



# PIET QUIZ PORTAL WEB APPLICATION USING MERN STACK

**Naman Gupta<sup>1</sup>, Muskaan Bhatia<sup>2</sup>, Namita Bilandi<sup>3</sup>, Abhishek Dadhich<sup>4</sup>**

<sup>1,2,3</sup>Final Year, B.Tech, Poornima Institute of Engineering and Technology

<sup>4</sup>Asst. Professor, Poornima Institute of Engineering and Technology

## Abstract

The MERN stack is a popular technology stack for building web applications that include MongoDB, Express.js, React.js, and Node.js. MongoDB is a NoSQL database that provides a flexible and scalable data storage solution. Express.js is a web application framework for Node.js that simplifies the process of building APIs and handling HTTP requests. React.js is a JavaScript library for building user interfaces, and Node.js is a JavaScript runtime for building server-side applications. The quiz management system using the MERN stack is a web application designed to simplify the process of creating, managing, and taking quizzes. This system allows users to create quizzes and customize settings such as time limits and scoring options. Users can add questions to quizzes, and the system provides real-time data processing, making it efficient to manage large amounts of data.

## Keywords

Web applications, React Js, NodeJS, ExpressJS, server-side applications, and User Interfaces

## Objective

It is designed to facilitate the creation, administration, and management of quizzes or assessments. The primary objective of

such a system is to provide a platform that enables educators or administrators to create quizzes and assessments easily, administer them to students or participants, and manage the overall process. The system should allow for a variety of question types, such as multiple-choice, true/false, or essay, as well as customizable settings like time limits and randomization of questions. The system should also provide automatic grading, allowing educators to save time and focus on other important tasks. The ability to monitor student progress and analyze their performance is another key objective of a quiz management system, allowing educators to identify areas of weakness and tailor their teaching accordingly. The system should provide real-time feedback to students, enabling them to track their progress and identify areas where they need to improve. Additionally, the system should be user-friendly and accessible, with support for multiple languages and platforms to reach a wider audience. Overall, the objective of a quiz management system is to streamline

the assessment process, making it more efficient, effective, and engaging for both educators and students.

## Introduction

The quiz management system project is a software application designed to manage and administer quizzes or tests, typically in an educational setting. The system is designed to simplify the process of creating and administering quizzes, making it easier for instructors to manage the learning process. The system includes a user-friendly interface that allows instructors to create and customize quizzes, set time limits, and specify grading criteria. Once the quiz is created, it can be distributed to students through the system, either online or in person. The system can track the progress of students as they take the quiz and provide immediate feedback on correct and incorrect answers. After the quiz is completed, the system automatically grades the quiz and generates a report on each student's performance.

This allows instructors to easily identify areas where students struggle and provide additional support as needed. The quiz management system project aims to provide a functional and user-friendly system that meets the users' needs, whether they are instructors, students, or administrators. The project may include the development of a custom software application, as well as integration with other systems, such as learning management systems, to provide a seamless user experience. Overall, the quiz management system project is designed to improve the efficiency and effectiveness of the learning and assessment process, while providing instructors with the tools they need to manage and track student progress. PIET Quiz is a portal for students and teachers where teachers can create tests of different subjects and set other parameters (Time duration, Test Expiry, and Number of Questions)(The major feature of the quiz

portal is that students can switch the tab and copy-paste the questions from the portal while giving the test. Teachers can do different operations on the portal like delete, edit, and add questions to the quiz and they can also block/unblock the students from the portal to create discipline within the portal.

Students will be able to view the quiz uploaded by the teacher on the portal including quiz description, quiz title, and quiz creator but they can perform the quiz for a limited time.

## Software Requirements

The following software is needed to launch a quiz management system:

**Node.js:** Node.js is a runtime environment for the server-side execution of JavaScript code. We must install Node.js on the computer to use a quiz management system that was built using it. The most recent version of Node.js is available for download from the official website.

**React:** A JavaScript package called React is used to create user interfaces. We must install the React library on the computer in order to use a React-based quiz management system. A package manager like npm (Node Package Manager) or Yarn may be used to install React.

**Database:** A database is necessary to hold quizzes, student information, and other pertinent data. Any database that supports Node.js, including MySQL, MongoDB, and PostgreSQL, can be used with the quiz management system. Before we can launch the quiz management system on our computer, we must install and set up the database.

We will need a web server that can serve both the React front-end and the Node.js back-end in order to operate the quiz management system. We may use any Node.js-compatible web servers, such as Nginx or Apache. The quiz management system cannot be launched until the web

server has been installed and configured on our computer.

## Hardware Requirements

Here are some more specifics on what hardware a quiz management system needs: The processor, sometimes known as the computer's "brain," is in charge of carrying out computations and executing instructions. The CPU needed for a quiz management system must be able to handle several users working simultaneously and sophisticated computations. It is advised to use a multi-core CPU with a clock speed of at least 2 GHz. The web server, database, and other programs that are operating on the system should not put too much strain on the CPU.

## Literature Survey

- A. "Design and Implementation of MEARN Stack-based Real-time Digital Signage System" [2]

The paper proposes a design and implementation of a real-time digital signage system called MR-DSS, which is based on the MEARN (ME(A+R)N) stack framework. The system is designed to efficiently handle real-time tasks such as urgent/instant messaging and system status monitoring, in addition to conventional digital signage CMS services. The CMS of MR-DSS, called MR-DSCMS, provides services through REST APIs for normal operations and Socket.IO for real-time messaging. The abstract highlights the architecture and components of MR-DSS, design and implementation issues, and experimental testing results, including the functional performance of MR-DSS, the networking load performance of MR-DSCMS's REST APIs compared to a well-known open-source CMS, and the faster

performance of real-time messaging via Socket.IO compared to REST APIs.

- B. "Comprehensive Study of MERN Stack - Architecture, Popularity, and Future Scope" [1]

The abstract discusses a study that compares the runtime performance of MERN (MongoDB, Express, React.js, and Node.js) and MEVN (MongoDB, Express, Vue.js, and Node.js) stacks, as well as their workability in the job market. The study includes an experiment building two to-do applications with MERN and MEVN stacks, measuring loading time, adding time, updating, and deleting time. Additionally, a survey was conducted among Swedish-based companies to identify the software stack trends. Although the low response rate challenges drawing conclusions from the survey, the results show that Vue.js and React.js are popular choices for client-side software, often combined with other server-side software like Java, Go, and Django. Factors influencing software choice include ease of learning, community support, client needs, and availability of developers.

- C. "MERN: A Full-Stack Development" [3]

describes a paper that focuses on modern web technologies used in the corporate world. The paper aims to familiarize readers with these technologies, including Back-End development, Database management, Digital Marketing, Domain and Hosting, Front-End development, Full-Stack Development, Git and Github for version control, Google Analytics for website analytics, Heroku for web hosting, Linux operating



system, MERN stack (MongoDB, Express, React, Node.js) for full-stack development, Netlify for web deployment, and VPS (Virtual Private Server) for hosting websites.

D. "Performance Optimization using MERN stack on Web Application" [4]

It emphasizes the significance of understanding customer demand in the fast-paced business environment of today. It highlights the growing importance of having an online presence to cater to customer needs effectively and efficiently. The abstract further mentions that an e-commerce web application can provide customers with access to a wide range of products, from basic necessities to luxury items. The project aims to create a user-friendly web application that allows registered users to easily view and purchase products online, using different payment options. Additionally, the project aims to provide business owners with a convenient way to view and manage orders placed through the web application. The technologies chosen for the project are React.js, MongoDB, Node.js, and Express.js. React.js is a popular JavaScript library for building user interfaces, while MongoDB is a widely used NoSQL database for storing data. Node.js is a JavaScript runtime that allows server-side execution of JavaScript code, and Express.js is a popular web application framework for Node.js. The abstract further states that the paper will provide a detailed discussion of each of these technologies and how they are utilized in building an e-commerce web application. The goal is to demonstrate the implementation of

these technologies in creating a comprehensive web application that facilitates online shopping and serves the needs of both customers and business owners.

## Conclusion

Based on the available research, building a quiz portal using the MERN stack has several advantages. The MERN stack offers a full-stack JavaScript solution that is highly scalable and efficient. Using MongoDB as the database allows for easy storage and retrieval of questions, answers, and user information. Express and Node.js provide a robust server-side framework for handling API requests and running the server, while React provides a dynamic and interactive user interface. To build a successful quiz portal using the MERN stack, several factors should be considered. Firstly, the application should be designed to be responsive and user-friendly. The quiz questions and answers should be organized in a clear and concise manner, and the user interface should be intuitive and easy to navigate. Secondly, security should be a top priority, and measures such as encryption and access control should be implemented to protect user data. Finally, the application should be tested thoroughly to ensure that it is stable and performs well under load. Overall, the MERN stack provides a powerful and versatile framework for building quiz portals that are scalable, efficient, and secure. By carefully considering the design, security, and testing of the application, developers can create a robust and reliable quiz portal that meets the needs of their users.

## References

1. Bhavyaa, Suhani Gupta, Ms. Vaishali, "Comprehensive Study of MERN Stack - Architecture, Popularity and Future Scope," 2021, International Journal of Scientific Research in Computer

Science, Engineering and  
Information Technology.

2. Trinh Duy Khue, Thanh Binh Nguyen, UkJIn Jang, Chanbin Kim, Sun-Tae Chung, "Design and Implementation of MEARN Stack-based Real-time Digital Signage System," 2017 Journal of Korea Multimedia Society.
  
3. Yogesh Baiskar, Priyas Paulzagade, Krutik Koradia, Pramod Ingole, Dhiraj Shirbhate, "MERN: A Full-Stack Development," 2022-01-18 IJRASET, doi: <https://doi.org/10.22214/ijraset.2022.39982>
  
4. Sourabh Mahadev Malewade & Archana Ekbote. "Performance Optimization using MERN stack on Web Application,"06, June-2021 International Journal of Engineering Research & Technology (IJERT) <http://www.ijert.org> ISSN: 2278-0181  
IJERTV10IS060239 Published by: [www.ijert.org](http://www.ijert.org) Vol. 10 Issue

