Software Requirements Specification (SRS)

Automated Question Paper (MCQ) Generator
Version 1.0

Prepared by:

Varun Acharya K.V (231IT083)
Suvarna G (231IT077)
Padi Napa (231IT045)

August 25, 2025

Table of Contents

1 Introduction

1.1 Purpose

The purpose of this document is to define the functional and non-functional requirements of an AI-Based MCQ Examination System. This system enables examiners to create question papers automatically using APIs by specifying a topic and difficulty level. It also allows students to attempt the exam online, where each student receives the same set of questions but in a different randomized order.
the intended audience for this document include:

1. Examiners/Faculty (to generate and conduct exams)
2. Students (to take exams)
3. Developers (to build and maintain the system)
4. Testers (to verify correctness and usability)

1.2 Scope

The system will have the following characteristics and features:

1. Allow examiners to log in and generate MCQ question papers automatically by providing a topic and difficulty level.
2. Uses API to backend to generate questions and answer choices for each of the questions.
3. Save the generated question paper in the system database and use it to generate a shuffled set of questions for each student.
4. Allow multiple students to log in and take the same exam online concurrently.
5. Present the same set of questions to all students but in different randomized orders to reduce cheating.
6. Automatically evaluate responses in the backend and display scores to the student.
7. Provide exam results and analysis to the examiner on their dashboard.
8. Provides a one-time email link to verify user email.
9. Provides measures to make sure that one student can log in from only one computer system.

Benefits of our system:

1. Reduces examiner workload for question creation.
2. Provides a secure and randomized exam experience.
3. Enables remote/online assessments.

1.3 Definitions, Acronyms and Abbreviations

1.3.1 Definitions

1. Student - person who is taking the quiz/test/exam.
2. Examiner - person who is examining the students.
3. RESTful API - web service that uses standard HTTP methods to access and manage resources through URLs, typically exchanging data in JSON.

1.3.2 Acronyms and Abbreviations

1. MCQ - Multiple Choice Questions.
2. API – Application Programming Interface.
3. REST - Representational State Transfer.
4. HTTPS - Hypertext Transfer Protocol Secure.
5. JSON - JavaScript Object Notation.

1.4 References

- OpenAI API Documentation (v1)
- React.js Documentation (v19.1)
- Express documentatino (v5.1.0)

1.5 Overview

This document is organized to provide a comprehensive understanding of the software system and its requirements. It is structured as follows:

- **Section 2: Overall Description** — Describes the general factors that affect the product and its requirements, including product perspective, functions, user characteristics, and constraints.
- **Section 3: Specific Requirements** — Details the functional and non-functional requirements that the system must satisfy, the external interfaces and the constraints that are to be kept in mind while making the system.
- Appendix
- Index

2 Overall description

2.1 Product Perspective

The system is a web-based application with separate roles for examiners and students. It uses a modular architecture, comprising the following components:

- **Frontend**: For user interaction (exam creation & taking)
- **Backend**: For exam management, authentication, question generation and storage, and randomization logic.
- **Database**: For storing users, exams, and results

A key aspect of the system is its reliance on an external API to generate questions. While the core functionality such as exam management and result storage is handled internally, the system requires connectivity to an external service for question generation.

2.1.1 System Interfaces

The software communicates with a third-party Question Generation API over HTTPS to retrieve question sets during exam creation. No other direct system-to-system interfaces are present.

2.1.2 User Interfaces

1. The web-based frontend features a responsive user interface for examiners and students, consistent with modern accessibility standards.
2. Key screens include login/authentication, question paper creation/editing/setup, exam-taking, and results display.
3. Standard web design elements (buttons, menus, forms) ensure intuitive navigation.

2.1.3 Hardware Interfaces

The application is intended for use on standard hardware (desktop, laptop, tablet, smartphone) running up-to-date browsers.

2.1.4 Software interfaces

- Relies on :
  - Supported browsers (e.g., Chrome, Firefox, etc of latest version).
  - Relational database (e.g., Supabase).
  - Question Generation API (eg. OpenAI API, Gemini API,etc).
- Interfaces are via RESTful HTTPS calls with JSON-formatted data.
- Web application frameworks (e.g.,React for frontend, Node.js/Express for backend).

2.1.5 Communications Interfaces

All communication with the external API and between frontend, backend and database is established using HTTP.

2.1.6 Operations

- The system supports user sessions, which are exam creation and conduction.
- The system includes a secure log-in, log-out, and session timeout feature.
- Provides one-time email link to verify user email.
- Provides measures to make sure that one student can log in from only one computer system.

2.2 Product Functions

2.2.1 Examiner Functions

- The application can create MCQ papers by entering topic & difficulty.
- The examiner should be able to preview questions and approve the questions.
- Should be able to allow the students to take the test after approval of questions.
- Should be able to access the student performance reports and analytics.

2.2.2 Student Functions

- Allow students to register and log in to their accounts securely.
- Allow to take any exam online smoothly and associate their account with their performance in the test.
- Provide each candidate with a randomized question set to ensure integrity of the exam.
- Allow each student to check their marks on a test after its completion on the platform.

2.2.3 Functions common to both students and examiners

- The application provides one-time email verification link feature.
- The application provides log-in, log-out, and session timeout feature.
- Each user can log in to their accounts from only one computer system.

2.3 User Characteristics

- **Examiner**: The examiner should be familiar with the exam subjects and basic web usage.
- **Students**: The student is assumed to have basic computer knowledge and can navigate web interfaces.

2.4 Constraints

1. Internet connection is required for the API to be used.
2. Exam sessions expire after a set time provided before the test starts.
3. The randomization algorithm must ensure the integrity of the question.
4. Must support multiple students taking the same test at the same time with minimum latency.
5. The application is designed to run on standard web-enabled devices and should not require any specialized equipment.

2.5 Assumptions and Dependencies

2.5.1 Assumptions

- All users are assumed to have internet access for the product to work.
- The API for question generation is assumed to work for question generation.

2.5.2 Dependencies

- Browsers used must support modern web technologies like React.

3 Specific Requirements

3.1 Functional requirements

User Authentication
F-001 : The system shall allow examiners to register on the website securely.
Sub-level requirements:

F-1.1: The registration form shall include fields for examiner name, email, and password.

F-1.2: The system shall validate that all mandatory fields are filled.

F-1.3: The system shall validate the email format and enforce strong password rules.

F-1.4: The system shall check whether the email is already registered.

F-1.5: The system shall hash and salt the password before storing it in the database.

F-1.6: The system shall confirm registration with a success message or error message.

F-1.7: Send one time link for verifying emails.

Condition	Valid Inputs	Missing Fields	Invalid Email/Password	Email Already Exists
Examiner enters all required details	Yes	No	Yes	Yes
Valid email format	Yes	N/A	No	Yes
Password meets security policy	Yes	N/A	No	Yes
Email already exists in DB	No	N/A	No	No
Action	-	-	-	-
Create account, hash password, store securely	Yes	No	No	No
Show error	No	Yes	Yes	Yes

Table 1: Decision Table for F-1

F-002 : The system shall allow examiners to log in securely into their accounts.
Sub-level requirements:

F-2.1: The login form shall accept email and password as inputs.

F-2.2: The system shall validate that all required fields are filled.

F-2.3:The system shall check whether the email exists in the database.

F-2.4: The system shall compare the entered password with the stored hashed password.

F-2.5: If authentication is successful, the system shall initiate a secure session.

F-2.6: If authentication fails, the system shall display an appropriate error message without revealing details.

F-2.7: Session timeout occurs after a fixed time.

Condition	Case 1: Valid Credentials	Case 2: Invalid Email	Case 3: Wrong Password	Case 4: Missing Fields
Examiner enters email and password	Yes	Yes	Yes	No
Email exists in DB?	Yes	No	Yes	Yes
Password matches stored hash?	Yes	N/A	No	N/A
Action	-	-	-	-
Grant access (login successful)	Yes	No	No	No
Show error: Email not found	No	Yes	No	No
Show error: Incorrect password	No	No	Yes	No
Show error: All fields required	No	No	No	Yes

Table 2: Decision Table for F-2

F-003 : The system shall allow students to log into their accounts securely.
Sub-level requirements:

F-3.1: The login form shall accept student email and password as inputs.

F-3.2: The system shall validate that all required fields are filled.

F-3.3:The system shall check whether the email exists in the database.

F-3.4: The system shall compare the entered password with the stored hashed and salted password.

F-3.5: On successful authentication, the system shall initiate a secure session.

F-3.6: On failed authentication, the system shall display an appropriate generic error message.

F-3.7: Session timeout occurs after a fixed time.

Condition	Case 1: Valid Credentials	Case 2: Invalid Email	Case 3: Wrong Password	Case 4: Missing Fields
Student enters email and password	Yes	Yes	Yes	No
Email exists in DB?	Yes	No	Yes	Yes
Password matches stored hash?	Yes	N/A	No	N/A
Action	-	-	-	-
Grant access, start secure session	Yes	No	No	No
Show error: Email not found	No	Yes	No	No
Show error: Incorrect password	No	No	Yes	No
Show error: All fields required	No	No	No	Yes

Table 3: Decision Table for F-3

F-004 : The system shall allow students to register on the website securely.
Sub-level requirements:

F-4.1: The registration form shall capture essential student details (name, email, password, etc.).

F-4.2: The system shall validate all required fields before submission.

F-4.3: The system shall check if the email is unique and not already registered

F-4.4: The system shall enforce strong password policies (e.g., min length, mix of letters/numbers/symbols).

F-4.5:The system shall hash and salt passwords before storing them in the database.

F-4.6:The system shall use HTTPS for secure data transmission during registration.

F-4.7: Send one-time link to verify emails.

Condition	Case 1: Valid Inputs	Case 2: Email Already Exists	Case 3: Weak Password	Case 4: Missing Fields
Student enters registration details (name, email, password, etc.)	Yes	Yes	Yes	No
Email unique in DB?	Yes	No	Yes	Yes
Password meets complexity policy?	Yes	Yes	No	N/A
Action	-	-	-	-
Create student account, hash & store password	Yes	No	No	No
Show error: "Email already registered"	No	Yes	No	No
Show error: "Password does not meet requirements"	No	No	Yes	No
Show error: "Missing required fields"	No	No	No	Yes

Table 4: Decision Table for F-4

Exam Creation
F-005 : The system shall allow examiners to specify the exam topic and difficulty.
Sub-level requirements:

F-5.1:The system shall provide an input field for the examiner to enter the exam topic.

F-5.2: The system shall provide a drop-down or selection menu for difficulty levels (e.g., Easy, Medium, Hard).

F-5.3: The system shall validate that the topic field is not left empty.

F-5.4: The system shall validate that only allowed difficulty levels are selected.

F-5.5: The system shall send the topic and difficulty to the question generation API.

F-5.6: The system shall display appropriate error messages when inputs are missing or invalid.

Condition	Case 1: Valid Inputs	Case 2: Missing Topic	Case 3: Invalid Difficulty	Case 4: Missing Both
Examiner provides topic?	Yes	No	Yes	No
Examiner selects valid difficulty	Yes	Yes	No	No
Action	-	-	-	-
Proceed to generate MCQs using API	Yes	No	No	No
Show error: "Topic is required"	No	Yes	No	No
Show error: "Invalid difficulty level"	No	No	Yes	No
Show error: "Topic and difficulty required"	No	No	No	Yes

Table 5: Decision Table for F-5 (MCQ Generation)

F-006 : The system shall generate MCQs using API.
Sub-level requirements:

F-6.1:The system shall send exam parameters (topic, difficulty, number of questions, etc.) to the API.

F-6.2: The system shall receive MCQs in a structured format

F-6.3: The system shall validate the API response to ensure MCQs are complete and valid.

F-6.4: The system shall display the generated MCQs to the examiner for review and approval.

F-6.5: The system shall log API errors for debugging and audit purposes.

Condition	Case 1: Valid API Call	Case 2: API Fails	Case 3: Invalid Parameters	Case 4: No Internet
Topic & difficulty provided?	Yes	Yes	No	Yes
API reachable?	Yes	No	Yes	No
API returns valid MCQs?	Yes	No	No	No
Action	-	-	-	-
Display generated MCQs for examiner review	Yes	No	No	No
Show error: "API not reachable"	No	Yes	No	No
Show error: "Invalid input parameters"	No	No	Yes	No
Show error: "No internet connection"	No	No	No	Yes

Table 6: Decision Table for F-6

F-007 : The system shall store generated questions in the database.
Sub-level requirements:

F-7.1:The system shall validate generated questions before saving them.

F-7.2: The system shall store each question with metadata.

F-7.3: The system shall ensure no duplicate questions are stored.

F-7.4: The system shall use secure queries.

F-7.5: The system shall confirm successful storage before making the exam available to students.

Condition	Case 1: Valid Questions	Case 2: Invalid Questions	Case 3: Database Failure	Case 4: Duplicate Question
Questions generated?	Yes	No	Yes	Yes
Database available?	Yes	Yes	No	Yes
Question already exists?	No	–	–	Yes
Action	-	-	-	-
Store questions successfully	Yes	No	No	No
Show error: "Invalid questions"	No	Yes	No	No
Show error: "Database unavailable"	No	No	Yes	No
Avoid duplication (skip or update existing entry)	No	No	No	Yes

Table 7: Decision Table for F-7

Exam Distribution
F-008 : The system shall make the same set of questions available to all students.
Sub-level requirements:

F-8.1:The system shall retrieve the stored set of questions from the database when students start the exam.

F-8.2: The system shall prevent modifications to the exam once it is published..

F-8.3: The system shall ensure exam session synchronization so all students get the same version of the paper.

F-8.4: The system shall ensure availability of questions even under concurrent student logins.

F-8.5: The system shall display an error if the exam is not yet published or session expired.

Condition	Case 1: Exam Published	Case 2: Exam Not Published	Case 3: Database Error	Case 4: Session Expired
Exam exists?	Yes	No	Yes	Yes
Exam published?	Yes	No	Yes	Yes
Database available?	Yes	Yes	No	Yes
Session valid?	Yes	Yes	Yes	No
Action	-	-	-	-
Deliver same set of questions to all students	Yes	No	No	No
Show "Exam not available"	No	Yes	No	No
Show error "Database unavailable"	No	No	Yes	No
Show error "Session expired"	No	No	No	Yes

Table 8: Decision Table for F-8

F-009 : The system shall randomize question order for each student.
Sub-level requirements:

F-9.1:The system shall retrieve the stored set of questions from the database when students start the exam.

F-9.2: The system shall prevent modifications to the exam once it is published..

F-9.3: The system shall ensure exam session synchronization so all students get the same version of the paper.

F-9.4: The system shall ensure availability of questions even under concurrent student logins.

F-9.5: The system shall display an error if the exam is not yet published or session expired.

Condition	Case 1: Exam Published	Case 2: Exam Not Published	Case 3: Randomization Error	Case 4: Session Expired
Exam exists?	Yes	No	Yes	Yes
Exam published?	Yes	No	Yes	Yes
Randomization successful?	Yes	Yes	No	Yes
Session valid?	Yes	Yes	Yes	No
Action	-	-	-	-
Deliver randomized order of questions to student	Yes	No	No	No
Show "Exam not available"	No	Yes	No	No
Deliver default stored order of questions	No	No	Yes	No
Show error "Session expired"	No	No	No	Yes

Table 9: Decision Table for F-9

Taking the Exam
F-010 : The system shall allow students to attempt MCQs online.
Sub-level requirements:

F-10.1: The system shall display MCQs one by one or as a full set, depending on examiner configuration.

F-10.2: The system shall allow students to select or change answers before final submission.

F-10.3: The system shall auto-save responses periodically to prevent data loss.

F-10.4: The system shall handle temporary network failures by caching responses locally and resyncing upon reconnection.

F-10.5: The system shall prevent unauthorized access by ensuring only authenticated students can attempt the exam.

F-10.6: The system shall provide navigation options (Next, Previous, Jump-to-Question) during the attempt.

Condition	Case 1: Exam Available	Case 2: Exam Not Yet Released	Case 3: Exam Expired	Case 4: Network Issue	Case 5: Invalid Session
Exam exists?	Yes	No	Yes	Yes	Yes
Exam active?	Yes	No	No	Yes	Yes
Session valid?	Yes	Yes	Yes	Yes	No
Network stable?	Yes	Yes	Yes	No	Yes
Action	-	-	-	-	-
Allow student to view and attempt MCQs	Yes	No	No	No	No
Show "No available exam"	No	Yes	No	No	No
Show "Exam has ended"	No	No	Yes	No	No
Retry submission / cache responses locally	No	No	No	Yes	No
Redirect to login page	No	No	No	No	Yes

Table 10: Decision Table for F-10

F-011 : The system shall auto-submit the exam when time expires.
Sub-level requirements:

F-11.1: The system shall display a countdown timer throughout the exam.

F-11.2:When the timer reaches zero, the system shall lock the interface and trigger automatic submission.

F-11.3: The system shall ensure all responses (answered/unanswered) are submitted at timeout.

Condition	Case 1: Time Remaining	Case 2: Time = 0 (Exam Over)	Case 3: Student Submits Early	Case 4: Network Issue at Timeout
Timer Running?	Yes	No	Yes	Yes
Time Expired?	No	Yes	No	Yes
Student Manually Submitted?	No	No	Yes	No
Network Stable?	Yes	Yes	Yes	No
Action	-	-	-	-
Continue exam	Yes	No	No	No
Auto-submit responses	No	Yes	No	No
Save responses & confirm submission	No	No	Yes	No
Cache final responses locally and retry until server acknowledges	No	No	No	Yes

Table 11: Decision Table for F-11

Result Processing
F-012 : The system shall automatically evaluate MCQ answers for all students.
Sub-level requirements:

F-12.1: The system shall store the correct answer key for each question in the database.

F-12.2: The system shall compare each student's submitted answer against the stored answer key.

F-12.3: The system shall generate a score immediately after auto-evaluation.

Condition	Case 1: Answer Matches Correct Option	Case 2: Answer Does Not Match	Case 3: Question Unanswered	Case 4: Multiple Attempts (if allowed)
Student Submitted?	Yes	Yes	Yes (but left blank)	Yes
Correct Option Selected?	Yes	No	No	Only last attempt considered
Action	-	-	-	-
Award full marks	Yes	No	No	No
Award 0 marks	No	Yes	Yes	No
Evaluate latest submitted answer	No	No	No	Yes

Table 12: Decision Table for F-12

F-013 : The system shall store and display results to students and examiners. Sub-level requirements:

F-13.1: The system shall save each student's evaluated score in the database after auto-evaluation.

F-13.2: The system shall allow students to view only their own results via a secure login.

F-13.3: The system shall allow examiners to view results of all students for a given exam.

F-13.4: The system shall maintain result history for future reference by both examiners and students.

Condition	Case 1: Answer Matches Correct Option	Case 2: Answer Does Not Match	Case 3: Question Unanswered
Student Submitted?	Yes	Yes	Yes (but left blank)
Correct Option Selected?	Yes	No	No
Action	-	-	-
Award full marks	Yes	No	No
Award 0 marks	No	Yes	Yes
Evaluate latest submitted answer	No	No	No

Table 13: Decision Table for F-12

3.2 External Interfaces

User Interfaces
I-001 : The system shall contain a dashboard for examiners.
I-002 : The system shall contain a suitable test screen for students.
I-003 : The system shall contain a profile section for the students.

Hardware Interfaces
I-004 : Standard computing devices such as laptops,desktop computers, tablets,etc.

Software Interfaces
I-005 : The system shall use API for question generation.
I-006 : The system shall use a database to store questions, students' performance, etc.

Communication Interfaces
I-007 : The system shall use the HTTPS protocol to communicate over the internet.

3.3 Non-functional requirements

Performance
N-001 : The system shall support at least 500 concurrent users without performance degradation.

Security
N-002 : All passwords will be stored in hashed format(along with salting).
N-003 : The system shall use the HTTPS protocol to communicate over the internet.

Usability
N-004 : The system shall have a responsive interface for desktop and mobile devices.

Reliability
N-005 : The system shall recover gracefully from server failures.

3.4 Constraints

C-001 : The exam sessions expire after a set time provided before the test starts.
C-002 : The randomization algorithm must ensure the integrity of the question.
C-003 : The system must support multiple students taking the same test at the same time.

4 Appendix

4.1 Appendix A

Use Case 1: Examiner Creates MCQ Paper
1. Examiner logs into the system
2. Selects "Create New Exam" option
3. Enters topic (e.g., "Data Structures") and difficulty level (e.g., "Medium")
4. System generates MCQs using the API
5. Examiner reviews and approves questions
6. Exam is saved and made available to students

Use Case 2: Student Takes Exam
1. Student logs into the system
2. Selects available exam from dashboard
3. System presents randomized question set
4. Student answers questions within time limit
5. System auto-submits and evaluates responses
6. Student views results after completion

Index