GROUP DREAM PLANNING.PDF

Introduction

In the dynamic and ever-evolving field of software development, the concept of a final, unchangeable product is more a myth than a reality. Recognising this, our approach to developing and enhancing software products is inherently iterative. Each phase of development not only leads to tangible outputs but also opens new avenues for understanding and meeting the ever-changing needs of our users.

The primary purpose of this report is to demonstrate these processes of iterative software practices, applicable to the problems faced by users of quiz tools like Toohak. We aim to delve into the specifics of these issues, develop a clear set of requirements, and sketch out the early stages of design solutions that address these needs effectively. The approach we have adopted is user-centric, ensuring that the development aligns closely with the real-world challenges and expectations of our target audience.

We will first present the process and outcomes of our requirements elicitation phase, where we interact directly with users to gather insightful data. Following this, we will articulate these findings in the form of user stories and a detailed use case, aligning closely with the articulated needs. We will also discuss the validation of these requirements through feedback from the same set of users. The latter part of the report will focus on the initial design aspects, primarily in terms of interface design and conceptual modelling, which aims to formulate a better understanding of the dynamic nature of Toohak.

Requirement: Elicitation

This section details the process and outcomes of our initial step in understanding user needs and problems associated with quiz tools like Toohak. Our methodology involved direct engagement with potential users through interviews.

Users:

Target User 1: Nicholas Tseung (47 year old teacher) yantseung@gmail.com.au
Target User 2: Deep Batra (18 year old student) dbatraa05@gmail.com.au

Question 1: What challenges have you faced while using current quiz tools?

Target User 1: Nicholas Tseung	**Target User 2:** Deep Batra
My primary challenge has been the limited flexibility in quiz formats. I often find it difficult to tailor quizzes to different learning levels and styles. The lack of features for automated feedback hinders my ability to provide immediate guidance to students.	I've struggled with user interfaces that aren't very intuitive. Sometimes it's hard to navigate through quizzes, especially on mobile devices. Also, the quizzes often lack interactive elements, which makes them less engaging. For example, the Kahoot mobile quiz interface is only 4 coloured squares, which I personally feel can be fleshed out into something more exciting and appealing for a user.

Question 2: Are there any specific features you feel are missing or underdeveloped in Toohak or similar tools?

Target User 1: Nicholas Tseung	**Target User 2:** Deep Batra
Definitely. There's a need for more diverse question types, like drag-and-drop or matching, to allow different learning objectives for students. Integration with other educational tools for seamless data sharing such as quiz features that directly import questions from a textbook, would also be a significant improvement.	I think there should be a feature for real-time feedback or hints. It’s frustrating when you get something wrong and have no idea why. All you’re really given is some words of ‘motivation’ for the next question. Also, a way to track my progress over time through the completion of multiple quizzes would be great.

Question 3: How do you typically use quiz tools, and what improvements could enhance your experience?

Target User 1: Nicholas Tseung	**Target User 2:** Deep Batra
I use them for both formative and summative assessments. Improvements? More analytics features to track student performance trends would be helpful. That way, it would help me keep track of content that may need to be retaught to the class or specific individuals that need more guided help. And tools to easily customise quizzes to match curriculum changes would save a lot of time.	Mostly for revision before tests. It would be better if quizzes were more interactive and visually appealing so it acts as a sort of 'mind de-stresser' before the exam, while simultaneously getting your study in. Also, having a community feature where we can share quizzes with peers or get recommendations would be cool.

Question 4: Can you describe a situation where a quiz tool did not meet your needs? What would have been the ideal outcome?

Target User 1: Nicholas Tseung	**Target User 2:** Deep Batra
Once, I wanted to create a scenario-based quiz for a critical thinking exercise, but the tool I used had very limited options. Ideally, the tool should have allowed for more complex question structures and branching scenarios.	I remember a time when I was preparing for a chemistry quiz, and the tool had no diagrams or visuals for complex topics. Ideally, there should have been more multimedia content to help understand difficult concepts better.

Based on the elicited responses, we propose the following solutions:

- Enhanced UI/UX: Redesign the user interface to be more intuitive and user-friendly, incorporating feedback from both users, Deep and Nicholas.
- Integration with Educational Platforms: Develop APIs or plugins for seamless integration with popular educational tools, addressing Nicholas's concerns.
- Advanced Quiz Customization: Implement features that support a wider range of question types and multimedia content, catering to Deep's needs.

Requirement: Analysis & Specifications

Following the elicitation of user requirements through interviews with Nicholas Tseung and Deep Batra, this section aims to consolidate and articulate these findings into structured requirements. This involves transforming their expressed needs into user stories and defining a use case to address these requirements.

1. As a teacher (Nicholas), I want to have quizzes with various formats and automated feedback options, so I can cater to different learning levels and provide immediate guidance.

Acceptance Criteria:

- The quiz tool should support multiple formats (multiple-choice, short answer, drag-and-drop, etc.).
- Automated feedback should be provided for each question type.

2. As a student (Deep), I want an intuitive and interactive quiz interface, so that I can have an engaging and easy-to-navigate quiz experience on any device.

Acceptance Criteria:

- The interface must be user-friendly, especially on mobile devices.
- Interactive elements (like multimedia integration) should be incorporated to enhance engagement.

3. As a student, I need features like real-time feedback and progress tracking in the quiz tool, so I can understand my mistakes and monitor my learning progress.

Acceptance Criteria:

- The tool should provide instant feedback or hints after each question.
- There should be a feature to track and display the user's progress over multiple quizzes.

The target user’s problems can be solved by a renege of implementations, such as the features within the quiz tool that offer real-time feedback or hints to students during quizzes. Additionally, introduce a progress tracking system that allows students to monitor their performance over time across multiple quizzes. This solution addresses students' needs for immediate feedback

and self-assessment tools, enabling them to better understand their learning progress and areas where they need improvement.

Quiz platforms can also develop a feature-rich quiz creation tool that allows for a wide range of question types, including multiple-choice, drag-and-drop, matching, and scenario-based questions. This tool will enable teachers like Nicholas to create quizzes that are tailored to different learning styles and objectives. Additionally, integrating automated feedback mechanisms will allow students to receive instant, personalised responses to their answers, enhancing the learning experience. This solution directly addresses the need for diversified and flexible quiz formats, ensuring that quizzes can cater to a broader spectrum of educational requirements and learning environments.

Requirement: Validation

In this step, we reached back out to Nicholas and Deep to validate the requirements and proposed solutions developed from their initial feedback. The objective is to ensure that the use cases and design proposals accurately reflect and effectively address their needs. We conducted this by scheduling a follow-up virtual meeting with each user, during which the documented user stories, use cases, and proposed design solutions were presented.

Feedback on Use Cases and Solutions

Nicholas Tseung (Teacher)

Nicholas appreciated the focus on flexible quiz formats and automated feedback. He was particularly interested in the integration capabilities with educational platforms and textbooks, which he felt would significantly streamline his quiz creation process. He suggested ensuring that the integration features are compatible with a wide range of educational tools and not just a few popular ones.

Deep Batra (Student)

Deep found the proposed interactive and intuitive user interface appealing. He was especially enthusiastic about the real-time feedback and progress tracking features, which he believed would greatly aid his study and revision process. Deep recommended including a feature for community engagement, where students could share quizzes and resources, further enhancing the learning experience.

The final validation outcome is that both Nicholas and Deep confirmed that the use cases and proposed solutions align well with the problems they had highlighted. Their feedback provided valuable insights for refining the features, particularly regarding the breadth of integration capabilities and the inclusion of community features for students.

Design: Interface Design

This section outlines the proposed interface design enhancements and new functionalities for the Toohak quiz tool, considering the needs of Nicholas Tseung and Deep Batra. The design will specify capabilities as HTTP endpoints, drawing inspiration from the existing structure in the `swagger.yaml` file.

1. POST /v1/quiz/create/formats
This endpoint allows for the creation of a new quiz with customizable formats and question types.

Parameters

- teacherId` (e.g., 12345)
- quizDetails (JSON object containing quiz title, description, question types, etc.)

Responses

- 201: Quiz successfully created (returns quiz ID and creation details).
- 400: Invalid input or missing required fields.

2. GET /v1/quiz/{quizid}/interactive-elements
Retrieves all interactive elements available for a specific quiz, enhancing student engagement.

Parameters

- quizid (e.g., 5546)

Responses

- 200: Successfully retrieved interactive elements (returns a list of elements).
- 404: Quiz not found.

3. PUT /v1/quiz/{quizid}/integration/textbook
Links a quiz to textbook content for easy import of questions.

Parameters

- quizid (e.g., 5546)
- textbookDetails (JSON object containing textbook name, edition, chapters, etc.)

Responses

- 200: Successful integration.
- 400: Integration failed (invalid textbook details or quiz ID).

Design: Conceptual Modelling

This section presents a state diagram to visually represent how the application's state changes in response to various user interactions, particularly for the new features we are proposing. The state diagram aims to provide developers with a clear understanding of the application's behaviour and flow.

State Diagram for Quiz Creation and Interaction Flow

1. State Diagram Overview
The diagram illustrates the flow from quiz creation by a teacher (like Nicholas) to quiz participation and progress tracking by a student (like Deep).

2. Key States
- Start: Initial state of the application.
- Create Quiz: State where a teacher creates a new quiz.
- Publish Quiz: The quiz is published and available for students.
- Quiz Taking: State where a student is actively taking the quiz.
- Quiz Interaction: Represents the student interacting with various quiz elements (like hints, multimedia).
- Progress Tracking: State where a student views their progress.
- Community Interaction: State representing the sharing and discussion of quizzes in the student community.
- End: Final state post-quiz interaction.

3. Transitions
- From Start to Create Quiz: Triggered by a teacher's decision to create a new quiz.
- From Create Quiz to Publish Quiz: Occurs after the quiz is created and set up for student access.
- From Publish Quiz to Quiz Taking: Initiated when a student starts the quiz.
- Within Quiz Taking, transitions to Quiz Interaction based on student actions (e.g., requesting hints).
- From Quiz Taking to Progress Tracking: After completing a quiz, the student can view their progress.
- From Progress Tracking to Community Interaction: Students choose to share results or discuss quizzes.
- From any state to End: Completion of the activity or exiting the application.

4. Diagram Implementation
- The diagram will be created using a tool like Lucidchart or Microsoft Visio to ensure clarity and professionalism.
- Each state will be clearly labelled and transitions will be marked with arrows, including conditions that trigger these transitions.

5. Inclusion in the Report:
    - The finalised state diagram will be included in the planning.pdf to assist in understanding the proposed enhancements to the Toohak application from a developer's perspective.