Online Quiz Management System for Institution

K. V. S. Prasanna Vasavi[1*]; P. Harshavardhan[2]; K. Pavani[3]; V. Jaya Kanth[4]; K. D. S. R. Manohar[5]; M. Nageswararao[6]

[1,2,3,4,5]Student, Department of CSE, Sri Vasavi Engineering College(A), Tadepalligudem, AP, 534101, India
[6]Assistant Professor, Department of CSE, Sri Vasavi Engineering College(A), Tadepalliguden, AP, 534101, India

Corresponding Authors: K. V. S. Prasanna Vasavi[1*]

Publication Date: 2025/12/09

Abstract: Institutions Online Quiz Management System is a web-based system that has been created based on ReactJS, Spring Boot, and MySQL to facilitate the automation of quiz creation, scheduling, and evaluation in schools. It offers three major user roles, which include Admin, Teacher, and Student, and offers a scalable and structured architecture. Through the system, administrators can make departments, semesters, and sections, and bulk register teachers and students with an Excel sheet. Teachers can create multimedia-supported quizzes with text and image questions, while students can attempt exams with randomized questions and options for fairness. Reports are generated at multiple levels, including student, section, and branch performance It is a securely authenticated system with efficient data management and dynamic UI interaction utilizing ReactJS and RESTful applications in Spring Boot. All in all, it enhances transparency while reducing manual workloads, thus providing institutions with an interactive, modern assessment solution.

Keywords: *Web Application, MySQL, ReactJS, Spring Boot, Online Assessment, Quiz Management.*

How to Cite: K. V. S. Prasanna Vasavi; P. Harshavardhan; K. Pavani; V. Jaya Kanth; K. D. S. R. Manohar; M. Nageswararao (2025). Online Quiz Management System for Institution. *International Journal of Innovative Science and Research Technology*, 10(11), 2824-2832. https://doi.org/10.38124/ijisrt/25nov1521

I. INTRODUCTION

In the era of technology, the digitization of educational zones is eliminating the process of writing and physical evaluations on piece of paper, and adopting the rapid and efficient online evaluations. Manual quizzes are time-intensive, subject to errors and they have no real-time analytics. In order to mitigate these constraints, the suggested Online Quiz Management System (OQMS) will be based on contemporary web technologies of ReactJS, Spring Boot, and MySQL to offer a safe, scalable, and convenient assessment platform. OQMS concentrates management, education, and students on a single system. Departments, semesters and sections can be created by the admins and users can be bulk uploaded in Excel files. Educators have the ability to create multimedia quizzes, distribute them to various groups and they can automatically score the quizzes. Learners are given a smooth quiz functionality, random question setting, and immediate creation of results. The system is constructed on a three-tier architecture and therefore guarantees high performance, maintenance, and scalability in the future. With ReactJS, one gets a responsive interface. In conjunction with this, secure backend processes are managed by Spring Boot, while data is efficiently stored in a MySQL database.

Together these technologies power OQMS an application that increases transparency, reduces reliance on manual tasks and provides administrators, teachers (and even students) with useful data-backed insights. It is a certified way of contacting the traditional methods of assessment to the current systems of automatic evaluation.

II. LITERATURE SURVEY

Over the last twenty years, online assessment systems have advanced with an extremely intense rate due to the revolution of the modern web technologies and the emergence of e-learning. The first systems were primarily digitizations of paper-based tests that provided primitive automatic scoring but were flexible and did not display user experience of today's standards [1], [6]. Subsequently, REST APIs and the use of single-page front-end systems like ReactJS, Angular, and Vue.js were announced with a second wave of systems, in order to build more interactive and responsive platforms [1], [3], [4], [7]. These advances also simplified systems maintenance, made them more scalable and more suitable to multimedia contents and real time updating.

In the literature, three significant requirements are reiterated, namely: (1) multiple question support, (2) highscale assessment integrity, and (3) powerful reporting and analytics [2], [6], [9], [10], [13].

Numerous research investigates multimedia-based inquiries to enhance the interaction response and assess the upper order aptitudes, yet they observe a complication in storage and streaming performance [5], [9]. In order to achieve integrity in exams, researchers can suggest randomized questions, secure timers, more extensive authentication, and detection systems, such as plagiarism checking and abnormal analysis [7], [8], [10].

Bulk user management importation of big lists of students in the form of an excel or CSV is one of the practical requirements which are not discussed in the research extensively. Checks on validity, duplicate detection, and transactional imports are recommended as ways of preventing mistakes [12]. The features of reporting and analytics have become center of focus, and innovative systems now provide questioned level of difficulty, time on task, and instructor and administrator dashboards [2], [6]. Scalability and performance are also both of concern.

Research attachment on the application of stateless APIs, horizontal scaled, database replication, caching and content delivery networks to address high load of exams has been stressed [3], [5], [8]. Strategies that can be employed to address security and privacy issues are the use of HTTPS-only communication, the hashing of passwords, token-based authentication (JWT or OAuth2), access controls, and the logging of sensitive activities [7], [12].

Finally, there are gaps in AI-based analytics for automated grading of long answers, privacy-friendly proctoring, and better offline exam capabilities. It is evident that institutions now require open and scalable platforms with the capacity to handle multimedia content and make available advanced analytics.

Most of the existing solutions have the form of either commercial systems or limited academic prototypes representing various ideas [9], [12].

- *Problem Statement*

Manual workload, slowness in assessment, and transparency are some of the disadvantages of traditional quiz management in institutions. Educators have to print out papers, mark them by hand, and distribute the results manually, which is inefficient. Additionally, randomization and question security are difficult to manage manually. The absence of centralized systems prevents performance analysis across students, sections, and branches. Hence, there is a need for an automated, secure, and efficient system capable of handling end-to-end quiz management online.

- *Objectives*

The OQMS is a platform that offers secure and automated processing for quizzes. It ensures safety, scalability, and automation right from the creation of quizzes through scheduling and administration to grading them. Since everything is automated, starting from the preparation of questions to the printing of results, there will be less chance of errors, plus it cuts down many tedious admin jobs, too! OQMS has tight role-based access, allowing the right permissions to Admins, Teachers, and Students, while keeping data security on track. Password encryption, secure login procedures, and token-based authentication were some of the ways user information was kept protected. Accommodates bulk user registration, which is perfect for big institutions. Just upload an Excel file when you need to enroll multiple departments, semesters, or sections all at once. Teachers can now create more stimulating assessments by incorporating text, images, audio, and video into their questions. A built-in randomization algorithm also helps maintain exam integrity through shuffling question orders as well as multiple-choice options differently for each student. Quiz scheduling is completely taken care of by the system; students simply log into their dashboards where they can see any tests that are either forthcoming or currently available to take. Instant evaluation offers instant feedback after finishing a test or exam meaning there is less work overall for teachers too! So you get these polished reports with useful info like how different groups did (say classes or genders), which questions were commonly missed, and how individuals are progressing over time– helping educators make smart decisions based on solid data. Crafted using ReactJS, Spring Boot, and MySQL the system consists of interchangeable parts (modular) that perform well individually. Put together they create something with growth potential for example integrating more advanced features such as AI analytics or companion mobile apps. OQMS is not just about convenience; it takes data privacy seriously. It uses HTTPS encryption and checks inputs to keep stored information safe from attacks– all within a handy ReactJS interface known for being easy to use. Both teachers and students find moving around simple things like assignment briefs or grade histories straightforward; plus the whole system runs smoothly on anything from phones and tablets up to high spec PCs making accessibility universal indeed!

III. METHODOLOGY

Making and managing quizzes for schools is simpler with this system. It's built with ReactJS on the front end, Spring Boot on the back end, and MySQL in a database– these technologies work together well to give you a safe and efficient platform.

- *This System Employs a Three-Tier Architecture Model Described as Follows:*

The front end manages everything users see utilizing an intuitive design that appears nice while also adapting well regardless of screen size or device type– perfect for Admins, Teachers, and Students alike. What's more? Because it's built with ReactJS, there's no need to refresh your browser manually to see things like quiz timers counting down or instant grades pop up; the page just knows how to do that stuff all by itself.

The back end (Spring Boot), overall logic is handled here. This includes user authentication, quiz creation, scheduling tests, grading them– plus generating reports; it also takes care of communication between frontend & database using REST APIs.

Spring Boot offers not only strong security features but also ensures everything runs smoothly. The database stores critical data like user details quiz questions answers and results reports etc. MySQL helps manage large amounts of data both safely and rapidly.

Initially, the process began with a thorough requirement analysis conducting an in-depth assessment of what teachers, students, and administrators need. After understanding their needs– different parts of the system were planned and developed:

The Admin Module allows the admin to create departments, semesters, sections, and register many teachers and students simultaneously through Excel files.

Within the Teacher Module teachers are enabled to design quizzes incorporating text images videos schedule tests and access detailed performance reports.

Finally, there is the Student Module through which pupils can log in see upcoming quizzes participate in exams— seeing their results immediately afterward! After constructing all components– comprehensive testing was carried out on the system to ensure that everything functioned correctly– it worked as planned!

IV. FEATURES OF THE SYSTEM

It handles quizzes from start to finish– making and grading them automatically.

It is a really secure system in place that makes sure everything's fair, running smoothly and at top speed.

Tasks for administrators, educators and learners are all brought together in one easy-to-access spot by it as well.

➢ *Role-Based Access*

OQMS uses clear roles:

- Admin: manages users and academic structure
- Teacher: creates and schedules quizzes, views reports
- Student: attempts quizzes and checks results

Secure login, password reset, and Spring Security ensure protected access.

➢ *Bulk User Registration*

Admins can upload Excel files to register large groups of students and teachers. The system validates data automatically, reducing errors and setup time.

➢ *Academic Structure Management*

With the ability to create departments, semesters, and sections, admins can ensure quizzes and reports stay well organized. Meanwhile, teachers and students are always linked back to their appropriate section.

➢ *Quiz Creation with Multimedia*

Teachers can add questions with text, images, audio, and video, assign marks and difficulty, and enable randomization of questions and options to reduce cheating.

➢ *Quiz Scheduling and Assignment*

The teachers decide when quizzes start and finish, as well as how long they last. They have the flexibility to assign quizzes to one class or several and students see only those quizzes that are currently available.

➢ *Real-Time Exam Environment*

Students get a clean interface with:

- Countdown timers
- Auto-save
- Navigation controls

➢ *Automated Evaluation & Results*

The system auto-grades objective questions and shows instant results, section averages, and detailed performance breakdowns. Teachers can view student-wise and question-wise analysis.

➢ *Reporting and Analytics*

OQMS provides:

- Individual student reports
- Section-level summary
- Semester/department-level analytics
- Question difficulty and success rates
- These insights help improve teaching and assessment strategies.

➢ *Security Features*

The system ensures strong protection through:

- Encrypted passwords (Bcrypt)
- JWT-based session handling
- HTTPS communication
- Audit logs for sensitive actions

➢ *Scalability and Architecture:*

OQMS, constructed with ReactJS, Spring Boot, and MySQL, boasts a modular design that enables scalability as well as simple extensibility. The system is interoperable with AI applications and mobile devices.

➢ *Easy-to-Use Interface:*

Thanks to its ReactJS build, OQMS features an easy-to-navigate user interface that works well on all screen sizes– making it simple for students, teachers and administrators alike. Plus accessibility features mean everyone can use it comfortably!

➢ *Insights*

During the development of OQMS, many valuable lessons were learned in system design, performance, UX and overall utility; all contributing to improvements for education providers.

- *System Design Insights:*

The system architecture was based on a three-tier design– ReactJS for the front end, Spring Boot for the middle layer, and MySQL for data storage. The use of REST APIs for communication between the front end and back end helped make each component independent so that maintenance would be easier and future expansion (such as adding a mobile app) could be done without too much trouble.

- *User Experience:*

One thing we learned is that users like simplicity in the interface. ReactJS allowed us to create separate dashboards-one each for students, teachers, and administrators-that made it easy for everyone to find relevant areas with speed and no confusion.

We also found that including clear calls-to-action-such as buttons with obvious next steps-plus details like countdown timers meant people felt more comfortable knowing how long they had left when taking tests; all these elements contributed towards an overall good feeling while using OQMS. We also found that including clear calls-to-action plus details like countdown timers meant people felt more comfortable knowing how long they had left when taking tests all these elements contributed towards an overall good feeling while using OQMS.

- *Performance and Scalability:*

In tests under load, Spring Boot demonstrated an ability to handle many concurrent users without slowing down. Features such as database indexing and caching provided further speed enhancements proving the system scalable: ready not just for today's big institutions but also tomorrow's ones.

- *Randomization and Integrity:*

Randomizing questions and options for each student reduced cheating and protected exam integrity. Teachers found this feature reliable and fair.

- *Security Measures:*

Strong security practices were used, including JWT tokens, bcrypt password encryption, HTTPS, and protection against SQL Injection and XSS. Auto-logout after inactivity improved safety on shared devices.

- *Automation Benefits:*

Bulk user uploads saved admins a lot of time. Teachers benefited from automatic quiz evaluation and instant results, reducing manual effort and errors.

- *Reporting and Analytics:*

The system made reports really clear for each student and each section too. The graphs and charts were a great help to the teachers– they could see where understanding was weak and then work on better ways to teach!

- *Problems and Fixes*

There were a few issues along the way– like when multimedia files were huge or lots of people tried to upload stuff at once. But we found solutions using asynchronous uploads and clever React state management plus some server tuning. These things also made everything much more stable.

- *What Users Said*

Students loved seeing their results straight away, teachers felt it saved them loads of time while administrators Appreciated how it kept everything neat and transparent. As a whole, OQMS proved itself effective, easy to use– and genuinely good at boosting both teaching techniques and student learning!

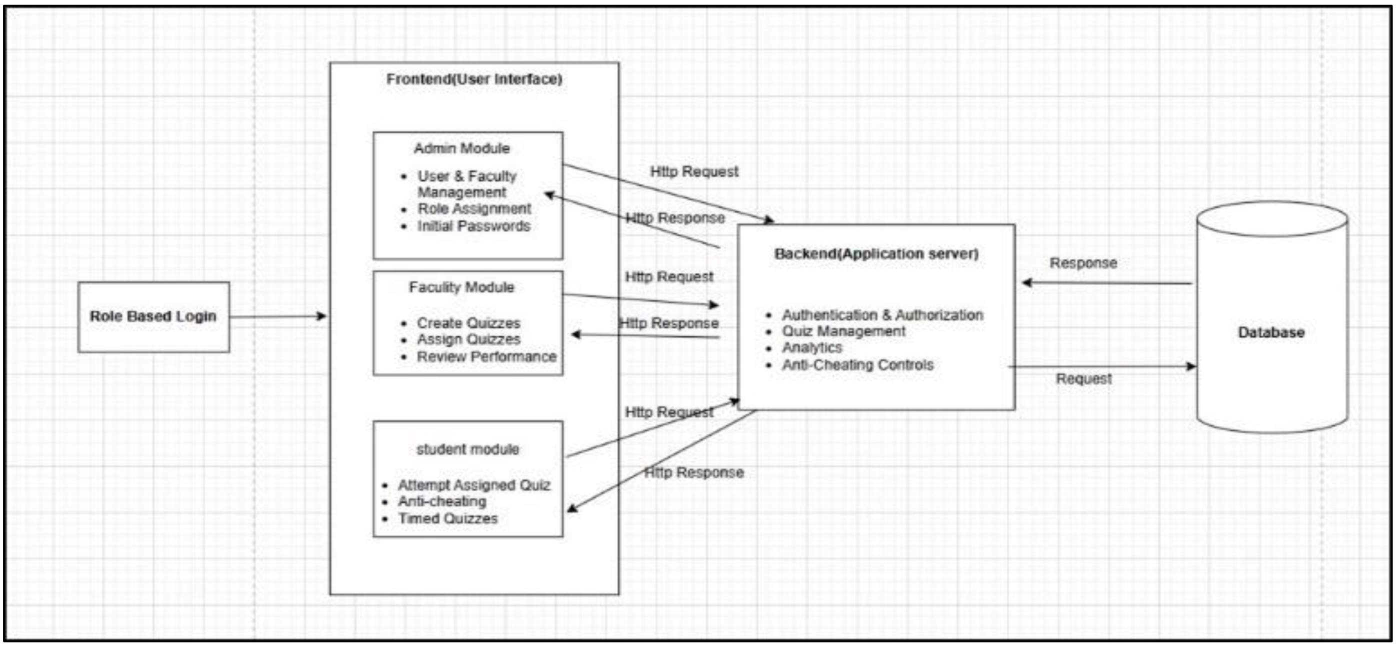


Fig 1 System Workflow

- *Proposed Work*

The proposed work will further enhance the Online Quiz Management System to make it feature-rich, efficient, secure, and user-friendly by providing more features. The future scope includes hosting on the cloud, the development of a mobile application, generating enhanced reports, and better security.

The system can be hosted on the cloud, allowing more users at one time without breakdowns or lag, and increasing its overall speed and reliability. It also allows for accessibility anywhere and automatic data backups.

A mobile version of the system will be developed so that students and teachers can have easy access to quizzes via their smartphones. In this way, users can attempt and manage quizzes even when they are away from a computer.

Security can be enhanced by adding two-step verification, stronger password encryption, and automatic session timeouts to protect user data. These updates will ensure safe and smooth operation for all users.

The reporting module can also be improved with more detailed analytics and visual charts to help teachers and administrators understand student performance more clearly.

Overall, the proposed work will make the Online Quiz Management System more flexible, secure, and convenient for institutions, teachers, and students.

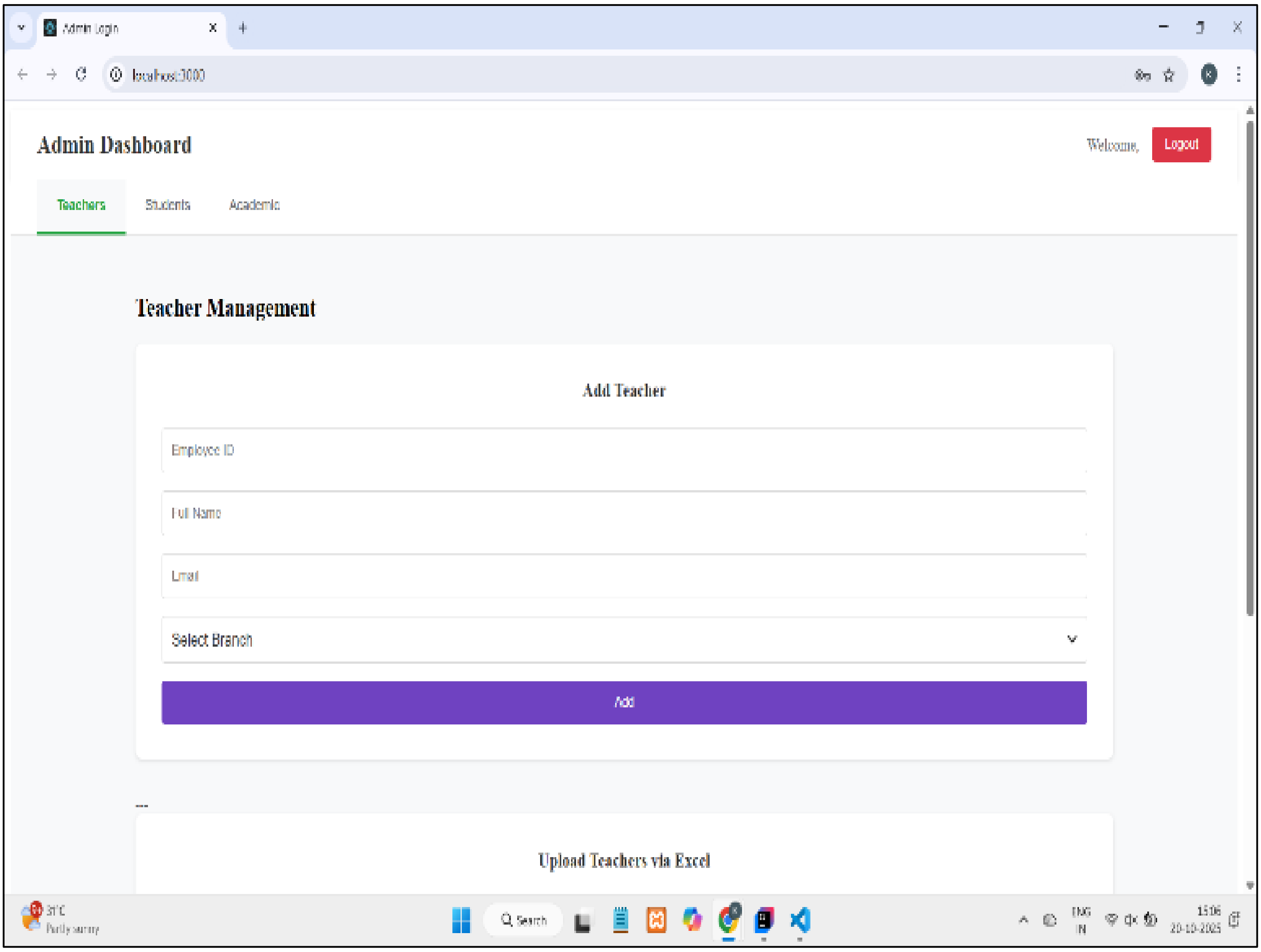


Fig 2 Admin Interface of the System

V. IMPLEMENTATION

The Online Quiz Management System was implemented using three main technologies: ReactJS, Spring Boot, and MySQL. Each part of the system performs a specific function to make the whole platform work smoothly and efficiently.

The front end of the system was built using ReactJS, which provides an interactive and responsive user interface. It allows users, Admins, Teachers, and Students to easily navigate the system. Dashboards were created for each role to manage their specific tasks. The use of ReactJS helped create a fast and dynamic website where users can view quizzes, results, and reports instantly.

The back end was developed using Spring Boot, which handles all the main logic of the system. It manages user authentication, quiz creation, scheduling, question randomization, and result calculation. The communication between the front end and back end is done through REST APIs, which ensures secure and reliable data transfer. Spring Boot also provides strong security using encrypted passwords and token-based login.

The database was designed in MySQL to store all important data, such as user details, quizzes, questions, answers, and reports. Proper table relationships were created to ensure smooth data retrieval and avoid duplication. Queries were optimized to provide faster responses during quiz attempts and result generation.

➢ *The System has Three Main Modules:*

- *Admin Module:*

The admin can create departments, semesters, and sections. Admins can also register students and teachers in bulk using Excel files and view performance reports.

- *Teacher Module:*

Teachers can create quizzes with multiple-choice questions, add images or videos if needed, schedule exams, and view student results and reports.

- *Student Module:*

Students can log in using their credentials, view available quizzes, attempt them within the given time, and immediately view their results after submission.

After development, the system was tested to check its functionality, performance, and security. Functional testing ensured all features worked correctly. Performance testing confirmed that the system could handle multiple users at the same time without lag. Security testing ensured that user data and quiz information were safe.

VI. LIMITATIONS

Although the Online Quiz Management System works efficiently and fulfills its main objectives, there are still some limitations that can be improved in future versions.

- ➢ The system requires a stable internet connection. If the network is slow or disconnected, students may face difficulties while attempting online quizzes.
- ➢ Currently, quizzes cannot be attempted offline. Students must stay connected to the internet during the entire quiz session.
- ➢ The system mainly supports multiple-choice questions. Other formats, like descriptive or fill-in-the-blank questions, are not yet included.
- ➢ The system can be accessed from mobile browsers, but does not yet have a dedicated mobile app, which limits ease of use for smartphone users.
- ➢ Reports are generated correctly but have limited graphical analysis. More visual tools, like charts or graphs, could make performance analysis easier.
- ➢ The system does not include online monitoring tools like webcam tracking or screen recording to prevent cheating during exams.
- ➢ Large Excel files with thousands of records may take more time to upload and validate, which could affect performance for very large institutions.

Despite these limitations, the system still provides a strong foundation for online quiz management. With future updates such as mobile app integration, offline mode, and improved analytics, these issues can be minimized or completely resolved.

VII. EXPERIMENTAL RESULTS

The Online Quiz Management System (OQMS) was tested in a real institutional setting to check the functionality, performance, and user experience of all three modules: Admin, Teacher, and Student. Each role worked smoothly and met institutional requirements.

➢ *Admin*

The Admin module successfully handled the creation of departments, semesters, and sections without errors. Bulk user registration through Excel uploads worked efficiently with proper validation. User management, password resets, and section creation were completed easily. Reports such as branch-wise, semester-wise, and section-wise summaries were generated clearly, and the system maintained fast performance even with large data.

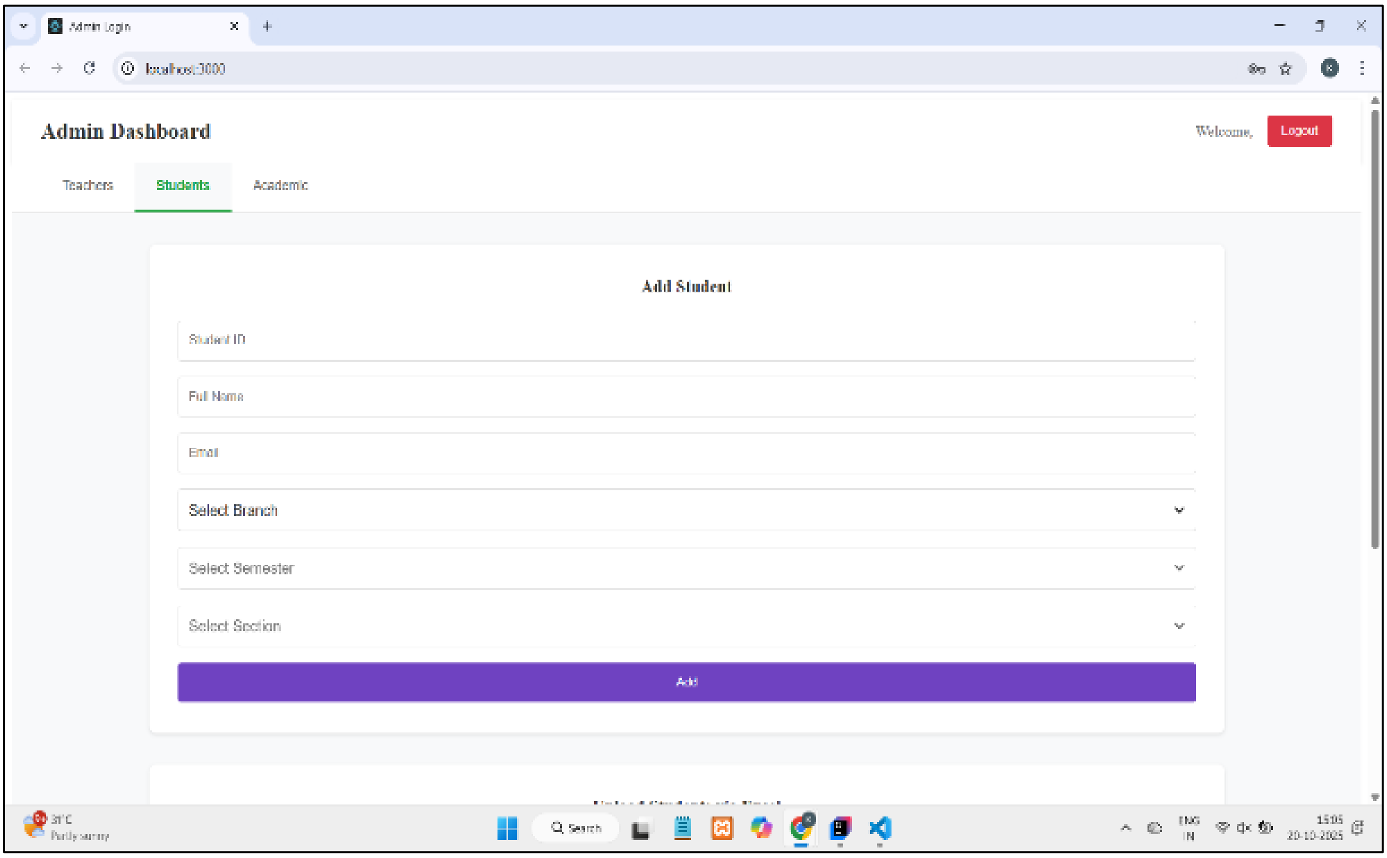
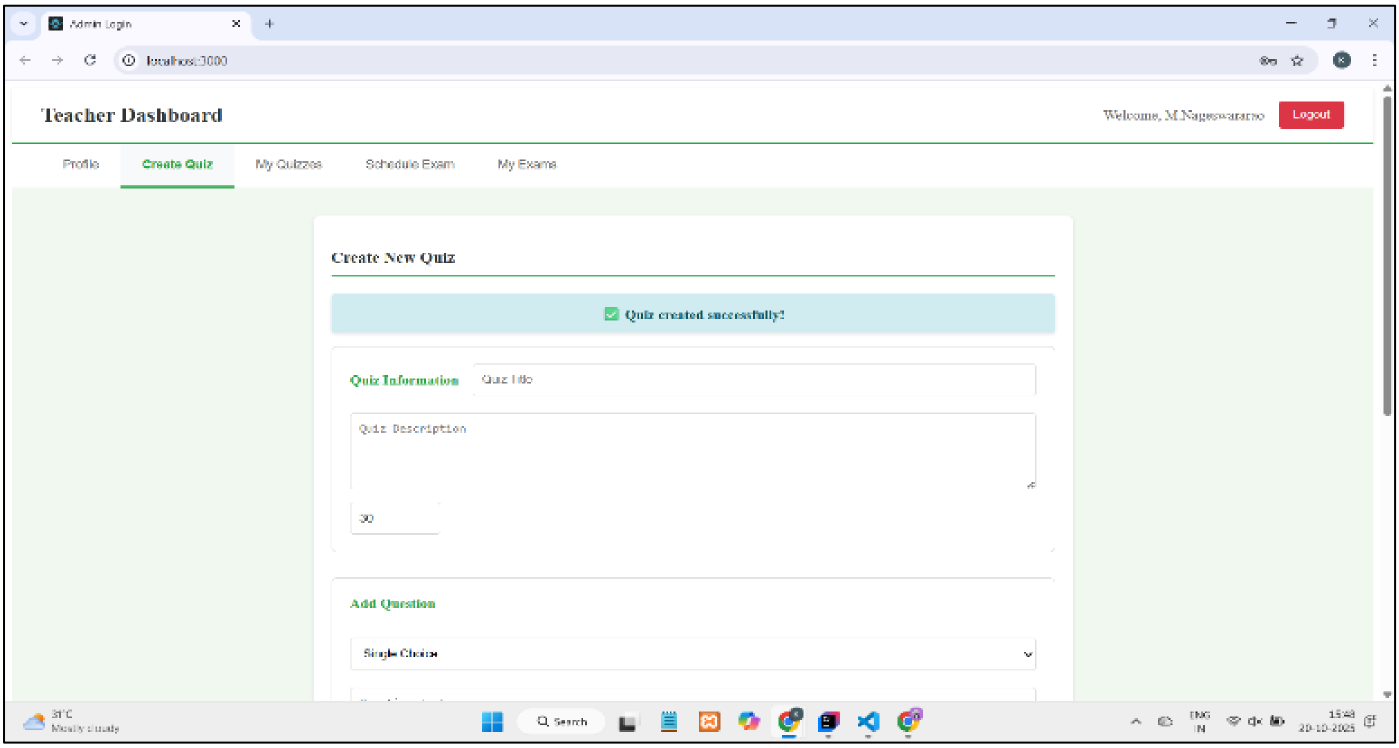


Fig 3 Admin Panel

➢ *Teacher*

Teachers were able to create quizzes with text, images, and multimedia. Quiz scheduling for single or multiple sections worked correctly. Question and option randomization provided each student with a unique quiz. Teachers viewed detailed reports such as student-wise marks and question-wise analysis. The interface was responsive and easy to use while creating and managing quizzes.



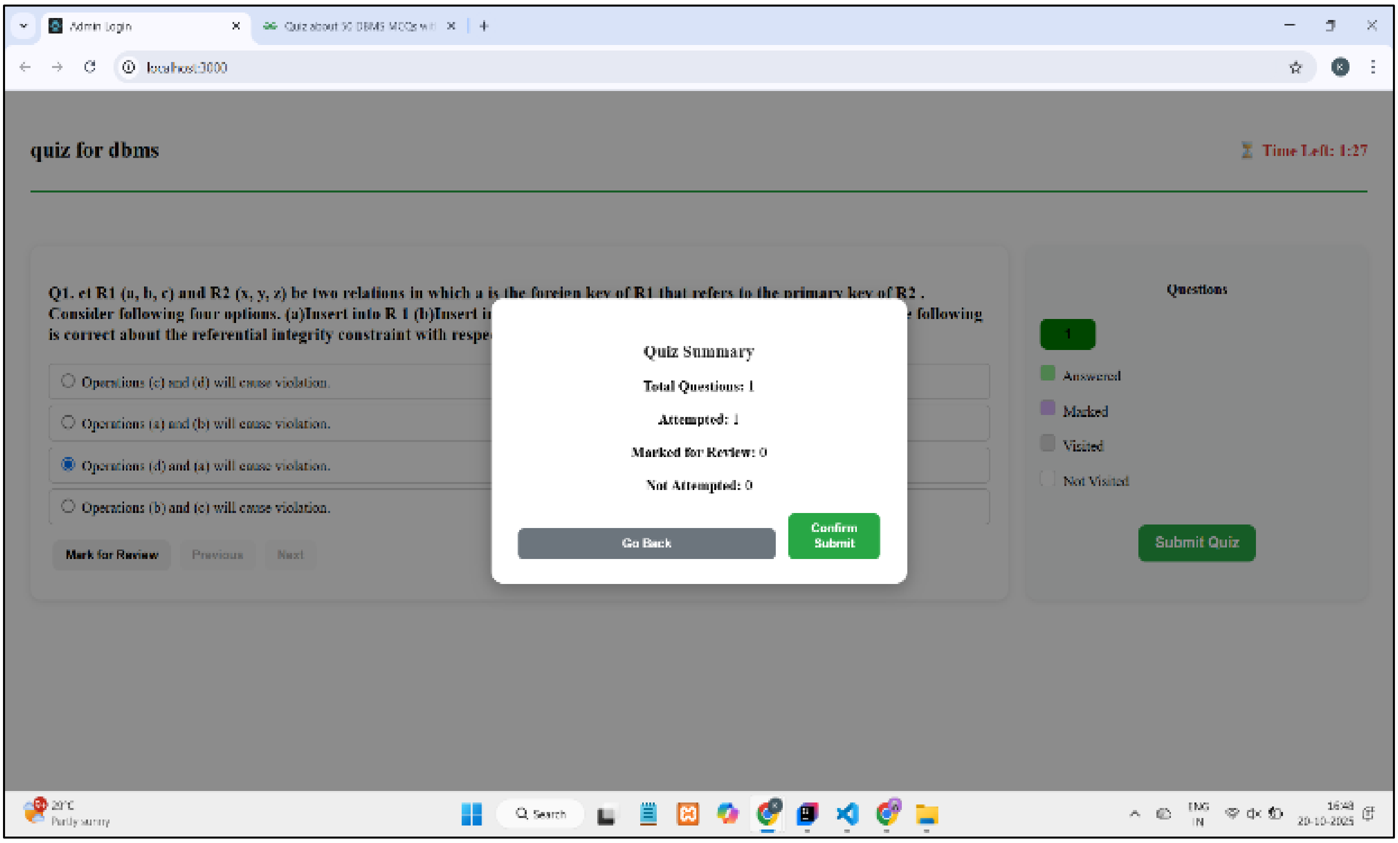
☑ Quiz created successfully!

Fig 4 Teacher Panel

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/25nov1521

➤ *Student*

Students logged in with admin-provided credentials and reset their passwords on first login. The dashboard clearly displayed active, upcoming, and completed quizzes. During quizzes, randomization, timers, and auto-submit worked properly. Students received instant results with detailed feedback and found the interface simple and accessible across devices.



quiz for dbms

⌛ Time Left: 1:27

Q1. et R1 (u, b, c) and R2 (x, y, z) be two relations in which a is the foreign key of R1 that refers to the primary key of R2 . Consider following four options. (a)Insert into R 1 (b)Insert i… following is correct about the referential integrity constraint with respe…

○ Operations (c) and (d) will cause violation.

○ Operations (a) and (b) will cause violation.

● Operations (d) and (a) will cause violation.

○ Operations (b) and (c) will cause violation.

Quiz Summary

Total Questions: 1

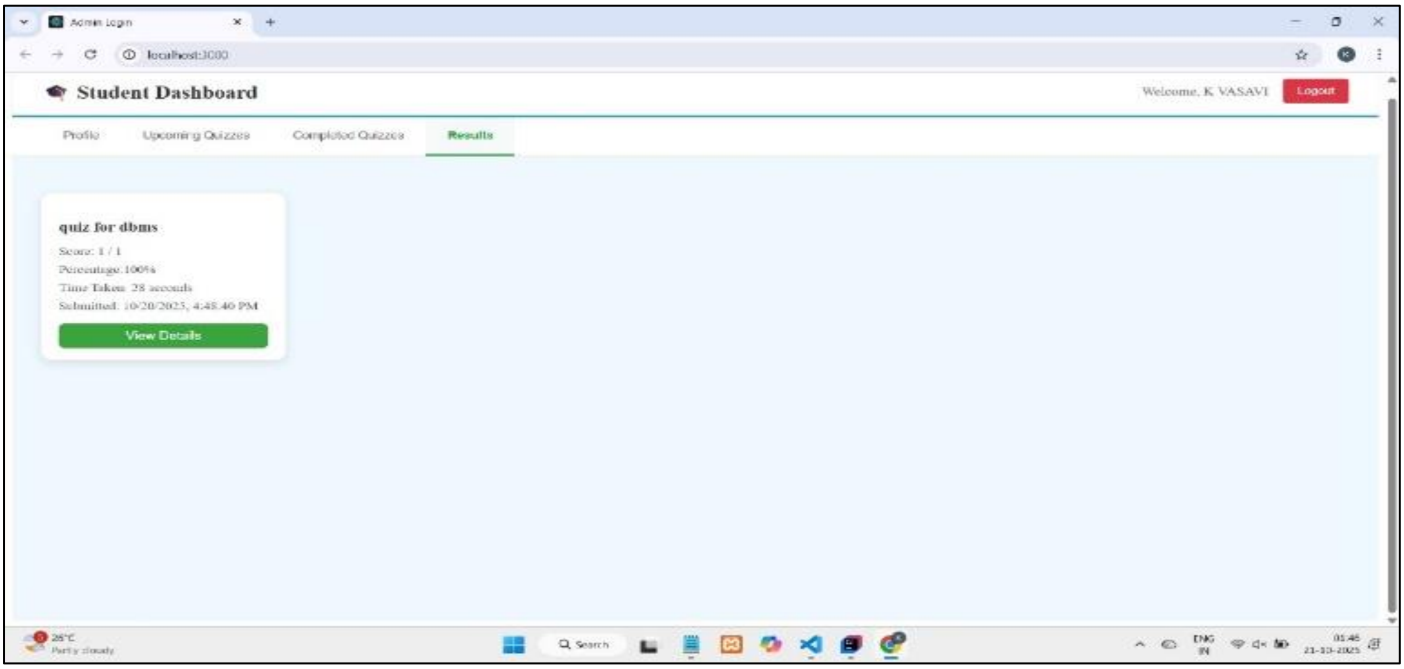
Attempted: 1

Marked for Review: 0

Not Attempted: 0

Fig 5 Student Panel

All modules worked smoothly with quick response times and strong coordination between the front end, back end, and database. Bulk uploads, quiz creation, scheduling, and quiz attempts ran without issues. The system proved stable, user-friendly, and effective for institutional online quiz management.



Student Dashboard

Welcome, K VASAVI

quiz for dbms

Score: 1 / 1

Percentage: 100%

Time Taken: 28 seconds

Submitted: 10/20/2025, 4:48:40 PM

Fig 6 Student Result

VIII. FUTURE ENHANCEMENTS

The Online Quiz Management System can be further improved by integrating advanced technologies to enhance efficiency, security, and accessibility. Future versions of the system may include AI-based performance analysis to track individual learning progress and suggest personalized quizzes based on student strengths and weaknesses. Integration with cloud platforms such as AWS or Firebase can enable real-time synchronization, secure data storage, and scalability for larger institutions.

To improve accessibility, a mobile application can be developed using React Native, allowing users to attempt quizzes and view reports on smartphones and tablets. The system can also incorporate live proctoring features using webcam monitoring and browser activity tracking to ensure exam integrity. Additionally, automated report generation in PDF or Excel format will help teachers and administrators easily share and store performance data.

Future enhancements can also include multilingual support, voice-based navigation, and offline quiz access through Progressive Web App (PWA) technology. These improvements will make the system more flexible, inclusive, and efficient, catering to the growing needs of modern educational institutions.

IX. CONCLUSION

The Online Quiz Management System effectively automates the process of quiz creation, scheduling, and evaluation, making assessments more organized and efficient for educational institutions. Built using ReactJS, Spring Boot, and MySQL, the system ensures seamless operation between users and components. It provides a user-friendly platform for administrators, teachers, and students, reducing manual effort and improving transparency in evaluations. Overall, the system offers a reliable and scalable solution for conducting online assessments in a structured and efficient manner.

REFERENCES

[1]. Smith, A., et al. “Design and Implementation of Online Quiz Portals Using ReactJS and NodeJS,” IJERT, 2023.

[2]. Ramesh, K., and Sharma, V., “Web-Based Assessment Systems for Educational Institutions,” IJCSMC, 2021.

[3]. Liu, D., and Wang, Y., “Developing Scalable Web Applications Using Spring Boot,” IEEE Access, 2020.

[4]. Gupta, R., and Jain, P., “Integration of ReactJS and REST APIs for Dynamic Web Applications,” IJERT, 2022.

[5]. Singh, A., et al. “Performance Analysis of Online Examination Systems Using MySQL Databases,” IJCA, 2021.

[6]. M. Ahmed and S. Hossain, “Web-Based Online Assessment System for Educational Institutions,” *International Journal of Emerging Technologies in Learning (iJET)*, vol. 15, no. 12, pp. 110–122, 2020.

[7]. S. K. Mishra and R. R. Patel, “Development of a Secure and Scalable Online Quiz Portal for Academic Institutions,” *International Journal of Engineering Research & Technology (IJERT)*, vol. 9, no. 6, pp. 510–514, 2020.

[8]. D. S. Kumar and T. S. Rao, “Automation of Online Examination System Using Modern Web Technologies,” *International Journal of Scientific & Technology Research (IJSTR)*, vol. 9, no. 5, pp. 300–305, 2020.

[9]. K. Pradhan and P. Sahoo, “A Web-Based Quiz Management System for Educational Evaluation,” *International Journal of Computer Science Trends and Technology (IJCST)*, vol. 8, no. 2, pp. 110–115, 2020.

[10]. S. Jain and R. Mehta, “An Improved Online Test Management System Using Cloud Infrastructure,” *International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)*, vol. 9, no. 4, pp. 2251–2257, 2021.

[11]. A. Patil and N. Deshmukh, “Online Quiz System for Academic Evaluation,” *International Research Journal of Engineering and Technology (IRJET)*, vol. 7, no. 5, pp. 2873–2877, 2020.

[12]. R. Sharma and D. Kaur, “A Study on the Effectiveness of Online Quiz Systems in Higher Education,” *International Journal of Education and Development Using ICT*, vol. 17, no. 2, pp. 90–98, 2021.