Creating a Digital Melee Combat System

Based off of Traditional Martial Arts

Author
Eric Lambert
15117626

Supervisor
Dr. Chris Exton

LM110 Computer Games Development
Final Year Project Report

UNIVERSITY *of* LIMERICK
OLLSCOIL LUIMNIGH

Computer Science & Information Systems
University of Limerick
County Limerick, Ireland

1 Abstract

Martial arts in video games are incredibly common, ranging from the core combat to an animation that plays in a strategy game or a story game's cinematic. Focusing on the former, combat being one of the pillars in game design and development, there are still a wide range of interpretations. Some are incredibly fun, futuristic, fantasy-based (not obeying the laws of physics), or more, while others start becoming more towards the realistic side of the spectrum.

I have over 10 years of martial arts experience, focusing on traditional combat swordsmanship from different cultures and time periods, combined with my current education in programming and switching careers into being a software developer. This puts me in a rather unique opportunity to interpret my first hand experience into the game mechanics for a game that relies heavily on martial combat (non-ranged). The goal here is not to replicate any specific martial art into a video game, as I do not have the legal rights to, nor the permissions of any governing bodies.

The intention is instead to design a combat system that would have a common interaction system. What I mean by that, is to build a combat system from the ground up, with the idea that any traditional or new martial art, that focuses on combat rather than competition/sport, could then be replicated in part, or eventually in full, in my combat system.

2 Declaration

This submission is to fulfill the report requirement for the final year project to graduate with a Bachelor of Science Degree in Computer Games Development.
All brands, product names, trademarks, or intellectual properties belong to their respective owners.
All of the work being submitted for this final year project has been done solely by the author, Eric Lambert.

3 Acknowledgments

I would like to thank my family, firstly. My partner, Leslie, for her love and support for these last 10 years, as well as putting up with spending 4 years over 5,000 miles apart. My friends for their encouragement, and my family for their support and guidance throughout my life.

I would also like to thank Dr. Chris Exton for being my supervisor, as well as everything he has done to teach and help the students. During this entire process, Dr. Exton has been nothing short of supportive, encouraging, and helpful.

Special thanks to Kaiso Obata Toshishiro for not only founding Shinkendo, but taking me on as a student well over 10 years ago. I would extend this thanks to the entire Obata family, fellow instructors, and fellow students.

Contents

4 Introduction

Figure 1: Screenshot of the project.

This final year project is an exploration of designing a digital melee combat system. Since there are a myriad of ways to designing melee combat in games, this is in no way an expression of anything negative towards what has been explored so far. Instead, I want to see if there was a way to make a melee combat system that could recreate the mechanics, strategy, and adrenaline of an actual sword fight, in a digital format. More important, the artifact has to be accessible to the general public that games, that will have every version of no martial art experience to someone with over 50 years of professional expertise in multiple styles.

It was very important to start off with no assumptions, decisions, or preconceptions of what the end system may end up looking like. The approach was to look at least two different sword martial arts, only what is available in the books available to the public either through a museum or purchase. Because these are the basics, if there are similarities in the systems, it will be at this base level. From there, I isolated the similarities between both systems, into the philosophy, mechanics, footwork, and swordwork. Once the commonalities had been identified, I then looked at all of the data, and worked to design a combat system that could be plugged into any game.

Once the design portion was done, I started prototyping the system in Unreal Engine 4. After creating the blueprints, in game assets, editing prefabs, importing packages, and building the level, I then started working on the actual gameplay. I knew I wanted to have a transition from the normal gameplay, into a transition state, into the actual combat, at that time from the design stage. That took the first major stage of programming, bug testing, and refining until I was happy with the transitions between the various game states. The subsequent milestone was creating a turn base interface as well as the turn base mechanics.

After that was achieved, I went on to tackle creating custom animations for this proof of concept artifact. This was because of the game design, I needed precise animations that begin and end at certain points. Not only would this make the game logic visually apparent to the user, but was needed for the collision detection on the player model and it's attached sword. I downloaded some free use animations from Mixamo, and then edited the .fbx files. This started with making poses in Blender for speeding up the development of the custom animations I needed to progress. This took a fair bit of time, as I have had no previous animation experience, and only knew the concepts of animations with some work with OpenGL.

4.1 Motivation

Throughout time, history has been recorded from word of mouth, to the written word, audio recordings, and then video recordings. While the ability to record our history into digital format, my mind goes to the martial arts. There is so much knowledge that has been invested in martial arts, with some of the earliest artifacts being dated to 1000 BCE[1]. While sword use died down, the was still used in combat all the way through World War 2, specifically in the Japanese officer school[11]. So roughly over 3000 years worth of use, which means that martial arts and weapon have history in most cultures.

Every generation of martial artist that passes lessens the collective knowledge that, for better or worse, makes up a tremendous period of human existence. I would love to see games to be a way of preserving knowledge, and even recreating history in games to try to give a user, as close as possible, a firsthand recreated experience whenever possible. I would love to create a combat system that is not only fun for users/players, but also could easily be used as a repository of knowledge.

4.2 Research Question

How feasible is a digital melee combat system, based off of two traditional sword martial arts?

4.3 Goals

The goals of this final year project is a series of three rather ambitious parts towards a complete project.

- Choose two existing traditional combat martial arts to research.
- Find core concepts in each of the chosen martial art.
- Define common core that will be used to design the program around.
- Design the combat system, including extensibility, and it's interaction within a complete game.
- Develop the designed system in Unreal Engine 4.
- Bug test and refine the system.
- Find, or create through Blender if needed, custom animations for controlling the skeletal mesh for the sword and characters.
- Refine collision detection, adding in a mesh slicer if possible.
- Link up the logic of the system with the animations.
- Finalise the system and check feasibility.

5 Research

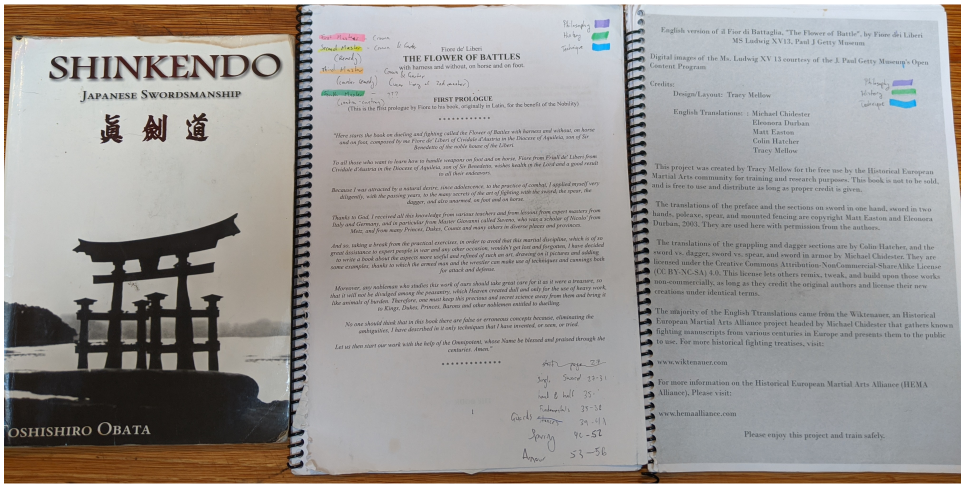


Figure 2: The three main books used for martial art research. Two are different versions/translations of Fiore's manuscript.

5.1 Research Approach

For this project, I wanted to make sure that the system would be able to accommodate multiple melee martial arts from different time period and cultures. It was very important to choose at least two different martial arts that have no known/major interaction. Next, it would be easier for the initial design be mirrored (same weapon and moveset for each in game character) at least until the system was fully developed and refined. For this reason, I chose to martial arts I have experience with, and both are swords that should be held in two hands, but can be wielded effectively with one hand given the circumstances of combat.

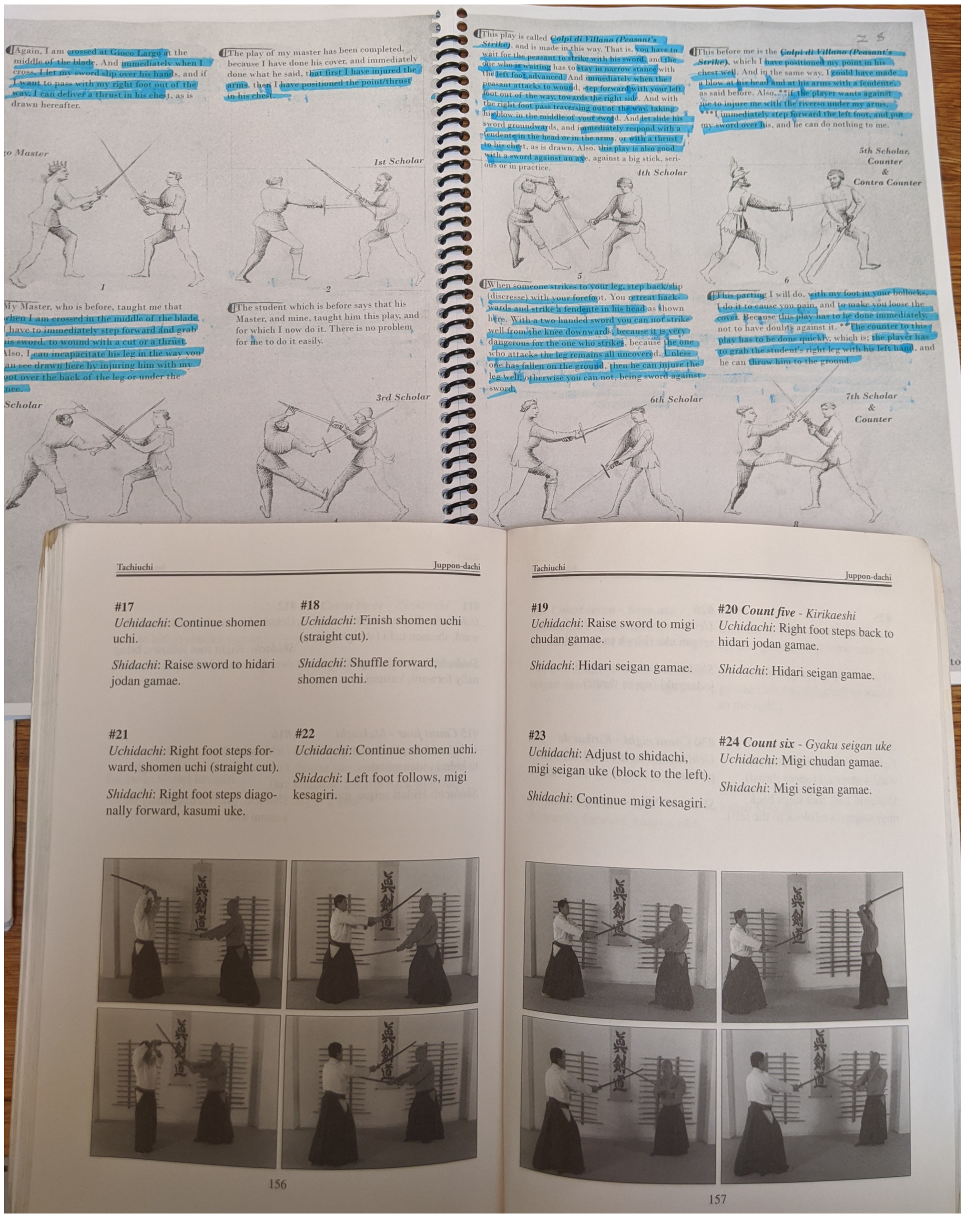
It was of the utmost importance, that any martial art that I included in the research of this project, not be affected by being used in competition. This is done for a very specific reason, as is not meant as any disrespect or anything negative. The specific reason for the harsh distinction and line that I would not cross (in terms of having more martial arts to draw on for this project), is rules. Competitions have rules, and they have to, especially when safety and weapons are being used. It would be irresponsible to toss two people some weapons... hopefully foam weapons, as wood, rebated steel, or sharp steel would all end with someone going to the hospital.

The rules cause a shift in focus from the practitioner, as logistical factors come to the forefront. I can say this with personal experience, fencing in college with foil, épée, and sabre back in 2004 - 2007. Because right-of-way is such an important part of fencing[4], it greatly affected how I played the sport. For instance, a mutual touch, an attack that connected with the opponents on target, would depend on the weapon, the judges, and the free. From there, a decision is reached as to if any point(s) is awarded, to whom, and then a reset to start again. There is nothing wrong with this approach, but it's clearly a competition. Compared to a martial art who's focus is use in combat where points, rules, referees don't exist. Instead it's a matter of, did your block and/or footwork keep you safe? If you did, you get to live, if not, you died. Did your attack launch properly, did you use proper muscle groups and extension to actually cut your opponent from the start of contact to when the blade finished through their body. This may sound gruesome, however it makes sense that the blade needs to be free after cutting or killing an opponent, as they may have friends or allies that are coming to return the favour to you. The mentality and approach of combat martial arts versus competitive is why I decided to model my system off of the former.

5.1.1 Research Parameters

- Two-handed sword.
- No armour.
- Combat focused.
- Historically correct.
- Established systems.
- Footwork.
- Swordwork.
- Non-combat concepts, training, and philosophies.
- Mechanics of a fight.

5.2 Traditional Martial Arts


Again, I am crossed at Gioco Largo at the middle of the blade. And immediately when I cross, I let my sword slip over his hands, and if I want to pass with my right foot out of the way, I can deliver a thrust in his chest, as is drawn hereafter.

The play of my master has been completed, because I have done his cover, and immediately done what he said, that first I have injured the arms, then I have positioned the point/thrust in his chest.

o Master

1st Scholar

1

2

y Master, who is before, taught me that vhen I am crossed in the middle of the blade, have to immediately step forward and grab is sword, to wound with a cut or a thrust. Also, I can incapacitate his leg in the way you an see drawn here by injuring him with my oot over the back of the leg or under the nee.

The student which is before says that his Master, and mine, taught him this play, and for which I now do it. There is no problem for me to do it easily.

Scholar

3rd Scholar

This play is called *Colpi di Villano (Peasant's Strike)*, and is made in this way. That is, you have to wait for the peasant to strike with his sword, and the one who is waiting has to stay in narrow stance with the left foot advanced. And immediately when the peasant attacks to wound, step forward with your left foot out of the way, towards the right side. And with the right foot pass traversing out of the way, taking his blow in the middle of your sword. And let slide his sword groundwards, and immediately respond with a fendente in the head or in the arms, or with a thrust to his chest, as is drawn. Also, this play is also good with a sword against an axe, against a big stick, serious or in practice.

This before me is the *Colpi di Villano (Peasant's Strike)*, which I have positioned my point in his chest well. And in the same way, I could have made a blow at his head and at his arms with a fendente, as said before. Also, **if the player wants against me to injure me with the riverso under my arms, ***I immediately step forward the left foot, and put my sword over his, and he can do nothing to me.

Z 8

4th Scholar

5th Scholar, Counter & Contra Counter

5

6

When someone strikes to your leg, step back/slip (discresse) with your forefoot. You retreat backwards and strike a fendente in his head as shown here. With a two handed sword you can not strike well from the knee downwards, because it is very dangerous for the one who strikes, because the one who attacks the leg remains all uncovered. Unless one has fallen on the ground, then he can injure the leg well, otherwise you can not, being sword against sword.

This parting I will do, with my foot in your bollocks. I do it to cause you pain, and to make you loose the cover. Because this play has to be done immediately, not to have doubts against it. **The counter to this play has to be done quickly, which is; the player has to grab the student's right leg with his left hand, and he can throw him to the ground.

6th Scholar

7th Scholar & Counter

8

Tachiuchi Juppon-dachi

#17
Uchidachi: Continue shomen uchi.

Shidachi: Raise sword to hidari jodan gamae.

#18
Uchidachi: Finish shomen uchi (straight cut).

Shidachi: Shuffle forward, shomen uchi.

#21
Uchidachi: Right foot steps forward, shomen uchi (straight cut).

Shidachi: Right foot steps diagonally forward, kasumi uke.

#22
Uchidachi: Continue shomen uchi.

Shidachi: Left foot follows, migi kesagiri.

156

Tachiuchi Juppon-dachi

#19
Uchidachi: Raise sword to migi chudan gamae.

Shidachi: Hidari seigan gamae.

#20 *Count five* - *Kirikaeshi*
Uchidachi: Right foot steps back to hidari jodan gamae.

Shidachi: Hidari seigan gamae.

#23
Uchidachi: Adjust to shidachi, migi seigan uke (block to the left).

Shidachi: Continue migi kesagiri.

#24 *Count six* - *Gyaku seigan uke*
Uchidachi: Migi chudan gamae.

Shidachi: Migi seigan gamae.

157

Figure 3: Examples of paired sparring training in both martial arts.

Word of mouth, practice, practical use are the main way of recording traditional manuscripts. Fiore wrote the the Flower of Battle for the Marquis of Ferrara, Modena, Reggio, and Parma[2]. aAs expected, there is a language barrier from old Italian to modern English, which the text that focuses on the translation and filling in the gaps from a direct translation[6]. Multiple translations are often helpful, as words and meaning can change depending on translations[3]. This includes Fiore's description of the four masters of each style, showing the different levels of mastery, of counters to the moves of the previous masters[2]. As such, we are relying on translations and hand drawn pictures from the 1400's to learn a martial art system from one of the best swordsmen of the time. Ideas that are common to traditional sword training, and recorded accounts of combat[8] which include the personal accounts of fencing duels with live (sharp) swords, and a fatal duel, from Aldo Nadi.

Shinkendo, while being more modern, still has similar issues, which is part of the reason I selected both martial arts as inspiration. In the book[10] the ideas, philosophy, and meaning had to be translated from Japanese, but still keeps the names, counting, and commands in Japanese. Instead of the hand-drawn pictures the Flower of Battle[2], Shinkendo[10] has multiple pictures, every one of the founder, Toshishiro Obata, and possibly an assistant or opponent depending on the circumstance. This allows for clear direction of body mechanics, snapshots in time, that should mirror the founder's exact position in the book. As a general rule in martial arts, and expanding into life, you carefully observe a skilled person doing something that you want to learn how to do. From there if you try to recreate the body mechanics in the same manner, you will end up with something similar. There is the ability to improve from there, depending on how you, the person wanting to learn in this case, best learn and refine new things.

5.2.1 Shinkendo, Japanese Swordsmanship, by Toshishro Obata

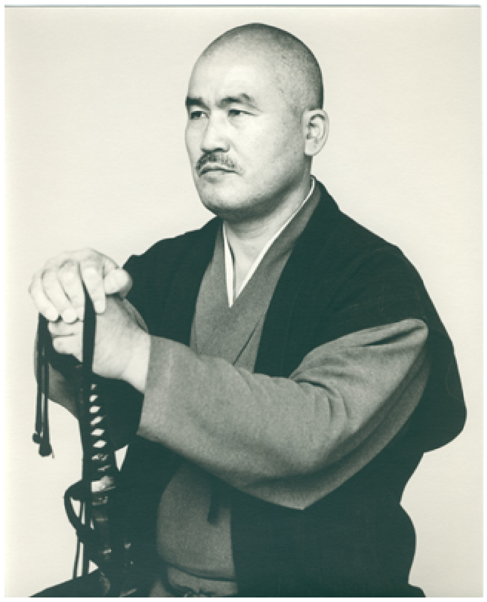


Figure 4: Toshishiro Obata, founder of Shinkendo (1948 -)[9]

Shinkendo does not have a perfect translation into English, as different parts of the word have multiple similar meanings[9], and can change the exact resulting definition. "Way of the true sword" is the easiest to explain. 'Shin' meaning true, or serious. 'Ken' translates into sword. And 'do', meaning "way of." Shinkendo is based on the traditional philosophies of Hachido (eight fold way) and Kuyo Junikun (twelve precepts of the nine planets strategem)[10] which we will get into later when we compare a similar system of describing a martial artist's control and use of the body in Fiore's manuscript. The book is broken up into important parts: history, philosophy, etiquette, and then techniques with many pictures and exact descriptions of what is being taught. Once it get's into techniques, it starts with footwork, then kammae (stances with the sword), followed by suburi, on so on, building on the previous lessons.

I will be attempting to limiting my knowledge of Shinkendo to only what is in the book, as well as what's publicly available through official sources (official website, YouTube clips, books). Shinkendo is a single sword technique, that has deep roots not only in the Samurai tradition, but also the Obata family which hailed from a region renowned in Japan for producing the best swordsmen. While it has techniques and elements from older Japanese martial arts, Shinkendo is also a modern martial art. As such, it allows the art to research and develop in the system. Unfortunately through history, when a founder of a system passes, it usually then locks the system and prevents any significant changes or innovation.

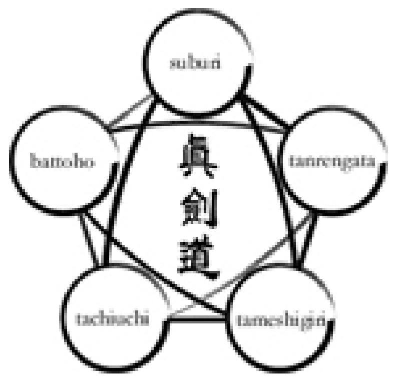


Figure 5: The five aspects of swordsmanship which are all interconnected, Gorin Gojo Gogyo[10]

Shikendo combines the five aspects of swordsmanship, suburi, tanregata, battoho, tachiuchi, and tameshigiri. Suburi refers to solo exercises, improving footwork, body movement, and sword movement. Tanregata is the solo forms, putting cuts and blocks to movement in specific sequences. Battoho, combative drawing and sheathing, is to allow for a practitioner to draw the sword safely, preform a cut, and then continue the fight. Once finished, it also covers how to put the sword away safely, but still on guard in case of an another attack. Tachiuchi is paired sparing with an opponent, that allow both students to switch between attacking and defending while moving. Finally, tameshigiri, is cutting targets to make sure that student is actually cutting with each swing, and maintaining edge control. Each aspect directly affects the other four, such as the edge control in tameshigiri being important in all. Making sure the cuts in tachiuchi are of the same, or better, quality as practiced during suburi, etc..

Usual test cutting material is tatami (rolled mats), bamboo, and tatami rolled around bamboo, which would simulate a small torso or thick arm. Obata was also able to preform Kabutowari, a case where a swordsman test their sword and their ability with a single cut into a helmet. This was often used as a contest between craftsman, armorers versus swordsmiths. Obata's cut resulted in a 13cm cut, a record breaking result, with the previous record being 10cm, 5.8mm back in November of 1886[9].

5.2.2 Flower of Battle by Fiore De Libre

Figure 6: Fiore De Libre (1350 - 1409)[16]

Fiore's manuscript, which now resides in the Paul Getty Museum, does not focus on a single weapon. Instead it uses the same vocabulary, body movement, mentality, and approach to multiple weapons. Wrestling, dagger, sword in one hand, sword in two hands, sword in armor, poleaxe, lance, and on horseback[2]. For purposes of this project, we are focusing only on the sword in two hands, though it would be ideal to eventually expand to more. The Flower of Battle has a system of four masters as stated early, but for each weapon to show the basics, the counters to the basics, the counters to the counters, and then one more layer of counters for good measure. This is only what Fiore wrote and had illustrated for each step of a certain movement, 'play', so it could be assumed he knew more than what was transcribed.

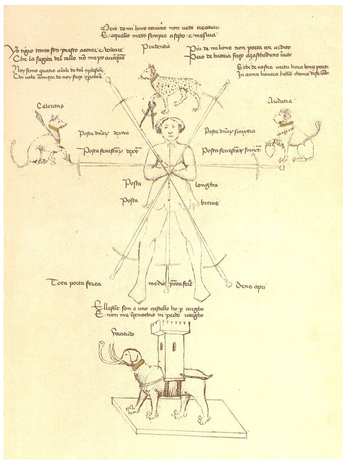


Figure 7: From Fiore's manuscript, showing the targets on an opponent. The animals used to symbolize the characteristics of the student's head, right hand, left hand, and base[2].

Fiore's section for the sword in two hands start with describing the footwork, and different ways of holding the sword. From there, the next writing and drawings are of the different targets (general) for cuts or thrusts with the sword. Afterwards, guards (stances) are introduced as well as descriptions of what each guard is meant to achieve. Finally, it starts going into the different plays (paired sparing) and exercises to practice. This is where it is very important to pay attention to garters and crowns being drawn on the two swordsman drawn, as it denotes which of the four masters is being drawn and usually involves a mistake from the non-master. Fiore's plays usually resolved in either a cut, thrust, throw, or grapple that defeats the opponent.

Fiore describes his cuts as flowing from guard to guard[3], as to keep as protected as possible. As such the way he recorded his system makes sense that the student first gets movement down. From there, it's important to learn all the guards, as they change depending on which foot is forward, and where the sword is relation to the body as well as the opponent's sword and body. Cutting from guard to guard (a high guard through a long guard, into a low guard, or vesa versa) not only keeps the sword in front protecting the person swinging, but it also keeps the cut controlled and consistent in power so that one isn't swinging wildly or timidly.

5.2.3 Personal Experience

Figure 8: Myself performing a tameshigiri demonstration at a festival in Morgan Hill, CA.

It's important to note that there are parts of sword combat that don't lend itself easily to being in digital form. For instance, it's next to impossible to try to capture every possible action a player may want to do in any given situation. As programmers, we can do our best to give the player options, some good and some bad for the situation. However, I believe I am in a rather unique place to try to add a new style of sword fighting in game, and the artifact that I am creating will be a test. I have over 10+ years of martial art experience, specifically in swords, at many times studying multiple different styles concurrently. I have trained with weapons and martial arts from Europe and Japan, from different time periods. In fact I received an instructor certificate, the right to teach, in Shinkendo, and ran a dojo for over 4 years before I decided to return to college. I have used wood swords, rebated steel swords, and live swords (2000+ grit edge polish for test cutting), both in personal practice and in public demonstrations. I have many holes in my knowledge, and many things left to learn, I do not pretend to be anything more than a journeyman in skill.

However, the way I was trained, combined with the programming skills I have learned during my attendance at the University of Limerick, that I could show a different side of sword martial arts. Instead of real time, have it turn based with concurrent execution of selected moves. Instead of free movement and action, being able to jump up and kick the air while the opponent is no where close and in no danger, limiting actions based on the situation. However with limiting the actions, it can add to the realism, and hopefully enjoyment, as more complex and situation specific moves are offered to the player. With this approach, it should take the focus away from quick reflexes, what attacks deal more damage, which is more common in today's game, and changed the focus to being about timing, distance, and swordplay in a deeper level, without requiring knowledge from the player.

5.3 Existing Combat Systems in Video Games

Fighting games have been a staple genre of video games as early as 1976 with the game Heavyweight Champ[15]. From there, the genre evolved and expanded to what it is today, including the plethora of sub genres. As a general rule, most games enjoy a lot of freedom and agency when it comes to the design portion of the project, including what genres the game would then fall under. The artifact will fall underneath the overall umbrella of fighting genre, but would also have the following tags or descriptors: turn based, fighting, strategy, swordplay.

Fighting games typically have certain characteristics that are common to most games that are encompassed under this broad category. Of course it goes without saying, that while some/all of these characteristics are common to the game, it is by no means a requirement. For example, arcade fighting games were quite common in arcades in the early 90's. Such games as Mortal Kombat, Street Fighter, Soulcalibur, etc. had certain gameplay mechanics and controls that were common to them. Most games were 2d, had character selections, special attacks, joystick controllers with buttons, jumps, blocks, matches and rounds.[15]

This is not a criticism of the games that came before, just a very broad overview identifying common characteristics, a reoccurring theme in this final year project. Fighting games have definitely evolved and changed beyond the early 90's. Different camera styles have been tried, physics engines, different controllers, new changes on existing characteristics, or entirely original ideas. As a rough generalization, most fighting games still are real time, 2d, incredibly responsive, and deep combat. Prediction, strategy, mind games, are all necessary in fighting video games.

Notable games, in no particular order:

5.3.1 Warrior

The first fighting game was a fist fighting game called Heavyweight Champ, released in 1967[15]. From there, expanding into martial arts was next for the emergence of what would become the gaming industry. In fact, the 1979 game called Warrior[20], is one of the first sword fighting games that I could find.

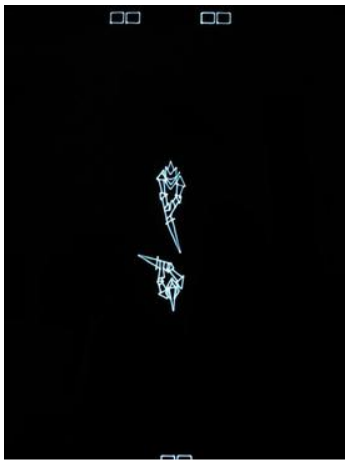


Figure 9: First video game featuring swordplay, Warrior[20]

5.3.2 God Hand

Although God Hand falls into the genre of an action game instead of fighting, there was a very interesting mechanic that was included in God Hand[17], released 2006. This is what was called the "Gods Reel" in which the player was able select a special ability after collecting enough of a certain resource. During this time the player could select from a range of powerful abilities, the best for the current situation the player finds themselves in. While the player is selecting the ability they want to execute, the enemy is stopped for the short duration that the player is forced to quickly finalize their choice. The mechanic that caught my eye was the time stoppage or time dialation while a player selects their next move(s) to be executed after.

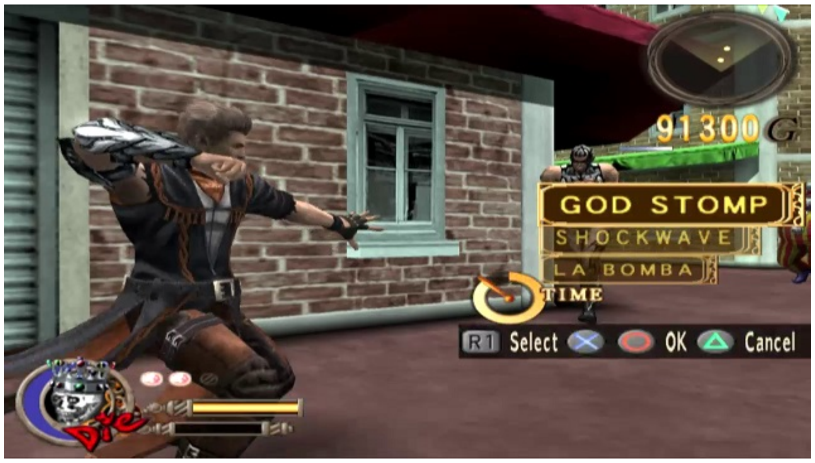


Figure 10: Godhand showing the Gods Reel[17]

5.3.3 Toribash

Toribash, released 2006, was a very interesting game for the amount of freedom of movement the player is allowed. The controls and joint manipulation that transitions between states, allow for basic to complex combinations. Once players gain an understanding of movement and able to execute desired moves is an incredibly rewarding moment, with a thrill of accomplishment. While I do not think the artifact I am creating will have anywhere near this level of free movement in terms of joint manipulation, I do want to have the same rewarding feeling of mastery from making the correct decisions that result in victories or remarkable gameplay.

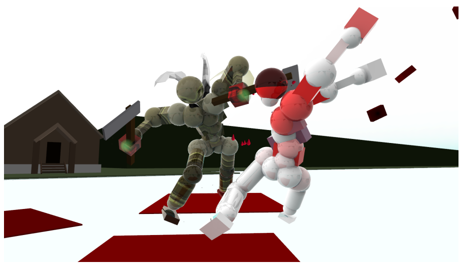


Figure 11: Example of freedom of moment allowed in Toribash[7]

5.3.4 Metal Gear Rising: Revengeance

Metal Gear Rising: Revengeance was released in 2013, and had a very interesting and unique sword combat system. Players were able to activate ”Blade Mode” which slowed time and allowed the player to make precise rapid cuts to destroy, disable, or dismember most game objects, not just enemies. Once time resumed, the damage calculations would resolve and physics applied in rather spectacular fashion. This again leads to deeper combat and options for the player to maximize their chances to win. Ideally what I would like to have in the artifact, maybe at a later time, is the ability to have more control over the cuts (and blocks).

As an example instead of a free aiming mechanic like in Metal Gear Rising: Revengeance, the player would choose the amount of power they want to put into a cut towards a chosen target on their opponent. Based off of how much power they select, that determines when the body animation begins and ends, independent of when the sword motion begins. What that means, is a player may do a huge wind up strike, but that involves them stepping into an attack, anchoring the body, and then perform a stronger cutting motion. If they do this, a block is almost useless. However there are two possible counters, either create distance so that the attack misses, or attack the opponent when the enter close distance, but before their weapon becomes a threat.

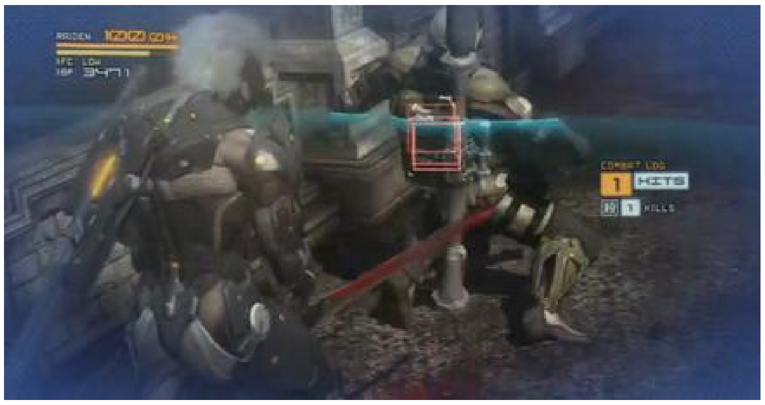


Figure 12: The "Blade Mode" in use in Metal Gear Rising: Revengeance[18]

5.3.5 Soul Calibur

The Soul Calibur series, originally released in 1998, is a very well known and established series developed by Project Soul[19]. It has many of the traditional characteristics of a fighting game, centered around melee combat. The camera system is 2d, was originally an arcade but now designed for controllers, special moves unique to each character, meaningful character selection, and rounds and matches. In terms of an industry standard for fighting games involving swordplay, this would likely be one of the strongest contenders.

Figure 13: A screenshot from the original Soul Calibur[19]

6 Design

6.1 Building the Common Core

To start this process, the first step was to select the weapon and armor style for the martial art that is being developed. When it comes to weapons, they're are millions of historical relics to use as is or as inspiration, however the single sword was selected for a specific reason. It is rarely the best in any circumstance, whether it be outside, inside, on foot, on horseback, but is also a weapon that can be effective in almost every situation. For example, the sheer range of a spear is superior to the sword when combat is outside, however the spear's range becomes a huge liability and unwieldy if used indoors. As for armor, it would also add complications. As such, for purpose of proof of concept, the project is desired to be single sword vs single sword, with no armor. Another benefit on focusing on single sword combat, is that because the sword was so common in history and across the globe, there is no shortage of sword martial arts that could be put into the system .

6.1.1 Weapon and Armour

Once you account for different weapons, resulting from different armor styles (European using combinations of plate and chain, with the Japanese using lacquered layered armor), there were many commonalities between approach and actual martial techniques. Since this is the overall combat system, it was important to see what both systems deemed as important in the approach and logistics of combat. Another important difference was the swords, a single edged curved sword, this is the famous katana, also known as a shinken. The sword being used by Fiore are double edged straight swords. There is also the length difference, weight, hand guards, and fullers (the groves down the center of the blade meant to give a wider blade without adding weight). For this artifact, the idea will be a double edged straight sword with no armour.

6.1.2 Mechanics and Philosophy

Kuyo Junikun, mentioned before, is at the core of Shinkendo. The 12 precepts include: ki (energy), shin (mind, heart, spirit), sei (accuracy), kan (distance in both space and time), chi (intelligence), soku (breathing), bin (speed), kan (5 senses), gi (technique), riki (physical and mental power), setsu (experience), and dan (judgment)[10]. For Fiore, he used 4 animals to describe something similar, the animals being the lynx, lion, elephant, and tiger[2]. The lynx is pictured at the top of the head, judging time and distance. The lion, by the left hand, doesn't stand for strength, but for courage. One theory is that this because of the grappling in Fiore's system, the left hand has to be courageous to extend and attempt to grapple an opponent. Next to the right hand is the tiger, which encourages speed with the technique and quickly finishing a movement or play and returning to an appropriate guard. Finally the elephant is based at the feet, note that there are no knees drawn on the elephant to emphasis stability and balance.

6.2 Combat Mechanics

Timing, judgment, distance, speed, and power are highlighted in both systems, which have no recorded interaction in history as far as I could discover. Note here with power, it means having enough to be effective in the physical aspect, but also having mental strength to stay composed. The physical power is not the idea of raw, overwhelming, brutal power. From here, a combat system would need cover these five components. Games typically rely on managing resources to create interesting choices. What weapon may do more damage than another, but be slower, similar with armor but with what it does to stamina, but allows for more damage to be received. With the five components in mind, the next step was to design a system around those components.

6.2.1 Timing

While real-time combat can be exhilarating, I wanted a slower pace game that more resembles speed chess with quick, decisive, deliberate actions, rather than having a player button mash while an animation. There is nothing wrong with this, I actually greatly enjoy a game in which I have grown to love their combat system. With this in mind, the combat for my artifact will be turn based. Both opponents select movement, and then based on their position relative to their opponent, the actions they can do. To keep the players engaged, there will be a timer on a short countdown so that players are forced to make a quick decision and keep the flow of combat. When both players select their movement and actions, the game then resumes and executes both actions concurrently.

6.2.2 Judgment

Judgement has multiple forms in this project. Firstly would be the 8 different stances to choose from, with two intents. This comes in the locked on portion of the game system, and is inspired by the For Honor, where a player moves their input in one of three directions, and the game would react accordingly. My system would have 8 sword positions, as well as if the intent of the player is to offend or defend, to draw inspiration from Fiore. If a player intends to offend (with the sharp part of a sword), or if a player wants to defend (and not have the sharp part of a sword enter them).

6.2.3 Distance

The approach to distance was actually more complicated than people would casually think. It should be a simple matter of if you're in range of the enemy when preforming a cut or thrust, you'll injure them! Range actually has three definitions, which are replicated in a different form in-game. Close range, swords completely crossed at the midpoint or closer. This is the space at which the Fiore would go either into half-swording[2] or grappling[3]. The next distance would be correct distance, the distance at which both swords are crossed only at the top third of the blade. An attack from this distance, with extension of the arm, would at most hit a limb, or with forward footwork, a cut or thrust to the body. Please note here, that I try to avoid the term of a 'strike' or 'hit'. This isn't just being pedantic, but an important reinforcement of the core concept of the game. Out of distance would be when the swords don't cross at all, and one person has to move towards the other to even get into correct range.

This was replicated in design and development by having the footwork be controlled by the player choosing an action, and then the correct animation would start that moved the skeletal mesh of the character model. This is done by making the animations root animations, and will be discussed later. Here is a portion where the game actually needs to step in and resolve player choices before playing animations. For example, if both players choose to step in from correct distance, they would run into the other (headbutt or accidental mutual skewering). While in real life, this would either be hilarious or the end of the fight, it wouldn't do well at fulfilling the goal of the project. For this reason, the game would measure the in-game distance between the two players, and chose between an animation that moves the character, or the other one that doesn't. The idea, if both players are or will be too close, the animation that doesn't move the character, otherwise the moving animation will be used. The same idea is for moving left and right. These fights are very angular rather than strictly linear. However, if one player moves to their left, and the other player moves to their right, if both try to that angular movement then they will also bump into the other. The game will be able to decide between playing an angular animation or a more linear animation. This way not only do both opponents keep facing one another, but would do what most of us do naturally, read body language. An example is walking through a crowded footpath, we've all had the dance when we come across someone walking in the same line on the same path. Sometimes we successfully negotiate the passing with a simple movement to the side and they move to the other. Other times there is an accidental dance of trying to navigate the impasse. The idea of this system would be to allow the game to recognise and resolve it.

One of the ways other ways this is being represented is the sphere triggers attached to the player and enemy game objects. When within the largest sphere, the player is able to lock on to the enemy, and the enemy automatically locks on and turns towards the player. A bit closer, and then the player is forced to lock onto the enemy, along with all the game mechanics that come to that. At this range, either player or computer can choose to start combat. If the player gets too close to the opponent, then combat is automatically started. The idea is this does replicate how one would approach an opponent. At long range, not too worried and can just keep a wary eye. Closer, almost on the edge of the out of range, defined earlier, the player is now in risk from a lunging attack, so the sword should be out to either threaten or proactively out to block an attack attempt. Once the swords come to correct distance, the two combatants are almost at the close range.

6.2.4 Speed

This was one of the harder to design around. Because of the turn base combat executing actions from both parties concurrently, it meant that all the animations have take a certain amount of time, and finish by that time so the next action can start. Speed would likely come up when dealing with different weapons and moves, however I did not want to rely on that alone. Because of the position and intent system for the swordwork, speed had to be addressed in terms of the flow of combat. So without any built in incentives (meaning things like a buff to a stat, temporary invulnerability, etc), to flow easily from a block to a different position to launch an attack can be done very clumsily or quickly. There is also different ways to recover from a downward cut that failed to connect, and coming back to the same line for another cut. For example, one way is very protective by immediately going to a defensive sword positions until raising back into a defensive top guard, and then into an offensive one at which point a cut could be launched. Another way would be after the initial cut, recover with a rising cut on the same angle which may strike the hand or body of the opponent as you arrive at the original launching point quicker. There are also many other options the player could come back to the same position, and that is the meaningful player choice that is part of successful games, or at least my attempt at it.

6.2.5 Power

There was an initial part of me that wanted to put a simple power selector and have that either slow or speed up the selected action. The more powerful the motion the slower it is, versus a faster action with less power. The idea is this mimics the speed of relaxed muscles versus tense muscles. However, power for both of these martial arts is based off of footwork. The weakest cut, in terms of power, is stepping backwards at the same time. Because of the footwork and body mechanics, the natural power is quite weak, compared to a small movement forward. The small movement forward allows the body weight to shift forward, instead of backwards with the step back, and allows the body to preform a stronger and more controlled cut. The strongest cut is stepping through a cut, meaning stepping forward at the same time as fully extending and cutting forward). This also takes the most distance and time to preform. This will be replicated in game through the technique part. So the player will be selecting their power, but with their footwork instead of their arm.

7 Translating into Game Mechanics

7.1 Footwork

7.1.1 Open World

Once I knew I wanted to do a turn based system, the next logical base was a grid travel system. This is very typically in tactical top-down turn base games. I knew I wanted to use a third person camera, that could still work with a tile system. However since I wanted to show this idea could work in a free roaming environment, I came up with the idea of moving the characters in set directions and angles during the execution stage of the turn system.

So when the game is in the world state it would be normally controls. This would be WASD for movement and mouse for camera control.

7.1.2 Turn Base

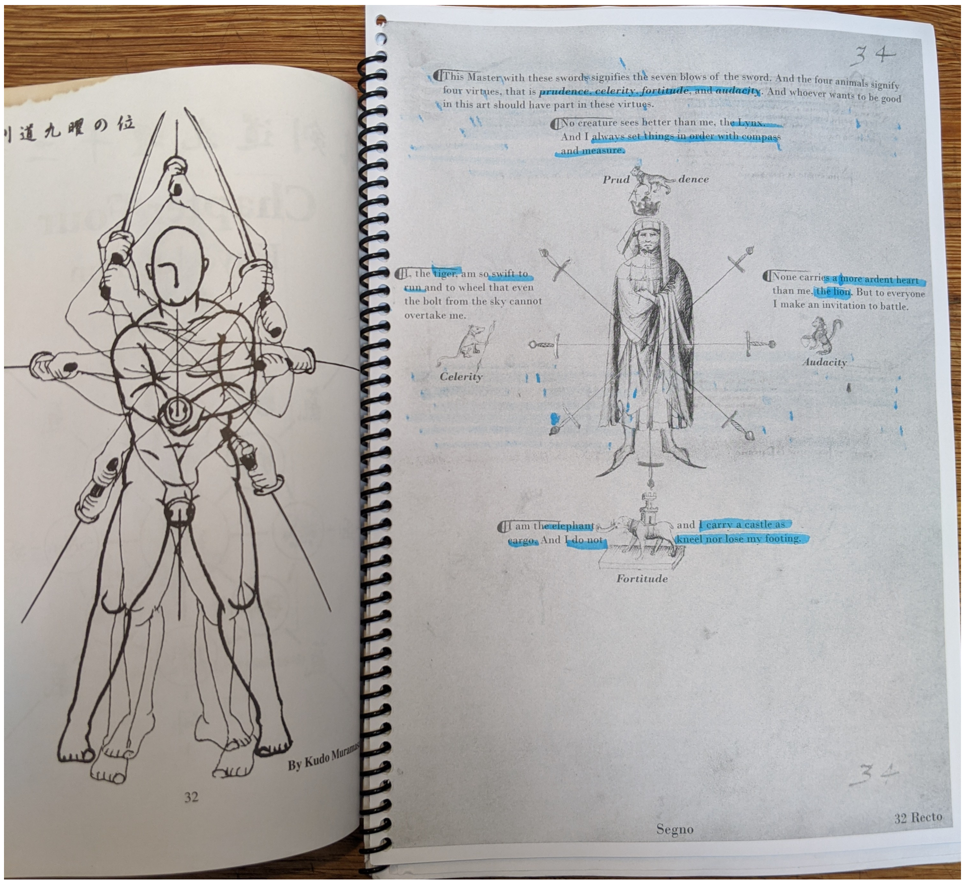
For the turn base combat portion, I danced between a bunch of implementations. In the end, I decided to have the player select which direction they wanted to move, as well as what foot forward they wanted at the end of the movement. This will play a crucial factor later. This would be the difference between if a movement is considered a step change or not. This will not only be critical in calculating technique, but also how big a movement is. A same foot forward movement is smaller in distance or angle than a step (foot change) movement. The only difference is if not direction is chosen, a player can keep same foot forward or step change to the other foot forward in the same location.

7.2 Swordwork

Swordwork would only be enabled during the lock on portion of the open world game state. Lock on, mentioned earlier in the report, has both players facing each other, so their WASD movement is in relation to where the camera is aimed, which is aimed at the opponent.

7.2.1 Sword Positions

Since the camera is controlled, the mouse then becomes the way to switch sword positions. Once the game state exits the world state, and then switches between the planning and execution portion of the turn base (until exiting combat range or one character dies). The sword position means where the sword is in relation to the body.



This Master with these swords signifies the seven blows of the sword. And the four animals signify four virtues, that is ***prudence, celerity, fortitude,*** and ***audacity***. And whoever wants to be good in this art should have part in these virtues.

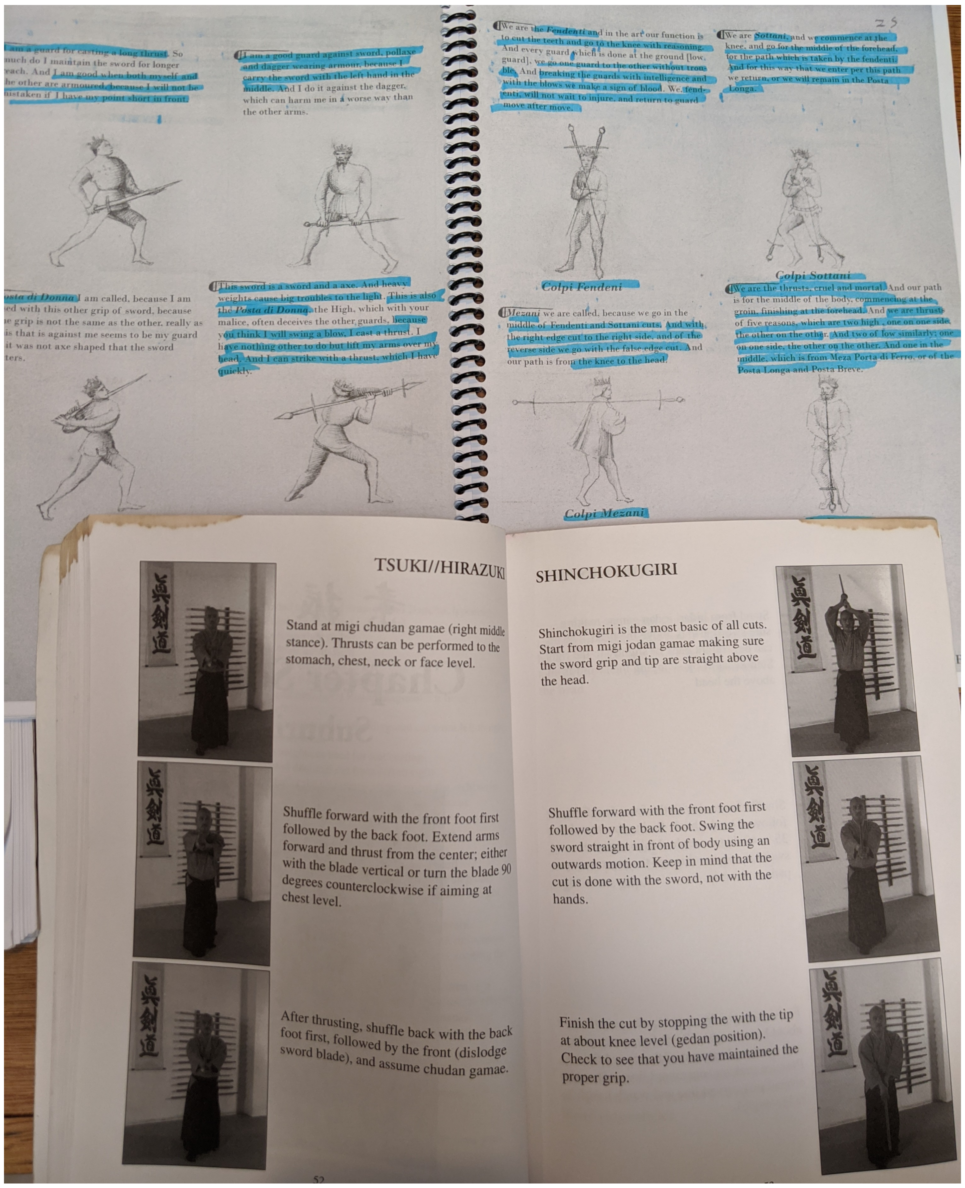
No creature sees better than me, the Lynx. And I always set things in order with compass and measure.

I, the tiger, am so swift to run and to wheel that even the bolt from the sky cannot overtake me.

None carries a more ardent heart than me, the lion. But to everyone I make an invitation to battle.

I am the elephant and I carry a castle as cargo. And I do not kneel nor lose my footing.

Figure 14: Side by side comparison of where attacks are launched from and expected to cut



am a guard for casting a long thrust. So much do I maintain the sword for longer each. And I am good when both myself and he other are armoured, because I will not be nistaken if I have my point short in front.

I am a good guard against sword, pollaxe and dagger wearing armour, because I carry the sword with the left hand in the middle. And I do it against the dagger, which can harm me in a worse way than the other arms.

We are the *Fendenti* and in the art our function is to cut the teeth and go to the knee with reasoning. And every guard which is done at the ground [low guard], we go one guard to the other without trouble. And breaking the guards with intelligence and with the blows we make a sign of blood. We, fendenti, will not wait to injure, and return to guard move after move.

25

We are *Sottani*, and we commence at the knee, and go for the middle of the forehead, for the path which is taken by the fendenti. And for this way that we enter per this path we return, or we will remain in the Posta Longa.

osta di Donna I am called, because I am ed with this other grip of sword, because e grip is not the same as the other, really as is that is against me seems to be my guard it was not axe shaped that the sword ters.

This sword is a sword and a axe. And heavy weights cause big troubles to the light. This is also the *Posta di Donna*, the High, which with your malice, often deceives the other guards, because you think I will swing a blow, I cast a thrust. I have nothing other to do but lift my arms over my head. And I can strike with a thrust, which I have quickly.

Colpi Fendeni

Mezani we are called, because we go in the middle of Fendenti and Sottani cuts. And with the right edge cut to the right side, and of the reverse side we go with the false edge cut. And our path is from the knee to the head.

Colpi Sottani

We are the thrusts, cruel and mortal. And our path is for the middle of the body, commencing at the groin, finishing at the forehead. And we are thrusts of five reasons, which are two high , one on one side, the other on the other. And two of low similarly; one on one side, the other on the other. And one in the middle, which is from Meza Porta di Ferro, or of the Posta Longa and Posta Breve.

Colpi Mezani

TSUKI//HIRAZUKI

Stand at migi chudan gamae (right middle stance). Thrusts can be performed to the stomach, chest, neck or face level.

Shuffle forward with the front foot first followed by the back foot. Extend arms forward and thrust from the center; either with the blade vertical or turn the blade 90 degrees counterclockwise if aiming at chest level.

After thrusting, shuffle back with the back foot first, followed by the front (dislodge sword blade), and assume chudan gamae.

SHINCHOKUGIRI

Shinchokugiri is the most basic of all cuts. Start from migi jodan gamae making sure the sword grip and tip are straight above the head.

Shuffle forward with the front foot first followed by the back foot. Swing the sword straight in front of body using an outwards motion. Keep in mind that the cut is done with the sword, not with the hands.

Finish the cut by stopping the with the tip at about knee level (gedan position). Check to see that you have maintained the proper grip.

Figure 15: Side by side comparison of where guards, as well as the theory of cutting guard to guard

If you then incorporate these pages in the Getty[2] and the Shinkendo book[10] together, there are quite a few similarities. The first is the idea of cutting from guard to guard in both books, just mechanical differences because both styles are unique. I copied this idea into the game, that whenever an action finishes, it has to finish in a designated sword position, guards in Fiore[2] and gammae[10] in Shinkendo.

The other similarity is there seems to be a pattern emerge between the positioning of the sword. There is three different vertical positions, as well as three different horizontal positions. This forms a 3 by 3 grid, of upper, middle, or lower for vertical and left, centre, and right for horizontal. This gives a total of 9 different positions for the sword to be in. I lowered this to 8, partially because for using the mouse input, there is only 8 directions to move in, with middle centre being no change.

7.2.2 Intent

There are two intents, to offend and to defend, as explained previously. A person who intent is to offend will not only have better technique, because they're launching the attack from a strong position, but be able to launch the attack quicker. Similarly someone who's intent is to defend can block, absorb, or deflect an incoming attack much easier and effectively. Switching from the offense to defense can be done while changing sword positions.

If a player tries to attack from a defensive intent, it will be a slower attack (by the time the blade reaches it's furthest extension towards the opposing character). Not only that, it would not have launched from a familiar position, and would be easier to block, absorb, or deflect (a blocking action).

7.2.3 Actions

Player actions was probably the hardest choices of what to include. Not only did it determine added scope and complexity of the project, but also what options the player had for the swordwork action to be launched at the same time as the chosen footwork. I eventually decided on attack, defend, change position/intent, and hold action. At the most basic level, this is the interactions that early students would be dealing with. This is also a place to revisit for adding in special moves based off of distance, like a jumping attack forward (more of a hop), halfswording (holding the sword at the handle and halfway up the sword, useful for leverage, throws, and targeted thrusts), or even grappling (attempts to throw or disarm the opponent).

7.2.4 Technique

Technique is a calculation done internally, who's purpose is to assign an enum value of poor, medium, or good. This value is attached to the current swordwork being performed. The two variables that go into calculating the technique of that queued swordwork (because two sets of footwork and swordwork actions can be queued before the execution phase is launched) is that current intent as well as what foot will be forward at the end of the footwork portion of that paired set of actions.

Technique comes into play, pun intended, when the sword registers a collision. If the colliding object is another character model, the technique doesn't matter currently as it would start the death process on the hit character, which would launch a series of events. This could be expanded to make sure a killing attack had to be a medium or strong technique, which is up to balance. The other collision, sword on sword, is where the technique shines. The player with the higher technique will knock the the other player's sword out of line and immediately return to the planning stage, forfeiting any queued actions. The player with the higher technique would be in the offensive intent and able to launch an attack from that position, or any of the other normal actions that are available. Meanwhile the defender has to use a sword action to recover their composure and bring their sword to a position and intent. Both players have the option of footwork to be launched concurrently. After that the game returns to it's normal loop. If both characters have the same technique value, then they both start with their sword at that position with their previous intent. From there they are able to choose any action for swordwork and footwork as normal for both of their actions.

7.3 Development Software

7.3.1 Github

Github was used as the version control for this project. I have experience with Github through my college experience, outside of college, and in Co-op. Github is an excellent tool for teams and companies to manage software development with multiple programmers and the possible merge conflicts. However, there was still a problem with the .gitignore and eventually there was branch conflicts that was deleting progress. The biggest one set me back about two weeks, when I emailed my supervisor. I was able to rebuild the lost work, and eventually a new repository for the project.

7.3.2 Unreal Engine 4

A decision was made early on to use Unreal, as I had some experience with it over Co-op placement. Originally there was a debate as whether to use Unity or Unreal for this project. Unreal has a better network and multiplayer interface currently, but Unity is also has a much more robust neural network package for AI in games. I would like both multiplayer and single player vs AI to be options. Unreal Engine 4 uses Blueprints, which is a visual programming language. To program the in-game logic, a programmer connects events, functions, in a visual interface that then compiles what the programmer designed into executable C++ code. Blueprints also allows for writing entire games strictly in C++, and using the engine editor.

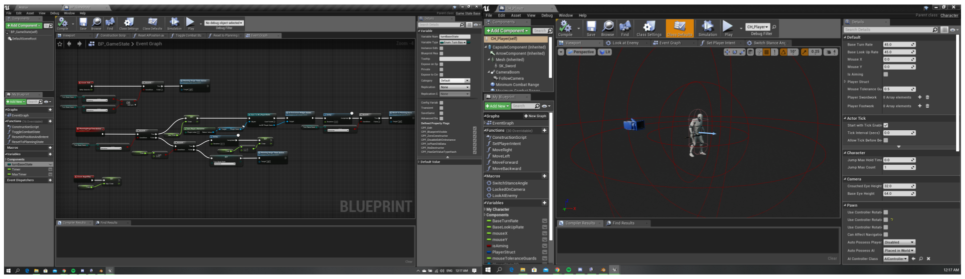


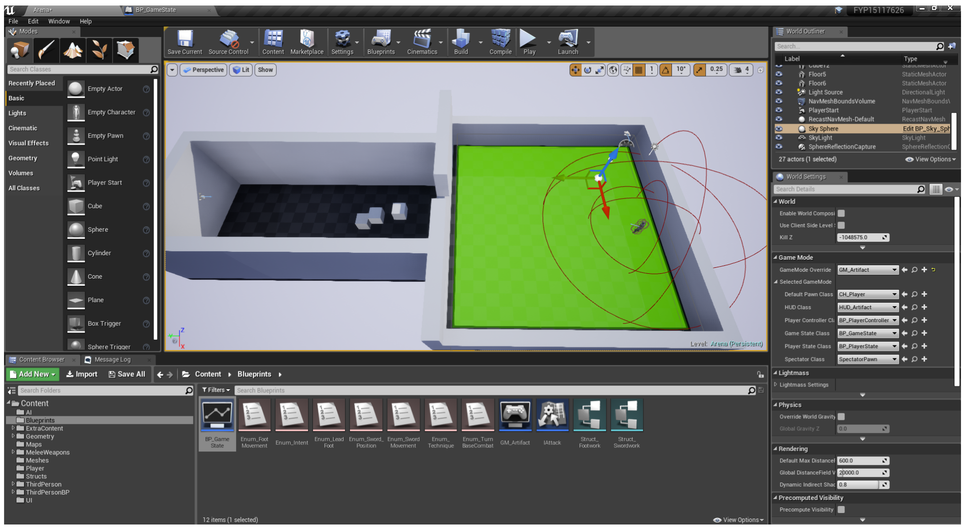
Figure 16: Example of the Blueprints(left) and Unreal Engine 4 editor(right)

7.3.3 Blender

Originally the plan was to use a Unreal Marketplace asset called ARTv1 (Animation and Rigging Tool) for quickly editting and creating the animations I needed for this project. Unfortunately, not having first hand experience of making animations at a high level, there were not a lot of tutorials to help me find my bearings. After exhausting that possibility, I moved quickly to Blender, as it's famous, free to use, with many options for tutorials, forums, and videos on how to do what I need.

8 Development

8.1 In-Game Assets



Label	Type
Floor5	StaticMeshActor
Floor6	StaticMeshActor
Light Source	DirectionalLight
NavMeshBoundsVolume	NavMeshBounds\
PlayerStart	PlayerStart
RecastNavMesh-Default	RecastNavMesh
Sky Sphere	Edit BP_Sky_Sph
SkyLight	SkyLight
SphereReflectionCapture	SphereReflectionC

27 actors (1 selected)

World Settings

World
- Enable World Composi
- Use Client Side Level
- Kill Z: -1048575.0

Game Mode
- GameMode Override: GM_Artifact
- Selected GameMode
  - Default Pawn Class: CH_Player
  - HUD Class: HUD_Artifact
  - Player Controller Cl: BP_PlayerController
  - Game State Class: BP_GameState
  - Player State Class: BP_PlayerState
  - Spectator Class: SpectatorPawn

Lightmass
- Lightmass Settings

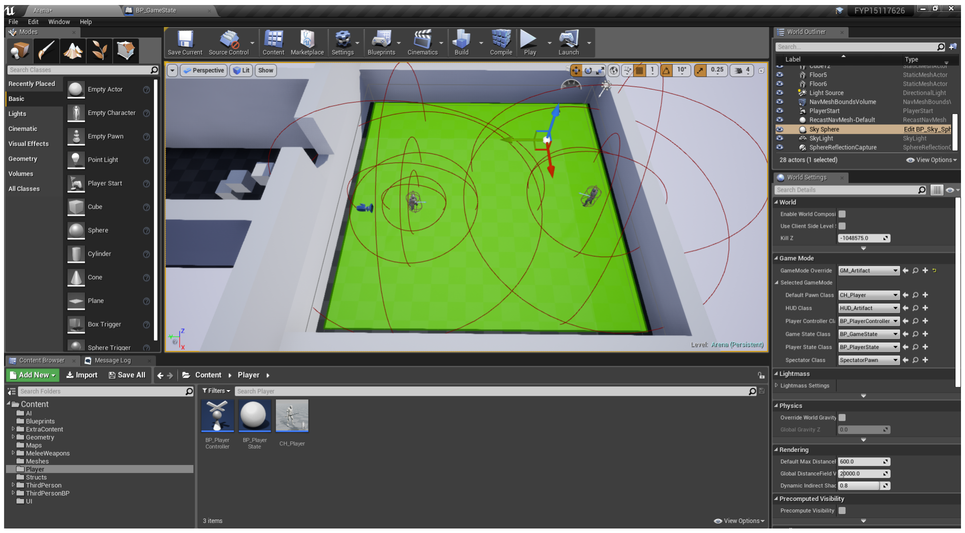
Physics
- Override World Gravity
- Global Gravity Z: 0.0

Rendering
- Default Max Distance: 600.0
- Global DistanceField V: 20000.0
- Dynamic Indirect Shac: 0.8

Precomputed Visibility
- Precompute Visibility

Figure 17: The simple arena I created for playtesting.

This is a simple arena I made to be able to control the playtesting of the project. It starts with a simple corridor with boxes to jump on to get into the square portion, the actual arena. Once in here, the player has enough room to play with the distances of max lock on, and minimum locked on & maximum combat range, and then minimum combat range, in descending radii. The area is enclosed to prevent players or the AI from running off the edge of the map. The green portion of the square arena is baked with a navmesh for the AI agent to navigate.



Label	Type
Floor5	StaticMeshActor
Floor6	StaticMeshActor
Light Source	DirectionalLight
NavMeshBoundsVolume	NavMeshBounds\
PlayerStart	PlayerStart
RecastNavMesh-Default	RecastNavMesh
Sky Sphere	Edit BP_Sky_Sph
SkyLight	SkyLight
SphereReflectionCapture	SphereReflectionC

28 actors (1 selected)

World Settings

Kill Z	-1048575.0
GameMode Override	GM_Artifact
Default Pawn Class	CH_Player
HUD Class	HUD_Artifact
Player Controller Cl	BP_PlayerController
Game State Class	BP_GameState
Player State Class	BP_PlayerState
Spectator Class	SpectatorPawn
Global Gravity Z	0.0
Default Max Distance	600.0
Global DistanceField V	20000.0
Dynamic Indirect Shac	0.8

Figure 18: This shows the CH_Player in the arena with CH_Enemy.

The same map as before, but in here is the placed CH_Player character that the player spawns in control of. Here you can see the three ride wire framed spheres that indicate the distance checks mentioned above. The CH_Enemy, the preplaced computer opponent that CH_Player responds to, has two checks, for the automatic lock (the larger sphere) and distance at which the computer can initiate combat (the smaller sphere).

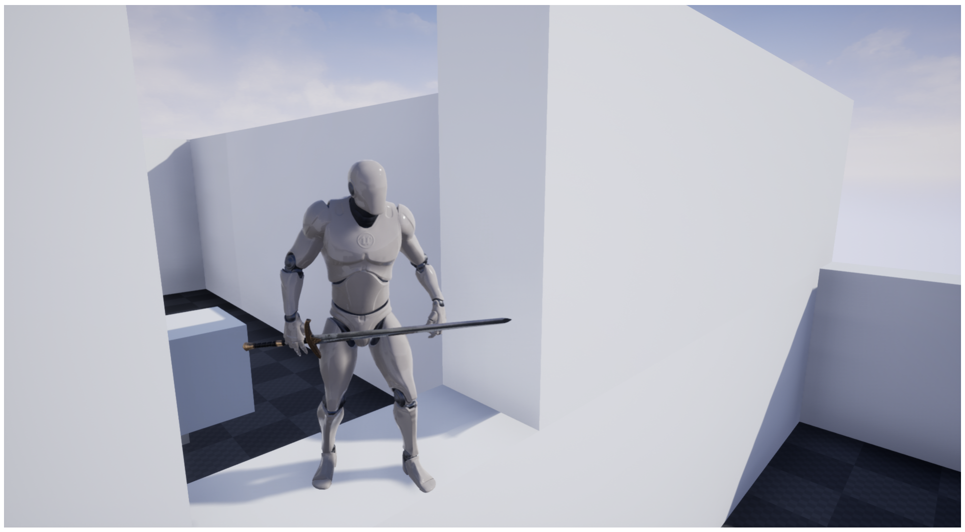


Figure 19: The Unreal 4 Mannequin Man with a free Unreal Marketplace Sword asset

This is the Unreal Engine 4 Mannequin Man[14] who's skeletal mesh I modified to include a socket. The next step was to attach the two handed sword I downloaded from the Unreal Marketplace[13]. Both CH_Player and CH_Enemy use the same in-game model.

8.2 Unreal Blueprints

Obviously I am not going to include every blueprint, equivalent to every screenshot of code, but will only be highlighting a few. I wrote this program with as much proper programming technique and standards that is expected from our lecturers. Meaning each blueprint is encapsulated that communicates through a planned hierarchy and proper method calls instead of direct variable manipulation.

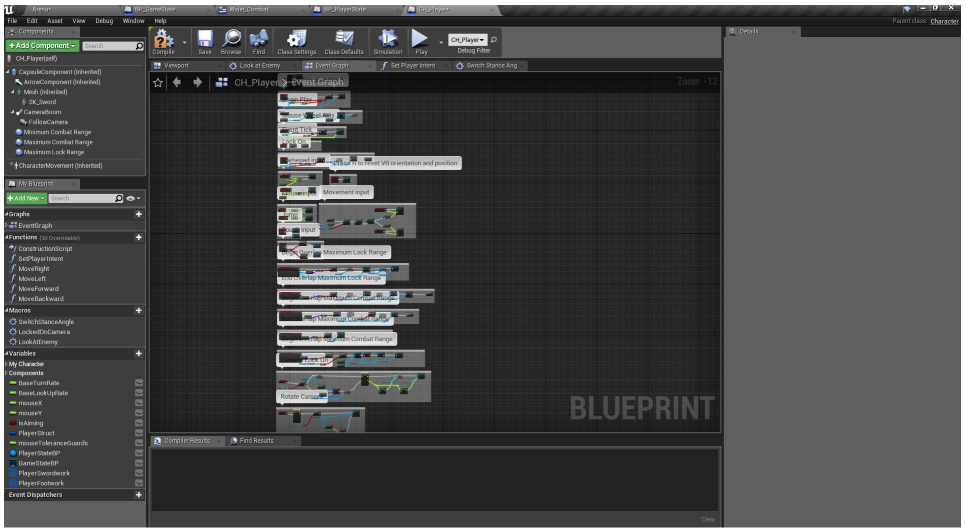


Figure 20: A zoomed out look at the many event calls and checks in CH_Player, a few were already generated and unchanged.

For example, CH_Player is responsible for capturing player input, collision detections, and the physics of the player object. BP_PlayerState sets the variables of CH_Player in terms of game logic or what the programming decides is needed to create. It also is responsible for the internal game logic when it comes to variables changing, sending out event calls or calling methods in either CH_Player or BP_PlayerController. BP_PlayerController communicates between BP_PlayerState and then BP_GameState. BP_PlayerController controls how significant changes in BP_PlayerState and how it affects the player in terms of the game logic, such as the telling CH_Player to launch a certain animation, with a sound effect, and notify the BP_GameState about the result of the shot (if programmed to).

BP_GameState is similar to BP_PlayerState, but is in charge of the game's variables such as a global timer, score, etc.. BP_GameState then sends out events or method calls to BP_GameController when certain variables pass thresholds. BP_GameController then is responsible for such things as changing levels, pausing, spawning, etc.[12]. This level of communication, hierarchy, encapsulation, and attention to the "-ilities" is consistent proper programming practices and should reflect as such.

8.2.1 Change Sword Position

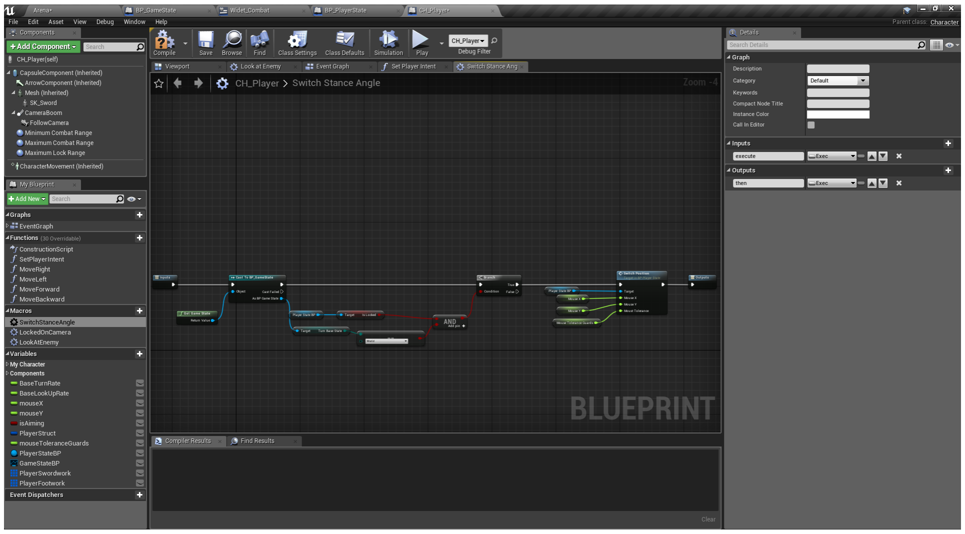


Figure 21: Passing the saved mouse input values, when appropriate, to the BP_PlayerState.

The first part of converting mouse input to a matching sword position is actually saving the current value of the mouse input. Before continuing, there is a check to make sure the player is in the locked on state. This is to allow for normal free camera motion when not locked on, but when locked on the player then can use the mouse to control the sword position. Those values are then passed to the BP_PlayerState method SwitchPositions as well as a threshold to be checked against.

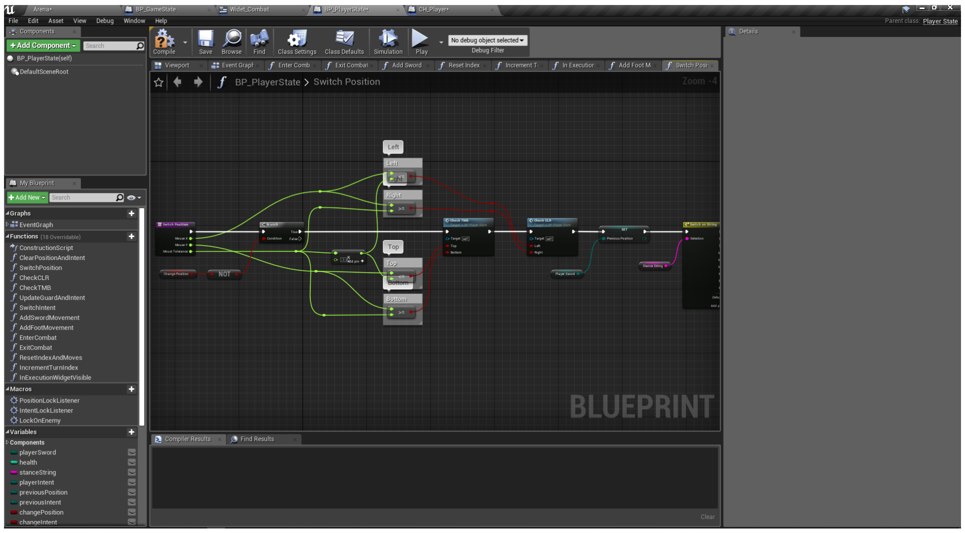


Figure 22: This is the first part of the function that converts the mouse input to sword position.

In this function, the first check is to see if a there is a change in position. This is because the mouse not moving, under the passed threshold, should be a dead zone in both directions. This is also how the game detects if the mouse is only moving vertically or horizontally. It then calls the CheckTMB and CheckCLR functions that set the value of a string.

The reason for the string is because of how switch statements are programmed in UE4. The options were switch cases from a string, int (integer), or enum state. The enum state would have taken a bit more programming, and the string or int was easier. Int was not a usable option, because BP incremented the values automatically, and I could not do my planned version of having a two digit value to represent what sword position the current mouse input was passing. The first digit would be the top, middle, or bottom check result, and then the second digit would be the left, centre, or right check result.

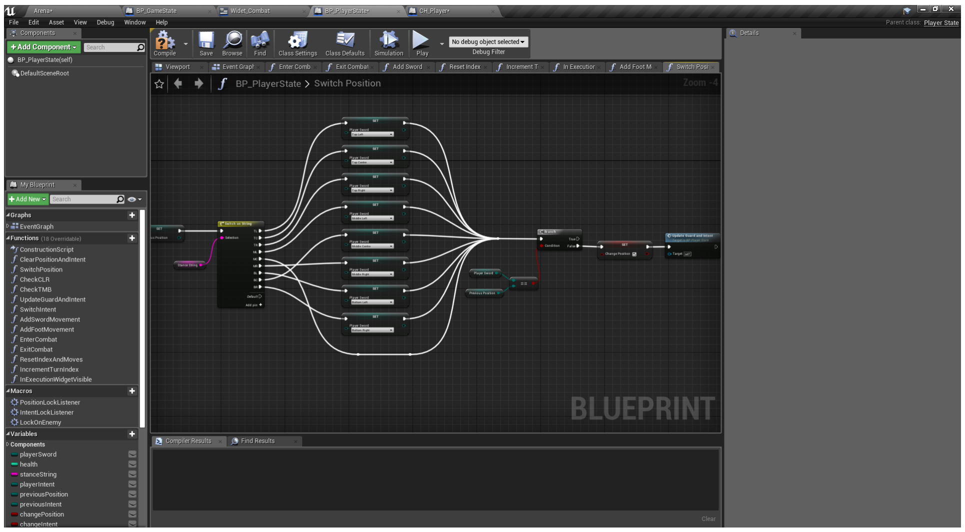


Figure 23: This is the second part of the SwitchPosition method.

The player's current sword position is saved as a temporary value to check later. Then the switch statement then executes the appropriate setter for the player's current position. MC, middle centre, bypasses the next check as it means there was no mouse input. This is the reason why the player has to pull straight down on the mouse to get to the middle centre sword position. The method then checks to see if the current position is the same as the previous, and then continues as appropriate.

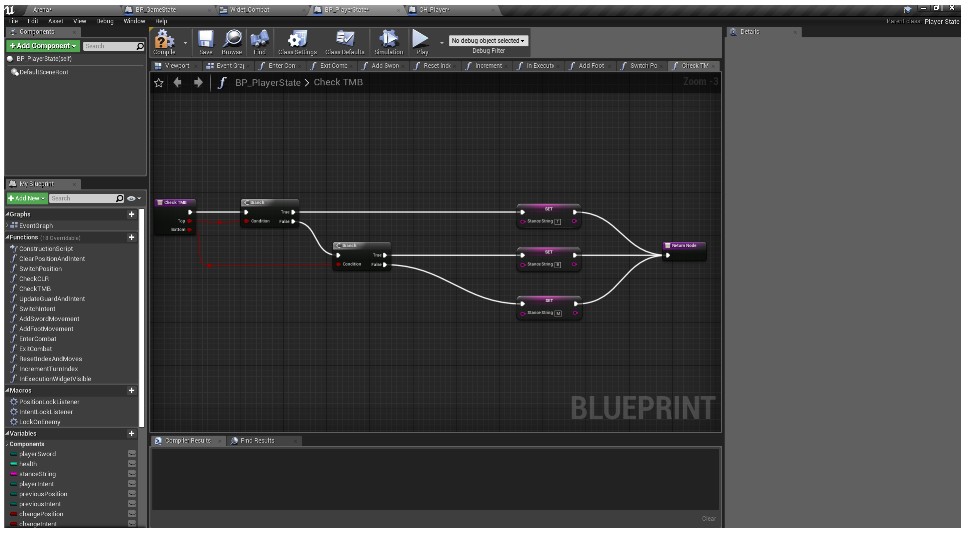


Figure 24: This checks the horizontal input of the mouse and sets the first value of the string.

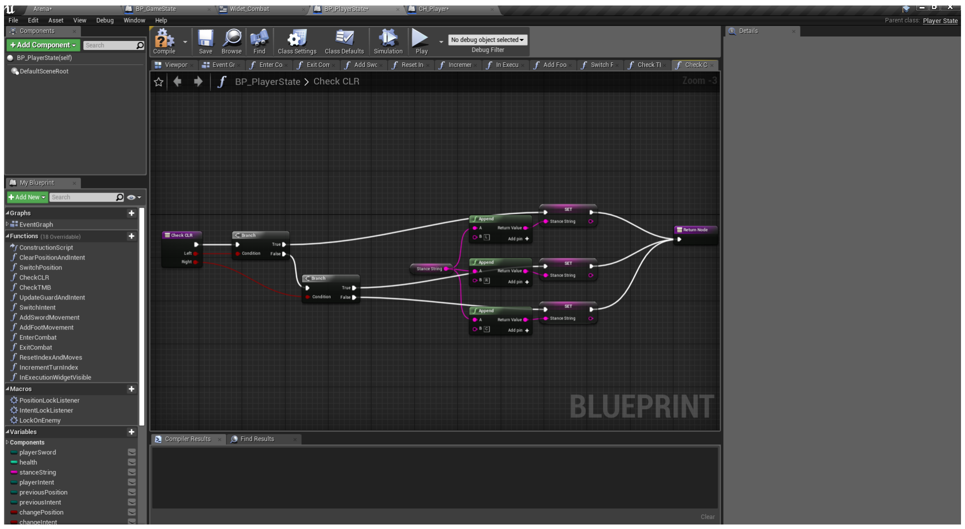


Figure 25: This checks the horizontal input of the mouse and appends to the string.

8.2.2 Turn Based System

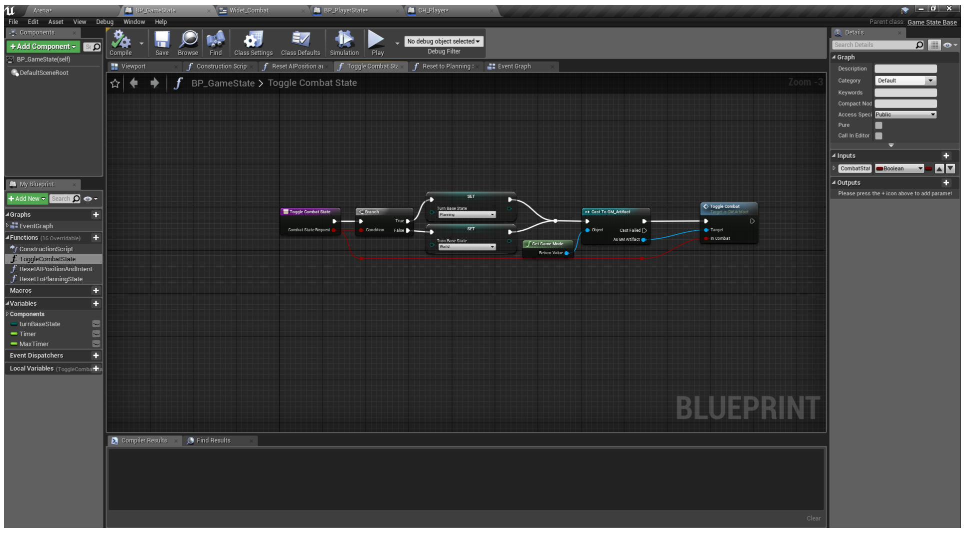


Figure 26: This is the blueprint that is responsible for transition between the paused turn base combat and the non-combat gameplay.

An example of how I kept all the game values and logic running smoothly. Any change that affects another blueprint is first checked, and then calls the appropriate methods in the other blueprint(s) to limit the change of important values. In this example, when the player enters combat and goes from the WASD and mouse input to having those disabled, mouse reenabled and the start of the turn based combat in the planning stage. The call to GM_Artifact ToggleCombat method then starts calling the appropriate methods and events to keep each game object and blueprint operating without conflicting method calls.

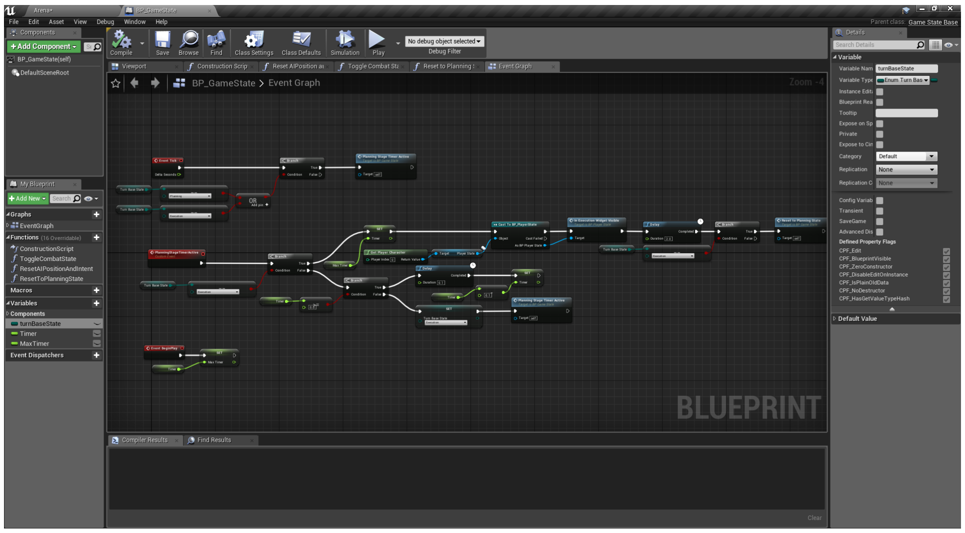


Figure 27: My way of creating a turn based combat system with a timer that can be interrupted.

This is my attempt at creating a turn based system in UE4. This took a while to design, test, and refine until I ended up with this. The requirements of my turn based system was that both combatants could queue up two footwork actions and two swordwork actions. The first footwork action and the first swordwork action would execute concurrently. If there was no change in the game state, meaning if one died or the distance was beyond the maximum combat range, then the second set of actions would execute concurrently. This is all happening at the same time the other combatants actions are playing out. If no action is queued, then obviously nothing happens for that slot.

I wanted to have a ten second timer to countdown on the screen that forced the player to quickly choose their next actions. This is the speed chess feeling I was going for that I mentioned earlier. Once the timer reduces to 0.0f, then the turn base state turns to the execution phase. However the timer can be bypassed, and properly reset, if the base state turns to execution from another source.

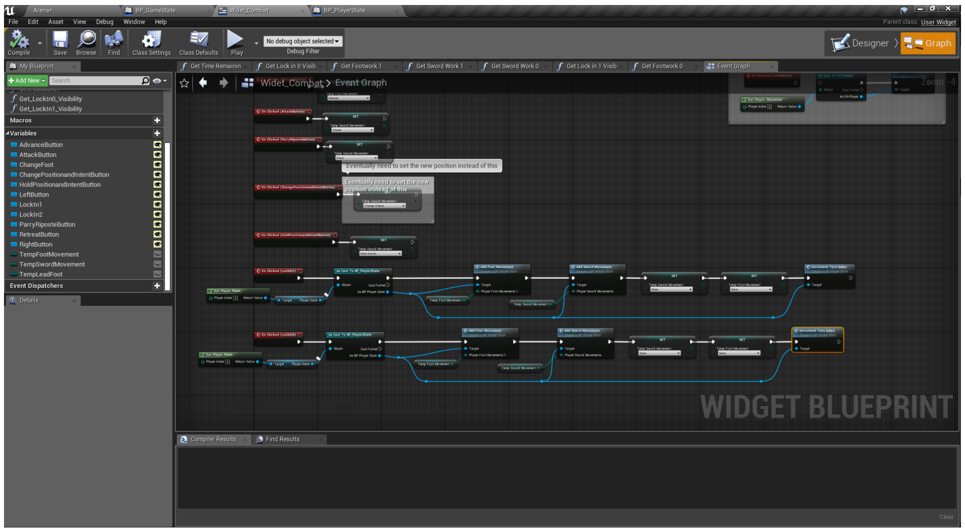


Figure 28: Adding player's both actions to the queue of a maximum of 2 pairs of actions.

Here is the programming that adds the players current selection of footwork and swordwork actions. The OnClick(LockedIn1) event sets the current selection sets the temporary selection to a 'none' value and increments the turn index. The turn index is actually the index for the spot in the footwork and swordwork arrays. What is not shown in this blueprint, is the increment turn index call to BP_PlayerState has a check to see if this is the second set of actions being locked in. If it is, then it resets the turn index and sets the base state is set to execution, which is then handled in the previous paragraph.

9 Animation

9.1 Mixamo

In an effort to speed the process along, instead of creating a new animation from scratch, I went to Mixamo and downloaded a free to use pack of animations[5]. By finding animations that are close to what I want, I was in the best position to learn Blender and how to animate in it.

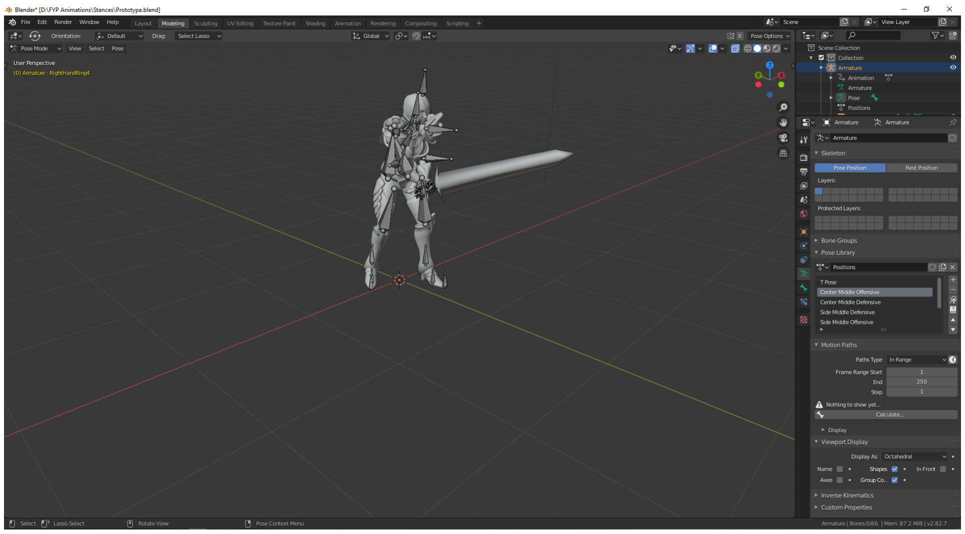


Figure 29: The start of one of the animations downloaded from Mixamo, inside of Blender.

9.2 Layered Animations

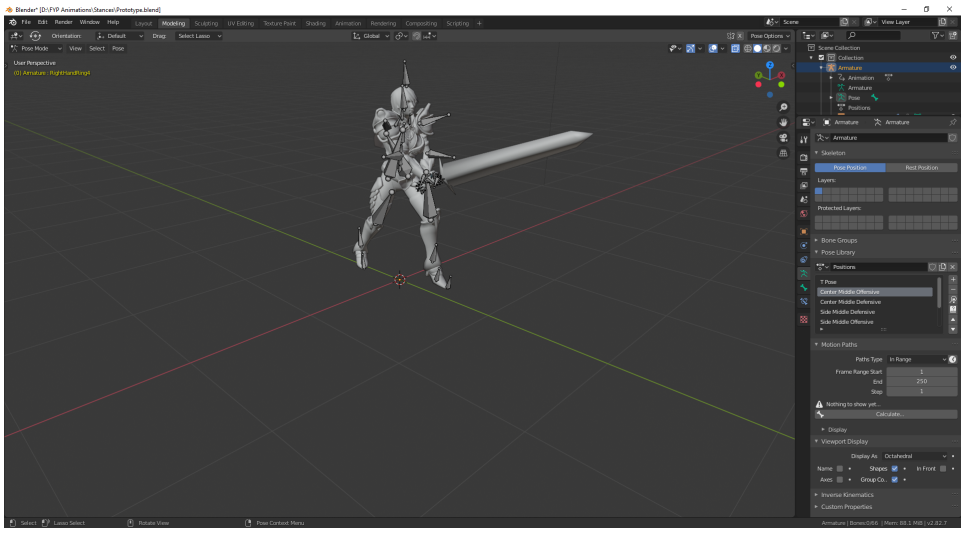
At first I looked at AimOffset and Blendspace as a way to have separate animations play for the lower body and upper body. These two are great technologies, but not what I was looking for and couldn't be implemented without major refactoring in variables. As a result of my research into this, I found layered animations and that it was supported in Unreal Engine 4. This would allow me to play animations on the lower half of the body, footwork, independently of the swordwork, the upper body. This would greatly reduce the number of animations I would have to create, otherwise I would have to make: 4 (directions) x 2 (step change) + 1 (in place step change) = 9 variations of each sword movement.

9.3 Root Animation

For an animation to move the in-game asset in Unreal Engine, if the animation is meant to move the Vector3 location, then a root bone has to be added to the animation. This root bone is parented to the hip (to move with the hip bone). Often constraints are places on the root bone so the character moves in a straight line (for a lunging attack as an example).

9.4 Created Poses

The first step to creating the custom animations I need for the project was to create all the poses I needed. By adding the poses in a certain key frames, and reusing as many poses as possible, Blender would have an easier time interpolating the movement between key frames. This also keeps with the idea of cutting from guard to guard. Also the sword position is important for collision detection depending on the game, and this is artifact is no exception. I was also able to switch left foot and right foot forward, as I created two poses that only affected the lower body, again speeding up my time in Blender.


Blender* [D:\FYP Animations\Stances\Prototype.blend]

User Perspective
(0) Armature : RightHandRing4

Pose Library — Positions:
- T Pose
- Center Middle Offensive
- Center Middle Defensive
- Side Middle Defensive
- Side Middle Offensive

Motion Paths: Paths Type: In Range; Frame Range Start: 1; End: 250; Step: 1

Nothing to show yet...

Viewport Display: Display As: Octahedral

Armature | Bones:0/66 | Mem: 88.1 MiB | v2.82.7

Figure 30: The sword position of middle center, offensive intent, created in Blender.

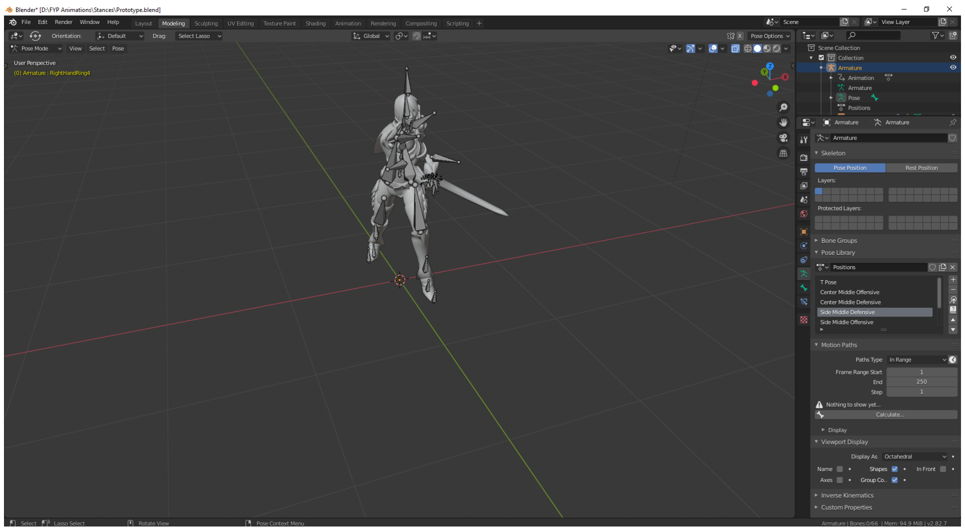


Figure 31: Sword position middle left, defensive intent.

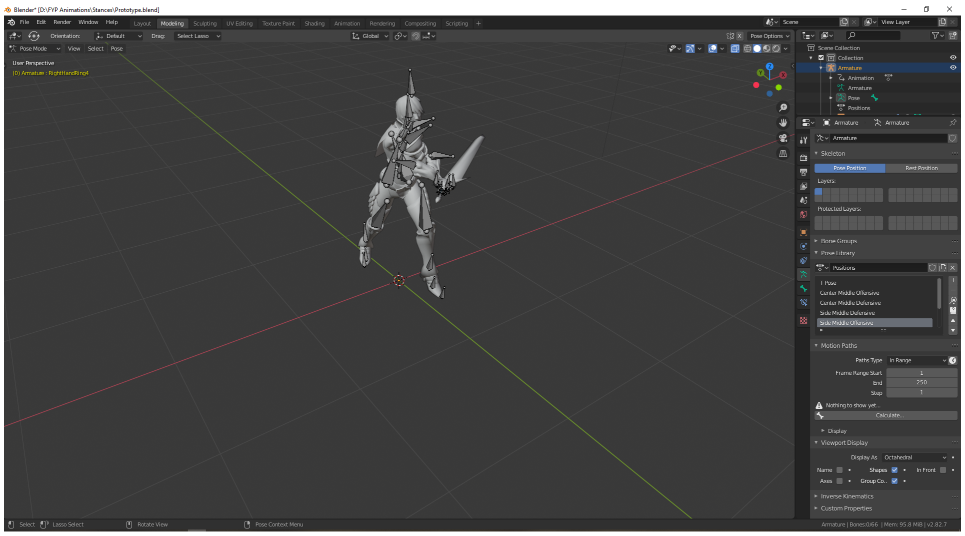


Figure 32: Sword position middle left, offensive intent.

9.5 Custom Animations

Unfortunately, this is where progress on the project is currently, as explained in the conclusion. I was unable to find premade animations that I needed, but I do understand the theory of animation between armatures, bones, and key frames thanks to Computer Graphics 2. I tried to arrange for a tutoring situation when the Covid-19 pandemic shut down the day to day operations of the University of Limerick. I had to self instruct myself on how to use blender, what to avoid, what I need to do to make it work properly in UE4, and how to reduce my workload. The list I had for animations that needed to be created:

- 9 x Footwork
- 8 x Intent changes
- 12 x Changing sword position
- 8 x Attacks
- 8 x Blocks
- 2 x Out of lines (technique resolution on a sword to sword collision)

10 How to Play

The current delivered object is obviously incomplete as mentioned before. Currently it is playable, but because the animations needed to be created and then implemented, there is no way to exit combat or the game besides closing.

10.1 Open World

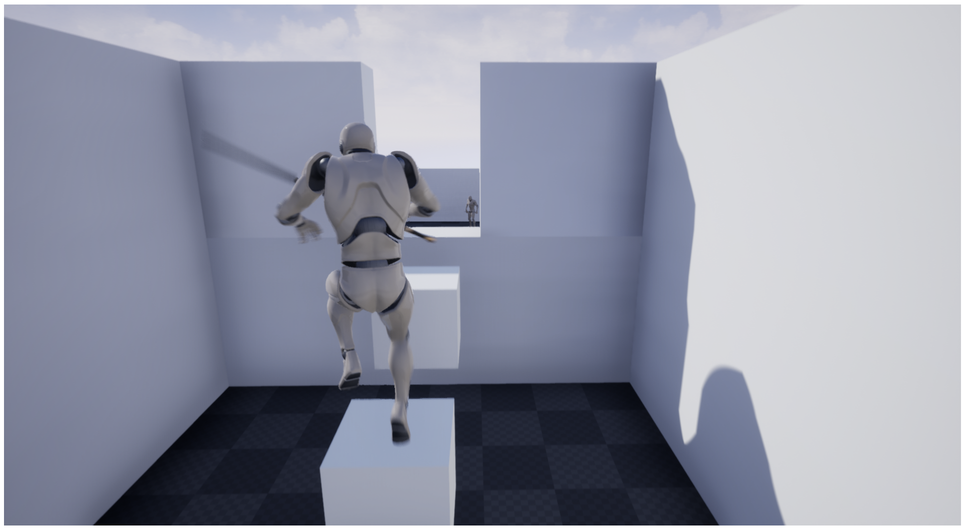


Figure 33: Screenshot of the player character jumping.

W - Forwards
A - Strafe Left
S - Backwards
D - Strafe Right
Mouse - Camera Look
Space - Jump

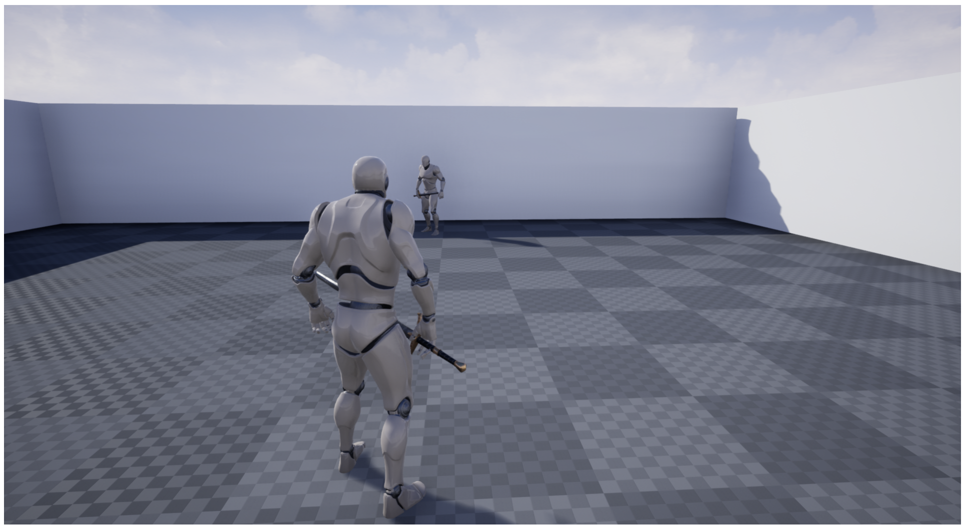


Figure 34: The maximum range at before the enemy AI locks on, and the player can lock on to the enemy.

Mouse Right Button - Lock on.

10.1.1 Locked On

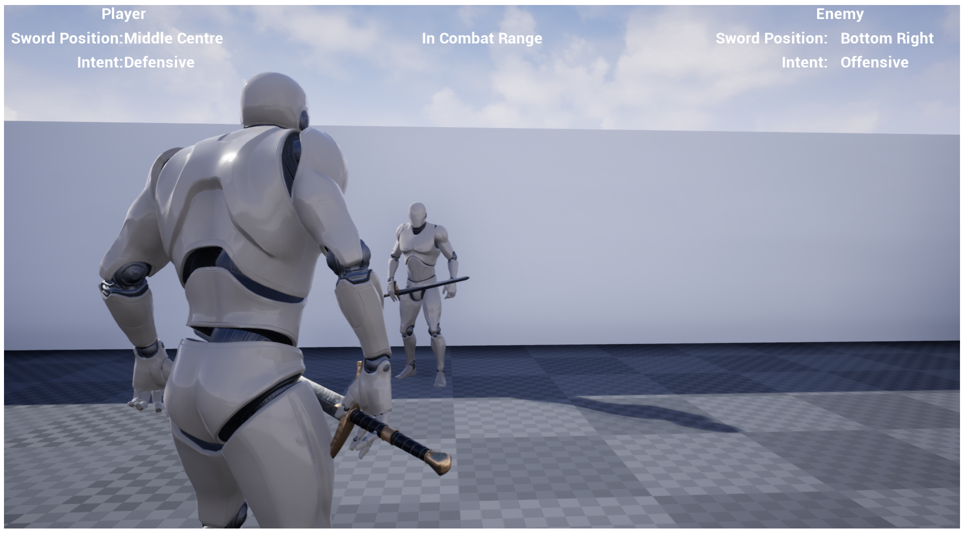


Figure 35: The player is now locked onto the enemy AI.

W - Towards Enemy
A - Circle Left
S - Backwards away from Enemy
D - Circle Right
Mouse - Direction determines word position. No input keeps the current position. Move mouse in an up and left motion to move the sword position to top left. From there, the player can immediately drag the mouse directly right, and the game will recognise that the player wants the sword in the middle right position.
Mouse Wheel Up - Set intent to offensive.
Mouse Wheel Down - Set intent to defensive.
Jump - Disabled
Mouse Right Button - Disabled

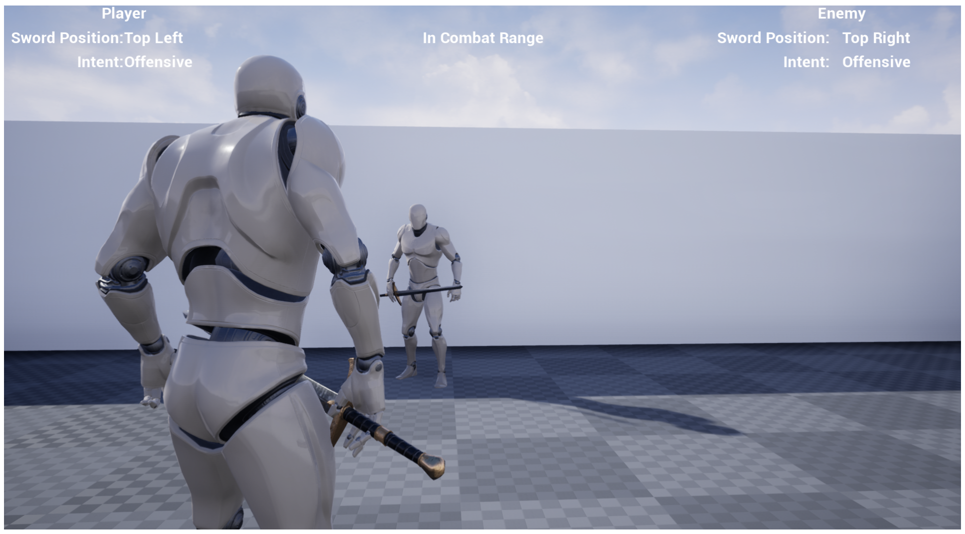
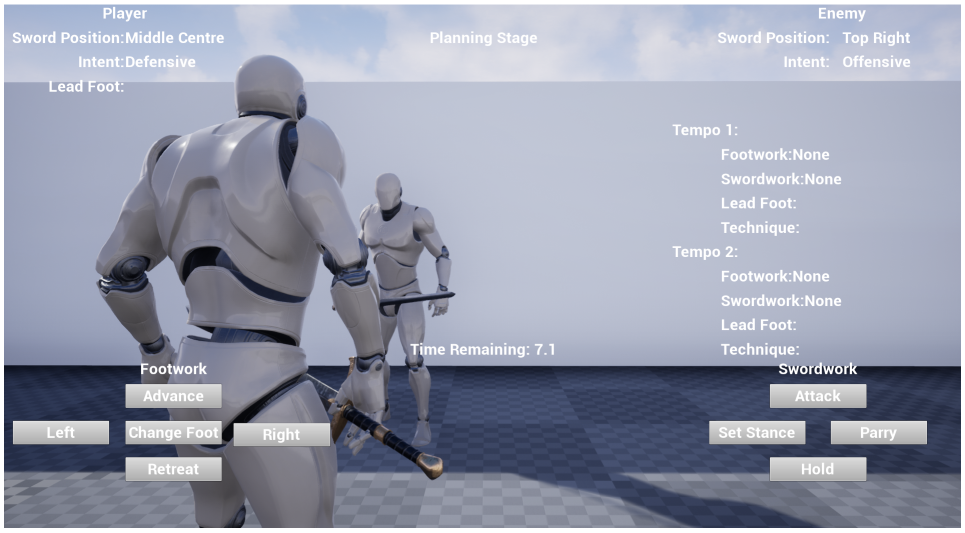


Figure 36: The player in range to start the turn based combat.

E or Mouse Left Button - Start combat (move into planning stage of turn based combat loop)

10.2 Planning



Player
Sword Position:Middle Centre
Intent:Defensive
Lead Foot:

Planning Stage

Enemy
Sword Position: Top Right
Intent: Offensive

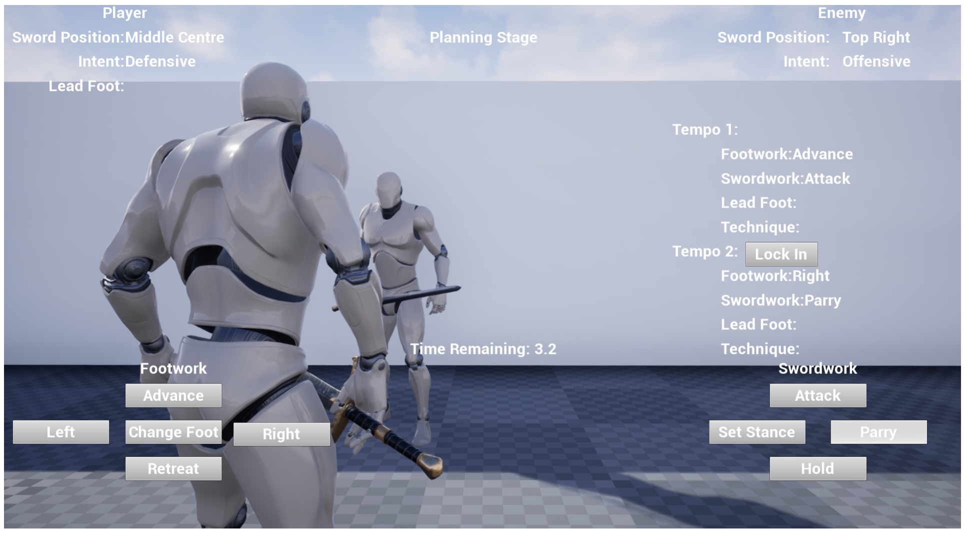
Tempo 1:
Footwork:None
Swordwork:None
Lead Foot:
Technique:
Tempo 2:
Footwork:None
Swordwork:None
Lead Foot:
Technique:

Time Remaining: 7.1

Figure 37: The player in range to start the turn based combat.

Switching from planning state of the turn based system, to the locked-on free world state is where the input drastically changes. For the turn base system, all previous controls are disabled, and will be enabled when the state switches back the locked-on free world state. During the turn base system, the player uses the mouse to select what options the player wants to execute.

The player must select one movement and one swordwork action to confirm to start selecting the second set. The change button will control what the end foot forward is, and technique is decided once the action is locked in. Unfortunately this part is currently not yet implemented. The player can keep on changing footwork and or swordwork until they hit the ”Lock In” button.



Player
Sword Position:Middle Centre
Intent:Defensive
Lead Foot:

Planning Stage

Enemy
Sword Position: Top Right
Intent: Offensive

Tempo 1:
Footwork:Advance
Swordwork:Attack
Lead Foot:
Technique:
Tempo 2:
Footwork:Right
Swordwork:Parry
Lead Foot:
Technique:

Time Remaining: 3.2

Figure 38: The player in range to start the turn based combat.

Here the player has until the timer runs down to zero, or click the "Lock IN" button next to "Tempo 2:" to commit the selected actions and values to the queue, and then immediately begin the execution phase.

10.3 Execution

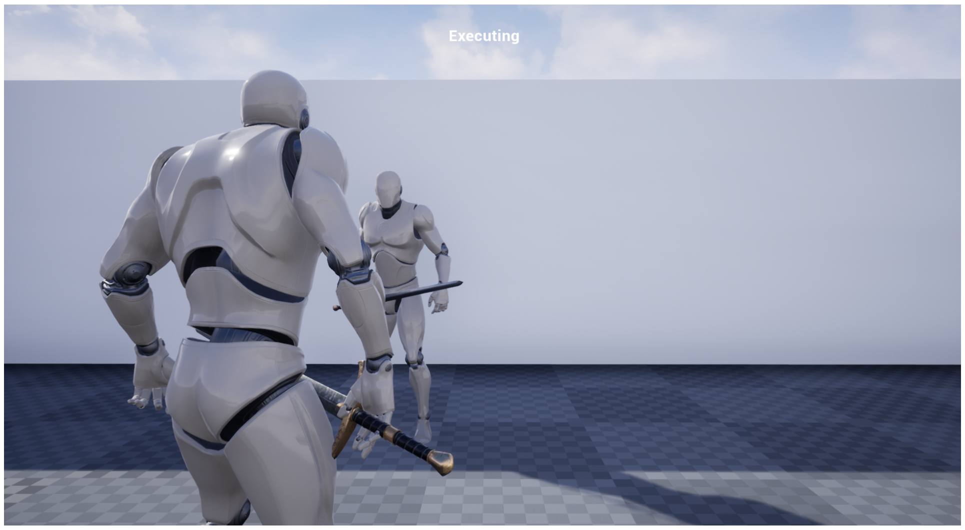


Figure 39: The player and opponent's moves executing.

Minimum UI, no input, each set of actions takes 1 sec, so this stage only lasts a maximum of two seconds. Once the animations, play out, assuming that both characters are alive and still in range, the game returns to the planning stage. If one player dies or if characters back too far apart, combat is broken and the game returns to the locked-on state, with the mouse controlling the sword position and intent, and WASD moving the character. If the player backs up enough, it ends the locked on state, and the player is an a free roam state, able to jump, use the camera, wand move as is common in video games.

11 Future Plans

11.1 Implement Custom Animations

Once the animations are completed, they need to be imported into Unreal Engine 4 with specific steps, such as setting which axis forward, unit scale, and correct file format. Once the animations are correctly imported, then would be the next massive milestone of setting up the correct calls to the correct animations (as it would likely be a separate upper body and lower body animation).

11.2 Anim-Notify

Anim-notify will be very handy to report various stages of the animation to whichever blueprint needs the information. This can inform the player that there sword didn't hit anything at the spot of maximum extension, as set up by the anim-notify on that specific frame.

11.3 Collision Resolution

The next step would be setting up a conflict collision channel customised for the sword skeletal mesh. This would allow me to distinct collisions from being against an opponent's sword, or the body (skeletal mesh) of the character. If the sword connects with the opponent's body, it makes sense to play a death animation, or preferably a mesh-slicer, as well as have the game logic to either end the game or reset to start again. If the sword collides with another sword, well that is where the technique comes in that was defined earlier.

12 Conclusion

Unfortunately due to the outbreak of Covid-19, there was a huge disruption to the work on the prototype. I was recalled to the US, and dealt with the arrangements of ending my 4 year residency in Ireland, while still turning in college work when due.

Given the current state of the project, the result would have to be inconclusive. However, the current state shows that the concept is possible, but would need a team (including animators) to properly finish prototyping and refining. I do believe that this is proof that there are still new ways to design digital melee combat systems that are worth exploring. Obviously with any new software, it would take time, changes, and love to make into a polished product. I do think this idea has good bones and worth further investigation.

13 Bibliography

References

[1] C Matthias Hüls, Ingo Petri, and Helmut Föll. “Absolute Dating of Early Iron Objects from the Ancient Orient: Radiocarbon Dating of Luristan Iron Mask Swords”. In: *Radiocarbon* 61.5 (2019), pp. 1229–1238. DOI: `10.1017/RDC.2019.13`.

[2] Fiore dei Liberi. *il Fior di Battaglia, MS Ludwig XV 13*. `http://wiktenauer.com/images/e/eb/Getty_MS_Ludwig_XV_13_Scans_with_English_Translation.pdf`. Online: Accessed 10-September-2019, Translated: ”Chidester, Michael and Durban, Eleonora and Easton, Matt and Hatcher, Colin and Mellow, Tracy”, Original Manuscript: ”J. Paul Getty Museum”. 1404.

[3] Fiore dei Liberi. *il Fior di Battaglia, MS Ludwig XV 13*. `https://www.aemma.org/onlineResources/liberi/wildRose/fiore.html`. Online: Accessed 15-September-2019, Translated: ”Michelini, Hermes”, Presented By: ”Knights of the Wild Rose”. 1404.

[4] London Fencing Club. *Priority (Right of Way) in foil fencing*. `https://www.londonfencingclub.co.uk/news/106-priority-right-of-way-in-foil-fencing`. [Online; accessed 27-April-2020]. 2012.

[5] Mixamo. *Great Sword Pack*. `https://www.mixamo.com/#/?page=1&query=great+sword`. [Online; accessed 02-January-2020]. Unknown.

[6] Ken Mondschein. “On the Art of Fighting: A Humanist Translation of Fiore dei Liberi’s Flower of Battle Owned by Leonello D’Este”. In: *Acta Periodica Duellatorum* 6 (June 2018), pp. 99–135.

[7] Nabi Studios and Hampa. *Toribash*. `https://www.toribash.com`. [Online; accessed 19-September-2019]. 2019.

[8] Aldo Nadi. *The Living Sword*. Sunrise, FL: Laureate Pr, 1995.

[9] Toshishiro Obata. *Shinkendo Tameshigiri, Samurai Swordsmanship & Test-Cutting*. San Gabriel, California: International Shinkendo Federation, 2005.

[10] Toshishiro Obata. *Shinkendo, Japanese Swordsmanship*. San Gabriel, California: International Shinkendo Federation, 1999.

[11] Power, Guy. *A Brief History of Toyama Ryu*. `https://www.smaa-hq.com/articles/article/a-brief-history-of-toyama-ryu`. [Online; accessed 2-February-2020]. 1998.

[12] Unreal Engine. *Game Mode*. https://docs.unrealengine.com/en-US/Gameplay/Framework/GameMode/index.html. [Online; accessed 26-January-2020]. 2004.

[13] Unreal Engine Marketplace. *Free Fantasy Weapon Sample Pack*. https://https://www.unrealengine.com/marketplace/en-US/product/e4494c76c3b348aba7ef9b263a6dd496. [Online; accessed 03-November-2019]. 2018.

[14] Unreal Engine Marketplace. *UE4 Mannequin: Mobile*. https://www.unrealengine.com/marketplace/en-US/product/ue4-mannequin-mobile. [Online; accessed 03-November-2019]. 2015.

[15] Wikipedia contributors. *Fighting Game*. https://en.wikipedia.org/wiki/Fighting_game. [Online; accessed 02-September-2019]. 2019.

[16] Wikipedia contributors. *Fiore dei Liberi*. https://en.wikipedia.org/wiki/Fiore_dei_Liberi. [Online; accessed 30-September-2019]. 2019.

[17] Wikipedia contributors. *God Hand*. https://en.wikipedia.org/wiki/God_Hand. [Online; accessed 22-September-2019]. 2019.

[18] Wikipedia contributors. *Metal Gear Rising: Revengeance*. https://en.wikipedia.org/wiki/Metal_Gear_Rising:_Revengeance. [Online; accessed 30-September-2019]. 2019.

[19] Wikipedia contributors. *Soulcalibur (video game)*. https://en.wikipedia.org/wiki/Soulcalibur_(video_game). [Online; accessed 30-September-2019]. 2019.

[20] Wikipedia contributors. *Warrior (arcade game)*. https://en.wikipedia.org/wiki/Warrior_(arcade_game). [Online; accessed 20-September-2019]. 2019.