



Project Report

A Quiz & Practice Test Platform

Course Code: CSE479

Course Title: Web Programming

Section: 1

Semester: Summer-2025

Group - 8

Prepared By

Name	ID
Sheikh sazzaduzzaman	2021-3-60-264
Md. Bayzid Ohin	2021-3-60-288
Md Shahbaz khan	2022-1-60-140

Submitted To

Nisat Tasnim Niloy

Lecturer

Department of Computer Science & EngineeringEngineering

Submitted on : 02 August, 2025

Table Of Contents

1. Abstract	2
2. Introduction	2
3. Literature Review	3
4. System Design	4
5. Methodology	5-6
6. Implementation.....	7
7. Graphical Models.....	8-11
8. Target Technology	11
9. Results & Analysis.....	11
10. Output	12-18
11. Discussion.....	19
11. Team Responsibility.....	21
11. Future Work.....	22
12. Conclusion	22

Abstract

Education is rapidly shifting towards the online platform through interactive learning tools in order to enhance the interaction of the students and keep track of their performance. The traditional assessment methods are time consuming, there are no real time results or monitoring of performance. This project generates a web-based quiz management system that enables instructors to create, deliver and publish quizzes and students to take quizzes, receive real-time grading and review their progress through leader boards.

It is described as a system of secure authentication, real-time automatic grading, quiz classification, analytics dashboards, randomized questions, and leaderboards. It is built around HTML, CSS, JavaScript (frontend), PHP (backend), and MySQL (database) and runs on an Apache server.

The prototype will serve to indicate how interactive online testing can improve learning, reduce the workload of instructors and provide students with an engaging experience. Scalable architecture can be supported with the system and hundreds of classes and multiple institutions.

Introduction

Background

Assessment of students is common through the use of quizzes and tests. However, conventional classroom quizzes are not mobile, must be graded manually and lack real-time feedback. With the increasing popularity of online education, online quiz solutions that offer automation, objectivity and functionalities to users are becoming highly sought after.

Problem Statement

Instructors are faced with the challenge of managing different quizzes, maintaining assessments integrity and providing feedback in a time-sensitive manner. On the other hand, the students need the tools that support them to track their improvements, compete on real-time leaderboards and improve their learning experience with immediate feedback.

Objectives

This project intends to:

1. Develop a web-based quiz manager that is safe and scalable.
2. Permission of automatic grading and automatic feedback of results.
3. Dashboard performance monitoring and design leaderboards.
4. Allow quizzes to be categorized, randomized and time-limited.

Literature Review

Various online quiz and assessment systems have become popular in the past few years in both academics and business training environments. Despite the fact, that these tools have proven to be useful, they also have specific disadvantages which restrict their application in structured classroom settings.

Google Form : Google Forms is a readily available tool which allows educators to make quizzes with very little effort. It also lacks high-level interactive elements such as leaderboards, performance analytics or randomized question banks. Therefore, it can be used in simple surveys and simple tests as opposed to the comprehensive academic analysis.

Kahoot : Kahoot is popular due to the gamified, quiz based format that allows students to be actively engaged. However, it is quite effective in improving motivation, but Kahoot is more of an entertainment tool, not offering the same level of management, grading, and academic reporting required in formal education.

Instead, the new system will attempt to balance between functionality and ease of use. It aims to offer the necessary features such as secure authentication, automated grading, leaderboards and analytical dashboards without the bigger LMS platforms and their accompanying scale and complexity. The teachers find it easy and the students find it amusing at the same time keep the standards in academics.

System Design

System Architecture

Actors:

- **Instructor:** Creates, edits, and publishes quizzes.
- **Student:** Takes quizzes, views results, checks leaderboard.
- **System:** Handles authentication, auto-grading, and analytics.

Core Components:

1. **Authentication Module** - Secure registration/Login with role-based access.
2. **Quiz Management Module**- Quiz creation, publishing, time limits.
3. **Auto-Grading & Result Module** - Real time assessment and storage of results.
4. **Leaderboard Module**- Places students by score and time.
5. **Analytics Module** – Calculations of the performance trends.

Database Design (ER Model)

Entities:

- **User:** user_id (PK), name, email, role, password.
- **Quiz:** quiz_id (PK), title, time_limit, category, is_published.
- **Question:** question_id (PK), quiz_id (FK), question_text, options, correct_answer.
- **Result:** result_id (PK), user_id (FK), quiz_id (FK), score, completion_time.
- **Leaderboard:** Tracks rankings by score and time.

Methodology

Web-based Quiz Management System is designed in a systematic manner and uses modern web development standards, security, and extensive testing protocols. There are various important components of the system architecture such as the frontend, the backend, the database, the security protocols, and the testing systems.

Frontend Development

The HTML, CSS, and JavaScript frontend of the system is simple to use and responsive to both teachers and students. The design is developed to be not only intuitive but also enables instructors to easily build and manage quizzes and students to browse quizzes and review their results without challenges. Its frontend is built using modern design ideas as it has the following:

- Responsive design to ensure that it can be accessed through different devices including desktops, tablets, and mobile phones.
- Interviews use of such elements, as instant feedback after submitting quizzes, the way results and leaderboards are shown.

Backend Development

The back end is developed using the PHP programming language, which provides the main business logic of the system. It handles user authentication, quizzes, grading automatically and it can generate dynamic contents. Secure API endpoint is used to communicate between the frontend and the database in the system. The backend does critical tasks such as:

- Role based access control and user authentication to differentiate between an instructor and a student.
- Quiz management, such as creation, editing, publication, and classification of quizzes.
- Auto grading and live calculation of scores, which gives immediate feedback to the students.

Database Management

The system has MySQL as its database manager. It has been created to have a database that can store and handle important data such as user accounts, quiz information, student answers and leaderboard information. The schema is high query performance and data integrity considered. The database format is able to support:

- Data about users, including usernames, user roles (instructor or student) and hashed passwords to enable secure authentication are stored.
- Quiz data, quiz IDs, time limits, category, and questions on that quiz.
- Leaderboard entries such as student scores, results, time taken to complete and rank information.

Security Measures

Security is also considered in the design of the system, as the system will help protect user information and facilitate safe communication. The above measures are taken to protect the integrity and confidentiality of the system:

- Encryption user passwords with industry-standard algorithms to store user passwords in a secure way.
- Session management to control the state of user login to avoid unauthorized access to sensitive system assets.
- Input verification to avoid SQL injection, XSS attacks and other typical security vulnerabilities. Any user input (instructors and students alike) is verified on client and server side to ensure that only secure information is handled.

Testing

The system is subjected to very serious functional tests so that everything is operating properly. Key areas of testing include:

- Submission and grading of the quiz to have confidence that the auto-grading logic is proper and accurate.
- Leader board facility to ensure that students are ranked correctly on the basis of their scores and time taken to complete.
- Testing between cross-browsers and devices to make sure that the application works reliably on other platforms.
- Load testing to test the performance of the system with different user loads, and make sure that it can support more users.

Implementation

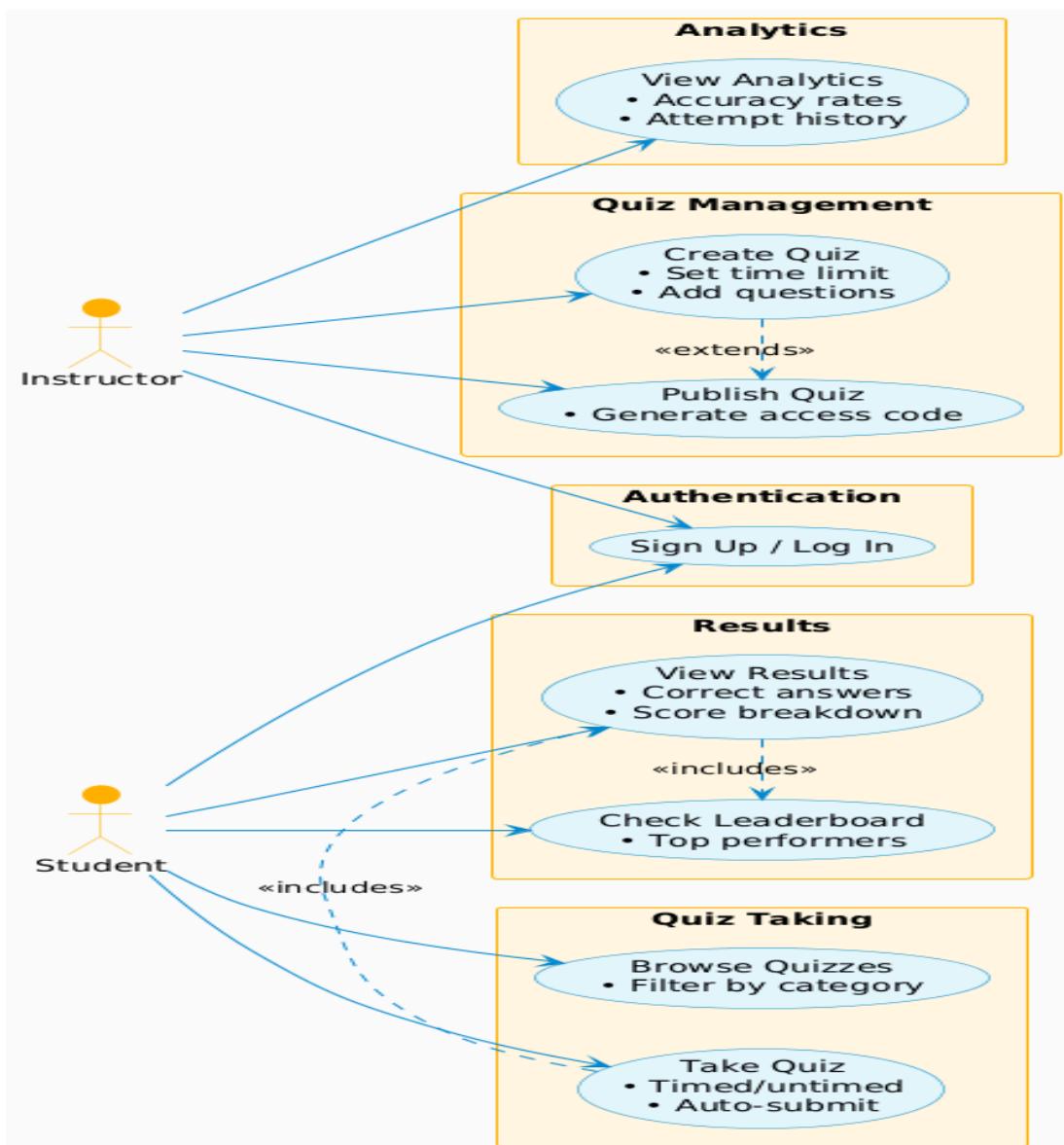
- **Authentication:** User-Role based user-login system (Instructor/Student).
- **Quiz Creation:** The instructor can use the dashboard to edit quiz, question, time restrictions, and publishing settings.
- **Quiz taking:** In quiz taking, students type in an access code to take the quizzes and are required to answer the questions until the time elapses.
- **Auto-Grading:** Responses are scored at the PHP server.
- **Leaderboard:** Leaderboard after each attempt of quiz is real-time as well.
- **Analytics:** It should display trends of average scores and student progress in charts.

Graphical Models

A. Use Case Diagram :

List of Primary Users / Actors:

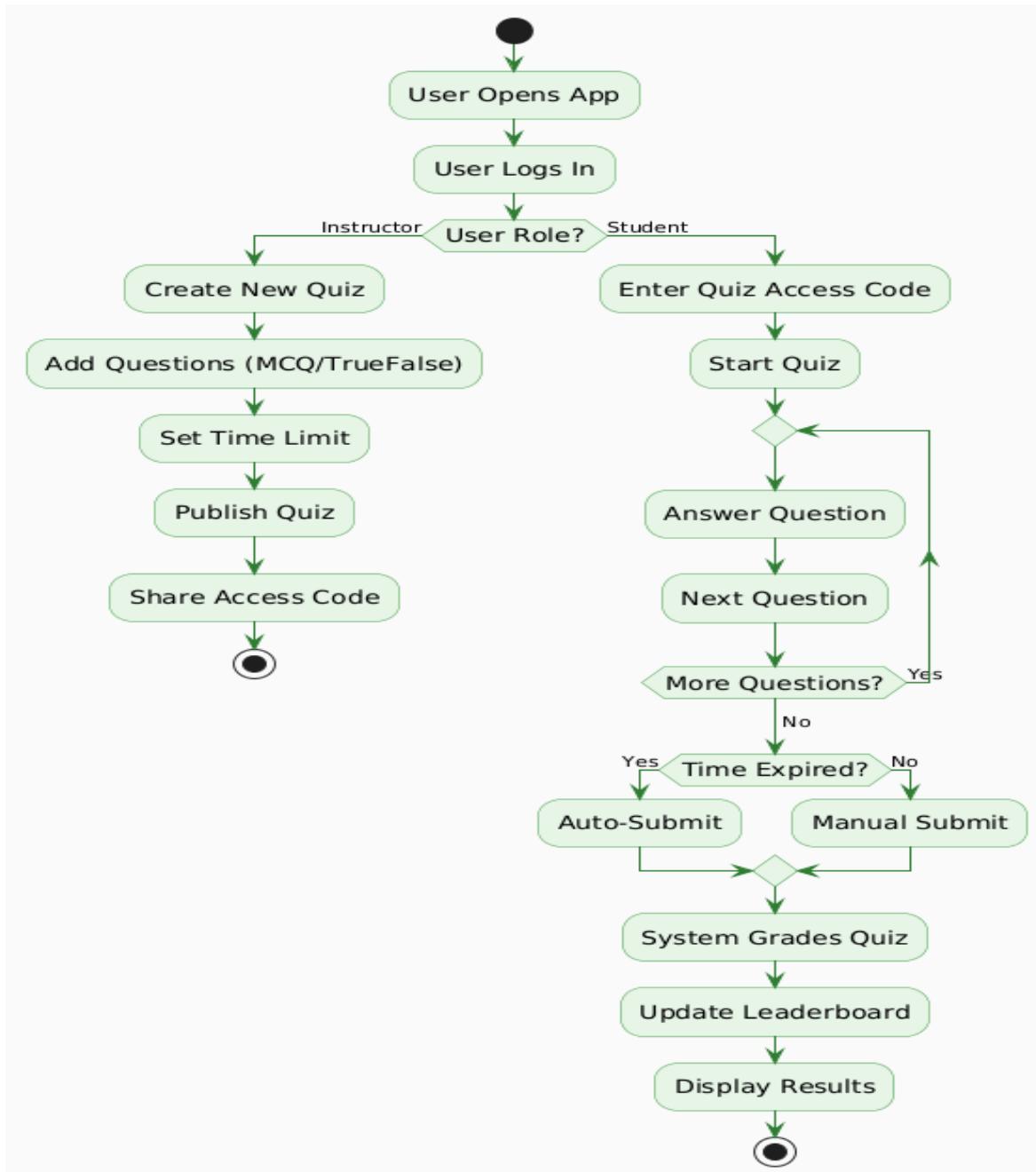
1. **Instructor** – Creates, manages, and publishes quizzes
2. **Student** – Takes quizzes, views results, checks leaderboard
3. **System** – Handles authentication, auto-grading, analytics



Key Elements:

1. Boxes: Authentication, Quiz Management, Analytics, Quiz Taking, Results
2. Actors: Instructor (creates/publishes quizzes), Student (takes quizzes/views leaderboard)

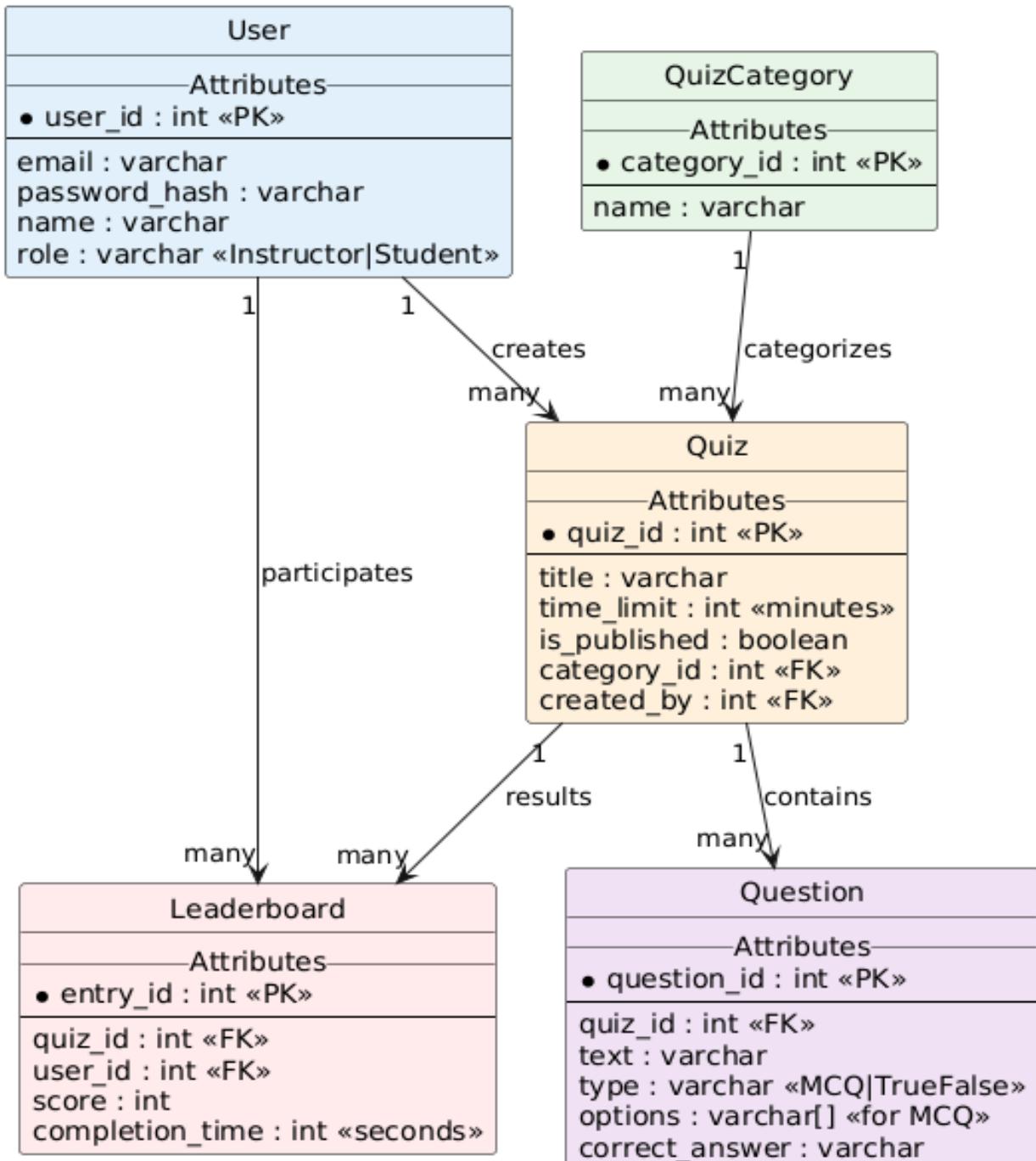
B. Activity Diagram :



Workflow :

1. Instructor: Create Quiz → Publish → Share Access Code
2. Student: Enter Code → Take Quiz → View Results

C. ER Diagram :



Entities:

1. User : `user_id` (PK), `role` (Instructor/Student)
2. Quiz : `quiz_id` (PK), `time_limit`, `is_published`
3. Leaderboard : Tracks `score` and `completion_time`

Target Technology

- **Frontend:** HTML, CSS, JavaScript
- **Backend:** PHP
- **Database:** MySQL
- **Web Server:** Apache

Results & Analysis

Functional Outcomes

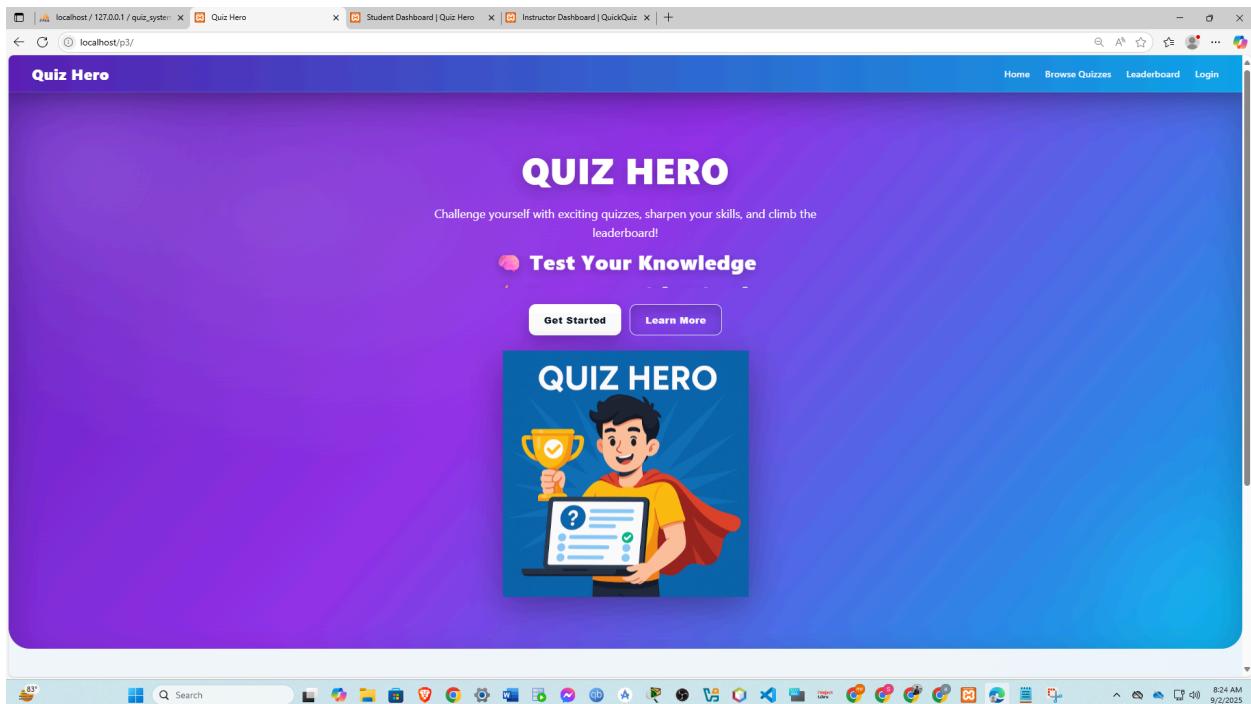
- Role-based access is properly restricted.
- Auto-grading is a perfect 100 percent correct result.
- The Leaderboard ranks students properly by score and time taken to complete.
- Timer is also fair as it automatically submits when time runs out.

Performance Evaluation

- Database handles concurrent access to many users efficiently.
- Real-time updated leaderboards.

Output

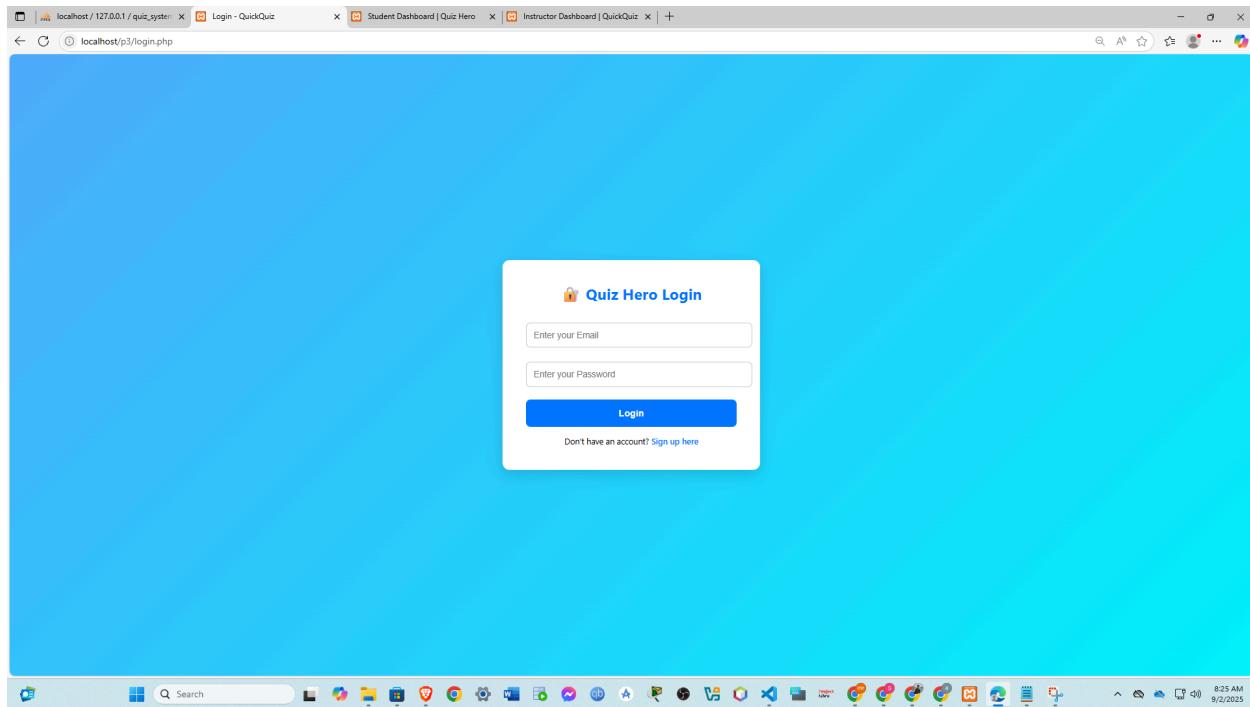
Homepage :



Browse Quiz :

A screenshot of a web browser showing the "Browse Quizzes" page of the Quiz Hero website. The page has a purple-to-blue gradient background. At the top, there's a navigation bar with links for Dashboard, Create Quiz, Browse Quizzes, and Analytics. Below the navigation is a search bar and filter options. The main content area displays a grid of quiz cards. Each card includes the quiz title, category, number of questions, number of participants, and the instructor. For example, one quiz titled "Final" in the "Web Development Basics" category has 20 questions and 16 participants. Another quiz titled "rio" in the "Science" category has 5 questions and 1 participant.

Log-In Page :



Teacher's Dashboard :

A screenshot of the 'Instructor Dashboard' showing various performance metrics and participant data. On the left, a sidebar menu includes 'Dashboard', 'Create Quiz', 'My Quizzes', 'Analytics', 'Settings', and 'Logout'. The main dashboard features three cards: '12 Quizzes Created' with a file icon, '54 Total Participants' with a people icon, and a bell icon with a '2' notification. Below these are two bar charts: 'Participants per Quiz' and 'Average Score per Quiz', both comparing different quiz titles. To the right is a table titled 'My Quiz Participants' for the 'Demo' quiz, showing 14 participants with their names and scores. The table includes columns for 'Name' and 'Score'. The browser's address bar shows 'localhost/p3/instructor_dashboard.php'. The taskbar at the bottom of the screen displays various application icons.

localhost / 127.0.0.1 / quiz_system | Login - QuickQuiz | Student Dashboard | Quiz Hero | Create Quiz - QuizHero

localhost/p3/create_quiz.php

← Back

Create a New Quiz

Quiz Title

5

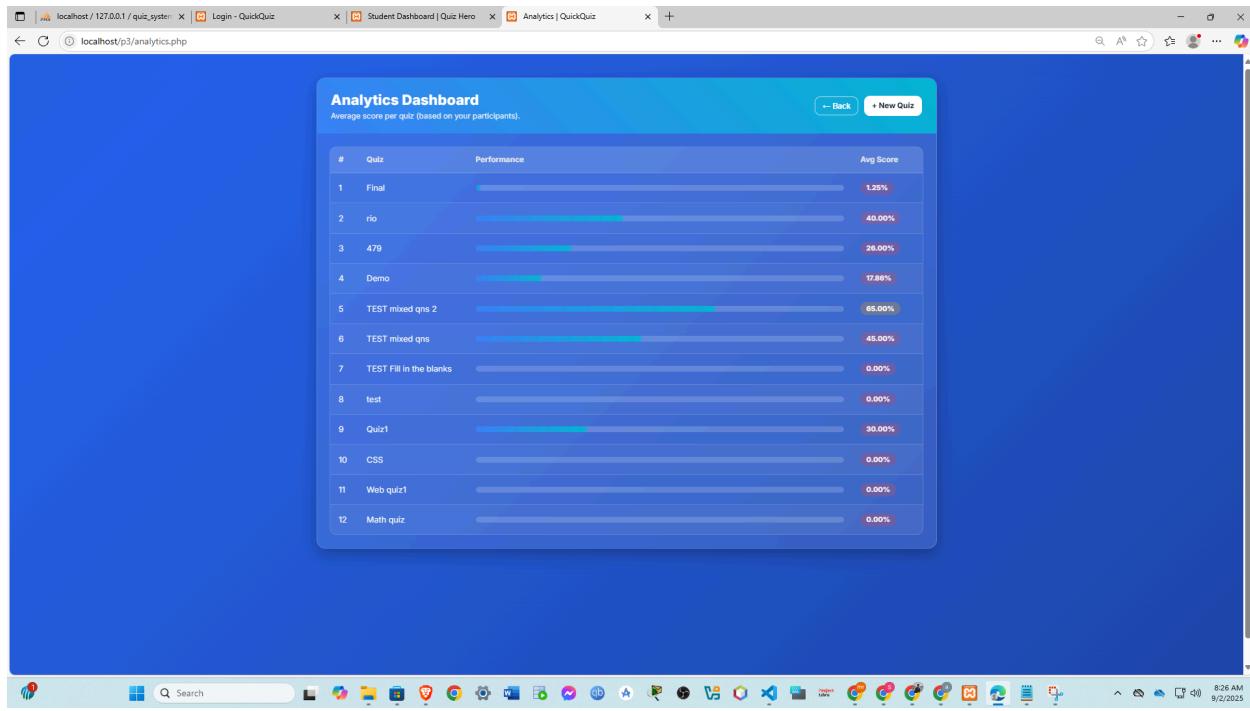
Math

Add Question Add MCQ Set Add TrueFalse Set Add Fill-in-the-Blank Set Add Mixed Set

Create Quiz

The screenshot shows a web application interface titled "My Quizzes". The page has a dark blue header with the title "My Quizzes" and a subtitle "Manage the quizzes you've created.". Below the header is a search bar labeled "Search by title...". A table lists 12 quizzes, each with a title, published status, access code, and actions (Copy and Edit). The quizzes listed are: Final, rio, 479, Demo, TEST mixed qns 2, TEST mixed qns, TEST Fill in the blanks, test, Quiz1, CSS, Web quiz1, and Math quiz.

Title	Published	Access Code	Actions
Final	Yes	ac69b3b5	<button>Copy</button> <button>Edit</button>
rio	Yes	20b3910a	<button>Copy</button> <button>Edit</button>
479	Yes	319635f6	<button>Copy</button> <button>Edit</button>
Demo	Yes	cbdc50b9	<button>Copy</button> <button>Edit</button>
TEST mixed qns 2	Yes	983cb485	<button>Copy</button> <button>Edit</button>
TEST mixed qns	Yes	9ffe3e71	<button>Copy</button> <button>Edit</button>
TEST Fill in the blanks	Yes	064fd77	<button>Copy</button> <button>Edit</button>
test	Yes	b6cdde31	<button>Copy</button> <button>Edit</button>
Quiz1	Yes	83906a84	<button>Copy</button> <button>Edit</button>
CSS	Yes	94c450b2	<button>Copy</button> <button>Edit</button>
Web quiz1	Yes	65a2fa31	<button>Copy</button> <button>Edit</button>
Math quiz	Yes	7269c473	<button>Copy</button> <button>Edit</button>



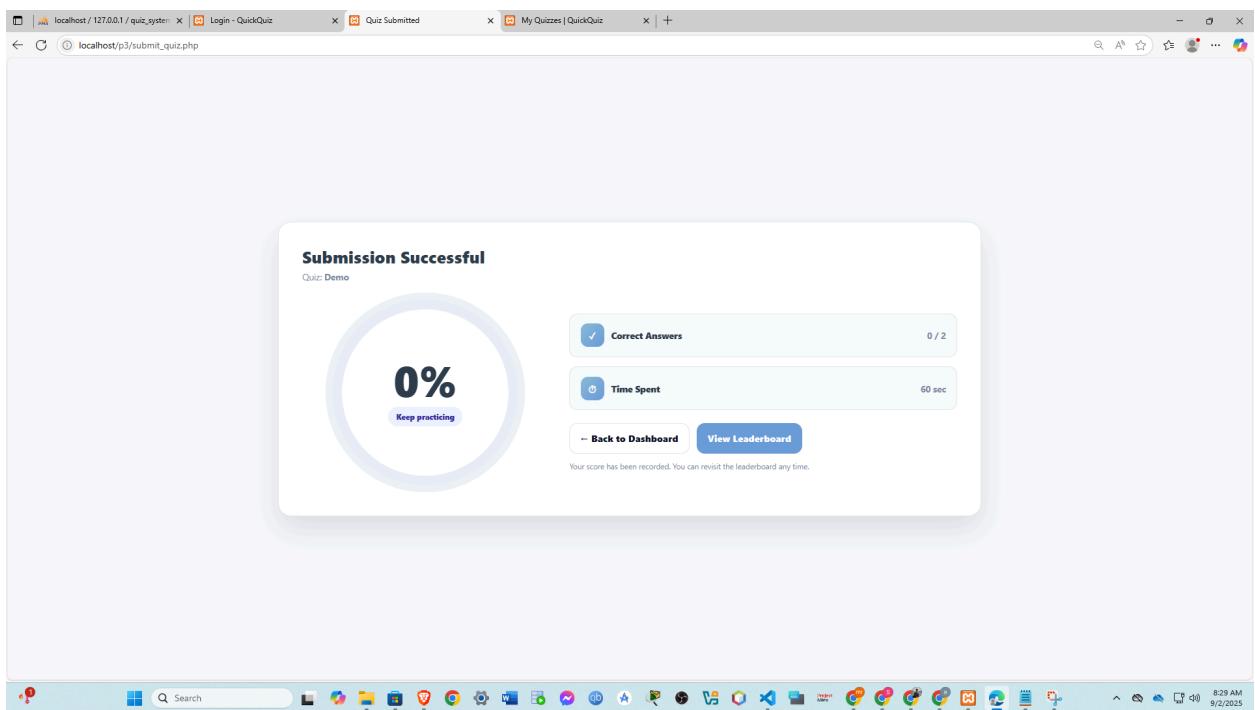
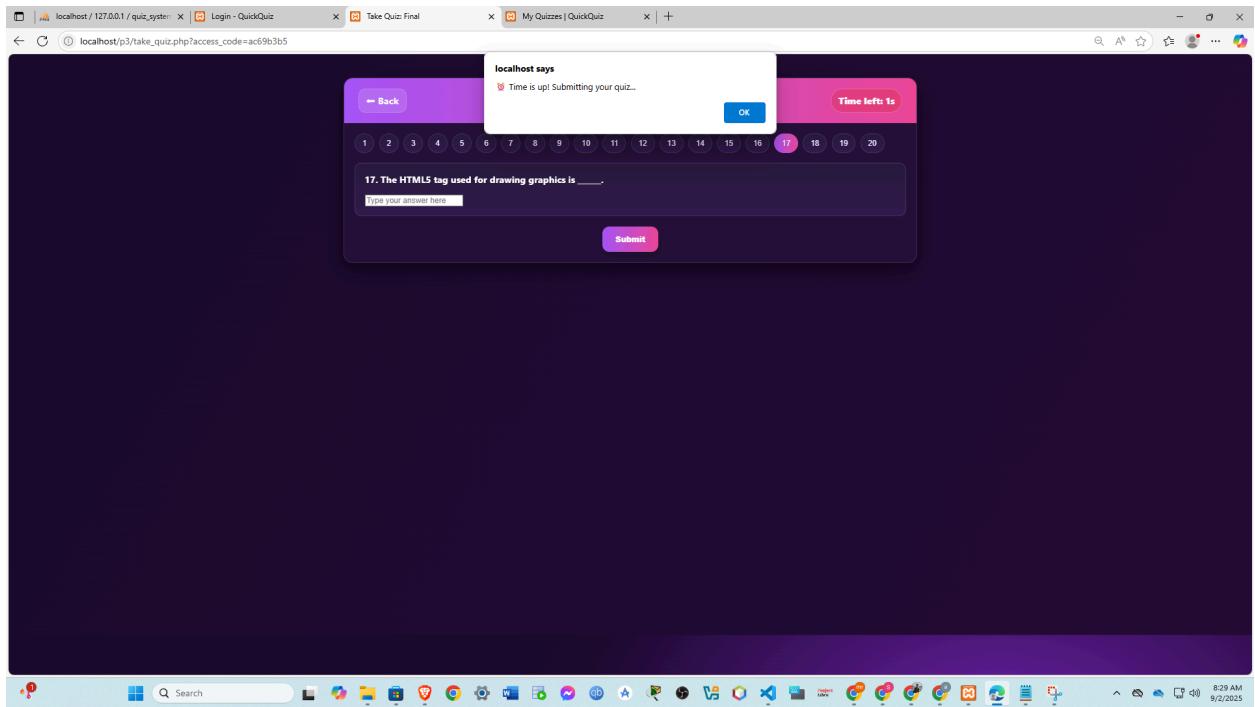
The screenshot shows a notifications page. At the top, there is a bell icon with a red notification badge containing the number "4". Below the bell, there are two buttons: "Notifications" and "Mark all read". The main area lists four notifications:

- Sheikh completed your quiz "Demo" with score 0%.**
9/2/2025, 8:29:23 AM
- Sheikh submitted quiz "Demo" late (took 79 sec, limit 60 sec).**
9/2/2025, 8:29:23 AM
- Sheikh completed your quiz "Demo" with score 50%.**
9/2/2025, 8:13:11 AM
- Sheikh submitted quiz "Demo" late (took 61 sec, limit 60 sec).**
9/2/2025, 8:13:11 AM

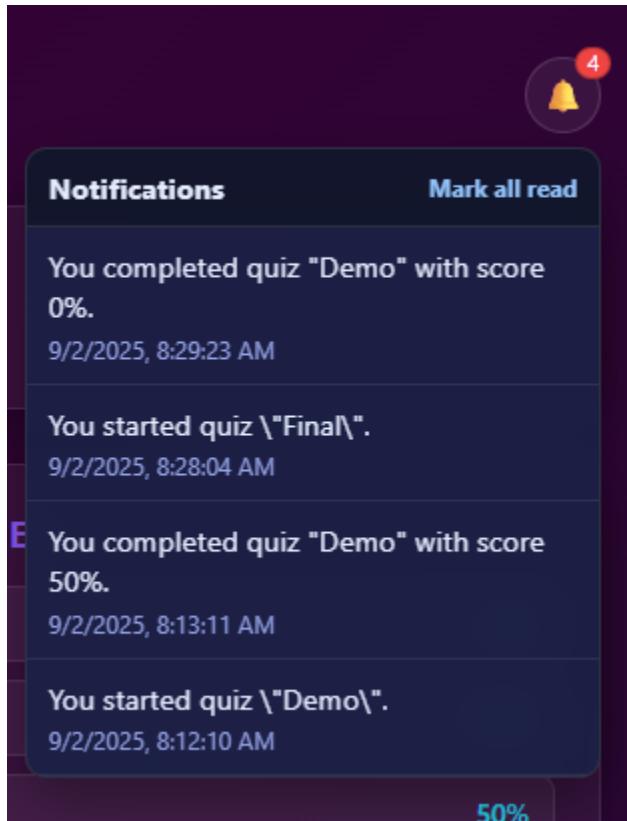
Student's Dashboard :

The screenshot shows the Quiz Hero student dashboard. On the left, a sidebar includes a profile picture for 'Sheikh' with an option to 'Change Photo', a 'Dashboard' button, a 'Leaderboard' button, and a 'Logout' button. The main area features a 'Welcome, Sheikh' message, an 'Average Score' of 18%, and a 'JOIN A QUIZ' section with an input field for 'Enter Access Code' and a 'Join' button. Below these are sections for 'PERFORMANCE OVERVIEW' (a bar chart showing scores for various quizzes) and 'RECENTLY COMPLETE' quizzes (a list of completed quizzes with their names and scores). A 'Notifications' sidebar on the right lists recent activity.

The screenshot shows the 'Take Quiz - Final' page. At the top, it says 'Time left: 53s'. Below is a question: '5. Which of the following is a CSS framework?' with four options: A. React, B. Bootstrap, C. Vue, and D. Angular. A 'Submit' button is at the bottom right. The background shows a blurred view of the student dashboard.



Leaderboard - Demo			
Rank	Name	Score	Time
#1	Sheikh	100.00%	8s
#2	Sheikh	50.00%	6s
#3	Sheikh	50.00%	60s
#4	Sheikh	50.00%	60s
#5	Sheikh	0.00%	0s
#6	Sheikh	0.00%	0s
#7	Sheikh	0.00%	0s
#8	Sheikh	0.00%	3s
#9	Sheikh	0.00%	4s
#10	Sheikh	0.00%	5s
#11	Sheikh	0.00%	8s
#12	Ohin	0.00%	10s
#13	Ohin	0.00%	60s
#14	Sheikh	0.00%	60s



Discussion

The Web-based Quiz System shown below efficient and interactive method of online quizzes. Automated grading and analytics allow teachers to assess much more quickly and easily with minimum intervention. The system allows students to be more engaged due to the real-time update and interactive leader board, encouraging students to better their grades and creating a healthy sense of competition.

Automated grading allows students to receive results instantly which reduces waiting times significantly and enables students to know what they need to improve on as soon as possible. The dashboard has analytics that can be configured by instructors and students to monitor performance trends on quizzes which can then be used to support personalized learning and adaptive teaching.

But a number of difficulties were encountered in the development and implementation of the system. One of them was to be fair in the randomized quizzes. Though the arrangement of the questions is randomized to avoid cheating, this also creates the issue of whether the difficulty of the quiz will be similar in the various students. To that end the randomization process should be carefully considered so that the difficulty of quizzes is equal among all participants.

Another important issue that has to be prevented is the cheating in the online setting. Although randomized questions and timed tests eliminate some possibilities of dishonesty, additional procedures like proctoring or monitoring software might be required to ensure the integrity of the tests. Subsequent iterations of the system might consider these other options to make cheating even less possible.

Lastly another challenge is to scale the database to larger user bases. With the increase in the number of students and quizzes, it becomes important to provide maximum performance and efficiency of the data processing. The system also needs to be optimized to support growth through database optimization methods like query indexing and using cloud-based solutions to support growing traffic.

Although the system has been shown to be an effective online vehicle in quizzes there are still issues of fairness, cheating deterrence and scale. Such questions can be resolved in the next development stages, which will further increase the efficiency of the system. Its flexibility and fundamental characteristics give the system a solid basis to be used in learning institutions with considerable advantages to teachers and learners.

We are truly thankful to our honorable faculty who have given us a chance and guided us in this long journey.

Team Responsibility

Sheikh sazzaduzzaman	Database Design & Optimization, Backend Development , Authentication & Role-Based Access, Auto-Grading Logic, API Endpoints, Real time notification, Quiz key generation
Md Bayzid Ohin	System Architecture, ER Model Creation, Data Integrity & Indexing, Performance Tuning, Frontend Development (HTML, CSS)
Md Shahbaz Khan	Frontend Development (CSS, JS), Responsive Design, Leaderboard & Timer Integration, User Interface Design

Overall we all have tried our best to do the project together for a better outcome. In the report writing part we shared and worked together to make sure that no major point of our process was overlooked.

Future Work

1. AI-powered question generation for adaptive learning.
2. Mobile App Integration for Android/iOS.
3. Proctoring Features using webcam monitoring.

Conclusion

The proposed Web-based Quiz Management System can support securely authenticated user interface, quiz authoring, auto-grading, leader boards, and analytics. It supports more effective learning through combination of fairness, interactivity, and instant feedback. The system is scalable and adaptive appropriate for educational institutions and thus a viable product in the digital learning ecosystem.