



University
of Glasgow

IT+ Masters Team Project M

2021-22

report

Members

Yunkai Li 2698474

Tianyuan Wang 2521798

Xinhao YI 2627827

Xunxi Weng 2629898

Yujie Xiao 2546699

Catalog

1.First sprint.....	3
1.1 Story List.....	3
1.2 Project Progress Chart.....	3
1.3 Sprint Scrum.....	4
1.4 Sprint Review.....	4
2. Second sprint.....	5
2.1 Story List.....	5
2.2 Project Progress Chart.....	5
2.3 Sprint Scrum.....	6
2.4 Sprint Review.....	6
3. Third sprint.....	7
3.1 Story List.....	7
3.2 Project Progress Chart.....	7
3.3 Sprint Scrum.....	8
3.4 Sprint Review.....	8
4. Fourth sprint.....	9
4.1 Story List.....	9
4.2 Project Progress Chart.....	9
4.3 Sprint Scrum.....	10
4.4 Sprint Review.....	10
5.Fifth sprint.....	11
5.1 Story List.....	11
5.2 Project Progress Chart.....	11
5.3 Sprint Scrum.....	12
5.4 Sprint Review.....	12
6.Sprint Retrospective.....	13
6.1 Project Progress Chart.....	13
6.2 Review.....	15

1.First sprint

1.1 Story List

- Story name: Administrator Generates board Module
- Story name: Administrator Generates the player Module
- Story name: Administrator Distributes three hands Module
- Story name: Administrator Updates the state of the hands Module
- Story name: Administrator Sets mana value Module
- Story name: Administrator Compares the selected cards with the remaining mana values Module
- Story name: Administrator Shows the range of the unitcard available Module
- Story name: Administrator Shows spellcard usable range Module
- Story name: Administrator Unit Movement expression Module
- Story name: Administrator Unit Attack Range Module Module
- Story name: Administrator see if there is still actionable behaviour Module

1.2 Project Progress Chart

First sprint				
Story card	priority	Expected cost	Real cost	status
Administrator Generates board	1/10	1	1	Done
Administrator Generates the player	1/10	1	1	Done
Administrator Distributes three hands	2/10	1	1	Done
Administrator Updates the state of the hands	4/10	2	2	Done
Administrator Sets mana value	3/10	2	2	Done
Administrator Compares the selected cards with the remaining mana values	5/10	3	2	Done
Administrator Shows the range of the unitcard available	3/10	2	3	Done
Administrator Shows spellcard usable range	5/10	2	3	Done
Administrator Unit Movement expression	7/10	4	4	Done
Administrator Unit Attack Range Module	7/10	4	4	To do
Administrator see if there is still actionable behaviour	8/10	3	5	To do

1.3 Sprint Scrum

During the first week, we first discussed and assigned the overall development process, followed by the completion of the initial functions, such as generating the board, drawing cards, and the initial construction of all data sets.

During the development process, we encountered some problems, for example, the first is to shift the cards at right side one grid to the left after using a handcard, we need to create a map to connect our cards and positions. This is because after removing a card, all remaining cards need to be shifted one grid left and we need to know the position of each card; the second is some thoughts on cardclick, is it necessary to determine if there is enough mana first? We made a rule that if there is not enough to check the card, the tile range will not light up on the board.

We have also encountered a few bugs, such as finding problems with delete, drawing the deleted card again after canceling the highlighted session, and the bug where the unit is designed to be summoned normally but appears every time a hand is played, all of which we will improve in the next development week!

1.4 Sprint Review

During the first week of development, we completed most of the story cards except for the admin unit attack range module and the admin to see if there were any actionable actions left, and all team members had a clear definition and understanding of the story cards before implementation

The main issues at the beginning were that the software version did not fit and the initialization pages were incomplete, we spent a lot of time discussing these issues, the main problem was that we did not have a card location and we are going to fix this in the next sprint.

During the week the testers also completed testing of the story card and provided short, focused reports to help the development team write better functional code when possible.

2. Second sprint

2.1 Story List

- Story name: Player Chooses a deck Module
- Story name: Player Draws a card from the deck Module
- Story name: Player User selects a card to compare Module
- Story name: Player Selects Unit Module
- Story name: Player Summons unit Module
- Story name: Player uses spell Module
- Story name: Player Moves unit Module
- Story name: Player Checks Unit Attack Available Module
- Story name: Player Unit Attack Module
- Story name: Player End of Round Module
- Story name: Player uses Planar Scout card Module

2.2 Project Progress Chart

Second sprint				
Story card	priority	Expected cost	Real cost	status
Player Chooses a deck	1/10	1	1	Done
Player Draws a card from the deck	1/10	1	1	Done
Player User selects a card to compare	5/10	1	1	Done
Player Selects Unit	3/10	1	2	Done
Player Summons unit	3/10	2	2	Done
Player Uses spell	4/10	2	3	Done
Player Moves unit	6/10	2	3	Done
Player Checks Unit Attack Available	2/10	3	3	Done
Player Unit Attack	6/10	2	2	Done
Player End of Round	4/10	1	1	Done
Player Uses Planar Scout card	5/10	1	2	To do

2.3 Sprint Scrum

In the second week we worked on player design and development, including player card extraction, player movement, attacking and summoning.

Some of the issues we encountered were that we could not see the range of the cards after clicking on them, and when it was time to highlight them again on the second turn, the previous ones would be unhighlighted and highlighted. We needed to add the TileUtils class, including red and white highlighting and de-highlighting, and a method to maintain a record of the highlighted cards. Each unit creates its own class that inherits from the unit and uses polymorphism to call the unit's call method, the move method that determines when the tile highlight range is selected, and the attack method that determines when the tile highlight range is selected.

we also needed to design a static file with tables corresponding to card and unit, as the red and white distinction was considered, and considered wrapping a method to look up the colour of the tile, and created a handcardnit that maintained the relationship between all card and unit.

2.4 Sprint Review

Before the week started, our team first analysed and clarified the definition and function of the story cards for the week to guarantee a better follow-up. Then in the second week of development we completed most of the story cards, with the exception of the planar scout card used by the player

The main problems we encountered were that we couldn't see the range of the cards when we clicked on them and the highlighting didn't work properly. Over the next week each of us contributed our own ideas to fix both of these problems. A lot of time and effort was spent on this. In addition, we found that everyone writing on the main branch could easily lead to confusing code, so we set up different branches to accommodate collaborative work. This week we

During the week the testers also completed testing of the story card and provided short, focused reports to help the development team write better functional code where possible.

3. Third sprint

3.1 Story List

- Story name: Player uses Rock Pulveriser card Module
- Story name: Player uses Pyromancer card Module
- Story name: Player uses Bloodshard Golem card Module
- Story name: Player uses Blaze Hound card Module
- Story name: Player uses Windshrike card Module
- Story name: Player uses Hailstone Golem card Module
- Story name: Player uses Serpenti card Module
- Story name: Player uses Staff of Y’Kir card Module
- Story name: Player uses Entropic Decay card Module
- Story name: Player uses Comodo Charger card Module
- Story name: Player uses Hailstone Golem card Module

3.2 Project Progress Chart

Third sprint				
Story card	priority	Expected cost	Real cost	status
Player uses Rock Pulveriser card	5/10	2	2	Done
Player uses Pyromancer card	4/10	2	2	Done
Player uses Bloodshard Golem card	4/10	3	3	Done
Player uses Blaze Hound card	7/10	2	2	Done
Player uses Windshrike card	7/10	2	2	Done
Player uses Hailstone Golem card	8/10	2	2	Done
Player uses Serpenti card	8/10	1	1	Done
Player uses Staff of Y’Kir card	8/10	2	2	Done
Player uses Entropic Decay card	4/10	3	3	Done
Player uses Comodo Charger card	4/10	1	1	Done
Player uses Hailstone Golem card	5/10	2	1	Done

3.3 Sprint Scrum

In the third week we still worked on player development, first addressing the remaining issues from last week and the parts that were not developed.

In this week we found that the existing data relationships do not support the ability to find a given unit based on its title, and we need to think about whether we need to consider special characters when summoning (unlimited movement distance, long-range attacks) and how special units should be handled.

In this case we write some tool classes in CradUtils to identify which card is currently in play, and then introduce public List<Tile> currentWhiteHighLightTiles and public List<Tile> currentRedHighLightTiles in the game state. they are used to save the currently highlighted cards in the clicked state, filtering them once so that only the lighted ones are available and the unlighted ones are not allowed. We also add unitutils to determine special status monsters, and another method to simulate unit movement.

3.4 Sprint Review

Before the week started, our team first analysed and clarified the definition and function of the story cards for the week to guarantee a better follow-up. Then, in the third week of development, we completed all the scheduled story cards.

The main problem we encountered during the week was that we could not find the corresponding unit by its branded title. Except for the first time, we did not have a solution for the corresponding special units. We had to use Utils to deal with this problem. Everyone in our group approached the problem positively and came up with different solutions. Secondly because we split the responsibility for writing different modules, this led to a lot of conflicts when merging and thus cost us a lot of time to deal with it. In the next week we will focus on this issue

During the week the testers also completed testing of the story card and provided short, focused reports to help the development team write better functional code where possible

4. Fourth sprint

4.1 Story List

Story name: Player uses Pureblade Enforcer card Module

- Story name: Player uses Azure Herald card Module
- Story name: Player uses Silverguard Knight card Module
- Story name: Player uses Azurite Lion card Module
- Story name: Player uses Fire Spitter card Module
- Story name: Player uses Ironcliff Guardian card Module
- Story name: Player uses Truestrike card Module
- Story name: Player uses Sundrop Elixir card Module
- Story name: AI draws Module
- Story name: AI chooses the hand to play Module
- Story name: AI gives priority of unit cards Module

4.2 Project Progress Chart

Fourth sprint				
Story card	priority	Expected cost	Real cost	status
Player uses Pureblade Enforcer card	5/10	2	2	Done
Player uses Azure Herald card	5/10	2	2	Done
Player uses Silverguard Knight card	5/10	2	2	Done
Player uses Azurite Lion card	5/10	2	2	Done
Player uses Fire Spitter card	5/10	2	2	Done
Player uses Ironcliff Guardian card	5/10	3	3	Done
Player uses Truestrike card	5/10	2	2	Done
Player uses Sundrop Elixir card	5/10	1	1	Done
AI draws	2/10	2	1	Done
AI chooses the hand to play	1/10	2	1	Done
AI gives priority of unit cards	3/10	3	3	Done

4.3 Sprint Scrum

During this week we worked on player refinement and the development of some of the AI's functions, such as the AI's choice of hand distribution already determining the priority of the order of cards played.

During this week we found that the code was not working due to several missing library files. And after merging the code of team members, the code did not work properly. Thinking about how unit could get it to attack remotely and found that clicking on the highlighting would result in an error reporting a null pointer.

In which we get the CardUtils tool class from the utils package. We found that the merge problem was solved by privately transferring the code package and restarting the computer, possibly related to browser cookies, and that for the unit we could tell it was a special unit by its name

4.4 Sprint Review

Before the week started, our team first analysed and clarified the definition and function of the story cards for the week to guarantee a better follow-up. After that, in the fourth week of development, we completed all the scheduled story cards.

During this week we focused on the issue of merging conflicts in order to prevent the problem of spending a lot of time dealing with conflicts from the previous week. The next major issue we encountered during the week was that we were unable to find the corresponding units by their unit branding titles. Except for the first time, we did not have a solution for the corresponding special units. We had to use Utils to deal with this problem. Everyone in our group approached the problem positively and came up with different solutions. But again because of the different opinions of each person it was also difficult to reach the same opinion and thus we spent a lot of time dealing with each person's understanding and problems. In the coming week we will focus on this issue

During the week the testers also completed testing of the story card and provided short, focused reports to help the development team write better functional code where possible

5.Fifth sprint

5.1 Story List

- Story name: AI summons the unit Module
- Story name: AI gives priority of spell usage in playable hands Module
- Story name: AI uses spell Module
- Story name: AI determines the priority of unit's movement Module
- Story name: AI moves unit Module
- Story name: AI Unit Dynamic Path Planning Module
- Story name: AI determines the priority of unit summoning among playable cards Module
- Story name: AI determines the attack priority of the unit Module
- Story name: AI attacks Module
- Story name: AI ends turn Module

5.2 Project Progress Chart

Fifth sprint				
Story card	priority	Expected cost	Real cos	status
AI summons the unit	6/10	4	3	Done
AI gives priority of spell usage in playable hands	3/10	3	3	Done
AI uses spell	3/10	3	2	Done
AI determines the priority of unit's movement	4/10	4	4	Done
AI moves unit	5/10	3	3	Done
AI Unit Dynamic Path Planning	7/10	5	4	Done
AI determines the priority of unit summoning among playable cards	3/10	5	5	Done
AI determines the attack priority of the unit	3/10	3	3	Done
AI attacks	8/10	2	2	Done
AI ends turn	2/10	2	2	Done

5.3 Sprint Scrum

Last week we have completed most of the code development and the overall project has entered the final stage, mainly the development of AI character related features and the overall code refinement.

In this process we found some problems, for example, we found that we need to change the death check and found that the delay is too short, the death animation does not play out, so finally changed to 2 seconds delay.

We also found that the `cardUtils.getCardFromHandCardPosition` section would report an error when the monster died and played the extracted unit hand again. The error also occurred when pressing endturn, which took a lot of time to fix. We tried to set `gameState.interval` for debugging when the animation was called with an error. We also used `movepoint` and `attackpoint` in `tileclick` to determine if the selected unit had already been moved or attacked.

5.4 Sprint Review

Before the week started, our team first analysed and clarified the definition and function of the story cards for the week to guarantee a better follow-up. After that, during the last week of development, we completed the code, fixed the bugs and finished the project.

During the week, we focused on the issue of collaboration between the various members of the team, and we worked on aligning ideas to prevent us from having to spend a lot of time integrating them again. Secondly this week was the last week of the development cycle and we finished writing, testing and bug fixing all the code. We had a brief stalemate in the logic of the unit quota move and the attack, we spent a lot of time discussing it and then we wrote code analysis of the attack and move to make the flow clearer to resolve the issue. During the testing and bug hunting process, there was a lot of discussion between the team members and Jan. The coding team was very good and everyone contributed their ideas to make the project go in a positive direction, so we were able to complete the project successfully.

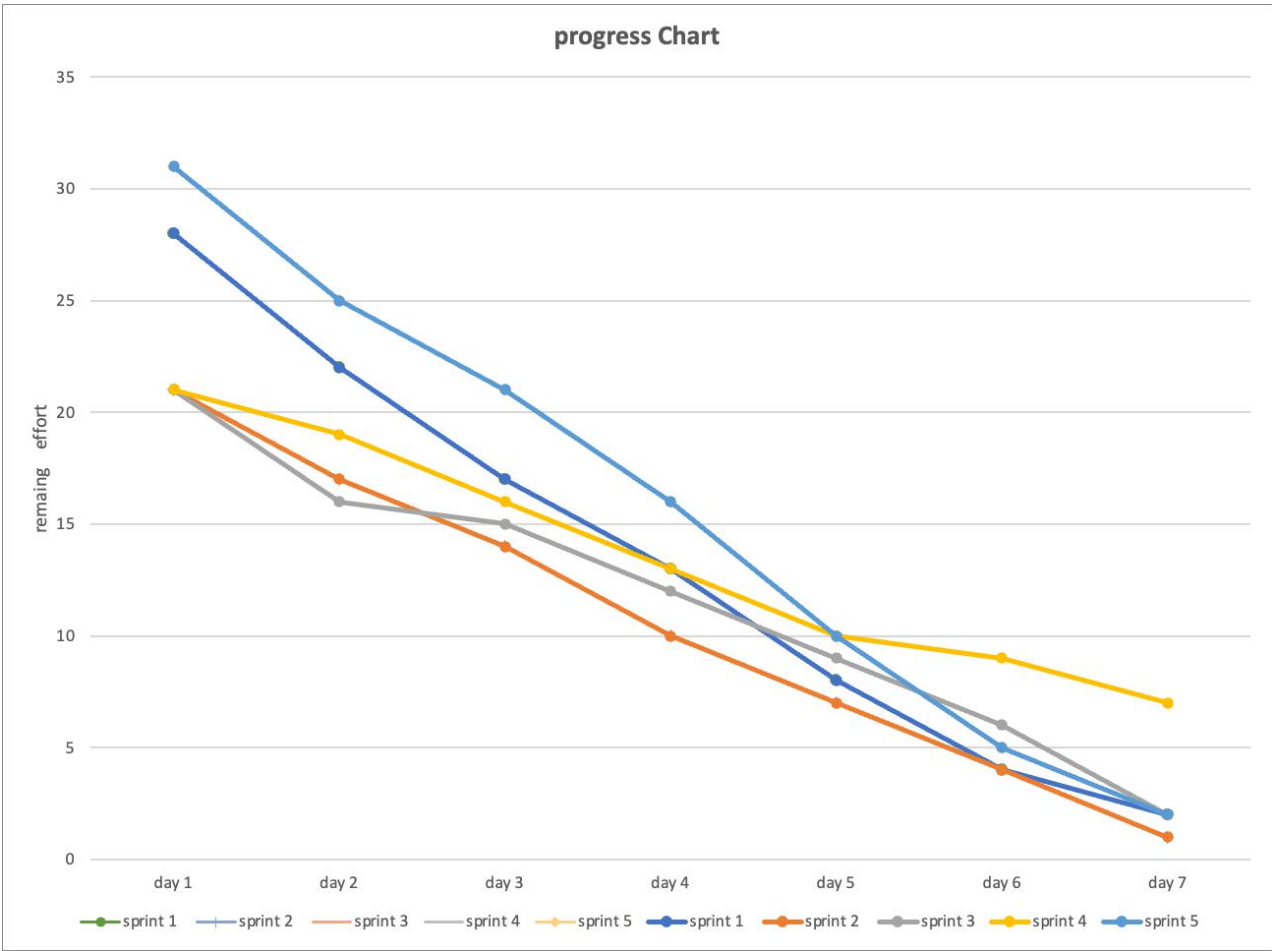
During the week the testers also completed testing of the story card and provided short, focused reports to help the development team write better functional code where possible

6.Sprint Retrospective

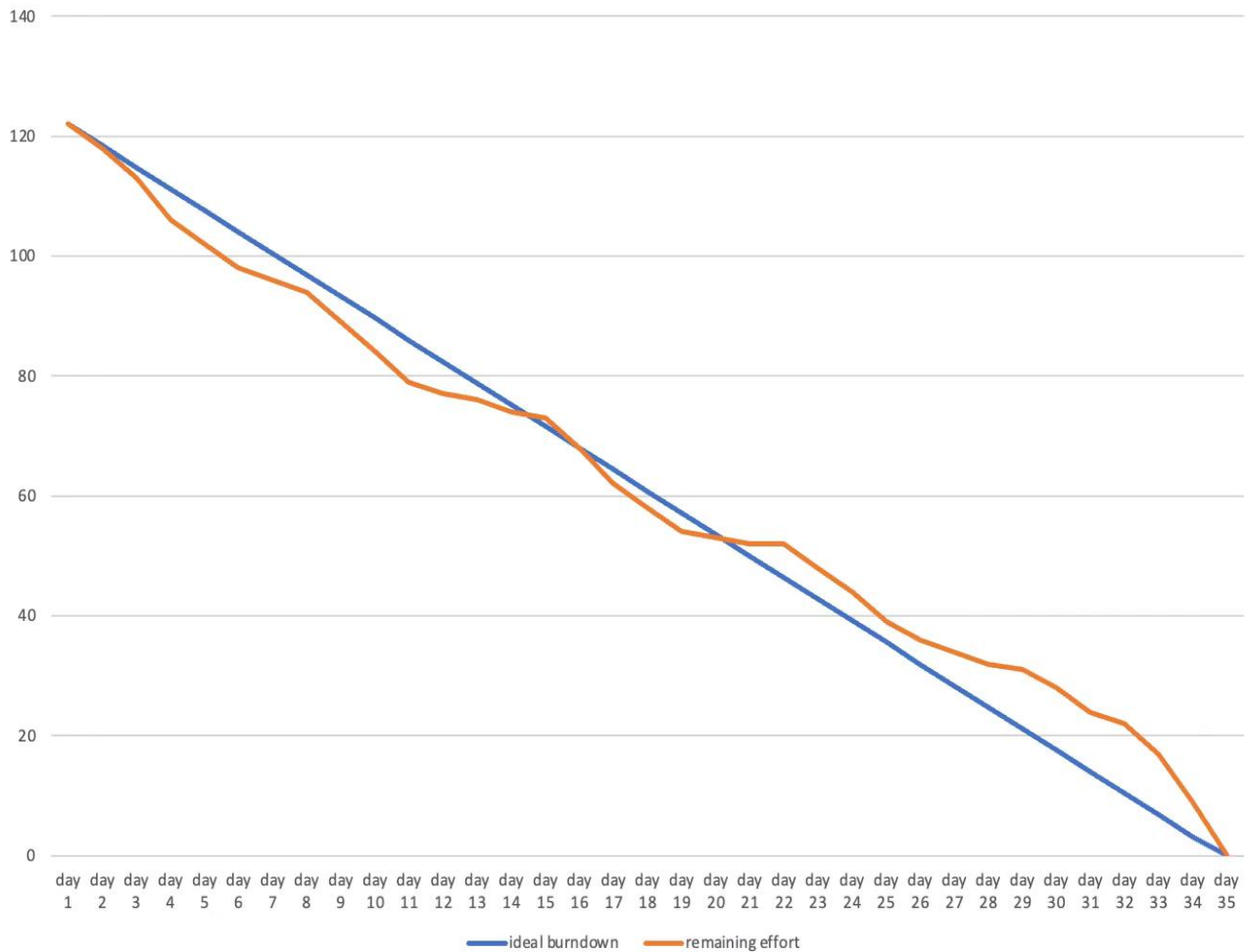
6.1 Project Progress Chart

Project Progress Chart

	Sprint 1	Sprint 2	Sprint 3	Sprint 4	Sprint 5	Average
Estimate	25	17	22	23	34	24.2
Actual	28	21	21	21	31	24.4
Diff from Estimate	3	4	-1	-2	-3	0.2
Diff as % of Estimate	112%	123.5%	95.5%	91.3%	91.2%	102.7%



Burndown Chart



6.2 Review

At the beginning, we adopted the method of remote collaboration. There was a situation where two people did not communicate and wrote the same method. After recognizing this situation, we tried our best to maintain a face-to-face working mode to solve this problem. Another problem is that it is found that there is a large difference in the level of code between different team members.

In order to solve this problem, we have stipulated the scope of work for everyone, and let a team member specialize in the work of testing and documentation. Finally, all story cards have passed the test, helping the main developers to develop code as soon as possible.

We used a total of five weeks to complete the task, and as you can see from the graph, the average working time per week was 24.5 hours. In the first two weeks, when we estimated the time needed of the sprint project, we always underestimated the time. In the next three weeks, as we become more familiar with the code, we will overestimate the time required due to increased efficiency. In general, the gap between our estimated working hours and actual working hours is about 10%. We believe that such errors are still within an acceptable range.

It is worth recalling that we originally wanted to use the tree data structure to solve the logic problems of AI. We list all possible ways to deal with AI. This is the biggest problem we have. We tried for three days and still couldn't solve the technical problem. Finally, we adopted the logic of priority and set the AI to the highest priority to attack the enemy avatar, which solved this problem.

One development experience is that we are not familiar with using GitLab as a tool to manage the development process at the beginning. There are always major changes to the main branch, which leads to confusion in the development process. Later, everyone uses their own branches to change different updates. It solved the problem.

Another development experience is that when estimating the sprint time needed, the amount of the last week was slightly underestimated, resulting in less code implemented in the third week than the last week. This led to a very tense situation in the last week, which should be avoided in the next project. Overall, we have completed a great project.