INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 03, March 2025, pp : 535-545

www.ijprems.com
editor@ijprems.com

e-ISSN : 2583-1062
Impact Factor : 7.001

TRAVEL AND TOURISM SYSTEM

Mrs. Samiksha Omkar Kamat[1], Ms. Snehal Shital Nakate[2], Ms. Pratiksha Surgonda Patil[3], Ms. Nikita Babaso Pujari[4]

[1]Guide, Sharad Institute Of Technology Polytechnic Yadrav

[2,3,4]Student, Sharad Institute Of Technology Polytechnic Yadrav

mantriswapnali25@gmail.com

ABSTRACT

A Travel and Tourism Management System (TTMS) is a software application designed to streamline and manage the various aspects of travel and tourism businesses. This system can be used by travel agencies, tour operators, hotels, transportation services, and customers to efficiently handle the end-to-end process of planning and booking travel services. Here's an overview of the key components and functions of the system. Enables customers to create and manage their profiles, including personal details, preferences, and travel history. Allows customers to view their past bookings and associated details.

1. INTRODUCTION

It describes the overall system design, focusing on how the modules interact with each other. Include details on the database schema, API integration, and the app's front-end and back-end layers. The system's architecture should prioritize a user-friendly interface, offering seamless navigation and quick access to necessary services. It should integrate third-party APIs for real-time data on flight schedules, hotel availability, and car rentals. The backend architecture may consist of a relational database to store user data, booking information, and payment history. Advanced features could include user reviews, recommendations based on previous bookings, multi-language support, and customer service chatbots. Security protocols must ensure the protection of sensitive data, while scalability is essential to accommodate increasing users and transactions. The system should also include reporting and analytics features for administrators to track booking trends, manage inventory.

2. EXISTING SYSTEM

The existing system in travel and tourism management refers to the combination of traditional and modern practices currently in use to manage travel services, bookings, customer experiences, and overall tourism activities. This system varies significantly based on the size of the organization, technological advancements, and the type of tourism sector involved (e.g., leisure, business, eco-tourism). Manual systems are still prevalent in small- scale operations and rural areas where technology adoption is low. Digital platforms have revolutionized how travelers book their trips, offering convenience, transparency, and global reach. GDS platforms are large-scale systems used by travel agencies and service providers to manage inventory and bookings. With the rise of smartphones, mobile apps and websites have become a cornerstone for modern tourism services. The pandemic has significantly impacted the travel and tourism industry, revealing limitations in the existing systems. The travel and tourism industry relies heavily on advanced systems and technologies to streamline operations and enhance customer experiences. Online booking systems are at the forefront, enabling travelers to easily reserve flights, hotels, car rentals, and tour packages.

2.1 Project overview

Travel And Tourism System is a complete tourist fully integrated tourism application. The application covers all the areas required for an including tourism, This project is developed to manage the tourist in the tourism management application. The main module in this project are login, tourist management, complaints and reports. The objective of the project is to develop a system that automates the processes and activities of a travel and. The purpose is to design a system using which one can perform all operations related to traveling. Travel and Tourism management system is used to book a tour from anywhere in the world by a single dynamic website which will help the user to know all about the places and tour details in our app. The admin can add packages to the application from a certain travel agents and hotels through create a tour page. Then the users can sign in and book each package, then it can be confirmed by the admin in their manage booking page.

2.2 Objective & Scope

1.Streamlining the booking process:-

Streamlining the booking process involves simplifying and optimizing how users search for, compare, and reserve travel services. This can include features like a user-friendly interface, fast search and filter options, integrated payment gateways, automated confirmations, and personalized recommendations to make the experience seamless and efficient.

2.Providing personalized travel recommendations:-

Providing personalized travel recommendations involves using user data like preferences, past trips, budget, and interests to suggest destinations, activities, and itineraries tailored to their needs. AI and machine learning algorithms can enhance this by analyzing patterns and offering relevant options to improve the travel experience.

3.Real-time data integration:-

Real-time data integration involves connecting the system to external sources like flight, hotel, weather, and traffic APIs to provide users with up-to-date information. This ensures availability, pricing, and conditions are accurate, allowing for dynamic updates and better decision-making for users.

4.Design a user-friendly interface for easy navigation:-

Designing a user-friendly interface means creating a simple, intuitive layout with clear navigation, easy-to-read text, and accessible buttons. It should allow users to quickly find what they're looking for, with minimal effort, and provide a smooth experience across devices by being responsive and organized.

5.Admin and reporting functionality:-

Admin and reporting functionality allows administrators to manage user accounts, bookings, and services efficiently. It includes generating reports on sales, user activity, and system performance to help make data-driven decisions and improve the system's operations.

6.Supporting multi-platform accessibility:-

Supporting multi-platform accessibilitymeans ensuring the travel system is compatible across various devices, such as desktops, smartphones, and tablets. This allows users to access and use the system seamlessly, regardless of the platform they're on, providing flexibility and convenience.

Applying Software Engineering Approach

1. Requirement Analysis:

Functional Requirements:-

- Search Engine: A powerful and fast search tool to explore travel options.
- User Interface (UI): Clean and intuitive UI for both customers and admins.
- Notification System: Send updates to users about booking confirmations, reminders, and promotions.

Non-Functional Requirements:-

- Performance: The system should be fast, with quick loading times.
- Reliability: The system should be available and operational with minimal downtime.
- Usability: Ensure ease of use for both end-users and admins.

2. System Design:

- **Client-Side (Mobile App)**: -The mobile app built with Cordova integrates HTML, CSS, and JavaScript for a seamless interface. The app is divided into three distinct login modules (Admin, Driver, Customer) and communicates with the server using REST APIs.
- **Server-Side (Back-end)**:- PHP and MySQL form the back-end. PHP scripts handle the user authentication and business logic. The MySQL database stores user data, bus routes, schedules, and bookings.
- **Database Layer**:- MySQL is used to store all the data, including user credentials, buses, routes, schedules, and bookings. Relationships are designed between tables such as Users, Buses, Routes, and Bookings.

3. Algorithm Development:

Login Algorithm:- Checks user credentials and redirects to the appropriate dashboard**.**

Route Allocation:- Matches buses to routes based on availability and schedule**.**

4. Integration and Testing:

- **Data Integration:-** Link user data (preferences, booking history) with the personalized recommendation engine. Integrate real-time data sources for dynamic pricing, availability, and updates on services.
- **Multi-Platform Integration:-** Ensure seamless synchronization between web and mobile platforms (bookings).Test that features like booking, profile management, and payment are consistent across devices.
- **Admin Dashboard Integration:-** Integrate admin functionalities such as booking management, reporting, and user management into the dashboard. Ensure admins can access real-time data and analytics with correct permissions**.**

- **Functional Testing:-** Verify that the system's features and functions (search, booking, payment, recommendations) work as per the requirements. Test the user flow to ensure the booking process is smooth and intuitive.
- **Usability Testing:-** Evaluate the system from the user's perspective to check if it is easy to navigate. Test on different screen sizes and devices to ensure the interface adapts well (responsive design).

5. Deployment and Maintenance:

- **Continuous Integration/Continuous Deployment (CI/CD):-** Use CI/CD pipelines to automate the deployment process, ensuring that updates and patches can be quickly rolled out without downtime.
- **Cloud Deployment:-** Deploy the system on a cloud platform (e.g., AWS, Azure, Google Cloud) to ensure scalability and reliability.
- **DNS Setup:-** Configure domain names and set up web hosting for the application.
- **SSL Certificate:-** Implement SSL encryption for secure communication.
- **Feature Enhancements:-** Periodically add new features (e.g., new destinations, services, or payment methods) to keep the system competitive.
- **Bug Fixes:-** Continuously monitor and resolve any bugs or issues reported by users or found in monitoring tools.

3. LITERATURE SURVEY

3.1 Literature Review

The literature on travel and tourism management systems highlights the significant role of technology in transforming the industry. Studies emphasize the adoption of Global Distribution Systems (GDS), such as Amadeus and Sabre, which revolutionized how travel agencies access real-time inventory and pricing. Researchers have also explored the growing reliance on online booking platforms, like Booking.com and Expedia, which offer users convenience, transparency, and comparative options. Advances in artificial intelligence (AI**)** and machine learning have been widely studied for their potential to personalize travel experiences, such as AI chatbots that provide 24/7 customer support and predictive algorithms for dynamic pricing. Scholars have investigated the integration of mobile applications to support travelers with itinerary planning, offline maps, and instant notifications. The literature also reflects the increasing emphasis on sustainable tourism, with systems incorporating eco-friendly practices, such as carbon footprint tracking and promotion of green travel options. Furthermore, studies underline the importance of customer relationship management (CRM) systems.

4. REQUIREMENT SPECIFICATION

4.1 Software Requirements

The software requirements for our project, "Travel And Tourism System," android for run our application, cordova which is basically used for communication. We utilize Visual Studio Code as our primary integrated development environment (IDE), providing a robust and customizable environment for coding and debugging. Additionally, we leverage Node.js and npm (Node Package Manager) for backend development, enabling efficient package management and server-side scripting.

For testing and compatibility purposes, we ensure our web application is compatible with major browsers such as Google Chrome (version 13 and above), Firefox, Safari, and Microsoft Edge. Chrome. we use CSS for many button and styles.we also used Javascript .And for backend purpose we use PHP, MySQL. And XAMP serever for local host.

4.2 Hardware Requirements

The hardware requirements for our project, "Travel And Tourism System" primarily revolve around a standard computer system capable of handling development tasks and running web applications smoothly. We recommend the following hardware specifications:

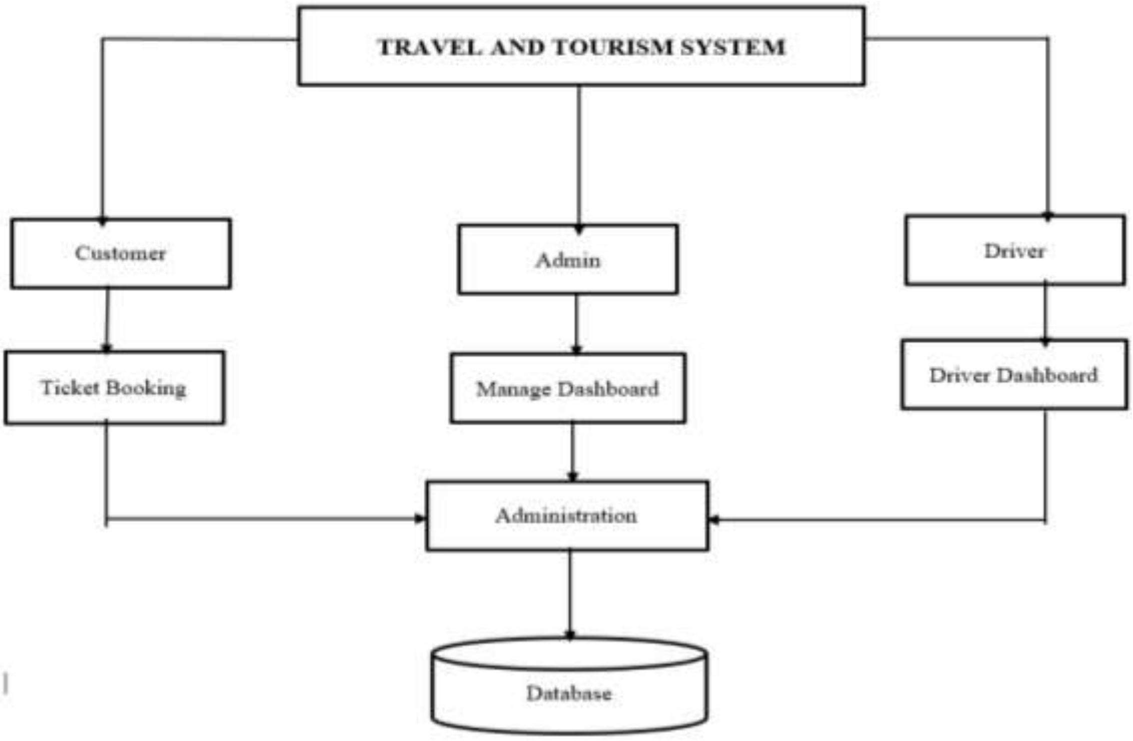
1. **Processor**:- A multicore processor (such as Intel Core i5 or AMD Ryzen 5) with sufficient processing power to handle AI algorithms, data processing, and multitasking efficiently.
2. **Memory** (**RAM**):- At least 8GB of RAM to ensure smooth operation of development tools, IDEs, and web browsers while running AI algorithms and handling large datasets.
3. **Storage**:- Solid-state drive (SSD) with adequate storage capacity (256GB or more) for storing project files, datasets, development environments, and software tools.
4. **Graphics Card**:- A dedicated graphics card (GPU) with CUDA support (NVIDIA GeForce GTX or AMD Radeon series) may be beneficial for accelerating AI computations, particularly for image and video processing tasks.

Display:- A high-resolution display (Full HD or higher) with good color accuracy for visualizing

5. SYSTEM DESIGN

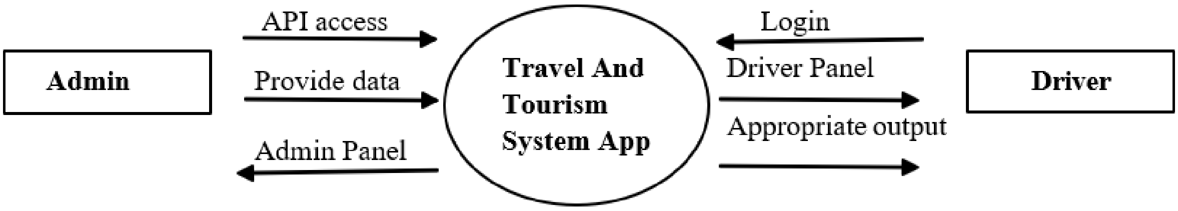
Architecture Design

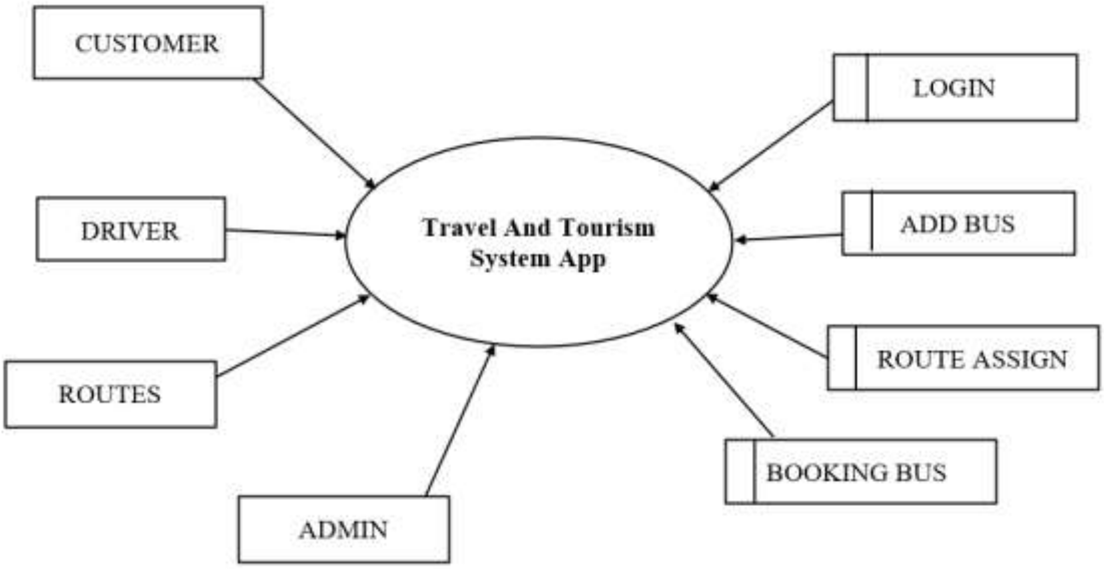
The architecture of the Tours & Travels app is structured to ensure modularity, security, and user-friendly interactions. The application follows a **Client-Server architecture** where the front-end interacts with the back-end using HTTP requests.



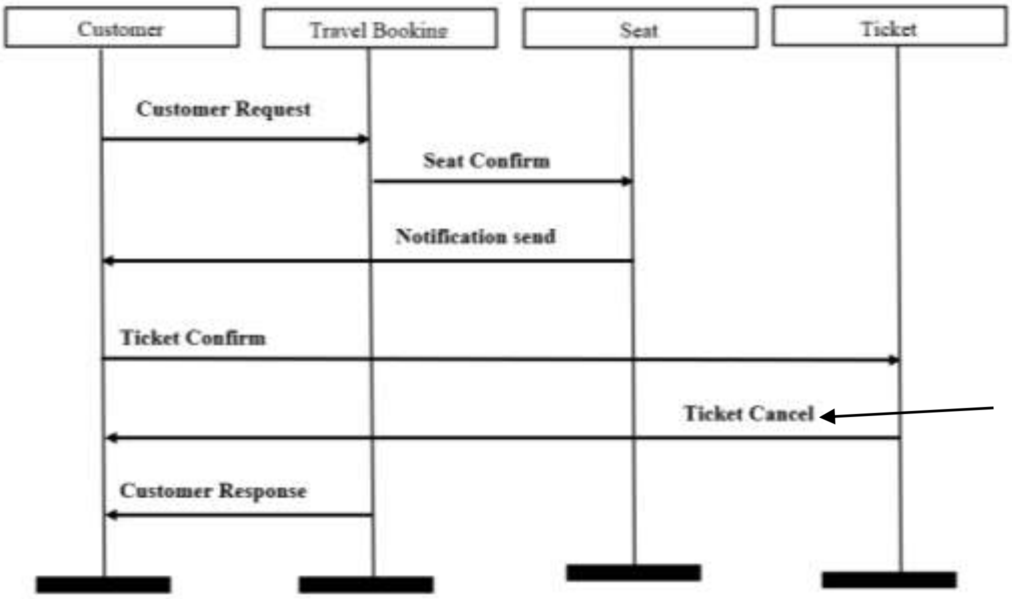
5.2 Data Flow Diagram:

1. Level 0 DFD

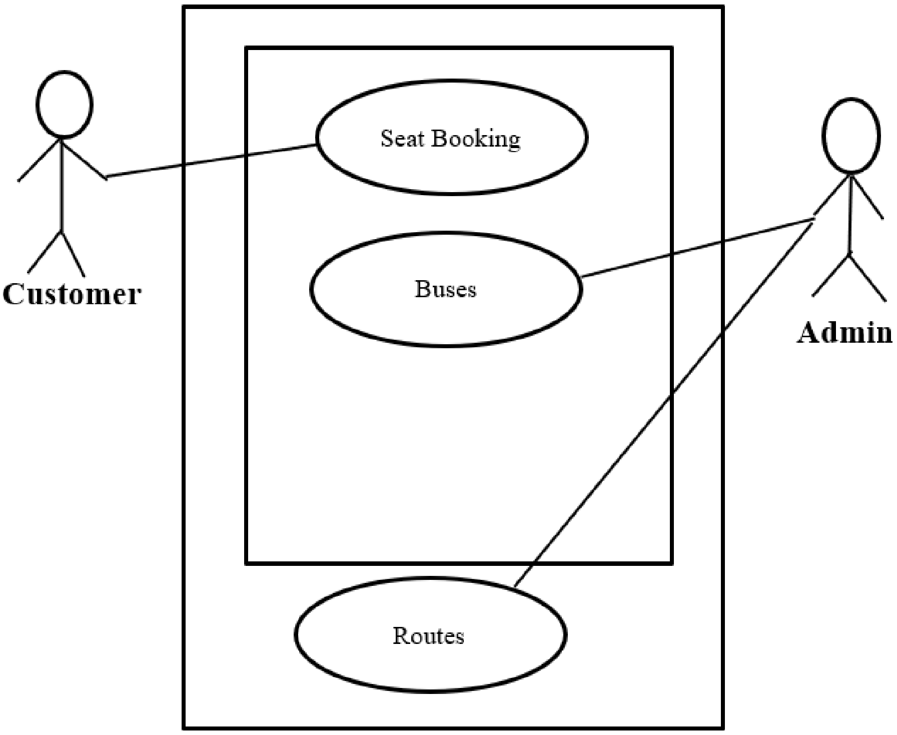




2. Level 1 DED



5.3 Use Case Diagram:-



6. ALGORITHM DESCRIPTION

6.1 Introduction to Login Algorithm

A login algorithm is a process designed to authenticate users and verify their identities before granting access to a system or application. It involves securely managing user credentials, typically a username and password, to ensure that only authorized individuals can access sensitive information or perform specific actions. The algorithm compares the user-provided credentials with stored data, ensuring proper security measures like encryption and hashing are implemented to protect against breaches. Modern login algorithms often incorporate additional security features like password hashing (using algorithms such as bcrypt or SHA-256), rate limiting to prevent brute force attacks, and two-factor authentication (2FA) for added protection. These algorithms may also manage session creation using tokens like JSON Web Tokens (JWT) to maintain secure user sessions. By balancing usability with robust security measures, login algorithms form the foundation of secure access in any digital system.

6.2 System Overview

Travel And Tourism System is a complete tourist fully integrated tourism application. The application covers all the areas required for an including tourism, This project is developed to manage the tourist in the tourism management application. The main module in this project are login, tourist management, complaints and reports. The objective of the project is to develop a system that automates the processes and activities of a travel and. The purpose is to design a system using which one can perform all operations related to traveling. Travel and Tourism management system is used to book a tour from anywhere in the world by a single dynamic website which will help the user to know all about the places and tour details in our app. The admin can add packages to the application from a certain travel agents and hotels through create a tour page. Then the users can sign in and book each package, then it can be confirmed by the admin in their manage booking page.

7. CODING

7.1 Packages

1**. Bootstrap 5**

Description- A popular, open-source CSS framework for developing responsive and mobile-first websites. It includes pre-built components like buttons, forms, navigation bars, and grid systems.

2.jQuery

Description: A fast, small, and feature-rich JavaScript library that simplifies things like HTML document traversal and manipulation, event handling, and AJAX interactions.

3.**Font Awesome**

Description: - A widely used icon library that provides scalable vector icons that can be customized with CSS.

4.**Custom JavaScript Files (index.js, pulltorefresh.js)**

Description:- Custom JavaScript files designed to handle additional user interactions or custom behaviors for the application.

7.2 Classes Methods

1.User Class

Description:- A class to manage user-related functionality such as authentication, registration, and data management for both customers and drivers.

Methods in this Class:

authenticate($mobile, $password):- Verifies a user's mobile number and password against the database to authenticate login.

register():- Handles the registration of new users (customer or driver).

getUserDetails():- Fetches user details based on user ID for profile management.

2. Database Class

Description:- A class to manage the database connection and query execution. It encapsulates the logic for connecting to the database and executing SQL queries.

Methods in this Class:

connect():- Establishes a connection to the database using MySQLite.

query($sql):- Executes SQL queries (like SELECT, INSERT, UPDATE) and returns the result.

close():- Closes the database connection.

3. Booking Class

Description:- This class handles functionality related to bookings, such as creating, updating, or retrieving travel bookings for customers and drivers.

Methods in this Class:

createBooking():- Creates a new booking for a customer, including destination, pickup, and drop-off details.

getBookingDetails():- Fetches details of a particular booking using the booking ID.

cancelBooking():- Cancels a customer's booking and updates the booking status in the database.

4. Driver Class

Description:- This class manages driver-specific operations such as managing available routes, vehicle details, and current trips.

Methods in this Class:

updateAvailability():- Updates a driver's availability status (whether they are available for a new booking).

getDriverDetails():- Retrieves the driver’s profile and other related details.

8. IMPLEMENTATION

8.1 Installation & Preparation

Visual Studio Code

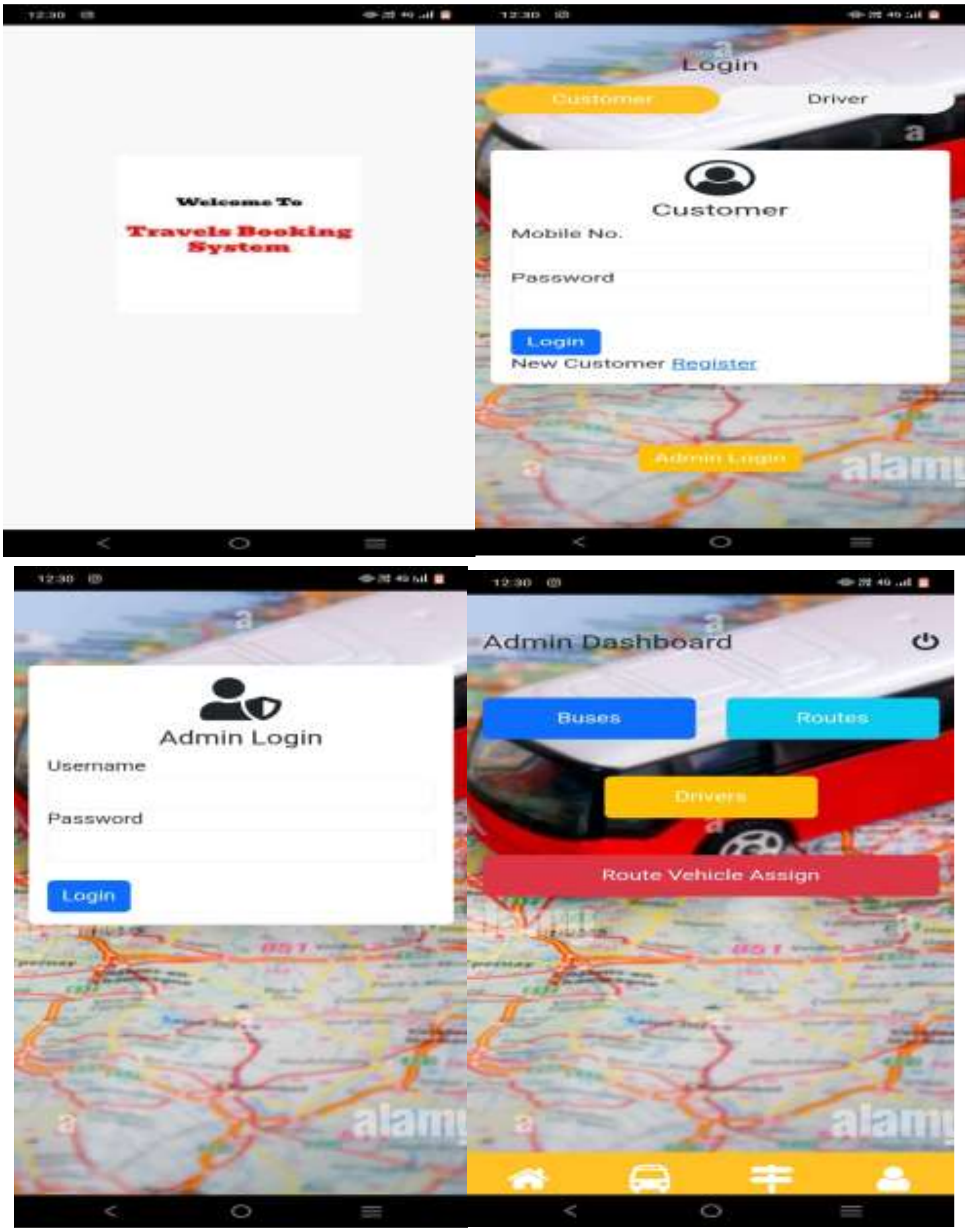
1. Download VS Code:- Open your web browser and go to the official Visual Studio Code website: [https://code.visualstudio.com/]() & Click on the "Download for Windows" button to download the installer.

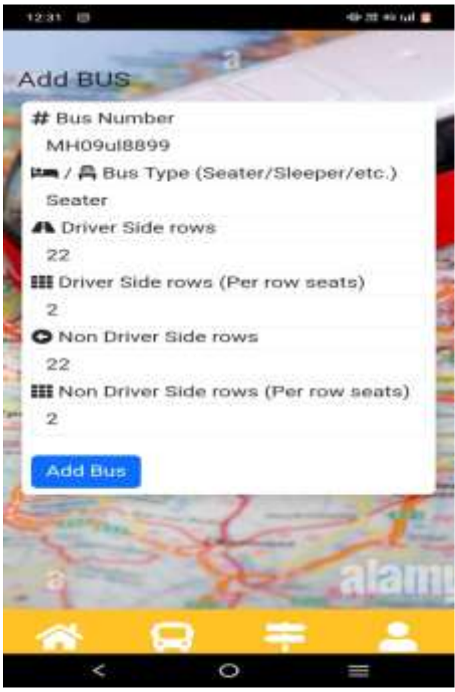
2. Run the Installer:- Once the download is complete, locate the downloaded installer file (usually named something like `VSCodeSetup.exe`) in your downloads folder or wherever you saved it. Double-click on the installer file to run it.

3. Install VS Code:- During installation, you might see a User Account Control (UAC) prompt; click "Yes" to continue. Follow on-screen instructions to complete the installation. Default installation location is recommended. Optionally, customize tasks like adding VS Code to PATH or creating desktop shortcuts as desired.

4. Launch VS Code:- To launch Visual Studio Code after installation, simply double-click its desktop shortcut or search for "Visual Studio Code" in the Start menu. Alternatively, right-click on a folder in File Explorer and select "Open with Code" to open it directly in VS Code.

8.2 Screen Shots Of The System





Add BUS

Bus Number
MH09ul8899

Bus Type (Seater/Sleeper/etc.)
Seater

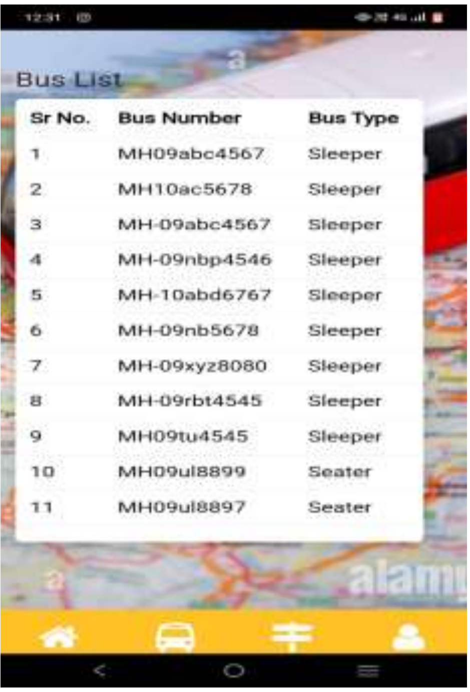
Driver Side rows
22

Driver Side rows (Per row seats)
2

Non Driver Side rows
22

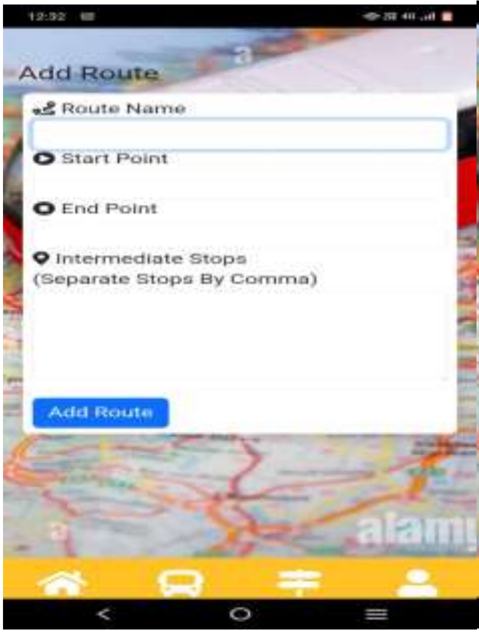
Non Driver Side rows (Per row seats)
2

Add Bus



Bus List

Sr No.	Bus Number	Bus Type
1	MH09abc4567	Sleeper
2	MH10ac5678	Sleeper
3	MH-09abc4567	Sleeper
4	MH-09nbp4546	Sleeper
5	MH-10abd6767	Sleeper
6	MH-09nb5678	Sleeper
7	MH-09xyz8080	Sleeper
8	MH-09rbt4545	Sleeper
9	MH09tu4545	Sleeper
10	MH09ul8899	Seater
11	MH09ul8897	Seater



Add Route

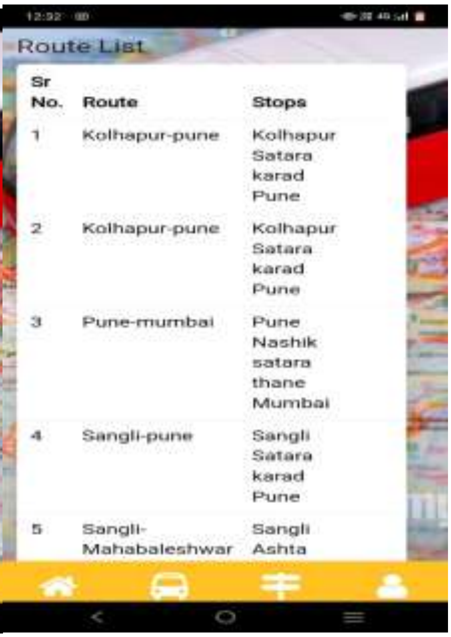
Route Name

Start Point

End Point

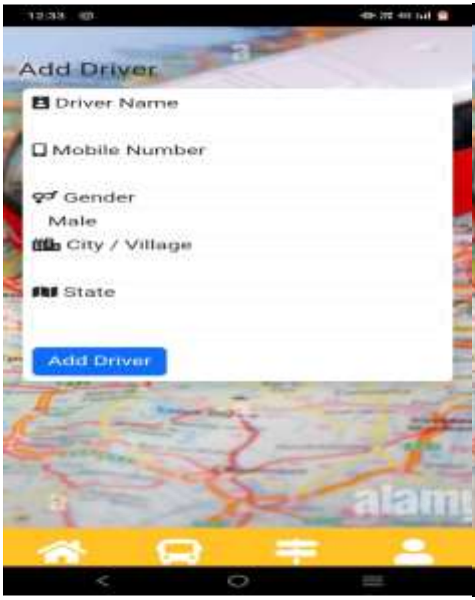
Intermediate Stops (Separate Stops By Comma)

Add Route



Route List

Sr No.	Route	Stops
1	Kolhapur-pune	Kolhapur Satara karad Pune
2	Kolhapur-pune	Kolhapur Satara karad Pune
3	Pune-mumbai	Pune Nashik satara thane Mumbai
4	Sangli-pune	Sangli Satara karad Pune
5	Sangli-Mahabaleshwar	Sangli Ashta



Add Driver

Driver Name

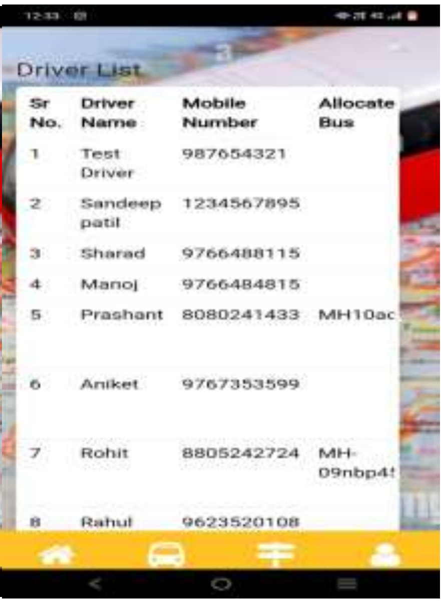
Mobile Number

Gender
Male

City / Village

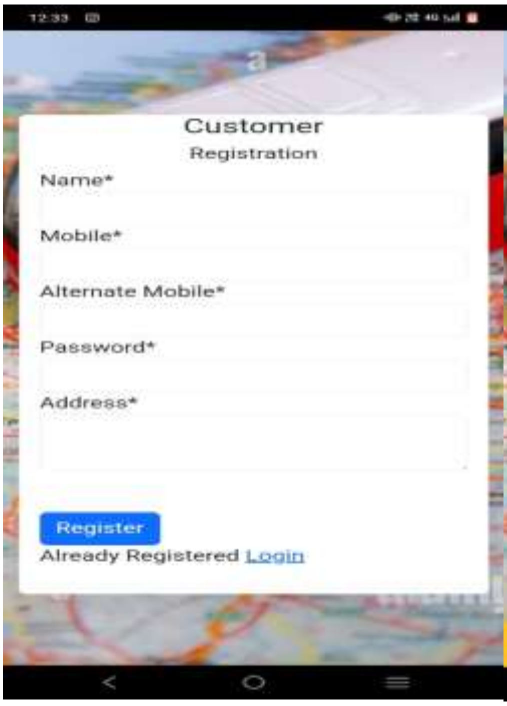
State

Add Driver



Driver List

Sr No.	Driver Name	Mobile Number	Allocate Bus
1	Test Driver	987654321	
2	Sandeep patil	1234567895	
3	Sharad	9766488115	
4	Manoj	9766484815	
5	Prashant	8080241433	MH10ac
6	Aniket	9767353599	
7	Rohit	8805242724	MH-09nbp4
8	Rahul	9623520108	



Customer
Registration

Name*

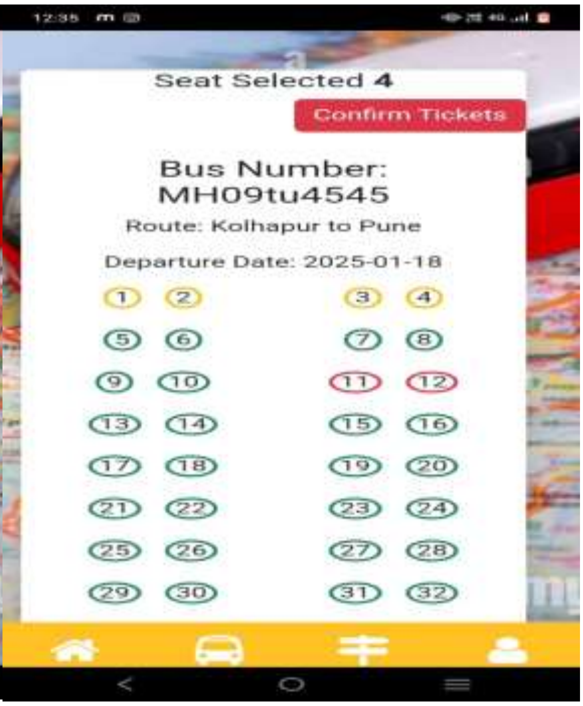
Mobile*

Alternate Mobile*

Password*

Address*

Already Registered Login

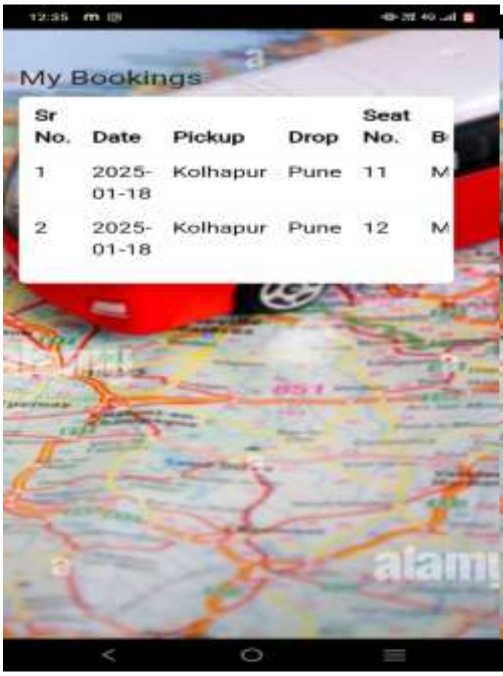


Seat Selected 4

Bus Number: MH09tu4545

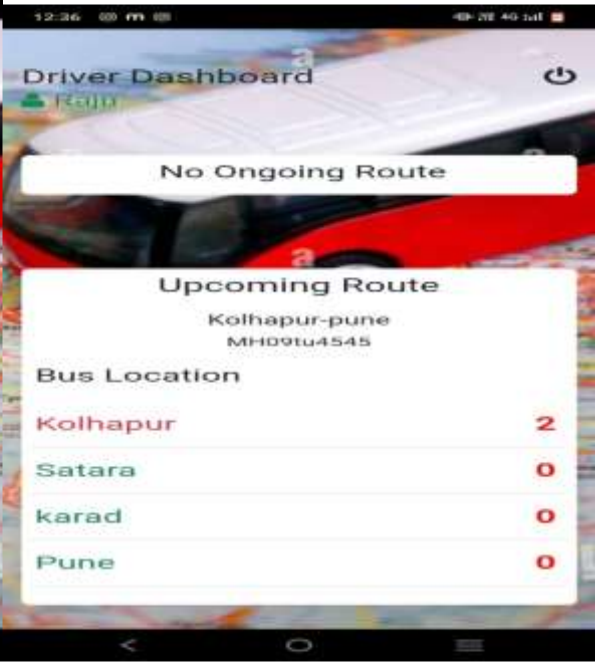
Route: Kolhapur to Pune

Departure Date: 2025-01-18



My Bookings

Sr No.	Date	Pickup	Drop	Seat No.	B
1	2025-01-18	Kolhapur	Pune	11	M
2	2025-01-18	Kolhapur	Pune	12	M



Driver Dashboard

Raju

No Ongoing Route

Upcoming Route

Kolhapur-pune
MH09tu4545

Bus Location

Kolhapur	2
Satara	0
karad	0
Pune	0

Test Cases

Test case Id	Test case Input	Expected result	Actual result	Pass/Fail
1.	Username	The username should be admin.	The entered username is admin.	Pass
2.	Password	The password should be admin.	The entered password is admin.	Pass
3.	Customer Login	The customer should be redirected to the customer dashboard page.	The customer redirected to the customer dashboard page (customer_dashboard.html).	Pass
4.	Driver Login	The driver should be redirected to the driver dashboard page.	The driver redirected to the driver dashboard page (driver_dashboard.html).	Pass
5.	Route	It should take the route data.	It take the route data.	Pass

Test Cases for Customer Login:

Test Case 1: Valid Customer Login

- **Test Case ID**:- TC_Customer_Login_01
- **Test Description**:- Verify that the customer can successfully log in with valid credentials.
- **Preconditions**:- The customer is registered in the system.
- **Test Steps**:-

1. Open the login page.
2. Select the "Customer" login option.
3. Enter a valid mobile number in the "Mobile No." field.
4. Enter the correct password in the "Password" field.
5. Click the "Login" button.

- **Expected Result**:- The customer should be redirected to the customer dashboard page (customer_dashboard.html).

Test Case 2: Invalid Mobile Number

- **Test Case ID**:- TC_Customer_Login_02
- **Test Description**:- Verify that the customer cannot log in with an invalid mobile number.
- **Preconditions**:- The customer is registered in the system.
- **Test Steps**:-

1. Open the login page.
2. Select the "Customer" login option.
3. Enter an invalid mobile number (not registered in the system).
4. Enter the correct password.
5. Click the "Login" button.

Test Cases for Driver Login:

Test Case 3: Valid Driver Login

- **Test Case ID**:- TC_Driver_Login_01
- **Test Description**:- Verify that the driver can successfully log in with valid credentials.
- **Preconditions**:- The driver is registered in the system.
- **Test Steps**:-

1. Open the login page.
2. Select the "Driver" login option.
3. Enter a valid mobile number in the "Mobile No." field.
4. (Leave the password field empty or enter as required).
5. Click the "Login" button.

- **Expected Result**:- The driver should be redirected to the driver dashboard page (driver_dashboard.html).
- **Test Case 4: Invalid Driver Mobile Number**
- **Test Case ID**:- TC_Driver_Login_02
- **Test Description**:- Verify that the driver cannot log in with an invalid mobile number.
- **Preconditions**:- The driver is registered in the system.
- **Test Steps**:-

1. Open the login page.
2. Select the "Driver" login option.
3. Enter an invalid mobile number (not registered in the system).
4. Leave the password field empty or enter a password.
5. Click the "Login" button.

- **Expected Result**:- An error message should appear, indicating "Invalid mobile number."

9. CONCLUSION

The **Travel and Tourism System** project, particularly focusing on the user login functionality, has been successfully developed and tested to ensure both customers and drivers can access their respective dashboards securely and efficiently. The system is designed with a user-friendly interface, making it easy for users to choose their login roles. Through rigorous testing, including unit testing, integration testing, and user acceptance testing (UAT), the login process has been validated for functionality and usability. The system ensures secure authentication for both customers and drivers, with role-based access control that ensures each user only accesses their designated dashboard. Additionally, the login page is responsive across various devices, enhancing user experience. While the project meets the primary requirements, there are areas for improvement, particularly in terms of security, such as implementing stronger password encryption and secure communication. The system also has potential for scalability improvements as the user base grows. Overall, the project lays a strong foundation for future development, and further enhancements will be made to improve security, performance, and user satisfaction as the system evolves.

10. REFERENCES

[1] (n.d.). World Wide Web Consortium (W3C). Retrieved from.

[2] (2023). Bootstrap. Retrieved from.

[3] (2023). JQuery. Retrieved from.

[4] Web Security for Developers" by Malcolm McDonald**.** (2021).

[5] Testing Computer Software" by Cem Kaner, Jack Falk, and Hung Q. Nguyen