

School of Sciences and Engineering

Department of Computer Science and Engineering

Senior Project 1

**“Classroom Monitoring Using AI”**

**Software Requirements Specification Document**

Presented To:

Dr. Sherif Aly

Dr. [Hesham Eraqi](mailto:heraqi@aucegypt.edu)

* Table of Contents:

1. Introduction:

1.1 Purpose and Scope

1.2 Intended Audience

1.3 Contact information/SRS team members

1.4 Document Conventions

1.5 Definitions and Acronyms

2. Overall Description:

2.1 Product Perspective

2.2 Product Functions

2.3 Operating Environment

2.4 User Classes and Characteristics

3. External Interface Requirements:

3.1 Hardware Interfaces

3.2 Software Interfaces

3.3 User Interfaces

3.4 Communication Protocols and Interfaces

4. System Features:

4.1 Overview of System Features with Results

5. Non- Functional Requirements

5.1 Performance Requirements

5.2 Safety Requirements

5.3 Security Requirements

5.4 Software Quality Attributes

* 1 Introduction:
* 1.1 Purpose and Scope:

The purpose of the Software Requirements Specification document is to describe and highlight the software specifications of our senior project titled “Classroom Monitoring Using AI” and how it is expected to operate and function. The document will also describe the functional and non-functional requirements needed for the software to operate and achieve its goals by fulfilling the needs of the software’s users.

The scope of this project is to create a safe and fair examination environment for students that eliminates the high rates of cheating during examinations in Egypt, by implementing a system that detects possible cheating incidents in physical exams. Consequently, violating academic integrity creates a critical issue that negatively affects the reputation of the national education system, quality, and competence of graduates.

* 1.2 Intended Audience:

This Software Requirements Specification document is to describe and highlight the software specifications of the project. The document is mainly intended to be written for the developers of the “Classroom Monitoring Using AI” project and for future references for the users of this project.

* 1.3 Contact information/SRS Team Members:

In case you have any inquiries or questions regarding this Software Requirements Specification Document, below are the contact details for the project/SRS team members:

* Noha Abdelkader

Undergraduate Computer Engineering Student at The American University in Cairo

[noha.alshabrawy@aucegypt.edu](mailto:noha.alshabrawy@aucegypt.edu)

* Marwan Amr Awad

Undergraduate Computer Engineering Student at The American University in Cairo

[marwanawad@aucegypt.edu](mailto:marwanawad@aucegypt.edu)

* [Mohamed Elsayed Gamil Elshabshiri](mailto:elshabshiri@aucegypt.edu)

Undergraduate Computer Engineering Student at The American University in Cairo

[elshabshiri@aucegypt.edu](mailto:elshabshiri@aucegypt.edu)

* [Mohamed Ashraf Taha](mailto:mohammedashraf@aucegypt.edu)

Undergraduate Computer Engineering Student at The American University in Cairo

[mohammedashraf@aucegypt.edu](mailto:mohammedashraf@aucegypt.edu)

* [Omar Sherif Elmahdy](mailto:omarmahdy122@aucegypt.edu)

Undergraduate Computer Engineering Student at The American University in Cairo

[omarmahdy122@aucegypt.edu](mailto:omarmahdy122@aucegypt.edu)

* [Youssef Khaled Beshir](mailto:youssefbeshir@aucegypt.edu)

Undergraduate Computer Engineering Student at The American University in Cairo

[youssefbeshir@aucegypt.edu](mailto:youssefbeshir@aucegypt.edu)

* 1.4 Document Conventions:

This document was created based on the IEEE (IEEE Standard 830-1998) template for System Requirement Specification Documents.

* 1.5 Definitions and Acronyms:

YOLO: You Only Look Once which is a real-time object detection algorithm that identifies specific objects in videos, live feeds, or images

SSL: An SSL (Secure Sockets Layer) certificate is a digital certificate that authenticates a the identity and enables an encrypted connection

TLS: Transport Layer Security, the advanced version of SSL

HTTPS: Hypertext Transfer Protocol Secure

* 2 Overall Description:
* 2.1 Product Perspective:

The project is a standalone project. The project is developed using YOLOv3 object detection tool, Keras, TensorFlow Deep learning libraries based on Python programming language. The project is not built on any other projects.

* 2.2 Product Functions:

The functionalities of the product enable the user to be notified of possible cheating incidents in examination processes in a classroom environment. This includes a notification alert to the proctor in the classroom in addition to a snapshot/footage of the possible cheating incident of the student with his/her location in the classroom. The software generates a report at the end of the examination with detailed information of possible and reported cheating incidents.

* 2.3 Operating Environment:

The operating environment of the project is divided into 2 modules: the backend and the frontend. There are no required specified operating systems to use the system.

* 2.4 User Classes and Characteristics:

1. Exam Proctors: these are the main users of the software, exam proctors will use the application to get notified of possible cheating incidents during the exam. They do not need to have extensive technical background/skills since the software is easy to use for non-technical users.
2. Software Maintainers: the maintainers of the software are responsible for making sure that the software is fully functional and operating.

* 3 External Interface Requirements:
* 3.1 Hardware Interfaces:

The software will be connected to an IP security camera that will be fixed in the middle-top of the classroom to capture the video footage of the examination that will be fed into the software for real-time processing. It is recommended to have a camera of 4MP resolution in order to capture the movement and object details in a classroom setting. The IP camera also should have a suitable range of a minimum of 8 meters in order to capture a classroom of average dimensions of 8\*6 meters.

* 3.2 Software Interfaces:

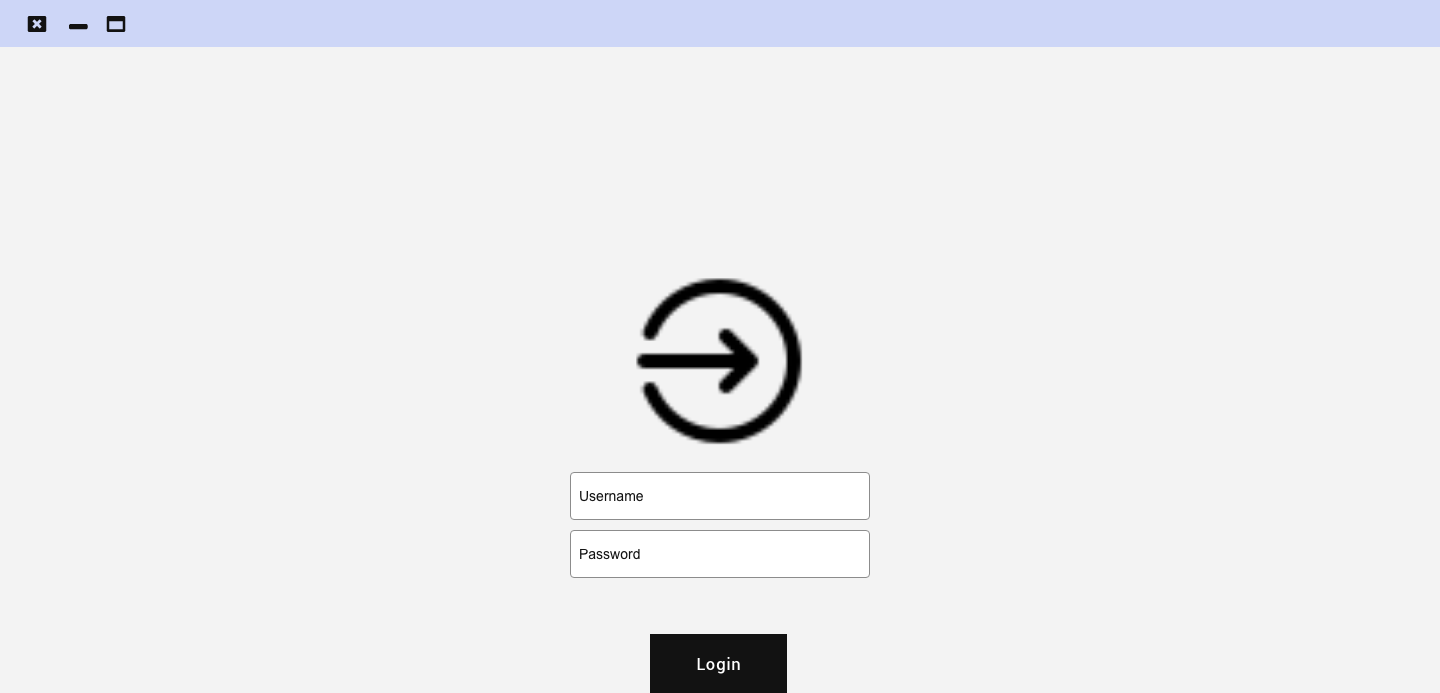
The software requires a Python 3 (or latest release) environment to run the system. The software can be connected to a database such as MySQL RDBMS (Relational Database management system) to store the login credentials of the proctors in addition to the locations of the students on each seat in the classroom setting.

* 3.3 User Interfaces:

The software has different interfaces that the user can interact with. Below are some of the interfaces that the user will interact with to do different functionalities in the software.

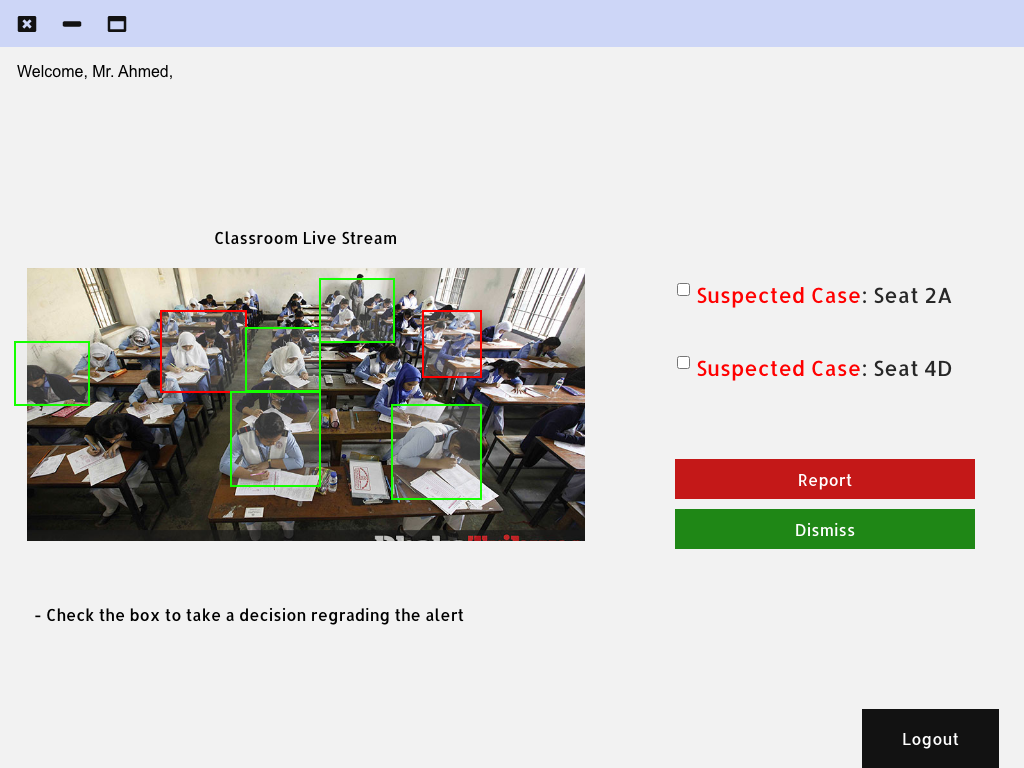
1- Login Page:

The user (proctor) is asked to enter his/her credentials to access the software.



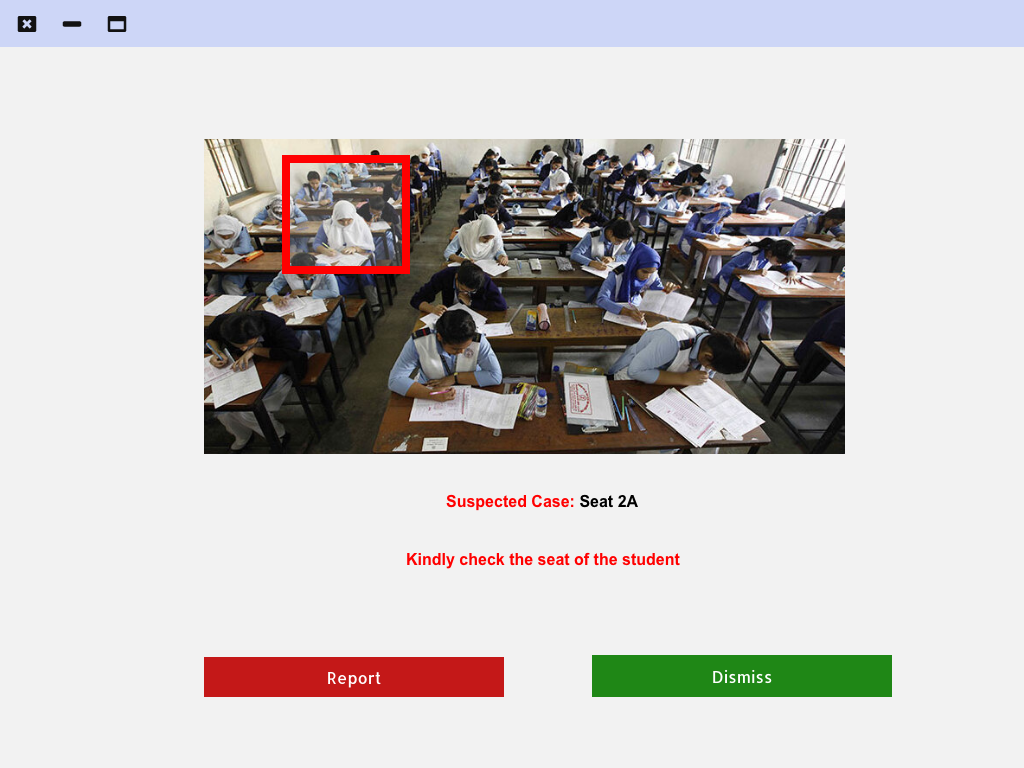
2- Home page:

The home page is where the user (proctor) can view a live stream of the classroom during the examination in addition to a notification panel where all possible cheating incidents are displayed on the panel to notify the proctor. The user selects any of the incidents to view more details, report, or dismiss the notification.

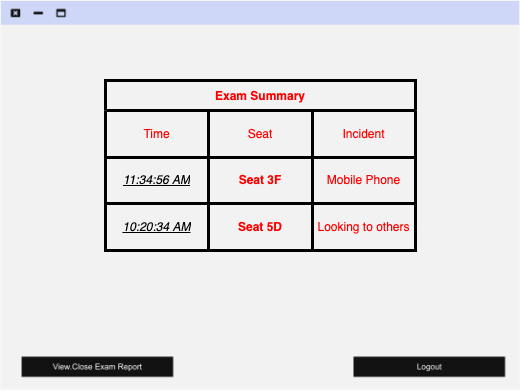


3- Case Details Page:

This page is viewed once the user selects any of the incidents on the notification panel on the home page. This page shows a snapshot of the incident with the seat number and an option to dismiss (after the student is checked by the proctor ) or report (to report the incident as a cheating incident)



4- Exam Summary Page:

The page shows a summary of all dismissed (in green) and reported (in red) incidents by the proctor during the exam.

* 3.4 Hardware Interfaces:

The minimum hardware specifications to operate the software is 8 Gigabytes of Random Access Memory (RAM) and an Intel Quad-core i7 processor, to be able to run the software with no complications.

* Communication Protocols and Interfaces:

The software requires an internet connection to connect to the remote database of the software, where exam reports and user information are stored. The connection between the security camera in the classroom and the computer is based on Real-Time Streaming Protocol (RTSP). The protocol is used to transmit the real-time live video to the software for processing and viewing.

* 4 System Features:
* 4.1 Overview of System Features and Results:

1- Viewing Possible Cheating Incidents:

System users (proctors) can view live streaming of the classroom setting with bounding boxes around each student in the classroom (green box indicating normal activity, the red box indicating suspicious cheating activity). A notification panel is displayed beside the live stream where all possible cheating incidents are viewed by the proctor with the seat number of the student, the user can dismiss or report the incident by selecting the incident.

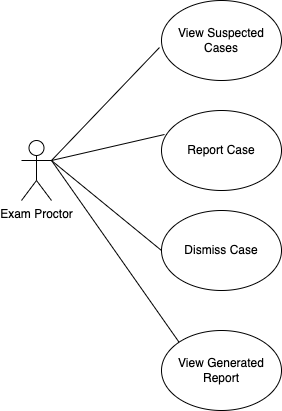
2- Reporting Cheating Incidents:

System users (proctors) can report a cheating case by viewing the details of a notification displayed on the notification panel on the software homepage. The user can view a snapshot of the suspected student while doing the suspicious cheating activity, in addition to his/her seat number.

3- Viewing The Generated Exam Report:

System users (proctors) can view a full examination report after the end of the exam. The report contains a summary of all the possible and reported cheating incidents during the exam with a timestamp for each incident and seat number.

The following is a use case diagram showing the functionalities of the system that the user can interact with:



* 5 Non-functional Requirements:
* 5.1 Performance Requirements:

The minimum hardware specifications to operate the software is 8 Gigabytes of Random Access Memory (RAM) and an Intel Quad-core i7 processor, to be able to run the software with no complications.

* 5.2 Safety Requirements:

To ensure the safety of all user information, user credentials are safely stored on the software’s database which offers network encryption to keep the communication between the client and the database as safest as possible, using TLS and SSL security certificates.

* 5.3 Security Requirements:

To ensure optimum security during the usage of the software. Users are required to enter their credentials [username and password] to allow the users to access the user privileges of the proctor.

* 5.4 Software Quality Attributes:

* Availability:

The system will be run frequently according to the exam time that may take about 1-3 hours. In case of any system crash, the system should recover within minutes and all the data would be stored and saved frequently.

* Reliability:

The system will ask the user to enter his/her login credentials to enter only allowed users to the system. In case of wrong login credentials, the system will prompt the user of the wrong login attempt. All generated data by the system will be stored in hard drives and cloud storage to ensure the security and accessibly of data.

* Security:

The system will use an online database that encrypts users’ data to ensure the security and confidentiality of the data. HTTPS will be also used to ensure the security of the connection between the client and server.

* Maintainability:

In case of a failure, any system module should be able to recover in a duration of 10-15 minutes to minimize the absence of the functionality of the system during the examination process

* Testability:

Testing will be done as a continuous process during the development of the system, including pre-testing stages, which ensures that system requirements are properly met. The following phase will be the system testing phase which will include testing the system after development. Several testing methodologies will be followed, such as, unit testing, functional testing, white box testing for the developed code, and the black box testing of the software in order to ensure the complete functionality of the system.