THE UNIVERSITY OF NEWCASTLE
SCHOOL OF INFORMATION AND PHYSICAL SCIENCES

COMPUTING AND INFORMATION SCIENCES
WORK INTEGRATED LEARNING
COMP3851A – SEMESTER 1, 2024

Project Plan

Authors:
ANSAR DANIYARULY, C3512057

RAUAN ZHOLMAN, C3501153

CHAN MYAE THINZAR TUN, C3512052

MAY THU KYAING, C3512127

THIN THET HTAR SAN, C3502751

JIAN XIN PNG, C3512213

QuizMaster – Intelligent Quiz Creation and Assessment Platform

1. Background

In modern education, the capability to measure students' knowledge effectively and track their progress is essential to higher learning achievement. With the use of electronic learning in every aspect of education, the demand for adaptive, efficient, and intelligent test tools has increased immensely (McMillan, 2020). The creation of the QuizMaster system was with the intent of responding to the challenges teachers face while designing quizzes, analyzing results, and getting the attention of students.

Teachers nowadays rely heavily on existing tools like Google Forms, Moodle, and Kahoot to administer tests. While these tools offer easy quiz creation and delivery features, they fall short in the areas of automated grading flexibility, real-time feedback, adaptive analytics, and ongoing web and mobile experience (MoodleDocs, 2024; Kahoot!, 2024). These limitations prevent teachers from seeing student learning in anything other than simple scores and students from receiving effective feedback on their performance.

The underlying problem QuizMaster addresses is the lack of a unified, intelligent system to perform both formative (ongoing) and summative (ultimate) assessments, along with tools that promote reflective learning. For instance, formative assessments typically include rich feedback and student retakes or self-assessment functionality, while summative assessments require protection, timing limitation, and marking accuracy. Existing tools do not generally solve both use cases equally well.

The motivation behind this project stems from a shared vision among the development team to create a setting that is not only feature-rich but also accessible, interactive, and informed. Through the integration of AI-based question suggestions, real-time analysis, and gamification learning capabilities, QuizMaster aims to enhance teaching efficacy and learner engagement.

In terms of feasibility, the project is well within reach. The development team comprises six individuals with a varying set of complementary skills in frontend and backend development, mobile app development, user interface design, project management, and technical writing. The technologies to be implemented—such as Firebase, React.js, and Flutter—are widely used and well-supported, and the team will be able to develop scalable, secure, and high-performing solutions within the project timeline.
Lastly, QuizMaster is a working, timely answer to a real issue in online education, and its implementation is well within the technical and organizational capabilities of the team.

2. Aims

The main objective of the QuizMaster initiative is to create an expansive, yet easy-to-use web and mobile program which simplifies the process of creating, distributing, and evaluating quizzes for teachers, as well as providing students with useful self-assessment mechanisms. The initiative seeks to do this by first creating an adaptable quiz generating module, which makes it easy for teachers to create quizzes with a wide variety of question formats, including multiple choice, true/false, and short answer questions. The module will allow this to be done in order to create interesting and diverse evaluations suited to various learning goals Second, the product will also include flexible assessment configurations, which allow for the capability of categorizing quizzes either as formative, used to deliver feedback as students learn, or summative, used to determine final grades. There are configurable time and open-access availability, allowing the platform to be used in various teaching approaches and assessment requirements. Of utmost importance, too, is automating the grading of appropriate question types and generating strong, comprehensive performance analytics for students as well as whole-class performance, giving teachers valuable feedback on students' understanding and patterns of learning. In addition, QuizMaster seeks to incorporate student self-assessment and tracking capabilities, providing students with independent tools to self-test and monitor their learning over time. This encourages autonomy in students and helps them better understand areas of academic excellence and areas where improvement is needed. Lastly, the project provides cross-platform access and data management efficiency by providing responsive web-based application accessibility on various devices (desktop and mobile) and the incorporation of a Firebase backend for secure real-time data storage, authentication, and handling of the question bank effectively. The platform will be able to export quiz analytics and results in standard formats (i.e., PDF/Excel) to allow easy record-keeping and sharing.

3. Methods and Activities

Requirements Analysis and Planning:

In this activity, functional requirements are defined, user needs are gathered, and user stories are created. Gantt charts and Lucid chart are two examples of tools which assist with process planning, milestone setting, and risk identification. Clear documentation provides the groundwork for efficient and organised development and guarantees alignment with stakeholder expectations.

Designing the User Interface (UI) of the web app:

The goal of this activity is to plan and design an intuitive web application, while taking into consideration elements that makes taking the quiz engaging. During the design phase, our team will utilise prototype development tools such as Draw.io and Figma to create wireframes and high-fidelity prototypes. Once the design has been finalised, we will leverage HTML to create the layout of the webpage, and CSS to keep the web page visually appealing to users (Bhanarkar et al., 2023). Interactivity can be enhanced with JavaScript or frameworks like React, which can handle dynamic content and transitions between quiz states and pages efficiently. Furthermore, our team will incorporate accessibility principles and ensure that navigation, color schemes, and font styles meet usability standards.

Backend Development of the web app:

In conjunction with front-end work, the back-end infrastructure that powers the quiz operations will be developed, which is the goal of this activity. Our team will use Firebase or Golang as the primary programming languages for this activity. RESTful APIs will be configured to handle quiz data, user sessions, and result administration. Our team will also employ databases like MySQL or MongoDB to store quiz questions, user responses, and analytics data. To streamline the coding process, we will use Visual Studio as our IDE. Furthermore, we will implement server-side validation to supplement front-end checks. This activity also involves creating a robust data model and ensuring that all components interact seamlessly across different modules and layers of the application.

System Integration and API Connectivity:

This stage ensures that the database, mobile, frontend, and backend components work seamlessly together. Firebase services are integrated, and Postman is used to test APIs. Role-based access and analytics are managed across platforms without UI or logic errors, and the team guarantees safe authentication, real-time updates, and smooth data transmission (Banerjee, 2024).

Testing, Deployment, and Maintenance:

This activity ensures stability and user satisfaction by adopting unit, integration, and end-to-end testing with Jest for JavaScript, JUnit for Java, and NUnit for C#. Automated frameworks like Selenium and Cypress simulate end-user scenarios across browsers. Bug fixes and performance issues are resolved before deployment. CI/CD pipelines using GitHub Actions or GitLab CI automate builds and deployments. Hosting on Azure ensures scalability and reliability. The application will be deployed in a controlled environment, monitored continuously, and maintained regularly based on user feedback.

Project Management and Collaboration:

With two-week sprints, the project follows Agile methodology and is monitored using Jira or GitHub Projects. GitHub handles bugs, code reviews, and version control. Tools such as Microsoft Teams and regular meetings facilitate collaboration. Milestones are coordinated utilising a shared calendar. Peer reviews and coding standards guarantee the quality of the documentation, which is versioned in Notion or GitHub Wiki.

4. Ethics, intellectual property and confidentiality considerations

QuizMaster is the pioneering project into ethical software development, considering issues of fairness, privacy, or responsible use of content. Given that the quizzes are AI-generated and that student performance is analysed, the team will work to review datasets and algorithms to ensure learning experiences remain as bias-free as possible.

User data are collected after gaining consent, and the data are held in a secure storage system with access control. They comply with data protection laws such as the GDPR, to ensure that privacy and accountability are maintained.

Intellectual property rights created in the form of source codes, designs, and quiz content during the development process shall form the property of the team. All external tools and resources used must comply with the licenses and be duly acknowledged.

Respect for confidentiality and ethics builds trust and upholds standards for professionalism throughout the life cycle of the project (Binns, 2018).

The QuizMaster project follows strict ethical standards to ensure privacy, ownership rights, and responsible development. Since the system handles sensitive student data, protecting user privacy is critical. Firebase Authentication and secure API practices will ensure that access to data is restricted to authorized users only, complying with relevant data protection laws such as the GDPR (Voigt & Von dem Bussche, 2017).

All original code, content, and media developed are considered the intellectual property (IP) of the team, unless otherwise agreed upon. Open-source tools like React.js and Flutter will be used in accordance with their licenses, with proper attribution maintained (Open-Source Initiative, 2021). Reuse of proprietary content without permission is strictly avoided.

Confidentiality is also prioritised. Project documentation, user data, and source code are stored on secured platforms like GitHub with access restrictions. Team members are expected to uphold non-disclosure principles. Following ethical software development standards ensures trust, sustainability, and professional integrity (Gotterbarn et al., 2018).

References

Banerjee, P. (2024) ‘System integration, from middleware to apis’, *International Journal of Computer Trends and Technology*, 72(3), pp. 37–45. doi:10.14445/22312803/ijctt-v72i3p106.

Bhanarkar, N., Paul, A. and Mehta, A. (2023) ‘Responsive web design and its impact on User Experience’, *International Journal of Advanced Research in Science, Communication and Technology*, 3(4), pp. 50–55. doi:10.48175/ijarsct-9259.

Binns, R.(2018). *Fairness in Machine Learning.* https://dl.acm.org/doi/10.1145/3287560.3287598

Gotterbarn, D., Miller, K., & Rogerson, S. (2018). *ACM Code of Ethics and Professional Conduct*. https://www.acm.org/code-of-ethics

Kahoot! (2024). *What is Kahoot?*. [online] Available at: https://kahoot.com/what-is-kahoot

McMillan, J.H. (2020). Classroom Assessment: Principles and Practice for Effective Standards-Based Instruction. 7th ed. Boston: Pearson Education. https://archive.org/details/classroomassessm0000mcmi_o2h4/page/n5/mode/2up

Open Source Initiative. (2021). *The open source definition*. https://opensource.org/osd

Voigt, P., & Von dem Bussche, A. (2017). *The EU General Data Protection Regulation (GDPR): A Practical Guide*. Springer. https://link.springer.com/book/10.1007/978-3-031-62328-8