

# Event Management System

## Presented By : Rakhi Khatri

## Registration Id-Section : 2312383-4G

## Department : Computer Science

## Course : Database Management System

# Instructor : Sir Abid Ali

## *Project Submission Date : 11-june-2025*

# Table of Contents

## 1. Abstract

## 2. Introduction

## 3. System Overview

## 4. Features

## 5. System Architecture

## 6. Technology Stack

## 7. Database Design

## 8. Implementation Details

## 9. Screenshots

## 10. Testing

## 11. Challenges Faced

## 12. Future Enhancements

## 13. Conclusion

## 14. References

# 1. Abstract

This project is an Event Management System built to handle data related to events, venues, tickets, and more. It offers a clean interface and visual dashboards for insights. Developed using HTML, Tailwind CSS, JavaScript, and Chart.js.

# 2. Introduction

The project helps manage event-related data efficiently. The aim is to provide a structured system that supports event organizers in handling large data sets.

The core of this project lies in robust database design (DBMS), while the web application was developed to present and manage the data in a practical and user friendly manner.

# 3. System Overview

The system supports viewing and managing entities like venues, events, tickets, and feedback. It includes visualizations for quick insights and navigation buttons for accessing individual tables.

# 4. Features

- Dashboard with entity stats

- Navigation to different data tables

- Charts for data visualization

- Clean and responsive UI

# 5. System Architecture

The system uses a client-server model. JavaScript fetches data from a backend server, and renders it using Chart.js on the frontend.

# 6. Technology Stack

- Frontend: HTML, Tailwind CSS, JavaScript

- Charts: Chart.js

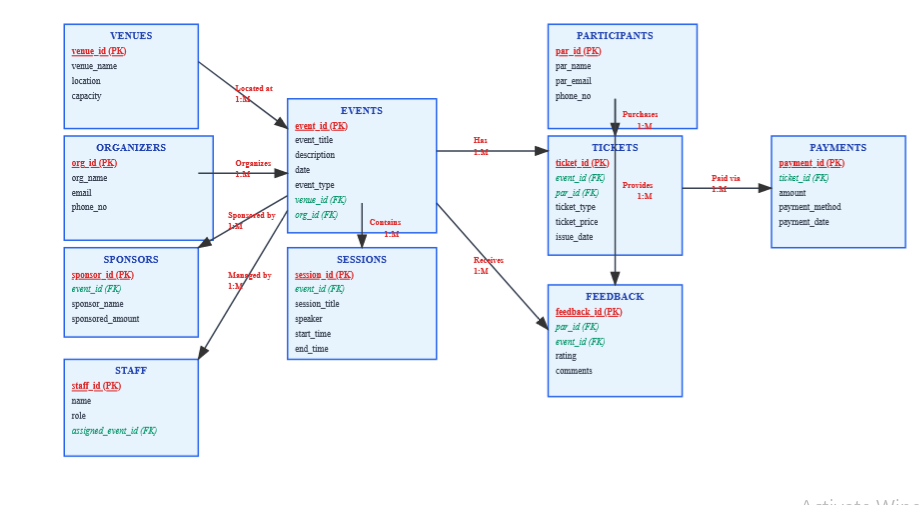
- Backend: Node.js (API based)

- Deployment: GitHub Codespaces

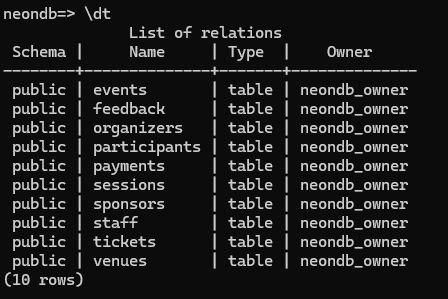
# 7. Database Design

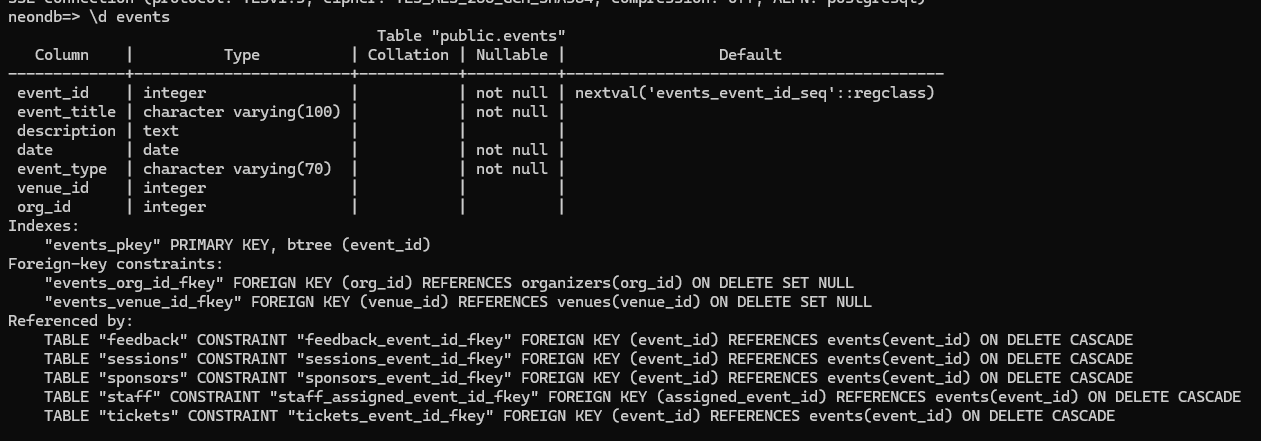
Each entity (like venue, event, etc.) has its own table. Tables include appropriate fields and primary keys. Feedback includes a rating field used for pie chart display.

# ERD:

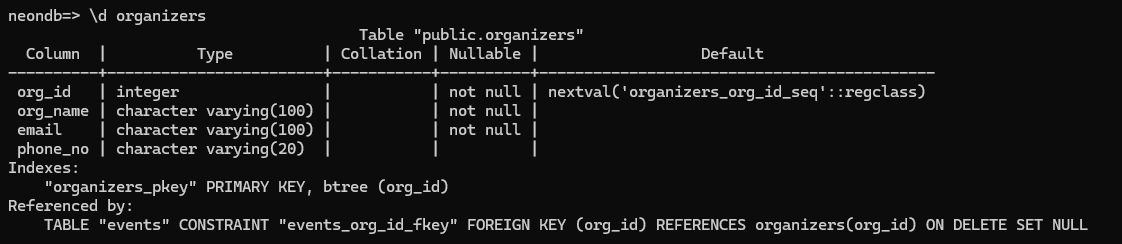


# Database Tables:

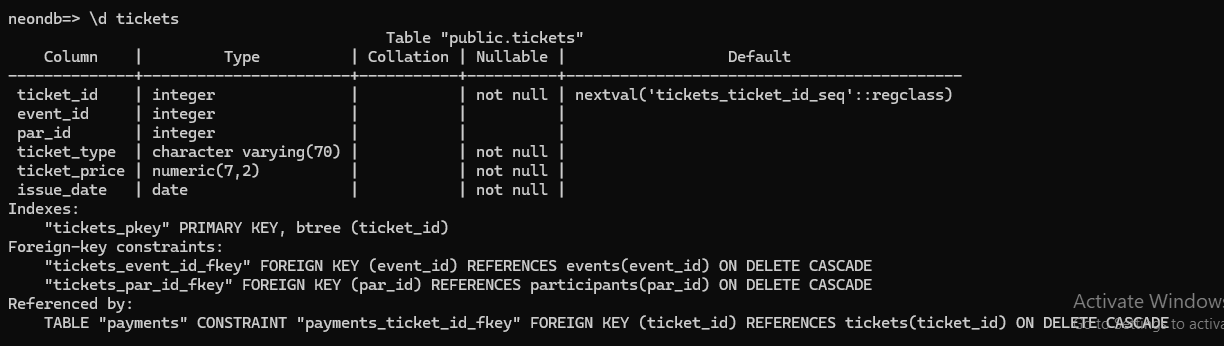


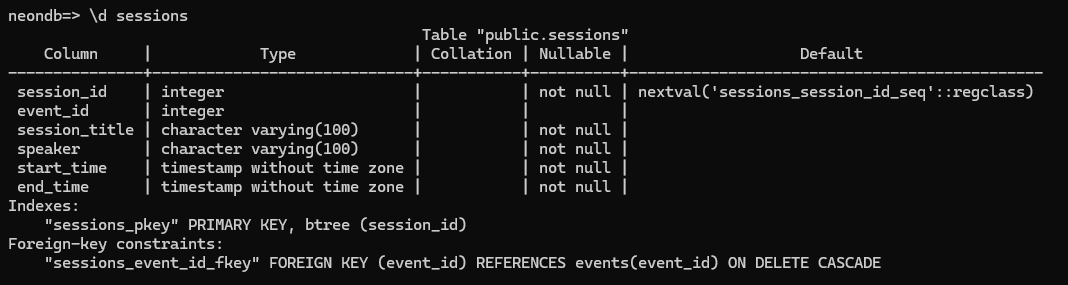






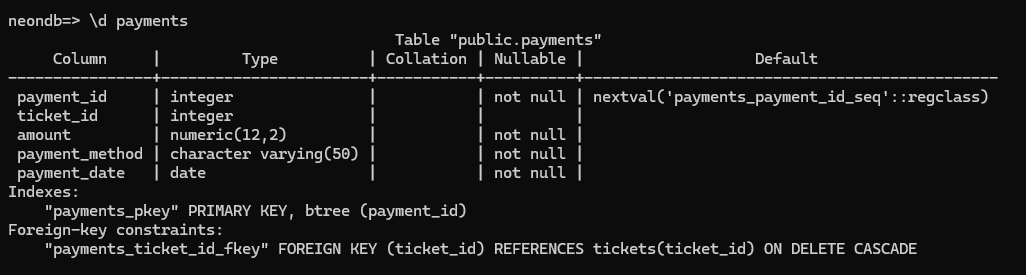


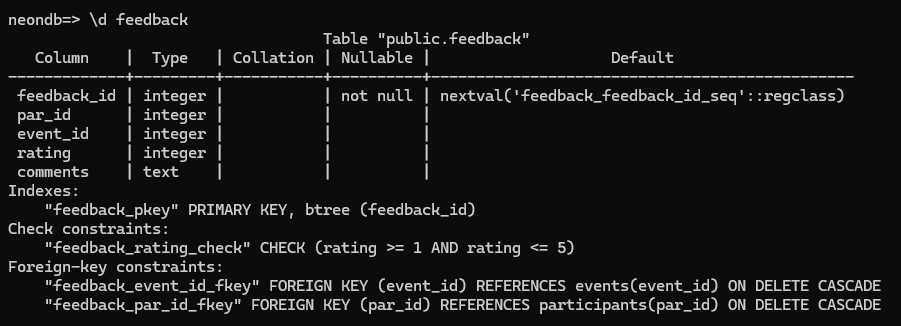










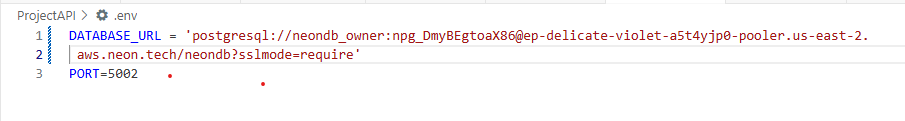


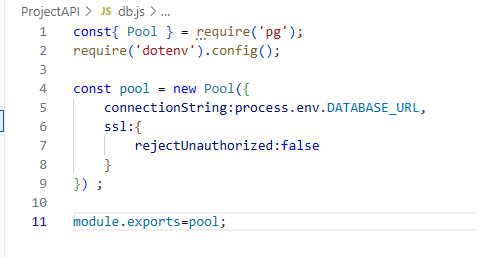
# 8. Implementation Details

JavaScript fetches data using APIs and renders it in bar and pie charts. Navigation is handled through anchor tags. Chart.js is used for visualizations.

# 9. Screenshots of Code Snippets

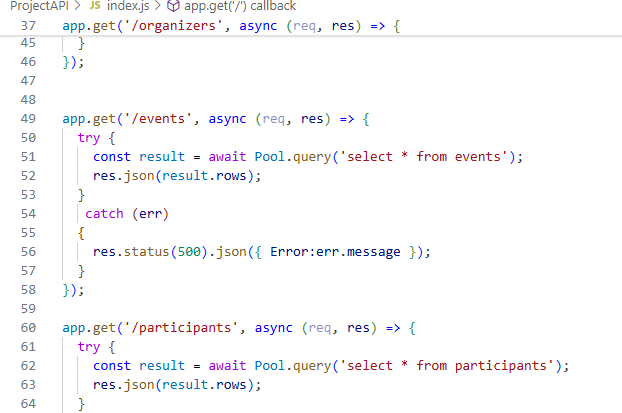
**ProjectAPI Folder (All Files)**

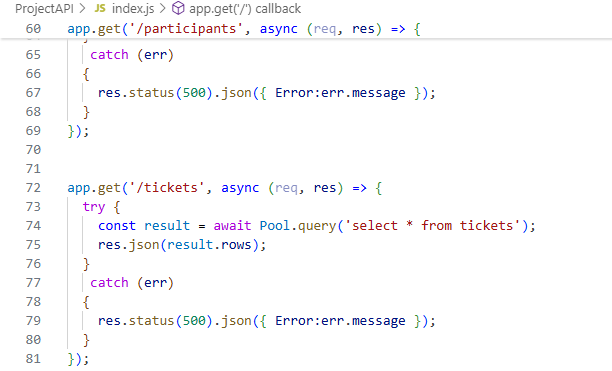














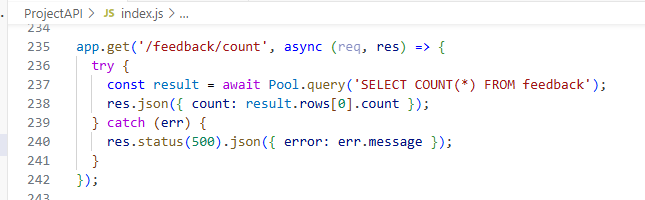




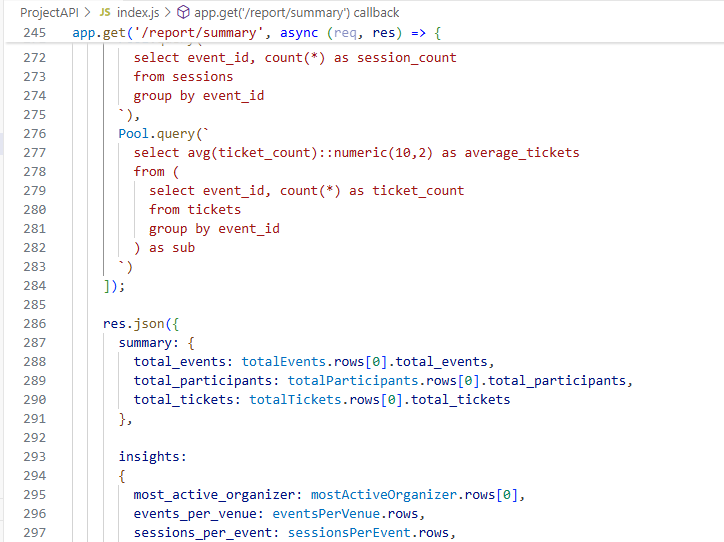




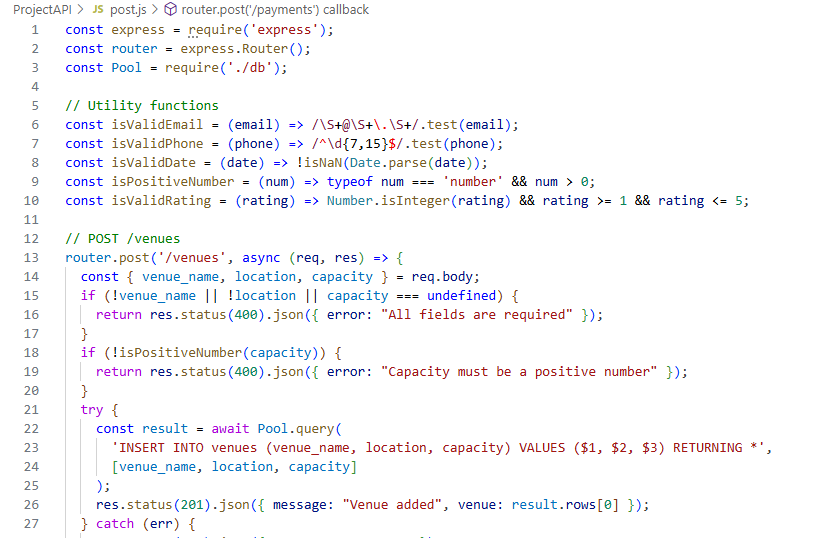


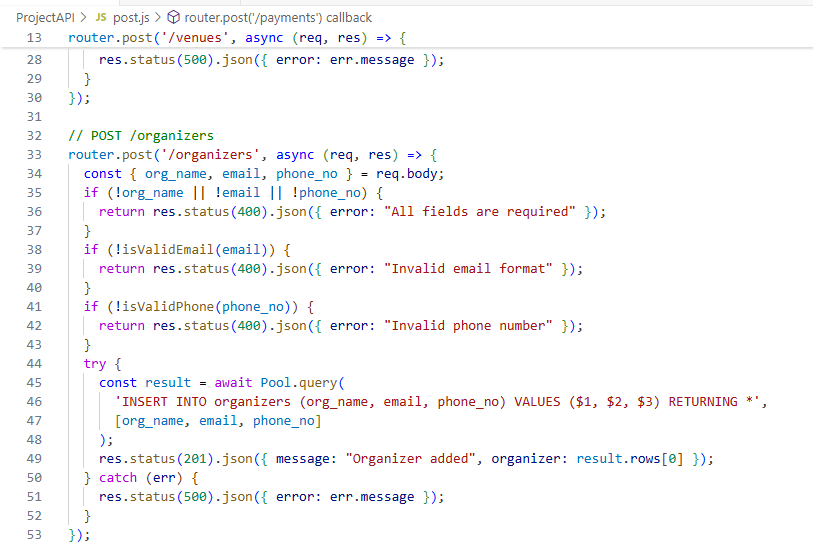


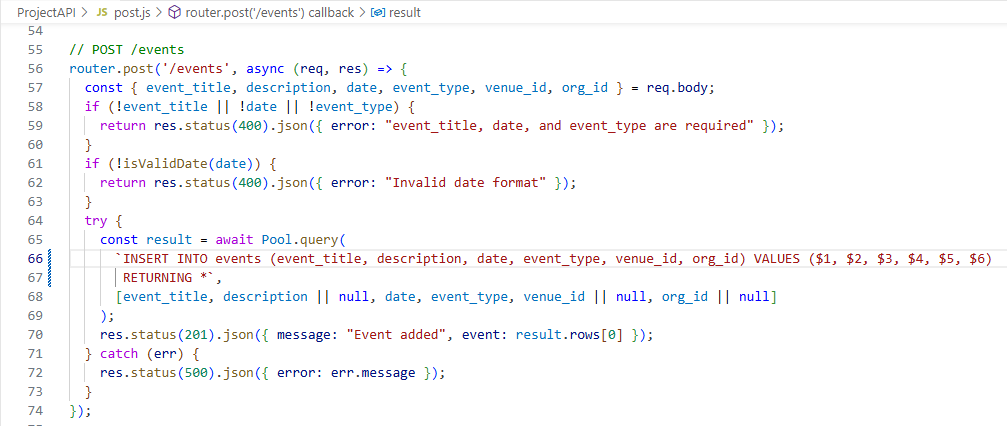




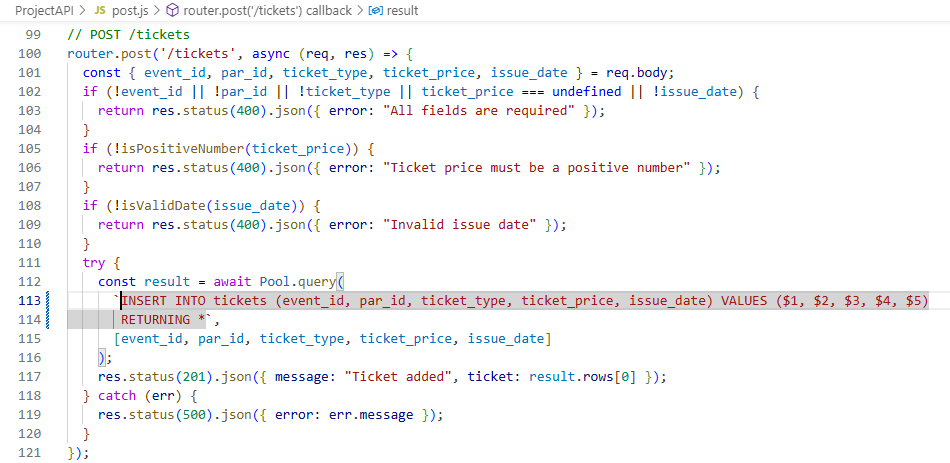


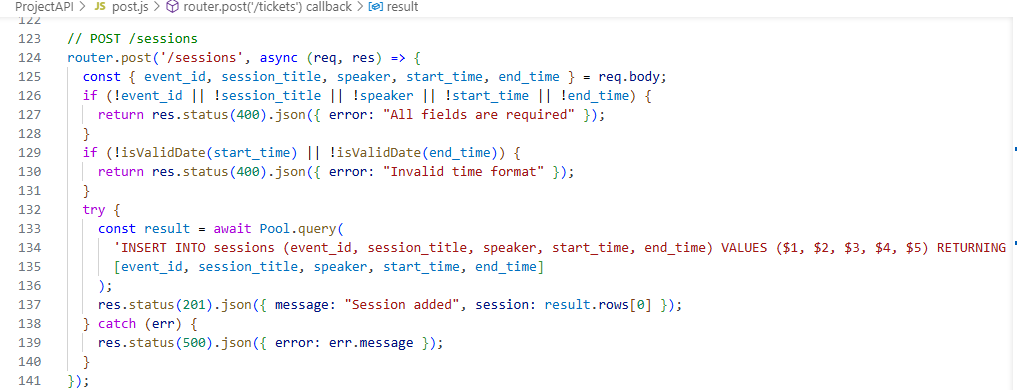




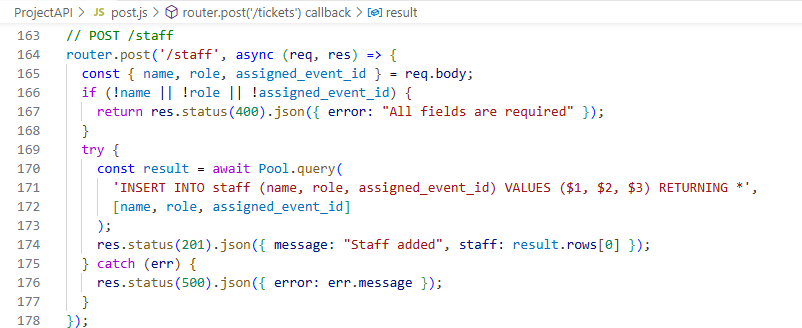


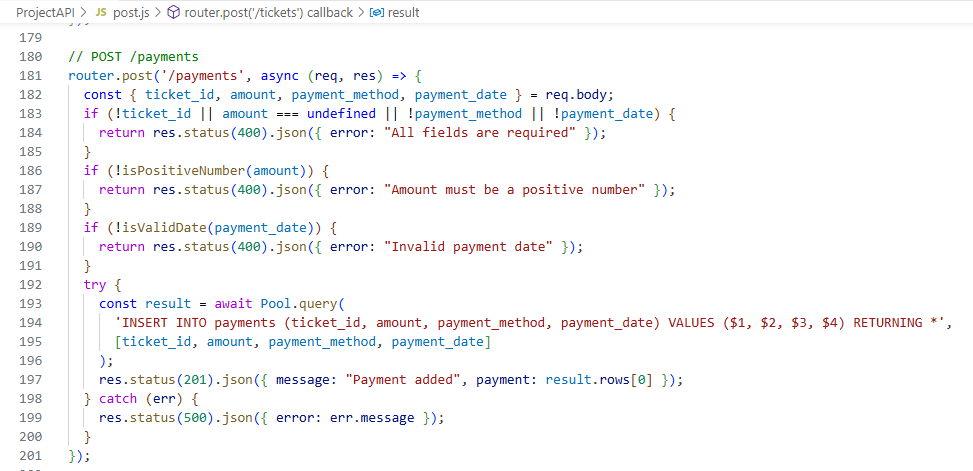


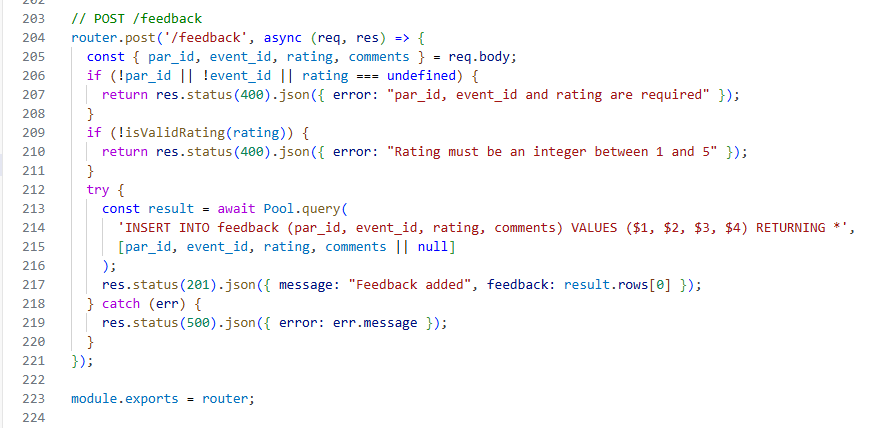












**Projectweb Folder (All Files)**

****

****

****

****

****

****

****

****

****

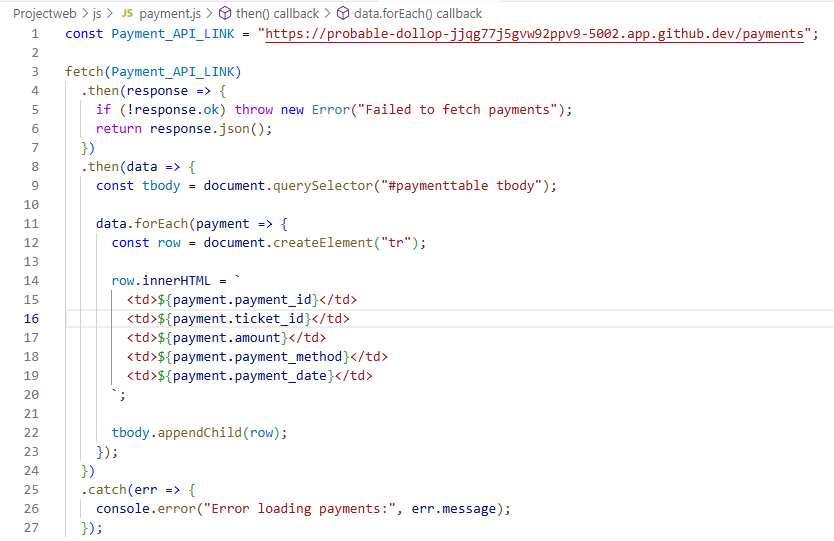
****

****

****

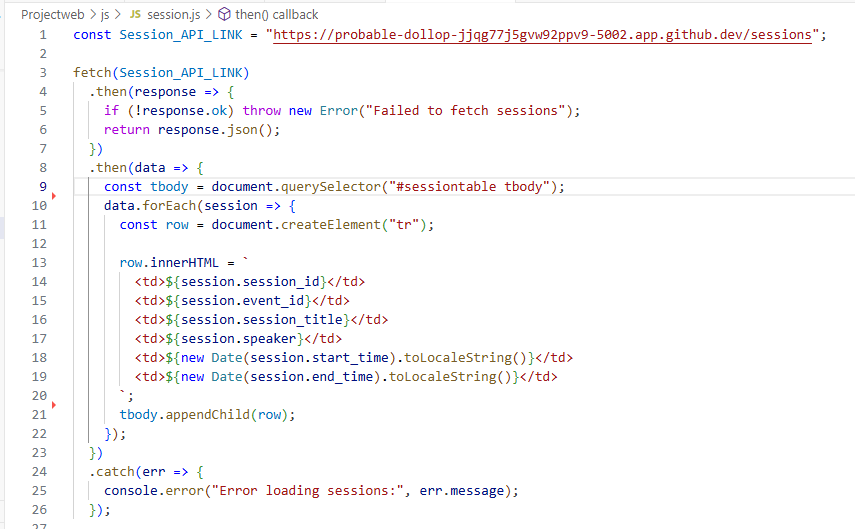
****

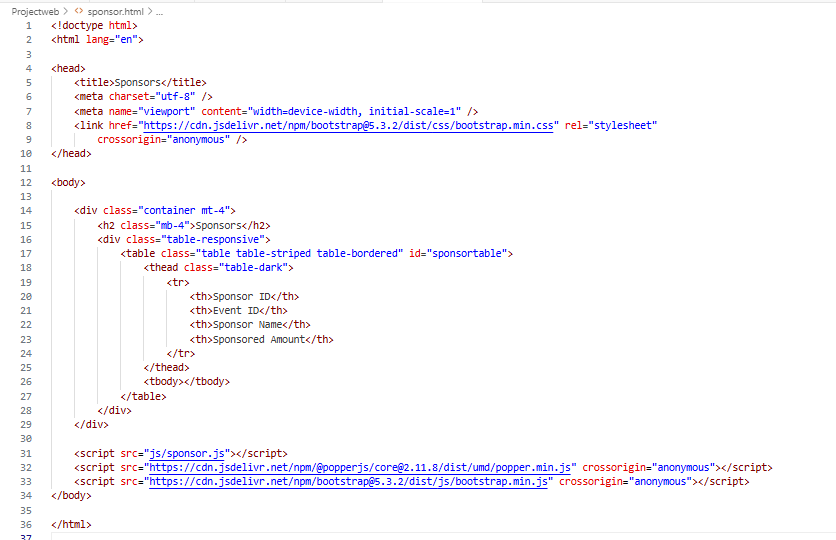
****

****

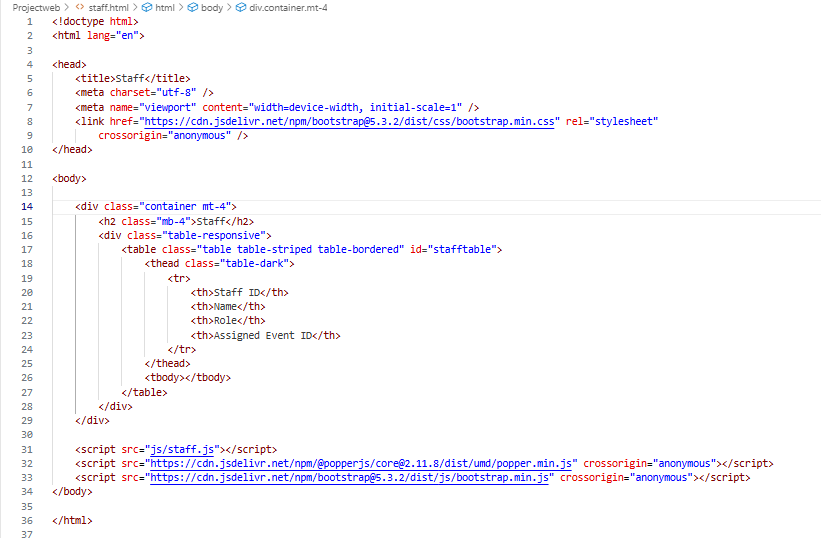
****

****

****

****

****

****

****

****

****

****

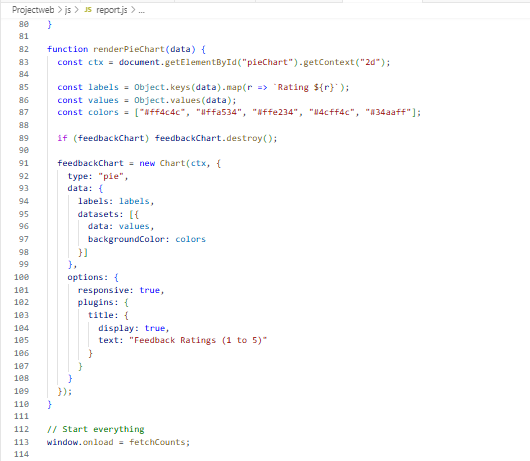
****

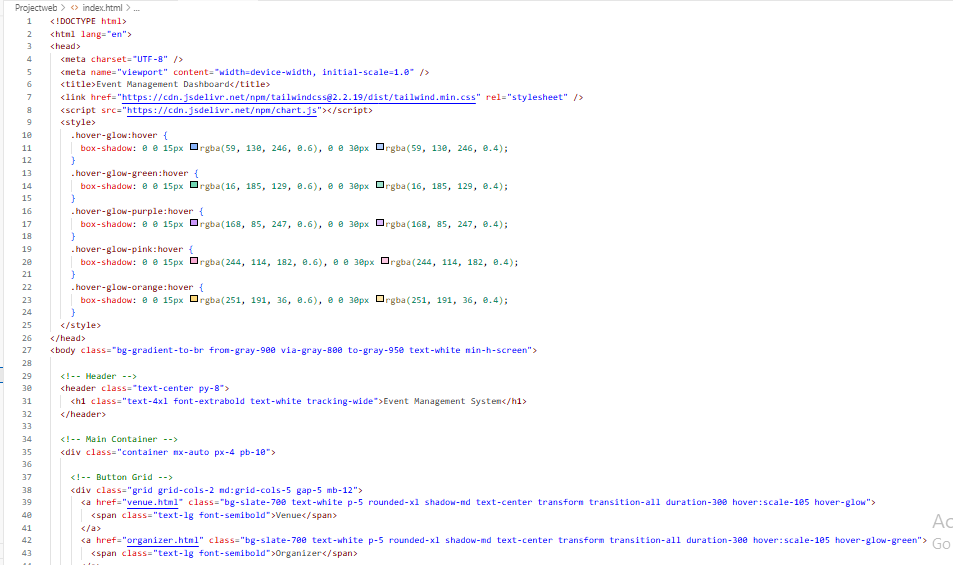
****

****

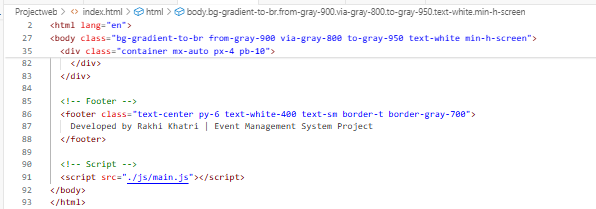
****

****

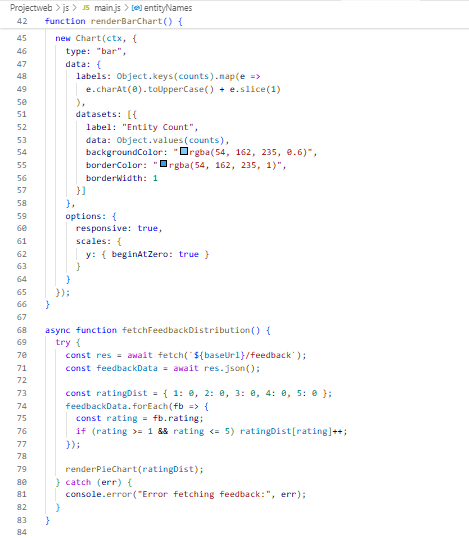
****

****

****

****

****

****

****

# 10. Testing

Tested manually using Postman and browser. Charts were verified using sample data.

# 11. Challenges Faced

- Styling charts professionally

- Managing multiple endpoints

- Ensuring responsiveness

# 12. Future Enhancements

- Filter and search options

- Other Functions(Update and Delete)

# 13. Conclusion

The project helped understand frontend integration, data handling, and visualization. It's a simple yet functional event dashboard.

# Final Summary of the Project:

This project is a complete Event Management System created as a major DBMS project. It is based on a well-designed database that includes tables like Venue, Organizer, Event, Participant, Ticket, Session, Sponsor, Staff, Payment, and Feedback. All the tables are connected using primary and foreign keys with proper relationships, including ‘ON DELETE CASCADE’ and ON ‘DELETE SET NULL’ where required.

To make the project more practical and user-friendly, a web application was also developed. The frontend allows users to view data in a clean and professional dashboard with navigation buttons, bar and pie charts, and a responsive layout. The frontend connects with the backend API to fetch and display live data from the database. This combination of DBMS and web technologies makes the system complete and realistic for real-world usage.