Campus Event Management - Project Report

Project Title: Campus Event Management

Student Name: VISHWAS M

Company/Drive: Webknot Campus Drive Assignment

# 1. Introduction

The Campus Event Management System is a web-based application designed to simplify the process of organizing and managing events within a campus. The platform allows event coordinators to create and manage events, while students can easily view event details and register for participation.  
  
The main motivation behind this project is to reduce manual effort in event handling and to provide a structured platform for smooth communication between organizers and participants.

# 2. Objectives

- Provide a centralized system for managing campus events.  
- Enable students to register for events with ease.  
- Reduce paperwork and streamline event communication.  
- Maintain records of events, participants, and schedules.

# 3. Technology Stack

- Frontend: HTML, CSS, JavaScript  
- Backend: Django (Python Framework)  
- Database: SQLite  
- Version Control: Git & GitHub

# 4. System Design

4.1 Architecture  
The project follows the MVC (Model-View-Controller) pattern provided by Django.  
- Model: Defines the database structure for events, users, and registrations.  
- View: Handles requests and returns responses.  
- Template (Controller part): Provides the user interface using HTML & CSS.  
  
4.2 Workflow  
1. Admin/Organizer creates an event.  
2. Event details are stored in the database.  
3. Students can view the event list and register.  
4. The system maintains participant records for future reference.

# 5. Features

Admin Features:  
- Login & authentication  
- Create, update, and delete events  
- View registered participants  
  
User Features:  
- Register and log in  
- Browse available events  
- Register for an event  
- Receive event confirmation

# 6. Implementation Details

- The project is divided into Django apps to maintain modularity.  
- Models are created for Users, Events, and Registrations.  
- Templates are used for the frontend.  
- Django’s ORM is used for database operations.  
- Authentication is handled using Django’s built-in user system.

# 7. Challenges Faced

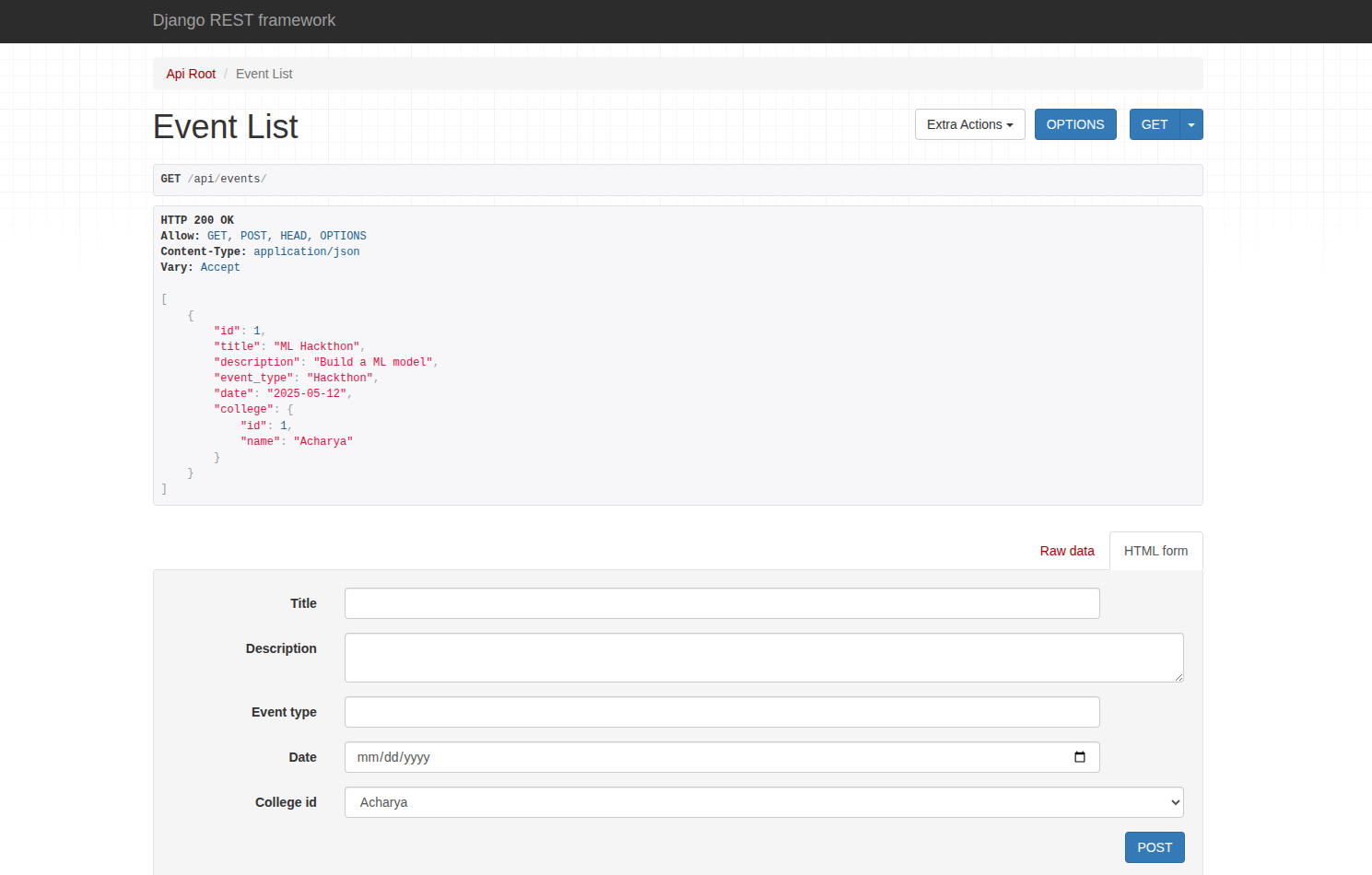
Challenges:  
- Designing a user-friendly UI with basic frontend skills.  
- Managing relationships between users and events in the database.  
- Ensuring proper validation during registration.  
  
Solutions:  
- Learned and applied Django’s relational model system.  
- Referred to Django documentation for authentication.  
- Tested modules step by step to resolve errors quickly.

# 8. Results / Outputs

- Successfully developed a working web application for managing campus events.  
- Students can log in, view events, and register.  
- Admin can manage events and track participation.

# 9. Conclusion

The Campus Event Management project successfully demonstrates how technology can simplify campus-level event coordination. By using Django, the application ensures modular development and provides a scalable base for future improvements like email notifications, advanced search, or mobile support.



# 

# 

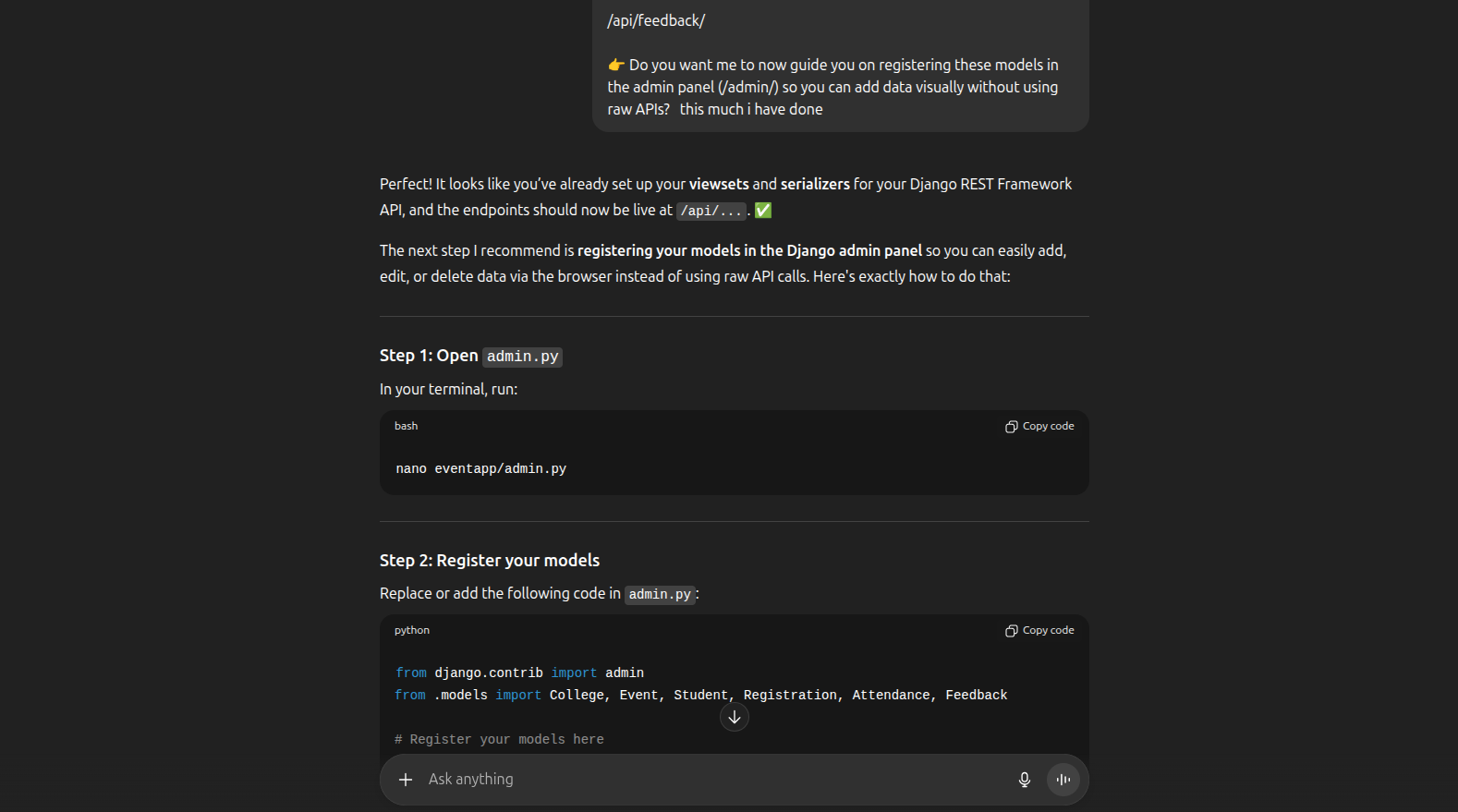
# 10. Future Enhancements

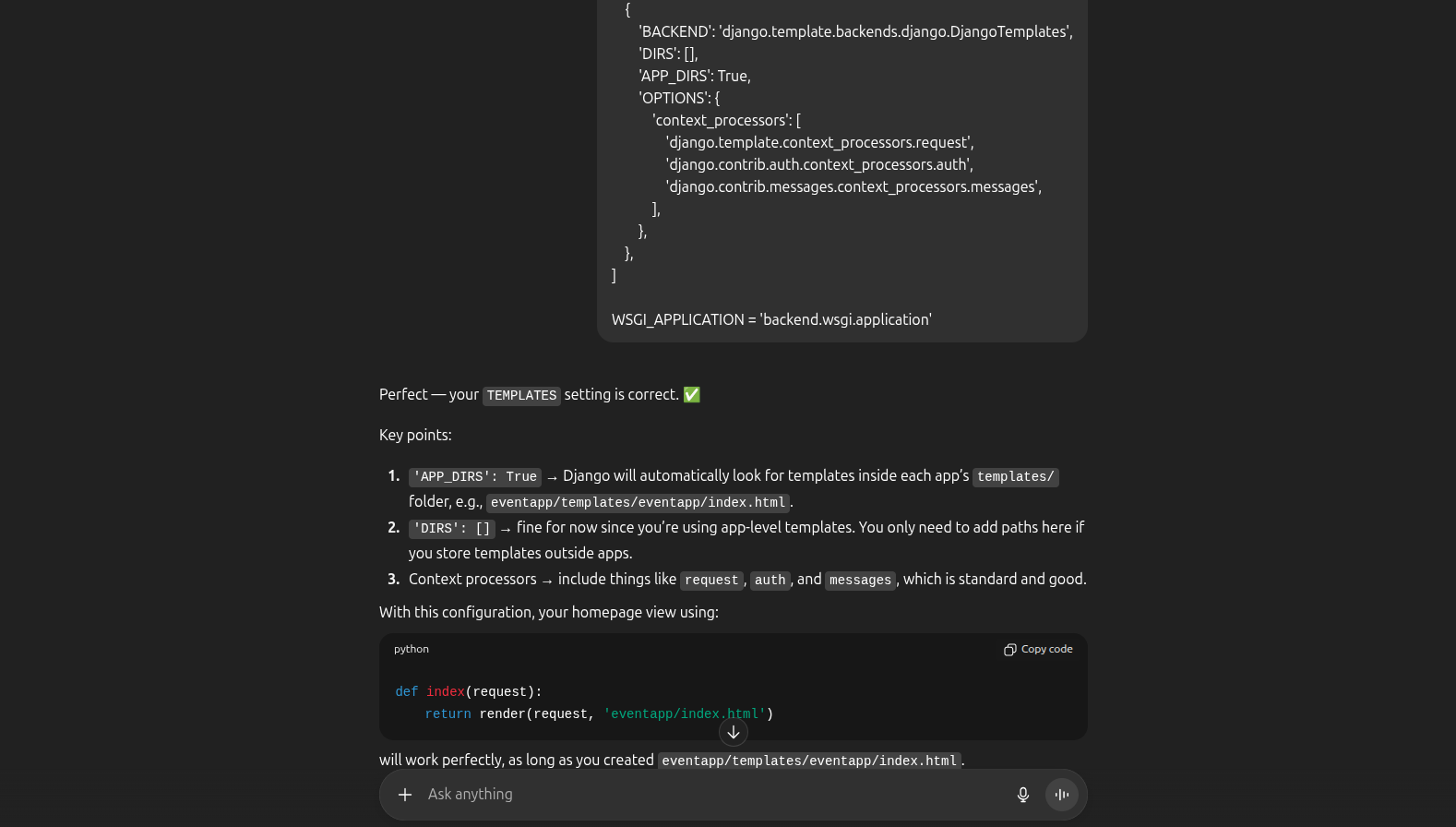
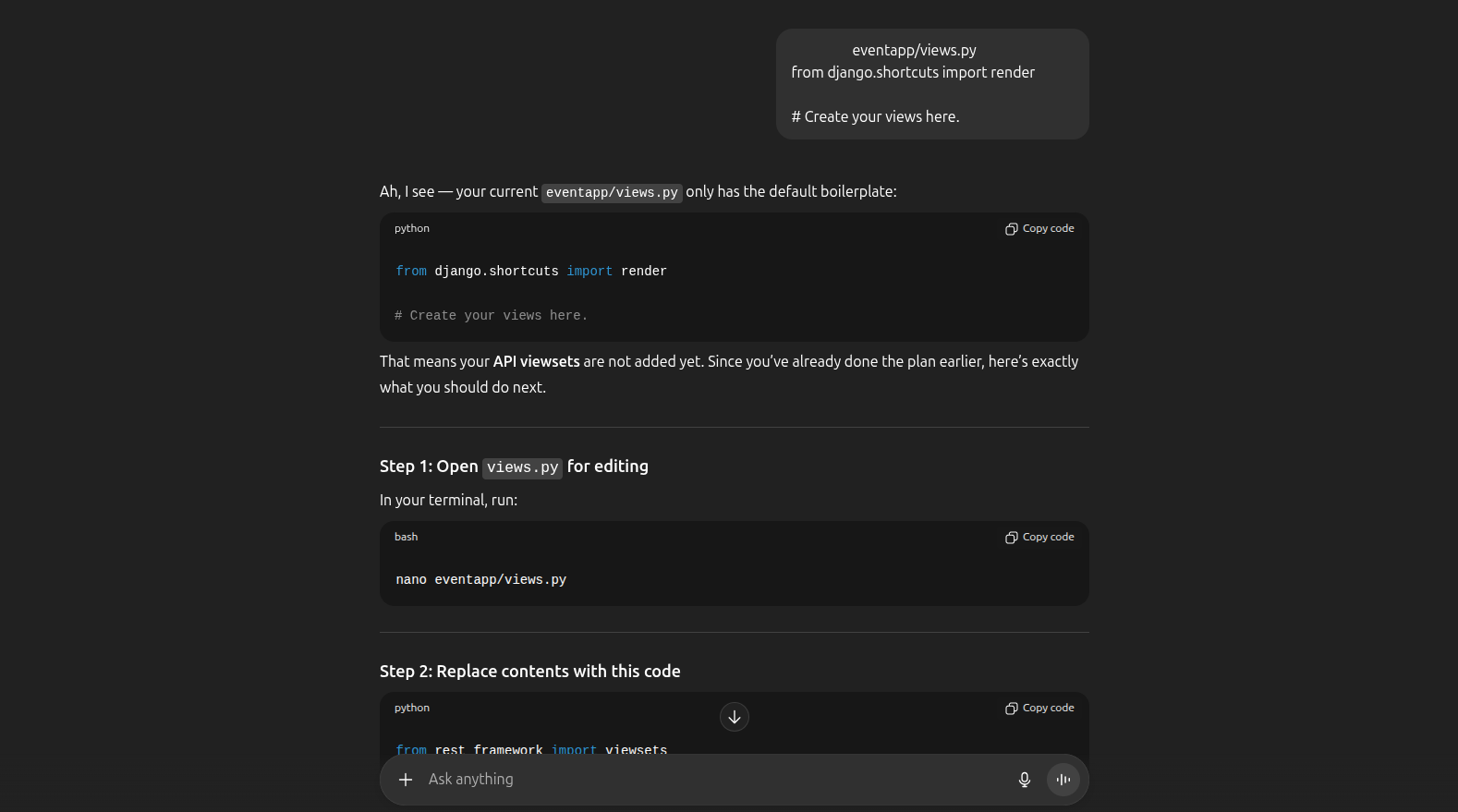
- Add role-based access control for multiple organizers.  
- Enable email/SMS notifications for participants.  
- Implement search and filter options for events.  
- Create a mobile-friendly responsive UI.

# 11. Learning Outcomes

- Gained practical experience with Django framework.  
- Improved understanding of database design and ORM.  
- Learned to integrate frontend with backend functionality.  
- Enhanced debugging and problem-solving skills.

# 12. AI Conversation Log





Submitted by: VISHWAS M

For: Webknot Campus Drive – Assignment Challenge