehicleIds]) {

            vehicleStockMap.put(vehicle.Id, vehicle);

        }

        List<Vehicle__c> vehiclesToUpdate = new
List<Vehicle__c>();

        for (Vehicle_Order__c order : orders) {

            if (vehicleStockMap.containsKey(order.Vehicle__c)) {

                Vehicle__c vehicle =
vehicleStockMap.get(order.Vehicle__c);

                if (vehicle.Stock_Quantity__c > 0) {

```

```

        vehicle.Stock_Quantity__c -= 1;

        vehiclesToUpdate.add(vehicle);

    }

}

}

if (!vehiclesToUpdate.isEmpty()) {
    update vehiclesToUpdate;
}
}
}
}
}

```

- **Trigger - VehicleOrderTrigger:**

```

trigger VehicleOrderTrigger on Vehicle_Order__c (before
insert,before update, after insert, after update) {

    VehicleOrderTriggerHandler.handleTrigger(trigger.new,
    trigger.oldMap, trigger.isBefore,trigger.isAfter, trigger.isInsert,
    trigger.isUpdate);

}

```

- **Asynchronous Apex - VehicleOrderBatch:**

```

global class VehicleOrderBatch implements
Database.Batchable<SObject> {

    global Database.QueryLocator start(Database.BatchableContext bc)
    {
        return Database.getQueryLocator([
            SELECT Id, Status__c, Vehicle__c
            FROM Vehicle_Order__c
            WHERE Status__c = 'Pending'
        ]);
    }

    global void execute(Database.BatchableContext bc,
    List<Vehicle_Order__c> orderList) {
        Set<Id> vehicleIds = new Set<Id>();

        for (Vehicle_Order__c order : orderList) {
            if (order.Vehicle__c != null) {
                vehicleIds.add(order.Vehicle__c);
            }
        }
    }
}

```

```

    }

    if (!vehicleIds.isEmpty()) {
        Map<Id, Vehicle__c> vehicleStockMap = new Map<Id,
Vehicle__c>(
            [SELECT Id, Stock_Quantity__c FROM Vehicle__c
WHERE Id IN :vehicleIds]
        );

        List<Vehicle_Order__c> ordersToUpdate = new
List<Vehicle_Order__c>();
        List<Vehicle__c> vehiclesToUpdate = new
List<Vehicle__c>();

        for (Vehicle_Order__c order : orderList) {
            if (vehicleStockMap.containsKey(order.Vehicle__c)) {
                Vehicle__c vehicle =
vehicleStockMap.get(order.Vehicle__c);

                if (vehicle.Stock_Quantity__c > 0) {
                    order.Status__c = 'Confirmed';
                    vehicle.Stock_Quantity__c -= 1;

                    ordersToUpdate.add(order);
                    vehiclesToUpdate.add(vehicle);
                }
            }
        }

        if (!ordersToUpdate.isEmpty()) {
            update ordersToUpdate;
        }

        if (!vehiclesToUpdate.isEmpty()) {
            update vehiclesToUpdate;
        }
    }

    global void finish(Database.BatchableContext bc) {
        System.debug('    Vehicle order batch job completed
successfully.');
```

- **Schedule Class - VehicleOrderBatchScheduler:**

```

global class VehicleOrderBatchScheduler implements Schedulable {

    global void execute(SchedulableContext sc) {
```

```

VehicleOrderBatch batchJob = new VehicleOrderBatch();

Database.executeBatch(batchJob, 50); // 50 is the batch size

    }

}

```

- **Scheduled Job:**

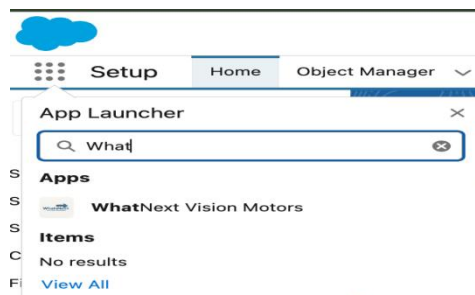
```
String cronExp = '0 0 12 * * ?'; // Runs daily at 12:00 PM
```

```
System.schedule('Daily Vehicle Order Processing', cronExp, new
VehicleOrderBatchScheduler());
```

Phase 3: UI/UX Development & Customization

Lightning App Setup through App Manager

Created a Lightning App named "WhatNext Vision Motors" via the App Manager in Setup, with a meaningful description. Added navigation items including Vehicle, Dealer, Customer, Order, Test Drive, Service Request, Reports, and Dashboard. Assigned the app to the System Administrator profile for access.



Page Layouts, Dynamic Forms

Designed page layouts for Vehicle__c and Vehicle_Order__c objects, incorporating Dynamic Forms to enhance field visibility and user interaction based on record type and user role.



WhatNext Vision M...

Vehicles ▾



Vehicles

Recently Viewed ▾

To unpin, pin another



1 item • Updated a few seconds ago



Vehicle Name

1



Mahendra



WhatNext Vision M...

Vehicles ▾

Vehicle Dealers ▾

Vehicle Customers ▾

Vehicle Orders ▾

Vehicle Test Drives ▾

Vehicle Servi

Search...



Vehicle Order
Vision

Related

Details

Order Name

Vision



Owner

Kinthada Baladithya



Customer

[Bal Reddy](#)



Vehicle

[Mahendra](#)



Order Date

7/19/2025



Status

Confirmed



Created By

Kinthada Baladithya, 7/17/2025, 2:30 AM


Last Modified By

Kinthada Baladithya, 7/17/2025, 2:30 AM

Phase 4: Data Migration, Testing, Security

Sample Records Created

Vehicle: Mahendra SUV Model

 Vehicle
Mahendra

Related

Details

Vehicle Name

Mahendra

Vehicle Model

SUV

Stock Quantity

4

Dealer


Price

\$1,000,000


Status

Available


Created By

 [Kinthada Baladithya](#), 7/15/2025, 7:52 AM

Owner

 [Kinthada Baladithya](#)

Last Modified By

 [Kinthada Baladithya](#), 7/17/2025, 9:14 AM


Testing

Done by flows of test drive remainder.

All Scheduled Jobs

Help for this Page

The All Scheduled Jobs page lists all of the jobs scheduled by your users. Multiple job types may display on this page. You can delete scheduled jobs if you have the permission to do so.

 **Percentage of Scheduled Jobs Used: 1%**
You have currently used 1 scheduled Apex jobs out of an allowed organization limit of 100 active or scheduled jobs. To learn about how this limit is calculated and what contributes to it see the [Lightning Platform Apex Limits](#) topic.

View: All Scheduled Jobs

Create New View

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z Other All

Schedule Apex

Action	Job Name	Submitted By	Submitted	Started	Next Scheduled Run	Type	Cron Trigger ID
Manage Del Pause Job	Daily Vehicle Order Processing	Baladithya, Kinthada	7/15/2025, 8:47 AM	7/16/2025, 12:01 PM	7/17/2025, 12:00 PM	Scheduled Apex	08egL000007jxMV
Del	Metalytics Data Loader Job for Org : 00DgL000006bXr	User, Integration	7/3/2025, 3:13 PM	7/16/2025, 1:54 PM	7/17/2025, 1:54 PM	Autonomous Data Loader Job	08egL000006dIZY
	Program Milestone Computation Cron Job	Process, Automated	7/3/2025, 3:13 PM	7/17/2025, 7:00 AM	7/17/2025, 11:59 AM	Program Milestone Computation Cron Job	08egL000006dIZW
	Program Status Update Cron Job	Process, Automated	7/3/2025, 3:13 PM	7/17/2025, 5:01 AM	7/17/2025, 8:00 PM	Program Status Update Cron Job	08egL000006dIZX

Phase 5 – Deployment Maintenance

Deployment

After completion of flow test drive remainder, the customer got the mail regarding test drive.

Why is this message in spam? This message is similar to messages that were identified as spam in the past.

Report not spam

Hi Bal Reddy,

This is a reminder that your test drive is scheduled for tomorrow.

Please contact us if you need to reschedule.

Thanks,

WhatNext Vision Motors

Maintenance

Use Apex logs for monitoring - Updates to flows or triggers via Setup

File | Edit | Debug | Test | Workspace | Help

VehicleOrderTriggerHandler.apex | VehicleOrderTrigger.apex | VehicleOrderBatch.apex | VehicleOrderBatchScheduler.apex | Log executeAnonymous @7/15/2025, 9:17:09 PM

Execution Log

Timestamp	Event	Details
21:17:09:002	USER_INFO	
21:17:09:002	EXECUTION_ST...	[EXTERNAL]0005Up000004p6E7baledthyskinthead436@agentforce.com([GMT-07:00] Pacific Daylight Time (America/Los_Angeles))[GMT-07:00]
21:17:09:002	CORE_INIT_ST...	[EXTERNAL]executeAnonymous.apex
21:17:09:002	VARIABLE_SCO...	[1]cronExpString:false
21:17:09:002	HEAP_ALLOCATE	[93]Bytes:3
21:17:09:002	HEAP_ALLOCATE	[98]Bytes:152
21:17:09:002	HEAP_ALLOCATE	[415]Bytes:408
21:17:09:002	HEAP_ALLOCATE	[406]Bytes:408
21:17:09:002	HEAP_ALLOCATE	[536]Bytes:48
21:17:09:002	HEAP_ALLOCATE	[247]Bytes:6
21:17:09:002	HEAP_ALLOCATE	[EXTERNAL]Bytes:3
21:17:09:002	STATEMENT_EX...	[1]
21:17:09:002	STATEMENT_EX...	[1]
21:17:09:002	HEAP_ALLOCATE	[1]Bytes:12
21:17:09:002	VARIABLE_ASS...	[1]cronExp"0 0 12 * * ?"
21:17:09:002	STATEMENT_EX...	[1]
21:17:09:002	HEAP_ALLOCATE	[3]Bytes:30
21:17:09:004	HEAP_ALLOCATE	[2]Bytes:1
21:17:09:004	METHOD_ENTRY	[1]01 pugl.000003Vp2vzVVehicleOrderBatchScheduler.VehicleOrderBatchScheduler()
21:17:09:004	STATEMENT_EX...	[1]
21:17:09:004	STATEMENT_EX...	[1]
21:17:09:004	METHOD_EXIT	[1]VehicleOrderBatchScheduler
21:17:09:004	HEAP_ALLOCATE	[2]Bytes:1
21:17:09:004	HEAP_ALLOCATE	[66]Bytes:5
21:17:09:004	HEAP_ALLOCATE	[74]Bytes:5
21:17:09:004	HEAP_ALLOCATE	[62]Bytes:7
21:17:09:004	SYSTEM_MODE...	false
21:17:09:004	HEAP_ALLOCATE	[3]Bytes:5
21:17:09:004	CONSTRUCTOR...	[1]01 pugl.000003Vp2vzVVehicleOrderBatchScheduler.VehicleOrderBatchScheduler()

This Frame

Events

Checkpoints

Query Editor

Filter

Click here to filter the log

Logs

Stack

Checkpoints

Query Editor

Filter

Click here to filter the log

View State

Progress

Problems

Problems

Name	Line	Problem
------	------	---------

Conclusion:

The WhatsNext Vision Motors CRM automates vehicle order processing and improves customer engagement with Salesforce tools. It removes manual stock checks and provides real-time updates using triggers and batch jobs. This creates a seamless experience for both customers and dealers. This solution meets the business needs of improving order accuracy, increasing customer satisfaction, and boosting operational efficiency. It positions WhatsNext Vision Motors as a leader in automotive CRM innovation.

Benefits:

- **Saves Time and Prevents Stockouts:** Automating stock validation and dealer assignments reduces manual effort and ensures availability.
- **Ensures Accuracy with Test Validation:** Detailed test cases and high code coverage guarantee reliable performance.
- **Role-Based Access Improves Security:** Customized profiles and sharing rules protect sensitive data while keeping it accessible.

Future Scope:

- **Add Slack Notifications:** Integrate real-time alerts for order and stock updates to improve team communication.
- **Develop LWC Dashboard for Live Stock Tracking:** Create a dynamic interface for real-time inventory monitoring.
- **Implement Multi-Level Approval for Orders:** Add a layered approval process to improve order management control.

THANK YOU