WT Lab Project Report

on

**ATTENDENCE MANAGEMENT SYSTEM**

**USING**

**HTML,CSS AND JAVASCRIPT**

**BACHELOR OF TECHNOLOGY**

in

**COMPUTER SCIENCE AND ENGINEERING**

**2021-2025**

Submitted by

D.Harika 22311A0579

V.Sai Sri Lohitha 22311A0569

V.Parichaya 22311A0571

M.Sumedha 22311A0572

Under the Guidance of

Mr.Ch.Sreedhar

Assistant Professor



**Department of Computer Science & Engineering**

**SREENIDHI INSTITUTE OF SCIENCE AND TECHNOLOGY**

**(AUTONOMOUS)**

Yamnampet (V), Ghatkesar (M), Hyderabad – 501301,T.S.

2021-2025



**CERTIFICATE**

This is to certify that this WT Project Report on carried on “**Attendence Management System**”, submitted by D.Harika (22311A0579), V.Sai Sri Lohitha (22311A0569), V.Parichaya(22311A0571), M.Sumedha(22311A0572) in the year 2025 in partial fulfillment of the academic requirements of Jawaharlal Nehru Technological University for the award of the degree of Bachelor of Technology in Computer Science and Engineering, is a bonafide work in project that has been carried out **during III B Tech CSE II semester**. This report has not been submitted to any other institute or university for the award of any degree.

Faculty Name Dr . Aruna Varanasi

Assistant Professor  **Head, Department of CSE**

Department of CSE

**Project Coordinator**

External Examiner

Date:-

# ABSTRACT

The Attendance Management System is a software application designed to automate and streamline the process of recording, tracking, and managing attendance data of students or employees. Traditional methods of attendance tracking, such as paper registers or manual entry, are time-consuming, error-prone, and inefficient. This system leverages modern technologies to provide a reliable, user-friendly, and scalable solution that ensures accuracy and reduces administrative workload.

The system offers functionalities such as user authentication, real-time attendance marking, report generation, and data analysis. Depending on the implementation, the system may incorporate biometric verification, RFID scanning, QR code scanning, or facial recognition to ensure secure and non-redundant attendance logging. It also provides features for managing leave requests, viewing attendance history, and generating reports for various timeframes.

This project contributes to digital transformation efforts by improving operational efficiency, ensuring data security, and enhancing transparency in attendance tracking. The system can be deployed in educational institutions, workplaces, or any organization that requires regular attendance monitoring.

# INDEX

|  |  |
| --- | --- |
|  | **Page NO** |
| **List of figures** |  |
| **1.INTRODUCTION(description)** |  |
| **1.1 Project Introduction** |  |
| **1.2 Project Overview** |  |
| **1.3 Project Objectives** |  |
| **1.4 Project Structures** |  |
| **1.5 Key Features** |  |
| **2.OUTPUT SCREENS** |  |
| **3.CODE IMPLEMENTATION AND RESULTS** |  |
| **3.1 Implementation Approach** |  |
| **3.2 Languague Used** |  |
| **3.3 Code** |  |
| **4.CONCLUSION** |  |

# INTRODUCTION

* 1. **Project Introduction**

The **Attendance Management System** is a software application designed to simplify and automate the process of tracking and managing student or employee attendance. Traditionally, attendance tracking has been done manually using paper-based registers or spreadsheets, which are time-consuming, error-prone, and inefficient. This project aims to eliminate such inefficiencies by providing a digital platform that records, stores, and manages attendance data in real-time.

The system allows authorized personnel (such as teachers, HR staff, or administrators) to mark attendance quickly through a user-friendly interface. It can generate detailed reports, identify absentee patterns, and even notify concerned individuals via email or SMS. Advanced versions of the system may include features like biometric integration, QR code scanning, and facial recognition to ensure accuracy and security.

By implementing this system, organizations can enhance operational efficiency, reduce administrative workload, and maintain accurate attendance records, which are crucial for performance evaluation, compliance, and payroll processing.

* 1. **Project Overview**

The **Attendance Management System** is a centralized platform developed to efficiently record, manage, and track attendance data for students or employees. The primary objective of this project is to replace traditional, manual attendance methods with a streamlined digital solution that is accurate, secure, and easy to use.

The system is designed with multiple modules, including **user authentication**, **attendance recording**, **report generation**, and **data management**. Depending on the requirements, it can support **manual marking**, **QR code scanning**, **RFID**, or even **biometric verification**. The backend database securely stores all attendance records, while the front end provides a simple and interactive interface for users.

Key stakeholders such as **administrators**, **teachers**, or **HR personnel** can access real-time data, generate attendance reports, and monitor attendance trends. Notifications can be sent automatically to alert absentees or inform parents or managers about irregularities.

The system contributes significantly to improving accuracy, reducing paperwork, and saving time. It is scalable and can be customized to meet the needs of schools, colleges, companies, or other organizations.

**1.3 Project Objectives**

1. **Automate Attendance Recording**  
   To eliminate manual attendance tracking and provide a digital, efficient way to mark and store attendance.
2. **Improve Accuracy and Reduce Errors**  
   To minimize human errors and data loss by maintaining a secure and reliable database of attendance records.
3. **Save Time and Effort**  
   To reduce the time spent on marking and calculating attendance manually, thereby improving productivity.
4. **Generate Real-Time Reports**  
   To enable generation of daily, weekly, or monthly attendance reports for analysis and decision-making.
5. **Enable Remote Access**  
   To provide access to attendance data from anywhere through a web-based or mobile interface (if applicable).
6. **Ensure Data Security and Privacy**  
   To implement authentication and access control mechanisms to protect sensitive attendance data.
7. **Support Multiple Users and Roles**  
   To accommodate different user roles such as admin, faculty, and student/employee, each with appropriate access levels.
8. **Provide Alerts and Notifications**  
   To send notifications or alerts for absentees, low attendance warnings, or reminders.
9. **Enhance Record Keeping and Compliance**  
   To help institutions or organizations maintain attendance records that support academic evaluation or payroll compliance.

**1.4 Project Structures**

#### 1. **User Interface (UI) Layer**

* **Login/Registration Page**  
  For user authentication (admin, teacher, student/employee).
* **Dashboard**  
  Displays a summary of attendance data and quick actions.
* **Attendance Marking Page**  
  Interface for marking and viewing attendance.
* **Reports Page**  
  Allows users to generate and view attendance reports.
* **Notifications Page**  
  Displays alerts, reminders, and messages.

#### 2. **Application Layer (Logic)**

* **Authentication & Authorization Module**  
  Handles login, role-based access, and user sessions.
* **Attendance Management Module**
  + Mark Attendance
  + View Daily/Monthly Attendance
  + Edit/Update Entries (if allowed)
* **Reporting Module**  
  Generates custom attendance reports in PDF/Excel formats.
* **Notification Module**  
  Sends alerts via email/SMS (optional).
* **Admin Module**
  + Manage Users
  + Configure system settings
  + Manage departments/classes

#### 3. **Database Layer**

* **User Table**  
  Stores user credentials, roles, and profile info.
* **Attendance Table**  
  Stores daily attendance records.
* **Course/Department Table**  
  Links students/employees to their respective classes or teams.
* **Logs Table**  
  Tracks login history and system activity.

#### 4. **Technologies Used**

* **Frontend:** HTML, CSS, JavaScript (or React/Angular)
* **Backend:** Python (Django/Flask), Java (Spring), or PHP
* **Database:** MySQL / PostgreSQL / MongoDB

**5**. **Documentation & Deployment**

* **User Guide**
* **Technical Documentation**
* **Deployment Scripts (Docker/Cloud)**

**1.5 Key Features**

#### 1. ****User Authentication & Role Management****

* Secure login for different user roles (Admin, Teacher/Manager, Student/Employee).
* Role-based access to functionalities.

#### 2. ****Attendance Marking****

* Manual attendance marking through web or mobile app.
* Support for **QR code**, **RFID**, or **biometric** attendance (optional).
* Bulk attendance marking for a class or department.

#### 3. ****Real-Time Attendance Tracking****

* Instant update and storage of attendance records.
* View daily, weekly, or monthly attendance data.

#### 4. ****Data Analytics & Dashboards****

* Visual dashboards showing attendance trends and summaries.
* Identify irregular patterns or frequent absentees.

#### 5. ****Leave Management Integration****

* Option for students/employees to apply for leave.
* Track approved/rejected leave requests and integrate with attendance.

#### 6. ****Admin Controls****

* Add/Edit/Delete users, departments, and attendance entries.
* Configure holidays, working hours, and attendance rules.

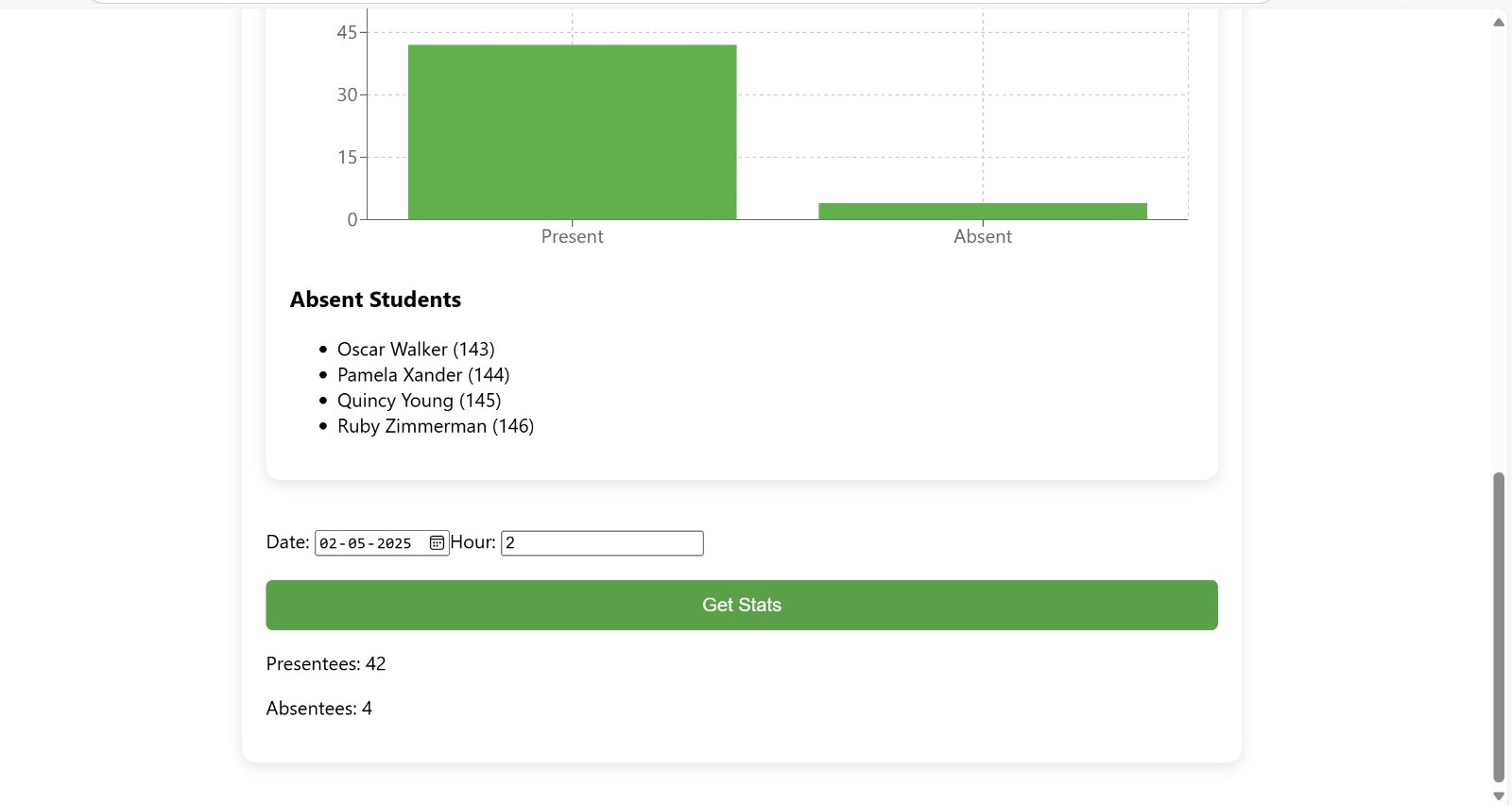
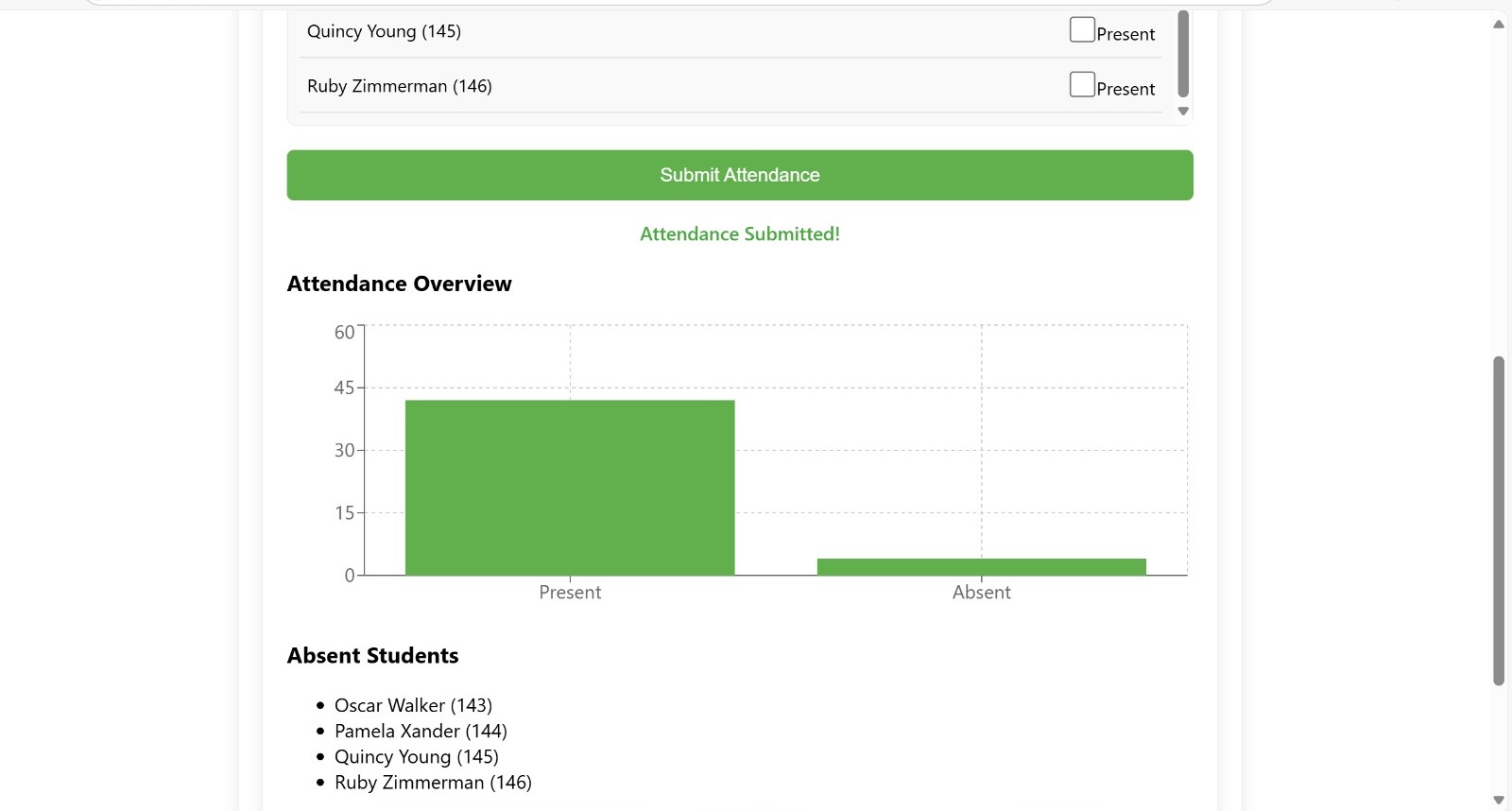
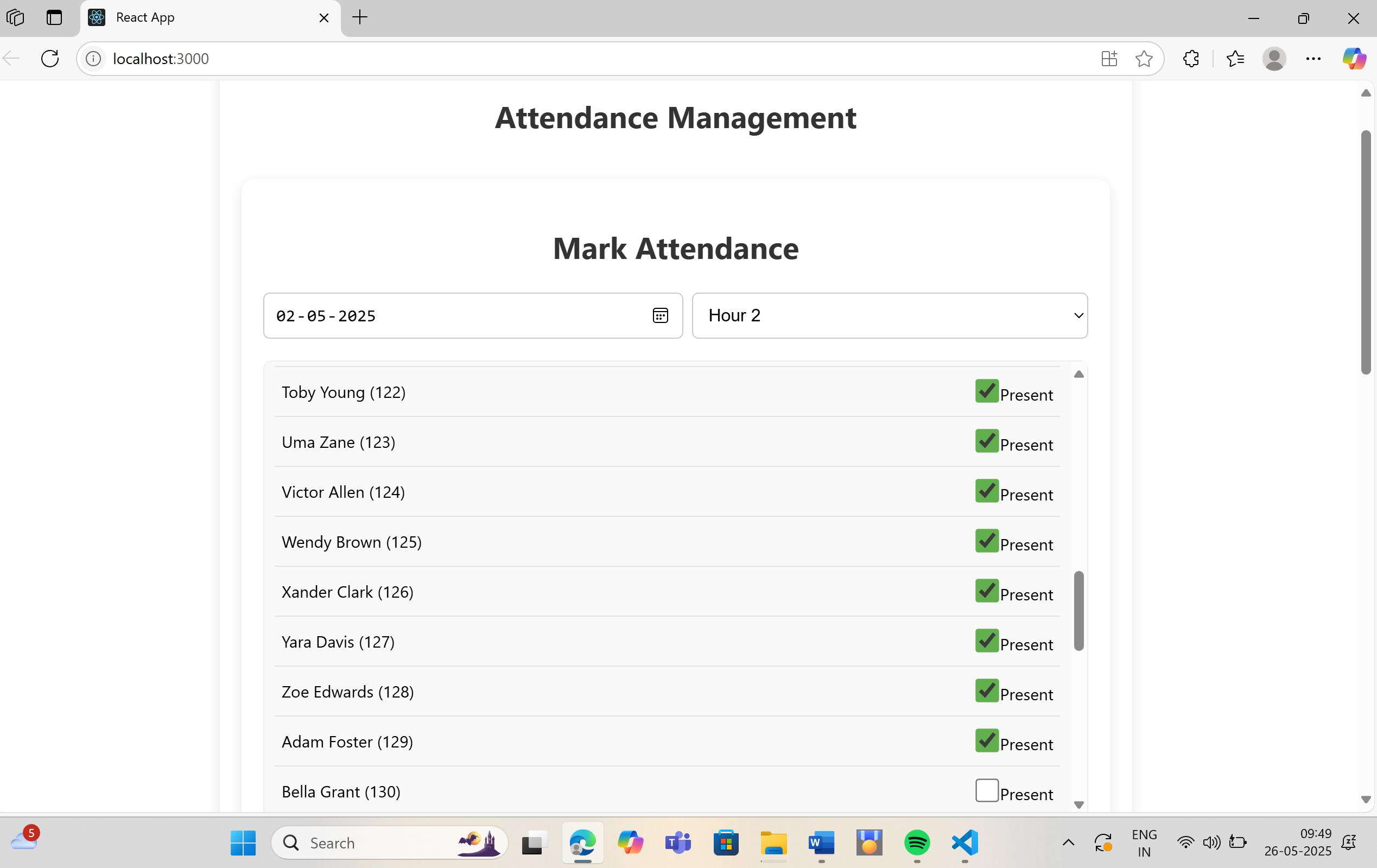
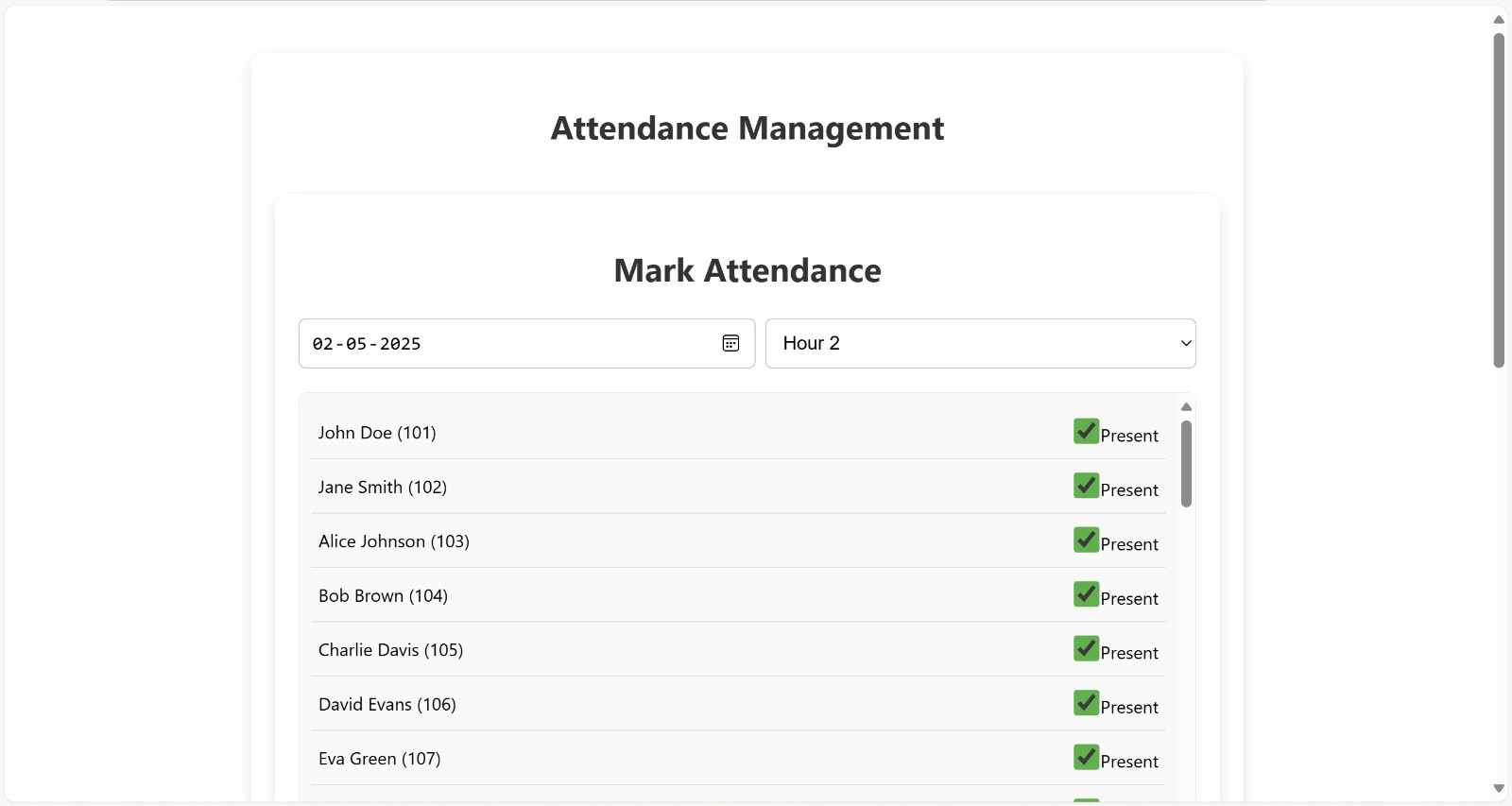
#### 7. ****Multi-Platform Support****

* Accessible on web browsers and mobile devices.
* Responsive design for different screen sizes.

#### 8. ****Secure Data Management****

* Encrypted database to protect sensitive attendance records.
* Backup and restore functionality.

# OUTPUT SCREENS



**CODE IMPLEMENTATION AND RESULTS**

**3.1 Implementation Approach**

### 1. ****Data Collection and Preparation****

* **Input Sources**: The system uses structured CSV files such as:
  + attendance.csv: Contains records of student/employee attendance with fields like ID, date, status (Present/Absent), and department.
  + users.csv or districts.csv: Contains user metadata such as name, department, class, or district for grouping and mapping.
* **Preprocessing Tasks**:
  + Cleaning null or duplicate entries.
  + Converting date formats to standard datetime objects.
  + Merging multiple data sources for a unified analysis dataset.

### 2. ****Attendance Analysis Logic****

* Encapsulated in modules like attendance\_analysis.py (originally crime\_analysis.py in your project).
* Key Functions:
  + Group attendance by student, department, or date.
  + Compute total classes, presents, absents, and attendance percentage.
  + Identify defaulters or students below threshold attendance.
  + Allow filtering by date range, department, or user ID.

### 3. ****Visualization and Reporting****

* Managed by visualization.py.
* Charts and graphs are generated using libraries like **Matplotlib**, **Seaborn**, or **Plotly** to display:
  + Daily attendance trends.
  + Department-wise attendance distribution.
  + Individual student attendance progress over time.
  + Heatmaps of attendance across dates and users.

### 4. ****Application Interface****

* Main logic or interface handled in app.py:
  + Could be a web interface using **Flask** or **Streamlit**, or a CLI tool.
  + Allows users (admins/instructors) to:
    - Upload new attendance files.
    - Filter and analyze specific records.
    - Visualize real-time attendance reports.
    - Export reports as images or PDFs.

### 5. ****Modular Code Organization****

* The code is organized in the src/ directory for better separation of concerns:
  + app.py: Launches the main application.
  + attendance\_analysis.py: Responsible for all backend calculations.
  + visualization.py: Generates charts and graphs.
  + \_\_init\_\_.py: Marks it as a Python package.

### 6. ****Extensibility and Future Enhancements****

* The current system can be extended with:
  + **Database Integration**: Use SQLite, MySQL, or Firebase for scalable storage.
  + **Authentication System**: Role-based login for Admins, Teachers, Students.
  + **QR/Face Recognition Integration**: Automate real-time attendance marking.
  + **Notifications**: Send alerts for low attendance via email or SMS.

**3.2 Languague Used**

The development of the **Attendance Management System** leverages a combination of backend programming, data handling, and optional frontend technologies to build a responsive and efficient system for tracking and analyzing attendance data. Below is a breakdown of the languages and tools used in the project:

**1. Python**

* Used as the **primary programming language** for both backend processing and optional frontend logic.
* Powers the core functionality of the system, including:
  + Reading and parsing CSV files.
  + Grouping and filtering attendance records.
  + Calculating attendance metrics and generating reports.
* Acts as the **engine** of the system, handling all business logic and data operations.

#### ****2. HTML (Optional for Frontend UI)****

* Used when integrating a web-based interface using frameworks like **Flask** or **Streamlit**.
* Provides the **structure** of the user interface including input forms, filters, and data views.
* Facilitates user interaction with the attendance dashboard or report generation tools.

#### ****3. CSS (Optional for Frontend UI Styling)****

* Applied for styling the HTML elements in case of a Flask-based frontend.
* Enhances the **visual appearance** of the web interface, including layout, colors, and responsiveness.
* Ensures a **clean and user-friendly design** when displaying reports or charts.

#### ****4. JavaScript (Optional for Interactivity)****

* Used in case the frontend is extended with interactive elements like dynamic filters, real-time search, or visual attendance charts.
* Enables enhanced **user interaction**, such as filtering attendance by date or department without refreshing the page.

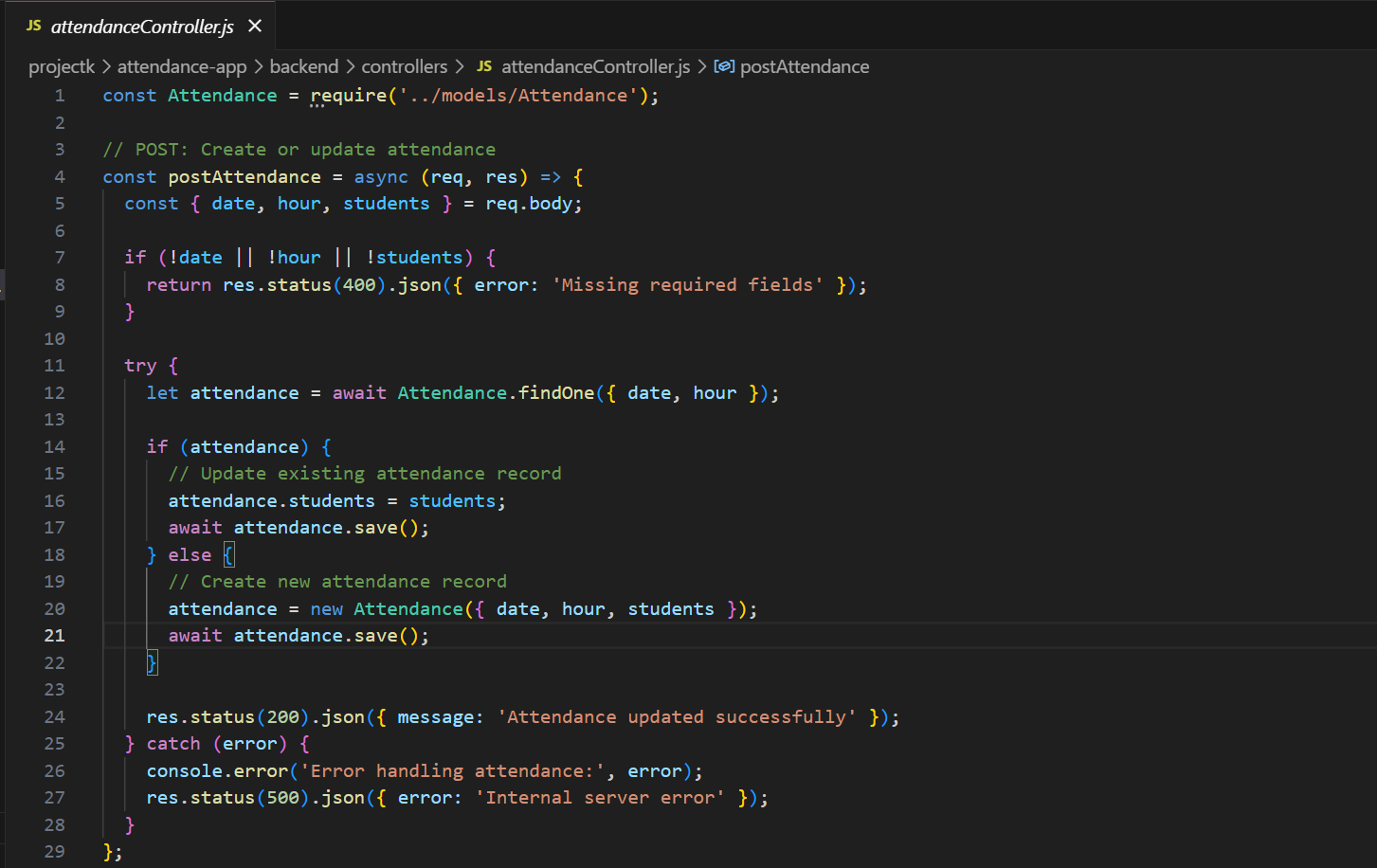
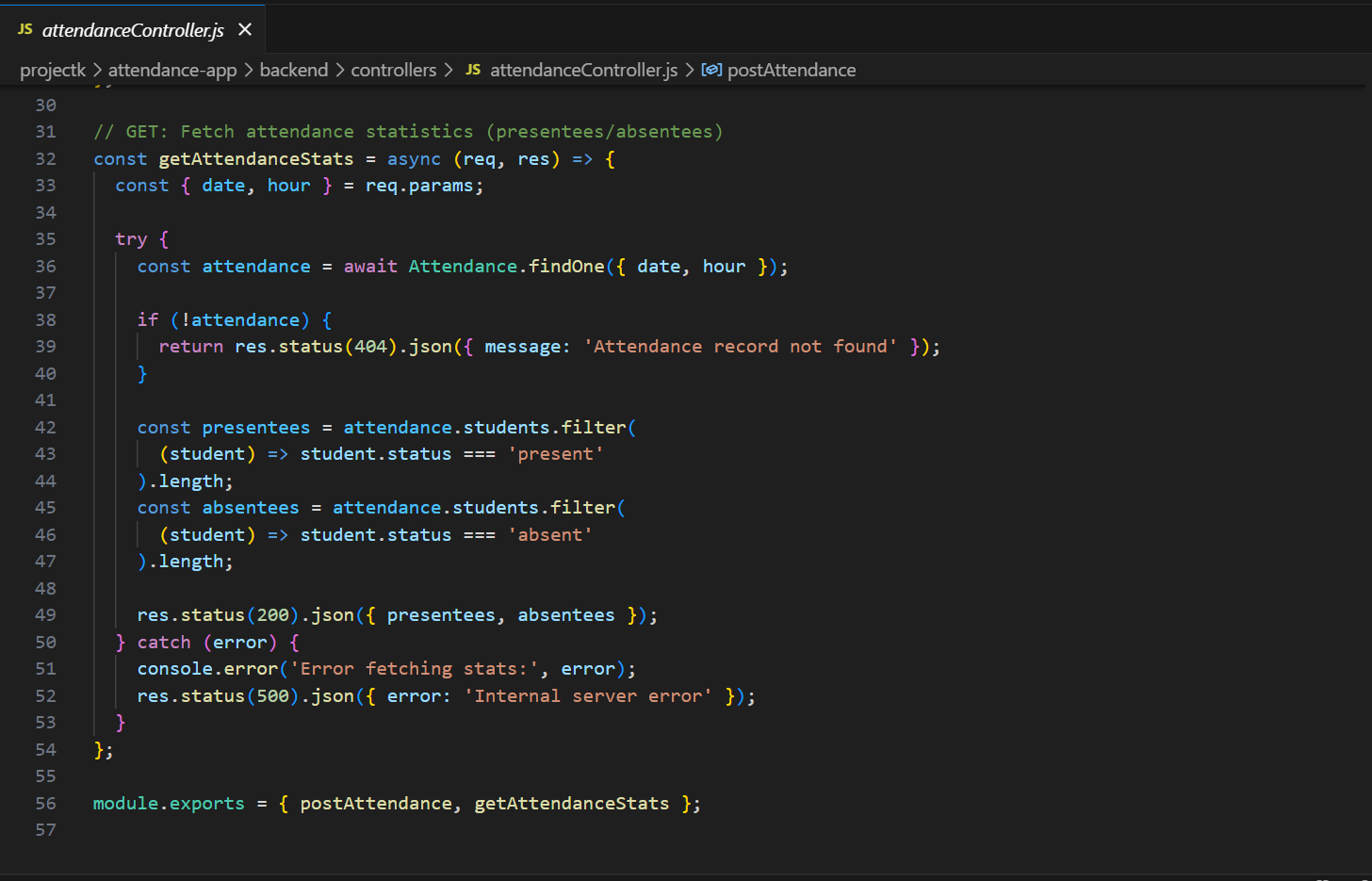
#### ****5. CSV (Comma-Separated Values)****

* Used as the **data storage format** for attendance records and user metadata (e.g., attendance.csv, districts.csv).
* Easily readable by both humans and Python programs.
* Serves as a lightweight and portable alternative to traditional databases in the current version of the project.

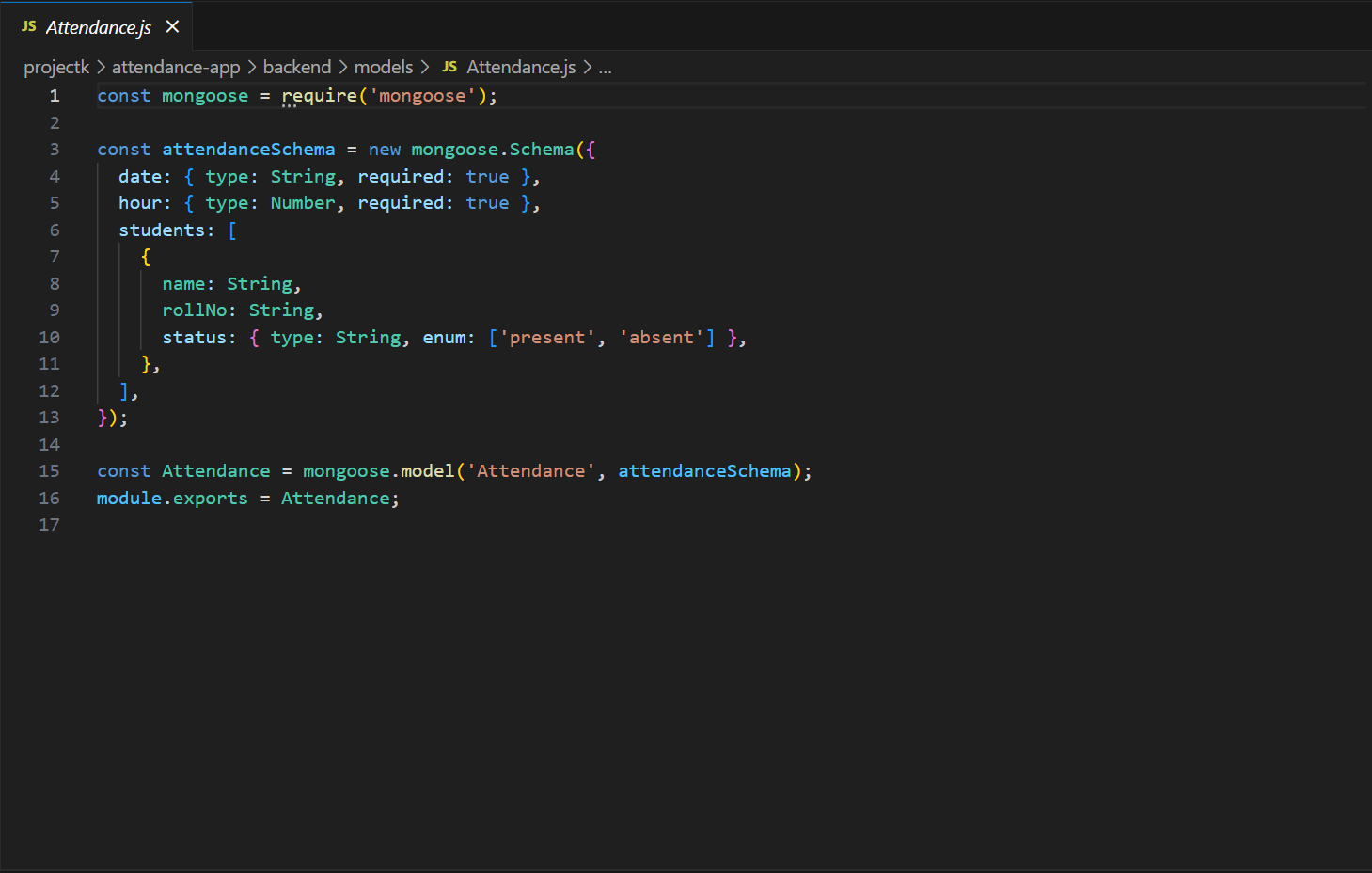
This technology stack ensures a modular and scalable solution that can be easily extended in future versions, such as integrating databases (MySQL/SQLite), QR code scanning, or facial recognition for automated attendance marking.

**3.3 Code**

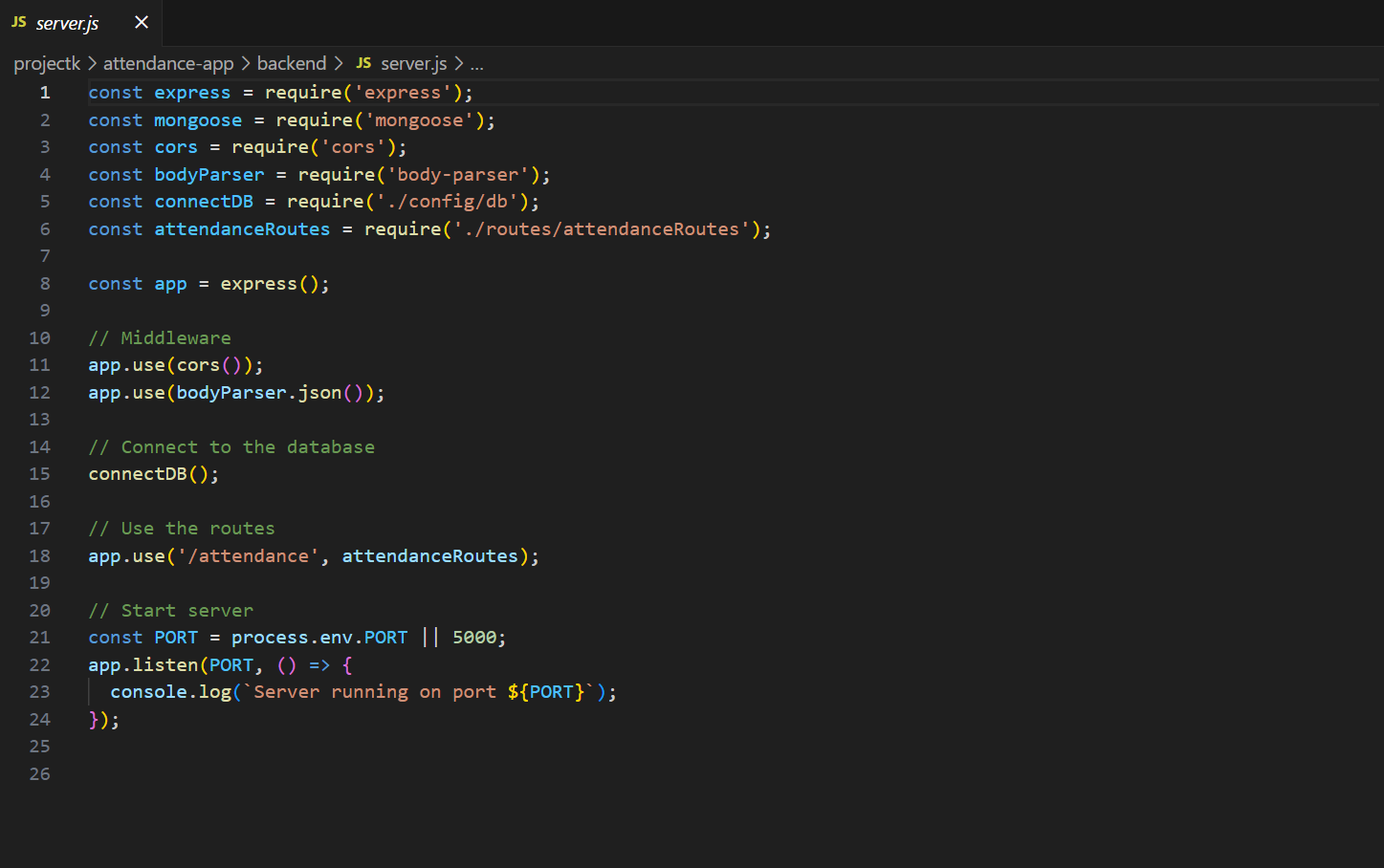
1.attendencecontroller.js



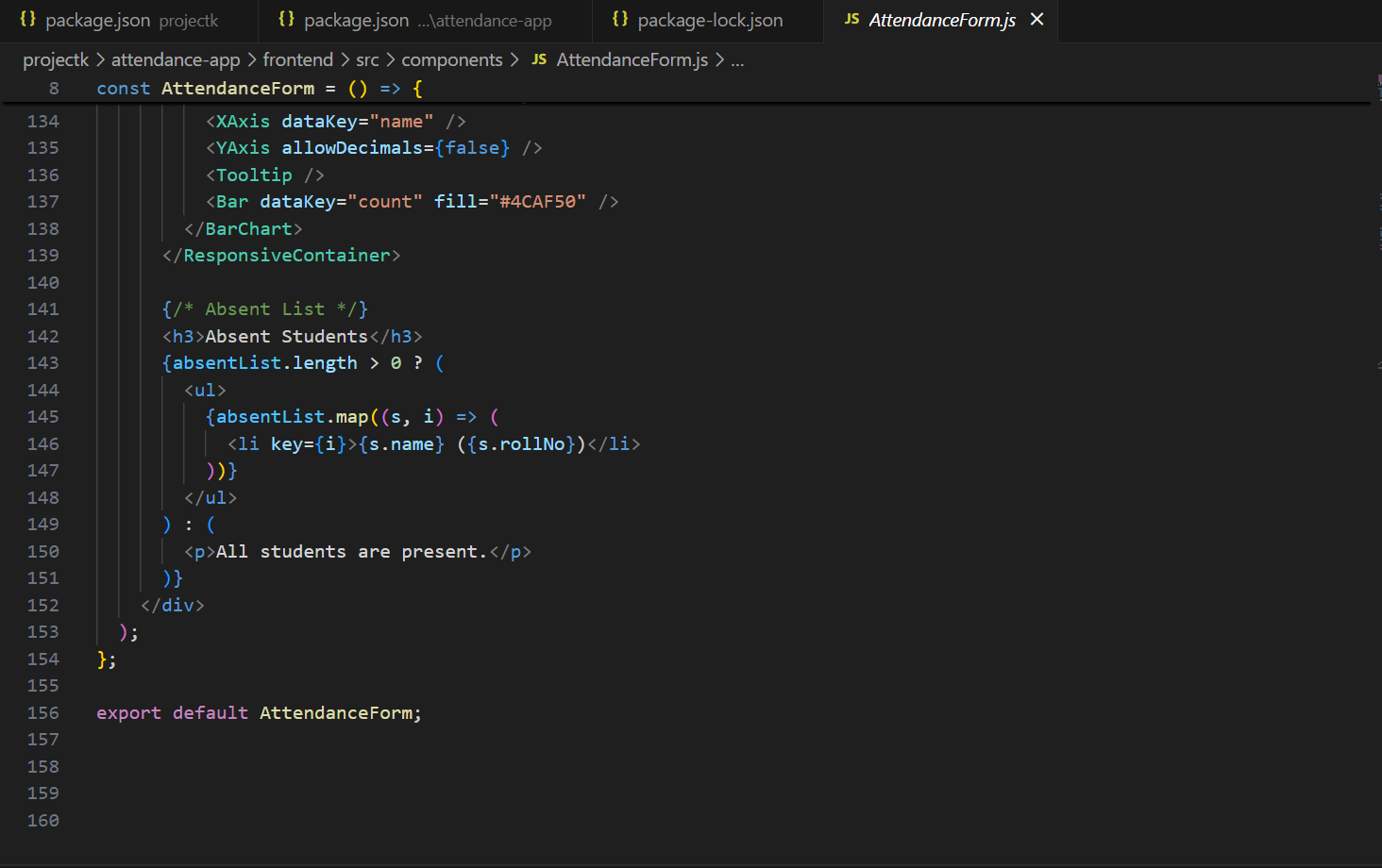
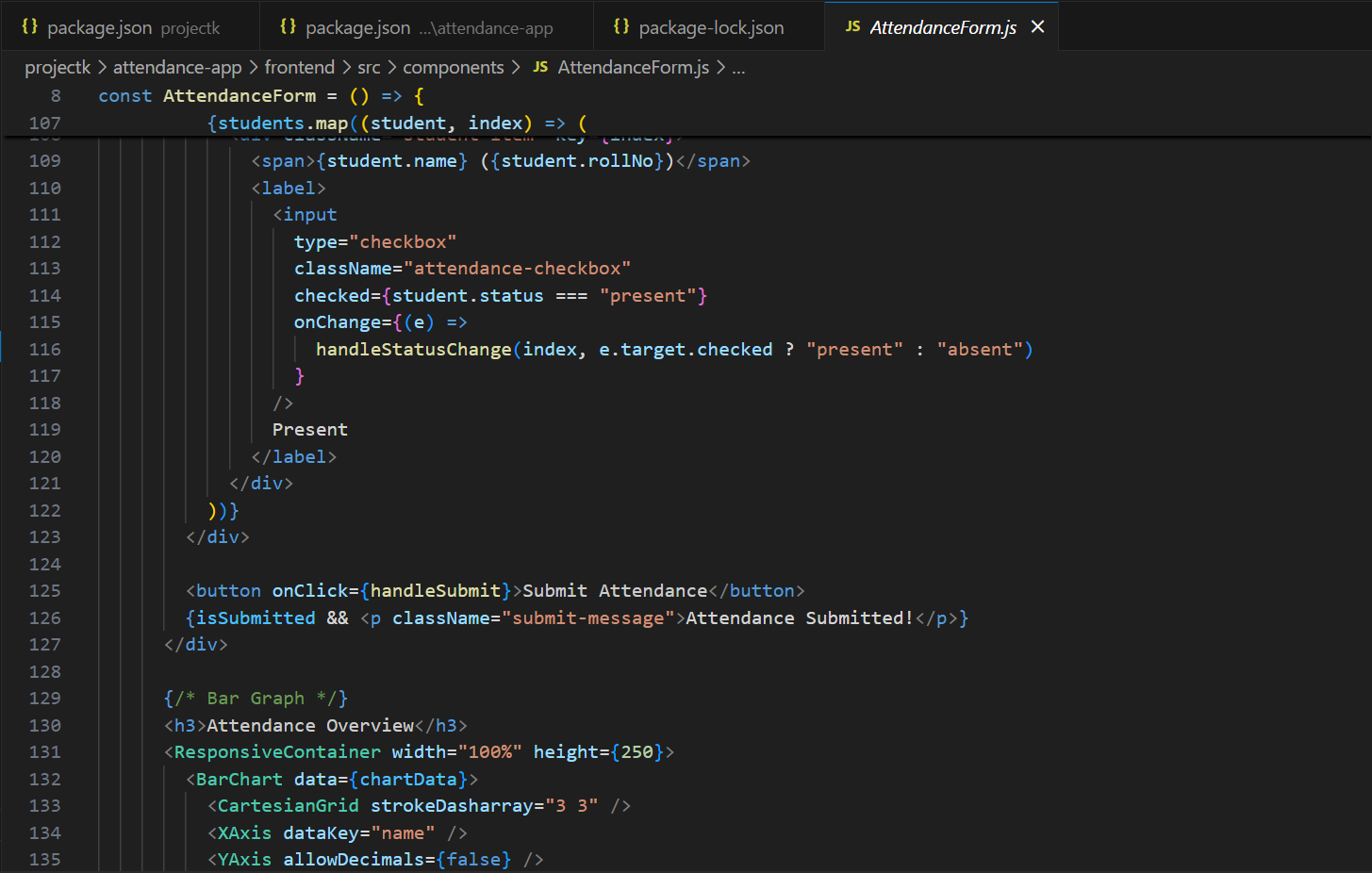
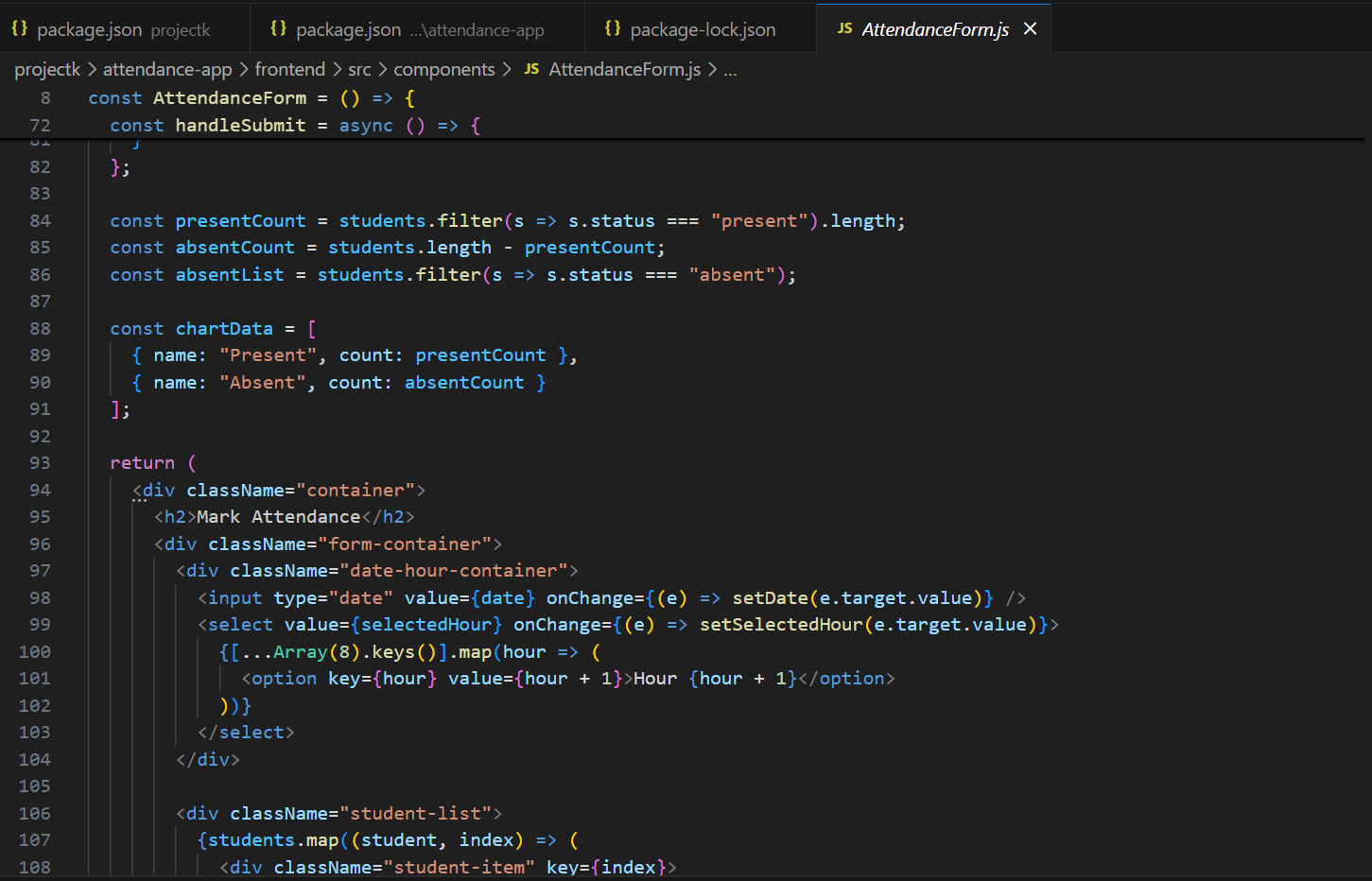
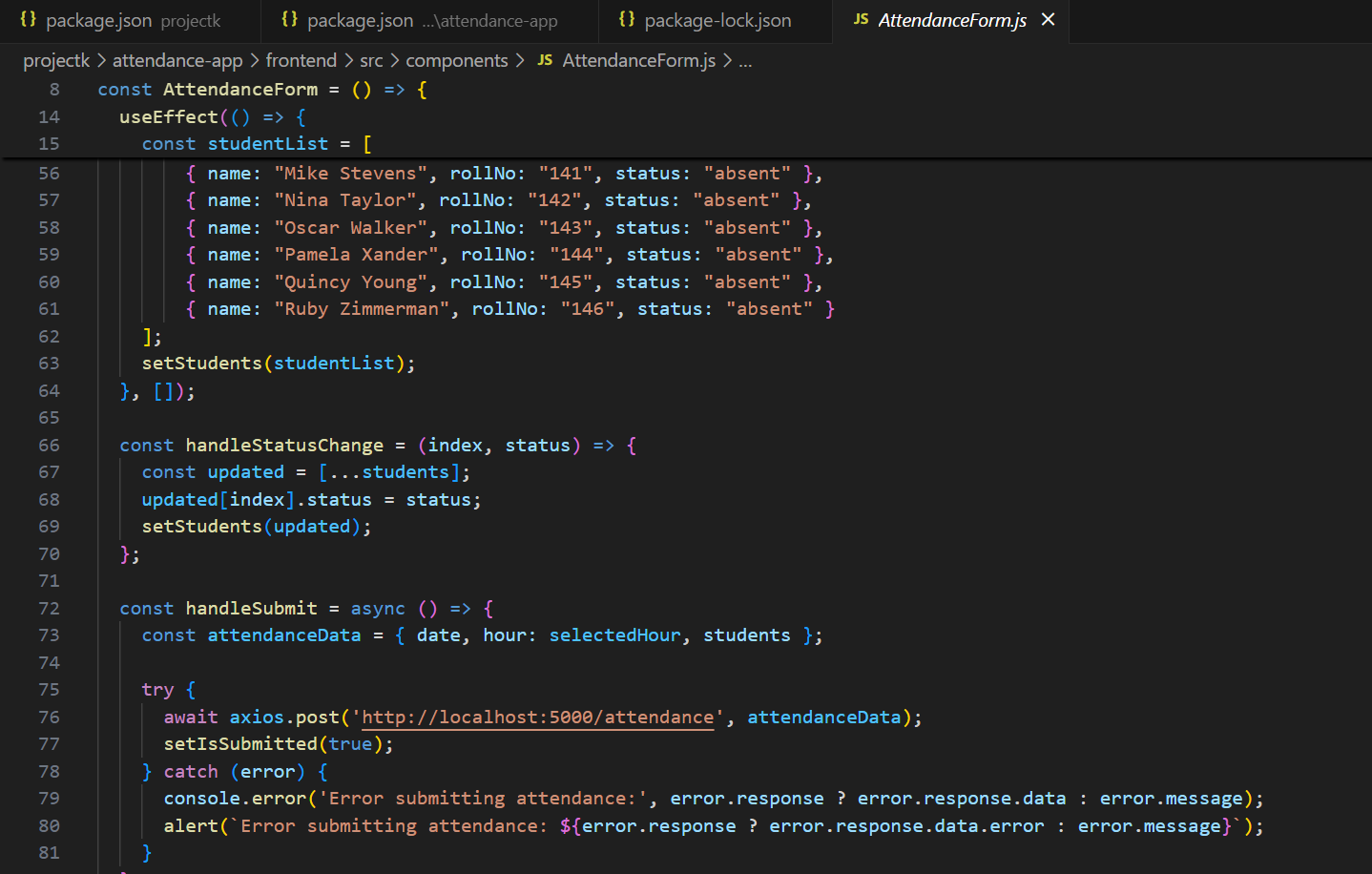
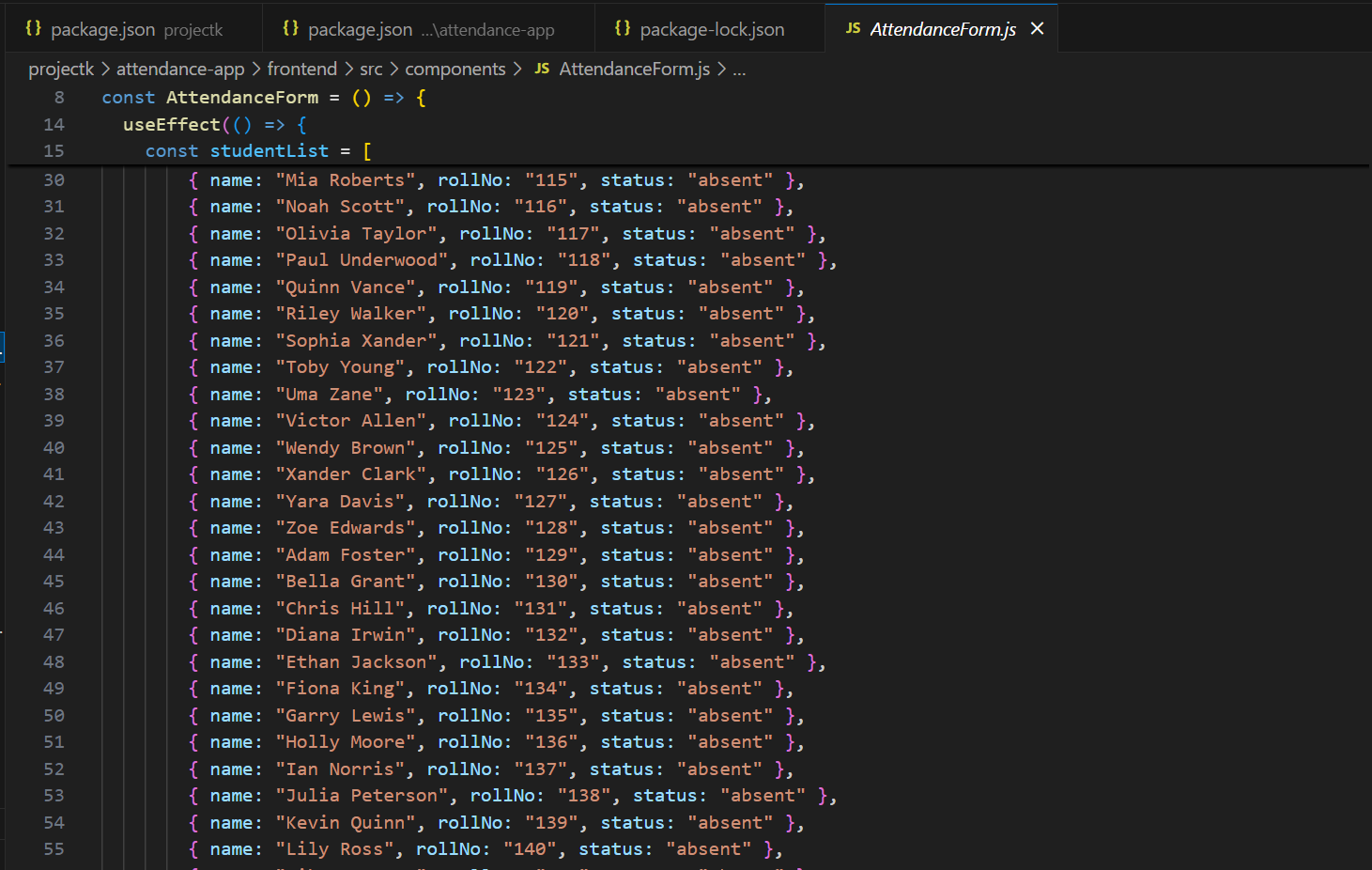
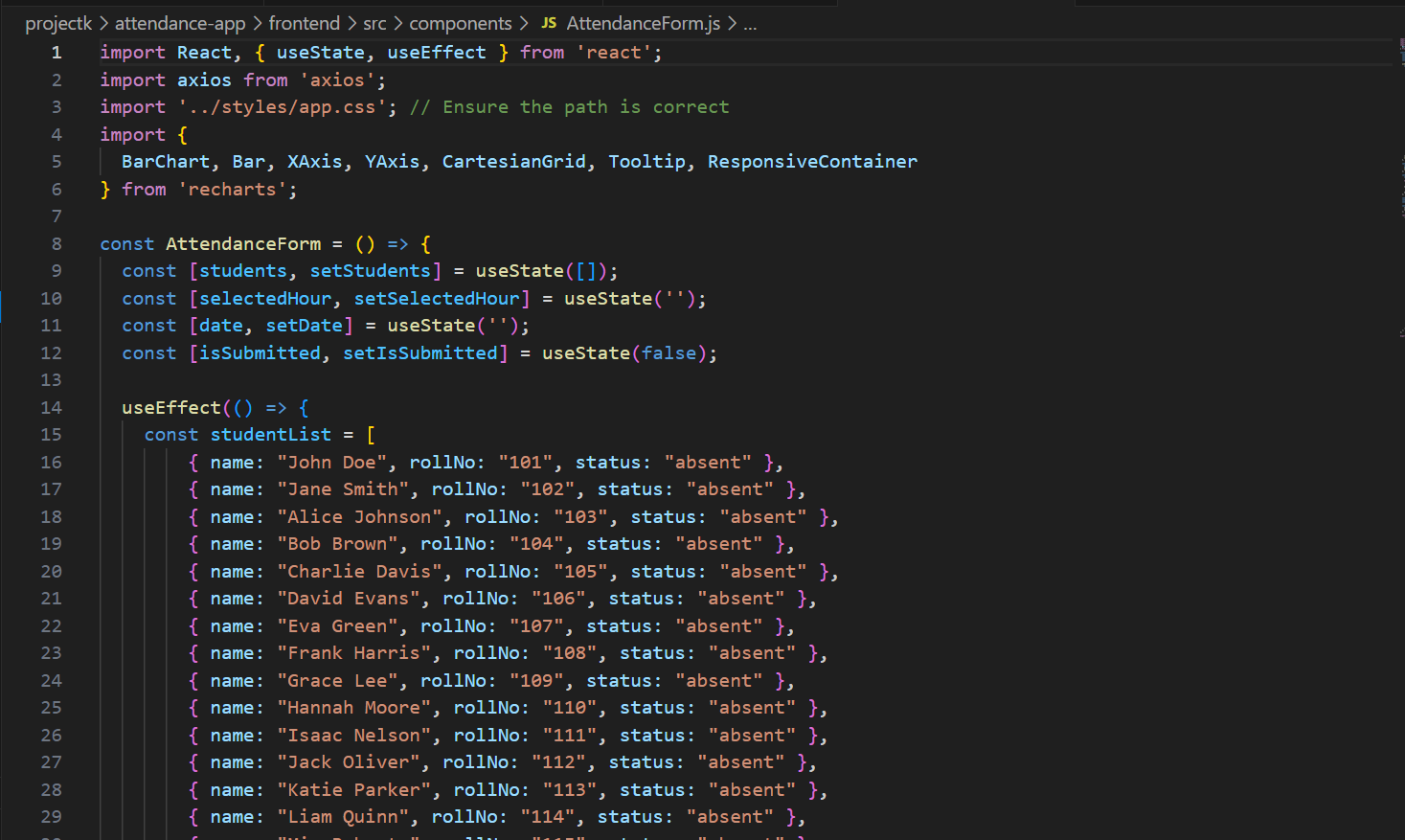
2.attendence.js



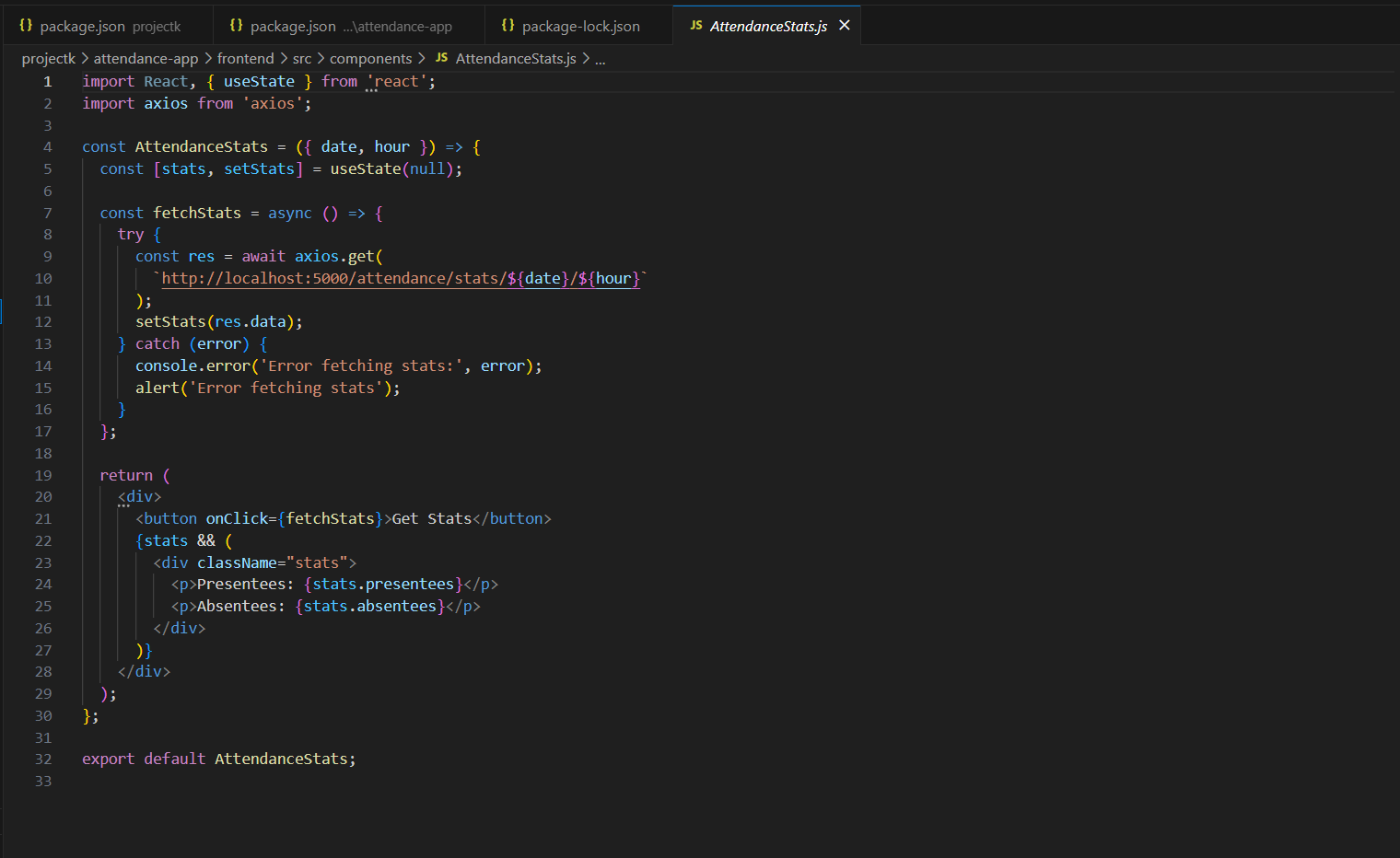
3.server.js



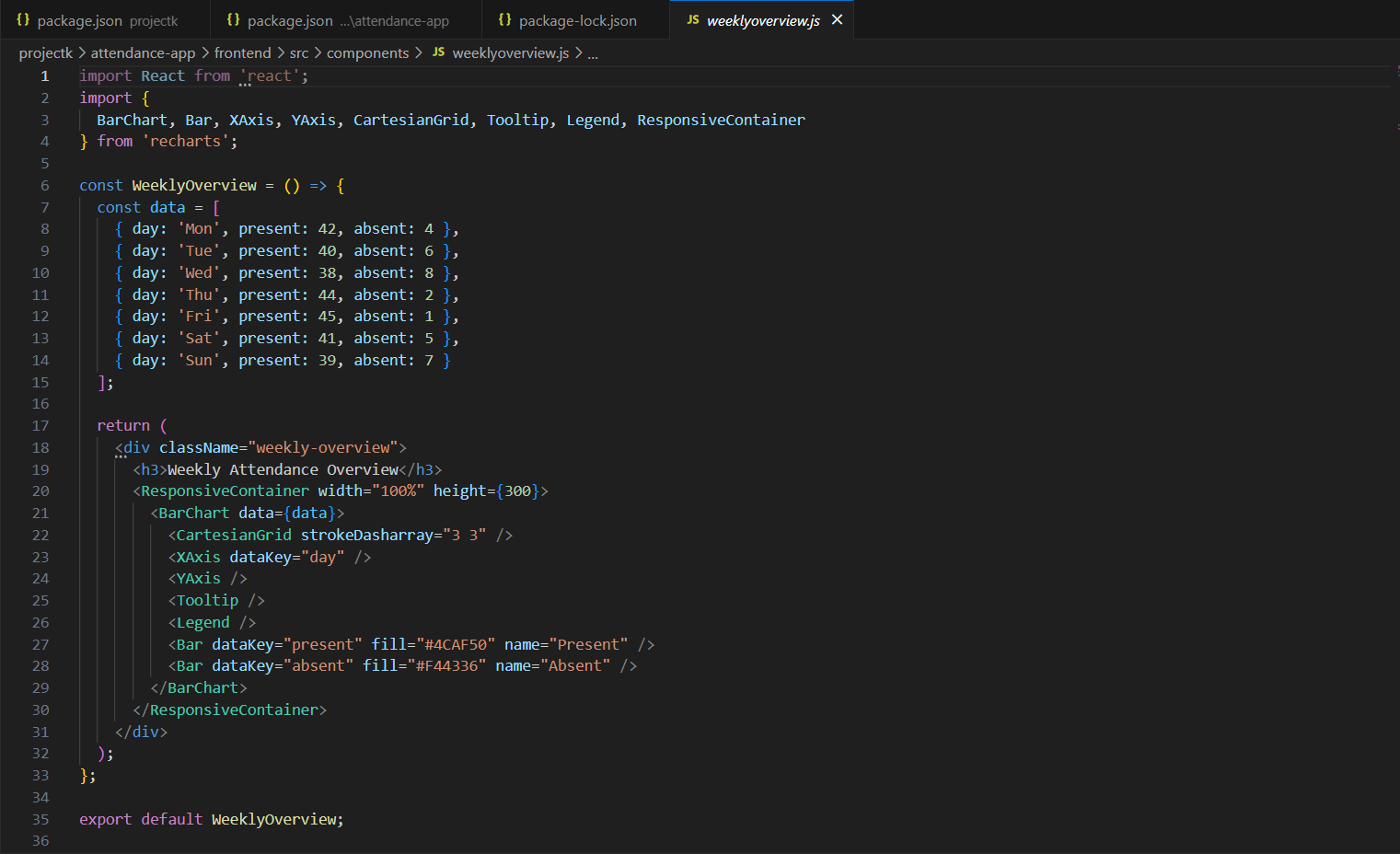
4.attendenceform.js



5.attendencestat.js



6.weeklyoverview.js



# CONCLUSION

The **Attendance Management System** is a robust, scalable, and user-friendly solution designed to automate the process of tracking and managing attendance records. Developed primarily using Python and supported by libraries such as pandas, matplotlib, and seaborn, the system efficiently processes CSV-based datasets and generates insightful reports and visualizations.

By eliminating the limitations of traditional manual attendance methods, this system helps institutions and organizations improve operational efficiency, ensure accurate record-keeping, and reduce administrative workload. The data-driven approach enables easy detection of irregular attendance patterns, identification of defaulters, and overall improvement in monitoring student or employee participation.

The use of Python offers numerous benefits, including:

* Simplified data handling and transformation.
* Quick development of backend logic.
* Easy integration with visualization tools and potential web frameworks (like Streamlit or Flask).
* Flexibility to scale or migrate to more advanced systems in the future.

Moreover, the modular architecture of the system allows for future enhancements, such as:

* Real-time attendance tracking using biometric or QR code technology.
* Integration with databases like SQLite or MySQL for better data management.
* Deployment as a web-based platform or mobile application for broader accessibility.
* Automated alert systems for low attendance via email or SMS notifications.

In conclusion, the Attendance Management System fulfills its core objective of providing a reliable and accurate way to manage attendance records while offering flexibility for future upgrades. It serves as a practical and educational example of how Python can be used to solve real-world problems through automation, data analysis, and visualization.