Software Requirements Specification

for

EZTransport

Transport and Logistics Management System

Manipal Institute of Technology, Manipal

Prepared By:

Anvita Warjri (230953100)
Riddhima Jain (230953108)
Adriteyo Das (230953244)

Version 1.0

1st April 2025

Contents

Revision History

Name	Date	Reason For Changes	Version

1 Introduction

1.1 Purpose

The objective of the "Transport and Logistics Management System" is to streamline the management of shipments, vehicles, drivers, and waypoints for logistics companies. This SRS describes the initial release of this software (Version 1.0). The scope of this SRS covers the full functionality of our project.

1.2 Document Conventions

Times New Roman is the font used throughout the document, with headings in Bold. Sections and Subsections are numbered to give a clear way of understanding the project's structure and development. These conventions enhance readability, organization, and clarity.

1.3 Intended Audience and Reading Suggestions

The SRS is intended for getting a comprehensive idea about the functionality of the software for consumers and developers alike. Special care has been taken to simplify the working of the various subsystems for ease in grading and gauging quality of the project by our respective examiners.

1.4 Product Scope

The Transport and Logistics Management System simplifies tracking shipments using waypoints, assigns drivers and vehicles to shipments, and provides real-time updates to customers. Its goals include eliminating manual processes, providing transparent tracking for customers, and enhancing efficiency in logistics operations. By achieving these objectives, the software aligns with corporate goals of enhancing customer satisfaction and operational efficiency in the logistics industry.

2 Overall Description

2.1 Product Perspective

The Transport and Logistics Management System specified in this SRS is an independent solution created to streamline logistics operations. It is not part of a product family or a replacement for existing systems. Instead, it functions as a self-contained product, facilitating interactions between administrators, drivers, and customers. This software operates autonomously, without relying on external systems or components.

2.2 Product Functions

The Transport and Logistics Management System comprises key system features to facilitate efficient logistics management. The system begins with a robust user authentication and authorization mechanism, allowing users to securely register, log in, and access the platform based on their roles—whether as Administrators, Drivers, or Customers.

Administrators can manage the overall system, including adding/removing vehicles, assigning drivers, and overseeing shipments. Drivers are responsible for updating the status of shipments as they reach predefined waypoints. Customers can place shipment requests, track the status of their shipments, and receive notifications at each waypoint.

When a customer places a shipment request, they provide details such as the sender, receiver, package details, and weight. The system assigns a driver and vehicle to the shipment and defines a sequence of waypoints for tracking. As the driver reaches each waypoint, they update the system, and the customer receives a notification. Once the shipment is delivered, the status is updated to "Delivered," and the customer is notified.

The system ensures transparency and efficiency in logistics operations by automating tasks like driver assignment, waypoint tracking, and customer notifications. All information is stored in the database, and the Administrator has full authority to view, add, delete, or modify any details in the system.

2.3   User Classes and Characteristics

We anticipate 3 major user classes for this software:

1. **Administrators**: Users who manage the overall system, including adding/removing vehicles, assigning drivers, and overseeing shipments. They have the highest level of system access and are responsible for system maintenance and management.

2. **Drivers**: Users who are responsible for transporting shipments and updating the status of shipments as they reach predefined waypoints. They interact with the system primarily through the Driver Module.

3. **Customers**: End-users who place shipment requests, track the status of their shipments, and receive notifications. They interact with the system through the Customer Module and represent the primary user base.

The functionality aims to satisfy all three user classes, with particular emphasis on providing transparent tracking for customers and efficient management tools for administrators.

2.4   Operating Environment

The database application will be set up on an MySQL server. It will be accessed using a React.js application as front end. The react application will implement an HTML website and CSS for aesthetics and utilize JavaScript to handle requests. Flask will be implemented for the API endpoints to connect to the backend MySQL server using the appropriate Python-MySQL Connector

2.5   Design and Implementation Constraints

While industry standards and software design conventions have been followed closely for this project, complete functionality and deployment will be taken up in the later versions. Deployment of the website and database along with debugging support have not been included in this iteration. Users will have an encryption-less login, which will be improved further. The guidelines explicitly dictate usage of SQL database systems, therefore other database systems haven't been explored.

2.6 User Documentation

A standard documentation underlining usage of the MySQL database from their official website along with a comprehensive GitHub repository will be provided to help users with the functionality of the website.

2.7 Assumptions and Dependencies

The project is designed using modern industry level applications which are tried and tested and regularly maintained, hence we do not take into account faults on a fundamental level to affect the project. The linking of the various platforms, however, are subject to constant updates in terms of libraries and functions used, under whose scrutiny our projects bugs and effectiveness depends on. Other miscellaneous details such as presence of tools and basic dependencies on the client's end may also have an effect on the user experience.

3 External Interface Requirements

3.1 User Interfaces

The user interface is designed with a landing page providing access to login and register pages. After login, each user type is directed to their specific module:

1. **Admin Module**: Administrators can manage users, vehicles, drivers, and shipments. They can add, edit, or delete any system entities and oversee the entire operation.

2. **Driver Module**: Drivers can view assigned shipments, update waypoints as they reach them, and track progress of their current deliveries.

3. **Customer Module**: Customers can place shipment requests, track shipments, view status updates, and receive notifications about their shipments.

Standard GUI elements ensure consistency, error messages follow conventions, and a separate specification details the comprehensive user interface design for a streamlined experience.

3.2 Hardware Interfaces

The website is accessible and compatible with various hardware devices, including desktop computers, laptops, tablets, and smartphones to access the interface across all the devices.

3.3 Software Interfaces

- **Database System:** Oracle (latest stable version):
  The website utilizes Oracle as its primary database system to store user profiles, shipment details, waypoints, etc.

- **Backend Framework:** The website utilizes Node.js or Flask as the primary backend framework, providing a robust and scalable platform for developing web applications.

- **Frontend Framework:** React.js (latest stable version):
  The user interface is developed using React.js for a dynamic webpage and the design is done through CSS.

- **Data Items:** User profiles, Vehicle details, Driver information, Shipment data, Waypoint tracking, Notification logs, Package information, and Location data.

3.4 Communications Interfaces

- **Communication Protocols:** The application uses standard HTTP/HTTPS protocols for communication between the front-end and back-end components.

- **Security:** Communication is secured with use of JWT or CSRF tokens to ensure the confidentiality and integrity of data during transmission.

- **Error Messages:** Clear and concise error messages are displayed to users in case of communication failures or other issues, aiding in troubleshooting.

- **Synchronization Mechanisms:** Real-time updates and synchronization are implemented to ensure consistent data across all connected devices.

4 System Features

4.1 User Authentication

4.1.1 Description and Priority

The User Authentication feature is designed to enable secure user access to the Transport and Logistics Management System platform. It is of high priority as it forms the foundation for personalized user interactions.

4.1.2 Stimulus/Response Sequences

- **Stimulus:** User clicks on the "Sign Up" button.
  Response: System prompts the user to provide valid details and password for account creation. Users select the type of account they want to register as (Administrator, Driver, or Customer) and fill in appropriate form fields which include:

    – Name

    – Contact Information (Phone, Email)

    – Address

    – Role-specific information:

        * For Drivers: License Number, Experience
        * For Customers: Company Name (if applicable)

Appropriate User IDs will be auto-generated and allotted to the user.

- **Stimulus:** User enters login credentials.
  Response: System verifies credentials; successful login redirects the user to their respective dashboards based on their roles.

4.1.3 Functional Requirements

- **REQ-1:** Users can only register using valid information in the form fields.

- **REQ-2:** Passwords must be securely hashed and stored.

- **REQ-3:** Successful login redirects users to role-specific dashboards.

- **REQ-4:** Appropriate prompts and error messages will be provided while filling in registration forms as well as on the login page.

4.2 Shipment Request Placement

4.2.1 Description and Priority

The Shipment Request Placement feature allows Customers to seamlessly initiate new shipment requests. This feature holds high priority due to its central role in facilitating user interactions.

4.2.2 Stimulus/Response Sequences

- **Stimulus:** Customer clicks "Place Shipment Request."
  Response: System prompts the user to input shipment details such as sender, receiver, package details, weight, dimensions, and desired delivery timeframe.

- **Stimulus:** Customer submits shipment request.
  Response: System acknowledges the request and notifies Administrators for processing. The request is stored in the database with a status of "Pending."

- **Stimulus:** Administrator assigns a driver and vehicle to the shipment.
  Response: System updates the shipment status to "Assigned" and notifies both the Customer and the assigned Driver.

4.2.3 Functional Requirements

- **REQ-5:** Customers provide complete shipment details including sender, receiver, package information, and desired delivery timeframe.

- **REQ-6:** The system stores the shipment request and assigns a unique tracking ID.

- **REQ-7:** Administrators can view all pending shipment requests and assign appropriate resources.

- **REQ-8:** Customers receive confirmation and tracking information once their request is processed.

4.3 Waypoint Tracking

4.3.1 Description and Priority

The Waypoint Tracking feature enables drivers to update shipment status at predetermined way-points and allows customers to track their shipments in real-time. This feature is classified as high priority due to its importance in providing transparency.

4.3.2 Stimulus/Response Sequences

- **Stimulus:** Driver reaches a waypoint and logs into the system.
  Response: System displays the driver's assigned shipments and allows them to select the current shipment.

- **Stimulus:** Driver updates the shipment status at the current waypoint.
  Response: System records the status update, timestamps it, and sends a notification to the customer.

- **Stimulus:** Customer checks shipment status.
  Response: System displays the current location and status of the shipment, including all waypoints reached.

4.3.3 Functional Requirements

- **REQ-9:** The system defines a sequence of waypoints for each shipment.

- **REQ-10:** Drivers can update shipment status at each waypoint.

- **REQ-11:** The system timestamps each status update and maintains a history.

- **REQ-12:** Customers can view real-time tracking information showing the current location and status of their shipments.

- **REQ-13:** Automated notifications are sent to customers when their shipment reaches each waypoint.

4.4 Vehicle and Driver Management

4.4.1 Description and Priority

The Vehicle and Driver Management feature provides administrators with tools to manage the fleet of vehicles and pool of drivers. This feature holds medium priority for maintaining system operations.

4.4.2 Stimulus/Response Sequences

- **Stimulus:** Administrator navigates to vehicle management.
  Response: System displays a list of all vehicles with their details and status.

- **Stimulus:** Administrator adds a new vehicle or updates existing vehicle information.
  Response: System validates and stores the vehicle information.

- **Stimulus:** Administrator navigates to driver management.
  Response: System displays a list of all drivers with their details, qualifications, and current assignments.

- **Stimulus:** Administrator assigns a driver to a vehicle or shipment.
  Response: System updates the driver's status and notifies them of the assignment.

4.4.3 Functional Requirements

- **REQ-14:** Administrators can add, edit, and remove vehicles from the system.

- **REQ-15:** Vehicle information includes type, capacity, license plate, maintenance schedule, and current status.

- **REQ-16:** Administrators can add, edit, and manage driver profiles.

- **REQ-17:** Driver information includes name, contact details, license information, qualifications, and availability status.

- **REQ-18:** The system prevents assignment of unavailable drivers or vehicles to new shipments.

4.5 Notification System

4.5.1 Description and Priority

The Notification System feature enables automated notifications to customers about shipment status updates. This feature is classified as medium priority.

4.5.2 Stimulus/Response Sequences

- **Stimulus:** Shipment status is updated at a waypoint.
  Response: System generates and sends a notification to the customer with details about the current status and location.

- **Stimulus:** Shipment is delivered to its final destination.
  Response: System generates a delivery confirmation notification and sends it to the customer.

- **Stimulus:** Customer views notification history.
  Response: System displays a chronological list of all notifications for the customer's shipments.

4.5.3 Functional Requirements

- **REQ-19:** The system automatically generates notifications when shipment status changes.

- **REQ-20:** Notifications include relevant information such as shipment ID, current location, status update, and timestamp.

- **REQ-21:** Customers can view their notification history.

- **REQ-22:** Customers can set preferences for notification delivery methods (e.g., email, in-app).

5 Other Nonfunctional Requirements

5.1 Performance Requirements

- **Response Time:** The website's endpoints must respond to requests within a short duration on average, around 300 ms under normal operating conditions. Quick response times are required for maintaining interactivity in the web application, so that users have a smooth experience with the platform.

- **Scalability:** The system must be able to handle increasing numbers of shipments, users, and transactions as the business grows.

- **Reliability:** The system should be available 99.9% of the time, with planned maintenance periods announced in advance.

- **Error Handling:** The platform shall handle errors smoothly to minimize disruptions. Poor handling of errors causes inconveniences while using the application and may even lead to security issues.

5.2 Safety Requirements

- **Actions Prevented:** Unauthorized access to user data, data breaches, unauthorized sharing of user information with third parties.

- **Data Integrity:** The system must ensure that shipment data, especially waypoint tracking information, is accurate and cannot be tampered with.

- **Data Encryption:** To protect sensitive user and business data during transmission, all data must be encrypted.

5.3 Security Requirements

- **Secure Communication:** Use secure communication protocols such as HTTPS to encrypt data transmitted between the app and its servers. This helps prevent eavesdropping and man-in-the-middle attacks.

- **User Authentication:** Implement secure user authentication mechanisms to ensure that only authorized users can access the app.

- **Role-Based Access Control:** Different user roles (Administrator, Driver, Customer) should have access only to the functionality and data relevant to their role.

- **Prevent Cyber Attacks:** Use strict measures to prevent unauthorized access using attacks like SQL Injection, XSS Attack etc.

5.4   Software Quality Attributes

- **Usability:** Users should be able to navigate the website easily and perform tasks intuitively. This could be measured by the average time it takes for a user to complete common tasks like creating a shipment request, checking status etc.

- **Security:** Users' personal data should be protected from unauthorized access or breaches.

- **Reliability:** The system should be able to operate consistently without failure under normal conditions.

- **Maintainability:** The codebase should be well-organized and documented, making it easier for developers to make updates or fix bugs.

5.5   Business Rules

- **Shipment Placement:**
  Rule: Only registered users (Customers) can place shipment requests.
  Implication: User authentication and authorization mechanisms must be in place.

- **Driver Assignment:**
  Rule: A driver can only be assigned to one active shipment at a time.
  Implication: The system must track driver availability and prevent over-assignment.

- **Waypoint Updates:**
  Rule: Only the assigned driver can update the status of a shipment at waypoints.
  Implication: The system must authenticate drivers before allowing status updates.

- **Delivery Confirmation:**
  Rule: A shipment is considered delivered only when the driver confirms delivery at the final waypoint.
  Implication: The system must track the complete journey of a shipment through all waypoints.