Requirement Analysis

Group-1

Software Requirements Specification (SRS) Document

Ahmed (Jimmy) Abdalla
Aditya Singh Gaharwar
Rhea Sera Rodrigues
Reuben Joslin Coutinho
Reshad Mohsini

Table of Contents

1. Introduction

1.1 Purpose

This document specifies the requirements for the Cloud Auto Repair Platform, designated as version 1.0. The SRS outlines the comprehensive details for the development and implementation of a cloud-based service intended to transform the automotive repair industry. This SRS applies to the web-based component of the platform, which is the focus of the initial release, delineated as a subsystem of a larger, more complex service offering.

The document is intended for use by a range of readers, including but not limited to:

- Software Developers who will implement and test the specified requirements.
- Project Managers who will plan and oversee the project's progress.
- Marketing Staff who will position the product in the market and define its value proposition.
- End-Users, car owners and service providers will interact with the platform.
- Quality Assurance Testers who will verify that the system meets these requirements.
- Technical Writers who will craft user manuals and help documentation based on this SRS.

1.2 Document conventions

- All acronyms will be spelled out in full upon first use and used consistently thereafter.
- Stakeholder feedback identifiers will be denoted using "STK##" (e.g., STK01).

1.3 Project scope

The Cloud Auto Repair Platform is an innovative cloud-based solution that optimizes the interaction between car owners and automotive service providers. Its purpose is to streamline the booking, tracking, and management of automotive repair services. The platform's features include online appointment scheduling, real-time service tracking, digital service documentation, and integration with a network of automotive centers. The aim is to enhance operational efficiency, improve customer satisfaction, and elevate the quality of service delivery. This SRS describes the scope of the platform's initial release, with subsequent enhancements planned in phased rollouts as described in the product roadmap.

1.4 References

- Wiegers, K., & Beatty, J. (2013). Software Requirements. Microsoft Press.

- ProductPlan. (n.d.). Minimum viable product. Retrieved April 4, 2024, https://www.productplan.com/glossary/minimum-viable-product/#:~:text=A%20mini

mum%20viable%20product%2C%20or,iterate%20and%20improve%20the%20produ
ct

- //Institute of Electrical and Electronics Engineers. (1998). IEEE recommended practice for software requirements specifications (IEEE Std 830-1998). IEEE. https://doi.org/10.1109/IEEESTD.1998.88286

- Team Asana. "The Eisenhower Matrix: How to prioritize your to-do list." Asana, 29 Jan. 2024, https://asana.com/resources/eisenhower-matrix

- A. Newportright, "Requirement Prioritization," LinkedIn, [Online]. Available: https://www.linkedin.com/pulse/requirement-prioritization-angel-newportright/

- "What is BDD Testing? A Comprehensive Guide." Katalon, https://katalon.com/resources-center/blog/bdd-testing

- "Achieving Requirements Testability." Prolifics Testing, 10 Oct. 2018, www.prolifics-testing.com/news/achieving-requirements-testability.

- Dean Leffingwell, 'Agile Software Requirements: Lean Requirements Practices for Teams, Programs, and the Enterprise', Addison-Wesley Professional, 2011.

2. Overall description

2.1 Product perspective

The Cloud Auto Repair Platform is an innovative addition to the automotive industry's evolving digital landscape. It does not replace but rather augments the traditional automotive repair process with a cloud-based system designed to simplify and enhance the service experience for both car owners and service providers. This system is a new product, envisioned to become a key player in a market that increasingly values technology integration for convenience and efficiency.

The platform will interface with existing automotive service centers, acting as a facilitator for service bookings and management.

2.2 User classes and characteristics

User Classes	Characteristics
Car Owners	- Seek convenience in booking and tracking automotive services. - Value transparency and timely communication. - Require a user-friendly interface for interacting with the service. - May have varying levels of technical proficiency.
Service Providers	- Automotive centers looking for efficient scheduling and management tools. - Require integration with existing service workflows. - Need detailed reporting and analytics features. - Prefer a system that minimizes administrative overhead.
Platform Administrators	- Monitor system performance and user activity. - Handle support tickets and troubleshoot issues. - Manage platform updates and configurations. - Ensure system security and data integrity.
Third-Party Developers	- Develop additional services or tools for the platform. - Require access to well-documented APIs. - Need a clear understanding of platform limitations and capabilities. - Work within the constraints of the system's architecture.

2.3 Operating environment

The Cloud Auto Repair Platform will operate in a cloud-computing environment, accessible via web browsers across various operating systems (Windows, macOS, Linux) and devices (desktops, laptops, tablets, and smartphones). The platform's servers and databases will be

hosted in geographically distributed data centers to ensure high availability and disaster recovery capabilities. It will need to coexist and integrate with various third-party services, including payment gateways and vehicle diagnostic tools.

2.4 Design and implementation constraints

Development will be constrained by the choice of technologies that best support cloud-based solutions, scalability, and integration with various APIs. The use of Docker containers orchestrated by Kubernetes is prescribed for deployment flexibility and scalability. Furthermore, compliance with data protection laws, such as GDPR, imposes constraints on the design, particularly in how user data is handled and stored.

2.5 Assumptions and Dependencies

Several assumptions underpin the project:
- A growing market acceptance of digital service platforms within the automotive industry.
- The readiness of service centers to integrate their systems with the Cloud Auto Repair Platform.
- Continuous availability of reliable cloud infrastructure services.

Dependencies include
- Stable and robust cloud services infrastructure.
- Third-party APIs for payment processing, GPS tracking, and vehicle diagnostics must remain compatible and fully functional.
- Users' devices must meet minimum hardware and software requirements to access the platform.

3. System features

3.1 User Registration and Login

3.1.1 Description

Enables new users to register for an account and existing users to securely access their accounts. It is a high-priority feature essential for user identity verification and access control.

3.1.2 Functional Requirements

FR3.1.2.1: The system shall provide a registration form that captures user details including name, email address, and password.
FR3.1.2.2: The system must implement email verification to activate new user accounts.
FR3.1.2.3: The system shall offer a secure login process with multi-factor authentication support.
FR3.1.2.4: The system must allow users to reset their password through a verified email.
FR3.1.2.5: The system shall log user access times and dates for security and auditing purposes.

3.2 Car Registration

3.2.1 Description

Allows users to register their vehicles with the platform, providing a basis for service bookings and maintenance records. This feature is of high priority due to its role in customizing the user experience and ensuring accurate service tracking.

3.2.2 Functional Requirements

FR3.2.2.1: The system shall enable users to add multiple vehicles to their profile, including vehicle make, model, year, and VIN.
FR3.2.2.2: The system must allow the editing and removal of vehicle details by the user.
FR3.2.2.3: The system shall verify that the VIN entered is in a valid format and not already registered.

3.3 Request Repair/Maintenance Services

3.3.1 Description

This feature enables users to request specific repair or maintenance services. Prioritizing this feature is vital to the platform's core functionality, allowing users to schedule and manage service appointments.

3.3.2 Functional Requirements

FR3.3.2.1: The system must provide a detailed catalog of services for users to select from.
FR3.3.2.2: The system shall allow users to request a specific date and time for their service appointment.
FR3.3.2.3: The system must send a confirmation notification to the user upon successful service request submission.
FR3.3.2.4: The system shall enable users to view pending and past service requests.

3.4 Specify Problem, Vehicle Details

3.4.1 Description

This feature allows users to provide detailed information about the problem and the specific details of their vehicle. It is crucial for service centers to prepare and provide accurate quotes.

3.4.2 Functional Requirements

FR3.4.2.1: The system shall present users with an interface to describe the problem in free text and optionally attach photos or videos.
FR3.4.2.2: The system must record and associate the problem details with the user's scheduled service appointment.
FR3.4.2.3: The system shall allow users to update the problem description before the service appointment.

3.5 Assign Pickup Request to Driver

3.5.1 Description

Facilitates the automatic assignment of vehicle pickup requests to available drivers, which is critical for the efficient logistics of the service. This feature is a high priority as it directly impacts the user experience and operational efficiency.

3.5.2 Functional Requirements

FR3.5.2.1: The system must automatically assign a pickup request to an available driver based on proximity and estimated time of arrival.
FR3.5.2.2: The system shall provide the driver with the user's preferred pickup time and location.
FR3.5.2.3: The system must allow for manual reassignment of pickup requests by an administrator if necessary.
FR3.5.2.4: The system shall send real-time notifications to users about the driver's identity and estimated pickup time.

3.6 Integration with GPS for Tracking

3.6.1 Description

This feature provides real-time tracking capabilities for users to follow their vehicle's journey from pickup to service center. This medium-priority feature enhances transparency and user trust.

3.6.2 Functional Requirements

FR3.6.2.1: The system must integrate with GPS services to provide real-time location tracking of the vehicle during transit.
FR3.6.2.2: The system shall display the vehicle's current location and the estimated time of arrival at the service center.
FR3.6.2.3: The system must ensure the privacy of the user's location details, revealing them only to authorized parties.
FR3.6.2.4: The system shall provide a way for users to enable or disable GPS tracking for their service requests.

3.7 List of Preferred Service Centers

3.7.1 Description

Allows users to view and select from a curated list of preferred service centers. This high-priority feature supports personalized user experiences and informed decision-making.

3.7.2 Functional Requirements

FR3.7.2.1: The system must generate a list of service centers based on user preferences and vehicle requirements.
FR3.7.2.2: The system shall allow users to favorite or blacklist service centers for future requests.
FR3.7.2.3: The system must provide detailed information about each service center, including services offered, ratings, and reviews.
FR3.7.2.4: The system shall update the list dynamically based on user feedback and service center performance.

3.8 Service Centers Receive Requests

3.8.1 Description

This feature ensures that service centers can receive and manage service requests efficiently. It's a high-priority feature that influences the responsiveness and quality of service.

3.8.2 Functional Requirements

FR3.8.2.1: The system must notify service centers immediately when a service request is made.
FR3.8.2.2: The system shall provide service centers with the ability to accept, reject, or request additional information on service bookings.
FR3.8.2.3: The system must allow service centers to view and manage their queue of upcoming service appointments.
FR3.8.2.4: The system shall enable service centers to communicate with users directly through the platform for clarifications or updates.

3.9 Generate/Send Quotes

3.9.1 Description

This feature enables service centers to generate and send cost estimates or quotes for requested services to users. It's a high-priority feature, integral for maintaining transparency and building trust between the service providers and users.

3.9.2 Functional Requirements

FR3.9.2.1: The system must provide service centers with the ability to generate itemized quotes for services requested by the user.
FR3.9.2.2: The system shall send quotes to users through the platform's messaging system and email notifications.
FR3.9.2.3: The system must allow service centers to modify and resend quotes based on further inspection or user requests.
FR3.9.2.4: The system shall ensure quotes include detailed descriptions of services, parts, labor, and any applicable taxes or discounts.

3.10 Review and Accept/Reject Services

3.10.1 Description

Allows users to review service quotes and either accept or reject the services proposed by the service centers. This feature is critical for user autonomy and satisfaction, making it a high priority.

3.10.2 Functional Requirements

FR3.10.2.1: The system must enable users to review detailed quotes within the platform.
FR3.10.2.2: The system shall provide users with the option to accept the entire quote, reject it, or request modifications.
FR3.10.2.3: The system must notify service centers immediately of the user's decision.
FR3.10.2.4: The system shall provide an option for users to communicate with the service center for any queries regarding the quotes.

3.11 Partial Acceptance of Services

3.11.1 Description

This feature allows users to select specific services they wish to proceed with from the quote, offering flexibility in managing their service requests. It is of medium priority, reflecting the platform's user-centric approach.

3.11.2 Functional Requirements

FR3.11.2.1: The system must allow users to accept only certain items from the quote and reject others.
FR3.11.2.2: The system shall update the service centers about the user's partial acceptance and adjust the service queue accordingly.
FR3.11.2.3: The system must recalculate the final quote based on the services accepted by the user.
FR3.11.2.4: The system shall ensure users can confirm their final selections before proceeding with the service.

3.12 Documentation of Repair Process

3.12.1 Description

A feature that enables service centers to document and update the repair process, providing users with transparency and assurance about the services performed. It's a medium-priority feature that emphasizes accountability and quality of service.

3.12.2 Functional Requirements

FR3.12.2.1: The system must provide a feature for service centers to upload textual and visual documentation of repairs.
FR3.12.2.2: The system shall allow users to view the status updates and documentation of their vehicle's repair process.
FR3.12.2.3: The system must maintain a secure log of all documentation for warranty and service history purposes.
FR3.12.2.4: The system shall enable notifications to users when new documentation is added or updated.

3.13 Detailed History of Services

3.13.1 Description

Enables users to access a comprehensive history of all services performed on their vehicle(s), facilitating easy tracking of maintenance schedules and service records. This feature is high priority for ensuring user satisfaction and vehicle care management.

3.13.2 Functional Requirements

FR3.13.2.1: The system must allow users to view a chronological list of all services performed on their vehicle, including dates, descriptions, and costs.
FR3.13.2.2: The system shall provide detailed service reports, which users can download for personal records.
FR3.13.2.3: The system must enable filtering and searching within the service history for ease of access.
FR3.13.2.4: The system shall securely store service histories, ensuring data integrity and privacy.

3.14 Secure Payment Gateway Integration

3.14.1 Description

Incorporates a secure payment system for users to pay for services rendered, utilizing reputable payment gateways. This is a high-priority feature crucial for transaction security and user trust.

3.14.2 Functional Requirements

FR3.14.2.1: The system must integrate with one or more secure, PCI-compliant payment gateways for processing transactions.
FR3.14.2.2: The system shall encrypt user payment information during transmission and storage.
FR3.14.2.3: The system must provide users with transaction receipts via email and within the platform.
FR3.14.2.4: The system shall support multiple payment methods, including credit/debit cards and online payment services.

3.15 Billing and No-Charge Policy

3.15.1 Description

This feature ensures transparent and fair billing practices, including a policy for not charging users for services they did not approve. It is a high priority to maintain trust and accountability.

3.15.2 Functional Requirements

FR3.15.2.1: The system must itemize bills, clearly listing all charges for approved services.
FR3.15.2.2: The system shall allow service centers to issue no-charge invoices for services declined by the user.
FR3.15.2.3: The system must notify users of any billing discrepancies or adjustments.
FR3.15.2.4: The system shall provide a dispute resolution mechanism for billing issues.

3.16 Towing Services for Non-Drivable Vehicles

3.16.1 Description

Offers a towing service option for users whose vehicles are not in a drivable condition to reach the service center. This medium-priority feature enhances the platform's accessibility and convenience.

3.16.2 Functional Requirements

FR3.16.2.1: The system must allow users to request towing services at the time of booking a repair or maintenance service.
FR3.16.2.2: The system shall display the cost for towing services upfront and include it in the service quote.
FR3.16.2.3: The system must coordinate the scheduling of towing services to align with the service center's availability.
FR3.16.2.4: The system shall provide users with the contact details of the towing service provider.

3.17 Data Protection and Privacy Compliance

3.17.1 Description

Ensures the platform adheres to international data protection and privacy laws, safeguarding user information. This high-priority feature is crucial for compliance and maintaining user trust.

3.17.2 Functional Requirements

FR3.17.2.1: The system must comply with GDPR, CCPA, and other relevant data protection regulations in the jurisdictions it operates within.
FR3.17.2.2: The system shall implement and maintain a comprehensive privacy policy accessible to all users.
FR3.17.2.3: The system must obtain explicit user consent before collecting, processing, or storing personal data.
FR3.17.2.4: The system shall provide users with the ability to access, correct, and delete their personal information upon request.
FR3.17.2.5: The system must regularly undergo privacy impact assessments and compliance audits.

3.18 Secure Data Storage and Transmission

3.18.1 Description

Focuses on implementing robust security measures for the protection of user data during storage and transmission. This feature is of high priority to prevent unauthorized access and data breaches.

3.18.2 Functional Requirements

FR3.18.2.1: The system must encrypt all sensitive data, including personal and payment information, during transmission using SSL/TLS protocols.
FR3.18.2.2: The system shall use secure, encrypted storage solutions for storing user data at rest.
FR3.18.2.3: The system must ensure that data transmitted to third-party services (e.g., payment gateways) is done securely and in compliance with the third party's security requirements.
FR3.18.2.4: The system shall implement access controls and auditing mechanisms to monitor data access and manipulation.
FR3.18.2.5: The system must have a robust data backup and recovery plan to protect against data loss scenarios.

4. Data Requirements

4.1 Logical data model

The Cloud Auto Repair Platform will utilize a relational database model to store and manage data. The core entities include:

- Users (car owners, service center personnel, platform administrators)
- Vehicles (car make, model, year, license plate number)
- Service Centers (name, location, contact information, service specialties, ratings/reviews)
- Service Requests (car owner, vehicle, requested service details, date/time submitted)
- Quotes (service center, service request, estimated cost breakdown)
- Service Orders (confirmed service request, accepted quote, service details, payment information)
- Payments (service order, payment method, amount, transaction date)

These entities will be linked through relationships to ensure data integrity and efficient retrieval.
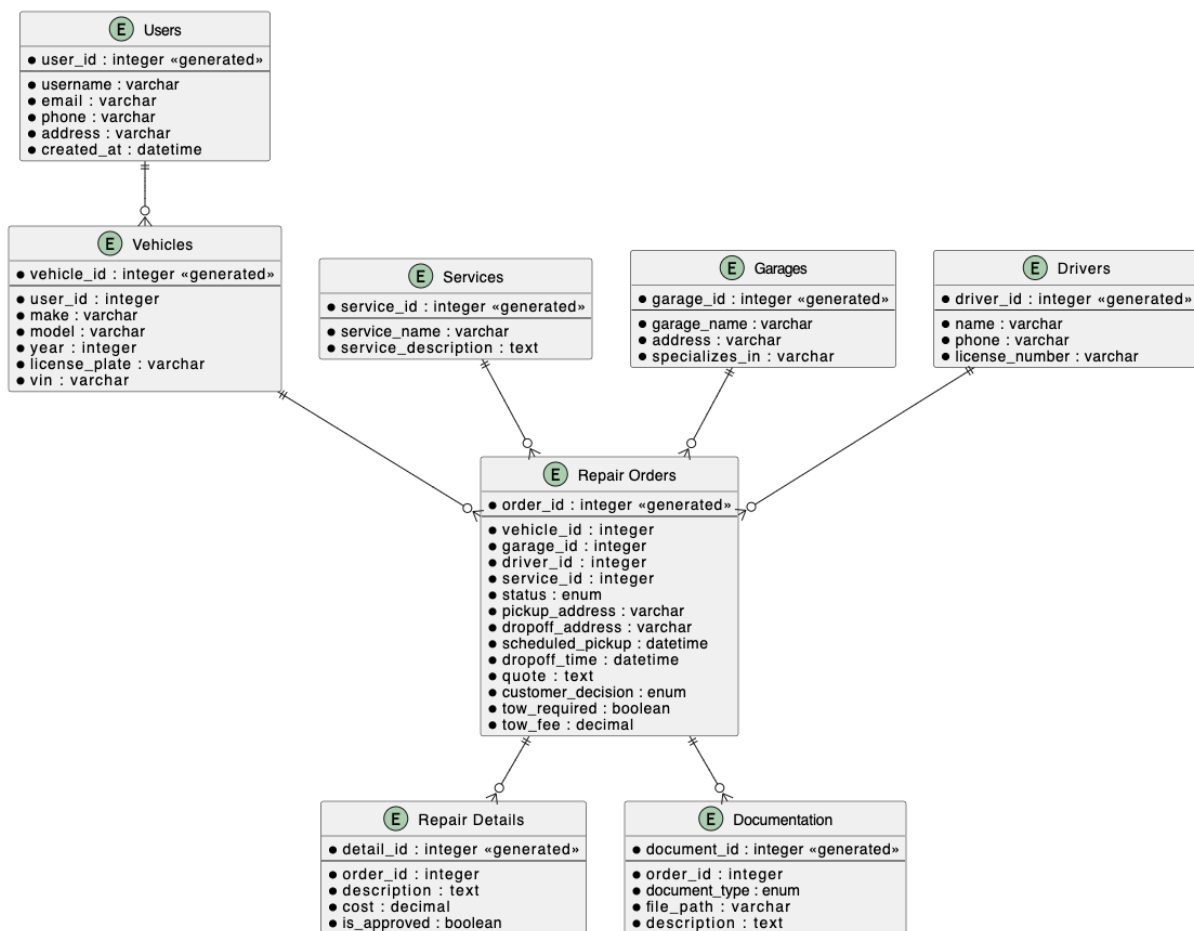
4.2 Data dictionary

Data Element	Description	Composition or Data Type	Length	Values
User	**Represents an individual user of the system, either a car owner or a service center operator.**	**UserID + Name + Email + PasswordHash + Phone**		
UserID	Unique identifier for a user	Integer		System-generated
Name	User's full name	String	Max 255	
Email	User's email address	String	Max 255	Valid email format
PasswordHash	Encrypted user password	String		Hashed format
Phone	User's contact phone number	String	Max 15	
Vehicle	**Details about vehicles owned by users, used to track service requests.**	**VehicleID + UserID + Make + Model + Year + VIN**		
VehicleID	Unique identifier for the vehicle	Integer		System-generated
UserID	Unique identifier of the user owning the vehicle	Integer		Reference to User
Make	Manufacturer of the vehicle	String	Max 100	
Model	Model of the vehicle	String	Max 100	
Year	Year of manufacture	Integer	4 digits	1900 - Current Year
VIN	Vehicle Identification Number	String	17 chars	Alphanumeric
Service Request	**A request made by the user for vehicle repair or maintenance services.**	**RequestID + UserID + VehicleID + ServiceCenterID + Description**		

		+ Status + Date		
RequestID	Unique identifier for a service request	Integer		System-generated
UserID	Unique identifier of the user making the request	Integer		Reference to User
VehicleID	Unique identifier of the vehicle needing service	Integer		Reference to Vehicle
ServiceCenterID	Unique identifier for the service center to fulfill the request	Integer		Reference to Service Center
Description	Description of the service requested	String		Textual description
Status	Current status of the service request	String		Pending, "In Progress", "Completed"
Date	Date when the service request was made	Date		YYYY-MM-DD
Service Center	**Information about service centers that provide vehicle repair and maintenance.**	**ServiceCenterID** **+ Name** **+ Location** **+ ServicesOffered**		
ServiceCenterID	Unique identifier for a service center	Integer		System-generated
Name	Name of the service center	String	Max 255	
Location	Physical location of the service center	String		Address format
ServicesOffered	Services provided by the service center	Text		List of services
Transaction	**Details of financial transactions for services rendered.**	**TransactionID** **+ RequestID** **+ Amount** **+ PaymentMethod** **+ Date**		
TransactionID	Unique identifier for a transaction	Integer		System-generated
RequestID	Unique identifier of the service request related to	Integer		Reference to Service Request

	the transaction			
Amount	Amount charged for the service	Decimal		Positive decimal number
PaymentMethod	Method used for payment	String	Max 50	Credit Card, "Debit Card", "PayPal", etc.
Date	Date when the transaction was processed	Date		YYYY-MM-DD

4.3 Reports

The platform will generate various reports to provide valuable insights for both car owners and service centers.

Report Type	Content Description	Sort Sequence	Totaling	Layout Constraints	Predefined Layout
Service History Report	List of service events with details	By service date, newest first	Total cost per service type and overall	Tabular layout with detailed expansion	Standard company branding template
Cost Breakdown by Service Type	Expenses broken down by service type	By total cost, descending	Subtotals by service type, grand total	Visual graphs or charts alongside tables	Accessibility-compliant visual elements
Service Center Performance Ratings	Average ratings and reviews	By average rating, the highest first	N/A	Graphical star-rating and textual feedback	User interface reflective of the platform's rating scale
Service Request Volume	Number of service requests by category	By volume, descending order	Counts per category, summary total	Comparative bar or line charts	Consistent with service center dashboard design
Customer Acquisition Report	New vs. returning customer stats	Chronological	Monthly and annual numbers	Funnel charts for conversion rates	Fits within the existing service center report framework
Revenue Reports	Earnings by service type and customer	By revenue, the highest-earnin	Total revenue per category, overall total	Detailed tables and summary visualizations	Easy reading and printing for financial

		g categories first			reporting
User Activity Reports	Login frequency, booking patterns	By activity level, most to least active	Total activities per feature and time frame	Interactive, dynamic for real-time analysis	Dashboard-oriented, adaptable to screens and devices
Platform Performance Metrics	System uptime, response times, errors	By component and severity	Summary metrics for performance indicators	Real-time updates with gauges and charts	IT department standards for system monitoring
Service Gap Analysis	Unmet demand and growth areas	By demand level, highest needs first	Not typically applicable	Heat maps and geographic visualizations	Compatible with strategic planning tools

4.3.1 Report Specification Template Example

Report Element	Service History Report Specification
Report ID	SHR-001
Report Title	Service History Report
Report Purpose	To provide a comprehensive record of all service events a vehicle has undergone, including maintenance and repairs.
Decisions Made from Report	Assisting vehicle owners in scheduling future services, assessing vehicle maintenance history for resale, and monitoring service expenses.
Priority	High
Report Users	Vehicle owners, service center staff
Data Sources	Vehicle service database
Frequency and Disposition	Generated on-demand by the user; available for display on the platform interface and printable upon request.
Latency	Instantaneous display with up-to-date information as soon as the user requests it.
Visual Layout	Portrait orientation with a clear and legible tabular layout for detailed viewing. Adheres to standard company branding template for consistency across the platform.
Header and Footer	Header includes the report title and generation date/time. Footer includes page numbers and a standard confidentiality statement.

Report Body	The body is organized in a table listing each service event, sorted by service date from newest to oldest. It includes columns for service type, date, service center, cost, and notes. Each row allows for expansion to view detailed service descriptions and outcomes.
End-of-Report Indicator	A clear statement or graphical indicator at the end of the report signifying no more data follows, such as "End of Report."
Interactivity	Users can interact with the report to filter by service type, date ranges, or service center. The detailed expansion of service events is clickable for ease of use.
Security Access Restrictions	Access is restricted to the vehicle owner and authorized platform administrators. Service center staff can access the report only with explicit permission from the vehicle owner.

4.4 Data acquisition, integrity, retention, and disposal

Data Acquisition:

- Data is collected directly from user inputs, service center submissions, and automated system tracking of transactions and service records.
- Initial data seeding for service centers includes bulk uploads of their service offerings, pricing, and available schedules.
- Subsequent data updates occur in real-time as users interact with the platform and service centers update their information.

Data Integrity:

- All data transactions are conducted over secure, encrypted channels to prevent unauthorized data interception.
- The platform employs field validation to ensure data accuracy and consistency at the point of entry.
- Regular data quality checks are implemented to identify and correct any anomalies or inconsistencies.
- A version control system is in place for all data entries, ensuring traceability and accountability.

Data Retention:

- User and service records are retained indefinitely to provide a comprehensive service history and support ongoing customer relationship management.
- Financial records are retained in accordance with legal and regulatory requirements, typically for a minimum of seven years.
- Data backups are performed daily, with incremental backups throughout the day and full backups nightly.
- Backups are stored in geographically separate, secure locations to ensure disaster recovery capabilities.

Data Disposal:
- Data disposal procedures are compliant with relevant privacy laws and regulations, such as GDPR or CCPA.
- Users can request the deletion of their personal data, which the platform will execute, leaving only anonymized records for historical analysis.
- When disposing of data, the platform employs secure deletion practices to ensure that data cannot be reconstructed or retrieved.
- Temporary data and cache are purged regularly, with the frequency determined by usage patterns and storage constraints.

Data Security:
- The platform uses role-based access control (RBAC) to limit data access to authorized users only.
- Regular security audits are conducted to identify and mitigate potential vulnerabilities.
- Incident response plans are in place for data breaches or loss, including immediate remediation actions and notifications to affected parties.

Archive and Backup:
- Archived data is subject to the same security controls as active data, ensuring its confidentiality and integrity.
- Backup data is tested regularly to ensure that it can be reliably restored in the event of system failure or data loss.

5. External Interface Requirements

5.1 User interfaces

- Standards to Follow: Adhere to the Material Design guidelines for intuitive and responsive interfaces.
- Fonts and Icons: Use Google's Roboto font for text and Material Icons for graphical elements.
- Screen Layout: Must be responsive, suitable for devices with screen sizes ranging from 360x640 to 1920x1080 pixels.
- Standard Features: Include a persistent navigation bar with Home, Services, History, and Help buttons. Implement global shortcut keys, e.g., Alt + H for Home.
- Validation and Error Handling: Real-time input validation with user-friendly error messages displayed near the form fields.
- Accessibility: Comply with WCAG 2.1 standards for accessibility, ensuring compatibility with screen readers and provisions for color blindness.

5.2 Software interfaces

- Database Communication: Utilize RESTful APIs for interactions between the front-end application and the backend database (PostgreSQL, version 12).
- External Services:
    - Payment Processing: Integrate with Stripe API for handling transactions.
    - Email Services: Integration with SendGrid for sending transactional emails and notifications.
- Data Formats: JSON for data exchange across APIs.
- Security: Implement OAuth 2.0 for authentication and authorization. Data encryption using TLS for data in transit.

5.3 Hardware interfaces

- Not applicable for the current scope of the Cloud Auto Repair Platform as it primarily operates as a web-based service. Future expansions into IoT for vehicle diagnostics will necessitate revisiting this section.

5.4 Communications interfaces

- Email Communication: SMTP protocol via SendGrid for outbound emails. Support HTML email templates with fallback to plain text.
- Network Protocols: HTTP/2 for enhanced performance in web communications. Secure WebSocket (wss://) for real-time notifications and updates.

- Security: Enforce HTTPS for all web traffic. Implement CSP and HSTS policies for web security.
- Data Transfer Rates: Ensure the system is optimized for minimum bandwidth consumption, with asynchronous data loading where possible.
- Synchronization: Use webhooks for real-time updates from integrated third-party services like payment gateways.

6. Quality Attributes

6.1 Usability

6.1.1 Ease of Use

- The platform shall be navigable by a user unfamiliar with the system within 5 minutes of initial use.
- Priority: High over ease of learning.

6.1.2 Error Recovery

- The system must provide clear, understandable error messages and guidance for recovery within 10 seconds of error occurrence.
- Priority: Critical for user confidence.

6.1.3 Accessibility

- The platform must comply with WCAG 2.1 AA standards, ensuring accessibility for users with disabilities.
- Priority: High, to ensure inclusivity.

6.2 Performance

6.2.1 Response Time

- The system shall respond to user input within 2 seconds under normal operation conditions.
- Priority: High for user satisfaction.

6.2.2 System Capacity

- Requirement: The platform must support up to 10,000 concurrent users without degradation of performance.
- Priority: Medium, scalable as the user base grows.

6.3 Security

6.3.1 Data Encryption

- All user data transmitted over the internet must be encrypted using TLS 1.2 or higher.
- Priority: High to protect user privacy.

6.3.2 Access Control

- The system must implement role-based access control (RBAC) with at least three levels of user roles.
- Priority: High to ensure data integrity and access restriction.

6.4 Safety

6.4.1 Data Loss Prevention

- The platform must automatically back up data daily to a remote server with a recovery point objective (RPO) of 24 hours.
- Priority: High to prevent loss or damage.

6.4.2 Hazard Prevention

- The system must log all failed login attempts and lock user accounts for 30 minutes after 5 failed attempts.
- Priority: Medium to prevent unauthorized access

6.6 Reliability

6.6.1 System Availability

- The platform must achieve a system availability of 99.9% uptime, excluding scheduled maintenance periods.
- Priority: High - To ensure continuous operation and maintain user trust and satisfaction.
- Measurement: Availability percentage calculated as (Total Time - Downtime) / Total Time * 100 over a calendar month.

6.6.2 Fault Tolerance

- The system must automatically recover from failure of any single component without user-visible downtime or data loss.
- Priority: High - Critical for minimizing impact on users during component failures.
- Measurement: Successful failovers without user intervention or data inconsistency issues during unplanned outages of individual components.

6.6.3 Backup and Recovery

- The platform must perform daily backups of all data, with the ability to restore any user data within 4 hours of loss or corruption.
- Priority: High - Essential for data integrity and recovery in case of system failures.
- Measurement: Recovery Time Objective (RTO) not exceeding 4 hours from the time of recovery initiation.

6.7 Scalability

6.7.1 User Load Handling

- The platform must be capable of scaling to support up to 100,000 concurrent users with no more than a 10% increase in response time.
- Priority: Medium - Important for maintaining performance during peak usage periods.
- Measurement: System response time under varying load conditions, measured and compared to baseline performance metrics.

6.7.2 Resource Elasticity

- The system infrastructure must automatically scale resources up or down based on real-time demand, ensuring efficient resource use without manual intervention.

- Priority: Medium - To optimize operational costs while accommodating fluctuating workloads.
- Measurement: Number of automatic scaling events that accurately match load changes without human intervention.

6.7.3 Modular Design

- The platform's architecture must support modular updates and enhancements without significant downtime or impact on existing functionality.
- Priority: Medium - To facilitate future growth and evolution of the platform.
- Measurement: Time and effort required to deploy updates or add new features, with minimal disruption to service availability.

7. Internationalization and Localization Requirements

7.1 Language Support

7.1.1 Multi-Language Interface

- The platform must support the display and input of text in English, Spanish, French, German, and Mandarin at launch, with the framework to add additional languages as needed.
- Priority: High - Essential for global user accessibility and usability.

7.2 Cultural Adaptability

7.2.1 Cultural Sensitivity

- The platform's content must adapt to avoid cultural sensitivities, including symbols, colors, and imagery, which could be considered inappropriate in target markets.
- Priority: High - To ensure cultural appropriateness and respect..

7.3 Formats and Conventions

7.3.1 Locale-Specific Formatting

- Automatically adapt the formatting of dates, times, currency, addresses, telephone numbers, and numerical values to match the user's locale settings.
- Priority: High - Critical for user understanding and engagement.

7.4 Regulatory Compliance

7.4.1 Legal and Regulatory Adherence

- Requirement: The platform must comply with international regulations and laws relevant to the automotive repair industry, including data protection and consumer rights, specific to each country of operation.
- Priority: High - Mandatory for legal operation in international markets.

7.5 Time Zone and Geolocation

7.5.1 Time Zone Support

- The platform must accurately reflect the user's local time zone in all time-related data displays and transactions.
- Priority: High - Essential for scheduling and service tracking functionalities.

7.6 Measurement Systems

7.6.1 Unit Conversion

- Provide the option for users to view and enter data in the measurement system of their choice (Imperial or Metric) and perform automatic conversions as necessary.
- Priority: Medium - Important for user convenience in international contexts.

8. Other Requirements

8.1 Legal and Regulatory Compliance

- Data Protection: The platform will comply with global data protection regulations such as GDPR in the EU and CCPA in California, USA, ensuring users' privacy rights are protected.
- Financial Transactions: All financial transactions will adhere to PCI DSS standards to ensure secure and compliant handling of credit card information.
- Vehicle Data: Compliance with local and international automotive service and repair standards to ensure that vehicle data handling meets industry regulations.

8.2 Installation and Configuration

- Installation Requirements: Detailed guidelines for installing any necessary software components on various platforms (Windows, macOS, Linux) or prerequisites for the web platform.
- Configuration Guidelines: Step-by-step configuration instructions for initial setup, including database configurations, API keys setup for third-party services, and environment-specific settings.

8.3 Startup and Shutdown Procedures

- Startup: Procedures for starting the service, including checks for data integrity, system dependencies, and preliminary data caching for performance optimization.
- Shutdown: Graceful shutdown procedures ensuring data integrity, proper disconnection from services, and cleanup of temporary files.

8.4 Logging, Monitoring, and Audit Trails

- Logging: Implement comprehensive logging for both system operations and user activities to support debugging, performance monitoring, and security auditing.
- Monitoring: Utilize tools and practices for real-time system monitoring, alerting for performance anomalies, and potential security incidents.
- Audit Trails: Maintain detailed audit trails for all critical activities, especially those related to user data access and manipulation, financial transactions, and system configuration changes, to support compliance and investigative activities.

8.5 Transition Requirements

- Data Migration: For migrating from existing systems to the Cloud Auto Repair Platform, specify requirements for data conversion tools or services, ensuring data integrity and compatibility.
- Training Materials: Development and availability of training materials for users and service center operators to facilitate a smooth transition to the new platform.
- Legacy System Integration: Where necessary, define requirements for integrating with or gradually phasing out legacy systems without disrupting current operations.

8.6 Environmental and Social Responsibility

- Eco-friendly Operations: Strategies for minimizing the environmental impact of the platform's operations, including optimizing data storage and transmission for energy efficiency.
- Community Engagement: Initiatives for engaging with and giving back to the communities the platform serves, such as promoting local service centers and supporting automotive safety and maintenance education.

Appendix A: Glossary

Term	Description
API	Application Programming Interface. A set of rules and definitions that allows software programs to communicate with each other.
Cloud Auto Repair Platform	The proposed cloud-based system designed to streamline the booking, tracking, and management of automotive repair services.
Data Protection and Privacy Compliance	Ensures that the platform adheres to international data protection and privacy laws, safeguarding user information.
ERP	Enterprise Resource Planning. A type of software that organizations use to manage day-to-day business activities.
GDPR	General Data Protection Regulation. A regulation in EU law on data protection and privacy in the European Union and the European Economic Area.
GPS	Global Positioning System. A satellite-based navigation system used to determine the ground position of an object.
PCI-compliant	Payment Card Industry-compliant. A set of security standards designed to ensure that all companies that accept, process, store, or transmit credit card information maintain a secure environment.
SRS	Software Requirements Specification. A document that describes what the software will do and how it will be expected to perform.
Service Center	An establishment where vehicles are repaired by auto mechanics and technicians.
Third-Party APIs	Interfaces provided by third parties that allow developers to access some features or data of an external software application or service.
User Interface (UI)	The means by which the user and a computer system interact, in particular the use of input devices and software.

Vehicle Identification Number (VIN)	A unique code used by the automotive industry to identify individual motor vehicles.
SSL/TLS protocols	Secure Sockets Layer/Transport Layer Security protocols. Cryptographic protocols are designed to provide communications security over a computer network.
Data integrity	The accuracy, consistency, and reliability of data throughout its lifecycle.
Data retention	Policies and procedures that govern the retention of data for compliance with legal and business data archival requirements.
User Experience (UX)	The overall experience of a person using a product such as a website or a computer application, especially in terms of how easy or pleasing it is to use.
Multi-factor Authentication (MFA)	A security system that requires more than one method of authentication from independent categories of credentials to verify the user's identity for a login or other transaction.
Cloud-computing environment	A model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources.
Data backup and recovery plan	Strategies and processes used to ensure the preservation and restoration of data in case of loss.
Responsive design	An approach to web design that makes web pages render well on a variety of devices and window or screen sizes.

Appendix B: Analysis Models

This section includes analysis models used during the development process:
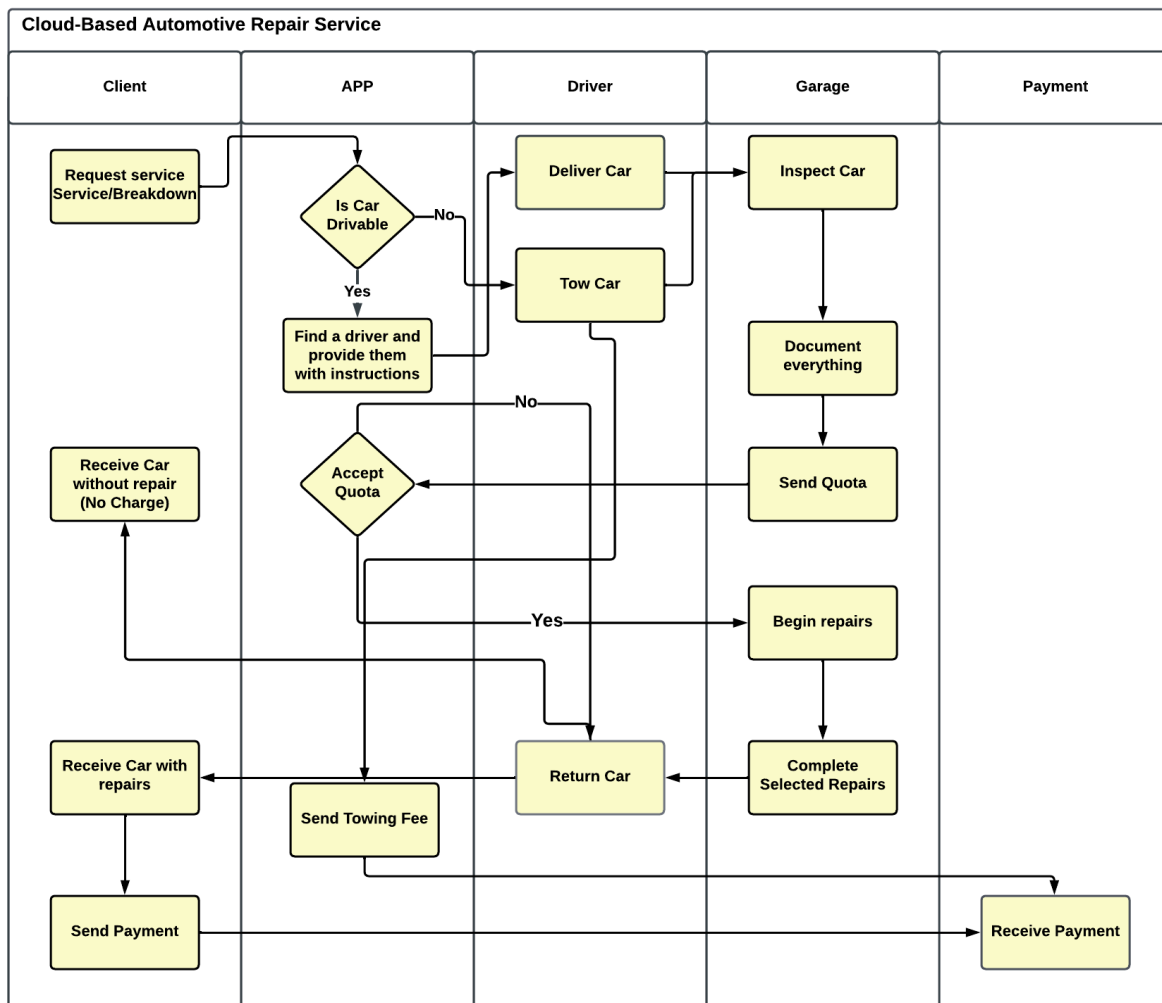
Use case diagram:
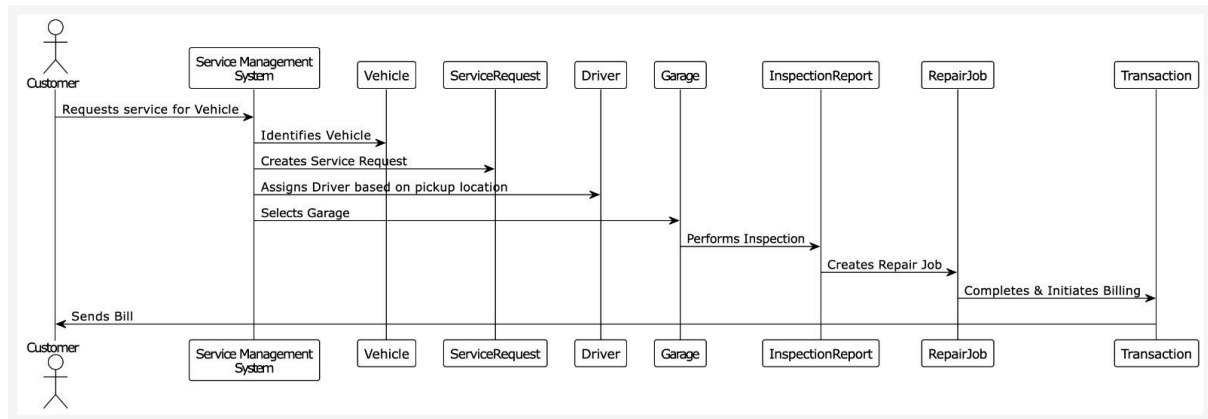
Actors/Participants:

- **Customer**: Initiates the service by requesting vehicle servicing.
- **ServiceManagementSystem**: The main system coordinating the service requests and interactions.
- **Vehicle**: An entity representing the customer's vehicle.
- **ServiceRequest**: A specific request for service lodged by the customer.
- **Driver**: Person responsible for transporting the vehicle to and from the garage.
- **Garage**: The place where the vehicle is serviced.
- **InspectionReport**: A report generated after inspecting the vehicle.
- **RepairJob**: Actual repair tasks performed on the vehicle.
- **Transaction**: Handles billing and payment processes.

Sequence of Events:

- The Customer requests service for their Vehicle.
- The ServiceManagementSystem identifies the Vehicle and captures its details.
- A ServiceRequest is then created by the system.
- The system assigns a Driver based on the pickup location.
- The Driver heads to the Garage with the vehicle.
- The Garage performs an inspection of the vehicle, which results in an InspectionReport.
- Based on this report, a RepairJob is created, detailing the repair work to be done.
- After the repair job is completed, the Transaction process begins, completing and initiating billing.
- The Customer is then sent the bill for the services provided.

Swimlane Diagram:

Cloud-Based Automotive Repair Service

Client	APP	Driver	Garage	Payment

Request service Service/Breakdown → Is Car Drivable — No → Deliver Car → Inspect Car

Is Car Drivable — Yes → Find a driver and provide them with instructions

Tow Car

Document everything

Accept Quota — No

Receive Car without repair (No Charge)

Send Quota

Accept Quota — Yes → Begin repairs

Receive Car with repairs ← Return Car ← Complete Selected Repairs

Send Towing Fee

Send Payment → Receive Payment

1. **Client**: The process begins when the client requests either service or breakdown assistance for their car.
2. **APP**: The application determines if the car is drivable.
   - If the car is drivable, the app finds a driver and provides them with instructions.
   - If the car is not drivable, it triggers the towing process.
3. **Driver**:
   - If the car is drivable, the driver delivers the car to the garage.
   - If the car is not drivable, the driver tows the car to the garage.
4. **Garage**:
   - Once the car arrives at the garage, it is inspected.
   - Everything about the condition of the car is documented.
   - The garage sends a quota to the client for the necessary services.
   - If the client accepts the quota, the garage begins repairs. If not, the process of returning the car is initiated.
5. **Client**:
   - If the client accepts the quota, they will eventually receive the car with repairs.

- If the client declines the quota, they receive the car without repairs and no charge is applied, unless the car was towed.
6. **Payment** (Newly added lane):
   - If towing is required, the client sends a towing fee.
   - After repairs are completed and the car is returned, the client sends payment for the services.

The additional "Payment" lane in the updated diagram helps to separate and organize the financial transactions from the service delivery aspects of the process, allowing stakeholders to more easily track the flow of money separately from the operational steps. This clear division enhances the readability of the diagram and helps in identifying any financial-specific steps or requirements, such as invoicing and payment processing.

State Transition Diagram:
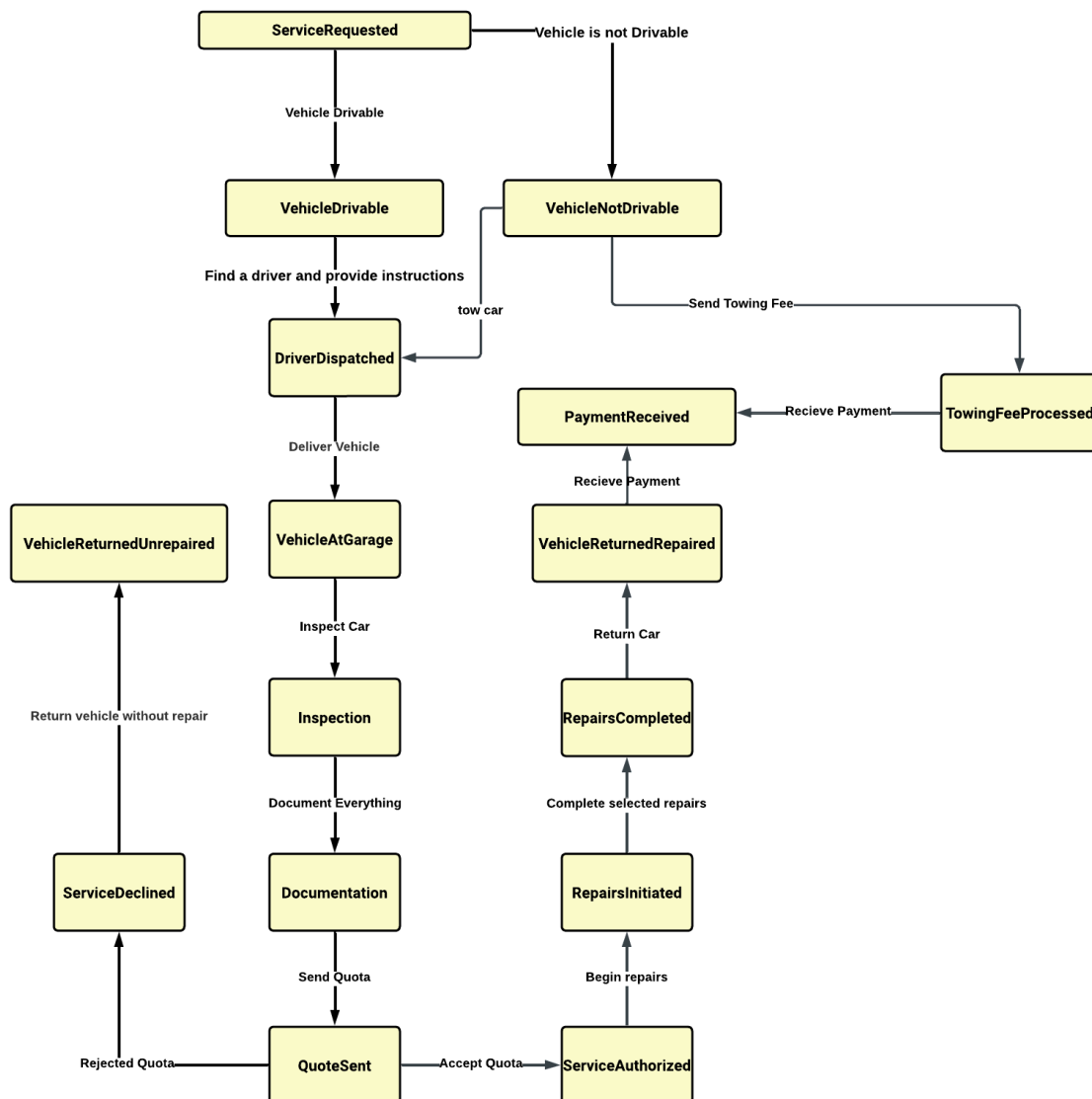
1. **ServiceRequested**: The process begins when a client requests a service or reports a vehicle breakdown through the application.
2. **VehicleDrivable** / **VehicleNotDrivable**: The application then determines if the vehicle can be driven to the garage.
   - If the vehicle is drivable, the application proceeds to dispatch a driver.
   - If not, the towing process is initiated, leading to a "TowingFeeProcessed" state before proceeding.
3. **DriverDispatched**: A driver is dispatched to the client's location to either.
   - Drive the vehicle to the garage, leading to the "VehicleAtGarage" state.
   - Tow the vehicle if it's not drivable, leading to "PaymentReceived".
4. **VehicleAtGarage**: Once at the garage, the vehicle undergoes:
   - An **inspection**, leading to documentation of its condition.
   - A quotation for repair work, moving to the "QuoteSent" state.

5. **QuoteSent**: After the client receives the quote:
   - If the service is declined, the vehicle is returned unprepared, leading to "VehicleReturnedUnrepaired".
6. **Service Authorized**: If the client authorizes the service based on the quote provided, this leads to "Repairs Initiated".
7. **RepairsInitiated**: The garage begins the repair process, leading to:
   - "RepairsCompleted": The vehicle is successfully repaired and returned to the client ("VehicleReturnedRepaired").
8. **RepairsCompleted**: After the repairs have been initiated and all necessary work has been carried out on the vehicle, the state transitions to:
   - **VehicleReturnedRepaired**: The vehicle is successfully repaired and ready to be returned to the client.
9. **PaymentReceived**: This state is reached after the towing fee has been processed (if there was a towing fee) or once the vehicle repair is completed and the client has been billed.
   - **TowingFeeProcessed**: If the vehicle is not drivable and requires towing, the fee is processed and the system transitions to this state awaiting payment.
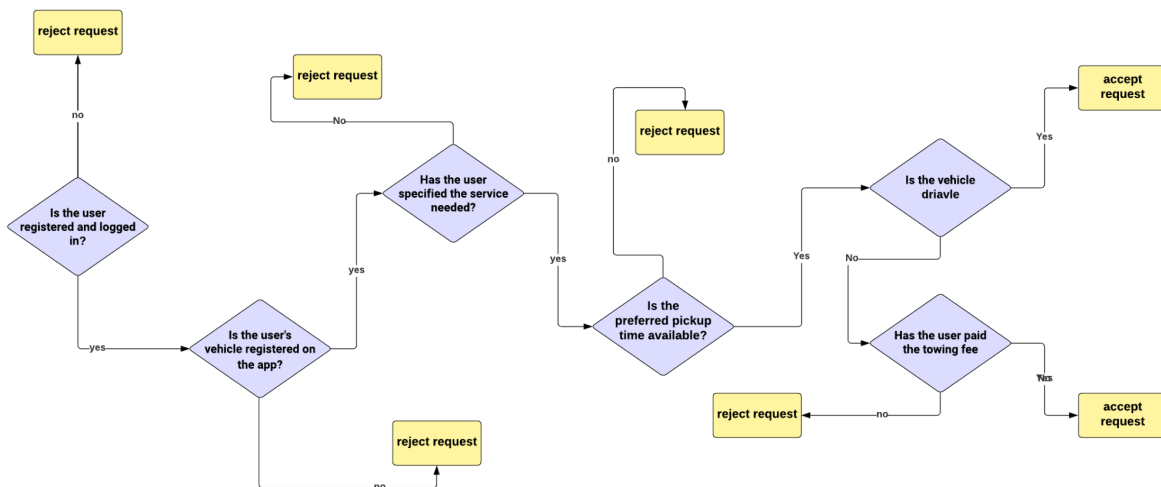
The decision tree for processing service requests within the cloud-based automotive repair service system consists of a series of checks that lead to an action to either accept or reject the request. The process is as follows:

1. **User Status Check**: The system first verifies if the user is registered and logged in. If this condition is not met, the request is rejected.

2. **Vehicle Registration Check**: For a logged-in user, the next check is whether the user's vehicle is registered on the app. If not registered, the request is rejected.

3. **Service Specification Check**: For a registered vehicle, the system then determines if the user has specified the required service. Absence of this specification leads to rejection of the request.

4. **Pickup Time Availability Check**: Upon service specification, the system checks the availability of the preferred pickup time. Unavailability of the time slot results in the request being rejected.

5. **Vehicle Drivability Check**: If the preferred pickup time is available, the system checks the drivability of the vehicle. If the vehicle is drivable, the system proceeds with the request.

6. **Towing Fee Payment Check**: If the vehicle is not drivable, the system requires confirmation of towing fee payment. Failure to confirm payment results in rejection of the request.

7. **Request Acceptance**: The request is accepted only if all conditions are positively met, including the payment of the towing fee when applicable.

This decision tree ensures that all necessary conditions are reviewed before a service request is processed, optimizing the flow of operations and ensuring that system actions are consistent with user inputs and statuses.