

DIGITAL GATE PASS SYSTEM

Mogili Nikhil Prasad

Students, Department of Emerging Technologies (CSE-AI&ML),

Mahatma Gandhi Institute of Technology, Hyderabad, Telangana-500075

mnikhilprasad_csm235A6605@mgit.ac.in

Guides: Ms. Karuna Verma, Assistant Professor

Department of Emerging Technologies,

Mahatma Gandhi Institute of Technology, Hyderabad, Telangana – 500075

ABSTRACT

The current manual gate pass system in colleges is inefficient, time-consuming, and vulnerable to security lapses. Students are required to approach the Head of Department (HOD) in person to request a gate pass, which often leads to delays and disrupts academic schedules. The approval process lacks flexibility, especially in urgent situations, and causes unnecessary congestion and confusion. Once a gate pass is issued, the security guard manually verifies the student's identity by matching their physical appearance with a photo ID. This method is prone to human error, increasing the risk of unauthorized access and misidentification. The absence of a centralized tracking system further complicates the process, making it difficult to audit gate pass usage or ensure accountability.

To resolve these challenges, this project proposes a digital gate pass system that

streamlines the request, approval, and verification process. Students can apply for gate passes online, and faculty can review and approve them digitally. The system will use digital photo verification and integrate modern authentication methods such as OTPs or QR codes to confirm identity and access rights securely.

The proposed solution aims to improve efficiency, reduce human error, and strengthen campus security. By automating the entire gate pass process and providing real-time access to records, the system will benefit students, faculty, and security staff alike. This shift toward a tech-enabled solution marks a significant step in modernizing campus infrastructure.

INTRODUCTION

In many educational institutions, especially colleges, the gate pass system plays a crucial role in regulating student movement and ensuring campus security. Traditionally, this

system has been managed manually, requiring students to physically approach the Head of Department (HOD) to request permission to leave the campus. This process often leads to unnecessary delays, long queues, and a loss of valuable academic time.

Once a gate pass is issued, identity verification is done manually by security personnel, who compare the student's physical appearance with their photo ID. This step is highly dependent on human judgment and can result in errors, including allowing unauthorized exits or denying valid ones. Additionally, there is no centralized system to track gate pass requests, approvals, or student movements, which further weakens the overall security and efficiency of the process. With the growing need for digital transformation in educational institutions, automating the gate pass system has become both feasible and necessary. A digital solution not only streamlines the approval process but also enhances security through features like digital photo verification and modern authentication methods.

This project aims to develop a secure, efficient, and user-friendly digital gate pass system that addresses the limitations of the manual method. By shifting to an automated platform,

colleges can save time, reduce administrative burdens, and significantly improve campus safety and record-keeping.

Motivation

In many educational institutions, the process of issuing gate passes is still handled manually, which involves writing student details in registers, obtaining signatures from the Head of Department (HOD), and verifying the pass at the security gate. This traditional approach is not only time-consuming but also inefficient, especially when there are many students requesting permission at the same time. Manual systems are prone to errors such as incorrect data entry, loss of records, and misuse of gate passes. Students often face inconvenience due to delays in approval when the concerned authority is unavailable. Moreover, maintaining and tracking records in physical registers becomes difficult over time. The motivation behind developing the Smart Gate Pass System is to create a digital solution that simplifies and automates the entire process. By implementing a web-based system, approvals can be processed quickly, records can be stored securely, and verification can be done instantly. This

not only reduces manual effort but also improves transparency, efficiency, and security within the institution.

Problem Definition

The existing gate pass system in many institutions suffers from several major issues that affect both efficiency and security. One of the primary problems is the delay in approval, as students are required to physically approach the HOD, which can be inconvenient and time-consuming. In many cases, students have to wait for long periods just to get permission, especially during busy hours. Another significant issue is the lack of proper verification at the security level, where security personnel rely on paper-based passes that can be easily lost, damaged, or even duplicated. This increases the risk of unauthorized exits and reduces the overall safety of the campus. Additionally, record management is a major concern, as maintaining registers manually requires extra effort and makes it difficult to retrieve past data when needed. There is no centralized system to store and monitor student movement effectively. Therefore, there is a need for a reliable and automated system that can handle approvals, ensure secure verification, and maintain accurate records in a structured digital format.

Existing System

The existing system for managing gate passes is primarily manual and depends

heavily on paperwork. In this system, students are required to fill out a paper form with their details and submit it to the Head of Department (HOD) for approval. Once approved, the student carries the physical gate pass to the security gate, where it is manually verified by security personnel. This process has several limitations, including delays in approval due to the availability of the HOD and the possibility of errors in manual data entry. The verification process at the security gate is also not reliable, as it depends on visual inspection of paper passes, which can be easily forged or altered. Furthermore, maintaining records in registers is inefficient and time-consuming, making it difficult to search for specific data or generate reports. There is also no proper mechanism to track student entry and exit in real time. Overall, the existing system lacks efficiency, accuracy, and security, which highlights the need for improvement.

Proposed System

The proposed Smart Gate Pass System is a web-based application designed to automate and improve the entire gate pass process. It consists of two main modules: the HOD panel and the Security panel, each with specific functionalities. In the HOD panel, the HOD can enter the student's roll

number and view complete details such as name, father's name, branch, attendance percentage, and photograph. Based on this information, the HOD can either check the details or directly approve the request. In the Security panel, the security personnel can enter the student's roll number to verify whether the student has been approved. If approved, the system displays the student's details and photograph for confirmation, ensuring that the correct student is leaving the campus. After verification, the approval status is reset to prevent reuse or misuse of the pass. This system reduces manual effort, speeds up the process, ensures accurate record-keeping, and enhances security by allowing only authorized students to exit the premises. It also provides a structured and reliable way to store and manage data digitally.

1.5 Requirements Specification

Software Requirements

- **Operating System:** Windows 10/11, Linux (Ubuntu), or macOS
- **Programming Language:** Python 3.x
- **Framework:** Flask (for backend and routing)

- **Frontend Technologies:**
 - HTML5 (for structure)
 - CSS3 (for styling)
 - JavaScript (for client-side validation and interactivity)
- **Database:**
 - SQL (MySQL / PostgreSQL / SQLite — based on project choice)
 - SQL Alchemy (for ORM integration with Flask)
- **Libraries and Packages:**
 - Flask-WTF (for form handling & validation)
 - Flask-Login (for authentication of admin/HOD/security)
 - Pillow (for student photo display and handling)
 - OpenCV (optional, if adding face recognition later)
 - Jinja2 (for templating in Flask)
- **Server/Environment Tools:**
- **WSGI server** (Flask built-in / Gunicorn for deployment)
- **Code Editor:** Visual Studio Code

Hardware Requirements

- **Processor:** Intel Core i3/i5 or AMD equivalent (i5 recommended)
- **RAM:** Minimum 4 GB (8 GB recommended for smooth development)

- Storage: 10–20 GB free space for project files, database, and setup
- GPU (Optional): Needed only if advanced face recognition is added
- Input Device: Standard webcam or laptop camera (for image verification option)
- Output Device: Monitor/Display for frontend UI

LITERATURE SURVEY

[1] Digital Gate Pass System Using QR Code Authentication [2024]

This study focused on designing a secure gate pass system where QR codes are generated for each approved pass. Students request passes through a web or mobile interface, and upon approval, a unique QR code is issued. At the gate, security personnel scan the code to instantly verify identity and permissions. This significantly reduced wait times and eliminated errors related to manual entry and ID card verification. The research concluded that QR-based systems offer a cost-effective and efficient way to automate campus access while maintaining a secure verification process.

[2] Facial Recognition-Based Smart Attendance and Access [2023]

This paper introduced an automated access control system using facial recognition, integrated with attendance tracking.

Cameras at entry points detect and match student faces with stored database records. If a match is successful and an approved gate pass exists, access is granted. The system improved accuracy in identity verification and eliminated misuse of ID cards. Challenges noted include sensitivity to lighting and camera angle, but overall, it provided a hands-free, secure solution suitable for modern campuses.

[3] IoT-Based Campus Entry Management System [2023]

This research proposed an IoT-enabled gate monitoring solution using sensors and cloud connectivity. It allowed real-time data collection and remote gate access control. The findings support the application of IoT in Digital Gate Pass systems to provide centralized monitoring and enhanced scalability.

[4] Mobile App for Digital Gate Pass with Admin Approval[2022]

A mobile-based system where students request gate passes through an app, and administrators approve requests digitally. The study emphasized user-friendly design and secure logins. It highlighted improved response time and better coordination between students, staff, and security

[5] RFID-Based Access Control for Educational Institutes[2022]

This paper presented an RFID-based model for managing entry and exit within campuses. RFID tags replaced manual IDs, reducing gate congestion and enabling efficient logging. Although effective, the system lacked flexibility for temporary or one-time passes, which QR and OTP systems can better manage.

[6] Real-Time Campus Access Management with Cloud [2021]

This study designed a cloud-based access control system to manage gate activity from any location. Administrators could approve or revoke passes in real time, and data logs were instantly stored in the cloud. This improves record-keeping and facilitates data retrieval during audits.

[7] Biometric Gate Pass Verification System for University Campuses [2020]

The use of fingerprint and iris recognition in gate pass verification proved to be highly secure but required specialized hardware. The research pointed out that while biometrics enhance accuracy, they are less cost-effective and scalable compared to QR or face- recognition methods.

[8] AI-Powered Campus Security System with Gate Pass Automation [2019]

This paper introduced AI for anomaly detection in gate pass logs and access patterns. The system flagged suspicious behavior and unauthorized

access attempts. It highlighted how artificial intelligence can complement Digital Gate Pass

METHODOLOGY

The Smart Gate Pass System is developed using a web-based approach with Python and the Flask framework. The system collects the student's roll number and retrieves details from the database for verification. The HOD checks and approves the request, and the approval status is stored digitally. The security panel then verifies the student details and resets the approval after validation to ensure secure and controlled exit.

Requirement Analysis

The requirement analysis for the digital gate pass system involves identifying the needs of students, faculty, and security personnel to create a seamless and secure workflow. The system must allow students to submit gate pass requests digitally from any device, reducing the need for physical interaction with the HOD. Faculty should have a dedicated approval dashboard where they can view pending requests, approve or reject them, and monitor student movement efficiently. Security staff require a simple verification interface that supports checking student details using digital photo matching, QR codes, or OTP verification. The system must support secure login for all users through authentication methods such as passwords or institutional credentials. Real-

time notifications should be integrated so students receive instant updates on approval status. The platform must maintain a centralized database that stores request logs, user details, and verification history for auditing purposes. Non-functional requirements include high availability, quick response time, and smooth performance even under heavy usage. The interface should be intuitive and compatible with both mobile phones and computers. Data privacy and protection must be ensured through encryption and role-based access control. The system should also offer scalability to support future expansion across departments. Overall, these requirements lay the foundation for a reliable and fully automated gate pass.

Database

The database design focuses on efficient storage, retrieval, and management of student data and approval information. The system uses SQLite as the database, which is lightweight and suitable for small to medium-scale applications. Each student is uniquely identified by a roll number, which serves as the primary key for accessing records. The database includes fields such as roll number, student name, father's name, branch, attendance percentage, photograph path, and approval status. The approval status indicates whether the student is allowed to exit or not. This structured design

ensures data consistency, avoids redundancy, and enables quick retrieval of information during both approval and verification stages. It also supports easy updates and maintenance of records, making the system efficient and reliable.

Frontend Development

Frontend development involves designing and implementing the user interface of the Smart Gate Pass System using HTML, CSS, and JavaScript. The system includes separate interfaces for the HOD panel and the Security panel to ensure clarity and proper functionality. The interface provides input fields for entering the roll number, buttons for actions such as checking details and approving requests, and display sections for showing student information and photographs. The design is kept simple, clean, and user-friendly so that users can easily navigate and perform tasks without confusion. Attention is given to layout, alignment, and responsiveness to ensure a smooth user experience. A well-designed frontend improves usability and allows efficient interaction between the user and the system.

Backend Development

Backend development is responsible for implementing the core logic and functionality of the Smart Gate Pass System using Python and the Flask framework. It acts as a bridge between the frontend interface and the database. The backend

handles tasks such as receiving user input, retrieving student details from the database, updating approval status, and verifying records. When the HOD enters a roll number, the backend fetches the corresponding data and displays it on the screen. Upon approval, the system updates the status in the database. Similarly, when the security personnel verifies a student, the backend checks the approval status and displays the required information. After successful verification, the approval status is reset to prevent misuse.

Evaluation and Optimization

The evaluation and optimization phase ensures that the Smart Gate Pass System functions correctly, efficiently, and without errors. In this phase, the system is tested by performing various operations such as entering roll numbers, checking student details, approving requests, and verifying them in the security panel. Each function is carefully evaluated to ensure accurate data retrieval, proper approval updates, and correct display of student information including photographs. Any issues such as incorrect outputs, delays in response, or system errors are identified and fixed.

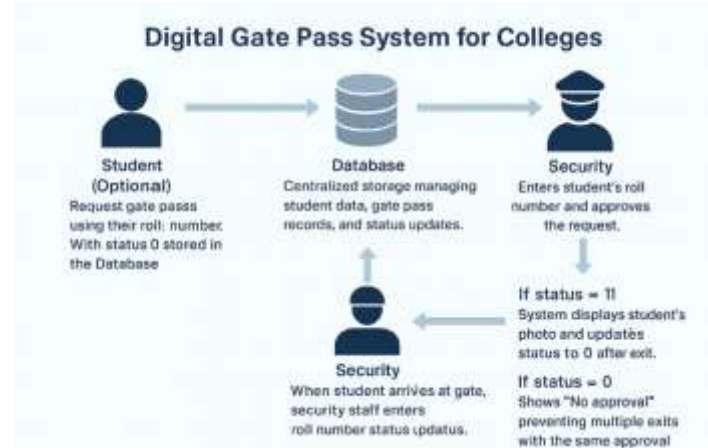
SYSTEM DESIGN

The Smart Gate Pass System is designed as a web-based application consisting of multiple interconnected panels that operate

through a centralized database to manage gate pass activities in an efficient and secure manner. The system follows a structured workflow that ensures smooth communication between different modules, including the Student Panel, HOD Panel, and Security Panel. In the Student Panel, when implemented, students can request a gate pass by entering their roll number, and the request is stored in the database with an initial status value of 0, indicating that it is pending approval. This eliminates the need for manual paperwork and allows requests to be processed digitally. The HOD Panel enables the Head of Department to access and review these requests by entering the student's roll number. Upon verification of details such as name, father's name, branch, attendance percentage, and photograph, the HOD can approve the request, which automatically updates the database status to 1, indicating that the request has been successfully approved. This digital approval mechanism reduces delays and ensures proper authorization. Once the student reaches the exit point, the Security Panel is used by the security personnel to verify the gate pass by entering the roll number. If the status is found to be 1, the system retrieves and displays the student's photograph and details for identity confirmation, ensuring that the correct student is leaving the campus. After successful verification, the system updates the status back to 0, marking

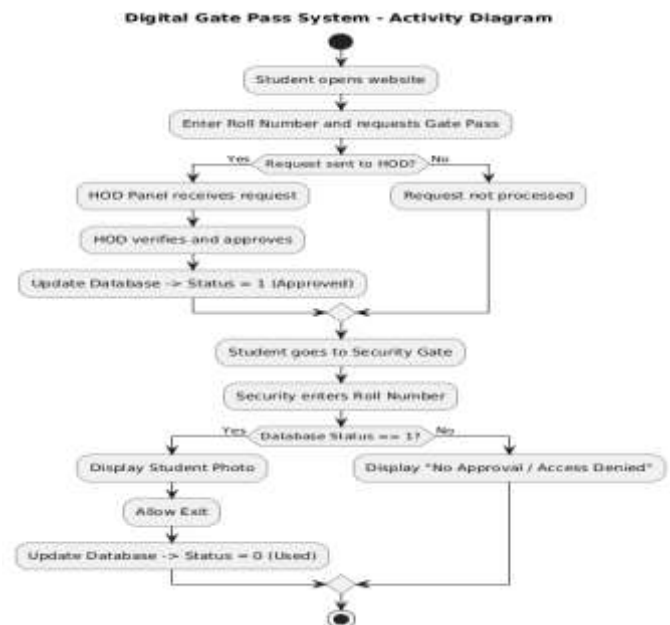
the gate pass as used and preventing any possibility of reuse or duplicate exits. If the status remains 0, the system displays a “No Approval” message, thereby restricting unauthorized movement and maintaining campus security. The centralized database plays a crucial role as the backbone of the system, storing all essential information including student details, gate pass requests, approval statuses, and verification logs. It continuously updates data in real time, ensuring accuracy, consistency, and quick retrieval of information whenever required. The system architecture is designed to handle multiple requests efficiently while maintaining data integrity and security. It also ensures transparency by keeping a proper record of all activities, which can be useful for monitoring and auditing purposes. Overall, the Smart Gate Pass System provides a seamless and automated flow from request generation to approval and final verification, reducing manual effort, minimizing errors, and enhancing operational efficiency.

System Architecture Diagram



[Figure 1] System Design of Digital Gate Pass

Activity Diagram



[Figure 2] Activity Diagram of Digital Gate Pass

The activity diagram shows how a student requests a gate pass online, which is then verified and approved by the HOD before updating the database. When the student reaches the gate, the security staff checks the roll number to confirm approval; if valid, the system displays the student's photo and allows exit, otherwise access is denied. Finally, the database updates the pass status to prevent reuse.

TESTING AND RESULT

TESTING

1. Input Testing

Input testing is performed to ensure that the system correctly accepts and processes user inputs such as student roll numbers. Various valid and invalid inputs are tested to check system behavior. When a valid roll number is entered, the system should fetch and display correct student details. If an invalid or non-existing roll number is entered, the system should display an appropriate error message. This testing helps in validating input fields and preventing incorrect data processing. It ensures that the system handles all types of user inputs effectively.

2. Functional Testing

Functional testing is carried out to verify that all features of the Smart Gate Pass System are working as expected. This includes checking operations such as retrieving student details, approving requests in the HOD panel, and verifying approvals in the Security panel. Each function is tested individually and as a complete workflow. The interaction between different modules is also verified to ensure smooth execution.

This testing ensures that the system performs all required tasks correctly without any functional errors.

3. Output Testing

Output testing ensures that the system displays correct and meaningful results based on user inputs. When a roll number is entered, the system should show accurate student details including name, branch, attendance, and photograph. It also checks whether approval messages and “No Approval” messages are displayed correctly. The formatting and clarity of the output are verified to ensure user understanding. This testing helps in confirming that the system provides correct and user-friendly outputs.

4. Database Testing

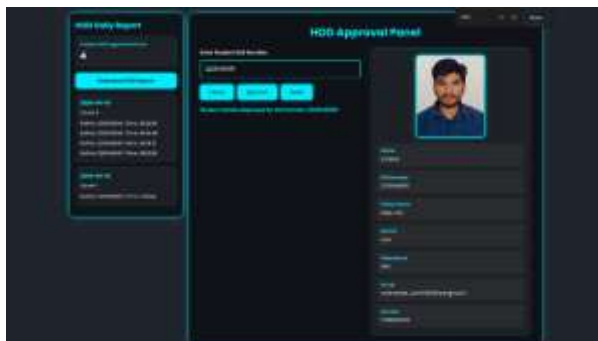
Database testing is performed to verify that data is stored, updated, and retrieved correctly from the database. It checks whether student details are accurately saved and whether the approval status changes properly from 0 (pending) to 1 (approved) and back to 0 (used). The system is also tested for data consistency and integrity. This ensures that there is no data loss or duplication. Database testing plays a key role in maintaining reliable

system performance.

5. User Interface Testing

User interface testing ensures that the system is easy to use and visually clear for users. It checks whether all buttons, input fields, and pages are properly aligned and functional. The navigation between HOD and Security panels is tested for smooth operation. The system is evaluated for simplicity so that users can perform actions without confusion. This testing helps in improving the overall user experience and usability of the system.

RESULTS



[Figure 3] HOD panel checking student Details



[Figure 4] HOD approving the student



[Figure 5] Security Verification

CONCLUSION AND FUTURE WORK

CONCLUSION

The Digital Gate Pass System effectively automates and secures the traditional gate pass process in educational institutions. By integrating features such as digital request submission, HOD approval, identity verification through image processing, and real-time status checks, the system enhances security, transparency, and operational efficiency. The streamlined workflow reduces delays, minimizes manual errors, and ensures that only authorized students can exit the campus. Overall, the project demonstrates a practical application of machine learning and web technologies to

solve real-world administrative challenges.

FUTURE WORK

Moreover, the system improves user experience for both students and staff by offering a simple and intuitive interface through platforms like Streamlit. The role-based access panels (Student, HOD, and Security) help define clear responsibilities, reducing communication gaps and improving accountability. This digital approach also supports better record-keeping and makes data easily retrievable for audits or incident tracking.

From a technical standpoint, the use of image pre-processing, feature extraction, and classification models ensures that student identity verification is reliable and fast. The integration of front-end and back-end components through Flask or similar frameworks provides a seamless pipeline from image capture to decision-making.

Regular testing and debugging have ensured the system runs smoothly across different modules, improving the overall robustness.

REFERENCES

□ **Flask Documentation – *Flask Web Framework***

<https://flask.palletsprojects.com/>

□ **Python Official Documentation – *Python Programming Language***

<https://docs.python.org/3/>

□ **SQLite Documentation – *SQLite DatabaseSystem***

<https://www.sqlite.org/docs.html>

□ **W3Schools – *HTML, CSS, and JavaScript Tutorials***

<https://www.w3schools.com/>

□ **GeeksforGeeks – *Web Development and Python Tutorials***

<https://www.geeksforgeeks.org/>