

Quiz App with AI Generative Questions

MRS.M. Bagya Lakshmi, G. Gnana Vasanth, I. Daniel Samraj.

*(Computer Science and Engineering, Francis Xavier Engineering College, Tirunelveli
Email: bagyalakshmi@francisxavier.ac.in)

** (Computer Science and Engineering, Francis Xavier Engineering College, Tirunelveli
Email: gnanavasanthg.ug22.cs@francisxavier.ac.in)

*** (Computer Science and Engineering, Francis Xavier Engineering College, Tirunelveli
Email: danielsamraji.ug22.cs@francisxavier.ac.in)

Abstract:

This system presents the development of an intelligent quiz application that leverages AI-driven content generation to dynamically produce questions based on user-selected topics. Traditional quiz systems rely on static question banks, which often result in outdated content, limited variety, and repetitive assessments. In contrast, this application integrates the Open Router API, a gateway to advanced language models, to create a flexible and scalable question-generation mechanism. Users can enter a topic of interest, and the backend system communicates with the Open Router API to retrieve fresh, relevant, and contextually appropriate questions. The system architecture combines ReactJS for the frontend, Node.js for the backend, and Open Router for AI integration. This not only enhances user engagement but also allows for real-time, personalized learning experiences. The results show a high degree of accuracy, relevance, and educational value in the AI-generated content. This approach contributes significantly to the field of adaptive learning technologies and demonstrates the potential of generative AI in modern educational tools.

Keywords — Quiz Application, Artificial Intelligence, Generative Model, Open Router API, ReactJS, Node.js, Dynamic Assessment, E-Learning.

INTRODUCTION

In the rapidly evolving digital education landscape, artificial intelligence (AI) is transforming the way learners interact with content, particularly in the areas of assessment and personalized learning. While traditional quiz systems have proven effective for reinforcing knowledge, they primarily rely on **static question banks**. These systems often require **manual**

updates, lack adaptability, and fail to align with the learner's current context or progress. As a result, user engagement decreases over time, and the effectiveness of the learning experience diminishes. This project introduces an innovative solution: an **AI-powered dynamic quiz application** that uses **real-time generative models** to automatically create quiz questions tailored to each learner's selected topic. Rather than relying on conventional, pre-written datasets, the system leverages **Open Router**, a powerful

API gateway that connects to advanced AI models, to generate **contextual and diversified questions** on demand. This significantly reduces the need for manual content creation while enhancing variety, relevance, and user engagement. The real-time generation of quiz content is powered by Open Router's integration with large language models capable of understanding and synthesizing educational material. Users can choose a subject or topic, and the system instantly crafts questions that are accurate, topic-aligned, and pedagogically appropriate.

In recent years, the educational sector has witnessed a significant shift towards digital and remote learning environments. As students increasingly engage with online platforms, the need for adaptive, interactive, and effective assessment tools has become more pronounced. Traditional quiz applications, while useful, are often limited in scope due to their reliance on manually curated question banks. These static systems not only lack personalization but also become repetitive over time, leading to reduced learner interest and engagement. This creates a demand for smarter, more dynamic tools that can evolve alongside the learner's progress.

What sets this system apart is its ability to **continuously adapt to the learner's needs** without requiring static content or manual intervention. The platform transforms traditional quizzes into an **interactive, intelligent learning experience**—enhancing engagement, retention, and accessibility across diverse knowledge domains.

The primary goal of this project is to develop a **real-time, AI-generated quiz platform** that modernizes the assessment process, making learning more **adaptive, scalable, and deeply personalized**. By integrating Open Router for dynamic content creation, ReactJS for an engaging user interface, and Node.js for backend operations,

this system demonstrates the powerful potential of conversational and generative AI in reshaping education.

OBJECTIVES

The main goal of this project is to create an AI-powered, real-time quiz application that generates personalized and dynamic quiz questions based on user-selected topics using natural language processing and generative AI. The system aims to replace traditional, static question banks with intelligent content generation that adapts to individual learning needs, ensuring greater engagement, variety, and relevance. By integrating cutting-edge AI technologies through Open Router, along with a modern web development stack, the project seeks to deliver an interactive, scalable, and personalized assessment experience that mirrors human-like adaptability in educational environments.

Main Points and Importance of Our Research

AI-driven dynamic quiz generation: Developed an intelligent quiz application that uses Open Router to connect with generative AI models, allowing real-time generation of personalized quiz questions based on user-selected topics, thus eliminating reliance on static question banks.

Scalable and efficient backend architecture: Implemented a robust backend using Node.js to manage quiz sessions, handle API requests, and coordinate seamless communication between the user interface and AI services, ensuring fast and reliable performance across devices.

Real-time content personalization: The system dynamically generates diverse, contextually relevant questions tailored to each user's selected subject, difficulty level, and learning preferences—enhancing engagement and improving the educational value of each session.

Seamless integration of modern technologies: Combined the power of ReactJS for a responsive and user-friendly frontend with generative AI for backend content creation, delivering a smooth and modern user experience across web platforms.

Enhanced learning through continuous variability: By leveraging generative models, the application ensures that learners are constantly presented with new and unique questions, reducing content repetition and promoting deeper understanding through dynamic reinforcement.

Promoting intelligent education systems: Demonstrated the potential of generative AI in the education sector by creating a flexible, scalable, and intelligent quiz platform that can adapt to various academic disciplines and individual learning goals.

In a practical educational technology application, the potential of artificial intelligence (AI) in the field of digital learning was showcased by developing a scalable and user-friendly quiz system that integrates generative AI and intelligent automation to provide personalized, dynamic assessment experiences.

MODULES AND ALGORITHM

The quiz app with AI Generative questions is divided into several functional modules, each responsible for handling a specific aspect of the quiz application. This modular architecture ensures better maintainability, scalability, and ease of testing. Key modules include user authentication, topic selection, dynamic question generation, real-time answer validation, and result tracking. By separating these concerns, developers can independently update or enhance specific components without affecting the overall system performance. This design also enables future integration with third-party tools, such as learning management systems (LMS) or analytics platforms, to further enrich the learning experience.

1. User Interface Module

Built using ReactJS and Material-UI, this module provides a responsive and interactive frontend. It allows users to input their desired quiz topics, interact with questions, and view results. It ensures a smooth user experience through intuitive design and dynamic rendering.

2. Prompt Generation Module

This module operates on the backend and is responsible for creating structured prompts tailored to the user's selected topic. The prompt instructs the Open Router API to generate relevant questions in a specific format (e.g., multiple-choice with four options and one correct answer).

3. AI Integration Module

Acting as a bridge between the application and the Open Router API, this module sends prompts and receives responses. It handles API communication, authentication, request throttling, and response validation.

4. Question Parsing Module

This module parses the raw JSON or string response from Open Router into usable data objects. It ensures the format is consistent, extracts individual questions and options, and identifies the correct answers.

5. Quiz Rendering Module

This module takes parsed questions and stoutly displays them to the stoner in a quiz format. It supports real-time picture, navigation between questions, and input shadowing for named answers.

6. Scoring and Result Module

Once the user submits the quiz, this module evaluates the answers, calculates the score, and generates a result summary. It may also provide optional feedback on incorrect answers and performance.

A. Algorithm

The process begins when the user accesses the web interface and selects a subject or topic of interest for the quiz. Upon submission, the frontend triggers an API request to the backend, passing the selected topic, preferred difficulty level, and the number of questions. The backend, built using Node.js, receives the request and formats a prompt which is then sent to Open Router, a gateway to advanced generative AI models. The AI processes this prompt and returns a set of dynamically generated quiz questions and multiple-choice options, all contextually relevant to the selected topic.

These questions are then validated for clarity, uniqueness, and completeness. If any errors or inconsistencies are detected—such as missing options or repeated questions—the system automatically regenerates the faulty entries by re-prompting the AI model. Once validated, the complete quiz set is sent back to the frontend, where it is rendered using ReactJS in a clean and interactive layout.

During the quiz session, the user selects answers in real-time, which are instantly evaluated to provide immediate feedback. A score counter and progress tracker update dynamically, giving the user a clear overview of their performance. At the end of the session, the result is displayed, along with a summary of correct answers and explanations (if enabled).

If the user remains inactive for a predefined period, the session is paused and the system prompts them to continue. Persistent inactivity results in automatic session termination, ensuring efficient resource use. Additionally, users can provide feedback on question quality, enabling future enhancements to the prompt design and AI tuning. The entire process ensures a dynamic, intelligent, and personalized assessment experience tailored to the user's learning journey.

METHODOLOGIES

A. Dynamic Quiz Generation Using Generative AI

To exclude the limitations of stationary question banks, the system uses Open Router to access advanced generative AI models capable of dynamically creating quiz questions. When a user selects a subject and difficulty level, a tailored prompt is sent to the AI model. This model generates a fresh set of quiz questions and options in real-time. Preprocessing is applied to the AI response to ensure the content is clear, coherent, and formatted correctly before rendering it in the user interface. This approach allows the application to deliver diverse, up-to-date, and contextually relevant questions every time, reducing content repetition and improving learning outcomes.

B. Personalized User Experience Through Environment-Aware Questioning

The system adapts to each user's preferences and performance history. Upon quiz initiation, the application captures user choices such as selected subjects, difficulty levels, and previous scores. This data is used to fine-tune the prompt sent to the AI model, ensuring that the generated content aligns with the user's current learning environment. The system also supports adaptive questioning, where the difficulty of subsequent questions adjusts dynamically based on user performance during the session. This personalized delivery encourages engagement and optimizes knowledge retention.

C. Real-Time Feedback and Intelligent Result Analysis

After each response, the system provides immediate feedback, indicating whether the selected answer is correct. The backend performs real-time evaluation and tracks scores, accuracy, and time taken for each question. Once the quiz is complete, a detailed performance report is generated, highlighting strengths and areas for improvement. This feedback loop not only reinforces learning but also motivates users to continue exploring new topics. If enabled, AI can

also generate brief explanations for each question, helping users understand the correct answers more deeply.

D. Scalable Architecture and Data Handling

The system's architecture is modular and built using ReactJS on the frontend and Node.js on the backend. This ensures fast, responsive interactions and a clean separation of concerns. All quiz metadata and user session data are managed securely on the backend, enabling scalability and real-time performance monitoring. This structure allows for smooth session management, secure storage of results, and potential integration of user progress tracking over time.

E. Dynamic Quiz Generation Using Generative AI

To overcome the limitations of stationary question banks, the system uses Open Router to access advanced generative AI models capable of dynamically creating quiz questions. When a user selects a topic and difficulty level, a tailored prompt is sent to the AI model. This model generates a fresh set of quiz questions and options in real time. Preprocessing is applied to the AI response to ensure the content is clear, coherent, and properly formatted before rendering it in the user interface. This approach enables the application to deliver diverse, up-to-date, and contextually relevant questions every time, reducing content repetition and improving learning outcomes.

F. Conflict Recognition and Resolution

To avoid scheduling clashes in quiz sessions or assessments, the system can be extended to check for conflicting appointments or academic activities using calendar integration (e.g., Google Calendar API). If a selected quiz slot overlaps with another event, the system promptly retrieves and presents alternative slots, ensuring a smooth and conflict-free experience.

G. Prompt Quiz Initialization and Delivery

Once the user selects a subject and difficulty level, the request is immediately processed by the AI model through Open Router, which generates tailored quiz content in real time. The questions, along with options and answers, are instantly delivered to the user. Notifications and next-step suggestions are provided without delay, ensuring a fast and engaging quiz experience from start to finish.

H. Safeguarding of User Data

The system places high priority on user data security and privacy. All user details—including quiz history, performance data, and preferences—are transmitted using encrypted communication protocols and securely stored in compliance with industry standards. Access controls and session logging are maintained to ensure transparency and safeguard user confidentiality at all times.

I. Evaluation of User Satisfaction

After each quiz session, users are prompted to rate their experience and optionally provide feedback. This data is analyzed to identify areas of improvement in both AI question generation and the responsiveness of the user interface. The feedback loop also contributes to refining the AI model, improving the relevance and clarity of future questions, and enhancing overall platform performance.

J. Capacity and Concurrent User Handling

The system is designed to support multiple users concurrently by employing asynchronous request handling and dynamic session allocation. With load balancing and distributed infrastructure, the quiz platform can maintain optimal performance during periods of high user activity, ensuring that each user enjoys a consistent and uninterrupted experience.

K. Performance and System Scalability

Built with a modular architecture using ReactJS and Node.js, the platform is inherently scalable. Cloud services are leveraged to dynamically allocate resources based on user demand. This ensures the platform remains responsive and reliable even as usage scales, delivering consistent performance without compromise.

EXISTING SYSTEM

Many traditional systems rely heavily on static question banks and fixed templates, which not only limit the diversity of questions but also fail to cater to individual learning paths and performance levels. These platforms typically lack integration with advanced AI technologies capable of generating dynamic, real-time quiz content based on user inputs and preferences. As a result, users are often presented with repetitive and outdated questions that do not evolve with their progress or knowledge gaps. Furthermore, these systems offer minimal feedback and limited adaptability, providing a transactional interaction that does little to engage or motivate learners. The absence of intelligent automation, contextual understanding, and real-time analysis ultimately leads to a disengaging experience that fails to support continuous learning and knowledge retention. A modern quiz solution must move beyond these limitations by embracing generative AI, personalized performance tracking, and intuitive user interaction to truly enhance learning outcomes.

1. Quality and Applicability of Generated Questions

While AI models can generate questions rapidly, ensuring their quality and relevance remains a significant challenge. Studies have raised concerns about the educational impact of AI-generated quizzes, emphasizing the need for integrating external knowledge sources to enhance question quality.

2. Lack of Contextual Understanding

AI-generated questions often lack deep contextual understanding, leading to ambiguity or inaccuracies. This limitation affects the effectiveness of quizzes in assessing true comprehension.

3. Bias in Question Generation

AI models can inadvertently introduce biases based on the data they were trained on. This includes cultural, gender, or political biases, which can compromise the fairness and inclusivity of the quizzes.

4. Difficulty in Handling Complex Logic

Generating questions that require multi-hop logic or advanced critical thinking is challenging for current AI systems. They often struggle to create questions that test higher-order cognitive skills.

5. Evaluation of Question Effectiveness

Assessing the effectiveness of AI-generated questions is non-trivial. Traditional evaluation methods may not adequately capture the pedagogical value or the appropriate difficulty level of the questions.

6. Over-Reliance on AI Without Human Oversight

Fully automated systems risk propagating errors or generating subpar content without human intervention. A human-in-the-loop approach is often necessary to ensure the quality and appropriateness of quiz content.

PROPOSED SYSTEM

The proposed system is an AI-powered quiz application engineered to deliver personalized, dynamic quiz experiences by generating contextually relevant questions in real-time based on user-selected topics and difficulty levels. Built using modern web technologies such as ReactJS for the frontend and Node.js for the backend, the application ensures a responsive and seamless user interface. At

its core, the system integrates with the Open Router API, enabling access to powerful generative language models capable of creating diverse, coherent, and pedagogically sound quiz content on demand. Each quiz session is tailored to individual user preferences, ensuring that no two quiz experiences are exactly the same. The architecture is modular and component-driven, promoting scalability, code reusability, and ease of maintenance. Moreover, the system includes robust preprocessing mechanisms to validate and format AI-generated content before it reaches the user interface, ensuring clarity and relevance. With features such as real-time feedback, performance tracking, and adaptive difficulty adjustment, the platform not only assesses knowledge but also supports continuous learning and engagement. This approach sets a new standard for interactive educational tools by combining intelligent automation with user-centric design.

A. Frontend Interface

The frontend is developed using **ReactJS** to ensure a highly interactive and dynamic user experience, with **Material-UI** providing a modern and clean design that is responsive across various devices. Users can input their preferred topic, select the difficulty level, and seamlessly begin generating AI-driven quizzes. Each quiz question and its multiple-choice options are rendered dynamically from the backend, offering real-time feedback as users interact with the quiz. The interface ensures smooth navigation between questions, displaying real-time results and feedback, which keeps the user engaged and informed.

B. Backend Server

The backend is built with **Node.js** and the **Express.js** framework to handle all routing and HTTP requests efficiently. When a user interacts with the frontend, the server is responsible for generating structured prompts based on the topic and difficulty selected. These prompts are then sent to the **Open Router API**, where AI-generated responses are retrieved. Once the AI responses are received, the backend

ensures the data is parsed, validated, and formatted according to the frontend's expectations, ensuring high-quality and accurate question presentation. This setup ensures minimal delay in response time, offering users a smooth and reliable experience.

C. AI Integration via Open Router API

The system utilizes the **Open Router API** to connect with advanced language models capable of generating relevant quiz questions. Upon receiving structured prompts from the backend, the API returns contextually appropriate and grammatically sound responses. These AI-generated questions are tailored to the user's topic and difficulty selection, ensuring that the quiz is engaging and educational. By leveraging Open Router's support for structured outputs in a predefined **JSON schema**, the backend can easily parse and integrate the questions, ensuring that they meet the standards of clarity, relevance, and accuracy before being presented to the user.

D. Deployment and Scalability

The application is deployed on scalable cloud platforms such as **Vercel** or **AWS**, offering robust performance, high availability, and rapid scaling capabilities. The system is designed to handle varying levels of user traffic without compromising speed or performance. **CI/CD pipelines** are integrated into the development process to automate testing, building, and deployment, ensuring faster updates and feature releases. This architecture allows for quick scaling to handle concurrent user requests, enabling the platform to remain responsive during high traffic periods while maintaining consistent performance across all user interactions.

E. Real-time User Interaction and Feedback

The system is designed to provide an interactive experience, where users can see real-time feedback while answering quiz questions. Upon completing a set of questions, users immediately receive a score and a detailed explanation of the answers. This instant feedback helps users understand where they made mistakes and how they can improve.

Additionally, the system can adjust difficulty dynamically based on the user’s performance, ensuring that the quiz remains challenging yet achievable. Personalized suggestions and insights are also provided, allowing users to track their learning progress and stay motivated throughout their journey.

F .Data Security and User Privacy

Ensuring the security and privacy of user data is a top priority for the system. All personal and interaction data are encrypted using industry-standard protocols, ensuring secure transmission and storage. The system complies with all relevant privacy regulations and follows best practices for data protection, including user authentication and access control measures. Additionally, detailed logging and monitoring are in place to prevent unauthorized access and maintain transparency. By safeguarding sensitive data, the system ensures users feel confident and secure while interacting with the quiz application.

G. Analytics and Reporting

The system utilizes robust analytics tools to collect comprehensive data on user performance, quiz completion rates, and overall engagement. This data is meticulously analyzed to uncover trends, identify areas for improvement, and ensure the quizzes are meeting their educational objectives effectively. Detailed insights such as user strengths and weaknesses, time spent per question, and response accuracy are gathered to fine-tune the user experience. Customizable reports are generated for administrators or educators, allowing them to monitor individual user progress, evaluate quiz difficulty, and assess the effectiveness of the content. These reports not only provide valuable feedback on how well users are progressing, but they also offer actionable insights to refine the quiz structure. By leveraging these analytics, the system can adapt future quiz content and features to better meet the evolving needs of the audience, ensuring that the learning experience remains engaging, challenging, and aligned with educational goals. audience.

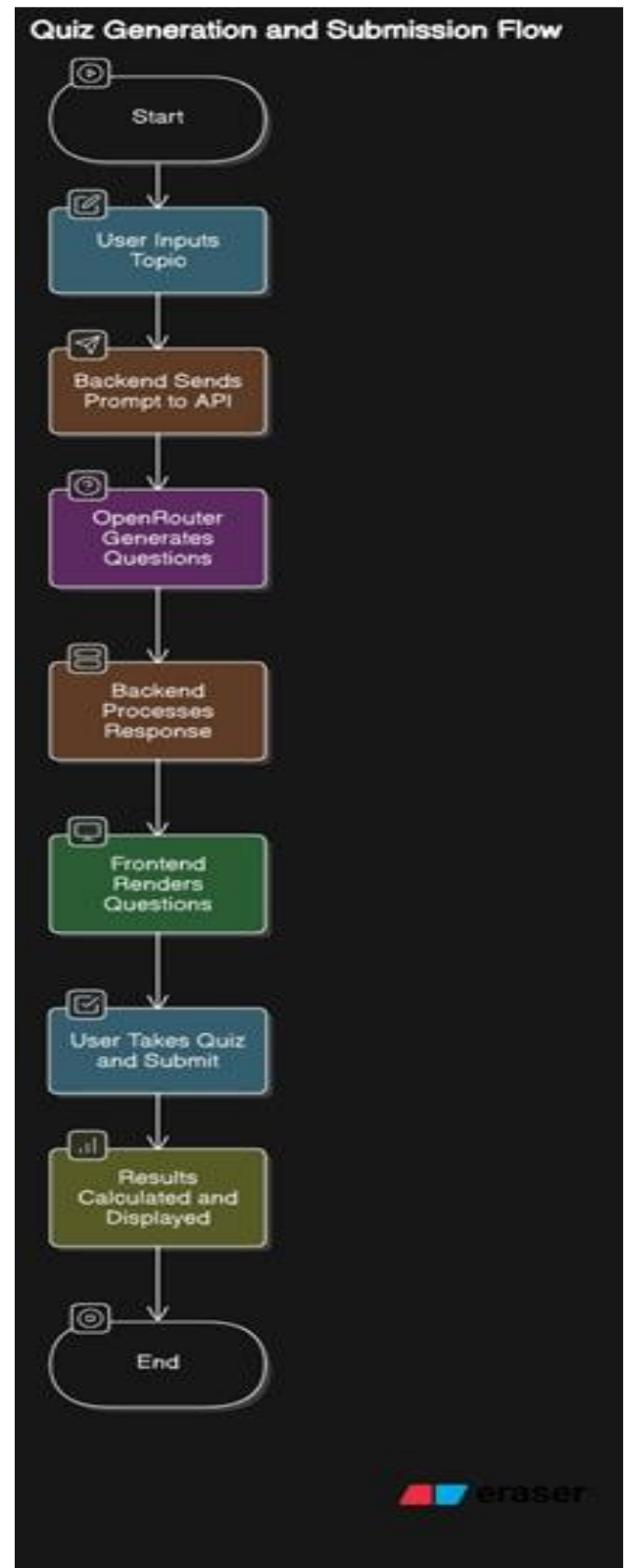
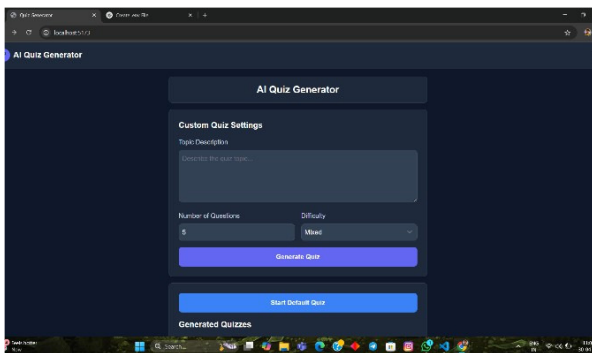


FIGURE 1: FLOW CHART

OUTPUT

A. Home Page

The journey begins on the home page of the AI quiz platform. This page serves as the central interface, enabling users to navigate through the application effortlessly. Upon arrival, users are greeted with a clean, responsive layout, where they can select their content of interest and the desired difficulty level for the quiz. The platform is designed to ensure smooth navigation, with intuitive buttons and clear instructions guiding users on how to proceed. After selecting the content and difficulty, users can initiate the quiz process by clicking the prominent "Start Quiz" button. Additionally, the page offers options for user registration and login, providing easy access for both new and returning users.



B. Content Prompt Selection

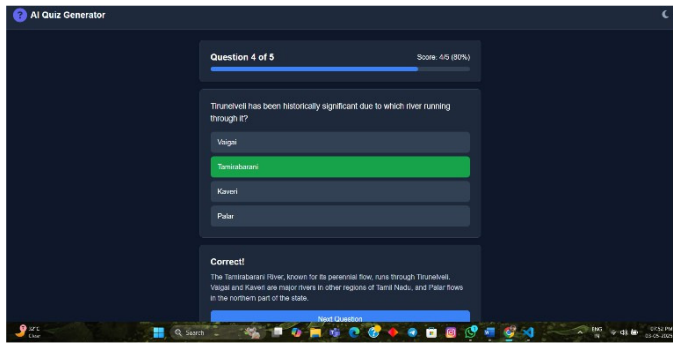
Once the user clicks the "Start Quiz" button, they are presented with the content prompt section. Here, the user can choose a subject area, such as History, Science, or Technology, and specify their preferred difficulty level—Easy, Medium, or Hard. This input is crucial as it directly influences the type of quiz the AI will generate. The system is designed to capture these preferences and send them as structured prompts to the backend, ensuring that the AI can tailor the quiz generation process according to the user's choices. Users have the flexibility to modify these selections at any point before the quiz begins.

C. Quiz Generation Using AI

After selecting the content and difficulty, the system triggers an AI-powered process to generate the quiz. The Open Router API is utilized to interact with advanced language models capable of dynamically creating a set of quiz questions based on the specified parameters. The AI generates questions in various formats, such as multiple-choice, true/false, or fill-in-the-blank, ensuring diversity and engagement throughout the quiz. Once the questions are generated, they undergo a preprocessing stage to ensure clarity, correctness, and the removal of any potential biases or inaccuracies. The validated questions are then sent back to the frontend, where they are dynamically rendered for the user.

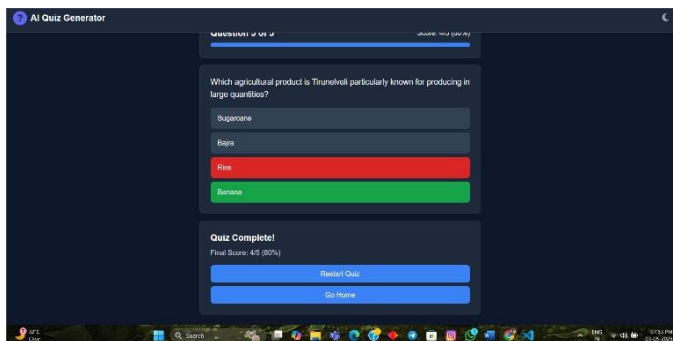
D. Taking the AI-Generated Quiz

Once the quiz questions are ready, users are guided through the quiz interface. Each question, along with its answer options, is displayed one by one, allowing users to select their responses in real-time. As users proceed through the quiz, the platform provides instant feedback for each question, indicating whether their answer was correct or incorrect while tracking their responses. The interface allows users to navigate back and forth between questions, providing flexibility. Additionally, a timer can be added to increase the challenge or track how long it takes users to complete the quiz. After answering each question, users are shown a summary of their answers, and the next question is automatically displayed.



E. Result

Once the user completes the quiz, the system processes their responses and presents a detailed result page. This page shows the total number of correct and incorrect answers and provides explanations for any incorrect responses. The result section also includes a score or grade based on the user's performance. Furthermore, users can view their historical performance, compare results over time, and receive personalized feedback or recommendations for improvement. The system may also highlight areas where further study is needed, contributing to a continuous learning experience.



CONCLUSIONS

The development of the AI-powered quiz generation platform marks a transformative step in modern education and personalized learning experiences. By integrating intelligent prompt engineering, real-time AI interaction, and seamless frontend-backend communication, the system redefines how quizzes are created, delivered, and experienced. Leveraging technologies like ReactJS, Node.js, and the Open Router API, the platform

ensures dynamic content generation, adaptability to user preferences, and an engaging user interface. It not only offers an intuitive way for learners to test their knowledge but also provides educators with powerful tools for content customization and performance tracking. As the system evolves, it lays the groundwork for advanced features such as adaptive difficulty levels, voice-enabled question answering, and multilingual support—paving the way for the next generation of AI-assisted educational platforms that are inclusive, scalable, and deeply personalized.

ACKNOWLEDGMENT

We would like to express our heartfelt gratitude to **Mrs. M. Bagya Lakshmi**, our project guide, for her invaluable guidance, continuous encouragement, and unwavering support throughout the development of this project. Her insights and expertise played a crucial role in shaping the direction and successful execution of our work. We also extend our sincere thanks to **Francis Xavier Engineering College** for providing us with the essential infrastructure, resources, and a motivating academic environment necessary for the completion of this project. A special note of appreciation goes to our teammates, **G. Gnana Vasanth** and **I. Daniel Samraj**, for their dedication, collaboration, and consistent efforts in bringing this system to life. The success of this project is a testament to the strong teamwork and shared commitment among all contributors.

REFERENCES

- [1] Y. Fu, Z. Wang, L. Yang, M. Huo, and Z. Dai, "ConQuer: A Framework for Concept-Based Quiz Generation," arXiv preprint arXiv:2503.14662, Mar. 2025.
This paper introduces the ConQuer framework, which aligns with our project's aim to generate concept-based quizzes dynamically using user-selected topics and difficulty levels.
- [2] E. Wallace, P. Rodriguez, S. Feng, I. Yamada, and J. Boyd-Graber, "Trick Me If You Can: Human-in-the-loop Generation of Adversarial Examples for

Question Answering,” arXiv preprint arXiv:1809.02701, Sep. 2018.

This study helped in understanding how human-in-the-loop techniques can improve quiz quality and adversarial robustness in automated question generation.

- [3] **J. Liu, “A Novel Interface for Adversarial Trivia Question-Writing,” arXiv preprint arXiv:2404.00011, Mar.2024.**

Provided insights on designing intuitive interfaces for trivia-based quiz systems, influencing our user-friendly front-end design.

- [4] **G. Boateng, V. Kumbol, and E. E. Kaufmann, “Can an AI Win Ghana’s National Science and Maths Quiz? An AI Grand Challenge for Education,” arXiv preprint arXiv:2301.13089, Jan.2023.**

Explores AI’s role in competitive quizzes, which inspired the incorporation of real-time performance tracking and scoring in our platform.

- [5] **ABM Technologies, “AI-Powered Quiz Generation: Transforming Educational Assessment,” ABM Technologies Blog, Jan.2025.**

Offered real-world applications of AI in assessment tools, reinforcing the practical viability of our project in academic environments.

- [6] **Monsha.AI, “Top 7 AI Quiz Generator Tools for Teachers,” Monsha.AI Blog, Mar.2025.**

Provided comparative insights into current tools, helping us design a more adaptive and flexible quiz generation system.

- [7] **Wikipedia Contributors, “Artificial Intelligence in Education,” Wikipedia, Apr.2025.**

Helped frame the background and significance of integrating AI into educational tools, giving our project a strong theoretical foundation.

- [8] **Harvard Online, “The Benefits and Limitations of Generative AI,” Harvard Online Blog, 2023.**

Provided a balanced view on generative AI, shaping our understanding of ethical considerations and system limitations.

- [9] **Sendsteps, “Exploring the Future of Online Quizzes with AI Technology,” Sendsteps Blog, 2025.**

Informed our approach to future scalability and AI-driven quiz personalization.

- [10] **Eggheads.ai, “AI Quiz Generator,” Eggheads.ai, 2024. Served as a reference for understanding back-end architecture and feature design of modern AI quiz generators.**

The study investigates the design and development of voice-activated AI systems for wellness, focusing on their impact on fitness and health management.