

# *STARBOOKS: WHIZ CHALLENGE*

Project Documentation Submitted to the Faculty of the  
School of Computing and Information Technologies

Asia Pacific College

In Partial Fulfillment of the Requirements for  
Introduction to Systems and Design for IT  
MNTSDEV

By

Kelly Dumbrique  
Shandrae Lois Quianzon

Arcielle Marie Gercan  
Janice Maxene Salipande

## Table of Contents

List of Figures.....	iv
List of Tables.....	1
Introduction .....	2
Project Context.....	3
Statement of the Problem .....	4
Objectives.....	5
Significance of the Project .....	6
Scope and Limitations .....	7
Review of Related Literature / Systems .....	9
Current System .....	16
Technical Background.....	16
List of Processes .....	17
SWOT Analysis .....	23
Proposed Solution.....	24
Technical Background.....	24
Feasibility.....	26
Operational Feasibility .....	26
Economic Feasibility .....	26
Technical Feasibility .....	27
Schedule Feasibility.....	27
Requirements Analysis .....	28
Project Vision.....	28
Prototype (Mock Flow / Wireframe).....	29
Project Lean Canvas .....	39
User Classes and Characteristics .....	40
Product Backlog.....	41
Product Roadmap.....	43
Release Plan .....	44
Use Case Diagram .....	46
Use Case Full Description.....	47
Conclusion .....	69
References.....	70

Appendices .....	75
Appendix A: Roles and Responsibilities .....	75
Appendix B: Minutes of the Meetings .....	77
Appendix C: Methodology .....	88
Appendix D: Project Sharepoint Link .....	90
Appendix E: Requirements Traceability Matrix .....	91
Appendix F: RACI Matrix.....	93

## List of Figures

Figure 1. Touchscreen All-in-One PC .....	16
Figure 2. Standard PC Setup .....	16
Figure 3. Registration Page: Terms and Conditions .....	19
Figure 4. Log in Page .....	19
Figure 5. Manage Profile .....	20
Figure 6. Homepage .....	20
Figure 7. Question Interface .....	21
Figure 8. Participant List Menu .....	21
Figure 9. Admin List Menu .....	22
Figure 10. Question List Menu .....	22
Figure 11. Difficulty Settings Menu .....	23
Figure 12. Player Registration: Terms and Conditions .....	29
Figure 13. Player Registration: Account Setup .....	29
Figure 14. Player Log in .....	30
Figure 15. STARBOOKS Whiz Challenge Homepage .....	30
Figure 16. Whiz Challenge: Select Category .....	31
Figure 17. Whiz Challenge: Science Easy Level .....	31
Figure 18. Whiz Challenge: Science Easy Level Question .....	32
Figure 19. Whiz Memory Match: Easy Level .....	32
Figure 20. Whiz Puzzle: Easy Level .....	33
Figure 21. Whiz Battle: Join or Create Battle .....	33
Figure 22. Whiz Battle: Waiting Room .....	34
Figure 23. Whiz Battle: Battle Result .....	34
Figure 24. Player Stats .....	35
Figure 25. Admin Dashboard Log in .....	35
Figure 26. Admin Dashboard: Analytics Dashboard .....	36
Figure 27. Admin Dashboard: List of Players .....	36
Figure 28. Admin Dashboard: List of Admins .....	37
Figure 29. Admin Dashboard: List of Questions .....	37
Figure 30. Admin Dashboard: Difficulty Settings .....	38
Figure 31. Product Roadmap .....	43
Figure 32. Use Case Diagram .....	46

## List of Tables

Table 1. Current System Process.....	17
Table 2. SWOT Analysis.....	23
Table 3. Project Vision.....	28
Table 4. Project Lean Canvas .....	39
Table 5. User Classes and Characteristics .....	40
Table 6. Product Backlog.....	41
Table 7. Release Plan .....	44
Table 8. Use Case: Register Account .....	47
Table 9. Use Case: Login .....	48
Table 10. Use Case: Edit Profile .....	49
Table 11. Use Case: Play Whiz Challenge .....	50
Table 12. Use Case: View Badges.....	51
Table 13. Use Case: Play Whiz Memory Match.....	52
Table 14. Use Case: Play Whiz Puzzle.....	54
Table 15. Use Case: Play Whiz Battle .....	55
Table 16. Use Case: View Stats.....	57
Table 17. Use Case: Logout.....	58
Table 18. Use Case: Manage Players.....	59
Table 19. Use Case: Manage Admins.....	61
Table 20. Use Case: Manage Quiz Questions.....	62
Table 21. Use Case: Manage Difficulty Levels .....	64
Table 22. Use Case: Manage Rewards .....	65
Table 23. Use Case: View Analytics .....	67
Table 24. Roles and Responsibilities .....	75
Table 25. Requirements Traceability Matrix.....	91
Table 26. RACI Matrix .....	93

## Introduction

The Department of Science and Technology (DOST) is the lead government agency in the Philippines tasked with directing, coordinating, and implementing policies, programs, and projects related to science and technology [1]. Its mandate covers a broad range of sectors, from education and industry to health and disaster resilience, all with the goal of using innovation to support national development and improve the quality of life of Filipinos [1].

Under DOST is the Science and Technology Information Institute (DOST-STII), which serves as the agency's official information arm. DOST-STII plays a key role in collecting, managing, and disseminating credible Science, Technology, and Innovation (STI) information across the country [2].

In line with this, DOST-STII's mission focuses on [2]:

- Providing credible and inclusive STI information through resource sharing and efficient delivery systems;
- Promoting public awareness, understanding, and appreciation of STI and its role in national development; and
- Empowering key stakeholders as partners and advocates in building a culture of STI.

To fulfill this mission, DOST-STII developed STARBOOKS, short for Science and Technology Academic and Research-Based Openly Operated KioskS [3]. STARBOOKS is the country's first offline digital library designed to bring S&T learning resources to communities that lack reliable internet access. Installed in schools, libraries, and public centers nationwide, it provides free and ready access to a wide array of educational materials—including science references, research papers, instructional and livelihood videos, K–12 learning modules, and financial literacy resources [4]. These materials are sourced from the DOST-STII Library and its institutional partners [3]-[4].

Beyond addressing the digital divide, the STARBOOKS initiative contributes to broader national and global goals. It aligns with the United Nations Sustainable Development Goals (SDGs), particularly:

- SDG 4: Quality Education, by promoting inclusive and equitable access to educational resources [6], and
- SDG 9: Industry, Innovation, and Infrastructure, through the use of innovative, offline technology to bridge gaps in education and knowledge access [6].

Moreover, STARBOOKS supports the National Innovation Agenda and Strategy Document (NIASD), particularly Innovation Priority Sector 1: Learning and Education [7]. This sector highlights the importance of enhancing the delivery of education through innovative solutions that address access, quality, and equity—objectives that STARBOOKS directly addresses through its scalable, inclusive approach to knowledge dissemination.

Through these efforts, STARBOOKS continues to be a critical tool in fostering a culture of science, technology, and innovation, and in empowering communities with knowledge—no matter their location or connectivity.

## **Project Context**

To further promote scientific literacy and student engagement, the STARBOOKS Whiz Challenge was introduced as a gamified educational platform that complements the STARBOOKS project [3]-[4]. Developed by DOST-STII, this application aims to make learning about Science, Technology, Engineering, and Mathematics (STEM) topics more enjoyable—especially for students attending DOST events, science fairs, and public exhibits.

The STARBOOKS Whiz Challenge is a standalone, offline quiz-based application that runs on STARBOOKS kiosk setups [3]-[4]. It primarily features a solo quiz game where players answer multiple-choice questions on various STEM topics, aligned with the Grade 1 to 12 curriculum. It allows players to choose a difficulty level—Easy, Average, or Difficult—and answer a fixed set of questions accordingly.

A unique feature of the Whiz Challenge is its badge and reward system. After completing a quiz:

- Players who answer all questions correctly earn a badge for the chosen difficulty level.
- Once a player collects three badges in the same difficulty, they become eligible to claim a physical reward, such as educational tokens or small prizes at the DOST booth.

This simple reward-based gamification approach motivates students to try multiple times, review their knowledge, and improve their performance—all while having fun. It transforms the traditional quiz experience into something more interactive, competitive, and goal-driven.

The STARBOOKS Whiz Challenge is currently deployed during DOST-led public events, science exhibits, education fairs, and outreach activities, where STARBOOKS kiosks are physically set up. Because it works offline, the system remains accessible and functional even in venues with no internet connectivity—ensuring equal participation from students in both urban and rural areas [3]-[4].

Through this initiative, DOST-STII continues to reinforce the value of accessible, innovative learning tools—empowering the next generation of Filipino learners to explore science and technology with curiosity, confidence, and excitement [8].

To better understand the existing processes of the current STARBOOKS system, including visual references, please refer to the List of Processes on Page 17.

## **Statement of the Problem**

DOST-STII has long advocated for making science and technology education more accessible and engaging, especially for learners in disadvantaged and underserved areas [3]-[5]. Through the STARBOOKS initiative and its outreach activities, one of their key efforts to spark interest among students has been the STARBOOKS Whiz Challenge—a gamified quiz application deployed during science fairs, exhibits, and educational events.

While the current version of the Whiz Challenge has successfully introduced fun, interactive STEM learning through a solo quiz game, DOST-STII has observed several limitations that affect the system's ability to fully meet its engagement and outreach goals:

### **1. Limited replayability and player retention**

As the app currently supports only a single game mode (solo quiz), DOST-STII has noted reduced student engagement over time. Once users have earned badges and rewards, the gameplay becomes repetitive, limiting opportunities for deeper learning or continued use.

### **2. Lack of diverse learning strategies**

The existing format does not accommodate different types of learners. Students who learn better through collaboration, competition, or visual-based games have fewer options to interact with the content in a way that suits their learning style.

### **3. Difficulty in measuring impact and player insights**

Without in-app performance tracking or analytics tools, it is difficult for DOST-STII to gather data about player behavior, content effectiveness, or trends during events. This hinders their ability to refine future outreach strategies or evaluate the educational impact of the app.

### **4. Scalability and deployment limitations**

The fully offline nature of the app means player progress is tied to individual kiosks, and admin-side content updates must be done manually per device. This fragmented setup limits the scalability of the system, especially during nationwide deployments or multi-location exhibits.

### **5. Need for richer, more modern user experience**

As digital learning tools continue to evolve, DOST-STII aims to keep the Whiz Challenge visually appealing and competitive with modern educational apps. However, the current system lacks dynamic visuals, animations, and polish that would better capture the attention of today's learners.



## Objectives

This project aims to redesign and enhance the existing STARBOOKS Whiz Challenge system in alignment with DOST-STII's goal of making STEM learning more engaging, data-driven, and scalable for students across the country. Specifically, the system seeks to achieve the following objectives:

### **1. Increase user engagement and replayability**

- Introduce multiple game modes to provide varied gameplay experiences and reduce repetition.
- Encourage longer and repeated usage of the app by integrating interactive and fun learning elements.

### **2. Support different learning styles**

- Develop features that cater to various learner types—such as competitive and visual-based.
- Promote both individual and social learning through solo and multiplayer experiences.

### **3. Enable performance tracking and player insights**

- Integrate in-app features to track individual player performance, quiz scores, and badge collection.
- Provide users with insights to support self-paced learning and continuous improvement.

### **4. Improve system scalability and manageability**

- Implement offline-first but sync-capable architecture to allow local data storage with optional synchronization across kiosks.
- Equip admins with tools to update quiz content, monitor usage trends, and manage game events more efficiently.

### **5. Enhance visual appeal and user experience**

- Redesign the user interface with modern layouts, animations, and intuitive navigation to improve overall aesthetics and usability.
- Use age-appropriate, engaging visuals to attract and retain attention—especially among younger users.

## Significance of the Project

The enhanced STARBOOKS Whiz Challenge app aims to provide a more engaging, gamified learning experience that supports both academic growth and digital inclusion. This section outlines the key groups who will benefit from the project:

**To exhibit visitors.** Visitors who attend DOST-hosted science fairs, exhibits, or events will have a more interactive and memorable experience through the improved STARBOOKS Whiz Challenge system. The system will serve not only as a learning tool but also as an attraction that draws attention to the importance of STEM education in a fun and game-based format.

**To students.** Grades 1 to 12 students, especially those in remote or underserved areas, will gain access to a fun, interactive, and self-paced way of learning science and math. The addition of new game modes, visual enhancements, and performance tracking can improve motivation, increase retention of STEM concepts, and encourage healthy competition. With or without constant internet access, students will have the opportunity to enjoy learning in a format that suits their environment and preferences.

**To educators and facilitators.** Educators and facilitators will benefit from a more engaging tool they can use to support classroom instruction or supplemental learning. The upgraded admin panel with game analytics will help them track user performance, identify learning trends, and adjust instructional strategies accordingly.

**To DOST-STII.** DOST-STII will benefit by having an improved version of an existing project that is more aligned with their long-term mission of spreading science and technology awareness. This upgraded system can serve as a showcase of their ongoing innovation efforts and help them reach more learners nationwide, especially in digital learning campaigns, educational exhibits, and STEM advocacy events.

**To communities and learning centers.** Local learning hubs, barangay e-centers, libraries, and school-based STARBOOKS kiosks will gain access to a more dynamic educational tool. The app can encourage more learners to use STARBOOKS and support community-based learning even in places with limited connectivity.

**To national education goals.** By improving digital and STEM education accessibility, the STARBOOKS Whiz Challenge contributes to SDG 4: Quality Education by ensuring inclusive and equitable learning opportunities for all [6], and to SDG 9: Industry, Innovation, and Infrastructure by promoting the use of innovative digital solutions in education [6]. It also aligns with the National Innovation Agenda and Strategy Document (NIASD) under Innovation Priority Sector 1: Learning and Education, supporting the country's push for a future-ready, innovation-driven education system [7].

## Scope and Limitations

The STARBOOKS Whiz Challenge application is developed as a gamified educational platform to supplement the existing STARBOOKS ecosystem. Its initial release focuses on delivering a solid and engaging experience that supports offline-first learning for students in schools, learning hubs, and DOST-STII exhibits. This first version aims to implement the project's Minimum Viable Product (MVP) — a functional prototype that prioritizes core features for deployment and testing.

Included in the **first major release**:

- **User registration and login system** for tracking individual player progress
- **Four game modes** available:
  - Whiz Challenge (Solo Quiz – offline)
  - Whiz Memory Game (offline)
  - Whiz Puzzle (offline)
  - Whiz Battle (1v1 Quiz Showdown – requires internet for game code matchmaking)
- **Science and Math subject categories** with three difficulty levels (**Easy, Average, Difficult**)
- **Badge and reward system** to recognize player achievements and boost motivation
- **Player statistics dashboard** showing earned badges, scores, best time, accuracy, battle history, win rate, and more
- **Admin panel** with functionalities to:
  - Manage player and admin lists
  - Manage quiz questions with AI-assisted question generation using OpenAI
  - Set number of questions, scores, and timer for each difficulty level
  - Manage badges and reward claiming
  - View and export game analytics
    - Total registered players
    - Player Registration Trend
    - Most Played Game Mode
    - Most Played Level Per Game Mode
    - Top Category
    - Average Session Duration
    - Reward Claims Summary
- Data is **stored locally first** and **syncs to the cloud** when internet becomes available

Due to resource constraints and the developmental scope, the STARBOOKS Whiz Challenge system has the following **limitations**:

- No integration with the main STARBOOKS digital library system
- No mobile or tablet version; limited to kiosk deployment only
- No topic-specific categories for Whiz Memory Match; only default STARBOOKS-related icons are used
- Only jigsaw-type puzzles are supported in Whiz Puzzle; other puzzle formats (e.g., word games) are not included

- No image customization for memory and puzzle games; game assets and settings are fixed and cannot be modified by admins
- Whiz Battle is limited to 1v1 public matchmaking via game code; private rooms and other multiplayer formats are not yet supported
- No player-specific analytics; the admin dashboard displays only overall trends and summary statistics across all users

This scope ensures that the project delivers its core value—an interactive, student-friendly, and offline-first learning companion—while staying realistic and achievable within the project’s timeline and available resources. The listed limitations represent future enhancement opportunities once the MVP has been successfully deployed and evaluated.

## Review of Related Literature / Systems

This section reviews key themes related to the STARBOOKS Whiz Challenge, including educational approaches like gamification and reward systems, as well as the technologies and tools used in similar offline and interactive learning platforms. These themes provide the foundation for the design and development of the proposed system.

### Gamification in Education

Gamification is the integration of game-like elements—such as points, badges, levels, and challenges—into non-game settings to enhance user engagement and motivation. In education, it has proven especially effective in increasing student participation, promoting retention, and making learning more enjoyable [11].

Studies have shown that gamified platforms help learners grasp complex subjects better by turning tasks into interactive challenges. This is particularly impactful in Science and Math education, where problem-solving and repeated practice are essential [8], [11]. Immediate feedback, progress indicators, and milestone rewards contribute to improved performance and deeper learning [11], [13].

The STARBOOKS Whiz Challenge applies these principles through features like quiz badges, progress levels, and 1v1 quiz battles. These mechanics not only increase engagement but also instill a sense of accomplishment, encouraging sustained use.

Gamification is also valuable in low-resource settings, where access to quality learning materials may be limited. It transforms learning into an immersive, game-like experience, helping students stay motivated even in offline environments [13]-[14].

By incorporating these proven strategies, STARBOOKS bridges education and entertainment—making learning more fun, rewarding, and effective for Filipino students in underserved communities.

### Offline Learning Technologies

Offline learning technologies are vital for delivering education in areas with limited or no internet access. These solutions help close the digital divide by enabling students in underserved or remote communities to access educational content without the need for continuous connectivity [15]. According to UNESCO, such technologies are key to achieving inclusive and equitable education, especially in rural and marginalized regions [6].

Kiosk-based systems like STARBOOKS exemplify this approach by storing digital libraries and interactive materials locally. Users can access videos, documents, and quizzes directly from the device without needing to go online, ensuring uninterrupted learning even in areas with poor infrastructure [4].

Globally, tools like RACHEL (Remote Area Community Hotspot for Education and Learning) and Kolibri serve similar goals, offering modular offline content delivery with minimal

hardware requirements [16]-[17]. These platforms are designed for low-power, low-maintenance environments and are optimized for accessibility in challenging contexts.

User-friendly interfaces—often built using HTML or touchscreen-friendly frameworks—further enhance accessibility, particularly for young learners or those unfamiliar with digital systems [18]. The STARBOOKS Whiz Challenge adopts these principles by combining offline functionality with gamified experiences and touchscreen interaction.

In sum, offline learning technologies offer a practical, scalable, and inclusive way to deliver education, making them highly suitable for initiatives like STARBOOKS that aim to empower learners regardless of internet availability.

### **Interactive Educational Kiosk Systems**

Interactive educational kiosks are standalone digital terminals designed to deliver educational content in public spaces like libraries, schools, and community centers [18]. These systems enable learners to access resources independently, often without personal devices or internet connectivity—making them ideal for underserved communities.

Such kiosks are typically built using cross-platform frameworks like Flutter Desktop or Electron, allowing developers to package applications as standalone systems compatible with touchscreen Windows PCs [20]-[21]. They may also include lightweight local web servers (e.g., Apache, Nginx) and UI toolkits optimized for touch navigation. Kiosk wrappers or system locking tools are often employed to prevent users from navigating outside the intended interface.

In the Philippines, DOST-STII's STARBOOKS exemplifies this model. As the country's first digital science library accessible offline, STARBOOKS offers curated content in Science and Technology to students in remote areas [4], [8]. While the current version is largely static or quiz-based, there is an increasing push to integrate gamified and interactive components to enhance engagement.

Globally, educational kiosks are evolving to include user accounts, instant feedback, and analytics—transforming them from passive information displays into dynamic, personalized learning environments [22]. These advancements support new features like multi-mode educational games, offline data storage, and real-time progress tracking—core to the design of STARBOOKS Whiz Challenge.

### **Offline-First System Architecture**

An offline-first system is designed to prioritize full functionality even in the absence of an internet connection, syncing data only when connectivity becomes available [23]. By enabling core features to run offline, such systems increase accessibility and reliability, especially in the education sector where consistent access to digital learning materials is critical [23]-[24].

Offline-first applications rely on storing data locally—usually within the device's internal storage or an embedded database—then synchronizing it to a central server once a stable connection is detected [24]. Technologies such as service workers, local databases (e.g., IndexedDB or SQLite), and caching strategies are often used in web and hybrid app development

to enable offline access [24]. These techniques allow systems like e-learning platforms, digital libraries, and health apps to serve content and capture user activity regardless of connectivity status.

In the case of STARBOOKS Whiz Challenge, adopting an offline-first architecture ensures that both players and administrators can access quizzes, track performance, and manage content without requiring a continuous internet connection. This aligns perfectly with the STARBOOKS initiative's commitment to bridging the digital divide, especially in underserved areas. Learners can play quizzes and earn rewards even when offline, while admin-side tools such as quiz editing or event setup remain usable on-site.

## **Digital Quizzing and Adaptive Learning**

Digital quizzing platforms have reshaped education by combining self-assessment, instant feedback, and gamified mechanics to boost student motivation and engagement [25]-[26]. Adaptive learning tools like Koobits tailor content based on a student's responses—adjusting question difficulty and scope to target learning gaps and reinforce mastery. This individualized approach enhances critical thinking and promotes better knowledge retention.

Koobits, popular in Southeast Asia, integrates real-time feedback, performance tracking, and gamified rewards such as badges and stars—encouraging students to stay motivated through measurable progress [27]. Similarly, platforms like Quizizz and Kahoot! have made quizzing interactive and social through live competitions and multiplayer game modes, fostering collaboration and active learning. However, their reliance on internet connectivity makes them less accessible to students in offline or low-bandwidth areas.

Technically, many of these platforms use backends built with Node.js, Firebase, or Laravel, while frontends are often developed using React or Angular. For offline setups, lightweight databases like SQLite or embedded JSON-based systems are commonly used. These allow quizzes to be delivered and tracked locally, ensuring smooth performance even without an internet connection [28]-[29].

The STARBOOKS Whiz Challenge draws from these models—offering quiz game modes with adjustable difficulty levels, immediate feedback, and motivational features like badges and progress tracking. Unlike fully online platforms, it's designed for offline use within educational kiosks, making adaptive learning accessible to remote learners while retaining the strengths of modern quiz systems.

## **Memory Matching Games and Cognitive Skills**

Memory matching games—also known as concentration or card-pairing games—are proven tools for enhancing cognitive functions such as short-term memory, attention span, and visual-spatial reasoning [30]-[31]. These games challenge players to recall the location of specific cards or images, reinforcing mental alertness, focus, and pattern recognition—skills crucial for academic success in science and mathematics.

Studies have found that consistent exposure to memory games can improve working memory and cognitive flexibility in learners, which supports better problem-solving and critical

thinking [30]-[31]. Because of their intuitive mechanics and minimal language dependency, these games are also well-suited for learners of various literacy levels and ages.

From a development standpoint, memory matching games are typically lightweight and can be built using engines like Phaser.js, Unity, or Flutter Flame, making them ideal for deployment on offline-capable kiosk systems [32]. These frameworks support smooth performance on low-spec hardware, ensuring broad accessibility.

Incorporating a memory game mode into the STARBOOKS Whiz Challenge adds a fun yet educational activity that supports mental development while maintaining offline functionality—making it both practical and pedagogically valuable for learners in underserved areas.

### **Educational Puzzles and Problem-Solving**

Educational puzzles—such as jigsaw puzzles, sequence games, and logic-based challenges—are powerful tools for enhancing critical thinking and problem-solving skills among learners [33]-[34]. These activities require players to analyze patterns, organize components, and develop strategies, thereby improving spatial awareness, persistence, and attention to detail.

Studies highlight that puzzle-based learning not only boosts cognitive functions like concentration and memory but also encourages flexible thinking—an essential skill in STEM education [33]-[34]. By engaging in hands-on problem solving, learners develop confidence in approaching unfamiliar tasks and applying logical reasoning.

From a development perspective, educational puzzle games are typically created using engines such as Unity, Godot, or Flutter Flame, depending on the platform's graphical and interaction needs [35]-[36]. Features like drag-and-drop mechanics and progress feedback can be built using HTML5 Canvas, JavaScript, or Flutter widgets, while offline progress saving is often handled through Hive or SQLite for local storage [35]-[36].

For the STARBOOKS Whiz Challenge, integrating puzzles adds variety to the learning experience while reinforcing analytical skills through fun, interactive challenges that are accessible even without internet connectivity.

### **Competitive Learning and 1v1 Game Dynamics**

Competitive learning leverages peer interaction and timed challenges to boost motivation, engagement, and knowledge retention among students [37]-[38]. Platforms like Kahoot!, Quizizz, and Gimkit have popularized multiplayer quiz formats, where learners race to answer questions correctly and quickly—turning learning into an exciting, game-like experience. These systems demonstrate that competition not only drives focus and active recall but also builds confidence and encourages repeated practice.

Technically, real-time multiplayer features in these platforms are powered by tools like Socket.IO, Realtime Database, or WebSockets, which allow synchronization of game data across players [39]-[40]. For systems like Kahoot!, match-based entry codes (room PINs) simplify multiplayer setup and invite participation without requiring complex logins. Frontend frameworks



like Flutter and React provide smooth, responsive interfaces for real-time play, while backend logic can be handled using Node.js or Laravel Echo servers.

In the context of the STARBOOKS Whiz Challenge, a 1v1 quiz battle mode is introduced with game-code matchmaking. This lets two players input a shared code to initiate a private, real-time duel—requiring minimal internet usage while preserving an engaging, competitive format. Unlike typical platforms, this design fits into a kiosk-based setup that works offline for most features and only briefly connects online for this battle mode.

By integrating this dynamic into the system, STARBOOKS brings the thrill of competitive learning into underserved areas, promoting cognitive agility, quick decision-making, and peer interaction—even with limited connectivity.

### **Reward Systems and Motivation in Learning Platforms**

Reward systems are fundamental in educational platforms, helping sustain learner engagement by reinforcing progress and achievement [41]-[42]. Gamified features like badges, experience points (XP), streaks, and level progression are widely used in platforms such as Duolingo [43], Khan Academy [44], and ClassDojo [45], offering students a sense of accomplishment and a visual representation of their learning journey.

These rewards act as immediate, positive feedback—boosting intrinsic motivation, improving task completion rates, and increasing long-term retention of content [41]-[42]. For younger or less confident learners, even simple visual rewards can significantly enhance focus and persistence.

The STARBOOKS Whiz Challenge applies similar principles through its badge collection system, level tracking, and a 1v1 battle win/loss record, offering recognition and incentive for consistent play and improvement. These systems are fully functional even in offline settings, using local storage methods such as SQLite or Hive, and can be synced when connectivity is available.

Technically, reward mechanics are implemented using internal logic managed by backend frameworks like Laravel, Firebase Firestore, MongoDB, or Supabase, depending on the architecture [46]. Progress data is typically tied to user sessions or device-specific identifiers in kiosk systems, ensuring continuity even without online accounts.

By integrating well-researched motivational mechanics into the STARBOOKS environment, the platform fosters self-driven learning and sustained academic curiosity—especially important in areas with limited educational support.

### **Player Statistics and Progress Tracking**

Tracking player performance is a vital feature in gamified learning platforms, providing learners with meaningful feedback and enabling educators or administrators to assess learning outcomes [47]. Popular platforms like Duolingo and Quizizz utilize detailed player statistics—including correct answers, streaks, time spent, and topic mastery—to personalize learning paths and maintain student engagement over time.

Progress tracking empowers students to monitor their own growth, recognize strengths and weaknesses, and stay motivated by seeing tangible evidence of their achievements [47]. In competitive or 1v1 game modes, statistics such as win/loss records, score histories, and response times enhance the competitive spirit and encourage ongoing improvement.

Technically, these tracking features are commonly supported by relational databases like MySQL or PostgreSQL, or embedded databases like SQLite for offline use [48]. Visualization tools such as Chart.js, D3.js, or Flutter's `charts_flutter` package are often employed to create intuitive graphical representations of player performance data.

For the STARBOOKS Whiz Challenge, player stats—including quiz results, badge achievements, and battle mode records—are stored locally, allowing the system to function fully offline. This approach is crucial for accessibility in remote or underserved communities where stable internet connections are scarce. Despite operating offline, the platform effectively mirrors the feedback and progress tracking loops found in online systems, promoting sustained learning and self-assessment.

### **Admin Interfaces in Educational Platforms**

Admin interfaces are essential components of educational platforms, enabling administrators and educators to efficiently manage content, monitor student progress, and extract meaningful insights from user data [49]-[50]. Well-known platforms like Kahoot! and Quizizz provide robust admin dashboards that facilitate quiz creation, performance analytics review, and customization of game settings to meet diverse learning objectives [50].

Key functionalities of an effective admin interface typically include:

- **Content Management:** Adding, editing, and organizing questions or games.
- **User Monitoring:** Tracking player activity, engagement, and progress.
- **Analytics and Reports:** Generating data-driven insights on student performance and usage patterns.
- **Settings and Customization:** Managing access controls, difficulty levels, and game modes.

For offline systems such as the STARBOOKS Whiz Challenge, the admin interface is designed to function fully on the local device. Despite the absence of internet connectivity, it supports critical features like badge tracking, player performance summaries, and quiz content management. This offline capability empowers administrators—such as teachers, librarians, or event facilitators—to maintain and oversee educational content and student engagement effectively in low-connectivity environments, without dependence on cloud services.

Admin dashboards are commonly developed using frameworks and templates like Django, Laravel, React Admin, or Vue Admin Panel, offering CRUD (Create, Read, Update, Delete) operations for managing quiz data, user progress, and exporting reports [51]. Offline-capable admin interfaces are often bundled with the main kiosk application and incorporate Role-Based Access Control (RBAC) systems to securely differentiate between admin and student users, with configurations managed locally or through JSON-based access controls.

## AI-Assisted Content Generation

The integration of artificial intelligence (AI) in educational systems has transformed the way content is created and managed [52]-[53]. One of the most notable applications is AI-assisted question generation [53]-[54], which allows systems to automatically produce quiz items tailored to specific learning objectives, difficulty levels, and cognitive domains. Studies have shown that natural language processing (NLP) and transformer-based models like GPT can effectively generate contextually relevant and pedagogically sound multiple-choice questions (MCQs) for academic use [53]-[54]. These AI models help reduce the manual workload of educators while maintaining the quality and variability of assessment items.

In the context of digital learning platforms, AI-assisted question generation improves scalability and personalization. Administrators or teachers can generate questions on-demand, modify them as needed, and align them with localized curriculum standards. This on-demand generation also enables quick adaptation to emerging topics or changes in educational focus without the need for extensive re-authoring [52]-[54].

For the STARBOOKS Whiz Challenge, incorporating AI-assisted quiz generation in the admin panel addresses one of the key administrative pain points: content management. Given the wide range of topics and grade levels covered by the app, manually curating quiz items can be time-consuming. By integrating AI tools into the backend, administrators can generate, review, and deploy new questions more efficiently, all while maintaining content relevance and consistency with the learning outcomes of the STARBOOKS ecosystem.

## Data Syncing and Cloud Integration

Data synchronization is a crucial component of modern distributed systems, especially when apps are expected to function offline and later sync to a central server [55]-[56]. In educational platform like the STARBOOKS Whiz Challenge, syncing ensures that player progress, badges, quiz attempts, and admin updates are not isolated to a single device. This is where platforms like Supabase—an open-source backend-as-a-service built on PostgreSQL—offer significant advantages in terms of ease of setup, real-time sync, and offline data handling [57].

Supabase provides developers with real-time APIs, authentication, and row-level security, enabling smooth interaction between local app data and cloud storage [57]. With built-in PostgreSQL replication features, Supabase can handle offline-ready scenarios by integrating with local storage and syncing changes once the app reconnects. This is particularly useful for educational apps deployed across multiple kiosk units, where syncing user data and system updates across devices is essential for scalability.

In the redesigned STARBOOKS Whiz Challenge system, Supabase can serve as the centralized backend to manage synced quiz data, player progress logs, badge achievements, and admin-created content. PostgreSQL ensures robust schema handling, indexing, and performance, while Supabase's client libraries allow seamless integration with the frontend [57]-[58]. This setup allows kiosk units to operate independently offline but still participate in a unified ecosystem when online, enabling centralized analytics, leaderboard synchronization, and data-backed decision-making.

## Current System

The current version of the STARBOOKS Whiz Challenge primarily supports a solo quiz mode with badges as a form of reward. The system is designed to operate offline and is used in DOST exhibits or fairs. Below is an overview of the technologies and components currently in use:

### Technical Background

#### a. Hardware

According to the DOST website [2], there are two current options for the STARBOOKS kiosk setup offered:

- **Option 1: Touchscreen All-in-One PC**
  - This setup integrates the display, CPU, and storage into a single device—eliminating the need for a separate system unit or external server.
  - It is a compact and modern solution used in recent deployments.
  - Mouse and keyboard can be optionally connected to the device for additional control.
  - This setup simplifies the kiosk and reduces hardware complexity.



*Figure 1. Touchscreen All-in-One PC*

- **Option 2: Standard PC Setup**
  - This is a more traditional setup involving a separate desktop PC acting as a server, connected to a terminal with a monitor and input devices.
  - This option is typically used when budget or deployment environment calls for separate server and terminal hardware.



*Figure 2. Standard PC Setup*

**b. Software**

- **Backend:** Laravel (PHP)
- **Frontend:** Vue.js
- **Database:** MySQL
- All player and admin data are stored locally on the kiosk.
- No internet-based syncing or cloud integration is supported.
- Admin tasks, such as managing quiz content and tracking badge progress, are performed directly on the kiosk.

**c. Network**

- The current STARBOOKS Whiz Challenge system is designed to function entirely **offline**, both for players and admins.
- This setup is ideal for deployment in remote areas or locations with limited or no internet connectivity, which aligns with DOST-STII's goal of promoting science and technology access in underserved communities.

**List of Processes**

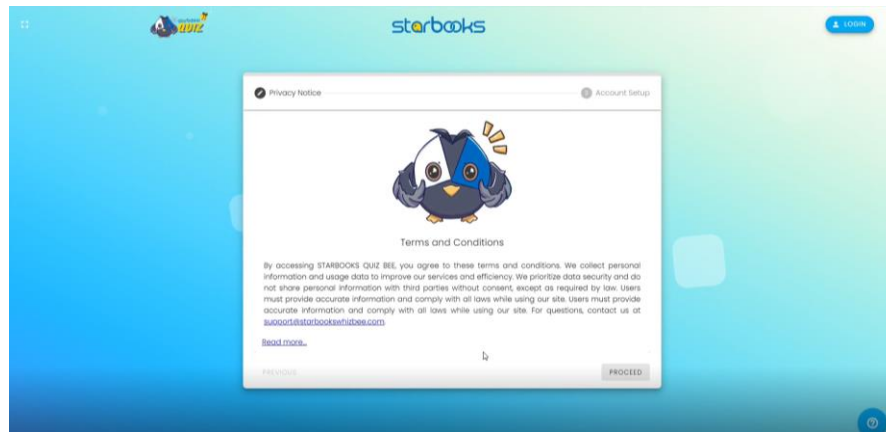
*Table 1. Current System Process*

PROCESS ID	PROCESS NAME	PROCESS DETAILS	DOER
P001	Player Registration	1. Player inputs username, password, avatar, player category, sex, and location (region, province, city).	Player
P002	Player Login	1. Player enters username and password to access the system.	Player
P003	Edit Player Profile	1. Player updates avatar, password, location, or other personal info via profile settings.	Player
P004	Select Difficulty Level	1. Player chooses a quiz difficulty (Easy, Average, Difficult) before starting the game.	Player
P005	Play Solo Quiz	1. System loads quiz questions based on selected level. 2. Player answers questions. 3. Score is computed.	Player & System

P006	Badge Progress Tracking	1. System awards badges when the player answers all questions correctly. 2. Player tracks badges earned.	System & Player
P007	Claim Rewards	1. When a player earns 3 badges in one level, they can claim a reward.	Player
P008	Player Logout	1. Player clicks Logout button to exit the system.	Player
P009	Admin Login	1. Admin enters credentials and is redirected to the dashboard.	Admin
P010	Manage Participant List	1. View/manage player profiles.	Admin
P011	Manage Admin List	1. View/manage admin profiles.	Admin
P012	Manage Badges/Rewards	1. View badge progress per player. 2. Mark rewards as claimed or distributed.	Admin
P013	Manage Quiz Questions	1. View/manage quiz questions.	Admin
P014	Manage Difficulty Levels	1. Set the number of questions. 2. Set the score points. 3. Adjust the timer.	Admin
P015	Admin Logout	1. Admin clicks Logout button to exit the system.	Admin

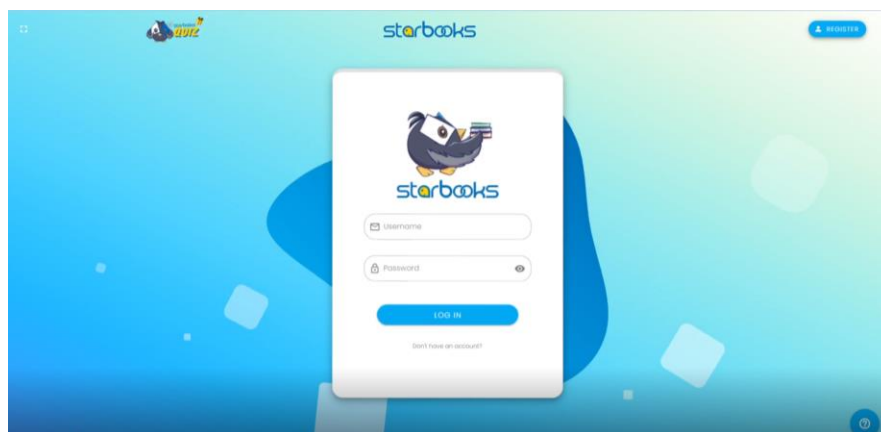
Table 1 contains a list of processes currently supported by the STARBOOKS system, covering both player and admin workflows. The table shows how players register, interact with the quiz system, and earn badges, while admins manage player and admin profiles, quiz content, and rewards from the backend.

The following figures illustrate selected interfaces from the current STARBOOKS Whiz Challenge system:



*Figure 3. Registration Page: Terms and Conditions*

Figure 3 shows the Terms and Conditions screen that appears during the registration process. Players must read and accept the terms before creating an account, ensuring they understand the rules, data usage policies, and responsibilities while using the STARBOOKS Whiz Challenge platform.



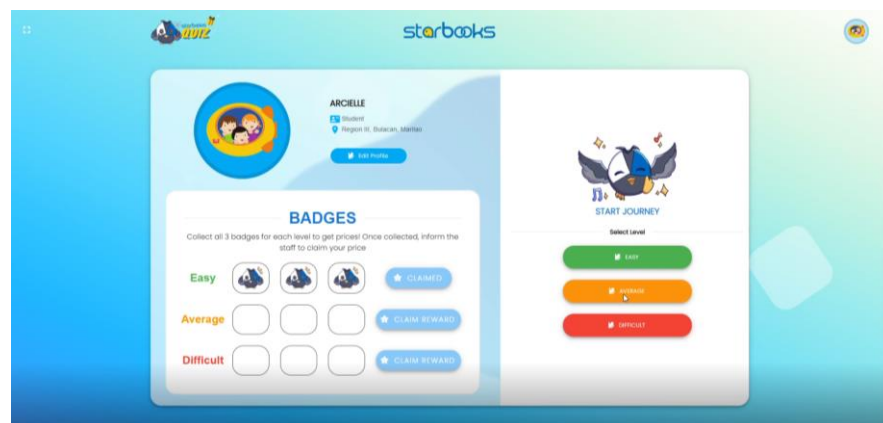
*Figure 4. Log in Page*

Figure 4 shows the player login screen where users enter their username and password to access the STARBOOKS Whiz Challenge. This serves as the entry point to the system, allowing personalized access to solo quiz and badge tracking.



*Figure 5. Manage Profile*

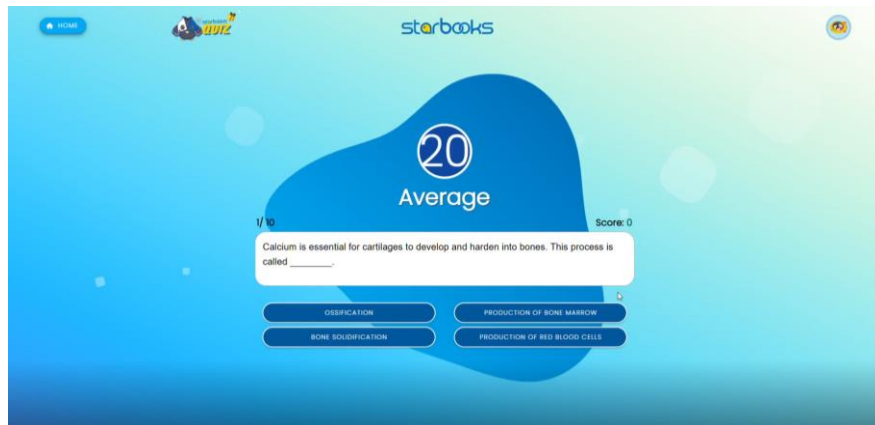
Figure 5 displays the Manage Profile screen, where users can update the personal information they provided during registration. This section also allows users to change their password.



*Figure 6. Homepage*

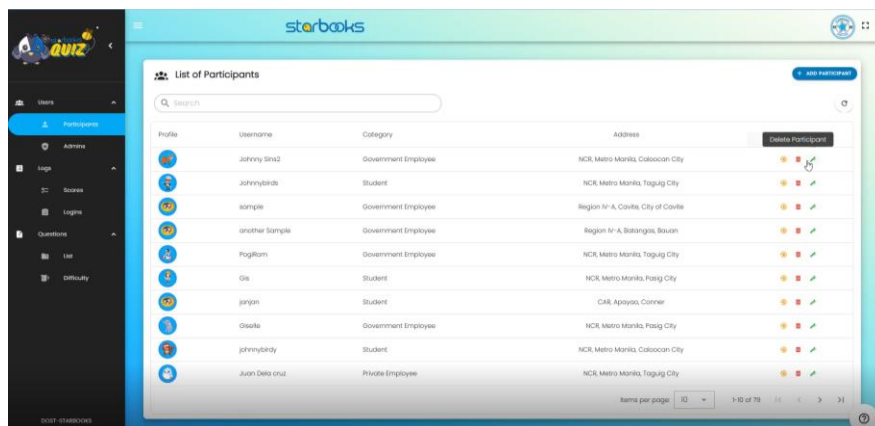
Figure 6 presents the homepage of the current STARBOOKS system, highlighting the badge collection for each difficulty level, along with options for players to select their preferred difficulty.





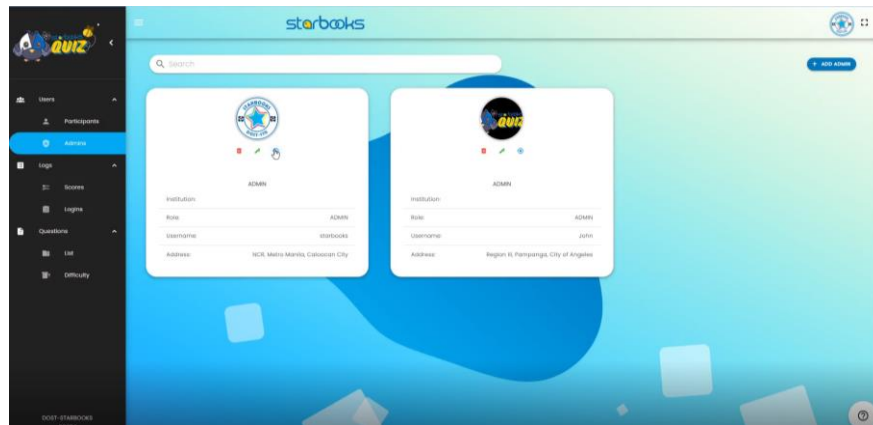
*Figure 7. Question Interface*

Figure 7 shows a sample question from the Average difficulty level of the quiz. It includes the question prompt, four answer options, a timer, and the player's current score.



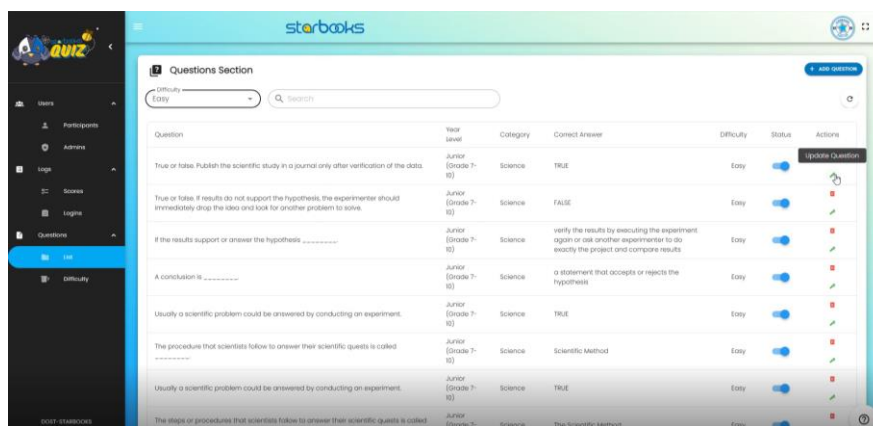
*Figure 8. Participant List Menu*

Figure 8 displays the Participant List menu, where admins can view player information, change passwords, delete accounts, and add new participants.



*Figure 9. Admin List Menu*

Figure 9 shows the Admin List menu, where existing admin accounts are displayed and can be managed by authorized users.



*Figure 10. Question List Menu*

Figure 10 shows the Question Menu, where admins can view, add, edit, or delete quiz questions.

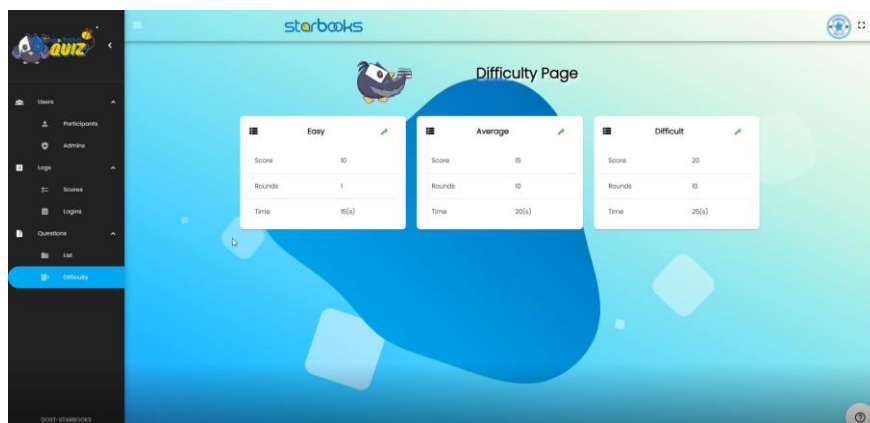


Figure 11. Difficulty Settings Menu

Figure 11 shows the Difficulty Settings Menu, where admins can set score points, number of questions, and time limit for each difficulty level.

## SWOT Analysis

Table 2. SWOT Analysis

STRENGTHS	WEAKNESSES
<ul style="list-style-type: none"> <li>Offline capability supports learning in low-connectivity areas</li> <li>Gamified features like badges and levels boost student engagement</li> <li>Simple and straightforward solo quiz format is easy to use</li> <li>Built on the trusted STARBOOKS platform familiar to users</li> </ul>	<ul style="list-style-type: none"> <li>Limited to solo quiz mode, lacking multiplayer or diverse gameplay</li> <li>Hardware dependency may restrict scalability in some remote areas</li> <li>Offline data synchronization issues if connectivity is intermittent</li> <li>Content updates are restricted without internet access</li> <li>User authentication is simplified, which may reduce security</li> </ul>
OPPORTUNITIES	THREATS
<ul style="list-style-type: none"> <li>Expand reach to underserved rural and remote communities</li> <li>Add more engaging game modes and adaptive learning features</li> <li>Partner with schools and libraries to increase adoption</li> <li>Develop mobile or hybrid versions to complement kiosk use</li> <li>Incorporate analytics to help educators track learning outcomes</li> </ul>	<ul style="list-style-type: none"> <li>Competition from online platforms offering richer multiplayer features</li> <li>Rapid tech changes could render kiosk hardware obsolete</li> <li>Funding constraints for hardware deployment and ongoing maintenance</li> <li>User engagement risk if content remains static or outdated</li> <li>Possible resistance from institutions favoring fully online solutions</li> </ul>

The SWOT analysis of the current STARBOOKS Whiz Challenge system in Table 2 highlights several key strengths, including its offline capability that ensures accessibility in areas with limited internet, engaging gamified features such as badges and levels that motivate learners, and a simple, easy-to-use solo quiz format built on the trusted STARBOOKS platform. However, the system faces weaknesses such as being limited to solo gameplay without multiplayer options, hardware dependency that may restrict deployment in some remote areas, and challenges with content updates and data synchronization due to intermittent connectivity. Opportunities exist to expand the system's reach to underserved communities, introduce more engaging game modes and adaptive learning features, and collaborate with educational institutions for wider adoption. Nonetheless, threats include competition from online platforms with richer multiplayer features, rapid technological changes that could make kiosk hardware obsolete, funding constraints for deployment and maintenance, and the risk of decreased user engagement if content is not regularly refreshed. Recognizing these factors is essential for guiding future improvements and deployment strategies to enhance the system's effectiveness and sustainability.

## Proposed Solution

This section outlines the technologies and components that will be used to develop and implement the enhanced version of the STARBOOKS Whiz Challenge. It includes the hardware setup, software stack, network considerations, and the peopleware involved.

### Technical Background

#### a. Hardware

The proposed system will remain compatible with the existing kiosk-based setups provided by DOST-STII.

- **Option 1: Touchscreen All-in-One PC**
  - Combines monitor, processor, and storage in one unit
  - Touchscreen interaction for intuitive gameplay
  - Optional keyboard and mouse
  - Ideal for exhibitions and learning centers with limited space
- **Option 2: Standard Desktop Setup**
  - CPU tower with separate monitor
  - USB keyboard and mouse
  - Cost-effective for stakeholders with limited budget

#### b. Software

The following technology stack was finalized in consultation with DOST-STII and the project adviser.

- **Backend: Django (Python)**
  - Manages authentication, game logic, player matchmaking, admin functions, and data processing.

- Widely used in analytics, AI, and backend-heavy apps.
- **Frontend: Flutter**
  - Cross-platform UI toolkit for gamified touchscreen interfaces and fluid animations.
- **Database: Supabase (PostgreSQL)**
  - Handles structured storage of users, quiz content, badges, and rewards.
  - Enables real-time matchmaking and quiz synchronization during 1v1 Whiz Battle.
  - Enables data syncing and cloud backup when an internet connection is available.

#### c. Network

- **Offline-First Design:**
  - All core features including solo quiz, memory game, puzzle, badges, and player stats work fully without internet.
  - Internet is only required for real-time 1v1 Whiz Battle and for AI-assisted question generation in the admin panel.
  - Player and admin data are stored locally on each kiosk. When internet becomes available, changes sync automatically to the cloud, allowing backup and cross-device access.

#### d. Peopleware

- **Student Developers:**
  - Develop all core features of the application, including the four game modes, reward system, player stats, and admin dashboard.
  - Design the user interface and ensure usability, accessibility, and engaging game mechanics.
  - Implement backend logic, offline data handling, and online functionality for Whiz Battle.
  - Conduct testing, debugging, and refinement to ensure the system runs smoothly on kiosk hardware.
  - Create system documentation and user manuals for administrators.
- **DOST-STII:**
  - Provide the core requirements, guidelines, and quiz content.
  - Review prototypes, provide feedback, and approve system features.
  - Coordinate where and how the system will be deployed.
  - Maintain long-term ownership and integration of the system into official STARBOOKS kiosks.
- **End Users – Exhibit Visitors and Students:**
  - Use the system to play games and engage with Math and Science content.
  - Track personal stats, unlock badges, and participate in quiz battles.
  - Provide indirect feedback through engagement and usage patterns.
- **Admin Users – Teachers, Librarians, or Exhibit Facilitators:**

- Access the built-in admin panel to manage player and admin lists, quiz questions, process reward claims, and view game analytics.
- Moderate gameplay during exhibits or in school-based learning environments.
- Assist users who need help with the system and ensure proper use of the kiosk.

## Feasibility

### Operational Feasibility

The proposed STARBOOKS Whiz Challenge system is considered operationally feasible based on the following factors:

- **User Familiarity:** Students are already accustomed to using interactive educational games and touchscreen interfaces, ensuring a smooth and intuitive user experience.
- **Deployment Readiness:** The system is designed for deployment within existing STARBOOKS kiosks and learning centers, requiring minimal adjustments to current operational workflows.
- **Admin Simplicity:** Administrative tools are integrated into the same interface, enabling teachers, librarians, and exhibit staff to manage content and monitor usage without needing advanced technical skills.
- **Offline-First Design:** The offline-first architecture ensures the system remains fully functional even in remote areas with limited or no internet access, supporting DOST's outreach goals for underserved communities.
- **Engagement-Oriented Design:** The gamified approach encourages regular student interaction, increasing both usage and educational impact.

Overall, the system is well aligned with the operational model of STARBOOKS and enhances its mission of delivering engaging science and math learning through interactive media.

### Economic Feasibility

The proposed STARBOOKS Whiz Challenge system is economically feasible, particularly given its student-led development and reliance on existing STARBOOKS infrastructure:

- **Minimal Cost to DOST-STII:** As an academic initiative, the project incurs no development labor costs for DOST-STII.
- **Utilizes Existing Hardware:** The system is designed to run seamlessly on current STARBOOKS kiosks—whether All-in-One units or standard PC setups—eliminating the need for new hardware investments.
- **Low Maintenance Requirements:** Maintenance is limited to occasional software updates and content revisions.

- **Scalability:** Once validated, the system can be easily replicated and deployed across additional STARBOOKS locations with minimal additional cost.

The project delivers high educational value and wide accessibility with minimal financial burden, making it a cost-effective solution for enhancing STARBOOKS engagement and outreach.

## Technical Feasibility

The proposed STARBOOKS Whiz Challenge system is technically feasible, considering the chosen technologies, the development team's skills, and hardware compatibility:

- **Proven Technologies:** The system will be developed using Flutter (for frontend), Django (for backend), and Supabase (for database, real-time multiplayer, and syncing). These technologies were selected based on their strong support for offline-first applications, smooth cross-platform performance, and real-time capabilities. They are also well-documented, developer-friendly, and aligned with modern best practices.
- **Offline-First with Smart Cloud Syncing:** The architecture supports local data storage to ensure smooth gameplay even without an internet connection. When the kiosk connects online, Supabase handles automatic syncing of data like player progress and admin updates. This approach keeps the system usable anywhere while still benefiting from cloud connectivity when available.
- **Hardware Compatibility:** The app is designed to run smoothly on existing STARBOOKS kiosks, whether All-in-One units or standard PCs, with no additional hardware upgrades required.
- **Developer Readiness:** The student development team possesses foundational skills in frontend, backend, and database development. While this is their first time building an offline-first system, the team is actively learning with guidance from academic advisers and DOST technical consultants.

The tools and architecture are practical, accessible, and aligned with the capabilities of the development team and the infrastructure provided by DOST, ensuring a technically sound solution.

## Schedule Feasibility

The development timeline for the STARBOOKS Whiz Challenge system is feasible within the academic period designated for the project:

- **Well-Defined Milestones:** The project follows the Software Development Life Cycle (SDLC), with clearly planned phases including requirements analysis, UI/UX design, core development, testing, deployment, and evaluation.
- **Manageable MVP Scope:** The first release focuses on four core game modes and essential features, ensuring the workload remains realistic and achievable within the set deadlines.
- **Ongoing Coordination with DOST:** Regular check-ins and feedback sessions with DOST-STII help validate progress early, enabling faster iteration and minimizing potential delays.

- **Team Role Distribution:** The student team is organized with assigned roles for project management, development, and QA, promoting efficient task execution and time management.

With a focused scope, defined development phases, and consistent collaboration, the project is on track to be delivered on time with all core features implemented and tested.

## Requirements Analysis

### Project Vision

*Table 3. Project Vision*

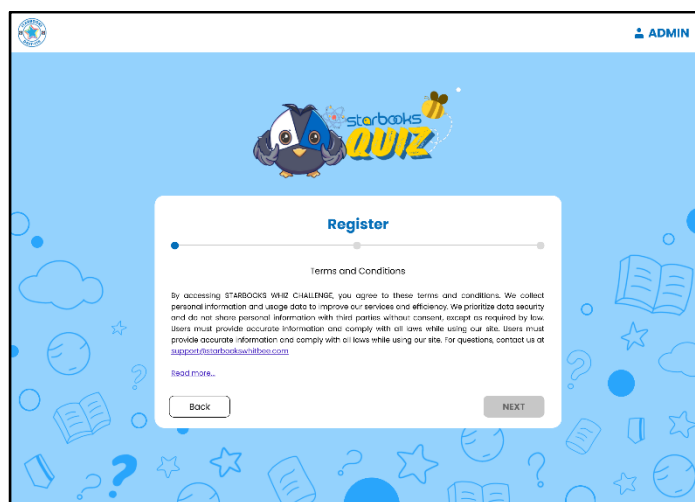
VISION	To modernize the current STARBOOKS with gamified, engaging Science and Math mini games that enhance learning and spark student interest in STEM.		
TARGET GROUP	NEEDS	PRODUCT	BUSINESS GOALS
<ul style="list-style-type: none"> <li>- Walk-in visitors and participants of DOST exhibits and events</li> <li>- Students (Grades 1 to 12) in schools and/or learning centers</li> <li>- Teachers, librarians, exhibit facilitators</li> <li>- DOST-STII stakeholders</li> </ul>	<ul style="list-style-type: none"> <li>- Existing quiz system lacks engagement and variety</li> <li>- Students need more interactive, motivating ways to learn</li> <li>- Admins need better tracking and management</li> </ul>	<ul style="list-style-type: none"> <li>- An interactive educational game app with 4 game modes: solo quiz, memory match, puzzle, and 1v1 quiz battle</li> <li>- Works offline except for the 1v1 Battle mode which requires internet</li> <li>- Sync automatically once internet is available</li> <li>- Includes badge rewards, player stats, and better admin tools for content management</li> </ul>	<ul style="list-style-type: none"> <li>- Reinforce DOST-STII's mission to promote STEM education across the country</li> <li>- Boost engagement with STARBOOKS through modern, game-based content</li> <li>- Equip DOST exhibits and learning centers with a scalable, offline-first educational solution</li> </ul>

Table 3 presents the project vision, outlining the intended purpose, target users, and desired impact of the STARBOOKS Whiz Challenge. It serves as a guiding statement for the development and implementation of the system.



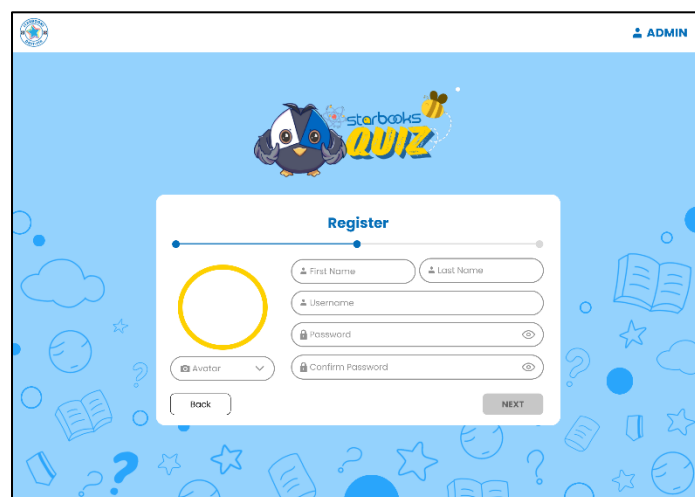
## Prototype (Mock Flow / Wireframe)

The following figures showcase selected key interface screens from the proposed STARBOOKS Whiz Challenge application. These are high-fidelity mockups created to illustrate the core functionalities and user experience of the system. Please note that these designs are not yet final and may still change based on further feedback, development adjustments, or usability testing. They serve as visual representations of the intended features and layout.



*Figure 12. Player Registration: Terms and Conditions*

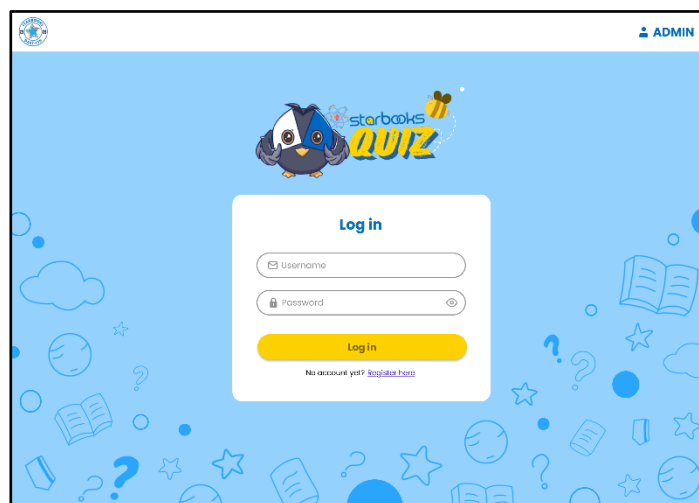
Figure 12 shows the Terms and Conditions screen that appears during the registration process. Players must read and accept the terms before creating an account, ensuring they understand the rules, data usage policies, and responsibilities while using the STARBOOKS Whiz Challenge platform.



*Figure 13. Player Registration: Account Setup*

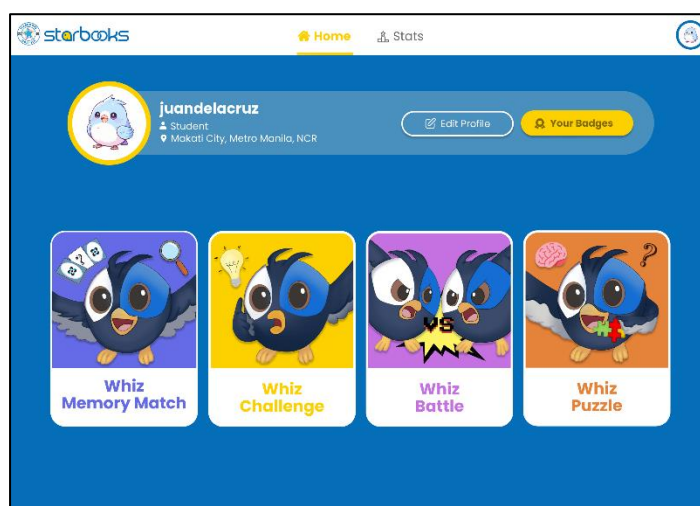
Figure 13 shows the account setup screen where players input their personal details such as avatar, first name, last name, username, password, and other necessary information. This step

finalizes the registration process and sets up the player's profile for a personalized experience in the STARBOOKS Whiz Challenge.



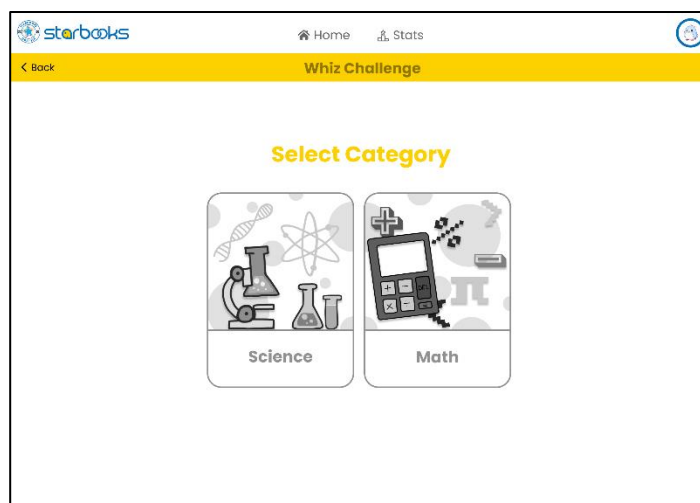
*Figure 14. Player Log in*

Figure 14 shows the player login screen where users enter their username and password to access the STARBOOKS Whiz Challenge. This serves as the entry point to the system, allowing personalized access to game modes, stats, and rewards.



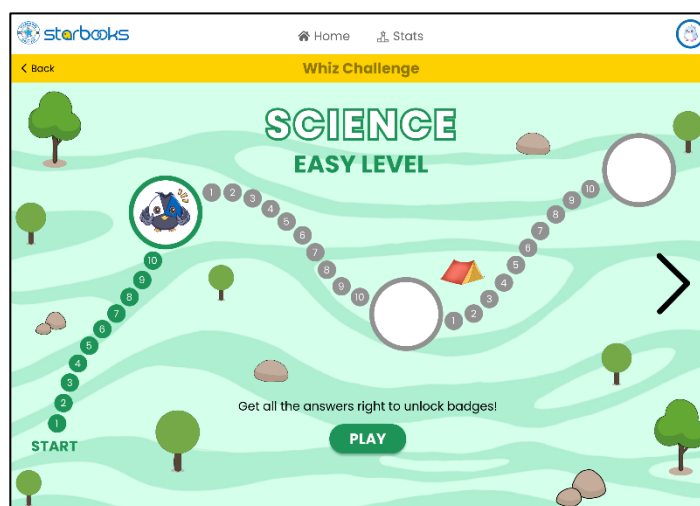
*Figure 15. STARBOOKS Whiz Challenge Homepage*

Figure 15 shows the homepage interface where players can choose from the four available game modes—Whiz Challenge, Whiz Memory Game, Whiz Puzzle, and Whiz Battle. This screen serves as the central hub for navigating the STARBOOKS Whiz Challenge gameplay experience.



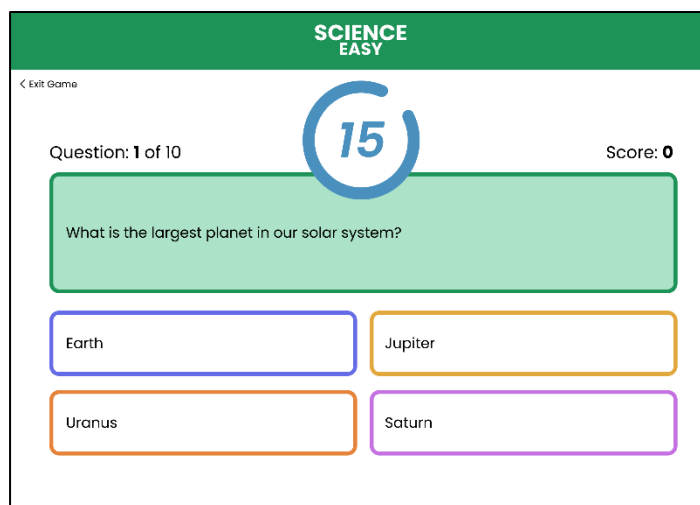
*Figure 16. Whiz Challenge: Select Category*

Figure 16 shows the category selection screen for the Whiz Challenge mode, where players choose between Science or Mathematics. This step tailors the quiz content based on the selected subject area, ensuring focused and relevant gameplay.



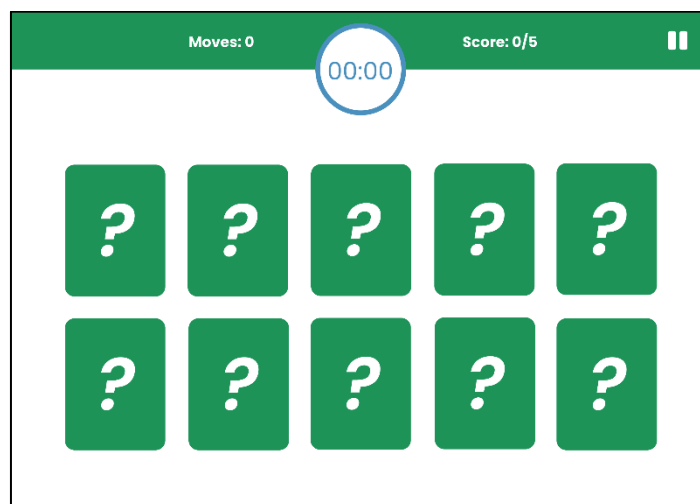
*Figure 17. Whiz Challenge: Science Easy Level*

Figure 17 shows the Science Easy level interface in Whiz Challenge mode, featuring a map-like path layout that visually represents quiz progression. Players can tap the play button to begin the solo quiz for this level.



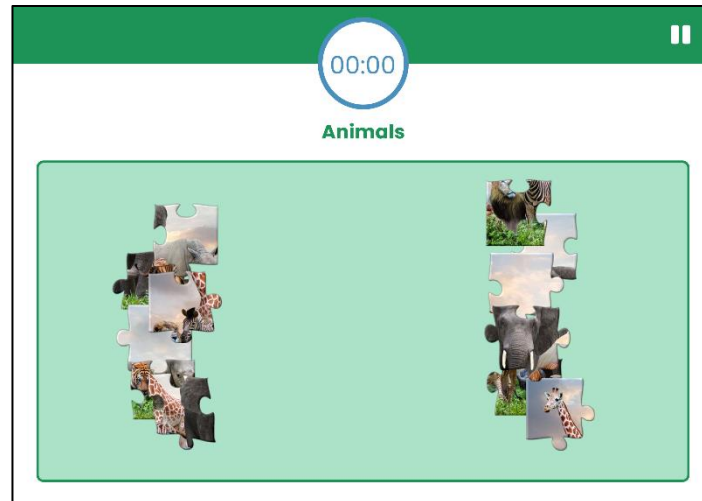
*Figure 18. Whiz Challenge: Science Easy Level Question*

Figure 18 shows a sample question screen from the Science Easy level in Whiz Challenge mode. It displays the quiz question along with four multiple-choice options and a countdown timer, encouraging quick thinking and engagement during gameplay.



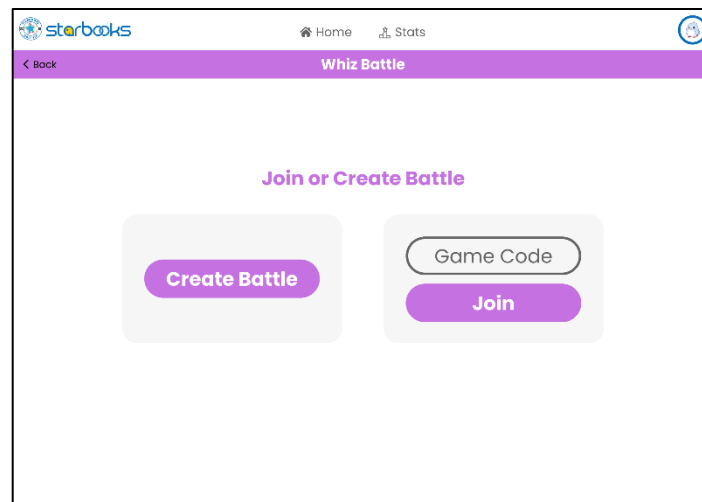
*Figure 19. Whiz Memory Match: Easy Level*

Figure 19 shows the Easy level screen of the Whiz Memory Match game mode, displaying 10 face-down cards arranged for matching. The interface includes a timer that records the time taken to complete the game, along with score and move counters to track the player's performance.



*Figure 20. Whiz Puzzle: Easy Level*

Figure 20 shows the Easy level of the Whiz Puzzle game mode, featuring scrambled jigsaw puzzle pieces spread across the screen. The puzzle belongs to the animals category, and a timer records the total time taken by the player to finish the puzzle.



*Figure 21. Whiz Battle: Join or Create Battle*

Figure 21 shows the Whiz Battle mode screen where players can either join an existing battle or create a new one. This interface facilitates real-time 1v1 quiz battles, allowing players to compete against other users.

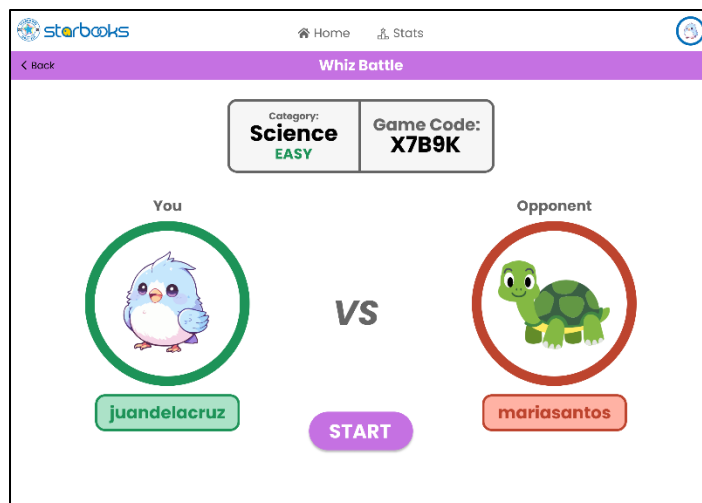


Figure 22. Whiz Battle: Waiting Room

Figure 22 shows the battle waiting room where the system generates a unique game code for the player. The screen displays the selected category and difficulty level while waiting for an opponent to join the 1v1 quiz battle.

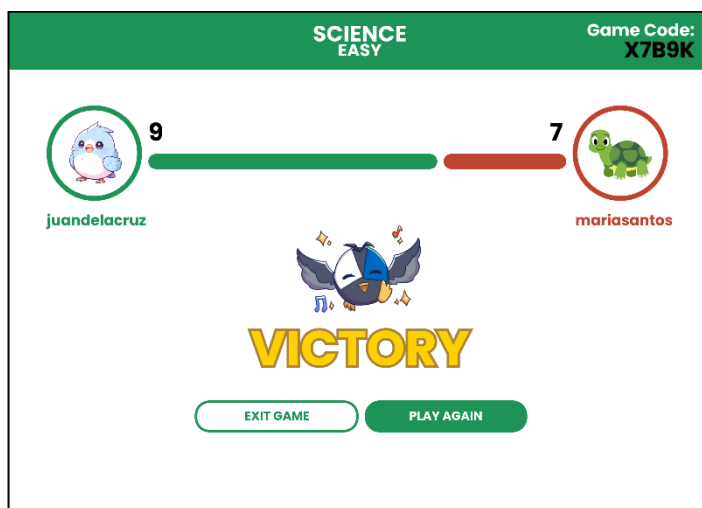


Figure 23. Whiz Battle: Battle Result

Figure 23 shows the results screen displayed at the end of a Whiz Battle match. It presents the player's victory or defeat status along with the final scores, summarizing the outcome of the 1v1 quiz battle.

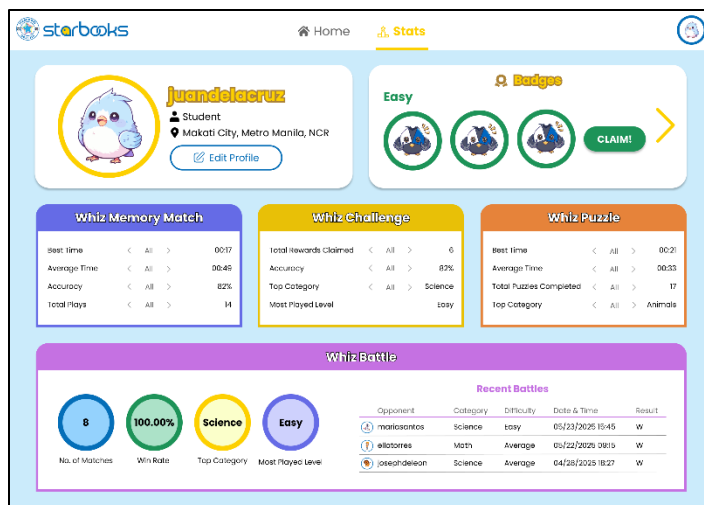


Figure 24. Player Stats

Figure 24 shows the Player Stats screen where users can view their overall game performance, including best time, top category, accuracy, most played level, battle history, badges earned, and progress across different game modes. This helps players track their growth and achievements in using the platform.

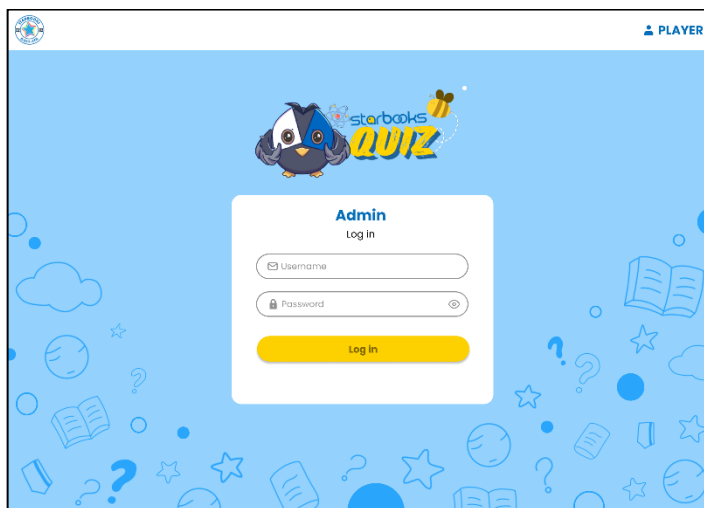


Figure 25. Admin Dashboard Log in

Figure 25 shows the admin login screen where authorized personnel enter their credentials to access the STARBOOKS Whiz Challenge admin dashboard. This secure entry point restricts access to administrative functions such as content management and game analytics.



Figure 26. Admin Dashboard: Analytics Dashboard

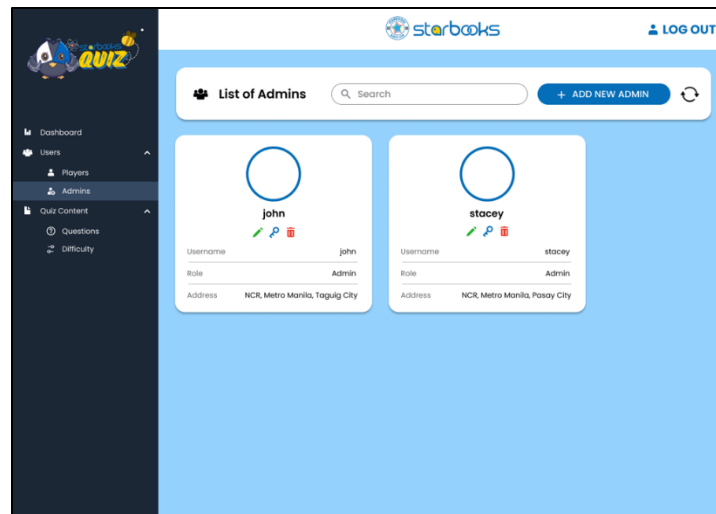
Figure 26 shows the Analytics Dashboard within the admin panel, where administrators can view data visualizations such as player trends, most played game mode, most played level, reward claims summary, and other key metrics to monitor user engagement.

Username	Category	Address	Actions	Badges
ronald	Student	NCR, Metro Manila, Caloocan City	<a href="#">Edit</a> <a href="#">Delete</a> <a href="#">View</a>	
carla	Student	NCR, Metro Manila, Makati City	<a href="#">Edit</a> <a href="#">Delete</a> <a href="#">View</a>	
clarisse	Student	NCR, Metro Manila, Taguig City	<a href="#">Edit</a> <a href="#">Delete</a> <a href="#">View</a>	
robert	Employee	NCR, Metro Manila, Paranaque City	<a href="#">Edit</a> <a href="#">Delete</a> <a href="#">View</a>	
jerome	Student	NCR, Metro Manila, Caloocan City	<a href="#">Edit</a> <a href="#">Delete</a> <a href="#">View</a>	
ariel	Student	NCR, Metro Manila, Pasay City	<a href="#">Edit</a> <a href="#">Delete</a> <a href="#">View</a>	
hannah	Student	NCR, Metro Manila, Las Pinas City	<a href="#">Edit</a> <a href="#">Delete</a> <a href="#">View</a>	
rico	Student	NCR, Metro Manila, San Juan City	<a href="#">Edit</a> <a href="#">Delete</a> <a href="#">View</a>	
marie	Employee	NCR, Metro Manila, Quezon City	<a href="#">Edit</a> <a href="#">Delete</a> <a href="#">View</a>	

Figure 27. Admin Dashboard: List of Players

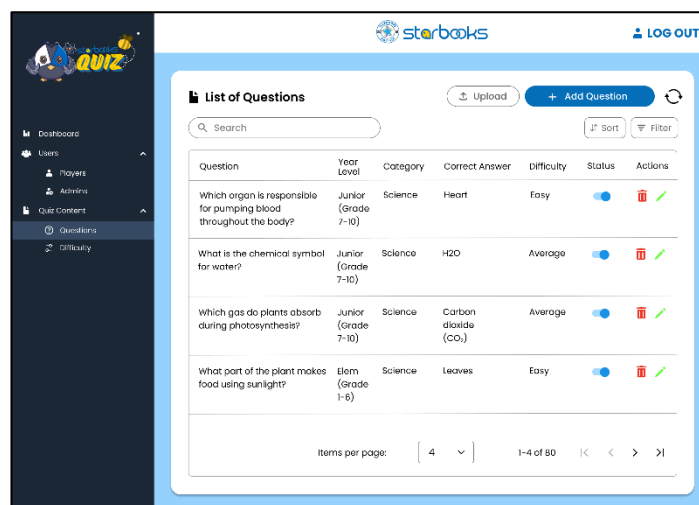
Figure 27 shows the List of Players screen in the admin dashboard, where administrators can view and manage registered player accounts.





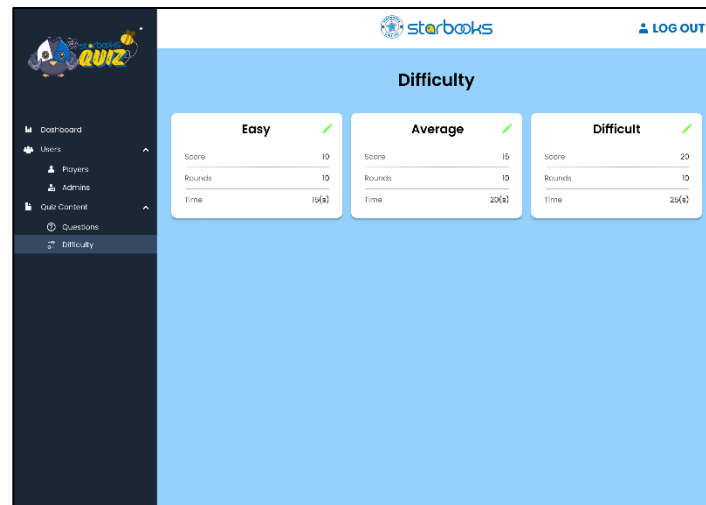
*Figure 28. Admin Dashboard: List of Admins*

Figure 28 shows the List of Admins screen within the admin dashboard, where authorized personnel can view and manage admin accounts.



*Figure 29. Admin Dashboard: List of Questions*

Figure 29 shows the List of Questions screen in the admin dashboard, where administrators can view, add, edit, or delete quiz questions.



*Figure 30. Admin Dashboard: Difficulty Settings*

Figure 30 shows the Difficulty Settings screen in the admin dashboard, where administrators can configure and adjust the difficulty levels for quiz questions to ensure balanced gameplay and appropriate challenge for players.

## Project Lean Canvas

Table 4. Project Lean Canvas

PROBLEM	SOLUTION	UNIQUE VALUE PROPOSITION	UNFAIR ADVANTAGE	CUSTOMER SEGMENTS
<ul style="list-style-type: none"> <li>- Existing STARBOOKS quiz is limited to a single game mode</li> <li>- No competitive gameplay feature</li> <li>- Lacks engaging, gamified user experience</li> <li>- No tracking of player stats</li> <li>- Lack of better admin dashboard management and analytics</li> </ul>	<ul style="list-style-type: none"> <li>- 4 interactive game modes (Solo quiz, Memory, Puzzle, 1v1 Battle)</li> <li>- Player stats to track performance</li> <li>- Works offline except 1v1 battle that needs internet for game code matchmaking</li> <li>- Sync automatically once internet is available</li> <li>- Upgraded admin dashboard with analytics</li> </ul>	<p>A fun, offline-first, gamified educational platform built specifically for schools and DOST exhibits—tailored for Filipino students and accessible even in low-connectivity areas</p>	<ul style="list-style-type: none"> <li>- Purpose-built for DOST STARBOOKS kiosks</li> <li>- Can operate both offline and online, ideal for rural deployment</li> <li>- Direct partnership with DOST</li> </ul>	<ul style="list-style-type: none"> <li>- Walk-in visitors at DOST exhibits, fairs, and events</li> <li>- Grades 1 to 12 students, particularly in schools/learning centers with limited digital access</li> <li>- Educators and LGUs supporting STARBOOKS</li> </ul>
EXISTING ALTERNATIVES	KEY METRICS	HIGH-LEVEL CONCEPTS	CHANNELS	EARLY ADOPTERS
<ul style="list-style-type: none"> <li>- Current STARBOOKS system</li> <li>- Online quiz apps or educational platforms like Quizizz, Kahoot, and Koobits</li> <li>- Mind games/puzzles</li> <li>- Printed learning materials like worksheets or textbooks</li> </ul>	<ul style="list-style-type: none"> <li>- Number of plays per game mode</li> <li>- Badge unlock count</li> <li>- Player progress and login stats</li> <li>- Admin usage and content updates</li> </ul>	<ul style="list-style-type: none"> <li>- Quizizz-style, but for offline kiosks</li> <li>- Kahoot-style learning, but built into STARBOOKS kiosks</li> <li>- Koobits-like educational games, but designed for offline use in remote areas</li> <li>- A gamified learning arcade, powered by STARBOOKS</li> </ul>	<ul style="list-style-type: none"> <li>- STARBOOKS kiosks</li> <li>- DOST exhibits, fairs, and outreach programs</li> <li>- Distributed through schools, learning centers, and local government units</li> </ul>	<ul style="list-style-type: none"> <li>- Visitors attending DOST exhibits</li> <li>- Students and educators in schools with STARBOOKS kiosks</li> <li>- LGUs deploying STARBOOKS in community centers</li> </ul>

		- Duolingo-style badge and reward system, but fully offline		
<b>COST STRUCTURE</b>		<b>REVENUE STRUCTURE</b>		
<ul style="list-style-type: none"> <li>- Development Time and Training</li> <li>- Software Development</li> <li>- Maintenance and Bug Fixing</li> <li>- Stakeholder Coordination</li> </ul>		<ul style="list-style-type: none"> <li>- No direct revenue; non-commercial academic project</li> <li>- Value measured through educational impact</li> <li>- Future funding possible via partnerships</li> </ul>		

Table 4 outlines the key elements of the STARBOOKS Whiz Challenge project using the Lean Canvas framework, including the problem, target users, unique value proposition, key metrics, and channels, to provide a clear snapshot of the project's strategic direction.

## User Classes and Characteristics

*Table 5. User Classes and Characteristics*

ROLES	DESCRIPTION
Player	A student or visitor who registers in the system to play games. Players can select difficulty levels, earn badges, and claim rewards.
Admin	A teacher, librarian, or exhibit staff who manages content, monitors player activity, and handles quiz questions, badges, and rewards.

Table 5 shows the different user roles within the STARBOOKS Whiz Challenge system. Each user class has specific responsibilities and access privileges tailored to support both gameplay and system administration.

## Product Backlog

Table 6. Product Backlog

ID	As a...	I want to be able to...	So that...	Priority	Sprint	Status
1	New player	Register for an account	I can access the system	High	1	To be started
2	Player	Log in to the system	I can access the system	High	1	To be started
3	Player	Edit my profile	I can update my personal info	Medium	1	To be started
4	Player	Play Whiz Challenge	I can answer quiz questions and earn badges	High	1	To be started
5	Player	View and claim Rewards	I can claim my earned badges	High	1	To be started
6	Player	Log out of the system	I can safely end my session	High	1	To be started
7	Player	Play Whiz Memory Match	I can match images and improve my memory	Medium	2	To be started
8	Player	Play Whiz Puzzle	I can solve jigsaw puzzles for fun learning	Medium	2	To be started
9	Player	Play Whiz Battle	I can compete with other players in a 1v1 quiz match	High	2	To be started
10	Player	View my stats	I can track my progress and performance	Medium	2	To be started
11	Admin	Log in to the admin dashboard	I can manage users, quiz content, rewards, and view analytics	High	3	To be started
12	Admin	Manage player accounts	I can view, edit, or delete player data	High	3	To be started
13	Admin	Manage admin accounts	I can view, edit, or delete admin data	Medium	3	To be started
14	Admin	Manage quiz content	I can create and update questions	High	3	To be started
15	Admin	Manage difficulty levels	I can set number of questions,	Medium	3	To be started

			timer, scores for each difficulty level			
16	Admin	Manage reward items	I can configure what players can claim	Medium	3	To be started
17	Admin	View analytics	I can monitor usage trends and improve content	High	3	To be started
18	Admin	Log out of the admin dashboard	I can secure admin access after use	High	3	To be started

Table 6 presents the product backlog for the proposed STARBOOKS Whiz Challenge system. It lists key user stories from both player and admin perspectives, outlining their goals, priorities, and planned sprint assignments. All tasks are currently marked as "To be started," reflecting the initial phase of the project.

## Product Roadmap

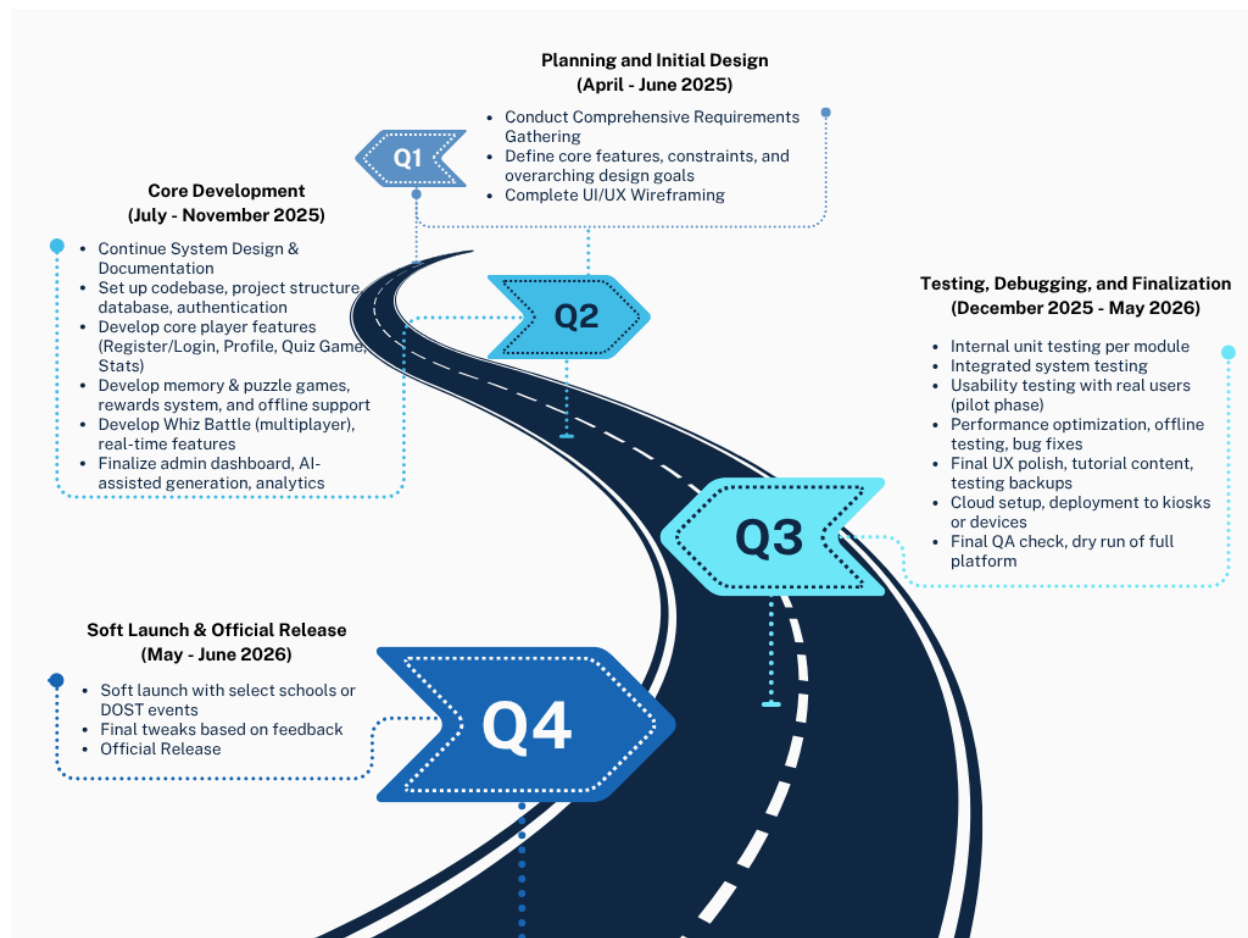


Figure 31. Product Roadmap

Figure 31 shows the product roadmap for the proposed STARBOOKS Whiz Challenge system. It outlines the key phases and major milestones from initial planning through to the official launch of v1.0 by June 2026. This roadmap is crucial for communicating our priorities for building the offline-first game modes, player progression system, and administrative tools, ensuring that all efforts are aligned towards delivering a functional and engaging Minimum Viable Product for DOST exhibits, schools, and learning centers.

## Release Plan

*Table 7. Release Plan*

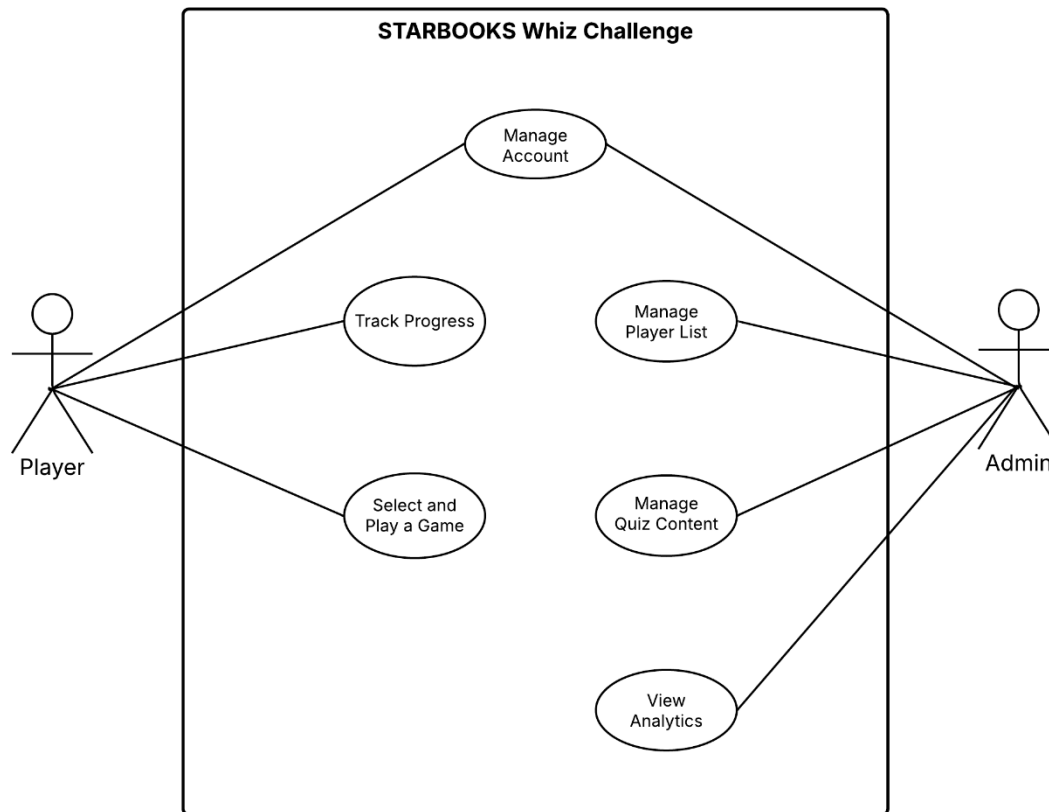
	Frontend	Backend	QA	Release
<b>May 2025</b>	UI/UX Wireframing (core pages, flows)			
<b>June 2025</b>	Refine mockups, prepare design system	Finalize requirements, database schema planning		
<b>July 2025</b>	Start development of Register/Login UI, Player Profile, Stats	Set up codebase, authentication, user management, core game logic		
<b>August 2025</b>	Continue development: Quiz Game UI, Game Modes	Implement reward system, game scoring backend		
<b>September 2025</b>	Puzzle/Memory game screens, battle mode UI	Whiz Battle multiplayer logic, offline mode backend		
<b>October 2025</b>	Admin dashboard UI, analytics views, text-to-speech input	Finalize admin tools: AI-assisted question generation, analytics tracking		
<b>November 2025</b>	UI polish, dynamic components, accessibility improvements	Final integrations and backend logic for all core features		
<b>December 2025</b>			Internal Unit Testing (module by module)	
<b>January 2026</b>			Integrated system testing, usability testing with pilot users	
<b>February 2026</b>			Performance optimization, offline testing, bug fixes	



<b>March 2026</b>			Finalize real-time sync logic, handle kiosk/cloud sync scenarios	
<b>April 2026</b>			Final UI polish, add tutorial content, test recovery/backup scenarios	
<b>May 2026</b>			Final QA Check	Soft Launch
<b>June 2026</b>			Monitor & support post-launch	Official Release

Table 7 outlines the release plan for the STARBOOKS Whiz Challenge system, detailing major development phases, and estimated timelines. This release plan serves as a guide for coordinating development efforts, stakeholder feedback, and system testing. If you want to learn more about the comprehensive software release plan, you can click this [link](#).

## Use Case Diagram



*Figure 32. Use Case Diagram*

Figure 32 presents the simplified use case diagram for the proposed STARBOOKS Whiz Challenge system. It illustrates the primary actors—Player and Admin—along with their interactions with core system functionalities. These include player-related activities such as account management, engaging in multiple game modes, and tracking progress, as well as admin functions like managing users, quiz content, rewards, and viewing analytics.

## Use Case Full Description

Table 8. Use Case: Register Account

<b>Use Case ID</b>	UC-01
<b>Use Case Name</b>	Register Account
<b>Created by</b>	Janice Maxene Salipande
<b>Date Created</b>	06-04-2025
<b>Purpose</b>	To allow new players to register and create an account for accessing the system.
<b>Requirement Traceability</b>	BR-01
<b>Priority</b>	High
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>The player has not yet registered in the system.</li> </ul>
<b>Postconditions</b>	<ul style="list-style-type: none"> <li>A new player profile has been created and saved.</li> </ul>
<b>Actors</b>	Player
<b>Include</b>	<ul style="list-style-type: none"> <li>Display Terms and Conditions</li> <li>Setup Account</li> </ul>
<b>Extends</b>	None
<b>Flow of Actions</b>	<p><b>Basic Flow:</b></p> <ol style="list-style-type: none"> <li>1. Player clicks on the "Register" button.</li> <li>2. System displays terms and conditions; player reads and accepts.</li> <li>3. Player selects an avatar.</li> <li>4. Player enters first name, last name, username, and password.</li> <li>5. Player selects a player category, sex, region, province, and city.</li> <li>6. Player clicks "Submit" to complete the registration.</li> <li>7. System saves the profile and confirms registration success.</li> </ol> <p><b>Alternative Flows</b></p> <p><b>Player Declines Terms and Conditions:</b></p> <ol style="list-style-type: none"> <li>1. The player clicks "Register here."</li> <li>2. The system displays the Terms and Conditions.</li> <li>3. The player does not click "Accept."</li> <li>4. The system cancels the registration process and returns the player to the welcome screen.</li> </ol> <p><b>Missing or Incomplete Fields:</b></p> <ol style="list-style-type: none"> <li>1. The player fills in the form and clicks "Submit."</li> <li>2. The system detects one or more empty or invalid fields (e.g., missing username, invalid password).</li> <li>3. The system displays an error message indicating the problematic fields.</li> <li>4. The player corrects the input and clicks "Submit" again.</li> </ol>

	<p>5. The system proceeds with registration if all inputs are valid.</p> <p><b><i>Username Already Exists:</i></b></p> <ol style="list-style-type: none"> <li>1. The player enters a username that already exists in the database.</li> <li>2. The system displays an error message: "Username already taken."</li> <li>3. The player edits the username to a new, unique one.</li> <li>4. The player clicks "Submit."</li> <li>5. The system proceeds with registration.</li> </ol>
--	--

Table 9. Use Case: Login

<b>Use Case ID</b>	UC-02
<b>Use Case Name</b>	Login
<b>Created by</b>	Janice Maxene Salipande
<b>Date Created</b>	06-04-2025
<b>Purpose</b>	To allow registered players and admins to access the system.
<b>Requirement Traceability</b>	BR-02, BR-11
<b>Priority</b>	High
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>• The user must be registered and have valid login credentials.</li> </ul>
<b>Postconditions</b>	<ul style="list-style-type: none"> <li>• The user is logged in and redirected to the homepage.</li> </ul>
<b>Actors</b>	Player, Admin
<b>Include</b>	None
<b>Extends</b>	<ul style="list-style-type: none"> <li>• Forgot Password</li> </ul>
<b>Flow of Actions</b>	<p><b>Basic Flow:</b></p> <ol style="list-style-type: none"> <li>1. The user opens the system.</li> <li>2. The login form is displayed asking for username and password.</li> <li>3. The user inputs their registered username and password.</li> <li>4. The system verifies the credentials against stored data.</li> <li>5. If the credentials are correct, the user is logged in.</li> <li>6. The system redirects the user to the homepage.</li> </ol> <p><b>Alternative Flows</b></p> <p><b><i>Invalid Credentials:</i></b></p> <ol style="list-style-type: none"> <li>1. User inputs an incorrect username or password.</li> <li>2. System shows an error message indicating invalid login.</li> <li>3. User is prompted to re-enter correct credentials.</li> </ol> <p><b><i>Missing or Incomplete Fields:</i></b></p> <ol style="list-style-type: none"> <li>1. User clicks the login button without filling in one or both fields.</li> <li>2. System highlights the missing fields and displays a "Please complete all fields" message.</li> </ol>

	3. User fills in the missing information and retries login.
--	---

*Table 10. Use Case: Edit Profile*

<b>Use Case ID</b>	UC-03
<b>Use Case Name</b>	Edit Profile
<b>Created by</b>	Janice Maxene Salipande
<b>Date Created</b>	06-04-2025
<b>Purpose</b>	To allow players to update their personal information and/or change their password.
<b>Requirement Traceability</b>	BR-03
<b>Priority</b>	Medium
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>• Player must be logged into the system.</li> </ul>
<b>Postconditions</b>	<ul style="list-style-type: none"> <li>• Player's updated profile information or new password is saved and reflected across the system.</li> </ul>
<b>Actors</b>	Player
<b>Include</b>	None
<b>Extends</b>	<ul style="list-style-type: none"> <li>• Change Password</li> </ul>
<b>Flow of Actions</b>	<p><b>Basic Flow:</b></p> <ol style="list-style-type: none"> <li>1. Player clicks the "Edit Profile" button located near their username on the homepage.</li> <li>2. System displays the editable profile form with pre-filled existing data.</li> <li>3. Player updates one or more fields (e.g., avatar, name, category, sex, region, province, city).</li> <li>4. Player may also click the "Change Password" option.</li> <li>5. Player clicks the "Save" button.</li> <li>6. System validates the updated information.</li> <li>7. If valid, system saves the new details to the local database.</li> <li>8. A confirmation message is displayed: "Profile updated successfully."</li> <li>9. System reflects the changes immediately on the homepage and other relevant areas.</li> </ol> <p><b>Alternative Flows</b></p> <p><b>Player Cancels Edit:</b></p> <ol style="list-style-type: none"> <li>1. Player clicks the "Edit Profile" button.</li> <li>2. After viewing the form, the player decides not to make any changes.</li> <li>3. Player clicks a "Cancel" button.</li> <li>4. System discards any unsaved changes and returns the player to the homepage.</li> </ol> <p><b>Validation Errors:</b></p>

	<ol style="list-style-type: none"> <li>1. Player clicks the “Save” button with one or more invalid or incomplete fields (e.g., missing required fields or invalid characters).</li> <li>2. System highlights the fields with errors and displays an appropriate message (e.g., “Username must be unique” or “Please select a city”).</li> <li>3. Player corrects the issues and resubmits.</li> </ol> <p><b>Save Failure:</b></p> <ol style="list-style-type: none"> <li>1. After the player submits valid inputs, a local storage failure occurs (e.g., full storage, write error).</li> <li>2. The system displays an error message: “Failed to save changes. Please try again.”</li> <li>3. Player is prompted to retry or cancel.</li> </ol>
--	--

Table 11. Use Case: Play Whiz Challenge

<b>Use Case ID</b>	UC-04
<b>Use Case Name</b>	Play Whiz Challenge
<b>Created by</b>	Janice Maxene Salipande
<b>Date Created</b>	06-04-2025
<b>Purpose</b>	To allow the player to play a solo quiz game, earn badges, and view performance stats.
<b>Requirement Traceability</b>	BR-04
<b>Priority</b>	High
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>• Player must be logged in.</li> </ul>
<b>Postconditions</b>	<ul style="list-style-type: none"> <li>• Game results (correct answers, wrong answers, average time per question) are displayed to the player.</li> <li>• Player may earn a badge if all answers are correct.</li> <li>• Player stats are updated.</li> </ul>
<b>Actors</b>	Player
<b>Include</b>	None
<b>Extends</b>	<ul style="list-style-type: none"> <li>• Earn Badge</li> </ul>
<b>Flow of Actions</b>	<p><b>Basic Flow:</b></p> <ol style="list-style-type: none"> <li>1. Player clicks the “Whiz Challenge” from the homepage.</li> <li>2. System prompts the player to select a category: Science or Math.</li> <li>3. Player selects the desired difficulty level: Easy, Average, or Difficult.</li> <li>4. Player starts the quiz game.</li> <li>5. System displays one question at a time, with answer options and a countdown timer.</li> <li>6. Player selects an answer.</li> <li>7. System provides immediate feedback: correct or incorrect.</li> <li>8. Steps 5–7 repeat until all questions are answered.</li> <li>9. System evaluates the results.</li> </ol>

	<ol style="list-style-type: none"> <li>10. If all answers are correct, player earns a badge for that difficulty level.</li> <li>11. System displays session performance stats: number of correct answers, wrong answers, and average time per question.</li> <li>12. Player returns to the homepage or selects another game.</li> </ol> <p><b>Alternative Flows</b></p> <p><b><i>Player Cancels Category Selection:</i></b></p> <ol style="list-style-type: none"> <li>1. Player opens the category selection screen but decides not to proceed.</li> <li>2. Player clicks the “Back” button.</li> <li>3. System returns the player to the homepage without starting the quiz.</li> </ol> <p><b><i>Player Times Out on a Question:</i></b></p> <ol style="list-style-type: none"> <li>1. Player does not answer the question within the allowed time.</li> <li>2. System automatically marks the question as incorrect.</li> <li>3. System proceeds to the next question.</li> </ol> <p><b><i>Player Closes Game Prematurely:</i></b></p> <ol style="list-style-type: none"> <li>1. During the quiz, player exits or closes the game (intentionally or accidentally).</li> <li>2. System records the session as incomplete.</li> <li>3. No badge or stats are saved.</li> <li>4. Player must restart the quiz from the beginning.</li> </ol>
--	--

*Table 12. Use Case: View Badges*

<b>Use Case ID</b>	UC-05
<b>Use Case Name</b>	View Badges
<b>Created by</b>	Janice Maxene Salipande
<b>Date Created</b>	06-04-2025
<b>Purpose</b>	To allow the player to view earned badges and claim rewards if criteria are met.
<b>Requirement Traceability</b>	BR-05
<b>Priority</b>	High
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>• Player must be logged in.</li> <li>• Player must have played game modes and earned badges.</li> </ul>
<b>Postconditions</b>	<ul style="list-style-type: none"> <li>• Player either views current badge progress or successfully claims a reward.</li> </ul>
<b>Actors</b>	Player
<b>Include</b>	None
<b>Extends</b>	<ul style="list-style-type: none"> <li>• Claim Reward</li> </ul>

<b>Flow of Actions</b>	<p><b>Basic Flow:</b></p> <ol style="list-style-type: none"> <li>1. Player clicks the “Your Badges” button on the homepage.</li> <li>2. System displays badge collections grouped by difficulty level.</li> <li>3. Player views which badges they’ve earned and which ones are missing.</li> <li>4. For any difficulty level with 3 badges (indicating full completion), a “Claim” button is enabled.</li> <li>5. Player clicks the “Claim” button.</li> <li>6. System confirms reward eligibility and updates the reward status to “Claimed”.</li> <li>7. System may also notify the admin (for DOST fair setups) for physical reward distribution.</li> <li>8. Player exits the reward section or continues using the system.</li> </ol> <p><b>Alternative Flows</b></p> <p><b><i>Player Has Incomplete Badge Set:</i></b></p> <ol style="list-style-type: none"> <li>1. Player views a difficulty level with less than 3 badges.</li> <li>2. “Claim” button is disabled and shows “Locked.”</li> <li>3. Player can choose to continue playing to earn missing badges.</li> </ol> <p><b><i>Player Tries to Claim Already Claimed Reward:</i></b></p> <ol style="list-style-type: none"> <li>1. Player clicks on a previously claimed reward.</li> <li>2. System shows a message: “Reward already claimed.”</li> <li>3. System disables re-claiming for the same reward.</li> </ol> <p><b><i>System Fails to Update Reward Status:</i></b></p> <ol style="list-style-type: none"> <li>1. Player clicks “Claim” but due to a technical error, the claim fails.</li> <li>2. System shows an error message: “Failed to claim reward. Please try again later.”</li> <li>3. Player may retry or contact staff at the exhibit/fair.</li> </ol>
------------------------	---

Table 13. Use Case: Play Whiz Memory Match

<b>Use Case ID</b>	UC-06
<b>Use Case Name</b>	Play Whiz Memory Match
<b>Created by</b>	Janice Maxene Salipande
<b>Date Created</b>	06-04-2025
<b>Purpose</b>	To allow the player to play the memory card matching game.
<b>Requirement Traceability</b>	BR-06
<b>Priority</b>	High
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>• Player must be logged in.</li> </ul>



<b>Postconditions</b>	<ul style="list-style-type: none"> <li>• Game result (time taken, best time) is displayed to the player.</li> <li>• Player stats are updated.</li> </ul>
<b>Actors</b>	Player
<b>Include</b>	None
<b>Extends</b>	None
<b>Flow of Actions</b>	<p><b>Basic Flow:</b></p> <ol style="list-style-type: none"> <li>1. Player clicks on the "Whiz Memory Match" game mode from the homepage.</li> <li>2. System displays the difficulty selection screen: <ul style="list-style-type: none"> <li>• Easy: 5 pairs</li> <li>• Average: 6 pairs</li> <li>• Difficult: 7 pairs</li> </ul> </li> <li>3. Player selects a difficulty level.</li> <li>4. System generates and shuffles the card grid based on selected difficulty.</li> <li>5. Game screen loads, showing the timer, score, moves, and card grid.</li> <li>6. Player begins clicking cards to reveal and match pairs.</li> <li>7. System: <ul style="list-style-type: none"> <li>• Records number of moves</li> <li>• Starts timer upon first click</li> <li>• Records score for correct matches</li> </ul> </li> <li>8. Once all pairs are matched: <ul style="list-style-type: none"> <li>• System stops the timer</li> <li>• Displays the game summary (time taken, best time)</li> <li>• Updates the best time if the current time is faster than previous best</li> </ul> </li> <li>9. Player can choose to replay, or return to homepage.</li> </ol> <p><b>Alternative Flows</b></p> <p><b><i>Player Closes Game Before Completion:</i></b></p> <ol style="list-style-type: none"> <li>1. Player starts a game but closes the app or navigates away.</li> <li>2. System does not save partial progress.</li> <li>3. Player must restart the game from the beginning.</li> </ol> <p><b><i>System Fails to Load Cards:</i></b></p> <ol style="list-style-type: none"> <li>1. Upon selecting a difficulty, the game screen fails to load due to a glitch.</li> <li>2. System shows error: "Failed to load memory cards. Please try again."</li> <li>3. Player is redirected to the difficulty selection screen to try again.</li> </ol> <p><b><i>New Time Is Not Faster Than Previous Best:</i></b></p> <ol style="list-style-type: none"> <li>1. Player finishes the game with a time slower than their best for that level.</li> </ol>

	2. System updates the current session's stats but does not change the best time.
--	--

Table 14. Use Case: Play Whiz Puzzle

<b>Use Case ID</b>	UC-07
<b>Use Case Name</b>	Play Whiz Puzzle
<b>Created by</b>	Janice Maxene Salipande
<b>Date Created</b>	06-04-2025
<b>Purpose</b>	To allow the player to play the jigsaw-style puzzle game.
<b>Requirement Traceability</b>	BR-07
<b>Priority</b>	High
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>• Player must be logged in.</li> </ul>
<b>Postconditions</b>	<ul style="list-style-type: none"> <li>• Game result (time taken, best time) is displayed to the player.</li> <li>• Player stats are updated.</li> </ul>
<b>Actors</b>	Player
<b>Include</b>	None
<b>Extends</b>	None
<b>Flow of Actions</b>	<p><b>Basic Flow:</b></p> <ol style="list-style-type: none"> <li>1. Player clicks on the "Whiz Puzzle" game mode from the homepage.</li> <li>2. System displays the difficulty selection screen: <ul style="list-style-type: none"> <li>• Easy: fewer puzzle pieces</li> <li>• Average: moderate puzzle pieces</li> <li>• Difficult: more puzzle pieces</li> </ul> </li> <li>3. Player selects a difficulty level.</li> <li>4. System displays available categories/themes (e.g., Solar System, Scientists, Human Body, Animals, Geometry, Random/Shuffle).</li> <li>5. Player selects a category/theme.</li> <li>6. Puzzle game screen loads based on selected difficulty and theme.</li> <li>7. Timer starts when the player moves the first piece.</li> <li>8. Player arranges pieces to complete the puzzle.</li> <li>9. Once the puzzle is completed: <ul style="list-style-type: none"> <li>• System stops the timer</li> <li>• Displays time taken to solve</li> <li>• Compares and updates best time if beaten</li> </ul> </li> <li>10. Player can choose to replay, or return to homepage.</li> </ol> <p><b>Alternative Flows</b></p> <p><b><i>Player Selects Random/Shuffle Category:</i></b></p>

	<ol style="list-style-type: none"> <li>1. Player selects "Random/Shuffle" instead of a specific theme.</li> <li>2. System randomly selects a category and loads the corresponding puzzle.</li> </ol> <p><b>Puzzle Fails to Load:</b></p> <ol style="list-style-type: none"> <li>1. System fails to load the puzzle due to technical issue.</li> <li>2. Error message appears: "Unable to load puzzle. Please try again."</li> <li>3. Player is redirected to category selection screen.</li> </ol> <p><b>Player Leaves Game Mid-Session:</b></p> <ol style="list-style-type: none"> <li>1. Player exits or navigates away from the game before completing it.</li> <li>2. System does not save incomplete puzzle session.</li> <li>3. Player must restart the puzzle if returning.</li> </ol> <p><b>Player Does Not Beat Best Time:</b></p> <ol style="list-style-type: none"> <li>1. System detects current session time is slower than previous best.</li> <li>2. System updates the current session's stats but does not change the best time.</li> </ol>
--	--

Table 15. Use Case: Play Whiz Battle

<b>Use Case ID</b>	UC-08
<b>Use Case Name</b>	Play Whiz Battle
<b>Created by</b>	Janice Maxene Salipande
<b>Date Created</b>	06-04-2025
<b>Purpose</b>	To enable players to engage in a 1v1 quiz battle by creating or joining a game using a game code.
<b>Requirement Traceability</b>	BR-08
<b>Priority</b>	High
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>• Player must be logged in.</li> </ul>
<b>Postconditions</b>	<ul style="list-style-type: none"> <li>• Battle result is displayed to the player.</li> <li>• Player stats are updated.</li> </ul>
<b>Actors</b>	Player
<b>Include</b>	<ul style="list-style-type: none"> <li>• Create Battle</li> <li>• Join Battle via Game Code</li> </ul>
<b>Extends</b>	None
<b>Flow of Actions</b>	<p><b>Basic Flow:</b></p> <ol style="list-style-type: none"> <li>1. Player clicks on the "Whiz Battle" game mode from the homepage.</li> <li>2. System displays two options: <ul style="list-style-type: none"> <li>• Create Battle</li> <li>• Join Battle</li> </ul> </li> <li>3. If player selects Create Battle:</li> </ol>

- System prompts the player to select a category (Science or Math).
  - Player selects a difficulty level (Easy, Average, Difficult).
  - System generates a unique game code and displays a “waiting for opponent” screen.
  - Second player joins using the code.
  - Once both players are ready, the battle begins.
4. If player selects Join Battle:
    - Player enters the game code received from another player.
    - System validates the code and connects both players.
    - Battle starts once both players are ready.
  5. During the battle:
    - A series of questions appear, same as in the solo quiz.
    - After each question, a progress bar displays which player is leading.
  6. After final question:
    - System calculates total scores.
    - Displays final result: Victory/Defeat/Draw.
    - Updates player stats (win/loss record, match history, etc.).

#### **Alternative Flows**

##### ***Opponent Fails to Join Created Battle:***

1. Player creates a battle and waits for an opponent.
2. Opponent does not join within a set time (e.g., 2–3 minutes).
3. System prompts: “No opponent found. Try again?”
4. Player may retry or return to homepage.

##### ***Invalid Game Code Entered:***

1. Player attempts to join a battle using an invalid or expired code.
2. System shows error: “Invalid game code. Please check and try again.”
3. Player can re-enter code or return to homepage.

##### ***Connection Interrupted During Battle:***

1. One player disconnects mid-game due to connectivity issue.
2. System pauses the game and attempts reconnection.
3. If reconnection fails, match ends with status “incomplete”.
4. Stats are not updated; player is redirected to homepage.

##### ***Both Players Get Equal Scores:***

1. After final question, system detects a tie.

	<ol style="list-style-type: none"> <li>2. Message displayed: "It's a draw! Great match!"</li> <li>3. Match is logged with "draw" status.</li> </ol> <p><b>Player Exits Before Match Ends:</b></p> <ol style="list-style-type: none"> <li>1. Player manually exits during battle.</li> <li>2. System counts it as a forfeit.</li> <li>3. Opponent wins by default; stats updated accordingly.</li> </ol>
--	---

Table 16. Use Case: View Stats

<b>Use Case ID</b>	UC-09
<b>Use Case Name</b>	View Stats
<b>Created by</b>	Janice Maxene Salipande
<b>Date Created</b>	06-04-2025
<b>Purpose</b>	To allow players to view their performance data and gameplay statistics for all game modes.
<b>Requirement Traceability</b>	BR-09
<b>Priority</b>	Medium
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>• Player must be logged in.</li> </ul>
<b>Postconditions</b>	<ul style="list-style-type: none"> <li>• Stats are displayed on-screen.</li> </ul>
<b>Actors</b>	Player
<b>Include</b>	None
<b>Extends</b>	None
<b>Flow of Actions</b>	<p><b>Basic Flow:</b></p> <ol style="list-style-type: none"> <li>1. Player clicks on the "Stats" button in the navigation bar on the homepage.</li> <li>2. System displays a tabbed or sectioned stats interface with the following breakdowns:             <ol style="list-style-type: none"> <li>a. For Whiz Challenge:                 <ul style="list-style-type: none"> <li>○ Total rewards claimed</li> <li>○ Accuracy per level and overall</li> <li>○ Top category (Math/Science)</li> <li>○ Most played level (Easy/Level/Difficult)</li> </ul> </li> <li>b. For Whiz Memory Match:                 <ul style="list-style-type: none"> <li>○ Best time per level and overall</li> <li>○ Average time per level and overall</li> <li>○ Accuracy per level and overall</li> <li>○ Total plays per level and overall</li> </ul> </li> <li>c. For Whiz Puzzle:                 <ul style="list-style-type: none"> <li>○ Best time per level and overall</li> <li>○ Average time per level and overall</li> <li>○ Total puzzles completed per level and overall</li> <li>○ Top puzzle category</li> </ul> </li> </ol> </li> </ol>

	<p>d. For Whiz Battle:</p> <ul style="list-style-type: none"> <li>○ Total number of matches</li> <li>○ Win rate percentage</li> <li>○ Top category</li> <li>○ Most played level (Easy/Average/Difficult)</li> <li>○ Recent battles (table format showing opponent, category, difficulty, date &amp; time, result)</li> </ul> <p>3. Player can navigate between sections to view statistics for each game mode.</p> <p>4. Player can exit the stats page and return to the homepage.</p> <p><b>Alternative Flows</b></p> <p><b><i>Player Tries to Access Stats Without Playing Any Game Yet:</i></b></p> <ol style="list-style-type: none"> <li>1. Player clicks on "Stats" with no recorded gameplay.</li> <li>2. System shows placeholders or a message: "No stats available yet. Start playing to view your performance."</li> </ol> <p><b><i>Stats Load Slowly:</i></b></p> <ol style="list-style-type: none"> <li>1. Player clicks on "Stats".</li> <li>2. Due to data size or device lag, the system takes a few seconds to load.</li> <li>3. A loading spinner appears with message: "Fetching your stats..."</li> <li>4. Stats are displayed once data is ready.</li> </ol> <p><b><i>Data Error or Corruption:</i></b></p> <ol style="list-style-type: none"> <li>1. Player clicks "Stats".</li> <li>2. System encounters an error retrieving saved stats.</li> <li>3. System shows error: "Oops! Something went wrong. Please try again later."</li> <li>4. Player is redirected back to homepage or given the option to retry.</li> </ol> <p><b><i>Stats Interface Becomes Unresponsive:</i></b></p> <ol style="list-style-type: none"> <li>1. Player navigates to a specific stats tab.</li> <li>2. Interface freezes or fails to switch sections.</li> <li>3. Player can refresh the app or navigate back to homepage to retry.</li> </ol>
--	---

Table 17. Use Case: Logout

<b>Use Case ID</b>	UC-10
<b>Use Case Name</b>	Logout
<b>Created by</b>	Janice Maxene Salipande
<b>Date Created</b>	06-04-2025
<b>Purpose</b>	To allow the user to securely log out of their account and end the current session.

<b>Requirement Traceability</b>	BR-10, BR-18
<b>Priority</b>	High
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>User must be logged in.</li> </ul>
<b>Postconditions</b>	<ul style="list-style-type: none"> <li>User is logged out and redirected to the login page.</li> </ul>
<b>Actors</b>	Player, Admin
<b>Include</b>	None
<b>Extends</b>	None
<b>Flow of Actions</b>	<p><b>Basic Flow:</b></p> <ol style="list-style-type: none"> <li>1. User clicks on their avatar icon at the top-right corner of the screen.</li> <li>2. A dropdown menu appears.</li> <li>3. User clicks the “Logout” button.</li> <li>4. System ends the session and logs the user out.</li> <li>5. System redirects the user to the login screen.</li> <li>6. A confirmation message may be displayed: “You have successfully logged out.”</li> </ol> <p><b>Alternative Flows</b></p> <p><b><i>User Clicks Avatar Icon But Dropdown Doesn’t Appear:</i></b></p> <ol style="list-style-type: none"> <li>1. User clicks on their avatar icon.</li> <li>2. Due to a UI bug or lag, the dropdown doesn’t load.</li> <li>3. User refreshes the page or retries.</li> <li>4. Dropdown menu appears on retry.</li> </ol> <p><b><i>Logout Button Fails to Respond:</i></b></p> <ol style="list-style-type: none"> <li>1. User clicks the “Logout” button.</li> <li>2. System fails to respond or freeze occurs.</li> <li>3. User waits or retries the logout action.</li> <li>4. If unresponsive, user force-closes or restarts the app.</li> </ol>

Table 18. Use Case: Manage Players

<b>Use Case ID</b>	UC-11
<b>Use Case Name</b>	Manage Players
<b>Created by</b>	Janice Maxene Salipande
<b>Date Created</b>	06-04-2025
<b>Purpose</b>	To allow the admin to add, view, edit, or delete player accounts registered in the system.
<b>Requirement Traceability</b>	BR-11
<b>Priority</b>	High
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>Admin is logged into the system.</li> </ul>
<b>Postconditions</b>	<ul style="list-style-type: none"> <li>Admin successfully add, views, edits, or deletes player account(s) as needed.</li> <li>Changes are saved and reflected in the system.</li> </ul>

<b>Actors</b>	Admin
<b>Include</b>	<ul style="list-style-type: none"> <li>• Add New Player</li> <li>• Edit Player Info</li> <li>• Change Player Password</li> <li>• Delete Player Account</li> </ul>
<b>Extends</b>	None
<b>Flow of Actions</b>	<p><b>Basic Flow:</b></p> <ol style="list-style-type: none"> <li>1. Admin accesses the admin dashboard.</li> <li>2. Admin logs into the Admin Dashboard.</li> <li>3. Admin clicks on Users &gt; Players.</li> <li>4. System displays the full list of registered players in a table with details (e.g., name, username, category, location).</li> <li>5. Admin uses search or filters to locate a specific player.</li> <li>6. Admin clicks Add New Player (if creating a new account).</li> <li>7. System displays a form for player registration.</li> <li>8. Admin fills out the required fields and clicks Save.</li> <li>9. System validates and creates the new player account.</li> <li>10. Admin clicks Edit on a listed player to update their profile.</li> <li>11. System displays a form with pre-filled player data.</li> <li>12. Admin modifies fields like name, category, or location.</li> <li>13. Admin optionally clicks Change Password.</li> <li>14. System prompts for new password input and confirmation.</li> <li>15. Admin enters the new password and confirms.</li> <li>16. System validates and updates the password.</li> <li>17. Admin clicks Save Changes to apply all updates.</li> <li>18. System confirms that the profile and password updates were saved.</li> <li>19. Admin clicks Delete for a player if removal is required.</li> <li>20. System displays a confirmation prompt.</li> <li>21. Admin confirms the deletion.</li> <li>22. System removes the player account from the list.</li> </ol> <p><b>Alternative Flows</b></p> <p><b>Add New Player - Validation Failure:</b></p> <ol style="list-style-type: none"> <li>1. Admin submits the Add Player form with invalid/missing fields.</li> <li>2. System highlights errors and prompts for correction.</li> <li>3. Admin corrects the form and resubmits.</li> </ol> <p><b>Edit Profile - Cancelled:</b></p> <ol style="list-style-type: none"> <li>1. Admin opens a player profile but decides not to continue.</li> <li>2. Admin clicks Cancel or navigates away.</li> <li>3. System discards any unsaved changes.</li> </ol> <p><b>Change Password - Cancelled:</b></p> <ol style="list-style-type: none"> <li>1. Admin clicks Change Password but then cancels.</li> <li>2. System discards password input and returns to profile form.</li> </ol>



	<p><b>Save Fails (Edit/Add):</b></p> <ol style="list-style-type: none"> <li>1. Admin submits changes, but the system encounters a storage error.</li> <li>2. System shows: “Unable to save changes. Please try again.”</li> <li>3. Admin retries or contacts support.</li> </ol> <p><b>Delete Cancelled:</b></p> <ol style="list-style-type: none"> <li>1. Admin clicks Delete for a player.</li> <li>2. System prompts: “Are you sure you want to delete this player?”</li> <li>3. Admin clicks Cancel.</li> <li>4. Deletion is aborted.</li> </ol> <p><b>Player List Fails to Load:</b></p> <ol style="list-style-type: none"> <li>1. Admin navigates to the player list, but data doesn’t load.</li> <li>2. System shows: “Failed to load player data.”</li> <li>3. Admin refreshes the page or checks connection.</li> </ol>
--	--

Table 19. Use Case: Manage Admins

<b>Use Case ID</b>	UC-12
<b>Use Case Name</b>	Manage Admins
<b>Created by</b>	Janice Maxene Salipande
<b>Date Created</b>	06-04-2025
<b>Purpose</b>	To allow an existing administrator to view the list of admins, add new admins, or edit admin account details as needed.
<b>Requirement Traceability</b>	BR-12
<b>Priority</b>	High
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>• Admin is logged into the system.</li> </ul>
<b>Postconditions</b>	<ul style="list-style-type: none"> <li>• New admin accounts may be created.</li> <li>• Existing admin accounts may be updated or deleted.</li> <li>• The admin list reflects all changes made.</li> </ul>
<b>Actors</b>	Admin
<b>Include</b>	<ul style="list-style-type: none"> <li>• Add New Admin</li> <li>• Edit Admin Info</li> <li>• Change Admin Password</li> <li>• Delete Admin Account</li> </ul>
<b>Extends</b>	None
<b>Flow of Actions</b>	<p><b>Basic Flow:</b></p> <ol style="list-style-type: none"> <li>1. Admin logs into the dashboard.</li> <li>2. Admin clicks on the Users &gt; Admins menu.</li> <li>3. System displays a table of current admins with their details (name, username, role, etc.).</li> <li>4. Admin clicks Add Admin.</li> <li>5. System presents a form to input new admin details (name, username, password, role).</li> </ol>

	<ol style="list-style-type: none"> <li>6. Admin fills in the form and clicks Save.</li> <li>7. System validates and creates the new admin account.</li> <li>8. Admin can also click Edit beside an existing admin.</li> <li>9. Admin updates the necessary details and clicks Update.</li> <li>10. System saves the changes.</li> <li>11. Admin can also click Delete to remove an admin.</li> <li>12. System asks for confirmation before deleting.</li> <li>13. Upon confirmation, the admin is removed from the system.</li> </ol> <p><b>Alternative Flows</b></p> <p><b><i>Admin Cancels Add Admin Process:</i></b></p> <ol style="list-style-type: none"> <li>1. Admin clicks Add Admin.</li> <li>2. Admin starts filling out the form but clicks Cancel.</li> <li>3. System closes the form and returns to the admin list.</li> </ol> <p><b><i>Admin Cancels Edit Process:</i></b></p> <ol style="list-style-type: none"> <li>1. Admin clicks Edit beside an admin.</li> <li>2. Admin changes their mind and clicks Cancel.</li> <li>3. System discards any changes and returns to the admin list.</li> </ol> <p><b><i>Admin Cancels Deletion:</i></b></p> <ol style="list-style-type: none"> <li>1. Admin clicks Delete beside an admin.</li> <li>2. System prompts: "Are you sure you want to delete this admin?"</li> <li>3. Admin clicks Cancel.</li> <li>4. System cancels the action and returns to the list.</li> </ol> <p><b><i>System Fails to Save:</i></b></p> <ol style="list-style-type: none"> <li>1. Admin clicks Save or Update, but the system encounters a server issue.</li> <li>2. System shows an error: "Failed to save changes. Please try again."</li> <li>3. Admin retries after some time.</li> </ol>
--	---

Table 20. Use Case: Manage Quiz Questions

<b>Use Case ID</b>	UC-13
<b>Use Case Name</b>	Manage Quiz Questions
<b>Created by</b>	Janice Maxene D. Salipande
<b>Date Created</b>	06-04-2025
<b>Purpose</b>	To allow administrators to view, add, edit, or remove quiz questions.
<b>Requirement Traceability</b>	BR-13
<b>Priority</b>	High
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>• Admin is logged into the system.</li> </ul>
<b>Postconditions</b>	<ul style="list-style-type: none"> <li>• The list of questions is updated (added, edited, or deleted).</li> <li>• Imported spreadsheet data is validated and saved.</li> </ul>

<b>Actors</b>	Admin
<b>Include</b>	<ul style="list-style-type: none"> <li>• Import Spreadsheet</li> <li>• Add New Question</li> <li>• Edit Question Info</li> <li>• Delete Question</li> </ul>
<b>Extends</b>	None
<b>Flow of Actions</b>	<p><b>Basic Flow:</b></p> <ol style="list-style-type: none"> <li>1. Admin navigates to Quiz Content &gt; Questions.</li> <li>2. System displays a table of all quiz questions with filters (e.g., by Subject, Difficulty, Status).</li> <li>3. Admin chooses one of the following options:             <ol style="list-style-type: none"> <li>a. Admin clicks Add Question.</li> <li>b. System displays a form with fields:                 <ul style="list-style-type: none"> <li>• Question text</li> <li>• Four multiple-choice options</li> <li>• Correct answer</li> <li>• Subject/Category (Science or Math)</li> <li>• Difficulty level (Easy, Average, Difficult)</li> <li>• Optional: Attach image</li> <li>• Status (Active/Inactive)</li> </ul> </li> <li>c. Admin completes the form and clicks Save.</li> <li>d. System validates and saves the new question to the database.</li> <li>e. System confirms: "Question added successfully."</li> </ol> </li> <li>4. Admin clicks Edit beside an existing question.             <ol style="list-style-type: none"> <li>a. System displays the form pre-filled with current values.</li> <li>b. Admin updates any of the fields and clicks Update.</li> <li>c. System validates and saves the changes.</li> <li>d. System confirms: "Question updated successfully."</li> </ol> </li> <li>5. Admin clicks Delete beside a question.             <ol style="list-style-type: none"> <li>a. System prompts: "Are you sure you want to delete this question?"</li> <li>b. Admin confirms deletion.</li> <li>c. System deletes the question from the database.</li> <li>d. System confirms: "Question deleted."</li> </ol> </li> <li>6. Admin clicks Import Spreadsheet.             <ol style="list-style-type: none"> <li>a. System displays a file upload prompt and import instructions (e.g., required format).</li> <li>b. Admin uploads a valid spreadsheet (.xlsx or .csv).</li> <li>c. System parses and validates the data:</li> <li>d. Ensures no duplicate questions</li> <li>e. Validates required fields per row</li> <li>f. If valid, system imports all entries into the question list.</li> <li>g. System confirms: "Questions imported successfully."</li> </ol> </li> </ol> <p><b>Alternative Flows</b></p>

	<p><b>Admin Cancels Adding Question:</b></p> <ol style="list-style-type: none"> <li>1. Admin opens the Add Question form.</li> <li>2. Admin fills some fields but clicks Cancel.</li> <li>3. System discards all input and returns to the question list.</li> </ol> <p><b>Incomplete or Invalid Input:</b></p> <ol style="list-style-type: none"> <li>1. Admin submits the Add/Edit form with missing or invalid data (e.g., duplicate options, missing correct answer).</li> <li>2. System displays validation errors and highlights problematic fields.</li> <li>3. Admin corrects the data and resubmits.</li> </ol> <p><b>Admin Cancels Edit:</b></p> <ol style="list-style-type: none"> <li>1. Admin clicks Edit for a question. Admin decides not to proceed and clicks Cancel.</li> <li>2. System discards changes and returns to the list.</li> </ol> <p><b>Admin Cancels Delete:</b></p> <ol style="list-style-type: none"> <li>1. Admin clicks Delete beside a question.</li> <li>2. System prompts for confirmation.</li> <li>3. Admin clicks Cancel.</li> <li>4. System aborts deletion and returns to the list.</li> </ol> <p><b>Import Fails (Invalid Format or Data):</b></p> <ol style="list-style-type: none"> <li>1. Admin uploads a malformed or incorrectly formatted file.</li> <li>2. System rejects the file and displays an error: "Invalid file format or missing required fields."</li> <li>3. Admin corrects the file and uploads again.</li> </ol> <p><b>Database Save Error:</b></p> <ol style="list-style-type: none"> <li>1. Admin clicks Save or Update, but a server or storage issue occurs.</li> <li>2. System shows: "Unable to save changes. Please try again."</li> <li>3. Admin retries the operation after resolving the issue.</li> </ol>
--	--

Table 21. Use Case: Manage Difficulty Levels

<b>Use Case ID</b>	UC-14
<b>Use Case Name</b>	Manage Difficulty Levels
<b>Created by</b>	Janice Maxene Salipande
<b>Date Created</b>	06-04-2025
<b>Purpose</b>	To allow administrators to configure the quiz difficulty levels by setting the number of questions, score, and timer.
<b>Requirement Traceability</b>	BR-14
<b>Priority</b>	Medium
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>• Admin is logged into the system.</li> </ul>

<b>Postconditions</b>	<ul style="list-style-type: none"> <li>• Difficulty level settings are updated.</li> <li>• These settings are reflected during gameplay.</li> </ul>
<b>Actors</b>	Admin
<b>Include</b>	<ul style="list-style-type: none"> <li>• Set Number of Questions</li> <li>• Set Score Points</li> <li>• Set Timer</li> </ul>
<b>Extends</b>	None
<b>Flow of Actions</b>	<p><b>Basic Flow:</b></p> <ol style="list-style-type: none"> <li>1. Admin navigates to the Quiz Content &gt; Difficulty menu.</li> <li>2. System displays difficulty levels.</li> <li>3. Admin selects a difficulty level (Easy, Average, Difficult) to configure.</li> <li>4. System displays a form to update: <ul style="list-style-type: none"> <li>• Number of questions</li> <li>• Score points</li> <li>• Timer per session</li> </ul> </li> <li>5. Admin edits the fields as desired.</li> <li>6. Admin clicks Save.</li> <li>7. System validates and updates the difficulty settings.</li> <li>8. Confirmation message is displayed.</li> </ol> <p><b>Alternative Flows</b></p> <p><b>Invalid Configuration Values:</b></p> <ol style="list-style-type: none"> <li>1. Admin enters non-numeric or out-of-range values for time or number of questions.</li> <li>2. System highlights the errors and displays a validation message.</li> <li>3. Admin corrects the values and clicks Save again.</li> </ol> <p><b>Admin Cancels Update:</b></p> <ol style="list-style-type: none"> <li>1. Admin opens a difficulty level but decides not to make changes.</li> <li>2. Admin clicks Cancel.</li> <li>3. System discards changes and returns to the difficulty list.</li> </ol> <p><b>Save Failure:</b></p> <ol style="list-style-type: none"> <li>1. Admin clicks Save, but a system error occurs.</li> <li>2. System displays: "Failed to save changes. Please try again."</li> <li>3. Admin retries later after system recovers.</li> </ol>

Table 22. Use Case: Manage Rewards

<b>Use Case ID</b>	UC-15
<b>Use Case Name</b>	Manage Rewards
<b>Created by</b>	Janice Maxene Salipande
<b>Date Created</b>	06-04-2025

<b>Purpose</b>	To allow administrators to view players' badge progress and manage the reward claiming process by validating badge completion and confirming or rejecting reward distribution.
<b>Requirement Traceability</b>	BR-15
<b>Priority</b>	High
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>• Admin is logged into the system.</li> <li>• At least one player has participated in games and earned badges.</li> </ul>
<b>Postconditions</b>	<ul style="list-style-type: none"> <li>• The updated badge progress is reflected in the player interface.</li> </ul>
<b>Actors</b>	Admin
<b>Include</b>	None
<b>Extends</b>	<ul style="list-style-type: none"> <li>• Give Reward</li> </ul>
<b>Flow of Actions</b>	<p><b>Basic Flow:</b></p> <ol style="list-style-type: none"> <li>1. Admin navigates to the Users &gt; Players menu.</li> <li>2. System displays a list of players with their badge progress.</li> <li>3. Admin selects a specific player to view badge details.</li> <li>4. System shows the badge status for each difficulty level.</li> <li>5. If a player has completed 3 badges for a specific level, a "Give Reward" button is active.</li> <li>6. Admin validates the badge completion.</li> <li>7. Admin clicks "Give Reward" to mark the reward as distributed.</li> <li>8. System updates the player's reward status and logs the claim.</li> <li>9. Confirmation message appears.</li> </ol> <p><b>Alternative Flows</b></p> <p><b>No Players Yet:</b></p> <ol style="list-style-type: none"> <li>1. Admin accesses the Users &gt; Players menu.</li> <li>2. System detects no player data.</li> <li>3. System displays: "No players available for reward management yet."</li> </ol> <p><b>Invalid Player Selection:</b></p> <ol style="list-style-type: none"> <li>1. Admin tries to select a player from the list.</li> <li>2. Due to a UI or sync issue, the player data fails to load.</li> <li>3. System displays: "Failed to load player details. Please try again."</li> <li>4. Admin retries or selects a different player.</li> </ol> <p><b>Incomplete Badges:</b></p> <ol style="list-style-type: none"> <li>1. Admin views a player who hasn't completed 3 badges for a level.</li> <li>2. "Give Reward" button is disabled and marked as Locked.</li> <li>3. Admin cannot proceed to validate or confirm.</li> </ol>

	<b>System Save Failure:</b> <ol style="list-style-type: none"> <li>1. Admin clicks “Give Reward”, but a server error occurs.</li> <li>2. System shows: “Reward claim not saved. Please retry.”</li> <li>3. Admin retries once system resumes normal function.</li> </ol>
--	--

Table 23. Use Case: View Analytics

<b>Use Case ID</b>	UC-16
<b>Use Case Name</b>	View Analytics
<b>Created by</b>	Janice Maxene Salipande
<b>Date Created</b>	06-04-2025
<b>Purpose</b>	To allow the administrator to monitor user engagement, player behavior, and game usage trends through data visualizations and summarized analytics.
<b>Requirement Traceability</b>	BR-16
<b>Priority</b>	High
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>• Admin is logged into the system.</li> <li>• Player data and gameplay activity have been recorded by the system.</li> </ul>
<b>Postconditions</b>	<ul style="list-style-type: none"> <li>• Admin views updated visual insights to make informed decisions.</li> </ul>
<b>Actors</b>	Admin
<b>Include</b>	<ul style="list-style-type: none"> <li>• View Total Registered Players</li> <li>• View/Filter Player Registration Trend</li> <li>• View Most Played Game Mode</li> <li>• View Most Played Level Per Game Mode</li> <li>• View Top Category</li> <li>• View Average Session Duration</li> <li>• View Reward Claims Summary</li> </ul>
<b>Extends</b>	<ul style="list-style-type: none"> <li>• Export Report</li> </ul>
<b>Flow of Actions</b>	<b>Basic Flow:</b> <ol style="list-style-type: none"> <li>1. Admin navigates to the Dashboard menu from the sidebar.</li> <li>2. System loads visual analytics including charts and graphs.</li> <li>3. Dashboard shows: <ul style="list-style-type: none"> <li>• Total number of registered players.</li> <li>• Trend graph of player registration over time.</li> <li>• Most played game mode.</li> <li>• Most played level per game mode.</li> <li>• Top quiz categories.</li> <li>• Average session duration per game mode.</li> <li>• Reward claims summary</li> </ul> </li> <li>4. Admin hovers over charts or selects filter options.</li> <li>5. System dynamically updates the visualizations based on selected filters.</li> <li>6. Admin interprets the data and optionally downloads the analytics report as a CSV.</li> </ol>

	<p><b>Alternative Flows</b></p> <p><b>No Data Yet:</b></p> <ol style="list-style-type: none"><li>1. Admin opens the Analytics Dashboard.</li><li>2. System detects insufficient data (e.g., no gameplay yet).</li><li>3. System displays: "Analytics will appear once there's enough gameplay activity."</li></ol> <p><b>Charts Fail to Load:</b></p> <ol style="list-style-type: none"><li>1. Admin navigates to the dashboard.</li><li>2. Due to a server or rendering error, some charts fail to display.</li><li>3. System shows: "Some data failed to load. Please refresh or try again later."</li><li>4. Admin refreshes the page; system retries chart rendering.</li></ol> <p><b>Invalid Filter Combination:</b></p> <ol style="list-style-type: none"><li>1. Admin selects filters that produce no matching data (e.g., filters by a time range when no player was active).</li><li>2. System shows blank charts or a message like: "No data available for the selected criteria."</li><li>3. Admin adjusts filters.</li></ol> <p><b>Download Fails:</b></p> <ol style="list-style-type: none"><li>1. Admin clicks the Export button.</li><li>2. System encounters an export error.</li><li>3. System shows: "Unable to download report. Please retry."</li><li>4. Admin attempts again after checking system status.</li></ol>
--	---



## Conclusion

At this stage of the STARBOOKS Whiz Challenge system development, we have successfully achieved several key objectives outlined in our initial project plan. Specifically, we have:

- **Completed the Planning and Requirements Analysis Phase:** We gathered and documented all necessary system requirements, identified key user roles, and created detailed use case descriptions for both player and admin functionalities.
- **Finalized Core System Features and Functionalities:** Through continuous meetings with DOST-STII and collaborative brainstorming sessions, we clarified and finalized the essential components of the system, including game modes, player rewards, and admin dashboards.
- **Developed Comprehensive Use Case Descriptions:** Each use case now includes clearly structured tables with preconditions, postconditions, basic flows, and alternative flows to guide the development phase.
- **Defined Analytics and AI Integration Points:** For the admin dashboard, we outlined essential data visualizations and AI feature designed to provide actionable insights and intelligent assistance, such as generating quiz questions.
- **Finalized the Technical Stack:** The technologies, tools, and frameworks to be used in the development of the system have been selected and confirmed.

Currently, we are in the **wireframing phase using Figma**, translating all the finalized functionalities into high-fidelity UI designs. This includes player interfaces, admin dashboards, and game flows. We are also maintaining close coordination with DOST-STII for continuous feedback, technical guidance, and system validation to ensure alignment with their expectations and real-world needs.

For the next iteration, we plan to:

- **Complete the UI/UX Wireframes**
- **Present the wireframes to DOST-STII for feedback and approval**
- **Begin the system design and prototyping phase**, focusing on frontend and backend integration planning
- **Set up the development environment based on the finalized tech stack**
- **Start the initial coding sprint based on approved designs**

By following our roadmap and timeline, we aim to keep the project on track and ensure a well-structured and meaningful output for the upcoming development phases.

## References

- [1] Department of Science and Technology, "Mandate, Mission, and Vision," *Dost.gov.ph*, Apr. 26, 2014. <https://www.dost.gov.ph/transparency/about-dost/mandate-mission-and-vision.html>
- [2] Super User, "Mission and Vision," *Dost.gov.ph*, 2022. <https://www.stii.dost.gov.ph/transparency/about-us/mission-and-vision>
- [3] Department of Science and Technology – Science and Technology Information Institute (DOST-STII), "STARBOOKS: A Decade of Bringing Library-in-a-Box to Communities," Department of Science and Technology, Taguig City, 2021.
- [4] Department of Science and Technology – Science and Technology Information Institute (DOST-STII), "About STARBOOKS," Department of Science and Technology – Science and Technology Information Institute, Taguig City.
- [5] United Nations, "Transforming our world: the 2030 Agenda for Sustainable Development," United Nations, New York, 2015.
- [6] UNESCO, "Education for Sustainable Development Goals: Learning objectives," UNESCO, Paris, 2017.
- [7] "National Innovation Agenda and Strategy Document (NIASD)," National Economic and Development Authority (NEDA). Available: <https://filipinnovation.depdev.gov.ph/material/img/NIASD-PUBLICATION-V24.4.pdf>
- [8] Department of Science and Technology – Science and Technology Information Institute, "DOST's STARBOOKS reinvents itself with more upgrades on its 10th year," Department of Science and Technology – Science and Technology Information Institute, Taguig City, 2021.
- [9] E. Ratinho and C. Martins, "The role of gamified learning strategies in students' motivation in high school and higher education: A systematic review," *Heliyon*, vol. 9, no. 8, 2023.
- [10] Department of Science and Technology – Science and Technology Information Institute, "2024 DOST-STII Performance Report," Department of Science and Technology – Science and Technology Information Institute, Taguig City, 2024.
- [11] P. Puntillo, "Why gamification increases student motivation and learning," 21 December 2023.
- [12] Department of Science and Technology – Science and Technology Information Institute (DOST-STII), "Young learners learn financial literacy from DOST-STII-BPI Foundation partnership," Department of Science and Technology – Science and Technology Information Institute, Taguig City, 2023.
- [13] D. Strmečki, A. Bernik, and D. Radošević, "Gamification in E-Learning: Introducing Gamified Design Elements into E-Learning Systems," *Journal of Computer Science*, vol. 11, no. 12, pp. 1108–1117, Dec. 2015, doi: <https://doi.org/10.3844/jcssp.2015.1108.1117>.

- [14] A. N. Saleem, N. M. Noori, and F. Ozdamli, "Gamification Applications in E-learning: A Literature Review," *Technology, Knowledge and Learning*, vol. 27, no. 1, pp. 139–159, Jan. 2021, doi: <https://doi.org/10.1007/s10758-020-09487-x>.
- [15] M. Munoto, M. S. Sumbawati, and S. F. M. Sari, "The Use of Mobile Technology in Learning With Online and Offline Systems," *International Journal of Information and Communication Technology Education*, vol. 17, no. 2, pp. 54–67, Apr. 2021, doi: <https://doi.org/10.4018/ijicte.2021040104>.
- [16] "RACHEL: Remote Area Community Hotspot for Education & Learning," *hundred.org*, Apr. 27, 2020. <https://hundred.org/en/innovations/rachel-remote-area-community-hotspot-for-education-learning>
- [17] "Kolibri," *hundred.org*, Jul. 01, 2019. <https://hundred.org/en/innovations/kolibri>
- [18] S. Yeon, "An User-Friendly Kiosk System Based on Deep Learning," *Journal of Korea Society of Industrial Information Systems*, vol. 29, no. 1, pp. 1–13, 2024, doi: <https://doi.org/10.9723/jksiis.2024.29.1.001>.
- [19] M. Dzhurow -Student and A. Atanasova, "Reports Awarded with 'Best Paper' Crystal Prize '21 APPLICATIONS AND BENEFITS OF USING KIOSKS AND INTEGRATED KIOSK SOFTWARE IN EDUCATION SPHERE." Available: <https://conf.uni-ruse.bg/bg/docs/cp21/bp/bp-39.pdf>
- [20] L. Dagne, "Lukas Dagne Flutter for cross-platform App and SDK development," 2019. Available: <https://www.theseus.fi/bitstream/handle/10024/172866/Lukas%20Dagne%20Thesis.pdf>
- [21] A. Tashildar *et al.*, "APPLICATION DEVELOPMENT USING FLUTTER," 1262. Available: [https://www.irjmet.com/uploadedfiles/paper/volume2/issue\\_8\\_august\\_2020/3180/1628083124.pdf](https://www.irjmet.com/uploadedfiles/paper/volume2/issue_8_august_2020/3180/1628083124.pdf)
- [22] S. A. Wani, "KIOSK: A NEW TREND FOR DIGITAL LIBRARY," Jan. 01, 2019. [https://www.researchgate.net/publication/330924708\\_KIOSK\\_A\\_NEW\\_TREND\\_FOR\\_DIGITAL\\_LIBRARY](https://www.researchgate.net/publication/330924708_KIOSK_A_NEW_TREND_FOR_DIGITAL_LIBRARY)
- [23] A. Paul and M. Haldar, "Offline-First App," *Apress eBooks*, pp. 145–164, Jan. 2023, doi: [https://doi.org/10.1007/978-1-4842-8707-1\\_5](https://doi.org/10.1007/978-1-4842-8707-1_5)
- [24] M. Kleppmann, A. Wiggins, P. van Hardenberg, and M. McGranaghan, "Local-first software: you own your data, in spite of the cloud," *Proceedings of the 2019 ACM SIGPLAN International Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software*, Oct. 2019, doi: <https://doi.org/10.1145/3359591.3359737>
- [25] A. I. Wang, M. Zhu, and R. Sætre, *The Effect of Digitizing and Gamifying Quizzing in Classrooms*. Academic Conferences and Publishing International, 2016. Available: <https://ntnuopen.ntnu.no/ntnu-xmlui/handle/11250/2426374>
- [26] K. P. Nuci, R. Tahir, A. I. Wang, and A. S. Imran, "Game-based digital quiz as a tool for improving students' engagement and learning in online lectures," *IEEE Access*, vol. 9, pp. 1–1, 2021, doi: <https://doi.org/10.1109/access.2021.3088583>.

- [27] “KooBits - About,” *KooBits*, Apr. 07, 2025. <https://www.koobits.com/about> (accessed May 29, 2025).
- [28] K. Wilkinson, “The Integration of Mobile Learning App-based Quiz-Games in Higher Education Teaching of Anatomical Sciences,” 2016. Accessed: Feb. 26, 2025. [Online]. Available: [https://repository.mdx.ac.uk/download/5ebec042a07ac302c635ad231b640ed66bc6596953fb6f4ac2971f9977d317a2/4511772/KWilkinson%20thesis\\_Redacted.pdf](https://repository.mdx.ac.uk/download/5ebec042a07ac302c635ad231b640ed66bc6596953fb6f4ac2971f9977d317a2/4511772/KWilkinson%20thesis_Redacted.pdf)
- [29] “Assessment for learning with mobile apps: exploring the potential of quizizz in the educational context | International Journal of Development Research (IJDR),” *www.journalijdr.com*. <https://www.journalijdr.com/assessment-learning-mobile-apps-exploring-potential-quizizz-educational-context>
- [30] R. Sivakumar, “Global and Lokal Distance Education-GLOKALde,” no. 1, 2022, Available: <https://www.glokalde.com/pdf/issues/21/Article2.pdf>
- [31] A. Al-Thaqib *et al.*, “Brain Training Games Enhance Cognitive Function in Healthy Subjects,” *Medical science monitor basic research*, vol. 24, pp. 63–69, 2018, doi: <https://doi.org/10.12659/msmbr.909022>.
- [32] None Hermahayu, Afifudin Isnain, None Mukholid, and Septi Hardiana, “The effectiveness of smart memory cards to improve working memory abilities of children aged 5-12 years,” *AIP Conference Proceedings*, Jan. 2023, doi: <https://doi.org/10.1063/5.0135434>.
- [33] A. Khadem, R. Khoshkholgh, F. Vafi Sani, and Z. Dolatabadi, “Using the Jigsaw (Puzzle) Method in Academic Environments: Benefits and Challenges,” *Med Edu Bull*, vol. 3, no. 2, 2022, doi: <https://doi.org/10.22034/MEB.2022.333390.1053>.
- [34] P. Fissler, O. C. Küster, D. Laptinskaya, L. S. Loy, C. A. F. von Arnim, and I.-T. Kolassa, “Jigsaw Puzzling Taps Multiple Cognitive Abilities and Is a Potential Protective Factor for Cognitive Aging,” *Frontiers in Aging Neuroscience*, vol. 10, no. 299, Oct. 2018, doi: <https://doi.org/10.3389/fnagi.2018.00299>.
- [35] H. Marin-Vega, G. Alor-Hernández, R. Zatarain-Cabada, M. L. Barron-Estrada, and J. L. García-Alcaraz, “A Brief Review of Game Engines for Educational and Serious Games Development,” *Journal of Information Technology Research*, vol. 10, no. 4, pp. 1–22, Oct. 2017, doi: <https://doi.org/10.4018/jitr.2017100101>.
- [36] E. Christopoulou and S. Xinogalos, “Overview and Comparative Analysis of Game Engines for Desktop and Mobile Devices,” *International Journal of Serious Games*, vol. 4, no. 4, Dec. 2017, doi: <https://doi.org/10.17083/ijsg.v4i4.194>.
- [37] N. E. Cagiltay, E. Ozcelik, and N. S. Ozcelik, “The effect of competition on learning in games,” *Computers & Education*, vol. 87, no. 87, pp. 35–41, Sep. 2015, doi: <https://doi.org/10.1016/j.compedu.2015.04.001>.
- [38] V. Riemer and C. Schrader, “Playing to learn or to win? The role of students’ competition preference on self-monitoring and learning outcome when learning with a serious game,” *Interactive Learning Environments*, pp. 1–13, Apr. 2020, doi: <https://doi.org/10.1080/10494820.2020.1752741>.

- [39] R. X. D. Hawkins, "Conducting real-time multiplayer experiments on the web," *Behavior Research Methods*, vol. 47, no. 4, pp. 966–976, Oct. 2014, doi: <https://doi.org/10.3758/s13428-014-0515-6>.
- [40] M. Korkmaz and D. A. Demir, "Real-Time Web Applications with Node.js: Leveraging WebSockets and Socket.IO for Seamless Communication," *International Journal of Trend in Scientific Research and Development*, vol. Volume-4, no. Issue-5, May 2025, Accessed: May 29, 2025. [Online]. Available: <https://www.ijtsrd.com/computer-science/other/31874/realtime-web-applications-with-nodejs-leveraging-websockets-and-socketio-for-seamless-communication/dr-ayse-demir>
- [41] P. Vranesic, K. Aleksic-Maslac, and B. Sinkovic, "Influence of Gamification Reward System on Student Motivation," *2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, May 2019, doi: <https://doi.org/10.23919/mipro.2019.8756848>.
- [42] S. A. Kochurina, O. O. Fedorova, and K. P. Zakharov, "Learning Motivation of Students of Adaptive Learning Platforms," *Lecture notes in networks and systems*, pp. 303–310, Jan. 2022, doi: [https://doi.org/10.1007/978-3-031-11435-9\\_33](https://doi.org/10.1007/978-3-031-11435-9_33).
- [43] S. N. Amin, "GAMIFICATION OF DUOLINGO IN RISING STUDENT'S ENGLISH LANGUAGE LEARNING MOTIVATION," *Jurnal Bahasa Lingua Scientia*, vol. 13, no. 2, pp. 191–213, Nov. 2021, doi: <https://doi.org/10.21274/ls.2021.13.2.191-213>.
- [44] A. Каримов, M. Saarela, and Tommi Kärkkäinen, "The impact of online educational platform on students' motivation and grades: the case of Khan Academy in the under-resourced communities," *Zenodo (CERN European Organization for Nuclear Research)*, Jul. 2023, doi: <https://doi.org/10.5281/zenodo.8115746>.
- [45] G. Y. Pratista, "Students' Perceptions of Using Class Dojo in a Teaching and Learning Process," *English Education: Journal of English Teaching and Research*, vol. 8, no. 1, pp. 77–90, May 2023, doi: <https://doi.org/10.29407/jetar.v8i1.19257>.
- [46] L. J. Tandey, "The Design and Development of Backend System for a Game Application," *Journal of Games, Game Art, and Gamification*, vol. 3, no. 1, pp. 14–20, Oct. 2021, doi: <https://doi.org/10.21512/jggag.v3i1.7230>.
- [47] R. J. Nadolski and H. G. K. Hummel, "Retrospective cognitive feedback for progress monitoring in serious games," *British Journal of Educational Technology*, vol. 48, no. 6, pp. 1368–1379, Aug. 2016, doi: <https://doi.org/10.1111/bjet.12503>.
- [48] D. Hooshyar, M. Yousefi, and H. Lim, "A systematic review of data-driven approaches in player modeling of educational games," *Artificial Intelligence Review*, vol. 52, no. 3, pp. 1997–2017, Dec. 2017, doi: <https://doi.org/10.1007/s10462-017-9609-8>.
- [49] A. De Santis *et al.*, "DASHBOARDS FOR ADMINS, TEACHERS, AND STUDENTS: TOOLS FOR MONITORING AND LEARNING," *EDULEARN Proceedings*, vol. 1, pp. 4666–4672, Jul. 2024, doi: <https://doi.org/10.21125/edulearn.2024.1150>.

- [50] P. Leitner, K. Maier, and M. Ebner, "Web Analytics as Extension for a Learning Analytics Dashboard of a Massive Open Online Platform," pp. 375–390, Jan. 2020, doi: [https://doi.org/10.1007/978-3-030-47392-1\\_19](https://doi.org/10.1007/978-3-030-47392-1_19).
- [51] A. Dwivedi, M. Dsouza, and A. Aggarwal, "AI-Driven School Management System: A React-Based Web Application Enhancing Educational Administration and Student Performance Analytics," *International Journal of Innovative Science and Research Technology (IJISRT)*, pp. 3042–3054, May 2024, doi: <https://doi.org/10.38124/ijisrt/ijisrt24apr2465>.
- [52] S. Pokhrel and Shiv Raj Banjade, "AI Content Generation Technology based on Open AI Language Model," *December 2023*, vol. 5, no. 4, pp. 534–548, Dec. 2023, doi: <https://doi.org/10.36548/jaichn.2023.4.006>
- [53] P. Kumar, S. Manikandan, and R. Kishore, "AI-Driven Text Generation: A Novel GPT-Based Approach for Automated Content Creation," pp. 1–6, Apr. 2024, doi: <https://doi.org/10.1109/icnwc60771.2024.10537562>
- [54] Tae Won Kim, "Application of artificial intelligence chatbot, including ChatGPT in education, scholarly work, programming, and content generation and its prospects: a narrative review," *Journal of Educational Evaluation for Health Professions*, vol. 20, pp. 38–38, Dec. 2023, doi: <https://doi.org/10.3352/jeehp.2023.20.38>
- [55] M. Larsson and M. Stigellid, "Energy efficient data synchronization in mobile applications. A comparison between different data synchronization techniques," *Handle.net*, 2015. <https://hdl.handle.net/20.500.12380/223406> (accessed Jun. 09, 2025).
- [56] Y. P. Faniband, I. Ishak, F. Sidi, and M. A., "A Review of Data Synchronization and Consistency Frameworks for Mobile Cloud Applications," *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 12, 2018, doi: <https://doi.org/10.14569/ijacsa.2018.091284>
- [57] A. Z. Ayezabu, "Supabase vs Firebase : Evaluation of performance and development of Progressive Web Apps," [www.theseus.fi](http://www.theseus.fi), 2022. <https://www.theseus.fi/handle/10024/771009>
- [58] Y. Luo, Y. Zhang, Z. Liu, G. Lai, Y. Wen, and W. Gao, "An Efficient Data Synchronization Tool Design and Implementation Based on Data Flow," pp. 1–5, Mar. 2025, doi: <https://doi.org/10.1109/csecs64665.2025.11010010>

## Appendices

### Appendix A: Roles and Responsibilities

*Table 24. Roles and Responsibilities*

TEAM MEMBER	ROLE	RESPONSIBILITIES
Kelly Dumbrique	Project Manager	<ul style="list-style-type: none"> <li>- Organizes tasks</li> <li>- Oversees progress</li> <li>- Sets deadlines</li> <li>- Manage timeline and roadmap</li> <li>- Lead team meetings</li> </ul>
	Liaison Officer	<ul style="list-style-type: none"> <li>- Communicates and coordinates with DOST-STII representatives</li> <li>- Schedule and document meetings</li> <li>- Relay feedback</li> <li>- Clarify concerns</li> </ul>
	QA Lead	<ul style="list-style-type: none"> <li>- Write test cases</li> <li>- Plan and oversee testing activities</li> <li>- Assign tasks to testers</li> <li>- Review test cases and results</li> </ul>
Arcielle Marie Gercan	UI/UX Designer	<ul style="list-style-type: none"> <li>- Creates high-fidelity wireframes</li> <li>- Creates design assets and animations in Figma</li> </ul>
	Backend Developer	<ul style="list-style-type: none"> <li>- Build APIs</li> <li>- Manage database</li> <li>- Ensure secure data handling</li> </ul>
	QA Tester	<ul style="list-style-type: none"> <li>- Write test cases</li> <li>- Perform manual/automated testing</li> <li>- Report and verify bugs</li> </ul>
Shandrae Lois Quianzon	UI/UX Designer	<ul style="list-style-type: none"> <li>- Creates high-fidelity wireframes</li> <li>- Creates design assets and animations in Figma</li> </ul>
	Frontend Developer	<ul style="list-style-type: none"> <li>- Translate Figma to code</li> <li>- Ensure responsive design</li> <li>- Handle user interactions</li> </ul>
	QA Tester	<ul style="list-style-type: none"> <li>- Write test cases</li> <li>- Perform manual/automated testing</li> <li>- Report and verify bugs</li> </ul>

Janice Maxene Salipande	UI/UX Designer	<ul style="list-style-type: none"> <li>- Creates high-fidelity wireframes</li> <li>- Creates design assets and animations in Figma</li> </ul>
	Documentation Lead	<ul style="list-style-type: none"> <li>- Handles all written outputs including reports, use cases, and final documents</li> <li>- Compile appendices</li> <li>- Maintain consistency in writing</li> </ul>
	QA Tester	<ul style="list-style-type: none"> <li>- Write test cases</li> <li>- Perform manual/automated testing</li> <li>- Report and verify bugs</li> </ul>

Table 24 outlines the current roles and responsibilities of each team member. Since we are still in the early stages and have not yet entered the development phase, the roles—especially those related to coding, testing, and deployment—are subject to change as the project progresses.



## Appendix B: Minutes of the Meetings

### *Meeting Minutes*

Meeting with DOST

Date: May 9, 2025

Discussion:

Miss Mavie

Sir McGyver

Miss Ella

Sir John – Web App Developer

### **Clarification with the Application**

- Before per question now by Quest
- Include Mini-Quest
- Whiz Memory Match, Whiz Puzzle Games, Whiz Battles
- The contents are from DOST, but the system design is us
- Login
  - Possible to have Basic login (Guest Name, Gender)
  - Password is for badges and for them if babalik balik sila
- Real-time Battles is for quizzes, but change the interface (Multiplayer Mode), NoSql
- Have inspiration with quiz games mobile apps
- AI to be discussed later because it needs a high type of hardware
- Stand Alone Server
- OS - Windows 10 or latest
- Framework Laravel – Backend Vuejs (previous starbooks) or Flutter(more prefer) for Frontend
- Built-in Admin Dashboard approved
- Local Wifi multiplayer game
- Offline (websocket) Online (NoSql)
- By category, random question, with points, leaderboard
- Memory Match (DOST Logos), Puzzles (Pictures or missing words)
- Battles and Quiz will be Question type

- Kahoot type for Battle Games, every end of question will be a leaderboard
- Consider Race Type design of Leaderboard for Battles
- Leaderboard for the Main App is already okay with list type?
- Basic Requirements Games – Memory Match, Puzzle Games, Quiz Type of Game, Battles
- Disregard the Grade Level Difficulty

### Meeting Minutes

ACTIVITY DETAILS			
Control Number	NEXUS - MOM05	Meeting Agenda	Documentation
Meeting Date	01 MAY 2025	Venue/Platform:	MS Teams
Start Time	8:00 PM	End Time	9:05 PM
Attendees			
Kelly Dumbrique		Shandrae Lois Quianzon	
Arcielle Marie Gercan		Janice Maxene Salipande	
Absent			

The screenshot shows a Microsoft Teams meeting interface. On the left, a SharePoint page titled 'Product Vision' is displayed. The page content includes:

- Product Title:** STARBOOKS Whiz Challenge (starbookswhez.ph)
- Project Description:** The STARBOOKS Whiz Challenge is an offline-capable, gamified educational application designed for kiosks that visitors will play during exhibits and events of DOST. This may include various quests such as Whiz Memory Match, Whiz Puzzle Games, Whiz Battles, and more. The game will also feature Whiz Rewards, allowing players to win special prizes through a leaderboard.
- What makes it stand out?** Unlike typical quizzes or trivia platforms, the STARBOOKS Whiz Challenge is specifically designed for educational exhibits organized or supported by the DOST. Its primary aim is to engage visitors through gamified learning focused on Science and Mathematics, making

On the right side of the Teams window, a vertical list of participants is shown with their profile pictures and names: Janice Maxene Salipande, Shandrae Lois Quianzon, Arcielle Marie Gercan, and Kelly Dumbrique.

PROCEEDINGS			
Agenda	Discussion Notes	Action Item	Person-in-charge
Product Documentation	Missing: <ul style="list-style-type: none"> <li>• Use Case Diagram</li> <li>• Roadmap</li> <li>• Kanban Board</li> <li>• Design Thinking</li> </ul>		
	Clarification on deliverable	Needed: <ul style="list-style-type: none"> <li>• Login System</li> </ul>	All

	<ul style="list-style-type: none"> <li>Priority on the deliverables wanted by client</li> </ul>	<ul style="list-style-type: none"> <li>Admin System</li> <li>Basic Structure</li> <li>Leaderboard</li> </ul> Client wants: <ul style="list-style-type: none"> <li>Kiosk</li> <li>Offline game</li> <li>Target market (public schools with little to no resources)</li> </ul>	
	Priority of the client / team <ul style="list-style-type: none"> <li>Games or Quiz</li> </ul>	Improve interface	
<b>System development</b>	Do we add AI Features? <ul style="list-style-type: none"> <li>Yes</li> </ul>	Text to Speech Natural Language Feedback AI Driven Insights (Admin)	Janice
	Centralized data center for tracking (admin) <ul style="list-style-type: none"> <li>Not yet developed</li> </ul>	To be done by client side Do not prioritize	Arcielle
<b>Functionality</b>	Level-based Gamified Learning	Students/users can choose difficulty	Kelly
<b>Use Case Diagram</b>	Does it need to be per feature <ul style="list-style-type: none"> <li>Yes (needs to be detailed)</li> </ul>	Make on lucid chart	Shandrae & Janice
<b>Roadmap</b>	How to make <ul style="list-style-type: none"> <li>Based on the school calendar</li> <li>Make it feasible</li> <li>Quarter Goals</li> </ul>	Do in Canva	Shandrae
	Q1	Gather requirements (May)	
	Q2	Design	
	Q3	Development	
	Q4	Test & Deployment	
<b>Board</b>	We'll use Kanban Board	Make use of planner feature of teams	Kelly
<b>Cost-Benefit Analysis</b>	No set price	Hardware systems are provided by client side	Kelly
		No specified price	
<b>UI/UX</b>		Login/Register	Arcielle & Shandrae
		Profile Settings	
		Homepage	
		Quiz Game Interface	
		Leaderboard	

## Meeting Minutes

ACTIVITY DETAILS			
Control Number	NEXUS - MOM06	Meeting Agenda	Pre-Presentation Planning
Meeting Date	03 MAY 2025	Venue/Platform:	MS Teams
Start Time	8:00 PM	End Time	8:35 PM
Attendees			
Kelly Dumbrique		Shandrae Lois Quianzon	
Arcielle Marie Gercan		Janice Maxene Salipande	
Absent			

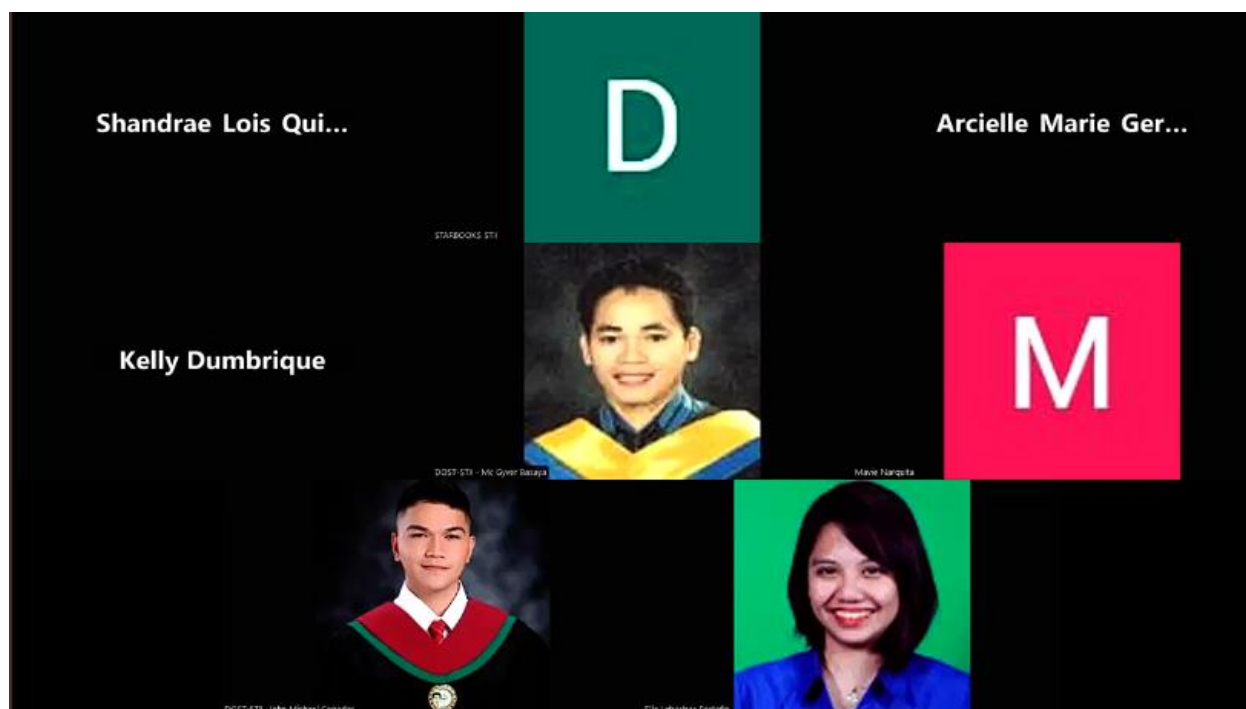
The screenshot shows a Microsoft Word document titled "Business Requirements" for the "NEXUS" project. The document is displayed in a web browser window (Safari) with the URL "asiapacificcollege-my.sharepoint.com". The document content includes a list of requirements for an interactive kiosk app. The requirements are categorized into Functional and Non-Functional Requirements. The Functional Requirements list includes: As a student, I want to play interactive quizzes on Science and Math topics, so I can learn in a fun, engaging way; As a player, I want to earn stars and unlock badges, so I feel rewarded and motivated to continue playing; As a visitor, I want to access the quiz easily on the kiosk without needing to sign in online, so I can quickly participate during exhibits; As an admin, I want to add, edit, and delete quiz questions within the kiosk app, so the content stays updated and relevant; As a teacher, I want to view the leaderboard to see how students are performing in real-time; As a player, I want to claim physical rewards after earning 10 stars, so I feel incentivized for participating. The Non-Functional Requirements list includes: As a system maintainer, I want the application to work offline on kiosks, so it can be used even in remote areas without internet access; As a user, I want the interface to be intuitive and visually engaging, so I can navigate and play without confusion or external help; As an IT administrator, I want user progress to be securely stored locally, ensuring data security.

DISCUSSIONS			
Agenda	Discussion Notes	Action Item	Person-in-charge
Documentation Review	Project Summary <ul style="list-style-type: none"> <li>Offline</li> <li>Features</li> <li>What makes it stand out</li> </ul>		
	Deliverables <ul style="list-style-type: none"> <li>Interactive kiosk app</li> <li>Games: we recommend games</li> </ul>		
	UI/UX Assets	To be asked in next meeting with DOST	All
	Admin Panel <ul style="list-style-type: none"> <li>Per kiosk for now</li> </ul>		

UI/UX Wireframe	To add in document	Waiting for the final design to be done	Arcielle and Shandraae
		Add in document after	Janice
Audience	Primary <ul style="list-style-type: none"> <li>Educational exhibits with DOST</li> </ul>		
	Secondary <ul style="list-style-type: none"> <li>Schools (rural learning centers)</li> </ul>		
Target Customer	DOST-STII		
	Event Organizers & Educational Institutions		
	LGUs		
Target User	Exhibit Visitors		
	Students		
	Teachers		
	Admins		
Summary		Introduction – Deliverable	Kelly
		UI/UX Wireframe – Environmental Scan	Arcielle
		User stories – CostBenefit Analysis	Shandrae
		Design Thinking – End	Janice

### Meeting Minutes

ACTIVITY DETAILS			
Control Number	NEXUS - MOM08	Meeting Agenda	DOST Follow-up Meeting
Meeting Date	09 MAY 2025	Venue/Platform:	Zoom
Start Time	2:00 PM	End Time	3:00 PM
Attendees			
Kelly Dumbrique		Shandrae Lois Quianzon	
Arcielle Marie Gercan		Mc Gyver Basaya	
Mavie Narquita		John Michael Cagadas	
Elle Labasbas Pestano			
Absent			
Janice Maxene Salipande			



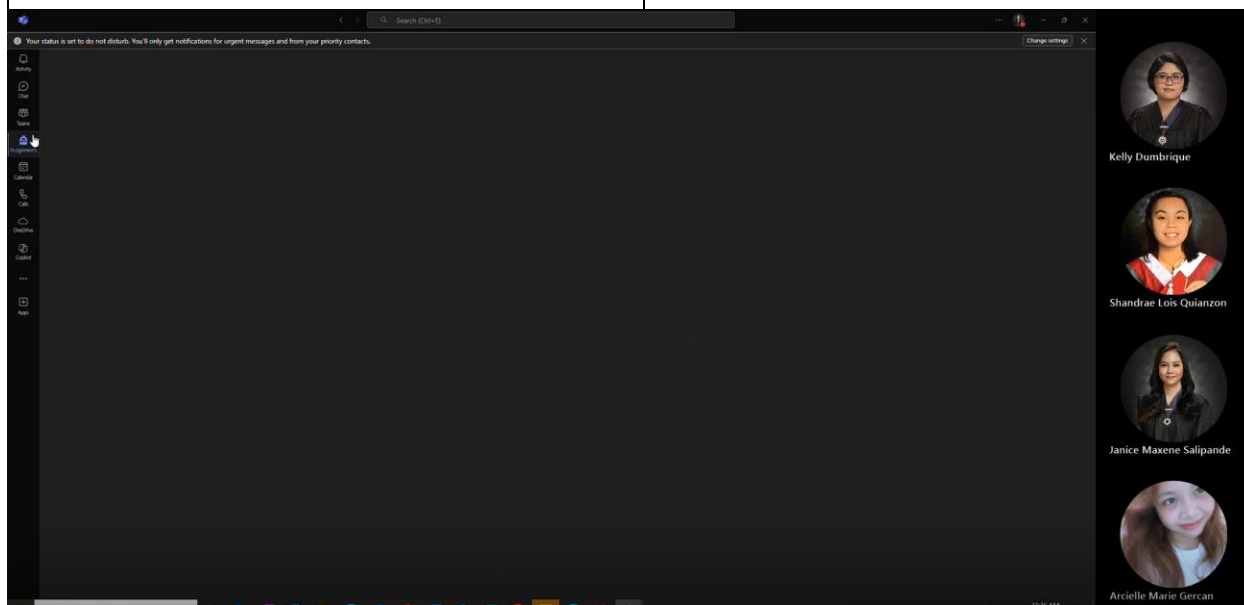
## DISCUSSIONS

Agenda	Discussion Notes	Action Item	Person-in-charge
Clarifications	Content Direction *documentation – system development <ul style="list-style-type: none"> <li>Games we (team) will develop initial output of games (modular)</li> <li>Leaderboard</li> </ul>		
	Registration/login <ul style="list-style-type: none"> <li>User pass needed because users can go back to kiosk to continue progress</li> <li>M/F needed for data of DOST</li> <li>Email not needed</li> </ul>		
	User Engagement <ul style="list-style-type: none"> <li>Quiz is just one type of game</li> <li>Whoever is top in leaderboard will win at the end (exhibit)</li> </ul>		

	<ul style="list-style-type: none"> <li>By topic/category (games)</li> </ul>		
	Kiosk Hardware <ul style="list-style-type: none"> <li>How they install on kiosk (Not USB, file size is 400MB, used XAMPP)</li> <li>AI problem (too big data, might not fit the system/unit)</li> <li>OS (Atleast Windows10)</li> <li>Framework (Laravel, UJS (Flutter is recommended))</li> </ul>		
	Built-in Admin <ul style="list-style-type: none"> <li>Admin dashboard, not data center</li> <li>The data center will be made and synced by DOST-STII</li> </ul>		
	UI/UX Assets <ul style="list-style-type: none"> <li>Will be sent for our reference</li> <li>They used vue.js</li> </ul>		
	Theme <ul style="list-style-type: none"> <li>Change our theme (wizard)</li> </ul>		
	Development <ul style="list-style-type: none"> <li>Just add the battles (duels) after</li> <li>Dual player can be done using local wi-fi (web socket)</li> </ul>		
	User Pains from their current system <ul style="list-style-type: none"> <li>Client does not have it in detail</li> <li>Difficult Questions from average and difficult modes</li> </ul>	Questions are by category Random Questions Point system	TBA

### Meeting Minutes

ACTIVITY DETAILS			
Control Number	NEXUS – MOM11	Meeting Agenda	Deliverables Alignment Meeting
Meeting Date	22 MAY 2025	Venue/Platform:	MS Teams
Start Time	11:30 AM	End Time	12:20 PM
Attendees			
Kelly Dumbrique		Shandrae Lois Quianzon	
Arcielle Marie Gercan		Janice Maxene Salipande	
Absent			



DISCUSSIONS			
Agenda	Discussion Notes	Action Item	Person-in-charge
Release Plan	V1 – don't prioritize category topics <ul style="list-style-type: none"> <li>- Focus on easy, average, difficult modes</li> </ul>		
	Whiz Battle <ul style="list-style-type: none"> <li>- Kahoot gameplay</li> </ul>		
	<ul style="list-style-type: none"> <li>- Dates (no released school calendar yet)</li> </ul> Projected dates: <ul style="list-style-type: none"> <li>- July – September (T1)</li> <li>- October – December (T2)</li> <li>- February – March/April (T3)</li> </ul>		



Promotional Ad	1 minute max		Arcielle
	Admin Interface doesn't need to show		
		Script	Janice

### Meeting Minutes

ACTIVITY DETAILS			
Control Number	NEXUS – MOM12	Meeting Agenda	Finals Connsultation
Meeting Date	03 JUNE 2025	Venue/Platform:	MS Teams
Start Time	5:00 PM	End Time	5:45 PM
Attendees			
Kelly Dumbrique		Shandrae Lois Quianzon	
Arcielle Marie Gercan		Janice Maxene Salipande	
Sir Jose Eugenio Queseda			
Absent			

DISCUSSIONS			
Agenda	Discussion Notes	Action Item	Person-in-charge
Project Context	Presented documentation and sample demo of the current system	Finalize AI feature set and define its scope (admin assistance, question generation, etc.)	All
	Discussed concern with 1v1 Battle Mode and its current feasibility	Research solutions for syncing offline data (look into Supabase, MongoDB with offline support)	

	Outlined the AI feature concept — including text-to-speech and AI-generated questions for admins	Review 1v1 Battle Mode logic and implementation challenges	
	Emphasis on syncing offline kiosk data to the cloud when back online		
<b>AI Feature Suggestions</b>	AI-assisted admin panel (similar to MS Teams AI)	Shortlist feasible AI features for prototype	
	Text-to-Speech functionality for accessibility	Explore Python NLP libraries (e.g., GPT-based API or Hugging Face) for text generation	
	AI-generated question banks for games/quizzes	Consider open-source TTS tools (e.g., gTTS or Coqui.ai)	
<b>Tech Stack Suggestions</b>	Python-based stack preferred due to familiarity and easier backend integration	Create proof of concept using Django + Supabase	
	Django (Backend) + Flutter (Frontend) + Supabase (Realtime DB/Auth)	Test Supabase's sync reliability in offline/online scenarios	TBA
	Offline-first and real-time sync capabilities are essential	Compare with Firebase or MongoDB Realm as alternatives	
<b>Timeline &amp; Milestones</b>	Only a best estimate for now — flexibility expected	Draft timeline with buffer periods and breakpoints	
	Second half of the term will focus on prototype development	Focus on diagrams and tech familiarization in early weeks	Kelly & Janice
	Initial focus on system design and documentation (Use Case Diagrams, Analysis)	Assign members to begin learning and experimenting with chosen tech stack	Kelly
<b>Use Case Diagram</b>	Keep diagrams simple to avoid clutter	Identify and draft only core use cases for the main diagram	
	Fully dressed use cases should include: <ul style="list-style-type: none"> <li>• Case ID</li> <li>• Use Case Name</li> <li>• Date Created / Modified / Updated</li> <li>• Detailed descriptions &amp; flows</li> </ul>	Prepare detailed use case specs separately for documentation	Janice

		Schedule a peer review of diagrams for clarity and correctness	Kelly
<b>Data Handling &amp; Synchronization</b>	Key concern: handling login/registration without online access	Research cloud-based real-time databases with offline sync (Supabase, Firebase, MongoDB Realm)	Arcielle
	Must sync data from multiple kiosks to a central cloud system when reconnected	Define sync logic (e.g., queue updates offline, push on reconnect)	
		Prototype offline registration and login process	TBA

## **Appendix C: Methodology**

This project followed a phased, iterative methodology to ensure structured development, continuous stakeholder involvement, and alignment with the goals of the STARBOOKS Whiz Challenge enhancement. The process was guided by the Software Development Life Cycle (SDLC), adapted to suit the needs of a student-led project. Each phase is described below:

### **1. Planning and Requirements Gathering**

We began by identifying the system's gaps and understanding the needs of both players and administrators. Regular consultations were conducted with DOST-STII to clarify existing issues, discuss desired features, and determine the technical limitations of the current system. Key deliverables included the problem statement, objectives, scope, and initial system concept.

### **2. Analysis and Documentation**

This phase focused on creating detailed documentation of the system requirements. We developed use case diagrams, full use case descriptions (including both player-side and admin-side functionalities), and functional requirements. We also identified priority features, actors, user roles, and core game flows.

### **3. Design Phase (Current Phase)**

We are currently in the design phase, developing high-fidelity wireframes for the player interface and admin dashboard using Figma. These wireframes reflect finalized features, game modes, and analytics components. We also finalized our proposed tech stack (Django for backend, Flutter for frontend, Supabase for database and real-time features).

### **4. Development Phase (Upcoming)**

Once designs are finalized and approved, the actual development will begin. Features will be built in iterations, starting with the player-side game modules followed by the admin dashboard. Testing will be done alongside development to identify bugs early.

### **5. Testing and Quality Assurance**

We plan to conduct both unit testing and user acceptance testing. Feedback from DOST-STII and selected testers will be gathered to ensure that the system functions as intended and aligns with user expectations.

### **6. Deployment and Handover**

The final system will be deployed to target STARBOOKS exhibit locations. A user manual and system documentation will be created for DOST-STII to support long-term use and maintenance.

## **7. Maintenance and Feedback**

After deployment, we will gather feedback and implement any necessary adjustments. While limited due to the scope of a project, we aim to leave the system in a maintainable and well-documented state for future enhancements.

## **Appendix D: Project Sharepoint Link**

[NEXUS - Sharepoint](#)

## Appendix E: Requirements Traceability Matrix

Table 25. Requirements Traceability Matrix

REQUIREMENT ID	REQUIREMENT DESCRIPTION	USE CASE ID	TEST CASE ID
BR-01	Player can register to the application.	UC-01	TC-01
BR-02	Player can log in using their username and password.	UC-02	TC-02
BR-03	Player can edit their profile information.	UC-03	TC-03
BR-04	Player can play Whiz Challenge (solo quiz).	UC-04	TC-04
BR-05	Player can view and claim earned rewards.	UC-05	TC-05
BR-06	Player can play Whiz Memory Match.	UC-06	TC-06
BR-07	Player can play Whiz Puzzle.	UC-07	TC-07
BR-08	Player can play Whiz Battle.	UC-08	TC-08
BR-09	Player can view game statistics.	UC-09	TC-09
BR-10	Player can log out from the system.	UC-10	TC-10
BR-11	Admin can log in using valid credentials.	UC-02	TC-11
BR-12	Admin can manage player profiles.	UC-12	TC-12
BR-13	Admin can manage the list of admins.	UC-13	TC-13
BR-14	Admin can manage quiz content (questions).	UC-14	TC-14
BR-15	Admin can manage difficulty settings for quiz.	UC-15	TC-15
BR-16	Admin can manage player badges and rewards.	UC-16	TC-16
BR-17	Admin can view analytics dashboard with game statistics.	UC-17	TC-17

BR-18	Admin can log out from the system.	UC-10	TC-18
-------	------------------------------------	-------	-------

Table 25 illustrates the relationship between the project's requirements and their corresponding design elements, development tasks, and testing cases. It ensures that all requirements are addressed throughout the development process, helping track progress and maintain alignment between stakeholders' needs and the final product.



## Appendix F: RACI Matrix

Table 26. RACI Matrix

TASK	PROJECT MANAGER	UI/UX DESIGNER	BACKEND DEVELOPER	FRONTEND DEVELOPER	QA/TESTER	LIAISON TO DOST-STII
Define project scope and objectives	A	C	C	C	I	R
Requirements gathering	R	C	C	C	I	A
Create use case diagrams & documentation	R	C	C	C	I	C
Design wireframes and mockups	I	R	C	C	I	C
Review and finalize system design	A	R	C	C	C	C
Select and finalize tech stack	R	C	A	A	I	C
Backend development	I	I	R	C	I	I
Frontend development	I	C	C	R	I	I
API integration	I	I	R	R	I	I
Database design and setup	I	I	A	C	I	I
Implement gamification features	I	C	C	R	C	I
Create admin dashboard	I	C	R	R	I	C
Develop player-side functionalities	I	C	R	R	I	I
Conduct functional testing	I	I	C	C	R	I
Conduct user testing (w/ DOST and target users)	I	I	I	I	R	A
Fix bugs and polish based on feedback	I	C	R	R	R	I
Prepare deployment build	R	I	A	A	C	I
Deployment and installation	A	I	R	R	I	C
Final demo to DOST and documentation submission	A	I	C	C	C	R

Legend:

R	Responsible: Does the work
A	Accountable: Approves the work and ensures task completion
C	Consulted: Offers input or guidance
I	Informed: Kept in the loop

Table 26 defines the roles and responsibilities of team members involved in the STARBOOKS Whiz Challenge project. It clarifies who is Responsible, Accountable, Consulted, and Informed for each major task or deliverable, promoting clear communication and efficient project management.