**Family Travel Tracker**

A web application to track the countries visited by family members. Each user can log in, track visited countries, and customize their profile with a unique color theme.

**Features**

* **User Management**: Add family members with a custom profile color.
* **Country Tracking**: Log countries visited by each user.
* **Database Integration**: Uses PostgreSQL to store user and country data.

**Tech Stack**

* **Backend**: Node.js, Express.js
* **Frontend**: EJS (Embedded JavaScript templates)
* **Database**: PostgreSQL
* **Dependencies**: body-parser, express, ejs, pg (as per package.json)

**Installation**

1. **Clone the repository**:

git clone <repository-url>

cd family-travel-tracker

1. **Install dependencies**:

npm install

1. **Set up PostgreSQL Database**:
   * Ensure PostgreSQL is installed and running.
   * Create a database named world.
   * Run SQL scripts to set up the required tables (users, countries, visited\_countries).
2. **Configure Database**: Update the database connection settings in index.js:

const db = new pg.Client({

user: "postgres",

host: "localhost",

database: "world",

password: "your\_password",

port: 5432,

});

1. **Run the Application**:

nodemon /index.js

The app will be available at http://localhost:3000.

**Usage**

* **Homepage**: View a list of countries visited by the currently selected user.
* **Add Country**: Enter a country name to add it to the user's visited list.
* **Switch User**: Select a user to view their travel history.
* **Add New User**: Create a new user with a unique color identifier.

**File Structure**

* **index.js**: Main server file, handles routes and database interactions.
* **public/**: Static assets (CSS, images).
* **views/**: EJS templates for rendering the frontend.

**Dependencies**

The project relies on the following npm packages:

* **express**: Web framework
* **body-parser**: Parses incoming request bodies
* **ejs**: Templating engine
* **pg**: PostgreSQL client for Node.js