Indian Institute of Information Technology Allahabad

Department of Software Engineering

Software Requirements Specification (SRS)

Automated Exam Monitoring and Grading System

Submitted by:
Aman Jaiswal - MSE2024010
Vaibhav Patidar - MSE2024009
Pratik Rathod - MSE2024004

Under the Guidance of:
Dr. Gaurav Srivastava
Assistant Professor, Department of Software Engineering

Academic Year: 2025 (2nd semester)

May 4, 2025

1 Revision History

Author	**Date**	**Reason for Change**	**Version**
vaibhav-pat	2025-04-04	Initial commit	0.0
vaibhav-pat	2025-04-04	Added features in README	0.1
vaibhav-pat	2025-04-04	Added technologies to README	0.11
vaibhav-pat	2025-04-04	Updated README.md	0.12
amanSjaiswal	2025-04-22	Added introduction	0.2
Pratikrathod1632	2025-05-03	Add files via upload	0.3
vaibhav-pat	2025-05-04	Added use case: Monitor Exam Session	0.4
amanSjaiswal	2025-05-04	Added evaluate and grade exam use case	0.5
vaibhav-pat	2025-05-04	Deleted previous TakeOnline use case	0.51
Pratikrathod1632	2025-05-04	Updated file	0.6
vaibhav-pat	2025-05-04	Updated README.md file	0.13
vaibhav-pat	2025-05-04	Updated README.md file	0.14
vaibhav-pat	2025-05-04	Created a separate section for each use case	0.7
vaibhav-pat	2025-05-04	Updated README.md file	0.15
vaibhav-pat	2025-05-04	Updated README.md file	0.16
amanSjaiswal	2025-05-04	Class diagram of system	0.8
amanSjaiswal	2025-05-04	Final class diagram	0.9
vaibhav-pat	2025-05-04	Added SRS	1.0

2 Introduction

2.1 Purpose

The purpose of this Software Requirements Specification (SRS) document is to provide a detailed description of the Automated Exam Monitoring and Grading System. It outlines the functional and non-functional requirements, system features, and design constraints that define the system's behavior and structure. This system is designed to streamline and automate the process of conducting online exams by enabling real-time monitoring to prevent malpractice and facilitating efficient evaluation and grading of responses. The document serves as a reference for developers, testers, project managers, and stakeholders throughout the software development lifecycle.

2.2 Document Conventions

This document follows certain conventions for clarity and consistency:

- Requirements are labeled as FR-x for Functional Requirements and NFR-x for Non-Functional Requirements.
- UML diagrams are used to model the system's behavior and structure.
- Actors are depicted using stick figures in use case diagrams.
- Class names follow the CamelCase naming convention.
- Italics are used to refer to system modules or subsystems.

2.3 Intended Audience and Reading Suggestions

This document is intended for the following readers:

- Developers – to understand system functionalities and component interactions.
- Testers – to create test cases based on the defined requirements.
- Project Managers – to track progress and ensure implementation completeness.
- Clients and Stakeholders – to verify that the system meets specified expectations.
- UI/UX Designers – to design user interfaces that align with the functional flow.

Readers are encouraged to begin with the overall description to understand the context of the system, and then proceed to system features and interface requirements for deeper insights.

2.4 Product Scope

The Automated Exam Monitoring and Grading System is a web-based application designed to facilitate secure and efficient online assessments. It provides features such as exam creation and scheduling, real-time monitoring of students via webcam and screen capture, and AI-powered cheating detection mechanisms. The system automates the evaluation and grading process and supports immediate result generation with performance analytics. It accommodates various assessment types, both objective and subjective, and is suitable for institutions, universities, and online learning platforms seeking a scalable and secure testing environment.

2.5 References

- IEEE Std 830-1998 – IEEE Recommended Practice for Software Requirements Specifications
- UML 2.5 Specification – Object Management Group (OMG)
- Institution/University guidelines on exam policies (if applicable)
- Research papers and tools related to AI-based proctoring (e.g., OpenCV, MediaPipe)
- Online Assessment Standards and Privacy Regulations (e.g., GDPR, FERPA)

3 Overall Description

3.1 Product Perspective

The Automated Exam Monitoring and Grading System is a standalone, web-based solution designed to support the entire lifecycle of online examinations, ranging from exam creation to real-time monitoring and automated grading. The system integrates artificial intelligence techniques to enhance the reliability and efficiency of assessments conducted in remote environments. While it is capable of functioning independently, it is also designed to integrate seamlessly with existing Learning Management Systems (LMS) such as Moodle or Canvas, thereby extending its utility in academic institutions.

The system architecture comprises the following core modules:

- **Exam Management Module** – Handles the creation, scheduling, and publishing of exams.
- **Monitoring Module** – Enables real-time proctoring using webcam, microphone, and screen activity tracking.
- **Evaluation Module** – Supports automatic evaluation of objective-type questions and facilitates manual or assisted grading of subjective responses.
- **User Management Module** – Manages different user roles including students, instructors, and proctors.

3.2 Product Functions

The system offers a comprehensive set of functionalities to support secure and streamlined online assessments. These include:

- Secure user authentication with role-based access control.
- Creation, editing, and scheduling of exams.
- Delivery of question papers within a controlled environment.
- Live monitoring of examinees using webcam feeds and screen activity tracking.
- Detection and logging of suspicious behaviors such as a missing face, presence of multiple individuals, or unusual background noise.
- Automatic evaluation of multiple-choice and numeric-response questions.
- A dedicated interface for instructors to manually assess subjective answers.
- Generation of detailed scorecards along with performance analytics.
- Export capabilities for results and monitoring logs in various formats.

3.3 User Classes and Characteristics

The system is designed to accommodate different types of users, each with specific responsibilities and requirements:

- **Students**: These are the primary users who will take the online exams. They are expected to have a reliable internet connection, access to a webcam and microphone, and basic proficiency in operating a computer.
- **Instructors**: Responsible for exam creation and grading. While they may also participate in live monitoring, this is not mandatory.
- **Proctors/Monitors**: Tasked with supervising exam sessions either in real time or by reviewing flagged activities after the exam.
- **Administrators**: Oversee system configuration, manage user accounts and roles, schedule exams, and generate institutional-level reports.

3.4 Operating Environment

The system operates in a standard web-based environment, with components designed for compatibility and scalability. Key technical specifications include:

- **Frontend**: Web-based interface accessible via modern browsers such as Chrome, Firefox, and Safari.
- **Backend**: Hosted on cloud platforms or institutional servers, with support for Node.js, Python, or Java-based services.
- **Database**: Compatible with relational and NoSQL databases including MySQL, PostgreSQL, and MongoDB.
- **AI/Monitoring Engine**: Powered by Python modules utilizing libraries such as OpenCV and TensorFlow.
- **Network Requirements**: Stable internet connectivity is essential for both students and faculty during exam sessions.

3.5 Design and Implementation Constraints

Several constraints influence the design and deployment of the system. It must adhere to data privacy regulations such as GDPR, particularly in relation to webcam and microphone access. The platform should perform reliably across diverse hardware configurations, including laptops and desktops, and must be capable of scaling to accommodate large numbers of simultaneous users. It is also important to note that the effectiveness of AI-based monitoring may vary depending on factors such as lighting conditions and device capabilities.

3.6 User Documentation

To support users in effectively operating the system, comprehensive documentation will be provided. This includes:

- User manuals tailored for different user roles (Students, Instructors, Proctors, and Administrators).
- Contextual help features and tooltips embedded in the user interface.
- Step-by-step video walkthroughs demonstrating system setup and usage.
- A frequently asked questions (FAQ) section and a troubleshooting guide.

3.7 Assumptions and Dependencies

The successful functioning of the system relies on certain assumptions and external dependencies. It is assumed that all users will have access to devices equipped with a functional webcam and microphone. A stable internet connection is considered a prerequisite to ensure uninterrupted exam monitoring. Unless AI-based grading is integrated for subjective responses, instructors are expected to manually evaluate them. Furthermore, the system assumes institutional support in enforcing examination policies and addressing grievances raised by users.

4 External Interface Requirements

4.1 User Interfaces

The system provides a set of responsive, web-based user interfaces tailored to different user roles, ensuring ease of use, clarity, and seamless interaction across devices. Each interface is optimized for the specific needs and workflows of its user class.

4.1.1 Student Interface

Students will interact with the system through a streamlined series of views, including:

- A login and registration screen.
- A page displaying exam instructions and institutional rules.
- A system check module for webcam and microphone access validation.
- A live exam interface comprising a question viewer, navigation controls, and an exam timer.
- A submission confirmation page and a post-exam feedback section.

4.1.2 Instructor Interface

Instructors are provided with comprehensive tools to manage and evaluate examinations:

- A dashboard for creating, editing, and scheduling exams.
- A manual grading panel for evaluating subjective responses.
- A review interface for examining flagged activities during exams.
- Access to student performance analytics and result reports.

4.1.3 Proctor Interface

Proctors are equipped with functionalities to monitor examinees effectively in real time:

- A live monitoring dashboard displaying multiple candidates simultaneously.
- Instant alerts for suspicious behavior such as face absence or audio anomalies.
- Controls to message or remove candidates from an exam session.
- A panel to review activity logs after the exam concludes.

4.1.4 Admin Interface

Administrators are granted access to higher-level system functions, including:

- User and role management tools.
- System configuration and maintenance options.
- Utilities for exporting logs, grading data, and audit records.

4.2 Hardware Interfaces

4.2.1 Student Systems

The system requires standard hardware configurations for student-side compatibility:

- A webcam with a minimum resolution of 720p and a functioning microphone.
- Desktop or laptop computers with a standard keyboard and mouse.
- Support for video and audio capture through a web browser using WebRTC technology.

4.2.2 Server Infrastructure

The backend infrastructure must support scalable performance and real-time monitoring:

- Cloud-hosted or on-premise servers capable of processing concurrent video streams.
- GPU acceleration is recommended to facilitate AI-powered monitoring features.

4.3 Software Interfaces

The system is designed to operate across commonly used platforms and relies on a range of third-party tools and APIs for core functionalities.

- **Operating Systems**: Compatible with the latest two versions of Windows, macOS, and Linux.
- **Web Browsers**: Fully supported on the latest releases of Google Chrome, Mozilla Firefox, and Microsoft Edge.

4.3.1 Integrated Libraries and APIs

The system will leverage various integrated libraries and APIs for key features:

- **OpenCV**, **MediaPipe**, and **TensorFlow** for facial detection and behavior analysis.
- **WebRTC** for secure, browser-based audio and video communication.
- **MySQL**, **PostgreSQL**, or **MongoDB** as supported database backends.
- **OAuth 2.0** or institution-specific Single Sign-On (SSO) for secure user authentication.

4.4 Communications Interfaces

4.4.1 Internet Connectivity

A stable and consistent internet connection is required for uninterrupted exam sessions:

- Minimum required bandwidth is 2 Mbps for both upload and download.
- Continuous connectivity must be maintained throughout the examination period.

4.4.2 Data Transmission Protocols

Secure and real-time data exchange is facilitated using:

- **HTTPS** for all data communications to ensure encryption and security.
- **WebSocket** or **WebRTC** protocols for transmitting real-time video and audio streams.
- **RESTful APIs** to handle backend interactions, including fetching exams, submitting answers, and logging events.

5 System Features

5.1 Take Online Exam

This feature enables students to securely participate in exams through an intuitive, browser-based interface. The process is carefully structured to ensure a smooth and secure experience, encompassing login authentication, instructions acknowledgment, question navigation, timed progression, and seamless submission. The system ensures that no exam is left incomplete by implementing an automatic submission mechanism once the allotted time expires.

Functional Requirements:

- **FR1.1 – User Authentication:** The system shall require students to log in using valid credentials before accessing any examination content.
- **FR1.2 – Display Instructions:** Prior to the start of the exam, the system shall present exam-specific instructions and behavioral rules to the student.
- **FR1.3 – Load Exam Questions:** The system shall retrieve and display questions based on the settings defined by the instructor or exam administrator.
- **FR1.4 – Question Navigation and Answering:** Students shall be able to navigate between questions, record their answers, and revise responses at any point until submission.
- **FR1.5 – Auto Submission on Timeout:** Once the exam duration expires, the system shall automatically submit the student's responses.
- **FR1.6 – Submission Confirmation:** A visual confirmation message shall be displayed to confirm successful submission of the exam.

5.2 Monitor Exam Session

To uphold academic integrity, this feature offers real-time monitoring of candidates through video and audio streams. It integrates AI-based facial recognition and anomaly detection to flag suspicious behaviors such as missing candidates, multiple faces, or unexpected audio signals. These events are logged and notified to human proctors who can intervene during or after the session.

Functional Requirements:

- **FR2.1 – Activate Monitoring Tools:** The system shall activate the student's webcam and microphone before the exam begins for continuous monitoring.
- **FR2.2 – Detect Anomalies:** The monitoring module shall detect and log issues such as face absence, multiple faces, and abnormal background sounds.
- **FR2.3 – Real-time Alerts:** Detected anomalies shall trigger alerts in real time, allowing proctors to respond promptly.
- **FR2.4 – Proctor Viewing Panel:** Proctors shall be provided with a dashboard showing live video feeds and flagged events for active or post-exam review.
- **FR2.5 – Logging and Auditing:** All monitoring logs, including timestamps and flagged anomalies, shall be securely stored for audit and investigation purposes.

5.3 Evaluate and Grade Exam

This feature streamlines the evaluation process by automatically grading objective questions and offering instructors a comprehensive grading interface for subjective responses. Instructors can adjust scores manually and generate detailed performance reports for each student.

Functional Requirements:

- **FR3.1 – Auto Grading for Objective Questions:** The system shall automatically evaluate and score questions such as multiple-choice and numeric entries using predefined answer keys.
- **FR3.2 – Manual Grading Interface:** Instructors shall be able to assess subjective responses using a grading dashboard designed for efficiency and clarity.
- **FR3.3 – Score Override:** Instructors shall have the ability to override automatically calculated scores when needed.
- **FR3.4 – Final Score Calculation:** The system shall compute the final grade by aggregating both automatically and manually assigned scores and store it in the student's record.
- **FR3.5 – Report Generation:** Grade reports and performance analytics shall be generated by the system and made available in downloadable formats.

5.4 Manage Users and Roles

User and role management ensures secure and structured access to system functionalities. This feature allows administrators to define roles and responsibilities, enforce access permissions, and maintain comprehensive audit trails of all system interactions.

Functional Requirements:

- **FR4.1 – User Account Management:** Administrators shall have the ability to create, update, and delete user accounts as needed.
- **FR4.2 – Role Assignment:** The system shall support predefined roles—Student, Instructor, Proctor, and Admin—and allow role assignment accordingly.
- **FR4.3 – Role-based Access Control:** Access to system functionalities shall be granted or restricted based on each user's role.
- **FR4.4 – Activity Logging:** The system shall track and store user activities to support transparency and accountability.

5.5 Exam Management

This feature empowers instructors and administrators to create and publish exams with a wide range of customization options. It includes support for diverse question types, question randomization, precise scheduling, and notification broadcasting to students.

Functional Requirements:

- **FR5.1 – Exam Creation:** Instructors shall be able to design exams using various question types, including multiple-choice, short-answer, and essay-style formats.

- **FR5.2 – Question Bank and Randomization:** The system shall support the use of question banks and enable randomized question sets for individual students to discourage cheating.

- **FR5.3 – Scheduling:** Exam start/end times and durations shall be configurable by the instructor at the time of exam creation.

- **FR5.4 – Exam Notifications:** Registered students shall receive system-generated notifications once an exam has been published.

6 Other Nonfunctional Requirements

6.1 Performance Requirements

To ensure a seamless experience for users during high-stakes examinations, the system must meet specific performance benchmarks:

- **NFR1 – Concurrent User Support:** The system shall support a minimum of 1,000 concurrent users without noticeable degradation in responsiveness or stability, maintaining consistent performance under peak loads.
- **NFR2 – Auto-Grading Response Time:** The system shall return results for objective-type questions within 5 seconds after submission, ensuring prompt feedback to students.
- **NFR3 – Real-Time Monitoring Performance:** The system's video and audio monitoring capabilities shall maintain a minimum frame rate of 15 fps under standard network conditions to facilitate effective and reliable proctoring.
- **NFR4 – System Availability During Exams:** The system shall provide an uptime of at least 99.5% during scheduled exam periods to ensure that services are continuously accessible without interruptions.

6.2 Safety Requirements

The system shall incorporate safety protocols to protect data integrity, prevent data loss, and safeguard user privacy during exam sessions:

- **NFR5 – Privacy of Monitoring Data:** Live video and audio streams shall be accessible only to authorized personnel (i.e., assigned proctors and administrators) to preserve student privacy and comply with institutional and legal guidelines.
- **NFR6 – Auto-Save on System Failure:** In the event of a system crash or abnormal shutdown during an exam, the system shall automatically save all current student responses and immediately notify the administrator.
- **NFR7 – Consent for Device Access:** Before accessing a student's webcam or microphone, the system shall explicitly notify the user and obtain their consent, in alignment with digital privacy norms and applicable data protection laws.

6.3 Security Requirements

Strong security measures are essential for maintaining trust, protecting sensitive data, and ensuring the authenticity of examination processes:

- **NFR8 – Data Encryption:** All sensitive information—including user credentials, exam content, and results—shall be encrypted both in transit (using HTTPS) and at rest (using robust encryption algorithms).
- **NFR9 – Role-Based Access Control (RBAC):** The system shall enforce RBAC to ensure that access to data and functionalities is appropriately restricted based on the user's role (e.g., Student, Instructor, Proctor, Admin).

- **NFR10 – Login Attempt Limitation:** The system shall limit failed login attempts to five (5). Exceeding this limit shall lock the account for 30 minutes, mitigating brute-force attack risks.
- **NFR11 – Audit Logging:** All critical user activities—such as login attempts, exam submissions, grading actions, and system setting modifications—shall be logged for auditing and traceability purposes.

6.4 Software Quality Attributes

To deliver a robust and dependable system, the software shall be built upon key quality attributes:

- **Availability:** The system shall be available 24/7, except during scheduled maintenance, to support users from different time zones and ensure uninterrupted access.
- **Scalability:** The software architecture shall support horizontal scalability, allowing for seamless addition of servers to handle increased loads and user traffic without affecting performance.
- **Usability:** The user interface shall be intuitive and user-friendly, designed to accommodate users with basic computer and digital literacy skills, reducing learning overhead.
- **Maintainability:** The system shall employ a modular design, enabling easy updates, maintenance, and integration of additional plugins or external modules (e.g., AI-based proctoring, analytics).
- **Portability:** The application shall be cross-platform compatible (including Windows, Linux, and macOS) and function effectively across all major modern web browsers such as Chrome, Firefox, Microsoft Edge, and Safari.

6.5 Business Rules

To support consistent enforcement of institutional policies and maintain academic integrity, the system shall adhere to the following business rules:

- **NFR12 – Student Verification:** Only students who are registered and institutionally verified shall be permitted to access and participate in any online examination.
- **NFR13 – Proctor Authorization:** Only authorized institution-assigned staff members with valid credentials shall be allowed to act as proctors during any exam session.
- **NFR14 – Grading Deadline:** All submitted exams must be graded and the results published within seven (7) days following the conclusion of the exam.
- **NFR15 – Anomaly Threshold for Review:** If a student is flagged more than three (3) times for suspicious behavior during an exam, their session shall be automatically escalated for manual review by the designated proctor or administrator.

7 Other Requirements

This section outlines additional requirements that ensure the system aligns with legal standards, institutional policies, and technical constraints not covered in earlier sections.

- **OR1 – Legal and Policy Compliance:** The system shall comply with all relevant institutional academic integrity policies and applicable data protection regulations, such as the General Data Protection Regulation (GDPR). This includes proper handling of user consent, data retention policies, and rights to access or erase personal information. Compliance must be ensured both during development and throughout the system's operational lifecycle.

- **OR2 – Time Synchronization:** To ensure fairness in examination scheduling, the system shall implement real-time synchronization between all client devices and the central server. This mechanism shall ensure that exam start and end times remain consistent, regardless of the user's geographic location or device clock settings.

- **OR3 – Log Retention Policy:** All system-generated logs—including user activity logs, monitoring records, and proctoring alerts—shall be securely retained for a minimum duration of six (6) months. This retention period supports regulatory audits, academic appeals, and internal investigations as needed.

- **OR4 – Exportable Logs:** The system shall provide functionality to export monitoring data, user interaction logs, and proctoring notes in widely supported formats such as CSV or JSON. This ensures compatibility with external reporting tools and enables institutional stakeholders to perform in-depth analysis or archival.

- **OR5 – Platform Accessibility Constraints:** To maintain exam integrity and ensure a standardized testing environment, the system shall support full access via desktop and tablet devices (equipped with modern operating systems and browsers). However, access from mobile phones shall be restricted, as these devices present higher risks for academic dishonesty and are less conducive to secure proctoring.

Appendix A: Glossary

This section provides definitions of key terms used throughout the Software Requirements Specification to ensure clarity and a shared understanding among all stakeholders.

Proctor: An individual, typically assigned by the institution, responsible for overseeing and monitoring students during an examination to ensure compliance with academic integrity guidelines.

Flagged Activity: Any event automatically or manually identified as suspicious or potentially violating examination rules—such as multiple faces on camera, prolonged absence, or unusual background noise—requiring review by a proctor or administrator.

Auto-Grading: A system-driven mechanism that evaluates objective-type questions (e.g., multiple-choice, true/false, numeric responses) by comparing student responses to a predefined answer key, eliminating the need for manual intervention.

Subjective Grading: The manual evaluation process applied to responses that require critical thinking or elaboration—such as short answers or essays—performed by instructors based on rubrics or qualitative judgment.

Role-Based Access: A security and access control strategy wherein system permissions and functionalities are granted based on the user's role (e.g., Student, Instructor, Proctor, Admin), ensuring appropriate and restricted access to data and actions.

Exam Session: A defined time period during which a student attempts and completes an examination. The session begins when the student starts the exam and ends upon submission or expiration of the allotted time.

Appendix B: Analysis Models

B.1 Use Case Diagrams

B.1.1 Use Case: Take Online Exam

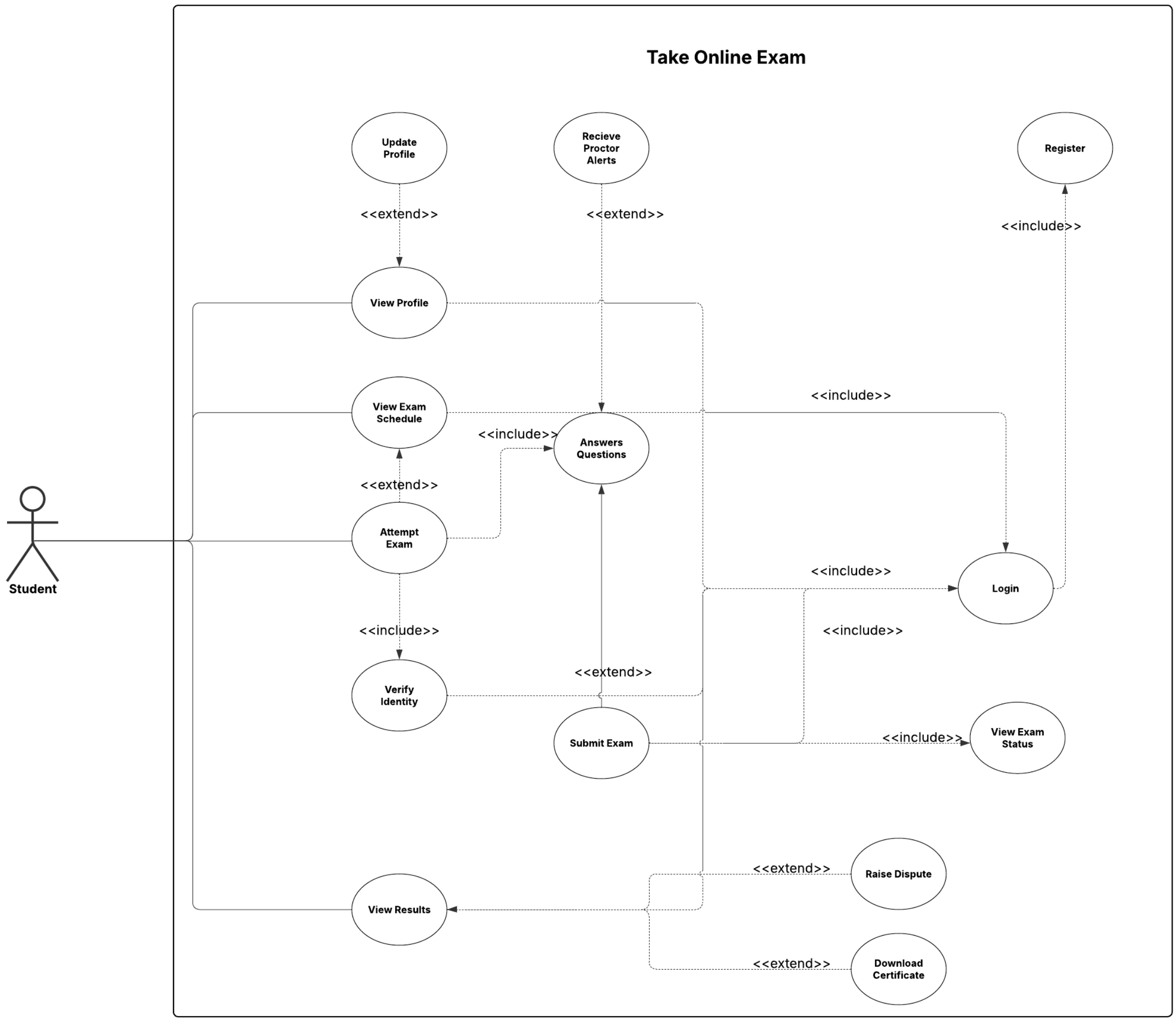

Figure 1: Use Case: Take Online Exam

Description: Depicts the student's interaction with the system during an exam session, covering the complete process from login to submission.

Actors:

[noitemsep]Student Exam System

Key Functionalities:

[noitemsep]Secure authentication and login Display of instructions and questions Real-time time tracking with auto-submit on timeout Confirmation message upon successful submission

B.1.2 Use Case: Monitor Exam Session

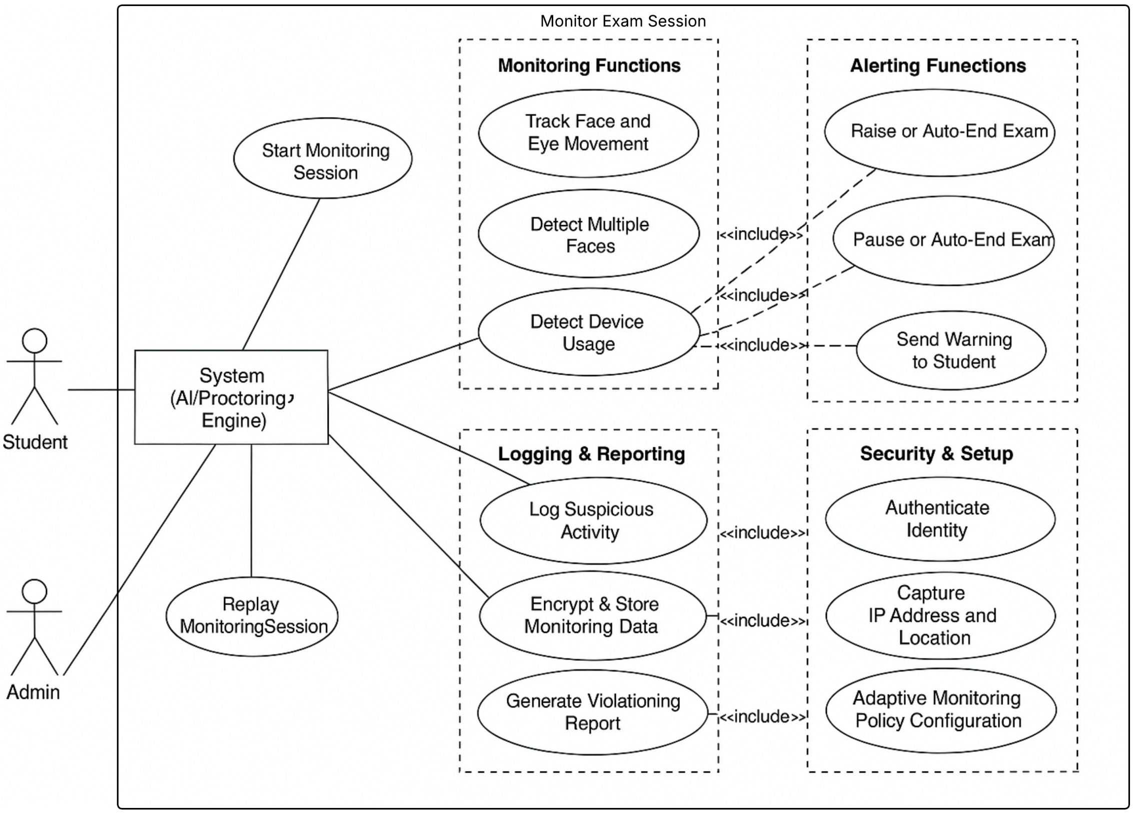

Figure 2: Use Case: Monitor Exam Session

Description: Shows the collaborative monitoring of the exam session using both AI-based detection and manual intervention by the proctor.

Actors:

[noitemsep]Student Proctor Monitoring Module

Key Functionalities:

[noitemsep]Live webcam and microphone streaming AI-driven anomaly detection (e.g., multiple faces, noise) Event logging and alert generation Real-time oversight and manual intervention by proctors

B.1.3 Use Case: Evaluate and Grade Exam

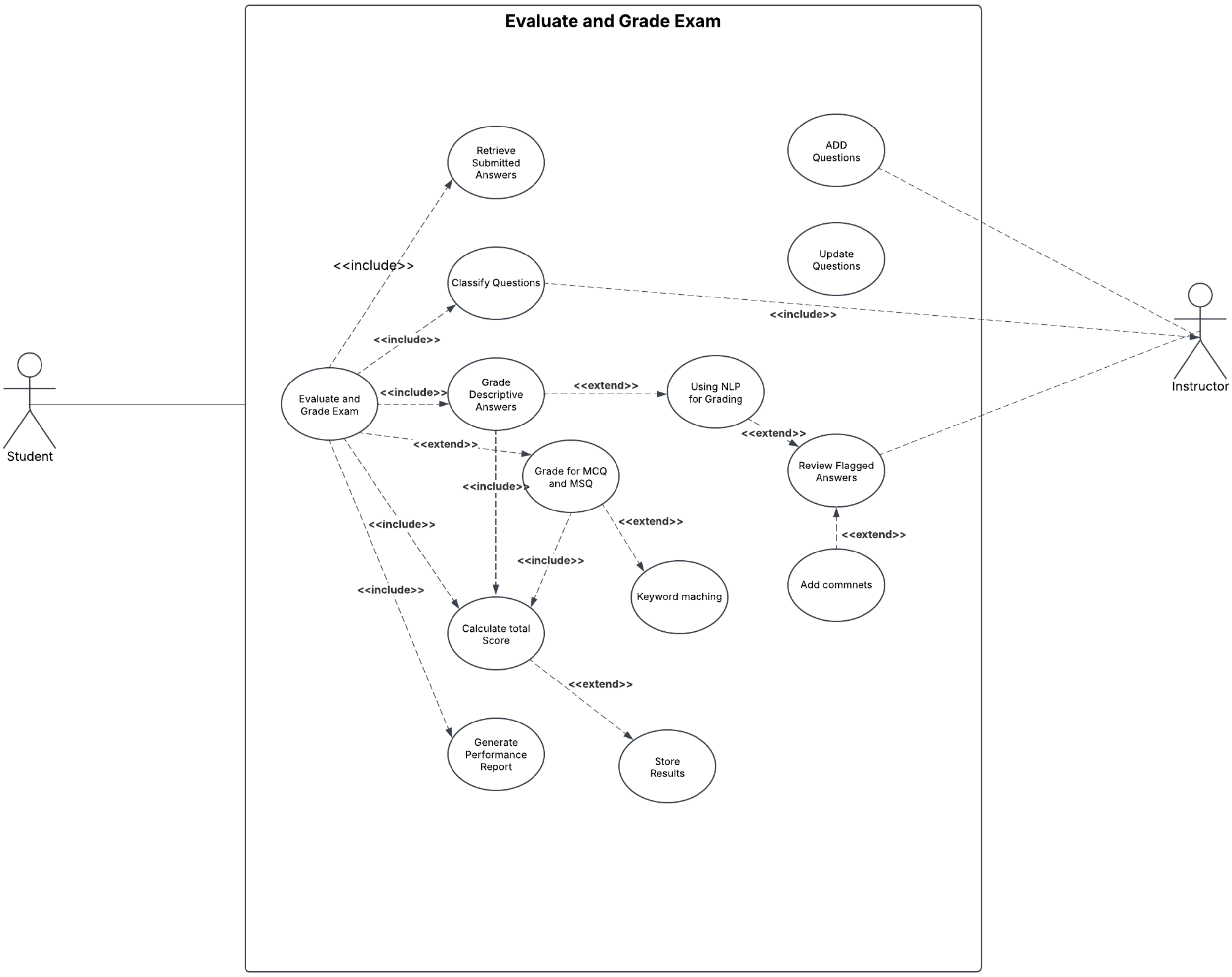

Figure 3: Use Case: Evaluate and Grade Exam

Description: Illustrates the grading process post-exam submission, including automatic evaluation of objective answers and manual grading of subjective ones.

Actors:

[noitemsep]Instructor Student Exam Grading Module

Key Functionalities:

[noitemsep]Auto-grading for objective-type questions Manual grading interface for essays and long answers Override mechanism for grading corrections Final report generation and publishing

B.2 Class Diagram

Description: The class diagram provides a static representation of the system's structure, illustrating its key classes, attributes, methods, and interrelationships.

Key Classes and Responsibilities:

- **User** – Abstract class with shared attributes (name, email, userID, role). Specialized into: Student, Instructor, Proctor, Admin.
- **Exam** – Attributes: examID, title, duration, startTime, endTime, questionList. Linked to one or more Question objects.
- **Question** – Supports various types (MCQ, Subjective, True/False). Contains questionID, text, options, correctAnswer, and grading schema.
- **ExamAttempt** – Records student responses and submission time. Connected one-to-one with Student.
- **ProctoringModule** – Performs real-time video/audio analysis, logs anomalies, and generates alerts.
- **GradingModule** – Handles both automated and manual grading workflows.
- **Report** – Generates result summaries and detailed performance breakdowns. Aggregates results from GradingModule.
- **Log** – Stores all user actions, system events, and monitoring records for audit trails.

Key Classes and Responsibilities:

- **User** – Abstract class with shared attributes (name, email, userID, role). Specialized into: Student, Instructor, Proctor, Admin.
- **Exam** – Attributes: examID, title, duration, startTime, endTime, questionList. Linked to one or more Question objects.
- **Question** – Supports various types (MCQ, Subjective, True/False). Contains questionID, text, options, correctAnswer, and grading schema.
- **ExamAttempt** – Records student responses and submission time. Connected one-to-one with Student.
- **ProctoringModule** – Performs real-time video/audio analysis, logs anomalies, and generates alerts.
- **GradingModule** – Handles both automated and manual grading workflows.
- **Report** – Generates result summaries and detailed performance breakdowns. Aggregates results from GradingModule.
- **Log** – Stores all user actions, system events, and monitoring records for audit trails.

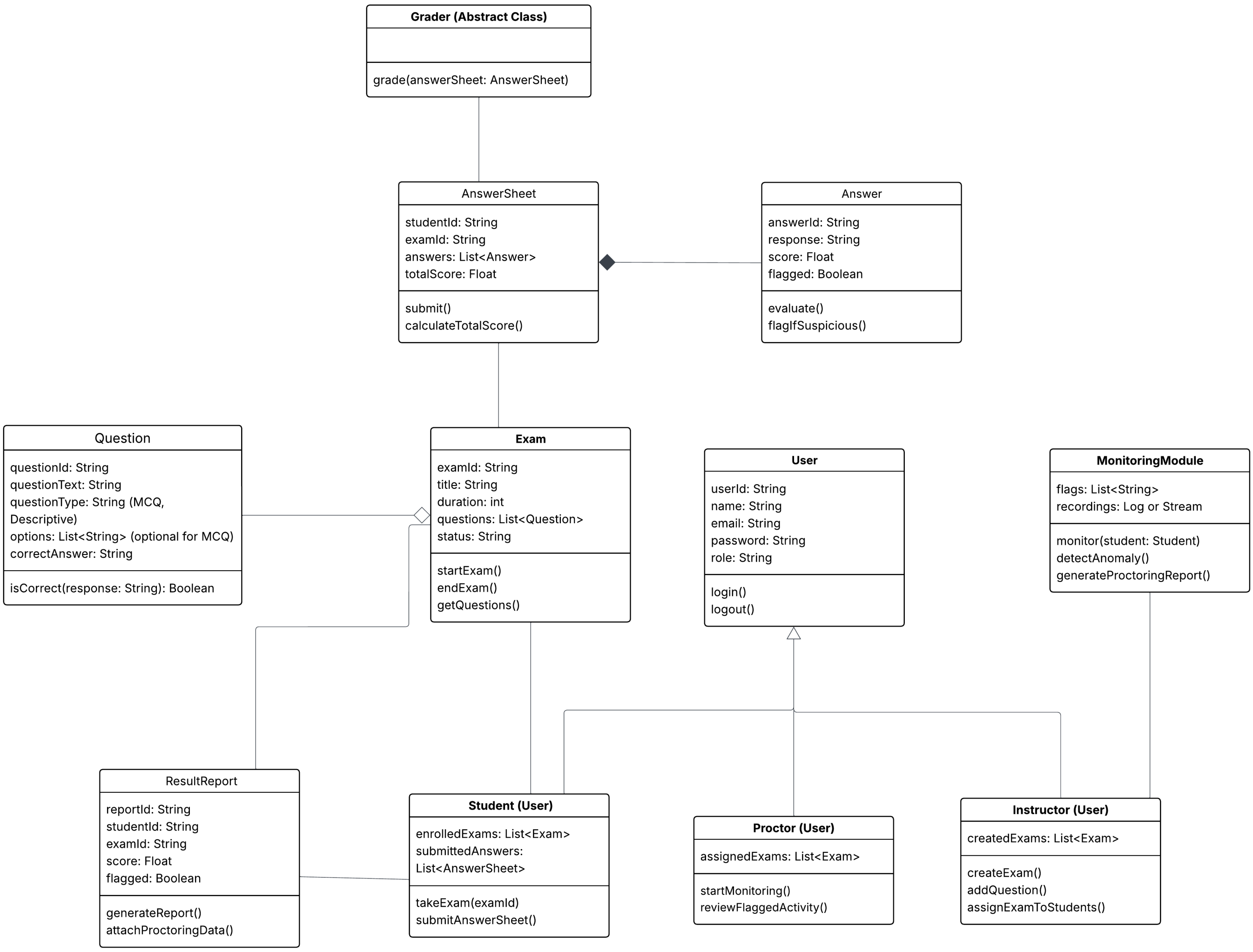

Figure 4: Class Diagram: System Structure

Appendix C: To Be Determined (TBD) List

This section lists items that are currently pending decisions or further clarification in the system's design and implementation.

TBD-1: Third-party Proctoring APIs A decision is pending on whether to integrate external proctoring solutions or develop proprietary real-time monitoring modules in-house.

TBD-2: Final Grading Algorithm Specific rules and weightages for combining subjective and objective scores are yet to be finalized.

TBD-3: Exam Export Formats The supported formats for exam/question data import/export (e.g., Excel, QTI) are under consideration.

TBD-4: Notification Methods The choice between SMS, email, or in-app notifications (or a combination) for alerts and updates is still being evaluated.

TBD-5: Language Support While the initial release will be in English, support for additional languages will be determined based on user demand and regional requirements.