Calendar App

This project is a simple calendar web application built using Python Django for the backend and ReactJS for the frontend. It allows users to create, view, edit, and delete events on their personal calendar. The app also supports user authentication and includes a REST API for interacting with calendar events.

# Features

* User Authentication: Secure user registration, login, and logout functionality.
* Event Management: Users can create, view, edit, and delete events on their personal calendar.
* Calendar View: A clean, intuitive interface to display events in a monthly, weekly, or daily view.
* REST API: Exposes calendar functionalities through a REST API for potential integration with other services or apps.
* Responsive Design: Works across multiple device types (desktop, mobile, tablet).

# Tech Stack

Backend: Python, Django, Django REST Framework

Frontend: ReactJS

Database: SQLite (or any other database of your choice)

Authentication: Django Authentication, JWT for token-based authentication

Deployment: Optional (Heroku, AWS, etc.)

# Prerequisites

Python (>=3.6)

Node.js (>=14)

npm or yarn

Django (>=3.0)

React (>=17)

# Getting Started

## Backend Setup (Django)

1. Clone the repository:  
   git clone https://github.com/yourusername/calendar-app.git  
   cd calendar-app
2. Create a virtual environment and activate it:  
   python3 -m venv env  
   source env/bin/activate # On Windows: `env\Scripts\activate`
3. Install the dependencies:  
   pip install -r requirements.txt
4. Apply migrations and start the backend server:  
   python manage.py migrate  
   python manage.py runserver

## Frontend Setup (ReactJS)

1. Navigate to the frontend directory:  
   cd frontend
2. Install the dependencies:  
   npm install # or `yarn install`
3. Start the React development server:  
   npm start # or `yarn start`

# Running the Full Application

Ensure both the Django backend and React frontend are running concurrently. By default, the backend runs on http://localhost:8000 and the frontend on http://localhost:3000.

# API Endpoints

## Authentication Endpoints

* POST /api/auth/login/: Logs in a user
* POST /api/auth/register/: Registers a new user

## Calendar Event Endpoints

* GET /api/events/: Get all events for the authenticated user
* POST /api/events/: Create a new event
* PUT /api/events/{id}/: Edit an existing event
* DELETE /api/events/{id}/: Delete an event

# Deployment (Optional)

If you want to deploy this app (e.g., on Heroku or AWS), you will need to follow specific deployment steps depending on the platform.

# License

This project is licensed under the MIT License - see the LICENSE file for details.