Online Proctoring System

Done by:

Abdulla Alnasser 21002682

Abdulrahman Aljammaa 21003117

Alghaith Alshareif 21003341

Mahra Almazzmi 21002612

Mohammad Kalbat 21002272

Nadia Alhashemi 21000667

System Design and Practical Project

Course instructors:

Abdallah Sherief

Hend ElMohandes

Table of Contents

Acknowledgement

We would like to express our deepest gratitude to Dr. Abdulla Alnasser, a computer scientist, for his valuable guidance provided throughout the project. His knowledge and perception were crucial in resolving our questions and giving us insightful suggestions. His mentoring was a major factor in our project's success.

We also want to express our deep gratitude to Murshid, a talented web developer, for insisting on learning about Spring Boot. His advice turned out to be a game-changer, streamlining numerous chores and enabling our code to function smoothly. His insightful advice had a big impact on our project's effectiveness and overall outcome.

Dr. Abdulla Alnasser and Murshid's knowledge, assistance, and encouragement are greatly appreciated. Their contributions were essential in shaping our project and enhancing its outcomes.

Introduction

2.1 Background and motivation

As a result of the global pandemic, the majority of educational institutions in today's society have been driven to adopt a model of education that is delivered online. A wide variety of colleges and universities started providing online instruction and examinations for their students. Also, the COVID-19 Epidemic had an effect on entrance examinations and recruitment processes, both of which use written tests to choose people for positions. In this setting, academic dishonesty is on the rise, whether it manifests itself as plagiarism or as cheating during tests. Students often have several ways of cheating in online exams, necessitating the implementation of some kind of online proctoring system to track the students' activities during the exam period. Even with a large number of checks already in place, such as the one-of-a-kind method that has been proposed to electronically invigilate students during examinations held in remote locations, monitoring online examinations can be difficult because of the possibility of a high incidence of exam fraud.

The prevalence of e-learning courses has led to an increase in the possibility that students may cheat on exams in a variety of different methods. These include multi-window surfing, asking classmates for answers, and even bringing dishonest materials into the testing environment. There is a wide variety of proctoring software available to aid educators in the administration of online examinations. The standard requirements for being able to take exams from any location are, in fact, a computer equipped with a camera and an active internet connection. In any case, the only way they guarantee the test's honesty is by accrediting the professionals who administer it. Yet, they are still putting their faith in the human exam monitoring system. As a consequence of this, automating the monitoring process in such a way that it keeps its reliability while also retaining a low cost is a challenging challenge that is tackled in this study.

We developed a model that performs three different functions to reduce the prevalence of plagiarism among students. Firstly, the system detects the examinees' faces during examination to prevent them from cheating. Secondly, it detects the presence of a cell phone, book or any other person that can be used for malpractice. Thirdly, it identifies the examinee's head posture and jaw movement. Lastly, it detects if there is more than one face present in the video.

2.2 Problem Definition

The COVID-19 pandemic has forced many educational institutions to move their courses and exams online. This has led to an increase in the number of online exams, which are more susceptible to cheating than in-person exams (Duhaim et al., 2021). Students can cheat in exams through several means. Students cheat through the use of books, notes or cheat sheets. They may also communicate with other during the examination period to help them with the questions. Furthermore, they may also use electronic devices to access unauthorised materials for closed exams. Another method of cheating in exam is through impersonation of others to take exams on their behalf. Therefore, developing strategies for addressing or reducing the prevalence of cheating in online exam is a concern in educational institutions.

Cheating in online exams can have significant consequences on students, instructors, and the administration. It can result in the devaluation of the academic achievement of students who do not cheat in exam (Baran & Jonason, 2021). It can also create unfair advantage for other students within the educational institutions. Moreover, it can undermine the educational system's academic integrity. These consequences may affect the educational institutions and undermine the objectives of the educational systems and the courses provided. Although cheating is a prevalent issue among students, this model can help solve these problems are reduce the cases of suspicious activities. This model has proven to be effective in detecting cheating in online exam, making exam fair and secure.

Project methodology

3.1 Project Analysis, Requirements Specifications

3.1.1 Hardware and Software Used

The hardware requirements for this project will vary depending on the number of users. The requirements include:

- A computer with a powerful processor and high memory. This is necessary to run the computer vision algorithms that will be used to detect cheating.
- A webcam that can capture high-quality video of the exam taker. This is necessary to track the exam taker's movements and facial expressions.
- A high-speed internet connection. This is necessary to stream the video from the exam taker's computer to the server.
- In addition to these general requirements, there may be other specific requirements depending on the specific needs of the application. For example, if the application will be used to detect cheating in a large-scale exam, then it may be necessary to use a cloud-based server to handle the large volume of data.

Hardware Requirements

- Computer: The computer should have a powerful processor, such as an Intel Core i7 or AMD Ryzen 7. It should also have at least 16GB of memory and a fast hard drive or solid-state drive.
- Webcam: The webcam should be able to capture high-quality video at a resolution of at least 1080p. It should also have a wide field of view so that it can capture the entire exam taker's face.
- Internet connection: The internet connection should be high-speed so that the video can be streamed to the server without any lag.

Software Requirements

The system has various software requirements for development. It will need to run on various operating systems, such as Windows, macOS, and Linux. The programming language used for the program development is java, html, JavaScript, and CSS. The databases used are MySQL and PostgreSQL. Besides, it will be hosted in a web server, such as Apache or Nginx. In addition to these general requirements, there may be other specific requirements depending on the specific needs of the application. For example, if the application will be used to detect cheating in a large-scale exam, then it may be necessary to use a cloud-based server to handle the large volume of data.

3.1.2 Requirement specification

An instructor for the project would be able to:

1. Determine whether more than one face is present in the video recording.
2. If the instructors find any items that students can use for the cheating, they can take appropriate actions.
3. Observe the position of the person's head to determine the direction in which they are looking.

Recognizing a Person's Face

The challenging problem of computer vision known as face detection involves identifying and localizing persons in pictures. Face detection is part of the field of computer vision. According to Roa’a et al. (2022), a combination of body movement can effectively produce desired results for verifying individuals, particularly when taking exams. The previous systems included face detection through the use of Dlib’s frontal face Histogram Oriented Gradients (HOG). Despite this, it did not result in outcomes that were satisfying. Comparisons were made between various face recognition models, such as Haar, Dlib, the

Multi-task Cascaded Convolutional Neural Network (MTCNN), and OpenCV's DNN module.

The DNN module that may be found in OpenCV generates output of the highest possible quality. Face detection is the foundation of tracking the movement or the expressions of individuals during the exam, including head position and movements.

You can utilize an additional quantized model for the face detector by running the get face detector method and setting the option quantized to the value Yes. This system will allow an individual to use the face detector. By changing the value of the quantized option to true, you can use this model. In our system, the traditional implementation of the face detector delivered 17.5 frames per second, whereas the quantized implementation delivered 19.5 frames per second. Since it is uint8 quantized, this would be extremely handy when distributing on edge devices.

Detection of People and Cell Phones

We implemented the system using pre-learned weights of YOLOv3 trained on the dataset to distinguish individuals in the exam from mobile phones and webcam feed. This allowed us to do so more accurately. We counted the number of people on a webcam by using a pre-trained version of the YOLOv3 algorithm. In the event that the count is not equal to one, an alert may be triggered. To identify mobile phones in the COCO dataset, it is necessary to compare the class indexes to the index value of 67. Prior to reporting a mobile phone, a thorough evaluation should be conducted to determine if any class indexes closely match this number. Meanwhile, we may utilize another thread to complete the YOLOv3 duties of counting persons and detecting motion. Additionally, YOLOv3 is used for streaming the video feeds from the webcam for detecting individuals during exam. After that, the classifications of the things that have been observed are analysed, and appropriate action is taken if more than one person or a mobile phone is found to have been observed.

Identification of Facial Landmarks

The process of locating and keeping track of important landmarks on a person's face is referred to as facial landmark detection. Earlier, Dlib's model was utilized, however due to the fact that it performs poorly when the face is angled, it was retired. As a consequence of this, the model that has been proposed for identifying landmarks is founded on a convolutional neural network that was created using tensorflow. It can identify when a person is moving their head, monitor their faces, and estimate their head position. In order to accomplish face detection, the Caffe model that is included in OpenCV's DNN module was utilized. For the purpose of facial landmark detection, we made use of a pre-trained model from Dlib. This provides 68 different facial landmarks. There are six facial Landmark points that were used, and they are as follows: the tip of the nose, the chin, the extreme left and right points of the lips, as well as the left corner of the left eye and the right corner of the right eye. The rotational and translational vectors near the tip of the nose were computed, and the conventional 3D coordinates of these face landmarks were used. After acquiring the necessary vectors, those three-dimensional dots were preserved on a two-dimensional surface, which is our image.

Head Pose Estimate

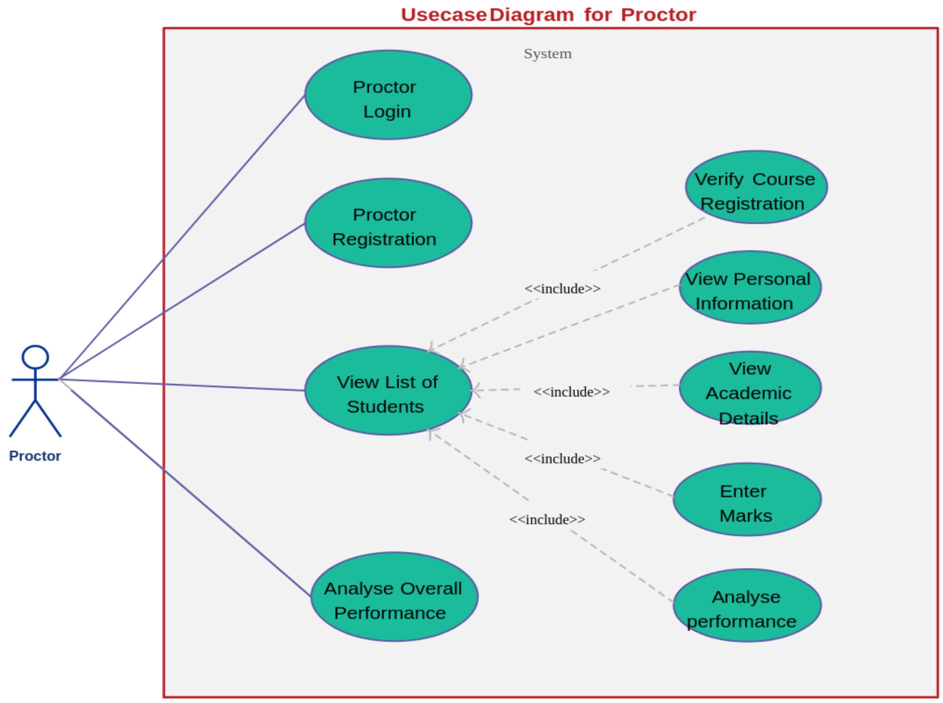
The estimate of the head's position is carried out in order to ascertain the direction in which it is facing. Estimating a person's head position is a tough topic in the field of computer vision since the solution requires a number of different processes. First, we have to find the face in the frame, and then we can move on to identifying all of the facial indicators.

In today's society, it may seem like a straightforward process to recognize a person's face, and this is especially the case when the individual is looking directly into the camera. The challenge arises if the face is angled in any direction. To this, you must also consider the fact that certain facial landmarks become covered when the head is moved. After that, we need to transform the points to 3D coordinates so that we can calculate the inclination of the surface. It is necessary to obtain measurements from six spots on the face, specifically the tip of the nose, the chin, the extreme left and right points of the lips, the left corner of the left eye, and the right corner of the right eye.

3.1.3 Use Case Diagrams

The system has four main actors:

- Examinee. The individual who is in charge of taking the test is referred to as the examinee.
- The person who is in charge of supervising the examination is referred to as the invigilator.
- The person who is in charge of organizing and supervising the test is called the administrator, and it is their job to ensure that everything goes smoothly.
- System: The term "system" refers to the piece of software that is utilized in the process of delivering the examination.



3.2 Project Design

3.2.1 Project Design Perspective

Proctoring systems are used extensively in online learning environments to monitor students while they are taking exams and participating in other online activities. This helps to prevent students from engaging in dishonest academic behavior. Within the framework of this design approach, we will discuss the critical components and elements to consider while building a proctoring system to ensure academic honesty.

The system includes a program that runs on a client computer, software on a server computer, or a database that holds all relevant data. Client-side software is often placed on the student's computer and is in charge of capturing video. The video is received by the server's software, which then processes them through the implemented algorithms to detect any head movement or the presence of a mobile phone. The software operating on the server also allows instructors and administrators to create reports and see recorded data. The database is where all of the relevant data, such as the specifics of the students and the reports.

Data protection, scalability, and dependability are three of the most important factors to take into account while developing a proctoring system. Because it involves the collecting and storing of information, such as the name of the student, date and time of the incident, and the incident, data privacy is a critical concern for any proctoring system. This is because proctoring entails the information of students. The system must comply with stringent privacy rules and regulations, and data encryption ought to be utilized in order to safeguard the data both while it is being transmitted and while it is being stored.

Another significant consideration when creating proctoring systems is scalability, as these systems may have to support a high number of users during peak hours. A scalable system should meet the performance objectives when the workload increases or environment changes (Al-Said Ahmad & Andras, 2019). The system should be able to manage the demand while also providing real-time statistics and notifications to administrators and instructors who use it. A distributed architecture can boost scalability by allowing many servers to manage the demand while also assuring redundancy and fault tolerance. Therefore, a scalable system is implementable in various settings to enable the system work.

Another critical thing to consider is reliability, because the proctoring system must run smoothly and without interruptions while the students take their tests. Software reliabilities is the ability of the ability of the program to complete defined functions as expected as per the user requirements (Xi et al., 2022; Shanthi et al., 2021). The system should be highly available, with features like load balancing, redundant servers, and automated failover protocols. In addition, the system should be inspected and updated on a regular basis to ensure that it is running appropriately and efficiently. Based on the user requirements, the system is reliable because it operates based on the features necessary.

3.2.2 Process

The project design comprises a series of steps that must be completed in the proper order in order to deliver a dependable proctoring system. It is critical to include stakeholders (such as students, instructors, and administrators) at all phases of a project's design process to ensure that the final result meets their needs and exceeds their expectations. Some of the best practices for software engineering that should be followed include using version control, documenting the code, and sticking to a thorough testing strategy. It is also critical to adhere to best practices for software engineering.

The steps for the process include:

- The first step is to identify the issue at hand, as the fundamental goal of the proctoring system is to address a specific issue. The issue in this scenario is to prevent students from cheating during online assessments.
- The next step is to identify the demands of the proctoring system, which is known as the "identifying requirements" procedure. This involves establishing the system's qualities and capabilities that it must have in order to address the problem.
- Making a high-level design After determining the criteria for the proctoring system, it is feasible to create a high-level design for the system. This involves describing the system's architecture as well as how the various pieces of the system will communicate with one another.
- Following the development of the high-level concept, a comprehensive design of the proctoring system may be created. This may be done once the high-level design is completed. This phase includes specifying how the system will be built, including the algorithms and data structures that will be used.

- Implementation: Once the system's exact design is complete, the next stage is to put it into action. This procedure also includes writing the system's code and merging its many components.
- Testing: After the system has been installed, it must be tested to ensure that it performs as intended. Functional testing compares the system to the requirements, whereas non-functional testing investigates the system's scalability, and performance.
- After the deployment phase is done, the system may be deployed to production after it has been tested and proven to perform as intended.
- Maintenance: As a last stage, the system must undergo regular maintenance to ensure that it continues to work as intended. This involves fixing bugs, adding new features, and updating the program as needed.

Activities for the Development Process

Week	Activity
1.	Conduct initial research on similar proctoring systems and their features, Define system requirements
2.	Develop wireframes and mockups for the UI/UX design, Develop the front-end interface using HTML/CSS/JavaScript
3.	Implement basic video streaming using WebRTC Implement the pose detection algorithm and integrate it with the video stream
4.	Develop the back-end using Java and Spring Framework Implement user authentication and authorization using Spring Security

5.	Develop the database schema and implement data access using Spring Data JPA
6.	Implement RESTful API endpoints for CRUD operations on student and incident data
7.	Conduct integration and system testing Conduct user acceptance testing and address any bugs or issues Conduct performance testing and optimization
8.	Prepare for deployment and production release
9.	Deploy the system to a production environment and conduct post-deployment testing
10.	Plan for future enhancements

3.2.4 Roles and activities of team

Alghaith Alshareif:

Alghaith participated by gathering information on necessary libraries and technology, laying the groundwork for the study. He developed the StudentController class and collaborated with Mohammad Kalbat to create the student.html class. Additionally, Alghaith contributed to the project implementation and requirement specification.

Mohammad Kalbat:

Mohammad’s contributions were in creating the AdminController class and collaborating with Alghaith Alshareif to develop the student.html class. These classes were critical to the project’s success as they allowed for effective administration and interaction within the program. In the report, Mohammad took charge of various aspects, including delivering through introduction, describing the project’s hardware and software infrastructure. Mohammad also worked with Abdulla Alnasser for project testing and assessment.

Abdulla Alnasser:

Abdulla Alnasser contributed in both the code and the report. He was responsible for creating the incidenceDaoService and incidenceDetails classes, which managed and processed incident-related data within the program. In the report, Abdulla provided a clear succinct problem statement, emphasizing the specific issue the project aimed to address. He also documented the process, outlining the necessary actions to achieve the desired results. Abdulla worked closely with Mohammad Kalbat on project testing and evaluation, ensuring the final product’s functionality and efficacy.

Abdulrahman Aljamaa:

Abdulrahman made contributions to both the coding and the report. He enhanced the application's functionality and user experience by implementing the StudentDetails method and creating the admin.html page. In the report, Abdulrahman addressed the project's design viewpoint and developed the project diagram, providing visual representations of system interactions. He collaborated with Nadia Alhashemi to provide background information and inspiration for the project.

Nadia Alhashemi:

Nadia played helped in improving the application's functionality and database connectivity. She added the student method, StudentDaoService class, and DatabaseConfig component to the codebase. In the report, Nadia contributed valuable insights into the project's history, motivation, testing, assessment, user input, and financial considerations. Her efforts contributed to the successful implementation and documentation of the project.

Mahra Almaazmi:

Mahra's contributions to the project included the development of the student-Details and Incidence classes, enhancing the application's functionality. In the report, Mahra generated the Table of Activities and made contributions to the Gantt Chart, providing visual representations of system interactions, project activities, and the project timeline. Additionally, Mahra contributed to the limitation section of the report.

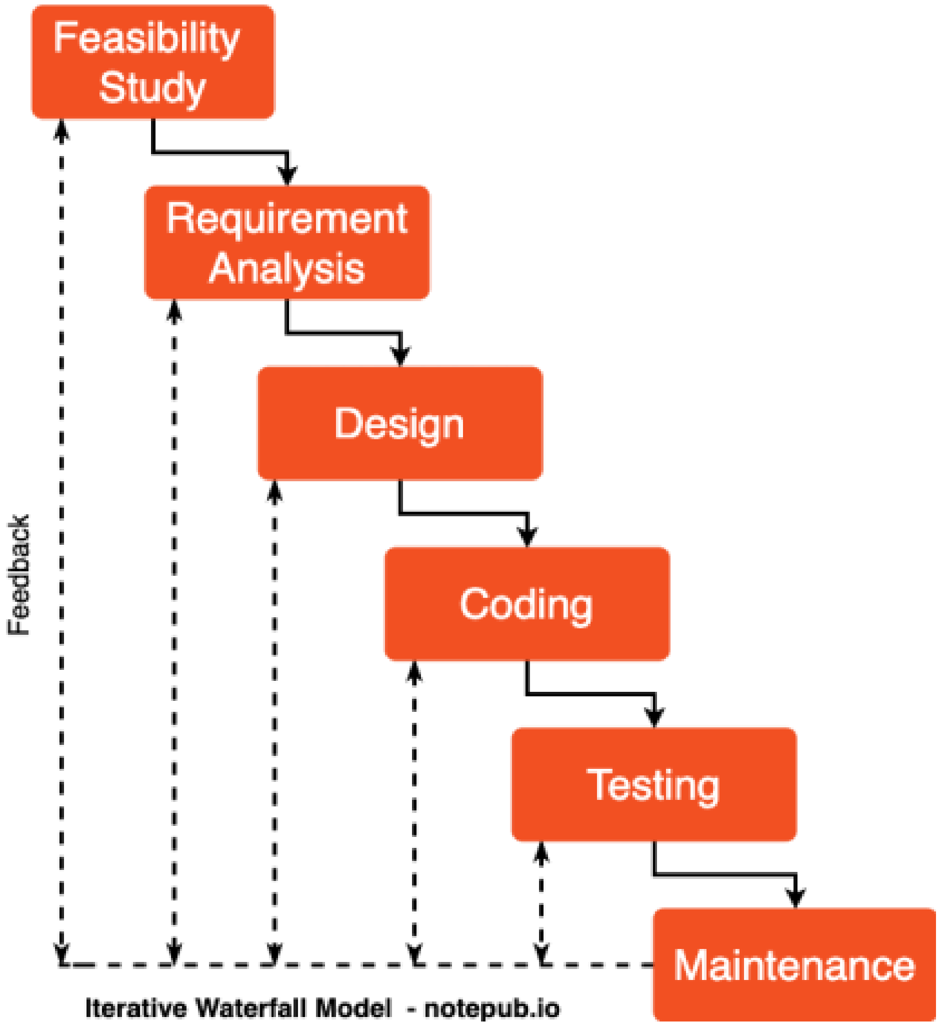
Everyone:

All team members actively shared their project learnings and ideas. The report's depth and insight showed the team's progress and new understanding. We learned how to cooperate and work as a team, which is essential to our future work environments. Each team member also meticulously cited sources and materials. This effort showed our dedication to academic practice and ethical project documentation. Continuous learning and strict referencing helped the team implement the project successfully.

3.2.5 Gantt Chart

Task	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7
Project Planning	X						
Research and Analysis	X	X					
Design		X	X	X			
Development			X	X	X		
Testing and QA					X	X	
Deployment						X	X
Documentation				X	X	X	X

3.2.6 Project diagram



3.2.7 Budget

Item	Estimated Cost
Personnel	$50,000
Software development	$80,000
Hardware	$20,000
Testing	$10,000

Marketing	$5,000
Contingency	$15,000
Total	$180,000

3.3 Project Implementation

A wide range of technologies were used to construct our online proctoring system, each one chosen for its unique advantages and contributions to the functionality of the system as a whole. HTML, CSS, JavaScript, SQL, and Java were the programming languages utilised to create the project.

Backend

Java serves as the backbone of our application and is utilised to build the server-side functionality and communicate with our SQLite database. We implemented inversion of control and dependency injection using Spring Boot, a solid framework that makes it easier to set up standalone Spring apps. Additionally, Spring Boot provides a number of practical features like an embedded server, automatic configuration, and the powerful JdbcTemplate of Spring to streamline database interfaces.

Additionally, SQLite was chosen as the database due to its straightforwardness, lightweight, and user-friendliness. SQLite is portable, serverless, and self-contained, thus meeting our needs, and requiring no configuration or set up.

Moreover, The application relies on a Model-View-Controller (MVC) architecture to manage its data, user interfaces, and control flow. The models represent the data structure of the application and communicate with the database using JdbcTemplate from Spring. The data models take care of storing and retrieving student data as well as handling the specifics of proctoring incidents. While the views are used to render the data, the controllers handle the user interactions and link the models and views together.

Furthermore, Java was used to model our data access objects (DAO) and Data Transfer Object (DTO). The DAO classes are in control of interacting with the student and incidence database through the JDBC (Java Database Connectivity) for connectivity and to perform operations. On the other hand, the DTO classes act as a conduit for transferring data between the web layer and the service layer.

Lastly, java was also used in the development of controller classes, which are fundamental classes on the server-side. They are responsible for handling HTTP requests related to both student and admin such as accessing the database, fetching data, updating data, and receiving incident reports.

Frontend

On the front end, was employed HTML and CSS with responsive styling provided by Bootstrap. JavaScript, along with TensorFlow.js, were used to provide dynamic functionality on the client-side, such as accessing the user's webcam and performing pose and object detection.

Furthermore, The proctoring system uses TensorFlows PoseNet and coco-ssd models to detect the student's pose and the presence of objects (cell phones) in the webcam feed.

In addition, Thymeleaf was incorporated as a server-side Java template engine to dynamically generate HTML views on the server side. Allowing data from the server to be inserted into these views before they are sent to the client.

Error Handling

For error handling in the backend, Spring’s BindingResult class is utilitzed. BindingResult class offers methods to reject values, get all errors, and check if there were any errors during the binding process. In case of any errors arising during the POST request handling, the errors are returned as a JSON (JavaScript Object Notation) response.

Data Flow

The system initiates by executing the Main class, which launches the Spring Boot application. Subsequently, the user can then access the system through the local host on the web browser. The user can navigate to different pages in the application by visiting the different URLs. When a student visits the proctoring page, JavaScript code accesses their webcam and starts detecting their pose and presence of cell phones. If any incidents occur, the JavaScript code sends this information back to the server by making a POST request to an endpoint.

During the development of the code, our aim was to adhere to best practices in terms of code organization, modularization, and commenting. With this approach, we have ensured that the system is extendable and maintained.

3.4 Project testing and evaluation

A significant phase in the development process was testing an online proctoring system for tests. We started by gathering requirements to verify that the system matches our stakeholders' demands. We then developed a detailed test strategy to run numerous scenarios and assess the system's performance against the specified criteria. We reported any flaws or difficulties detected during the testing process to the development team, who worked to remedy them and release new versions of the program. We also ran regression tests to make sure the fixes didn't add any new flaws or issues into the system. We conducted acceptance testing after extensive testing to confirm that the system satisfies the specified criteria and is ready for usage. We can guarantee that our online proctoring system works effectively, satisfies the demands of our stakeholders, and delivers a great user experience for students and teachers alike by following this rigorous testing approach.

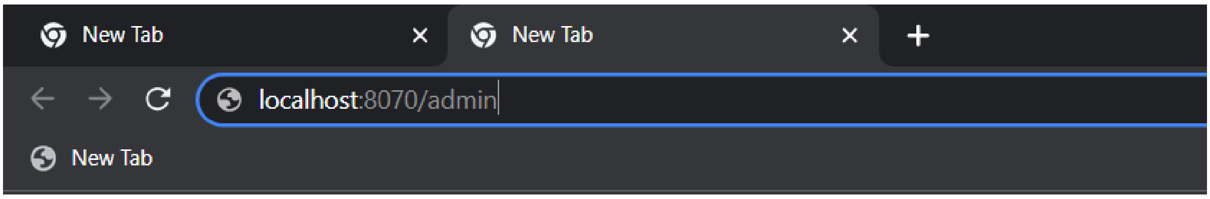
Procedures for Testing

Testing the proctoring system involved multiple stages, including unit testing, integration testing, and system testing. Unit testing verified the functionality of the system's separate parts, such as procedures and functions, conducted by developers. Integration testing verified that the system components could work together, carried out by the development team. System testing included testing the entirety of the system, verifying its operation, performance, and security, carried out by a dedicated testing team. Through a rigorous testing process, we were able to ensure that the proctoring system is reliable, efficient, and secure, addressing any issues found during the testing process. Overall, the testing process for the proctoring system played a critical role in identifying and addressing potential issues and ensuring that the system meets the desired standards.

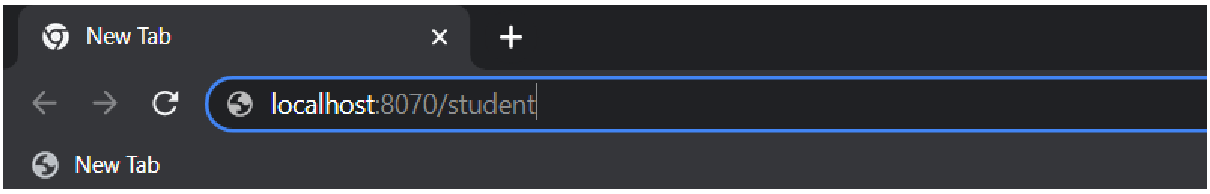
Evaluation Process

The method of evaluating the proctoring system consisted of numerous steps, including testing for both performance and user approval. The usability of the system as well as its functionality were put through their paces during user acceptance testing. The testing was done by a group of users who were not involved in the process of development in any way. The users were provided with a set of activities to complete on the system, and the feedback they provided was used to make changes to the system's usability as well as its functionality. Testing for performance consisted of evaluating how well the system functioned under a variety of stress settings. A dedicated testing team was responsible for carrying out these tests, which included testing the response time, throughput, and scalability of the system. The outcomes of these tests were analyzed in order to determine how the system's performance may be enhanced.

To access admin page and student details page

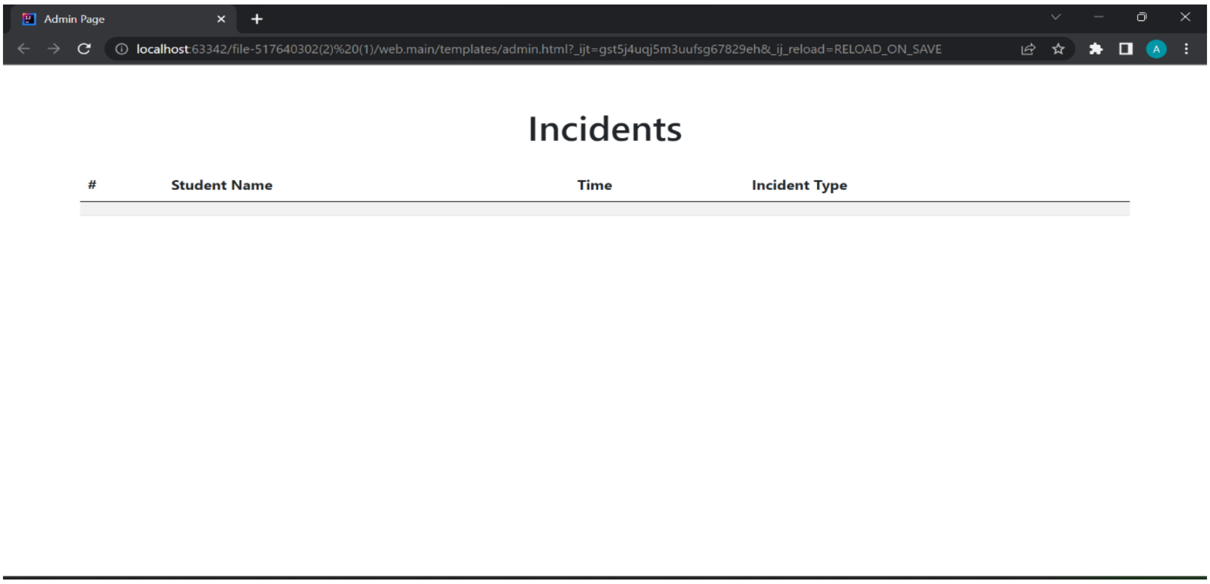


localhost:8070/admin



localhost:8070/student

Admin page

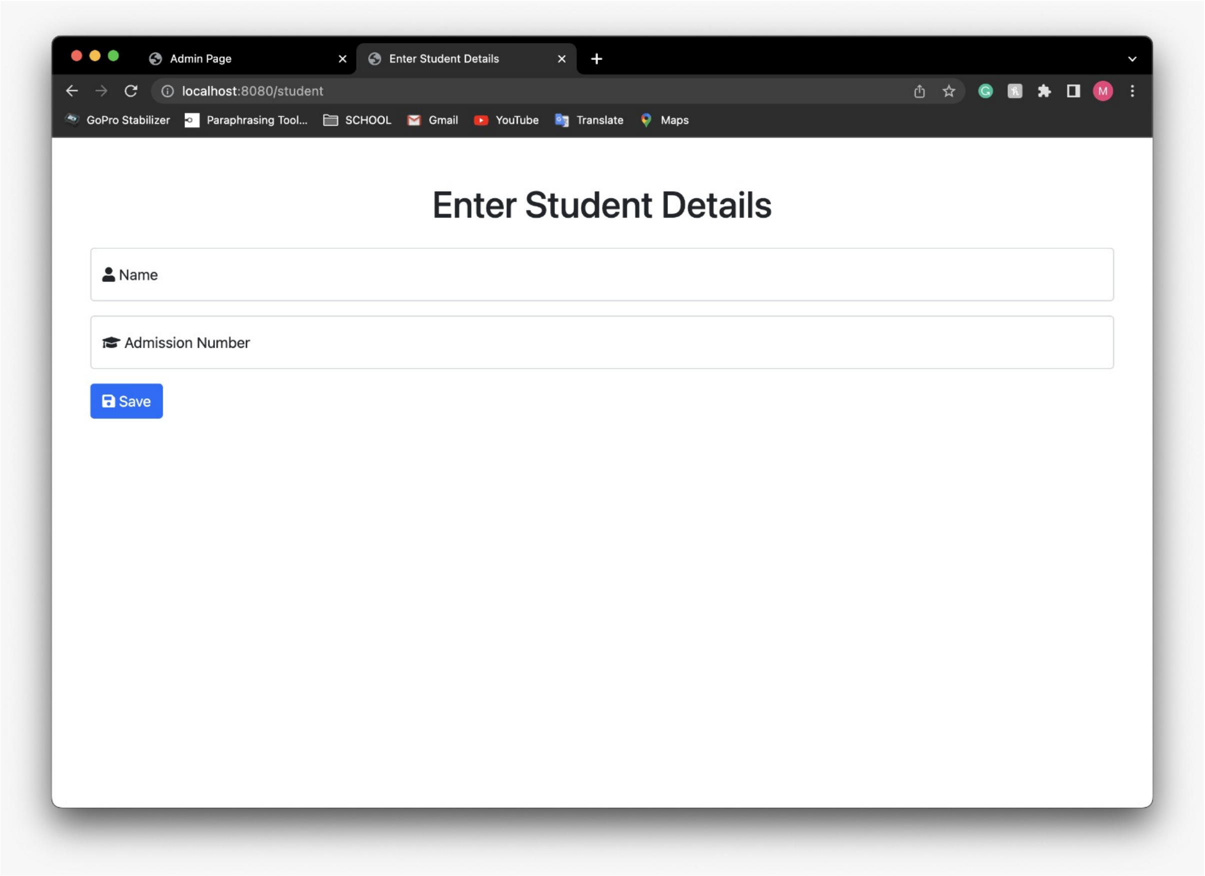


localhost:63342/file-517640302(2)%20(1)/web.main/templates/admin.html?_ijt=gst5j4uqj5m3uufsg67829eh&_ij_reload=RELOAD_ON_SAVE

Incidents

#	Student Name	Time	Incident Type

Student details page



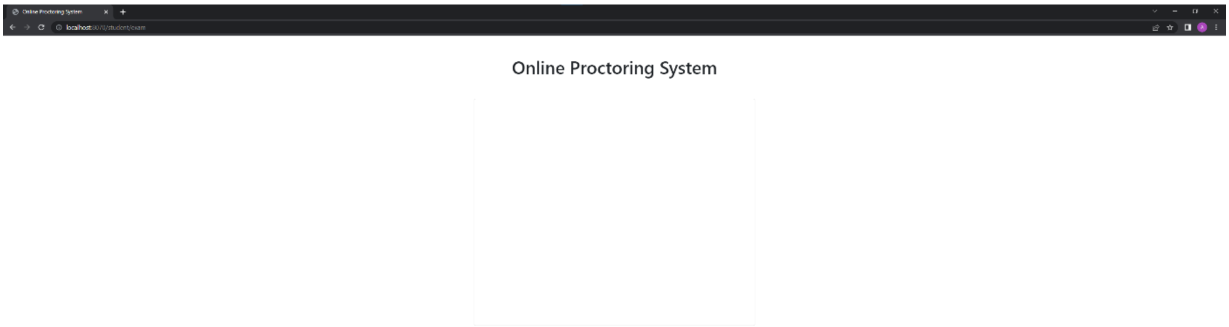
localhost:8080/student

Enter Student Details

Name

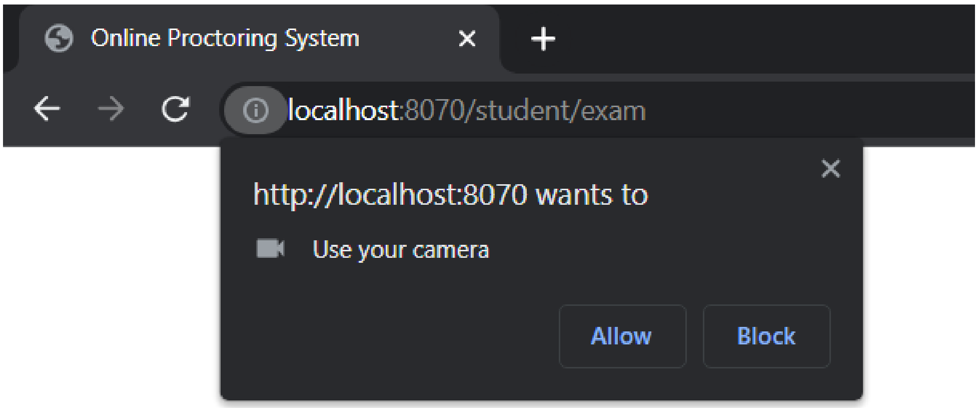
Admission Number

Proctoring page before Camera Access



Online Proctoring System

Asking for Camera Access



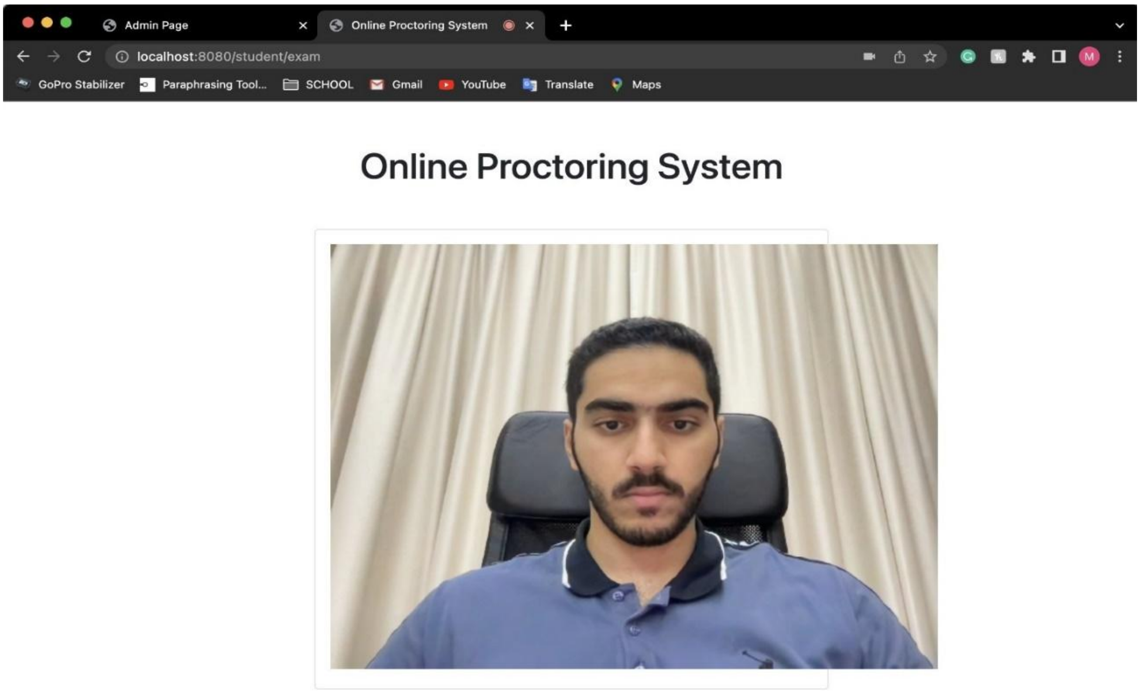
Online Proctoring System

localhost:8070/student/exam

http://localhost:8070 wants to

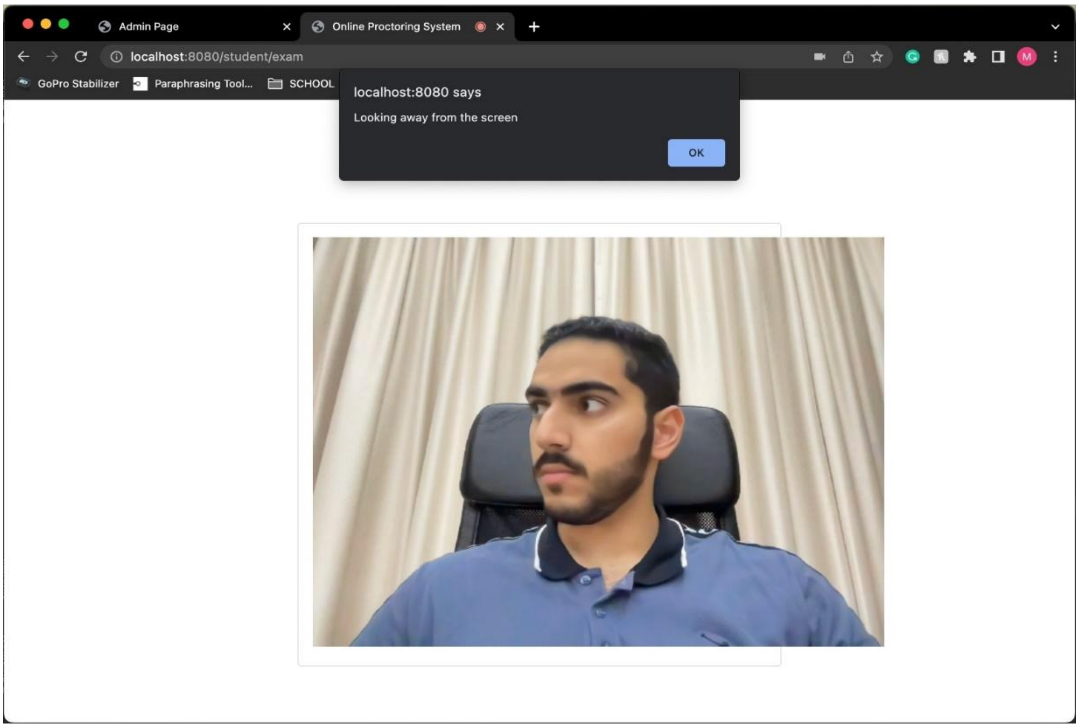
Use your camera

Proctoring page After Access



Online Proctoring System

Looking away from the screen

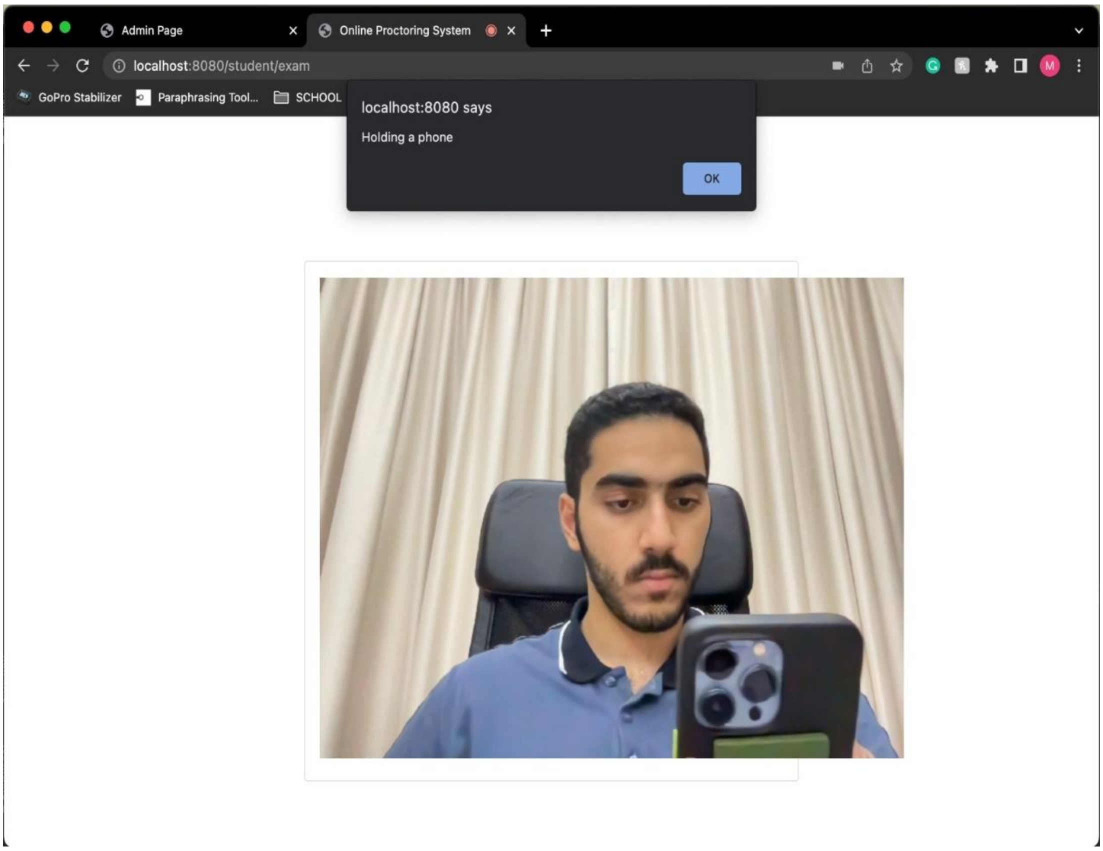


localhost:8080 says

Looking away from the screen

OK

Holding a Cell Phone



localhost:8080 says

Holding a phone

Evaluation Criteria and Metrics

In order to conduct an accurate assessment of the proctoring system, several measures were utilized, including the following:

- Usability is a metric that was utilized in order to evaluate the user-friendliness of the system from the user's point of view. The System Usability Scale (SUS), which is a questionnaire-based tool that is commonly used for evaluating the usability of software systems, was the instrument that was utilized to measure it.
- Functionality: This metric was used to evaluate the capability of the system to carry out the functions for which it was designed. It was evaluated by comparing the

functionality of the system to the requirements and specifications that were set at the beginning of the project.

- Performance: Using this metric, we were able to analyze the performance of the system under a variety of different load scenarios. The performance of several different testing tools, such as Apache JMeter and Gatling, was evaluated and compared.

Discussion, Conclusion & Future Work

4.1 Lessons Learnt

Through the development and implementation of the proctoring system, several important lessons have been learned that could benefit future projects.

- Before beginning the project, it is essential to have a clear understanding of the issue that needs to be resolved. This is the first step. In this example, the requirement to prevent academic dishonesty in online exams was the motivating reason behind the proctoring system. Understanding the problem allowed the team to develop a comprehensive solution that addresses the root causes of the issue.
- Secondly, it is important to have a clear project plan that outlines the scope of work, timeline, and resource allocation. A Gantt chart was utilized to guarantee that each task was assigned to a specific team member, and deadlines were met. In addition, a budget table was crafted in order to keep tabs on costs and verify that the undertaking remained within its allotted spending limit.
- Lastly, it is absolutely necessary to have a software development process that is well outlined and in place. The agile software development process was utilized by the team, which enabled them to be adaptable and flexible in response to shifting requirements. This technique enabled the team to develop a working product within the timetable and budget restrictions.
- Fourthly, the team learned the importance of thorough testing before deployment. The system was put through a number of different rounds of testing to guarantee that it was running properly and that it was safe. In addition, the team carried out user acceptance testing to ensure that the system was simple to operate and that it fulfilled the requirements outlined by the end-users.

- Fifthly, communication is important to the success of any enterprise. The team used tools such as Trello and Slack to promote communication and collaboration between team members. Regular meetings of the team were held to discuss the progress that had been made, identify any problems that had arisen, and make decisions.
- Sixthly, it is vital to take into consideration the ethical repercussions that may result from the development of the technology. In the case of the proctoring system, the team made certain that the system was open and transparent, and that users were informed that they were being monitored. In addition, the team took precautions to guarantee that the system did not gather any unnecessary identifying information from users.
- Last but not least, the group realized the need of thorough documentation. Documenting the code, system architecture, and user manuals allowed for easier maintenance and troubleshooting in the future. It also made it easier for new team members to understand the system and contribute to its development.

4.2 Comments on Testing and evaluation

In order to ensure that the proctoring system is functional and satisfies the demands of its users, it is vital to receive feedback and comments from users through testing and assessment. Using this feedback, one can determine whether there are any areas in which the system could be improved and then make any necessary adjustments to the system.

Usability testing is one method that can be utilized to collect feedback from users. This entails observing users while they engage with the system and asking them to complete specified tasks. Testing the system's usability can help identify any problems with the user interface or the functionality of the system that may be preventing users from accomplishing their objectives. Users may, for instance, discover that it is challenging to traverse the system, or they may be unable to locate the information that they require. The developers can discover these issues and make the necessary modifications to improve the usability of the system by observing users and obtaining feedback from those using the system.

Conducting polls or questionnaires directed toward users is yet another method for eliciting their feedback. These can be utilized to collect data regarding the overall level of happiness that users have with the system, in addition to information regarding specific areas such as the simplicity of use, reliability, and correctness. Feedback on certain features or functionality can also be gathered through the use of surveys and questionnaires, which enables developers to discover any areas that might benefit from further development. Getting feedback from users can also be done through interviewing them or participating in focus groups. Comments provide an opportunity for users to discuss their experiences with the system in more depth, providing valuable information into how the system is being used and how it might be improved. It is also possible to gather feedback on new features or functionality that is being considered for the system by conducting interviews or holding focus groups. This provides the developers with the opportunity to evaluate the requirements and preferences of the users.

Resistance from users to the system is a typical problem that may appear during testing and evaluation. This may occur if users perceive the system as being intrusive or overly controlling, or if they do not trust the system's ability to accurately monitor their behavior. Additionally, this may occur if users do not trust the system's ability to monitor their behavior accurately. In order to find a solution to this problem, it is essential to include users in the process of design and development as much as is practically possible. This can help to develop trust and guarantee that the system is constructed in a way that satisfies the needs and expectations of its users.

While receiving input from consumers, the necessity for the privacy and security of their data is another key factor to take into mind. Users may be hesitant to submit feedback if they are concerned about the security of their personal information or data. In order to address these concerns, it is essential to make certain that the system is designed in a manner that safeguards the privacy of users and that any data that is collected is used exclusively for the purpose of enhancing the functionality of the system.

In conclusion, obtaining comments and feedback from users by means of testing and evaluation is a vital component of the process of designing a proctoring system that is both effective and user-friendly. The developers are able to discover any areas that might be improved and make the required modifications when they collect input from users through a range of different means. This helps to guarantee that the system satisfies the requirements and expectations of its users. It is crucial to involve users in the design and development process as much as possible, and to ensure that the system is designed in a way that preserves user privacy and security. Keeping these factors in mind enables software developers to design a proctoring system that is not only efficient but also meets the requirements of the people who will be using it.

4.3 Advantages of the project

When compared to more conventional approaches to supervising students during examinations, the proctoring system that was developed as part of this research offers a number of major advantages. These benefits can be broken down into a few different categories, the most important of which are ease of use, cost-effectiveness, precision, and safety.

The simplicity of the proctoring method is among the most significant advantages that it offers. Students do not require a physical testing center or an on-site proctor in order to take their examinations from any location of their choosing because the system is totally automated. Students, particularly those who are studying remotely or who have busy schedules, are given an easier time scheduling and taking tests as a result of this development. It also does away with the necessity for paper-based examinations and the logistics that come along with them, such as printing, transporting, and storing the paperwork.

The cost-effectiveness of the proctoring system is yet another advantage associated with using it. Because the system is completely automated, there is no longer a requirement to employ a sizable number of proctors, the management of which can be both costly and time-consuming. In addition to this, there is less of a demand for traditional testing facilities, which can be prohibitively expensive to both run and rent. This means that the cost of giving exams is substantially reduced, which can contribute to cost savings for both students and educational institutions.

Also, the proctoring method has a very high degree of accuracy. Because the system is completely computerized, there is no opportunity for a human to make a mistake, such as incorrectly grading or misinterpreting the outcomes of an exam. Because it employs sophisticated algorithms to detect and eliminate any instances of cheating behavior, it also removes the possibility of dishonesty on the part of users. This indicates that the results of the examination are extremely accurate and reliable, which is necessary for making significant decisions regarding the performance of students.

The proctoring system, in addition to its precision, also has a very high level of security. Because the system is entirely computerized, there is no longer any opportunity for the actual theft or misplacement of examination resources. In addition to this, it employs sophisticated encryption methods and security measures in order to prevent unauthorized access to or manipulation of examination data. This ensures that the data from the examination is kept in a highly safe and secured environment, which is absolutely necessary for preserving the honesty of the testing procedure.

In comparison to more conventional approaches to the monitoring of students while they are taking tests, the proctoring system that was developed as part of this project possesses a number of key features that make it superior in many respects. Because of its ease of use, low cost, high accuracy, and high level of safety, it is a great alternative for educational institutions that are trying to increase the overall quality of their examinations while also streamlining the processes involved in administering them. The significance of automated proctoring systems is only going to continue to increase as both the rate of technological advancement and the number of educational institutions that are moving toward online and remote learning increases.

4.4 Disadvantages of the Project

Although there are a number of benefits associated with the proctoring system, there are also a number of significant drawbacks that need to be taken into consideration. These potential drawbacks can include things like:

Issues About Privacy. The usage of proctoring systems gives rise to significant worries regarding individuals' privacy. These kinds of technologies gather a substantial quantity of personally identifiable information, including video recordings of people taking examinations. It is possible that if this information is obtained by the wrong people, it will be utilized for unethical reasons. There is also the possibility that the private information of individuals could be utilized improperly or sold for financial gain.

Proctoring systems can be difficult to set up and keep up to date due to the associated technical difficulties. They demand a great amount of technical expertise, and it's possible that they'll also need continuous assistance to make sure they're operating correctly. Exams are not immune to the possibility of being plagued by technical issues, which can cause both disruptions and added anxiety for candidates.

Proctoring systems are not failsafe and can produce false positives and negatives. False positives and negatives can be caused by false positives. It's possible that a false positive will accuse an honest student of cheating, while a false negative could mean that real wrongdoing goes unnoticed. This might result in outcomes that are unfair and damage to the reputations of the students. The cost of implementing a proctoring system can be prohibitive, particularly for educational establishments of a more modest size. The exorbitant cost of hardware, software, and continuing support can make it difficult for certain institutions to deploy these systems. This makes it possible for some institutions to implement these systems.

Another disadvantage of the system is its accessibility. It is possible that certain students will have difficulty using the proctoring systems due to accessibility concerns. For instance, those who have disabilities may have trouble placing themselves in front of a camera, and people who have sluggish internet connections may face challenges with technology.

Considerations of a moral and ethical nature must be made prior to the implementation of any proctoring system. Some people believe that the usage of such systems violates the norms of trust and respect that should exist between educational institutions and the students who attend those institutions. They call such practices invasive. Proctoring systems run the potential of being prejudiced against particular student groups due to the presence of bias in the systems. For those with darker skin tones, for instance, the facial recognition technology used in proctoring systems may not work as well as it does for people with lighter skin tones.

Students who use proctoring systems may experience adverse psychological impacts as a result of their use of these systems, including elevated levels of stress and anxiety. This may have a detrimental effect on both their overall health and their level of productivity. Proctoring methods are not a comprehensive answer to the problem of academic dishonesty; despite the fact that they may prevent some students from cheating, their effectiveness is limited. Pupils may still discover ways to cheat, such as using materials from outside the classroom or working together with other students.

Conclusion

The online proctoring system was developed as a solution to prevent cheating in online exams. The system function seamlessly by identifying the students' movements of the face, and the body movements during exam. As such the examiner can track whether the student sis cheating during exam. For instance, if the student comes with items, such as phones or tablets to the exam room, the system would detect these items and record them in the logs. This strategy enables the examiners to track or investigate any cheating in the exam. The process of developing this system is one that was tiresome and involving by all members of the group. The research work, coding, and documentation were needed to make everything work as required. Therefore, the efforts put by the team members show their ability to work together as a team, which prepares us for the job market in the future.

References

Al-Said Ahmad, A., & Andras, P. (2019). Scalability analysis comparisons of cloud-based software services. *Journal of Cloud Computing*, *8*(1), 1-17. https://doi.org/10.1186/s13677-019-0134-y

Baran, L., & Jonason, P. K. (2022). Contract cheating and the dark triad traits. In *Contract Cheating in Higher Education: Global Perspectives on Theory, Practice, and Policy* (pp. 123-137). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-031-12680-2_9

Duhaim, A. M., Al-Mamory, S. O., & Mahdi, M. S. (2021). Cheating detection in online exams during COVID-19 pandemic using data mining techniques. *Webology*, *19*, 1-26. https://www.webology.org/data-cms/articles/20220122120510pmWEB19026.pdf.

Li, X., Ma, M., Zhang, J., Zhao, H., Zhang, W., & Liu, Z. (2022, August). Software reliability evaluation system of ship information management System. In *2022 9th International Conference on Dependable Systems and Their Applications (DSA)* (pp. 81-85). IEEE. https://doi.org/10.1109/DSA56465.2022.00019

Roa'a, M., Aljazaery, I. A., ALRikabi, H. T. S., & Alaidi, A. H. M. (2022). Automated cheating detection based on video surveillance in the examination classes. *iJIM*, *16*(08), 125. https://www.academia.edu/download/88653247/11175.pdf

Shanthi, D. (2021, November). Ensemble approach of ACOT and PSO for predicting software reliability. In *2021 Sixth International Conference on Image Information Processing (ICIIP)* (Vol. 6, pp. 202-207). IEEE. https://doi.org/10.1109/ICIIP53038.2021.9702555