



ZENTRIX_VERN QUIZ APPLICATION

A Web-Based Interactive Quiz System

S Roshan & F. Mohammed Jafraan

Bachelor of Computer Applications (BCA)

Shyamala Devi

Department of Computer Applications

ABSTRACT

The Zentrix_Vern Quiz Application is a modern, web-based quiz platform designed to replace inefficient paper-based testing methods with a fast, automated, and user-friendly digital solution. The application is built using HTML for page structure, CSS for visual styling, and JavaScript for interactive quiz logic including real-time answer validation and score calculation. Google Firebase is used as the cloud-based backend to store quiz questions and save user results securely without the need for a traditional server. The system presents multiple-choice questions one at a time, evaluates answers instantly, and displays a detailed score report upon completion. It is accessible from any modern web browser, making it suitable for students, educators, and training platforms. This project demonstrates practical integration of frontend web technologies with a cloud database, producing a scalable, cost-effective, and efficient quiz solution.

Keywords: *Online Quiz System , Web-based Application , HTML , CSS , JavaScript , Firebase , Automated Evaluation , MCQ*

INTRODUCTION

A quiz system is a software tool that presents a set of questions to users and automatically evaluates their responses. In traditional education settings, quizzes were conducted using printed papers, which required significant time and effort for creation, distribution, evaluation, and result management. With the advancement of web technologies, it has become possible to automate this entire process using browser-based applications that are fast, reliable, and easy to use.

The Zentrix_Vern Quiz Application was developed to address the limitations of traditional quiz methods by offering a fully digital alternative. The application is designed to load questions dynamically from a Firebase database, display them to the user through a clean interface, evaluate answers in real time using JavaScript, and store the results securely in the cloud. The project was developed as part of the final year BCA curriculum to

demonstrate the practical use of frontend web development combined with cloud database integration.

Objectives

- To build a fully browser-based quiz application that requires no software installation.
- To implement real-time answer validation and automatic score calculation using JavaScript.
- To use Firebase as a cloud backend for storing and retrieving quiz data and results.
- To deliver a responsive, visually clean UI that works on desktops, tablets, and mobile devices.
- To eliminate manual effort in quiz evaluation and result management.

PROPOSED SYSTEM

The Zentrix_Vern Quiz Application is the proposed solution to the problems found in conventional quiz systems. Unlike paper-based or basic offline digital quizzes, this application leverages cloud technology through Firebase to provide a seamless and automated quiz experience that can be accessed from anywhere with an internet connection.

The application features a home page with a Start Quiz button, a quiz interface that presents one multiple-choice question at a time, and a result page that displays the final score upon completion. All quiz questions are stored in the Firebase Realtime Database and fetched dynamically when the quiz begins. User results are saved back to Firebase automatically after quiz completion, ensuring data is always persistent and accessible for review.

Key Features

- Dynamic question loading from Firebase — no hardcoded questions in the frontend code.
- Real-time answer validation with instant visual feedback for correct and incorrect selections.
- Automatic score calculation and a detailed result summary at the end of the quiz.
- Responsive design using CSS that adapts to different screen sizes and devices.
- Secure cloud storage of quiz data and results through Google Firebase.
- Simple restart option to retake the quiz without refreshing the page manually.

Compared to the existing system, the proposed application eliminates printing costs, reduces evaluation time to zero, provides instant results, and stores all data securely in the cloud. It is scalable, easy to maintain, and suitable for use in schools, colleges, and online training platforms.

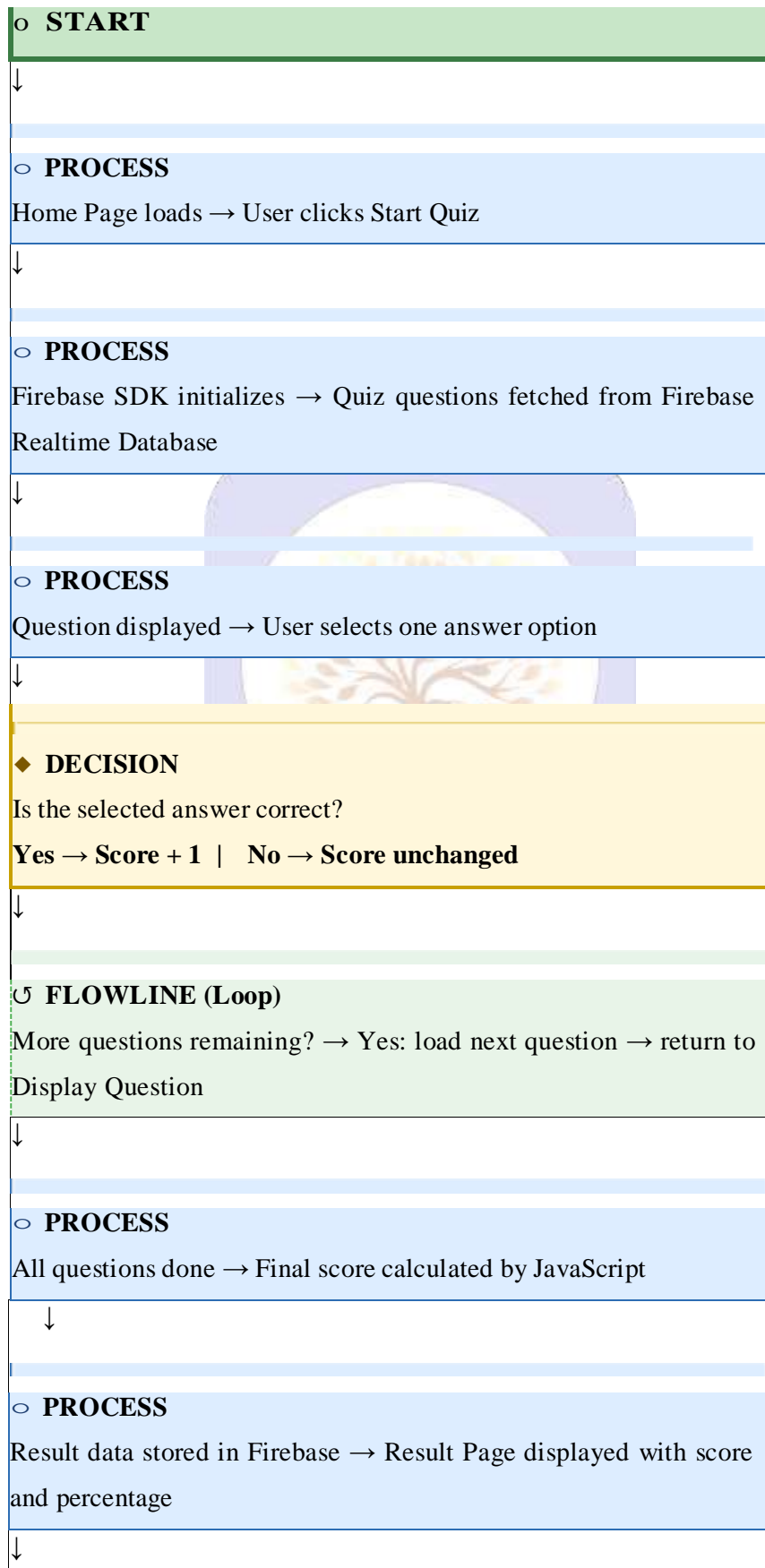
SYSTEM DESIGN

Architecture Overview

The Zentrix_Vern Quiz Application follows a two-layer client-server architecture. The frontend layer, built with HTML, CSS, and JavaScript, runs entirely in the user's browser and handles all user interactions, question display, and score logic. The backend layer is powered by Google Firebase, which provides cloud-based data storage and retrieval without requiring a traditional application server or server-side language. The Firebase JavaScript SDK is linked via CDN and initialized within the frontend code to enable seamless communication between the browser and the Firebase database. This serverless approach keeps the application lightweight, fast to deploy, and easy to maintain.

Application Flowchart

Legend: ○ **Oval** = Start / End ◻ **Rectangle** = Process Step ◆ **Diamond** = Decision → **Arrow** = Flowline



END

The flowchart above follows the standard flowchart convention: oval shapes represent the Start and End terminals, rectangles represent process steps, the diamond represents the decision point (correct or incorrect answer), and arrows indicate the flowlines connecting each stage of the quiz application.

MODULE DESCRIPTION

The application is divided into five functional modules, each handling a specific part of the quiz workflow. This modular approach ensures clean code organization and easier maintenance.

1. User Interface (UI) Module

This module is responsible for the visual presentation of the entire application. Built using HTML and CSS, it consists of three pages: the Home Page, the Quiz Page, and the Result Page. The Home Page displays the application name and a Start button. The Quiz Page shows the current question and answer options. The Result Page presents the final score and performance summary. CSS ensures the layout is responsive, visually consistent, and accessible across screen sizes.

2. Quiz Initialization Module

When the user clicks Start Quiz, this module activates the Firebase connection using the SDK. It fetches all quiz questions from the Firebase Realtime Database, initializes state variables such as current question index and score counter, and passes the data to the Question Display Module. It handles the loading phase and ensures that the quiz begins only after data is successfully retrieved.

3. Question Display Module

This module renders each question and its four answer options on the Quiz Page using JavaScript DOM manipulation. It tracks the current question index, updates the interface when the user moves to the next question, and shows a question progress indicator. Questions are displayed sequentially, and the module loops through all fetched questions until the last one is answered.

4. Answer Evaluation Module

This is the core logic module of the application. When the user selects an answer, this module compares the choice with the correct answer stored in the Firebase data. If the answer is correct, the score is incremented and visual feedback (such as a green highlight) is shown. If incorrect, the wrong option is marked and optionally the correct answer is revealed. After evaluation, the module triggers progression to the next question.

5. Result and Storage Module

Once all questions are answered, this module calculates the total score, formats the result data including total questions, correct answers, incorrect answers, and percentage, and writes this information to the Firebase database. It then renders the Result Page with a clear score summary. A Restart button is also provided, allowing the user to retake the quiz without reloading the page.

IMPLEMENTATION

Technology Roles

HTML provides the structural skeleton of the application. It defines the layout of the Home Page, Quiz Page, and Result Page using semantic elements such as headings, divs, buttons, and paragraphs. The Firebase JavaScript SDK is linked in the HTML file using a script tag pointing to the Google CDN.

CSS handles all visual styling including colors, fonts, spacing, button effects, and responsive layouts. Flexbox is used for centering and aligning content. Hover and active pseudo-classes add smooth interactive effects to buttons. Media queries ensure the design adapts properly to mobile and tablet screens.

JavaScript contains all quiz logic. It initializes Firebase using the project configuration object, fetches questions using Firebase read methods, renders questions dynamically via DOM manipulation, validates answers, updates the score, and writes results back to Firebase using async/await-based write operations.

Firebase serves as the backend. Quiz questions are stored in a structured JSON format in Firebase Realtime Database. Each question contains fields for question text, four options, and the correct answer index. After quiz completion, result data including score and timestamp is saved to Firebase, making it retrievable for future analysis.

System Requirements

Software requirements include Windows 10/11 or macOS as the operating system, Google Chrome or Mozilla Firefox as the browser, and Visual Studio Code as the code editor. The application is built with HTML, CSS, and JavaScript, with Firebase as the backend. Git is optionally used for version control. Hardware requirements include an Intel Core i3 processor or above, a minimum of 4 GB RAM (8 GB recommended), at least 500 MB of free storage space, a 64-bit system, and a stable internet connection for Firebase access.

RESULTS AND SCREENSHOTS

The Zentrix_Vern Quiz Application was successfully tested across multiple browsers and devices. All core functionalities including question loading from Firebase, real-time answer evaluation, score calculation, result display, and Firebase data storage performed as expected. The following screenshots illustrate the key interfaces and features of the application.

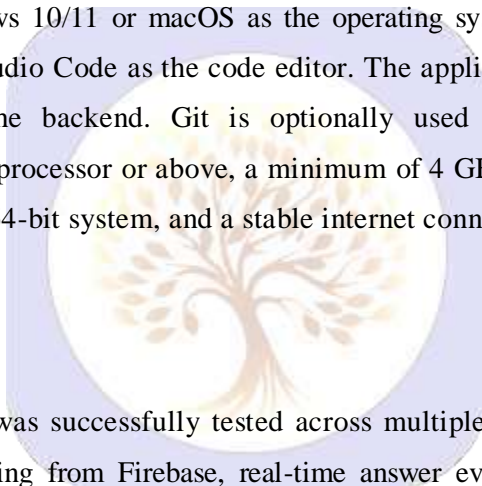
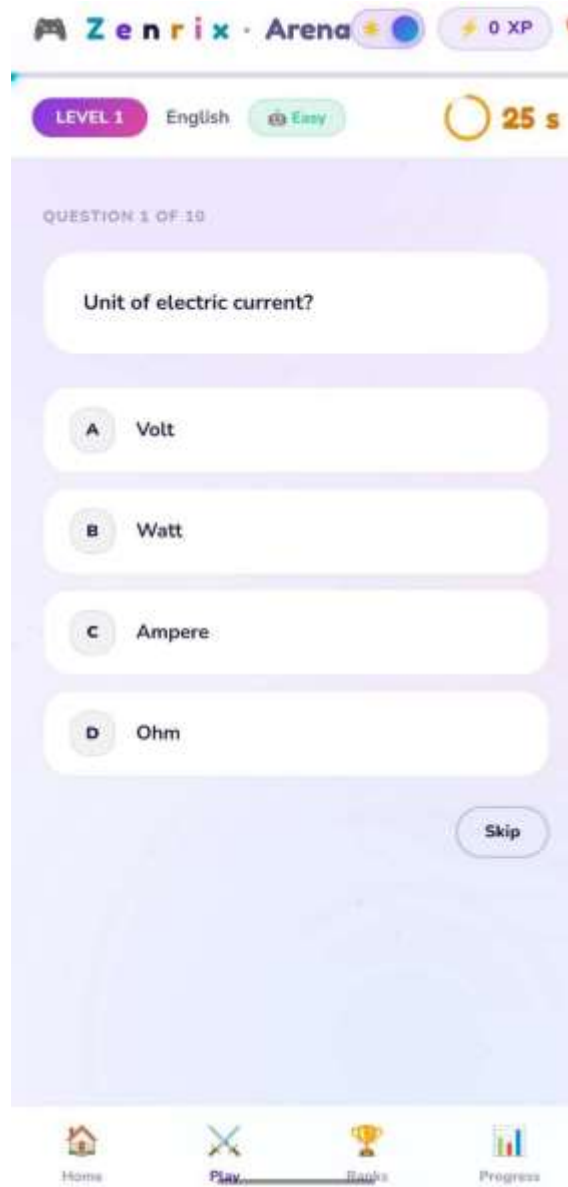


Figure 1: Home Page of Zentrix_Vern Quiz Application

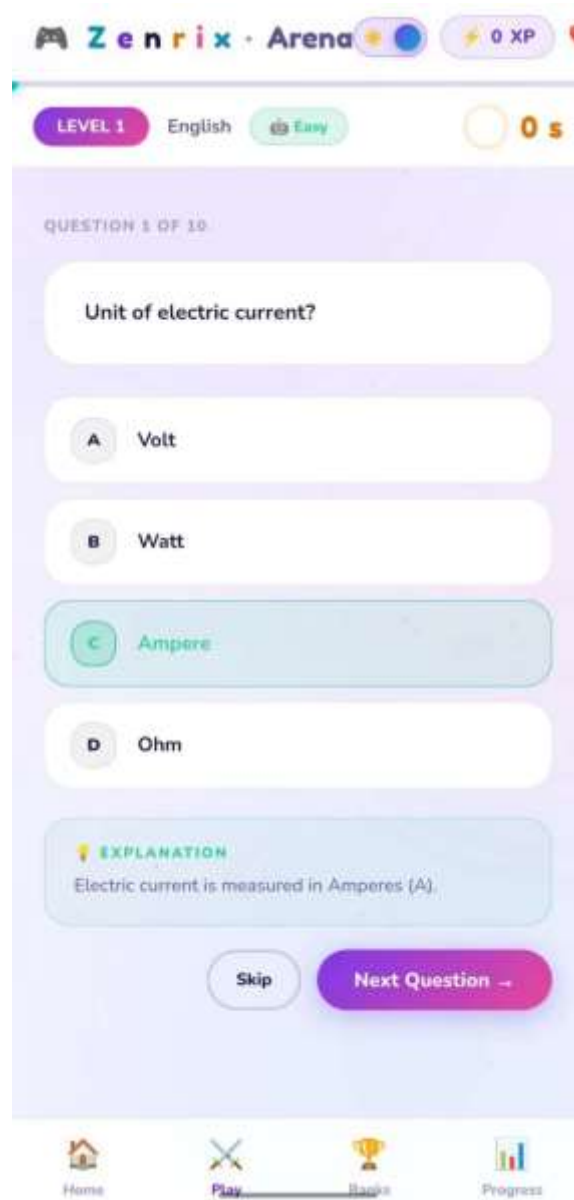


Fig. 1: Home Page of Zentrix_Vern Quiz Application

This screen shows the main landing page of the application. It displays the application title, a brief description, and a prominent Start Quiz button. The clean layout ensures the user immediately understands how to begin the quiz.

Figure 2: Quiz Question Interface*Fig. 2: Quiz Question Interface*

This screen shows how each question is presented to the user. The question text is displayed clearly at the top, followed by four labelled multiple-choice options below. A question progress counter is visible to indicate how far the user is in the quiz.

Figure 3: Answer Selection Screen*Fig. 3: Answer Selection Screen*

This screen captures the moment a user selects an answer option. The selected option is visually highlighted to confirm the choice. If configured, incorrect selections are marked in red while the correct answer is shown in green before moving to the next question.

Figure 4: Result Page with Score Display*Fig. 4: Result Page with Score Display*

This screen shows the final result page displayed after the user answers all questions. It presents the total score, number of correct and incorrect answers, and the overall performance percentage. A Restart Quiz button allows the user to attempt the quiz again.

Figure 5: Firebase Database Console

Firebase Realtime Database console with the stored quiz questions and user result entries. Each question is saved in a structured JSON format, and result records include the score and timestamp, demonstrating successful cloud data storage.

CONCLUSION

The Zentrix_Vern Quiz Application was successfully designed and developed as a complete web-based quiz platform using HTML, CSS, JavaScript, and Google Firebase. The application fulfills all stated objectives by providing an automated, user-friendly, and cloud-connected quiz experience that eliminates the limitations of traditional paper-based methods.

Through this project, the development team gained hands-on experience in building interactive web applications, integrating cloud database services, implementing real-time data operations, and designing responsive user interfaces. The modular architecture of the application ensures it is easy to maintain and

extend with additional features in the future.

Future enhancements planned for this application include user authentication via Firebase Auth, multiple quiz categories, a real-time leaderboard, timer-based questions, and an admin dashboard for managing quiz content. Overall, the Zentrix_Vern Quiz Application serves as a practical and well-executed example of applying modern web and cloud technologies to solve a real-world educational problem.

REFERENCES

- [1] **Google Firebase Documentation** — Firebase Realtime Database and Firestore Guide. Google LLC, 2024. Available at: <https://firebase.google.com/docs>
- [2] **Mozilla Developer Network (MDN)** — HTML5, CSS3 and JavaScript Web API Reference. Mozilla Foundation, 2024. Available at: <https://developer.mozilla.org>
- [3] **Jon Duckett** — HTML and CSS: Design and Build Websites. John Wiley & Sons, Inc., 2011. ISBN: 978-1-118-00818-8.
- [4] **Jon Duckett** — JavaScript and jQuery: Interactive Front-End Web Development. John Wiley & Sons, Inc., 2014. ISBN: 978-1-118-53164-8.
- [5] **W3Schools Online Web Tutorials** — HTML, CSS, JavaScript and Firebase Integration Guides. Refsnes Data, 2024. Available at: <https://www.w3schools.com>
- [6] **Roger S. Pressman** — Software Engineering: A Practitioner's Approach, 8th Edition. McGraw-Hill Education, 2014.
- [7] **Visual Studio Code Documentation** — Getting Started Guide and Extensions Reference. Microsoft Corporation, 2024. Available at: <https://code.visualstudio.com/docs>

