B.Tech. CSE (III YEAR – V SEM) (2025-2026)

DEPARTMENT OF COMPUTER ENGINEERING & APPLICATIONS

GLA University

**17km Stone, NH-19, Mathura-Delhi Road P.O.
Chaumuhan, Mathura - 281406 (Uttar Pradesh) India**

Report on
(AI Proctoing System For Online Exams)

Team Member 1: Rahul Gupta	UR: 2315800067
Team Member 2: Shivam Rajput	UR: 2315800076

Mentor Name: Dr. Gaurav Bathla

INDEX

1. Introduction

In recent years, the demand for online examinations has grown rapidly due to advances in digital learning platforms and the need for flexible education systems. However, this shift has introduced new challenges in maintaining the fairness, integrity, and security of assessments conducted remotely. Traditional invigilation methods are impractical for online settings, creating opportunities for malpractice and impersonation.

The **AI Proctoring System for Online Examination** aims to address these challenges through the integration of artificial intelligence technologies with a robust, scalable, and secure examination platform. Built on the MERN (MongoDB, Express, React, Node.js) stack, and incorporating TensorFlow.js for intelligent proctoring features, this system provides a comprehensive solution for conducting online examinations with real-time monitoring, automated cheat detection, and efficient reporting mechanisms.

The traditional examination environment relies heavily on human invigilators to supervise candidates, verify their identities, and monitor their behavior. In an online environment, however, human invigilation becomes resource-intensive, error-prone, and logistically unfeasible, particularly for large-scale examinations involving hundreds or thousands of candidates. Moreover, online exams conducted without adequate supervision have been shown to suffer from high rates of academic dishonesty, including impersonation, the use of unauthorized resources, and collaboration among candidates. These challenges undermine the validity and credibility of the examination process.

By integrating computer vision and machine learning techniques, the AI Proctoring System automates essential invigilation tasks such as identity verification, face detection, multiple-face detection, and activity monitoring. The system can detect suspicious behaviors in real time, including the presence of unauthorized devices, attempts to bypass the webcam, or leaving the examination area. These features dramatically reduce the risk of cheating while allowing examinations to be conducted at scale, across diverse geographical regions.

The proposed system also improves the user experience for both students and instructors. Students benefit from a streamlined examination workflow, while instructors gain powerful dashboards to configure exams, monitor live proctoring results, and review logs of suspected malpractice. Automated reporting ensures that actionable information is available immediately after an exam concludes, facilitating faster evaluation of cheating incidents.

In conclusion, the **AI Proctoring System for Online Examination** represents a transformative approach to secure, fair, and scalable online assessment. By leveraging artificial intelligence, modern web technologies, and an open-source framework, the system addresses the fundamental challenges of traditional proctoring and delivers a future-ready solution for the evolving world of education.

2. Organization / Company Profile

The industrial training for this project was completed at **GLA University, Mathura** as a part of the **JOVAC Online Training Program**. GLA University is a renowned institution known for its strong academic curriculum, industry-focused training programs, and emphasis on practical learning. **JOVAC Online** is GLA University's digital training initiative designed to equip students with modern technical skills through hands-on learning. The platform focuses on emerging technologies such as Artificial Intelligence, Machine Learning, Full Stack Development, and Cybersecurity.

Through the JOVAC Online program, students receive:

- Structured training modules aligned with industry requirements.
- Practical exposure through real-time projects and assignments.
- Mentorship from experienced trainers and domain experts.
- Opportunities to build technical and problem-solving skills.

As part of this training, I worked on the **AI-Based Proctored Examination System**, which helped me gain hands-on experience in:

- MERN Stack development
- AI model integration using YOLOv8
- Computer Vision techniques with OpenCV
- Designing real-time proctoring solutions

The training provided by **GLA University (JOVAC Online)** has contributed significantly to my technical growth and understanding of real-world AI applications.

3. Objectives of the Project

The main objectives of developing the **AI-Based Proctored Examination System** are:

1. **To enhance online examination security** by detecting cheating behaviors using AI and machine learning models.
2. **To implement real-time monitoring** through face tracking, eye tracking, head pose estimation, and object detection.
3. **To automate the proctoring process** and reduce the dependency on manual human invigilators.
4. **To build a user-friendly examination portal** using the MERN stack for students and teachers.
5. **To generate detailed cheating logs and alerts** for teachers through an interactive dashboard.
6. **To ensure identity verification and integrity** of candidates throughout the exam duration.
7. **To integrate ML models like YOLOv8 and OpenCV** for accurate detection of multiple faces, mobile phones, and suspicious movements.

4. Technology Used

The AI-Based Proctored Examination System utilizes a combination of modern web development tools and advanced machine learning models to provide real-time monitoring and secure examination functionality. The major technologies used in this project include:

1. Frontend Technologies

- **HTML, CSS, JavaScript** – Core technologies for structuring, styling, and adding dynamic behavior to the web interface.
- **React.js** – Used to build an interactive and component-based user interface for students and teachers.
- **Redux Toolkit** – Manages global state, including authentication, exam data, and cheating logs.
- **Material UI** – Provides modern, responsive, and user-friendly UI components.
- **React Router** – Enables smooth navigation across different pages of the application.
- **React-Webcam** – Captures real-time video stream from the user's webcam for AI monitoring.
- **TensorFlow.js** – Executes AI models directly in the browser for real-time proctoring tasks.

2. Backend Technologies

- **Node.js** – JavaScript runtime for building a scalable backend server.
- **Express.js** – Lightweight web framework used for building APIs and managing routing.
- **MongoDB** – NoSQL database for storing user accounts, exam details, and cheating logs.
- **Mongoose** – ODM library for modeling MongoDB documents with ease and validation.
- **JSON Web Tokens (JWT)** – Ensures secure user authentication and authorization.

- **bcryptjs** – Used for hashing passwords and securing sensitive user credentials.

3. AI & Proctoring Technologies

- **YOLOv8** – Real-time object detection model used for identifying mobile phones, multiple faces, and unknown persons during exams.
- **OpenCV** – Used for face tracking, eye tracking, and head pose estimation to detect suspicious movements.
- **Python** – Integrates ML models and handles processing of frames and detection pipelines.
- **NumPy & Pandas** – For data handling, frame processing, and storing detection results.

5. System Overview / Proposed System

The **AI-Based Proctored Examination System** is designed to provide a secure and reliable online exam environment by integrating advanced machine learning models with a modern web-based architecture. The system monitors the candidate in real time using webcam input, detects suspicious activities, and logs cheating incidents for teacher review.

The solution is divided into two major parts:

1. AI Proctoring Module (Python + YOLOv8 + OpenCV)

This module handles all real-time monitoring activities during an online exam. It performs:

- **Face detection & tracking**
- **Multiple-person detection** (to ensure only one candidate is present)
- **Mobile phone detection** using YOLOv8
- **Eye movement tracking** to identify if the candidate is looking away
- **Head pose estimation** to detect abnormal head rotations
- **Absence detection** when the candidate leaves the screen

The module continuously analyzes webcam video frames and sends alerts or cheating logs to the backend.

2. MERN-Based Web Portal (Frontend + Backend)

The system includes a complete exam portal for teachers and students with the following features:

For Teachers

- Create and manage exams
- Configure questions and exam rules
- View real-time alerts and cheating logs
- Access student-wise reports generated by the AI module

For Students

- Log in securely
- Attempt exams with a timer
- Get automatically monitored through the webcam
- All actions during the exam are tracked for integrity

System Flow Summary

1. Student logs in and starts the exam.
2. Webcam feed is captured and analyzed using YOLOv8 + OpenCV.
3. Detected cheating events (mobile phone, extra person, looking away, etc.) are sent to the backend.
4. Teacher dashboard displays real-time alerts and stores logs in MongoDB.
5. At the end of the exam, a complete summary is generated.

The proposed system eliminates the need for manual invigilation by providing an **automated, intelligent, and scalable proctoring solution**. It ensures exam integrity while offering convenience for both teachers and students.

6. ML Models & Proctoring Technologies (YOLOv8 + OpenCV)

The AI-Proctored Examination System uses a combination of advanced Machine Learning (ML) and Computer Vision (CV) technologies to detect suspicious behavior during online exams. These models work together to continuously analyze the candidate's webcam feed and ensure the integrity of the examination process.

1. YOLOv8 (You Only Look Once – Version 8)

YOLOv8 is a state-of-the-art real-time object detection model used in this project for identifying cheating-related objects and activities. It provides high accuracy and fast processing, making it suitable for live proctoring.

Key Functions:

- **Mobile phone detection** during the exam.
- **Multiple person detection** to ensure only one candidate is present.
- **Unknown face detection** (if an additional person appears).
- **Background object detection**, improving situational awareness.

YOLOv8 ensures reliable object-level monitoring with minimal latency.

2. OpenCV (Open Source Computer Vision Library)

OpenCV is used extensively for facial and behavioral tracking in real time. It processes video frames and extracts face-related features to determine user behavior.

Key Functions:

- **Face Tracking:** Ensures the candidate remains in front of the camera throughout the exam.
- **Eye Tracking:** Detects lateral eye movement to identify if the candidate is looking away.
- **Head Pose Estimation:** Calculates head rotation (yaw, pitch, roll) to detect abnormal head movements.
- **Face Absence Detection:** Triggers alerts when no face is detected.

These techniques help in identifying behavioral anomalies and maintaining exam discipline.

3. TensorFlow.js Models

TensorFlow.js allows lightweight ML models to run directly in the web browser without backend dependency.

Key Functions:

- Additional **browser-based cheating checks**
- Lightweight **face verification** and **presence detection**
- Helps reduce server load and improves real-time performance

This makes the proctoring system more responsive and scalable.

4. Python-Based Detection Pipeline

Python acts as the integration layer for YOLOv8 and OpenCV, processing video frames and generating cheating alerts.

Key Responsibilities:

- Running ML inference
- Handling frame-by-frame analysis
- Communicating results to backend APIs
- Logging detection events

7. Implementation & Workflow

The implementation of the **AI-Based Proctored Examination System** integrates machine learning models, computer vision techniques, and a full-stack web application. The workflow ensures continuous monitoring, secure exam conduct, and efficient log generation.

1. System Initialization

- The student logs into the exam portal using valid credentials.
- Once the exam begins, the webcam is activated and video frames are captured using **React-Webcam**.
- The backend server initializes communication with the AI module running in Python.

2. Real-Time Video Processing

Each captured frame is sent to the **Python-based AI pipeline**, where the following operations occur:

a. Object Detection using YOLOv8

- Detects **mobile phones**, **multiple persons**, and **unknown faces**.
- YOLOv8 inference runs continuously to identify suspicious objects or individuals.

b. Face & Eye Tracking using OpenCV

- Tracks the candidate's face position and ensures they stay within the frame.
- Eye tracking detects if the candidate repeatedly looks left or right.

c. Head Pose Estimation

- Calculates **yaw, pitch, and roll** to detect unnatural or frequent head turns.
- Alerts are generated if the candidate looks away from the screen for too long.

d. Absence Detection

- If no face is detected for a certain duration, a warning or cheating log is generated.

3. Cheating Detection & Alert Generation

When any suspicious activity is detected:

- The event is classified (e.g., *Mobile Phone Detected*, *Multiple Faces*, *Looking Away*).
- The system timestamps the incident.

- The cheating log is sent to the backend via API.

These logs are stored in **MongoDB** for teacher review.

4. Exam Portal Functionality (MERN Stack)

Student Side

- Students can start exams, answer questions, and submit responses.
- Auto-submit is triggered when the timer ends.

Teacher Side

- Teachers can create exams, add questions, and configure rules.
- Real-time alerts and logs are displayed in the teacher dashboard.
- Cheating incidents can be reviewed after the exam.

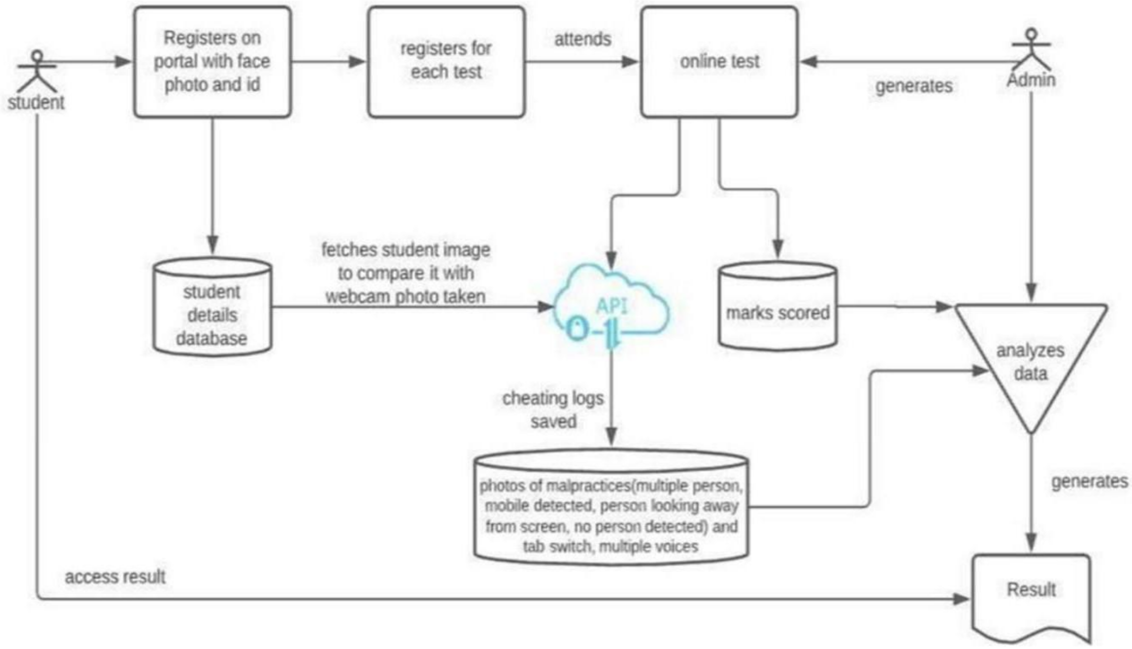
4. Backend Processing

- The backend (Node.js + Express) receives logs from the AI module.
- JWT-based authentication protects routes and user data.
- Mongoose handles structured storage of exam data, user details, and cheating logs in MongoDB.

5. Examination Completion

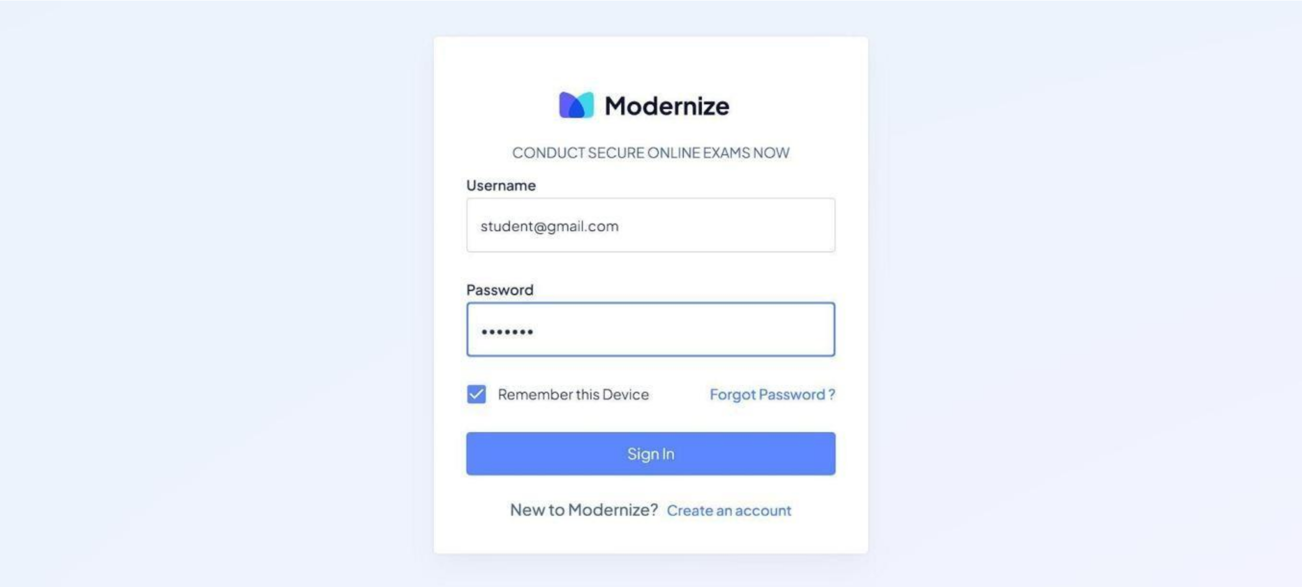
- When the exam ends, all detection activities stop.
- Final cheating summary is generated and made available to teachers.
- Screenshots and logs can be reviewed for verification.

Workflow Diagram


student
Registers on portal with face photo and id
registers for each test
attends
online test
generates
Admin
student details database
fetches student image to compare it with webcam photo taken
API
marks scored
analyzes data
cheating logs saved
photos of malpractices(multiple person, mobile detected, person looking away from screen, no person detected) and tab switch, multiple voices
generates
access result
Result

8. Screenshots of the Project

Login Page


Modernize
CONDUCT SECURE ONLINE EXAMS NOW

Username: student@gmail.com

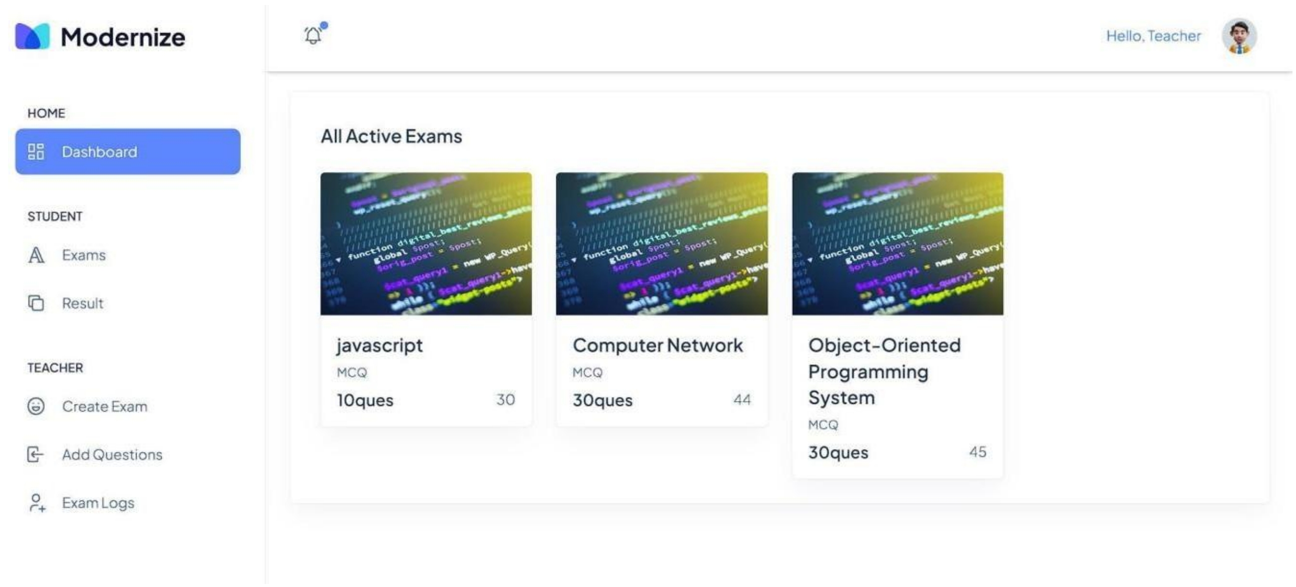
Password: •••••••

☑ Remember this Device

Forgot Password ?
Sign In
New to Modernize? Create an account

Dashboard

- **Student**

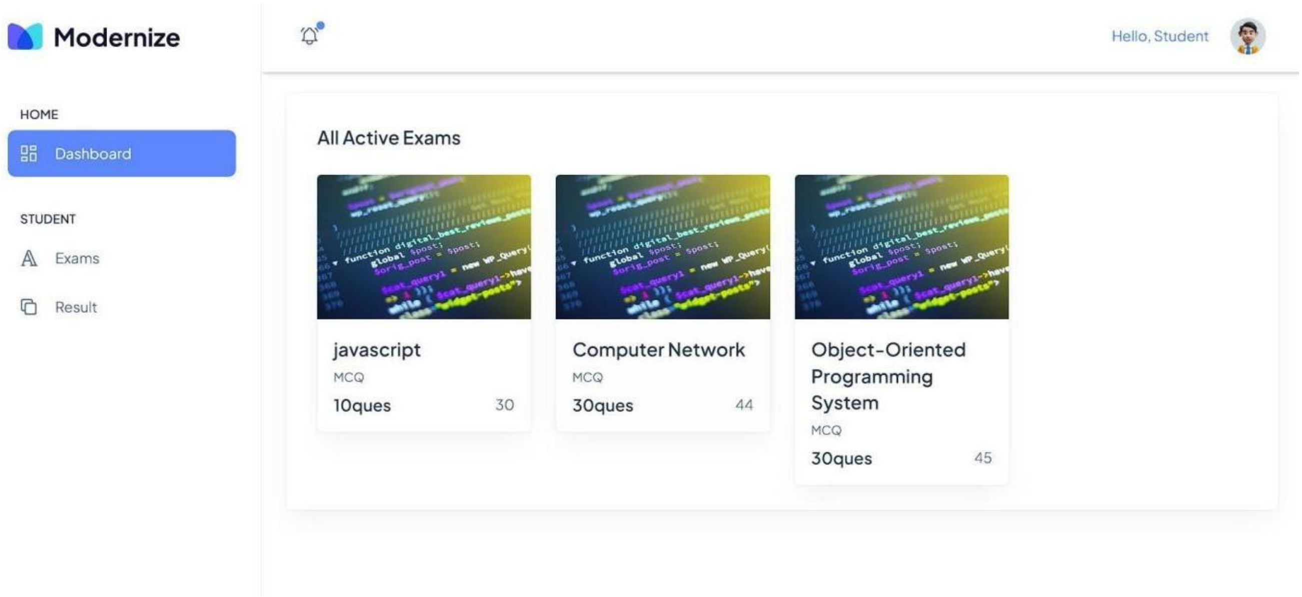


All Active Exams

javascript
MCQ
10ques 30

Computer Network
MCQ
30ques 44

Object-Oriented Programming System
MCQ
30ques 45



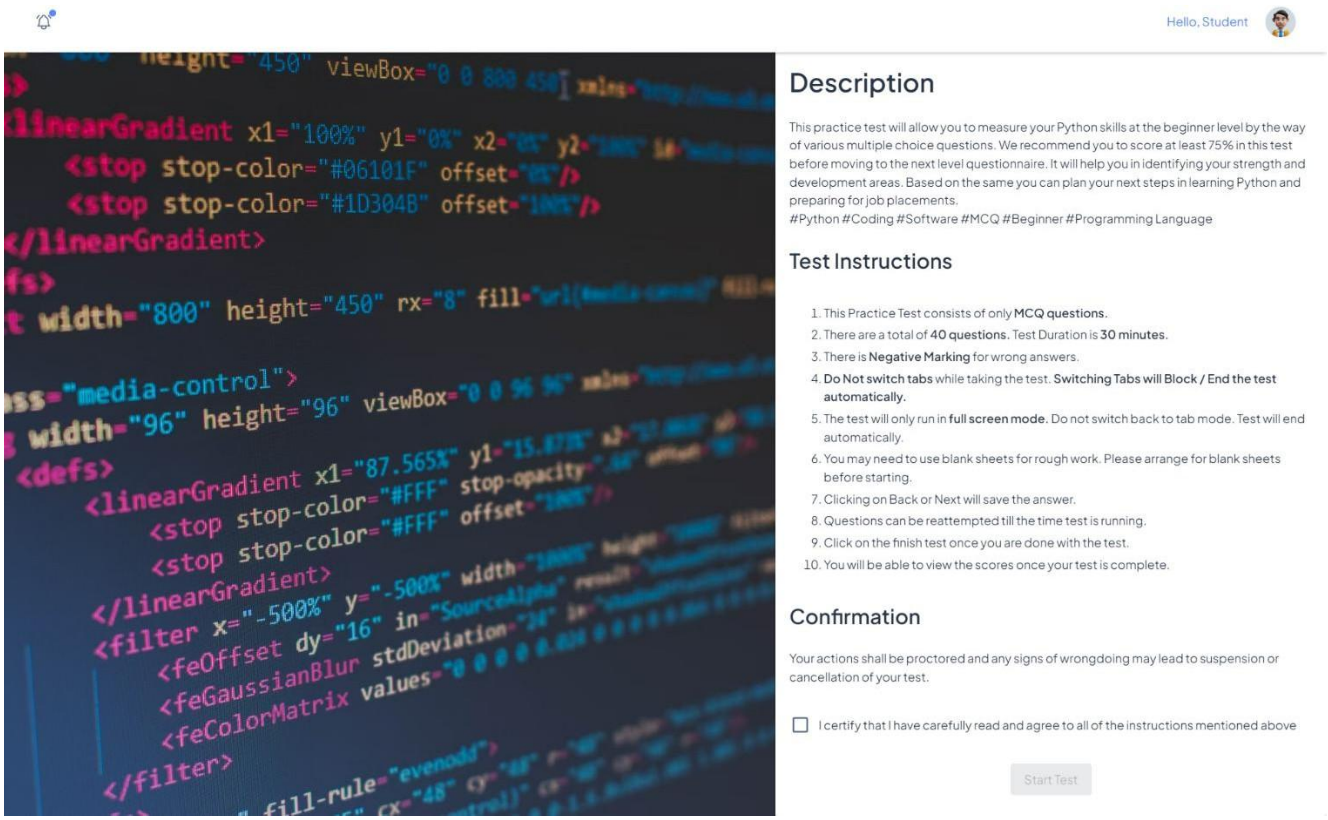
All Active Exams

javascript
MCQ
10ques 30

Computer Network
MCQ
30ques 44

Object-Oriented Programming System
MCQ
30ques 45

Exam Page



Description

This practice test will allow you to measure your Python skills at the beginner level by the way of various multiple choice questions. We recommend you to score at least 75% in this test before moving to the next level questionnaire. It will help you in identifying your strength and development areas. Based on the same you can plan your next steps in learning Python and preparing for job placements.
#Python #Coding #Software #MCQ #Beginner #Programming Language

Test Instructions

1. This Practice Test consists of only **MCQ questions.**
2. There are a total of **40 questions.** Test Duration is **30 minutes.**
3. There is **Negative Marking** for wrong answers.
4. **Do Not switch tabs** while taking the test. **Switching Tabs will Block / End the test automatically.**
5. The test will only run in **full screen mode.** Do not switch back to tab mode. Test will end automatically.
6. You may need to use blank sheets for rough work. Please arrange for blank sheets before starting.
7. Clicking on Back or Next will save the answer.
8. Questions can be reattempted till the time test is running.
9. Click on the finish test once you are done with the test.
10. You will be able to view the scores once your test is complete.

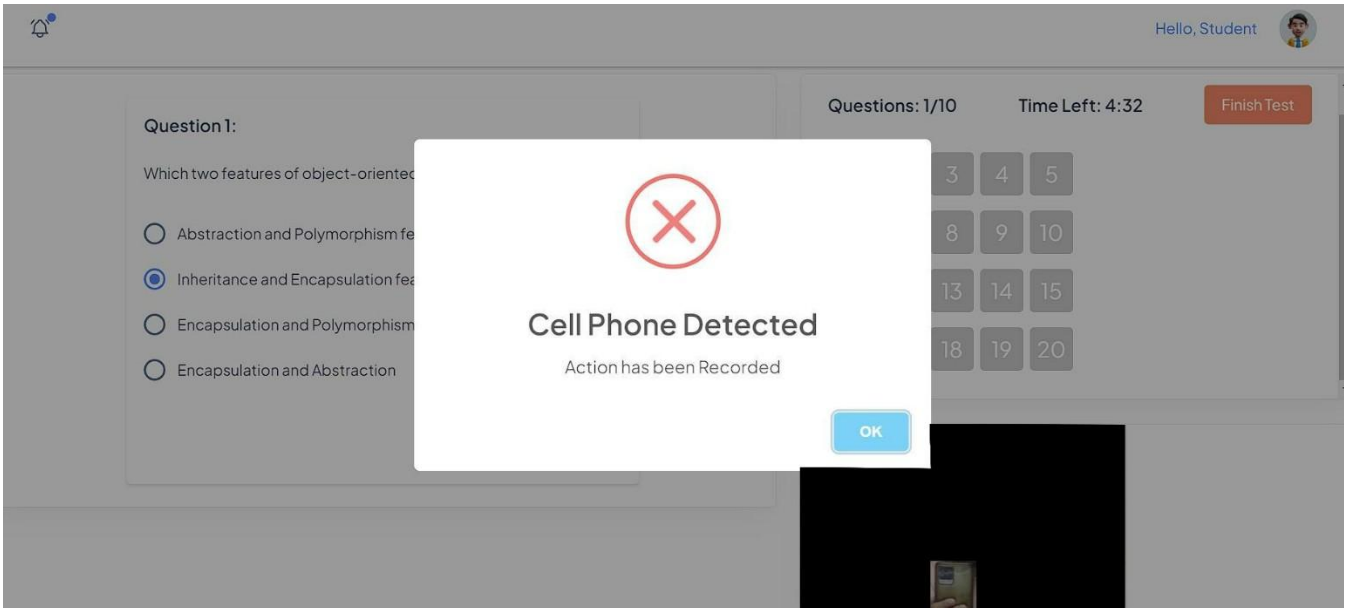
Confirmation

Your actions shall be proctored and any signs of wrongdoing may lead to suspension or cancellation of your test.

□ I certify that I have carefully read and agree to all of the instructions mentioned above

Cheating Detection During Exam

Cell Phone Detection



Question 1:

Which two features of object-oriente

○ Abstraction and Polymorphism fe

● Inheritance and Encapsulation fea

○ Encapsulation and Polymorphism

○ Encapsulation and Abstraction

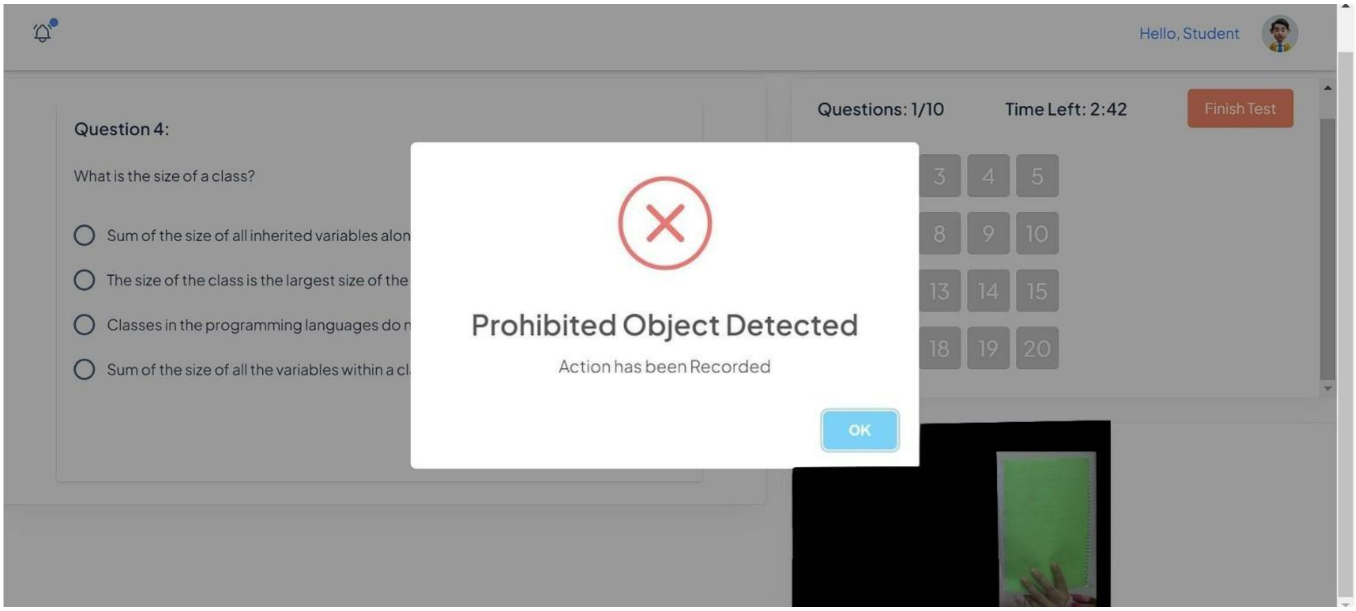
Questions: 1/10

Time Left: 4:32

Cell Phone Detected

Action has been Recorded

Prohibited Object Detection



Question 4:

What is the size of a class?

○ Sum of the size of all inherited variables alon

○ The size of the class is the largest size of the

○ Classes in the programming languages do n

○ Sum of the size of all the variables within a cl

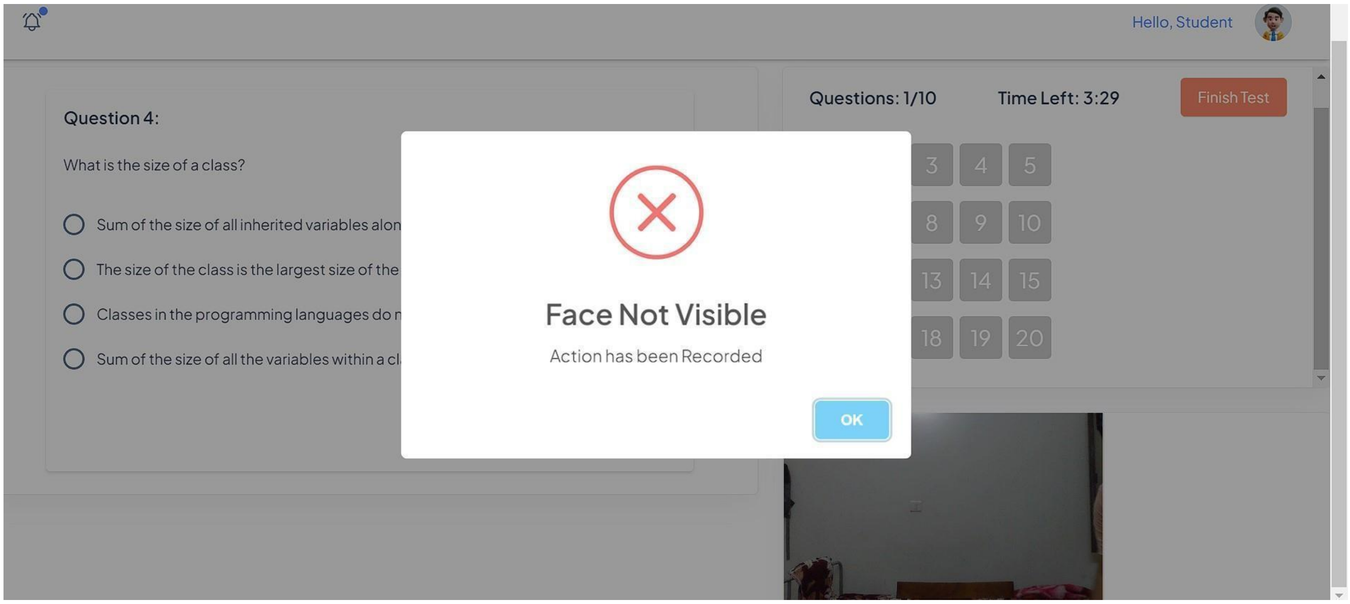
Questions: 1/10

Time Left: 2:42

Prohibited Object Detected

Action has been Recorded

Face Not Visible Detection



Question 4:

What is the size of a class?

○ Sum of the size of all inherited variables alon

○ The size of the class is the largest size of the

○ Classes in the programming languages do n

○ Sum of the size of all the variables within a cl

Questions: 1/10

Time Left: 3:29

Face Not Visible

Action has been Recorded

9. Conclusion & Future Scope

The AI-Based Proctored Examination System provides a secure, automated, and efficient solution for conducting online exams. By integrating **YOLOv8**, **OpenCV**, and **TensorFlow.js**, the system effectively detects cheating behaviors such as mobile phone usage, multiple persons, and suspicious head or eye movements. The MERN-based portal ensures smooth handling of exam creation, student interaction, and teacher monitoring.

Overall, the system:

- Enhances exam integrity through real-time AI-based monitoring
- Reduces dependency on manual invigilation
- Provides accurate and consistent detection of suspicious activities
- Offers a scalable and user-friendly platform for institutions

This project demonstrates how artificial intelligence and modern web technologies can work together to address security challenges in online education.

Future Scope

1. **Identity Verification:**
   Implement face recognition to match the candidate with their registered profile before and during the exam.
2. **Voice Recognition & Audio Analysis:**
   Detect background voices, whispers, and audio anomalies to identify verbal cheating attempts.
3. **Secure Exam Environment:**
   Restrict or block unauthorized applications on students' devices during exams.
4. **Unified Examination Portal:**
   Create a single platform for login, exam access, chat communication, and answer sheet uploads through an integrated scanner.
5. **Advanced Behavioral Analysis:**
   Introduce emotion detection, lip-sync analysis, and deeper behavioral patterns to strengthen the proctoring system.
6. **Cloud Deployment:**
   Deploy the system on cloud platforms for large-scale usage across institutions.