

A Mini Project Report
on
ATTENDANCE MANAGEMENT SYSTEM

Submitted to CMREC HYDERABAD

In Partial Fulfillment of the requirements for the Award of Degree of

**BACHELOR OF TECHNOLOGY
IN
COMPUTER SCIENCE AND ENGINEERING**

Submitted By

D.Ajith (218R1A0512)

E.Chaithanya (218R1A0514)

A.Sindhu (228R5A0501)

Under the Esteemed guidance of

Mrs.M.Swathi

Assistant Professor

Department of CSE



Department of Computer Science & Engineering
CMR ENGINEERING COLLEGE
(UGC AUTONOMOUS)

(Approved by AICTE, NEW DELHI, Affiliated to JNTU, Hyderabad)
Kandlakoya, Medchal Road, R.R. Dist. Hyderabad-501 401)

2024-2025

CMR ENGINEERING COLLEGE

(UGC AUTONOMOUS)

(Accredited by NBA, Approved by AICTE NEW DELHI, Affiliated to JNTU, Hyderabad)

Kandlakoya, Medchal Road, Hyderabad-501 401

Department of Computer Science & Engineering



CERTIFICATE

This is to certify that the project entitled “ATTENDANCE MANAGEMENT SYSTEM” is a bonafide work carried out by

D.Ajith (218R1A0512)

E.Chaithanya (218R1A0514)

A.Sindhu (228R5A0501)

in partial fulfillment of the requirement for the award of the degree of **BACHELOR OF TECHNOLOGY** in **COMPUTER SCIENCE AND ENGINEERING** from CMR Engineering College, affiliated to JNTU, Hyderabad, under our guidance and supervision.

The results presented in this project have been verified and are found to be satisfactory. The results embodied in this project have not been submitted to any other university for the award of any other degree or diploma

Internal Guide

Mrs. M. Swathi
Assistant Professor
CSE Department
CMRC

Mini Project Coordinator

Mr. S. Kiran Kumar
Assistant Professor
CSE Department
CMREC

Head of the Department

Dr. Sheo Kumar
Professor & H.O.D
CSE Department
CMREC

Examiner

DECLARATION

This is to certify that the work reported in the present project entitled “**ATTENDANCE MANAGEMENT SYSTEM**” is a record of bonafide work done by us in the Department of Computer Science and Engineering, CMR Engineering College, JNTU Hyderabad. The reports are based on the project work done entirely by us and not copied from any other source. We submit our project for further development by any interested students who share similar interests to improve the project in the future.

The results embodied in this project report have not been submitted to any other University or Institute for the award of any degree or diploma to the best of our knowledge and belief.

D.Ajith (218R1A0512)

E.Chaithanya (218R1A0514)

A.Sindhu (228R5A0501)

ACKNOWLEDGMENT

We are extremely grateful to **Dr. A. Srinivasula Reddy**, Principal and **Dr. Sheo Kumar**, HOD, **Department of CSE, CMR Engineering College** for their constant support.

I am extremely thankful to **Mrs. M. Swathi**, Internal Guide, Assistant Professor, Department of CSE, for her constant guidance, encouragement and moral support throughout the project.

I will be failing in duty if I do not acknowledge with grateful thanks to the authors of the references and other literatures referred in this Project.

I thank **S Kiran Kumar** Mini Project Coordinator for his constant support in carrying out the project activities and reviews.

I express my thanks to all staff members and friends for all the help and co-ordination extended in bringing out this project successfully in time.

Finally, I am very much thankful to my parents who guided me for every step.

D.Ajith (218R1A0512)

E.Chaithanya (218R1A0514)

A.Sindhu (228R5A0501)

CONTENTS

TOPIC	PAGENO
ABSTRACT	I
INTRODUCTION...	1
1.1 Introduction of the project.....	1
1.2 Purpose of the project	2
1.3 Existing system&Disadvantages.....	2
1.4 Proposed system with features.....	2
2.LITERATURE SURVEY.....	3.
3.SOFTWARE REQUIREMENTS ANALYSIS	4
3.1 SDLC.....	4
3.2 System study.....	5
3.3 Modules and their functionalities.....	6
4. Feasibility study.....	10
4.1 Feasibility Report.....	10
4.2 Technical Feasibility.....	11
4.3 Operational Feasibility.....	11
4.4 Economical Feasibility.....	12
5.System Requirements Specification.....	13
5.1 Requirement Specification.....	13
5.2 Hardware Requirements.....	13
5.3 Software Requirements.....	14
6.System Design.....	26
6.1 UML Diagrams.....	27
6.2 Data Flow Diagram.....	32
7. CODING AND IMPLENTATION.....	33
7.1Sample code.....	33
8.SYSTEM TESTING.....	67
9.OUTPUT SCREENS.....	69

10.CONCLUSION.....72

11.REFERENCES.....73

ABSTRACT

Over the years the manual attendance management has been carried across most of educational institutions. To overcome the problems of manual attendance, I have developed “web based attendance Management System”. Attendance Management System is based on web server, which can be implemented on any computer. In This application, PHP is server side language, MySQL and PHP is used as back-end design and HTML, CSS and JavaScript are used as front-end tools. The system communicates with database residing on a remote server. It calculates automatically, the attendance percentage of students without any manual paper-based work. The system facilitates the end users with interactive design and automated processing of attendance management.

Attendance management is a critical function in educational institutions and organizations alike, ensuring accountability, monitoring of student or employee participation, and compliance with attendance policies. The Attendance Management System (AMS) proposed in this project aims to automate and digitalize the process of recording and tracking attendance. This system will offer administrators, teachers, and students a robust platform to manage attendance efficiently. Key features include secure user authentication, multiple methods for recording attendance (such as manual entry and biometric scanning), real-time data updates, comprehensive reporting capabilities, and automated notifications to relevant stakeholders. The AMS seeks to improve accuracy, reduce administrative overhead, enhance transparency, and facilitate data-driven decision-making. By implementing this system, institutions can streamline operations, optimize resource allocation, and foster a more accountable and productive environment.

1.INTRODUCTION

1.1. Introduction to Project:

Existing attendance management system is manual, in which everything has to be maintained with the manual process. Attendance records are maintained in register, book, files, etc. These records have to be kept safe and secure without being lost or damaged. So, a secure and safe place is required to keep these records, so that no any liquid substance or any animal like mouse, etc. could harm these data. Generally, they are stored in cup boards. For example, if someone has marked attendance on register, and unfortunately a register has come in contact with water or any liquid substance or any animal like mouse, it will lose the data which is stored in register. And if anyone has to check attendance record, then he or she has to find the register first, then he or she can access it.

This system leverages modern web technologies to streamline the attendance management process. It is based on a web server, making it accessible from any computer with an internet connection. The application uses PHP as the server-side language, with MySQL as the database management system. The front-end is designed using HTML, CSS, and JavaScript to ensure an interactive and user-friendly experience. The Web-Based Attendance Management System automates the calculation of attendance percentages for students, eliminating the need for manual, paper-based work. It communicates seamlessly with a database hosted on a remote server, ensuring data is accurately recorded and easily accessible. This innovative solution not only enhances efficiency but also provides a reliable and convenient platform for managing attendance in educational institutions.

An Attendance Management System (AMS) represents a pivotal tool in modern organizational efficiency, streamlining the process of tracking and managing attendance for employees or students. By replacing traditional manual methods with automated solutions, AMS enhances accuracy in attendance recording, reduces administrative burdens, and provides real-time insights into attendance patterns. Utilizing various technologies such as biometric scanners, RFID cards, or mobile apps, AMS not only simplifies the task of clocking in and out but also facilitates seamless integration with payroll systems and generates comprehensive reports for informed decision-making. Ultimately, AMS fosters a more productive and compliant environment by ensuring adherence to attendance policies and labour regulations, thereby optimizing resource allocation and operational effectiveness.

1.2. Purpose of the Project

The primary purpose of the Web-Based Attendance Management System is to modernize and improve the attendance tracking process in educational institutions. By eliminating manual, paper-based methods, the system enhances efficiency and accuracy in recording attendance. It offers a digital, automated solution that calculates attendance percentages, ensuring reliable data storage and accessibility from any computer with internet connectivity. The system's interactive design provides a user-friendly interface for educators and administrators, facilitating easier management of attendance records and supporting administrative tasks with secure, centralized data storage.

1.3.Existing System

The existing system of attendance management typically involves a manual process where teachers or administrative staff record student attendance using paper-based registers or spreadsheets. This method is time-consuming, prone to errors, and inefficient in terms of data retrieval and analysis. Additionally, it lacks real-time updates and can be easily manipulated, leading to inaccuracies in attendance records. Communication of attendance information to students and parents often relies on manual reports or phone calls, further adding to the inefficiency. Overall, the manual attendance management system is outdated and struggles to meet the needs of modern educational institutions in terms of accuracy, efficiency, and data security.

1.4. Proposed system

Now as we all know, almost everything can be done online. Like Money transfer, Shopping, Booking, Teaching, Data sharing, Admissions, Job search, etc. And so many other activities are done with the help of internet. So with the easy access and use of internet, we are going to take this existing attendance system on advance level. We are going to develop an online platform with high security so that the same process could be done easily without the waste of time, afford, and energy. So firstly, teachers/faculties are required to register on attendance management system. Once registration is done, faculty can easily add/remove students and marks attendance by just signing in to the application. And the records are maintained in a table format according to date, so that faculty can easily see it. So this system will save a lot of time, energy, and afford for teachers as well as institution.

2. LITERATURE SURVEY

S. no	Page Title	Journal Name	Year	Abstract	Methodologies/ Technology	Conclusion
1.	Advanced Analytics for Attendance Management using Machine Learning	Journal of Machine Learning Research	2024	Explores advanced machine learning techniques for analyzing attendance data, focusing on predictive analytics and anomaly detection	Machine learning techniques	Machine learning enhances attendance management through predictive analytics and anomaly detection, indicating future research potential
2.	Integration of IoT in Attendance Management Systems	Journal of Internet of Things	2023	Investigates how IoT devices can be integrated into attendance systems for real-time data collection and automation	IoT devices	IoT integration offers real-time data and automation, improving attendance management but requiring additional infrastructure.
3.	Enhancing User Experience in Web-Based Attendance Systems	UX/UI Design Journal	2022	Focuses on improving UX/UI in web-based attendance systems to enhance user experience and system usability.	UX/UI design principles	Improved UX/UI design significantly enhances user experience and system efficiency in web-based attendance systems.
4.	Blockchain Technology for Secure Attendance Management	International Journal of Blockchain Technology	2021	Examines the use of blockchain to secure attendance data, preventing tampering and fraud through decentralized ledgers.	Blockchain technology	Blockchain technology improves security and transparency in attendance systems, addressing issues related to data tampering and fraud.
5.	Artificial Intelligence in Attendance Management: Future Directions	Journal of Artificial Intelligence Research	2020	Investigates AI and machine learning in attendance systems, focusing on predictive analytics and anomaly detection.	AI, machine learning	AI can enhance predictive analytics and anomaly detection in attendance systems, suggesting promising future research directions.

3.SOFTWARE REQUIREMENT ANALYSIS

3.1.SDLC

The **Systems Development Life Cycle (SDLC)** or Software Development Life Cycle in systems engineering, information systems and software engineering, is the process of creating or altering systems, and the models and methodologies use to develop these systems.



Figure 3.1(a): Software Development Life Cycle

Requirement Analysis and Design:

The functional requirements for the system include user authentication for both students and staff, automated attendance marking and calculation, real-time generation of attendance reports and statistics, and a user-friendly interface to ensure easy interaction. Additionally, the system must meet several non-functional requirements: data security through encryption and access controls, scalability to accommodate increasing numbers of users and records, and accessibility from any internet-enabled device. The high-level design of the system proposes a web-based application utilizing a client-server architecture. The low-level design breaks down the system into several modules: an authentication module for managing user login and access, an attendance tracking module for marking and monitoring attendance, and a reporting module for generating attendance reports.

Implementation:

In this phase involves translating the design specifications into code using a combination of PHP, MySQL, HTML, CSS, and JavaScript. PHP is utilized for server-side logic and database interactions, while MySQL manages the attendance records. HTML, CSS, and JavaScript are employed for front-end development and user interaction. The development process is facilitated by tools such as Visual Studio Code for the development environment and Git for version control.

Testing:

In this phase includes several key components to ensure the system functions correctly and meets user requirements. Unit testing involves testing individual modules separately to verify their proper functionality. Integration testing follows, ensuring that all modules work together seamlessly within the complete system. Volume testing assesses the system's performance under a large volume of data to confirm its efficiency in handling high loads. Finally, acceptance or beta testing involves testing the system with actual users to validate that it meets their needs and requirements

Maintenance:

The maintenance phase involves post-implementation support and updates to address any issues, incorporate user feedback, and adapt to changes in user requirements. Effective change management ensures that the system remains flexible and adaptable, allowing for seamless updates, bug fixes, and the addition of new features as needed

3.2. System Study:

User Interface Systems

User interface systems can be broadly classified into two categories:

3.2.1. Computer-Initiated Interfaces:

- **Description:** In these interfaces, the computer guides the user through the interaction. The system presents information or prompts, and the user responds accordingly.

- **Relevance to Attendance Management System:** This type is useful for automated notifications and reminders about attendance deadlines or anomalies. For example, the system might alert teachers when attendance is missing or show notifications about upcoming attendance reviews.

3.2.2. User-Initiated Interfaces:

- **Description:** The user controls the interaction, inputting commands or filling out forms:

- **Relevance to Attendance Management System:** This interface is essential for users to interact with the system effectively. There are two main types within this category:

a. Command-Driven Interfaces:

- **Description:** Users input commands or queries which the system interprets and responds to.

- **Relevance to Attendance Management System:** Although less common in web-based systems, command-driven interfaces can be used for advanced search functionalities or administrative commands, like generating attendance reports based on specific queries.

b. Forms-Oriented Interfaces:

- **Description:** Users interact with the system by filling out forms displayed on their screens.

- **Relevance to Attendance Management System:** Forms-oriented interfaces are ideal for this system as they provide a structured way for users (students, teachers, administrators) to input and manage attendance data. Forms can be used for marking attendance, entering student details, and generating reports.

3.3. Modules and their Functionalities:

There are 3 modules:

1.Faculty

2.System

3.Student

1.Faculty :-

Faculties or teachers are the people who will first sign up on online attendance management system as faculty. And then after successful registration, they can login and add students and can mark attendance on daily basis.

Following data from faculty side will be provided to the system at the time of registration:

- Name
- Mobile
- Date of Birth
- Qualification
- Username

- User ID
- Password

Name:

The name field represents the full name of the user, which typically includes first name, middle name (if applicable), and last name. This information is essential for identification purposes, personalizing user experience, and formal communication.

Mobile:

The mobile field captures the user's mobile phone number. This is used for contact purposes, including sending notifications, alerts, and for two-factor authentication to enhance security.

Date of Birth:

The date of birth field records the user's birth date. This information is used for age verification, tailoring services to specific age groups, and personalizing user interactions based on age-related data.

Qualification:

The qualification field details the user's educational background or professional qualifications.

Username:

The username field is a unique identifier chosen by the user or assigned by the system. It is used for logging into the system and differentiates users.

User ID:

The user ID is a unique identifier assigned to each user by the system.

Password:

The password field is a secret string of characters chosen by the user to secure their account.

Faculty Responsibility:

- Registration on system
- Login to system
- Add students
- Mark attendance

2.SYSTEM:

System is an online platform where attendance is maintained in a secured way. The faculties are registered here, and with the help of system, they can add/remove students, can marks attendance and maintain attendance records.

System responsibilities:

- Registration of faculties and admission of students
- Adding or removing students by faculties
- Display of registered student in faculty dashboard
- Display of current attendance status in faculty dashboard
- Display of attendance records in record tab of faculty dashboard according to date in a table format
- Maintaining record for each candidate and voter without making any duplicate record
- Forget password facility for faculties

3.STUDENT:

Students are those who will be added by faculties into the system.

Following data from group/party side will be provided to the system at the time of registration:

- Name
- Roll no
- Course
- Semester
- Branch

Name:

The name field represents the full name of the user, which typically includes first name, middle name (if applicable), and last name. This information is essential for identification purposes, personalizing user experience, and formal communication.

Roll no:

The roll number is a unique identifier assigned to each student by an educational institution.

Course:

The course field specifies the particular program or subject the student is enrolled in.

Semester:

The semester field indicates the current term or period of the academic year in which the student is enrolled.

Branch:

The branch field refers to the specific area of study or specialization within a broader course of study.

3.4. Existing System

Existing attendance management system is manual, in which everything has to be maintained with the manual process. Attendance records are maintained in register, book, files, etc. These records have to be kept safe and secure without being lost or damaged. So, a secure and safe place is required to keep these records, so that no any liquid substance or any animal like mouse, etc. could harm these data. Generally, they are stored in cup boards. For example, if someone has marked attendance on register, and unfortunately a register has come in contact with water or any liquid substance or any animal like mouse, it will lose the data which is stored in register. And if anyone has to check attendance record, then he or she has to find the register first, then he or she can access it.

The existing system of attendance management typically involves a manual process where teachers or administrative staff record student attendance using paper-based registers or spreadsheets. This method is time-consuming, prone to errors, and inefficient in terms of data retrieval and analysis. Additionally, it lacks real-time updates and can be easily manipulated, leading to inaccuracies in attendance records. Communication of attendance information to students and parents often relies on manual reports or phone calls, further adding to the inefficiency. Overall, the manual attendance management system is outdated and struggles to meet the needs of modern educational institutions in terms of accuracy, efficiency, and data security.

4. Feasibility Study

4.1. Feasibility Report

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Projects are initiated for two broad reasons:

1. Problems that lend themselves to systems solutions
2. Opportunities for improving through:
 - (a) upgrading systems
 - (b) altering systems
 - (c) installing new systems

A feasibility study should provide management with enough information to decide:

- Whether the project can be done
- Whether the final product will benefit its intended users and organization
- What are the alternatives among which a solution will be chosen
- Is there a preferred alternative?

Three key considerations involved in the feasibility analysis are:

- Technical Feasibility
- Operational Feasibility
- Economical Feasibility

4.2. Technical Feasibility:

A large part of determining resources has to do with assessing technical feasibility. It considers the technical requirements of the proposed project. The technical requirements are then compared to the technical capability of the organization. The systems project is considered technically feasible if the internal technical capability is sufficient to support the project requirements.

Technical feasibility assesses the technical requirements of the proposed project and compares them to the organization's technical capabilities. The project is considered technically feasible if the internal technical capability is sufficient to support the project requirements.

The essential questions that help in testing the technical feasibility of a system include the following:

- Is the project feasible within the limits of current technology?
- Does the technology exist at all?
- Is it available within given resource constraints?
- Is it a practical proposition?
- Do we possess the necessary technical expertise, and is the schedule reasonable?

4.3. Operational Feasibility:

Operational feasibility is dependent on human resources available for the project and involves projecting whether the system will be used if it is developed and implemented.

Operational feasibility is a measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development.

Operational feasibility reviews the willingness of the organization to support the proposed system. This is probably the most difficult of the feasibilities to gauge. In order to determine this feasibility, it is important to understand the management commitment to the proposed project. If the request was initiated by management, it is likely that there is management support and the system will be accepted and used. However, it is also important that the employee base will be accepting of the change.

The essential questions that help in testing the operational feasibility of a system include the following:

- Does the current mode of operation provide adequate throughput and response time?
- Does the current mode provide end users and managers with timely, pertinent, accurate, and useful formatted information?
- Does the current mode of operation provide cost-effective information services to the institution?
- Could there be a reduction in cost and/or an increase in benefits?
- Does the current mode of operation offer effective controls to protect against fraud and ensure data accuracy and security?
- Are current work practices and procedures adequate to support the new system?
- If the system is developed, will it be used?

4.4.Economical Feasibility:

Economic feasibility, or cost/benefit analysis, determines the benefits and savings expected from the system and compares them with costs. The decision to design and implement the system is made if the benefits outweigh the costs.

The essential questions that help in testing the operational feasibility of a system include the following:

- Is the system cost-effective?
- Do benefits outweigh costs?

5.SYSTEM REQUIREMENTS SPECIFICATION

5.1.Requirement Specification:

A requirement specification for a software system is a complete description of the behavior of a system to be developed. It includes a set of use cases that describe all the interactions the users will have with the software. In addition to use cases, the SRS also contains non-functional requirements. Non-functional requirements which impose constraints on the design or implementation such as performance engineering requirements, quality standards.

System requirement specification is a structured collection of information that embodies the requirements of a system. A business analyst, sometimes titled system analyst, is responsible for analysing the business needs of their clients and stakeholders to help identify the business problems and propose solutions. Within the system development life cycle domain, the business analyst typically performs a liaison function between the business side of an enterprise and the information technology department or external service providers.

5.2.Hardware Requirements:

MINIMUM (Required for Execution)		MY SYSTEM (Development)
System	Pentium IV 2.2 GHz	i3 Processor 5 th Gen
Hard Disk	20 Gb	100 Gb or above
Ram	1 Gb	4 Gb

5.3. Software Requirements:

Operating System	Windows 10/11
Development Software	Web based Application
Programming Language	PHP
Domain	Web Development
Integrated Development Environment (IDE)	Visual Studio Code
Front End Technologies	HTML5, CSS3, Java Script
Front End Technologies or Framework	Bootstrap
Database Language	SQL
Database (RDBMS)	MySQL
Database Software	XAMPP Server
Web Server or Deployment Server	Apache
Design/Modelling	Rational Rose

Selected Software:

1.Introduction to PHP:

PHP (Hypertext Preprocessor) is a widely-used, open-source scripting language especially suited for web development. It can be embedded into HTML, making it ideal for creating dynamic web pages and applications. PHP runs on the server side, which means it executes on the web server, generating HTML that is sent to the client's web browser.

- PHP can generate dynamic page content
- PHP can create, open, read, write, delete, and close files on the server
- PHP can collect form data
- PHP can send and receive cookies
- PHP can add, delete, modify data in your database
- PHP can be used to control user-access
- PHP can encrypt data

The biggest strength of PHP is huge collection of standard libraries which can be used for the following

1. Database Access
2. File Handling
3. Session and Cookie Management
4. Form Handling
5. Data Encryption
6. XML and JSON Parsing

Advantages of PHP

1. Ease of Learning and Use:

PHP has a simple and intuitive syntax, making it accessible for beginners. Its similarity to C and Java also helps those familiar with these languages.

2. Open Source and Free:

PHP is open-source, meaning it is free to use and modify. This reduces development costs and encourages a large community of developers to contribute and share resources.

3. Cross-Platform Compatibility:

PHP runs on various operating systems, including Windows, Linux, and macOS, making it highly versatile for different development environments.

4. Robust Community Support:

PHP has a large, active community that provides extensive documentation, forums, tutorials, and solutions for common issues, aiding developers in problem-solving.

5. Database Integration:

PHP integrates seamlessly with numerous databases, such as MySQL, PostgreSQL, Oracle, and SQLite, offering flexibility in data storage solutions.\

6. Performance:

PHP is efficient for creating dynamic web pages quickly. Its server-side execution reduces client-side processing time, enhancing overall performance.

7. Security Features:

PHP offers built-in security features to protect against common web threats, such as SQL injection, cross-site scripting (XSS), and cross-site request forgery (CSRF).

8. Integration Capabilities:

PHP can integrate with various web technologies, including HTML, CSS, JavaScript, and XML, making it suitable for full-stack web development.

Advantages of PHP Over Other Languages

1. Embedded in HTML:

PHP can be directly embedded within HTML, making it straightforward to add dynamic content to static web pages. This can be more complex in other languages that require separate server-side processing, such as JavaScript running on Node.js.

2. Efficient for Web Development:

PHP was designed specifically for web development, with features tailored to the needs of web developers. Other languages like Python or Ruby are more general-purpose, and while they are powerful, they may not offer the same level of specialization for web tasks.

3. Rapid Development:

PHP's simplicity and the availability of numerous frameworks facilitate rapid application development, enabling quicker deployment compared to more complex languages like Java or C++.

4. Strong Community Support:

The large, active PHP community provides extensive documentation, forums, and resources, making it easier to find solutions to problems and stay updated with best practices.

Disadvantages of PHP:

1. Poor Error Handling:

PHP's error handling can be inconsistent and less robust compared to other languages. Exception handling is available but is not as well-integrated or powerful as in languages like Java or C#.

2. Performance Issues:

While PHP is suitable for many web applications, it can struggle with performance under heavy loads compared to languages like Java or C#.

3. Weak Typing:

PHP's dynamic and weak typing can lead to unexpected behavior and bugs that are difficult to trace. This lack of strict type enforcement can lead to issues, particularly in larger, more complex applications.

PHP Development Steps

PHP, which originally stood for "Personal Home Page" and later changed to "PHP: Hypertext Preprocessor," was created by Rasmus Lerdorf in 1994. He developed it initially to manage his personal website and track visitors.

The first version, PHP/FI (Form Interpreter), released in 1995, allowed users to create dynamic web content and process web forms.

In 1997, PHP 2.0 gained more features and a larger user base. This version still carried the limitations of its origins, and it wasn't until Andi Gutmans and Zeev Suraski rewrote the parser in 1998 that PHP saw a major transformation.

The release of PHP 3.0, marked a significant improvement, with a more robust and extensible architecture, leading to its widespread adoption in web development.

The introduction of PHP 4.0 in 2000, powered by the new Zend Engine developed by Gutmans and Suraski, brought enhanced performance, reliability, and additional features, making PHP suitable for more complex web applications.

PHP 5.0, released in 2004, introduced a new object-oriented programming model, improved XML support, and the Zend Engine II, further solidifying PHP's role in modern web development.

PHP 6.0 was an ambitious attempt to bring native Unicode support to the language but was ultimately abandoned due to complexities and performance issues.

In 2015, PHP 7.0 was released with the new Zend Engine 3.0, which provided significant performance improvements, reduced memory usage.

PHP 8.0, released in 2020, continued this trend with the introduction of Just-In-Time (JIT) compilation, which further improved performance by optimizing code execution.

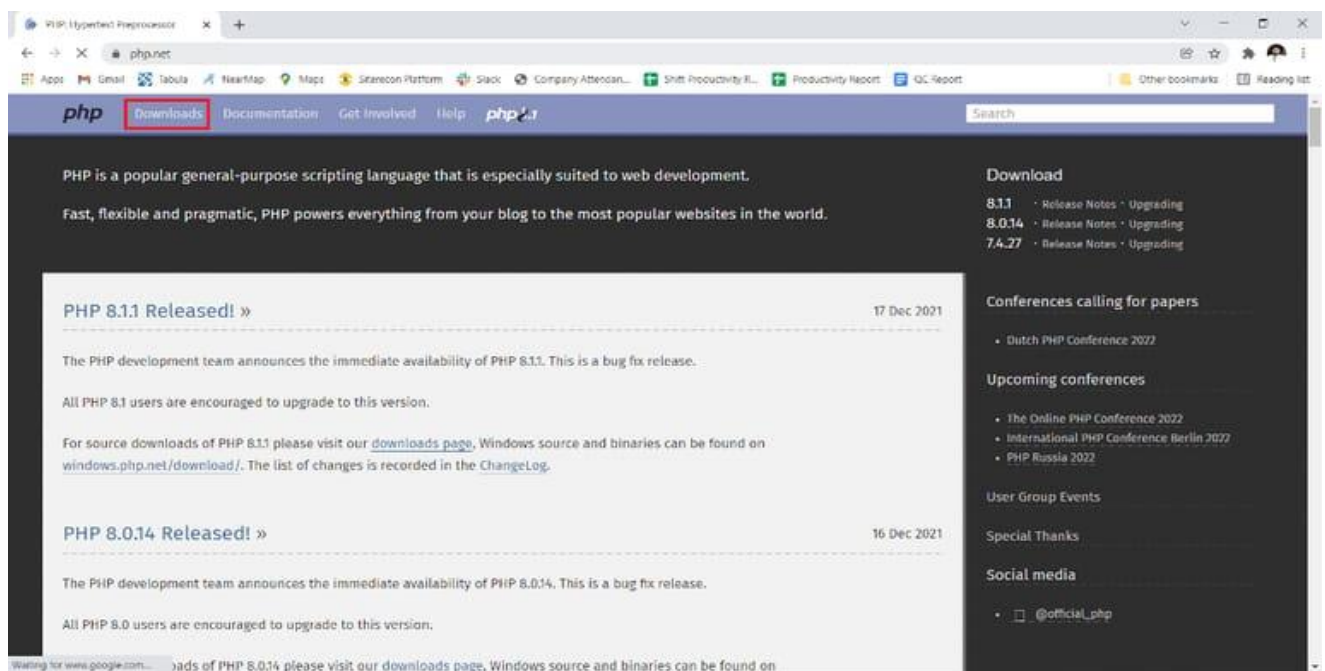
- PHP remains a cornerstone of web development due to its ease of use, flexibility, and a vast ecosystem of frameworks and libraries.
- Popular frameworks like Laravel, Symfony, and CodeIgniter, along with content management systems like WordPress, Joomla, and Drupal, rely on PHP, contributing to its enduring popularity.
- Regular updates and a strong community ensure that PHP continues to evolve, meeting the demands of modern web development and maintaining its relevance in the tech industry.

How to Install PHP on Windows and Mac

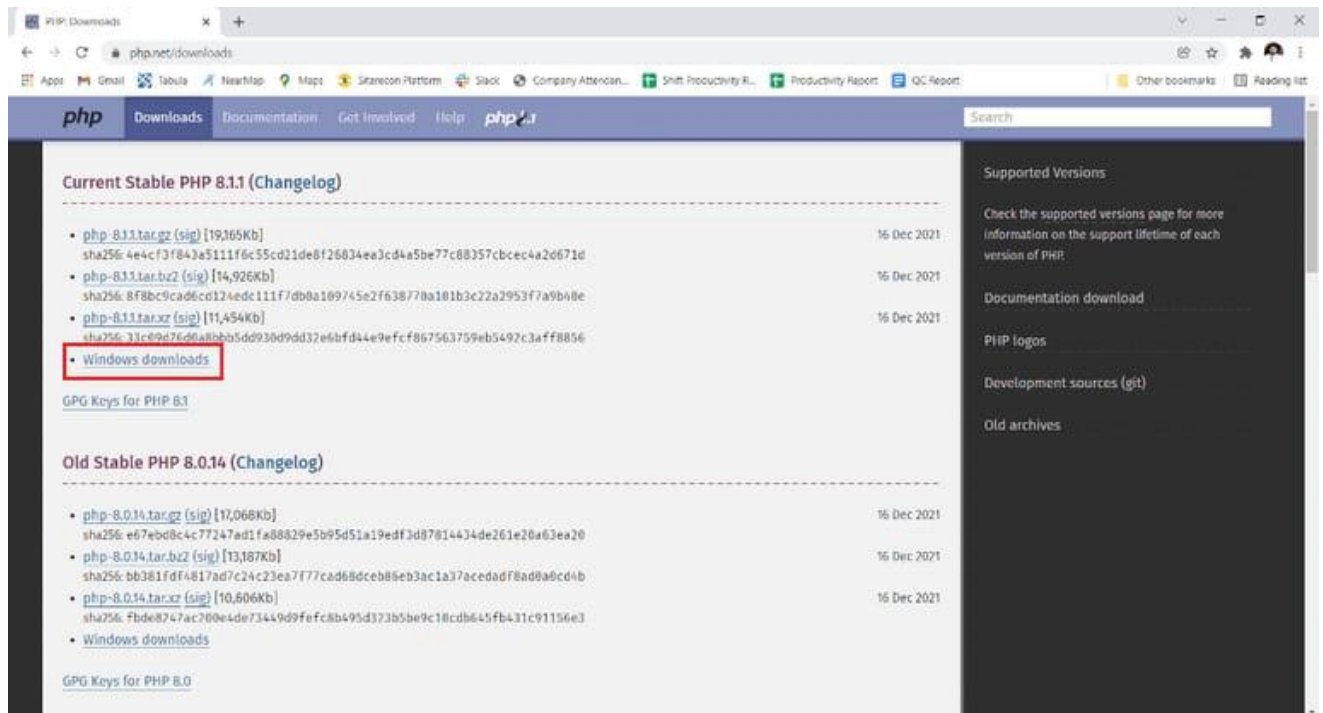
To install PHP on Windows, first download the latest version from the official PHP for Windows page, selecting the Thread Safe version that matches your system architecture (x86 for 32-bit or x64 for 64-bit). Extract the downloaded ZIP file to a folder (e.g., C:\php). Next, rename the php.ini-development file to php.ini and configure it by enabling necessary extensions (e.g., extension=curl, extension=mbstring). Then, add the PHP folder path (e.g., C:\php) to the Path variable in your system's Environment Variables. Finally, open Command Prompt and type `php -v` to verify the installation. For Mac, install Homebrew first by running the installation command in Terminal. Once Homebrew is installed, run `brew install php` to install the latest version of PHP. After installation, type `php -v` in the terminal to confirm PHP is installed correctly. Both systems should now have PHP ready for use.

Download the Correct version into the system

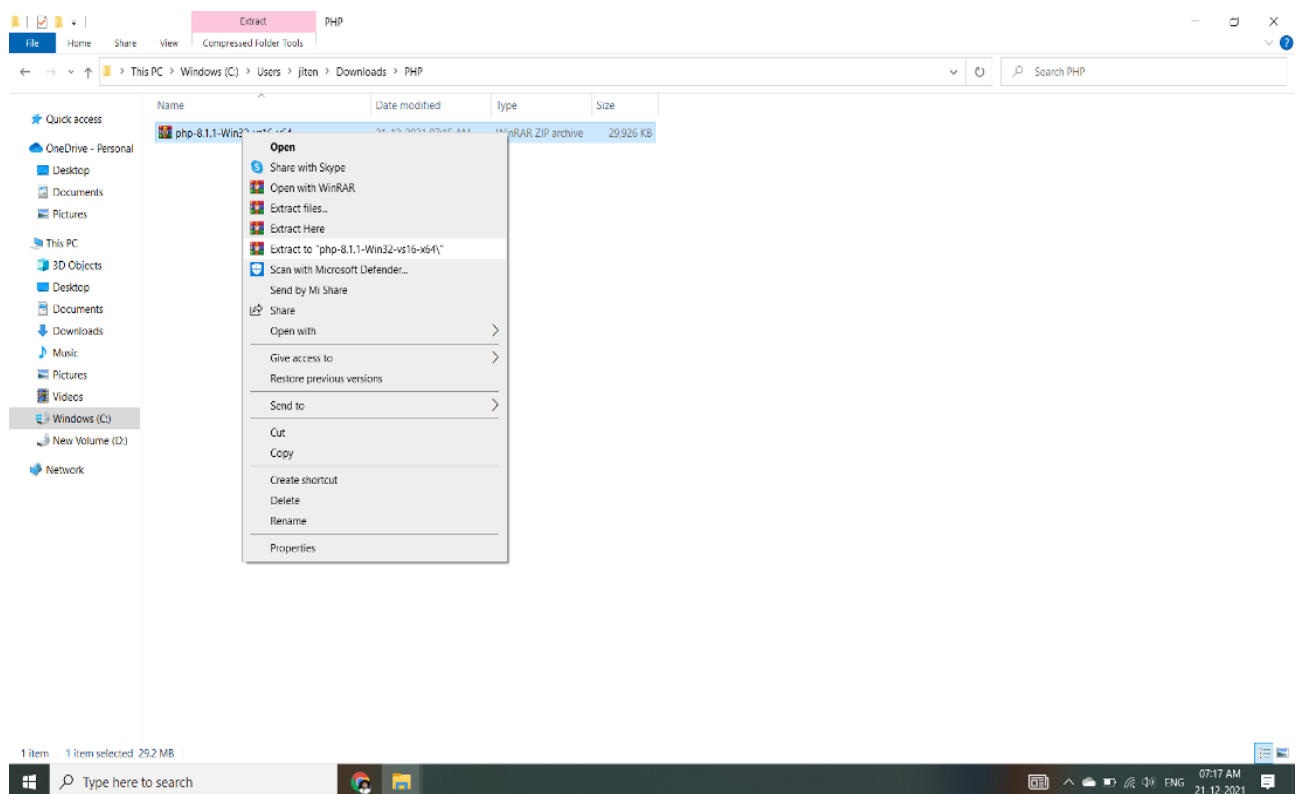
Step 1: Go to the official site to download and install PHP using Google Chrome or any other web browser. OR Click on the following link: <https://www.php.net/downloads>



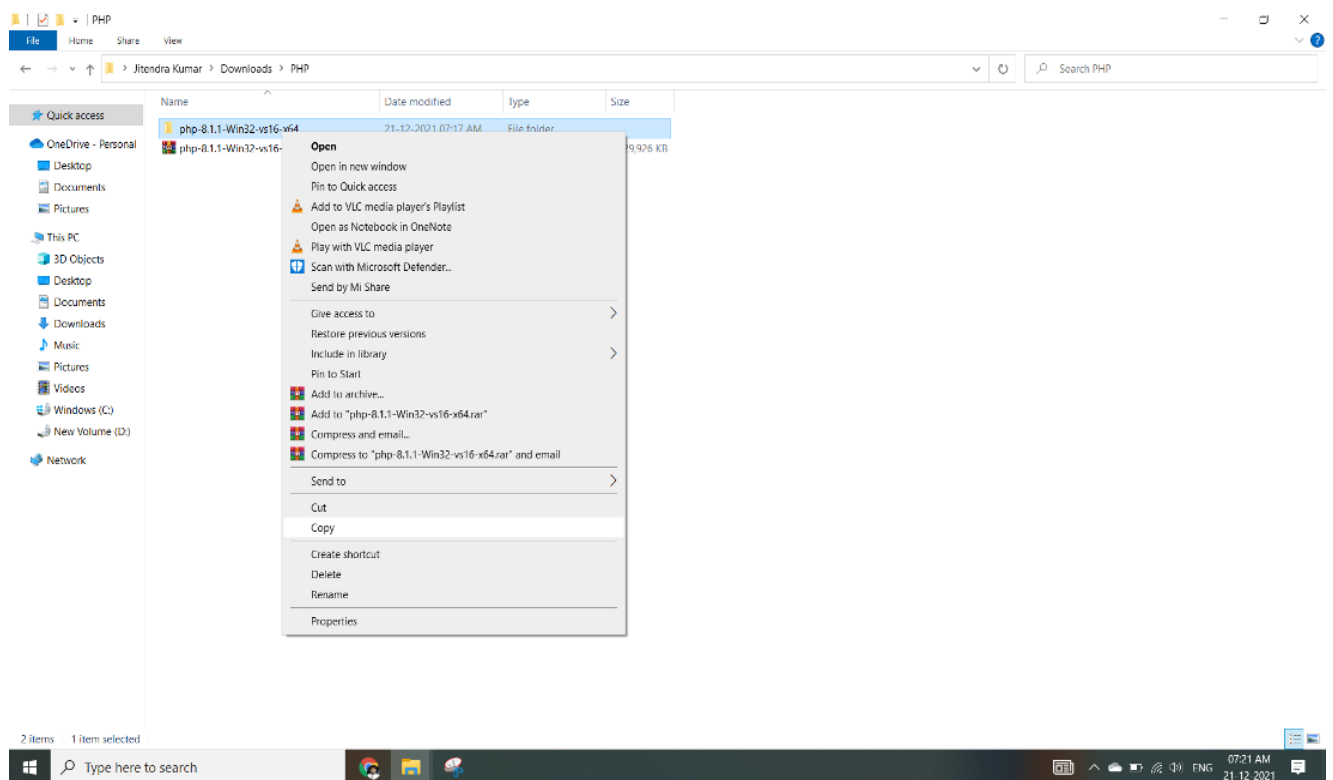
Step 2: Click on the Download Tab



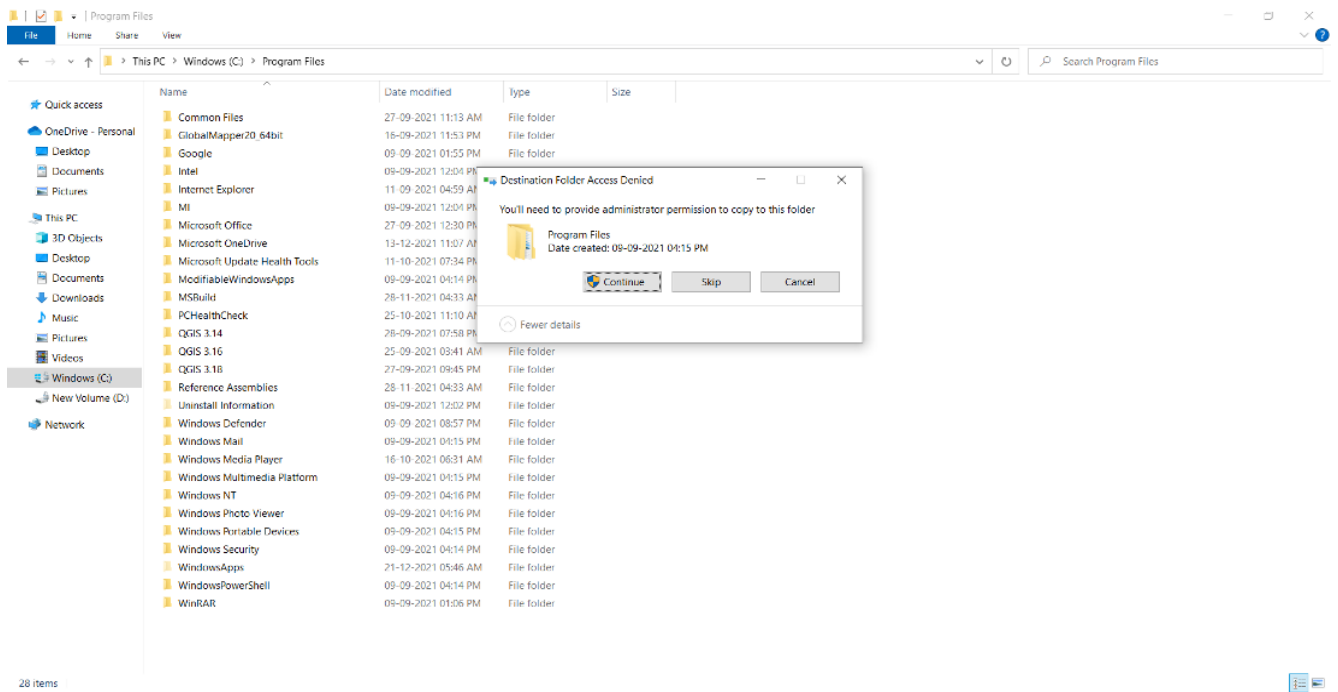
Step 3: Now check for the zip file in downloads in your system and extract it.



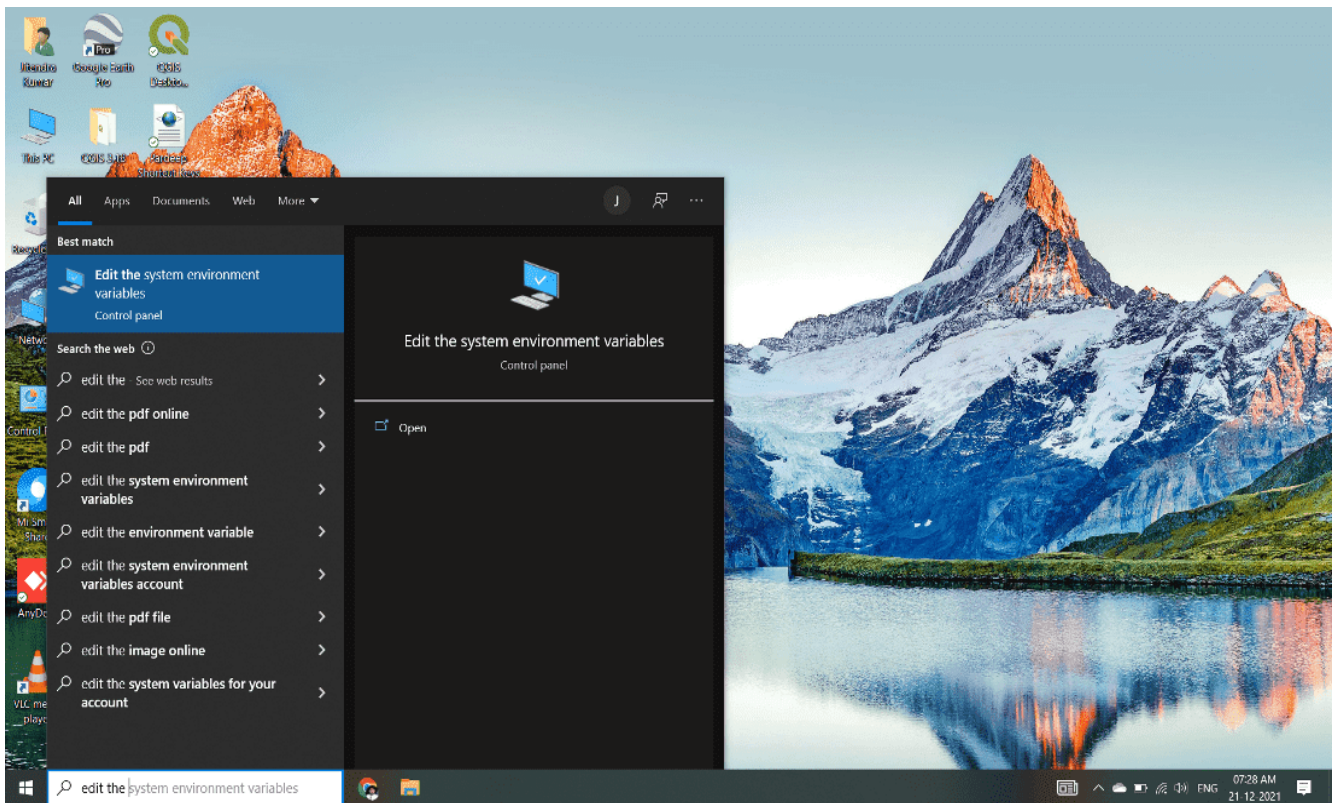
Step 4: Now copy the extracted folder.



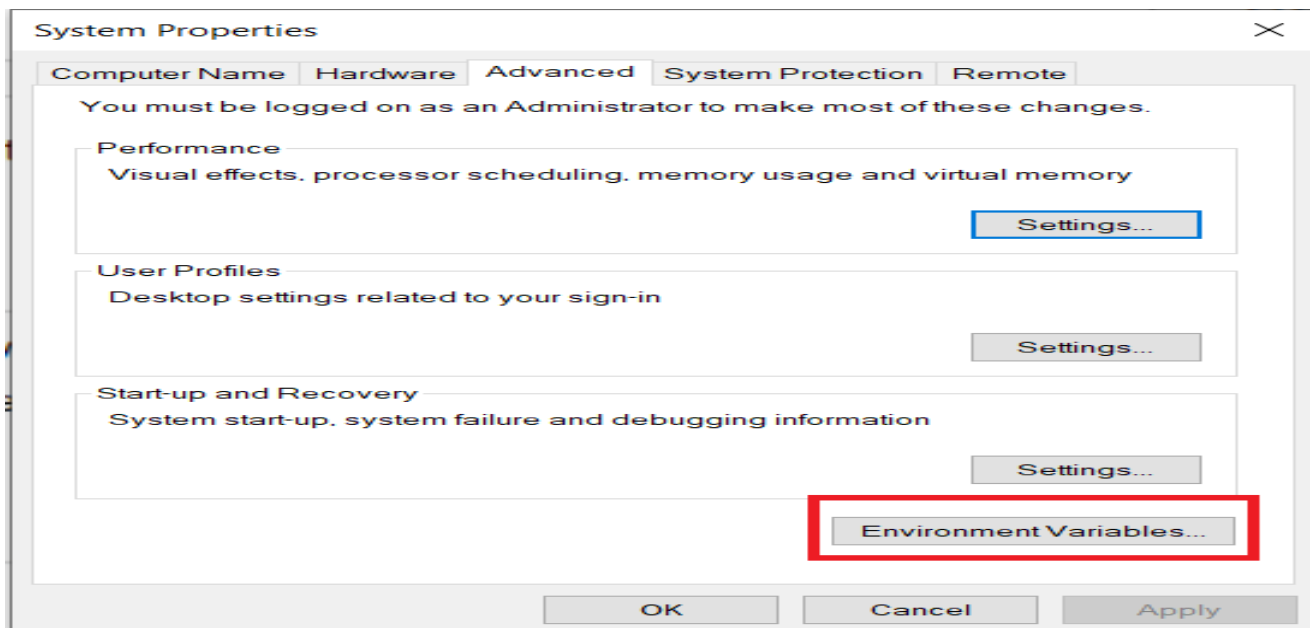
Step 5: Now the Permission Windows appears to paste the folder in program files then click on “Continue”.



Step 6: Now click on Start Menu and search “Edit the system environment variables” and open it.

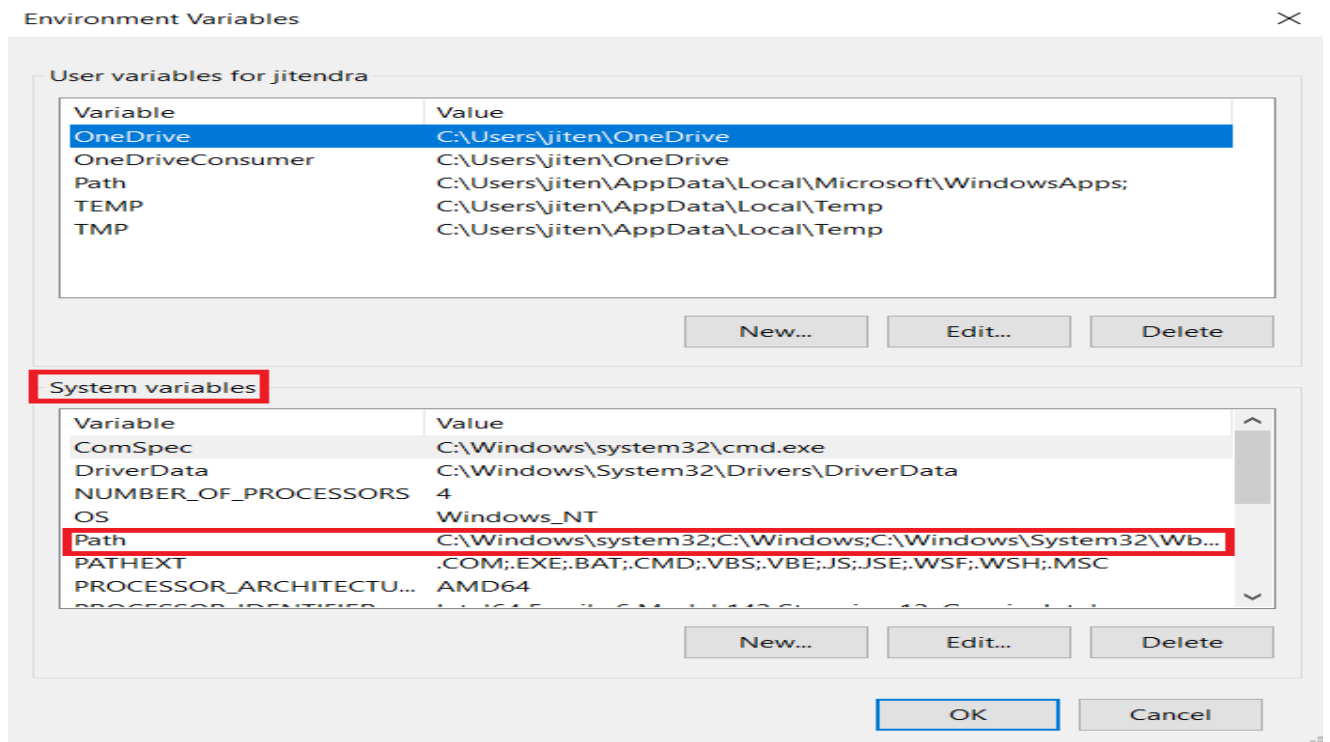


Step 7: After opening System, Variable New window appears, and click on “Environment Variables”.



:

Step 8: Now go to the “System variables” Path option and double click on Path.



Step 9: Next screen will open and click on the “New” button.

%SystemRoot%\system32
%SystemRoot%
%SystemRoot%\System32\Wbem
%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\
%SYSTEMROOT%\System32\OpenSSH\
C:\Program Files\QGIS 3.14\bin

New

Edit

Browse...

Delete

Move Up

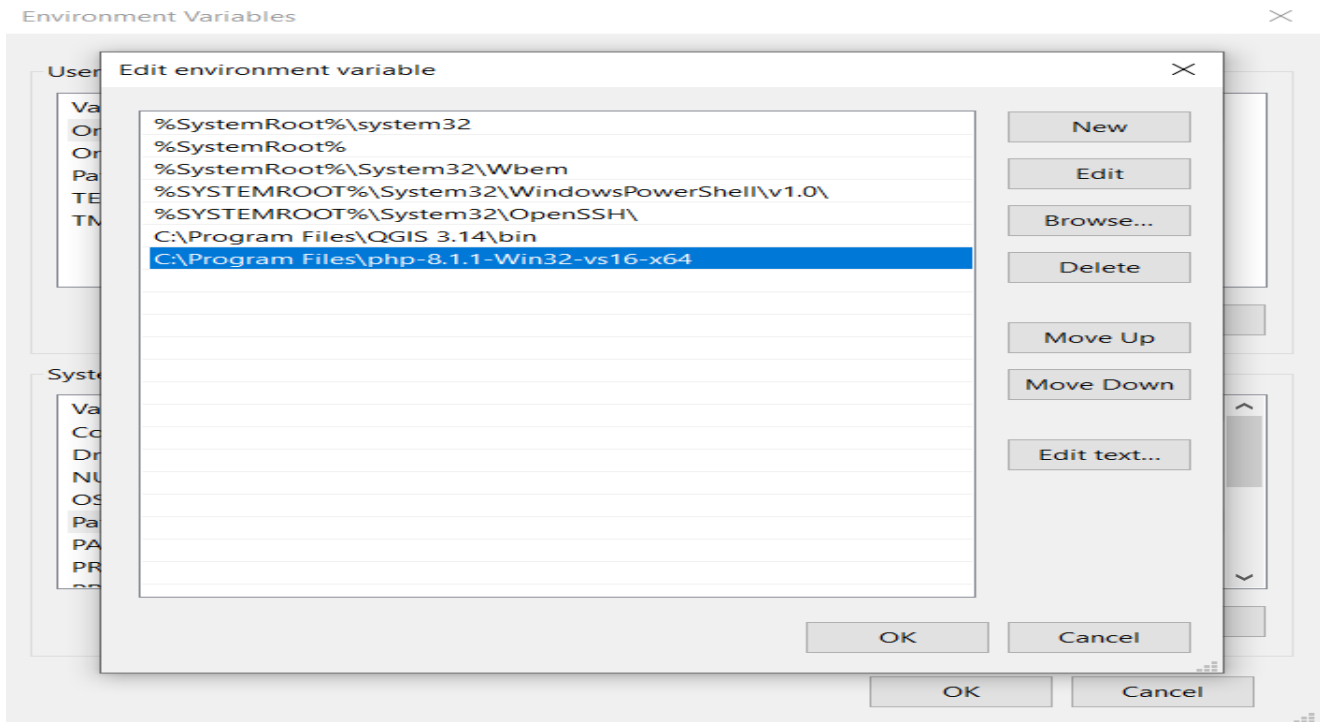
Move Down

Edit text...

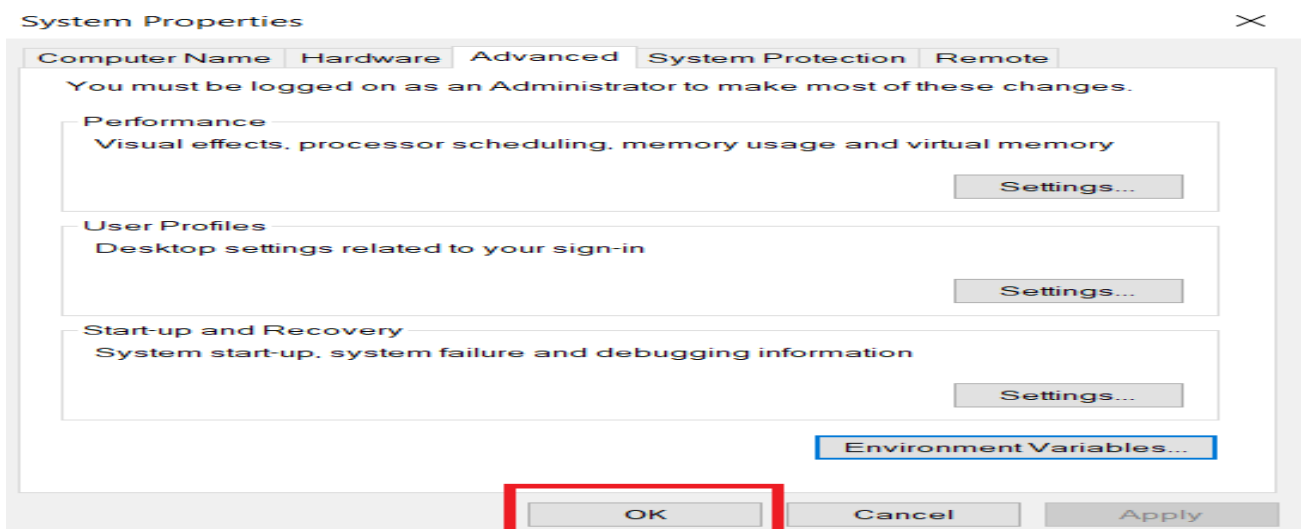
OK

Cancel

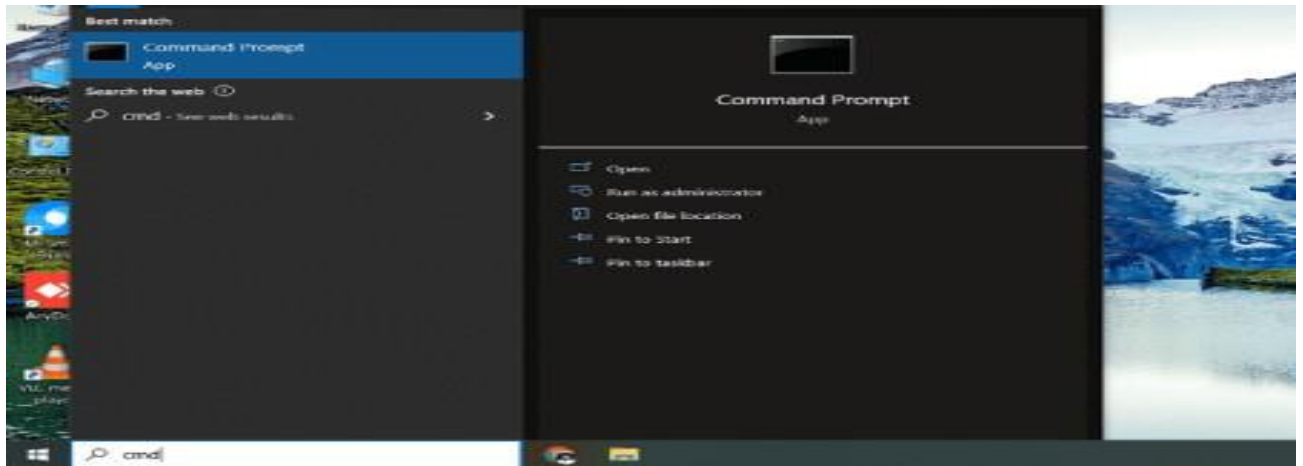
Step 10: After New Paste the address we copy from program files to new and click on Enter button.



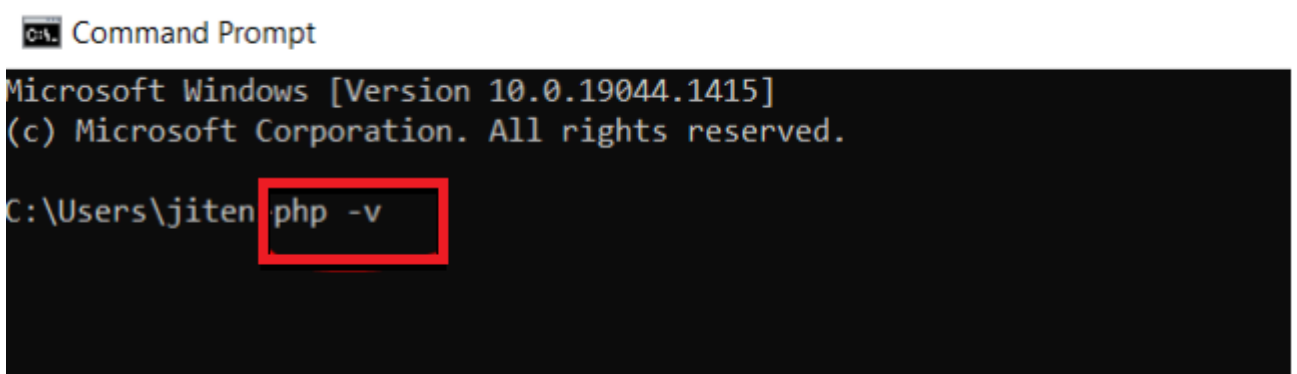
Step 11: Click on OK for saving changes.



Step 12: Now your PHP is installed on your computer. You may check by going to the “Start” menu typing Command Prompt. Open it.



Step 13: When the Command Prompt opens, type php -v



Step 14: Now enter the command prompt to show the version of PHP installed on your computer.

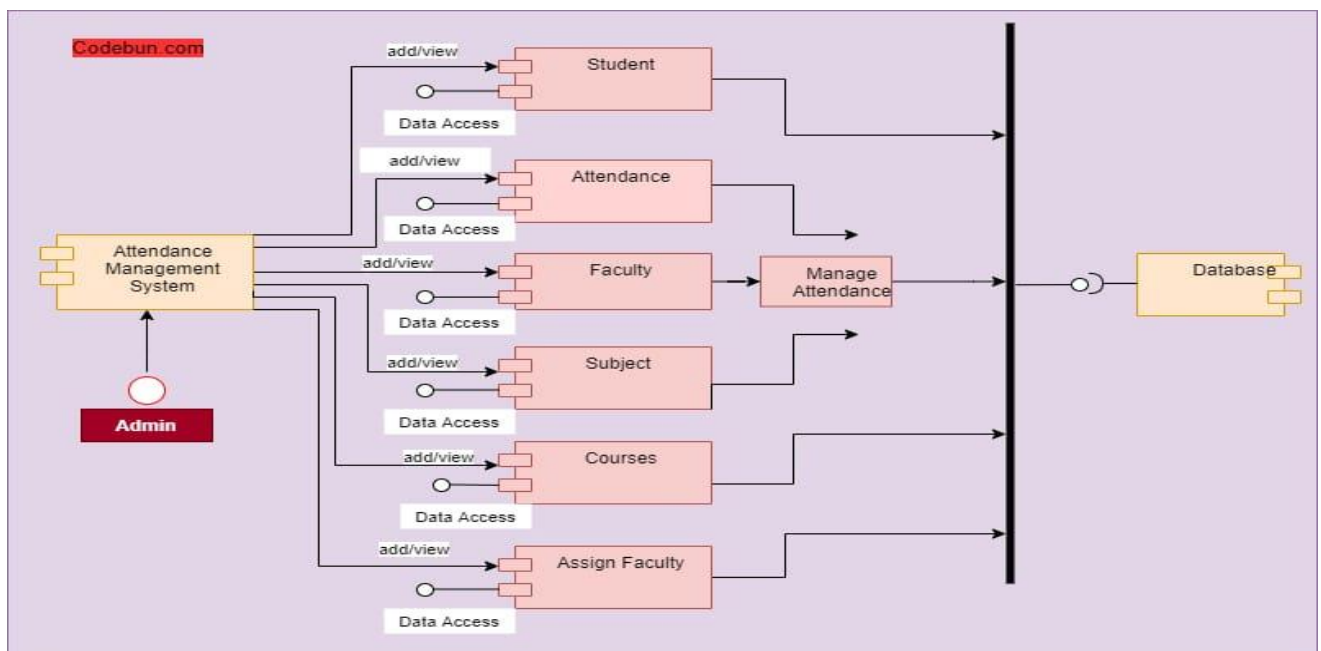


Finally, You have successfully installed PHP on your Windows 10 system.

6 SYSTEM DESIGN

System Architecture:

An Attendance Management System (AMS) typically follows a client-server architecture, where the frontend (client layer) includes web or mobile interfaces for students, teachers, and administrators to interact with the system. The backend (application layer) handles business logic, such as user authentication, attendance tracking, and report generation, using technologies like Node.js, Java, or Python. The database layer stores attendance records, user details, and course information, often utilizing relational databases like MySQL or PostgreSQL. The system may integrate with external devices like biometric scanners or QR code readers for automated attendance tracking. A notification system may also be incorporated to send alerts and reminders through email, SMS, or push notifications, using services like Twilio or Firebase. Additionally, the AMS can generate detailed reports and analytics on attendance trends and performance, leveraging tools like Power BI or custom dashboards. In terms of deployment, the system can be structured using monolithic architecture for simplicity, microservices architecture for scalability and flexibility, or a cloud-based architecture for higher availability, security, and elasticity. Security is critical, with robust authentication and authorization mechanisms (e.g., OAuth, JWT) and encryption for data protection. To handle increasing demand, the system can be scaled horizontally with load balancing and caching strategies (e.g., Redis) for enhanced performance.



6.1 UML Diagrams:

UML is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems. UML was created by Object Management Group (OMG) and UML 1.0 specification draft was proposed to the OMG in January 1997.

OMG is continuously putting effort to make a truly industry standard.

- UML stands for **U**nified **M**odeling **L**anguage.
- UML is a pictorial language used to make software blue prints.

UML Modeling Types:

It is very important to distinguish between the UML model. Different diagrams are used for different type of UML modeling. There are three important type of UML modelings:

6.1.1 STRUCTURAL THINGS:

Structural things are classified into seven types those are as follows:

CLASS DIAGRAM:

Class diagrams are the most common diagrams used in UML. Class diagram consists of classes, interfaces, associations and collaboration. Class diagram represents the object orientation of a system. So it is generally used for development purpose. This is the most widely used diagram at the time of system construction.

The purpose of the class diagram is to model the static view of an application. The class diagrams are the only diagrams which can be directly mapped with object oriented languages and thus widely used at the time of construction.

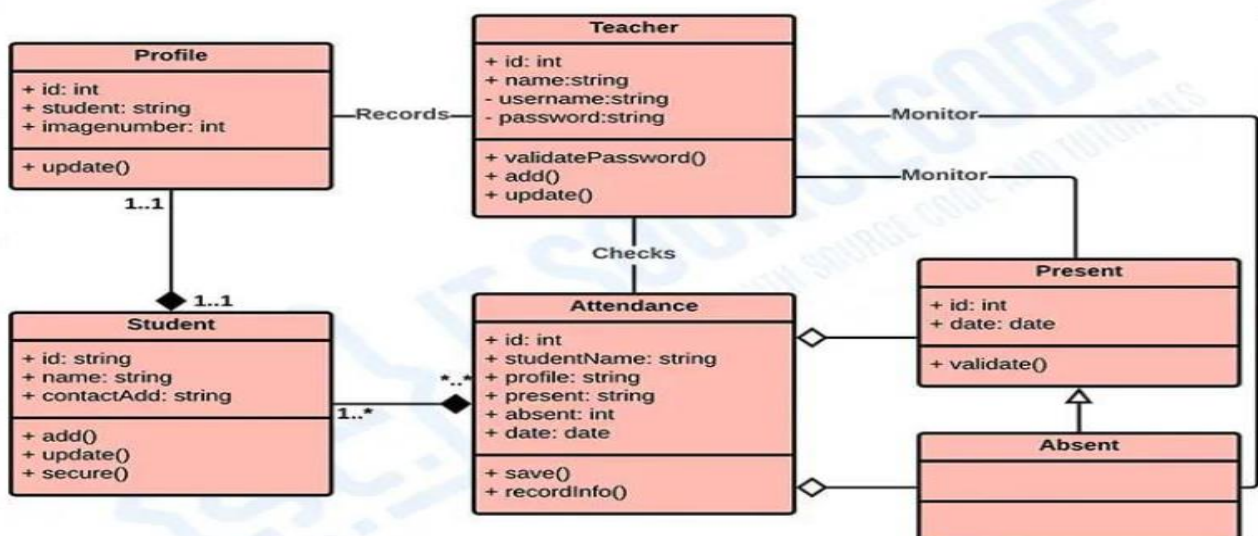


Figure Class Diagram

USE CASE DIAGRAM:

Use case diagrams are considered for high level requirement analysis of a system. So when the requirements of a system are analyzed the functionalities are captured in use cases. So we can say that use cases are nothing but the system functionalities written in an organized manner. Now the second things which are relevant to the use cases are the actors. Actors can be defined as something that interacts with the system. The actors can be human user, some internal applications or may be some external applications. So in a brief when we are planning to draw an use case diagram we should have the following items identified.

Functionalities to be represented as an use case

Actors

Relationships among the use cases and actors.

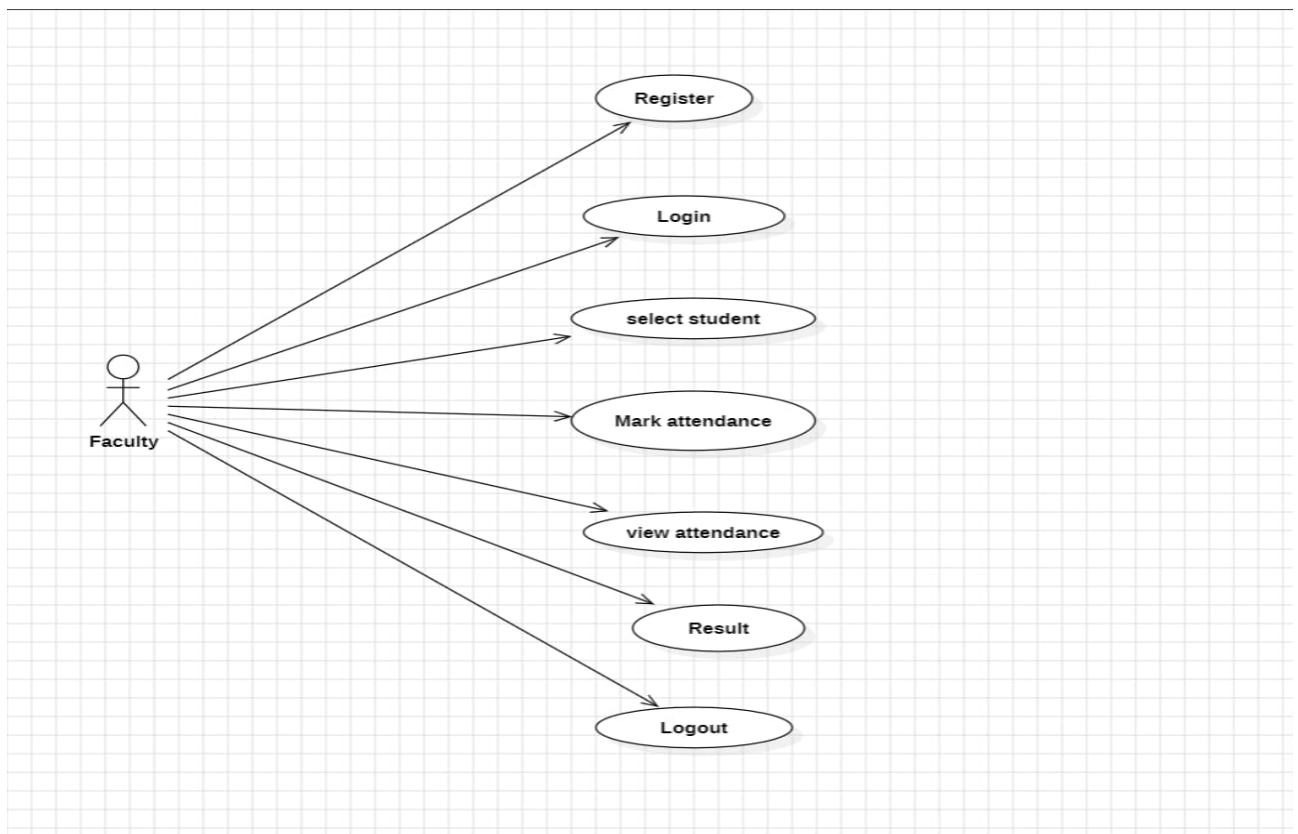


Figure Use Case Diagram

BEHAVIORAL THINGS

Behavioural things are considered as verbs of a model. These are the 'dynamic' parts which describes how the model carry out its functionality with respect to time and space. Behavioral things are classified into two types:

One is the sequence diagram and the other is the collaboration diagram. The sequence diagram captures the time sequence of the message flow from one object to another and the collaboration diagram describes the organization of objects in a system taking part in the message flow.

SEQUENCE DIAGRAM

The sequence diagram has four objects (Customer, Order, Special Order and Normal Order).

The following diagram shows the message sequence for Special Order object and the same can be used in case of Normal Order object. It is important to understand the time sequence of message flows.

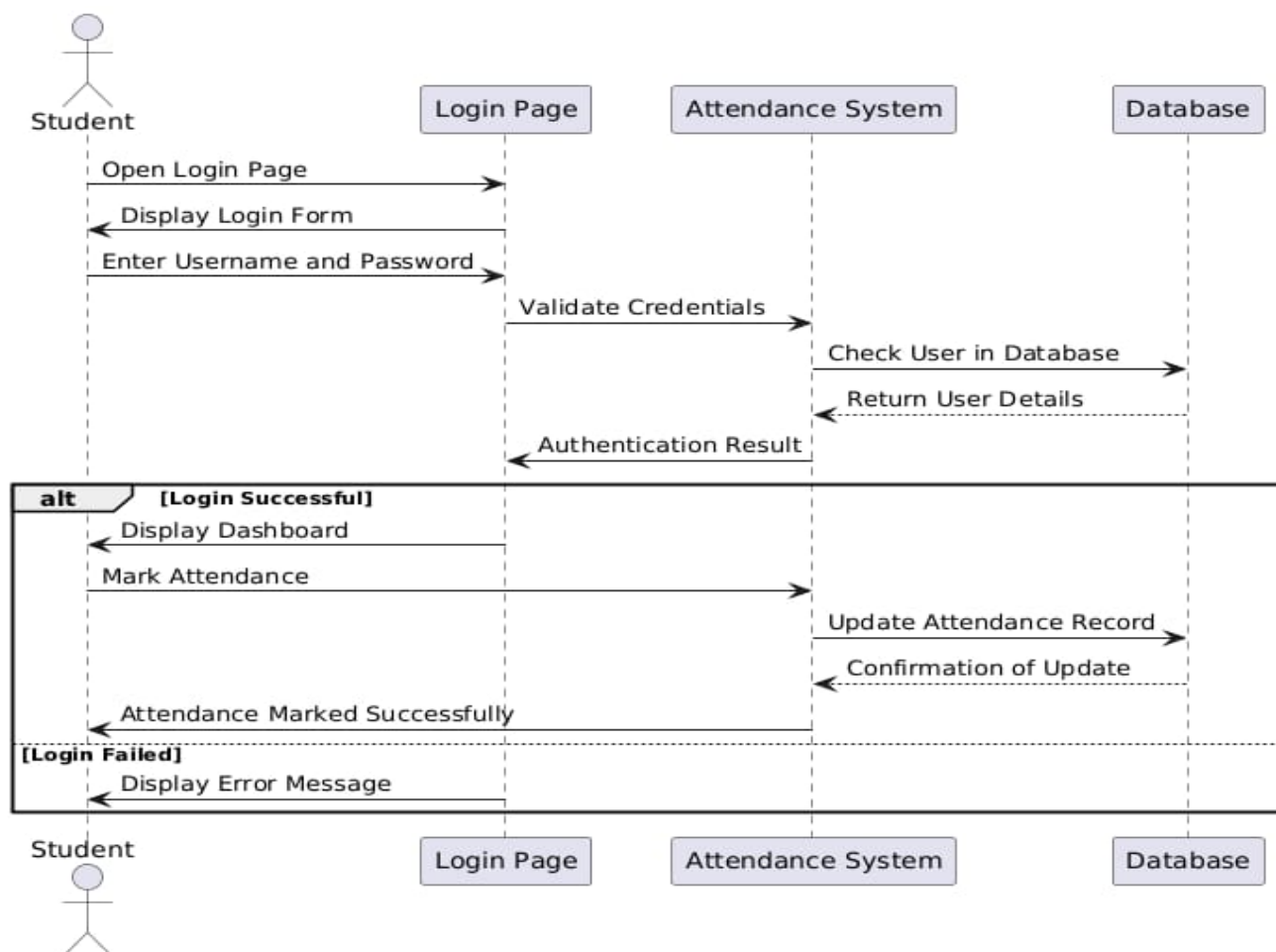


Figure Sequence Diagram

Activity diagram is another important diagram in UML to *describe the dynamic aspects of the system*.

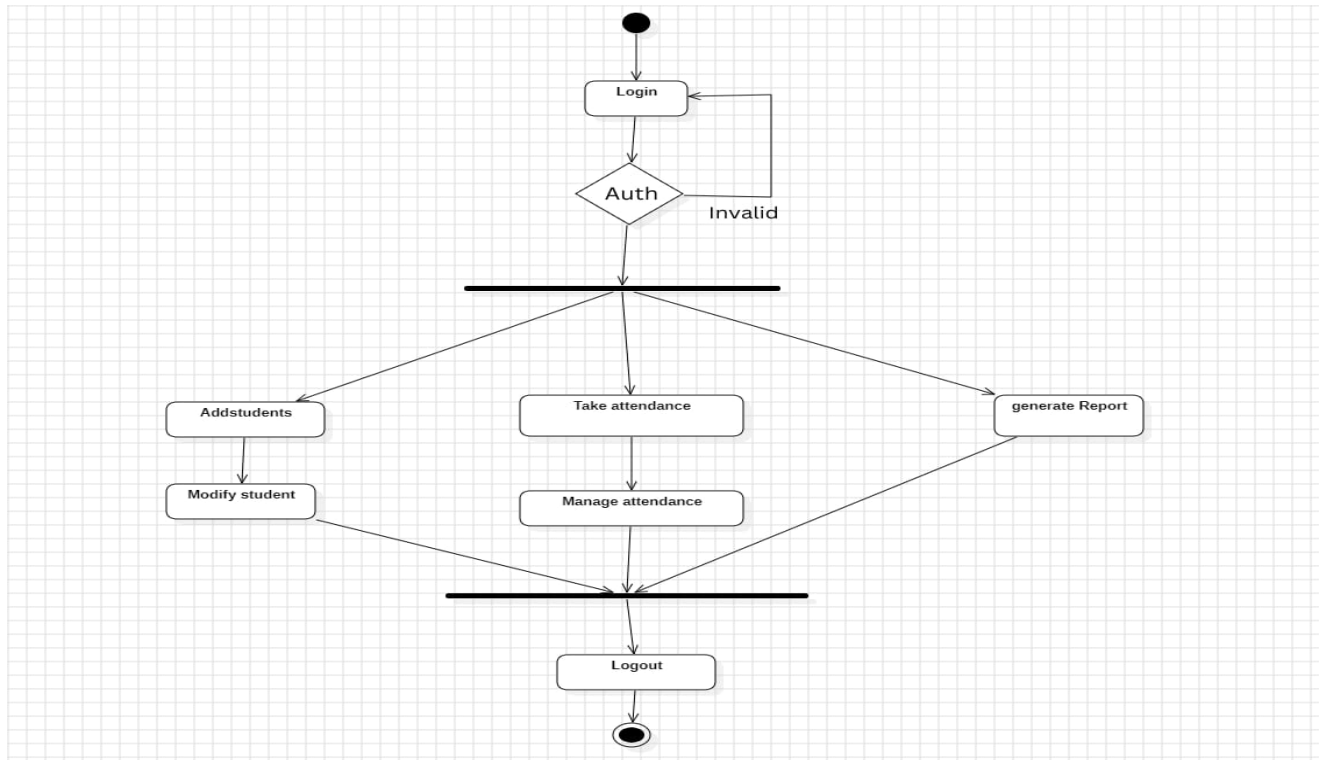


Figure Activity Diagram

DEPLOYMENT DIAGRAM:

The deployment diagram visualizes the physical hardware on which the software will be deployed. Deployment Diagram is a type of diagram that specifies the physical hardware on which the software system will execute. It also determines how the software is deployed on the underlying hardware. It maps software pieces of a system to the device that are going to execute it.

The deployment diagram maps the software architecture created in design to the physical system architecture that executes it. In distributed systems, it models the distribution of the software across the physical nodes.

The software systems are manifested using various artifacts, and then they are mapped to the execution environment that is going to execute the software such as nodes. Many nodes are involved in the deployment diagram; hence, the relation between them is represented using communication paths.

There are two forms of a deployment diagram.

- Descriptor form
- It contains nodes, the relationship between nodes and artifacts.
- Instance form
- It contains node instance, the relationship between node instances and artifact instance.
- An underlined name represents node instances.

Purpose of a deployment diagram

Deployment diagrams are used with the sole purpose of describing how software is deployed into the hardware system. It visualizes how software interacts with the hardware to execute the complete functionality. It is used to describe software to hardware interaction and vice versa.

Deployment Diagram Symbol and notations

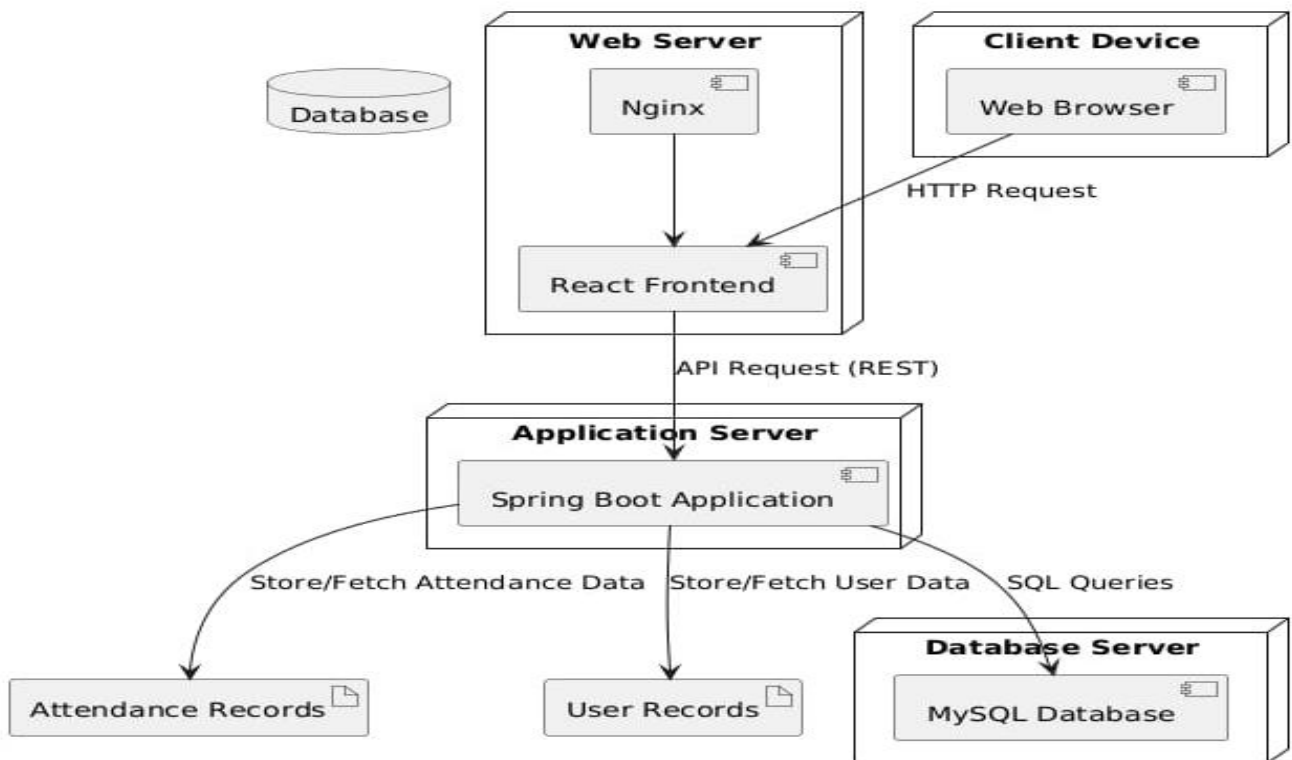


Figure Deployment Diagram

6.2 DATA FLOW DIAGRAM

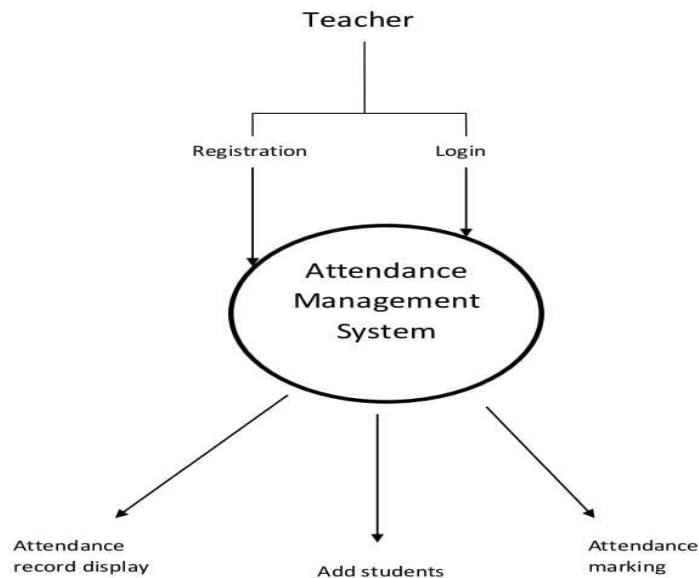


Figure 6.2.1 Data Flow Diagram

- ✓ The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.
- ✓ The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.
- ✓ DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.

7. CODING AND IMPLEMENTATION

7.1. SAMPLE CODE:

Homepage (HTML)

```
<!doctype html>
<html lang="en">
<head>
<!-- Required meta tags -->
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
<!-- Bootstrap CSS -->
<title>Home - Attendance Management System</title>
<link rel="stylesheet" href="resources/Bootstrap/css/bootstrap.min.css">
<link rel="stylesheet" href="resources/font-awesome/css/font-awesome.min.css">
<link rel="stylesheet" href="resources/css/stylesheet.css">
<script src="resources/Jquery/jquery-3.5.1.js"></script>
<script src="resources/Bootstrap/js/bootstrap.min.js"></script>
<script src="resources/js/sweetalert.min.js"></script>
</head>
<body>
<div id="headerSection" class="sticky-top">
<div class="container" >
<div class="row">
<div class="col-sm-12 text-center pt-3">
<p id="brand">Attendance Management System</p>
</div>
</div>
</div>
</div>
```

```

<div id="bodySection">
<div class="container">
<div class="row pt-4 pb-2 align-items-center">
<div class="col-md-12 text-center py-3">
<h2>Ramniranjan Jhunjhunwala College of Arts, Science & Commerce</h2>
</div>
</div>
<div class="row py-1 align-items-center">
<div class="col-md-3"></div>
<div class="col-md-6 text-center">
<div id="loginSection" class="text-center">
<br>
<h4>Faculty Login</h4>
<h4><i class="fa fa-user-circle fa-3x py- 2" style="color:#be2edd"></i></h4>
<form>
<div class="form-row py-1 px-5">
<div class="form-group col-md-12">

```

Registration (HTML)

```

<?php
session_start();
?>
<!doctype html>
<html lang="en">
<head>
<!-- Required meta tags -->
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
<!-- Bootstrap CSS -->
<title>Registration - Attendance Management System</title>
<link rel="stylesheet" href="../resources/Bootstrap/css/bootstrap.min.css">

```

```

<link rel="stylesheet" href="../resources/font-awesome/css/font-awesome.min.css">
<link rel="stylesheet" href="../resources/css/stylesheet.css">
<link rel="stylesheet" href="../resources/Jquery/jquery-ui.css">
<script src="../resources/Jquery/jquery-3.5.1.js"></script>
<script src="../resources/Jquery/jquery-ui.js"></script>
<script src="../resources/Bootstrap/js/bootstrap.min.js"></script>
<script src="../resources/js/sweetalert.min.js"></script>
</head>
<body>
<div id="headerSection" class="sticky-top">
<div class="container" >
<div class="row">
<div class="col-sm-12 text-center pt-3">
<p id="brand">Attendance Management System</p>
</div>
</div>
</div>
</div>
<div id="bodySection">
<div class="container">
<div class="row align-items-center pt-5 text-center">
<div class="col-md-12"><h3>Faculty Registration</h3></div>
</div>
<div class="row py-4">
<div class="col-md-12">
<div id="regSection" class="text-center">
<form id="regForm" enctype="multipart/form-data">
<div class="form-row">
<div class="form-group col-md-2 m-0 p-0"></div>
<div class="form-group col-md-8">
<input id="fname" type="text" class="form-control" placeholder="Full name">
</div>

```

```

<div class="form-group col-md-2 m-0 p-0"></div>
</div>
<div class="form-row">
<div class="form-group col-md-2 m-0 p-0"></div>
<div class="form-group col-md-4">
<input id ="mobile" type="number" class="form- control" placeholder="Mobile">
</div>
<div class="form-group col-md-4">
<input id ="dob" type="text" class="form- control" placeholder="Birth Date">
</div>
<div class="form-group col-md-2 m-0 p-0"></div>
</div>
<div class="form-row">
<div class="form-group col-md-2 m-0 p-0"></div>
<div class="form-group col-md-4">
<input id ="quali" type="text" class="form- control" placeholder="Qualification">
</div>
<div class="form-group col-md-4">
<input id ="uname" name="mobile" type="text" class="form- control" placeholder="Username">
</div>
<div class="form-group col-md-2 m-0 p-0"></div>
</div>
<div class="form-row">
<div class="form-group col-md-2 m-0 p-0"></div>
<div class="form-group col-md-4">
<input id ="uid" type="password" class="form- control" placeholder="User ID">
</div>
<div class="form-group col-md-4">
<input id ="pass" type="password" class="form- control" placeholder="Password">
</div>
<div class="form-group col-md-2 m-0 p-0"></div>
</div>

```

```

<div class="form-row py-1">
<div class="form-group col-md-3"></div>
<div class="form-group col-md-6">
<input onclick="regFun()" type="button" style="background- color:#be2edd" class="form-control btn btn-
success" id="register" value="Register">
</div>
<div class="form-group col-md-3"></div>
</div>
</form>
<a href=".."><button type="button" class="btn btn- primary">Back</button></a>
</div>
</div>
</div>
</div>
</div>
</div>
<script src="../../resources/js/register.js"></script>
</body>
</html>

```

Main Dashboard (HTML)

```

<?php
session_start(); if(!isset($_SESSION['uid'])){
header('location:logout.php');
}
?>
<!doctype html>
<html lang="en">
<head>
<!-- Required meta tags -->
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

```

```

<!-- Bootstrap CSS -->
<title>Dashboard - Attendance Management System</title>
<link rel="stylesheet" href="../resources/Bootstrap/css/bootstrap.min.css">
<link rel="stylesheet" href="../resources/font-awesome/css/font-awesome.min.css">
<link rel="stylesheet" href="../resources/css/stylesheet.css">
<script src="../resources/Jquery/jquery-3.5.1.js"></script>
<script src="../resources/Bootstrap/js/bootstrap.min.js"></script>
<script src="../resources/js/sweetalert.min.js"></script>
</head>
<body>
<div id="headerSection" class="sticky-top">
<div class="container" >
<div class="row">
<div class="col-sm-12 text-center pt-3">
<p id="brand">Attendance Management System</p>
</div>
</div>
</div>
</div>
<div id="bodySection">
<div class="container">
<div class="row align-items-center py-3">
<div class="col-md-10 text-center">
<h3>Welcome <?php echo $_SESSION['faculty']?>!</h3>
</div>
<div class="col-md-2 text-center">
<a href="logout.php"><button class="btn btn-sm btn-primary">Logout <i class="fa fa-user-
circle"></i></button></a>

```

```

</div>
</div>
<hr>
<div class="row py-1">
<div class="col-md-12 text-center">
<form>
<div class="form-row">
<div class="form-group text-center col-md-8">
<h4>Add new student</h4>
</div>
<div class="form-group text-center col-md-4">
<a href="record.php"><button type="button" class="btn btn-sm btn- success">Check record <i class="fa fa-
book"></i></button></a>
</div>
</div>
</div>
</form>
<form>
<div class="form-row pt-3">
<div class="form-group col-md-8">
<input class="form-
control" type="text" placeholder="Student Name" id="name">
</div>
<div class="form-group col-md-4">
<input class="form-
control" type="text" placeholder="Student Roll No" id="roll">
</div>
</div>
</div>
<div class="form-row">
<div class="form-group col-md-4">
<select class="form-control" id="course">

```



```

<option value="">Course</option>
<option value="Diploma">Diploma</option>
<option value="B.E.">B.E.</option>
<option value="M.Tech">M.Tech</option>
</select>
</div>
<div class="form-group col-md-4">
<select class="form-control" id="semester">
<option value="">Semester</option>
<option value="I">I</option>
<option value="II">II</option>
<option value="III">III</option>
<option value="IV">IV</option>
<option value="V">V</option>
<option value="VI">VI</option>
<option value="VII">VII</option>

```

Records (HTML)

```

<?php
session_start(); if(!isset($_SESSION['uid'])){
header('location:logout.php');
}
?>
<!doctype html>
<html lang="en">
<head>
<!-- Required meta tags -->

```

```

<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
<!-- Bootstrap CSS -->
<title>Dashboard - Attendance Management System</title>
<link rel="stylesheet" href="../resources/Bootstrap/css/bootstrap.min.css">
<link rel="stylesheet" href="../resources/font-awesome/css/font-awesome.min.css">
<link rel="stylesheet" href="../resources/css/stylesheet.css">
<script src="../resources/Jquery/jquery-3.5.1.js"></script>
<script src="../resources/Bootstrap/js/bootstrap.min.js"></script>
<script src="../resources/js/sweetalert.min.js"></script>
</head>
<body>
<div id="headerSection" class="sticky-top">
<div class="container" >
<div class="row">
<div class="col-sm-12 text-center pt-3">
<p id="brand">Attendance Management System</p>
</div>
</div>
</div>
</div>
<div id="bodySection">
<div class="container">
<div class="row align-items-center py-3">
<div class="col-md-1 text-center">
<a href="main.php"><button class="btn btn-sm btn-warning">Back</button></a>
</div>
<div class="col-md-9 text-center">

```

```

<h3>Welcome <?php echo $_SESSION['faculty']?>!</h3>
</div>
<div class="col-md-2 text-center">
<a href="logout.php"><button class="btn btn-sm btn-primary">Logout <i class="fa fa-user-
circle"></i></button></a>
</div>
</div>
<hr>
<div class="row">
<div class="col-md-12 text-center">
<div class="py-3" id="studentList"></div>
</div>
</div>
</div>
</div>
<script src="../../resources/js/record.js"></script>
</body>
</html>

```

Dashboard (Javascript)

```
getStudents();

function addStudent() {
var name = $("#name").val();
var course = $("#course").val(); var roll = $("#roll").val();
var semester = $("#semester").val(); var branch = $("#branch").val();
if (name == "" ||
course == "" ||
roll == "" ||
semester == "" ||
branch == "")
{
alert("Fields should be blank!");
} else {
$.ajax({
url: "../api/api.php", type: "POST", dataType: "json",
contentType: "application/json", data: JSON.stringify({
call: 3,
name: name,
course: course,
roll: roll,
semester: semester,
branch: branch,}),
success: function (data) {
if (data == 1) { swal({
title: "Student added!",
text: "New student is added on E-Attendance!", icon: "success",
button: "OK!",
});
$("#name").val("");
$("#course").val("");
$("#roll").val("");
$("#semester").val("");
```

```

$("#branch").val(""); getStudents();
} else {
swal({
title: "Error!",
text: "Some error occurred!", icon: "error",
button: "OK!",
});
},
});
}
}

function searchStudent() {
var course = $("#course1").val();
var semester = $("#semester1").val(); var branch = $("#branch1").val();
if (course == "" || semester == "" || branch == "") { alert("Fields should be blank!");
} else {
$.ajax({
url: "../api/api.php", type: "POST", dataType: "json",
contentType: "application/json", data: JSON.stringify({
call: 6, course: course,
semester: semester, branch: branch,
}),
success: function (data) { if (data.length > 0) {
var sr = 1;
var students = "";
$.each(data, function (i, d) { students +=
"<tr>" +
'<th scope="row">' + sr +
"</th>" +
'<td colspan="2">' + d.name +
"</td>" +
'<td scope="col">' + d.roll_no +
"</td>" +

```

```

'<td scope="col">' + d.course +
"</td>" +
'<td scope="col">' + d.semester + "</td>" +
'<td scope="col">' + d.branch +
"</td>" + "</tr>";
sr++;
});
$("#studentList").html(
'<div class="table-responsive-md" style="background-color:white">' +
'<table class="table table-bordered">' +
"<thead>" + "<tr>" +
'<th scope="col">Sr.no.</th>' +
'<th colspan="2">Name</th>' +
'<th scope="col">Roll No</th>' +
'<th scope="col">Course</th>' +
'<th scope="col">Semester</th>' +
'<th scope="col">Branch</th>' +
"</tr>" +
"</thead>" +
"<tbody>" +
students +
"</tbody>" +
"</table></div>"
);
} else {
$("#studentList").html("<p><b>No matching records found!</b>");
}
},
});
}
}

function getStudents() {
$.ajax({
url: "../api/api.php", type: "POST", dataType: "json",
contentType: "application/json",

```

```

data: JSON.stringify({ call: 4,
}),
success: function (data) { console.log(data);
var studentsData = data[0];
var day = parseInt(data[1][0]); var month = parseInt(data[1][1]); var year = parseInt(data[1][2]); var todayDate
= data[2];
var sr = 1;
var students = ""; var present = 1; var absent = 0; var status = "

$.each(studentsData, function (i, d) {

if((d.date==null) || (todayDate!=d.date && d.status!=0)){ status = '<button class="btn btn-sm btn-
success"  onclick="conFirm(\'"+d.id+\'\',\'"+1+\'\'')"  type="button"><b>P</b></button>  &nbsp; <butt  on
class="btn btn-sm btn-
danger" onclick="conFirm(\'"+d.id+\'\',\'"+0+\'\'')" type="button"><b>A</b></button>'; students +=
"<tr>" +
'<th scope="row">' +
sr +
"</th>" +
'<td colspan="2">' +
d.name +
"</td>" +
'<td scope="col">' +
d.roll_no +
"</td>" +
'<td scope="col">' +
d.course +
"</td>" +
'<td scope="col">' +
d.semester + "</td>" +
'<td scope="col">' +
d.branch +
"</td>" +
'<td scope="col">' +

```

```

status +
"</td>" +
"</tr>";
sr++;
}
else if(day==d.day && month==d.month && year==d.year && d.record==1){
status = '<span class="badge badge-success badge-pill px-3 py-2">Present</span>'; students +=
"<tr>" +
'<th scope="row">' +
sr +
"</th>" +
'<td colspan="2">' +
d.name +
"</td>" +
'<td scope="col">' +
d.roll_no +
"</td>" +
'<td scope="col">' +
d.course +
"</td>" +
'<td scope="col">' +
d.semester + "</td>" +
'<td scope="col">' +
d.branch +
"</td>" +
'<td scope="col">' +
status +
"</td>" +
"</tr>";
sr++;
}
else if(day==d.day && month==d.month && year==d.year && d.record==0){ status = '<span class="badge
badge-danger badge-pill px-3 py-
2" type="button"><b>Absent</b></span>'; students +=

```



```

"<tr>" +
'<th scope="row">' +
sr +
"</th>" +
'<td colspan="2">' +
d.name +
"</td>" +
'<td scope="col">' +
d.roll_no +
"</td>" +
'<td scope="col">' +
d.course +
"</td>" +
'<td scope="col">' +
d.semester +
"</td>" +
'<td scope="col">' +
d.branch +
"</td>" +
'<td scope="col">' +
status +
"</td>" +
"</tr>";
sr++;
}
else{
return null;
}
});
$("#studentList").html(
'<div class="table-responsive-md" style="background-color:white">' +
'<table class="table table-bordered">' +
"<thead>" +
"<tr>" +

```

```

'<th scope="col">Sr.no.</th>' +
'<th colspan="2">Name</th>' +
'<th scope="col">Roll No</th>' +
'<th scope="col">Course</th>' +
'<th scope="col">Semester</th>' +
'<th scope="col">Branch</th>' +
'<th scope="col">Attendance</th>' +
"</tr>" +
"</thead>" +
"<tbody>" +
students +
"</tbody>" +
"</table></div>"
);
},
});
}
function conFirm(id, record){ var sid = id;
var record = record; swal({
title: 'Are you sure?',
text: "Confirm once before marking attendance!", icon: "warning",
buttons: ['Cancel', 'Confirm'], dangerMode: true,
})
.then((ok) => { if (ok) {
addAttendance(sid, record);
} else {
swal("Think again!");
}
});
}
function addAttendance(sid, record) { var sid = sid;
var record = record;
$.ajax({

```

```

url: "../api/api.php", type: "POST", dataType: "json",
contentType: "application/json", data: JSON.stringify({
call: 8, sid :sid,
record : record
}),
success: function (data) { if (data == 1) {
getStudents();
} else {
swal({
title: "Error!",
text: "Some error occurred!", icon: "error",
button: "OK!",
});
}
},
});
}

```

Records (Javascript)

```

getStudents();
function getStudents() {
$.ajax({
url: "../api/api.php", type: "POST", dataType: "json",
contentType: "application/json", data: JSON.stringify({
call: 12,
}),
success: function (data) { var studentsData = data; var sr = 1;
var students = ""; var record = ""; var sid = "";
var studentList = ""; var status = "";
$.each(studentsData, function (i, d) { console.log(studentsData);
status = (d.record==1) ? '<span class="badge badge-success badge-pill px-3 py- 2">Present</span>' : '<span
class="badge badge-danger badge-pill px-3 py- 2"><b>Absent</b></span>';

```

```

if(record==""){ students = "<tr>" +
'<th scope="row">' +
sr +
"</th>" +
'<td colspan="2">' +
d.name +
"</td>" +
'<td scope="col">' +
d.roll_no +
"</td>" +
'<td scope="col">' +
d.course +
"</td>" +
'<td scope="col">' +
d.semester +
"</td>" +
'<td scope="col">' +
d.branch +
"</td>" +
'<td scope="col">' +
d.date +
"</td>" +
'<td scope="col">' +
status +
"</td>" +
"</tr>";
studentList += '<h5 class="pb-2">Attendance date: '+d.date+'</h5><div class="table- responsive-md"
style="background-color:white">' +
'<table class="table table-bordered">' +
"<thead>" +
"<tr>" +
'<th scope="col">Sr.no.</th>' +
'<th colspan="2">Name</th>' +
'<th scope="col">Roll No</th>' +

```

```

'<th scope="col">Course</th>' +
'<th scope="col">Semester</th>' +
'<th scope="col">Branch</th>' +
'<th scope="col">Date</th>' +
'<th scope="col">Status</th>' +
"</tr>" +
"</thead>" +
"<tbody>" +
    students;
record = d.date;
sr++;
}
else if(record==d.date){
students = "<tr>" +
'<th scope="row">' +
    sr +
"</th>" +
'<td colspan="2">' +
    d.name +
"</td>" +
'<td scope="col">' +
    d.roll_no +
"</td>" +
'<td scope="col">' +
    d.course +
"</td>" +
'<td scope="col">' +
    d.semester +
"</td>" +
'<td scope="col">' +
    d.branch +
"</td>" +
'<td scope="col">' +
    d.date +

```

```

"</td>" +
'<td scope="col">' +
status +
"</td>" +
"</tr>";
studentList+=students;
record = d.date;
sr++;
}
else if(record!=d.date){
sr = 1;
students = "<tr>" +
'<th scope="row">' +
sr +
"</th>" +
'<td colspan="2">' +
d.name +
"</td>" +
'<td scope="col">' +
d.roll_no +
"</td>" +
'<td scope="col">' +
d.course +
"</td>" +
'<td scope="col">' +
d.semester +
"</td>" +
'<td scope="col">' +
d.branch +
"</td>" +
'<td scope="col">' +
d.date +
"</td>" +
'<td scope="col">' +

```

```

status +
"</td>" +
"</tr>";
studentList+="</tbody>" +
'</table></div><br><h5 class="pb-
2">Attendance date: '+d.date+'</h5><div class="table-responsive-md" style="background-
color:white">' +
'<table class="table table-bordered">' + "<thead>" +
"<tr>" +
'<th scope="col">Sr.no.</th>' +
'<th colspan="2">Name</th>' +
'<th scope="col">Roll No</th>' +
'<th scope="col">Course</th>' +
'<th scope="col">Semester</th>' +
'<th scope="col">Branch</th>' +
'<th scope="col">Date</th>' +
'<th scope="col">Status</th>' +
"</tr>" +
"</thead>" +
"<tbody>" +
students;
record = d.date;
sr++;
}
});
$("#studentList").html(studentList);
},
});
}

```

Registration (Javascript)

```
$(document).ready(function () {  
    $("#dob").datepicker({  
        maxDate: 0,  
        changeMonth: true,  
        changeYear: true,  
        yearRange: "1950:2020",  
    });  
});  
  
function regFun() {  
    var fname = $("#fname").val();  
    var uname = $("#uname").val();  
    var uid = $("#uid").val();  
    var quali = $("#quali").val();  
    var pass = $("#pass").val();  
    var dob = $("#dob").val();  
    var mobile = $("#mobile").val();  
    if (  
        fname == "" ||  
        uname == "" ||  
        uid == "" ||  
        quali == "" ||  
        pass == "" ||  
        dob == "" ||  
        mobile == ""  
    ) {  
        alert("Fields should be blank!");  
    } else {  
        $.ajax({  
            url: "../api/api.php",  
            type: "POST",  
            dataType: "json",  
            contentType: "application/json",  
        });  
    }  
}
```



```
data: JSON.stringify({
call: 1,
fname: fname,
  uname: uname,
  uid: uid,
quali: quali,
pass: pass,
dob: dob,
mobile: mobile,
}),
success: function (data) { if (data == 1) {
swal({
title: "Registration successfull!",
text: "Welcome to Online Attendance Management System!",
  icon: "success",
button: "OK!",
}).then((value) => { window.location = "../";
});
} else if (data == 2) { swal({
title: "User already exists!",
text: "Mobile number is already taken. Try another!",
  icon: "warning",
button: "OK!",
});
} else {
swal({
title: "Error!",
text: "Some error occured!",
  icon: "error",
button: "OK!",
});
}
},
});
```

```
}  
}
```

Login (Javascript)

```
function loginFun() {  
var uid = $("#uid").val();  
var pass = $("#pass").val();  
if (uid == "" || pass == "") {  
alert("Fields cannot be empty!");  
} else {  
$.ajax({  
url: "api/api.php",  
type: "POST",  
dataType: "json",  
contentType: "application/json",  
data: JSON.stringify({  
call: 2,  
uid: uid,  
pass: pass,  
}),  
success: function (data) { if (data == 1) {  
window.location = "routes/main.php";  
} else {  
swal({  
title: "Invalid credentials!",  
text: "Enter proper details!",  
icon: "error",  
button: "OK!",  
});  
}  
},  
});  
}  
}
```

Forget Password (Javascript)

```
$(document).ready(function () {
    $("#dob").datepicker({
        maxDate: 0,
        changeMonth: true,
        changeYear: true,
        yearRange: "1950:2020",
    });
    $("#changePass").click(function () {
        var uid = $("#uid").val();
        var dob = $("#dob").val();
        var pass = $("#pass").val();
        if (uid == "" || dob == "" || pass == "") {
            alert("Fields cannot be left blank!");
        } else {
            $.ajax({
                url: "../api/api.php",
                type: "POST",
                dataType: "json",
                contentType: "application/json",
                data: JSON.stringify({
                    call: 5,
                    uid: uid,
                    dob: dob,
                    pass: pass,
                }),
                success: function (data) {
                    if (data == 1) {
                        swal({
                            title: "Password changed!", text: "New password is set!", icon: "success",
                            button: "OK!",
                        }).then((vote) => { window.location = "../";
```

```

});
} else {
swal({
title: "Invalid Credentials!",
text: "Invalid! user id / date of birth",
icon: "error",
button: "OK!",
});
}
},
});
}
});
});
});

```

Backened API (PHP)

```

<?php
session_start(); include('connect.php');
$json = json_decode(file_get_contents("php://input"),true);
// Faculty registration if($json['call']==1){
$fname = $json['fname'];
$uname = $json['uname'];
$uid = $json['uid'];
$mobile = $json['mobile'];
$dob = $json['dob'];
$quali = $json['quali'];
$pass = md5($json['pass']);
$date = date('d-m-Y');
$check=mysqli_query($con,"select*fromfacultywheremobile='$mobile'");
if(mysqli_num_rows($check)>0){
echo json_encode($response['success']=2);
}
else{

```

```

$query = mysqli_query($con, "insert into faculty (name, mobile, dob, qualification, username, uid, password,
created_at) values('$fname','$mobile','$dob','$quali', '$uname', '$uid', '$pass', '$date')");
if($query){
echo json_encode($response['success']=1);
}
else{
echo json_encode($response['success']=0);
}
}
}

// Faculty login
if($json['call']==2){
$uid = $json['uid'];
$pass = md5($json['pass']);
$check = mysqli_query($con, "select * from faculty where uid='$uid' and password='$pass'");
if(mysqli_num_rows($check)>0){
$_SESSION['uid']=$uid;
$fetch = mysqli_fetch_array($check);
$_SESSION['faculty'] = $fetch['username'];
echo json_encode($response['success']=1);
}
else{
echo json_encode($response['success']=0);
}
}

// Add student
if($json['call']==3){
$name = $json['name'];
$course = $json['course'];
$roll = $json['roll'];
$semester = $json['semester'];
$branch = $json['branch'];
$date = date('d-m-Y');

```

```

$query = mysqli_query($con, "insert into student (name, roll_no, course, semester, branch, created_at)
values('$name','$roll','$course','$semester', '$branch', '$date')");
if($query){
echo json_encode($response['success']=1);
}
else{
echo json_encode($response['success']=0);
}
}

// Get students
if($json['call']==4){
$day = date('d');
$month = date('m');
$year = date('Y');
$date = [$day, $month, $year];
$todayDate = date('d-m-Y');
$getStudents = mysqli_query($con, "select student.id, student.name, student.roll_no, student.course,
student.semester, student.branch, student.created_at, record.sid, record.status, record.record, record.day,
record.month, record.year, record.date from student LEFT JOIN record ON student.id = record.sid ORDER
BY student.id ASC");
if(mysqli_num_rows($getStudents)>0){
$students = mysqli_fetch_all($getStudents, MYSQLI_ASSOC);
$empty = mysqli_free_result($getStudents);
echo json_encode([$students, $date, $todayDate]);
}
else{
echo json_encode($response['success']=0);
}
}

// Forget password
if($json['call']==5){
$uid = $json['uid'];
$dob = $json['dob'];
$pass = md5($json['pass']);

```

```

$check = mysqli_query($con, "select * from faculty where uid='$uid' and dob='$dob'");
if(mysqli_num_rows($check)>0){
$update = mysqli_query($con, "update faculty set password='$pass' where uid='$uid' and dob='$dob'");
if($update){
echo json_encode($response['success']=1);
}
else{
echo json_encode($response['success']=0);
}
}
else{
echo json_encode($response['success']=0);
}
}

// Search student
if($json['call']==6){
$course = $json['course'];
$semester = $json['semester'];
$branch = $json['branch'];
$getStudents = mysqli_query($con, "select * from student where course='$course' and semester='$semester' and branch='$branch'");
if(mysqli_num_rows($getStudents)>0){
    $students = mysqli_fetch_all($getStudents, MYSQLI_ASSOC);
    $empty = mysqli_free_result($getStudents); echo json_encode($students);
}
else{
echo json_encode($response['success']=0);
}
}

// Populate students data
if($json['call']==7){
$getStudents = mysqli_query($con, "select * from student"); if(mysqli_num_rows($getStudents)>0){
    $students = mysqli_fetch_all($getStudents, MYSQLI_ASSOC);
    $empty = mysqli_free_result($getStudents); echo json_encode($students);
}
}

```

```

else{
echo json_encode($response['success']=0);
}
}

// Add record
if($json['call']==8){
$sid = $json['sid'];
$record = $json['record'];
$day = date('d');
$month = date('m');
$year = date('Y');
$date = date('d-m-Y');
$changeStatus = mysqli_query($con,"update record set status=0 where sid=$sid"); if($changeStatus){
$query = mysqli_query($con, "insert into record (sid, status, record, date, day, month, year) values('$sid',
1 , '$record', '$date', '$day', '$month', '$year')"); if($query){
echo json_encode($response['success']=1);
}
else{
echo json_encode($response['success']=0);
}
}
}

// Get records
if($json['call']==9){
$getRecords = mysqli_query($con, "select student.name, student.course, student.branch, stu dent.semester,
record.subject_name, record.sid, record.id, record.subject_code, record.esse, record.ct, record.ta from student
LEFT JOIN record ON student.id=record.sid WHERE student.id=rec ord.sid ");
if(mysqli_num_rows($getRecords)>0){
$records = mysqli_fetch_all($getRecords, MYSQLI_ASSOC);
$empty = mysqli_free_result($getRecords); echo json_encode($records);
}
else{
echo json_encode($response['success']=0);
}
}

```



```

}
}
// Edit record
if($json['call']==10){

$sid = $json['sid'];
$id = $json['id'];
$getRecord = mysqli_query($con, "select ese, id, sid, ct, ta from record WHERE sid='$sid' and id='$id' ");
if(mysqli_num_rows($getRecord)>0){
$record = mysqli_fetch_array($getRecord, MYSQLI_ASSOC);
$empty = mysqli_free_result($getRecord); echo json_encode($record);
}
else{
echo json_encode($response['success']=0);
}
}

// Save record
if($json['call']==11){
$sid = $json['sid'];
$id = $json['id'];
$new_ese = $json['new_ese'];
$new_ct = $json['new_ct'];
$new_ta = $json['new_ta'];
$update = mysqli_query($con, "update record set ese='$new_ese', ct='$new_ct', ta='$new_ta' where sid='$sid'
and id='$id'");
if($update){
echo json_encode($response['success']=1);
}
else{
echo json_encode($response['success']=0);
}
}

if($json['call']==12){

```

```

$query = mysqli_query($con, "select student.id, student.name, student.roll_no, student.cou rse,
student.semester, student.branch, student.created_at, record.sid, record.status, record.d ate, record.record from
student LEFT JOIN record ON student.id = record.sid WHERE student.id = record.sid ORDER BY record.id
DESC");
if(mysqli_num_rows($query)>0){
$records = mysqli_fetch_all($query, MYSQLI_ASSOC);
$empty = mysqli_free_result($query); echo json_encode($records);
}
else{
echo json_encode(0);
}
}
?>

```

Database Connectivity (PHP)

```

<?php
$con = mysqli_connect('localhost','root','','student- attendance') or die('connection error');
?>

```

Logout (PHP)

```

<?php
session_start();
session_destroy();
header('location:../index.php');
?>

```

Stylesheet (CSS)

```
#headerSection {  
background-color: #be2edd;  
}
```

```
#brand {  
font-weight: 600;  
color: white;  
font-style: italic;  
font-family: cursive;  
font-size: xx-large;  
}
```

```
#loginSection {  
padding: 15px;  
border: 1px solid #be2edd;  
background-color: white;  
border-radius: 20px;  
}
```

```
#adminSection { padding: 15px;  
border: 1px solid #34495e;  
background-color: white;  
border-radius: 20px;  
}
```

```
#component {  
background-color: white;  
border-radius: 10px;  
}
```

8. System Testing

Implementation and Testing:

Implementation is one of the most important tasks in project is the phase in which one has to be cautions because all the efforts undertaken during the project will be very interactive. Implementation is the most crucial stage in achieving successful system and giving the users confidence that the new system is workable and effective. Each program is tested individually at the time of development using the sample data and has verified that these programs link together in the way specified in the program specification. The computer system and its environment are tested to the satisfaction of the user.

Implementation

The implementation phase is less creative than system design. It is primarily concerned with user training, and file conversion. The system may be requiring extensive user training. The initial parameters of the system should be modifies as a result of a programming. A simple operating procedure is provided so that the user can understand the different functions clearly and quickly. The different reports can be obtained either on the inkjet or dot matrix printer, which is available at the disposal of the user. The proposed system is very easy to implement. In general implementation is used to mean the process of converting a new or revised system design into an operational one.

Testing

Testing is the process where the test data is prepared and is used for testing the modules individually and later the validation given for the fields. Then the system testing takes place which makes sure that all components of the system property functions as a unit. The test data should be chosen such that it passed through all possible condition. Actually testing is the state of implementation which aimed at ensuring that the system works accurately and efficiently before the actual operation commence. The following is the description of the testing strategies, which were carried out during the testing period.

System Testing

Testing has become an integral part of any system or project especially in the field of information technology. The importance of testing is a method of justifying, if one is ready to move further, be it to be check if one is capable to with stand the rigors of a particular situation cannot be underplayed and that is why testing before development is so critical. When the software is developed before it is given to user to use the software must be tested whether it is solving the purpose for which it is developed. This testing involves various types through which one can ensure the software is reliable. The program was tested logically and pattern of execution of the program for a set of data are repeated. Thus the code was exhaustively checked for all possible correct data and the outcomes were also checked.

Module Testing

To locate errors, each module is tested individually. This enables us to detect error and correct it without affecting any other modules. Whenever the program is not satisfying the required function, it must be corrected to get the required result. Thus all the modules are individually tested from bottom up starting with the smallest and lowest modules and proceeding to the next level. Each module in the system is tested separately. For example the job classification module is tested separately. This module is tested with different job and its approximate execution time and the result of the test is compared with the results that are prepared manually. The comparison shows that the results proposed system works efficiently than the existing system. Each module in the system is tested separately. In this system the resource classification and job scheduling modules are tested separately and their corresponding results are obtained which reduces the process waiting time.

Integration

Testing After the module testing, the integration testing is applied. When linking the modules there may be chance for errors to occur, these errors are corrected by using this testing. In this system all modules are connected and tested. The testing results are very correct. Thus the mapping of jobs with resources is done correctly by the system. Acceptance Testing When that user find no major problems with its accuracy, the system passers through a final acceptance test. This test confirms that the system needs the original goals, objectives and requirements established during analysis without actual execution which elimination wastage of time and money acceptance tests on the shoulders of users and management, it is finally acceptable and ready for the operation.

9 .OUTPUT SCREEENS

9.1 Screenshots:

Homepage

Attendance Management System

CMR Engineering College



Faculty Login



User ID

Password

Login

Forget password? [Click here](#)

New registration? [Click here](#)

Registration Page

Attendance Management System

Faculty Registration

Full name

Mobile

Birth Date

Qualification

Username

User ID

Password

Register

Back

Faculty Dashboard

Attendance Management System

Welcome rutuja!

Logout

Add new student

Check record

Student Name

Student Roll No

Course

Semester

Branch

Add student

Mark Attendance 21-04-2021

Sr.no.	Name	Roll No	Course	Semester	Branch	Attendance
1	Shoaib	101	Diploma	I	Mechanical	Present
2	shakib	102	Diploma	I	Mechanical	Absent
3	zamir	123	Diploma	I	Mechanical	Present

Adding Student

Attendance Management System

Welcome rutuja!

Logout

Add new student

Check record

Student Name

Student Roll No

Course

Branch

Add student

Student added!

New student is added on E-Attendance!

OK

Mark Attendance 21-04-2021

Sr.no.	Name	Roll No	Course	Semester	Branch	Attendance
1	Shoaib	101	Diploma	I	Mechanical	Present
2	shakib	102	Diploma	I	Mechanical	Absent
3	zamir	123	Diploma	I	Mechanical	Present
4	Ram	101	Diploma	I	Mechanical	P A

Marking Attendance

Attendance Management System

Welcome rutuja!

Logout

Add new student

Check record

Student Name

Student Roll No

Course

Branch

Add student

Are you sure?

Confirm once before marking attendance!

Cancel

Confirm

Mark Attendance 21-04-2021

Sr.no.	Name	Roll No	Course	Semester	Branch	Attendance
1	Shoaib	101	Diploma	I	Mechanical	Present
2	shakib	102	Diploma	I	Mechanical	Absent
3	zamir	123	Diploma	I	Mechanical	Present
4	Ram	101	Diploma	I	Mechanical	P A

Attendance Records

Attendance Management System

Back

Welcome rutuja!

Logout

Attendance date: 21-04-2021

Sr.no.	Name	Roll No	Course	Semester	Branch	Date	Status
1	Ram	101	Diploma	I	Mechanical	21-04-2021	Present
2	zamir	123	Diploma	I	Mechanical	21-04-2021	Present
3	shakib	102	Diploma	I	Mechanical	21-04-2021	Absent
4	Shoaib	101	Diploma	I	Mechanical	21-04-2021	Present

Forgot Password

Attendance Management System	
Forgot Password	
<div><input type="text" value="Enter User ID"/> <input type="text" value="Enter Birth Date"/> <input type="text" value="New password"/> Change Back</div>	

Database Structure

phpMyAdmin

Server: 127.0.0.1 » Database: student-attendance

Structure SQL Search Query Export Import Operations Privileges Routines Events More

Filters

Containing the word:

Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> faculty	Browse Structure Search Insert Empty Drop	1	InnoDB	latin1_swedish_ci	16 K1B	-
<input type="checkbox"/> record	Browse Structure Search Insert Empty Drop	4	InnoDB	latin1_swedish_ci	16 K1B	-
<input type="checkbox"/> student	Browse Structure Search Insert Empty Drop	4	InnoDB	latin1_swedish_ci	16 K1B	-
3 tables Sum		9	InnoDB	latin1_swedish_ci	48 K1B	0 B

☐ Check all With selected:

[Print](#) [Data dictionary](#)

[Create table](#)

Name: Number of columns: 4

[Go](#)

Console

10. CONCLUSION

The Attendance Management System (AMS) project has been successfully developed to automate the process of tracking and managing attendance records. The system provides a user-friendly interface for users to register, log in, and mark their attendance, while administrators have access to additional features such as generating reports and managing user accounts.

11. REFERENCES

- [1] Patel UA, Swaminarayan Priya R. Development of a student attendance management system using RFID and face recognition: a review. *Int J Adv Res Comput Sci Manag Stud*. 2014;2(8):109–19.
- [2] Jacksi K. Design and Implementation of Online Submission And Peer Review System: A Case Study Of E-Journal Of University Of Zakho. *Int J Sci Technol Res*. 2015;4(8):83–5.
- [3] Anitha V Pai, Krishna A, Kshama PM, Correa M. Web service for student attendance management system. *www.ijarse.com*. 2016 Mar;5(3).
- [4] Srinidhi M, Roy R. A web enabled secured system for attendance monitoring and real time location tracking using Biometric and Radio Frequency Identification (RFID) technology. In *IEEE*; 2015. p. 1–5.
- [5] Kassim M, Mazlan H, Zaini N, Salleh MK. Webbased student attendance system using RFID technology. In *IEEE*; 2012. p. 213–8.

