

SOFTWARE DESIGN DOCUMENT

BY G-33

COLLEGE OF COMPUTING AND INFORMATION SCIENCES

INTERNATIONAL EDUCATION SERVICES MATHEMATICS CHALLENGE SYSTEM

DESIGN DOCUMENT FOR THE INTERNATIONAL EDUCATION SERVICES MATHEMATICS CHALLENGE SYSTEM

GitHub Link: <https://github.com/KisomoseArnoldPatrick/Recess.git>

G-33

Supervisor: Mary Nsabagwa

|  |  |  |
| --- | --- | --- |
| Name | Registration Number | Student Number |
| Kisomose Arnold Patrick | 23/U/10079/EVE | 2300710079 |
| Mugisha Modern | 23/U/11587/PS | 2300711587 |
| Kabalebe Joshua | 23/U/27067/PS | 2300727067 |
| Kukiriza Sinai | 23/U/0646 | 2300700646 |
| Tumuhaise Prudence Donald | 23/U/18065/EVE | 2300718065 |

Table of Contents

[1. Introduction 1](#_Toc169809741)

[1.1 Purpose 1](#_Toc169809742)

[1.2 Scope 1](#_Toc169809743)

[1.2.1 Goals and Objectives of the project 1](#_Toc169809744)

[1.2.2 Benefits of the Project 1](#_Toc169809745)

[1.3 Document Overview 1](#_Toc169809746)

[1.4 Reference Material 3](#_Toc169809747)

[1.5 Definitions and Acronyms 3](#_Toc169809748)

[1.5.1 Definitions 3](#_Toc169809749)

[1.5.1 Acronyms 3](#_Toc169809750)

[2. System Overview 4](#_Toc169809751)

[3. System Architecture 6](#_Toc169809752)

[3.1 Architectural Design 6](#_Toc169809753)

[3.2 Decomposition Description 7](#_Toc169809754)

[3.3 Design Rationale 8](#_Toc169809755)

[4. Data Design 10](#_Toc169809756)

[4.1 Data Description 10](#_Toc169809757)

[4.2 ERD 15](#_Toc169809758)

[4.2 Data Dictionary 16](#_Toc169809759)

[5. Component Design 22](#_Toc169809760)

[5.1 Applicant registration algorithm 22](#_Toc169809761)

[5.2 Confirmation algorithm 22](#_Toc169809762)

[5.3 Uploading school algorithm 22](#_Toc169809763)

[5.4 Uploading question bank and answer bank algorithm 23](#_Toc169809764)

[5.4 Uploading a challenge algorithm 23](#_Toc169809765)

[5.5 Challenge registration 23](#_Toc169809766)

[5.6 Attempt challenge algorithm 23](#_Toc169809767)

[5.7 Report algorithm at the close of a challenge 24](#_Toc169809768)

[6. Human Interface Design 25](#_Toc169809769)

[6.1 Overview of User Interface 25](#_Toc169809770)

[6.1.1 Participant Interactivity 25](#_Toc169809771)

[6.1.2 Administrator Interactivity 25](#_Toc169809772)

[6.1.3 School Representative Interactivity 25](#_Toc169809773)

[6.2 Screen Images 25](#_Toc169809774)

[6.3 Screen Objects and Actions 27](#_Toc169809775)

**List of Tables**

[Table 4. 1 Administrator 10](#_Toc169809713)

[Table 4. 2 School 10](#_Toc169809714)

[Table 4. 3 SchoolRepresentative 10](#_Toc169809715)

[Table 4. 4 Participant 10](#_Toc169809716)

[Table 4. 5 Rejected 11](#_Toc169809717)

[Table 4. 6 Answer 11](#_Toc169809718)

[Table 4. 7 Question 12](#_Toc169809719)

[Table 4. 8 Challenge 12](#_Toc169809720)

[Table 4. 9 QuestionBank 12](#_Toc169809721)

[Table 4. 10 AnswerBank 13](#_Toc169809722)

[Table 4. 11 Attempt 13](#_Toc169809723)

[Table 4. 12 AttemptQuestion 13](#_Toc169809724)

[Table 4. 13 ParticipantChallenge 14](#_Toc169809725)

[Table 4. 14 Applicant 14](#_Toc169809726)

[Table 4. 15 Administrator 16](#_Toc169809727)

[Table 4. 16 School 16](#_Toc169809728)

[Table 4. 17 SchoolRepresentative 16](#_Toc169809729)

[Table 4. 18 Participant 16](#_Toc169809730)

[Table 4. 19 Rejected 17](#_Toc169809731)

[Table 4. 20 Answer 17](#_Toc169809732)

[Table 4. 21 Question 18](#_Toc169809733)

[Table 4. 22 Challenge 18](#_Toc169809734)

[Table 4. 23 QuestionBank 18](#_Toc169809735)

[Table 4. 24 AnswerBank 19](#_Toc169809736)

[Table 4. 25 Attempt 19](#_Toc169809737)

[Table 4. 26 AttemptQuestion 19](#_Toc169809738)

[Table 4. 27 ParticipantChallenge 20](#_Toc169809739)

[Table 4. 28 Applicant 20](#_Toc169809740)

[**Table 6. 1 Screen Objects and Actions** 27](#_Toc169809706)

**List of figures**

[Figure 4. 1 ERD of system 15](#_Toc169809697)

[Figure 6. 1 Challenge configuration view 26](#_Toc169780705)

[Figure 6. 2 Analytics View 26](#_Toc169780706)

[Figure 6. 3 Command line interface view 27](#_Toc169780707)

# Introduction

## Purpose

This Software Design Document (SDD) provides a comprehensive architectural blueprint and technical specifications for the development of the International Education Services Mathematics Challenge System.

The system facilitates an international mathematics competition for primary school students across the country, enabling secure registration, challenge management, scoring, analytics, and reporting functionalities.

## Scope

The system will allow interested primary school children to take part in the competition by allowing the participant to access valid challenges, sending a report of answers for all attempted questions to all the participants at the close of the challenge and recognizing the first two winners on the website.

### Goals and Objectives of the project

**System Design**. Architect a scalable and secure system infrastructure that can handle multiple users simultaneously.

**Content Creation**. Develop a comprehensive set of math challenges covering various topics and difficulty levels.

**Testing & Feedback**. Conduct user testing to gather feedback and refine the system accordingly.

**Interface Development**. Design and implement a user-friendly interface with clear navigation and instructions

### Benefits of the Project

Enhanced Problem-Solving Skills.

Improved mathematical proficiency.

## Document Overview

This document has been arranged in chapters that reflect or show the system's requirements.

Chapter 1: Introduction

Identifies the purpose of this SDD and its intended audience.

Chapter 2: System Overview

Gives a general description of the functionality, context and design of the project.

Chapter 3: System Architecture

This is divided into sub-levels:

Chapter 3.1. Architectural Design This shows a modular program structure and explains the relationships between the modules to archive the complete functionality of the system.

Chapter 3.2. Decomposition Description

This provides a decomposition of the subsystems in the architectural design.

Chapter 3.3. Design Rationale

This provides the rationale for selecting the architecture described in 3.1 including critical issues and trade/offs that were considered.

Chapter 4. Data Design

This is also divided into sub-levels;

Chapter 4.1. Data Description

This explains how the information domain of the system is transformed into data structures.

Chapter 4.2. ERD

This provides a blueprint for organizing data structures. Ensuring efficient and effective system design

Chapter 4.3. Data Dictionary

This provides a list of the system entities or major data and their types.

Chapter 5. Component Design

This provides a more detailed view of the systematic operation of each component.

Chapter 6. Human Interface Design

This is divided into sub-levels;

Chapter 6.1. Overview of User

This describes the functionality of the system from the user’s perspective.

Chapter 6.2. Screen Images

This displays screenshots showing the interface from the user’s perspective.

Chapter 6.3. Screen Objects and Actions

This describes the screen objects and actions associated with these objects.

## Reference Material

J. S. van der Ven, A. Jansen, J. Nijhuis, and J. Bosch. Design Decisions: The Bridge between Rationale and Architecture

## Definitions and Acronyms

|  |  |
| --- | --- |
| Entity Relationship Diagram | This provides a visual starting point for the database design. |
| Command line interface | This is a text-based interface used to interact with the system, allowing users to enter commands and perform tasks. |
| Context diagram | This provides a basic sketch that defines an entity based on its scope, boundaries and relation to external components. |
| Data flow diagram | This is a visual representation of how information flows through a process or system. |

### 1.5.1 Definitions

### Acronyms

|  |  |
| --- | --- |
| SDD | Software Design Document |
| CLI | Command Line Interface |
| ERD | Entity Relationship Diagram |

# System Overview

In the pursuit of enhancing mathematical education, it has become increasingly evident that traditional teaching methods are not fully equipped to address the needs of today’s pupils. The advent of digital technology presents an opportunity to revolutionize how we approach math learning. Our math mathematics Challenge System is designed to bridge the gap between conventional practices and modern educational demands.

The International Educational Services Mathematics Challenge System is a comprehensive platform designed to facilitate and manage mathematics competitions for primary school pupils across the country. The system offers a range of functionalities to cater to the needs of administrators, school representatives, and participants (pupils).

A registered administrator registers and manages schools participating in the competition. An administrator uploads a school into the system indicating the name, district, school registration number, email and name of the representative. The school representative is validated before being registered into the system.

An administrator uploads questions and answers into the web system. If a challenge is to consist of 10 questions, 100 questions should be uploaded such that for each attempt the 10 questions are selected randomly from the 100 questions. Questions are picked from an Excel document while the answers with accompanying marks are also picked from another Excel document.

To open a competition, an administrator sets challenge parameters including the date when the challenge will be opened and closed, the duration of the challenge, and how many questions will be presented.

For a pupil to participate in the competition, they register using a command line interface. The pupil is presented with a menu of instructions to register their details (first name, last name, user name, password, email address, image, date of birth, and school registration number). If the school registration number does not match the registered school numbers, the pupil is informed. If there is a match, the record is added to a file and an email notification is sent to the respective school representative reminding them to confirm the applicant.

School representatives log into the system via the command line interface to confirm the newly registered prospective participants via a menu item “viewApplicants”. The menu displays all applicants with their registration numbers. To reject or activate an applicant, the representative enters a menu item “confirm yes/no username”. If one is rejected, they are sent an email, deleted from the file and moved into a database table rejected. Those that are accepted are sent an email, deleted from the file and inserted into the participant table in the database. If one tries to register again under the same school after being rejected, they are informed on the command line interface and denied from registering under the school.

Accepted applicants can log in and see details of challenges using “viewChallenges” command. If the challenge is still valid, the participant can issue a command “attemptChallenge challengeNumber” to start the challenge. A participant has a maximum of three chances to attempt a challenge and each attempt presents random questions. The questions are presented one by one and each time a question is presented, the number of remaining questions and time are indicated above the question. If a wrong answer is given by a participant, 3 marks are deducted, while if a participant is not sure, they enter a negative or - and are awarded 0 for that question. Else, the participant gets the marks that were assigned to that particular question. When the time for attempting the question expires, the participant challenge is closed and the participant is given their score and report. The report shows their scores and time taken, for each attempted question and the total time they have taken to complete the challenge.

At the close of the challenge, a report of answers for all attempted questions is sent to all participants and the first two winners are recognized on the website. Each participant is automatically sent an email providing them with a PDF report of the right answers.

The web system also gives analytics including school rankings, the performance of schools and participants over the years, the most correctly answered questions, Percentage repetition of questions for a given participant across attempts, a list of worst performing schools for a given challenge, a list of participants with incomplete challenges, list of best performing schools for all challenges.

# System Architecture

## Architectural Design

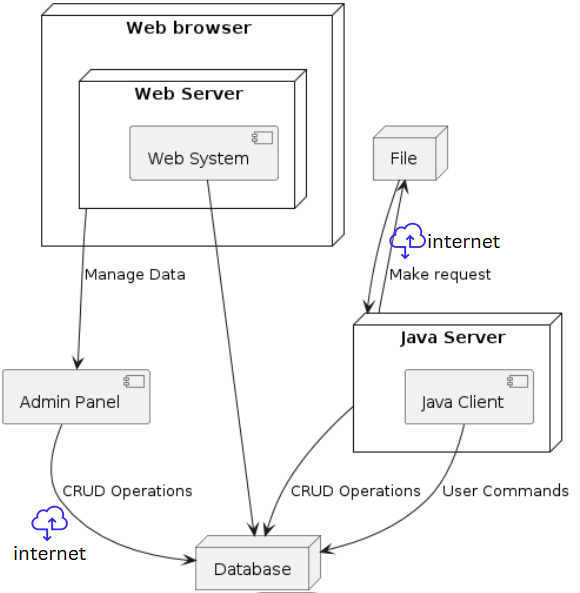


Figure 3. 1 System Architectural Design

Figure 3.1 shows the interaction between system components.

The system follows a robust and scalable three-tier web architecture, designed to ensure efficient management, security, and high performance for the international mathematics competition. The architecture comprises the following key components:

1. **Web Browser**: The primary user interface for administrators, enabling them to register schools participating in the challenge, configure challenges, upload questions and answers, and access analytics and reports through a web-based platform.
2. **Client**: This component represents the Command-Line Interface (CLI) used by school representatives and participants. School representatives can confirm or reject participant registrations, while participants can register, view challenges, attempt questions, and access their performance reports through the CLI.
3. **Web Server**: This server acts as the intermediary between the web browser and the application server, facilitating secure communication and delivering the web-based interface to administrators.
4. **Java Server**: The application server, implemented using Java technologies, serves as the backbone of the system. It handles core functionalities such as challenge management, registration processes, scoring algorithms, analytics generation, and report generation. This server interacts with the web server and the database server to process requests and deliver the required functionality.
5. **File**: This stores the participants’ details, the questions and answer files and also the image data which can be retrieved through the web browser or CLI.
6. **Database Server**: The database server houses the system's data, including schools, participants, challenge configurations, responses, and analytics. It provides a secure and reliable data storage solution, enabling efficient data management and retrieval for various system operations.

## Decomposition Description

The decomposition of the system's architecture into distinct components promotes modularity and facilitates independent development and maintenance. Each component can be designed, implemented, and tested independently, reducing the complexity of the overall system and enabling parallel development efforts.

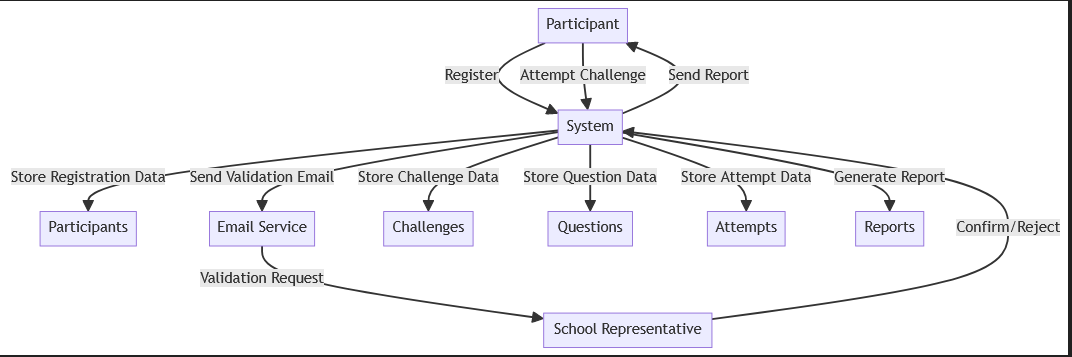
 Figure 3. 2 Data flow diagram

Figure 3.2 shows the flow of data within the International Education Services Mathematics Challenge System, illustrating the processes, data stores and external entities involved.

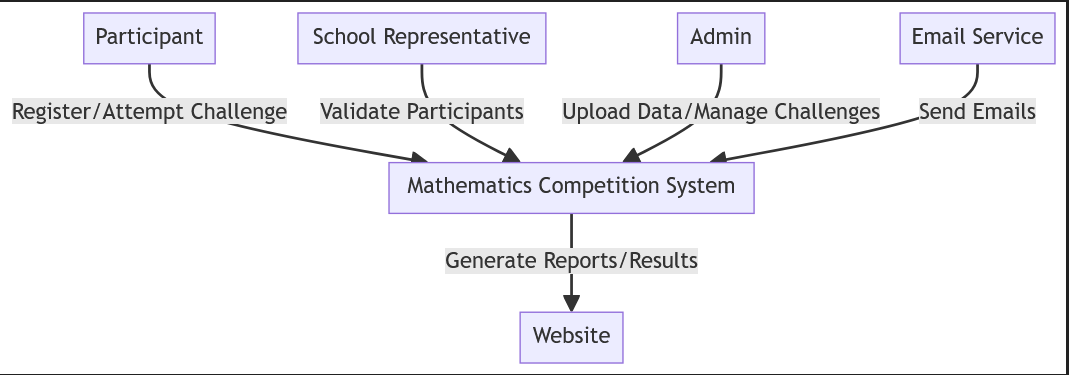
 Figure 3. 3 Context diagram

Figure 3.3 provides a high-level view of the International Education Services Mathematics Challenge System, illustrating the entities interacting with the system and the data flows between them.

## Design Rationale

The architecture follows a clear separation of concerns, with each tier responsible for specific tasks, ensuring modular development, scalability, and maintainability. The presentation tier (web browser and CLI) handles user interactions, the application tier (Java server) manages the business logic and data processing, and the data tier (file and database server) manages data storage and retrieval.

This three-tier architecture allows for independent development and deployment of each component, facilitating parallel development and easier maintenance. Additionally, the architecture supports horizontal and vertical scaling, ensuring the system can handle increasing workloads and user demand by adding more servers or enhancing the capacity of existing servers.

The system's architecture is designed to provide a secure, efficient, and user-friendly experience for administrators, school representatives, and participants throughout the entire competition process, promoting mathematical proficiency and problem-solving skills among primary school students nationwide.

# Data Design

## Data Description

Table 4. 1 Administrator

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| adminID | Unique identifier of an administrator |
| adminName | Name of the administrator |
| userName | The login name for the administrator used to access the system |
| password | Encrypted code used to permit system access |

Table 4. 2 School

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| schoolName | Name of the school |
| schoolRegNo | Unique identifier of a school |
| district | District of school |
| schoolRepID | References to the school representative a school has |
| adminID | References to the administrator who uploaded the school |

Table 4. 3 SchoolRepresentative

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| schoolRepID | Unique identifier of a school representative |
| userName | The login name the school representative uses to access the system |
| password | Encrypted code used to permit system access |
| emailAddress | Email of the representative |
| repName | Name of the representative |

Table 4. 4 Participant

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| participantID | Unique identifier of a participant |
| applicantID | Unique identifier given to the pupil after registration |
| firstName | First name of a participant |
| lastName | Last name of a participant |
| emailAddress | Email of a participant |
| dateOfBirth | Date of birth of a participant |
| schoolRegNo | Identifies the school the participant belongs to |
| userName | The login name used by a participant to access the system |
| imagePath | Image of the participant |
| password | Encrypted code used to permit system access |
| schoolRepID | References the school representative who confirmed the participant |

Table 4. 5 Rejected

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| rejectedID | Unique identifier of a rejected applicant |
| schoolRegNo | Identifies the school the applicant belongs to |
| emailAddress | Email of the applicant |
| applicantID | Unique identifier given to the pupil after registration |
| userName | The login names the applicant filled in during registration |
| imagePath | Image of the applicant |
| schoolRepID | References the school representative who rejected the applicant |
| firstName | First name of the applicant |
| lastName | Last name of the applicant |
| password | Encrypted code the applicant filled in during registration |
| dateOfBirth | Date of birth of the applicant |

Table 4. 6 Answer

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| answerNo | Unique identifier of an answer |
| answer | Solution of a question |
| marksAwarded | Marks awarded if a participant gives the answer |
| answerBankID | References to the answer bank the answer belongs to |

Table 4. 7 Question

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| questionNo | Unique identifier of a question |
| question | Represents set question |
| questionBankID | References to the question bank the question belongs to |
| answerNo | References the answer corresponding to a question |

Table 4. 8 Challenge

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| challengeNo | Unique identifier of a challenge |
| attemptDuration | Time to do the challenge |
| noOfQuestions | Number of questions to be answered in the challenge |
| overallMark | Total mark of the answer marks |
| openDate | Start date of the challenge |
| closeDate | Close date of the challenge |
| adminID | References to the administrator who uploaded the challenge |

Table 4. 9 QuestionBank

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| questionBankID | Unique identifier of the question file |
| name | Name of the file with questions |
| challengeNo | References the challenge which uses the question bank |
| adminID | References to the administrator who uploaded the question file |

Table 4. 10 AnswerBank

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| answerBankID | Unique identifier of the answer file |
| name | Name of the file with answers |
| adminID | References to the administrator who uploaded the answer file |

Table 4. 11 Attempt

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| attemptID | Unique identifier of an attempt |
| startTime | Time the attempt began |
| endTime | Time the attempt ended |
| score | The mark a participant obtained in an attempt |
| percentageMark | The percentage mark a participant obtained in an attempt |
| participantID | References a participant to the attempt |
| challengeNo | References to the challenge being undertaken by the registered participant |

Table 4. 12 AttemptQuestion

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| attemptID | Unique identifier of an attempt |
| questionNo | References a question to the attempt |
| wrong | Represents if the participant gave a wrong answer |
| correct | Represents if the participant gave a correct answer |
| negative | Represents if the participant gave a negative as an answer |
| givenAnswer | Answer given by a participant |

Table 4. 13 ParticipantChallenge

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| challengeNo | References to the challenge being undertaken by the registered participant |
| participantID | References to the registered participant undertaking the challenge |

Table 4. 14 Applicant

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| applicantID | Unique identifier of an applicant |
| schoolRegNo | Identifies the school the applicant belongs to |
| emailAddress | Email of the applicant |
| userName | The login names the applicant filled in during registration |
| imagePath | Image of the applicant |
| firstName | First name of the applicant |
| lastName | Last name of the applicant |
| password | Encrypted code the applicant filled in during registration |
| dateOfBirth | Date of birth of the applicant |

## 4.2 ERD

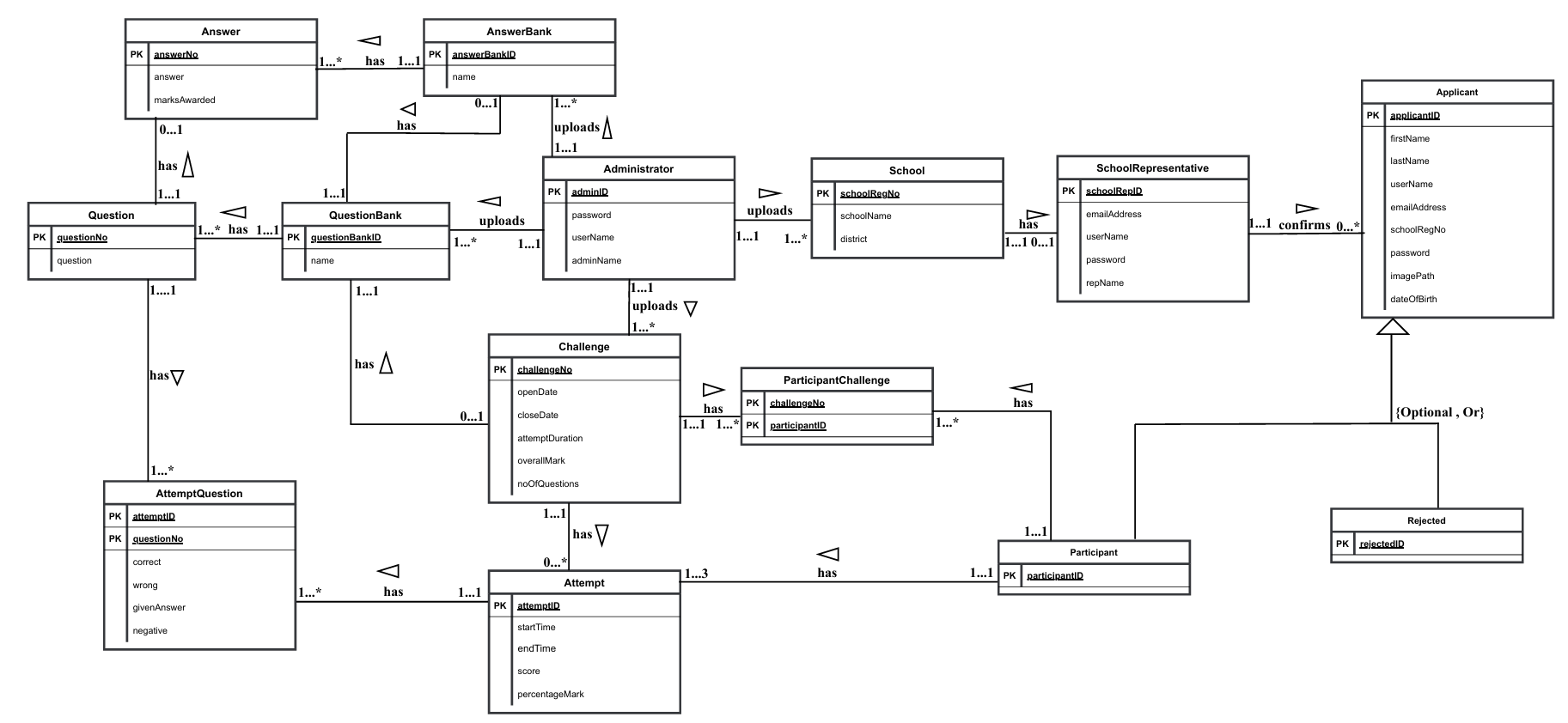


Figure 4. 1 ERD of system

Figure 4.1 shows an ERD of the system that provides a blueprint for organizing data structures.

## Data Dictionary

Table 4. 15 Administrator

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Size** | **Constraint** |
| adminID | INT |  | PRIMARY KEY |
| adminName | VARCHAR | 20 | NOT NULL |
| userName | VARCHAR | 20 | NOT NULL |
| password | VARCHAR | 50 | NOT NULL |

Table 4. 16 School

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Size** | **Constraint** |
| schoolName | VARCHAR | 30 | NOT NULL |
| schoolRegNo | INT |  | PRIMARY KEY |
| district | VARCHAR | 15 | NOT NULL |
| schoolRepID | INT |  | NOT NULL, FOREIGN KEY |
| adminID | INT |  | NOT NULL, FOREIGN KEY |

Table 4. 17 SchoolRepresentative

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Size** | **Constraint** |
| schoolRepID | INT |  | PRIMARY KEY |
| userName | VARCHAR | 20 | NOT NULL |
| password | VARCHAR | 50 | NOT NULL |
| emailAddress | VARCHAR | 30 | NOT NULL, UNIQUE |
| repName | VARCHAR |  | NOT NULL |

Table 4. 18 Participant

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Size** | **Constraint** |
| participantID | INT |  | PRIMARY KEY |
| applicantID | INT |  | NOT NULL, UNIQUE |
| firstName | VARCHAR | 10 | NOT NULL |
| lastName | VARCHAR | 15 | NOT NULL |
| emailAddress | VARCHAR | 30 | NOT NULL, UNIQUE |
| dateOfBirth | DATE | 10 | NOT NULL |
| schoolRegNo | INT |  | NOT NULL |
| userName | VARCHAR | 20 | NOT NULL |
| imagePath | VARCHAR | 10 | NOT NULL |
| password | VARCHAR | 50 | NOT NULL |
| schoolRepID | INT |  | NOT NULL, FOREIGN KEY |

Table 4. 19 Rejected

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Size** | **Constraint** |
| rejectedID | INT |  | PRIMARY KEY |
| schoolRegNo | INT |  | NOT NULL |
| emailAddress | VARCHAR | 30 | NOT NULL, UNIQUE |
| applicantID | INT |  | NOT NULL, UNIQUE |
| userName | VARCHAR | 20 | NOT NULL |
| imagePath | VARCHAR | 10 | NOT NULL |
| schoolRepID | INT |  | NOT NULL |
| firstName | VARCHAR | 10 | NOT NULL |
| lastName | VARCHAR | 15 | NOT NULL |
| password | VARCHAR | 50 | NOT NULL |
| dateOfBirth | DATE |  | NOT NULL |

Table 4. 20 Answer

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Size** | **Constraint** |
| answerNo | INT |  | PRIMARY KEY |
| answer | TEXT |  | NOT NULL |
| marksAwarded | INT |  | NOT NULL |
| answerBankID | INT |  | FOREIGN KEY |

Table 4. 21 Question

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Size** | **Constraint** |
| questionNo | INT |  | PRIMARY KEY |
| question | TEXT |  | NOT NULL |
| questionBankID | INT |  | FOREIGN KEY |
| answerNo | INT |  | FOREIGN NUMBER |

Table 4. 22 Challenge

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Size** | **Constraint** |
| challengeNo | INT |  | PRIMARY KEY |
| attemptDuration | TIME |  | NOT NULL |
| noOfQuestions | INT |  | NOT NULL |
| overallMark | INT |  | NOT NULL |
| openDate | DATE |  | NOT NULL |
| closeDate | DATE |  | NOT NULL |
| adminID | INT |  | FOREIGN KEY |

Table 4. 23 QuestionBank

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Size** | **Constraint** |
| questionBankID | INT |  | PRIMARY KEY |
| name | VARCHAR | 10 | NOT NULL |
| challengeNo | INT |  | NOT NULL, FOREIGN KEY |
| adminID | INT |  | NOT NULL, FOREIGN KEY |

Table 4. 24 AnswerBank

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Size** | **Constraint** |
| answerBankID | INT |  | PRIMARY KEY |
| name | VARCHAR | 10 | NOT NULL |
| adminID | INT |  | FOREIGN KEY |

Table 4. 25 Attempt

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Size** | **Constraint** |
| attemptID | INT |  | PRIMARY KEY |
| startTime | TIME |  | NOT NULL |
| endTime | TIME |  | NOT NULL |
| score | INT |  | NOT NULL |
| percentageMark | DEC |  | NOT NULL |
| participantID | INT |  | NOT NULL, FOREIGN KEY |
| challengeNo | INT |  | NOT NULL, FOREIGN KEY |

Table 4. 26 AttemptQuestion

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Size** | **Constraint** |
| attemptID | INT |  | PRIMARY KEY, FOREIGN KEY |
| questionNo | INT |  | PRIMARY KEY, FOREIGN KEY |
| wrong | BOOLEAN |  | NOT NULL |
| correct | BOOLEAN |  | NOT NULL |
| negative | BOOLEAN |  | NOT NULL |
| givenAnswer | TEXT |  | NULL |

Table 4. 27 ParticipantChallenge

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Size** | **Constraint** |
| challengeNo | INT |  | PRIMARY KEY, FOREIGN KEY |
| participantID | INT |  | PRIMARY KEY, FOREIGN KEY |

Table 4. 28 Applicant

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Size** | **Constraint** |
| applicantID {PK} | INT |  | PRIMARY KEY |
| schoolRegNo | INT |  | NOT NULL |
| emailAddress | INT |  | PRIMARY KEY |
| userName | VARCHAR | 20 | NOT NULL |
| imagePath | VARCHAR | 10 | NOT NULL |
| firstName | VARCHAR | 10 | NOT NULL |
| lastName | VARCHAR | 15 | NOT NULL |
| password | VARCHAR | 50 | NOT NULL |
| dateOfBirth | DATE |  | NOT NULL |

# Component Design

## 5.1 Applicant registration algorithm

Select the “register” option.

Input the first name, last name, email address, school registration number, date of birth, image, user name and password details.

If the applicant's school registration number matches a school registration number in the school table in the database, store the registration details in a pending file and send an email to the school representative for confirmation.

Else

Show error message.

End.

## 5.2 Confirmation algorithm

The school representative selects the “login” option.

Select the “view applicants” option.

If the school representative confirms, the applicant is deleted from the file and inserted into the participant table in the database.

Else.

The applicant is deleted from the file and inserted into the rejected table in the database.

Send an email to the applicant.

End.

## 5.3 Uploading school algorithm

The administrator clicks the login button via the web browser.

Click the “Manage schools” button.

Fill out a form with school details.

Click the submit button.

End.

## 5.4 Uploading question bank and answer bank algorithm

The administrator clicks the login button via the web browser.

Click the “Upload questions and answers” button.

Upload the question Excel file and answer Excel file.

End.

## 5.4 Uploading a challenge algorithm

The administrator clicks the login button via the web browser.

Click the “Set Challenge” button.

Set challenge parameters (opening date, closing date, question bank and answer bank to be used, number of questions to be presented).

End.

## 5.5 Challenge registration

The participant selects the “login” option.

Select the “View challenges” option in the menu item.

Select the “Participate” option for the interested challenge.

Send an email to the participant.

End.

## 5.6 Attempt challenge algorithm

The participant selects the “login” option.

Select the “Attempt Challenge” option in the menu item and provide the challenge number.

Answering questions one by one.

Close an attempt at the end of answering questions or the time duration.

Calculate the score and time taken.

Score = (Correct Answers\*Question Marks) - (Wrong Answers\*3)

Time taken = end time – start time

Send a report to the participant’s email that shows the score and time taken for each attempted question and the total time taken they have taken to complete the challenge.

End.

## 5.7 Report algorithm at the close of a challenge

Send a report showing the answers to all attempted questions to a participant via email.

Recognize the first two winners on the website.

End.

# Human Interface Design

## Overview of User Interface

### Participant Interactivity

The participant logs in to the system via the command line interface, views valid challenges, chooses what challenge to participate in and attempts the challenge.

### Administrator Interactivity

An administrator logs in to the system via the web browser, uploads schools, question and answer banks to be used by challenges. An administrator also sets challenge parameters and can view challenge analytics.

### School Representative Interactivity

A school representative logs in to the system via the command line interface, views applicants and confirms or rejects applicants.

## Screen Images

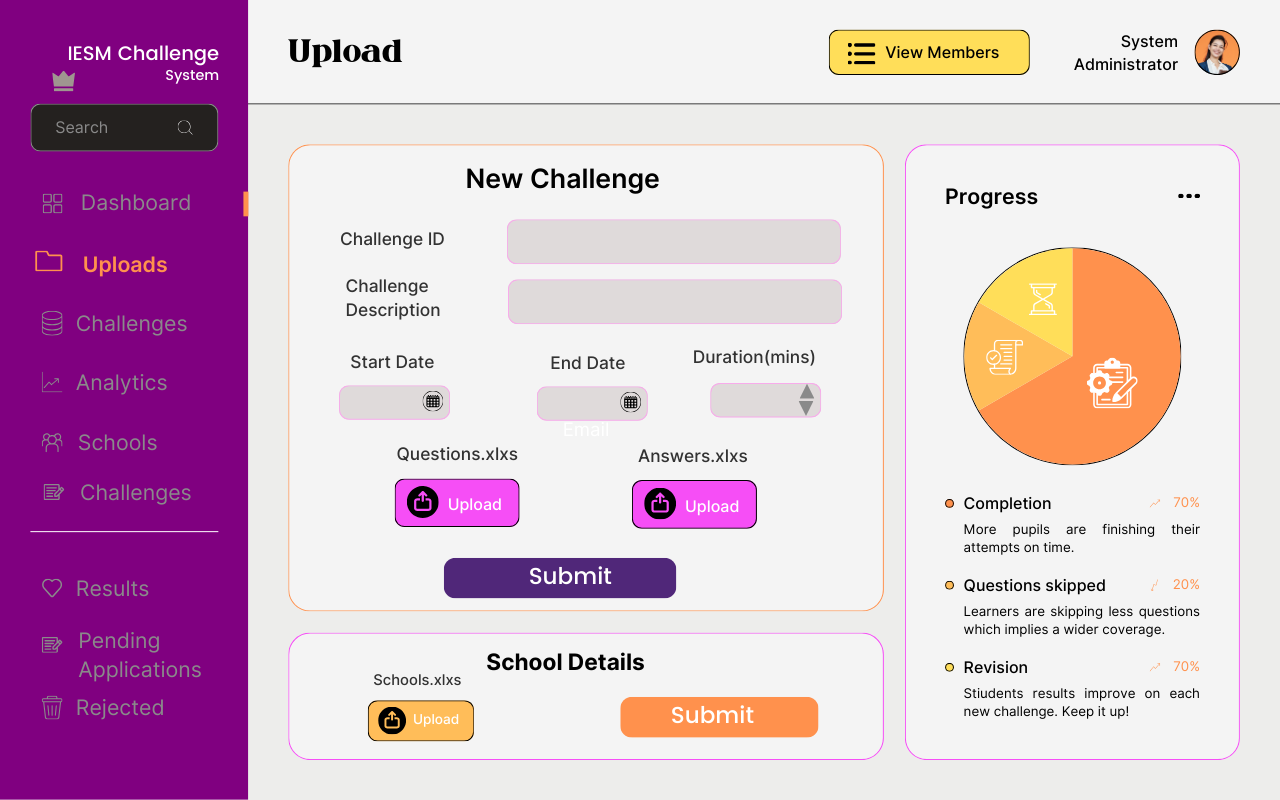
 Figure 6. 1 Challenge configuration view

Figure 6.1 shows how an administrator sets the parameters of a challenge.

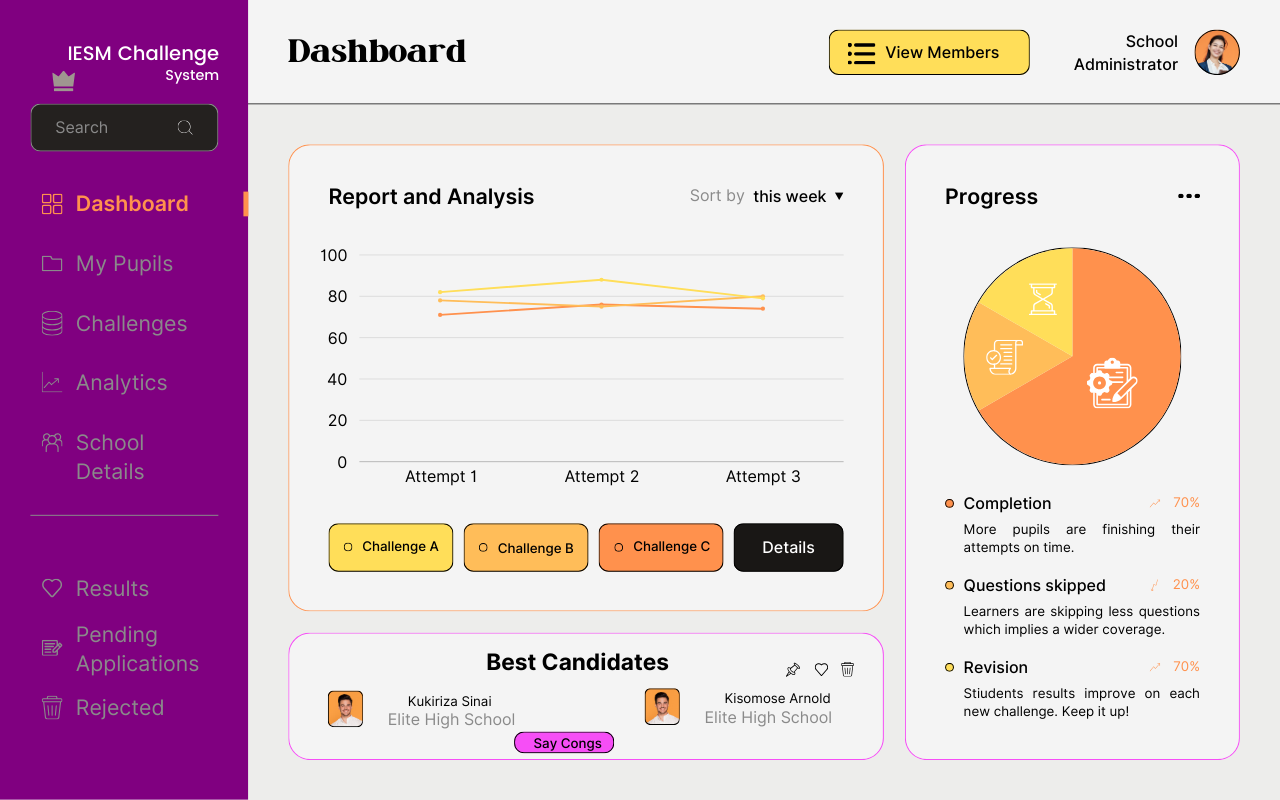
****

Figure 6. 2 Analytics View

Figure 6.2 shows how an administrator can view challenge analytics such as best candidates, school rankings, and worst performing schools for a given challenge.

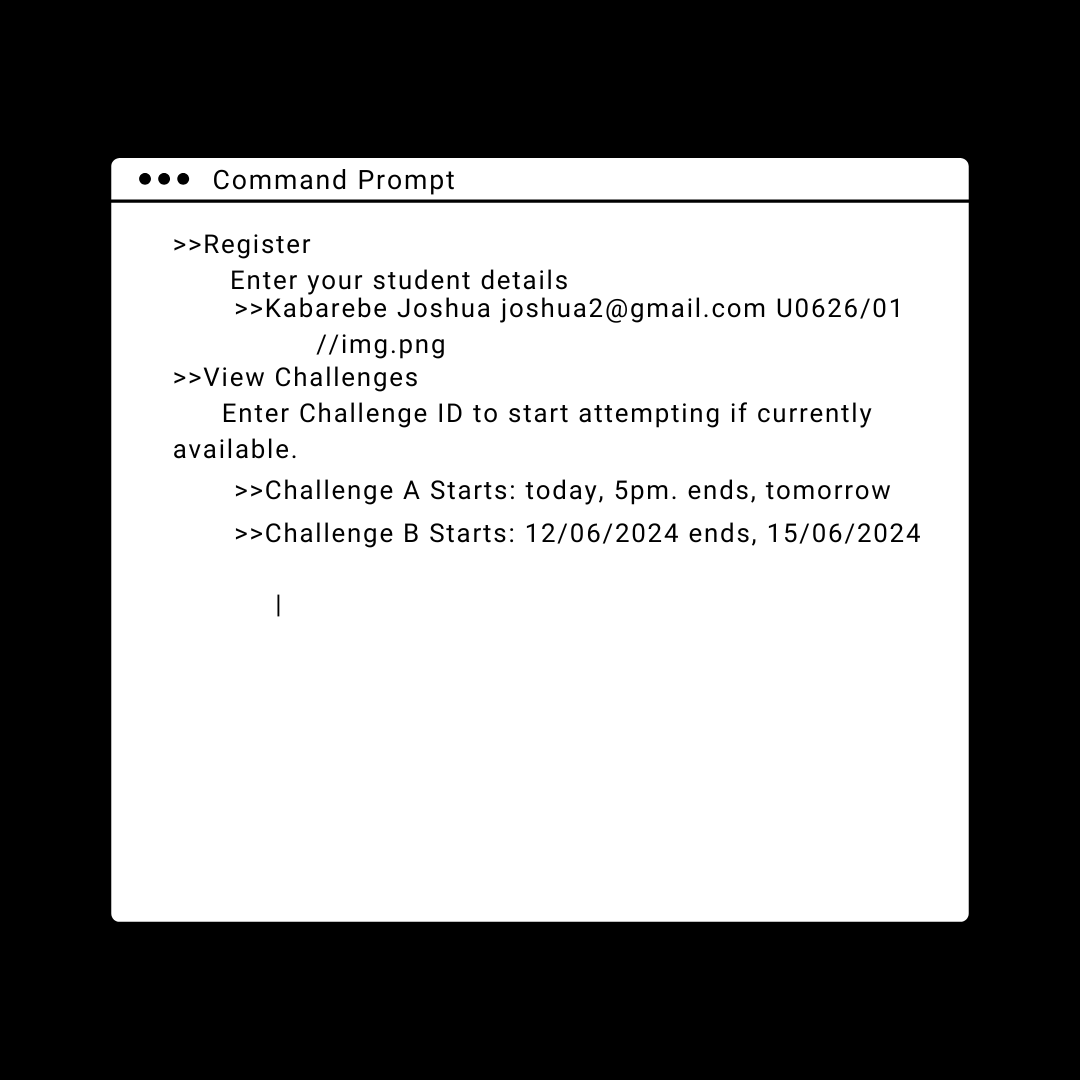


Figure 6. 3 Command line interface view

Figure 6.3 shows how a participant logs in via a command line interface and views valid challenges.

## Screen Objects and Actions

**Table 6. 1 Screen Objects and Actions**

|  |  |  |
| --- | --- | --- |
| **Screen object** | **Action** | **Result** |
| “Manage schools” button | Click | A form with school details pops up |
| “Upload questions and answers” button | Click | A file upload area appears |
| “Set Challenge” button | Click | A form with challenge parameters appears |
| “Sign up” button | Click | A registration form pops up |
| “Sign in” button | Click | A login form pops up |
| “Log in” button | Click | Validation of the login details and then allowing the administrator to access the system if the details are valid |
| “Reports and Analytics” button | Click | An administrator can view various reports (e.g. school rankings, performance over time) |
| Submit button | Click | Input data is submitted to the server for validation and then stored in the database |