

SOFTWARE REQUIREMENT FOR VEHICLE MANAGEMENT

NAME	SANJAY G
ROLL NUMBER	7376221CS287
SEAT NUMBER	233
PROJECT ID	13
PROJECT TITLE	VEHICLE MANAGEMENT

Tech-Stack:

- *Frontend*: REACT
- *Backend*: EXPRESS , NODE JS
- *Database*: MONGODB
- *Authentication*: Google Authentication

1. Introduction

1.1 Purpose

This document details the requirements for a Vehicle Booking and Management System, designed to facilitate vehicle booking by staff and manage these bookings through an administrative interface.

1.2 Scope

The system allows staff to book vehicles and administrators to manage these bookings. It includes functionalities for user authentication, booking applications, approval/rejection of bookings, and data storage.

2. Overall Description

2.1 Product Perspective

The system is a standalone application interfacing with a central database, containing modules for staff booking and admin management.

2.2 Product Functions

- Staff login and booking application
- Admin login, booking management, and report generation
- Data storage and retrieval

2.3 User Characteristics

Primary users are staff members booking vehicles and administrators managing these bookings.

3. Functional Requirements

3.1 Staff Vehicle Booking

3.1.1 Login

- **Description:** Staff log in with credentials.
- **Outputs:** Access granted or error message.

3.1.2 Apply Booking

- **Inputs:** Staff ID, purpose, city, car type, member count, going and return dates.
- **Outputs:** Booking confirmation or error message.

3.2 Admin Vehicle Report

3.2.1 Login

- **Description:** Admin log in with credentials.
- **Outputs:** Access granted or error message.

3.2.2 View and Manage Bookings

- **Inputs:** Booking ID, approval status, rejection reason.
- **Outputs:** Update confirmation or error message.

4. Non-Functional Requirements

4.1 Performance

- Efficient handling of concurrent logins and bookings.

4.2 Security

- Secure storage and transmission of credentials.
- Restricted access to authorized users.

4.3 Usability

- Intuitive interface for staff and administrators.

4.4 Reliability

- High availability during working hours.

5. System Architecture

5.1 Components

- **Database:** Stores user credentials, booking, vehicle, and driver details.
- **Login Module:** Handles authentication.
- **Booking Module:** Manages bookings.
- **Admin Module:** Manages approvals and reports.

5.2 Data Flow

1. User (staff/admin) logs in.
2. System verifies credentials.
3. Users access respective interfaces (booking or management).
4. Booking data is submitted and validated.
5. Admins view, approve, or reject bookings.
6. Database updates and confirms actions.

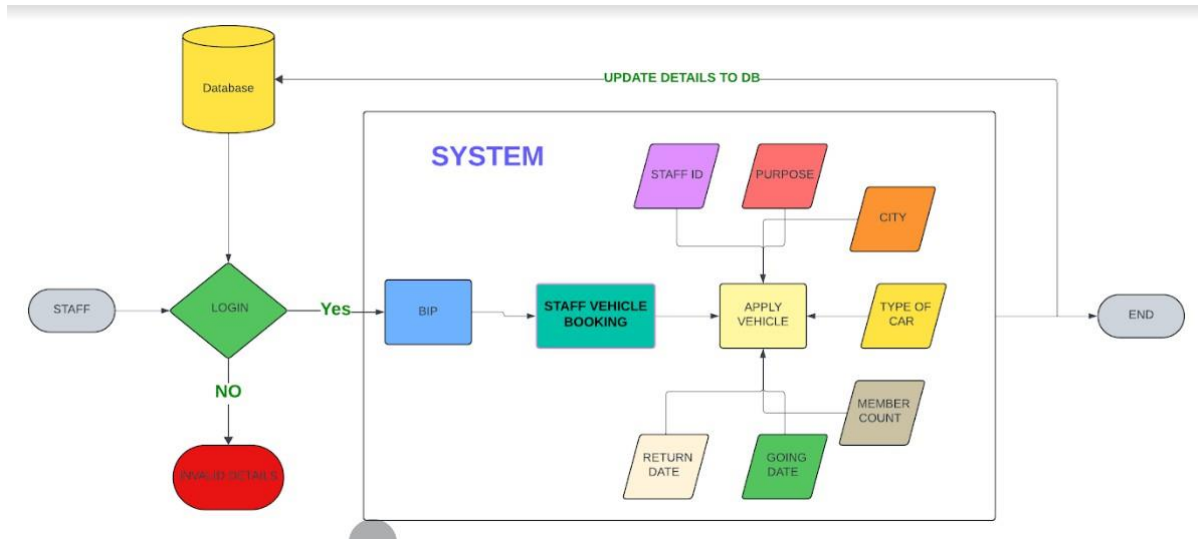
6. Appendix

6.1 Assumptions

- Unique IDs and valid credentials for all users.
- Central database dependency for data storage and retrieval.

FLOW CHART:

STAFF LOGIN:



ADMIN LOGIN:

