Capstone Project Report  
WhatNext Vision Motors

# Abstract

The automotive sector is undergoing a digital revolution where CRM platforms like Salesforce play a key role in driving operational efficiency and customer satisfaction. The "WhatNext Vision Motors" project is a capstone initiative aimed at leveraging Salesforce capabilities to build a fully functional automobile dealership management system. The platform integrates vehicle inventory, order processing, stock management, reporting, and user-specific dashboards, all tailored to streamline operations and improve the customer experience. This document outlines the complete development lifecycle, technical implementation, business objectives, and the achieved results.

# Introduction

WhatNext Vision Motors is a conceptual automobile dealership firm envisioned to digitally manage its core processes. The business operations include managing vehicle inventory, customer and dealer orders, vehicle allocation, and tracking delivery statuses. Traditional systems often rely on fragmented data and manual entries, leading to inefficiencies and customer dissatisfaction.  
  
This capstone project introduces a Salesforce-based CRM solution that addresses these issues. The platform ensures centralized data management, automated workflows, accurate stock control, and enhanced reporting functionalities. The project not only showcases technical skills but also simulates real-world business scenarios for practical learning.

# Objectives

- To design a centralized CRM solution for automobile dealership management.  
- To automate and validate vehicle orders based on stock availability using Apex triggers and flows.  
- To provide role-based access to customers, dealers, and admins.  
- To build Lightning Pages and custom tabs for an intuitive user interface.  
- To generate real-time reports and dashboards for key performance indicators.  
- To ensure data consistency and prevent stock duplication or overbooking.

# Problem Statement

Automobile dealerships often face challenges such as lack of synchronization between sales and stock, manual errors in order processing, and limited visibility for stakeholders. There is a pressing need for a dynamic and automated solution that enables real-time updates, improves customer engagement, and streamlines the order lifecycle. The WhatNext Vision Motors project addresses these issues by designing an end-to-end solution using Salesforce's declarative and programmatic tools.

# System Design

The system is designed with the following Salesforce components:  
- Custom Objects: Vehicle\_\_c, Vehicle\_Order\_\_c, Dealer\_\_c, Customer\_\_c.  
- Relationships: Lookup and master-detail relationships for linking orders with vehicles and dealers.  
- Validation Rules: To restrict invalid order entries and ensure business logic enforcement.  
- Apex Triggers: For automatic stock verification and allocation before order confirmation.  
- Flows: Used to automate notifications and record updates.  
- Lightning App Builder: To design custom user interfaces and role-specific pages.  
- Reports & Dashboards: To visualize data such as top-selling models, pending orders, and stock levels.

# Modules Overview

1. \*\*Vehicle Inventory Management\*\* – Admins can add, edit, and view available vehicles with details such as model, price, and quantity.  
2. \*\*Order Management\*\* – Customers and dealers place vehicle orders, which are validated in real-time.  
3. \*\*Stock Verification\*\* – Apex trigger ensures orders are only placed for vehicles in stock.  
4. \*\*Flow Automation\*\* – Flows send alerts and update order status.  
5. \*\*Custom UI\*\* – Role-specific Lightning pages for customers, dealers, and admins.  
6. \*\*Dashboards\*\* – Graphical representation of KPIs including monthly sales, pending deliveries, and stock trends.

# Implementation Details

The project is implemented in a Salesforce Developer Edition Org. Key technical features include:  
- Custom Metadata and Settings for flexible configurations.  
- Apex classes and trigger handlers for scalable logic.  
- Record-triggered flows for background automation.  
- Use of SOQL queries to fetch real-time data in triggers.  
- Security settings such as profiles and permission sets.  
- Use of schema builder for object relationships and ERD.

# Results and Benefits

The WhatNext Vision Motors application has led to significant improvements:  
- Streamlined order lifecycle from request to delivery.  
- Real-time vehicle availability check reduces delays.  
- Visual dashboards offer management insights into stock and performance.  
- Reduced manual intervention with automated validations and flows.  
- Improved data quality and customer satisfaction through error-free processing.

# Conclusion

This project demonstrates how Salesforce can be effectively used to develop complex, real-world CRM solutions. WhatNext Vision Motors acts as a blueprint for other automotive businesses aiming to adopt digital platforms. The modular design ensures scalability, and the use of both declarative and programmatic tools ensures maximum flexibility and maintainability.

# Future Enhancements

- Integration with payment gateways for online transactions.  
- Use of Einstein AI for personalized vehicle recommendations.  
- SMS and email alerts for order status.  
- Dealer rating and review system.  
- Mobile-first responsive design for field access.

# Source Code

The following source code snippets are core components of the WhatNext Vision Motors Salesforce implementation. These include Apex classes, triggers, and flows used for validation, automation, and order processing.

Apex Trigger: VehicleOrderTrigger

trigger VehicleOrderTrigger on Vehicle\_Order\_\_c (before insert, before update) {  
 if (Trigger.isBefore) {  
 if (Trigger.isInsert || Trigger.isUpdate) {  
 VehicleOrderTriggerHandler.preventOrderIfOutOfStock(Trigger.new);  
 }  
 }  
}

Apex Class: VehicleOrderTriggerHandler

public class VehicleOrderTriggerHandler {  
 public static void preventOrderIfOutOfStock(List<Vehicle\_Order\_\_c> orders) {  
 Set<Id> vehicleIds = new Set<Id>();  
 for (Vehicle\_Order\_\_c order : orders) {  
 vehicleIds.add(order.Vehicle\_\_c);  
 }  
  
 Map<Id, Vehicle\_\_c> vehicleMap = new Map<Id, Vehicle\_\_c>(  
 [SELECT Id, Stock\_\_c FROM Vehicle\_\_c WHERE Id IN :vehicleIds]  
 );  
  
 for (Vehicle\_Order\_\_c order : orders) {  
 Vehicle\_\_c vehicle = vehicleMap.get(order.Vehicle\_\_c);  
 if (vehicle != null && vehicle.Stock\_\_c <= 0) {  
 order.addError('This vehicle is out of stock.');  
 }  
 }  
 }  
}

# Output Screens

Below are sample output screens from the Salesforce application including dashboards, Lightning pages, and record views for Vehicle and Vehicle Orders.

[Screenshot: Vehicle Inventory Lightning Page]

[Screenshot: Dealer Vehicle Order Form]

[Screenshot: Order Validation Error Message]

[Screenshot: Admin Dashboard Overview]

[Screenshot: Vehicle Order Report View]