**Project Proposal for SCD Lab**

# **Software Requirements Specification (SRS) for Transportation Reservation System**

# **Members:**

Muhammad Ammad BSSE ()

Hamza Gul Khan BSSE (121)

Ahmed Asad Butt BSSE ()

**Version:** 1.0

**Date:** [Date]

# **Document Control**

| **Version** | **Description** | **Author** | **Date** |
| --- | --- | --- | --- |
| **1.0** | **Initial Draft** | **Muhammad Ammad**  **Hamza Gul Khan**  **Ahmed Asas Butt** | **[Date]** |

**1. Introduction**

1.1 **Purpose**

The purpose of this document is to outline the software requirements for the development of a web-based Transportation Reservation System using the MERN (MongoDB, Express.js, React, Node.js) stack. This document defines the scope, features, and task division among team members for the project.

1.2 **Scope**

The Transportation Reservation System will provide users with the ability to:

* Reserve local transportation.
* Book vehicles for specific purposes.
* Reserve and purchase airplane tickets.
* Track live vehicle locations.

**2. Project Overview**

2.1 **Project Description**

The Transportation Reservation System aims to simplify the process of reserving and booking various means of transportation, including local transportation, vehicles, and airplane tickets. Additionally, users can track the live location of vehicles.

2.2 **Key Features**

The key features of the system include:

* User registration and authentication.
* Reservation and booking of transportation services.
* Online payment for services.
* Real-time vehicle tracking.
* User profiles and history tracking.

**3. Team and Task Division**

3.1 **Team Members**

* [Member 1 Name]: Front-end Developer
* [Member 2 Name]: Back-end Developer
* [Member 3 Name]: Database Administrator

3.2 **Task Division**

The project tasks are divided among team members as follows:

* [Member 1 Name]:
  + Develop the user interface (UI) using React.
  + Implement the front-end design.
  + Implement real-time vehicle tracking features.
* [Member 2 Name]:
  + Develop the server-side application using Node.js and Express.js.
  + Implement the reservation and booking logic.
  + Integrate payment processing.
* [Member 3 Name]:
  + Design and maintain the database using MongoDB.
  + Ensure data consistency and performance.
  + Implement user authentication and authorization.

**4. System Requirements**

4.1 **Hardware Requirements**

* Specify hardware requirements for server hosting, database storage, and client devices.

4.2 **Software Requirements**

* List the required software components, including MERN stack technologies and any third-party libraries.

**5. Functional Requirements**

5.1 **User Registration and Authentication**

* Users should be able to register and log in securely.
* User data should be stored securely.

5.2 **Local Transportation Reservation**

* Users should be able to reserve local transportation services.
* Confirmation details should be provided to users.

5.3 **Vehicle Booking**

* Users should be able to book vehicles for specific purposes.
* Vehicle availability and booking status should be updated in real-time.

5.4 **Airplane Ticket Reservation**

* Users should be able to search for and reserve/purchase airplane tickets.
* Booking confirmation and e-tickets should be generated.

5.5 **Live Vehicle Tracking**

* Users should be able to track the live location of vehicles.
* Real-time updates on vehicle positions should be provided.

**6. Non-Functional Requirements**

6.1 **Performance**

* The system should handle a large number of concurrent users.
* Response times should be minimal.

6.2 **Security**

* User data should be encrypted and stored securely.
* Payment processing should adhere to industry security standards.

6.3 **User Experience**

* The user interface should be intuitive and user-friendly.
* The system should be accessible on various devices and browsers.

**7. Design Patterns**

The following design patterns will be implemented in the project:

7.1 **MVC (Model-View-Controller) Pattern**

* The MVC pattern will be used to separate the application's concerns, enhancing maintainability and scalability.

7.2 **Factory Method Pattern**

* The Factory Method pattern will be employed for creating transportation reservation objects based on user requests.

7.3 **Observer Pattern**

* The Observer pattern will be used for real-time updates and notifications in the live vehicle tracking feature.

**8. Constraints**

* Specify any constraints, such as project timeline, budget, or external dependencies.

**9. References**

* List any references or resources used in the preparation of this SRS.

**10. Appendices**

* Include any additional information, diagrams, or mockups supporting the SRS.

This SRS document outlines the software requirements for the Transportation Reservation System project, including task division among team members and the implementation of three design patterns. It serves as a blueprint for the project's development and ensures that all stakeholders have a clear understanding of the project's scope and objectives.