

**AGILE HERO: AN INTERACTIVE PLATFORM TO FACILITATE  
AGILE AND SCRUM TRAINING**

**Agile Mini Games**

**BY**

**MR. HAO HUANG 6488121**

**ADVISOR**

**ASST. PROF. DR. MORAKOT CHOETKIERTIKUL**

**CO-ADVISOR**

**LECT. DR. PISIT PRAIWATTANA**

**A Senior Project Submitted in Partial Fulfillment of the Requirements for  
THE DEGREE OF BACHELOR OF SCIENCE  
(INFORMATION AND COMMUNICATION TECHNOLOGY)**

**Faculty of Information and Communication Technology  
Mahidol University  
2024**

**COPYRIGHT OF MAHIDOL UNIVERSITY**

## **ACKNOWLEDGMENTS**

The success of this senior project would not have been possible without the guidance and support of many individuals.

First and foremost, we extend our heartfelt gratitude to our advisor, Asst. Prof. Dr. Morakot Choetkiertikul, for their invaluable wisdom, constructive feedback, and encouragement throughout the project. We are also deeply thankful to our co-advisor, Lect. Dr. Pisit Praiwattana, whose insightful advice and expertise greatly enhanced the quality of our work.

Additionally, we are grateful to our classmates and peers who contributed through collaborative efforts, brainstorming sessions, and moral support. Their involvement has been instrumental in our project's development. We also thank our friends and families for their unwavering encouragement and understanding during the demanding phases of this project. Their support was a constant source of strength and motivation.

Finally, we acknowledge the academic community and institutions that provided the necessary resources and research materials critical to our success. This project's realization is a testament to the collective efforts of everyone involved, and we express our sincere gratitude for their contribution to this journey.

## **Abstract**

The **Abstract** Agile and Scrum training activities are often conducted through physical workshops and games to enhance team collaboration and understanding of Agile principles. However, such methods can be difficult to scale, monitor, and engage participants in remote or digital environments.

Existing training tools are mostly either manual or require extensive setup, and many lack real-time interaction or visual engagement. To address these challenges, we developed a set of multiplayer online games designed for Scrum training, including the Coin Game, Ball Point Game, and Drawing Together Game.

Our solution simulates key Scrum practices such as iteration, communication, and task handover through interactive browser-based gameplay. The system enables real-time synchronization via WebSocket and supports both collaborative play and facilitator control. The platform improves accessibility, scalability, and participant engagement in Agile education.

## CONTENTS

	Page
<b>1 INTRODUCTION .....</b>	<b>1</b>
1.1 BACKGROUND AND MOTIVATION .....	1
1.2 PROBLEM STATEMENT .....	1
1.3 PROJECT OBJECTIVE .....	2
1.4 SCOPE OF THE PROJECT .....	3
1.5 EXPECTED CONTRIBUTIONS .....	4
1.5.1 FOR AGILE TRAINER .....	4
1.5.2 FOR AGILE TRAINEE .....	5
1.6 ORGANIZATION OF THE DOCUMENT .....	5
<b>2 BACKGROUND .....</b>	<b>6</b>
2.1 AGILE METHODOLOGY .....	6
2.1.1 SCRUM FRAMEWORK .....	6
2.1.2 IMPORTANCE OF AGILE EDUCATION AND TRAINING .....	7
2.2 AGILE GAMES FOR EXPERIENTIAL LEARNING .....	7
2.2.1 BALL POINT GAME .....	7
2.2.2 COIN GAME .....	7
2.2.3 DRAWING TOGETHER GAME .....	8
2.2.4 LIMITATIONS OF TRADITIONAL AGILE GAMES .....	8
2.3 GAMIFIED DIGITAL AGILE TRAINING .....	8
2.3.1 AGILE HERO: A DIGITAL TRAINING PLATFORM .....	8
2.4 TECHNOLOGIES USED IN THE PROJECT .....	9
2.4.1 FRONTEND TECHNOLOGIES .....	9
2.4.2 WEBSOCKET COMMUNICATION .....	9
2.4.3 BACKEND (COIN GAME ONLY) .....	9
2.4.4 VERSION CONTROL AND DEPLOYMENT .....	9
2.5 CHAPTER SUMMARY .....	10

	3
<b>3 GAME DESIGN</b>	11
3.1 OVERVIEW OF GAME SYSTEM ARCHITECTURE AND MECHANISM	11
3.2 GAME DETAILS	13
3.2.1 AGILE COIN GAME	13
3.2.2 AGILE BALL POINT GAME	13
3.2.3 DRAWING TOGETHER GAME	14
3.3 STORYBOARD	14
3.3.1 AGILE COIN GAME	14
3.3.2 BALL POINT GAME	15
3.3.3 DRAWING TOGETHER GAME	16
3.4 WINNING CONDITIONS	17
3.5 GAMEPLAY EXPLANATION + UI CONTROLS	17
3.6 CONFIGURATION	17
3.7 TECHNICAL ARCHITECTURE	17
<b>4 EVALUATION</b>	20
4.1 FUNCTIONALITY EVALUATION	20
4.1.1 TEST ENVIRONMENT	20
4.1.2 AGILE COIN GAME	20
4.1.3 BALL POINT GAME	20
4.1.4 DRAWING TOGETHER GAME	21
4.2 USER EVALUATION	21
4.2.1 PARTICIPANTS	21
4.2.2 EVALUATION METRICS	22
4.2.3 RESULTS AND ANALYSIS	22
4.2.4 LIMITATIONS	22
4.2.5 FEEDBACK SUMMARY AND RECOMMENDATIONS	23
4.2.6 CONCLUSION	24
<b>5 CONCLUSION</b>	25
REFERENCES	27
APPENDIX A	29



## **CHAPTER 1**

### **INTRODUCTION**

#### **1.1 Background and Motivation**

Agile methodologies, particularly Scrum, have gained widespread adoption in modern software engineering due to their focus on delivering value through iterative development, adaptive planning, and frequent stakeholder feedback [1, 2]. Compared to traditional methodologies like the Waterfall model [3], Agile promotes a mindset of collaboration, responsiveness to change, and continuous improvement.

One of the key aspects of Agile success lies in proper education and team alignment. To ensure that Agile values such as communication, transparency, and adaptability are understood and practiced, many training programs incorporate hands on activities. Physical Agile training games like the Ball Point Game and Coin Game have proven effective in helping teams internalize Agile principles in an engaging and memorable way [4].

These activities create experiential learning environments where teams can observe how changes in communication and process affect performance. However, these traditional games require physical presence, props (e.g., balls, coins, whiteboards), and experienced facilitators. As work environments shift toward hybrid and remote models, these physical setups become impractical or impossible to conduct [5].

Additionally, younger generations of learners increasingly prefer digital-first platforms that are interactive and gamified. The current gap in widely available, engaging, and scalable digital Agile training tools presents an opportunity to reimagine how Agile education is delivered in both academic and industry contexts.

#### **1.2 Problem Statement**

Although traditional Agile training methods have demonstrated pedagogical effectiveness, they are accompanied by several limitations [6, 7]:

- They rely heavily on physical interaction and materials, which are not feasible in remote or distributed teams.
- The quality of the learning experience can vary greatly depending on the facilitator's skill and group dynamics.
- They are time-consuming to set up and not easily repeatable across sessions or organizations.
- There is often no way to track player actions or measure outcomes for further reflection or improvement.

While some digital solutions such as Miro or Jamboard provide virtual collaboration boards, they do not offer structured, rule-driven gameplay or simulate Agile ceremonies and team roles in a meaningful way [8]. Most lack real-time synchronization, and even fewer integrate game mechanics that mirror Sprint planning, backlogs, retrospectives, or communication breakdowns.

This results in a learning experience that may fail to capture the core dynamics of Agile teams quick feedback cycles, task handoff efficiency, cross-functional interaction, and adaptive behavior. As Agile continues to be embraced outside of software development (e.g., education, marketing, manufacturing), there is a growing demand for educational tools that scale across disciplines and delivery modes.

### 1.3 Project Objective

The primary goal of this project is to develop a web-based, multiplayer game platform—**Agile Hero**—to support Agile and Scrum training through interactive gameplay. The system is designed to simulate real-world Agile scenarios in a digital environment, enabling both in-person and remote teams to participate actively in training sessions.

The platform consists of three mini-games:

- **Agile Coin Game:** Players simulate a task handoff pipeline to understand bottlenecks and improve team flow.
- **Ball Point Game:** A multi-round game designed to reinforce the inspect-and-adapt principle by measuring iterative improvements.



- **Drawing Together:** A role-based drawing game that simulates communication challenges between Product Owners and Developers.

Each game is carefully designed to align with one or more Agile values. They emphasize collaboration, self-organization, fast feedback, and delivering working outcomes in short iterations. The system aims to:

- Provide an engaging and gamified learning environment suitable for teams with different levels of Agile experience.
- Support real-time synchronization using WebSocket technology.
- Reduce reliance on physical materials or facilitators, enabling easy deployment in academic or enterprise settings.
- Collect basic metrics and logs that could support post-game debriefs and retrospectives.

#### 1.4 Scope of the Project

This project focuses on designing, developing, and evaluating three multiplayer Agile training games accessible via modern web browsers. The scope includes both the technical implementation and the pedagogical design of gameplay mechanics to ensure alignment with Scrum practices.

The system's scope includes the following:

- Developing the frontend using JavaScript, HTML, and CSS for responsiveness and interactivity.
- Implementing real-time multiplayer communication via WebSocket.
- Designing a RESTful backend using Python (FastAPI) for one game (Coin Game) that requires game state validation and session tracking.
- Creating a simple in memory session manager in the backend—no persistent database is used.

- Testing functionality using simulated scenarios such as “happy path,” “invalid action,” and “host disconnection.”
- Evaluating usability, engagement, and learning impact through user feedback and expert reviews.

## **1.5 Expected Contributions**

This project is expected to contribute in both technical and educational dimensions:

- A complete browser-based Agile training platform that supports real-time, multi-player gameplay.
- A set of three games representing different facets of Scrum (workflow, iteration, and communication).
- A modular and lightweight architecture using common web technologies (JavaScript + WebSocket + optional REST backend).
- Empirical findings from evaluations with real users and Agile experts, providing insights for further development or classroom adoption.
- A foundation for future gamified Agile training tools, adaptable to other methodologies like Kanban or SAFe.

There are several expected benefits from this project, as follows:

### **1.5.1 For Agile Trainer**

- This project will help Agile trainers conducting Agile training whom originally offered with only two methods which are on-site offline Agile activities and remote online Agile activities. The platform offer an additional option as Hybrid-Agile activities to conduct training for the Agile trainees while being onsite with the trainees but instead moderate through digital devices.

- This project created prototype of Agile platform which the Agile trainers can teach Agile values through group activities to any audience to catch the glimpse of content related to Agile before go learning deeper which the Agile Hero itself is readily available to play.

### **1.5.2 For Agile Trainee**

- This project will help Agile trainees to learn more about Agile and Scrum throughout the platform Agile Hero by training activities and training in the comprehensive level for trainee with interactive features.
- This project let the trainees practice Agile through the platform and enhance Agile value using key framework like Scrum.

## **1.6 Organization of the Document**

**The document consists of 6 chapters, as follows:**

- Chapter 1 Introduction
- Chapter 2 Background
- Chapter 3 Game Design
- Chapter 4 Evaluation
- Chapter 5 Conclusion

## **CHAPTER 2**

### **BACKGROUND**

This chapter provides the theoretical foundation and contextual background for the development of the Agile Hero platform. It outlines the evolution of Agile methodologies, the pedagogical role of Agile training games, the challenges with traditional training delivery, and how digital solutions specifically browser-based games offer a promising alternative. Additionally, the technologies used in the implementation of the platform are described.

#### **2.1 Agile Methodology**

Agile methodologies represent a shift from rigid, sequential software development processes to more flexible, people-centered approaches. The publication of the Agile Manifesto in 2001 by a group of industry experts marked a transformative period in software engineering [9]. The manifesto introduced four core values and twelve principles that prioritize communication, working software, and responsiveness to change over formal documentation and rigid planning.

Agile approaches emphasize continuous delivery of value, stakeholder collaboration, and the ability to adapt to feedback. These characteristics make Agile particularly useful in dynamic environments where requirements evolve rapidly [1]. Among Agile frameworks, Scrum has become one of the most widely adopted due to its simplicity and effectiveness in promoting iterative improvement.

##### **2.1.1 Scrum Framework**

Scrum is structured around short development cycles called Sprints, each typically lasting one to four weeks. Within each Sprint, a cross-functional Scrum Team delivers a potentially shippable product increment. The team comprises three key roles: the Product Owner, who defines the backlog and prioritizes tasks; the Developers, who execute the work; and the Scrum Master, who ensures that Scrum principles are followed

[10].

Scrum incorporates events such as Sprint Planning, Daily Scrum, Sprint Review, and Sprint Retrospective to enable frequent inspection and adaptation. These events promote transparency and continuous improvement, both of which are central to Agile values.

### **2.1.2 Importance of Agile Education and Training**

As Agile practices became mainstream, the demand for proper Agile training has grown. Effective Agile training is essential for ensuring that teams understand not only the mechanics of Agile, but also its mindset. It equips participants with the ability to apply Agile principles across real-world scenarios and team structures.

Traditional training programs often rely on formal certification, in-person workshops, and printed materials. While effective in some cases, these approaches can be costly, logistically demanding, and difficult to scale across geographies or large organizations [5, 11].

## **2.2 Agile Games for Experiential Learning**

Experiential learning, learning by playing a critical role in Agile training. By simulating real-world team dynamics and project constraints, games help participants internalize Agile values and practices more effectively than passive instruction alone.

### **2.2.1 Ball Point Game**

The Ball Point Game is a widely used Agile simulation that teaches process efficiency and self-organization. Participants are asked to pass balls through all team members as many times as possible within a fixed time frame. Over multiple iterations, the team is encouraged to reflect and adapt their strategy. This cycle mirrors the principles of iterative improvement and feedback in Scrum [12].

### **2.2.2 Coin Game**

The Coin Game is used to demonstrate the cost of context-switching and the importance of reducing work-in-progress. Players process coins in a sequential flow, experiencing firsthand how batch sizes and task switching affect throughput. This game

aligns with Lean principles and supports discussions on flow optimization.

### **2.2.3 Drawing Together Game**

Drawing Together Agile game simulate real-world communication challenges. For example, one player acts as a Product Owner and must describe an image, while the other player (Developer) attempts to recreate it based on verbal instructions. These exercises highlight the importance of clear communication, iterative clarification, and stakeholder feedback—core aspects of Agile team interaction.

### **2.2.4 Limitations of Traditional Agile Games**

Although physical games are useful in promoting Agile mindsets, they suffer from several drawbacks:

- Require physical props (balls, coins, Lego, etc.) and printed materials.
- Not suitable for remote or hybrid teams.
- Depend heavily on facilitator availability and skill.
- Offer limited opportunity for automated feedback, tracking, or replay.
- Challenging to scale across large organizations or distributed teams.

These challenges necessitate the development of digital alternatives that maintain interactivity while improving accessibility and scalability.

## **2.3 Gamified Digital Agile Training**

Gamification applies game design elements to non-game contexts to enhance engagement and motivation. In the context of Agile training, gamification can make abstract principles more tangible by embedding them into rules, challenges, and feedback systems.

### **2.3.1 Agile Hero: A Digital Training Platform**

The Agile Hero platform transforms traditional Agile games into interactive multiplayer browser experiences. It supports both remote and in-person play, enabling teams

to collaborate in real-time through WebSocket communication. Each game is built to simulate core Scrum activities and roles:

- **Agile Coin Game:** Simulates coin batching and workflow handoffs.
- **Ball Point Game:** Models process optimization across iterations.
- **Drawing Together:** Emphasizes PO-to-developer communication and feedback loops.

Players are required to coordinate, adapt, and improve over multiple rounds, providing a rich learning experience aligned with Agile values. The platform removes logistical constraints and allows training to scale across teams, organizations, or academic settings.

## 2.4 Technologies Used in the Project

### 2.4.1 Frontend Technologies

The platform's frontend is developed using HTML, CSS, and JavaScript. Game UI elements, animations, and timers are rendered client-side. For drawing interactions, HTML5 Canvas API is utilized.

### 2.4.2 WebSocket Communication

WebSocket is a bidirectional protocol that enables persistent real-time communication between clients and the server [13]. This is critical for maintaining synchronized game state across all players in a session.

### 2.4.3 Backend (Coin Game Only)

For the Coin Game, a backend is implemented using Python's FastAPI framework. It manages game room creation, player sessions, round transitions, and action validation. Game state is stored in memory, ensuring simplicity and low latency without external databases.

#### **2.4.4 Version Control and Deployment**

Version control enables collaboration and rollback during testing. The system is designed for lightweight deployment, allowing usage in classrooms, workshops, and virtual Agile coaching sessions.

### **2.5 Chapter Summary**

This chapter provided an overview of Agile principles and training practices, focusing on the use of games for experiential learning. It explained the rationale for transitioning from physical to digital Agile training tools and introduced the concept of the Agile Hero platform. By leveraging modern web technologies such as WebSocket and FastAPI, the platform enables scalable, interactive, and accessible Agile education. The next chapter will detail the game design and implementation structure of each training module.



## CHAPTER 3

### GAME DESIGN

This chapter presents the design of three multiplayer mini-games—Agile Coin Game, Ball Point Game, and Drawing Together—developed to support Agile training. Each game is tailored to simulate Agile practices such as communication, collaboration, and iteration.

The chapter outlines key game elements, including gameplay mechanics, UI, configurations, and their mapping to Scrum concepts. It also introduces the technical architecture, which combines JavaScript for frontend development and FastAPI (Python) for backend logic in the Coin Game. Real-time synchronization is achieved using Web-Socket, allowing smooth multiplayer interaction.

#### 3.1 Overview of Game System Architecture and Mechanism

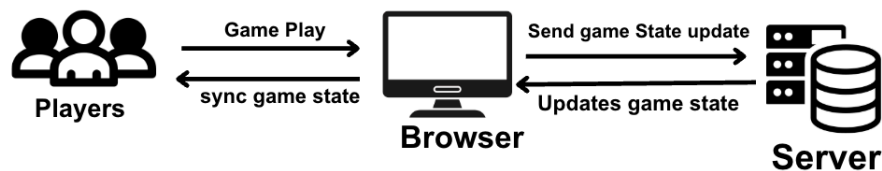


Figure 3.1: System Architecture

The System Architecture (Figure 3.1) shows that the three games developed under the Agile Hero platform share a common multiplayer structure but differ slightly in their implementation technologies:

- **Online Multiplayer Coin Game** uses a full-stack architecture:
  - Frontend: JavaScript with HTML/CSS
  - Backend: Python (FastAPI) for REST API and WebSocket handling
  - Backend manages player roles, coin transfer status, and round tracking

- **Ball Point Game and Drawing Together** are implemented as frontend-only applications:
  - Frontend: React-based SPA with local game logic
  - WebSocket for real-time communication
  - Game state is managed within the browser session

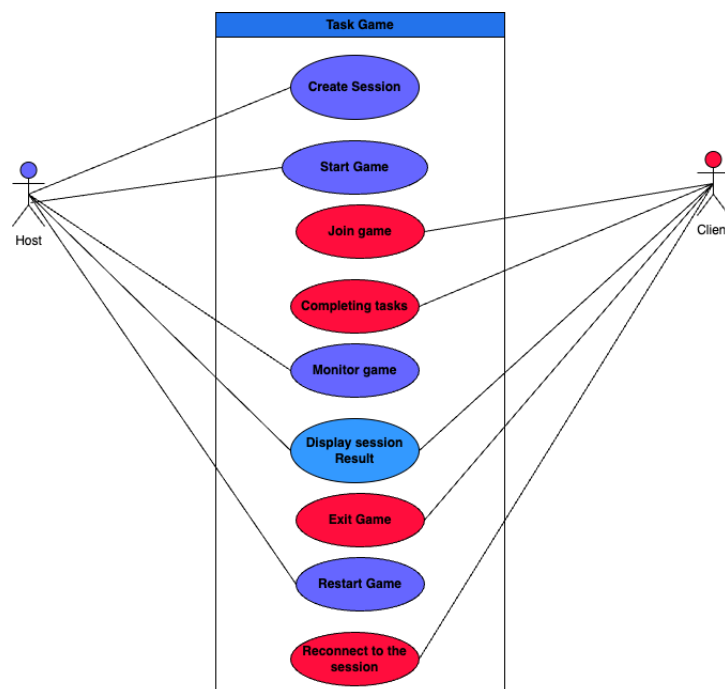


Figure 3.2: Host and Client Use Case

The Use Case diagram (Figure 3.2) shows the interaction between host and client systems. The host controls the game logic, while clients participate through interaction.

- The platform supports synchronous multiplayer gameplay over LAN or the Internet.
- Each game is structured around key Scrum values: collaboration, iteration, feedback.
- All games operate in rounds, with each round enforcing timing, handover, and synchronization rules.

## 3.2 Game Details

### 3.2.1 Agile Coin Game

**Objective:** Maximize delivery efficiency and minimize delays through process optimization.

**Mechanics:** Players form a virtual circle. Coins are passed from one player to the next based on predefined batch sizes. The game consists of four rounds, each simulating different batching strategies to illustrate the impact on flow efficiency.

**How to Play:**

- The game consists of four rounds, each involving 20 virtual coins:
  1. **Round 1:** All 20 coins are passed as a single batch.
  2. **Round 2:** Coins are passed in two batches of 10 coins each.
  3. **Round 3:** Coins are passed in four batches of 5 coins each.
  4. **Round 4:** Coins are passed in ten batches of 2 coins each.
- Players click the “Complete this batch” button to send coins to the next player.
- Each player can pass coins per batch after receiving them.
- The backend records timestamps and calculates flow time and delays.
- At the end of each round, the game displays metrics to encourage team reflection.

### 3.2.2 Agile Ball Point Game

**Objective:** Increase the number of points scored per round through process optimization.

**Mechanics:** Players pass a virtual “ball” between them. Each completed loop scores a point. Teams are encouraged to inspect and adapt strategy.

**How to Play:**

- Each player must pass a ball according to the ball’s color.
- the ball has three colors, which are red, green and blue.

- If the color of the ball is red, press the R key to pass the ball. Similarly, if the balls are green and blue, press the G key and the B key respectively to pass the ball.
- If you are the leader player, you need to press the N key to generate new ball, and when the ball pass back to you , then you need to press P key to receive this ball.
- After each round, teams reflect and adjust.
- At the end of the game, there is a game summary that you can check how many balls that you successfully pass for each round.

### 3.2.3 Drawing Together Game

**Objective:** Communicate effectively to reproduce an image based on PO instructions.

**Mechanics:** The PO describes an image while the Developer draws it, highlighting communication challenges in Agile development.

#### **How to Play:**

- This is a two-player game and there are two rounds in total.
- Describer sees an image and describes it using a textbox.
- Painter draws on canvas using brush tools.
- The first-round describer can only send one message and can only see the picture in 5 seconds, also cannot see what painter drawing.
- The second-round describer can send message and see the picture in limited time, also painter can send the drawing to describer two times for checking.
- At the end of the game, there is a game summary that you can see the original image and first-round or second-round drawing results.

## 3.3 Storyboard

### 3.3.1 Agile Coin Game

- **Landing Page:** Player chooses to host or join the Coin Game.

- **Configuration:** Host sets player count, round time, and batch configurations (As 20, 10, 5, 2 coins per batch across four rounds). Players enter names and join the lobby.
- **Lobby:** All players wait in a virtual room. Instructions about coin passing mechanics and round structure are shown.
- **Gameplay:** In each round, players pass coins to the next player using a “Completed this batch” button. The backend tracks timing and flow per batch. Rounds become shorter and more iterative.
- **Review:** After each round, the game shows delivery times and flow efficiency. Players are encouraged to reflect on bottlenecks.
- **Game Summary:** After all four rounds, final delivery data and improvement charts are shown. The team discusses the impact of batch sizes on flow.

### 3.3.2 Ball Point Game

- **Landing Page:** Player chooses to host or join the Coin Game.
- **Configuration:** Host sets Target score per round. Players input names and wait in the lobby.
- **Lobby:** The Host arranges the positions of the players according to the game rules.
- **Gameplay:** Players press the “R”, “G” or “B” keys to simulate passing according to the different colors of the ball. After a circular network all players can get 1 point after completing the game. The timer ends the round..
- **Review:** After each round, the number of completed passes is shown. Players discuss strategies to improve efficiency.
- **Game Summary:** Final team score is shown. The team reflects on what changes improved throughput across iterations.

### 3.3.3 Drawing Together Game

- **Landing Page:** Players select the Drawing Together Game. One player hosts, the other joins. The system assigns roles: one as Describer (Product Owner), the other as Painter (Developer).
- **Configuration:** Role explanation and game instructions are shown to both players. The game rules differ between rounds and are clearly introduced.
- **Lobby:** Players wait for both participants to join. Once ready, the game begins automatically with Round 1.
- **Gameplay – Round 1:**
  - Describer sees the original image for only 5 seconds.
  - Describer can send **one text message only**.
  - Describer cannot see what the Painter is drawing.
  - Painter starts drawing on the canvas based solely on that one message.
  - Timer runs until the drawing is submitted or time runs out.
- **Gameplay – Round 2:**
  - Describer can view the image for a limited time.
  - Describer is allowed to send **multiple messages**.
  - Painter can draw and **submit the draft for review up to 2 times**.
  - Describer can provide feedback after each draft, enhancing communication.
  - Final drawing is submitted before the timer ends.
- **Review:** After both rounds, players are shown:
  - The original image
  - The drawing from Round 1
  - The drawing from Round 2

Players are encouraged to compare how limited vs. iterative communication impacts result accuracy.

- **Game Summary:** A final result screen displays all three images side by side. Players discuss which communication mode was more effective and how iterative feedback improved outcomes.

### 3.4 Winning Conditions

- **Coin Game:** Lower average delivery time indicates higher efficiency.
- **Ball Game:** Team with the highest point total wins.
- **Drawing Game:** Closest visual match to the original wins.

### 3.5 Gameplay Explanation + UI Controls

- Button-based interaction: Start Game, Pass, Submit, Next Round
- Mouse for drawing; keyboard shortcuts for quick commands.
- Countdown timer and progress bar are always visible.

### 3.6 Configuration

Game setup is customizable:

- Player count (2–6 players)
- Round duration (30s to 3 mins)
- Number of rounds
- Optional features (e.g., show/hide roles, shuffle turn order)

### 3.7 Technical Architecture

- **Frontend:** HTML, CSS, JavaScript (React.js for SPA)
- **Backend (Coin Game only):** Python + FastAPI

- **Communication:** WebSocket protocol
- **Data Handling:** In-memory session state
- **Responsive UI:** Works on desktop and tablet

### Containerized Deployment with Docker

To address the committee's recommendation for containerized deployment, the Agile Hero platform has been successfully packaged into a Docker container to enable easy setup, testing, and demonstration across different environments.

The Docker deployment includes the following:

- **Agile Coin Game:** The backend service built with FastAPI is containerized to handle REST APIs and WebSocket communication. It manages player sessions, batch logic, and timing.
- **Ball Point Game and Drawing Together Game:** These frontend-only games are served together with the Coin Game using a lightweight static HTTP server within the same container image.
- **Deployment Artifacts:** The project includes a Dockerfile and a docker-compose configuration for future multi-container expansion. A README.md is also provided with detailed usage instructions.

This setup allows the platform to be deployed and launched using a single command. It ensures that the environment remains consistent across various machines, reducing manual setup and enabling portability.



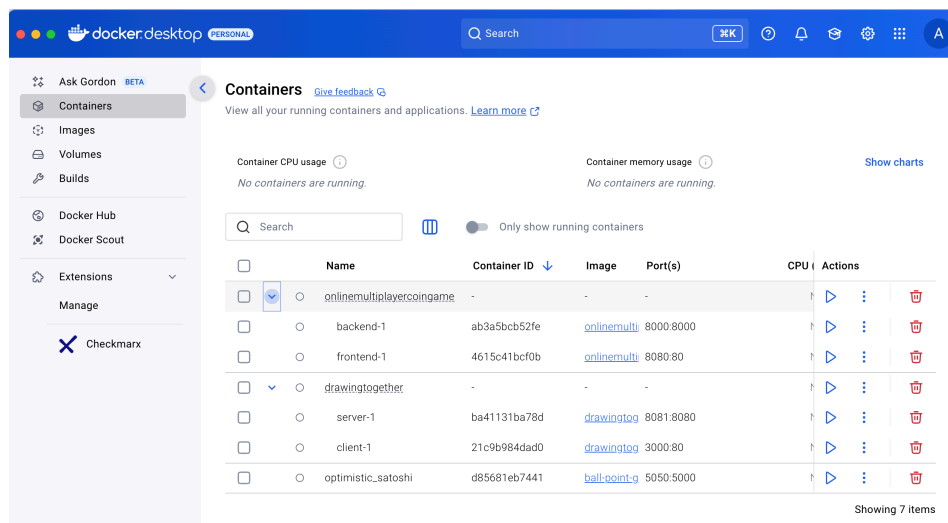


Figure 3.3: Docker container running backend services and serving frontend assets

## CHAPTER 4

### EVALUATION

This chapter presents a formal evaluation of the Agile Hero platform, focusing on functional correctness, usability, and educational effectiveness. The assessment is divided into two major sections: functionality-based evaluation and user-based evaluation. The objective is to determine how well the platform supports the intended learning outcomes of Agile training through web-based games.

#### 4.1 Functionality Evaluation

The functionality evaluation assesses whether the system components behave according to their specifications. Each of the three mini-games (Agile Coin Game, Ball Point Game, and Drawing Together) was tested using black-box methods across three main test scenarios: expected behavior, invalid interaction, and edge-case handling.

##### 4.1.1 Test Environment

All tests were conducted on modern desktop web browsers (Google Chrome and Mozilla Firefox) using a local server with Python FastAPI and WebSocket. The system was deployed in a controlled LAN environment to simulate multi-user sessions.

##### 4.1.2 Agile Coin Game

- **Scenario 1 – Normal Operation:** All players pass 20 coins correctly based on the batch rules. The round ends as expected after all passes are completed.
- **Scenario 2 – Rule Violation:** A player attempts to pass before receiving a coin; the system blocks the action and maintains turn order.
- **Scenario 3 – End of Round Detection:** The system correctly detects the end of the round when the final player flips the last coin in the final batch. Only then is the round completion triggered, and results are displayed.

#### 4.1.3 Ball Point Game

- **Scenario 1 – Successful Cycle:** Players complete a full ball-passing cycle. The score counter increments correctly.
- **Scenario 2 – Wrong Sequence:** Out-of-order passes do not increase the score, preserving team logic.
- **Scenario 3 – Restart Condition:** Each round resets correctly with updated time and scores.

#### 4.1.4 Drawing Together Game

- **Scenario 1 – Round 1 Constraints:** Describer sees the image for 5 seconds, sends one message, and has no drawing preview. Constraints are strictly enforced.
- **Scenario 2 – Round 2 Feedback Loop:** Painter submits up to two drafts, and Describer provides real-time feedback via messaging.
- **Scenario 3 – Result Display:** End screen displays original image, Round 1 drawing, and Round 2 drawing side-by-side.

The functionality testing confirmed that all game rules and flows were implemented correctly. Minor latency observed in WebSocket updates was negligible and did not affect usability.

### 4.2 User Evaluation

A user evaluation was conducted to measure the perceived usability and educational impact of the Agile Hero platform from a learner's perspective. Participants were invited to test the games and provide structured feedback.

#### 4.2.1 Participants

A total of 5 individuals participated in the evaluation. This included:

- 2 undergraduate students with some exposure to Agile
- 3 industry practitioners

### 4.2.2 Evaluation Metrics

A Likert-scale questionnaire was used to collect responses on the following aspects:

- Ease of learning the game interface and rules
- Realism of Agile principles simulated
- Engagement and interactivity
- Usefulness for training or classroom settings

### 4.2.3 Results and Analysis

- **Drawing Together Game** received the highest educational value rating. Participants highlighted how communication limitations realistically simulate the challenges of Sprint Planning and Story Handoff.
- **Agile Coin Game** helped visualize the effects of batch sizes and delays in workflow, aligning with Lean-Agile thinking.
- **Ball Point Game** encouraged teamwork and iteration, though some users felt the game would benefit from more rounds.
- Participants found the user interface generally intuitive, but suggested adding a tutorial page for first-time users.

### 4.2.4 Limitations

- Sample size was small and mostly comprised of students.
- Sessions were conducted on a single LAN; online deployment testing was not included.
- Quantitative logging (e.g., click frequency, response time) was not implemented.

#### 4.2.5 Feedback Summary and Recommendations

Quantitative and qualitative feedback was collected from participants after the game sessions. The average satisfaction score for Agile Hero was 4.0 out of 5, indicating a generally positive reception. Most users agreed that the games were easy to understand, well-integrated, and useful for learning Agile concepts.

- **Overall Experience Rating:** 4.0 / 5.0
- **Ease of Use:** 100% of participants agreed or strongly agreed that the platform was easy to use.
- **Confidence in Use:** All participants reported feeling confident using the mini-games.

Open-ended responses emphasized the drawing game as the most engaging component:

“Overall, the concept is interesting. The drawing game caught most of my attention.”

In addition, structured expert feedback suggested several directions for improving educational value and alignment with Agile learning goals:

- **Agile Metric Integration:** Each game outcome should reflect measurable Agile metrics (e.g., delivery speed, iteration improvement, WIP limits).
- **Debrief Discussion Prompts:** Include structured post-game reflection questions related to Agile principles like “Inspect and Adapt” and “Responding to Change.”
- **Customizability:** Enable instructors to modify rules or inject constraints (e.g., scope creep, multitasking limits) to reflect real-world challenges.

Example suggestions include:

- Adding a “WIP overload” rule to simulate multitasking inefficiency.

- Introducing random interruptions during gameplay to mimic shifting priorities or unplanned work.

These recommendations underscore the importance of aligning interactive game mechanics not just with entertainment, but with structured Agile learning outcomes. Future versions of the Agile Hero platform should incorporate these enhancements to support deeper reflection and trainer facilitation.

#### **4.2.6 Conclusion**

The evaluation results indicate that the Agile Hero platform is functionally reliable and pedagogically effective for introducing Agile principles in an interactive format. User responses affirm the usefulness of these games in both classroom and workshop settings.

## **CHAPTER 5**

### **CONCLUSION**

This chapter concludes the Agile Hero project by summarizing the development process, challenges, achievements, and future directions.

#### **Project Summary**

This project developed a web-based multiplayer Agile training platform with three interactive games: Agile Coin Game, Ball Point Game, and Drawing Together. These games were designed to simulate real Agile practices including flow optimization, iteration, and communication.

Through web technologies such as JavaScript, WebSocket, and FastAPI, the games support real-time gameplay for both onsite and remote teams. Each game reflects key Scrum values and was evaluated via functional testing and user feedback.

#### **Challenges Faced**

- Managing real-time synchronization between multiple clients.
- Designing logic for round-based gameplay with dynamic rules per game.
- Limiting instructions and communication in the Drawing Game while still enabling effective teamwork.

#### **Lessons Learned**

- Gamification is effective in teaching Agile, especially when roles and rules simulate real-life scenarios.
- Real-time multiplayer games require careful state tracking and user input handling.
- Agile values like “inspect and adapt” were essential not just in-game, but throughout development.

### **Future Work**

- Add analytics dashboard for trainers to monitor game outcomes.
- Support more complex Scrum roles or larger team dynamics.
- Improve mobile compatibility for hybrid use.
- Expand to more games (e.g., retrospective simulation, backlog grooming).

### **Final Reflection**

The Agile Hero platform demonstrates that Agile training can be delivered effectively through multiplayer web games. The project bridges the gap between physical training exercises and remote learning by providing engaging, interactive, and practical learning experiences that reflect real Scrum practices.



## REFERENCES

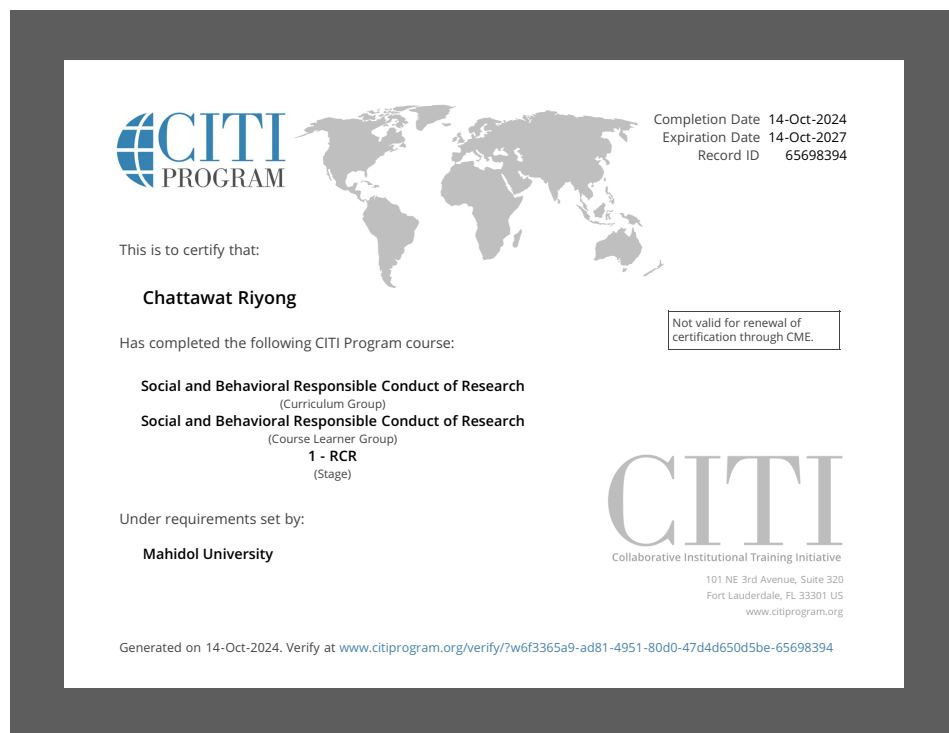
- [1] Dingsøyr T., Nerur S., Balijepally V., Moe NB.. “A decade of agile methodologies: Towards explaining agile software development”, Elsevier Inc.; 2012, <https://www.sciencedirect.com/science/article/pii/S0164121212000532>.
- [2] Sommerville I., Software engineering / Ian Sommerville. — 9th ed. vol. 9, Pearson; 2011.
- [3] Claes, Kai BDP., Wohlin, “The Waterfall Model in Large-Scale Development”, In: Markku, Päivi J., Frank APB., Oivo, editors. Product-Focused Software Process Improvement. Springer Berlin Heidelberg; 2009. p. 386–400.
- [4] McAvoy J., Sammon D., “Agile Methodology Adoption Decisions: An Innovative Approach to Teaching and Learning”, Journal of Information Systems Education. 2005;16.
- [5] Member FC.. “20 Common Challenges When Introducing Agile”, forbes; 12 2023, 20 Common Challenges When Introducing Agile (And How To Overcome Them).
- [6] Boehm B., Turner R., “Management challenges to implementing agile processes in traditional development organizations”, IEEE Software. 2005;22(5).
- [7] Rudd A.. “Agile for All — Part 1: Problem Solving”; 3 2022, <https://medium.com/@AndyPaulRudd/agile-for-all-part-1-problem-solving-95e561ffc473>.
- [8] Miro. “MIRO RECOGNIZES 50 MILLION MINDS ON THEIR WAY TO THE NEXT BIG THING”; May 2023, <https://www.prnewswire.com/news-releases/miro-recognizes-50-million-minds-on-their-way-to-the-next-big-thing-301771048.html>.
- [9] Hazzan O., Dubinsky Y.. “The Agile Manifesto”, Springer; 2014, <http://agilemanifesto.org/>.

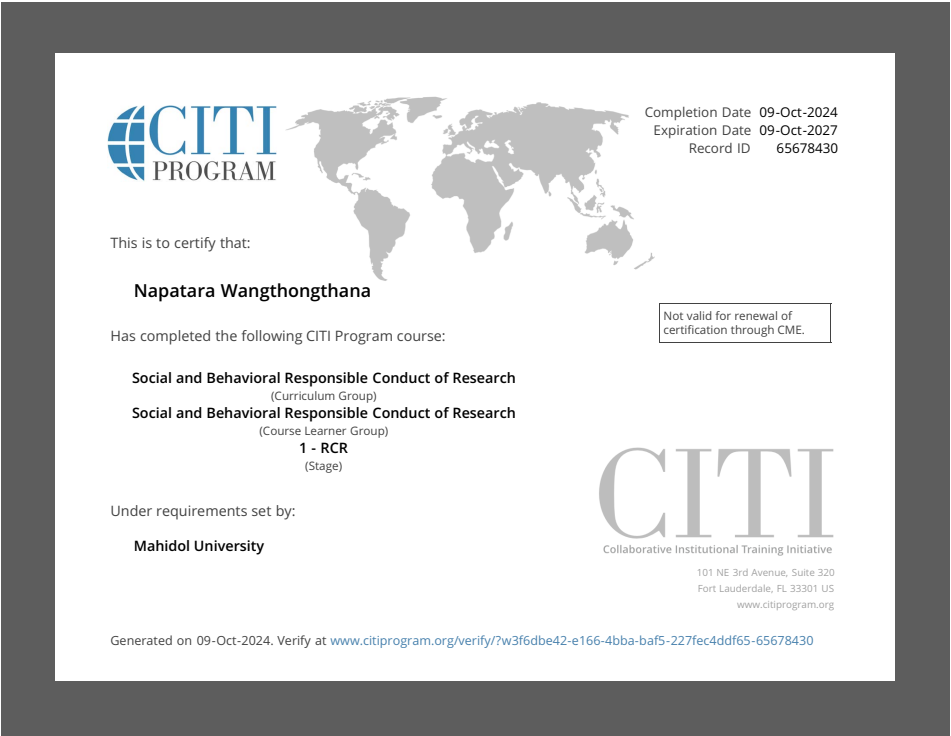
- [10] Schwaber K., Sutherland J.. “The 2020 Scrum Guide™”, [scrumguide.org](https://scrumguides.org/scrum-guide.html); 2020, <https://scrumguides.org/scrum-guide.html>.
- [11] “What is Agile Training?”, [excella](https://www.excella.com/insights/what-is-agile-training); 3 2018, <https://www.excella.com/insights/what-is-agile-training>.
- [12] Team E., [clickup](https://clickup.com/blog/agile-games/); 10 2024, <https://clickup.com/blog/agile-games/>.
- [13] WHATWG. “WebSockets Standard”; 2023, Archived from the original on 2023-03-12, Accessed: 2024-10-08, <https://websockets.spec.whatwg.org/>.

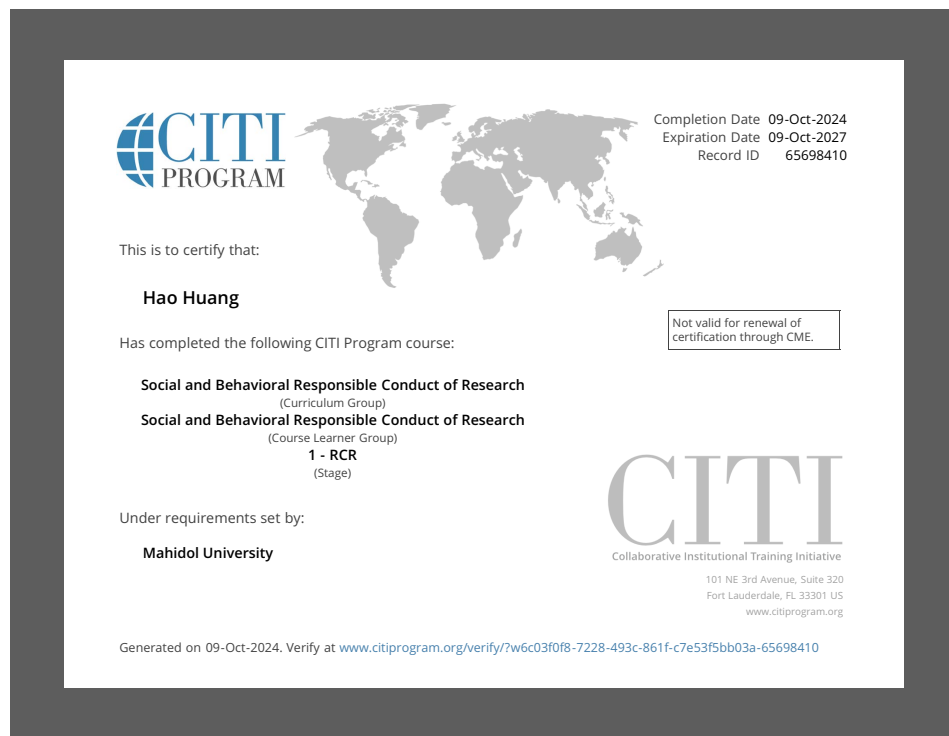
## APPENDIX A

### ETHICS APPROVAL OF THE STUDIES

#### A.1 IRB Certification







## **A.2 ICF**

## **A.3 PIS**

## APPENDIX B

### APPENDIX B: RESPONSE LETTER TO SENIOR PROJECT COMMITTEES

The following section contains the formal response to comments provided by the Senior Project Committees during the final defense of the Agile Hero project.

The full PDF response letter is included on the next page.

#### Responses to the Senior Project Committees

##### Agile Hero: Agile Mini-game

I sincerely thank the senior project committees for their valuable feedback and suggestions. Below is our detailed response to the comments received. This letter will be added as **Appendix B** in the final report.

---

#### 1. The committee suggested delivering the project using a containerised solution like Docker.

##### Response:

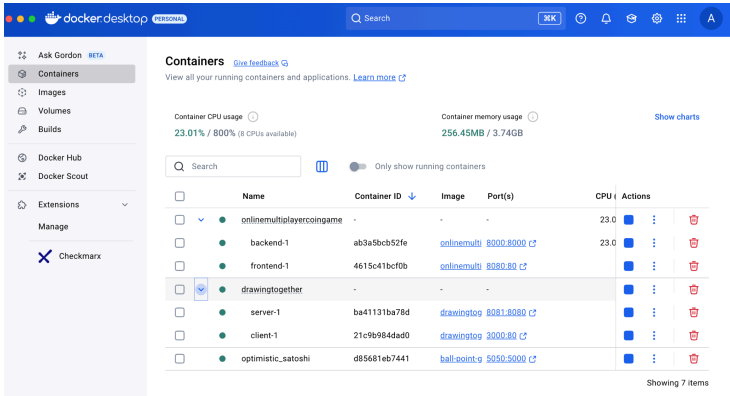
In response to this request, we have successfully containerised the **Agile Hero: Agile Mini-game**, including all three training games:

- **Agile Coin Game** (requires backend):
  - The backend implemented in **Python (FastAPI)** is containerised via a dedicated **Dockerfile**.
  - Frontend files are served via a simple **Node.js or Python HTTP server** within the container.
  - Handles WebSocket connections for real-time game synchronisation.
- **Ball Point Game** and **Drawing Together Game**:
  - Both games are frontend-only (pure HTML/CSS/JavaScript with WebSocket) and are served from the same container.
  - Game logic runs fully client-side, but real-time multiplayer is preserved via shared WebSocket channels.

To simplify deployment, we created:

- A single **Docker image** that includes all frontend assets and the backend for the Coin Game.

- A `docker-compose.yml` file for future scalability (e.g., separating frontend and backend).
- A full `Readme.md` with setup instructions, environment ports, and sample commands.



This clarification and deployment process have been added to **Section 3.7 (Technical Architecture)** on **page 18** of the report.

**2. The committee recommended improving the user interface of the game.**

**Response:**

We have reviewed and polished the user interface of all three mini-games. Improvements include:

- Improved all game styles (fonts, button colours, layout),
- Larger buttons and clearer labels to improve usability,
- Improved screen instructions and visual feedback during gameplay,
- highlighted active states for better clarity.



**Game ID: UCHRJW3** **Waiting** [START GAME](#)

Current Round: 1

Status: **Waiting**

Round Score target \*

5

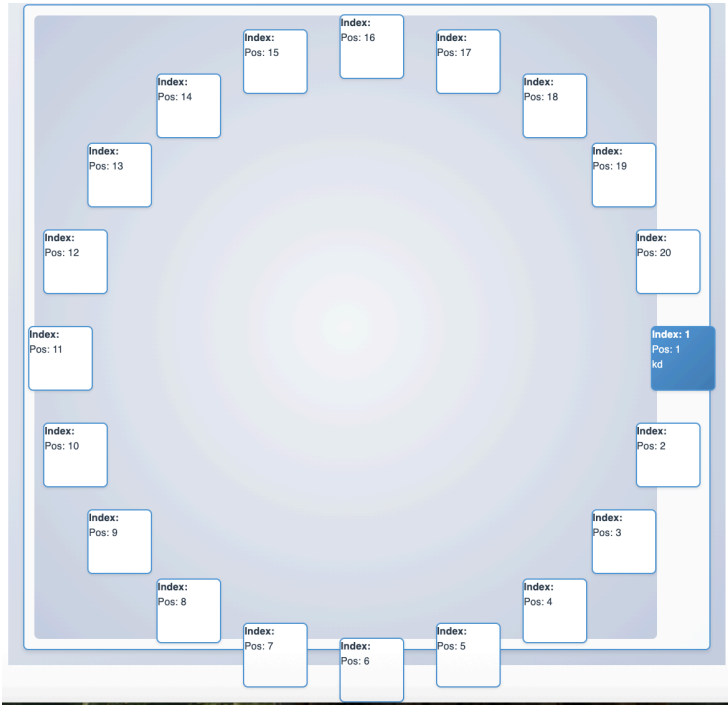
Round Score target

Round	Score
Round 1	0
Round 2	0
Round 3	0
Round 4	0
Round 5	0

Player 1: kd

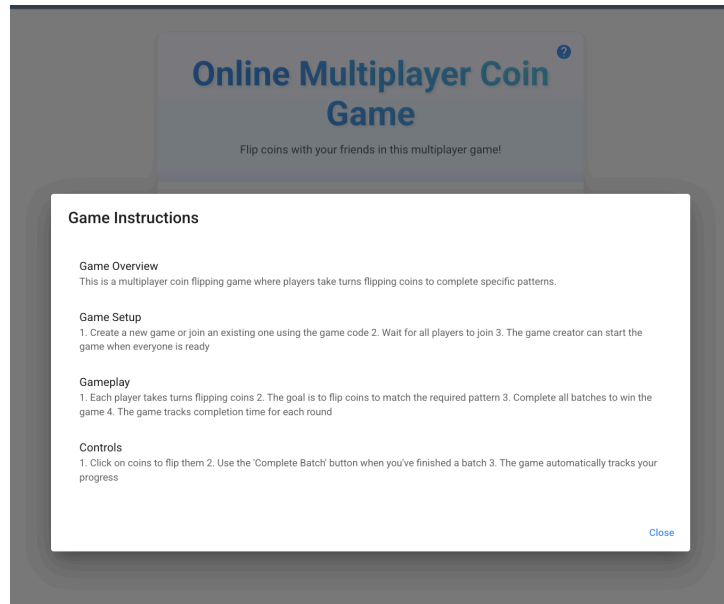
Position

1



These changes improve the game's accessibility and player experience while maintaining alignment with Agile training goals.

**3. The committee suggested adding the game instructions to the web.**



This screenshot shows that I added the game instructions to the web game. For example, in the main menu of the game, clicking on the little question mark in the top right corner will show the game instructions.

Once again, we are grateful to the committee for the helpful feedback. The above improvements have been implemented and documented in the final report accordingly.