

Project Structure Document

Computer Games Laboratory

Summer Term 2021



Meeple People

Anastasia Pomelova

Eugene Ghanizadeh Khoub

Mert Ülker

Shyam Rangarajan

Contents

1	Formal Game Proposal	3
1.1	Game Description	3
1.1.1	Gameplay Details	3
1.1.1.1	Goal	3
1.1.1.2	Game Flow	3
1.1.1.3	Additional Rules and Content	6
1.2	Technical Achievement	8
1.3	Big Idea Bullseye	8
1.4	Development Schedule	9
1.4.1	Layered Task Breakdown	9
1.4.2	Timeline	10
1.5	Assessment	13
1.5.1	Thematic Binding	13
1.5.2	Strategic Gameplay	13
1.5.3	Our Definition of Success	13
2	Game Prototype	14
2.1	Prototype Description	14
2.1.1	Gameplay Details	14
2.1.1.1	Game Flow	15
2.1.1.2	Additional Rules and Content	15
2.2	Gameplay Experience	15
2.3	Lessons Learned and Design Revisions	16
2.3.1	Goal I	16
2.3.2	Goal II	17
2.3.3	Goal III	18
2.3.4	Conclusion	18
3	Interim Demo	19
3.1	Design Approach	19
3.2	Interim Status	20
3.2.1	UI Design	20
3.2.2	Game Logic	20
3.2.3	Game Manager	21
3.2.4	Networking	22
3.3	Challenges	22
3.4	Next Steps	23
4	Alpha Release	24
4.1	Status Overview	24
4.2	Game Logic and UI	24
4.2.1	Game Logic	25
4.3	Challenges	26
4.4	Next Steps	26

5	Playtesting	27
5.1	Playtest Setting	27
5.1.1	Scope	27
5.1.2	Playtester Selection	27
5.1.3	Testing Method	27
5.2	Playtest Results	28
5.2.1	Play Matrix	28
5.2.2	Quantitative Feedback	28
5.2.3	Qualitative Feedback	31
5.3	Changes and Next Steps	33
6	Conclusion and Final Release	34
6.1	Summary	34
6.2	Changes since Alpha Release	35
6.2.1	Networking	35
6.2.2	Complete Priority System	35
6.2.3	Sound	36
6.2.4	Menu Sections	36
6.2.5	Player Help	37
6.2.6	Miscellaneous Changes	37
6.3	Group Experience	38
6.4	Course Impressions	38
6.4.1	Expectations	38
6.4.2	Our Achievements	39
6.4.3	Our Learnings	39
6.4.4	Course Suggestions	39

1 Formal Game Proposal

1.1 Game Description

“Halt! What business have you in our glorious city of Kota?”

“Sir, kind sir, please let us in! Our small village of Dharti was plundered by the giants not two days past. Flee we did, but they come behind us! Won’t you take us in?”

“Dharti you say? Hmm, you provide us much grain. Our liege lord perchance might let you in. But you must show us your papers. Queue up! Our glorious knights can fend off the giants.”

Welcome to **Flee Fi Fo From**, where players take on the role of medieval guilds attempting to protect villagers from rampaging giants by bringing them to safety within their city walls. Altruism is not the primary motive though – Guilds are trying to complete hidden objectives and look as good as possible during this crisis management, in order to earn goodwill influence points in the city. Guilds must maintain the illusion of fairness and order in the midst of the chaos or risk the displeasure of the city population. Stakes are high, morals are low, and no one is above some skullduggery. Do you have what it takes to effectively manage this chaotic environment and out maneuver your opponents?

1.1.1 Gameplay Details

1.1.1.1 Goal

The goal of the game is to earn the most Influence Points (IP) within the city during the crisis management process. Players earn IP by saving villagers, completing hidden objectives, maintaining a high rating and earning inner castle rewards. The player with the highest IP total is the winner.

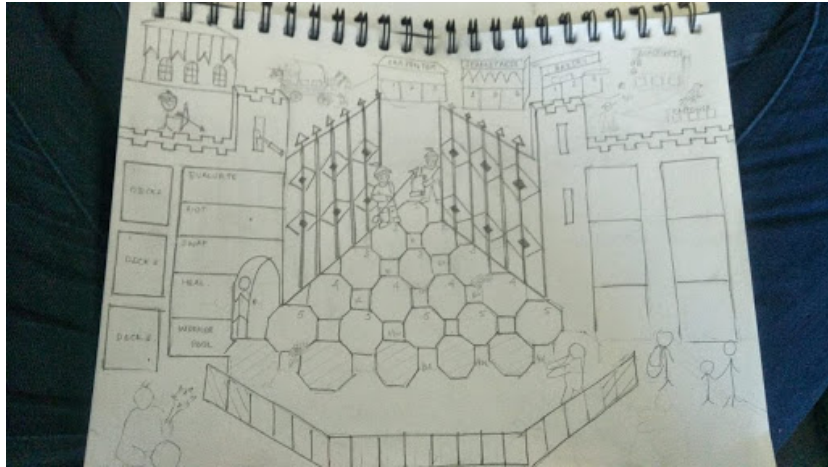


Figure 1: Game Board

1.1.1.2 Game Flow

The game is played over a series of Action and Reset turns. The player with the Action marker starts the game and plays their Action turn. Immediately after, the player with the Reset marker plays their Reset turn. After both players complete their Action and Reset turns respectively, they move each marker clockwise.

Action Turn:

The purpose of the action turn is primarily to save people. Players may choose to do so in an orderly fashion, or they may even create additional **chaos** to save more people and meet secret objectives. Players play actions by moving workers from their available pool to the action mat.

A player may use up to 2 workers on their Action turn, by placing their workers on the corresponding location on the Action mat. A player may choose from the following 5 actions:

- Authorize: Permit someone into the city, different types of villagers have restrictions.
- Swap: Interchange the position of two people.
- Riot: Create chaos with a Guild Knight and run into the city. Commoners may follow, receive Disgrace based on injuries caused.
- Revive: Assist an injured villager up and gain Honor.
- Objective: Draw a new objective card.

Reset Turn:

The purpose of the reset turn is to restore **order** by:

- Maintaining the flow in the evacuation board by moving villagers forward while maintaining queuing priority and refilling the empty spaces with new villagers.
- Relieving used workers from the action mat back to make them available in the players' pool again.
- Maintaining knight formation against the giants or retreating.

The Reset player chooses one ability from each side of the Reset mat (Guild Orders and Relief Orders). The player may use these chosen abilities in any sequence during their turn.

Move Turn Markers:

Once the two players have taken an Action and Reset turn respectively, each marker is moved one step clockwise to the next player. Play then continues with the player with the Action marker taking their Action turn and the player with the Reset marker following with their Reset turn, until the Game End is triggered.

Game End:

The end of the game is triggered when:

1. All villagers have been brought to safety OR
2. The total guild knight strength in the battlefront is lower than the strength of the giants



(a) Actions Possible



(b) Reset Abilities

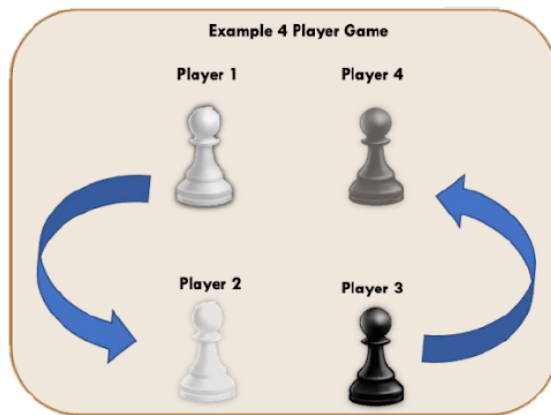


Figure 3: Move Turn Markers

1.1.1.3 Additional Rules and Content

Further rules and envisioned content are listed as follows:

Objectives:

Objective cards are hidden goals that each player is trying to complete. Objectives may be completed by a player at any point during their Action or Reset turn. There is no limit to the number of objectives that a player may hold in their hand, or the number that they may complete on a single turn.

Objective cards will be of three types: Rescue (set collection of villagers), Observe (pattern formation in the crisis area), and Admin (miscellaneous logistics activities). A set of standard game objectives is present inside the city, which may be claimed once by each player with the use of a Guild Knight.

Movement Rules:

Pieces may only move forwards towards the city gate. Backwards and lateral movement is not allowed.

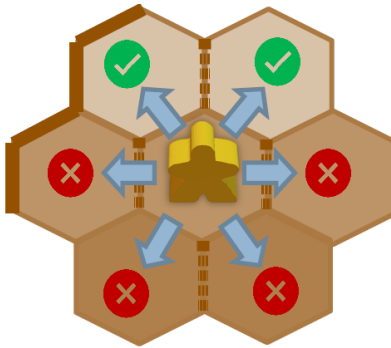


Figure 4: Movement Rules

Queuing Order:

As upholders of justice and equality, the city of Kota prides itself on its adherence to order at all times, even in the midst of a crisis. The movement of people into the city is tightly regulated by a FIFO (First In First Out) system, to ensure that there is no discrimination of any kind. FIFO is maintained on the basis of priority levels, and is tracked via rows (distance to the city gate).

- Untapped and injured pieces are considered high priority, tapped pieces are lower priority.
- Within a row, all high priority pieces must be moved first. If this order is violated, the player receives a Disgrace for each piece that breaks FIFO. (5b)
- An adult moved forward must tap. (5a)
 - Children never tap and are always high priority.
 - An adult moving forward to an empty row does not tap.
 - Tapped pieces remain tapped while moving forward.
- If an entirely filled row contains only tapped pieces, then all of the pieces in the row must be immediately untapped. (5c)

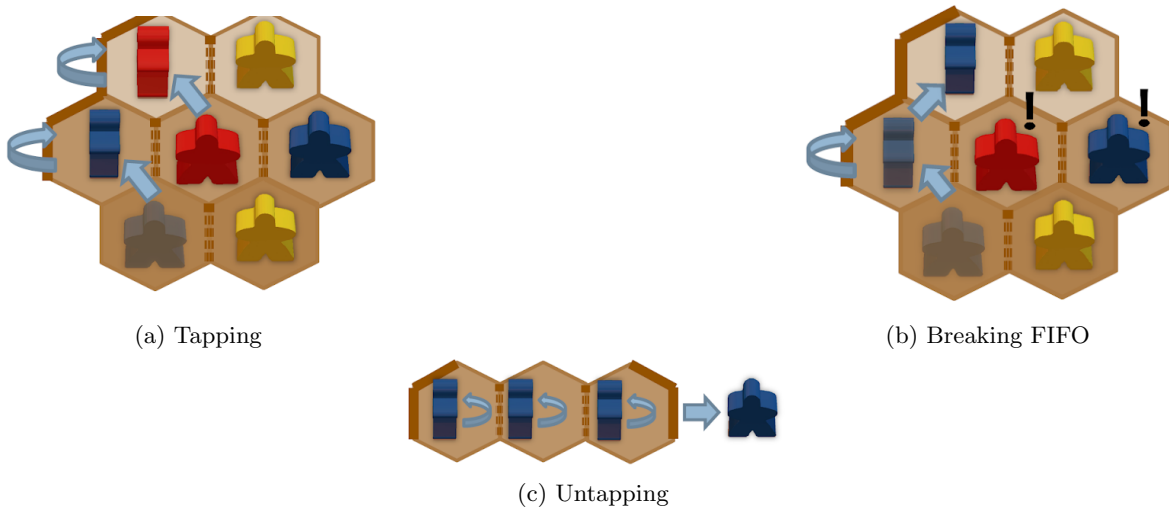


Figure 5: Queuing Rules

Types of Pieces:

The pieces involved in the crisis are of four types:

- Children: Possess mobility in the queue and are always high priority. But they must be accompanied by an adult, as the city does not take kindly to negligence.
- Commoners: Run of the mill villagers. They possess herd mentality, which allows them to become followers in Riots.
- Elderly: They are slow moving, but the city expects special care to be given to accommodate them, in which case they provide Honor. Failure to do so leads to Disgrace.
- Guild Knights: Player specific pieces, which participate in the defence against the giants. They may be used to start Riots in the queue, as well as to secure additional influence in the castle.

Glory Tracker:

The city of Kota takes special pride in the glory of its guilds. Good deeds (maintaining order, going above and beyond) are rewarded with Honor. Misdemeanors (creating chaos or breaking rules) will be deemed a Disgrace.



Figure 6: Glory Tracker for Disgrace and Honor

1.2 Technical Achievement

Our technical innovation revolves around the networking capabilities of our game, and aims to enhance the game system that we build to make it enjoyable by as many players as possible. We believe that board games are meant to be a social experience, and we hope to achieve this goal by making it possible for groups of individuals to partake in our game at the same time. Our goal is to facilitate the simultaneous participation of up to 4 players within the same game.

In order to establish a multiplayer board game session, we plan a network model composed of a dedicated server, which would host the session for the clients and act as the main authoritative component within the model. The server will be the hub for remote procedure calls from/to clients and the sole channel of communication among multiple devices. While we initially aim for a networked multiplayer session using a local area network (LAN), we also want to allow sessions over the internet as we progress through our project milestones, likely by employing a relay server or performing NAT punch-through. Furthermore, we envision extra features in relation with our technical achievement such as game lobbies, rank system & matchmaking and text/voice chat throughout the game session to further enhance the traditional board game experience.

1.3 Big Idea Bullseye

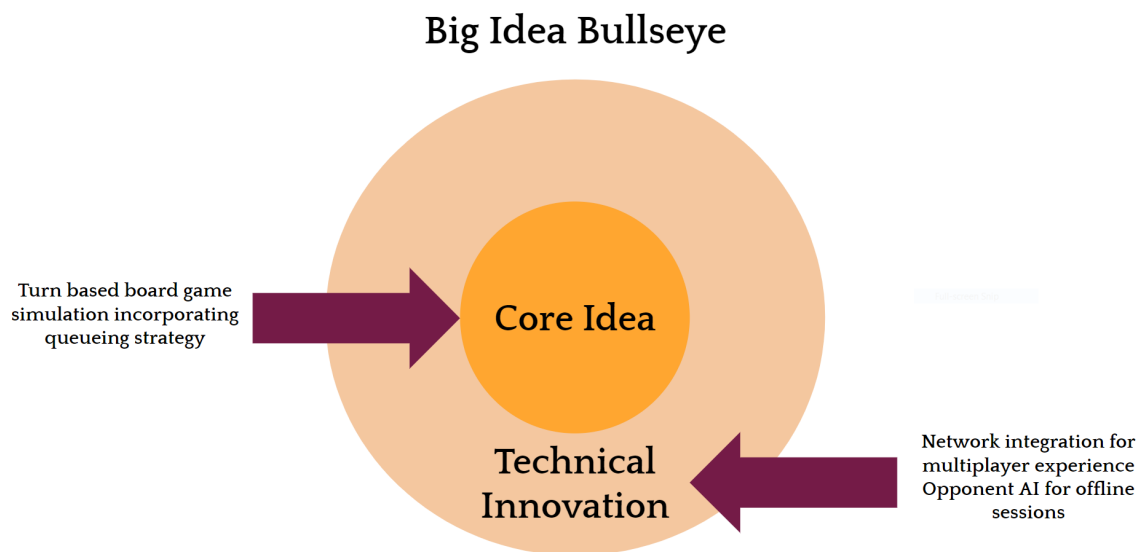


Figure 7: Big Idea Bullseye

1.4 Development Schedule

1.4.1 Layered Task Breakdown

We have divided our game elements and goals into five layers as shown below, in a manner that we believe is most crucial to the game and best sets us up to achieve incremental success at each layer:

Functional Minimum:

- Game logic
- Simple UI, fixed input (mouse or touch)
- Playable on single device

Our functional minimum is aligned with our core idea of building a board game simulation. The game logic involved is fairly complex due to the rules of the game, and our goal is to make sure that our fundamental game structure is capable of handling this.

Low Target:

- Playable on LAN
- Keyboard shortcuts & improved UI
- Higher quality visual feedback

Our low target aims to build further on our gameplay simulation by focusing a bit more on the UI available to the player. We also want to start providing the integration points to incorporate multiple players in the game environment.

Desired Target:

- Playable over internet
- High fidelity piece models
- Castle reward system

Our desired target aims to primarily further our technical achievement by making the game available to players at different locations via the Internet. At this stage we would also want to incorporate our castle reward system, which is a more complex piece of game logic meant to improve the end game experience, although not required for the fundamental functioning of the game.

High Target:

- 3D asset animation & shader integration
- Objective card system
- Friend system
- Tutorials
- Haptic feedback (platform dependent)

Time permitting, we want to add in quality-of-life improvements to the game such as tutorials and asset animation, as well as improve the strategic gameplay even further by adding in the complex objective card system. This would build further from the castle reward system added earlier. At this stage, based on our degree of satisfaction with our original technical goal of networking, we may embark on a secondary technical achievement, namely the addition of AI opponents into the game to improve the single player experience.

Extras:

- AI opponents
- Game lobbies
- Rank system & matchmaking
- In-game text/voice chat
- Cross platform

This section includes aspects that we do not expect to reach during the course. We would aim to improve our features even further to bring them to a highly polished and professional standard where they could support a formal game release.

1.4.2 Timeline

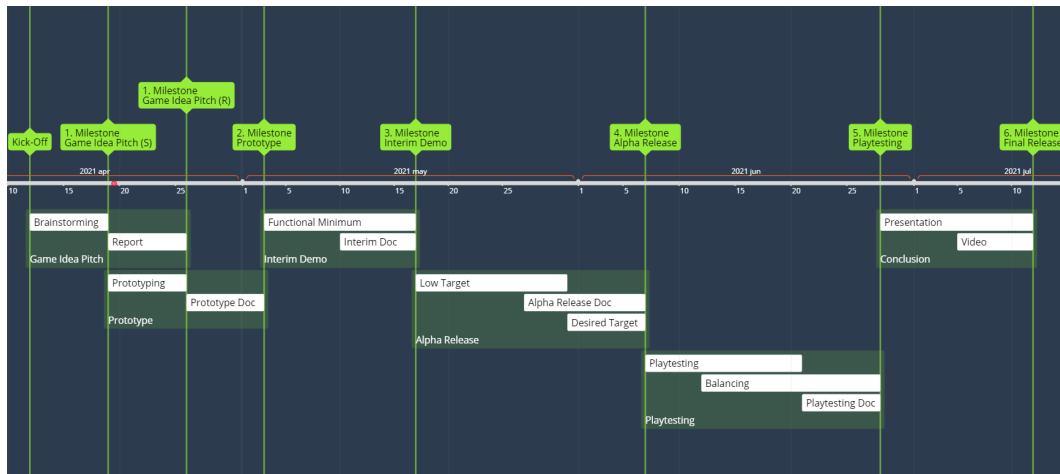


Figure 8: Development Timeline

Week	Goal	High Level Tasks	Assigned To	Estimated Hours
Week 1: 12.4 - 18.4	Game idea	Game idea	All	10 each
		Game idea pitch presentation	All	5 each
Week 2: 19.4 - 25.4	Document idea	Further work out gameplay mechanics	All	10 each
		Start working on prototype	All	5 each
		Write report	All	5 each
Week 3: 26.4 - 2.5	Prototype and research	Finish prototype	Shyam, Eugene	10
		Research technologies and fitting code patterns	Anas, Mert	5
		Prototype doc	All	5 each

Week 4: 3.5 - 9.5	Functional Minimum: Setup game logic	Handling basic player input	Anas, Mert	5
		Game field representation	Eugene	10
		Logic checks - E.g. FIFO	Shyam	5
		Mockup UI	Eugene	5
		Representation for all game pieces with placeholder methods	Anas, Mert	10
		Commands & game states	Anas, Shyam	10
		Research networking models/APIs	Mert	5
Week 5: 10.5 - 16.5	Functional Minimum: Implement game logic	Implement placeholder methods from last week	Anas, Mert	15
		Game Manager & turn mechanic	Shyam, Eugene	10
		Logic for action functions like Riot, Swap, ..	Shyam, Anas	10
		Connect UI to game logic	Eugene	5
		Prepare Game Manager for networking integration	Mert	5
		Interim doc	All	5 each

Week 6: 17.5 - 23.5	Alpha Release: Local player-game interaction	Collect player commands to perform action functions	Anas, Shyam	20
		Improve player-game interaction	Eugene	10
		Model network abstraction	Mert	10
Week 7: 24.5 - 30.5	Alpha Release: Network player-game interaction	Implement server/client architecture	Mert, Eugene	20
		Higher goal mechanics	Anas, Shyam	20
Week 8: 31.5 - 6.6	Alpha Release: Polish what we have	Implement network communication manager	Mert	20
		Testing & bug fixing	Anas, Shyam	20
		Polishing & visual improvements	Eugene	10
		Test local area network connection	All	2 each
		Alpha Doc	All	10
Week 9: 7.6 - 13.6	Playtesting	Prepare questionnaire for evaluation	All	5 each
		First playtesting session	All	5 each
		Identify possible issues	All	10
		Set up and utilize relay server to support connection over internet	Mert	10
Week 10: 14.6 - 20.6	Playtesting	Balancing & tweaking values (no new mechanics)	Shyam	15
		Bug fixing	Anas, Mert	15
		Second playtesting session	All	5 each
		Record video material during playtesting	Eugene	5
Week 11: 21.6 - 27.6	Playtesting	Evaluation	Anas, Mert	10
		Playtesting doc	Shyam, Eugene	10
		Look through current video material	All	2
Week 12: 28.6 - 4.7	Final Release	Final doc	All	15 each
		Presentation	All	5 each
		Record more video material if necessary	All	4 each
Week 13: 5.7 - 11.7	Final Release	Finish wiki page	All	5 each
		Cut trailer video	All	10 each

1.5 Assessment

We believe that the strongest features about our game will be the thematic binding and the deep strategic gameplay. Our target audience will be individuals who are comfortable with a long sit down gaming session of about 90 minutes, and enjoy strategic considerations and negotiations before each move.

1.5.1 Thematic Binding

Our game not only links strongly to the Order and Chaos theme of the course, but also uses this opportunity to present a satirical depiction of real-world scenarios. The presentation of moral choices in the game (maintaining order) is complemented by the ability to break ethical rules (create chaos) and perform ‘disgraceful’ actions in the game in the pursuit of self-interest. We want our game to stay true to its setting even outside of the course, such that players feel immersed in the world that we create and are engaged in the gameplay experience from start to end.

1.5.2 Strategic Gameplay

The turn-based nature of board games typically allows for careful strategic thought before each turn, rather than rewarding dexterity and motor skills. Our game would present the opportunity for multiple paths to victory and several contrasting options to the player at each turn, such that every decision made in the game would have a meaningful impact on the final result. As believers in the concept of board games being a social experience, our game would also include several avenues for interaction between players, both positive and negative.

1.5.3 Our Definition of Success

We want to evaluate our success in terms of the gameplay opportunities as well as technical accomplishment. From a gameplay experience we want the rules governing the gameplay system to feel challenging to just the right degree, such that players are constrained by the decisions that they can make, and rewarded for maximizing their gains from their limited pool of resources. Expressions such as ‘If only I had one more turn’, or ‘I wish I had one more action’ would fit in line with the tightly balanced nature of our gameplay. Our intention is to make the game easy to play mechanically but hard to master. We expect the final scores of our game to be extremely tight depending on player skill levels.

With regards to our technical accomplishment, the primary aim of digitizing board games is to make it feasible for individuals in different locations to play together. We thus want to provide a seamless experience with regards to UI and networking such that players can interpret the feeling of playing a physical board game. We expect that the digitization of a physical medium would create certain challenges. For instance, some aspects such as the tactile nature of board games would be difficult to replicate. However, we could compensate for this by making the less engaging aspects of typical board games such as rules enforcement, setup and cleanup much easier in a digital form. Our criteria of success would be if players are able to have a seamless gameplay experience with their friends, and feel mildly exasperated when they lose such that they are itching to play again.

2 Game Prototype

2.1 Prototype Description

Motivation:

Our goal with the prototyping process was to help identify the following:

- Solidify the core mechanics and clean up any inconsistencies in the gameplay loop.
- Identify challenges that could occur when transitioning from a physical medium to digital space.
- Identify areas of improvement that a digitized game provides over the traditional board game experience.

Approach:

In order to model our board game easily over a virtual environment, we used Tabletop Simulator (TTS) by Berserk Games. TTS is a physics sandbox that can be used for rapid prototyping and refinement of tabletop games. After uploading our assets into the simulator, we ran multiple tests to help identify issues with the gameplay loop and solidify the core mechanics.

We also researched existing digitized adaptations of board games that were recently released, in order to understand how they approached the digitization process. The games Scythe Digital, Root, and Battle Brothers were played multiple times, so in short this was probably the most fun research experience we had. We also looked up documentation of such releases online, and were able to connect with like minded developers from indie studios, to help understand some of the challenges they encountered and lessons learned when transitioning from the physical to digital space.

At all stages of the process, we kept referring back to our knowledge of the Unity game engine and related networking tools, to draw parallels in the way certain functionality could be implemented, as well as potential challenges and gaps that would exist. These learnings are summarized in the section following the gameplay details.

2.1.1 Gameplay Details

Although parts of the following section was already addressed in the previous Gameplay Details section, we made some minor additions which should be mentioned.

Prototype Summary:

The following image lists the overall prototype fully laid out in Tabletop Simulator.



Figure 9: Prototype Overview

2.1.1.1 Game Flow

Scoring:

Players earn Influence Points for the following categories, the player with the most IP wins the game:

- 1 IP per piece rescued by the player (includes villagers AND guild knights)
- Value from glory tracker
- Value from completed objectives
- Value from claimed inner city rewards

2.1.1.2 Additional Rules and Content

Objectives:

Objective cards are hidden goals that each player is trying to complete. Objectives may be completed by a player at any point during their Action or Reset turn. There is no limit to the number of objectives that a player may hold in their hand, or the number that they may complete on a single turn.

Objective cards will be of three types: Rescue (set collection of villagers), Observe (pattern formation in the crisis area), and Admin (miscellaneous logistics activities). A set of standard game objectives is present inside the city, which may be claimed once by each player with the use of a Guild Knight.

2.2 Gameplay Experience

Our gameplay experience varied on the basis of our overall experience with board games (casual gamers or hobby gamers).

Casual Gamers:

Our casual board game players found the game itself very enjoyable. While the game is not a brain teaser (in a good way, since we would want to keep the gameplay somewhat casual), it still requires strategic thinking and adaptability.

Overall, the game is quite sufficient when it comes to content. Elements such as objective cards and global objectives (inner city reward system) definitely help a lot keeping the game alive and avoid getting any stale vibes. The pattern recognition tasks were particularly cool since players don't know which pattern objective their opponents have, which in turn leads to more unpredictability (also toning up the chaos/order aspect of the game). Another aspect of the game which our casual gamers really enjoyed was the two turn mechanic (action/reset): it severely decreases the downtime for players and almost constantly keeps players in the loop, which could be particularly useful for a "digital" game since players usually tend to lose concentration faster/easier when there's no physical presence.

Hobby Gamers:

Our hobby gamers echoed most of the feedback of the casual gamers, with the additional note that the ability to form combinations and optimize actions in the game is quite high. This means that although each individual action is simple to execute, a player can get a high reward for strategically chaining actions in a certain way to accomplish multiple objectives simultaneously.

The above distinction helps the game fit its targeted audience, namely hobby gamers who would also look to play with their casual gamer friends. The low skill floor but high skill ceiling of the game means that a larger demographic of players can be involved in the game. The hidden nature of the scoring means that new players won't often feel out of the game, which helps them retain their engagement. The low luck in the game means that experienced players are rewarded for their planning.

2.3 Lessons Learned and Design Revisions

The lessons learned and subsequent design revisions made have been condensed into a single section, so that it is easier to flow from point to point. As discussed at the start of the report, there were three goals identified when entering the prototyping process. Our lessons learned and design revisions made have been organized across these goals.

2.3.1 Goal I

Solidify the core mechanics and clean up any inconsistencies in the gameplay loop.

- **Observation:** The abstract notions of the queue were not intuitive to players. In particular, a funnel structure is necessary for the FIFO system of the game to function, however this is an extension of the traditional interpretation of a queue. Multiple pieces can be at the same priority in later rows, in contrast to a single piece holding a priority at each slot in a queue.

Solution: Thematic changes were adopted where the villagers are attempting to enter a guarded city with the partially opened gates forming the required funnel shape. This also led to the ripple effect of renaming a few of the basic actions of the game to help it fit with this new theme of a guard authorizing an individual to pass.

- **Observation:** Gameplay loop breaks in a few edge cases. The gameplay loop was found to break in rare cases. One instance is where a player has no available workers to reset on the action mat since they have been targeted by multiple players with the poach action. At the same time, they could not counter poach if they were out of objective cards to use as payment.

Solution: The Countermand reset option was modified from a card swap to instead allow the drawing of a card, followed by card return only if a player had two or more cards in hand. This would allow the stuck player to guarantee their use of the poach action, and also provide a useful optimization tool to other players who were out of objective cards, even if they didn't want to poach.

- **Observation:** Further strategy potential utilizing the queue exists. The rules around FIFO help build a fundamental system on top of which a lot of gameplay is layered. While the incentives of the higher level layers are easy to see (such as rescue of a piece, completion of an objective etc.), the direct value of FIFO is a bit hard for players to interpret, and could thus be mistakenly viewed as unnecessary to the game.

Solution: Our solution with regards to this is to explore some additional incentives around FIFO which occur at a lower layer, such that players can see immediate tangible benefits of smart queuing.

2.3.2 Goal II

Identify challenges that could occur when transitioning from a physical medium to digital space.

We consider this our most important goal in the prototyping process, since it could make our digitization process orders of magnitude easier or more difficult.

- **Observation:** Objective cards and inner city rewards will be complex to code. The nature of the objective cards and by extension inner city means that each individual card would need a custom set of rules or checks to decide if a valid completion has occurred.

Solution: Fortunately, all of these checks would derive from a fundamental rule set required by the game system to operate. Our solution is thus to move this requirement to the high target section, so that we can tackle it after the rules engine is stable.

- **Observation:** Drag and drop is clunky. Our experience with interacting with pieces in Tabletop Simulator gave us a good precursor into how gameplay experience can be ruined by latency. The drag and drop effects can be extremely clunky and create frustration for the player.

Idea: Having explored other digital board games, we are exploring the idea of using a flat UI in the form of a click and select to interact with pieces. This would be followed by the required drag and drop animation being loaded from the main system instead.

- **Observation:** Game logic is not the same as rules. Although the rules of the game are already complex, enforcing them with a rules engine would require further additions and logic checks.

Idea: We plan to split our core game logic into two parts. The first part would include performing the actions assuming that the players maintained the validity of rules internally. This is similar to how sandboxes like Tabletop Simulator operate, placing the onus of responsibility on the players instead of the engine. This step would allow us to first incorporate all the possible actions and required animations and effects into the system. Our second step would build in a rules enforcer to check if actions are valid. This would be consistent with our tiered approach of eventually adding objectives as a layer above this.

Some challenges where board games can't be rivalled easily are:

- **Observation:** Tactile experience cannot be easily replicated. Physical games are great because players can touch and feel components in the game. This is naturally an extremely difficult feat to replicate in a digital version.

Idea: We will continue to explore solutions, but our preliminary solution is to add satisfying sound effects for tactile interactions, along with other aural cues that may not exist in the physical space. The hope is that this can provide a viable trade-off between the two senses.

- **Observation:** Board games provide a social experience.

Idea: We are aiming for good networking, but even this may not be a perfect substitute for a table of interacting board gamers. The addition of text and audio could perhaps help bridge some of this gap.

2.3.3 Goal III

Identify areas of improvement that a digitized game provides over the traditional board game experience.

Our final goal seeks to take a positive look at the opportunities that our prototype provides.

- **Observation:** Rulebooks are cumbersome to read in board games, and can be a huge barrier to entry.

Solution: A digital adaptation provides a unique way to have an interactive tutorial, such that players can learn as they play. This could help expand the reach of traditionally difficult board games.

- **Observation:** Administration tasks such as setup and cleanup are cumbersome, which is also applicable to our game. Complex board games often have myriad components with detailed setup instructions that may even vary across different scenarios and player counts.

Benefit: Digital board games provide a unique way to automate all of the background admin tasks related to board games, drastically speeding up the setup and teardown of sprawling games. This can in fact help increase the reach and replayability of such games.

- **Observation:** FIFO Tapping and rules in general are complex to remember in board game.

Benefit: Our game system would eventually be able to do the rules enforcement and tracking for the players, greatly simplifying the process and providing visual cues as required. For instance, rule checking could be performed after an action is performed, and cues such as lighting could be used to highlight viable options for the player.

2.3.4 Conclusion

In summary, our prototyping process was an extremely illuminative exercise that has helped us identify potential challenges with our game design process as we enter the next phase of development. Based on some of the changes we have made to account for the challenges, we feel well equipped to proceed with development.

3 Interim Demo

This part of the report will discuss our development process for Flee Fi Fo From, and an update of our status at this interim stage. We will then cover the challenges we encountered as well as our next steps.

3.1 Design Approach

For our development process, we first outlined the architecture for the project. Following this, four subsections were identified as illustrated in Figure 1 below. These sub sections or modules would allow for the compartmentalization of development efforts within a particular module to allow for independent progress across the board. Once a stable code base is ready, we will integrate these modules with each other for the complete version of the game.

The subsections are:

- **UI Design:** Presentation layer that the user interacts with.
- **Game Logic:** Defines the logic for individual actions within the game.
- **Game Manager:** Central control of the game which connects to the other elements.
- **Networking:** Provides the capability for gameplay across different clients.

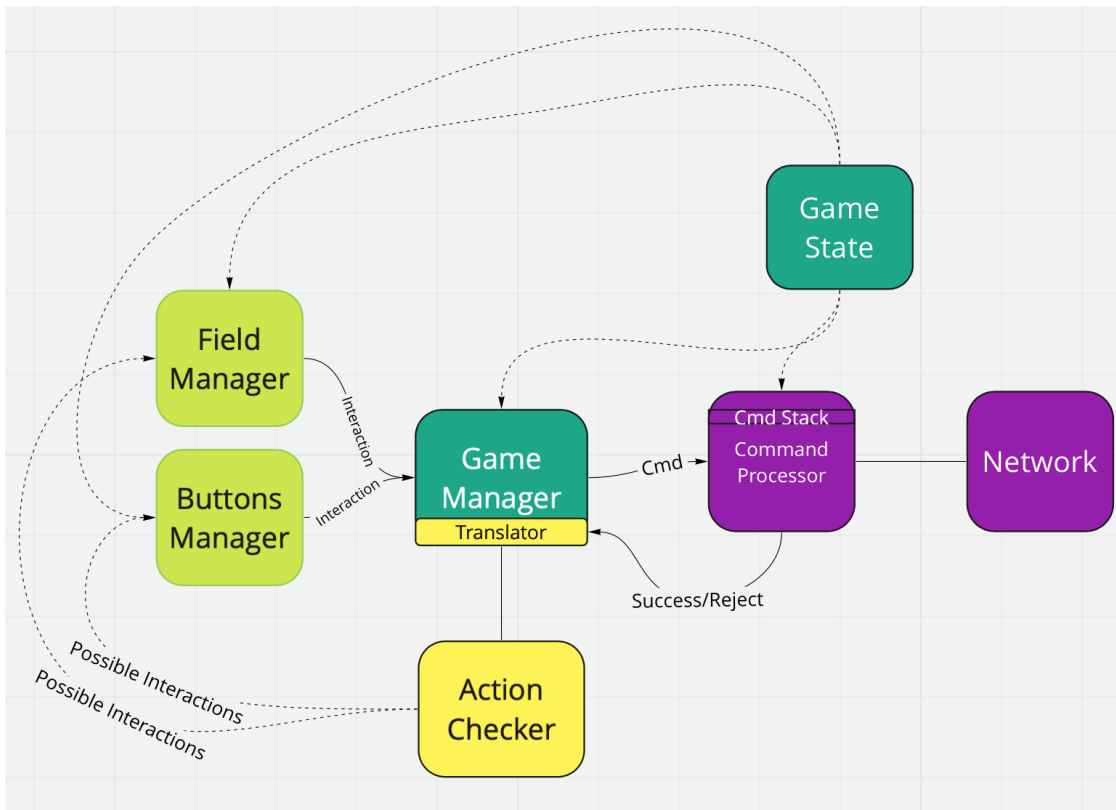


Figure 10: Design Architecture

3.2 Interim Status

In the following section, we will discuss the status update for each individual subsection.

3.2.1 UI Design

This layer consists of the visual components needed to play the game (perform actions part). In its current state, the UI already allows for a player to interact with the game board and overlay UI elements. This includes the base field commands and undo commands. For each command the interactable pieces are highlighted. A simplified version of the riot command has been implemented, allowing for a piece to be injured. This in turn allows for the testing of the Revive command. Furthermore, preliminary work on the worker resets has commenced, with the ability to depict workers per action space, as well as remove workers via the reset commands. All objective related actions are implemented as placeholder methods, until we start working on the objective mechanic in our higher goal tasks. The honor tracker above the field is already depicted in our desired UI style. An illustration is shown below in Figure 11.

The UI leverages the use of Button Manager and Field Manager classes to depict the player interactable actions and the game board respectively. These classes would interface with the Game Manager to allow for rules enforcement.

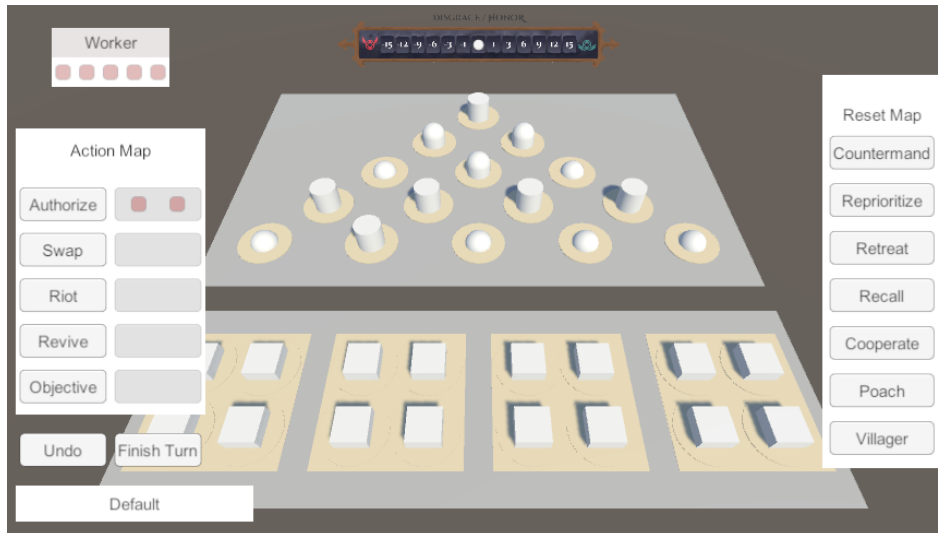


Figure 11: User Interface

3.2.2 Game Logic

This piece of the project contains the game logic, which adapts the rules of the board game into a code structure that can be processed by the game manager.

The architecture consists of a set of classes for each of the game pieces and the available actions that a player could take. The viability of these actions are evaluated via checker functions within each action (and corresponding reset). These checker functions leverage helper functionality provided by the game manager. Actions and resets selected by a player are then processed by the game manager, and added to a command stack. Further details can be found in the game manager section.

3.2.3 Game Manager

This layer includes the Game State data structure, which is a purely logical representation of the board, workers, player scores, etc. Due to the complex nature of many rules the game has, the Game State is augmented with utility helper functions that allow expression of such complex logic in an intuitive and reusable way, and allows for faster iteration on the ruleset of the game.

```
private void EnableRiotPath(List<Tile> path)
{
    var last = path[path.Count - 1].Position;

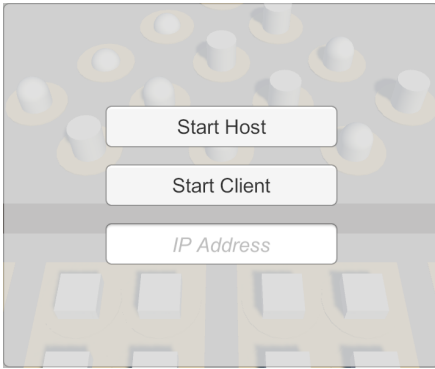
    GameState.Instance.TraverseBoard(p => {
        var tile = TileByPosition(p);
        var meeple = GameState.Instance.AtPosition(p);
        tile.Interactable = (
            last.CanMoveTo(p) &&
            (
                meeple == null ||
                [
                    meeple.IsHealthy() &&
                    meeple.GetType() != typeof(DKnight)
                ]
            )
        );
    });
}
```

Figure 12: Game Manager Code Snippet

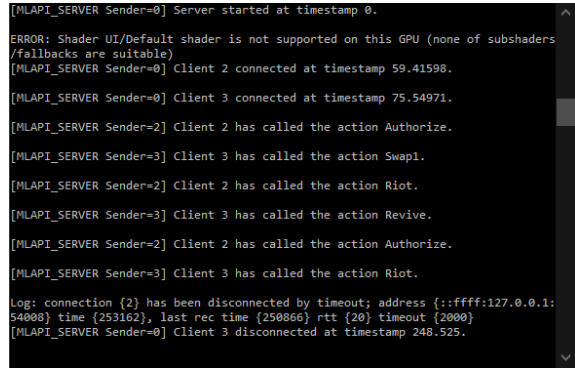
Figure 12 contains an example code for highlighting tiles that can be selected for a step of a riot action, and the expression basically boils down to a traversal of the board, and marking tiles that can be moved to from the last tile of the riot, and they are either empty or a healthy villager (not a knight) is on said tile.

The Game State also provides an Observable interface, allowing the visual layer to listen for changes to particular values and reflect them efficiently on the board (e.g. position of a villager or whether they are injured or not).

The Game Manager (still WIP) has the responsibility of translating raw interactions (clicking on a button, a piece, etc) into game commands (e.g. swap two pieces, riot, etc.) based on context and game rules. It acts as a central bus between various other modules: the visual elements communicate interactions and their feedback to the Game Manager, Action checkers communicate with the Game Manager to retrieve a list of possible interactions at current state, and passes logical commands to the command processor which in turn results in updates in the Game State and will be synced via network.



(a) Connection UI



(b) Server Logging

3.2.4 Networking

The networking layer has been fully decoupled from the remaining system, with the exception of command processor, in an effort to realize a more adaptable system, in turn also leading to an ease of parallel development of distinct layers and subsystems. This allowed for a simultaneous development of game logic/manager and the networking layer, while also rendering a smooth networking integration possible. With the current state of the networking, we are able to host a session on the local area network, which will be further extended into internet connection over the course of remaining development phase.

Client-Server model was realized by employing a dedicated server to host the session in the form of a headless server build and multiple players connecting the host as clients. In contrast to a direct communication between the clients, the model only allows a communication of clients through the host device, which is also responsible for processing commands and maintaining their history for potential undo operations. An initial version of the communication manager was implemented to publish actions through the server to other clients, utilizing remote procedure calls and executing commands locally on each client.

Furthermore, a simple UI was employed to establish connection for the clients, along with a server event listener, which proves to be especially helpful for debugging purposes by logging information such as server start and client connection/disconnection.

3.3 Challenges

We encountered a few challenges during our planning and development efforts. Firstly, the rules-heavy nature of the game meant that we had to decide our architecture and integration points before implementing any base functionality, in contrast to iterative development from a complete barebone minimum game in the typical layered approach of the course. We decided to split the architecture into subsections to allow for independent development before merging. The advantage of this approach is that we were also able to commence working on items from future layers at an earlier stage, for instance the playtesting of the gameplay and rules, as well as the networking components.

Secondly, our desire to implement an undo functionality created significant design constraints. This affected the choice for the command pattern architecture, as well as the use of a command stack for easy reversion of previous moves by a player. Finally, we are working on constantly updating the rule set of the game, modifying and trimming the rules where necessary to lower the cognitive load on the player and allow for a streamlined experience.

The process of playtesting the rules of the game in the board game form thus far has yielded feedback regarding the complexity of certain actions, specifically around the myriad options for Authorize. Furthermore, the FIFO system of tracking has proven to be challenging to track, and also has the potential for tactile mistakes during tapping. The current piece orientation also presents visual challenges when viewed from different sections of the board.

In light of this, a modified FIFO system using an externally tracked priority system across each piece type is being experimented with. Furthermore, the Authorize and Riot specific rules for the different piece types could now be removed, as they could be internally accommodated in the FIFO system. The change of these game rules will simplify the gameplay, as well as the corresponding game logic. These changes will be tested before implementation in the alpha release.

3.4 Next Steps

In terms of next steps, we will be working on integrating the independent subsections into a cohesive whole, as well as removing any duplications currently existing as placeholders. In addition, our next steps for each layer include:

- **UI Design:** We will be focusing on adding the connections for remaining worker actions, and identifying areas of duplication between UI and Game Manager. Following this, we will be performing visual improvements to the game interface and assets.
- **Game Logic:** Apart from the addition of worker based logic, we will be focusing on verifying the modified rules before integration with the code base.
- **Game Manager:** We will be focusing on decoupling overloaded classes such as Field Manager, Button Manager and Action Checker, and absorbing some of this functionality within the game manager module instead. We also need to expand the logic for the worker rules.
- **Networking:** For networking, we will be working on adjusting the system for connection over the internet. Following this, we will be adjusting the connection manager and corresponding UI to support DNS address and/or server name along with IPv4 address. Finally, we will be implementing fallback methods to handle potential server connection errors.

4 Alpha Release

This part of the report will discuss our alpha release for Flee Fi Fo From, and an update of our status at this stage. We will then cover the challenges we encountered as well as our next steps.

4.1 Status Overview

As discussed in the interim part of the report, we split the project into four subsections to facilitate independent development, and have now integrated a majority of the components. At a high level, the game logic and UI is complete with all components in place. The game is now fully playable on a single device. The networking branch will also be integrated shortly to facilitate play over the internet.

4.2 Game Logic and UI

This layer consists of the visual components needed to play the game (perform actions part). In its current state, the UI allows for a player to interact with all of the pieces on the game board, as well as the UI elements including actions, resets and priority tracker. The riot command is now fully implemented, along with undo functionality for all sub commands. The worker loop system is also fully implemented now, allowing for worker depiction per action space as well as the removal of workers via the reset commands. All objective related actions are implemented as placeholder methods, until we start working on the objective mechanic in our higher goal tasks. These facets will be toggled off during our alpha release and playtesting sessions. The honor tracker above the field is depicted in our desired UI style, along with a scoreboard to track victory progress across players. An illustration is shown below in Figure 11.

The UI leverages the use of Button Manager and Field Manager classes to depict the player interactable actions and the game board respectively. These classes have been significantly reworked and will continue to be refactored to create as much decoupling as possible between visuals and logic in the game.

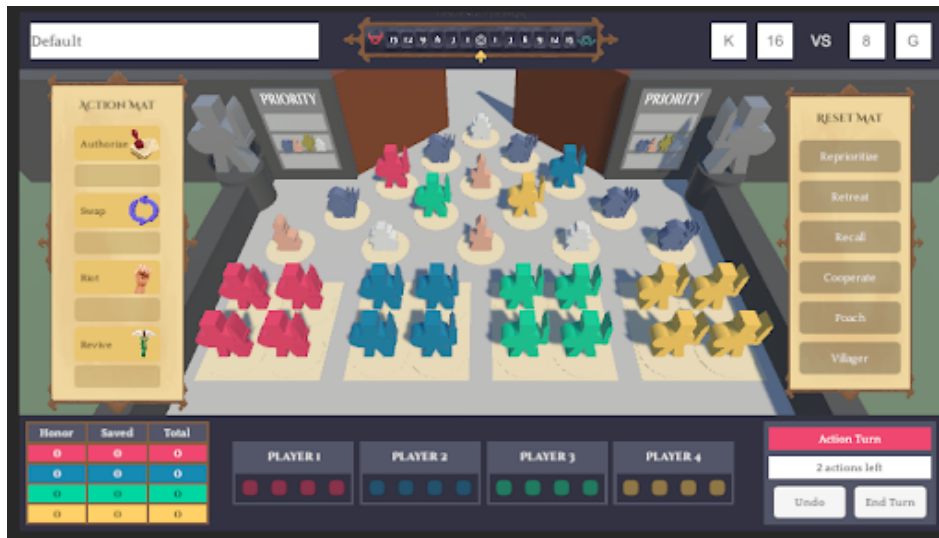


Figure 14: User Interface

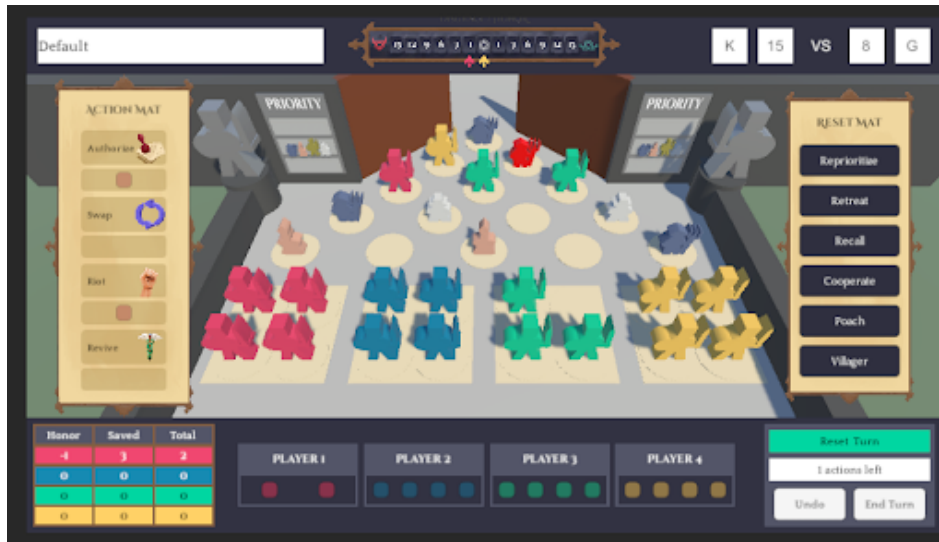


Figure 15: Mid-Game State

4.2.1 Game Logic

The Observable interface of the Game State has been extended to encompass as many of the game components as possible, allowing the visual layer to listen for changes to particular values and reflect them efficiently on the board (e.g. position of a villager or whether they are injured or not).

The Game Manager now acts as a central bus between various other modules: the visual elements communicate interactions and their feedback to the Game Manager, Action checkers communicate with the Game Manager to retrieve a list of possible interactions at current state, and passes logical commands to the command processor which in turn results in updates in the Game State and will be synced via network.

Rule Changes:

In light of playtesting feedback, a modified FIFO system using an externally tracked priority system across each piece type has been implemented. The UI for this is already available, while the logical connections are in progress.

Networking:

The networking capability of the game has been enhanced from LAN to internet connection, in order to reach a wider user base. This has been performed by means of an AWS Linux instance deployed with a relay server. The board initialization (random) has been synchronized across players, such that each local client works with the same original seed and sees the identical game state. The logic for the turn control (current action and reset player) has also been completed. The synchronization of the actions across each local instance will be implemented in the next phase.

4.3 Challenges

Our encountered challenges are a continuation of those identified during the interim phase. The rules-heavy nature of the game has been greatly simplified, but might still have potential to reduce the cognitive load on the player further. A by-product of these myriad rules and action combinations is that there is the potential for edge cases that are contrary to what the game logic would expect. Our approach to this is to playtest internally and externally as much as possible, in order to identify and address as many bugs as possible. This process has already commenced. Lastly, in order to keep the stress on the network as low as possible, we are regularly reviewing our code base to identify any refactoring opportunities. Some of the earlier challenges around undo functionality and command stack have already been addressed with our current architecture, but we will continue to test them for bugs, as listed above.

4.4 Next Steps

In terms of next steps, we will be working on preparing the game for playtesting, as well as further refining the current version.

Game UI and Logic:

For the game UI, we will be focusing on visual improvements to the current interface, including piece models and animations. We also plan to implement a simplified version of the inner castle, which contains all non-objective related castle spots. We want to round up the player experience via helpful tooltips or messages on the UI, which would provide guidance on the actions that a player could take on their turn.

From a game logic standpoint, the integration with the existing priority UI is in the works, such that the various actions and resets earn the corresponding honor effects based on priority. We are also interested in monitoring and refactoring the code base of the game to make sure that the game components are decoupled as much as possible, as well as to remove any existing duplication of code. We will also be verifying that the modified rules work well with the game, while continuing to explore further opportunities to clean up and simplify any rules that are cumbersome or unclear.

Networking:

For networking, we will be working on synchronizing the player actions across the network, so that all clients have the same local copy of the actions. We are also aiming to implement fallback methods to handle potential server connection errors. These steps will facilitate the playtesting of our game across the Internet, as we move into the next phase of the project.

5 Playtesting

This chapter will discuss our playtesting approach for Flee Fi Fo From, and the feedback that we received from our play testers. We will also briefly discuss some changes that we anticipate making as a result of these suggestions.

5.1 Playtest Setting

5.1.1 Scope

The focus of the sessions was on the overall experience including gameplay, theme, user interface, visuals and any other critiques. In order to standardize our playtest feedback across sessions, we used a stable branch of our project for all playtests. Due to network disconnections and issues with action syncing, we decided to omit the networking piece from our playtests. We also made a conscious tradeoff to incorporate a fully fledged priority system into our game at the expense of the inner castle and preliminary objectives that we had originally planned to implement. These pieces could be added at a later stage, not as part of this course.

5.1.2 Playtester Selection

Our game is meant to simulate a typical strategy board game experience, thus being a 60-90 sit down kind of exercise. Our target audience would thus be hobby and enthusiasts board gamers who would enjoy such an experience. We decided to focus our playtester selection criteria around our target demographic. This was done by playtesting with gamer friends, as well as relative strangers on virtual board game playtesting servers on Discord. In all, we were able to receive feedback from more than 15 individuals.

5.1.3 Testing Method

We used a combination of video conferencing and screen share tools to conduct our playtest sessions. This involved one single game instance being presented via screen share, and interacted with by participants via screen control. Contrary to blind playtesting experiences with digital games, where players are meant to find out information on their own, we adopted a slightly hybrid approach. We presented a brief introduction to the game including the theme and setting, as well as an overview of the possible actions available to the player. We believe that this was a necessary step that would be in line with playtesting actual board games. Most competitive board gamers would not embark into a game session without having at least a high level idea of the game.

Following the introduction, our players would jump into the game session. We would ask a few in-game questions if the opportunity arose, or else be silent spectators. We decided to rely on our in-game help messages for specific rule questions during play. Following the game session, we asked players to answer a brief post game survey, which included a few quantitative rating questions, as well as plot a quick response on a play matrix. Since our players were on voice chat, we collected most qualitative feedback in the form of a post game discussion following the survey.

5.2 Playtest Results

5.2.1 Play Matrix

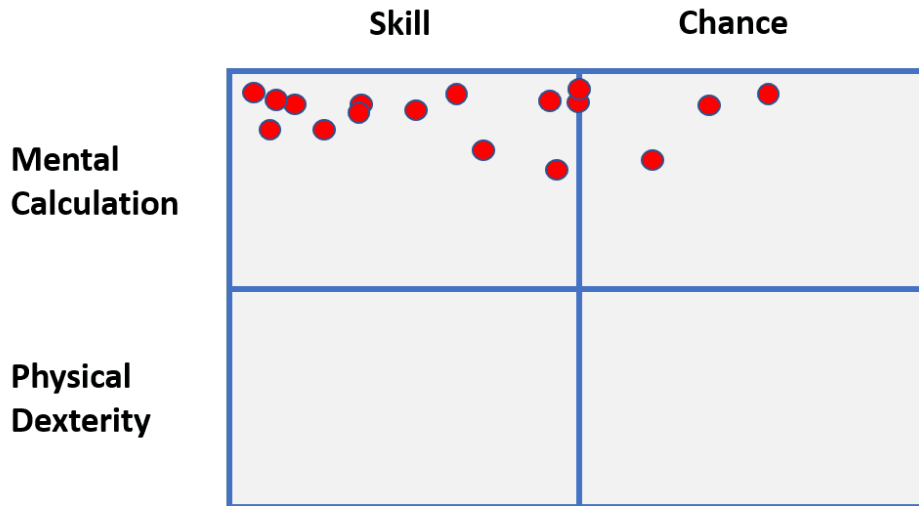


Figure 16: Play Matrix Results

Where players thought the game currently is:

As can be seen from the matrix, a majority of the opinions were in the Mental Calculation and Skill quadrant, which is what we expected for our game. However the opinion of the degree of Skill required appeared to vary across players, with a fair number of responses moving into the Chance end of the spectrum. On questioning the playtesters, we believe that this opinion stemmed from the presence of some randomness in the game with regards to the villager draw.

What changes the players would like to see to get it to where we want:

The Chance quadrant responders mentioned that they would like some degree of probability control over the villager draw in order to be able to choose what got added to the queue. Furthermore, they felt that the queue movement was not always consistent across players, and that sometimes limited the reset options they had. We will be considering these opinions to see if we would like to make any changes, or whether we are comfortable with the elements of luck in the game. A second critique that began to emerge from this stage was that although the mental calculation of the game was quite high, the reward for it was not necessarily at par with it. We will discuss this further in a later section.

5.2.2 Quantitative Feedback

We asked our testers to rate the different aspects of the game on a scale of 1 to 5, with 1 being Terrible and 5 being Terrific. The results can be seen in Figure 17 below.

On a scale of 1 (Terrible) to 5 (Terrific), how would you rate the following aspects of the game?

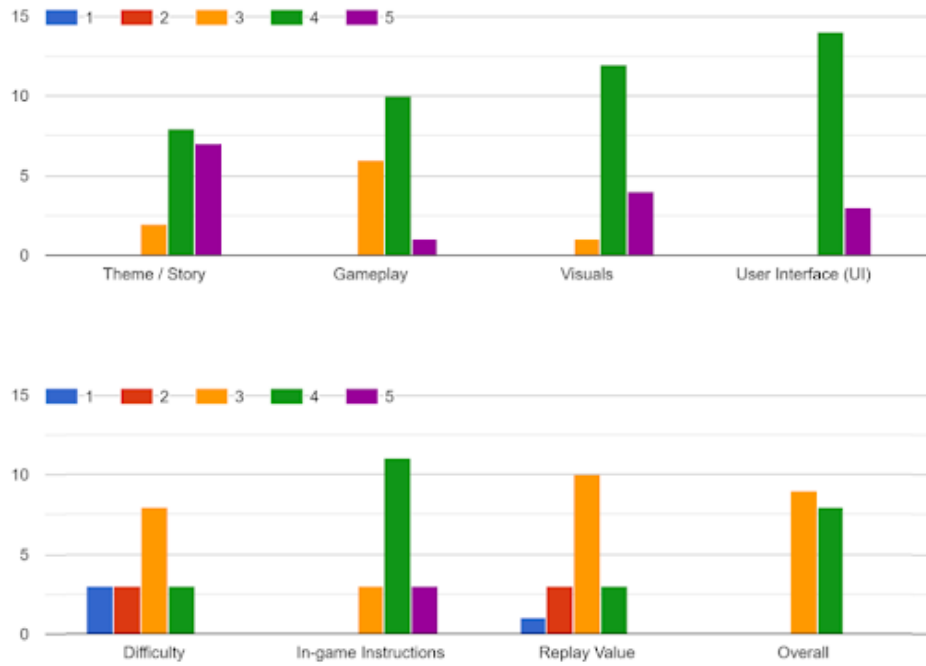


Figure 17: Ratings for Different Aspects of the Game

When looking at the positives from the above ratings, we can see that the theme was consistently well received. The gameplay trended well, with its unique aspects and mechanics being noticed in particular. The visuals and UI were generally well received, while no issues were found with in-game instructions. In fact several players found the in game instructions to be more helpful than what they could have received with a tabletop experience.

When looking at the negatives, what stands out is that the replay value of the game was deemed to be below par. The game overall trended at slightly above average, which is lower than what we would want it to eventually be. On further discussion, we found that this overall rating was heavily influenced by the lack of strategic choices and replay value, which we will discuss later.

We also included survey questions to gauge the audience’s perception of their fit for the game, as well as what they thought about the game length and difficulty. The results are summarized below.

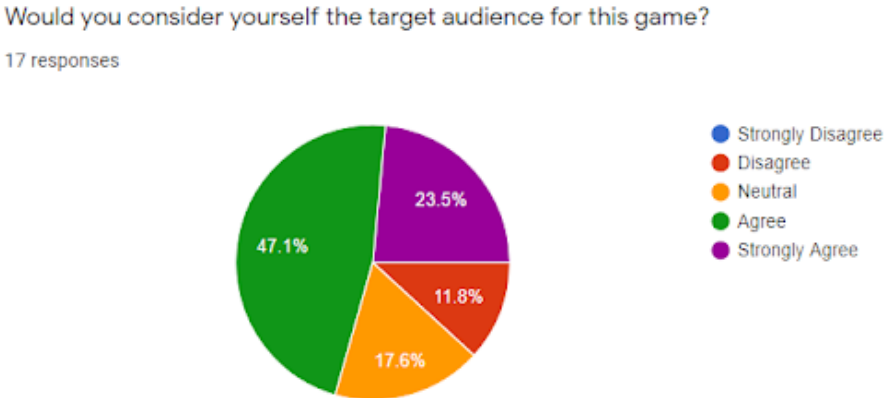


Figure 18: Target Audience Fit

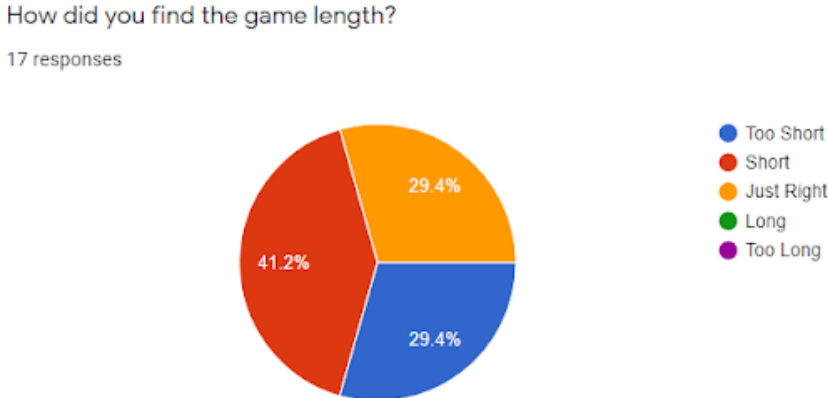


Figure 19: Game Length Rating

From the above two figures, as well as the difficulty rating in Figure 2 earlier, we can make a few observations. Firstly, the selected playtesters were an apt fit for our game, since they tend to be experienced board gamers who play often. Secondly, the difficulty level of the game (Figure 17) varied a bit across players, however in general it was considered to be on the easy side of the spectrum. In fact, one player remarked that “The math is not difficult, but it doesn’t feel worth it”. Finally, a majority of the respondents found the overall game length to be shorter than preferred. We think these are good pieces of feedback to have, because it shows that we have plenty of room to expand on the difficulty level and game length by adding components that would provide more strategic value. We will discuss this further in the section on changes and next steps.

5.2.3 Qualitative Feedback

We followed up from our survey questions with a post game discussion. We generally asked our participants questions regarding their most and least favourite aspects of the game, as well as things that they found frustrating or that they would like to change. The exact mode of the questions varied based on player responses, and resembled more of a conversation.

In general, we found that players really enjoyed the theme as well as some unique aspects of the gameplay. They found the UI and visuals to be clear and polished to meet their requirements, and felt that the help messages improved their experience over a traditional tabletop setting. Some specifics are listed below.

Theme:

As mentioned above, the players were able to connect well with the game setting, and believed that it represented the Chaos and Order theme well.

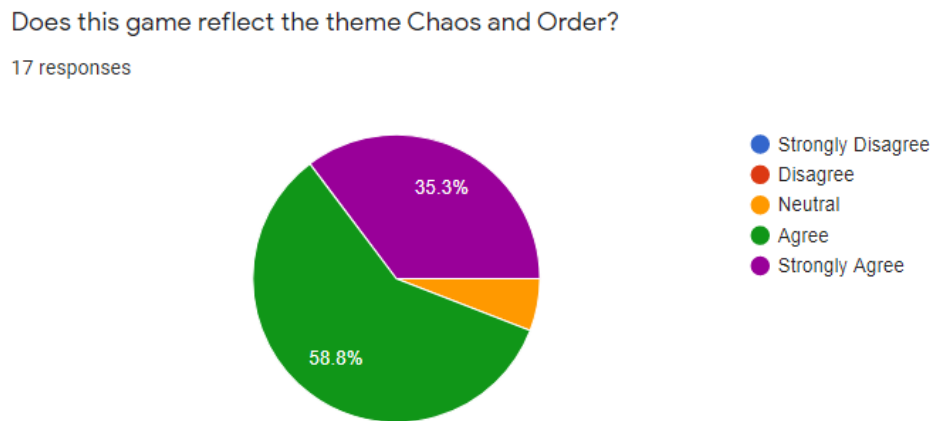


Figure 20: Feedback on Thematic Fit

The priority mechanic was very well received, and players felt that there was tremendous potential with this unique system. They also felt there was strong thematic resonance with real world aspects such as immigration and refugees, with some even suggesting vaccine priority queues as a linkage. The only significant feedback they had here was that there were even more opportunities possible to link the priority mechanic with other aspects of the game. One response stuck with us in particular:

*”I see some British humour here ...why are medieval people lining up in an orderly way in the middle of a **** storm..and being awfully polite about it”*

UI and Visuals:

Both the quantitative and qualitative feedback seemed to indicate that the UI and visuals for the game were polished, clean and intuitive to understand. The undo functionality was particularly praised, with players noting that mistakes were even easier to undo in our digital version than a typical tabletop version. The information messages on the screen at every game state were well received by the players, who felt that it greatly contributed to their understanding of the game. One playtester even remarked that *“This is a heck of a lot more useful than a rule book.”*



Figure 21: Example In-game Prompts

With regards to suggestions, the players felt that some of the buttons, in particular for the workers, were not always easy to see. They would have liked to see this slightly bigger, and if possible also ‘pop out’ better when they were interactable. In addition, they also had some similar feedback with regards to tile movement, in order to better indicate the selected tiles and possible forward movement for the next tile. We also received some paradoxical feedback, where a couple of super competitive players would have preferred that there was no live scoreboard in the game, similar to how it would be in a tabletop version. In their mind *“Seeing the scores makes it too easy to know when you’re winning. I like being forced to gauge the scores.”* We are not yet sure if this should be considered a disadvantage of our digital version, or whether we can consider it as a niche opinion.

Gameplay:

Finally, with regards to the gameplay, we received a fair amount of feedback in this area. On the plus side, the priority mechanic was very well received, with players believing that this could have great further potential. The double sided end condition of the game was also something that added a dramatic element and strategic nature to both sides (action and reset) of the gameplay. In fact, the action and reset concept in general was well liked by our testers, with multiple individuals noting that it significantly lowered downtime and increased player engagement.

We did however receive significant negative feedback regarding the overall strategic opportunities and the replay value of the game. Players were not always able to see the value of saving their knights, and also felt that there was not enough variety in the options that they could pursue. One remark that particularly stuck with us was when a tester said *“I think all the fundamental pieces are there to make this game really good, but it needs a bit more variety and oomph to it before I would play it again.”*. We are overall quite glad to hear this feedback, because it indicates to us that our game has the potential to be successful if we are able to incorporate some of the changes we are contemplating. There were also a couple of edge cases and negative play observations we made regarding the Poach action in the game. Players felt that they could be unfairly targeted by players within the game and left without resettable workers, leading to kingmaking scenarios. Finally, they felt that there could perhaps be a bit more Honor earning opportunities in the game, as it felt quite penalizing currently.



Figure 22: Broken Game Loop with Poach

5.3 Changes and Next Steps

We are considering several changes based on the feedback we received. In terms of easy wins, the edge cases and UI based feedback we received should be quite easy to fix, and we also plan to remove the Poach action from the game to prevent negative play and game loop breakages.

In general, we felt that the strategic decisions and replay value of the game would be dramatically improved once the objectives and inner castle are in play. This would also target the feedback around the game length and difficulty level currently, since we feel that we have plenty of room to increase both. While the inner castle was originally targeted for our alpha release, we made a conscious decision to implement a full fledged priority system instead. Based on the success and reception of the priority system, we feel comfortable with this tradeoff. We will likely implement a slight proxy buff to the knights' point values to simulate the inner castle for now.

Overall, our key priority is on resyncing the networking piece of the project and refining the components that we already have in place, in order to have a fully playable and polished version of our game for the final presentation.

6 Conclusion and Final Release

This chapter will discuss our final release for Flee Fi Fo From, based on planned refinements and changes made after receiving feedback from our play testers. We will also briefly discuss our overall experience with the course project.

6.1 Summary

The following screenshot shows the final version of our game for the purpose of this course. We have successfully built a digital version of a board game Flee Fi Fo From, incorporating a turn based system with deep strategic gameplay. Our game offers a varied take on the Chaos and Order theme, and features a unique Priority mechanic as well as a Action-Reset turn mechanic for downtime reduction. The game core is supported by command and client server architecture patterns, allowing for consistent serialization and synchronization of actions across players. The game can be played locally with up to 4 players and also offers multiplayer connectivity over the internet.

We believe that our digital version is able to offer some unique advantages over a traditional board gaming experience, via the supportive in-game help prompts to guide players through the game, as well as immersive sound effects that would not be possible in the physical medium. Furthermore, the online connectivity allows for access to a player base that may not normally be possible due to geographical constraints.



Figure 23: Final Release

6.2 Changes since Alpha Release

The game scope evolved over the course of the project, and was accompanied by subsequent modifications to our development schedule as we deemed fit. The following changes were implemented in our game between the alpha release and the final version.

6.2.1 Networking

The most significant change since the alpha release revolves around the networking capabilities of the project. At the time of the alpha release, the game was fully playable with up to 4 players on a single device. Due to challenges with network stability and synchronization, the networking branch was also not used for the playtesting phase, which was instead done via local screen share.

The framework for the networking capabilities was built by making every action in the game a serializable command. Furthermore, the game state including random draws were also made serializable to allow for synchronized seeding across all the clients.

For the final release, full networking capabilities over LAN as well as Internet is now possible. The Internet multiplayer utilizes an MLAPI relay server for connectivity. Port Forwarding on the side of the server host is also recommended to add to the network stability and prevent some synchronization issues. These steps have been listed in the project documentation.

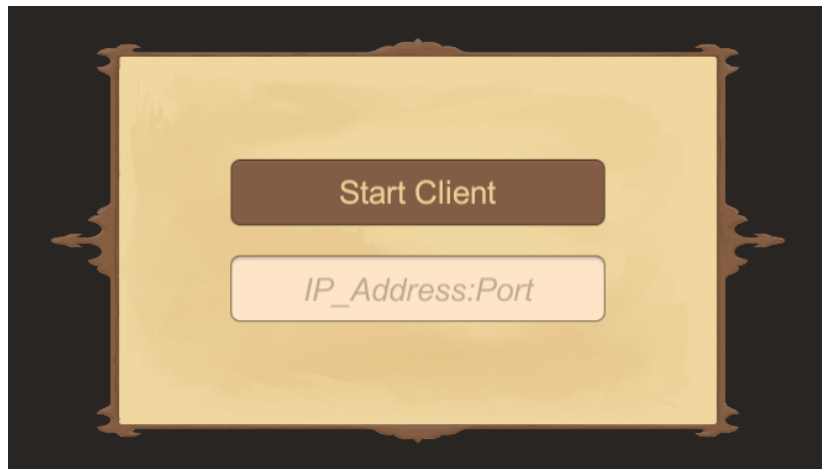


Figure 24: Online Multiplayer Startup

6.2.2 Complete Priority System

In the duration between the interim and alpha release, we had made a conscious decision to implement a priority system in the game instead of the tapping mechanism for FIFO tracking. We believe that the advantages that it brought us in terms of simplified gameplay and visuals would be significant. This new system came at the expense of the inner castle and objective card system, which we had originally planned to target for our final release. Between the alpha and final release, we were able to complete our priority system such that it is now integrated with the core actions of the game, and feeds into the corresponding glory effects. This fundamental system could also be leveraged by the objective cards in the future.



Figure 25: Priority System with Reprioritize and Glory Effects

6.2.3 Sound

While sound is typically a vital component of digital games, we had scheduled sound to be much later in our priority list of tasks, mainly due to the fact that sound is not typically a medium that is encountered in physical board games. Once we had the rest of the core components in place, we were also able to implement background sound effects and music at the relevant sections in the game, which thus brings the aural elements into play as well. We believe that this further enhances the value of our digitized board game, providing an experience that would not normally be possible. We hope to expand on this domain further by adding more original sound tracks and effects, if we decide to develop the project further beyond the course.

6.2.4 Menu Sections

We also added a main menu to the game containing toggleable options to choose between local and multiplayer gameplay, as well as to modify the player counts.



Figure 26: Main Menu

6.2.5 Player Help

In order to facilitate an easy gameplay experience for the player and to assist them with information when they were stuck, in-game help messages had been added to the game since the alpha release. These are displayed in one corner of the screen and reflect what a player can do at every step during their turn. Furthermore, a rules section was added to the main menu to provide information about the board game related rules of the game. These are meant to supplement the in-game help messages. We believe that this combination of aids will help familiarize the players with the gameplay much faster than what could be conveyed via a physical version of the game.

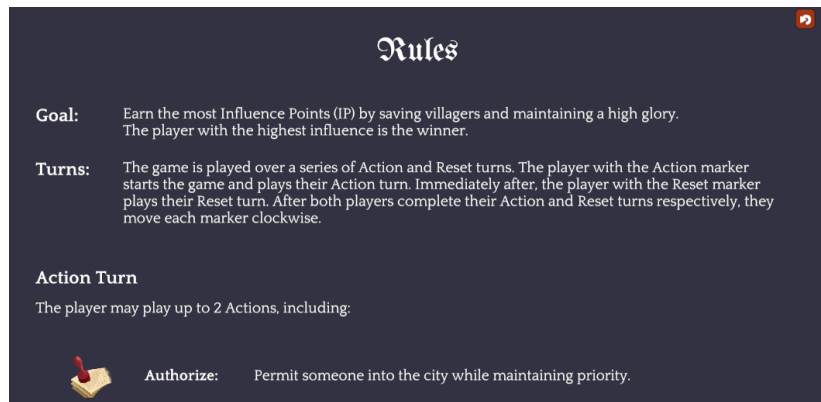


Figure 27: Rules Menu

6.2.6 Miscellaneous Changes

In addition to the above changes, several documentation pieces, visual updates, and balance tweaks were made to the game. These were intended to refine the game further for the final release, and also factored in feedback received during the playtesting phase of the project. Examples of these include the size of the worker icons, negative gameplay elements being removed from the game, and less cluttered iconography.

6.3 Group Experience

We began this project with a view to trying our hand at building an unconventional project, namely a digital board game. We believe we had a unique take on the Chaos and Order theme, and were excited to see how it could materialize in a digital form. Our initial design ideas largely remained consistent over the course of the project. The most significant changes revolved around the core gameplay rules, which were simplified in several areas to allow for a better gameplay experience. The most stark of these was the priority system in place of the tapping method for FIFO tracking. We believe that with these changes, our initial design ideas manifested into the final version even better than we had planned, and we are thus very satisfied with it.

Regarding our development schedule, we were able to follow sections of it quite well, but had to adapt when we encountered challenges. The most significant of these revolved around the networking aspects of the project, which was also our technical achievement goal. We probably underestimated the challenge of creating serializable actions to synchronize a turn based game, and also encountered challenges with network stability. These did require changes to our development schedule, however we were able to adapt the various work streams of our project well enough to dedicate sufficient time and resources to the networking branch. This eventually paid off at the end of the project.

Another change in our development schedule was regarding the decision to implement the priority system at the expense of the inner castle and the objectives. While we knew that it would lower the replay experience in the short term, we still believe that it provides a very strong core for our game to work with for the future.

The project structure put forth by the course team was at different times both a bane and a boon. The starting milestones provided us with structure and clarity regarding the project, and helped us stay on track. However, since we were developing a turn based game that was slightly different from typical projects, we needed to dedicate significant effort to planning and architecture upfront, rather than commence layered development immediately. In this case we found that the interim and alpha release project milestones were challenging for us, because the timelines did not suit the ideal development cycle of our project. We were forced to make a few concessions on our project architecture in order to meet these milestones, which will probably have to be refactored in the future. When viewed as a whole though, the project structure helped us stay on track and deliver a polished final release for the class.

6.4 Course Impressions

6.4.1 Expectations

Our expectations entering the course was that we would be able to build a fully playable game by the end of it. To that end, the course certainly met our expectations. We were initially apprehensive about the theme of the project and whether it would be very restrictive. However, we found that the theme was quite open-ended and permitted multiple interpretations, as can be seen from the myriad different projects that emerged in the class. We also think that the theme was a very apt choice to reflect the current global times, which in fact motivated us to structure our game around some concepts that would provide subtle social messaging. Overall, we felt that the theme helped bound our creativity in a positive way, and led to more fruitful ideation.

6.4.2 Our Achievements

We are very satisfied with the state of the game that we achieved, knowing full well the conscious design decisions and trade-offs that we made over the course of the project. We feel that we were able to successfully abstract a complex set of rules and interactions into a relatively simple user interface in the game. Furthermore, we were able to leverage software design principles when planning the architecture of our project, as reflected in the command and client-server patterns. This command stack also allowed for a very streamlined Undo system in the game, which we feel provides an even better user experience than a traditional board game experience, where players would have to remember their past moves instead.

We are also satisfied with the way we were able to internally manage our project schedule as a team, in order to adapt to challenges we encountered, such as with networking. We feel that this was a valuable experience that would serve us well in professional workplace environments as well. We are confident that we have a very solid version of the game that has been well structured in order to support future development and refinement after the course, and hope to take this project further.

6.4.3 Our Learnings

We had several valuable learnings during the project. Primarily, these revolve around gaining a good understanding of the work required for a project, and estimating the effort and time for it better. We probably underestimated the complexity of the networking, which led to some challenges in the interim milestones. This is something that we feel we will be able to better gauge after this course experience, and would be able to allocate time and resources accordingly. We are overall quite satisfied that we kept to our project schedule fairly well and met every course milestone, while also balancing our work streams in the face of challenges.

6.4.4 Course Suggestions

In terms of schedule, the course certainly felt rushed. This may be due to the fact that the final presentations take place a few weeks before the exam period of the semester, effectively lowering the development time for any students who would want to focus on project related classes in the semester. There is also the elephant in the room regarding the exceptional circumstances under which this course was undertaken. While the in-person interaction was certainly missed, we felt that there were opportunities to have more interaction between the groups taking part in the course. For example, perhaps each group should have playtested their games with every other group in the class. This of course could have been proactively done by the students, however perhaps there were a few opportunities to incentivize more interaction between the students in the class, even in an online domain. The original feedback given by each individual to every other group after the game pitch phase was quite nice, and we feel that it could have been done more often.

Another suggestion albeit difficult to implement could be to allow for groups to continue projects across semesters. For those students who need to take two Master Praktikums as part of their degree, this could be a good way to incentivize more detailed game development, such that some of these projects could even become professional releases. Currently, that would be difficult to achieve because students would need to meet the individual milestones in each iteration of the course. Since we had first hand experience of a need to have more time for planning and architecture of our game, this is a challenge that resonates with us.

Overall, we are glad to have a fully playable version of the game as a product of this course, and hope to work on it in the future as well.