**Requirement Analysis**

Project Name: Garage Management System

Team ID: LTVIP2025TMID28970

Date: 19-06-2025

**1. Customer Journey Map**

This map meticulously illustrates the step-by-step process a vehicle owner undertakes, from their initial vehicle issue or service need to the final notification post-payment for services rendered. It critically highlights the points of interaction with the proposed CRM system, detailing how each system interaction not only supports the customer's action but also streamlines the garage's operations and enhances data capture for future engagement.

|  |  |  |
| --- | --- | --- |
| **Step** | **Customer Action (Detailed Scenario & Customer Intent)** | **System Interaction (Detailed CRM Role & Data Capture)** |
| **1** | **Vehicle Owner Contacts Garage or Arrives for Service**: A potential customer (vehicle owner) experiences a vehicle issue (e.g., check engine light, strange noise) or has a scheduled maintenance need. They initiate contact online (via website form, chat, or online booking portal), by phone call, or simply drive into the garage. Their primary intent is to get their vehicle diagnosed, repaired, or serviced. | **New Customer & Vehicle Record Creation/Retrieval (Customer\_\_c & Vehicle\_\_c Objects)** Upon first interaction, the service advisor or system automatically captures essential customer details (Name, Contact Information, Preferred Communication Method) into a new **Customer\_\_c** record. If it's a returning customer, their existing record is quickly retrieved, populating their history and preferences. Concurrently, a new **Vehicle\_\_c** record is created or retrieved, capturing vital vehicle details (Make, Model, Year, VIN, License Plate, Mileage). This ensures a personalized service from the outset and begins building a comprehensive customer and vehicle profile. |
| **2** | **Vehicle Diagnosis & Service Request Confirmation**:  The vehicle owner describes the symptoms or service needed to the service advisor. They may approve initial diagnostic procedures. The customer's intent is to accurately convey their concerns and understand the potential issues and recommended services. | **Service Request Record Created & Linked (ServiceRequest\_\_c Object)**  As the customer explains the issue, the service advisor logs these details into a new **ServiceRequest\_\_c** record. This record is directly linked to the **Customer\_\_c** and **Vehicle\_\_c** records. It captures reported symptoms, diagnostic findings, and recommended services (e.g., oil change, brake replacement, engine diagnostic). This may involve checking historical service records from the **Vehicle\_\_c** object for recurring issues or past maintenance, ensuring a comprehensive understanding of the vehicle's history. |
| **3** | **Receives Service Estimate & Approves Work**  The customer receives a detailed estimate outlining the scope of work, parts required, labor costs, and estimated completion time. They might ask for clarifications or prioritize certain repairs. Their intent is to understand the full cost and duration before committing to the service. | **Service Estimate Generation (ServiceEstimate\_\_c Object)**  A new **ServiceEstimate\_\_c** record is automatically generated within the CRM, drawing data from the **ServiceRequest\_\_c** record and **PartsInventory\_\_c** (for parts pricing) and **LaborRates\_\_c** (for labor calculations). This record details line items for parts, labor hours, applicable taxes, and any discounts. It is intelligently linked via Lookup relationships to the **Customer\_\_c**, **Vehicle\_\_c**, and **ServiceRequest\_\_c** records. The system records the customer's approval status (Approved/Declined) and any specific notes or partial approvals. |
| **4** | **Vehicle Service/Repair Commences**  The vehicle is moved into the service bay, and the mechanic begins the approved work. The customer might receive updates on progress if agreed upon. Their goal is to have their vehicle professionally repaired or maintained. | **Work Order Creation & Progress Tracking (WorkOrder\_\_c & Task\_\_c Objects)**  Upon service approval, a new **WorkOrder\_\_c** record is generated, linking to the **ServiceEstimate\_\_c**. This triggers the assignment of tasks (e.g., "Brake Pad Replacement," "Oil Filter Change") to specific mechanics or bays via **Task\_\_c** records. The system facilitates real-time updates on task status (e.g., "In Progress," "Pending Parts," "Completed") by mechanics using a mobile interface, which updates the **WorkOrder\_\_c** status. This provides managers with real-time operational visibility and allows for accurate time tracking for labor billing. |
| **5** | **Receives Completion Notification & Prepares for Pickup**  The customer is notified that their vehicle service is complete and it's ready for pickup. They might arrange their schedule to collect the vehicle and prepare for payment. Their intent is to retrieve their repaired vehicle efficiently. | **Service Completion Status Update & Notification (WorkOrder\_\_c & Flow)**  When the final task on the **WorkOrder\_\_c** is marked "Completed," an Apex Trigger or Record-Triggered Flow automatically updates the **WorkOrder\_\_c** status to "Ready for Pickup." This automation then triggers an outbound email or SMS notification (using the customer's preferred communication method from their **Customer\_\_c** record) informing them that their vehicle is ready. The notification includes final service details, outstanding balance, and garage operating hours, enhancing transparency and customer convenience. |
| **6** | **Makes Payment & Picks Up Vehicle**  The customer arrives at the garage, reviews the final invoice (which might include minor adjustments if scope changed), and tenders payment using their preferred method (cash, card, digital payment). Their intent is to finalize the transaction and drive away with their repaired vehicle. | **Payment Record & Invoice Finalization (Payment\_\_c & Invoice\_\_c Objects)**  Upon receipt of payment, a new **Payment\_\_c** record is created, linking to the **WorkOrder\_\_c** and **Customer\_\_c**. This **Payment\_\_c** record captures the amount paid, payment method, and date. An automated Apex Trigger or Flow on the **Payment\_\_c** object updates the **Outstanding\_Balance\_\_c** field on the **WorkOrder\_\_c** to zero for full payments. A final **Invoice\_\_c** record is generated from the **WorkOrder\_\_c** and **Payment\_\_c** data, automatically calculating final totals, taxes, and applied payments, ensuring a comprehensive financial record and audit trail. |
| **7** | **Receives Digital Invoice/Receipt & Follow-up**  After successful payment, the customer expects a digital invoice or receipt for their records, detailing the services, parts, and total cost. They might also expect a follow-up about their experience. This fulfills their need for documentation and provides an opportunity for feedback. | **Automated Invoice Email & Follow-up Task/Survey (Flow & Task/Survey Objects)**  A robust Record-Triggered Flow executes automatically upon finalization of the **Invoice\_\_c** record and confirmation of full payment. This Flow dynamically retrieves the customer's email address from their **Customer\_\_c** record and dispatches a personalized digital invoice using an Email Template. Concurrently, the Flow creates a follow-up **Task\_\_c** for the service advisor (e.g., "Call for satisfaction survey in 3 days") or triggers an automated customer satisfaction survey (linking to a **Survey\_\_c** object). This proactive communication and feedback loop significantly enhances customer satisfaction and provides valuable data for continuous service improvement. |

**2. Data Flow Diagram (DFD)**

**Level 0 Description :**

The Level 0 DFD is the highest-level overview of the Garage Management System. It's also known as the Context Diagram.

* **Single Process:** It shows the entire system as a single process (often represented as a single circle or rounded rectangle). In this case, it's the entire "Garage Management System."
* **External Entities:** It identifies the external entities that interact with the system. In this diagram, these are primarily the [Vehicle Owner], [Mechanic], and potentially [Admin/Service Advisor].
* **Data Flows:** It shows the major data flows between the external entities and the single system process. It doesn't show internal processes or data stores within the system itself at this level, only the primary inputs and outputs from and to the external world.

**Level 0 – Context Diagram**

**Entities:**

* **Vehicle Owner:** The customer seeking vehicle services.
* **Mechanic:** The individual performing diagnostics and repairs.
* **Admin/Service Advisor:** The staff managing appointments, estimates, billing, and customer interactions.

**Data Flows:**

* **Vehicle Owner** sends Service Request, Vehicle Info, Payment.
* **Admin/Service Advisor** inputs Service Details, Estimate Approval, Payment Confirmation.
* **Mechanic** inputs Diagnostic Results, Work Progress, Parts Used.
* **Garage Management System** provides Service Estimate, Work Order, Completion Notification, Invoice, Digital Receipt.

**Level 1 Description:**

The Level 1 DFD breaks down the single "Garage Management System" process into its major sub-processes and associated data flows and data stores, providing a more detailed view of internal system operations.

* **Vehicle & Customer Registration:** This process captures new or retrieves existing customer and vehicle details, linking them appropriately.
* **Service Request Management:** This process captures reported symptoms, diagnostic findings, and recommended services, creating a structured service request.
* **Estimate Generation:** This process calculates the cost of service, parts, and labor, generating a detailed estimate for customer approval.
* **Work Order & Progress Tracking:** This process creates work orders based on approved estimates, assigns tasks to mechanics, and tracks the real-time progress of repairs, including parts usage.
* **Invoice & Payment Processing:** This process finalizes the service invoice, records payment transactions, and updates the outstanding balance.
* **Notification & Follow-up:** This process automates sending service completion notifications, digital invoices/receipts, and triggers follow-up tasks or surveys.

**Level 1 – Detailed DFD**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Step** | **Process** | **Input** | **Output** | **Data Store** | | **1** | **Manage Customer & Vehicle** | Customer Info, Vehicle Info | Customer\_\_c record, Vehicle\_\_c record | Customer\_\_c, Vehicle\_\_c | | **2** | **Create Service Request** | Symptoms, Diagnoses, Service Needs | ServiceRequest\_\_c record | ServiceRequest\_\_c | | **3** | **Generate Service Estimate** | Service Request, Parts Pricing, Labor Rates | ServiceEstimate\_\_c record, Estimate Details | ServiceEstimate\_\_c, PartsInventory\_\_c, LaborRates\_\_c | | **4** | **Process Work Order & Track Progress** | Approved Estimate, Mechanic Updates, Parts Used | WorkOrder\_\_c record, Task Status | WorkOrder\_\_c, Task\_\_c, PartsInventory\_\_c | | **5** | **Process Payment & Invoice** | Payment Amount, Payment Method, Work Order | Payment\_\_c record, Final Invoice | Payment\_\_c, Invoice\_\_c | | **6** | **Send Notifications & Follow-ups** | Invoice Record, Customer Email ID, Service Completion Status | Completion Notification, Digital Invoice, Follow-up Task/Survey | Sent Log, Customer\_\_c, Task\_\_c, Survey\_\_c | |

**3. Solution Requirements**

**Functional Requirements:**

* **Ability to create customer records (Vehicle Owner):** The system must allow for the creation, storage, and comprehensive management of vehicle owner information.
* **Ability to create vehicle records (Vehicle\_\_c):** The system must support the creation, detailed categorization, and management of all vehicles (make, model, VIN, mileage, service history).
* **Create and track Service Requests & Work Orders:** The system must provide functionality to generate, track the status of, and manage all service requests from initial inquiry to completed work orders.
* **Auto-update Service Status & Notification (Trigger/Flow):** A system trigger or flow must automatically update the WorkOrder\_Status\_\_c field and trigger notifications upon key event updates (e.g., "In Progress," "Ready for Pickup").
* **Send service completion/invoice emails (Flow):** An automated Flow must be in place to send service completion notifications and digital invoices/receipts to customers upon successful service finalization and payment.
* **Generate reports and dashboards:** The system must enable users to create, customize, and view various reports and dashboards for insightful analysis of service trends, parts inventory, labor utilization, and financial performance.
* **Validate Payment Amount <= Total Invoice Amount (Validation Rule):** A validation rule must be implemented to prevent data entry errors by ensuring that the Paid\_Amount\_\_c never exceeds the Total\_Invoice\_Amount\_\_c on an Invoice\_\_c record.

Following are the functional requirements of the proposed solution.

|  |  |  |
| --- | --- | --- |
| **FR No.** | **Functional Requirement (Epic)** | **Sub Requirement (Story / Sub-Task)** |
| FR-1 | User & Vehicle Registration | - Registration/Login through Form ­- Vehicle Onboarding (VIN lookup) - Customer Account Management |
| FR-2 | Appointment & Service Request Management | - Schedule/Reschedule Appointments - Log Customer Reported Symptoms - Create & Track Service Request records |
| FR-3 | Vehicle & Part Inventory Management | - Create Vehicle\_\_c records (Make, Model, VIN) - Manage PartsInventory\_\_c (Stock, Price, Supplier) - Track mileage and service history |
| FR-4 | Work Order & Task Management | - Generate Work Order records from Service Requests - Assign tasks to Mechanics - Track task status (In Progress, Completed) |
| FR-5 | Estimate & Invoice Automation | - Generate ServiceEstimate\_\_c from Work Orders - Auto-calculate parts, labor, taxes - Generate Invoice\_\_c upon work completion |
| FR-6 | Payment & Balance Update | - Auto-update Paid\_Amount\_\_c via Trigger/Flow - Validate Paid\_Amount\_\_c ≤ Total\_Invoice\_Amount\_\_c - Manage Outstanding\_Balance\_\_c for partial payments |
| FR-7 | Communication & Notifications | - Send service completion notifications (Email/SMS) - Send digital invoices/receipts - Use Flow with customer contact lookup |
| FR-8 | Role-Based Access Control | - Define profiles for Service Advisor, Mechanic, Admin - Set permissions using Permission Sets for features and data |
| FR-9 | Reports & Dashboards | - Generate reports for Service History, Revenue, Parts Usage - Create dashboards for operational performance & customer trends |
| FR-10 | Data Validation & Integrity | - Implement validation rules for accurate data entry - Use Lookup relationships to connect Customer, Vehicle, Service Request, Work Order, Invoice |

**Non-Functional Requirements:**

* **System responsiveness and low latency:** The system should operate quickly and efficiently, with minimal delays in response times for all user interactions, particularly during vehicle check-in and service status updates.
* **Ensure user access control (Profile & Permission Sets):** Robust security measures, including Profiles and Permission Sets, must be implemented to control user access to sensitive vehicle, service, and financial data based on their roles (e.g., Mechanics only see relevant work orders).
* **Secure communication:** All automated email/SMS communications sent from the system (e.g., service reminders, completion notifications) must adhere to security best practices to protect customer and vehicle data.
* **Maintain data integrity across related objects:** The system must ensure consistency and accuracy of data across all linked objects (e.g., Customer, Vehicle, Service Request, Work Order, Invoice, Parts) through appropriate relationships and validation rules.
* **Maintain audit trail with field history tracking:** The system should track changes to key fields on important objects (e.g., WorkOrder\_\_c, Invoice\_\_c, Vehicle\_\_c) to provide an audit trail for accountability and historical analysis of services and changes.

Following are the non-functional requirements of the proposed solution.

|  |  |  |
| --- | --- | --- |
| **NFR No.** | **Non-Functional Requirement** | **Description** |
| NFR-1 | Usability | The Garage Management System must be intuitive and user-friendly for service advisors, mechanics, and administrators, minimizing training time. |
| NFR-2 | Security | Enforce robust access control using profiles and permission sets to protect sensitive customer, vehicle, and financial data. |
| NFR-3 | Reliability | Ensure the system works consistently without errors during all standard garage operations, including busy periods. |
| NFR-4 | Performance | The system should load vehicle histories, update work order statuses, and generate estimates/invoices with minimal latency. |
| NFR-5 | Availability | The system should be accessible during all garage operating hours with minimal downtime to support continuous operations. |
| NFR-6 | Scalability | Capable of supporting increased vehicle throughput, customer base, parts inventory, and transaction volume as the garage expands. |
| NFR-7 | Maintainability | Easy to update and enhance automation components like Flows and Triggers as service processes and business logic evolve. |
| NFR-8 | Auditability | Field history tracking must be enabled on key objects like Work Orders and Invoices to provide change logs for transparency and accountability. |

**4. Technology Stack**

The Garage Management System project was developed on the Salesforce platform, leveraging its robust low-code and pro-code capabilities. The system incorporates custom data modeling, extensive automation, precise validation, comprehensive reporting, and granular role-based access control, making it a complete CRM tailored for efficient garage operations.

|  |  |  |
| --- | --- | --- |
| **Category** | **Tools/Technologies Used** | **Explanation** |
| **Platform** | **Salesforce Lightning Experience** | The entire application is built on Salesforce Lightning Experience, providing a modern, intuitive UI and a component-based architecture. This enhances the user experience for service advisors, mechanics, and administrators, improving interaction speed and clarity across all modules. |
| **Automation** | **Record-Triggered Flows**, **Workflow Rules** | - **Flows** are extensively used for automating complex actions such as sending service completion notifications, creating follow-up tasks for customers, and updating work order statuses. - **Workflow Rules** can be employed for simpler, declarative automations like triggering email alerts for critical vehicle events (e.g., overdue service). |
| **Scripting** | **Apex Triggers,** **Apex Classes** | **Apex Triggers** handle specific business logic that cannot be achieved declaratively, such as automatically updating Outstanding\_Balance\_\_c on Invoice\_\_c records when Payment\_\_c records are created/updated, or complex calculations for labor pricing. **Apex Classes** support reusable logic for integrations or complex data processing. |
| **Data Modeling** | **Custom Objects: Customer\_\_c, Vehicle\_\_c, ServiceRequest\_\_c, WorkOrder\_\_c, ServiceEstimate\_\_c, Invoice\_\_c, PartsInventory\_\_c, Mechanic\_\_c, Task\_\_c, Payment\_\_c** | Custom objects are created to represent key real-world entities within a garage environment, such as vehicle owners, specific vehicles, detailed service requests, ongoing work orders, parts inventory, and mechanic assignments. **Relationships (Lookup/Master-Detail fields)** are established among them (e.g., ServiceRequest\_\_c linked to Vehicle\_\_c and Customer\_\_c) to ensure data integrity and traceability. |
| **Validation & Rules** | **Validation Rules**, **Formula Fields** | - **Validation Rules** prevent data entry errors and ensure business logic adherence (e.g., preventing Paid\_Amount\_\_c on an invoice from exceeding Total\_Invoice\_Amount\_\_c, ensuring VIN format is correct).<br>- **Formula Fields** are used for real-time calculations such as total labor hours, estimated service completion dates, or remaining part quantities. |
| **Communication** | **Email Alerts, Email Templates, Flows, SMS Integration (via AppExchange/API)** | **Record-Triggered Flows** are configured to send automated notifications (email/SMS) to customers using **Email Templates** for events like service completion, appointment reminders, or overdue service alerts. This significantly enhances customer communication and reduces manual follow-ups. |
| **Reporting & Insights** | **Reports, Dashboards, Report Types** | Custom **Reports** are created (e.g., Service History Report, Revenue by Service Type, Parts Usage Report), and **Dashboards** are built to visualize key performance indicators (KPIs) such as total services completed, average repair time, customer satisfaction scores, and inventory levels, providing actionable insights for garage management. |
| **Access Control** | **Profiles, Permission Sets, Sharing Rules** | Different user roles like Service Advisor, Mechanic, and Garage Admin are granted specific access levels to objects and fields via **Profiles**. **Permission Sets** are used for granular access control (e.g., giving mechanics access only to their assigned work orders). **Sharing Rules** can be implemented for record-level visibility control. |

**Why This Stack Was Chosen:**

* **Salesforce Lightning** allows for rapid development and customization, providing a modern, intuitive, and mobile-ready interface essential for garage staff on the go.
* **Apex** offers the flexibility and power to implement complex business logic unique to vehicle service and repair, such as advanced scheduling algorithms or intricate parts management workflows, beyond what declarative tools can achieve.
* **Flows and Email/SMS Alerts** significantly reduce manual administrative tasks by automating customer communication (e.g., service completion notifications, reminders) and internal process updates, freeing up staff for direct customer interaction and service delivery.
* **Reports and Dashboards** provide comprehensive and real-time insights into garage performance, service trends, labor utilization, and inventory management, enabling proactive decision-making for business growth and efficiency.
* **Validation Rules and Formula Fields** ensure high data integrity, accurate financial calculations, and adherence to service protocols, minimizing errors and improving reliability.
* The **robust data model capabilities** allow for precise tracking of complex relationships between customers, vehicles, parts, services, and mechanics, which is vital for a comprehensive garage management system.