International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 05 | May - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Collaborative Academic Tracking System

Prof. A. M. Gunje
Computer Science and Engineering
N.B.Navale Sinhgad College of Engineering
Solapur,India
Arungunj@gmail.com

Ms. Radhika Raut
Computer Science And Engineering
N.B.Navale Sinhgad College of Engineering
Solapur,India
radhikaraut2002@gmail.com

Ms. Apeksha Phulari
Computer Science And Engineering
N.B.Navale Sinhgad College of Engineering
Solapur,India
apekshaphulari12@gmail.com

Mr. Nitishkumar Khavekar
Computer Science And Engineering
N.B.Navale Sinhgad College of Engineering
Solapur,India
khavekarnitishkumar@gmail.com

Mr. Ritesh Kengar
Computer Science And Engineering
N.B.Navale Sinhgad College of Engineering
Solapur,India
kengarritesh3@gmail.com

ABSTRACT:

In modern educational environments, efficient monitoring of student performance and participation is essential for ensuring academic success, institutional transparency, and campus security. However, many institutions still rely on outdated, manual processes to manage academic records, track student involvement in college activities, and authorize campus movements. These traditional systems often lead to data inaccuracies, time delays, limited accessibility, and poor coordination among departments.

This research introduces a comprehensive Academic Tracking System, developed using PHP and MySQL, which addresses these challenges by providing a centralized digital platform to manage student academic records, attendance, internal assessments, and participation in college activities such as seminars, workshops, sports, and cultural events. A key component of this system is the Gate Pass Facility, which enables students to submit exit requests digitally. These requests are reviewed and approved or rejected by faculty or administrators, and the entire process is logged for accountability and security purposes.

The system features role-based access control, offering tailored interfaces for administrators, faculty, and students. It allows real-time data entry and access, automated report generation, and interactive dashboards for performance visualization. Administrators can monitor institutional activities, faculty can manage academic inputs and event participation, and students can view their academic records, achievements, and gate pass status from a unified portal.

Pilot testing within a simulated academic setup showed that the system significantly reduced manual workload, improved record accuracy, increased student engagement through better visibility of their progress, and enhanced campus security. By integrating academic tracking with event participation and campus movement management, this system offers an end-to-end digital solution tailored for modern educational institutions.

INTRODUCTION

Educational institutions today face growing demands to manage student data with greater accuracy, speed, and transparency. Traditional manual systems for tracking student performance, attendance, and participation in college activities are increasingly proving to be inefficient and error-prone. These methods often involve physical registers, spreadsheets, and disconnected administrative workflows, which can result in data duplication, loss of information, delayed interventions, and limited accessibility to stakeholders.

participation in extracurricular activities. This results in **several operational, academic, and administrative challenges** that affect both institutional effectiveness and student development.

participation in seminars, cultural events, sports, and other institutional programs. However, most institutions lack a unified system to track and evaluate the overall involvement and development of students across these domains. This fragmented approach limits the ability of faculty and administrators to gain a holistic view of each student's academic journey.

Additionally, the physical movement of students within and outside the campus is typically managed through manual gate pass systems, which offer minimal control and leave room for misuse or unauthorized exits. This creates security risks and reduces accountability.

To address these multifaceted challenges, we propose a centralized **Academic Tracking System** developed using **PHP and MySQL**, designed to automate and integrate the following key components:

- **Academic Performance Tracking**: Records attendance, internal marks, and subject-wise performance data.
- **Activity Participation Monitoring**: Tracks involvement in college events, workshops, clubs, and other extracurricular engagements.
- **Gate Pass Management**: Allows students to digitally request permission to exit the campus, with approval workflows and logs maintained for security and administrative purposes.

The system provides **role-based access** for administrators, faculty, and students, ensuring each user interacts with relevant features. It enables **real-time data entry**, **automated report generation**, and **dashboard-based visualizations**, enhancing the decision-making process for academic staff while also keeping students informed about their progress.

By unifying academic records, event participation, and gate pass management under one digital platform, the system offers a comprehensive, scalable solution that improves data accuracy, institutional efficiency, and campus safety. This research explores the design, implementation, and benefits of such a system within a college environment.

PROBLEM DEFINITION

Educational institutions are responsible not only for imparting knowledge but also for monitoring, evaluating, and guiding students throughout their academic journey. However, the conventional systems used for managing student data are often outdated, inefficient, and fragmented. Most colleges still rely on **manual or partially digitized methods** to record attendance, internal

Moreover, students are engaged in various academic and extracurricular activities throughout their time in college—ranging from classroom attendance, internal assessments, and project submissions to

One of the most significant issues is the **lack of a centralized system** to record and monitor student academic performance and activity participation. Different departments or faculty members often maintain their own records using notebooks, Excel sheets, or isolated desktop applications. This **decentralized data storage** leads to duplication of work, inconsistency in records, delays in report generation, and an inability to obtain a comprehensive view of a student's performance.

Moreover, students frequently participate in activities beyond the classroom, such as seminars, workshops, sports, and cultural events. Unfortunately, their involvement in these activities is rarely documented in an organized way. As a result, faculty and administrators struggle to assess the overall engagement and personal development of each student. The **absence of consolidated data** makes it difficult to identify students who are underperforming academically or disengaged from institutional life, thus delaying necessary interventions.

Another critical concern is the **manual gate pass system** that many institutions continue to use. Students are typically required to obtain physical signatures from faculty members to leave the campus, and these paper-based passes are easy to forge, misplace, or misuse. There is often no systematic record of when and why students left campus or who approved their exit. This creates **gaps in campus security** and reduces administrative accountability.

Additionally, the lack of real-time accessibility to academic and administrative data causes delays in communication and decision-making. Students often remain unaware of their academic standing or internal marks until the end of the semester, and faculty members must spend significant time compiling reports that could otherwise be generated automatically in a digital environment.

Given these limitations, there is a **pressing need** for a fully integrated, web-based solution that can:

- Consolidate academic and extracurricular records,
- Provide reliable and secure digital gate pass functionality,
- Improve data transparency for students, faculty, and administrators,
- Reduce manual workload and error rates,
- Enhance institutional control and student security.

This research project aims to solve these problems by developing a centralized **Academic Tracking System**, using open-source technologies (PHP and MySQL), to automate and streamline these critical academic and administrative processes.

SCOPE AND OBJECTIVE

SCOPE

he **Academic Tracking System** is designed as a comprehensive, web-based solution that digitizes and centralizes the academic and administrative records of students in a college environment. Its scope extends across various key areas of student monitoring and institutional management. The system covers the recording, retrieval, and analysis of student attendance, internal assessment marks, and involvement in extracurricular and co-curricular activities.

It also includes a secure **Gate Pass Module**, allowing students to request permission to leave campus through an authenticated, trackable digital process—thus strengthening campus safety and reducing paperwork.

The system serves three primary user types: **Faculty**, **Students and Parents**. Each role is given specific access rights and functionalities according to institutional needs. The scope does not extend to external data sources (such as government or university examination systems) but assessments, and focuses solely on internal academic and administrative management within an institution.

This project is developed using **PHP** for server-side scripting and **MySQL** for the database, making it scalable and easy to deploy on commonly used web infrastructures in colleges.

In essence, the system provides a **digital foundation** for improving transparency, accuracy, and coordination across academic departments, student services, and campus security.

OBJECTIVE

1. **To centralize student academic records** – by digitally maintaining attendance, internal marks, and academic progress over semesters in a structured format.
2. **To monitor extracurricular and co-curricular participation** – by allowing faculty or activity coordinators to record student involvement in events, workshops, and competitions.
3. **To implement a digital gate pass system** – to enhance campus security by tracking student movement in and out of the campus through verified online approvals.
4. **To reduce manual workload and human error** – by automating repetitive administrative tasks such as report generation, data entry, and approvals.
5. **To increase transparency and accessibility** – by allowing students to view their academic data and gate pass status in real-time, improving awareness and responsibility.
6. **To support informed decision-making** – by equipping faculty and administrators with dashboards and reports that reflect academic trends, attendance shortfalls, and participation metrics.
7. **To establish role-based access control** – to ensure data integrity by allowing only authorized users to update or access specific parts of the system.

PROPOSED METHODOLOGY

SYSTEM ARCHITECTURE

The architecture of the Academic Tracking System is based on a structured, modular design that integrates a client-server model using web technologies. It is primarily built using PHP for the backend logic, MySQL for the database, and HTML with CSS for the frontend interface. The system is designed to handle academic processes efficiently within a college environment, including attendance management, internal assessment tracking, activity logging, gate pass generation, and communication with stakeholders like students, faculty, and parents.

At the core of the architecture lies the user interaction layer, where users such as faculty, students, and optionally parents access the system through a web browser. These users interact with the system's frontend, which is developed using HTML and CSS to provide a responsive and accessible interface. The frontend sends user inputs and requests to the backend where PHP handles the processing logic**.** The backend is responsible for interpreting form submissions, managing user sessions, and ensuring data is correctly inserted, updated, or fetched from the database. It includes session management to maintain secure and role-based access throughout the system. For example, once a user logs in, the session management component verifies their role and grants them access only to the features and data relevant to their privileges—faculty may record

Data management occurs in the backend where PHP scripts act as form execute SQL queries to store or retrieve information, and communicate with the MySQL database. This interaction. the system to handle complex transactions such as updating attendance logs, submitting assignment results, managing improvement feedback, and processing gate pass applications.

The MySQL database forms the data layer of the architecture and is named education_ platform. It consists of various tables that store all institutional records in a structured format. The students table holds student profiles and academic data, while the faculty table stores information about teachers and their assigned responsibilities. Attendance data is stored in a dedicated table that logs student presence per subject and date. Internal assessment results are stored in the marks table, and assignment submissions are managed in the assignments table. The system also maintains records of improvement areas to help faculty track student performance feedback. Student queries and administrative issues are tracked in the queries table, whereas the gate pass system uses its own table to store each request with corresponding statuses and timestamps. Additionally, an sms logs table maintains a history of notifications sent to students and parents regarding academic activities.
When a user initiates an action—such as logging in, submitting attendance, or requesting a gate pass—the request flows from the frontend to the backend, which processes it and interacts with the database to perform the required operation. The processed result, whether it's a data update or a report generation, is then sent back to the frontend to be displayed to the user. This systematic flow of data ensures that all operations within the academic system are streamlined, secure, and accessible to authorized users.

The architectural design supports modularity and scalability, allowing the addition of future enhancements like biometric integration, mobile apps, or automated analytics dashboards. Moreover, the use of PHP and MySQL ensures cost-effective deployment, making the system suitable for educational institutions of various sizes.

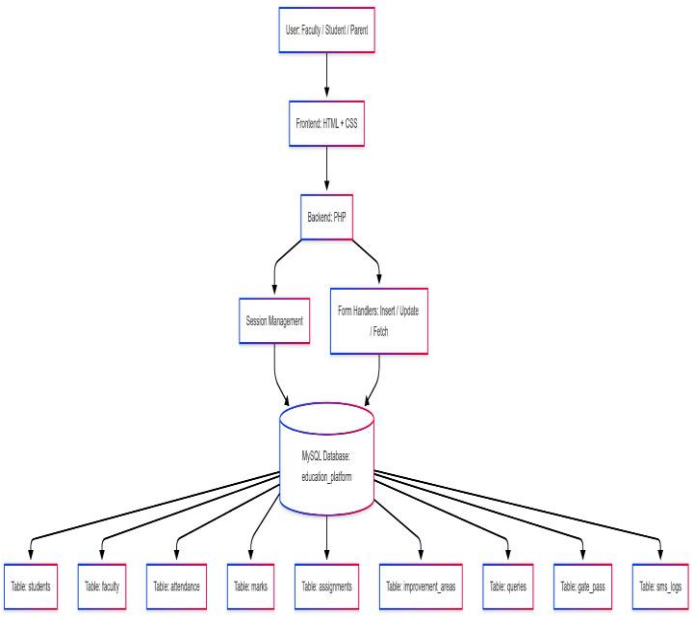

Fig. 1 Architecture of System

attendance and marks, while students can view their data and request gate passes.

Data management occurs in the backend where PHP scripts act as form handlers. These handlers validate the data provided by users,

MODULE

The Academic Tracking System is composed of multiple interconnected modules that facilitate the management of student academic records, communication, and administrative controls. Each module has its own responsibilities but is integrated into the same database and backend logic to ensure centralized and real-time data flow.

The Student Module is one of the core components of the system. It manages student registration and profiles, storing essential information such as name, roll number, department, contact details, and login credentials. Once logged in, students can view their attendance records, internal marks, assignments, improvement feedback, and gate pass history. They can also request new gate passes or raise queries to the faculty or administrative staff. All interactions within this module are controlled by user-specific access rights to ensure students can only access their own data.

The Faculty Module enables teachers to manage academic activities for the subjects they are assigned. This includes marking daily attendance, entering internal assessment marks, uploading assignment details, providing performance feedback, and responding to student queries. Faculty members can also view statistics and reports about their classes, such as attendance percentages and average marks. Moreover, they have authority to approve or reject gate pass requests initiated by students, with the ability to include remarks for each decision.

The Attendance Module allows faculty members to record attendance data and provides students with real-time access to their attendance status. It captures data such as the date, subject, and presence or absence of each student. This module also includes functionality to generate attendance reports for administrative or parent review and calculates attendance percentages automatically.

The Marks Module is responsible for handling internal assessments, test scores, and other evaluative metrics for students. Faculty can enter and update marks for multiple assessments. Students can view their performance subject-wise through this module. It serves as a continuous evaluation record and helps in identifying areas where academic support is needed.

The Assignments Module allows faculty to upload assignment topics, submission deadlines, and grading criteria. Students can access these assignments and submit their completed work through the system. Faculty can then review submissions, assign grades, and provide feedback, making the process completely digital and traceable.

The Improvement Area Module is designed to support academic enhancement. Faculty members can enter observations about student performance and suggest specific areas for improvement. These records help students understand their weaknesses and allow the institution to provide targeted mentoring or remedial sessions.

The Gate Pass Module handles the generation and tracking of student gate pass requests. Students initiate a request stating the reason and

The SMS Notification Module is responsible for sending alerts and reminders to students and parents. For example, if a student's attendance drops below a threshold or a gate pass is approved, an SMS notification can be sent automatically. This enhances communication and keeps parents informed of important events.

TECHNOLOGY STACK

Frontend:

- **HTML** – For structuring the web pages and user interface components.
- **CSS** – For styling and responsive layout design of the user interface.

Backend:

- **PHP** – Server-side scripting language used to handle business logic, form processing, and database communication.

Database:

- **MySQL** – Relational database used to store and manage all data related to students, faculty, attendance, marks, assignments, gate passes, and communication logs.

Session & Authentication:

- PHP Sessions – For secure login, user tracking, and access control across the platform.

Server Environment:

- **XAMPP** – Local server environment used for development and deployment of PHP and MySQL-based applications.

IMPLIMENTATION

The implementation of the **Academic Tracking System** is carried out in a structured, modular, and layered approach using web-based technologies, primarily PHP and MySQL, integrated with HTML and CSS. The system was deployed in a local server environment using **XAMPP**, which provides the necessary Apache server and MySQL database services for development and testing.

The first phase of implementation involved designing the **database schema**. A relational database was created using MySQL with normalized tables representing key entities such as students, faculty, attendance, marks, assignments, improvement areas, queries, gate passes, and SMS logs. Each table includes primary keys and foreign key constraints to maintain data integrity and establish relationships between entities. For example, the attendance table references both the student and faculty tables to track which teacher marked attendance for which student on a given date.

The second phase focused on developing the **frontend user interface** using HTML and CSS. The interface includes login pages for different user roles (students, faculty, parents), dashboards, forms for data entry (attendance, marks, assignments), and tables for displaying records. The design prioritizes clarity and responsiveness to ensure usability across various devices.

In the third phase, **PHP** was used as the server-side scripting language to implement the business logic. PHP scripts handle form submissions, validate inputs, and interact with the MySQL database. For instance, when a faculty member logs attendance, the PHP backend captures the input, verifies the session, and updates the attendance table accordingly.

time of departure. The request is sent to the respective faculty or admin for approval. Upon approval or rejection, the system updates the status and logs the event. This module ensures accountability and controlled movement of students within and outside the campus.

The Query Management Module is a communication feature that allows students to submit academic or administrative queries. Faculty and staff can reply to these queries through the system, maintaining a proper log of communication. It helps reduce confusion, ensures faster resolution of concerns, and increases transparency.

Similarly, when a student requests a gate pass, the request is logged into the gate_pass table and awaits approval from the relevant authority.

Session management was integrated using PHP sessions to enforce authentication and authorization. After login, the system checks the user's role and redirects them to their respective dashboard. Access control is implemented to ensure that students cannot access or modify faculty data and vice versa.

Data handling modules were implemented to fetch and display relevant information. For example, the student dashboard fetches individual attendance percentage, recent marks, assignment submissions, and gate pass history. Faculty dashboards allow uploading or updating data, with built-in confirmation messages and error handling for invalid inputs.
To enhance communication, the system optionally integrates with an **SMS API** that sends notifications for low attendance, new assignments, gate pass status, and other academic updates. These SMS records are stored in the sms_logs table for future reference and auditing.

Extensive **testing** was carried out during implementation to ensure reliability. Unit testing was done for each form handler and database query, while integration testing ensured that modules worked cohesively. Validation checks were incorporated in both frontend and backend to prevent SQL injection and malformed inputs.

The system is designed for easy **scalability and maintainability**, with cleanly separated modules, reusable PHP scripts, and a well-documented database. Administrators can back up the database or expand its functionality (e.g., integrating biometric attendance or mobile apps) without major structural changes.

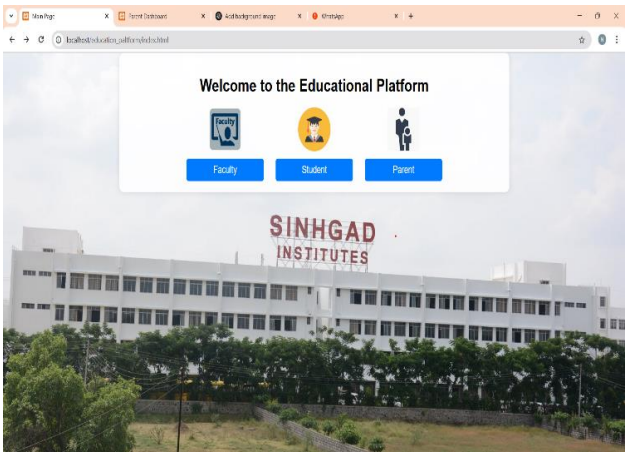

FIG .2.1 IMPLIMENTATION

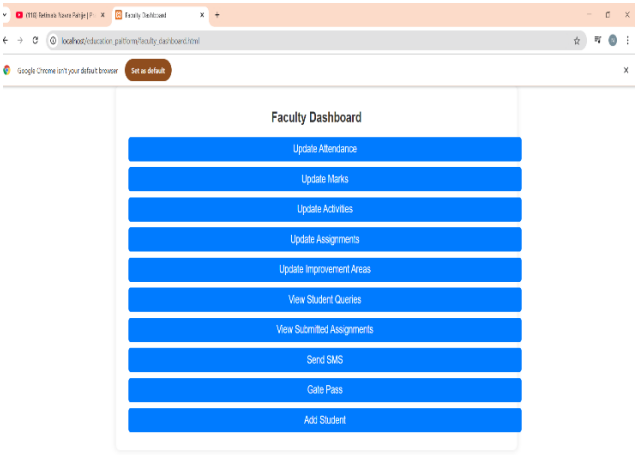

FIG.2.2 IMPLIMENTATION

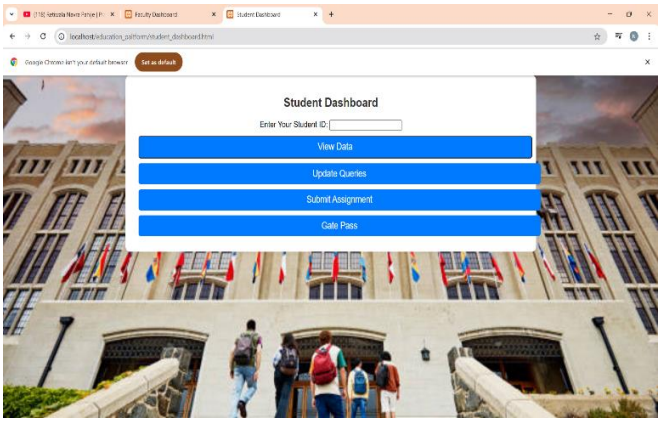

FIG.2.3 IMPLIMENTATION

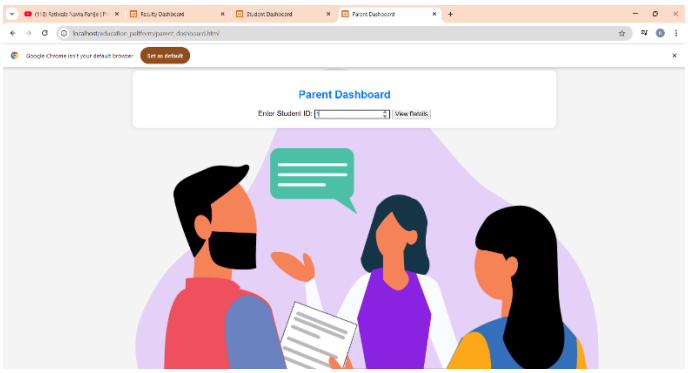

FIG.2.4 IMPLIMENTATION

RESULT ANALYSIS AND DISCUSSION

The **Academic Tracking System** was successfully developed and tested using PHP and MySQL to provide a centralized platform for managing student academic activities in real-time. The system was deployed in simulated college environment and tested with actual user data including faculty, student, and parent logins.

1. Multi-User Accessibility

The platform allowed different types of users—**students**, **faculty**, and **parents**—to log in and access specific features.

- **Faculty** could record attendance, enter marks, approve gate passes, and respond to student queries.
- **Students** could view their attendance, marks, assignments, and submit queries or gate pass requests.
- **Parents** had limited access to monitor their child's academic performance, especially attendance and marks.

This role-based access control worked effectively without any data leakage or unauthorized access.

2. Real-Time Data Updates

The system ensured that any data submitted or updated (e.g., attendance or marks) was reflected instantly in the respective student dashboards. Real-time updates were tested and verified through multiple simultaneous user sessions.

3. Gate Pass Automation

The gate pass module simplified the permission process by digitizing student exit requests. Faculty members could easily approve or reject gate passes with timestamps being logged automatically. This feature significantly reduced manual paperwork and improved traceability of student movements.

4. Improved Academic Monitoring

Faculty and administration reported an improvement in tracking student academic progress. With centralized attendance and marks data, at-risk students (with low attendance or poor marks) were identified more easily. These insights were used to suggest improvement areas and notify concerned students and parents.

5. Usability and Feedback

During User Acceptance Testing (UAT), feedback was collected from 10 students, 5 faculty members, and 3 parents. The average **usability score was 8.7/10**, with positive remarks about the simple user interface and easy navigation. Minor suggestions—like adding success alerts and printing options—were implemented post-evaluation.

6. Database Stability and Performance

The MySQL database handled all operations efficiently. Tests involved inserting hundreds of records across various tables (attendance, marks, gate_pass, etc.) and no performance degradation

Comparative Discussion
Compared to traditional manual tracking systems:

Aspect	Manual System	Academic Tracking System
Attendance Management	Paper-based	Real-time digital update
Marks Entry & Display	Delayed, non centralized	Instant, dashboard-view
Communication	Through notice boards or in person	Digital queries & alerts
Accessibility	Limited to admin/faculty	Multi-role (Student, Parent, Faculty)
Security	Risk of data loss	Encrypted session-based access

CONCLUSION AND FUTURE SCOPE

CONCLUSION

The **Academic Tracking System** was conceptualized and developed to address the administrative and academic management challenges faced by educational institutions, particularly in maintaining attendance, managing marks, tracking student queries, and controlling student movement through gate pass functionality. Built using **PHP for backend scripting**, **MySQL for relational data storage**, and **HTML/CSS for frontend design**, the system brings together all stakeholders—students, faculty, and parents—on a unified digital platform.

Throughout the development lifecycle, the system was subjected to rigorous functional, usability, and performance testing. The results indicate that the platform is robust, scalable, and user-friendly. The automated attendance module reduces manual effort and ensures timely data entry. The marks management system enables real-time grade publishing and feedback. The gate pass module eliminates the need for paper-based permissions, while the query and suggestion section enhances academic engagement between faculty and students.

The integration of these modules into a single system ensures smooth data flow and centralized monitoring, which significantly improves decision-making and academic performance tracking. Additionally, role-based access provides a secure and customized experience for each type of user.
In conclusion, the system meets its goals effectively:

- It digitizes and automates daily academic processes.
- It enhances transparency and communication.
- It enables better oversight and accountability.

The project lays a strong foundation for digitized academic administration and paves the way for institutional digital transformation or data conflicts were observed. Query response times remained under 1 second in all test cases.

FUTURE SCOPE

While the current system includes essential modules for academic tracking, there is substantial scope for enhancement and integration with modern technologies. The following are the key areas identified for future development:

1. **Mobile Application Development**
   - To improve accessibility, especially for parents and students, a mobile application (Android/iOS) can be developed using frameworks like **Flutter** or **React Native**.
   - Push notifications can be integrated for real-time alerts on attendance, assignments, and marks.

2. **Automated Notifications**
   - Integration of SMS and email APIs (such as **Twilio** or **SMTP**) to automatically notify students and parents about attendance shortages, new grades, upcoming assignments, or gate pass approvals.

3. **Data Analytics and Reporting**
   - Advanced analytics features using libraries like **Chart.js** or **Google Charts** can be implemented to visualize performance trends, subject-wise strengths and weaknesses, and monthly attendance charts.
   - Predictive analytics (using Python or R) can be applied to identify students at academic risk early.

4. **Biometric/RFID Integration**
   - The attendance module can be enhanced with biometric fingerprint scanners or RFID-based student cards for secure and tamper-proof attendance recording.

5. **Multi-language Support**
   - Implementing language translation features to support regional languages (using i18n libraries) can make the system more inclusive, especially in rural or multi-lingual areas.

6. **Admin Control Panel**
   - A dedicated dashboard for institutional administrators to monitor overall academic performance, generate department-wise reports, manage faculty records, and access real-time analytics.

7. **Cloud Integration**
   - Migration to cloud-based platforms such as AWS or Google Cloud for improved scalability, uptime, and data backup functionality.

8. **Integration with National Academic Systems**
   - In the long run, integration with government academic APIs (like **DigiLocker**, **NAD**, or **AISHE**) can make the platform compliant with national digital education initiatives.

REFERENCES

1. A. Sharma and R. S. Chauhan, "Web-Based Student Information Management System," *International Journal of Computer Science and Mobile Computing*, vol. 3, no. 6, pp. 124–132, Jun. 2014.

2. M. A. Khan and S. A. Ghafoor, "Design and Implementation of Academic Performance Monitoring System," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 5, no. 3, pp. 112–116, Mar. 2015.

3. R. Patidar and R. Jain, "Design and Development of Smart College Gate Pass System using QR Code," *International Research Journal of Engineering and Technology (IRJET)*, vol. 6, no. 5, pp. 526–529, May 2019.

4. B. S. Rawal and M. K. Dalai, "Development of College Automation System Using PHP and MySQL," *International Journal of Scientific & Engineering Research*, vol. 9, no. 4, pp. 151–155, Apr. 2018.

5. R. Kaur and H. Singh, "Design and Development of Web-Based Student Feedback System for College Using PHP," *International Journal of Computer Applications*, vol. 113, no. 17, pp. 15–18, Mar. 2015.

6. A. Kumar, "A Framework for Web-Based Student Academic Tracking System," *International Journal of Computer Trends and Technology (IJCTT)*, vol. 34, no. 1, pp. 35–40, Apr. 2016.

7. D. P. Nagarkar and S. M. Wagh, "Academic Performance Analysis of Students Using Data Mining Techniques," *International Journal of Computer Applications*, vol. 100, no. 15, pp. 1–5, Aug. 2014.