Online Attendance Through Web Application

Mrs. P. Saranya[1], Mr. K. V. Naveen Kumar Reddy[2], Mr. K. Omkar Sai[3], Mr. K. Dileep[4], Mr. M. Pothanesh[5], Mr. K. Prasanth[6]

[1]Assistant Professor, Department of Electronics and Communication Engineering, Sri Venkatesa Perumal college of Engineering and Technology, Andhra Pradesh, India

[2,3,4,5,6] UG Scholar, Department of Electronics and Communication Engineering, Sri Venkatesa Perumal college of Engineering and Technology, Andhra Pradesh, India

ABSTRACT

The College Attendance System is a web-based application designed to streamline and automate attendance management within educational institutions. Traditional manual attendance tracking methods are often time-consuming, error-prone, and lack real-time accessibility. This system leverages web technologies, including HTML, CSS, JavaScript, and Google Apps Script, coupled with Google Sheets as a backend database, to provide an efficient and accurate solution.

The system features a secure faculty portal for digital attendance marking, a student portal for real-time attendance record viewing, and automated report generation. By eliminating manual data entry and facilitating instant data retrieval, the system significantly improves administrative efficiency and reduces errors. The use of Google Sheets ensures data integrity and accessibility from any device with internet connectivity. This project aims to enhance transparency, improve accuracy, and streamline the attendance process, ultimately contributing to a more organized and effective learning environment.

Keyword: - *HTML, CSS, JAVASCRIPT, GOOGLE SHEETS, DATABASE*

1. INTRODUCTION

Attendance management is a crucial aspect of educational institutions and organizations. Traditional attendance systems rely on manual processes such as paper-based registers or spreadsheets, which are prone to errors, manipulation, and inefficiencies. In recent years, the advancement of web technologies has provided an opportunity to develop automated attendance systems that improve accuracy, security, and ease of use.

A web-based attendance system allows teachers to mark attendance digitally, store records in a database, and generate reports with minimal effort. Such systems provide real-time access to attendance data, reducing paperwork and improving administrative efficiency. Additionally, integrating cloud-based solutions, such as Google Sheets, enables seamless data storage and retrieval.

2. Online Attendance Through Web Application

The limitations of the existing manual attendance system necessitate the development of a more efficient, automated, and secure system. The proposed Digital Attendance System is a web-based solution designed to streamline attendance management by leveraging modern technologies such as HTML, CSS, JavaScript, Google Sheets, and Google Apps Script.

It ensures real-time attendance tracking, transparency, and accessibility by integrating cloud-based storage. Faculty members can securely log in and mark attendance digitally, while students can only view their records without modification privileges. The integration of Google Sheets as a cloud database enables seamless data synchronization, reducing the administrative burden and minimizing errors associated with manual record-keeping.

By implementing role-based access control, the system enhances security by preventing unauthorized access or modifications. The proposed methodology aims to provide an intuitive interface for both faculty and students, ensuring ease of use and efficiency. This chapter discusses the system architecture, workflow, security mechanisms, and overall advantages of the proposed approach.

2.1 Implementation

2.1.1 UML Diagrams

Agile software developers, be they application developers or data engineers, must minimally have an understanding of object orientation if they are to be effective. This includes understanding basic concepts such as inheritance, polymorphism, and object persistence. Furthermore, all developers should have a basic understanding of the industry-standard Unified Modeling Language (UML). A good starting point is to understand what I consider to be the core UML diagrams – use case diagrams, sequence diagrams, and class diagrams – although as I argued in An Introduction to Agile Modeling and Agile Documentation you must be willing to learn more models over time.

Let's begin with what I consider to be the three core UML diagrams for developing business software: UML use case diagrams, UML sequence diagrams , and UML class diagrams. These are the diagrams that you will see used the most in practice – use case diagrams to overview usage requirements, sequence diagrams to analyze the use cases and map to your classes, and class diagrams to explore the structure of your object-oriented software (what I like to refer to as your object schema). These three diagrams will cover 80% of your object modeling needs when building a business application using object technology.

This diagram depicts several use cases, actors, their associations, and optional system boundary boxes. A use case describes a sequence of actions that provide a measurable value to an actor and is drawn as a horizontal ellipse. An actor is a person, organization, or external system that plays a role in one or more interactions with your system. Actors are drawn as stick figures. Associations between actors and classes are indicated in use-case diagrams, a relationship exists whenever an actor is involved with an interaction described by a use case. Associations between actors and use cases are modeled as lines connecting them to one another, with an optional arrowhead on one end of the line indicating the direction of the initial invocation of the relationship.

Use Case Diagram:

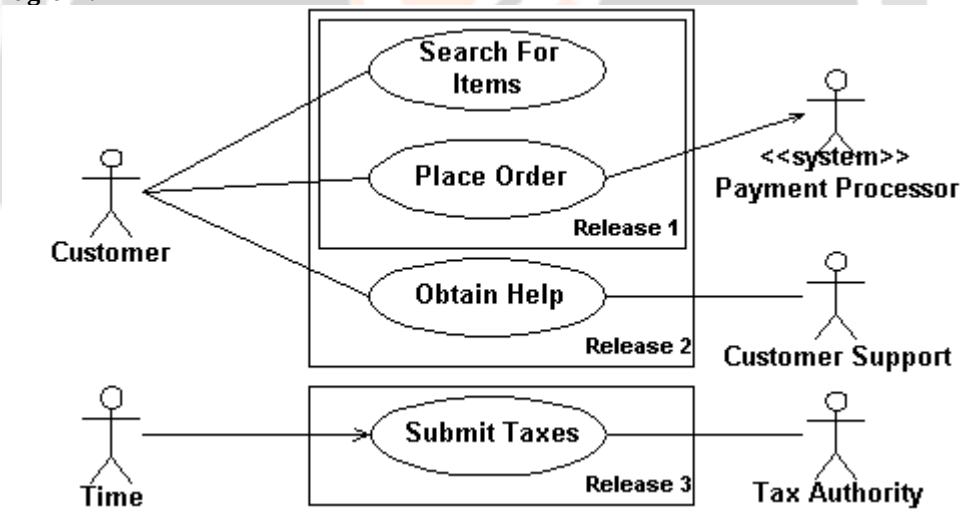

Fig 2.1: Sample Use Case Diagram

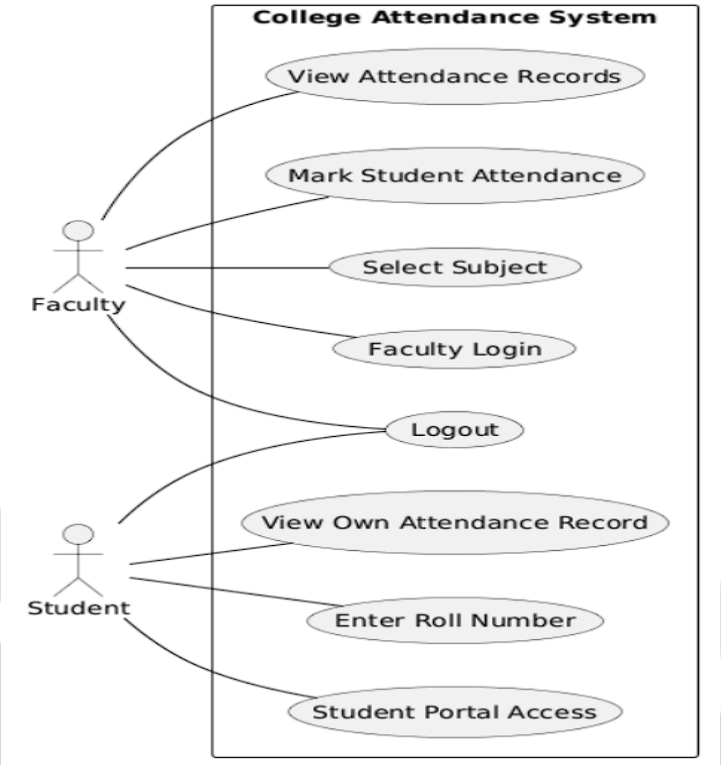

Fig 2.2: Use Case – interaction between Faculty , Student and Actions

2.2 Experimental Results

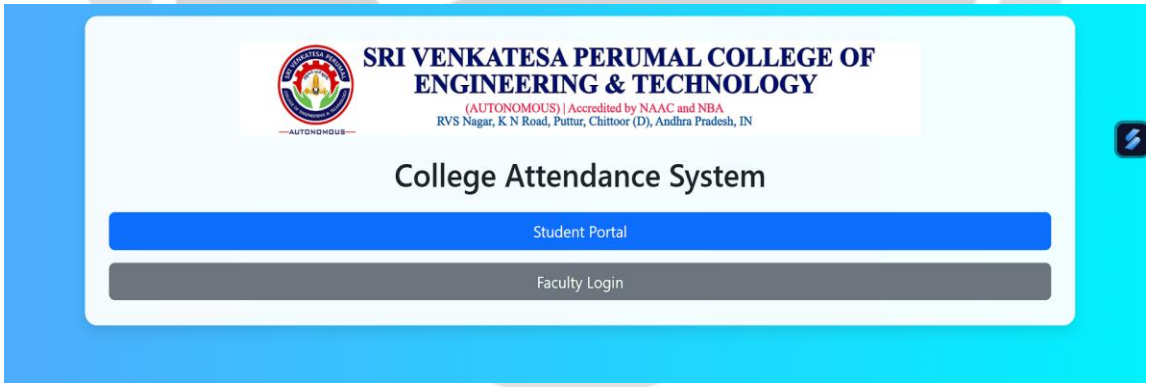

Fig. 2.2.1: Landing Page

The main menu screen, as depicted in the fig.2.21, serves as the central navigation hub for the College Attendance System. It is the first screen users encounter upon accessing the web application and provides access to the two primary functionalities of the system: the Student Portal and the Faculty Login.

Functionality:

- **Student Portal Button:** Clicking this button redirects the user to the Student Portal, where students can view their attendance records.
- **Faculty Login Button:** Selecting this button leads to the Faculty Login screen, where faculty members can enter their credentials to access the attendance marking and management functionalities.

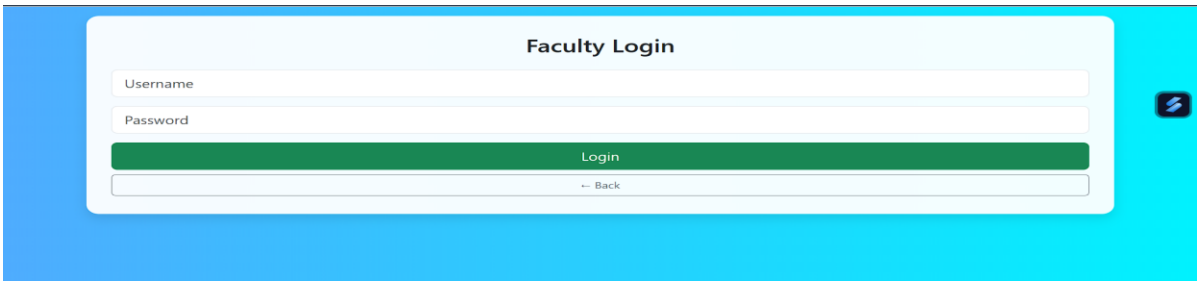

Fig. 2.2.2: Authentication With Password

The Faculty Login screen, as depicted in the fig.2.2.2, is designed to provide secure access for faculty members to the attendance management functionalities of the College Attendance System. This screen requires faculty members to enter their credentials to proceed.

Functionality:

- **Username and Password Input:** Faculty members enter their credentials into the respective fields.
- **Login Submission:** Clicking the 'Login' button triggers the system to verify the entered credentials against the stored faculty database.
- **Navigation:** The '← Back' button allows users to return to the main menu without logging in.

Fig. 2.2.3: List Of Subjects

The 'Select Subject' screen, as depicted in the fig.2.2.3, is presented to faculty members after successful login. It displays a list of subjects for which the faculty member is responsible, allowing them to choose the subject for which they want to mark or view attendance.

Functionality:

- **Subject Selection:** Faculty members can click on a subject button to proceed to the attendance marking or viewing screen for that particular subject.
- **Navigation:** The '← Back' button allows faculty members to return to the previous screen if they do not wish to select a subject.

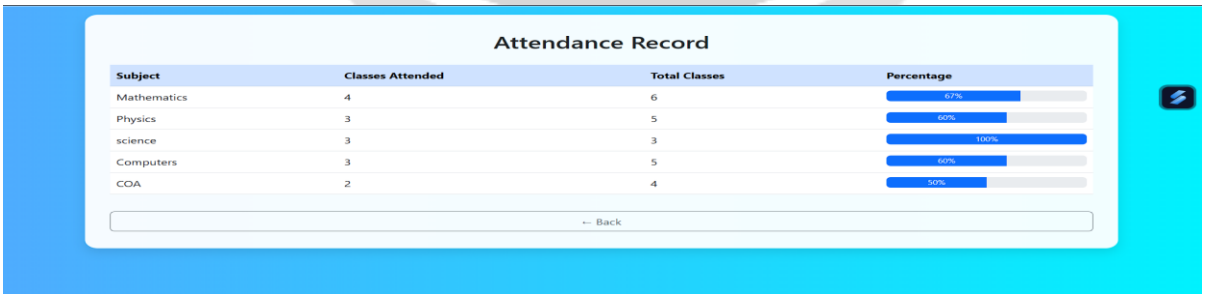

Fig. 2.2.4: Attendance records

The 'Attendance Record' screen, as depicted in the fig.2.2.4, is presented to students after they enter their roll number in the Student Portal. It displays a summary of their attendance records for each subject, showing the number of classes attended, total classes held, and the calculated percentage.

Functionality:

- **Attendance Display:** The table displays the student's attendance records for each subject.

- **Percentage Visualization:** The progress bar in the 'Percentage' column provides a visual representation of the student's attendance percentage.
- **Navigation:** The '← Back' button allows students to return to the previous screen.

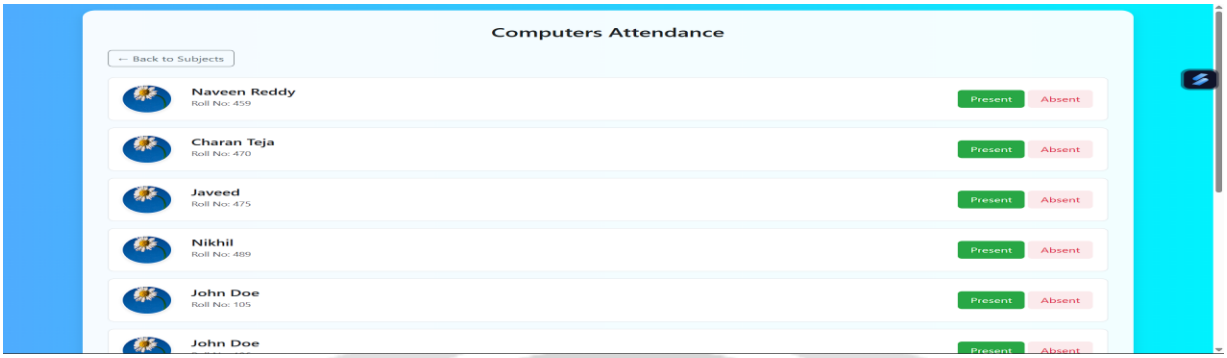

Fig. 2.2.5: Attendance form

The 'Attendance Marking' screen, as depicted in the fig.2.2.5, is presented to faculty members after they select a subject from the 'Select Subject' screen. It displays a list of students enrolled in the selected subject, allowing the faculty member to mark their attendance as 'Present' or 'Absent'.

Functionality:

- **Attendance Marking:** Faculty members can mark a student as 'Present' or 'Absent' by clicking the respective buttons.
- **Navigation:** The '← Back to Subjects' button allows faculty members to return to the subject selection screen.
- **Data Saving:** The "saved to the DB" label, shows the faculty that the data has been saved.

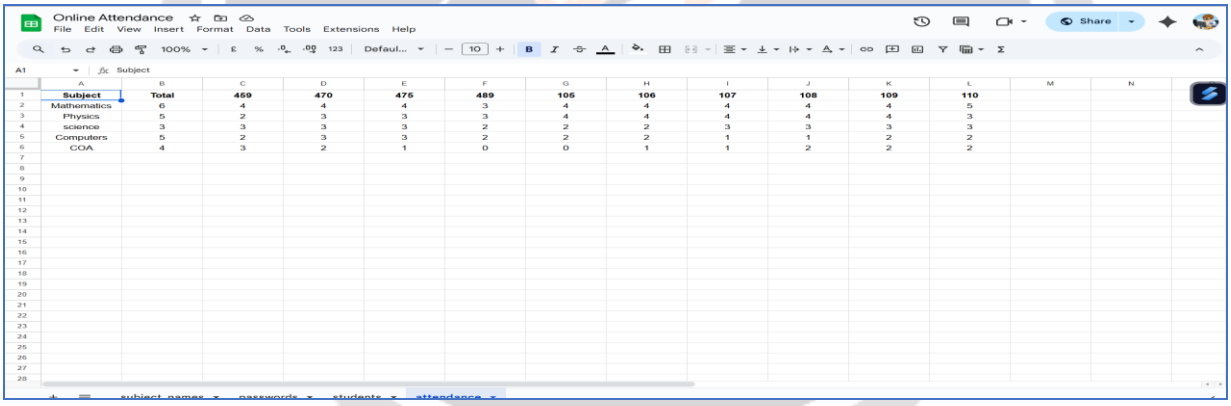

Fig. 2.2.6: Student Attendance database

The image depicts the 'attendance' sheet within the Google Sheets document used as the backend database for the College Attendance System. This sheet is responsible for storing and managing attendance records for each subject and student.

Key Elements:

- **Columns:**
  - **Subject (Column A):** Lists the names of the subjects.
  - **Total (Column B):** Indicates the total number of classes held for each subject.
  - **Student Roll Numbers (Columns C onward):** Each subsequent column represents a student, identified by their roll number. The values within these columns indicate the number of classes attended by each student for the corresponding subject.
- **Rows:**
  - **Header Row (Row 1):** Contains the column headers.
  - **Subject Rows (Rows 2 onward):** Each row represents a subject and its associated attendance data.

- **Data Representation:** The attendance data is represented numerically, indicating the number of classes attended by each student for each subject.

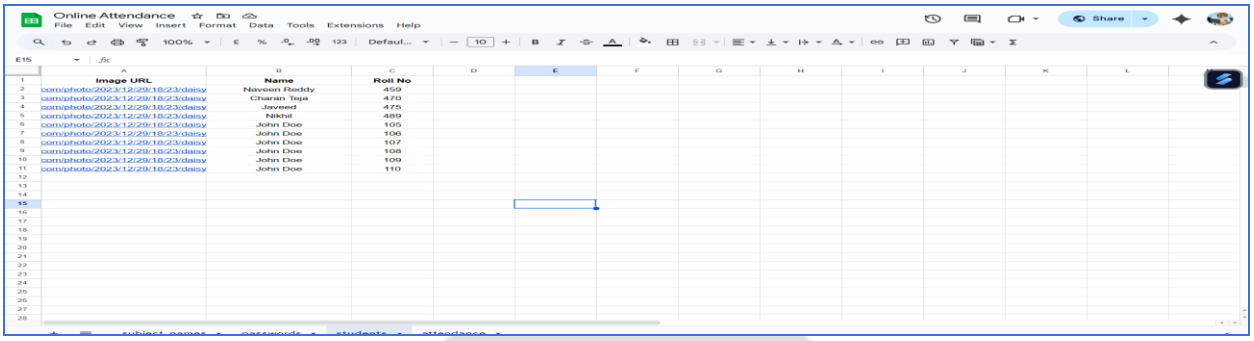

Fig.2.2.7: Student details in database

The image depicts the 'students' sheet within the Google Sheets document used as the backend database for the College Attendance System. This sheet is responsible for storing and managing student information, which is used to display student details in the attendance marking and attendance record screens.

Key Elements:

- **Columns:**
  - **Image URL (Column A):** Stores the URL of the student's image.
  - **Name (Column B):** Stores the student's full name.
  - **Roll No (Column C):** Stores the student's unique roll number.
- **Rows:**
  - **Header Row (Row 1):** Contains the column headers.
  - **Student Rows (Rows 2 onward):** Each row represents a student and their associated information.
- **Data Representation:** The student data is represented as text and URLs.

3 CONCLUSIONS

The implementation of an online attendance system through a web application has revolutionized the way attendance is managed in educational institutions and organizations. By replacing traditional manual methods with a digital system, this solution enhances efficiency, reduces human errors, and provides real-time data accessibility. The developed system utilizes modern web technologies such as React.js for the frontend, Node.js and Express.js for the backend, and MongoDB integrated with Google Sheets for database management. These technologies work together to create a seamless, automated, and user-friendly experience for both students and teachers. One of the major advantages of this system is its ability to eliminate common issues associated with manual attendance tracking.

Traditional paper-based methods often result in data loss, errors in record-keeping, and the possibility of proxy attendance. By implementing secure authentication mechanisms such as JWT-based authentication, unauthorized access is prevented, ensuring that only authorized users can mark or view attendance records. Additionally, real-time synchronization with Google Sheets ensures that data is always up-to-date and accessible from any location, providing flexibility to institutions and teachers.

In conclusion, the online attendance system developed in this study represents a significant step toward modernizing attendance management. By leveraging the power of web technologies, the system offers a cost-effective, secure, and efficient solution for tracking attendance in academic and professional settings. The successful implementation of this system highlights the potential of digital transformation in streamlining administrative processes, improving accuracy, and enhancing user experience. Future advancements in AI, mobile applications, and automation will further elevate the system’s functionality, ensuring that it continues to meet the evolving needs of institutions worldwide.

4. REFERENCES

[1]. URL: [Insert the main URL for Google Apps Script documentation, e.g., https://developers.google.com/apps-script]

Description: The official documentation for the Google Apps Script platform, which was used for developing the backend logic of the attendance system, including data handling, user authentication, and interaction with Google Sheets.
Google Sheets API Documentation (via Apps Script):

[2]. URL: [Insert the specific URL for the Spreadsheet Service in Apps Script, e.g., https://developers.google.com/apps-script/reference/spreadsheet]
Description: Documentation detailing how Google Apps Script interacts with Google Sheets for data storage, retrieval, and manipulation within the attendance system.
HTML (Hypertext Markup Language):

[3]. URL: [Insert a general URL for HTML documentation, e.g., https://developer.mozilla.org/en-US/docs/Web/HTML]
Description: The standard markup language used for creating the structure and content of the web-based user interface for both faculty and student portals.
CSS (Cascading Style Sheets):

[4]. URL: [Insert a general URL for CSS documentation, e.g., https://developer.mozilla.org/en-US/docs/Web/CSS]
Description: The stylesheet language used for controlling the presentation, layout, and visual appearance of the web interface, enhancing user experience.

[5]. URL: [Insert a general URL for JavaScript documentation, e.g., https://developer.mozilla.org/en-US/docs/Web/JavaScript]
Description: The scripting language used for adding interactivity and dynamic behavior to the web interface, such as form handling, navigation, and data display.
Bootstrap (CSS Framework):

[6]. URL: [Insert the URL for Bootstrap documentation, e.g., https://getbootstrap.com/docs/5.3/getting-started/introduction/]
Description: A popular CSS framework utilized for creating a responsive and visually consistent user interface across different devices. The styling and layout components used in the student and faculty portals are based on Bootstrap.