

# Online Exam Proctoring System

Shreeshail<sup>1</sup>, Salini Abraham<sup>2</sup>, Piyush S B<sup>3</sup>, Shree Hari<sup>4</sup>, Sushanth M S<sup>5</sup>

<sup>1,3,4,5</sup>Member, Srinivas Institute of Technology, Mangalore, 574143, Karnataka, India

<sup>2</sup>Professor, Srinivas Institute of Technology, Mangalore, 574143, Karnataka, India

doi.org/10.64643/IJIRTV12I12-206842-459

**Abstract**—With increasing popularity of remote learning, online exams have been widely adopted in many institutions of higher education. There are, however, a number of difficulties involved with online testing like cheating, impersonation, tab switching, and help from external sources. Currently existing systems for online testing lack proper supervision for the student taking the exam. To solve such problems, an Online Exam Proctoring System will be designed which will monitor the face of the examinee throughout the examination using face recognition technology and log suspicious activities such as tab switching or inactivity on the computer screen. For coding-related exams, coding sandbox on the web browser will be provided to write and run code without any risk.

**Index Terms**—Online examination, AI proctoring, face recognition, tab-switch detection, coding sandbox, behavior analysis, academic integrity.

## I. INTRODUCTION

The current post-COVID period witnessed the emergence of online testing systems due to increased usage of digital learning platforms. While providing much needed flexibility to the students, the online exam format also makes it easier for them to indulge in unethical practices. Such acts like impersonation, tab switching, phone usage, or using help from outside become difficult to detect in a conventional online examination system. In this project, a solution for such problems will be proposed in the form of Online Examination Proctoring System which will incorporate facial recognition, behavior tracking, web browser activity tracking, and coding environment for online testing.

## II. LITERATURE REVIEW

Recent trends in the field of e-learning have led to the necessity for more advanced remote examination

evaluation processes which are reliable, intelligent and scalable, leading to academic focus on proctoring solutions powered by AI. In the paper by Sanjana Yadav and Archana Singh they came up with a system that uses the webcam and checks what the user is doing to find out if someone is cheating. This system looks for things like someone taking the test doing something off-screen or moving around in a weird way. It uses patterns and rules to figure this out. This system is good because it saves people from having to watch everyone all the time and it can check in time. It has some problems because it needs to have rules set up first and it can be affected by the environment so it is not very flexible.

Something similar happened in the paper by I. Ahmad and friends they made a system that uses learning to watch the test-taker and make sure they are not cheating. This system uses face recognition checks if the person is blinking and looks for objects that should not be there. It is better than the ways of doing things but it can be affected by how bright the room is, the quality of the webcam and if the internet is working well. All these things can make it hard for the system to work in time. In the paper by S. R. Chowdhury and friends they made a system that uses intelligence to watch the test-taker from far away. This system looks at the persons face. Checks what they are doing, like where they are looking and if they are moving their head. This system is good because it is more reliable and does not need many people to watch the test-takers. It has some problems like if the room is not bright enough if there is too much noise or if the computer is not good enough.

Navya Thampan and friends made a system that uses tools like LBPH, Dlib, OpenCV and YOLOv3 to watch the test-taker all the time and make sure they are not cheating. This system can find out if someone is looking away whispering or using something they should not be using. It has some problems like if the

webcam is not good enough if the room is too bright or too dark or if the person is too far away. Wijesekara and friends made a system that uses different ways to check if someone is cheating, like looking at what they are doing listening to what they are saying and checking the environment. This system is good because it can find out if someone is cheating accurately and it is more reliable than other systems. It needs a very powerful computer and it raises some concerns about privacy and if it can work in real-time.

All these studies show that artificial intelligence is getting better at helping with tests and making sure people are not cheating. They are good, at using face recognition checking what people are doing and using different ways to check for cheating. They all have some problems like needing good hardware being affected by the environment being too complicated and not being able to work well together. This means we need a system that can work on the web is strong and can use intelligence to make sure tests are safe and reliable. We need an artificial intelligence proctoring system that can make sure online tests are secure. Intelligence proctoring systems are the future of online tests.

### III. PROBLEM STATEMENT

Online examinations are really simple to set up. Students can take them from pretty much anywhere. They do have some problems though. For example, it is hard to stop students from cheating because there is no one watching them in person. Students can do things like switch to another tab on their computer, use a hidden phone, copy answers from the internet or even get someone to take the online examination for them.

Online examinations are not very secure because of this. Students can easily cheat during examinations. Most existing online exam platforms only verify the student once before the exam starts and do not monitor activities properly during the test. Programming exams also face problems because students can copy code from external websites or IDEs. To overcome these issues, this project proposes an AI-based online exam proctoring system that continuously monitors students using webcam analysis, behavior tracking, and browser activity detection along with a secure coding environment.

### IV. EXISTING SYSTEM

Online examination systems currently permit examinations to be held online and facilitate submission of answers online by the student. Nonetheless, the majority of the current systems lack essential capabilities to guarantee the integrity of the examination process. Almost all the platforms employ basic forms of identification such as passwords or one-time passcodes where the verification process occurs prior to the commencement of the exam. This makes way for impersonation and examination malpractice. Even though some of the existing systems include webcam surveillance features, they cannot distinguish between different faces within the same environment and also cannot identify any form of malpractice including frequent looking away from the laptop and using hidden devices while taking an examination. Some universities use human monitors who watch students through webcam stream; however, this poses financial strain on universities and is not always reliable. Sophisticated techniques such as biometric recognition or eye movement tracking help improve security measures but pose financial problems for universities as well as being impractical for remote learners using their personal computers. One other shortcoming is that there is no secure coding environment available when conducting programming exams. This allows students to easily change to another IDE or websites and thus increase the risk of plagiarism.

### V. PROPOSED SYSTEM

The suggested Online Exam Proctoring System aims to make online exams more secure and reliable. In the proctoring process, the student has to login through facial recognition apart from standard login procedures. It ensures that the right person attends the exam. While taking the exam, the system monitors the student through their webcam and detects any suspicious activity like presence of several faces, extended periods away from the computer screen, irregular head movements, and tab switching. Moreover, the system watches for any unusual browser activity that could be associated with attempting to exit the exam. In case of coding questions, the system offers an integrated code editor and the secure Node.js environment where the students

can execute their codes in the browser itself without the help of any other application software.

## VI. METHODOLOGY

The online exam proctoring tool does things in a careful and ordered way. It makes sure that the people taking the exam are who they say they are. The online exam proctoring tool also watches what the people are doing all the time and checks their work. The online exam proctoring tool does all of this to make sure everything is fair and safe. The online exam proctoring tool is very good, at keeping an eye on things. In the first step, live pictures are captured for identity verification, and a face recognizer algorithm analyzes those images and confirms that it is indeed the identity of the candidate taking the exam. Along with the confirmation of identity, the system also confirms if there are adequate lighting conditions to capture the face, and if the candidate's face is clearly visible. Once this step is completed successfully, the second step involves Live Proctoring, which includes tracking the candidate's face orientation, gaze orientation, multiple faces, unusual head movements, etc. Simultaneously, the system is tracking the browser activities like tab switching's and navigation, which might result in assigning an integrity score to the candidate if any unusual activities take place. The automated scoring module immediately calculates the grades for objective questions and programming answers executed by the candidate are validated through a secure sandbox and test cases and marks are awarded accordingly. HTTPS protocol is used. Bcrypt hashing is used for passwords. Client-side webcam data processing happens unless the candidate chooses to record.

### A. Overall Design

The system is built in a way. This means features like authentication, exam management, proctoring and evaluation work together as parts. The frontend is built with React.js. The backend uses Node.js, Express.js and MongoDB for handling APIs, workflows and storing data. This setup makes the system scalable and easy to maintain. The modular design also helps to update features without disrupting the whole application. By doing the system becomes more reliable. It can handle scale

online exams smoothly. The system operates efficiently. It supports a number of users. This approach helps to prevent performance issues. The system works under pressure. It provides an experience, for users. The features work together. The system is easy to use. It meets the needs of users. The modular architecture is useful. It helps to achieve the goals of the system. The system is designed to perform. It handles exams efficiently. The features are well integrated. The system is reliable. The performance is smooth.

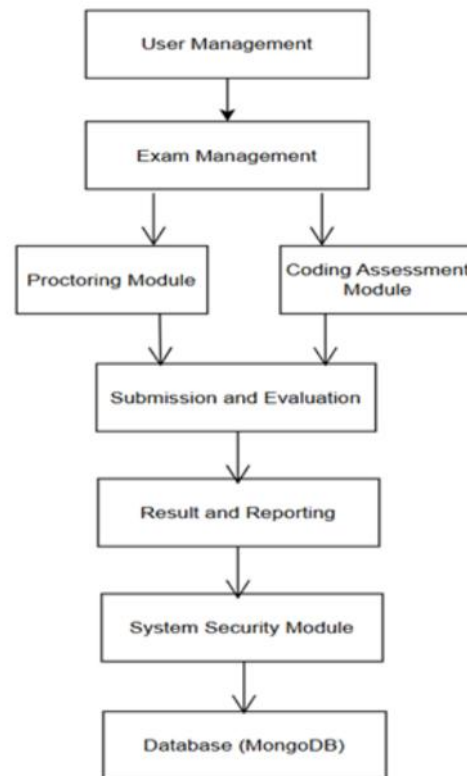


Fig. 1. Module Design

We have broken the system down into its essential parts with our Module Design: User and Exam Management, Proctoring, a Coding Environment, plus Result Processing and Reporting. The idea is for each of these to stand on its own, so you can put one together, test it or maintain it without any ripple effect on the rest. It makes for a more robust design and allows the front-end, back-end and database to interact in a seamless fashion through some well-defined APIs. When it comes to real-time orchestration of things like code execution, tab switching, webcam use or auto-saving, the modules

are in constant communication. You will find the Proctoring Module has its hands on the browser and webcam, while the Exam side of things dictates how and when questions are put before the user. The Coding Module will run your code in a secure sandbox, and the Result Module is where scores and integrity records are handled.

### B. Proctoring Engine

At the heart of the system's security lies the proctoring engine. Relying on face-api.js models, it is able to spot and verify faces on the fly, even when there are several in view, while also keeping an eye on where the subject is looking. The engine will flag anything that seems amiss by way of background motion, head position or a change in brightness. It is also monitoring the browser for any telltale signs such as a tab being switched or the exam window closed down. All of this is put on record with a time stamp; from these logs an integrity score is derived so that once the exam is over, a teacher can have a look at the student's behavior and form an opinion.

### C. Exam Management Module

The exam management module helps manage the delivery and execution of assessments in a controlled way. It makes sure that assessments are delivered and executed properly. The module oversees assessments. It controls how assessments are delivered. The exam management module is, in charge of assessments. It facilitates test creation, test scheduling, question bank allocation, and authentication of users. When a test starts, the entire system locks itself down, ensuring that all the timers are synchronized on both the front and back ends and activating all the proctoring systems. The module keeps communicating with the proctoring engine at all times, logging all behavioral events as well as all violations. Automatic saving, resuming, and submission management are some features that ensure that nothing goes missing while keeping track of all the students' actions during the test.

## VII. IMPLEMENTATION

The system is implemented using a modular and scalable architecture consisting of dedicated frontend, backend, database, and AI-driven proctoring components. React.js technology is used for the

creation of a highly reactive and adaptive interface, providing self-adjustability of the platform on laptop, tablet, and smartphone screen sizes by employing CSS flex and grid layouts along with the application of media queries. It guarantees the flawless operation of vital features in online exams like live popup monitoring, violations identification, timing verification, transition between questions, and in-browser code editor. Backend development is carried out with the help of Node.js programming language and Express.js framework. It serves as a connection point for performing tasks such as authentication, scheduling and distributing exams, handling real-time proctoring events, storing behavioural history, saving data and executing code requests. A highly scalable distributed NoSQL database - MongoDB - is employed for storing user credentials, configuration settings of online exams, question bank, coding test cases, students' answers, logging proctoring events, and user identities embedding with high consistency and rapidity of read-write operations.

The real-time AI proctoring is implemented via face-api.js library, providing client-side face detection and recognition features with multi-facial check to decrease latency time and preserve users' privacy. All modules operate by means of encrypted HTTPS REST API, ensuring continuous synchronization between frontend and backend throughout the exams. The platform was tested on browsers and devices to make sure it looks the same and works properly everywhere. We checked that the webcam works well there is not delay and the system can detect what is happening during the test. This way we know the platform will work well for a lot of people at the time. It will be safe and reliable, during the test process. The platform will keep all information safe and secure.

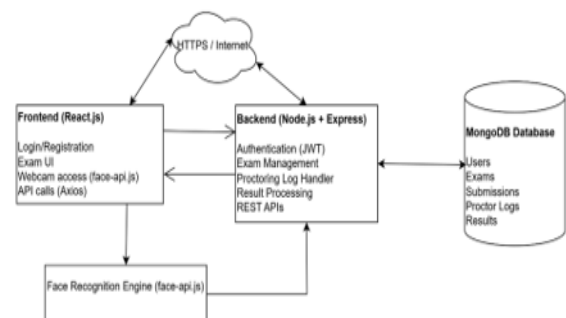


Fig. 2. System Architecture

## VIII. RESULTS AND DISCUSSIONS

The Online Exam Proctoring System worked well when we tested it. It was able to find things like when someone had faces on the screen or when they switched tabs. The Online Exam Proctoring System also found out when someones identity did not match. All of this happened in time. The face recognition part of the Online Exam Proctoring System worked fine when the lighting was normal. The behaviour monitoring system part of the Online Exam Proctoring System was good at finding actions that seemed suspicious during the exam. The coding environment part of the Online Exam Proctoring System was also very good. It ran the student's programs in a place and checked the results with some test cases that we already knew. The Online Exam Proctoring System had some features. It saved the work automatically. Also sent it in automatically. This helped when the internet was not working properly. Overall, the Online Exam Proctoring System made the exam process very secure and smooth. It helped make the online tests fair and transparent. The Online Exam Proctoring System is very helpful, for assessments.

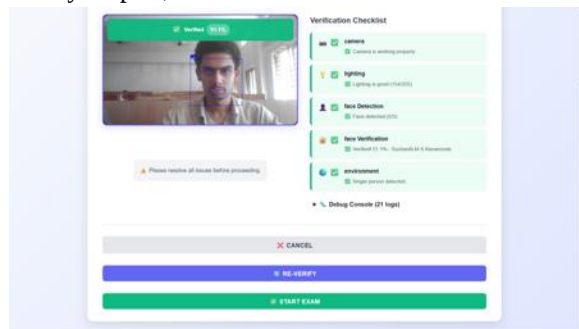


Fig. 3. Proctoring the Student

The Pre-Exam Setup page ensures all proctoring conditions are met before the exam begins. The application shows a live feed from the webcam, and the student's face is verified using the face-api.js library. The system tests whether the camera works, the illumination, the quality of face detection, the matching of the person's face with their identity, and whether there is only one person in the room. All these factors are highlighted with a green light if everything is correct or an exclamation mark if something should be checked again. The student is not allowed to start the exam until all these conditions are met. The second step is considered a mandatory entrance check that

ensures the legitimacy of the examination process and eliminates the risk of cheating by the student. The real-time proctoring engine monitors the behavior of the student during the exam. The system matches the person's identity based on live data and saved photos and sends notifications when there are any inconsistencies or more than one person is found on the video. In addition, this system analyses such actions as switching tabs, minimizing windows, and changing the focus of attention. If something is not done correctly it will be written down. This will have a bad effect, on the integrity score. The integrity score is important. Any mistakes will hurt the integrity score.

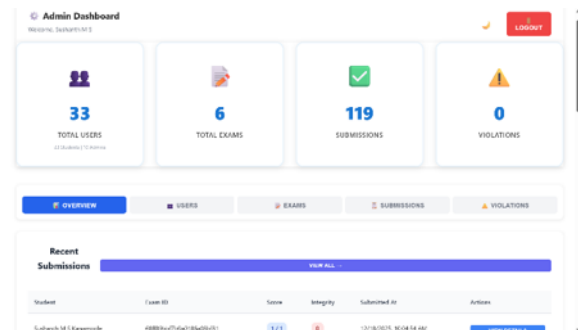


Fig. 4. Admin Dashboard

The Admin Dashboard is, like the control room for administrators to manage the examination system. From here they can create exams. They can also add choice questions and coding problems. Admins can even. Delete existing questions if needed. They can manage users by turning their accounts on or off. They can also change student and admin accounts. The dashboard also helps admins to: 1. See exams, 2. Change scheduled exams 3. Watch what students are doing, 4. Look at reports. These reports include: Proctoring logs, Coding results, and Integrity scores. This dashboard makes it easy for admins to do all their work in one place.

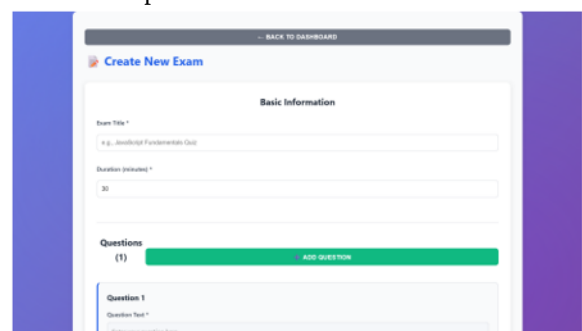


Fig. 5. Exam Creation by Admin

The Online Exam Proctoring System is where the administrator manages the exam creation process. They do this through a secure and easy to use interface. The administrator can decide all the things about the exam. This includes the Online Exam Proctoring System exam title, how time students have to take the Online Exam Proctoring System exam, when the Online Exam Proctoring System exam is scheduled how many marks the Online Exam Proctoring System exam is worth and what kind of questions are on the Online Exam Proctoring System exam. The questions on the Online Exam Proctoring System exam can be choice or coding questions. The administrator can upload questions to the question bank. Add them by hand. They can also add how hard the question is, pictures and coding samples. They can even decide how the question will be graded. For coding exams on the Online Exam Proctoring System, the administrator can add test cases to see if the code works right. The test cases are used in a place called a sandbox environment. The administrator can also decide if they want to use things like face monitoring or alerts when someone switches tabs or loses their screen. They can turn these things on or, off depending on what the Online Exam Proctoring System exam needs. After setting up these details, the administrator proceeds to publish the exam, which makes it available only to authorized and authenticated users for the designated schedule period.

#### REFERENCES

- [1] N. H. I. Teo, N. Abu-Bakar, and S. Karim, "Designing GA-Based Auto-Generator of Examination Questions," in *Proc. UKSim-AMSS European Modelling Symposium*, 2012.
- [2] B. Keresztury and L. Cser, "New cheating methods in the electronic teaching era," *Procedia – Social and Behavioural Sciences*, vol. 93, pp. 1516–1520, 2013.
- [3] P. V. Secreto and R. L. Pamulaklakin, "Learners' satisfaction level with online student portal as a support system in an open and distance e-learning environment (ODEL)," *Turkish Online Journal of Distance Education*, vol. 16, no. 3, pp. 33–46, 2015.
- [4] S. Prathish, A. N. S., and K. Bijlani, "An Intelligent System for Online Examination," in *Proc. International Conference on Information Science (ICIS)*, pp. 138–143, 2016.
- [5] S. Yadav and A. Singh, "An Image Matching and Object Recognition System Using Webcam," in *Proc. 4th International Conference on Parallel, Distributed and Grid Computing (PDGC)*, 2016.
- [6] R. Bawarith, A. Basuhail, A. Fattouh, and S. Gamalel-Din, "E-exam cheating detection system," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 4, pp. 176–183, 2017.
- [7] T. Weihui, L. Bin, Z. Haitao, *et al.*, "Internet-Based Financial Job Qualification Training and Certification System Design," *Science and Technology Wind*, no. 24, 2020.
- [8] N. Parveen, S. Ranga, G. Nishanth, C. S. Abhijith, and A. H. K. Reddy, "Work Force Management System Using Face Recognition," *Turkish Journal of Computer and Mathematics Education*, vol. 12, no. 9, pp. 56–61, 2021.
- [9] I. Ahmad, F. AlQurashi, E. Abozinadah, and R. Mehmood, "A Novel Deep Learning-Based Online Proctoring System Using Face Recognition, Eye Blinking, and Object Detection Techniques," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 10, 2021.
- [10] N. C. M., P. Sharma, U. Tripathi, U. Kumar, and G. C. B. Prakash, "Automating Online Proctoring Through Artificial Intelligence," *International Research Journal of Engineering and Technology (IRJET)*, vol. 8, no. 1, 2021.
- [11] S. R. Chowdhury, P. Biswas, and M. S. Hossain, "AI-Enabled Remote Exam Proctoring System Using Facial Recognition and Activity Monitoring," *International Journal of Computational Intelligence and Applications*, vol. 20, no. 3, 2021.
- [12] A. Tweissi, W. Al Etaiwi, and D. Al Eisawi, "The Accuracy of AI-Based Automatic Proctoring in Online Exams," *Electronic Journal of e-Learning*, vol. 20, no. 4, 2022.
- [13] N. Thampan, S. A. Muthukumaraswamy, and S. A., "Smart Online Exam Invigilation Using AI-Based Facial Detection and Recognition Algorithms," in *Proc. 2nd Asian Conference on Innovation in Technology (ASIANCON)*, 2022.
- [14] L. Liu, "Facial Authentication System Design of Online Interactive Platform for Innovation and

- Entrepreneurship Courses for Mobile Platform Terminals,” in *Proc. 3rd International Conference on Smart Electronics and Communication (ICOSEC)*, 2022.
- [15] A. Al-Hassan, M. Al-Mayouf, and Y. K. Malek, “Deep Learning-Based Cheating Detection for Online Examinations,” *IEEE Access*, vol. 10, pp. 98765–98778, 2022.
- [16] R. Jain, S. Sharma, and P. Chauhan, “Automated Online Examination Monitoring Using Computer Vision Techniques,” *International Journal of Engineering Research & Technology (IJERT)*, vol. 11, no. 6, pp. 1–6, 2022.
- [17] L. Priyanka and K. Shravani, “An Intelligent Web-Based Tracking System Powered by Artificial Intelligence,” *Material Science and Technology*, vol. 22, no. 3, 2023.
- [18] M. M. Rahman, F. Altaf, and S. Almotiri, “A Hybrid Deep Learning Model for Online Exam Proctoring and Student Authentication,” *Sensors*, vol. 23, no. 4, 2023.
- [19] K. S. Wijesekara, Y. R. Wickramasinghe, and N. D. Kodikara, “Real-Time Cheating Detection in Remote Exams Using Multimodal AI Techniques,” *Journal of Information Security and Applications*, vol. 76, 2023.
- [20] T. Veeramani, Tharun, S. Krishna, T. Kumar, and L. Prasad, “Online Exam Proctoring System Based on Artificial Intelligence,” *International Journal of Novel Research and Development*, vol. 9, no. 4, 2024.
- [21] J. Oeding, T. Gunn, and J. Seitz, “The Mixed-Bag Impact of Online Proctoring Software in Undergraduate Courses,” *Open Praxis*, vol. 16, no. 1, 2024.
- [22] N. Bhatnagar, M. Kakkar, M. Dahiya, and L. Mangla, “Online Proctoring System Using Machine Learning and Image Processing,” *International Journal of Current Science*, vol. 14, no. 2, pp. 413–417, 2024.
- [23] G. Borade, S. Rahane, N. Walke, and R. M. Shaikh, “Smart Proctor Hub – Online Exam Proctoring System,” *International Research Journal of Modernization in Engineering, Technology and Science*, vol. 7, no. 4, 2025.
- [24] K. L. Narayana, G. Dinesh, K. K. Kumar, P. P. Adithya, and M. H. V. S. S. Reddy, “Online Exam Proctoring,” *International Journal of Innovative Science and Research Technology*, vol. 10, no. 4, pp. 1568–1582, 2025.
- [25] S. Neeraj, “Advanced AI Examination Monitoring and Security System,” in *Proc. 2025 International Conference on Smart Computing and Artificial Intelligence*, 2025.