



PROCTOR EXAM SOFTWARE

¹Aditya Madhukar Bhat, ²Athmaranjan K, ³Anjankumar Nitin Nayak, ⁴Prasanna P, ⁵Vinayak Shivalingayya Hiremath,
^{1,3,4,5}Student, ²Associate Professor,

¹Department of Computer Science and Design

^{2,3,4,5}Department of Information Science & Engineering

^{1,2,3,4,5}Srinivas Institute of Technology, Mangalore, India.

ABSTRACT

In the evolving landscape of digital education, ensuring the integrity and fairness of online examinations has become a critical concern. This paper presents the design and implementation of a secure and scalable Proctor Exam Software aimed at preventing malpractice during remote assessments. The system architecture is divided into three hierarchical roles: Super Admin, Faculty, and Student. The Super Admin is responsible for approving faculty access and defining exam permissions. Approved faculty members can create and manage exams, receive a unique alphanumeric exam code, and analyse results. Students, upon entering the correct exam code and during the assigned time window, are granted a single opportunity to attempt the exam. The platform enforces strict proctoring controls such as full-screen mode enforcement, disabling of shortcut keys, prevention of tab-switching, and automatic submission in case of repeated violations. By integrating administrative control, secure exam creation, and real-time malpractice detection, the proposed system ensures a transparent and controlled environment for online assessments. This paper discusses the need, methodology, system workflow, and benefits of the proposed model.

1. INTRODUCTION

The increasing adoption of online learning platforms has brought about a significant transformation in the education sector, especially in the way assessments are conducted. As institutions shift towards digital methods of evaluation, maintaining the credibility and fairness of online examinations has emerged as a major challenge. Traditional examination systems, when adapted to a virtual format, often fail to implement effective monitoring and control mechanisms, resulting in increased instances of malpractice and compromised academic integrity.

To address these issues, there is a growing need for intelligent, automated, and secure proctoring solutions that not only facilitate remote exams but also ensure strict adherence to examination protocols. A robust online exam platform must incorporate multi-layered access control, seamless exam creation, real-time monitoring, and automatic violation handling to maintain the reliability of the assessment process.

This paper introduces a Proctor Exam Software system that offers a comprehensive framework for managing and conducting secure online examinations. The system is designed around three user roles: Super Admin, Faculty, and Student. Each role has clearly defined responsibilities and access levels, which together contribute to an organized, transparent, and secure examination environment. Key features include faculty approval workflows, secure exam creation with auto-generated codes, role-based permissions, full-screen enforcement, and real-time malpractice detection mechanisms. By integrating these components, the system offers a viable solution for institutions seeking to uphold academic standards in a digital setting.

2. PROBLEM STATEMENTS AND OBJECTIVES

Problem Statement

The shift to online education has exposed significant weaknesses in existing virtual examination systems, particularly in the areas of monitoring, security, and fairness. Many current platforms lack robust mechanisms to prevent cheating, making it difficult for institutions to guarantee the authenticity of student performance. The absence of real-time proctoring, limited control over user activity, and lack of accountability further diminish the reliability of online assessments. Additionally, without proper authorization workflows and role-based control, exam data can be misused or mismanaged. Therefore, there is a critical need for a secure, well-structured, and easy-to-administer online examination system that can effectively monitor student activity and prevent malpractice during exams.

Objectives

The main objectives of the proposed Proctor Exam Software are:

- To design a secure, role-based system where Super Admin, Faculty, and Students operate with distinct permissions and responsibilities.
- To implement a faculty approval mechanism controlled by the Super Admin, ensuring regulated access to exam creation features.
- To allow faculty to schedule exams with customizable parameters such as date, duration, total marks, passing criteria, and result declaration time.
- To auto-generate unique, tamper-proof alphanumeric exam codes for faculty to manage exam access securely.
- To develop a secure exam environment for students by enforcing full-screen mode and disabling shortcut keys, copy-paste functions, and tab switching.
- To detect and record malpractice attempts in real-time and automatically terminate exams if violations exceed a defined threshold.
- To ensure students can view results only after the specified result date, maintaining transparency and fairness in evaluation.

3. PROPOSED SYSTEM / METHODOLOGY

The proposed Proctor Exam Software adopts a structured, role-based methodology to manage and conduct online examinations securely and efficiently. The system is divided into three distinct modules: **Super Admin**, **Faculty**, and **Student**, each responsible for specific operations that collectively uphold the integrity of the examination process.

3.1 Super Admin Module

The Super Admin functions as the central authority responsible for approving faculty access to the platform. Faculty members are verified based on their subscription plan or institutional credentials. Once approved, a time-limited access period is granted during which faculty can conduct exams. The Super Admin is also tasked with monitoring faculty activities and ensuring compliance with institutional policies.

3.2 Faculty Module

Once access is granted, the faculty user can log in and proceed to create exams by specifying various parameters such as exam date, duration, total marks, passing marks, result declaration schedule, and exam timing. After these details are submitted, the system generates a unique 10-digit alphanumeric exam code, sent to the faculty via email. This code is used to manage question entry, student access, and result analysis.

The faculty is then required to enter the appropriate number of questions based on the total marks configured. The question set may be either objective or subjective, depending on the nature of the assessment. Additionally, the faculty can later use the exam code to review student performances and download result summaries.

3.3 Student Module

Students gain access to exams through the unique exam code shared by the faculty. They must log in and enter the code at the scheduled time. Access is only granted if the system clock matches the exam start time. Each student is allowed a single attempt per exam. Upon entry, the exam interface is locked to full-screen mode. All keyboard shortcuts, right-click functions, copy-paste features, and tab switching are disabled.

The system monitors the student's behavior in real-time. Any attempt to exit full-screen mode or switch tabs is recorded as a violation. If the number of violations exceeds a pre-defined threshold (e.g., three attempts), the exam is auto-submitted, and the student receives a score of zero. If the student successfully completes the exam without any malpractice, the submission is processed, and the result is made available on the pre-scheduled result date.

3.4 Implementation Highlights

- **Role-Based Access Control:** Each user role is granted only the necessary privileges, reducing the risk of unauthorized actions.
- **Secure Exam Code Generation:** Auto-generated, unique codes ensure exams are accessible only to authorized participants.
- **Real-Time Monitoring:** Active tracking of user actions during the exam helps detect malpractice early.
- **Single Attempt Policy:** Each student is allowed only one attempt per exam, ensuring fairness.
- **Delayed Result Release:** Results are disclosed only at the predetermined time, ensuring consistent release for all participants.

4. SYSTEM ARCHITECTURE & WORKFLOW

The architecture of the Proctor Exam Software is designed to provide a robust, scalable, and secure environment for online assessments. It consists of several components that interact seamlessly to deliver a user-friendly experience for Super Admin, Faculty, and Students while ensuring the integrity of the examination process. The system follows a client-server architecture, utilizing a web-based interface for ease of access and management.

4.1 System Architecture

The system is divided into three main modules based on user roles: **Super Admin**, **Faculty**, and **Student**. The architecture is structured as follows:

1. Client Layer:

- Users (Super Admin, Faculty, and Students) interact with the system through a browser-based interface, accessible on any device with internet connectivity.
- The interface is designed to be responsive, ensuring compatibility across multiple devices, including desktops, tablets, and smartphones.

2. Application Layer:

- The core logic and business processes are handled in this layer. This includes authentication, role-based access control, exam creation, result management, and real-time monitoring.
- The system also manages the communication between modules, such as sending exam codes, receiving exam submissions, and generating results.
- Security measures, including encryption, secure login processes, and monitoring of user activity, are also implemented in this layer.

3. Database Layer:

- A relational database management system (RDBMS) is used to store all data, including user credentials, exam details, student submissions, and result history.
- The database ensures data integrity, prevents unauthorized access, and supports efficient query processing for exam data retrieval and reporting.

4. Security Layer:

- This layer ensures that the system operates securely, protecting user data, preventing malpractice, and maintaining the integrity of exams.
- Security protocols, such as SSL encryption, user authentication, and real-time monitoring, are incorporated to safeguard the examination process.

4.2 Workflow

The workflow of the Proctor Exam Software follows a systematic, step-by-step process that integrates all three user roles efficiently.

1. Super Admin Workflow:

- The Super Admin registers and authenticates faculty users, granting access based on their subscription plan or institutional affiliation.
- Once approved, faculty can create exams and assign specific durations for exam availability.

2. Faculty Workflow:

- The faculty user logs into the system and creates exams by providing essential details like exam date, total marks, passing marks, and result date.
- After exam creation, the system generates a unique 10-digit alphanumeric code, which is sent to the faculty's email.
- Faculty can then add the number of questions based on the total exam marks and prepare the exam for student participation.
- The faculty can review the results after the exam is completed, analyzing the performance based on the exam code.

3. Student Workflow:

- Students log in using the unique exam code provided by the faculty.
- The system checks the current time to ensure that the exam is being accessed within the defined start and end times.
- Once the exam starts, the student's screen enters full-screen mode, and all non-essential features, such as right-click and keyboard shortcuts, are disabled.
- Students are given only one attempt to complete the exam. If they attempt to exit full-screen mode or switch tabs, it is flagged as a malpractice attempt.
- If malpractice is detected more than three times, the system auto-submits the exam, marking the student's score as zero.
- After the exam is submitted, students can view their results once the predefined result release time has arrived.

4.3 Interaction Diagram

The interaction between Super Admin, Faculty, and Student is visualized through a flow diagram, demonstrating the sequence of actions each user undertakes:

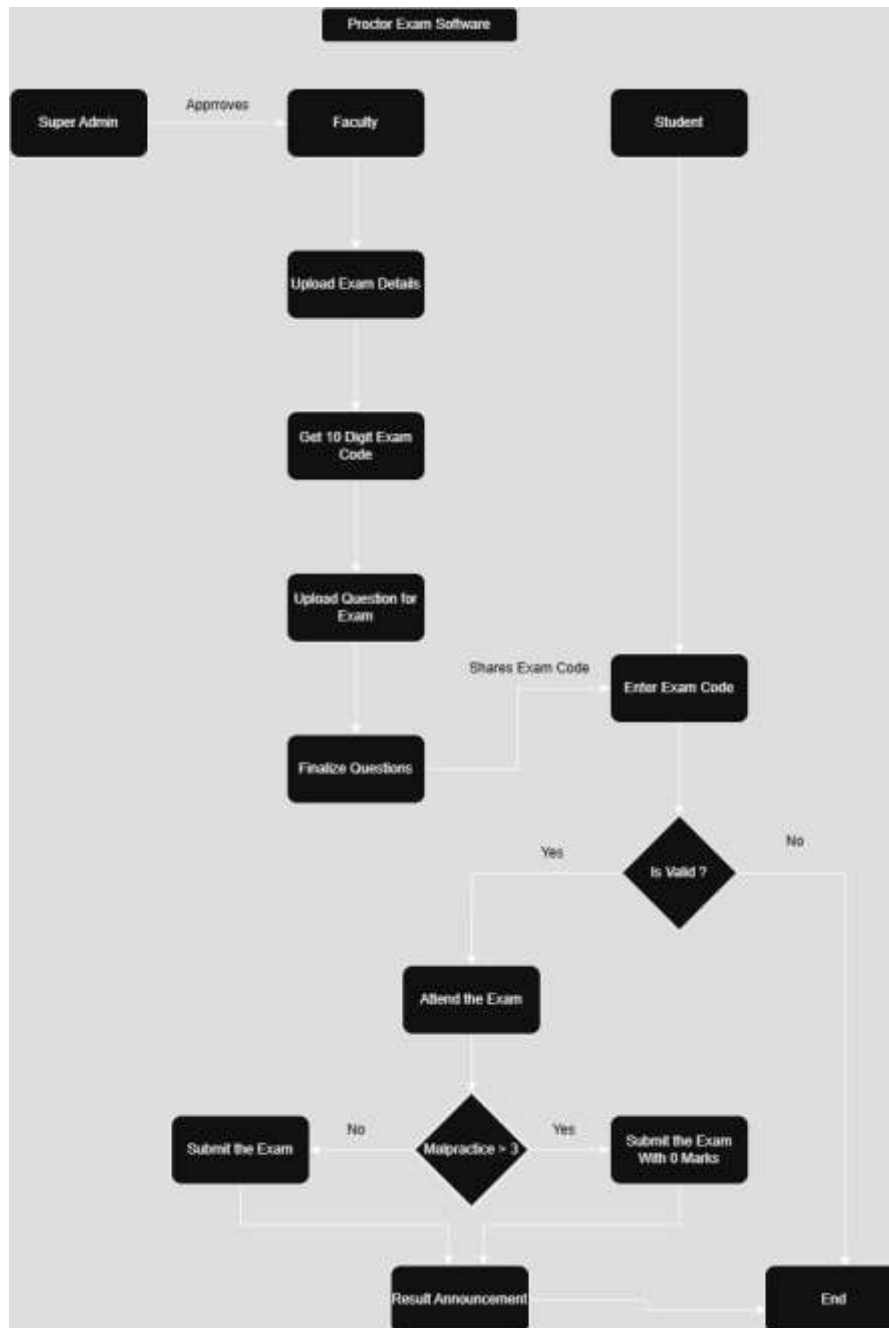


Figure 4.3

1. Super Admin approves faculty users and assigns exam access rights.
2. Faculty enters exam details, which triggers the generation of a unique exam code.
3. Students access the exam using the provided code, following the specified time window and security protocols.
4. The system monitors student behaviour, ensures no malpractice occurs, and provides exam results as per the faculty-defined schedule.

5. EXPECTED OUTCOMES / ADVANTAGES

The implementation of the Proctor Exam Software is expected to deliver a comprehensive and secure platform for conducting online examinations, addressing both logistical and integrity-related challenges associated with digital assessments.

5.1 Expected Outcomes

1. **Streamlined Examination Process:** The system will enable educational institutions and individual faculty members to organize and manage exams with ease, eliminating the need for physical supervision and reducing administrative overhead.
2. **Automated Monitoring and Control:** By incorporating real-time restrictions like screen locking, tab-switch detection, and shortcut key disabling, the software ensures continuous surveillance throughout the exam duration, minimizing the likelihood of malpractice.
3. **Customizable Exam Creation:** Faculty will be able to design exams with flexible parameters, such as setting exam duration, marks distribution, and scheduling result announcements, tailored to their academic requirements.
4. **Single-Attempt Enforcement:** Students are restricted to a one-time login and submission attempt per exam, ensuring fairness and preventing multiple trials that could otherwise be exploited.
5. **Immediate Response to Malpractice:** The system's built-in behaviour detection automatically responds to suspicious actions, such as exiting full-screen mode or changing browser tabs. Accumulated violations trigger automatic submission and scoring penalties.
6. **Time-Gated Access:** Exams are accessible only within the scheduled start and end times, ensuring that no student gains an unfair advantage by starting early or late.
7. **Scheduled Result Declaration:** Students will receive their results at the pre-defined date and time, eliminating manual intervention and promoting transparency in score release.

5.2 Key Advantages

- **Security and Integrity:** The system enforces strict exam protocols, promoting academic honesty and deterring cheating through intelligent monitoring and automatic action-taking mechanisms.
- **Scalability:** Designed to support multiple institutions and faculty members, the platform is scalable for wide usage across departments, colleges, or universities.
- **Ease of Use:** With a user-friendly interface for all three roles—Super Admin, Faculty, and Students—the system ensures ease of navigation and minimal learning curve.
- **Automation and Efficiency:** Automating exam creation, monitoring, and result publication drastically reduces the manual workload on faculty and administrative staff.
- **Cost-Effectiveness:** As an online solution, it reduces the costs associated with physical infrastructure, printed question papers, and in-person invigilation.
- **Accountability and Transparency:** All activities are logged, and the result system is tamper-proof, ensuring accountability at every stage of the exam process.

6. CONCLUSION

The transition to digital education has necessitated the development of secure and reliable online examination systems. This paper presented a comprehensive Proctor Exam Software tailored to meet the growing demand for integrity-driven assessment platforms. By integrating role-specific functionalities for Super Admins, Faculty, and Students, the system ensures a streamlined and controlled exam environment from setup to result declaration.

With features such as real-time monitoring, screen lock enforcement, single-attempt access, and automated malpractice detection, the platform addresses the critical challenges of academic dishonesty and exam integrity. Additionally, time-based access controls and secure result handling further enhance the transparency and trustworthiness of the process.

The modular and scalable architecture allows institutions to adopt the solution with minimal infrastructure while maintaining high security standards. This paper demonstrates that the proposed system not only simplifies examination logistics but also significantly enhances credibility and fairness in the digital evaluation process.

7. BIBLIOGRAPHY

- [1] N. Bansal and R. K. Mishra, "Securing Online Examinations through Browser Locking and AI Proctoring Mechanisms," *International Journal of Computer Science and Applications*, vol. 12, no. 2, pp. 34–41, 2021.
- [2] A. Kumar, S. Dutta, and M. Rao, "AI-Based Online Proctoring System for Secure E-Assessments," in *Proceedings of the 2022 International Conference on Smart Computing and Cyber Security*, IEEE, pp. 97–102.
- [3] T. Singh and R. Kaur, "A Study on Cheating Prevention Techniques in Online Exams," *International Journal of Education and Development using ICT*, vol. 17, no. 3, pp. 89–95, 2021.
- [4] P. S. Rao and S. A. Narayan, "Design and Implementation of a Web-Based Remote Examination System," *Journal of Educational Technology Systems*, vol. 50, no. 1, pp. 25–38, 2022.
- [5] M. J. Roy, "An Analytical Review of Role-Based Access in Educational Platforms," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 10, no. 4, pp. 56–61, 2021.
- [6] R. Sharma, "Secure Online Exam Platforms and Academic Integrity," *Education and Information Technologies*, Springer, vol. 27, no. 5, pp. 6569–6585, 2022.

