AN AUTOCHESS GAME DESIGNED TO IMPROVE THE CRITICAL THINKING SKILLS OF ITS PLAYERS

Yinjie Wu[1], Yu Sun[2]

[1]Virginia Episcopal School, 400 VES Rd, Lynchburg, Virginia 24503
[2]California State Polytechnic University, Pomona, CA, 91768,
Irvine, CA 92620

ABSTRACT

Providing critical thinking skills to people in an exciting and engaging manner is an issue that can hopefully be resolved with the introduction of a newly created autochess game [1][2]. Because this game only allows player interaction to occur before the battle starts in the preparation phase, the game encourages the player to think ahead and predict how the player's team can defeat the enemy team based on which cards are selected to be added. To test the effectiveness of the game at improving the critical thinking skills of its players, an experiment was conducted in which five example questions from the Watson-Glaser test were provided to the participants of the survey to answer, then the participants would play through all three levels of the game [3]. Then, the participants would try the same five questions from the Watson-Glaser test again [4]. The results of the survey indicate that the game only provided a slight improvement in critical thinking skills.

KEYWORDS

Autochess, Critical Thinking, Mythical Creatures, Unity

1. INTRODUCTION

The game is an autochess about mythical creatures [5]. Players would choose minions representing different creatures in mythology each round with combat between minions played out automatically. This game is made this way because it drew inspiration from a gamemode from one of my favorite games, Hearthstone. I chose the topic of mythological monsters because it is a topic very interesting but not mentioned a lot in games. As people may not be too familiar with mythological creatures, this game can help to educate them on what some popular mythological creatures can do and what they look like. Furthermore, not many people are familiar with autochess games and do not understand the appeal of these games or how to play these games. Autochess games are strategy games since they involve the player performing all their actions during the preparation phase and watching how well they did in the battle phase. This is very different from many other popular game genres, such as first-person shooters, as the player cannot interact with the game at all during the battle phase.

The most popular existing game of the genre is Hearthstone Battlegrounds mode [6]. It is based on the auto-battler genre with eight players competing in matches by recruiting minions over several rounds [7]. Auto-battler is a genre of games with automatic gameplay by the minions. Players would recruit their own minions and let them fight with enemies. Hearthstone

David C. Wyld et al. (Eds): ICAITA,CDKP, CBW, SOFT, CMC, Signal, MLT-2023
pp. 149-160, 2023. CS & IT - CSCP 2023 DOI:10.5121/csit.2023.130612

Battlegrounds combines this genre with its trading card game nature to form the new mode. Another existing game is Teamfight Tactics, which also introduces the same game mechanics as a typical autochess game does. A unique mechanic that Teamfight Tactics introduces is items that can be added to units to enhance their stats or can give them new abilities or effects. While there are not many glaringly large issues in each of these games, these games may have minor bugs that could hamper the gameplay experience, which is usually patched a few days after a new update is released. Another complaint that these games commonly have is how long it takes to understand and become good at the game. People who are proficient at the game often spend hundreds upon hundreds of hours on the game, and newer players become discouraged from continuing to play the game after realizing how wide the skill gap between themselves and more experienced players is.

The process of the gameplay is first, the player will be shown a panel with the enemy (computer) minions and their powers. Then the player would choose their minions. In battle, the player's minion on the left will attack first, followed by the enemy's minion on the left and the player’s second minion on the left, etc.

The main difference between Hearthstone Battlegrounds and my game is that the minions in Battlegrounds are brought each round and the quality of them varies depending on upgrades while all minions in my game are chosen pre-match and there is no obvious difference between the strength of minions. Also, there are eight players in a Hearthstone Battlegrounds game while my game is purely PVE. The same differences can be said about Teamfight Tactics as well, but Teamfight Tactics also has an item mechanic to further improve a unit, which my game does not. Overall, these two games are much more complex than my game.

An experiment was conducted to see whether the game has a significant effect on improving the player’s critical thinking skills. To test this, eleven participants were gathered to take a test for critical thinking skills. There were five questions on the test, and each one tested a different field of critical thinking. Then, the participants would play the three levels of the game. After completing the game, they would come back and complete the same test again. They were encouraged to read the questions carefully and see if a different answer would fit better as opposed to the one that was chosen before. Because the test and the questions were the same before and after, this would reduce any confounding variables. A table was used to record whether the answers that each participant chose improved from before and after playing the game, as well as which question each participant did well in. The goal of the experiment is to compare each participant’s results before and after and see whether there was a significant impact on critical thinking skills from playing the game. Furthermore, by listing down which questions each participant got correct, information can be gathered as to whether a specific critical thinking skill was impacted more significantly than others.

The rest of the paper will contain paragraphs about the challenges that were faced during the process of making the game. The details of the overall creation of the game as well as the solutions to some of the challenges are mentioned in the next section. In the section after, an experiment was designed and performed to test whether the game had a significant impact on critical thinking skills. Relevant games were mentioned and compared and contrasted to my game. Finally, concluding remarks and a self-reflection on how the game could be improved in the near future were mentioned.

2. CHALLENGES

In order to build the project, a few challenges have been identified as follows.

2.1. The UI

One challenge I faced while making the game was the UI. Designing and implementing a good UI is a difficult feat. The background cannot be too messy, dark, or distracting. The buttons and fonts all need to look appealing and visible. When the player sees a button, the player should be able to immediately recognize that it is a pressable button that has an effect on the game. Also, interfaces for displaying cards are difficult to design to make them easily readable for the players. The battle scene is the hardest part because it needs to have clear indications of what's going on and interesting effects.

2.2. The artistic parts of the game

Another challenge was the artistic parts of the game such as the background, icons, and images for cards. I didn't want to download existing drawings or designs for them, as this would take a long time and not all of the downloaded drawings would work well with one another. Therefore, I drew the icons and images, which would ensure that they would work well with each other and help to maintain a consistent artistic style in the game. It's very time-consuming work to do; each image for cards took at least two hours to draw. However, I believe that it was well worth the effort.

2.3. Finding resources

Finding resources about sound effects, visual effects, and fonts is also very challenging [8]. There are plenty of sound effects online but many are not suitable and the few that are suitable sometimes don't fit with each other very well. So, it took me a long time to find and download the sound effects that I wanted. Visual effects are less difficult to find but harder to implement with particles and other features. Fonts are hard because they need to match the artistic style of the game and many fonts have their flaws such as not having certain punctuation marks. Taking all of these asset choices into consideration and making sure they worked smoothly with one another was something that took much time and effort, but the game now has a very well-defined and consistent style as a result.

3. SOLUTION

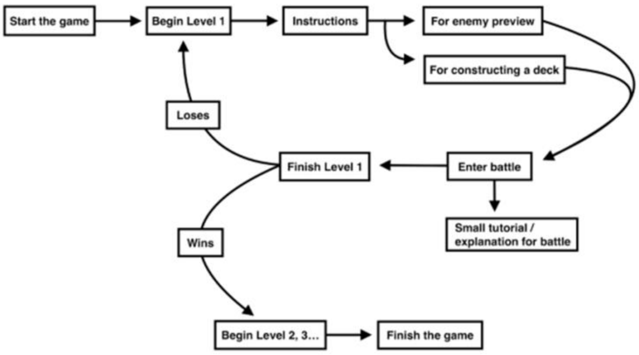


Figure 1. Overview of the solution

The game was created in Unity, and the functionality of the game was implemented using C# scripts [9]. When the game first starts up, the player is brought into the main menu screen, and when the player enters the game scene, they are introduced to a map featuring three levels. Regardless of which level the player chooses, they are brought into the preparation phase of the game. In this phase, the enemy cards are revealed to the player, and several possible card choices are provided for the player. The player will then choose five of the possible cards for their team, then continue into the battle phase [10]. In the battle phase, the player will not be able to interact with the game, and the player and enemy cards will battle each other automatically. Whichever team is left with cards remaining at the end of the battle wins. If the player wins, they get to move on to the next level. Otherwise, they will have to restart the level in hopes that they can win the next time.

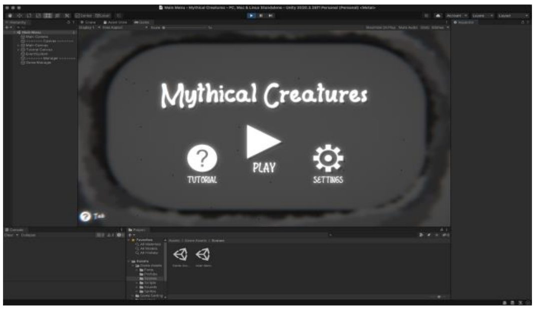


Figure 2. Main menu

We made the main menu like this, with a title, play button, tutorial, and settings button. A particle system is used to create some background effects of drifting ashes.

Figure 3. Text in main menu

The text in the menus is implemented using TextMeshPro objects, which makes it easy to customize high-quality text. When the tutorial button or tab key is pressed, the tutorial page will pop up.

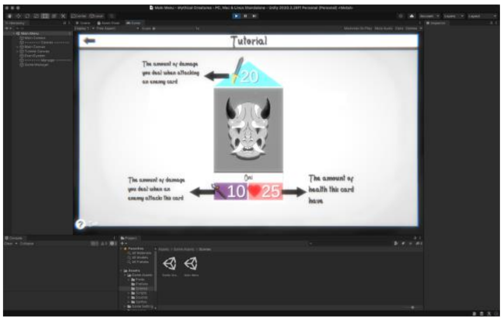

Tutorial
The amount of damage you deal when attacking an enemy card
20
Oni
The amount of damage you deal when an enemy attacks this card
10
25
The amount of health this card have

Figure 4. Tutorial page

The tutorial page explains the components of a card, including the health of the card, the amount of damage it deals when it attacks, and the amount of damage it deals when it defends. It can be opened at any time throughout the game in the bottom left corner. This is done by assigning a button to the question mark icon at the bottom left, and the button contains a script that toggles the tutorial panel.

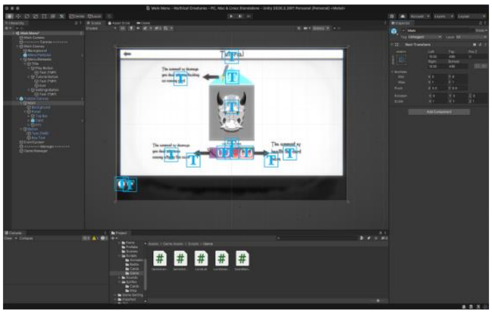


Figure 5. Text in tutorial page

The play button in the main menu opens the map, as the play button prompts the GameSceneManager script to switch to a different scene in the game. The same particle system is still used here for the ashes.

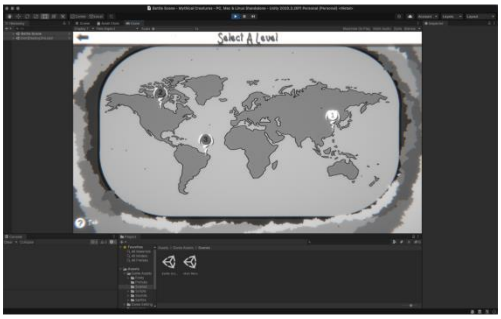
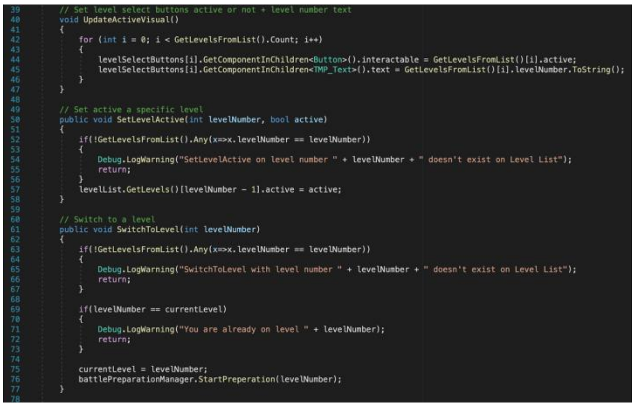


Figure 6. Map

Each button on the map is linked to a level. In the game, only the first level is available, and the next level will only be available after the first level is completed.



```
    // Set level select buttons active or not + level number text
    void UpdateActiveVisual()
    {
        for (int i = 0; i < GetLevelsFromList().Count; i++)
        {
            levelSelectButtons[i].GetComponentInChildren<Button>().interactable = GetLevelsFromList()[i].active;
            levelSelectButtons[i].GetComponentInChildren<TMP_Text>().text = GetLevelsFromList()[i].levelNumber.ToString();
        }
    }

    // Set active a specific level
    public void SetLevelActive(int levelNumber, bool active)
    {
        if(!GetLevelsFromList().Any(x=>x.levelNumber == levelNumber))
        {
            Debug.LogWarning("SetLevelActive on level number " + levelNumber + " doesn't exist on Level List");
            return;
        }
        levelList.GetLevels()[levelNumber - 1].active = active;
    }

    // Switch to a level
    public void SwitchToLevel(int levelNumber)
    {
        if(!GetLevelsFromList().Any(x=>x.levelNumber == levelNumber))
        {
            Debug.LogWarning("SwitchToLevel with level number " + levelNumber + " doesn't exist on Level List");
            return;
        }

        if(levelNumber == currentLevel)
        {
            Debug.LogWarning("You are already on level " + levelNumber);
            return;
        }

        currentLevel = levelNumber;
        battlePreparationManager.StartPreperation(levelNumber);
    }
```

Figure 7. Level code

In the code, methods called UpdateActiveVisual, SetLevelActive, and SwitchToLevel are used to control the activeness of levels and the switching to a specific level. For UpdateActiveVisual, each level from a list of specified levels is iterated through and is set to active.

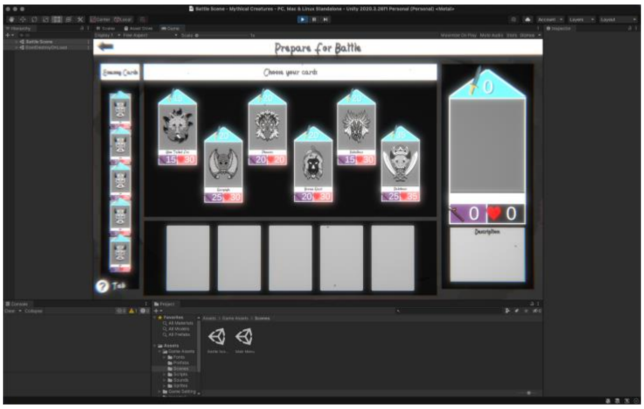
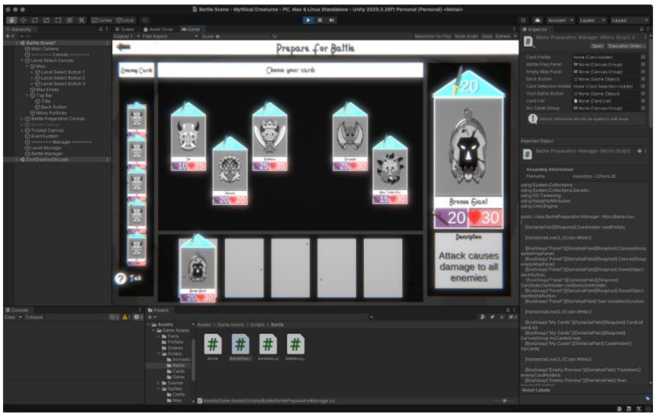


Figure 8. The preparation phase scene

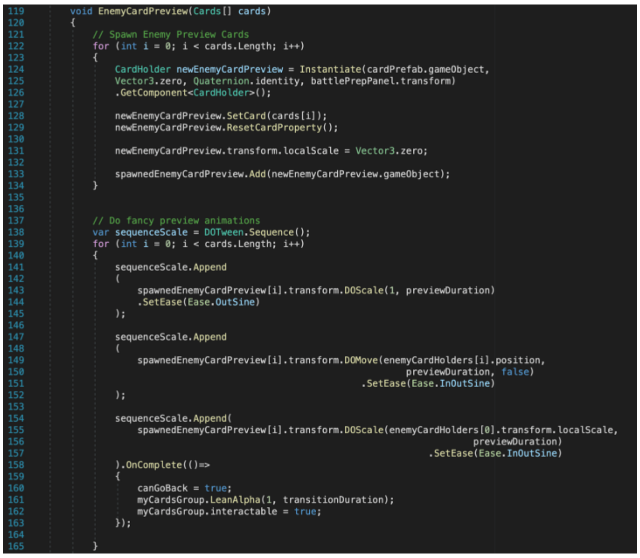
The preparation phase scene involves constructing a deck and viewing enemy cards before the battle. Because the player has information as to what cards the enemy will use, the player can more easily come up with a strategy to counter the enemy team. When any of the cards are clicked, they will be placed in the card slots at the bottom of the screen, indicating that they will be brought into the battle phase.



Attack causes damage to all enemies

Figure 9. Cards

The stats and skill information of a selected card would be listed at the right of the screen. This not only provides the user with a close look at the stats of a card but also gives a brief description of the card's unique skill or ability. The stats and the description are all sections of the main card game object, which are composed of an icon, an attack damage stat, a deflect damage stat, a health stat, and a skill description.



```
void EnemyCardPreview(Cards[] cards)
{
    // Spawn Enemy Preview Cards
    for (int i = 0; i < cards.Length; i++)
    {
        CardHolder newEnemyCardPreview = Instantiate(cardPrefab.gameObject,
        Vector3.zero, Quaternion.identity, battlePrepPanel.transform)
        .GetComponent<CardHolder>();

        newEnemyCardPreview.SetCard(cards[i]);
        newEnemyCardPreview.ResetCardProperty();

        newEnemyCardPreview.transform.localScale = Vector3.zero;

        spawnedEnemyCardPreview.Add(newEnemyCardPreview.gameObject);
    }


    // Do fancy preview animations
    var sequenceScale = DOTween.Sequence();
    for (int i = 0; i < cards.Length; i++)
    {
        sequenceScale.Append
        (
            spawnedEnemyCardPreview[i].transform.DOScale(1, previewDuration)
            .SetEase(Ease.OutSine)
        );

        sequenceScale.Append
        (
            spawnedEnemyCardPreview[i].transform.DOMove(enemyCardHolders[i].position,
                                                        previewDuration, false)
                                                .SetEase(Ease.InOutSine)
        );

        sequenceScale.Append(
            spawnedEnemyCardPreview[i].transform.DOScale(enemyCardHolders[0].transform.localScale,
                                                                    previewDuration)
                                                            .SetEase(Ease.InOutSine)
        ).OnComplete(()=>
        {
            canGoBack = true;
            myCardsGroup.LeanAlpha(1, transitionDuration);
            myCardsGroup.interactable = true;
        });

    }
```

Figure 10. Card code

A segment of code in the function EnemyCardPreview is used to create a copy of the enemy's deck to show players what to expect. The bottom part of the code, which is a for loop that loops through the cards, is used to animate the cards' path from spawning in the middle of the scene towards the left of the screen where they will stay. This animation can be seen at the beginning of the preparation phase.

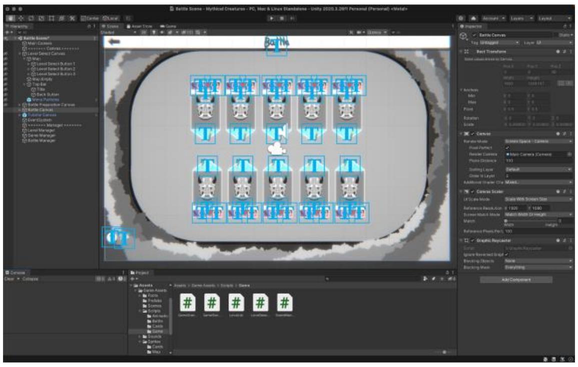
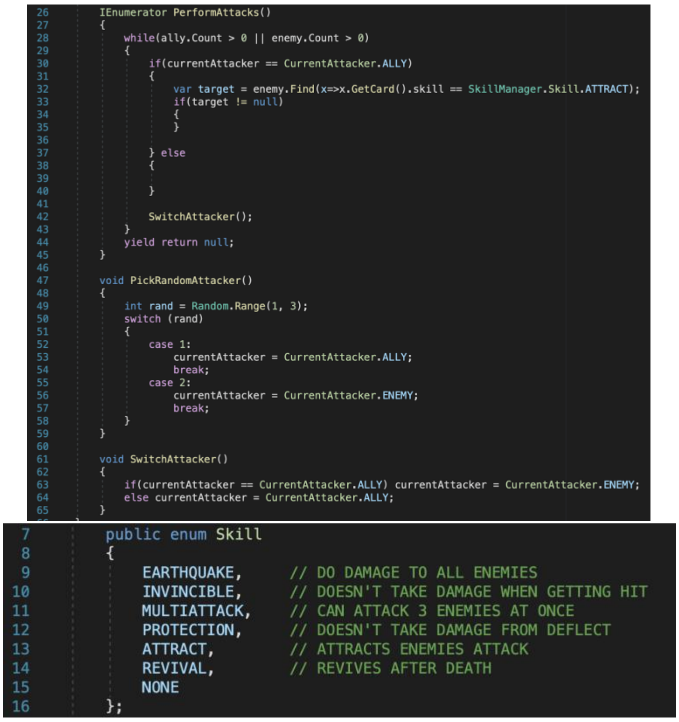


Figure 11. Text in cards

In the battle scene, the bottom cards are the player's cards and the cards at the top are the enemy's. The player always attacks first starting from the first one at left.



```
IEnumerator PerformAttacks()
{
    while(ally.Count > 0 || enemy.Count > 0)
    {
        if(currentAttacker == CurrentAttacker.ALLY)
        {
            var target = enemy.Find(x=>x.GetCard().skill == SkillManager.Skill.ATTRACT);
            if(target != null)
            {
            }

        } else
        {

        }

        SwitchAttacker();
    }
    yield return null;
}

void PickRandomAttacker()
{
    int rand = Random.Range(1, 3);
    switch (rand)
    {
        case 1:
            currentAttacker = CurrentAttacker.ALLY;
            break;
        case 2:
            currentAttacker = CurrentAttacker.ENEMY;
            break;
    }
}

void SwitchAttacker()
{
    if(currentAttacker == CurrentAttacker.ALLY) currentAttacker = CurrentAttacker.ENEMY;
    else currentAttacker = CurrentAttacker.ALLY;
}
```

```
public enum Skill
{
    EARTHQUAKE,     // DO DAMAGE TO ALL ENEMIES
    INVINCIBLE,     // DOESN'T TAKE DAMAGE WHEN GETTING HIT
    MULTIATTACK,    // CAN ATTACK 3 ENEMIES AT ONCE
    PROTECTION,     // DOESN'T TAKE DAMAGE FROM DEFLECT
    ATTRACT,        // ATTRACTS ENEMIES ATTACK
    REVIVAL,        // REVIVES AFTER DEATH
    NONE
};
```

Figure 12. Code of choosing cards

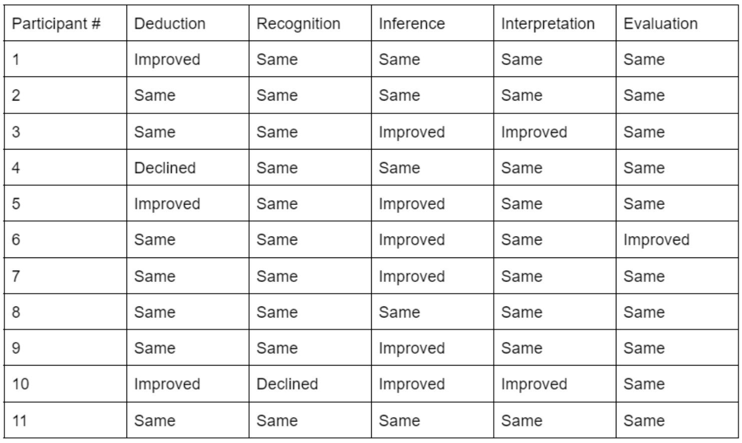
The code for choosing a card to attack and changing the card attacking and the code for skills of different cards is inside a battle script, which activates during the battle phase of the game and causes the cards to automatically attack the opposing team. The skills of each of the cards are stored in an enumeration.

4. EXPERIMENT

4.1. Experiment 1

To test if the game could improve critical thinking skills, eleven participants were asked to take a Google Survey form with five questions on it that tested for critical thinking skills. The first one tested for deductions, the second one tested for recognition of assumptions, the third one tested for inferences, the fourth one tested for interpretations, and the fifth one tested for evaluation of arguments. Since there are eleven participants in the experiment, the sample size should be large enough to account for any variability. Furthermore, the participants were able to share their thoughts about the experiment or game freely in a free-response section at the bottom of the Google Survey form. Once the participants were done, they would play all three levels of the game and try the same test again. They were encouraged to reread the question and explore the different answer choices.

Table 1. Result of experiment 1 (1)



Participant #	Deduction	Recognition	Inference	Interpretation	Evaluation
1	Improved	Same	Same	Same	Same
2	Same	Same	Same	Same	Same
3	Same	Same	Improved	Improved	Same
4	Declined	Same	Same	Same	Same
5	Improved	Same	Improved	Same	Same
6	Same	Same	Improved	Same	Improved
7	Same	Same	Improved	Same	Same
8	Same	Same	Same	Same	Same
9	Same	Same	Improved	Same	Same
10	Improved	Declined	Improved	Improved	Same
11	Same	Same	Same	Same	Same

The results of the experiment show that the majority of the critical thinking categories were improved after playing the game. The inference category experienced the most improvement among the participants. Interestingly, the only critical thinking category in which no improvements were made at all was the recognition category. Many of the results stayed the same before and after playing the game, although it was encouraged for the participants to reread the questions carefully and consider if a different answer would be better. This could indicate that some participants were insistent that their original answer was the correct one or felt that they would choose an incorrect answer if they chose differently. This idea can be reinforced by the participants’ feedback, in which some participants mentioned that they felt as if second-guessing themselves and choosing a different answer from their original answer would lead to a lower score.

The results of the experiment indicate that while the game does correlate with an improvement in the scores of critical thinking questions, the impact is not very significant. This is an expected result, as the improvement of critical thinking skills that can be achieved from playing three levels in a game will likely not be very large. Furthermore, the types of critical thinking skills vary greatly between the game and the questions. While the game tests for coming up with a strategy based on given information and coming up with an optimal team, some of the questions in the test require skills in recognition of assumptions and evaluation of arguments. However, with the question that tested for the ability to make inferences, there was a more noticeable improvement among the participants. Therefore, it appears that while the game can positively impact its players in terms of critical thinking skills, it appears to specialize in improving a particular type of critical thinking.

5. RELATED WORK

The game is under the genre of auto-battler. Auto-battler is a subgenre of strategy games that features chess-like elements in which the player places minions on a grid-like battlefield during the preparation round and then fights the opponent’s minions without the player's direct control. The genre was popularized by Dota Auto Chess in early 2019, and more studios have since released games in the genre, such as Teamfight Tactics by Riot Games, Dota Auto Chess by Valve, and Hearthstone Battlegrounds by Blizzard.

Teamfight Tactics by Riot Games is one of the most famous auto-battler games [11]. It is a spin-off of League of Legends and was drastically different compared to my game although they are under the same genre. Firstly, the Battlegrounds of Teamfight Tactics is like a chessboard. Minions would walk around and fight in the chessboard-like arena and it is influential to the gameplay because distance and formation are important for a battlefield that uses the x-axis and y-axis. Secondly, the minions of Teamfight Tactics are like characters instead of mere foot soldiers. They have complex stats, skills, traits, and equipment that could be utilized to influence the outcome of each battle. Equipment and traits systems are also big differences as there is plenty of equipment in Teamfight Tactics derived from League of Legends that are critical to the capacity of minions. A minion with good equipment is far more powerful than the ones without any. Traits are like positions labeled to each minion but can effectively buff them when multiple minions of the same traits are placed in the battle. The last main difference is the money system. There is a complete monetary system in Teamfight Tactics. Players would gain interest according to the money they have, the winning strikes, and winning a fight. The money would be used to buy minions and equipment (Casian & Zannato). In my game, there is nothing to be brought during the gameplay; players would build their deck out of existing cards pregame.

Dota Auto Chess is another favorite game of the genre, and it is the first auto-battler game to be created [12]. Dota Auto Chess was made in a mode in the game Dota 2 and became famous shortly after its release. This game also has a chess-like board, relatively more complex minions, traits, and a monetary system but no equipment. Players also need to buy minions each round, and this inspired other auto-battler games to be created in the future (Ponomarenko & Sirotkin). While the characters or units in Dota Auto Chess are able to be bought each round with currency, the units in our game do not require any currency, and the player is able to choose any card that appears in the pool of selectable characters in the preparation phase.

Hearthstone is the source of inspiration for the gameplay of my game [13]. The Battlegrounds mode of Hearthstone converted the chess-like auto-battler into the card game format of Hearthstone perfectly. Hearthstone Battlegrounds is an unusual auto-battler game in the form of a turn-based card game. The biggest difference between Hearthstone Battlegrounds (and my game) and other auto-battler games is card minions vs. character minions. Hearthstone is a trading card game, and all minions and the fundamental principles of gameplay are all based on cards. This was transformed into Battlegrounds mode and was drastically changed as a result. One difference is that the cards in Battlegrounds have their own or shared mechanics that can strengthen the whole deck. Some mechanics are shared and are endowed to the cards in a word, such as Taunt ("Enemies must attack minions that have Taunt.") and Windfury ("Can attack twice each turn."); some need a whole line written on the minion to explain, such as "Whenever a friendly Beast dies, gain +2/+1." Besides that, because there is always a side who attacks first in a card game of any kind, in Battlegrounds, a player is chosen randomly to attack at the beginning of each fight followed by the attack of their opponent. Each of their first cards at the left would attack a random target of the opposing team. Also, different from the minions' skills of other auto-battlers, there is a hero in Battlegrounds that has a skill. Each player will choose a hero before the game. One last big difference is the quest system. There are many quests in Battlegrounds that players can choose when they enter a given round. Each quest requires players to do some specific work such as "Play 5 Battlecry minions." and will provide powerful rewards when the quest is done (Zolboot et al.). While Battlegrounds is more complex and has a wider selection of cards, my game is much simpler and sticks to a simple artistic style.

6. CONCLUSIONS

In short, I have made an auto-battler card game with the theme of mythical creatures [14]. All of the UI elements and images of the game are drawn by myself. The general concept of an auto-battler is that all of the player interaction is done during a preparation phase, then the player units and the enemy units battle each other. During the battle phase, the user is not able to interact with the game. Whichever team still has units left alive at the end wins. The game was designed to improve the critical thinking skills of its players, as it requires thinking ahead and coming up with a strategy to defeat the enemy's team. An experiment was conducted to test whether the game significantly improved its players' critical thinking skills. Eleven participants gathered to take a critical thinking test, then played all three levels of the game [15]. They went back to the test to take it again, and the differences in scores before and after playing the game were compared. Although the other critical thinking skills did not seem to have much of a significant improvement, the inference skill had a fairly noticeable improvement. While the game does not seem to improve all types of critical thinking, it is effective at improving its players' ability to make inferences.

The current limitations are that the sound effects and music of the game are not up to standard. The sound effects are limited and imperfect and there is no background music. Also, there are just a few cards and levels in the game. Only a handful of cards exist and they don't have much of a combo. Ideally, having a synergy between a specific group of cards could allow for more strategy and provide more depth to the game. These limited amounts of cards with just three levels make the game not replayable, and players will likely get bored after just fifteen minutes of playing the game.

If I have time, I would implement better sound effects, some background music, and perhaps a story. More cards need to be added so I would also add more levels and unlock cards for each level. There should be some combos between cards featuring mythical creatures of similar elements or traits. When more cards are added, an inventory system could be added to the game as well.

REFERENCES

[1] Moore*, Tim. "The critical thinking debate: How general are general thinking skills?." Higher Education Research & Development 23.1 (2004): 3-18.

[2] Salam, Fajar. Implementasi Qlearning Pada Game Strategi Auto Chess. Diss. Universitas Komputer Indonesia, 2020.

[3] Behar-Horenstein, Linda S., and Lian Niu. "Teaching critical thinking skills in higher education: A review of the literature." Journal of College Teaching & Learning (TLC) 8.2 (2011).

[4] Bauwens, Eleanor E., and Gwenyth G. Gerhard. "The use of the Watson-Glaser Critical Thinking Appraisal to predict success in a baccalaureate nursing program." Journal of Nursing Education 26.7 (1987): 278-281.

[5] Harrington, Johnathon. "Games and Games: Dota 2, Dota Auto Chess, and Auto Chess." Chinese Digital Game Research 2019 Conference. 2019.

[6] Zolboot, Namuunbadralt, et al. "Hearthstone Battleground: An AI Assistant with Monte Carlo Tree Search." Proceedings of 37th International Confer 82 (2022): 131-140.

[7] Lee, Won-Hyeok, and Hyo-Nam Kim. "A Study on the Direction of Future Development of Auto Battler Genre." Proceedings of the Korean Society of Computer Information Conference. Korean Society of Computer Information, 2020.

[8] Prince, Stephen. Digital visual effects in cinema: The seduction of reality. Rutgers University Press, 2011.

[9] Nicoll, Benjamin, and Brendan Keogh. "The Unity game engine and the circuits of cultural software." The Unity game engine and the circuits of cultural software. Palgrave Pivot, Cham, 2019. 1-21.

[10] Baker, Alex, Emily Oh Navarro, and Andre Van Der Hoek. "An experimental card game for teaching software engineering processes." Journal of Systems and Software 75.1-2 (2005): 3-16.

[11] Zolboot, N., Johnson, Q., Shen, D., & Redei, A. (2022). Hearthstone Battleground: An AI Assistant with Monte Carlo Tree Search. Proceedings of 37th International Confer, 82, 131-140.

[12] Casian, A., & Zannato, M. (2020). Predicting Teamfight Tactics Results with Machine Learning Techniques.

[13] Ponomarenko, A. A., & Sirotkin, D. V. (2020). Dota Underlords game is NP-complete. arXiv preprint arXiv:2007.05020.

[14] Budijono, Nathaniel, et al. "LUDUS: An Optimization Framework to Balance Auto Battler Cards." (2022).

[15] Scriven, Michael, and Richard Paul. "Critical thinking." The 8th Annual International Conference on Critical Thinking and Education Reform, CA. Vol. 7. No. 9. 1987.