Requirements Document

Say Don’t Say a Taboo! Online Game

Group 4: Mohammad Baqer, Brad Donohue, Johnathan Gilmore, Sean Nimmo, Stephen Lin

Contents

1. Introduction

1.1. Business Purpose

Due to Covid, the switch from in-person to virtual meetings made playing more difficult. During this time, the online conversion of classic board games has become prominent, and casual online multiplayer game websites like skribbl.io and Codenames have risen in popularity. However, Taboo! does not have such a conversion, and for this project, we would like to make an online multiplayer game website for this game. Taboo! is a game where teams of two go in rounds, where you are given a card with a target word with 5 taboo words, the point is to get your partner to say the target word without you saying the taboo words, e.g. target: Aladdin, taboo: movie, Disney, Jasmin, Genie and Jafar. This project will build on top of an already started personal project (by the stakeholder and colleagues), where the goal is to be able to publish this game to the public to be used.

1.2. Business Scope

Business Domain: Online Application Store or Online Web Game

Range of Business: Potential uses include application store purchases or online ad revenue if the game is successful

1.3. Business Structure:

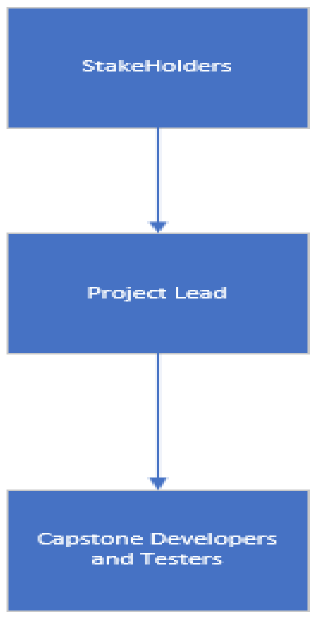


Figure 1 Business Diagram

2. System Requirements

2.1. The system shall use AI to generate cards

2.2. The system shall use Jotai for state management

2.3. The system shall use Firebase for state synchronization

2.4. The system shall use Next.js React Meta-framework

2.5. The system shall use CI/CD pipelines for Git Repository

2.6. The system shall support mobile

2.7. The system shall have a landing page

3. Game Requirements

3.1. The game must be able to support up to but not more than 12 players

3.2. The game shall allow players to join teams with a minimum of 2 players per team

3.3. The game shall allow players to create a team and join or leave a team

3.4. The game shall have a drag and drop indicator

3.5. The game shall contain a shareable link

3.6. The game shall have a exiting room

3.7. The game shall have the ability to create a new room

3.8. The game shall keep track of the score for both teams

3.9. The game shall have a Result/Summary page

3.10. The game shall allow ten rounds a game to be played

3.11. The game contains a timer that counts down 1 min per turn.

3.12. The game shall disconnect a player when the player exits their browser

3.13. The game shall allow players to select an avatar

3.14. The game shall contain a Result/Summary page at the end of 10 turns

3.15. The game shall contain a landing page for new players

3.16. The game shall alternate between teams per turn

3.17. The game shall feature a leaderboard that tracks the best players of all-time (not done)

4. Testing Requirements

4.1. Unit Testing

Each individual component of the game software must be tested to validate that the software code is performing as expected. This must be done during the development process of new game features. The component must pass the unit test before the integration process to ensure that the software is robust by eliminating most of the system bugs. Obviously, if the game feature did not pass the unit test, then it means that there are bugs in the components and cannot be integrated unless it is fixed.

4.2. User Interaction Testing

The user interactions test is crucial before the launch of the game to the public. This type of test showed how the user interacts with the software, in this case, the game. It helps the developer identify which parts of the game the users are confused about and what parts of the game can be improved upon. This type of test makes sure that the game will be in a healthy state, by eliminating confusion and updating the game based on user feedback.

4.3. Regression Testing

The regression test helps identify if new updates or changes have created new problems in the existing system. This test ensures the unification of the software where every change made did not accidentally created bugs in the existing features. The test cases will be automated so that whenever new changes are made, the test will be executed to check for bugs.

4.4. Visual Testing

Visual testing validates whether the developed software UI is compatible with the suer’s view. It ensures that the game website layout follows the correct visual elements like spacing, sizes, shapes, and UI elements positions. The game must render properly with various devices and browsers. This test helps identify how the visual aspect of the games affects the game system.

4.5. End-to-End Testing for the Happy Path

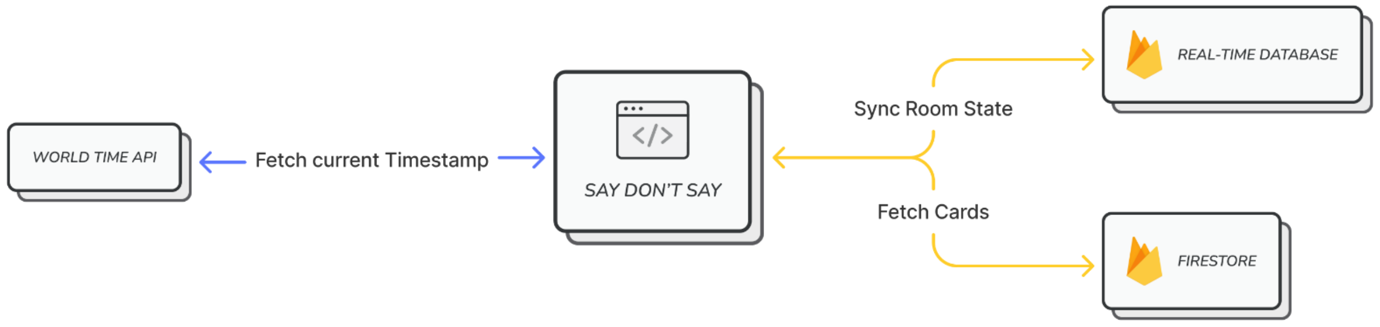
The happy path test is a type of software test where it takes a known input and produces an expected output. The test does not represent real-world conditions and only verifies if the functionality is there and if the functionality is operating correctly. If there is another viable alternative path, the happy path is the default scenario where there are no exceptional or error conditions. The failure of this test means that the game does not work since the user cannot play the game without encountering bugs.

4.6. Manual Usability Testing

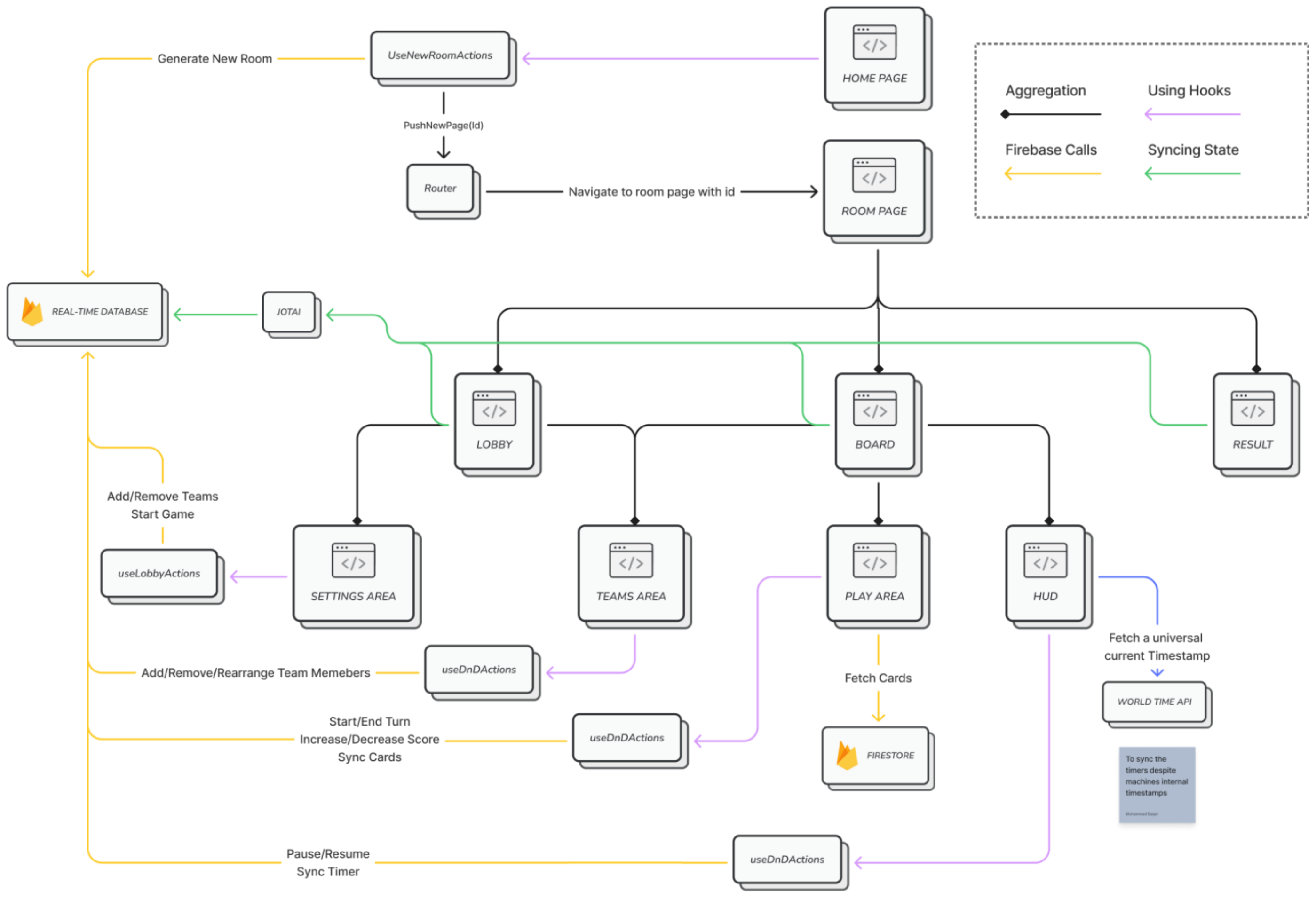
The game must be playable, and to test the playability of the game, the game must be tested by people. This test will consist of two phases where phase one the test is performed on our friends and get their feedback. Phase two the game will be tested against a larger random sample size. A larger sample random size can provide statical data on the future development of the game and gauge the game’s potential success in the population. The feedback from the tests can help identify what game features and visual elements need to be included or changed.

5. Software Architecture

5.1. Top Level Design

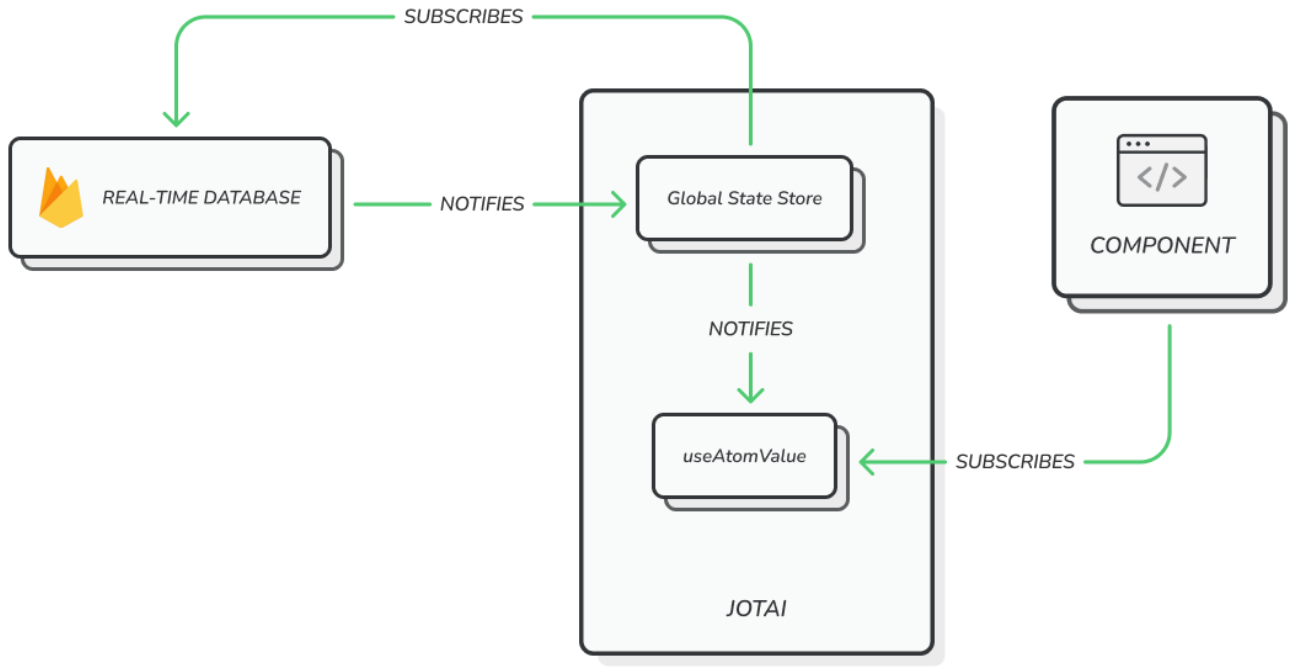


5.2. Component Level Design

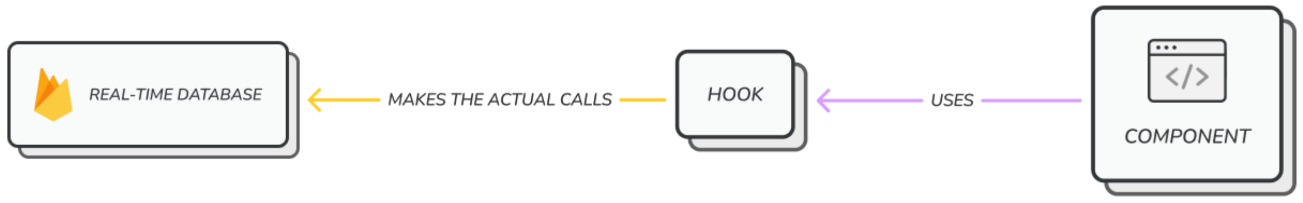


5.3. Design Patterns Used

5.3.1. Observer Pattern



5.3.2. Facade Pattern



6. Project Constraints

6.1. Time

The project time has been scheduled for a 6-month period. More time may be required depending on the scope of the project. The team currently consists of 5 members, and the plan was to implement additional features mentioned under the system and game requirements.

6.2. Cost

The game is being developed on a voluntary basis. The software components of the game are being developed using free software. There should not be any cost associated with the development of the game at the time of planning. However, it is also possible in the future there may be costs associated with the game if the current software cannot support the player base. The only cost required was due to the need to host the game on a domain. A domain was purchased for $30/year to host the game, and players access the domain via a search of the game name or by passing a game link to another player.

6.3. Scope

All in-scope tasks are listed under the game requirements. For out-of-scope tasks, the game will only be focus on the English language at the time of release. The current team size is small, therefore, developing the game features is prioritized over translation at the time.

6.4. Quality

The project is considered completed once all the game requirements are completed. Once the game requirements are completed and the game is playable, then the project is considered completed. The tests that we plan on doing like the usability test and user interaction test can help gauge the player's interest and enjoyability. Further game development and improvement will be developed according to the player feedback.

6.5. Resources

The current teams consist of 5 software developers with various roles. There is no additional external equipment required besides a computer. The project does require some software to develop games like Next.js, Jotai, Firestore, GitHub, Netlify, and Cypress.

6.6. Risks

None of the current team members work or are familiar with AI, and one of the system requirements is to develop an AI that will auto-generate the cards. This could potentially increase the time it takes to complete the project, however, the game can still function without needing AI to generate the cards.

Another risk could be that there are competitors that have already created this similar game in terms of concept. This game started during the Covid-19 Pandemic where the market does exist for an online version of *Taboo!* There is a possibility that by the time this project is ready for publishment, the game may not be profitable. However, the game was originally developed not intending to generate any profit.

7. Use Case

Players use the system to play the game with the rules described in the project description.

Players access the UI via a sharable URL link which leads them to the landing page.

Players can start the game once all the players click the start button.

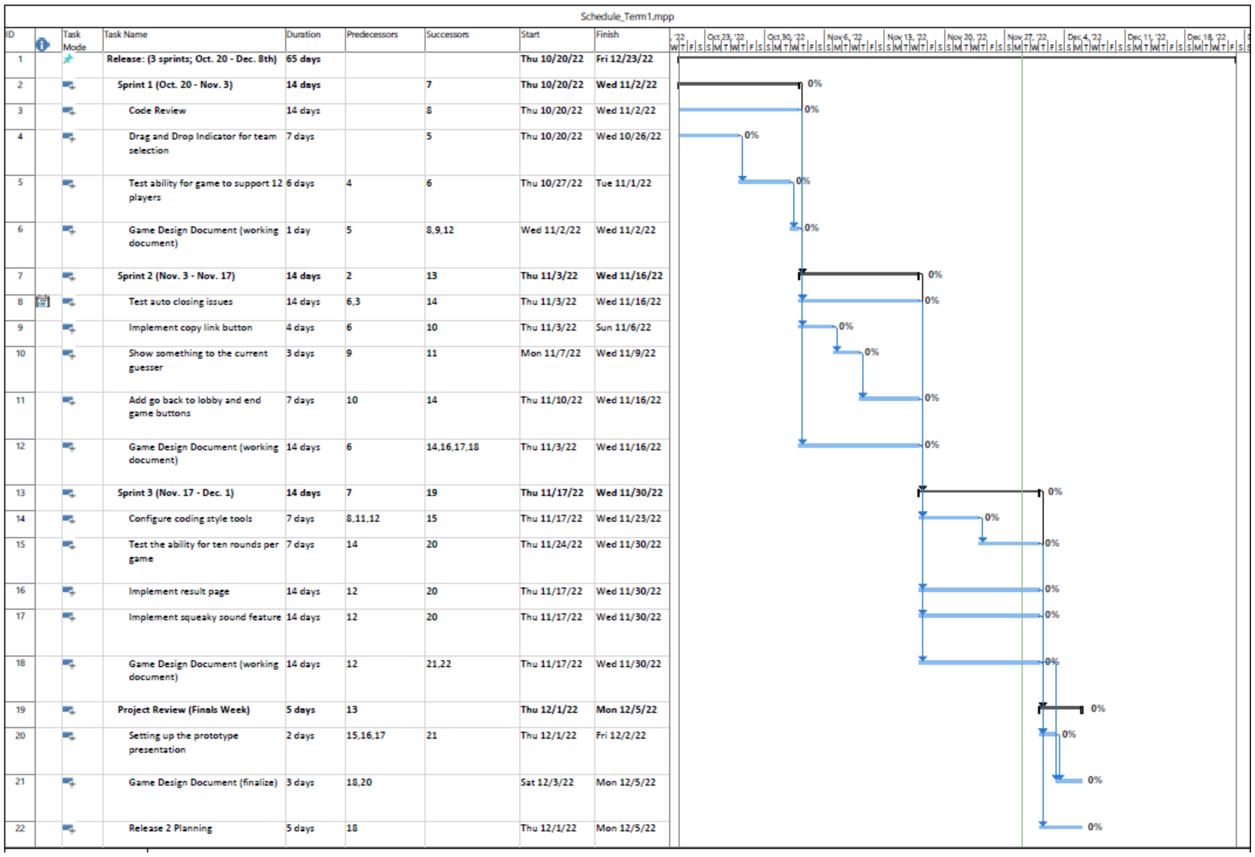
During the game, a player can:

- pause the game by clicking on the timer
- earn point for guessing correctly
- take points away from opposing team for saying Taboo words

8. Scheduling

Scheduling shall be based on a two-week sprint cycle with each sprint incorporating days allocated to feature testing and feature development. An additional block of time shall be set aside for backlog grooming to ensure we are keeping on schedule and to ensure important features are prepared to be merged into the baseline.

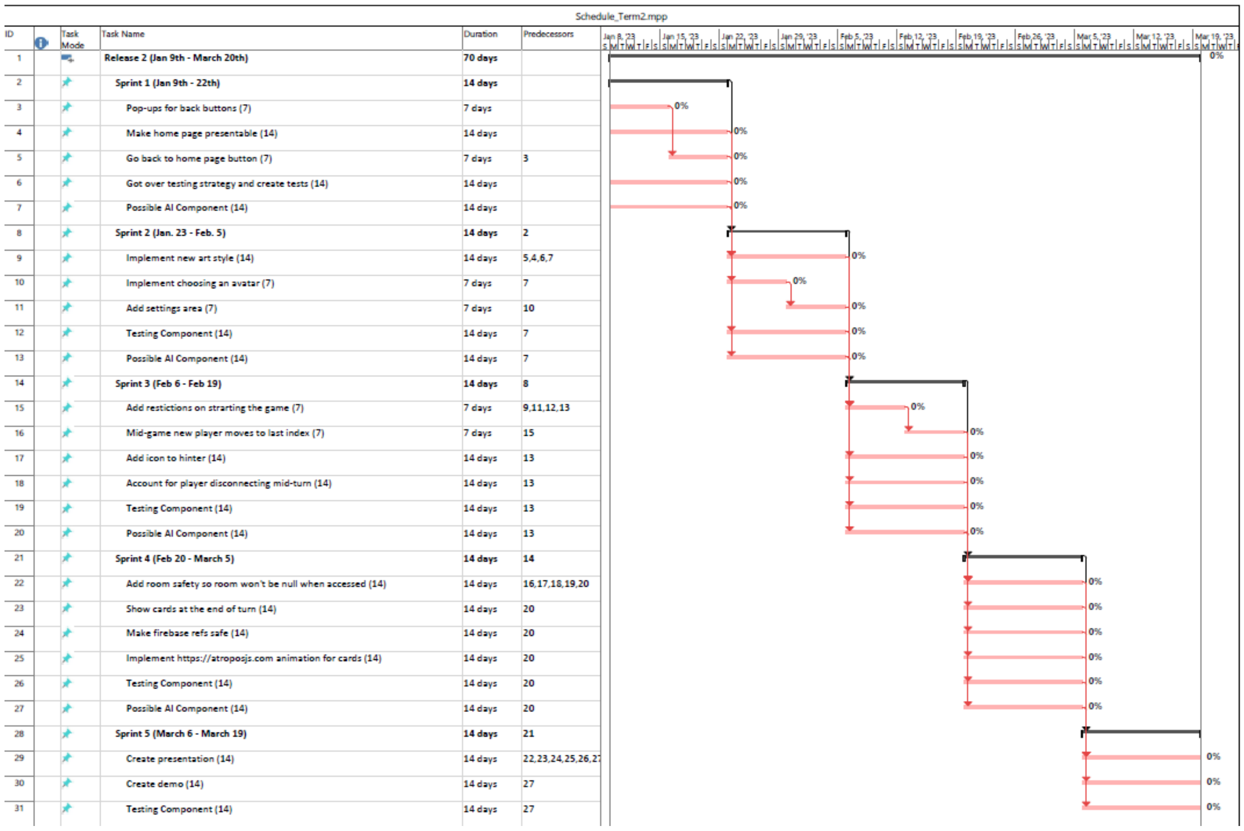
8.1. Term 1



ID		Task Mode	Task Name	Duration	Predecessors	Successors	Start	Finish
1			**Release: (3 sprints; Oct. 20 - Dec. 8th)**	**65 days**			**Thu 10/20/22**	**Fri 12/23/22**
2			**Sprint 1 (Oct. 20 - Nov. 3)**	**14 days**		**7**	**Thu 10/20/22**	**Wed 11/2/22**
3			Code Review	14 days		8	Thu 10/20/22	Wed 11/2/22
4			Drag and Drop Indicator for team selection	7 days		5	Thu 10/20/22	Wed 10/26/22
5			Test ability for game to support 12 players	6 days	4	6	Thu 10/27/22	Tue 11/1/22
6			Game Design Document (working document)	1 day	5	8,9,12	Wed 11/2/22	Wed 11/2/22
7			**Sprint 2 (Nov. 3 - Nov. 17)**	**14 days**	**2**	**13**	**Thu 11/3/22**	**Wed 11/16/22**
8			Test auto closing issues	14 days	6,3	14	Thu 11/3/22	Wed 11/16/22
9			Implement copy link button	4 days	6	10	Thu 11/3/22	Sun 11/6/22
10			Show something to the current guesser	3 days	9	11	Mon 11/7/22	Wed 11/9/22
11			Add go back to lobby and end game buttons	7 days	10	14	Thu 11/10/22	Wed 11/16/22
12			Game Design Document (working document)	14 days	6	14,16,17,18	Thu 11/3/22	Wed 11/16/22
13			**Sprint 3 (Nov. 17 - Dec. 1)**	**14 days**	**7**	**19**	**Thu 11/17/22**	**Wed 11/30/22**
14			Configure coding style tools	7 days	8,11,12	15	Thu 11/17/22	Wed 11/23/22
15			Test the ability for ten rounds per game	7 days	14	20	Thu 11/24/22	Wed 11/30/22
16			Implement result page	14 days	12	20	Thu 11/17/22	Wed 11/30/22
17			Implement squeaky sound feature	14 days	12	20	Thu 11/17/22	Wed 11/30/22
18			Game Design Document (working document)	14 days	12	21,22	Thu 11/17/22	Wed 11/30/22
19			**Project Review (Finals Week)**	**5 days**	**13**		**Thu 12/1/22**	**Mon 12/5/22**
20			Setting up the prototype presentation	2 days	15,16,17	21	Thu 12/1/22	Fri 12/2/22
21			Game Design Document (finalize)	3 days	18,20		Sat 12/3/22	Mon 12/5/22
22			Release 2 Planning	5 days	18		Thu 12/1/22	Mon 12/5/22

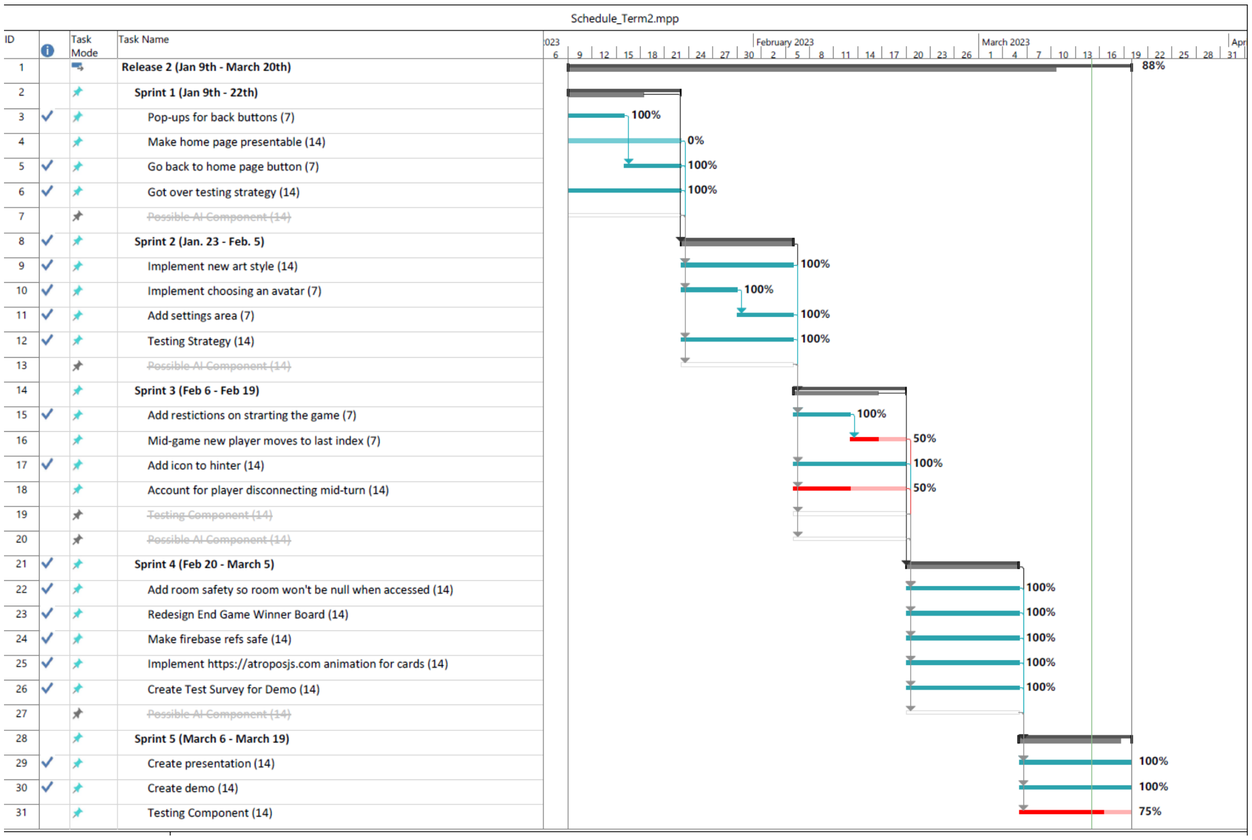
Figure 2 Term 1 Project Timeline

8.2. Term 2


Schedule_Term2.mpp

ID		Task Mode	Task Name	Duration	Predecessors
1			**Release 2 (Jan 9th - March 20th)**	**70 days**	
2			**Sprint 1 (Jan 9th - 22th)**	**14 days**	
3			Pop-ups for back buttons (7)	7 days	
4			Make home page presentable (14)	14 days	
5			Go back to home page button (7)	7 days	3
6			Got over testing strategy and create tests (14)	14 days	
7			Possible AI Component (14)	14 days	
8			**Sprint 2 (Jan. 23 - Feb. 5)**	**14 days**	2
9			Implement new art style (14)	14 days	5,4,6,7
10			Implement choosing an avatar (7)	7 days	7
11			Add settings area (7)	7 days	10
12			Testing Component (14)	14 days	7
13			Possible AI Component (14)	14 days	7
14			**Sprint 3 (Feb 6 - Feb 19)**	**14 days**	8
15			Add restictions on strarting the game (7)	7 days	9,11,12,13
16			Mid-game new player moves to last index (7)	7 days	15
17			Add icon to hinter (14)	14 days	13
18			Account for player disconnecting mid-turn (14)	14 days	13
19			Testing Component (14)	14 days	13
20			Possible AI Component (14)	14 days	13
21			**Sprint 4 (Feb 20 - March 5)**	**14 days**	14
22			Add room safety so room won't be null when accessed (14)	14 days	16,17,18,19,20
23			Show cards at the end of turn (14)	14 days	20
24			Make firebase refs safe (14)	14 days	20
25			Implement https://atroposjs.com animation for cards (14)	14 days	20
26			Testing Component (14)	14 days	20
27			Possible AI Component (14)	14 days	20
28			**Sprint 5 (March 6 - March 19)**	**14 days**	21
29			Create presentation (14)	14 days	22,23,24,25,26,2
30			Create demo (14)	14 days	27
31			Testing Component (14)	14 days	27

Figure 3 Term2 Project Timeline

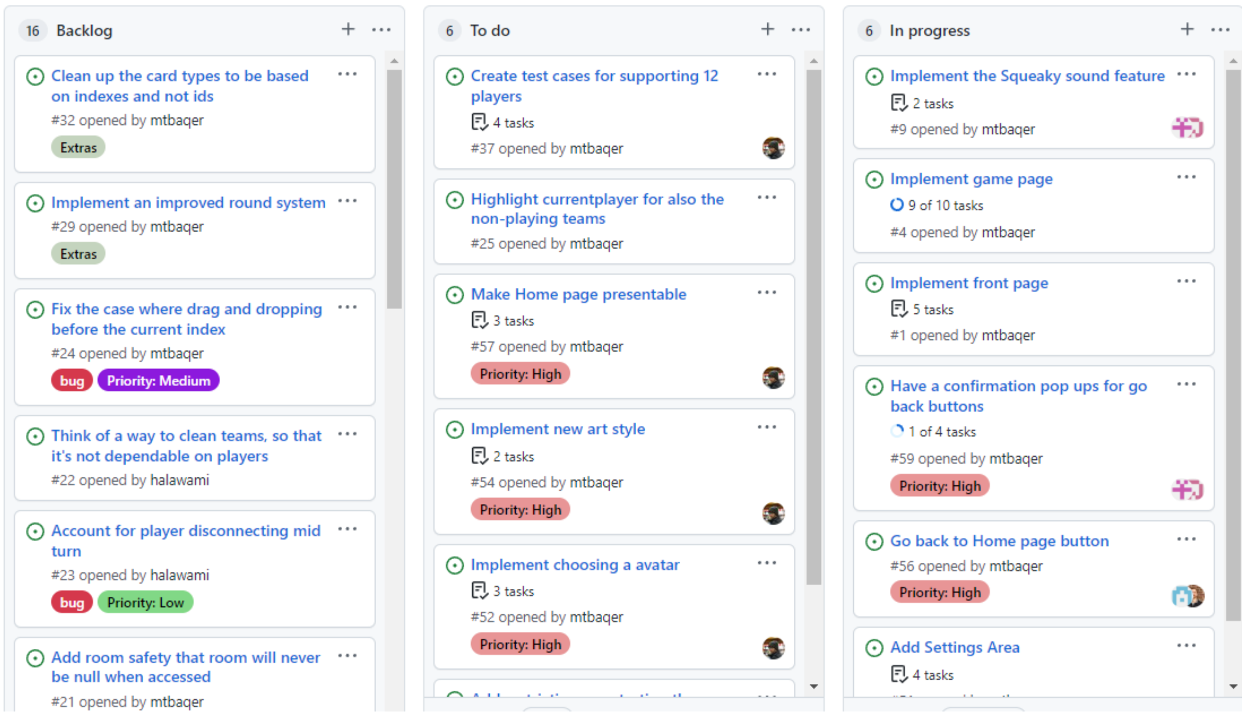


ID	ℹ	Task Mode	Task Name
1			**Release 2 (Jan 9th - March 20th)**
2			**Sprint 1 (Jan 9th - 22th)**
3	✓		Pop-ups for back buttons (7)
4			Make home page presentable (14)
5	✓		Go back to home page button (7)
6	✓		Got over testing strategy (14)
7			~~Possible AI Component (14)~~
8	✓		**Sprint 2 (Jan. 23 - Feb. 5)**
9	✓		Implement new art style (14)
10	✓		Implement choosing an avatar (7)
11	✓		Add settings area (7)
12	✓		Testing Strategy (14)
13			~~Possible AI Component (14)~~
14			**Sprint 3 (Feb 6 - Feb 19)**
15	✓		Add restictions on strarting the game (7)
16			Mid-game new player moves to last index (7)
17	✓		Add icon to hinter (14)
18			Account for player disconnecting mid-turn (14)
19			~~Testing Component (14)~~
20			~~Possible AI Component (14)~~
21	✓		**Sprint 4 (Feb 20 - March 5)**
22	✓		Add room safety so room won't be null when accessed (14)
23	✓		Redesign End Game Winner Board (14)
24	✓		Make firebase refs safe (14)
25	✓		Implement https://atroposjs.com animation for cards (14)
26	✓		Create Test Survey for Demo (14)
27			~~Possible AI Component (14)~~
28			**Sprint 5 (March 6 - March 19)**
29	✓		Create presentation (14)
30	✓		Create demo (14)
31			Testing Component (14)

Figure 4 Actual Term Schedule Implementation

9. Project Plan

To manage the project, we currently utilize GitHub. Here we manage a Kanban board where all tasks are assigned to one or two developers to work on in each sprint. Some tasks may take longer than a sprint, however, we estimate the times and then adjust as the project continues.


16 Backlog

Clean up the card types to be based on indexes and not ids
#32 opened by mtbaqer
Extras

Implement an improved round system
#29 opened by mtbaqer
Extras

Fix the case where drag and dropping before the current index
#24 opened by mtbaqer
bug Priority: Medium

Think of a way to clean teams, so that it's not dependable on players
#22 opened by halawami

Account for player disconnecting mid turn
#23 opened by halawami
bug Priority: Low

Add room safety that room will never be null when accessed
#21 opened by mtbaqer

6 To do

Create test cases for supporting 12 players
4 tasks
#37 opened by mtbaqer

Highlight currentplayer for also the non-playing teams
#25 opened by mtbaqer

Make Home page presentable
3 tasks
#57 opened by mtbaqer
Priority: High

Implement new art style
2 tasks
#54 opened by mtbaqer
Priority: High

Implement choosing a avatar
3 tasks
#52 opened by mtbaqer
Priority: High

6 In progress

Implement the Squeaky sound feature
2 tasks
#9 opened by mtbaqer

Implement game page
9 of 10 tasks
#4 opened by mtbaqer

Implement front page
5 tasks
#1 opened by mtbaqer

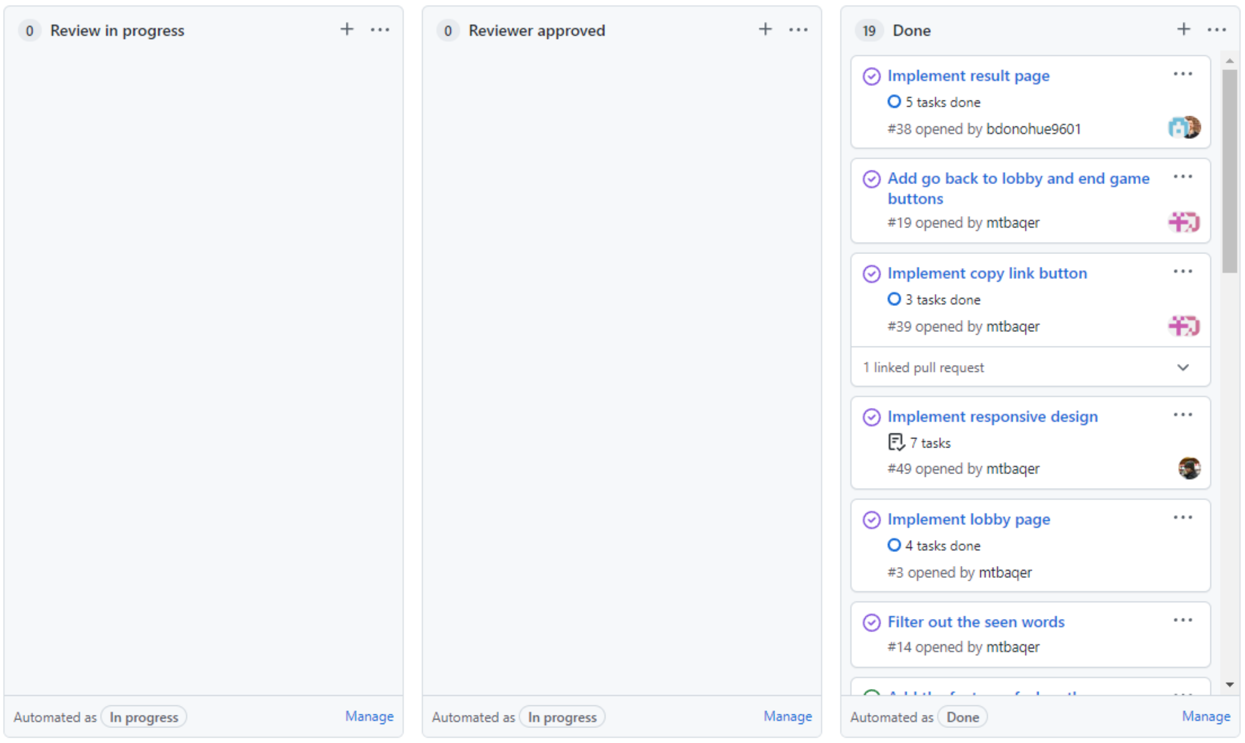
Have a confirmation pop ups for go back buttons
1 of 4 tasks
#59 opened by mtbaqer
Priority: High

Go back to Home page button
#56 opened by mtbaqer
Priority: High

Add Settings Area
4 tasks

Figure 4: Kanban Board

Tasks are organized in the backlog by severity with Priority High as the most urgent task. These are moved into the "To Do" column. Once the task is assigned, these are moved into "In Progress" where the task will be completed.



Implement result page
5 tasks done
#38 opened by bdonohue9601

Add go back to lobby and end game buttons
#19 opened by mtbaqer

Implement copy link button
3 tasks done
#39 opened by mtbaqer
1 linked pull request

Implement responsive design
7 tasks
#49 opened by mtbaqer

Implement lobby page
4 tasks done
#3 opened by mtbaqer

Filter out the seen words
#14 opened by mtbaqer

Figure 5 Kanban Board 2

Once a task has been completed, the task is then moved over to the "Review in Progress" column and subsequently moved into the "Reviewer Approved" column. Finally, it is moved into the "Done" column. All merge requests are managed by the project lead and are subject to his approval.

10. Acronyms and Abbreviations

UI - user interface

User – the player

11. Shareholders Signature

Name: Mohammad Baqer

 Date 03/15/23

Name: Hussein Alawami

Date: 03/15/23

Name: Omar Fayoumi

 Date: 03/15/23