ABSTRACT

The purpose of this application is to be a novel and flexible educational game with the ability to enhance students’ willingness to learn and memorize material. It focuses significantly more on the ‘fun’ gaming aspect as compared to existing competitors with the aim of being more engaging for students, making them more willing to learn the appropriate material. It also has a focus on teaching vocabulary through association with existing knowledge. The impact, if successful, would be to overall raise academic performance and increase potential college acceptance rates.

1. INTRODUCTION

Every year, 2 million students take the SAT [11], with millions more undergoing a variety of other related standardized examinations, such as the ACT or GRE. A major aspect of preparing for these examinations is the necessary studying and memorization of vocabulary and definitions. Currently, some of the most commonly used methods of learning take the form of flashcards or quizzes. However, these processes of learning are inherently monotonous due to their repetitive and non-interactive nature, and studies [8] have shown that students respond better to a more interactive and fun methods of learning. Furthermore, the learning and acquisition of vocabulary remains a useful skill for people of all ages and groups and can always enrich somebody intellectually. Therefore, this created the motivation to design War of Words.

War of Words is an educational game created to improve the vocabulary of individuals with a fun and interactive experience. In its current state, its primary target demographic is high school teenagers preparing to take standardized examinations. However, it is designed to allow for the enhancement and reinforcement of vocabularies in general, and is therefore suitable for any person with this goal in mind. War of Words is a useful addition to the current stable of educational games due to existing competitors suffering from a variety of flaws causing them to be ineffective at allowing users to gain significant results in the long run. War of Words provides a novel learning mechanism which separates it from existing alternatives and possesses a variety of features which allow it to be an efficient method for learning.

A core concept utilized in War of Words is that of word association, which is the idea of linking words together by a common thread and learning them in relation to each other.

A study by Gu and Johnson [1] found that foreign language speakers who favoured this method exhibited larger vocabulary sizes. This concept was therefore adapted as a core learning mechanism within the game. The game uses the relationships between groups of words, particularly synonyms and antonyms, to instil the general meaning of a word into the player’s mind. A good reward system motivates players to keep playing the game, until they complete the final objective of acquiring all words. Enemies are the obstacles in completing this objective, and thus War of Words incorporates the fundamental features of a successful video game as described below in Section 3.1.

In the game, players control a character and encounter a variety of monsters with distinct abilities, and are rewarded for defeating them efficiently and with tact. Words are resources that are acquired by the player, and enemies are fought using a complex interaction model. This interaction model is a turn based battle simulation that is entered when a player encounters an enemy, in which opposing parties can also use words as ammunition. As players progress through the game, they acquire more advanced and complex words and face stronger enemies, and thus are motivated by the sense of progression. The game rewards players for knowing word definitions as well as related words, but does not make it a requirement to play the game.

2. RELATED WORK

There are several games out today that aid in preparing for the SAT, including several commercial [5] titles, and countless others that simply aim to provide some form of a learning experience. The research sample used accurately describes the wide variety of SAT games available today, and the following is a selection of these.

2.1 My Sat Coach

The first is My SAT Coach [7], published by leading SAT preparation company The Princeton Review. This game has several strengths and also several weaknesses. One major strength is that it presents data in a simple manner, with time-limited mini-quizzes meant to reinforce learning of definitions and a simple points-based reward system. Perhaps the biggest strength of this game is within the time-limits, as they are a motivation for players to increase their performance over time, something also rewarded in point rewards based on time taken. However, its simplicity is also a major weakness, as it becomes highly repetitive. Furthermore, the minimalist user interface results in a very bland experience, and the game experience therefore resembles that of a vocab-
ullary quiz rather than an interactive game, with very little rewarding feedback for either success or failure. Without any context, the points that are gained from being correct or incorrect is very meaningless. Within the framework of the definition of a game provided earlier, it becomes apparent that without enough consequences for being incorrect and the very basic points system this example lacks both an appropriate level of challenge and an adequate feedback system. Without the negative feedback for failure, the feeling of being rewarded for success is also diluted, and this is further exacerbated by the lack of a strong rewards system. Due to the factors failing to differentiate the game experience from traditional methods of learning, there seems to be very little incentive for students to play it.

2.2 FutureU

Another example, FutureU [6] from competing examination preparation company Kaplan, suffers from a similar set of drawbacks. Its strengths include an intuitive interface, as well as some quizzes and a variety of different ways to learn and memorize. However, it again has minimal feedback for success and failure, and plays out like an interactive practice test, where a lot of the material is just randomly generated questions which are solved following Kaplan’s study framework. Therefore, again, it fails to fit the framework of a game specified above, as without appropriate challenges and a strong reward system, it results in a diluted set of responses leading to unrewarding gameplay.

2.3 Tyrannosaurus Prep

While the previous examples are commercial retail games, weaker yet are the free online variants. The first and most popular result when searching for SAT Games online is a SAT Vocabulary Quiz Game site called Tyrannosaurus Prep [9]. This is a multiple choice game in which the player matches definitions to words. It is simple to play, has immediate feedback on correctness, and offers the definition of all word choices after a guess. However, there is very little sensory feedback or reward, other than a ‘level system’, where a bar slowly creeps towards the next level, without any context. Furthermore, with only one game mode and a limited database of words, it quickly becomes extremely repetitive. This is exacerbated by the lack of any visible goal to progress towards, and once again, this results in a product which is lacking in innovation or novelty compared to existing frameworks.

2.4 Memory Matches

An example of a game with educational value that is not solely focused on preparation for the SAT is Memory Matches [2], a puzzle game for iOS. Players start out by looking at pairs of cards arranged randomly, memorizing their locations, and picking out pairs once the cards are flipped. This memory game has an appealing color scheme, a scoring system and a multiplayer system where you can compete with players on-line to get all the pairs. Its straightforward interface makes an already simple game even simpler, but it also lacks variety. If there were different themes of pairs to pick from, the experience would improve.

2.5 SAT Vocab

SAT Vocab [4] for iOS is an example of an educational game today that is of higher quality than the aforementioned rest but still is somewhat lacking. It does better than its competitors in providing a more enjoyable learning experience with pictures corresponding to definitions, and mini games that explain definitions in an intuitive way. It also has a reward system which holds the player responsible for wrong guesses, but again falters in the lack of a progression element. Some players may not feel rewarded after correct answers as there is no clear way to utilize these points. Perhaps if players could use these points to buy power ups that make the game easier/change the game, the game experience would improve by a lot. This sort of reward system, while an improvement over prior titles, still provides limited motivation for the player to continue progressing with the game.

2.6 play2prep

play2prep [10] for iOS is another example of an educational game that fails to capture most of the essential components of one. It provides a bland interface for a timed quiz game without any reward or incentive or even score reporting. While there are a diverse range of topics tested by the game including mathematics and logical reasoning, the testing mechanism is simplistic and non-interactive. This game also follows the recurring theme among most of the games we have examined they are user friendly and easy to learn, but does not offer a very enjoyable or unique experience or any feeling of progression. Without an enjoyable experience, the educational aspect is all that is visible to the player, and therefore the game possesses insignificant advantages over existing modes of study.

Upon examinations, all of the existing games listed above have several shortcomings in common with respect to the general game framework. The first of those is that, while they all possess a final goal, this goal is exclusively simply learning. While it is true that the goal of any educational game should be learning, that is only the external goal of the game, or the reason why the game is useful in one aspect of game design. However, for all these games, from an internal gameplay perspective, there lacks the concept of a goal or in-game objective to progress towards. While this makes sense within the outer context of the games being created as examination preparation tools, it also causes the gameplay itself suffer and not be self-motivating, and results in necessary outside motivation to play the game. Secondly, their challenge is generally limited to the words themselves, and tends to remain constant throughout, meaning that they is no real progression in difficulty, which is an important element to make a variety of players with different levels of aptitude feel as though the game is a useful and worthwhile experience. Therefore, some players may quickly find the challenges quickly overcome and therefore lose interest in the game.

In the process of designing War of Words, extensive testing of the aforementioned titles has been performed, with thorough analysis of the strengths and shortcomings shared by some of these titles. The most significant shortcoming for many of these games is the lack of meaningful feedback or progression within the context of the game, with the player’s role limited to picking the right answer in a limited amount of time. The SAT preparation games mentioned above succeed in helping the player learn, but do not make it enjoyable. Several of the samples above barely satisfy the requirements for a game, and this is one of the primary main
motivations behind the creation of War of Words. War of Words is first and foremost a game, which happens to possess educational elements as an additional motivator. The word learning is therefore a necessary educational prerequisite for the player to progress, and is designed to be masked within the interaction model.

3. SYSTEM MODEL

To begin, it is important to provide an understanding of how an educational game fits within the larger framework of video games in general.

3.1 Video Games

To start, it is important to formally define a ‘Video Game’ in relation to this use case. According to McGonigal [3], any video game, inherently, must possess three major characteristics that ultimately make it a rewarding and desirable experience. The first is some sort of internal goal within the game, something that must be achieved, and with it a sense of progression. In this case, it is the combination of the acquisition of game resources over time through the completion of objectives, as well as the progressively more complex and difficult situations the player encounters over time.

The second is a set of rules, or the introduction of some number of unnecessary obstacles, which add challenge, engagement and difficulty to the game. This second characteristic is especially important as, without an appropriate level of challenge, a game quickly becomes uninteresting and abandoned. If the game is too easy, it is quickly completed and left behind. Too difficult, and gamers grow frustrated and eventually give up. A level of obstacles that seem surmountable but still significant reinforces the goals from the first characteristic and cause players to spend more and more time playing. Therefore, other than being appropriately challenging, these obstacles must also be interesting and ultimately linked to the goals from the first characteristic. Furthermore, different players have different levels of challenge appropriate for them, and therefore the game must be appropriately targeted, either at a specific demographic of players for which there is a certain appropriate challenge level, or have an challenge level that is adaptable to the abilities and desires of the player. This is achieved here through the a connection with the first tenet, which is the progression feedback. While a beginner player may remain at the easier parts of the game for a prolonged period of time, with simple enemies and words, a more experienced or skilled user will quickly progress past that initial part and reach a level appropriate for their expertise, as the game and situations become increasingly difficult and complex the more the player succeeds. This feedback for player success also means that, as a player improves, they will constantly face appropriate challenges, resulting in a continuously engaging level of gameplay.

The final key characteristic is that of a feedback system. In many games, this is some sort of points where the goal is to surmount the highest recorded score, or a sort of level system. Without feedback, gamers are left without much motivation - good feedback is therefore key. The strongest type of reward system is one that makes the player feel like they are earning something for progression through a game and defeating the challenges posed. As a player progresses through the game, overcoming the aforementioned appropriately challenging obstacles on their way to the final goal, it is important to see some tangible progression throughout the game in the form of the feedback system. All of these things lead into the very crux of a successful video game: it must be entertaining, and it must be something that players want to willingly spend time on because of its enjoyment value. The feedback system in this game takes on several forms. The first is that of progression, as already covered in the previous two points. The player is rewarded with a tangible sense of improvement and reward as they navigate the obstacles present throughout the game. The second is on a more micro scale, where the game utilizes significant visual feedback to reflect success or failure, to present a more lasting impression with regards to whether they are learning and appropriately using resources. The final point is tied in with the educational aspect and is somewhat external to the game, as if the player observes a noticeable improvement in the skills tested and improved by the educational aspects of the game, then it will provide additional feedback for the efficacy of the game as a learning tool and encourage repeated and prolonged play.

An important system feature is that the educational aspect should be integrated within the framework of the game as defined by the prior three points in order to maintain the fulfillment of the aforementioned criteria. This is to maintain the enjoyability of the game’s educational aspects for the target audience, keeping it challenging and rewarding, and because of that more appealing than the alternatives, namely rote memorization or studying by traditional methods. This game is therefore created with this in mind to establish a framework for learning without the perception of tedious memorization or overused and stagnant methods. With this in mind, the design and gameplay mechanics of the game itself can thus be established.

3.2 Basic Features

3.2.1 User Interface

The player itself is represented by a controllable 3D visible character avatar in a third-person view, and the world as a series of rooms that contain enemies and objectives. Players are free to roam around the world and interact with the different enemies. The player has certain properties, such as health, level and an inventory of the words acquired by the player thus far. Each enemy also has a health, level and an inventory of the words acquired by different enemies. The player has certain properties, such as health, level and collection of words, and this is based off the player progression thus far in order to create the scaling sense of difficulty and challenge explained previously. The player level starts out at 1 and increases at a decreasing rate as experience is gained from defeating enemies. The game also provides an internal dictionary which maps words to their definitions, allowing the player to look up and initially gain an understanding of newly acquired words, as well as refreshing this as necessary. The player’s dictionary is appended every time a new word is acquired and is always available to view. By making it always available, players are encouraged to look up words and their definitions whenever they feel the need. However, when in combat, there is a time-based element which disincentivizes the lookup of words in that situation, and therefore encourages the player to learn and memorize these words rather than being entirely reliant on this dictionary.

3.2.2 Words as a Resource
In War of Words, individual words are the main resource in the game with which the player defeats enemies. Each word is associated with a word class, and each class has opposing sections of ‘positive’ and ‘negative’ words. Each word in the same section of a word class are synonyms, and are antonyms of words in the opposing section. Furthermore, each individual word has a ‘power’ and ‘cooldown’ which are dependant on the complexity of the word. The power of the word determines the damage that individual word does, and therefore causes scaling such that the usage and therefore the understanding and knowledge of the definitions of complex words are important and necessary to complete more difficult content. The cooldown associated with words is a mechanic designed to encourage players to broaden their vocabulary within synonym classes, and will be elaborated on in depth below. Words are mostly acquired from defeating enemies and therefore gaining whatever is present in their word lists, and once picked up cannot be lost by the player, creating a sense of persistence and aiding in the progression.

3.3 Combat Overview

Each battle or encounter is a turn-based interaction between a player and enemy, in which words serve as skills or weapons, and the relationships between different words create strategic choices in which words are used by both parties. The basic framework of these encounters is that each word or attack used deals a certain amount of damage based on a variety of factors that will be covered in more depth below. The overall objective for each of these encounters is to reduce the enemy ‘Hit Points’ to 0, while preventing the enemy from doing the same to the player. Victory in battles is rewarded by the player acquiring a selection of words employed by the enemy, advancing the player closer to the final game objective. Furthermore, the player is granted experience based on the quality of the victory, working towards increasing their ability to overcome more powerful enemies.

3.4 Player Choices: Synonyms

Each turn the player can chose to use a weak ‘wordless’ attack, which tends to be fairly ineffective and exists solely as a last resort should the player be unable to use a word, given that the usage of words is meant as the main method in which a player gains an understanding of their definitions. Otherwise, a player may choose to select a word with which to conduct the round of combat. While this uses a word, it in itself does not actually result in the player having to learn anything and therefore is also designed to be somewhat ineffective, although more powerful than wordless combat. The most effective option that players can employ and the main vector in which the player acquires associations between words in their mind is the ‘combine’ mechanic. The ‘combine’ mechanic enables the player to combine two words together to perform an attack. If the two words are synonyms, then the player gains an attack with power significantly enhanced due to the employment of this tactic, as a method of reinforcement and encouragement for successfully identifying synonyms. Furthermore, visual feedback appropriate to the power in the form of a powerful animation and the enemy avatar being physically forced backwards is employed in order to emphasize to the player that their choice was successful and allow them to know to form an association between the two chosen words. However, if the two words are incorrectly chosen such that they are not synonyms of each other, the player instead performs an attack which is weakened considerably compared to using a single word. This form of negative feedback is also a method in which a player can ensure to not repeat a mistake, as well as allowing them to correct faulty associations employed. This, too, is accompanied by an appropriate accompanying animation showing off the ineffectiveness of the choice.

3.5 Player Choices: Antonyms

After performing an action, the enemy then performs its own action, similarly to the player. This part is where the antonym part of the concept comes into action. The enemy can select words from the words that it owns, and the player is then offered a choice to select a word in order to counter the opposing attack. If the player chooses not to do so, then the enemy attack will be at full power and will inflict significant damage, encouraging the player to participate in the counter system and therefore develop additional understanding of the relationships between words. If the player successfully selects an antonym to counter, then, the power of the incoming attack will be reduced based off the power and complexity of the chosen word used to counter. Furthermore, a source of visual feedback, represented by an animated rippling ‘energy shield’, allows the player to quickly identify the success of this choice and make appropriate mental associations as a result. Similarly to the previous system, if a word is chosen incorrectly and does not represent an antonym, the player will fail to shield against the incoming attack and the failure will even enhance its power, as visualized by an unstable reaction. The fact that the antonyms used by the player have to be in response to words chosen by the AI means that the player cannot grow reliant on a small stable of words which they commit to memory, as they have to constantly adapt to the changing situations offered by the game.

Figure 1: Visual shield feedback when an antonym is successfully used

The mixture of positive and negative feedback, especially the addition of strong visual cues, causes the player to develop an intuitive understanding of their general performance, as well as overall encourage the player to want to know and understand more words and their relations in order to successfully complete game objectives. The negative feedback in particular is a strong incentive to develop a more comprehensive knowledge of the available resources.

3.6 ‘Cooldown’ System
The Cooldown System is built off a simple idea - that every time a word is used, it is ‘put on cooldown’ for a certain amount of time based on the word, in which it cannot be used again. This system, although seemingly simple, has a variety of implications which significantly incentivize the player to broaden their vocabulary. Firstly, the fact that words cannot be consistently used over and over means that the player cannot grow reliant on a small subset of words in the game that they know the meanings to in order to defeat the enemies they encounter. The fact that not every word can be used all in every encounter means that the player has to learn more words in order to consistently defeat enemies. Furthermore, if a player understands that it is strategically advisable to keep a few words of every category in reserve in the cast that it is necessary to use them as an antonym and reduce incoming damage, the fact that words go on cooldown after being used means that the player not only has to learn a variety of words, but that they should learn words in every category such that they do not expend all the words they possess in a single category. Finally the presence of cooldown as the main demotivator for defeat means that, upon defeat, the player is put in a situation where they can not use their already known, familiar words for a longer period of time, thus causing them to have to learn the meanings of additional words in order to survive and continue. This added incentive and encouragement for poorer players to learn additional words provides an additional layer of difficulty scaling that allow for the game to match itself to players of varying skill levels.

3.7 Strategical Implications

The varying choices that a player has to make throughout the course of combat and overall gameplay serve several purposes. Firstly, the strategic factors cause a player to think more about the gameplay overall, with the result of being intellectually stimulating and putting the player in a position conducive to learning. Furthermore, the combination of having to react and adapt to changing game conditions and the cooldown system result in the player acquiring a wide array of knowledge in order to successfully navigate the obstacles faced. Each turn, the player has several choices which strategically decide the course of not just the battle, but future encounters. They must take into consideration the consequences of making wrong decisions and weigh the pros and cons of each choice, and in doing so repeatedly consider the meanings of words and the variety of words and their associations available.

4. SYSTEM IMPLEMENTATION

4.1 Languages and Platforms

The Unity Game Development Engine [12] was chosen as the base upon which to design to game, in order to use a variety of pre-made animations and simplify the design of some UI Elements, with the underlying scripting that controls the game behaviour done in C#. The current version of the game itself is designed to be run on PC.

4.2 User Interface

The User-facing Interface of War of Words was primarily designed in the Unity design view, using the new Canvas system to draw UI elements, such as the Health Bars, Dictionary and Information boxes at the forefront of the scene. The various elements are dynamically updated through the underlying scripts when battle occurs in order to accurately reflect the current game state in real time. There are also some legacy UI elements mixed in, specifically the combat options and word selection dynamically generated in the C# scripts, using the legacy Unity GUI [13] function calls. This was necessary to work with some of the resources used in the rest of the User Interface. The User Interface is designed to intuitively use mouse and keyboard-based inputs, with the mouse being the primary option to toggle and select the various buttons used to make input selections, and the keyboard used to control the character avatar.

4.3 Model

The character and enemy based models and animations were imported pre-made assets from the Unity Web Store. The player character is designed to be controllable outside of battle using both of the common arrow key and WSAD-based input styles, and becomes locked when the player enters combat. The camera chosen is a third-person view, with the camera fixed in a position slightly above and behind the player avatar. This offers a sense of immersion to the player and implies their association with the character avatar. In combat, the player and enemy models are forced to face each other and their positions and rotations are locked unless otherwise determined in order to give visual feedback and ensure that the player can clearly see the results of his actions in combat. One example of this is that the player is knocked back, that is their position is transformed in a direction opposite to the enemy, when struck by a powerful attack, with the distance based off the power. This is one of the mechanics for a player to gauge the effectiveness of choices and attacks without necessarily having to explicitly be told.

4.4 Animations

4.4.1 Character Animations

The animations of the character and enemy models are premade animations, imported from the Unity Web Store. Due to compatibility issues between models designed for different versions of Unity, the player model uses legacy animations bundled with the avatar itself rather than one of the more advanced humanoid animation packages available, while the enemy models uses a separate set of animations prefabricated for its specific frame.

4.4.2 Skill and Particle Animations

The assets used for spell and attack animations are quite varied and from a variety of sources. Most of the elemental spell effects are prefabricated particle system assets from various packages imported from the Unity Web Store, with modifications as necessary to fit the interaction model, and instantiated when necessary at the appropriate positions and
velocities based on the player and enemy positions and the nature of the attack. The ‘energy shield’ is designed on the base of a physical sphere, with collision such that it absorbs the projectile attacks used. The animation for the rippling, semi-transparent effect playing on the shield that emphasizes its nature as something generated from the power of the antonym is created through the use of a custom shader, which has the effect of creating a smoky, semi-vaporous effect on the surface of the shield as well as rendering the rest transparent. At runtime, when the shield is active, a script is used to change the effect over time, causing the aesthetic of the moving ripples on the surface of the shield.

### 4.5 Words

Word are implemented in a ‘BaseWord’ class. This class encapsulates the core information which we need our words to contain, which are the text of the word itself, the word definition and word type, as well as whether the word is a member of the positive or negative sections of that particular word type. This class also contains the attributes of the individual word, being its power and base cooldown, which are dependent on the word complexity. The base cooldown is defined to be the default length on which a word goes on cooldown after being used.

Individual words are defined and stored in a separate file, wordDatabase.xml, which allows us to add words to the game as necessary without hardcoding it into the C# scripts. This can also theoretically allow players to customize the list of words used in the game if they intend on adapting it to their own use and needs in terms of specific vocabulary terms. On game launch, that XML document is parsed into BaseWords and stored in a master dictionary that contains all the words and is used as a reference.

#### 4.5.1 Word Ownership

The player and enemy characters each possess a List of BaseWords as part of their character information class. As the player character defeats an enemy character, he gains BaseWords as a reward and those are added to his list of words. The player can only give the option to select words within his own word list, that are not currently on cooldown, and cooldowns are tracked within this list.

### 4.6 Character Information Page

Each character is encapsulated in the CharacterInfo class, which stores the relevant Character Information connected with the current controlled character. This class contains all the relevant information for the character, including the Character Name, Maximum Health, Current Health, Word List, Level and Current Experience. It also contains a BaseDamage statistic, which is dependent on factors such as the current level and determines the amount of damage dealt by attacks. It serves as a base value for multipliers due to word usage and combinations to be applied to. This class also contains methods to modify these values, in particular one detecting if the character has gained enough experience to level up, and if so modifying the rest of the character statistics in order to reflect the increase in power and durability as a result of levelling up. Enemies have a similar character page encapsulating their stats, but with less customization, as they are unable to level up or gain words. Instead, they are given a level and base statistics based on the player’s details upon generation, as well as a list of words.

### 4.6.1 Saving and Loading

When outside of battle, the player has the option to save and load the game. Saving is achieved via the Unity ‘PlayerPrefs’ system, where the data is written to the local disk. For most of the character data, including current position, facing direction and statistics, this is quite trivial, as PlayerPrefs supports storage of Floats, Integers and Strings. However, the word list has to be serialized into string form to be able to be stored, and later deserialized upon load. Since it is just a list of words, this can be converting all the data in the individual BaseWord classes into String form, choosing an appropriate delimiter, and storing them as a single String under the key “words”.

### 4.7 Game State

The core Game State data is contained in the GameState class, which tracks the current game state. The most important part of this class is that it tracks the current phase of the game, whether the player is in the main overworld, where the character avatar can be controlled, or in combat. If the player is in combat, then the Game State also keeps track of the current words in use and the current phase of combat.

### 4.8 Combat States

While in combat, the combat UI is enabled and character positions and rotations are locked. The element of the combat UI that is present during all of the phases is the bar showing the current enemy health. There are 8 Combat States used in this implementation. Following is what they all are and what they enable:

#### 4.8.1 PLAYERTURN

‘PLAYERTURN’ is the State representation that it is currently the player’s turn. During this state, the player is given a selection of options, including ‘Use Word’ and ‘Normal Attack’. The normal attack is, as mentioned previously, very weak in order to encourage players to use words and generally given as a last resort option. If this option is selected, then that weak normal attack is performed, the animation will play, and then the Combat State will go to ‘ENEMYTURN’. Otherwise, if the player selects ‘Use Word’, the Combat State will go to ‘CHOOSEWORD’.

#### 4.8.2 CHOOSEWORD

During the ‘CHOOSEWORD’ State, a Word Selection GUI Element, which is seen as a scrollable box, is generated, with buttons pertaining to all of the current valid words. A word will only show up as valid to be selectable if the word is currently owned by the player, and it is currently not on cooldown. The player can now select either to ‘Cancel’ this action and therefore return to the previous phase (PLAYERTURN), or otherwise select a word from this list. If a word is selected, then the word is saved and then the Combat State will progress to ‘COMBINE’.

#### 4.8.3 COMBINE

During this state, the Word Selection menu is again generated. This time, the currently selected word is also unavailable to be picked, as a word should not be able to be combined with itself. During this phase, a UI element is also enabled reminding the player of the initial word selected. The player is given the option to combine the existing selec-
tion with a new word via the synonym system, to go back, or to not combine. If the player chooses 'Go Back', then the Combat State will return to CHOOSEWORD and the currently chosen word will be forgotten. If the player chooses 'Don’t Combine', then the player will attack with only the originally chosen word. That word will go on cooldown, damage will be dealt appropriate to the power of that individual word. Otherwise, if the player chooses a word, then the script computes whether or not the two words are synonyms and sets the power accordingly. In either of the two attack cases, all the used words will go on cooldown, and the game will then detect whether or not the enemy player was defeated. If the enemy player was defeated, then the state will progress to WIN. Otherwise, the state will progress to ENEMYTURN.

4.8.4 ENEMYTURN

During this phase, the AI selects its course of action, and whether or not it will select a word and which word to use. If the AI chooses to use a weak generic attack, then the game will check if the player is dead. If the player is dead, then the game state will advance to LOSE. Otherwise, it will go back to the PLAYERTURN. Otherwise, if the AI chooses to use a word, then the selected word will be stored and the game state will advance to COUNTER.

4.8.5 COUNTER

In this state, the player can choose whether or not to counter the enemy word and if so, with which word. Similarly to the COMBINE phase, the Word Selection menu is generated and a UI element is enabled showing the word selected by the enemy. The player can then choose 'Don’t Count’ or a word. If the player chooses a word, then the game will compute the effectiveness of the counter attempt based off whether or not it is an antonym and the complexity of the word, as covered previously. If the player chooses to not counter, then the full damage based on the enemy word chosen will be dealt. In either case, the game then checks for player defeat, and progress to LOSE if so. Otherwise, it will progress to PLAYETRTURN again.

4.8.6 WAIT

This state is not really a true combat state, but rather an temporary state that the State Machine resides in while animations are playing. The WAIT state disables all the state-specific UI elements so that the player cannot select any action while the animations are playing, and then transitions to whatever the next state should be after the animation is done playing. The reason the WAIT state is utilized is because Unity cannot synchronously wait, and therefore if the player were to select any interactive UI element during animations being played with the wait implementation, it could lead to unexpected side-effects such as progressing to the wrong state, attacking multiple times, or storing incorrect data.

4.8.7 WIN

The WIN state is progressed to after the game detects that the enemy has had all of its health depleted. The WIN state then rewards the player appropriately for the victory with experience and words, and then sets the overarching state back to WORLD, exiting combat.

4.8.8 LOSE

The LOSE state, like the WIN state, handles post-battle results. In this case, the player is penalized for his defeat with the doubling of the cooldowns of all words. Furthermore, this state must restore the player health back, so that the player does not persistently have zero or negative health after the battle.

5. RESULTS

Initial feedback and results gathered from 16 beta testers, mostly High School students, College students, or recent College graduates has been positive so far.

5.1 Testing methodology

2 separate corpuses of words, Corpus A and Corpus B were designed (See Appendix 1). Each corpus consists of 24 words, 12 that are present in War of Words (selected at random) and 12 that are not. Furthermore, the 16 testers were separated into 4 groups (1A, 1B, 2A, 2B). The 2 groups 1A and 1B were the control groups, and they were given specially designed flashcards corresponding to the words existing in the current War of Words corpus. Groups 2A and 2B were given a beta version of the game executable. The 2 groups were then tested. Group 1A and 2A were tested with Corpus A, and 1B and 2B were tested with Corpus B, on how many words they could either define or give synonyms for. Both groups were then instructed to spend an hour, with 1A and 1B studying the flashcards and 2A and 2B playing the game. After this hour, Groups 1A and 2A were tested with Corpus B and 1B and 2B were tested with Corpus A. The reason why the groups were split like this is so to account for possible differences in difficulty in the 2 corpuses. The reason why 2 corpuses were necessary was because otherwise students might deliberately look up words they did not know in the first corpus they encountered, which would skew the results in favor of both learning methods.

5.2 Results

<table>
<thead>
<tr>
<th></th>
<th>Before</th>
<th>Flash</th>
<th>Game</th>
</tr>
</thead>
<tbody>
<tr>
<td>Corpus A (WoW)</td>
<td>7.125</td>
<td>8.0</td>
<td>9.0</td>
</tr>
<tr>
<td>Corpus A (Control)</td>
<td>7.125</td>
<td>6.75</td>
<td>6.25</td>
</tr>
<tr>
<td>Corpus B (WoW)</td>
<td>6.25</td>
<td>8.0</td>
<td>8.5</td>
</tr>
<tr>
<td>Corpus B (Control)</td>
<td>7.25</td>
<td>7.00</td>
<td>6.75</td>
</tr>
<tr>
<td>Overall (WoW)</td>
<td>6.0875</td>
<td>8.0</td>
<td>8.75</td>
</tr>
<tr>
<td>Overall (Control)</td>
<td>7.1875</td>
<td>6.875</td>
<td>6.5</td>
</tr>
</tbody>
</table>

Table 1: Scores (out of 12) on vocabulary quiz. For raw data, see Appendix 2

Note that Flash refers to the group using Flashcards, and Game refers to the Group playing War of Words. (WoW/Control) refer to whether those words were present in War of Words and the Flashcards.

While this is a small sample size (16 people) and no firm conclusions can be drawn from the data, we can still observe that both Flashcards and playing War of Words seems to have a positive effect on Vocabulary, and that the effect of the game is approximately on par with, if not slightly better than that of Flashcards. The next step towards gathering more solid data is to test this upon a larger set of users, preferably mostly of the main target demographic of high school students preparing to take their standardized exam.
Furthermore, the test corpus was specially designed for this use case, and it may be useful to examine changes, for example, in standardized practice examinations for the SAT.

6. FUTURE WORK
While the initial returns on War of Words seem to be quite successful, there are a variety of ways to expand the game to make it more successful.

6.1 Expansion of Relational System
One piece of feedback shared amongst several beta testers is that the idea of ‘Word Types’ for Synonyms and Antonyms is too binary, and indeed, some words can have multiple meanings or may not have the same antonyms as their synonyms in practice. Therefore, one logical next step for the game is to change the word encoding such that every word has a specific list of synonyms and antonyms unique for the word. Equally, while words currently only have one listed definition as they pertain to their current word class, it would also make logical sense to open up this system such that each word has a list of definitions that more closely matches how they are used in English. Another suggestion was also to have a system where every word could be placed in a context as relating to an example of its use in actual written English, and to match up words which could all fit within a context. A major technical difficulty that implementers of this improvement should be wary of is that this additional complexity must be correctly encoded such that it works on a general level for all words, and it may be necessary to hardcode in much of the behaviour as relating to specific word synonym/antonym and contexts.

6.2 Application of System to Different Topics
Another potential future expansion would be to take the existing system, but instead of using it with vocabulary, apply it to various other contexts where such binary groupings could be made. One example of this would be to teach basic Mathematical principles, with examples such as $e^x$ and $10^{2x}$ be ‘synonyms’ (as they are both exponential functions) and $\ln(x)$ be an ‘antonym’ (as logarithms are the ‘opposing’ function to exponentials). This type of model could be used in a variety of educational contexts, and may become a useful tool for students unwilling or unable to learn these topics in more traditional manners.

7. ETHICS
There are several potential ethical dilemmas present were War of Words to see widespread use.

7.1 Correctness
One dilemma lies within the correctness of the educational aspect of this game. Since it has the aim of improving the vocabularies of the players and helping them learn, care must be taken to make sure that the educational content of the game remains valid and correct. If words are incorrectly defined or put in place such that they have the wrong synonyms or antonyms, players may learn this incorrect information. This could then lead to the players in question having difficulty or losing points during standardized examinations or in general being misled about certain words, and this could lead to emotional or educational distress.

7.2 Effectiveness
If War of Words is shown or marketed as an effective educational game, students may wish to play or use this game as an alternative to traditional learning methods. Therefore, if the game is in reality less effective than these methods, either as a whole or in that particular case, then the player could incur a large opportunity cost if they are devoting significant time to playing this game rather than pursuing other potentially more effective methods, and again this could potentially lead to comparative harm in their studies or vocabulary skills overall.

8. CONCLUSIONS
Overall, in a small sample size, War of Words has displayed effectiveness in aiding in the learning and memorization of vocabulary. It succeeds in its stated goal of improving upon perceived shortcomings in its competitors and providing a novel and ‘fun’ alternative to traditional methods of learning vocabulary. It also succeeds in its goal of teaching vocabulary through a system that is separate from rote memorization in the word-association, synonym and antonym based combat system, and it does so within the context of the game.

9. REFERENCES
10. APPENDIX 1: TESTING CORPUS

In both of these Corpuses, the first 12 words are words that exist within the given version of War of Words, and the latter 12 are SAT vocabulary words which are not yet in the game.

Corpus A:
1. Obliterate
2. Peace
3. Cajole
4. Animosity
5. Flummox
6. Calm
7. Strike
8. Ire
9. Douse
10. Rectitude
11. Illuminate
12. Shield
13. Antediluvian
14. Anachronistic
15. Impeach
16. Consensus
17. Obtuse
18. Lavish
19. Pulchritude
20. Cheerful
21. Novel
22. Alacrity
23. Amalgamation
24. Wisdom

Corpus B:
1. Efface
2. Hit
3. Parry
4. Conflagration
5. Tranquility
6. Baffle
7. Elucidate
8. Inveigle
9. Overcome
10. Animosity
11. Discombobulate
12. Serenity
13. Ebullient
14. Didactic
15. Tutorial
16. Adversity
17. Frugal
18. Procrastinate
19. Spurious
20. Gilded
21. Exalted
22. Wary
23. Aesthetic
24. Discredit

11. APPENDIX 2: TESTING RESULTS

Format of results: Before: (War Of Words corpus score, control corpus score), After: (War of Words Corpus Score, Control Corpus Score)

Group 1A:
1. Before: (6,7), After: (9,6)
2. Before: (8,6), After: (9,6)
3. Before: (11,12), After: (12,11)
4. Before: (3,4), After: (2,5)

Group 2A:
1. Before: (10,10), After: (12,9)
2. Before: (8,6), After: (9,4)
3. Before: (5,4), After: (7,4)
4. Before: (6,8), After: (6,10)

Group 1B:
1. Before: (8,6), After: (9,8)
2. Before: (9,12), After: (12,10)
3. Before: (2,4), After: (4,3)
4. Before: (5,7), After: (7,6)

Group 2B:
1. Before: (7,9), After: (7,5)
2. Before: (5,4), After: (7,2)
3. Before: (10,12), After: (12,12)
4. Before: (4,4), After: (10,6)