VIETNAM NATIONAL UNIVERSITY,
HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY (HCMUT)
FACULTY OF COMPUTER SCIENCE & ENGINEERING

SOFTWARE ENGINEERING

Assignment

Verbal Mate

Lecturer: Prof. Trương Tuấn Anh
SV: Lê Hoàng Thịnh - 2252775
Phạm Thành Nam - 2153603
Hồ Gia Tường - 2252887
Lương Quang Khánh - 2053115
Nguyễn Minh Tiến - 2150033
Đoàn Tấn Sang - 2252711

HO CHI MINH CITY, December 8^{th}, 2025

Contents

Revision History

Project Timeline:

Date	Content	Name	Version
19/10/2024	First Meeting	Group Meeting	1.0
27/10/2024	Main topic of project, requirement elicitation, domain context	Group Meeting	1.0
1/11/2024	Use-case table, stakeholders and needs	Group Meeting	1.0
9/11/2024	Whole Diagram	Group Meeting	1.0
15/11/2024	Functional & Non-functional Requirements	Group Meeting	1.1
17/11/2024	Activity Diagram, Sequence Diagram, Class Diagram, MVP	Group Meeting	1.1
1/12/2024	Architectural Diagram & Technology Consideration	Group Meeting	1.2
16/12/2024	Standardize writing & Proofing	Group Meeting	1.3
28/12/2024	Component Diagram & Testing	Group Meeting	1.4
30/12/2024	Writing in LaTeX	Group Meeting	1.4
8/1/2025	Finalize the project and record the report video	Group Meeting	1.4

Task 1: Requirement elicitation

1.1 Domain Context

1.1.1 Domain Context

The project aims to assist learners in improving their spoken English by practicing with structured speaking exercises. Learners answer preset questions using given cue words, and their responses are transcribed for evaluation. The system checks if learners correctly use the provided cue words, offering feedback to help them refine their language skills. The platform will be managed by a teacher, who oversees learner progress, question creation, and cue word management.

1.1.2 Stakeholders and Needs

Learners:

- Need to practice speaking English through structured exercises.
- Need immediate feedback on whether they used the correct cue words.
- Need access to varied questions that cover different topics to improve vocabulary and sentence structure.
- Need a user-friendly interface to submit spoken answers.

Teacher:

- Needs to manage the system and user database, including adding or removing learners.
- Needs to create and manage a library of preset questions and cue words.
- Needs to monitor learners' progress and offer feedback when necessary.
- Needs the ability to view transcribed learner responses to ensure cue word usage.
- Needs reports and analytics on overall learner performance.

1.1.3 Benefits of the System

Personalized Learning: Learners receive tailored feedback based on their responses, helping them improve their speaking abilities.

Efficient Practice: The system provides an automated way for learners to practice at their own pace, without needing live instructors.

Progress Monitoring: Both learners and teachers can track progress over time, identifying areas for improvement.

Scalability: The system can handle multiple learners simultaneously, making it suitable for both individual and institutional use.

Consistency: The use of automated transcription ensures consistent evaluation of learners' responses.

1.2 Functional & Non-Functional Requirements

1.2.1 Functional Requirements

For Learners:

1. The system must allow learners to answer preset questions by recording spoken responses.
2. The system must provide cue words for each question to guide learners' responses.
3. The system must transcribe learners' spoken answers into text automatically.
4. The system must check whether the transcribed responses contain the given cue words.
5. The system must provide immediate feedback to learners on whether they used the correct cue words.

For Teacher:

1. The system must allow teachers to add, modify, or remove learners' accounts.
2. The system must allow teachers to create and edit a library of preset questions and cue words.
3. The system must generate reports on learners' progress and performance over time.
4. The system must allow teachers to manage system settings, such as enabling or disabling feedback features for learners.

1.2.2 Non-Functional Requirements

1. **Usability:** The system must have an intuitive and user-friendly interface for both learners and teachers, ensuring easy navigation and use.
2. **Performance:** The system must transcribe responses and provide feedback in less than 5 seconds for a seamless user experience.
3. **Scalability:** The system must support up to 10,000 concurrent learners without degradation in performance.
4. **Security:** The system must secure learners' personal data and transcriptions by implementing encryption and following best practices for data protection.
5. **Reliability:** The system must have 99.9% uptime to ensure continuous availability for learners to practice at any time.

1.3 Use-case Diagrams

1.3.1 Use-case Diagram for the Whole System

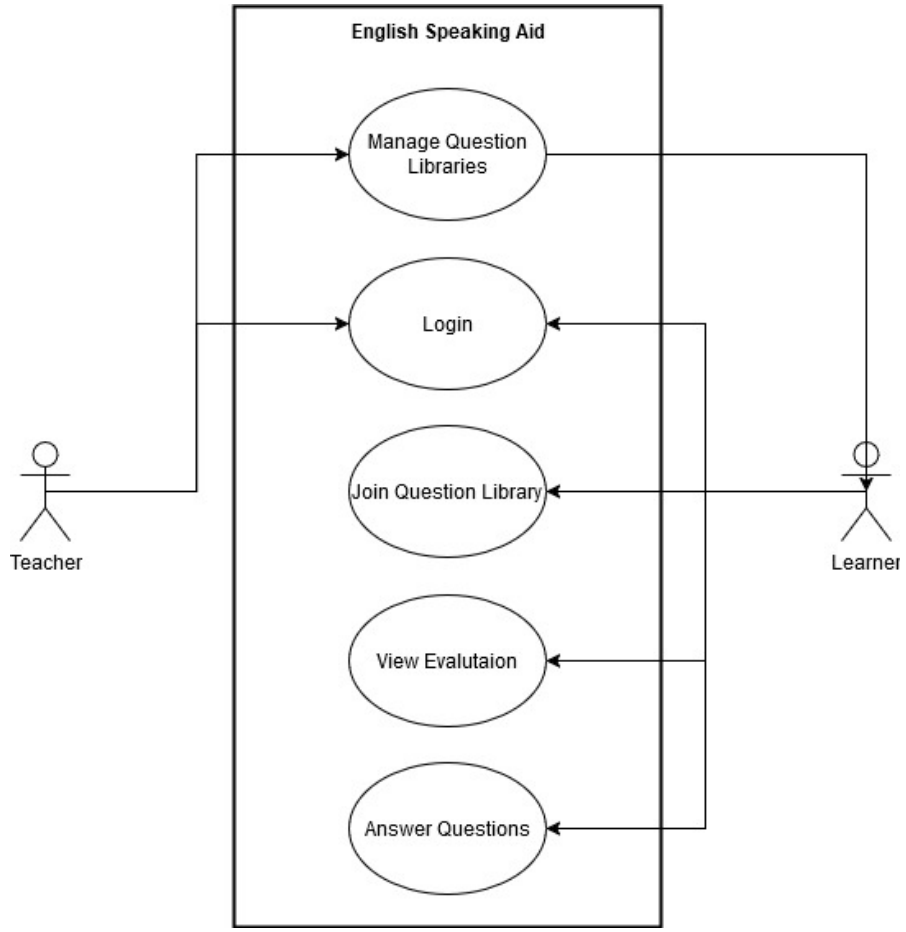


Figure 1: Use-case Diagram for the Whole System

1.3.2 Use-case Diagram for Manage Question Libraries Module

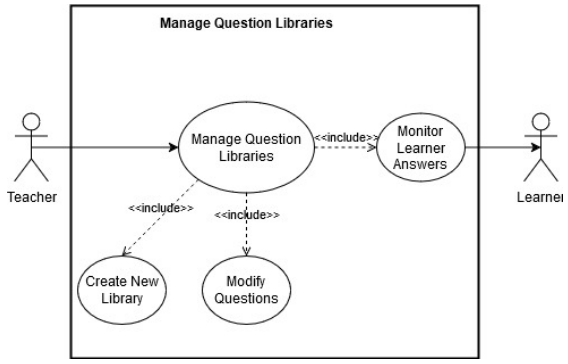


Figure 2: Use-case Diagram for Manage Question Libraries Module

1.4 The Details of Usecases in Manage Question Libraries Module

1.4.1 Usecase Create New Library

Name	Create New Library
Created by	Gia Tường
Date Created	29/10/2024
Primary actor	Teacher
Secondary actor	N/A
Description	The teacher creates a new library of questions from scratch or from an existing library.
Trigger	The teacher chooses the Create New Library function of the app on their device.
Precondition	PRE-1. The user's identity has been authenticated.
Postcondition	POST-1. A new question library is created and is accessible to learners and the teacher.
Normal Flow	1. The teacher chooses between creating a blank library or cloning an existing question library. 2. The teacher chooses a name for the question library. 3. The system creates the question library. 4. The system assigns the teacher as the creator of the question library, having teacher privilege.
Alternative Flow	2.1. Invalid Name: Notify the teacher and suggest choosing another name. Return to step 2.
Exception	Network Failure: Notify the teacher and suggest documenting changes before refreshing the page or app.

1.4.2 Usecase Modify Questions

Name	Modify Questions
Created by	Gia Tường
Date Created	29/10/2024
Primary actor	Teacher
Secondary actor	N/A
Description	The teacher modifies the question library by adding, removing, or changing questions and cue words, or changing the question library's name.
Trigger	The teacher chooses the Modify Questions function of the app on their device.
Precondition	PRE-1. The user's identity has been authenticated. PRE-2. There are existing question libraries which the user can access with teacher privilege.
Postcondition	POST-1. The chosen question library is updated, as specified by the teacher.
Normal Flow	1. The teacher chooses an existing question library to modify. 2. The teacher chooses an existing question to modify, or creates a new question. 3. The teacher modifies the question, its answer, and its cue words, as well as the feedback for specific incorrect answer(s). 4. Return to step 2. If the teacher is done with modifications, continue to step 5. 5. The system saves the changes made to the questions.
Alternative Flow	3.1. Teacher deletes the question: Return to step 2. 3.2. Invalid data: Notify the teacher and suggest changing the question. Retry step 3.
Exception	Network Failure: Notify the teacher and suggest documenting changes before refreshing the page or app.

1.4.3 Usecase Monitor Learner Progress

Name	Monitor Learner Progress
Created by	Gia Tường
Date Created	29/10/2024
Primary actor	Teacher
Secondary actor	Learner
Description	The teacher views learners' progress and reviews the mistakes made in their submitted responses.
Trigger	The teacher chooses the monitor learners progress function of the app on their device.
Precondition	PRE-1. The user's identity has been authenticated. PRE-2. The user has teacher privilege for the question library.
Postcondition	POST-1. Learner's progress record is updated with teacher's feedback, if applicable.
Normal Flow	1. The system will display the answers of learners participating in the question library. 2. The teacher selects an answer to examine. 3. The system displays a transcript of the learner's response, their identity, and the cue words used. 4. Optionally, the teacher provides additional feedback.
Alternative Flow	N/A
Exception	Network Failure: Notify the teacher and suggest documenting changes before refreshing the page or app.

Task 2: System modeling

2.1 Activity Diagrams

2.1.1 Usecase for the learner

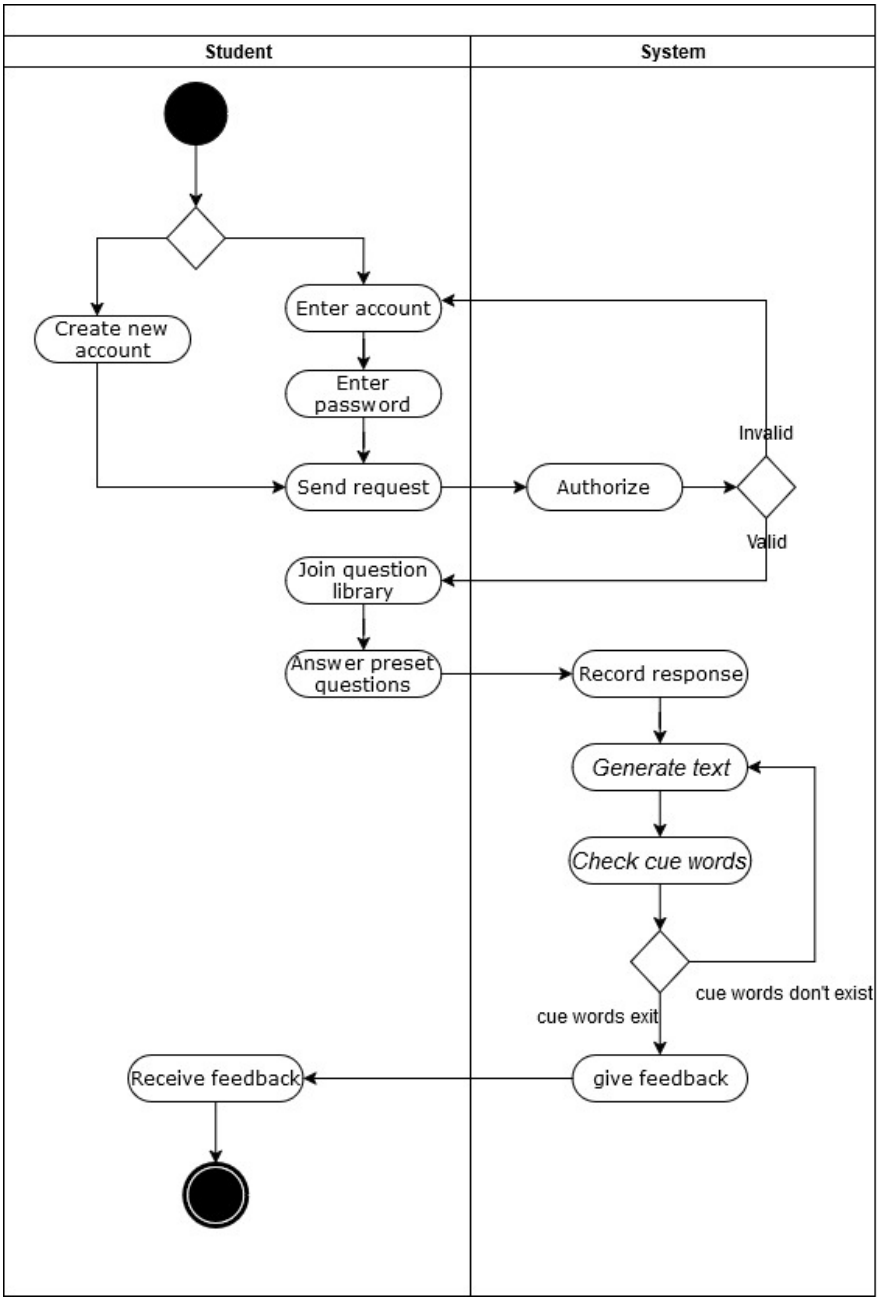


Figure 3: Activity diagram for Student

In this activity diagram, students will first either create their new account or enter their available account to login the website. When submitting the request, the system will authorize their request, if it is valid, they will continue to move to the website, if not, they get back to the starting point. Once they get in, we can look for the question library they want to join in. In that library, there will be preset questions for them to answer. Their answers will be recorded and generated to text. System will then check cue words in the test, if there is no cue word, system will regenerate the text. Finally, it will give the feedback to the student.

2.1.2 Usecase for the teacher

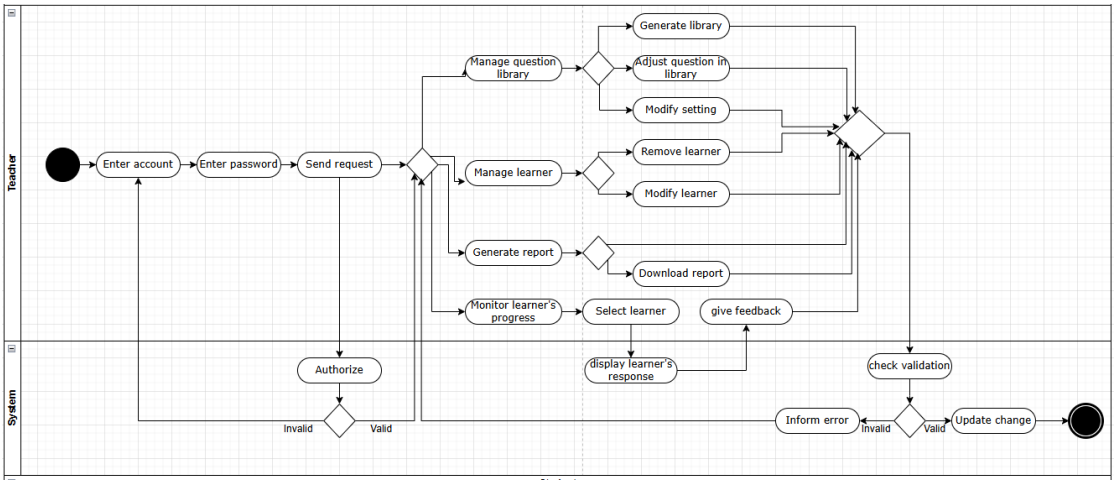


Figure 4: Activity diagram for Teacher

In this activity diagram, teachers will first login their account in order to enter the website. When submitting the request, the system will authorize their request, if it is valid, they will continue to move to the website, if not, they get back to the starting point. Once they get in, they can choose to manage their question library, learner, generate report or monitor their learner progress. When managing question library, they can generate library, adjust question in the library, modify setting of the library. When managing learner, they can remove or modify learner. When generate report, they can download back their report if they want to or just have it in the system. When monitoring learner's progress, they can select learner's response they want to display in their screen, which they can after giving feedback. All action will be checked by the system to see if these action is valid or not to inform the error of their action and have them back previous action or make update and complete the action

2.2 Sequence Diagrams

This sequence diagram illustrates the flow of actions between a User, a System, and a Database. The User begins the interaction by sending a request to the System. This is represented by the first solid arrow pointing from the User lifeline to the System lifeline. The system then becomes active, as shown by the vertical activation bar. The System, after receiving the user's request, sends a message to the Database. This action is represented by another solid arrow from the System lifeline to the Database lifeline. The Database activation bar indicates that it is processing the request. Once the Database processes the request, it returns a response

to the System. This is depicted by a dashed arrow going back from the Database to the System. The System, upon receiving the response from the Database, continues processing and prepares a result for the User. Finally, the System sends the response back to the User. This is shown by a dashed arrow from the System lifeline to the User lifeline.

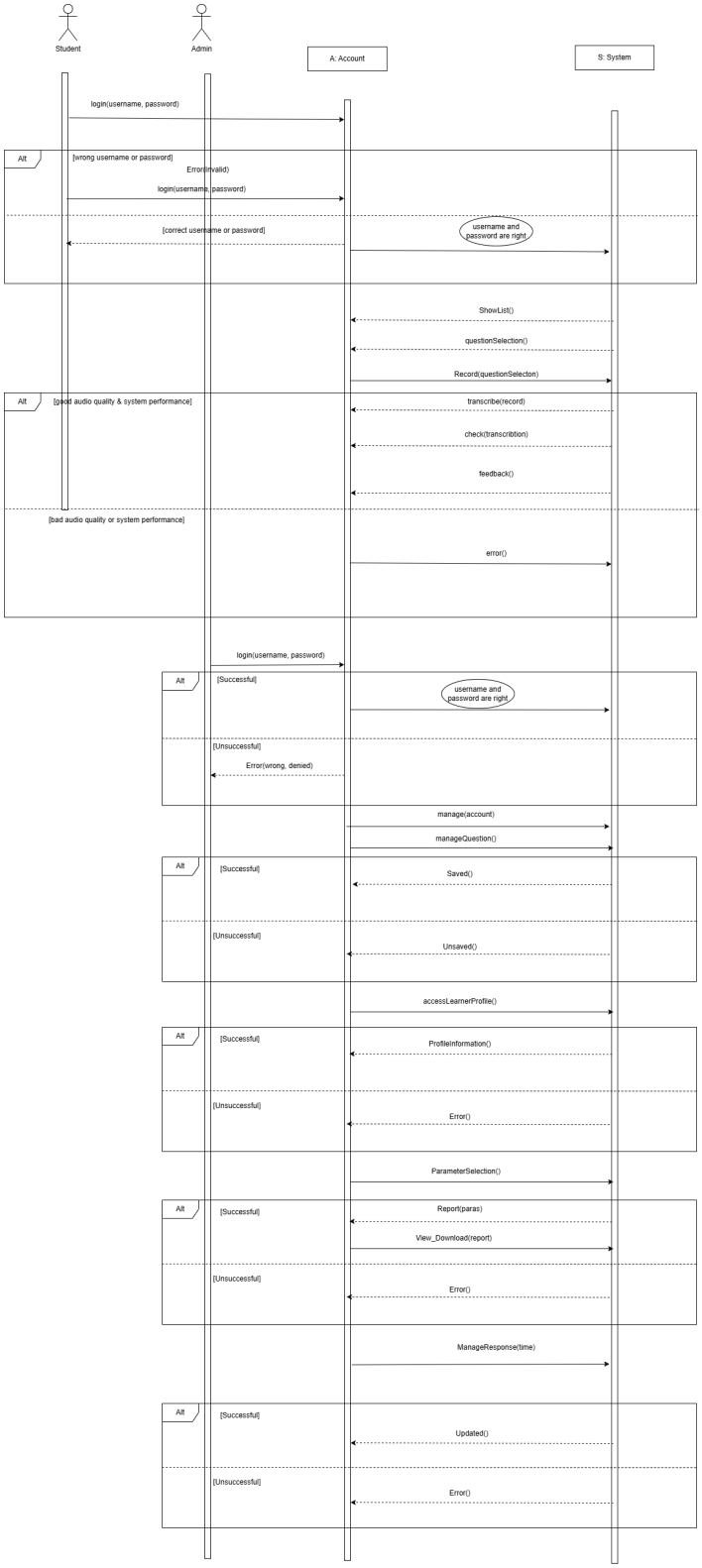


Figure 5: Sequence diagram of the system

2.3 Class Diagrams

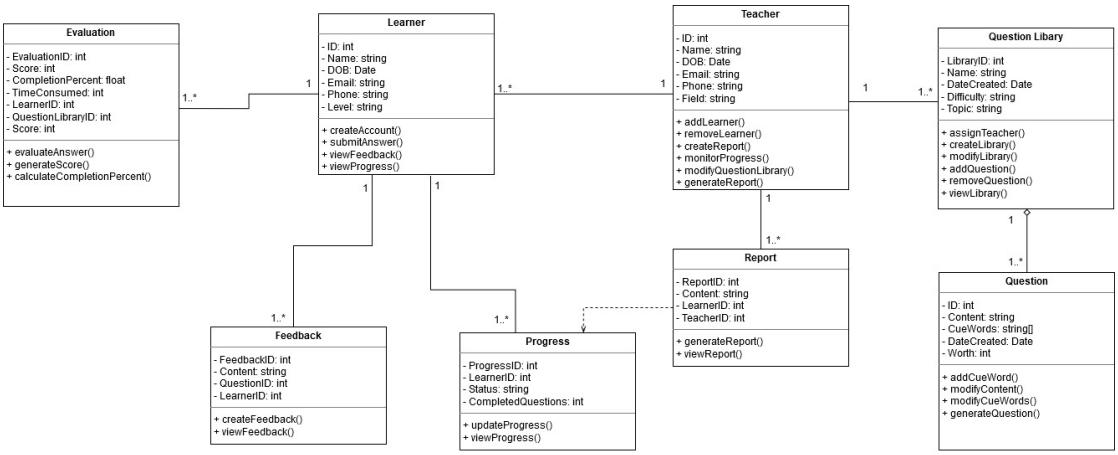


Figure 6: Class diagram of of the system

2.4 MVP Development

The MVP (Minimum Viable Product) is developed using Laravel as the backend framework, focusing on a minimalist interface design that prioritizes simplicity, usability, and efficiency. The interface employs a modern aesthetic, characterized by rounded edges, clean lines, and a balanced layout. As for the color scheme, the pages are primarily white with black buttons, providing a clean and uncluttered look while also offering strong visibility and contrast. A tool bar on top provides quick access to the homepage, quickly changing account roles, and user account management.

2.4.1 Student's Homepage

The Dashboard of Verbal Mate serves as the homepage, featuring an intuitive and functional layout, ensuring easy navigation and access to essential features. Here, the student can access question libraries to begin engaging with them. Each library is addressed by their name and number of questions and separated into organized blocks.

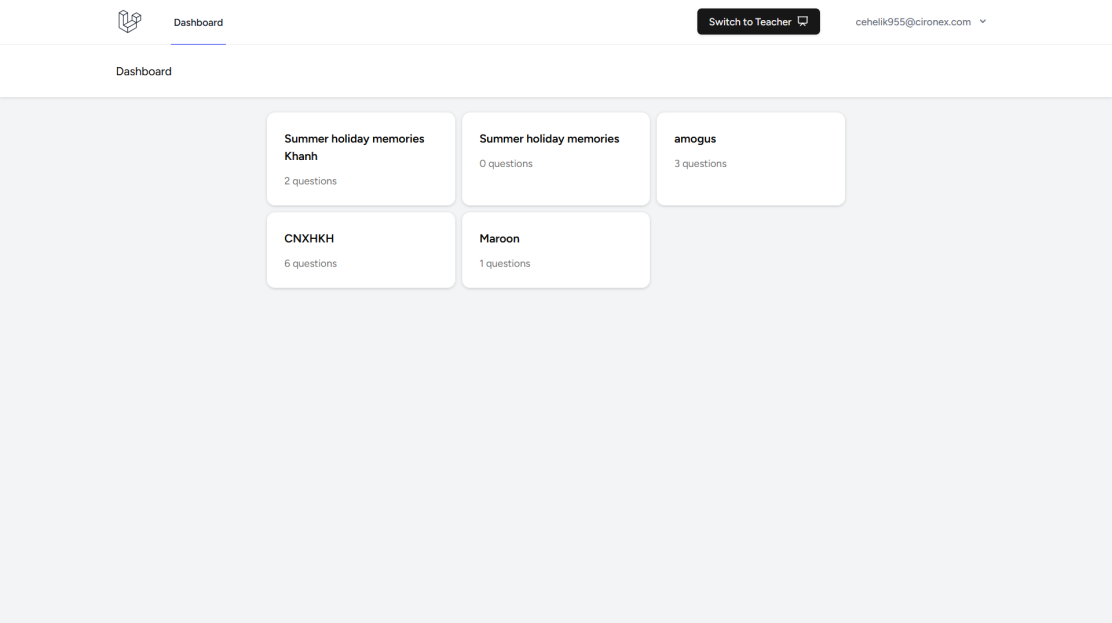


Figure 7: The student's homepage of Verbal Mate

2.4.2 Login/Register page

This screen allows users to securely access the system. It features fields for entering a username and password, a "Login" button, and a "Remember me" checkbox for saving your session. Alternatively, the user can select "Don't have an account?" to register an account. The screen provides instructions and error messages should errors occur, such as invalid credentials.

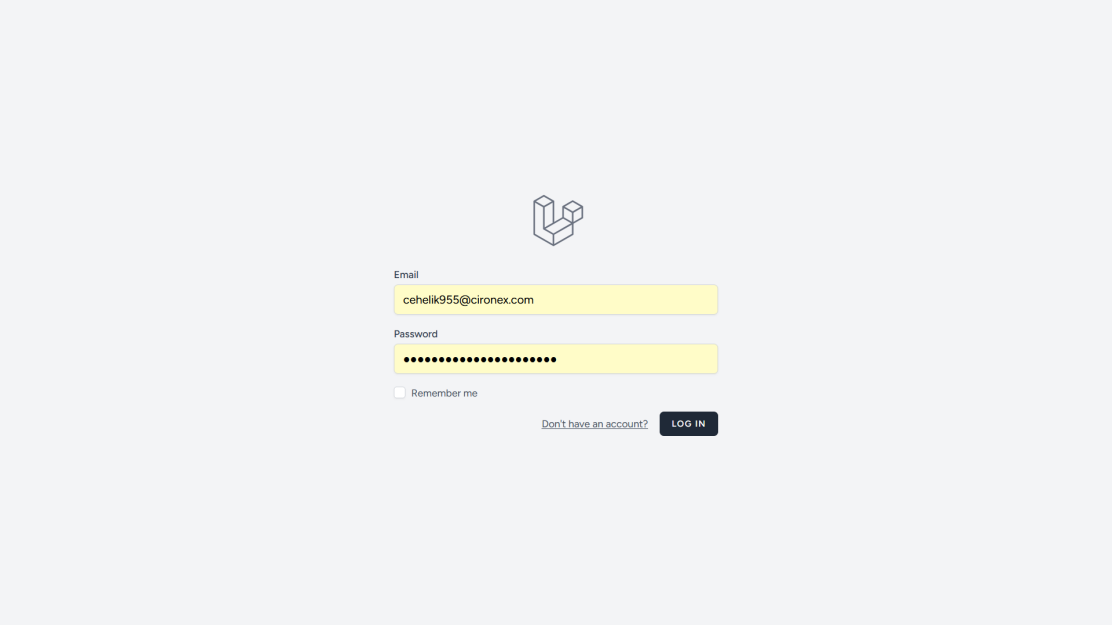


Figure 8: The Login page of Verbal Mate

2.4.3 Profile page

Here, the user can manage and modify their credentials, such as name, email, and password. The user is also provided the option to delete their account, should a need occur.

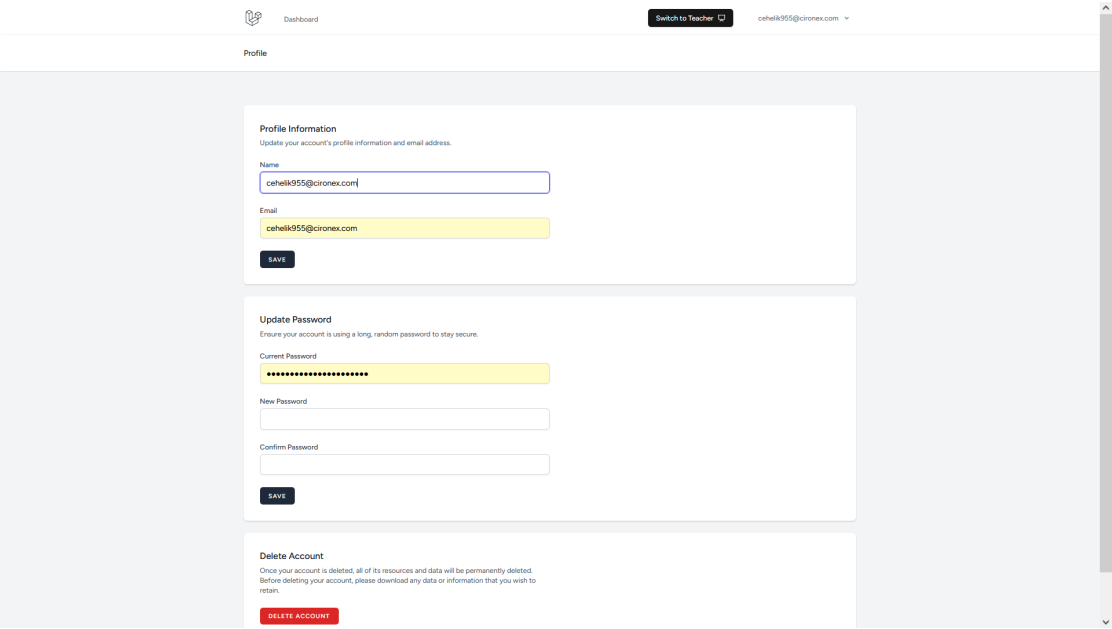


Figure 9: The profile page of Verbal Mate

2.4.4 Inside a Question Library

Accessing a question library, the student is provided an overview of the comprising questions and their cue words. By selecting "Attempt" next to the question, the student will transfer to another page to answer the respective question.

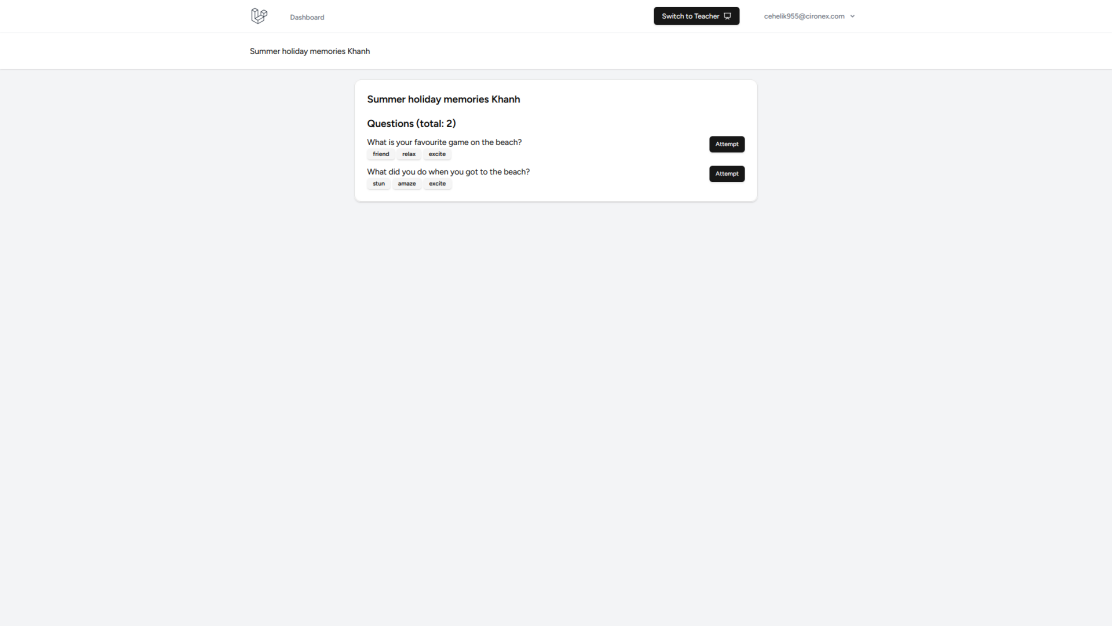


Figure 10: The list of Questions in a Library

2.4.5 Answering a Question

The interface is structured to encourage focus on the task, while providing necessary options to navigate, track progress, upload responses, and review previous answers. The question and cue words are featured prominently alongside a record button for user accessibility. This page also contains several utilities to ensure a seamless experience, such as replay recording, an option for file upload instead of live recording, and a summary of previous attempts.

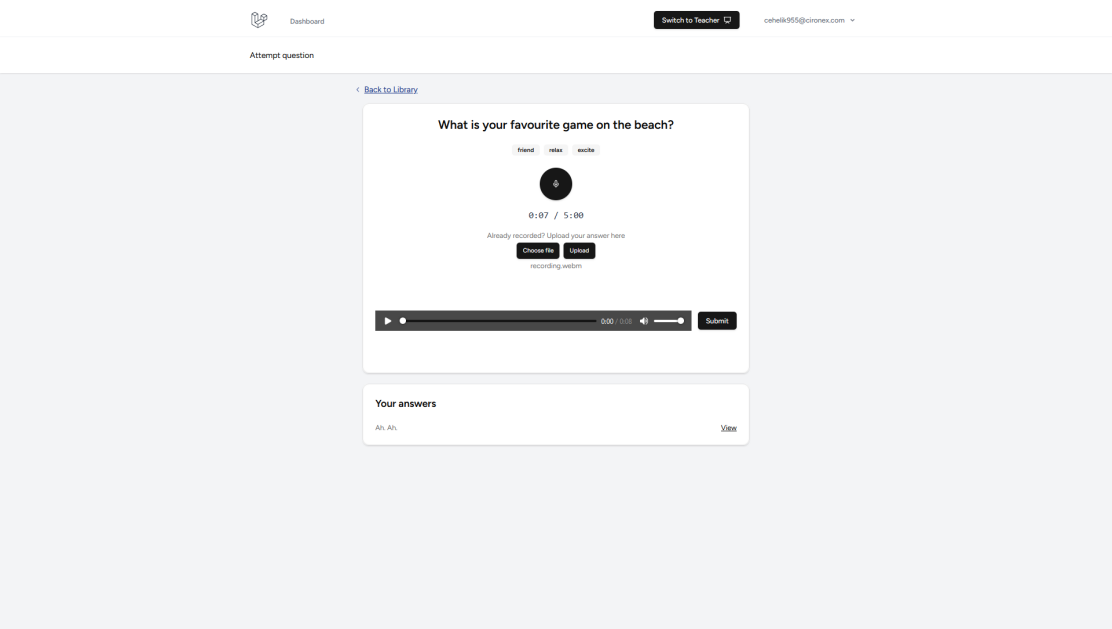


Figure 11: Answering a Question

After submission, the page shows a transcript of the student's recorded or uploaded response, any cue words that were mentioned in the transcript are automatically highlighted in green, helping students see how they have incorporated the provided hints into their answer.

The cue words displayed below the question undergo a visual change. Each cue word that was used in the student's response turns green and a small checkmark appears next to them, providing additional visual feedback that the student has successfully integrated the guidance into their response.

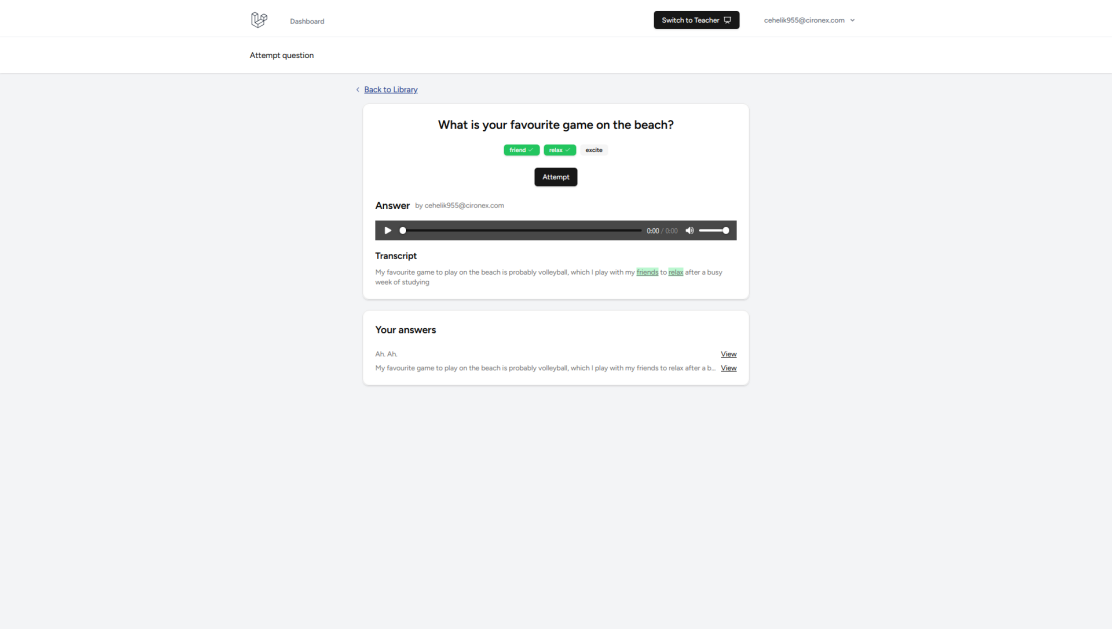


Figure 12: Feedback for answer

2.4.6 Teacher's Homepage

The interface for the teacher is nearly identical to that of the student, with the exception of the option to access the question library, providing familiarity to the user and ensuring a consistent theme.

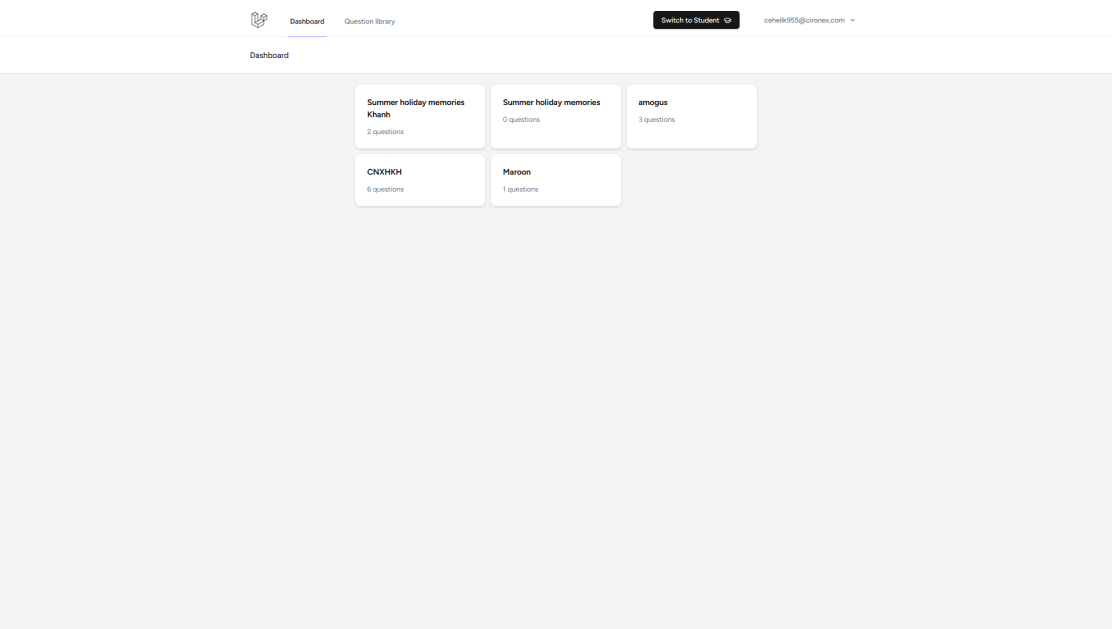


Figure 13: The teacher's homepage of Verbal Mate

2.4.7 Question Library page

For the same reason, the question library page is also similar to the dashboard. Here the teacher can access their own question libraries to review and modify their questions, create a new library using the "Create" button at the bottom right corner, or check their students' answers.

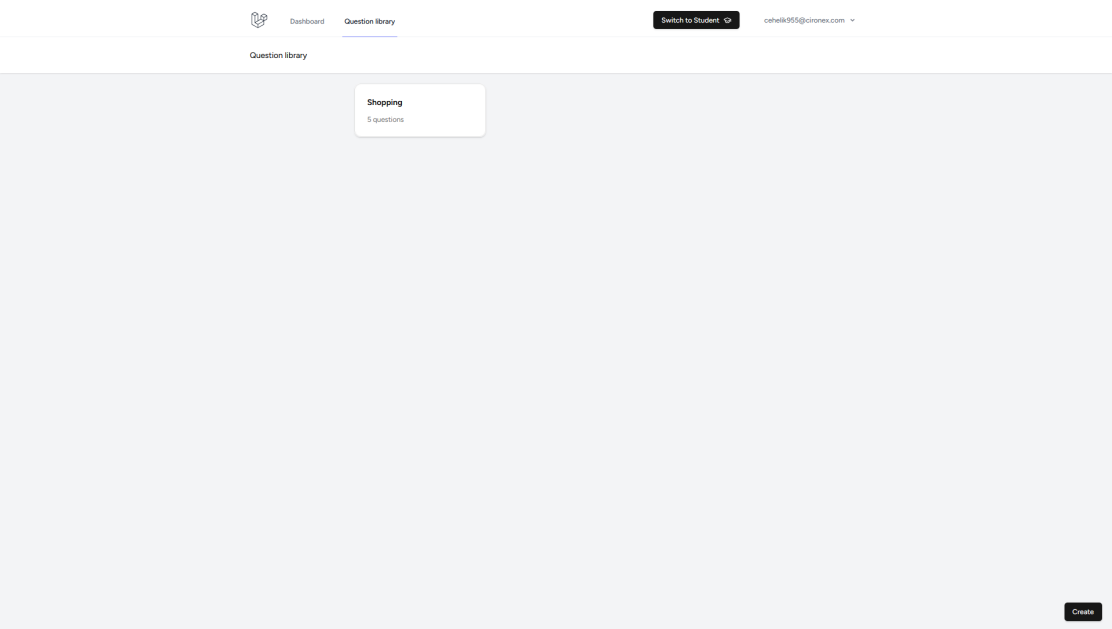


Figure 14: The Question Library page

By clicking the "Edit" button next to a question, the teacher can modify the question and their cue words. This action is done in a textbox and the background is dimmed to minimize visual noise and make using the page more intuitive.

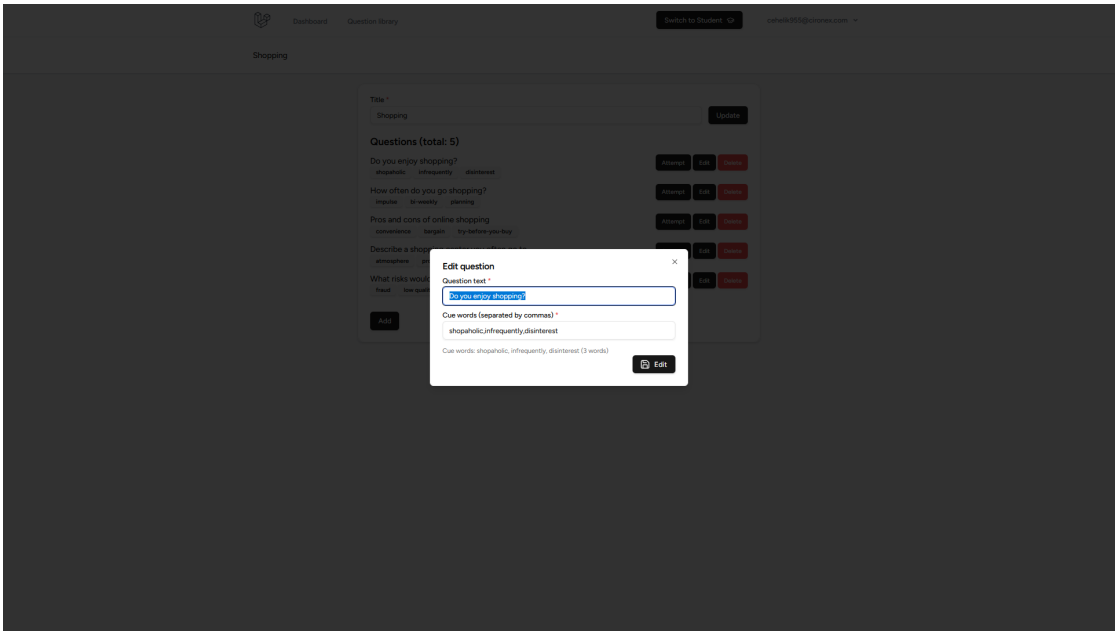


Figure 15: Editing a question

Task 3: Architecture design

3.1 Architecture

3.1.1 Architectural diagram

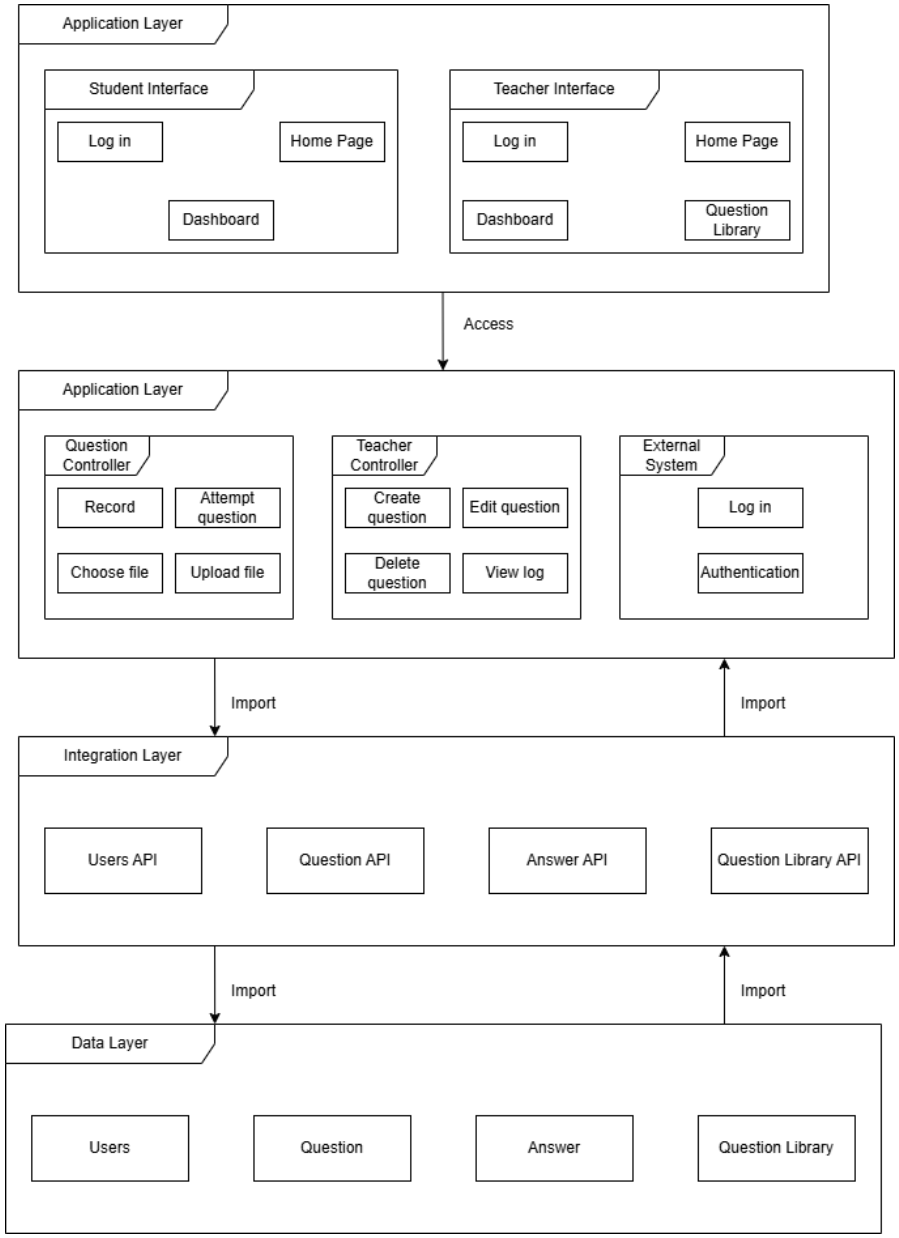


Figure 16: The Layer Architecture diagram

3.1.2 Presentation strategy

Student interface:

- Login
- Dashboard
- Attempt + Record function

Teacher interface:

- Login
- Dashboard
- Question library
- Attempt + Record function

The Presentation Strategy for the system is designed to cater to both students and teachers, ensuring an intuitive and effective interface for each user group. For students, the interface begins with a secure login mechanism that personalizes their experience. Upon logging in, students access a centralized dashboard that provides a comprehensive overview of their progress, assignments, and upcoming activities. An essential feature of the student interface is the "Attempt + Record" function, which allows students to engage with tasks such as quizzes and automatically saves their performance data. This fosters a structured learning process and helps students track their growth over time.

On the other hand, the teacher interface prioritizes efficiency and content management. Teachers also start with a secure login to maintain the confidentiality of their resources. Their dashboard is designed to provide a clear overview of their activities, including task management and performance monitoring. The interface includes a "Question Library" feature, enabling teachers to create, organize, and access a repository of questions for assignments or tests. Additionally, the "Attempt + Record" function is available for teachers to review and record student attempts or manage task-related data. Together, these features ensure a seamless workflow for educators while enhancing their ability to deliver quality education.

3.1.3 Application layer

1. Question library: is a centralized repository where teachers can store, organize, and manage a wide variety of questions. This feature supports multiple question types, enabling flexibility in creating assignments, quizzes, or tests. Teachers can easily update the library to ensure the content remains relevant and aligned with learning objectives.

2. Users: addresses the two main roles within the system: students and teachers. Each user is assigned specific permissions and access levels tailored to their needs. Teachers manage educational content and monitor student progress, while students focus on engaging with assignments and tracking their performance. This role-based structure ensures secure and efficient use of the platform.

3. Questions: is integral to the system, allowing the creation, categorization, and retrieval of questions. Questions can be linked to specific subjects, topics, or difficulty levels, making it easier for teachers to design targeted assessments and for students to engage in relevant practice.

4. Answers: complements the question functionality by enabling users to input and evaluate responses. For students, this feature records their answers, which are then automatically or manually assessed depending on the question type. For teachers, it provides a mechanism to review, analyze, and provide feedback on student submissions. Together, these components create a cohesive and efficient application layer.

3.1.4 Data Layer

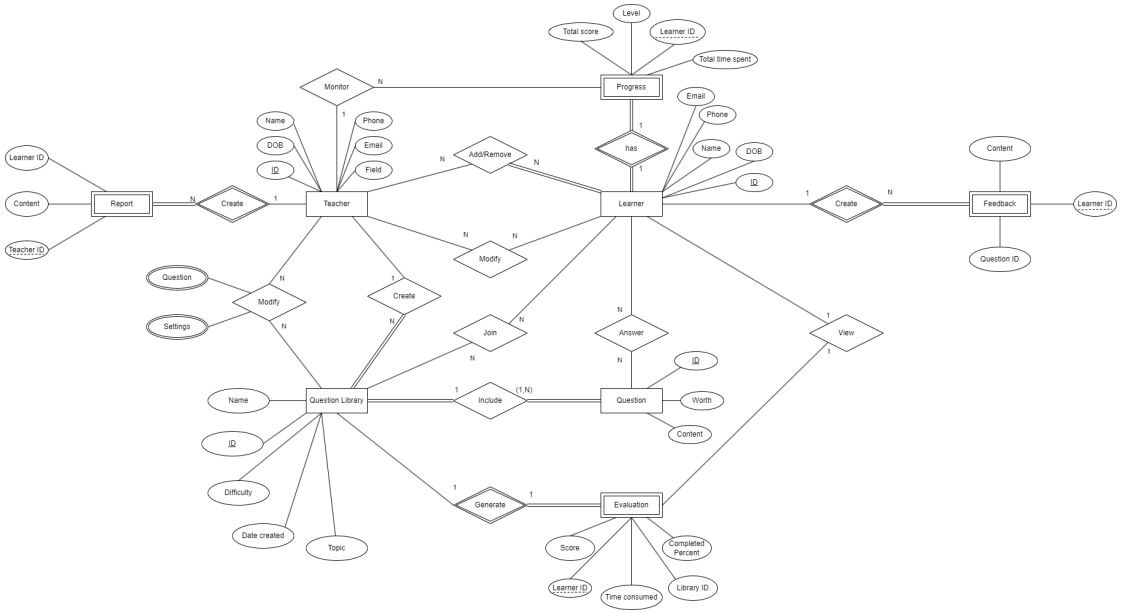


Figure 17: Entity Relation diagram of the database

Users:

Attribute	Data Type	Description
Name	String	Use to identify the student
Email	String	For the contact
Password	String	Authentication
ID	Number	Use to identify the student
Is_teacher	Boolean	To separate teacher and student

Question:

Attribute	Data Type	Description
ID	Number	A unique identifier for each question, used to distinguish it within the system.
Text	String	The main content or body of the question, describing the problem or query.
Cue Words	String	Keywords or hints associated with the question to help categorize or search it.
Question library ID	Number	A reference ID linking the question to its corresponding question library.

Answer:

Attribute	Data Type	Description
ID	Number	A unique identifier for each answer, used to differentiate it within the system.
Question ID	Number	The identifier of the question to which this answer corresponds.
User ID	Number	The identifier of the user (student or teacher) who submitted the answer.
Audio Link	String	A URL or file path linking to the audio recording of the answer, if applicable.

Question Library:

Attribute	Data Type	Description
ID	Number	A unique identifier for each question library, ensuring its distinction.
Answer ID	Number	A reference ID linking the question library to a specific set of answers.
Title	String	The name or title of the question library, describing its content or purpose.

3.1.5 API Management (Integrated Layer):

Users:

Method	Passing Parameter	Description	Return Value
registerUser	username, password	Registers a new user.	success/failure
loginUser	username, password	Authenticates the user and starts a session.	sessionToken
getUserProfile	userID	Retrieves the profile of the given user.	userProfile

Question:

Method	Passing Parameter	Description	Return Value
createQuestion	userID, questionText	Creates a new question.	questionID
getQuestionDetails	questionID	Fetches details of a specific question.	questionDetails
updateQuestion	questionID, newText	Updates the content of an existing question.	success/failure
deleteQuestion	questionID	Deletes the specified question.	success/failure

Answer Component:

Method	Passing Parameter	Description	Return Value
submitAnswer	questionID, userID, answerText	Submits an answer to a question.	answerID
getAnswers	questionID	Retrieves all answers for a question.	answerList
updateAnswer	answerID, newText	Updates the content of an existing answer.	success/failure
deleteAnswer	answerID	Deletes the specified answer.	success/failure

Question Library Component:

Method	Passing Parameter	Description	Return Value
getAllQuestions	category (optional)	Retrieves all questions, optionally filtered by category.	questionList
searchQuestions	keywords	Searches for questions containing keywords.	searchResults
getPopularQuestions	None	Fetches the most popular questions.	popularQuestions

3.2 Component diagram

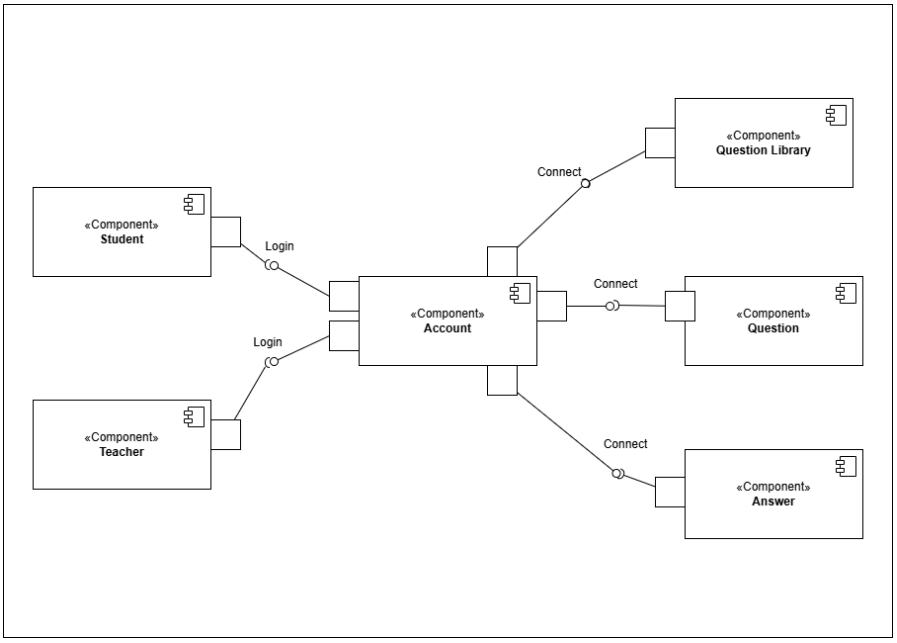


Figure 18: Component diagram of the system

Student Component

The Student component allows students to interact with the system by logging in through their accounts. This component provides students with the ability to access learning resources,

submit questions, and review answers in the system. It serves as a gateway for students to participate in the learning process and collaborate with teachers.

Teacher Component

The Teacher component enables teachers to log in and manage their activities in the system. Teachers can contribute by creating questions, providing answers, and curating content within the Question Library. This component facilitates the teacher's role as a guide and resource provider for students.

Account Component

The Account component acts as the central hub for authentication and authorization. Both students and teachers use this component to log in and access their respective functionalities. It also manages connections to other components such as Question Library, Question, and Answer, ensuring smooth and secure data flow across the system.

Question Library Component

The Question Library component stores and organizes all the questions in the system. It allows users to search, filter, and retrieve questions efficiently. Teachers can manage the library by adding or editing questions, while students can explore the library to find relevant learning materials.

Question Component

The Question component is responsible for handling individual questions created by users. It provides the functionality to create, update, and delete questions. This component acts as the intermediary for storing and displaying questions within the system, enabling interactions between students and teachers.

Answer Component

The Answer component manages the answers submitted in response to questions. It supports the creation, updating, and deletion of answers. This component ensures that answers are stored and displayed in connection with their corresponding questions, fostering a collaborative learning environment.

Task 4: Implementation

4.1 Online repository

The project is maintained and updated utilizing Github for colaborative tasks and version control.

https://github.com/Tuong-hcmut/Programming-Intergration-Project-Gr-8

4.2 Usability Test

4.2.1 Introduction

The usability test aimed to evaluate the user experience and effectiveness of the Minimum Viable Product (MVP) 1 for an English-speaking practice system. The primary focus was on assessing the platform's accessibility and efficiency in supporting learners in improving their spoken English and enabling teachers to manage question libraries effectively. This report documents the test's methodology, execution, and findings, emphasizing user impressions of visual design, ease of use, and core functionality. Key objectives included identifying usability challenges, validating the platform's ability to meet functional requirements, and gathering actionable feedback for refining the system. The results aim to guide improvements that enhance user satisfaction and ensure a user-friendly experience.

Khảo sát chất lượng thử nghiệm Hệ Thống hỗ trợ học Tiếng Anh giao tiếp cho sinh viên

Giới thiệu:

Nhằm đảm bảo chất lượng và phát triển lượng **Hệ Thống hỗ trợ học Tiếng Anh giao tiếp**, nhóm 8 chúng tôi gửi đến bạn một số câu hỏi nhằm khảo sát trải nghiệm của bạn trong suốt quá trình tương tác với hệ thống hệ thống.

Để đảm bảo tính hoàn thiện của cuộc khảo sát, khuyến khích các thí sinh trải nghiệm tất cả các tính năng của hệ thống, bao gồm:
Tasks:

1. Đăng ký tài khoản (Registration)
2. Đăng nhập/ Đăng xuất (Login/ Logout)
3. Thay đổi thông tin người dùng (Modify user info)
4. Tạo một thư viện với ít nhất 5 câu hỏi (Create a library with at least 5 questions)
5. Chỉnh sửa thư viện đã tạo (Modify library)
6. Trả lời câu hỏi bằng cách ghi âm (Answer by recording)
7. Trả lời câu hỏi bằng cách đăng tải file ghi âm từ máy tính (Answer by uploading)
8. Xác nhận và kiểm tra lại câu trả lời (Review previous answers)

Website: https://khanhhne.com/

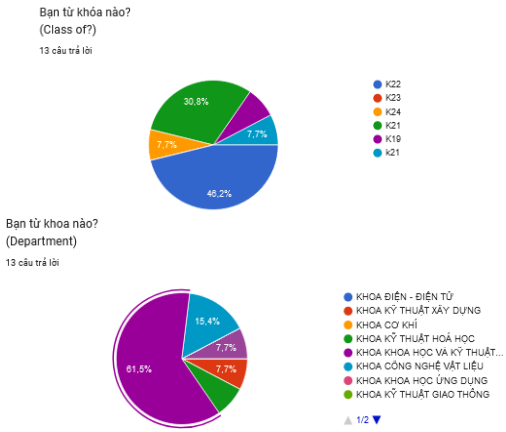
Figure 19: Usability Testing

4.2.2 Participants and Tasks

The usability test involved a total of 13 participants, who were selected to represent the target users of Verbal Mate. The participants were primarily students from HCMUT, aged between 18

and 23. This demographic is chosen as they best represent the target userbase. The participants had varying levels of experience with learning aids, with some having used similar services before, while others were completely new to such concepts.

Participants were asked to complete the following tasks, designed to evaluate the system's usability:



- Account Registration
- Login/ Logout
- Modify user info
- Create a library with at least 5 questions
- Modify library
- Answer by recording
- Answer by uploading
- Review previous answers

4.2.2.a Test Strategy

The usability test employed a qualitative approach to gather both subjective impressions and measurable data.

The test was conducted in person and using Google Forms to record and analyze data

4.2.2.b Test Procedure

1. Participants were briefed on the purpose of the test and provided with tasks.
2. Independently, each participant performs the assigned tasks, receiving additional guidance or moderation during the process only if necessary.
3. Upon completing the tasks, participants were asked to evaluate the system's usability and design and provide suggestions for potential improvements.
4. All observations and feedback are then recorded for detailed analysis.

4.2.2.c Metrics for Evaluation

The following metrics were used to assess the system's usability:

- Aesthetics: Participants' impressions of the page's visual design and overall appeal.
- Visibility: Clarity and accessibility of buttons and interactive elements.
- Tutorialization: Effectiveness of guidance or instructions provided to users.
- Intuitiveness: Ease of understanding and using the system without prior experience.
- Navigability: Smoothness of navigation between pages and features.

- Streamlining: How efficiently the system facilitates task completion.
- User Feedback: Qualitative insights on the system's design, functionality, and overall user experience.

The methodology ensured a comprehensive evaluation of both user performance and subjective experiences with the platform.

4.2.3 Test Results

4.2.3.a Findings Summary

The usability test revealed the following key findings:

Strengths:

Aesthetics: Participants found the overall design visually appealing, with a modern and clean interface.

Feedback System: Learners appreciated the immediate and clear feedback on cue word usage.

Intuitiveness: Most users were able to complete tasks without significant prior knowledge or guidance.

Areas for Improvement:

Visibility: Some buttons and interactive elements were difficult to locate, particularly on The question library page.

Navigation: Some participants expressed that some buttons and prompts are spaced too far apart, leading to slight difficulties in navifgation.

Tutorialization: New users expressed a desire for better action feedback such as successful logins or modifications.

4.2.3.b Qualitative Insights

User Feedback on Aesthetics:

Positive: "The interface is visually appealing and easy on the eyes."

Suggestions for Improved Usability:

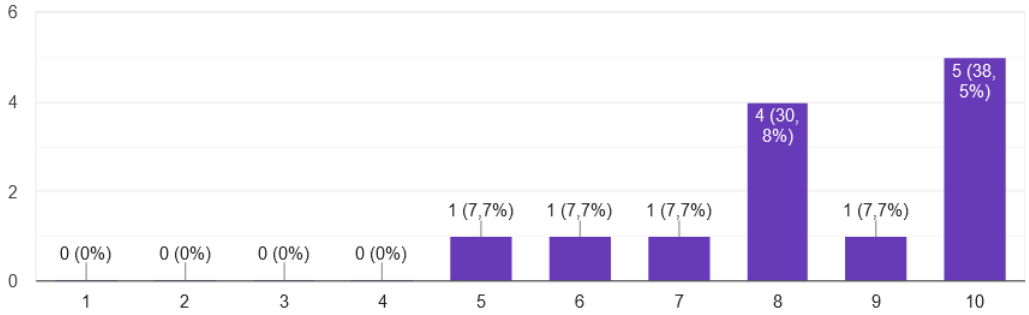
"Add more notifications to notify users of success."
"UI elements are too few, and the taskbar is too small."

4.2.3.c Figures

The findings demonstrate that while the system provides a strong foundation for English-speaking practice, enhancements in navigation, visibility, and tutorialization are required to optimize the user experience.

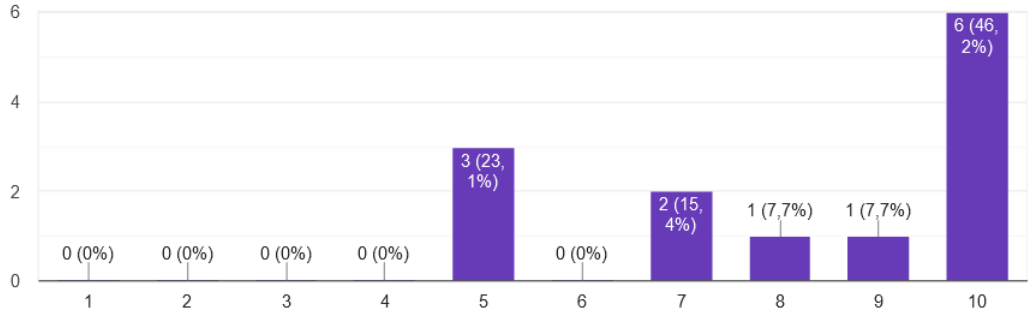
Bạn cảm thấy giao diện trang web như thế nào?
(Aesthetics?)

13 câu trả lời



Bạn có thể dễ dàng nhận diện các nút bấm không?
(Visibility?)

13 câu trả lời



Các mô tả có hữu dụng?
(Tutorialization?)

13 câu trả lời

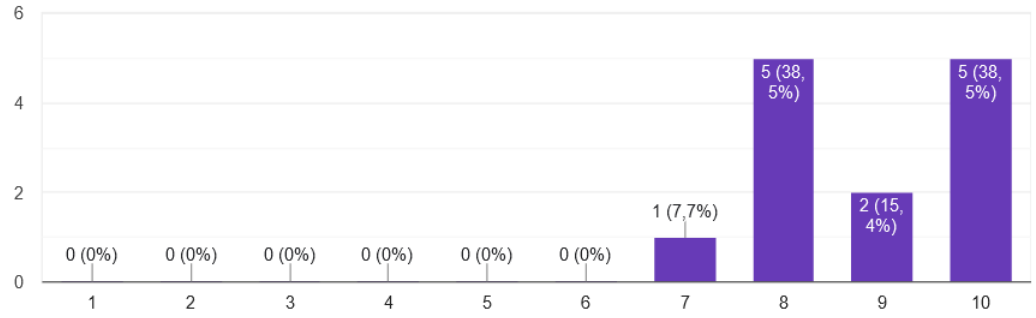
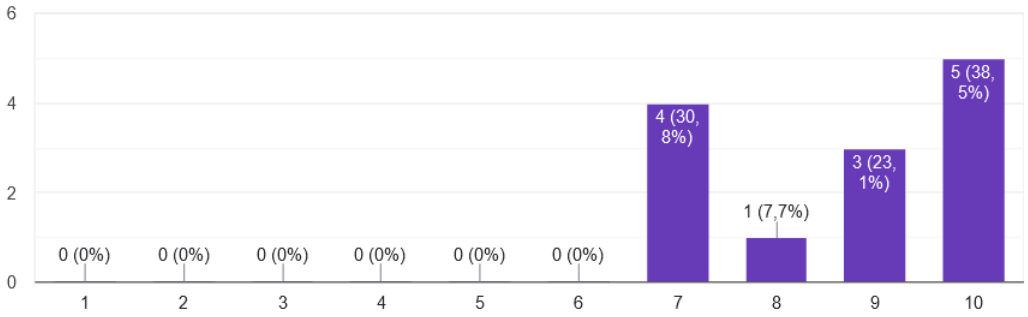


Figure 20: Questions about the interface

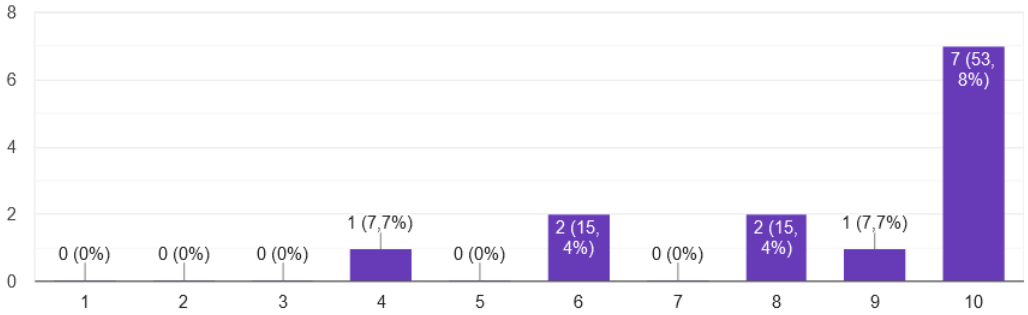
Trang web có trực quan dễ sử dụng?
(Intuitiveness?)

13 câu trả lời



Bạn có thể dễ dàng di chuyển giữa các menu không?
(Navigability?)

13 câu trả lời



Các công đoạn có rườm rà không cần thiết?
(Streamlining?)

13 câu trả lời

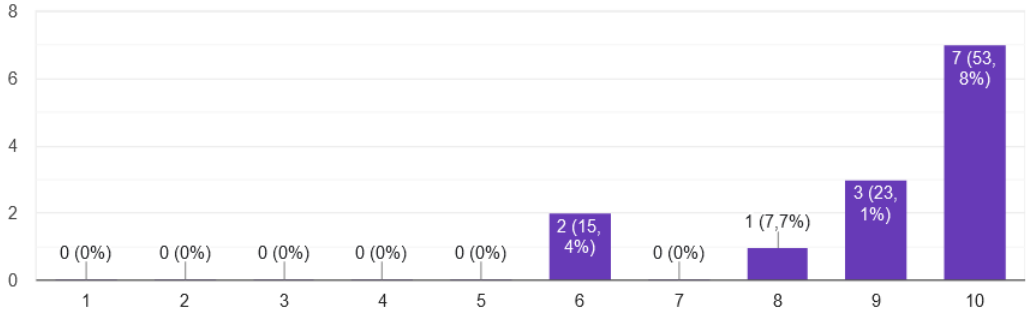


Figure 21: Questions about the user experience

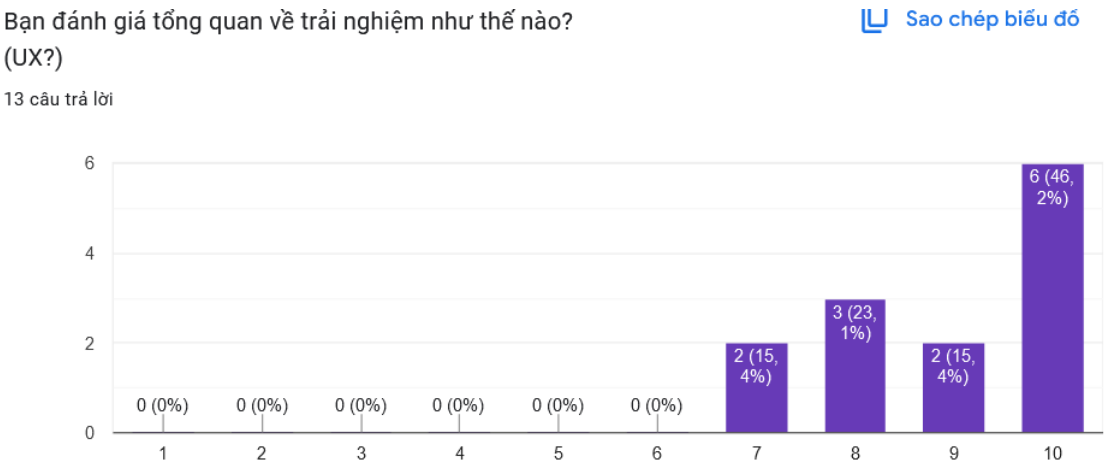


Figure 22: Overall evalutions

4.2.4 Summary

The usability test for MVP 1 of the English-speaking practice system provided valuable insights into its strengths and areas for improvement. Overall, participants found the platform visually appealing and intuitive, with learners appreciating the immediate feedback on cue word usage. Teachers valued the ability to manage learners and create structured exercises.

Key areas for improvement include enhancing button visibility, streamlining navigation, and providing quality-of-life features for users. While most participants were able to complete tasks successfully, challenges with specific features, such as locating buttons or switching between pages, highlighted the need for refinements.

Task 5: Implementation - Sprint 2

5.1 MVP 2

The MVP 2 builds on the foundation established by MVP 1, incorporating feedback and insights from Task 2.4 (dashboard design and stakeholder involvement) and Task 4.3 (usability testing). It retains the core functions and minimalist interface of MVP 1 but introduces minor adjustments to enhance usability, address identified pain points, and better align with stakeholder expectations.

KHANHHNE.COM

The entirety of the project and relating documentations are stored in the Github repository HERE

5.2 Product demonstration

The recording for this demo is included in the submission file or can accessed HERE

Conclusion

This report has detailed the comprehensive process of developing the software engineering integration project, encompassing its conceptualization, design, implementation, and refinement phases. Through iterative development and integration of feedback, the project has successfully delivered a functional, user-centric system that meets the outlined requirements while adhering to software engineering best practices.

Implemented Features and Functions:

- MVP 1: Demonstrated core the user interface, including question libraries, answer submissions, and role-specific dashboards. Minimalist interface emphasizing usability and clarity.
- MVP 2: Implemented the functionality based on MVP 1(Task 2.4), and iteratively improved on user experience with feedback from usability testing (Task 4.3). Enhancements included interface adjustments, and optimizations for navigation and runtime.

References

[] Ian Sommerville (2015), Software Engineering (10th ed.), ISBN 978-0133943030, Pearson

[] G. Booch, J. Rumbaugh, I. Jacobson (1998), The Unified Modeling Language User Guide, Addison- Wesley.

[] https://www.nngroup.com/articles/usability-testing-101/

[] https://www.nngroup.com/articles/quant-vs-qual/

[] https://www.nngroup.com/articles/talking-to-users/

[] https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/