Project Proposal: Travel and Tourism

Project Title: Travel and Tourism
Developers: M.C. Arun and D. Alfred Sam
Project Coordinator: Deepa Mam

---

1. Introduction

With the rise of technology in everyday life, making travel arrangements has become increasingly streamlined. The *Travel and Tourism* project aims to simplify booking transportation, accommodations, and tourist spots within a desktop-based application. This application uses a Swing interface with a straightforward, intuitive design to access core functionalities. It leverages MongoDB for storing all booking data and Spring Boot as the backend server, enabling efficient, secure, and scalable interactions with the database.

---

2. Project Overview

The *Travel and Tourism* project provides a desktop application with a minimalistic Swing interface, designed for users to book various travel services like buses, trains, hotels, and tourist places. With just four main buttons, users can insert, update, delete, and view travel information. Each button opens a new Swing window, where users perform actions related to bookings and data management, ensuring a clean and easy-to-use interface.

Key Objectives

- **Easy-to-Navigate Interface**: Four main buttons—Insert, Update, Delete, and View—simplify all user interactions.
- **Efficient Data Management**: MongoDB stores and organizes booking information.
- **Automated Processing**: Spring Boot automates backend processes to handle CRUD (Create, Read, Update, Delete) operations for bookings.

Swing Interface Design

The interface consists of a main window with four buttons, each leading to a new Swing window for specific actions. This approach keeps the interface uncluttered while providing all essential features:

1. **Insert** – Opens a new window to input booking data for buses, trains, hotels, or tourist spots.
2. **Update** – Opens a new window to modify existing booking details.
3. **Delete** – Opens a window to remove bookings.
4. **View** – Opens a window to display all existing bookings.

---

3. System Architecture

The system's backend architecture relies on Spring Boot and MongoDB, while the front end uses Swing. The MongoDB collections within the `DBMS` database store travel booking data, enabling efficient data retrieval and modification. The backend manages data transactions and provides endpoints for each CRUD operation.

The database is organized into the following collections:

1. **Available Buses**
2. **Available Trains**
3. **User Details**
4. **My Bookings**
5. **Available Hotels**
6. **Available Tourist Places**
7. **Receipts**

---

4. Features and Functionalities

4.1 User-Facing Features

Each button in the main Swing interface allows users to interact with the database in a dedicated window.

- **Insert Button**
  Opens a new window to input booking details such as transportation type, destination, date, and pricing. Users can choose the type of service (e.g., bus, train, hotel) to add relevant details.
- **Update Button**
  Opens a new window where users can modify booking information. A search feature allows users to locate specific records to edit.
- **Delete Button**
  Opens a window to remove specific bookings by selecting the type of service and entering the booking ID.
- **View Button**
  Opens a window that displays a list of all bookings with search options to filter results by service type, date, or user ID.

4.2 Backend Functionalities

- **Database Management**
  MongoDB collections store details for each type of service, enabling rapid retrieval, updates, and deletion of records.
- **Data Security**
  The application uses secure backend protocols to safeguard user details and booking data.

---

5. Database Schema Overview

The database schema is organized by collections, each representing a type of travel service or user information. Fields within each collection store specific attributes required for booking management.

1. **Available Buses**

    - Fields: Bus ID, Origin, Destination, Timing, Available Seats, Price.
2. **Available Trains**
    - Fields: Train ID, Origin, Destination, Timing, Available Seats, Price.
3. **User Details**
    - Fields: User ID, Name, Contact Information, Payment Details, Previous Bookings.
4. **My Bookings**
    - Fields: Booking ID, User ID, Bus/Train/Hotel/Tourist Place Details, Date of Booking, Status.
5. **Available Hotels**
    - Fields: Hotel ID, Location, Room Type, Available Rooms, Price, Amenities.
6. **Available Tourist Places**
    - Fields: Place ID, Location, Entry Fee, Opening Hours, Description.
7. **Receipts**
    - Fields for each type of receipt: Receipt ID, User ID, Booking ID, Amount Paid, Date, Service Details.

---

6. Technology Stack

- **Frontend**: Swing
  A Java Swing interface provides a simple, desktop-based user experience, reducing complexity by using four main buttons for all interactions.
- **Backend Framework**: Spring Boot
  Enables efficient management of database operations and interactions with MongoDB.
- **Database**: MongoDB
  Chosen for scalability, MongoDB allows the flexible storage and organization of booking data.

---

7. Workflow and Booking Process

1. **User Authentication**
   Users register and log in through a secure process, enabling them to access booking features.
2. **CRUD Operations via Swing Interface**
   - **Insert**: Users add new bookings.
   - **Update**: Users update booking details.
   - **Delete**: Users remove bookings.
   - **View**: Users view all bookings.
3. **Real-Time Data Sync**
   All CRUD operations are synchronized with MongoDB, ensuring real-time availability and accurate booking data.

---

8. Conclusion

The *Travel and Tourism* project provides a streamlined experience for booking travel services through a desktop-based application. By combining the simplicity of a Swing interface with MongoDB and Spring Boot, the application is both user-friendly and highly functional, capable of handling complex booking scenarios with ease.