## Web-Based Railway Ticketing system

# RailYatri Project


### Description

The **RailYatri Project** aims to provide a seamless experience for railway travelers. This platform integrates features such as ticket booking, train search, PNR status checking, and curated tourism packages, all while maintaining a user-friendly interface, leveraging technology to make train journeys smoother and more accessible.

### Features

- **Ticket Booking**: Easily book train tickets directly through the platform.
- **Authentication System**: Authentication system implemented using firebase
- **Search Trains**: Find trains between given stations for any date.
- **PNR Status Check**: Check your ticket's PNR status and updates in real-time.
- **Tourism Packages**: Explore curated travel packages designed for tourists, including accommodation options.
- **Feedback System**: An easy way for users to share feedback, helping us improve their experience.
- **Contact Us:** Connect with us for support, inquiries, or assistance anytime.

### Setup Instructions

# 1. Setup through Direct Link (Website Hosted on InfinityFree)

This project is already hosted on **InfinityFree**, follow these steps to access it:

1. Open a browser and go to the URL where your site is hosted( `http://railyatri.infinityfreeapp.com`).

   **(Make sure that the protocol is http and not https)**

2. All the necessary setup should already be done on the server side. The website will be live and accessible via this link.

# 2. Setup via GitHub Repository (Clone the Repository)

**Prerequisites:**

Before starting, ensure you have the following installed:

- **PHP** (version 7.4 or above)
- **MySQL**(through php myadmin)
- **Apache** or **Nginx** (for local server setup)
- **Code Editor** (VSCode, Sublime Text, etc.)
- **XAMPP** or **MAMP** (optional for easy local setup)

## 1. Clone the Repository

1. Clone the project repository to your local machine in the htdocs folder inside xampp or mamp:

   git clone https://github.com/arpitkumar0007/CSP_Project_RailwayTicketBooking.git

2. Navigate to Project Directory

   cd CSP_Project_RailwayTicketBooking

### 2. Set Up Apache and MySQL

If you're using **XAMPP** or **MAMP**, follow these steps:

- Install **XAMPP** (which includes Apache, MySQL, and PHP).
- Start **Apache** and **MySQL** servers from the XAMPP/MAMP control panel.

If not using **XAMPP** or **MAMP**, manually install **Apache** and **MySQL** and configure them to run on your local machine.

### 3. Configure Database

- Open **phpMyAdmin** (usually available at `http://localhost/phpmyadmin/`), and create a new database for the project (`railway_db`).
- Import the SQL file provided in the repo to create tables and populate the database.
- Run this command to populate the database

  mysql -u root -p railway_db < railway_db.sql

### 4. Run the Application

- Place your project files in the **htdocs** (XAMPP) or **www** (MAMP) directory.
- If using Apache, open your browser and navigate to `http://localhost/CSP_Project_RailwayTicketBooking` to see the app in action.

## 3. Setup using the Zip File of the Directory

1. **Download the ZIP File**:

   Download the ZIP file of the project and extract it to a folder on your local machine.
2. **Set Up Apache and MySQL**:

   If you're running this locally, ensure you have **XAMPP** or **MAMP** installed and running **Apache** and **MySQL**.
3. **Place the Files in the Web Directory**:
   - Extract the ZIP content into the **htdocs** (XAMPP) or **www** (MAMP) folder.

    - If using **XAMPP**, the path would typically be `C:/xampp/htdocs/`.
4. **Configure the Database**:
    - Open **phpMyAdmin** and create a new database (e.g., `railway_db`).
    - Import the SQL file from the ZIP (if provided) to create tables and populate data.
5. **Run the Application**:
    - Open the project in your browser at `http://localhost/repository-name`.
    - The website should be accessible and functional.

# Technologies Used

## Frontend

- **HTML**: Markup language for structuring the website content.
- **CSS**: Stylesheet language for designing the layout and look of the website.
- **JavaScript**: Programming language for adding interactivity and dynamic features to the website.

## Backend

- **JavaScript**: Programming language for adding authentication and implementing different logic based parts.
- **PHP**: Server-side scripting language used for handling logic, data processing, and communication with the database.
- **Firebase**: Used for implementing authentication system using google and email password

## Database

- **MySQL**: Relational database management system for storing user data, train information, and other related records.

## API

- **Web3Forms**: API used for creating and handling web forms for user submissions (for user contact us form submission).

## Hosting

- **InfinityFree**: Free hosting platform used to host the website and handle server-side requests.

### Version Control

- **Git**: Version control system used for tracking changes in the codebase.
- **GitHub**: Repository hosting service for version control and collaboration.

# Features Supported/working outline

## 1. User Authentication

- **Login/Register:**
  Users should log in or register to access most features, ensuring secure and personalized services. Firebase is integrated for authentication.

## 2. Core Features

- **Book Trains:**
  - Search for trains by entering source, destination, and date.
  - Users must log in to proceed with ticket booking.
  - Fill in passenger details, finalize booking, and download tickets.
- **Find Trains:**
  - Search for train availability based on stations and dates without logging in.
  - Direct link to booking if a train is found.
- **PNR Status:**
  - Check the status of a booked ticket by entering the PNR number.
- **Travel Packages:**
  - Explore and book travel packages curated for convenience.
  - Requires login for booking packages.

## 3. Support and Feedback

- **Feedback:**
  - View recent feedback from other users.
  - Submit feedback to improve services.
- **Contact Us:**
  - Fill a contact form for inquiries or assistance.
  - Uses `web3forms API` to send queries directly to admin's Gmail.

## 4. Database and Integration

- **Database:**
    - A central **MySQL Database** stores user, train, booking, and feedback data securely.
- **Firebase:**
    - Used for user authentication and managing user-related data.
- **Web3forms API:**
    - Handles contact form submissions effectively.

## 5. Ticket Management

- Users can download their tickets after booking.
- Real-time updates on booking status are integrated.

## 6. User Flow

- Users can navigate through Home Page > Login/Register > Select Features like Booking, Checking PNR, or Contact Us.
- Seamless redirection ensures smooth user experience across sections

**(workflow of the website in next page)**

# Workflow of the Website