



Classroom Management With Facial Recognition Attendance System

A Contactless Solution for Modern Academia

¹Vaibhav Laxman Bondge, ²Ashitosh Santosh Ghodke, ³Ravindra H. Adekar

^{1,2}U. G. CSE Student, ³Assistant Professor

^{1,2,3}Computer Science and Engineering,

^{1,2,3}Shri Tulja Bhavani College of Engineering, Tuljapur, India

Abstract: Anyone who has been in a classroom knows that taking attendance can be a disruptive chore. Traditional roll calls are slow, and even modern solutions like fingerprint scanners can cause bottlenecks and raise hygiene concerns. We've developed a fresh approach: a smart, web-based system that uses facial recognition to take attendance automatically and without contact. We've built an entire digital classroom around this idea, making it simple for teachers to manage their courses and for students to stay on track. The platform handles everything from enrolling students and creating new classes to providing easy-to-read dashboards for tracking performance. This paper will walk you through how it was built using technologies like Node.js, Python, OpenCV, and MongoDB, and show you how it works in the real world. With its user-friendly design and ability to spot multiple faces at once, our system proves that technology can make our schools more efficient and secure.

Index Terms - Facial Recognition, Attendance System, Digital Classroom, Web-based System, Computer Vision, Education Technology, Contactless Attendance.

I. INTRODUCTION

Anyone who's been in a lecture hall knows the drill: attendance is a chore. Manually checking off names takes forever and is prone to errors, not to mention students signing in for their absent friends. While physical scanners are a step up, they're not perfect either. They create queues at the classroom door and, in a world where we're all more conscious of germs, having everyone touch the same device isn't ideal. Simply put, the traditional ways of tracking attendance are broken.

Facial recognition technology offers a genuinely smart fix. It can tell who's in the room without anyone needing to touch a thing—a perfect solution for today's world. But just taking attendance isn't enough. Schools need a single, connected system to manage everything else, from class assignments to student data. Too often, attendance software is completely separate from the main learning platform, forcing teachers and staff to juggle multiple systems. It's an inefficient, disconnected experience.

To solve this, we built a web-based facial recognition attendance system that's also a complete digital classroom hub. We wanted to create something that was not only automatic and hyper-accurate but also completely contactless, leaving the old, risky methods behind. More importantly, we've wrapped it in a powerful classroom management framework that lets educators handle courses, manage student profiles, and communicate effortlessly. The idea is to bring everything under one roof, making school administration simpler and providing a clearer view of what's happening through data.

Here's what our system brings to the table:

- **Smart, real-time attendance:** It spots and identifies every student in the room simultaneously, so there's no disruption to the lecture.
- **An all-in-one web portal:** A simple, intuitive place to manage user profiles, create classes, and handle enrollment.
- **Actionable analytics:** Clear dashboards that break down attendance data, helping teachers spot trends and giving students ownership over their records.
- **Truly contactless check-ins:** A safe, hygienic way to record attendance that eliminates queues and physical touchpoints.
- **Easy reporting:** Attendance records can be downloaded in standard formats with just a click.

By taking this integrated approach, our system doesn't just fix a single problem. It fosters a more connected and efficient learning environment, empowering schools to run more smoothly and with greater insight.

II. RELATED WORK

The way we take attendance has been stuck in the past for too long. To understand the need for a new approach, we need to look at the tools currently available and pinpoint where they fall short.

It started with the classic roll call—a method that's not just slow but also fundamentally flawed. It's a system built on trust that's easily broken by proxy attendance, burdened by human error, and a drain on teachers' time. Early tech solutions like barcode and RFID scanners were a marginal improvement, but they still relied on a physical ID card that could be lost, forgotten, or passed around, making the attendance data just as shaky.

Biometric scanners, using fingerprints or palm prints, represented a major step up in security, virtually eliminating the issue of "buddy punching." Their fatal flaw, however, is the very thing that makes them work: physical touch. In any large institution, this creates hygiene risks and frustrating bottlenecks as students line up to scan in. The cost and complexity of deploying hundreds of these devices across a campus also make them an impractical choice for many.

While this was happening, facial recognition technology quietly became incredibly powerful. Fueled by deep learning and libraries like OpenCV, these systems can now identify faces with high accuracy. People have tried applying this to attendance, and the results are often good, but the solutions themselves are incomplete. They tend to be one-trick ponies—standalone apps that do nothing but scan faces, failing to connect with the other essential tools that make a school run.

Think about the digital learning platforms we now rely on, like Moodle or Canvas. They are excellent at managing coursework, grades, and communication, forming the core of the modern educational experience. And yet, almost none of them have a smart, automated attendance feature. Teachers using these incredible platforms are often forced to revert to a manual checklist, a jarring disconnect that creates unnecessary work.

The core problem is this: the market is filled with pieces of a solution, but no one has put them together. You can get contactless biometrics or you can get a great digital classroom, but you can't get both in one seamless package. Our system is designed to bridge that exact gap. We have built a single, comprehensive platform that integrates state-of-the-art facial recognition directly into a powerful classroom management suite, offering a solution that is finally as smart and efficient as modern education demands.

III. SYSTEM ARCHITECTURE AND METHODOLOGY

The development of an effective facial recognition attendance system that integrates with a digital classroom requires a well-designed and modular architecture. This section outlines the overall system design, detailing the interaction between its components, the main functions of each module, and the core technologies used.

A. Overall System Architecture

The system is designed as a multi-tiered web application to ensure it is scalable, easy to maintain, and has a clear separation of tasks. It operates on a client-server model where users interact with a web-based interface. This interface communicates with a backend server that handles the system's logic, database operations, and the facial recognition engine.

The system consists of three main layers:

1. **Presentation Layer:** The client-side web interface.
2. **Application Layer:** A Node.js/Express.js backend server and a Python/Flask server for facial recognition.
3. **Data Layer:** A MongoDB database.

User requests from the web interface are sent to the Node.js server, which coordinates with the MongoDB database and calls on the Python-based facial recognition service when needed. An integrated email service also handles sending out notifications.

B. Core Modules

To provide a complete solution, the system is divided into several interconnected modules, each managing specific functions for effective attendance and classroom management.

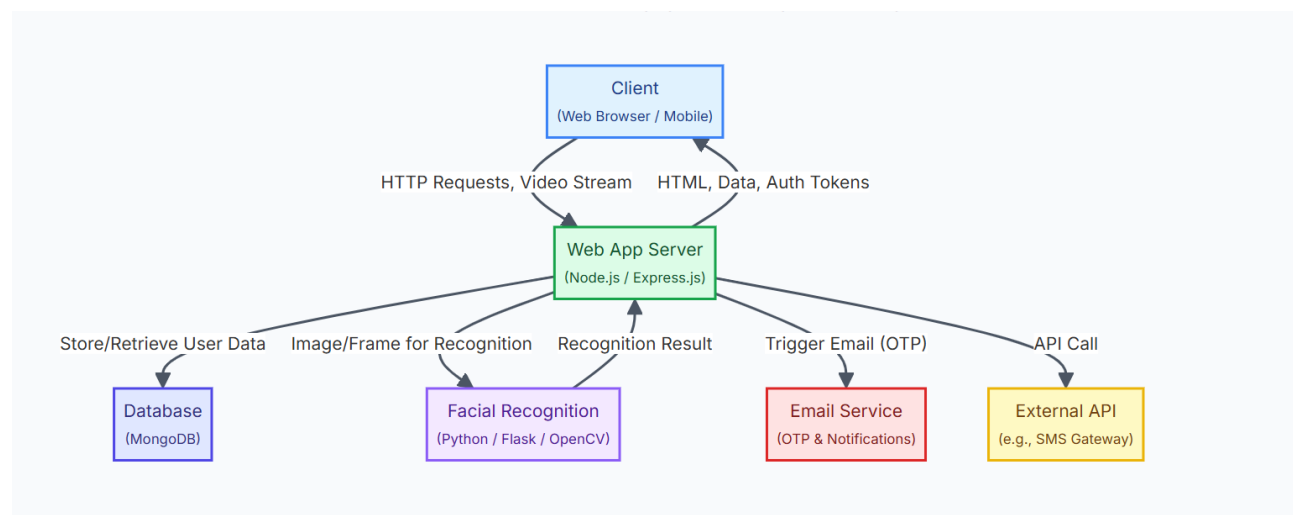


Fig. 1: Overall system architecture of the facial recognition attendance platform

1. User Management Module

This module is the system's foundation, handling the registration and management of student and teacher profiles. The registration process is secure and streamlined, requiring email verification with a 6-digit One-Time Password (OTP) to ensure users are part of the institution. This enhances security by requiring users to register with a college-domain email. After verification, users complete a detailed registration form. The system includes validation rules, such as preventing duplicate roll numbers for students and enforcing strong password policies.

2. Classroom Management Module

This module helps teachers efficiently create and manage their virtual classrooms. Teachers can set up new classrooms by providing course details through a dedicated interface. Each classroom automatically generates a unique join code, which students can use to enroll. Teachers can also add students manually, select them from a filtered list (e.g., by department or year), or add all eligible students at once. The module also notifies students when they are enrolled in a new classroom, encouraging immediate engagement.

3. Facial Recognition Module

This is the technical heart of the system, responsible for accurate face enrollment and real-time attendance marking.

- **Face Enrollment:** Students must upload a clear photograph of their face. A key validation step ensures the image is a JPG file and follows a specific naming convention (e.g., rollnumber.jpg) to easily link the biometric data to student records. This image also becomes the student's default profile picture.
- **Attendance Recording:** To take attendance, a teacher starts a lecture session and uses the "Take Attendance" feature. This activates a camera (either a webcam or an Android device streaming via IP Webcam). The Python-based backend uses OpenCV for real-time face detection and recognition. As students are recognized, their names appear on the screen with a sound to confirm their attendance has been marked. The system is designed to handle multiple faces in a single frame, making group attendance efficient. It can also record a "mass bunk" if no students are detected.

4. Attendance Tracking and Analytics Module

The system's strength lies in its comprehensive attendance tracking and analytics, which give both teachers and students detailed insights. Teachers have access to a dashboard showing overall attendance trends, class-wise attendance percentages, and lists of students with attendance below a set threshold (e.g., 75%). From this list, teachers can send automated warning emails. Students can view their attendance records for all their classes, allowing them to proactively monitor their own progress.

5. Reporting and Export Module

For administrative and record-keeping purposes, the system offers robust data export options. Teachers can download complete attendance records for their classes in an Excel (.xlsx) format. The system also generates

absentee reports for any given date, which helps teachers quickly identify which students were absent from which classes. These reports are also exportable to Excel for easy sharing and archiving.

6. System Updates and Accessibility Module

The system allows user profiles and class details to be updated directly through the web interface, keeping information current. To make attendance capture more flexible, the system supports connecting external Android device cameras. By using an IP Webcam app, a mobile device's camera can stream to the attendance module, offering an alternative to standard webcams.

C. Technologies Used

The system is built on a modern technology stack chosen for its reliability, scalability, and ease of use. The main components are:

- **Node.js and Express.js:** These form the core of the web application server, handling routing, APIs, and server-side logic.
- **Python and OpenCV:** Python is used for the facial recognition engine, with the OpenCV library handling image processing, face detection, and recognition. A Flask server exposes these functions as an API for the Node.js backend.
- **MongoDB:** This NoSQL database was chosen for its flexibility in storing diverse data, including user profiles, classroom information, and attendance records.

This comprehensive architecture ensures the facial recognition attendance system is not only efficient and accurate but also adaptable to the needs of educational institutions.

IV. RESULTS AND DISCUSSIONS

The web-based facial recognition attendance system offers a complete solution for modern educational settings. This section shows the system's key features with screenshots and discusses their effectiveness in solving the challenges mentioned earlier. The system is designed to be user-friendly, accurate, and have strong data management, making it a valuable tool for both educators and students.

A. User Registration and Profile Management

The sign-up process for students and teachers is designed to be secure and verified. The system requires email verification with a 6-digit OTP sent to institutional email addresses, as seen in **Fig. 2**, which adds a layer of security. After verification, users fill out a registration form (**Fig. 3**) with checks for unique student roll numbers and strong password policies. When users log in for the first time, they see dashboards tailored to their roles.

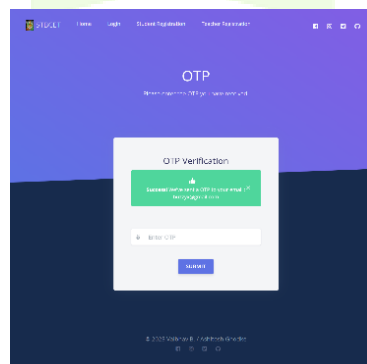


Fig. 2: Successful email verification via OTP, a crucial step in secure user registration.

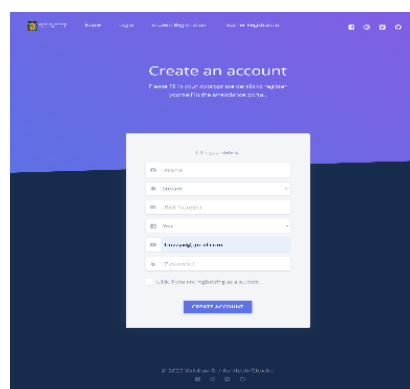


Fig. 3: The student registration form, designed for comprehensive data capture and validation.

B. Classroom Creation and Enrollment

The system provides intuitive tools for teachers to create and manage their courses. Teachers can easily set up new classrooms by providing essential details (**Fig. 4**). Each classroom gets a unique code that students can use to enroll (**Fig. 5**), automatically adding them to the class roster. Teachers also have flexible options for managing rosters, such as adding students manually or in groups..

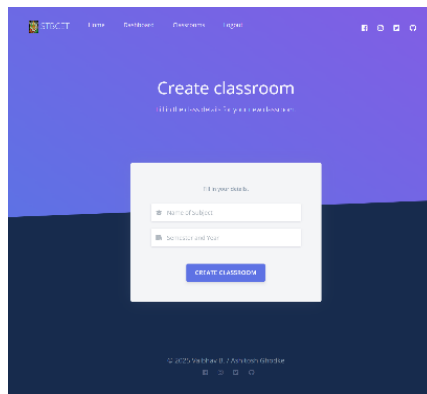


Fig. 4: Interface for teachers to create and configure a new classroom.

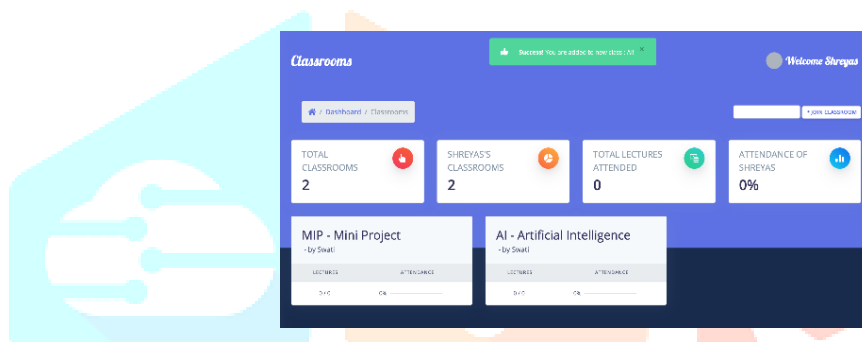


Fig. 5: Student interface for joining a classroom using a unique access code.

C. Face Enrollment and Attendance Recording

The system's core feature is its contactless attendance through facial recognition. Students upload a clear photo for enrollment, which is used to train the recognition model and also serves as their profile picture (**Fig. 6**). To take attendance, teachers start a "lecture" and use the "Take Attendance" feature (**Fig. 7**). This opens a live camera feed where the system scans for registered faces. When a student is identified, their name appears on the screen (**Fig. 8**), and a sound confirms their attendance. The system can recognize multiple faces at once, speeding up the process.

D. Attendance Statistics and Reporting

The system offers powerful analytical tools that turn attendance data into useful insights. Teachers can view a comprehensive dashboard (**Fig. 9**) with overall statistics, class percentages, and lists of students with low attendance (**Fig. 10**). From this list, teachers can send automated warning emails. Students can also see their own attendance records (**Fig. 11**), which helps them stay on track.

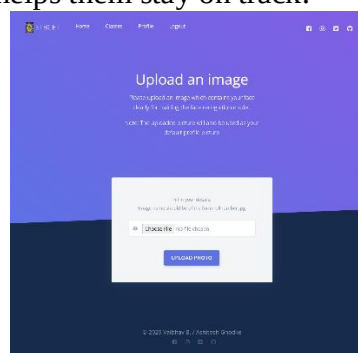


Fig. 6: Student interface for uploading a photograph for facial enrollment and profile picture.

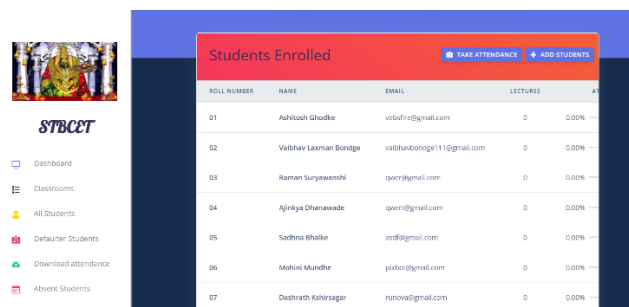


Fig. 7: The "Take Attendance" button, initiating the facial recognition capture session.



Fig. 8: Real-time attendance recording displaying detected and recognized student faces.

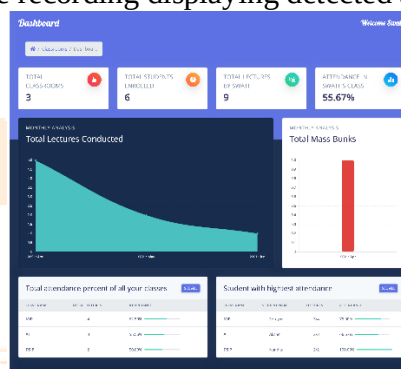


Fig. 9: Teacher's dashboard providing a statistical overview of attendance across classes

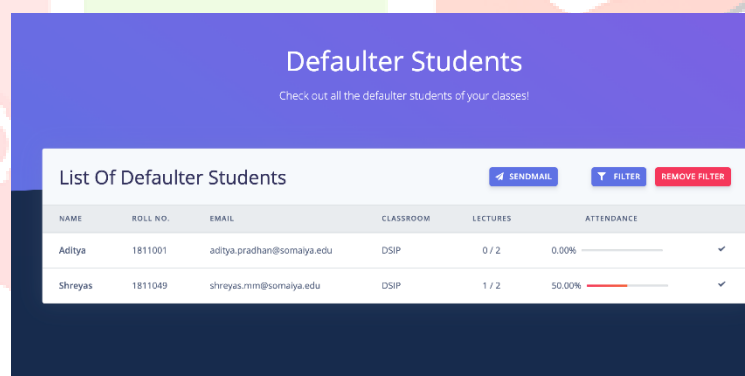


Fig. 10: Filterable defaulter list allowing teachers to identify and communicate with at-risk students.

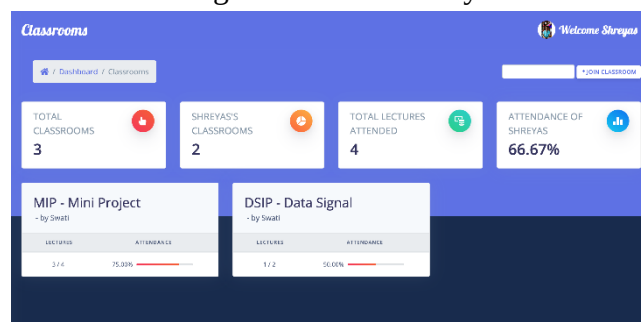


Fig. 11: Student's portal displaying their individual attendance records for enrolled classes.

E. Data Export and Accessibility

For record-keeping, the system allows teachers to download detailed attendance logs in Excel format (Fig. 12). It also generates absentee reports for specific dates, which can also be exported (Fig. 13). The system also allows users to update their profiles and class details (Fig. 14) and offers the innovative feature of connecting an Android device camera for attendance recording (Fig. 15), adding flexibility.

SrNo	Rollno	Name	MIP	AI	DSIP
1	1811084	Nandita	50.00	100.00	100.00
2	1811049	Shreyas	75.00	Not a part of class	50.00
3	1811053	Sakshi	Not a part of class	33.33	Not a part of class
4	1811042	Akshit	Not a part of class	66.67	Not a part of class
5	1811001	Aditya	Not a part of class	Not a part of class	0.00

Fig. 12: Example of a downloaded attendance report in XLSX format.

Name	Roll	Class	Date	Email
Mass Bunk	-	ML	Thu Apr 22	-
Aditya	1811001	DSIP	Thu Apr 22	aditya.pradhan@somaiya.edu
Akshit	1811042	AI	Thu Apr 22	akshit.gs@somaiya.edu
Sakshi	1811053	AI	Thu Apr 22	sakshi@somaiya.edu
Nandita	1811084	MIP	Thu Apr 22	nandita.kadam@somaiya.edu

Fig. 13: Absentee report showing students absent on a selected date, with export option.

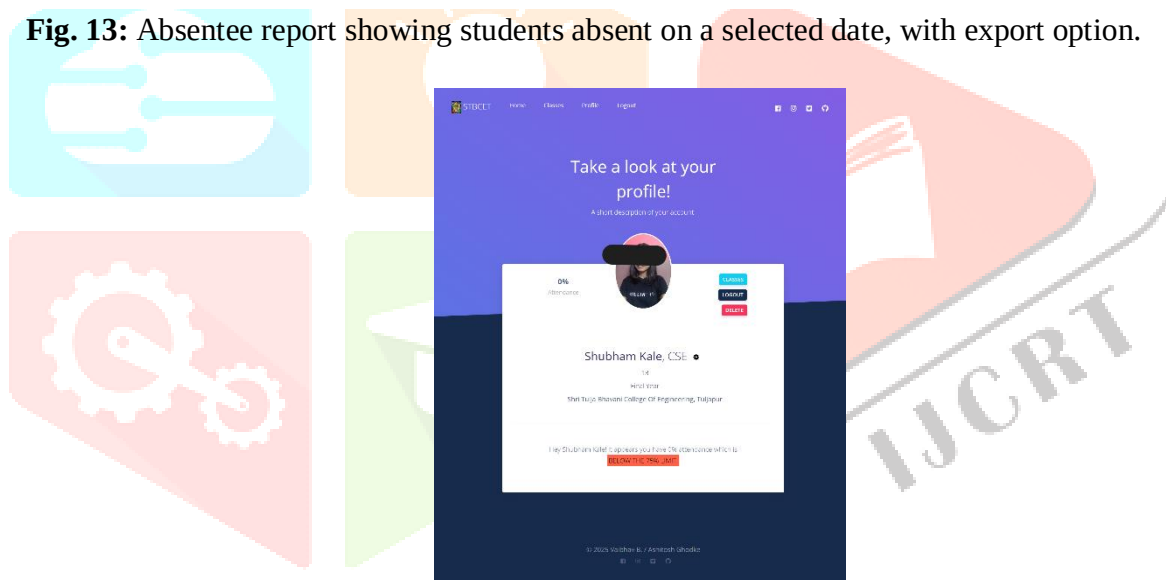


Fig. 14: User interface for updating profile details, ensuring current information is maintained.



Fig. 15: Interface demonstrating the integration of an Android device camera for attendance capture.

F. Discussion

The successful implementation of these features shows the effectiveness of the facial recognition attendance system. It significantly reduces the time spent on attendance and prevents proxy attendance. The contactless nature is especially relevant today. The integrated digital classroom framework provides a seamless experience for managing courses and communication. While the system works best with good lighting and clear face visibility, future improvements will focus on handling varied conditions and adding anti-spoofing measures. Overall, the system is a major step forward in automating academic administration with smart technology.

V. CONCLUSION AND FUTURE WORK

In this project, we've successfully built a web-based facial recognition attendance system that directly tackles the long-standing frustrations of traditional attendance methods. By combining smart facial recognition with a full-featured digital classroom, we've created a solution that is automatic, accurate, and completely contactless. Our work proves that you can do more than just streamline attendance and eliminate errors; you can create a richer, more interactive classroom experience. With an intuitive design for both students and teachers, powerful analytics dashboards, and simple data export features, this system is a practical and user-focused tool. It's a significant step toward a more organized, efficient, and technologically advanced learning environment, meeting the real-world needs of a modern smart campus.

As for what comes next, we see several exciting paths forward to make the system even better:

Our top priority is making it even more secure and reliable. We plan to integrate advanced "liveness detection" to prevent any attempts to fool the system with photos or videos. At the same time, we'll work on making the recognition technology tougher, training it to perform flawlessly in tricky lighting, or even when faces are partially covered.

We also want to ensure the system fits seamlessly into the tools schools already use. This means building deep integrations with popular Learning Management Systems (LMS) like Moodle and Canvas, creating a truly unified experience. To handle the scale of large universities, we'll continue to optimize performance to support thousands of students at once. Developing dedicated mobile apps is also on the roadmap to give everyone easy access on the go. Finally, we envision expanding our analytics to give students personalized feedback on how their attendance habits connect to their success, empowering them to take charge of their learning. By continuing down these paths, we believe this system can become an essential tool for any forward-thinking educational institution.

VI. ACKNOWLEDGMENT

We would like to thank our mentors for their expert guidance and continuous support, which were crucial in overcoming the challenges of this project. Our development was significantly accelerated by the use of design resources from Creative Tim, for which we are very grateful. We also thank our institution and colleagues for providing the supportive environment necessary to complete this research.

REFERENCES

- [1] Shreyas, S., Sakshi, S., & Akshit, A. (2021). CaptureIt! - A Web-based Attendance System Using Face Recognition. *2021 6th International Conference on Communication and Electronics Systems (ICCES)*, 283-288.
- [2] Viola, P., & Jones, M. J. (2001). Rapid object detection using a boosted cascade of simple features. *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001)*, 1(I), 511–518.
- [3] Schroff, F., Kalenichenko, D., & Philbin, J. (2015). FaceNet: A Unified Embedding for Face Recognition and Clustering. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 815-823.
- [4] Deng, J., Guo, J., Xue, N., & Zafeiriou, S. (2019). ArcFace: Additive Angular Margin Loss for Deep Face Recognition. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 4690-4699.
- [5] Jafri, R., & Arabnia, H. R. (2009). A survey of face recognition techniques. *Journal of Information Processing Systems*, 5(2), 41-68.

- [6] Kumar, A., & Singh, R. (2020). Automated attendance management system using face recognition. *International Journal of Engineering Research & Technology (IJERT)*, 9(05), 1083-1087.
- [7] Islam, R., & Hasan, M. K. (2021). A Comprehensive Review of Face Recognition for Attendance Systems. *Journal of Computer Science and Systems Biology*, 14(1), 1-8.
- [8] Node.js. (n.d.). *Node.js Official Documentation*. Retrieved from <https://nodejs.org/en/docs/> (Accessed: June 23, 2025).
- [9] Express.js. (n.d.). *Express.js Official Website*. Retrieved from <https://expressjs.com/> (Accessed: June 23, 2025).
- [10] MongoDB. (n.d.). *MongoDB Documentation*. Retrieved from <https://docs.mongodb.com/> (Accessed: June 23, 2025).
- [11] OpenCV. (n.d.). *OpenCV Official Documentation*. Retrieved from <https://docs.opencv.org/> (Accessed: June 23, 2025).
- [12] Flask. (n.d.). *Flask Documentation*. Retrieved from <https://flask.palletsprojects.com/> (Accessed: June 23, 2025).
- [13] Creative Tim. (n.d.). *Creative Tim Official Website*. Retrieved from <https://www.creative-tim.com/> (Accessed: June 23, 2025).

