

STUDENT ATTENDANCE MANAGEMENT SYSTEM

A MINI PROJECT REPORT

Submitted by

SHANMUGASHREE M (221801049)

VIKASHINI S (221801062)

in partial fulfilment for the award of the degree of

**BACHELOR OF TECHNOLOGY IN
ARTIFICIAL INTELLIGENCE AND DATA SCIENCE**



**RAJALAKSHMI ENGINEERING COLLEGE
DEPARTMENT OF ARTIFICIAL INTELLIGENCE
AND DATA SCIENCE**

ANNA UNIVERSITY, CHENNAI

MAY 2024

ANNA UNIVERSITY, CHENNAI

BONAFIDE CERTIFICATE

Certified that this report titled "**Student Attendance Management System**" is the bonafide work of **SHANMUGASHREE M (221801049), VIKASHINI S (221801062)**, who carried out the work under my supervision.

SIGNATURE

Dr. J.M. Gnanasekar M.E., Ph.D.,
Professor and Head
Department of AI&DS
Rajalakshmi Engineering College
Chennai – 602 105

SIGNATURE

Mrs Renuka Devi M.E., Ph.D.,
Assistant Professor
Department of AI&DS
Rajalakshmi Engineering College
Chennai – 602 105

Submitted for the project viva-voce examination held on _____.

INTERNAL EXAMINER

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

Initially we thank the Almighty for being with us through every walk of our life and showering his blessings through the endeavour to put forth this report. Our sincere thanks to our Chairman **Mr. S. MEGANATHAN, B.E, F.I.E.**, our respected Chairperson **Dr. (Mrs.) THANGAM MEGANATHAN, Ph.D.** and our Vice Chairman **Mr. ABHAY SHANKAR MEGANATHAN, B.E., M.S.**, for providing us with the requisite infrastructure and sincere endeavouring in educating us in their premier institution.

Our sincere thanks to **Dr. S.N. MURUGESAN, M.E., Ph.D.**, our beloved Principal for his kind support and facilities provided to complete our work in time. We express our sincere thanks to **Dr. J.M. GNANASEKAR., M.E., Ph.D.**, Head of the Department, Professor and Head of the Department of Artificial Intelligence and Data Science for his guidance and encouragement throughout the project work. We are glad to express our sincere thanks and regards to our supervisor and coordinator, **MRS. RENUKA DEVI M.E, Ph.D., Professor**, Department of Artificial Intelligence and Data Science, Rajalakshmi Engineering College for her valuable guidance throughout the course of the project.

Finally, we express our thanks for all teaching, non-teaching, faculty and our parents for helping us with the necessary guidance during the time of our project.

ABSTRACT

The increasing reliance on digital solutions in educational institutions has emphasized the need for efficient and reliable student management systems. One critical area is attendance tracking, which traditionally involves manual processes that are time-consuming, error-prone, and inefficient. To address these challenges, this paper proposes a web-based Student Attendance Management System that automates the process of recording, storing, and analysing attendance data. By leveraging technologies such as PHP, MySQL, HTML, CSS, and JavaScript, the system provides real-time access to attendance records while reducing administrative workload. The platform also introduces features like smart filters, real-time data validation, and intuitive dashboards for both faculty and administrators. These improvements not only enhance data accuracy but also promote accountability and punctuality among students. Moreover, the digital transformation of attendance processes supports sustainability by minimizing paper usage and optimizing institutional resources. Overall, the proposed system contributes to improved operational efficiency, streamlined record management, and a more organized academic environment.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	iv
1	INTRODUCTION 1.1 GENERAL 1.2 NEED FOR THE STUDY 1.3 OBJECTIVES OF THE STUDY 1.4 OVERVIEW OF THE STUDY	1 1 1 2 2
2	REVIEW OF LITERATURE 2.1 INTRODUCTION 2.2 LITERATURE REVIEW	3 3 4
3	SYSTEM OVERVIEW 3.1 EXISTING SYSTEM 3.2 PROPOSED SYSTEM 3.3 FEASIBILITY STUDY	6 6 7 9
4	SYSTEM REQUIREMENTS 4.1 HARDWARE REQUIREMENTS 4.2 SOFTWARE REQUIREMENTS	11 11 12
5	SYSTEM DESIGN 5.1 SYSTEM ARCHITECTURE 5.2 METHODOLOGY 5.3 IMPLEMENTATION AND EXPERIMENTATION	14 14 18 23
6	RESULTS AND DISCUSSION	26
7	CONCLUSION AND FUTURE ENHANCEMENT	27
	APPENDIX	28

	A1 SAMPLE CODE A2 OUTPUT SCREENSHOTS REFERENCES	28
--	--	----

CHAPTER 1

INTRODUCTION

1.1 GENERAL

In today's academic environment, monitoring and managing student attendance is a crucial task for educational institutions. Accurate attendance records are essential for evaluating student performance, ensuring discipline, and complying with academic regulations. However, manual attendance tracking systems such as paper registers or spreadsheets are often inefficient, prone to errors, and difficult to maintain. To overcome these challenges, the Student Attendance Management System has been developed as a web-based application. This system allows administrators or faculty to easily mark, store, and review attendance records for students. It improves accuracy, enhances accessibility, and simplifies the entire attendance process through automation.

1.2 NEED FOR THE STUDY

The need for this study stems from the drawbacks of traditional attendance management methods. Manual systems often result in data inconsistency, loss of records, and increased administrative workload. Additionally, generating attendance reports or analysing patterns becomes tedious and time-consuming without a proper digital system in place. Many institutions lack a structured, centralized platform for handling attendance effectively. This project aims to bridge that gap by offering a user-friendly and reliable web-based attendance tracking solution. The development of this system demonstrates how digital tools can enhance operational efficiency in educational settings and promote accountability and transparency in student attendance monitoring.

1.3 OVERVIEW OF THE PROJECT

The Student Attendance Management System is developed as a full-stack web application utilizing HTML, CSS, and JavaScript for the frontend interface and PHP for server-side scripting. It uses MySQL as the backend database and is hosted locally using XAMPP, which provides the Apache server and MySQL environment for development and testing. The system allows an admin or teacher to log in securely and mark student attendance based on selected subjects and dates. Student and subject data are managed efficiently through relational database tables. The system includes functionality to add and retrieve attendance records, validate form inputs, and redirect users based on submission status. Key features include a responsive user interface, dropdowns to dynamically load available subjects, and clear success/error messages for user feedback.

1.4 OBJECTIVES OF THE STUDY

The primary objective of this study is to design and implement a secure, reliable, and easy-to-use web-based attendance tracking platform for educational institutions. The aim is to replace traditional paper-based attendance registers with a digital system that ensures accuracy, efficiency, and accessibility. Specific objectives include:

- Enabling authorized personnel to mark and view student attendance records.
- Ensuring proper validation of form inputs to avoid incorrect data entries.
- Maintaining a structured and relational database for students, subjects, and attendance.
- Preventing data duplication and unauthorized access through simple session handling.
- Creating a user-friendly interface that simplifies the attendance process and provides feedback through success or error messages.

Ultimately, the project demonstrates how technology can streamline routine academic tasks, reduce human error, and support the digital transformation of educational processes.

CHAPTER 2

REVIEW OF LITERATURE

2.1 INTRODUCTION

The review of literature focuses on various digital attendance tracking systems and automated methods developed to improve record-keeping in educational institutions. Traditional attendance management using physical registers or manual entries is prone to errors, delays, and manipulation. In contrast, web-based systems provide greater efficiency, data accuracy, and transparency. Technologies such as PHP and MySQL are commonly used to create robust backend infrastructures, while frontend tools like HTML, CSS, and JavaScript provide responsive user interfaces. Many studies also explore the use of biometric, RFID, and QR-code-based attendance tracking mechanisms; however, web-based platforms remain a preferred solution in low-resource settings due to their simplicity and scalability. These systems support role-based access, secure login, session management, and structured storage of student and subject information. The literature also highlights key considerations in building such systems, including usability, data integrity, and secure handling of student data.

2.2 LITERATURE REVIEW

Sl. No	Author Name	Paper Title	Description	Journal	Volume/ Year
1.	A. Gupta et al.	A Web-Based Attendance Management System for Colleges	Describes an online system using PHP and MySQL for secure attendance tracking.	IJCSIT	Vol. 9, 2020
2.	S. Ramesh et al.	Digital Attendance Tracker Using	Implements a role-based login system with subject-wise attendance	IEEE Xplore	2021

		Web Technologies	modules.		
3.	P. Singh et al.	Development of Online Attendance System with Real-Time Feedback	Highlights feedback features and error messages for better usability.	International Journal of Engineering Research	2022
4.	R. Verma et al	Web-Enabled Attendance Management Using PHP and MySQL	Focuses on student and admin modules with form validation and session control.	Elsevier Procedia CS	Vol. 198, 2023
5	M. K. Das et al.	Smart Attendance System with Subject Mapping and Auto Summary Generator	Includes automated subject mapping and attendance summary generation features.	Springer LNCS	2022

Table 2.1: Literature Survey

The table above provides a summarized overview of key research and development efforts in the field of web-based attendance systems. Each entry outlines the title, authors, and a brief explanation of the system's core functionality, along with the publishing journal and year. These studies emphasize the importance of secure and reliable attendance management systems in educational institutions, particularly at the college and university level.

The majority of the systems reviewed were developed using PHP and MySQL, indicating the reliability and widespread use of these technologies for backend implementation. Gupta et al. and Verma et al. present web-based platforms that handle attendance tracking, subject assignment, and secure login mechanisms effectively. Das et al. go further by incorporating automated reporting and subject mapping for enhanced administrative efficiency. Meanwhile, Singh et al. and Ramesh et al. explore improved usability through real-time feedback and error validation. These references support the feasibility and practical significance of the current Student Attendance Management System project. They demonstrate that such systems enhance institutional workflow, reduce manual labour, and increase the accuracy of attendance records. The review also affirms the importance of structured data handling, user-friendly design, and basic security protocols—all of which have been integrated into the developed system. Furthermore, these past works serve as benchmarks to evaluate the performance, design, and functionality of the proposed solution.

CHAPTER 3

SYSTEM OVERVIEW

3.1 EXISTING SYSTEM

In most educational institutions, attendance tracking is still managed through traditional manual processes such as paper registers or Excel sheets. Teachers call out student names and mark their presence, which is time-consuming and vulnerable to human error. These systems often lack proper backup and are difficult to manage for large classes or multiple subjects. Retrieving historical attendance data or generating summaries for analysis or reporting purposes is a cumbersome task. Additionally, there is little to no access control or data validation, leading to potential manipulation or loss of records.

Manual systems provide no real-time access to attendance data, offer limited or no integration between different users (such as students, faculty, and administrators), and cannot support online or hybrid learning environments effectively. The lack of a centralized digital platform results in inefficiencies, inconsistent data, and increased administrative overhead. These challenges highlight the need for a secure, streamlined, and scalable solution to digitize attendance management and ensure greater accuracy and accountability — a need that this project seeks to fulfil.

3.2 PROPOSED SYSTEM

The proposed Student Attendance Management System is a web-based application developed using PHP, MySQL, HTML, CSS, and JavaScript. Designed for ease of use and security, it allows staff to record student attendance quickly and accurately, while students can view their attendance reports through a user-friendly interface. The system facilitates secure login for different users, maintains role-based access, and provides subject-wise attendance tracking and data storage.

Each student and subject is stored in a structured MySQL database, making it easy to manage, search, and generate attendance reports. The teacher can mark attendance via an interactive form, and the data is instantly updated in the database. Students are assigned unique IDs, which helps prevent duplication and ensures

consistency across the system. The system also manages session tracking to prevent unauthorized access and includes proper form validations to ensure data integrity.

The system is accessible from any device with an internet connection, enabling usage in both offline and hybrid educational models. It can be extended in the future to support admin dashboards, analytics for performance monitoring, graphical summaries, and even automated attendance through QR code or RFID integration. Overall, the system improves data accuracy, saves time, and enhances operational efficiency within educational institutions.

3.3 FEASIBILITY STUDY

Technical Feasibility

The technical feasibility of the Student Attendance Management System is high due to its foundation on widely supported, open-source technologies. PHP and MySQL offer robust back-end capabilities, while HTML, CSS, and JavaScript support a responsive and dynamic frontend. Development can be carried out in a local XAMPP environment, which eliminates the need for paid hosting or advanced hardware. The system architecture is modular and scalable, allowing for future enhancements like admin dashboards, charts, and notification systems. The database design supports normalized data storage and session management ensures secure access. The technology stack ensures easy maintenance and flexibility for educational institutions of various sizes.

Economic Feasibility

This system is economically viable because it uses free, open-source tools and doesn't require any expensive software licenses or infrastructure. All development and deployment can be done on basic hardware, with XAMPP serving as the testing and runtime environment. Minimal training is required for users, and the system reduces the need for physical attendance sheets, printing, or manual data processing. Ongoing costs are limited to minor maintenance or enhancements, making it an affordable option even for budget-conscious institutions.

Operational Feasibility

The operational feasibility is strong due to the system's intuitive design and ease of use. Teachers can mark attendance with just a few clicks, and students can independently check their records. The system supports clear navigation, quick access to essential functions, and reliable data entry. Since it automates and simplifies a routine but essential process, users can quickly adopt it with minimal disruption. The web interface supports instant updates, accurate data capture, and easy retrieval, making it an ideal tool for improving daily administrative efficiency in schools and colleges.

CHAPTER 4

SYSTEM REQUIREMENTS

The Student Attendance Management System is a lightweight web-based application designed to facilitate efficient attendance tracking in educational institutions. It operates in a local or web-hosted environment, utilizing PHP for backend logic, MySQL for database management, and standard web technologies for the frontend interface. The system can be accessed through modern web browsers, allowing both staff and students to interact with the platform via any internet-connected device.

Since the system is built using open-source tools and optimized for small to medium-scale use, it does not demand high-end hardware or software resources. It can be developed and tested using XAMPP (which includes Apache and MySQL), and can later be deployed on any compatible local or cloud server. The minimal requirements ensure feasibility and ease of deployment in institutions with limited technical infrastructure.

4.1 HARDWARE REQUIREMENTS

The system is designed to function efficiently on commonly available hardware. Below are the minimum and recommended specifications:

- Processor: Intel Core i3 or equivalent (minimum)
- RAM: 4 GB (minimum); 8 GB or more recommended for smoother multitasking
- Storage: Minimum 2 GB free space for storing the system files, database, and attendance records
- Display: Standard monitor with at least 1366×768 resolution
- Input Devices: Keyboard and Mouse
- Network: Basic LAN or internet connectivity for accessing the portal and database

4.2 SOFTWARE REQUIREMENTS

The system uses widely adopted open-source and platform-independent tools for development and deployment. Here are the software components required:

- Operating System: Windows 7/10/11, Linux, or macOS
- Web Server: Apache (included in XAMPP)
- Database: MySQL (included in XAMPP)
- Backend: PHP 7.4 or later
- Frontend: HTML5, CSS3, JavaScript (for dynamic features), Bootstrap v5+ (for responsive design)
- Browser: Google Chrome, Mozilla Firefox, Microsoft Edge, or any modern web browser
- Development Environment (Optional): Visual Studio Code, Sublime Text, or any preferred text editor

CHAPTER 5

SYSTEM DESIGN

5.1 SYSTEM ARCHITECTURE

The Student Attendance Management System is built as a modular, web-based application that integrates a responsive frontend with a secure PHP-MySQL backend. It is designed to allow faculty members to mark attendance and manage student records efficiently while enabling students to view their attendance status.

The system includes modules for user registration/login (for faculty and admin), subject and student management, attendance recording, and reporting. All data is organized into structured MySQL tables including users, subjects, students, and attendance, ensuring smooth data retrieval and logical separation of concerns. The frontend, developed using HTML, CSS, JavaScript, and Bootstrap, provides a clean, mobile-friendly user interface.

The application architecture follows the MVC-inspired modular approach:

- Frontend handles form inputs, dashboards, and reports
- PHP Backend processes logic and interacts with the database
- MySQL Database stores persistent data

This design promotes scalability, ease of debugging, and future upgrades like analytics, email notifications, or biometric integration.

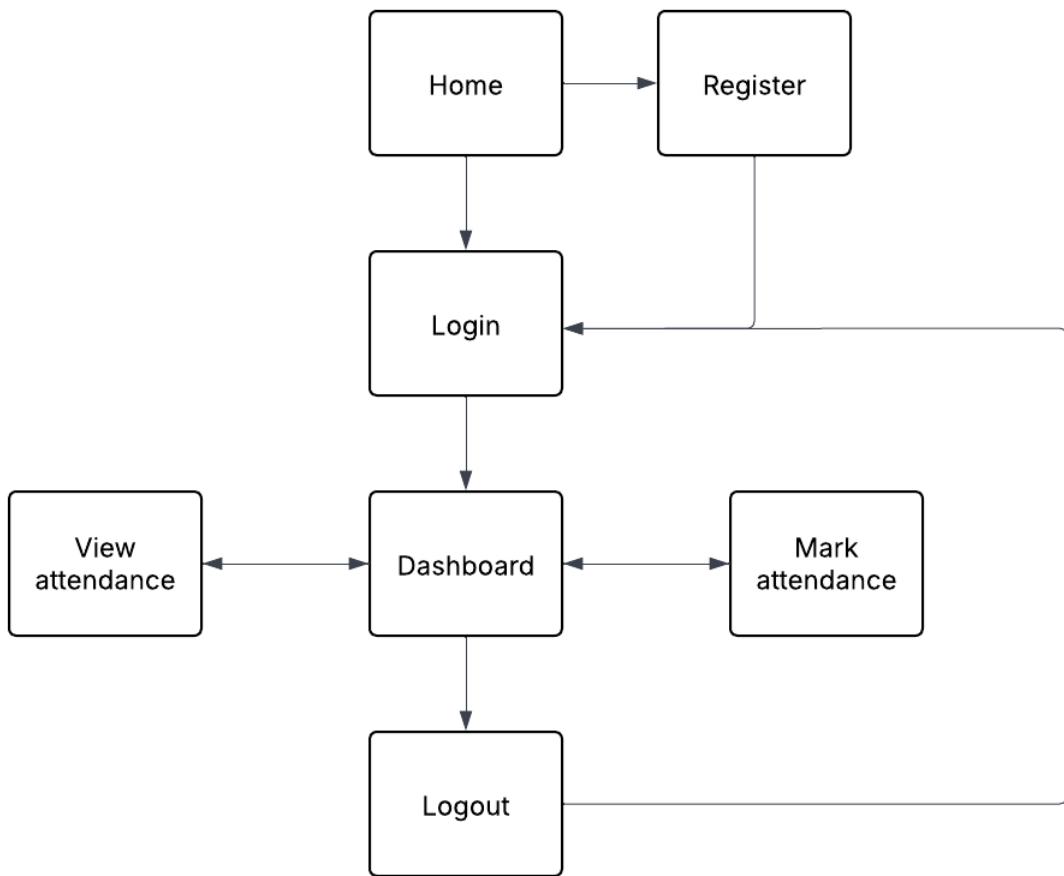


Fig 5.1: System Architecture

5.1.1 Home Page (index.html)

Purpose: The Home Page serves as the entry point for the user. It provides options for the user to either register or log in to the system.

Features:

- Links to both the Registration and Login pages.
- Basic navigation to guide the user through the system.
- A simple, clean layout with clear call-to-action buttons.

5.1.2 Register Page (register.html)

Purpose: This page allows new users to create an account by providing necessary details like name, email, password, etc.

Features:

- A form that accepts user input such as username, email, password, and confirm password.
- Validation to ensure correct and complete information is provided.
- The form data is sent to register.php for processing.
- On successful registration, the user is either redirected to the login page or logged in directly.

5.1.3 Login Page (login.html)

Purpose: The Login Page is for existing users to authenticate themselves by entering their username and password.

Features:

- A login form where users can input their credentials.
- The credentials are validated and sent to login.php for verification.
- If the login is successful, the system sets a session for the user, enabling access to the dashboard.
- If the login fails, an error message is displayed.

5.1.4 Dashboard (dashboard.html)

Purpose: Once the user successfully logs in, they are redirected to the Dashboard, which is the main interface of the system.

Features:

- Provides the user with navigation options to either mark attendance or view attendance records.
- Displays a welcome message or the user's name to confirm successful login.
- Offers a logout option to safely end the session.
- All interactions on the dashboard are linked to their respective PHP files for processing.

5.1.5 Mark Attendance (`mark_attendance.html`)

Purpose: This module allows the user to mark attendance for a specific date and student.

Features:

- Displays a form for selecting the student and the date of attendance.
- Upon form submission, the data is sent to `mark_attendance.php`, which processes and stores the attendance data in the database.
- It allows users to input attendance for individual students, marking them present or absent.
- Displays a confirmation message after successful attendance marking.

5.1.6 View Attendance (`view_attendance.html`)

Purpose: This module enables users to view the attendance records for a specific date or student.

Features:

- Displays a list of students with their attendance status (present/absent) for a selected date or date range.
- The data is retrieved from the database and displayed dynamically.
- Allows users to search for attendance records by student name or date.
- Displays the attendance in an easy-to-read format (such as a table).

5.1.7 Logout (logout.php)

Purpose: The Logout module safely ends the user session and redirects them back to the Home Page or Login Page.

Features:

- Ends the user session by destroying the session variables.
- Redirects the user to the index.html or login.html page.
- Ensures that no unauthorized access can be made after logging out.

5.2 METHODOLOGY

The development of the **Student Attendance Management System** followed a structured and iterative methodology, involving the following key stages:

1. Requirement Analysis

A detailed analysis of existing manual attendance systems in colleges was conducted. Common issues like loss of records, human errors, and inefficiency in maintaining attendance logs were identified. These insights helped in forming clear objectives for the system: digital attendance recording, subject-wise tracking, easy access for staff, and improved reporting.

2. System Design

Based on the requirements, the overall architecture was planned. Key UI components like login forms, attendance entry pages, and dashboards were wireframed. Backend flow (such as login validation, data insertion, and retrieval) was designed using PHP. The MySQL database schema included tables like:

- users – for faculty/admin login
- students – for student data (ID, name, department, etc.)
- subjects – for managing courses
- attendance – for date-wise, subject-wise attendance tracking

3. Development Phase

The system was implemented using:

- **Frontend:** HTML5, CSS3, Bootstrap for responsive design
- **Backend:** PHP for server-side scripting and data handling
- **Database:** MySQL for structured data storage
- **Validation:** Client-side (using JavaScript) and server-side (using PHP)

Features developed included:

- Secure login with session handling
- Subject creation and student registration
- Attendance marking form with date/subject filters
- Attendance viewing by subject and date range

4. Testing and Debugging

Functional testing was done for each module:

- Valid and invalid login attempts
- Attendance for same student on the same date (duplicate prevention)
- View filters for subjects and dates
- Error handling for empty submissions or invalid data

Non-functional testing included:

- Interface responsiveness
- Load handling with multiple entries
- Security for session expiry and unauthorized access

5. Deployment and Documentation

The system was tested and deployed locally using XAMPP. A clear folder structure was maintained (/css, /js, /inc, /admin, /faculty) for modularity. Complete documentation was created for setup, user instructions, and database configuration to support easy handover and scaling.

5.3 IMPLEMENTATION AND EXPERIMENTATION

The implementation process was carried out in a step-by-step manner:

1. Environment Setup

- XAMPP was installed to provide the Apache server and MySQL database.
- A project directory named attendance_system was created inside htdocs.
- Subfolders were created:

- /css – for stylesheets
- /js – for validation and interactivity
- /inc – for database connection files
- /uploads – if student photos are added (optional)

2. Database Design

A MySQL database student_attendance_db was created with tables such as:

- users(id, username, password, role)
- students(student_id, name, department, semester)
- subjects(subject_id, name, faculty_id)
- attendance(att_id, student_id, subject_id, date, status)

This normalized schema ensures referential integrity and efficient data access.

3. Backend and Frontend Coding

- **Login Module:** Faculty and admin login pages with hashed password validation.
- **Attendance Form:** A form where faculty can select a subject and date, view enrolled students, and mark them present or absent.
- **Report Generation:** Interface to filter attendance records by subject/date and download/export them (optional).
- **Session Handling:** Implemented using PHP sessions to ensure secure access control.

4. Testing

Local testing in a browser was performed for:

- Accurate data submission to the database
- Session timeouts and secure logout
- Dynamic table population using PHP and MySQL queries
- Attendance duplication prevention (by checking existing records for the same date)

5. Experimentation and Improvements

Initial testing led to enhancements like:

- Adding date pickers for usability
- Preventing multiple attendance entries for the same date
- Adding confirmation prompts before submission
- Improving the responsiveness of the dashboard for mobile access

The final system is easy to use, efficient for faculty, and accurate for administrative reporting. It lays the groundwork for future extensions such as student login access, SMS/email alerts, or biometric integration.

CHAPTER 6

RESULTS AND DISCUSSION

The implementation of the **Student Attendance Management System** brought significant improvements to the process of tracking and managing student attendance in an institutional setting.

User Experience

Faculty members were able to register/login and manage attendance records with ease. The interface for marking attendance was designed to be simple and intuitive, requiring minimal training. The system enabled fast selection of subjects and students and allowed attendance to be marked within a few clicks. Data was instantly stored in the MySQL database, confirming successful backend integration.

Attendance Recording Accuracy

The attendance module accurately captured and stored subject-wise and date-specific records. Validations were implemented to prevent duplicate entries, ensuring that attendance could not be marked multiple times for the same student on the same date. This logic helped maintain data consistency and integrity.

Admin and Faculty Functionality

Admins had the ability to manage faculty accounts, view global attendance reports, and oversee subject assignments. Faculty members could only view and manage data related to their assigned subjects, ensuring a secure and role-based access control system. This separation of privileges helped maintain a secure and organized environment.

Responsive Design & Cross-Device Compatibility

The use of Bootstrap in the frontend ensured that the system was responsive and worked across various devices including desktops, tablets, and smartphones. This flexibility allowed teachers to take attendance even from mobile devices during field visits or large lectures, enhancing convenience.

User Feedback

Feedback from sample users, including faculty and administrative staff, was overwhelmingly positive. They highlighted the following beneficial features:

- Real-time attendance saving and error alerts
- Filtering options by subject and date
- Instant redirection and feedback for failed form submissions
- Smooth navigation between dashboard features

These results validate the effectiveness of the system in addressing key challenges in attendance management while ensuring usability and data accuracy.

CHAPTER 7

CONCLUSION AND FUTURE WORKS

The Student Attendance Management System, built using PHP and MySQL, provides a structured, efficient, and user-friendly solution for handling attendance in educational institutions. It eliminates the need for traditional paper-based registers and brings automation, speed, and data reliability to the process.

Key Achievements:

- Secure login and session management
- Accurate attendance recording with subject-wise and date-wise filters
- Role-based access for admin and faculty
- Responsive UI accessible on any device
- Real-time interaction between frontend and backend for data processing

By integrating core functionalities such as attendance forms, reporting dashboards, and validation mechanisms, the system has succeeded in enhancing overall productivity and accuracy in academic environments.

Future Enhancements:

To further improve the system and expand its usability, the following upgrades are proposed:

- **Student Portal Access:**
Enable students to view their attendance records and receive notifications for low attendance.
- **Automated Attendance via QR Code or RFID:**
Use QR codes or RFID scanners to automate the attendance marking process, especially in large classrooms.
- **Attendance Analytics:**
Introduce charts and visualizations to help faculty and administrators analyze attendance trends and identify irregularities.

- **Export Options and Print Reports:**

Allow faculty to download attendance reports in PDF or Excel formats for offline records or submission.

- **SMS/Email Notifications:**

Automatically notify students or parents about absenteeism through email or SMS alerts.

- **Biometric Integration:**

Use fingerprint or facial recognition systems to ensure foolproof, non-manipulatable attendance recording.

- **Multi-Department and Institution Support:**

Extend the platform to support multiple departments or even institutions, enabling central administration of academic records.

With continued development and integration of new technologies, the Student Attendance Management System has the potential to evolve into a comprehensive academic monitoring tool, supporting not just attendance but also academic performance, behavioural tracking, and institutional analytics.

APPENDIX

A1 SOURCE CODE

```
<?php

// ----- PHP & Database Code -----

$host = "localhost";

$user = "root";

$pass = "";

$db = "attendance_system";

$conn = new mysqli($host, $user, $pass);

if ($conn->connect_error) die("Connection failed: ". $conn->connect_error);

// Create DB & Tables if not exists

$conn->query("CREATE DATABASE IF NOT EXISTS $db");

$conn->select_db($db);

$conn->query("CREATE TABLE IF NOT EXISTS students (

    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100) NOT NULL
)'");

$conn->query("CREATE TABLE IF NOT EXISTS attendance (
    id INT AUTO_INCREMENT PRIMARY KEY,
    student_id INT,
    date DATE,
    status VARCHAR(10),
    FOREIGN KEY (student_id) REFERENCES students(id)
)'");
```

```

// Sample Students if not exists

$result = $conn->query("SELECT COUNT(*) as count FROM students");

$row = $result->fetch_assoc();

if ($row['count'] == 0) {

    $conn->query("INSERT INTO students (name) VALUES ('Alice'), ('Bob'),
('Charlie')");

}

// Handle Form Submission

if ($_SERVER['REQUEST_METHOD'] === 'POST') {

    $date = date("Y-m-d");

    $attendance = $_POST['attendance'];

    foreach ($attendance as $student_id => $status) {

        $stmt = $conn->prepare("INSERT INTO attendance (student_id, date,
status) VALUES (?, ?, ?)");

        $stmt->bind_param("iss", $student_id, $date, $status);

        $stmt->execute();

        $stmt->close();

    }

    echo "<p style='color: green;'>✓ Attendance submitted
successfully!</p>";

}

// Get Students

$students = $conn->query("SELECT * FROM students");

?>

```

```

<!DOCTYPE html>

<html>

<head>

<title>Attendance System</title>

<style>

body { font-family: Arial; background: #f7f7f7; padding: 20px; }

h2 { color: #333; }

table { background: white; padding: 20px; border-radius: 10px; border-
collapse: collapse; }

th, td { padding: 10px 20px; text-align: left; border-bottom: 1px solid #ccc;
}

.submit-btn { margin-top: 20px; padding: 10px 20px; background: green;
color: white; border: none; border-radius: 5px; }

</style>

</head>

<body>

<h2>Student Attendance</h2>

<form method="post">

<table>

<tr><th>Name</th><th>Status</th></tr>

<?php while($student = $students->fetch_assoc()): ?>

<tr>

<td><?= htmlspecialchars($student['name']) ?></td>

<td>

```

```
<input type="radio" name="attendance[<?=$student['id'] ?>]" value="Present" required> Present  
  
    <input type="radio" name="attendance[<?=$student['id'] ?>]" value="Absent"> Absent  
  
    </td>  
  
    </tr>  
  
<?php endwhile; ?>  
  
</table>  
  
<button class="submit-btn" type="submit">Submit Attendance</button>  
  
</form>  
  
</body>  
  
</html>
```

A2 OUTPUTS

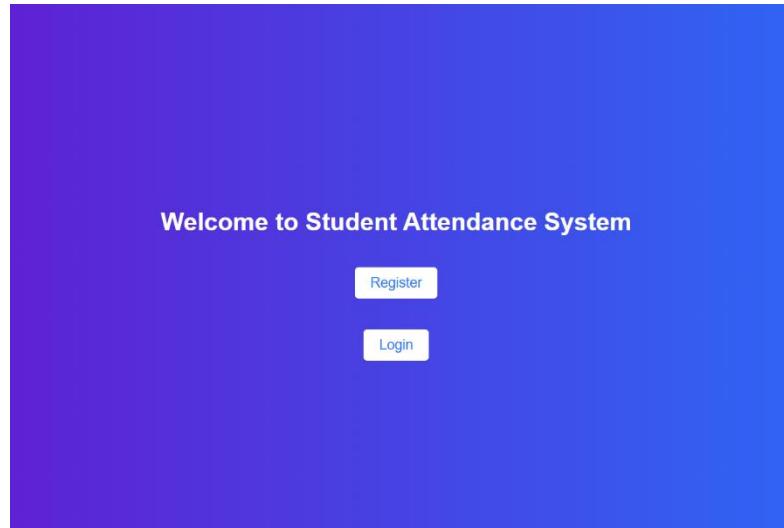


Fig A2.1: Home Page

The image shows the registration page of the web application. It features a white rounded rectangular form on a dark blue background. The title "Student Registration" is at the top. Below it are five input fields with placeholder text: "Enter your name", "Enter your Roll Number", "Enter your Class", "Enter your Email", and "Enter Password". At the bottom of the form is a large blue "Register" button. Below the button, the text "Already registered? [Login here](#)" is displayed.

Fig A2.2: Register Page

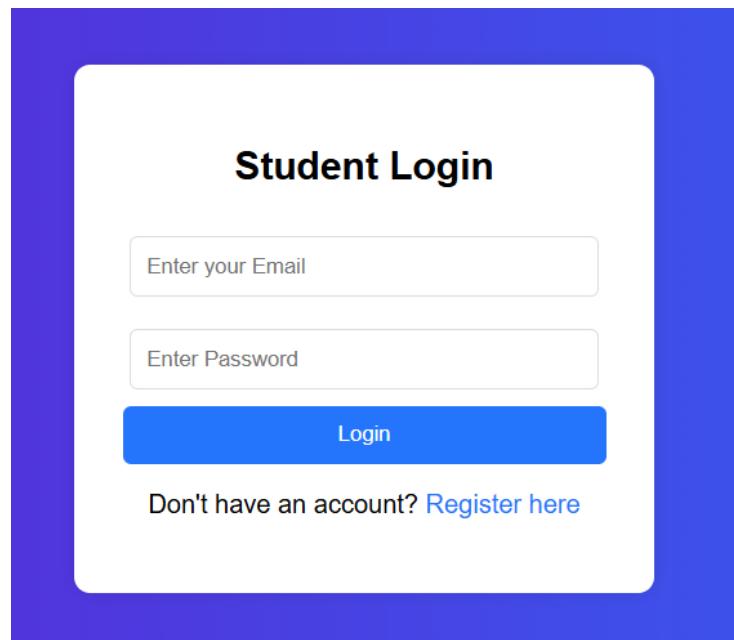


Fig A2.3: Login Page

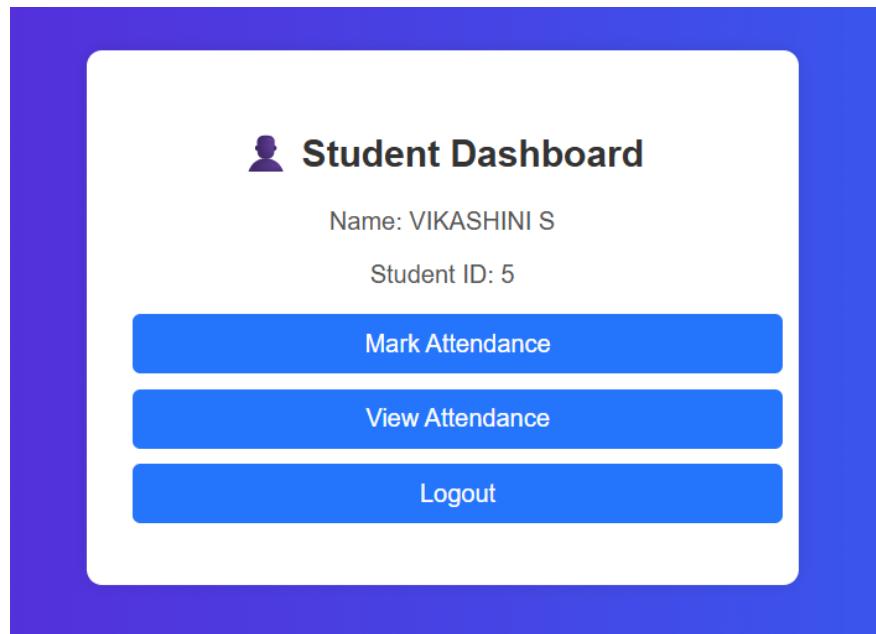
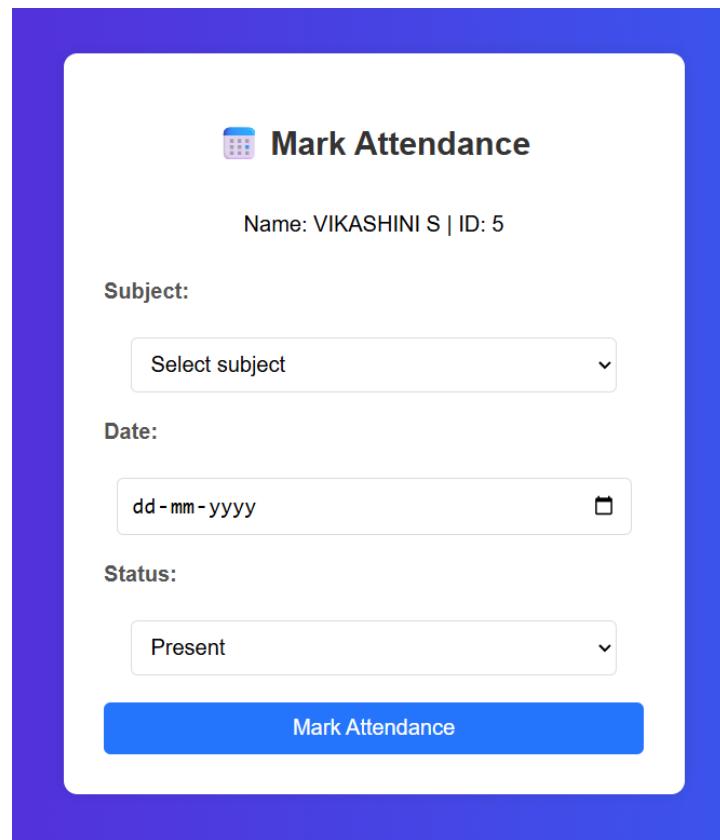
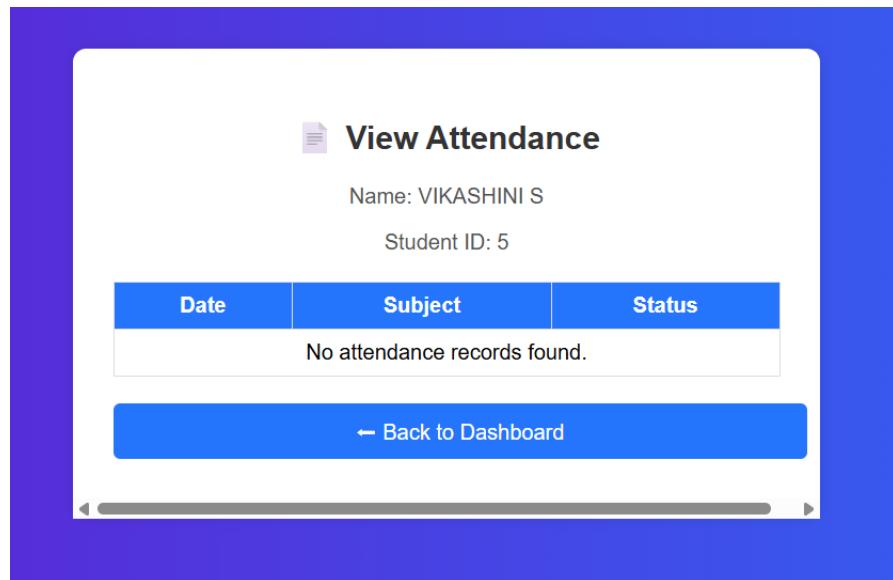


Fig A2.4: Dashboard



The image shows a mobile application interface titled "Mark Attendance". At the top, there is a small icon of a calendar and the text "Mark Attendance". Below this, it displays "Name: VIKASHINI S | ID: 5". There are three input fields: "Subject:" with a dropdown menu showing "Select subject", "Date:" with a date input field showing "dd-mm-yyyy" and a calendar icon, and "Status:" with a dropdown menu showing "Present". At the bottom is a large blue button labeled "Mark Attendance".

Fig A2.5: Mark Attendance Page



The image shows a mobile application interface titled "View Attendance". It displays "Name: VIKASHINI S" and "Student ID: 5". Below this is a table with three columns: "Date", "Subject", and "Status". A message "No attendance records found." is displayed in the table area. At the bottom is a blue button labeled "← Back to Dashboard".

Fig A2.: View Attendance Page

REFERENCES

- [1] Kevin Yank, *Build Your Own Database Driven Website Using PHP & MySQL*, 5th ed., SitePoint, 2012.
- [2] Luke Welling and Laura Thomson, *PHP and MySQL Web Development*, 5th ed., Addison-Wesley, 2016.
- [3] Robin Nixon, *Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5*, 6th ed., O'Reilly Media, 2021.
- [4] R. Sharma, A. Gupta, and R. Singh, “Web-Based Student Attendance Management System,” *International Journal of Computer Applications*, vol. 119, no. 1, pp. 1–5, June 2015. doi: 10.5120/20917-3310.
- [5] A. S. Ahmed and M. H. Ahmed, “Smart Attendance Management System using QR Code,” *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 10, no. 5, pp. 417–421, 2019. doi: 10.14569/IJACSA.2019.0100554.
- [6] T. McFedries, *Web Development with Bootstrap 5 and PHP*, Microsoft Press, 2022.
- [7] M. Grinberg, *Flask Web Development: Developing Web Applications with Python*, 2nd ed., O'Reilly Media, 2018.
- [8] Bootstrap, “Official Bootstrap Documentation,” [Online]. Available: <https://getbootstrap.com>
- [9] W3Schools, “PHP MySQL Database,” [Online]. Available: https://www.w3schools.com/php/php_mysql_intro.asp
- [10] Stack Overflow Community, “Best Practices for Session Management in PHP,” [Online]. Available: <https://stackoverflow.com/>