

Development of an English Proficiency Assessment System: A Web-Based Approach

Anwar Hilman^{1*}, Wafiqah Yasmin A.², Vivi Ayu Lestari³, Ahmad Najib Mutawaly⁴, Supriyadi⁵

¹Program Studi Informatika, Fakultas Teknologi Informasi dan Komputer, Universitas Horizon Indonesia,

²Program Studi Sistem Informasi, Fakultas Teknologi Informasi dan Komputer, Universitas Horizon Indonesia,

Jalan Pangkal Perjungan No.1 By Pass, Karawang, Provinsi Jawa Barat, 41316, Indonesia

e-mail : anwar.hilman.krw@horizon.ac.id

Abstract: English proficiency assessment is a vital component of the learning process, especially in tailoring instructional materials to students' needs and abilities. This study aims to develop and implement a web-based English proficiency assessment system that is efficient, adaptive, and capable of providing real-time feedback to users. The research adopts a Design and Development Research (DDR) approach, utilizing the Spiral Model as the software development methodology. The implementation results demonstrate that the system improves the efficiency of assessment processes, reduces manual grading errors, and offers a more interactive learning experience. User evaluations indicate a high level of satisfaction, particularly in terms of accessibility, quick result display, and user-friendly interface design. The study concludes that the developed system not only enables more accurate and relevant language skill assessment, but also contributes to enhancing students' learning motivation and the overall effectiveness of English language instruction in digital environments.

Keywords: english assessment; web-based system; system development

INTRODUCTION

In the globalization era, English proficiency has become a fundamental skill required in education, career development, and international communication. The increasing demand for a competent English-speaking workforce encourages educational institutions to conduct effective and efficient assessments of language skills. However, traditional English proficiency assessments are often time-consuming, subjective, and inefficient when applied on a large scale, creating gaps in accurate evaluation and timely reporting.

Advancements in information technology, especially web-based applications, offer solutions to these challenges by enabling integrated, automated, and accessible assessment processes. A web-based English proficiency assessment system can streamline the entire assessment workflow—from participant registration, online testing with automatic scoring for objective question types, to result processing and reporting—thus improving the efficiency, objectivity, and scalability of the evaluation process. Previous studies have shown that such systems also enhance data analysis, support decision-making in curriculum development, and increase assessment accessibility.

The following table presents a comparison between the previous state of the web-based assessment system (shown in the Present column) and the ideal conditions expected in the newly

developed system (shown in the Expectation column). This comparison covers various technical and functional aspects that previously posed challenges, such as network issues, browser compatibility, and user interface limitations, while outlining expected improvements in system stability, user experience, and the reliability of core assessment features.

Table 1. Comparison Between Existing System Conditions and Expected Improvements

Present	Espectation
Network problems (network problems)	Improved network stability and automatic recovery when disconnected
Browser compatibility issues	Optimization to be compatible across all browsers and devices
Audio doesn't work	Improved audio listening quality and clarity
The test page does not load or freezes	Autosave feature for answers to each question
Login error or account locked	The interface (UI) is more responsive and user-friendly
The test is interrupted and answers are not saved	Clear error notifications with suggested solutions

This study focuses on the development of a web-based English proficiency assessment system tailored to educational needs. It explores the design process, system development, and practical implementation, while identifying the challenges and opportunities faced during its application in the educational context. The results of this research are expected to contribute to a better understanding of how web-based assessment systems can improve the quality of English proficiency evaluations and provide practical recommendations for future system development.

LITERATURE REVIEW

Web-based English proficiency assessments offer an alternative approach that improves the validity and reliability of the evaluation process. Research by Emmiyati et al. (2023) confirmed that online testing systems provide high validity and acceptable reliability levels, with practical benefits for both learners and administrators. Complementing this, Fitriyah and Widiati (2021) emphasized that classroom-based assessments allow teachers to conduct continuous evaluations that reflect real learning conditions, thus improving students' comprehension and satisfaction. Taken together, these studies highlight the need for an effective English assessment method that combines validity, efficiency, and contextual relevance, making technology-enabled solutions increasingly vital.

Septiarananda et al. (2022) also demonstrated that students perceive web-based assessments positively, appreciating their convenience and accessibility, although technical constraints such as server stability and internet connectivity remain challenges. From these studies, it can be concluded that web-based assessment systems not only improve the efficiency and accessibility of language testing but also enhance user engagement and data-driven analysis for continuous improvement.

Design and Development Research (DDR) is a methodological approach that emphasizes both the creation and validation of innovative educational technologies (Sabri Sahrir et al., 2012). In educational technology development, DDR facilitates the iterative creation of prototypes and continuous formative evaluations to refine products in real learning contexts. Sabri Sahrir et al. (2012) successfully applied DDR in developing a web-based Arabic vocabulary learning game, showing the approach's effectiveness in aligning product design with user needs.

In parallel, Boehm (1986) introduced the Spiral Model as a software development approach characterized by iterative cycles that integrate risk analysis, system design, implementation, and evaluation. Sommerville (2011) and Pressman (2014) reinforced the Spiral Model's relevance in developing complex and adaptive systems, such as web applications, where continuous refinement based on feedback and risk management is crucial. Combining the DDR approach with the Spiral Model offers a robust framework for educational system development, ensuring user-centered design, iterative testing, and risk-mitigated development processes.

RESEARCH METHODS

Research Approach

This study adopts a Design and Development Research (DDR) approach. DDR is a research method aimed at designing and developing specific products or systems, as well as evaluating their effectiveness and efficiency. In the context of this research, DDR is used to develop a web-based information system that functions as an English proficiency assessment tool.

System Development Model

The system development process in this research follows the Spiral Model, one of the models in the Software Development Life Cycle (SDLC). The Spiral Model is selected because it emphasizes iterative and incremental development, allowing the system to be developed in stages through repeated evaluation and improvement. This model also facilitates risk identification at

each cycle, resulting in a more stable system that meets user needs. The software development model using the spiral method is illustrated in the following figure, adapted from the study by Sidiki et al. (2023).

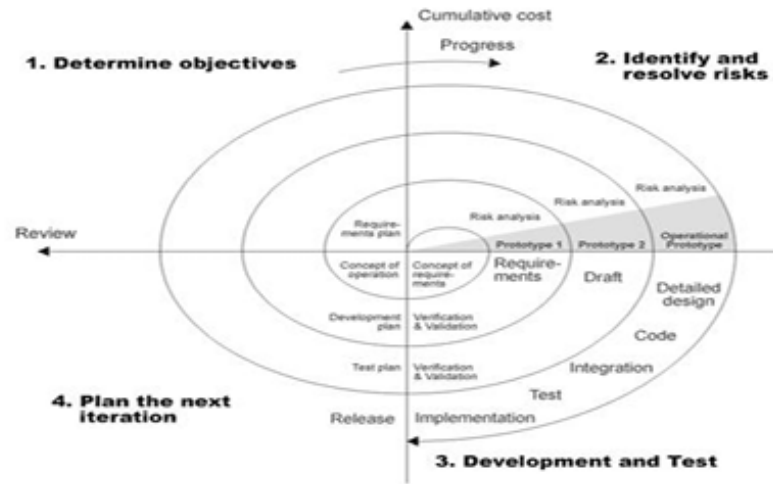


Figure 1. SDLC Spiral Model

Research Procedure

The overall procedure of this research consists of the following stages:

1. Needs Analysis

Conducting literature reviews and interviews with potential users (such as teachers or school administrators) to determine the functional and non-functional requirements of the system.

2. System Design

Designing the system using diagrams such as use case diagrams, activity diagrams, and database schemas according to the application's needs.

3. Application Development

Developing the web-based application using appropriate web programming technologies and frameworks.

4. System Testing

Testing the system using black box testing methods to verify whether the system functions as intended.

5. Implementation and Evaluation

Deploying the system in an educational environment for actual use, followed by evaluating its effectiveness.

RESULTS

1. Needs Analysis

This process was carried out through a literature review related to web-based English proficiency assessment systems and best practices in educational assessment. In addition, qualitative interviews were conducted with potential users, such as English lecturers and students, to understand their expectations, the assessment processes they commonly perform, and the challenges they face. The outcome of this stage was a system requirements document that outlined the core application functions, user interface design considerations, security aspects, and system performance expectations.

2. System Design

The system design stage was carried out using various modeling tools, such as class diagrams. A use case diagram was created to describe the interactions between users and the system. Activity diagrams were developed to illustrate the process flows, such as user registration, test execution, and assessment mechanisms.

a. Class Diagram

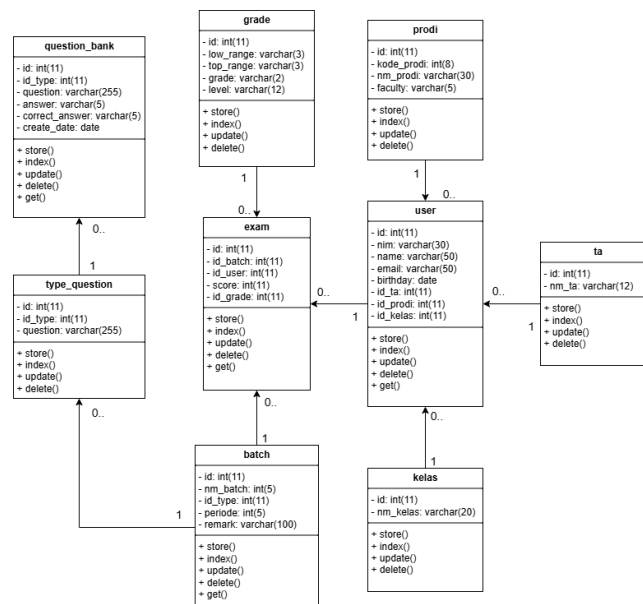


Figure 2. Class Diagram

This class diagram illustrates the structure of a web-based English proficiency assessment system consisting of several main entities, including question_bank, type_question, exam, grade, batch, user, prodi (study program), kelas (class), and ta (academic year). The question_bank entity stores question data categorized by type_question. Exam results are stored in the exam entity, which is linked to batch (exam

periods), user (participants), and grade (score categories). The batch entity organizes exams based on periods and question types. The user entity contains student information, including relationships with prodi, kelas, and ta (academic year). Each entity has clear one-to-many relationships, such as one user participating in multiple exams and one batch containing many exams. The diagram's structure demonstrates a modular and well-organized system, supporting effective and structured management of questions, participants, and exam results.

b. Use Case Diagram

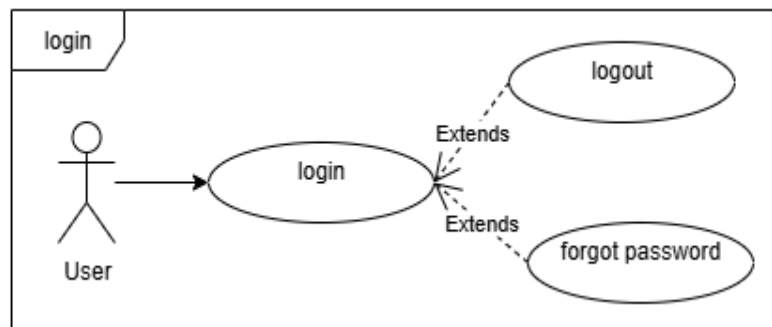


Figure 3. Use Case Login

This use case diagram illustrates the interaction between the User actor and the system in the authentication process. The user can perform two main activities: login and logout, both represented as independent use cases directly connected to the actor. In the login process, there is an additional feature, forgot password, represented as an extension use case using the <<extend>> relationship, indicating that this feature is optional and only triggered when the user encounters login issues. The structure of this diagram adheres to UML standards, with a clear distinction between primary processes and supporting features.

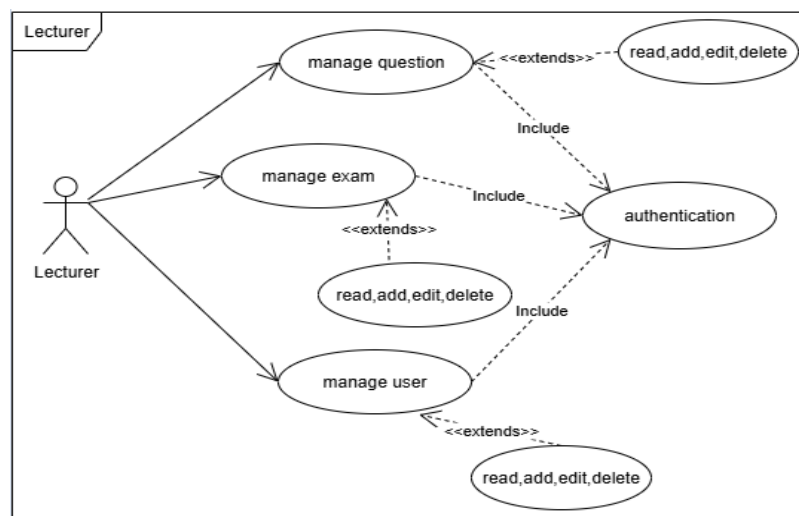


Figure 4. Use Case Diagram Lecturer Actor

This use case diagram illustrates the interactions between the Lecturer actor and the system's core functionalities, including managing questions, managing exams, and managing users. Each of these main use cases is connected to the read, add, edit, delete operations through `<<extend>>` relationships, indicating optional or extended actions. The authentication process is included as a shared prerequisite for executing all main tasks, represented using the `<<include>>` relationship. This structure highlights how access control and CRUD (Create, Read, Update, Delete) operations are integrated within each management process while maintaining modularity through reusable components.

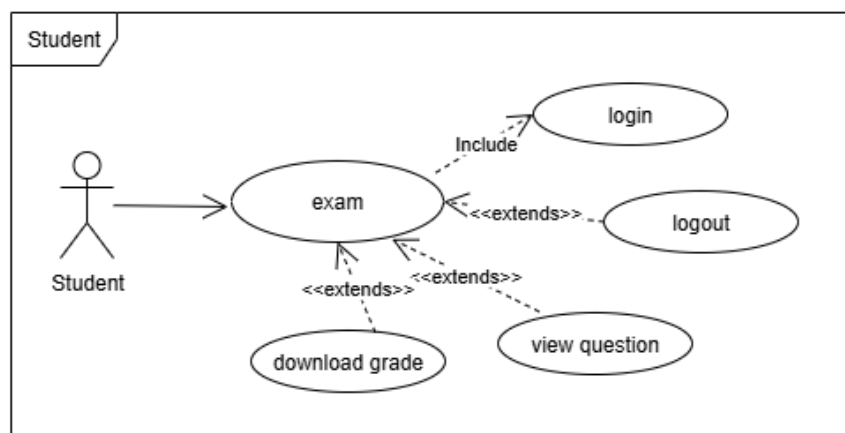


Figure 5. Use Case Diagram Student Actor

This use case diagram illustrates the interaction between the Student actor and the exam system. The student accesses the main use case exam, which is extended by optional features view question and download grade via `<<extend>>` relationships, indicating these are conditional parts of the exam process. Prior to taking the exam, the student must complete the login process, represented by an `<<include>>` relationship to enforce authentication as a required step. After the exam, the student may perform a logout, shown as a separate and independent use case.

c. Activity Diagram

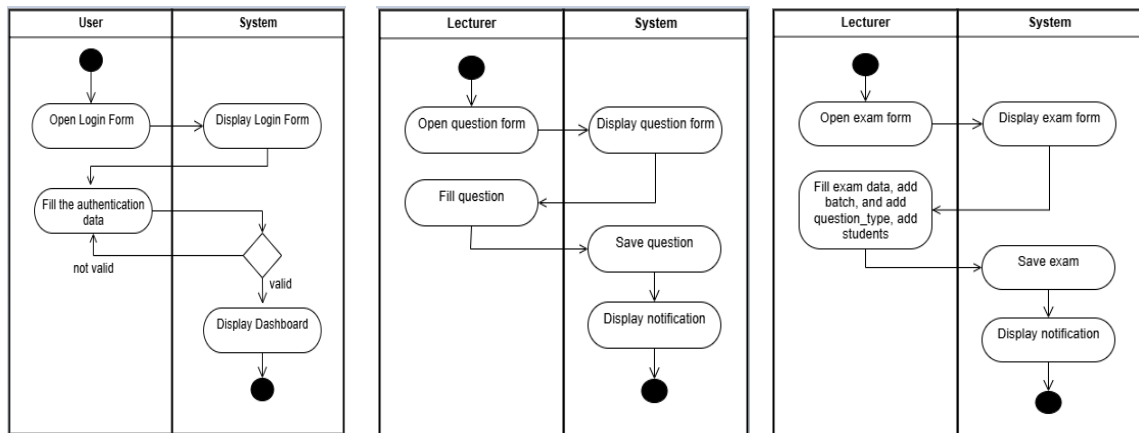


Figure 6. Activity Diagram Login, Add Question, Add Exam

These three activity diagrams illustrate the main workflows in a web-based assessment system. The first diagram shows the user login process, starting from opening the login form, entering authentication data, and if valid, proceeding to the dashboard. The second diagram describes how a lecturer adds questions by opening the question form, filling in the content, saving it, and receiving a confirmation notification. The third diagram outlines the process of creating an exam, where the lecturer inputs exam data, adds a batch, questions, question types, and students, then saves the data and receives a success notification.

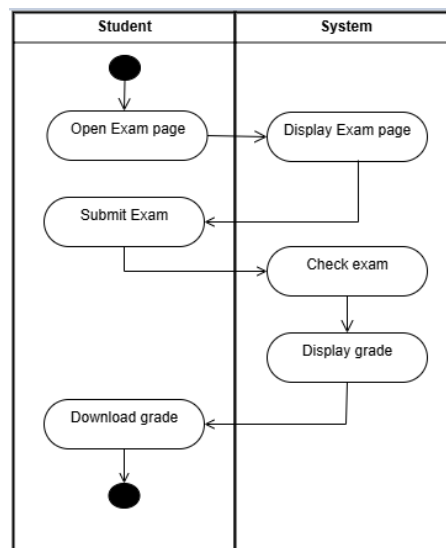


Figure 7. Activity Diagram Exam process

This activity diagram illustrates the process flow of a lecturer creating an exam within the system. The process begins when the lecturer opens the exam form, after which the

system displays it. The lecturer then fills in the exam data, adds a batch, question types, and students. Once all data is complete, the lecturer saves the exam, and the system displays a notification confirming that the exam has been successfully saved. The diagram highlights the interactive steps between the user and the system in organizing the exam in a structured manner.

d. Website Design

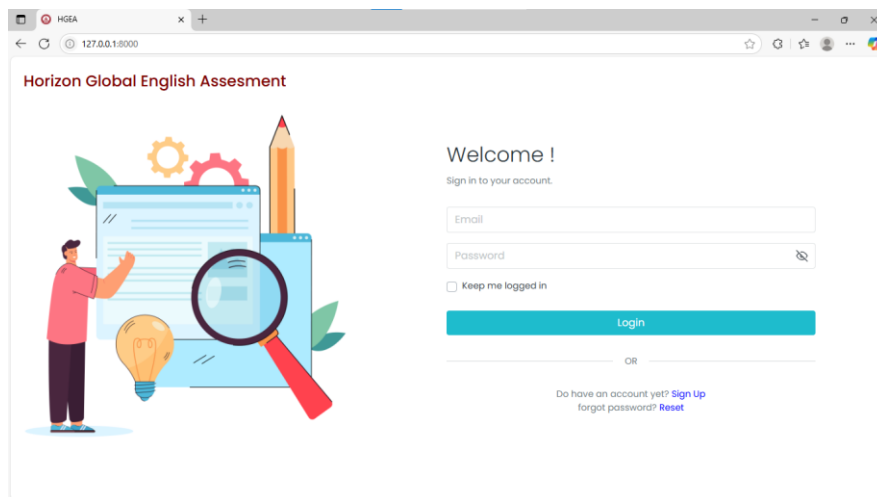


Figure 8. Login Form Design

The login form interface of the English Assessment Application features a clean, modern, and user-friendly design. On the left side, a visual illustration represents the learning and assessment process, showing a magnifying glass, a light bulb, and digital elements. On the right, the login form includes input fields for Email and Password, along with a Keep me logged in checkbox. The Login button is styled in a bright teal color to stand out clearly. A Sign Up link is also provided at the bottom for new users who do not have an account.

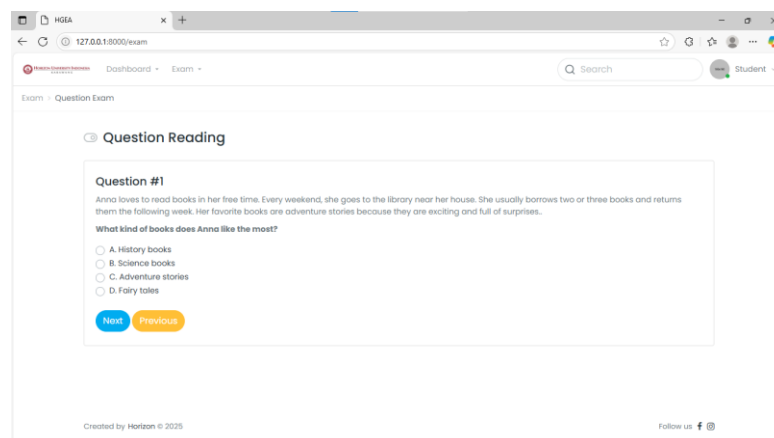


Figure 9. Question Page Design

The exam form interface is designed to clearly present questions and multiple-choice options in a structured layout. At the top of the screen, the question number and text are displayed in an easy-to-read format. Below the question, four answer choices (options A to D) are shown, each with a corresponding radio button for the user to select their answer. This design ensures that all key elements—question, choices, and navigation—are easy to access and understand. The interface typically includes Next, Previous, and Submit buttons to navigate between questions and submit answers at the end of the exam session. The overall layout is kept minimal to help students stay focused during the test.

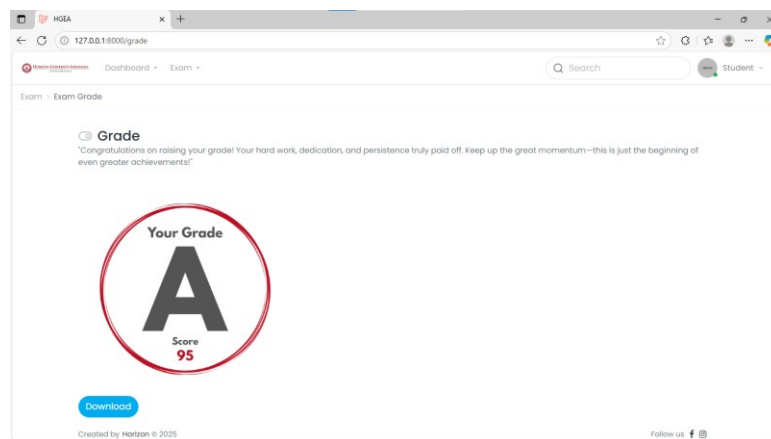


Figure 10. Grading Page Design

The exam grading interface is designed to display the final results of the test completed by the student. On the screen, students can view a visual representation of their score in the form of a letter grade (such as A, B, or C), presented clearly and attractively—often accompanied by icons or colors that reflect their achievement. Below that, the numerical score is shown, based on the number of correct answers from all the questions completed. The design may also include additional information, such as a download button to allow students to download their grade transcript. This interface aims to provide quick, visual, and easily understandable feedback for students.

3. Application Development

The application was developed using the Laravel 12 framework, running on PHP version 8.2. Laravel is a modern and widely adopted PHP framework that supports clean architecture, built-in security features, and efficient development through features such as routing, middleware, authentication, and Eloquent ORM for database handling.

Web Development Stack:

- a. Backend Framework : Laravel 12 (PHP)
- b. Programming Languages : PHP 8.2, HTML5, CSS3, JavaScript

- c. Frontend Tools : Laravel Blade Templating Engine
- d. Database Management : MySQL or MariaDB
- e. Version Control System : Git
- f. Environment Configuration : Managed using Laravel's .env file system

4. System Testing

Unit Testing

a. AuthenticationController

Table 2. AuthenticationController Testing

Test Case	Description	Expected Result
Test valid login	Provide correct email and password	Redirect to dashboard
Test invalid login	Provide incorrect password	Show error message
Test middleware	Access dashboard without login	Redirect to login

b. ExamController

Table 3. ExamController Testing

Test Case	Description	Expected Result
Store Exam	Pass valid exam form data	Save exam and return success
Missing Field	Leave required field empty	Return validation error
Edit Exam	Load existing exam	Data prefilled in form
Delete Exam	Send delete request	Exam deleted and redirected

c. QuestionController

Table 4. QuestionController Testing

Test Case	Description	Expected Result
Add Question	Submit valid question data	Question stored in database
Invalid Input	Submit with missing or invalid options	Validation error shown
Retrieve Questions	Call index()	Return list of questions

5. Implementation and Deployment

To ensure stable, secure, and high-performance deployment of the web-based English Proficiency Assessment System, the application is hosted on a Linux-based server environment. The following are the recommended technical specifications for the production server:

Server Configuration:

- a. Operating System: Ubuntu Server 22.04 LTS
- b. Web Server: Apache
- c. Database: MySQL 8.0
- d. Package Manager: Composer
- e. Memory (RAM): Minimum 4 GB (8 GB recommended for production stability)
- f. Processor (CPU): At least 2-Core (4-Core recommended)
- g. Storage: SSD with at least 40 GB of free space
- h. SSL Certificate: Let's Encrypt (for secure HTTPS access)
- i. Firewall & Security: UFW (Uncomplicated Firewall), Fail2Ban (optional), and OpenSSL for encryption.

CONCLUSION AND RECOMMENDATIONS**Conclusion**

The web-based English assessment system supports adaptive evaluation tailored to students' learning needs, enhancing engagement and motivation. It allows for the design of tests that accurately reflect practical English skills and align with learning areas. Furthermore, the system provides effective assessment tools with immediate feedback, contributing positively to student motivation and the overall quality of English language education.

Recommendations**Integration with Learning Management Systems (LMS)**

To improve efficiency and scalability, the system can be further developed by integrating it with Learning Management Systems (LMS) such as Moodle or Google Classroom. This would allow assessment results to be directly linked to overall teaching and learning activities within the educational institution.

Continuous Evaluation

It is recommended that regular evaluations be conducted on system performance, user experience, and its alignment with current educational needs to ensure that the platform remains relevant, effective, and responsive to future developments.

BIBLIOGRAPHY

- Boehm, B. W. (1986). A spiral model of software development and enhancement. *ACM SIGSOFT Software Engineering Notes*, 11(4), 14–24. <https://doi.org/10.1145/12944.12948>.
- Brown, H. D. (2004). *Language assessment: Principles and classroom practices*. Pearson Longman.
- Degeng, P., Wahyuni, S., & Septiarananda, L. (2023). Development of a Web-Based Writing Assessment System to Support Independent Learning. *Englisia: Journal of Language, Education, and Humanities*, 11(1), 1–15. <https://www.jurnal.ar-raniry.ac.id/index.php/englisia/article/view/20386>.
- Emmiyati, N., Ariani, S., & Andayani, F. (2023). Development of an Online English Proficiency Test and Its Application in Language Learning Contexts. *Indonesian Journal of English Education*, 10(1), 1–15. <https://journal.uinjkt.ac.id/index.php/ijee/article/view/37276>.
- Fitriyah, A. A., & Widiati, U. (2021). Classroom-Based Language Assessment: The Perceptions of Indonesian EFL Teachers. *CELT: A Journal of Culture, English Language Teaching & Literature*, 21(1), 55–72. <https://journal.unika.ac.id/index.php/celt/article/view/3055>.
- Pressman, R. S. (2014). *Software engineering: A practitioner's approach* (8th ed.). McGraw-Hill Education.
- Sabri Sahrir, M., Husnin, H., & Alias, A. (2012). Employing design and development research (DDR) approaches in the design and development of online Arabic vocabulary learning games prototype. *Procedia - Social and Behavioral Sciences*, 67, 537–546. <https://doi.org/10.1016/j.sbspro.2012.11.361>.
- Septiarananda, L., Wahyuni, S., & Puspitasari, R. (2022). Students' Perceptions of Using Web-Based Tests in Learning English. *TELL Journal*, 10(2), 123–135. <https://ejurnal.unima.ac.id/index.php/tell/article/view/7762>.
- Sidiki, A. N. N., Goni, A., Blessar, A. R., Amanda, A. P., Aziz, M. R., Airlangga, M. V., & Zuraidah, E. (2023). Design of web-based inventory information system using the spiral method. *IN: Terapan Informatika Nusantara*, 4(6), Article e4587. <https://doi.org/10.47065/tin.v4i6.4587>.
- Sommerville, I. (2011). *Software engineering* (9th ed.). Pearson Education.