

WEEK 1: Idea Generation

Weekly objective: Brainstorm ideas, define the core game mechanics.

Research Sources:

- [*The Making of Divinity: Original Sin 2*](#)
- [*How Do You Improve Turn Based Combat?*](#)
- [*Game Combat Design, Mechanics and Systems Guide*](#)

Research Analysis:

One of the biggest complaints about turn-based combat is the speed: players always feel like it is too slow. To help speed it up, one of the solutions could be to think about which elements can happen together or make the actions more interactive by mixing a mini game or a Quick Time Event (QTE), rewarding the player with boost on movement or attack.



Quick Time Event functionality used during combat (Clair Obscur: Expedition 33, 2025)

During a turn-based combat system, the player spends a fair amount of time navigating the menus, constantly looking for the best skill needed to damage a specific enemy. That time spent interacting with the UI must feel fun, intuitive. One of the greatest examples in the industry is the Persona franchise.



User interface in Persona 5 (2016)

Because we play a turn-based game, there is a specific play order set at the beginning of the combat. If we do not try to break what we call the turn economy, the gameplay will become more and more boring. If we look at Child of Light (2014), their input on this was to add a timeline: to perform an action, the player icon has to be located on a specific area of the timeline, and some bad decisions or special attacks from enemies can push him back further down the timeline.



The turn economy broke down by a timeline in Child of Light (2014)

Finally, one of the aspects that would make a turn based combat game feeling redundant is the grind feeling: fighting for fighting can be pleasant in Street Fighter VI or Tekken 8, but when we think about games that are cited in our original brief or art bible (Baldur's Gate 3, Clair Obscur: Expedition 33), there are multiple rewards regarding the narrative in the game.

Because my specialism is narrative design, I am going to think in what ways I could imagine the combat as storytelling: could it have different endings? Events during combat?

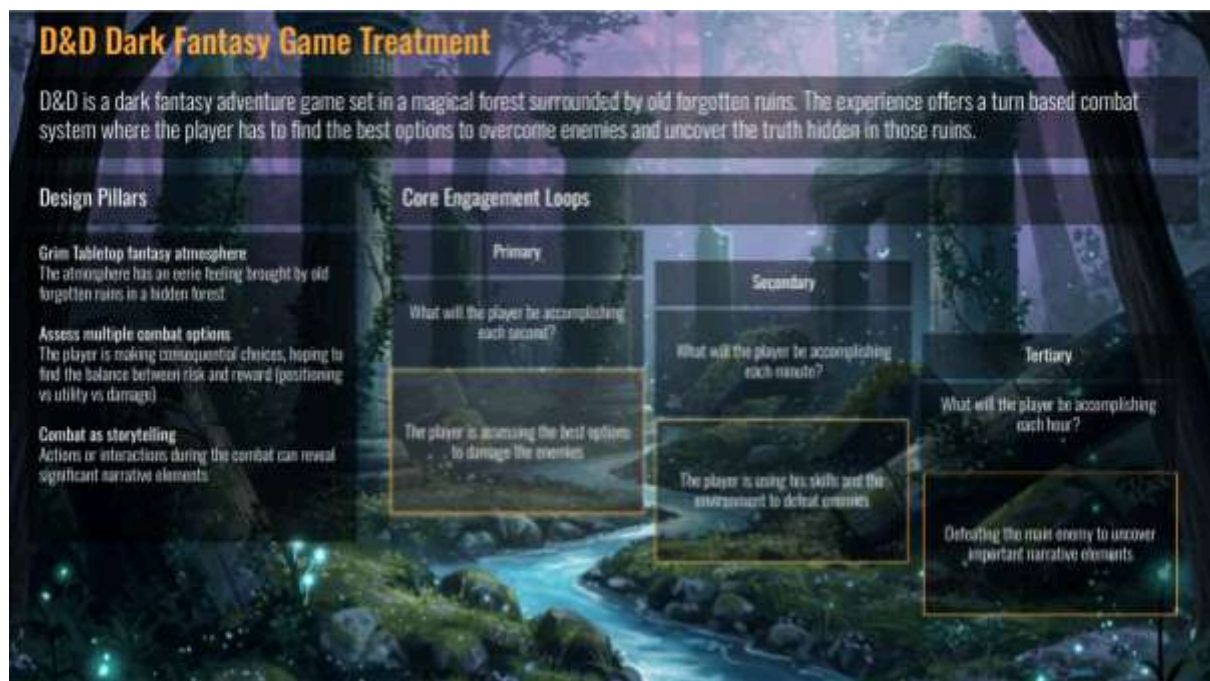
Weekly Summary:

On this project I am going to work in collaboration with another game designer, Marie Libeau, with a strong emphasis on UI and UX, and I will refer to her as Marie.

The first week in a team project feels always a bit overwhelming as we are just meeting each other, and yet we have to make crucial decisions about the game we want to create. It was a very short stretch between the start of the project and a first pitch presentation, barely 2 days and a half.

Beside my research time on turn-based combat highlighted above, we had to define as a team the Dungeon and Dragon feel, analyse the references from the original brief and where we could go from there.

I started quite early brainstorming ideas regarding the narrative, the design pillars and mechanics. Even though it is not definitive and we might adjust those during the development, I think it is essential to have a starting point and unlike other team projects I made sure the game treatment has been read by everyone.

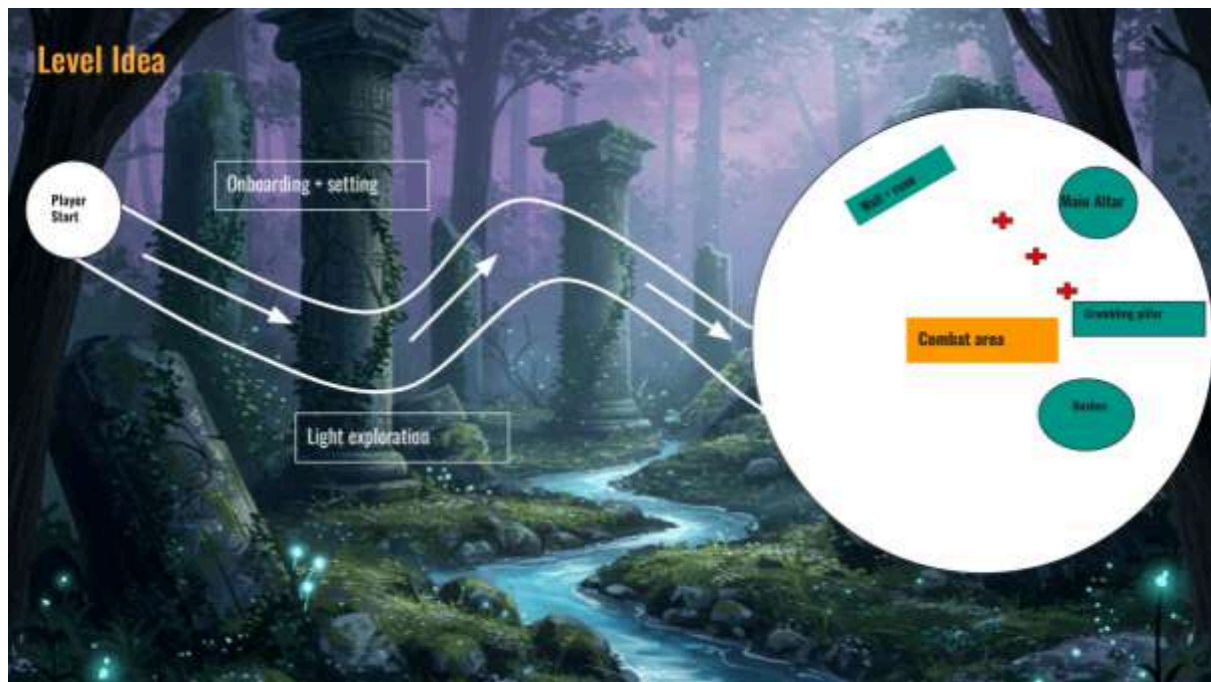


Here is my first iteration of the game treatment, as shown during the pitch presentation on Friday 28/11/25

In this image we can see most of my work summarized within the elevator pitch, the 3 design pillars, and a very broad breakdown of the core engagement loops.

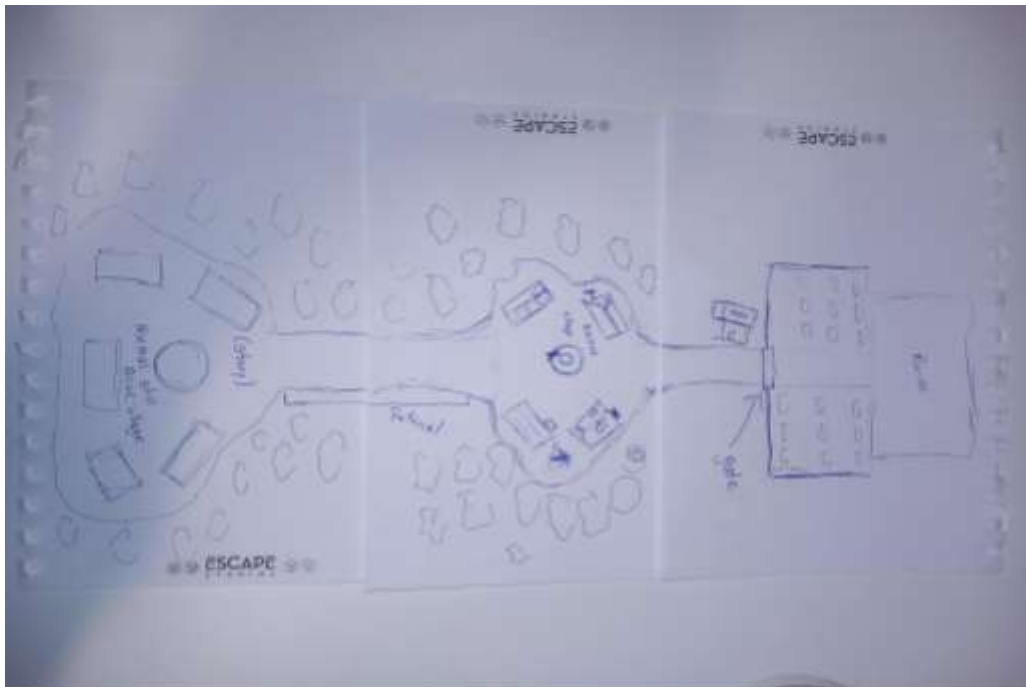
I have been working on the main narrative plot, alongside the first draft of the level layout: my main concern was to keep my teammates from getting out of scope easily. It is quite often early in the process we think too big: "we should have a village, with people walking, then a forest, then another area, then another one". As I keep telling them but with no success (at least until the feedback given during pitch presentation), small done well.

As an example of the meetings we had, here is my first layout on the level and how it could work:



First iteration on the level layout

And here is a second layout discussed in a team meeting:



Second iteration of the level layout

As we can see multiple areas have been added such as a village, a central location and a cemetery before the ruins. We have been strongly advised to reduce the scope here and we might go back to the original layout.

Finally, most of my early days brainstorming the project was spent thinking about what kind of turn-based combat system we wanted for our game, and here, clearly, two options were on the table: a real Dungeon & Dragon strategic to down view battle system with a very strong recent example, Baldur's Gate 3, and a slightly more classic turn-based system well defined with the Final Fantasy franchise but recently brought back to life with Clair Obscur: Expedition 33.

I originally intended to get closer to a more strategic approach, where I could have the player interact with the environment in combat, using crumbling pillars, bushes. While I was discussing with my teammates and listening to their input I was feeling that maybe I was heading in the wrong direction. The combat as shown in the image below is asking too many skills that I probably don't have as narrative designer, within the timeframe required for the project. And the feedback given by my tutor after the pitch presentation comforted me in my doubts; after discussing the matter with my team, we finally decided what we wanted for our game.



Combat in Baldur's Gate 3 (2023)



Battle system in Clair Obscur: Expedition 33 (2025)

Weekly Reflection:

This first week was barely like an introduction week with the team, and having the core ideas about the game we want to make. I have been getting a lot of information regarding combat design, as it is not my specialism.

I tried to specify design pillars and generate what the mechanics could be. As we are not going for a Baldur's Gate oriented combat system, I may have to break down these mechanics differently. Even if I am tempted to start implementing basic mechanics already in Unreal, I want to make sure that the design of the game is as precise as possible. For now, I would say I am on track.

For next week, I am willing to have a list of the abilities and skills player and enemy would use, and start implementing the combat in game.

WEEK 2: Polishing core ideas

Weekly objective: Adjust mechanics after feedback, defining characters skillset, start implementing on Unreal Engine 5.

Research Sources:

- [*What Makes a Good Combat System?*](#)
- [*Who Gets to be Awesome in Games?*](#)
- [*How Level Design Can Tell a Story*](#)

Research Analysis:

To improve my knowledge in combat design, I keep looking at ideas or studies that will help me make a better game. Something I have never done before is thinking of combat as frames of animation. Every animation can be split into three phases:

- Anticipation (wind up to the attack)
- Contact (when the attack lands and does damage)
- Recovery (character is going back to a neutral state)

The anticipation phase dictates the speed of the attack as shown in the picture below:



But combat is not only about attacking, but also defending: dodge and parry are the usual actions available to the player. Dodging uses stamina and changes the position (not really effective in a turn based combat), parrying allows the player to perform a counter attack most of the time, and it can feel overpowered very easily.

To fix this, one of the solutions could be to make the parry a bit more risky; or I could make the punishment bigger for missing a parry. Also could the number of parries be limited ?

Going back to three phases of animation when attacking, the frame time is likely to look like this: huge anticipation, quick strike then hefty recovery.

Our tutors had been sharing with me resources, very helpful pieces of research as we are still in our preproduction phase.

One of them is a video going through the aspect of fantasy in video games: who gets to be awesome? As a combat oriented game, it is an important question I should ask myself. Should I challenge the player to earn the fantasy, through different combats and master some abilities? Or should I give the fantasy? Considering the narrative of our game, giving the player abilities and power at the start of the adventure just feels right for our character, a knight on a mission to kill a monster.

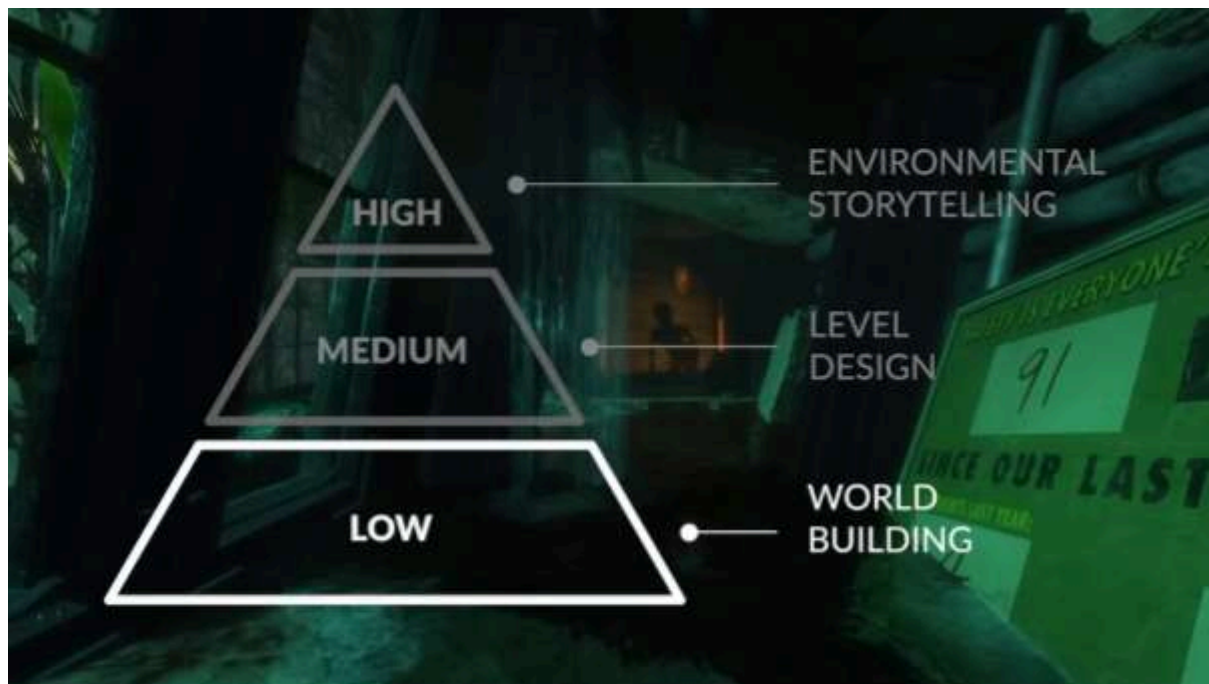
There would be others to increase or dilute the challenge to master some skills, such as the will to provide options for the player. This could be done through the accessibility settings (Forza Horizon with the brakes line switch). Bayonetta also offers a rather rewarding challenge: smashing the action button will have the character performing a whole range of efficient combos. But once again, this aspect of the game can be changed in the settings menu and creates a harder challenge, with a very high ceiling skillset.

Ori and the Will of Wips have a different way to reward the player: the game gives upgrades as long as we progress in the game. This is what we call "layer on complexity"; Ori starts with a simple jump but by the end he has a whole set of skills, attacks, moves, all introduced to the player and mastered along the journey.

Another resource shared with us students as we are in the process of blocking out our level is about how level design can tell a story.

We can divide level design into three different sections: environmental storytelling, actual level design, and finally worldbuilding. There is no specific

order to these, but they should be able to respond to each other in any circumstances.



Pyramidal structure of Level Design

On top of that, the level design must be made according to a range of emotions we want to give to the player. As for example, in Mass Effect 2, when we arrive on the planet we do get that feeling of hope and confidence because of the warm lighting, the area is wide and open, plus the player is backed by a whole army convoy behind him.

Hollows	Krogan Alliance	Hope
Pre-Crash	Calm Before the Storm	Confidence
Post-Crash	Survival	Chaos
Catacombs	History of the Krogan	Mystery
City Reveal	Krogan's Mighty Empire	Awe
City	Race to the Finish	Urgency
Reaper Fight	Final Push	Triumph

Level Design structure applied to emotional flow (Mass Effect, 2010)

That emotional flow of the level design can also be matched with an intensity chart, or curve. As shown below, each section of the level is meant to increase the feeling of stress, or offer a momentary relief. Ideally, this would be tested and adjusted after collecting feedback from playtesters. The catacombs section originally contained aliens or big spiders to kill and the intensity experienced by the player was too high. To match the original intention of giving a moment of peace after the post-crash stress, developers deleted all enemies from the catacombs



Intensity chart in Mass Effect 2 (2010).

A great example of efficient level design is the video game Journey (2010). Without any words, dialogues, barely nothing, it manages to bring feelings to the player, to show him the direction. It is something we might want to think about when building our game, particularly during the block out process: we have to find the right environmental tricks to convey the necessary emotion.



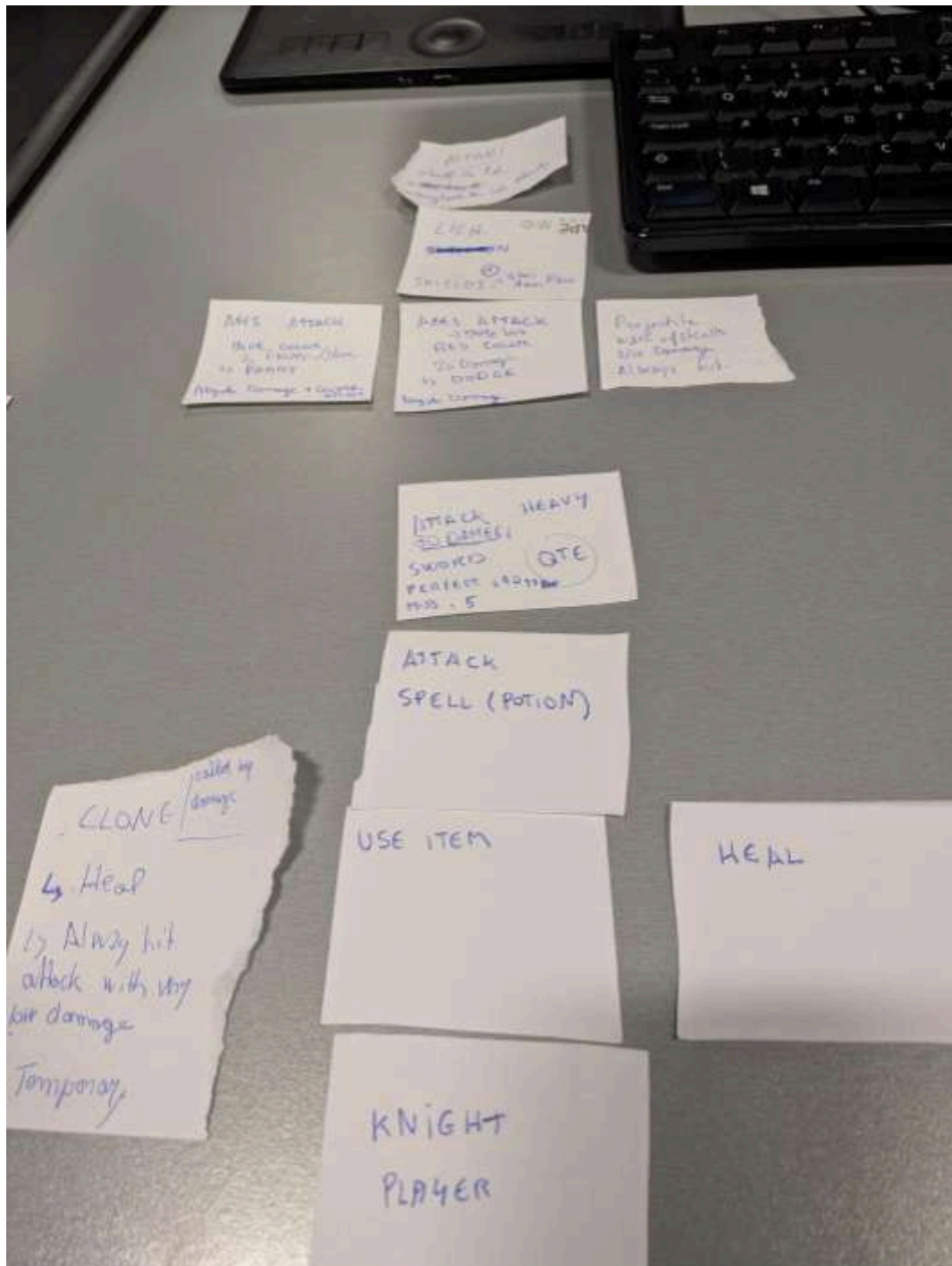
When level design needs no words (Journey, 2012)

Weekly Summary:

Paper prototype , Character/Enemy design, trying to get everyone on board with the agile methodology, research, work with animators. Issue again with level size

After the first week and finally converging for a more classic turn-based combat experience, it was time to readjust some mechanic ideas, and try to break them down a bit more specifically as asked by my tutor during the pitch presentation. When it comes to that kind of technical system, I need to put them on paper and the best solution we can have as game designers is to make a paper prototype, literally think about the game as a board game. It also brings me back to the original advice from tutors when we started that module: “design a more simple system inspired by the tabletop game, focus on the combat elements in particular”.

We can see in the image below the first iteration of the paper prototype; it does not have to be fancy, just functional so we can see what works or not and bring new ideas.



Playing a prototype version of the game

A few ideas came up with this, the main one is the possibility for the player to add a second player to the party, temporarily. This ghost clone of the player (he has no name for now), will be available to use for the player once he has his supercharge bar complete (filling up with any damage done to the enemy). Some of the new ideas are now being updated in the Game Design Document.

D&D Mechanic Ideas				
Core Player Character Actions				
1 Physical Action: Sword attack 1 Skill Action: Cast a spell (fire elemental attack) 1 Use Item Action: Health potion 1 Supercharge Action: Temporary release a ghost clone of yourself into the combat				
Core Ghost Clone Actions				
Each damage done to the enemy fills up a supercharge bar; when full, the supercharge is ready to be used. It takes as much MP as the sword.				
The supercharge allows the player to release a double ghost of himself for a very short period of time. The clone comes with two abilities:				
Heal the player to the max				
Or				
Special slashing attack that deals extra damage				
Action Type	Name	Energy Damage	MP Cost	Function
Physical	Sword attack	15		Standard sword attack
Skill	Cast Attack Spell	25	30	Throw a fireball, deals damage or triggers prop interaction
Item use	Use Potion			Restore health (15MP)
Supercharge	Send ghost to party	40	Empty the supercharge bar	Ghost clone is slanking the enemy, clones hit
Heal	Single healing		Empty the supercharge bar	Ghost clone is healing the player to max

Breaking down the primary mechanic in our game.

Additionally I exchanged many ideas with animators and riggers on what we are going to need for the game. As said in my initial section of this production log week 2, I have a feeling that I will have to work very closely with them to make the best of the combat.

Finally it is very complicated to get everyone on board with the agile methodology: some of us have never used it before and they do not see the need of having a meeting daily. I am doing my best to explain this to them; same with working on Unreal with a revision control system.

Some issues have shown up regarding the level size of the layout as some of us do not want to listen to the feedback given by tutors but something similar happened last year in my team project. Hopefully this is just an early adjustment and we will get better at receiving feedback.

Weekly Reflection:

It is time for me to start implementing the basics of my combat system. When I look at my week I think I may have lost some time to have the MVP ready but this is not the case.

It is essential to me and for the game that I am careful with the proper design aspect of my project: it needs to be well polished, and I want this game to be fun to play as a board game. This is why I will keep paper prototyping this game all along the production process, and receive as much feedback as possible.

The research has been taking me a lot of time too: once again, being out of my specialism is as much exciting than time consuming. This research work will pay off later, and learning something new also.

To summarize, this turn-based combat is making more sense to me now, the direction where I want to go and the feeling I want to give to the player is clearer. I am rather satisfied with the new mechanic ideas that make the primary mechanic loop: it has been broken down a bit more specifically and I can start implementing them.

WEEK 3: Creating the base system for my turn-based combat

Weekly objective: Implementing the basics of the combat system: UI, attacks from player and enemy, but also keep on polishing the mechanics with paper prototyping.

Research Sources:

- [*Understanding Balance in Video Games*](#)
- [*Defending Turn-Based Combat*](#)
- [*5 Core Pillars That Make Turn-Based Combat Fun*](#)
- [*Embracing Motion Matching: Using a Little to Get a Lot | Unreal Fest Bali 2025*](#)

Research Analysis:

I was asked by one of my teammates if there were any cool features or mechanics I was excited to implement, my answer left him speechless: "I will be excited if I managed to design a well balanced combat system".

This is precisely where the paper prototype was taking me and the research I have done this week comforted me on this idea.

First, as mentioned by Keith Burgun (2011), we have to cherish weaknesses. When designing my combat system, weaknesses are as important as powers or spells: when you identify a specific weakness, you can act with a specific character to apply a specific ability. And secondly, balance will be a top priority for my game: I don't want to take any shortcuts with that and I will leave time on my schedule to work on that.

More specifically about turn-based combat, one aspect I would like to emphasize is the "decision making and resource management", highlighted by Hookshot (2024) in his blog Hookshot, Charge Beam, Revive. This is not because the player is not pressing buttons that he is not playing a game. Once again, referring to the Dungeons and Dragons tabletop game: playing a boardgame has your mind filled with options to consider and strategies to execute. It would be great to have this feeling when playing our digital game: the player will have to think about what is the best decision.

The last research I have done this week was more in depth of the essential core pillars that make turn-based combat more fun and less repetitive.

- Players have to think before making a move

When designing the combat, I have been careful with adding abilities and weaknesses. A certain ability may be great in some situations and really bad in others.

- Players must decide when and how to use resources

I want the players to manage a limited amount of resources. As for example, The Eidolon supercharge is not easy to gather for the player, and when he can use it, he might be in a position to use another resource.

- Players should have opportunities to experiment and take risks

That idea feels a bit more complicated to implement as it is very complex to design systems to provide opportunities for the player to experiment. Maybe it can be just an unexpected outcome from a specific action.

- Encounters should have an engaging context

When the narrative comes into play, every decision carries more weight:

“Building up the encounter beforehand, adding unique dialogue during fights, story-driven objectives, and rich lore in the environment ties the combat to the bigger narrative.” (Niki, 2024)

- Encounters must reward players

I want to satisfy the player after defeating the boss: it could be an item, gear but the way I see it for now it would probably be a story revelation.

Weekly Summary:

I created a new version of the paper prototype and we played it with other teammates. The reason I did this is because I wanted to identify what was working or not: having abilities and powers and using them as a card game is very unforgiving.

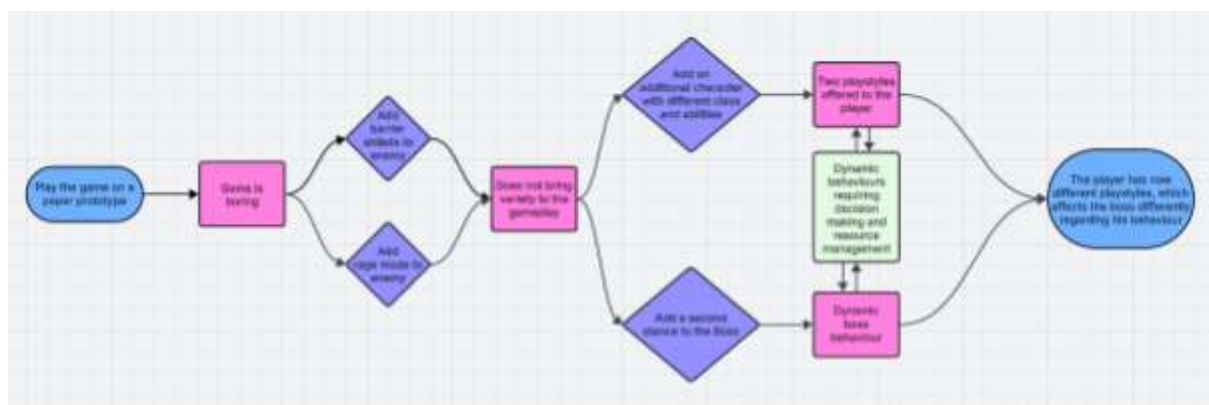
We found out very quickly that our game was not fun at all, boring to play to say the least. My initial reflexion was to add a complexity layer on the enemy, such as the need for the player to take down barrier shields before dealing any HP damage, or having that boss in a rage mode when low on HP.

But this wasn't working too and I decided to do some research and look at more games with a turn-based combat system.



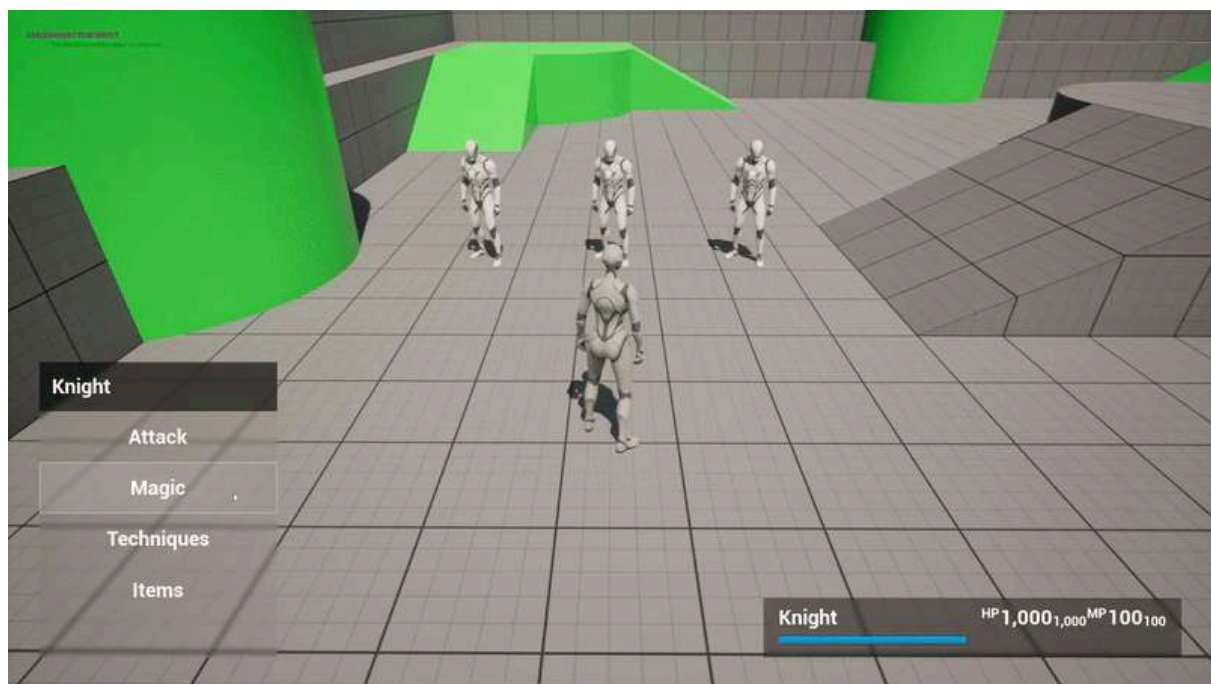
The result of my research was the addition of a second playable character: that brings a whole different playstyle, new gameplay ideas. That second character would be a priest, acting more like a wizard; he can heal himself or the other character, but it can also deal some important damage in specific situations.

The other idea is to give the boss two different stances, which requires different abilities to damage him. At this point I think these two ideas combined bring a whole new dynamic to the combat: the player has to figure out the best way to deal with the Lich King. How can I damage him the most? How to force him to the ground ?



How the paper prototype led my reflexion in building a better combat system

On top of that, I started implementing the very base system of a turn-based combat system with some base animations and UI. If I am thinking ahead there would be a couple of issues I will need to fix pretty early in development: the first one is the Game Mode override. At this very moment, you need to override the game mode from third person to battle mode but this is not the way I want the battle to launch in the game, it would probably have to happen inside a triggerbox, by calling an actor or a combat manager blueprint. The second issue concerns the “select target” action (as shown in the video below): for a more dynamic aspect and if I am willing to give this combat a sensation of speed, it would be beneficial to avoid that step, especially with one and single enemy.



Weekly Reflection:

This week has been quite challenging. Most of the time I did not know where I was headed with that combat system. It looked boring, and was very short on

ideas. But at the end the benefits of the paper prototype start being visible, and it looks like I have now a clear idea of what this game could be.

This week has been challenging as a team too: it is obvious that barely half of the team is really committed to the project and because of that, some of us are quite discouraged. The attendance is really low, online or in class, and most of the conversations have switched to Discord, which was not supported by tutors. The result of this was a very poor presentation during the mentor feedback session, just because no one had been working on it or even thinking about it until Thursday 4pm the day before.

I took this as a big warning.

WEEK 4 : What will make our game fun?

Weekly objective: Before implementing anything relatively specific to our game, the last week of pre-production will be spent on playing the paper prototype. At the moment, the gameplay does not quite work: some attacks are too powerful, the two boss states have no impact or so little on the gameplay.

Research Sources:

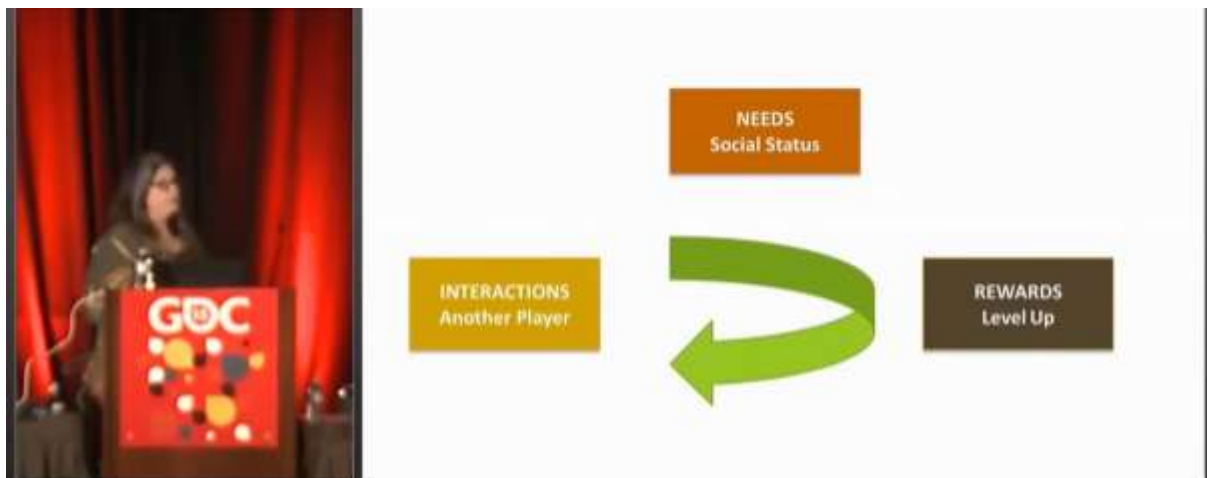
- [Emotional Journey: BioWare's Methods to Bring Narrative into Levels](#)
- [Rules of the Game: Five Tricks of Highly Effective Designers](#)

Research Analysis:

The first talk I watched is quite recent (Rules of the Game: Five Tricks of Highly Effective Designers, 2025) : five experienced designers shared their tricks and gave advice. Two of those caught my attention: first Laralyn McWilliams then Nels Anderson.

Laralyn McWilliams demonstrates the importance of playtesting, observing player's behavior and emotions. When developing Free Realms (Daybreak Game Company, 2009), she noticed players were missing a loop of the

gameplay, they were not levelling. Instead, they were sharing their cool outfits and partying together. The reward for the player was not attractive enough.

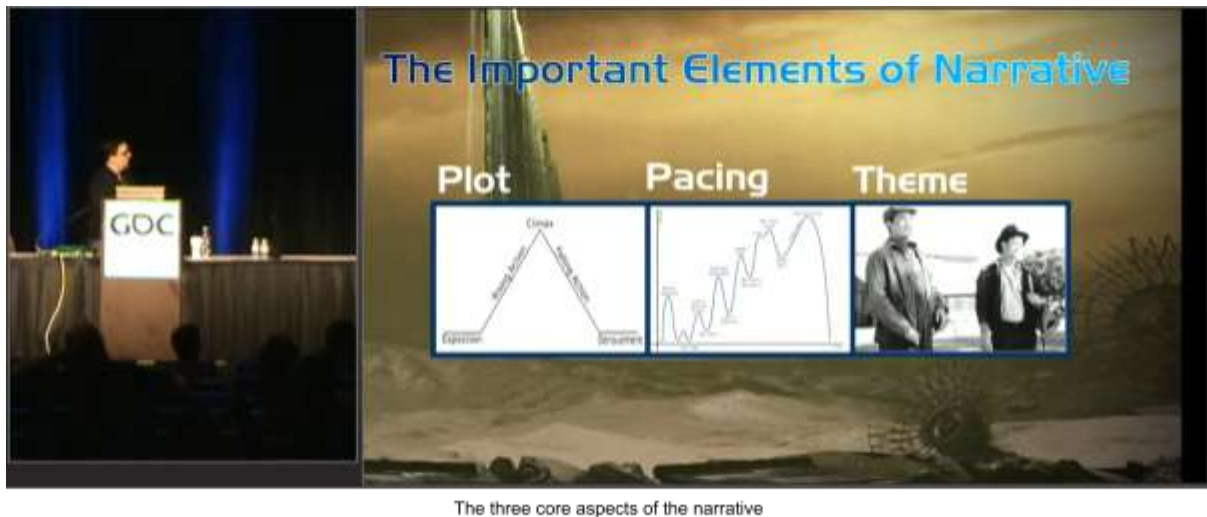


Laralyn McWilliams showing the player's behavior during playtest.

The solution was to reward the player with outfits, skins. If the player loves to do this, use it at your convenience, you have to treat the “player emotions as a core part of your systems design” (Laralyn McWilliams, 2025).

Nels Anderson was game designer on Firewatch (Campo Santo, 2016). After devising the use of the word “fun” in games, preferring the term “compelling”, he came up with the definition of what he calls the “design intent”, and this has a resonance with the paper prototype of our game. We need to have a clear vision of a specific mechanic or moment, how it should feel like and it needs to be tangible and evaluable.

The second talk I watched was given by Dave Feltham, and it is called “Emotional Journey: BioWare's Methods to Bring Narrative into Levels”. As an aspiring narrative game designer, there is a lot of information I can learn from this. The issue the development team had with the narrative was the progression: it felt it was moving forward during the cinematics only.



To fix this issue, Bioware and more precisely Dave Feltham (Level Designer on Mass Effect 2 and Mass Effect 3) had been looking at other games such as Uncharted 2: Among Thieves (Naughty Dog, 2009) or Red Dead Redemption (Rockstar, 2010). They particularly got interested in the way to implement the storytelling through the level.

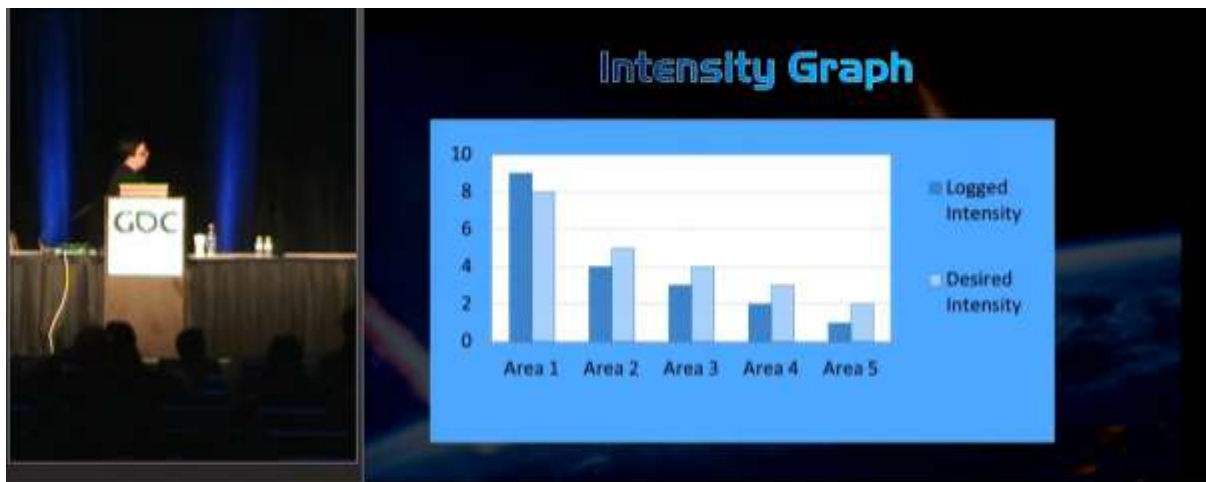
They organised their process in three different steps : Plan, Integrate, Measure. We can see in the picture below how they have a clear idea of themes and emotions for each area: this is important because it would have a significant impact on the gameplay during the integration phase.

Level Area	Theme	Emotion
Hollows	Two Tribes work Together	Surprise, Tension Hope
Pre-Crash Highway	Calm Before Storm	Hope, Anxiety
Post-Crash	Survival	Chaos, Survival
Catacombs	Reveal Krogan Empire	Curiosity, Caution Mystery

Before implementing the storytelling, we plan the emotions on the level layout

Even though our project is reduced to a boss battle, emotions and themes can be applied to a specific mechanic, attack or event during the combat.

During the integration phase, narrative was implemented through combat, level events and psychology.



The Intensity Graph allows the development team to measure and correct player's emotions.

And finally the last is the Measure: how far we are from the design intent? During intensive playtest sessions, we can measure the intensity of different areas with the help of an Intensity Graph. Then, developers can adjust the gameplay through combat, events or psychology to correct the intensity.

Weekly Summary:

There have been a lot of downsides this week. First we did a poor presentation last friday during the mentorship session, being told off quite poorly but fairly about it. We clearly did not spend enough time getting ready for it and to be fair I was ashamed watching it live.

Watching other teams' presentations was rather instructive: we are clearly quite behind, and if we want to come up with a compelling game in May much more will need to be done.

That is the reason why I have kept playing the game on paper prototype again and again, with different people.

Right now there are issues with some attacks that are unused by the player, why? How do I send the information to the player that he needs a combination of two attacks to make the most damage?

I had important feedback from my tutors Gregor and Glyn : what builds? what spends? I have to think about the functionalities, and find another way to resource generation.

Weekly Reflection:

I feel like everything I am doing is pretty poor and nothing seems to work. Probably under pressure when I know some of my fellow game designers have a game ready to be played already and I am still stuck in a paper prototype to find the fun of the game...Hopefully it will pay out at the end but it is hard to see the benefit of it right now.

WEEK 5 : First week of Production

Weekly objective: First week of production means the objective for the week is to be able to play the combat.

Research Sources:

- [Boss Up: Boss Battle Design Fundamentals and Retrospective](#)

Research Analysis:

In this 2018 GDC session, Untame's Itay Keren discusses the various aspects that make a well-designed, effective and memorable boss battle. The first example is referring to is the boss battle in Mega Man (Capcom, 1987).

When we think about boss battle and more particularly boss design, large means power. It has to be bigger than the player character, impressive. In some games the boss can be absurdly way bigger than the player in order to make this combat feel like no other.

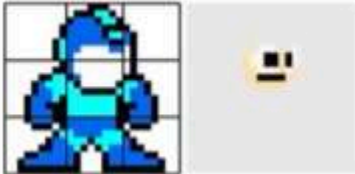
It is very important to bring the unexpected during the combat, surprise with theme, tech, visuals or mechanics. The combat against Yellow Devil in Mega Man (Capcom, 1987) is a great example of how to bring those different particularities together: first we can see in the image below the boss is at least four times the size of Mega Man. Then the developers have been using the tech at their disposal of the time to create this mechanic: when the boss starts shooting at you, he is building another self on the other side of the scene, whereas the original is disappearing. For a brief moment, the player is stuck between these two temporary bosses and needs to avoid projectiles at the same time. All this together generates a peak of difficulty, increases the difficulty curve but most of all it is a memorable and clear attack pattern, and gives place for true mastery.

GDC
GAME DEVELOPERS CONFERENCE

MARCH 19-23, 2018 | EXPO: MARCH 21-23, 2018 #GDC18

Large

Tech in service of design



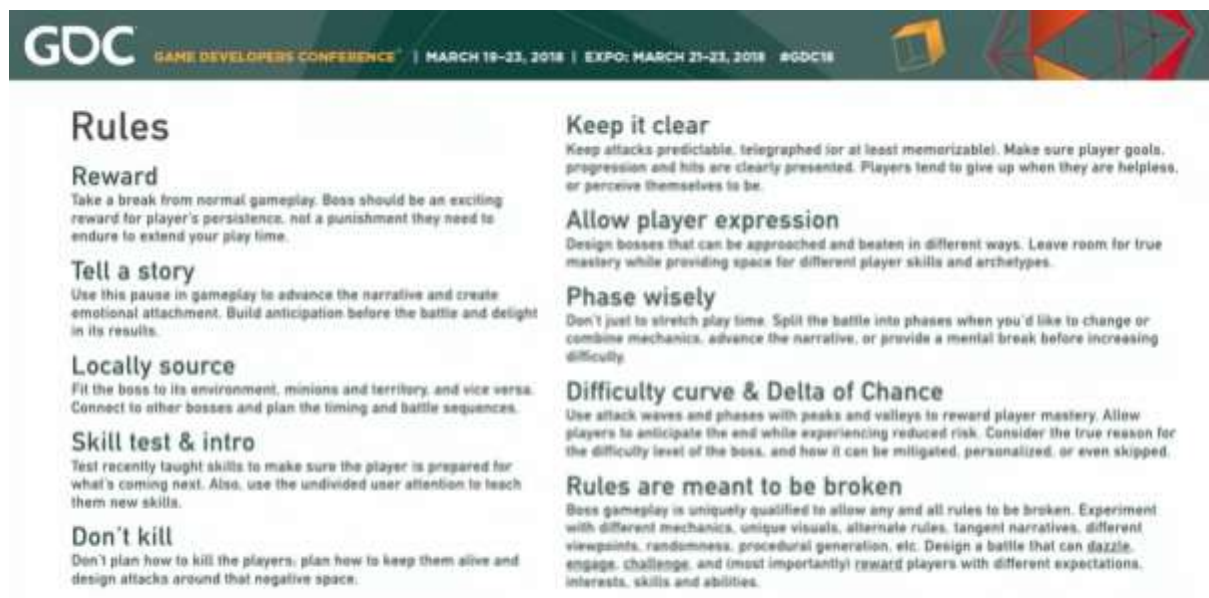
64 foreground sprites
All 8x8 or all 8x16
3 colors (out of 56) + transparent
No more than 8 sprites per scan line



Yellow Devil
Mega Man © 1987 Capcom

Mega Man combat against Yellow Devil.

Itay Keren also discusses the intro and outro of a boss battle. Because we think about adding cinematics at the beginning and end of the battle, this aspect of the talk was quite relevant for our project. An introduction to a battle advances the narrative and builds the rage: it allows the player's emotions to reach that intensity the developers are looking for when designing a boss combat. On the other hand, an outro would give the player a break after an intense battle.



The image shows a banner for the Game Developers Conference (GDC) 2018, held from March 18-23, 2018, with an expo from March 21-23, 2018. Below the banner is a list of rules for boss battles, attributed to Itay Keren. The rules are organized into two columns. The left column includes: 'Rules' (a section header), 'Reward' (take a break from normal gameplay), 'Tell a story' (use this pause in gameplay to advance the narrative), 'Locally source' (fit the boss to its environment), 'Skill test & intro' (test recently taught skills), and 'Don't kill' (don't plan how to kill the players). The right column includes: 'Keep it clear' (keep attacks predictable), 'Allow player expression' (design bosses that can be approached and beaten in different ways), 'Phase wisely' (don't just stretch play time), 'Difficulty curve & Delta of Chance' (use attack waves and phases with peaks and valleys), and 'Rules are meant to be broken' (boss gameplay is uniquely qualified to allow any and all rules to be broken).

GDC GAME DEVELOPERS CONFERENCE | MARCH 18-23, 2018 | EXPO: MARCH 21-23, 2018 #GDC18

Rules

Reward
Take a break from normal gameplay. Boss should be an exciting reward for player's persistence, not a punishment they need to endure to extend your play time.

Tell a story
Use this pause in gameplay to advance the narrative and create emotional attachment. Build anticipation before the battle and delight in its results.

Locally source
Fit the boss to its environment, minions and territory, and vice versa. Connect to other bosses and plan the timing and battle sequences.

Skill test & intro
Test recently taught skills to make sure the player is prepared for what's coming next. Also, use the undivided user attention to teach them new skills.

Don't kill
Don't plan how to kill the players, plan how to keep them alive and design attacks around that negative space.

Keep it clear
Keep attacks predictable, telegraphed (or at least memorizable). Make sure player goals, progression and hits are clearly presented. Players tend to give up when they are helpless, or perceive themselves to be.

Allow player expression
Design bosses that can be approached and beaten in different ways. Leave room for true mastery while providing space for different player skills and archetypes.

Phase wisely
Don't just stretch play time. Split the battle into phases when you'd like to change or combine mechanics, advance the narrative, or provide a mental break before increasing difficulty.

Difficulty curve & Delta of Chance
Use attack waves and phases with peaks and valleys to reward player mastery. Allow players to anticipate the end while experiencing reduced risk. Consider the true reason for the difficulty level of the boss, and how it can be mitigated, personalized, or even skipped.

Rules are meant to be broken
Boss gameplay is uniquely qualified to allow any and all rules to be broken. Experiment with different mechanics, unique visuals, alternate rules, tangent narratives, different viewpoints, randomness, procedural generation, etc. Design a battle that can dazzle, engage, challenge, and (most importantly) reward players with different expectations, interests, skills and abilities.

The main rules for a boss battle, according to Itay Keren.

Additionally, Itay Keren emphasises again on the unexpected and surprising feeling we want to give to the player: during the confrontation, altering the rules works well. The idea is to modify skills that the player has mastered before or during the combat and we can do that because the "boss gameplay is uniquely qualified to allow any and all rules to be broken".

This reminds me of conversations I had with my tutor where he wanted me to think differently about rules, and experiment with different mechanics.

Even though we want to break rules, the plan is always to keep the player, to make a path for him. As I said before, that can come from memorizable attack patterns, but also we want to make sure the player's goals or hits are clearly presented. Even a counter attack mechanic can be used to involve the player again in the combat after taking damage.

Finally, Itay Keren defines what he calls the “Delta of Chance”: this happens when a challenging peak is followed by an anticipated triumph. He gives an example from his own game, where he thinks that the boss fights he designed are too hard. The player hardly reaches that triumph feeling because he is not rewarded when going through this intense fight, and data proves they were giving up before defeating the boss.

Weekly Summary:

The winter holidays have been the opportunity to play the paper prototype with friends and family. Considering the last feedback I had from different tutors at University, I planned those playtests with some questions in mind: What build? What spend? Why are some abilities never used? Why are some always used? I wanted to think in terms of functionalities, and I have been advised to think of another source of generation.

The results of those playtests are:

- Additional light attack (Valeria)
- Skip turn (Valeria & Lucien)
- Armor Points needed for Valeria to parry
- Draining mechanic for Kael



Playing the paper prototype with a friend.

More questions and thoughts came from those sessions:

- Do I need a dice check on Glacial Tomb when Kael is grounded?
- Does the HP gain when using Prayer of Mending too high?
- Kael HP is draining fast
- Soul Sever meter is building slow
- Sundering Blow should deal more damage when Kael is grounded?
Should it be still effective when Kael is floating?

I will take those observations into consideration during the next playtest.

Regarding the digital prototype, I have managed to implement the basic turn system, basic physical and spell attack, the use of MPs when using black magic and damage done to the player. I am still learning the fundamentals of this specific combat system but I am doing this step by step and the logic of the implementation is making sense now.

We are going to hold our first playtest next Thursday: ideally I will have the basic combat implemented. The idea is to have something playable as soon as possible and iterate from there.



Spell ability available for Lucien.

Weekly Reflection:

I am feeling more comfortable with navigating and adding my personal touch to this boss battle. I can't wait to have animations and VFX to play with so it looks a bit more entertaining for the player. Once again, after having the basic functionalities playable I will start adding specific attacks and mechanics resulting from the paper prototype and will playtest and iterate from there.

In conclusion of this sprint we had an online session with our tutors where they gave us some feedback regarding our project. The main issue we still have (as it is not the first time) is the scope of the project.

As an example we have a teammate who spent weeks on the outside of the cathedral whereas the battle scene is inside. This is another example of miscommunication and poor agile scrum methodology. Another example I can give is this monday we were only two people including me during the call and my other teammate was sharing her concern about being left on the side by the other game art people.

WEEK 6 : Implementing basic functionalities before first playtest

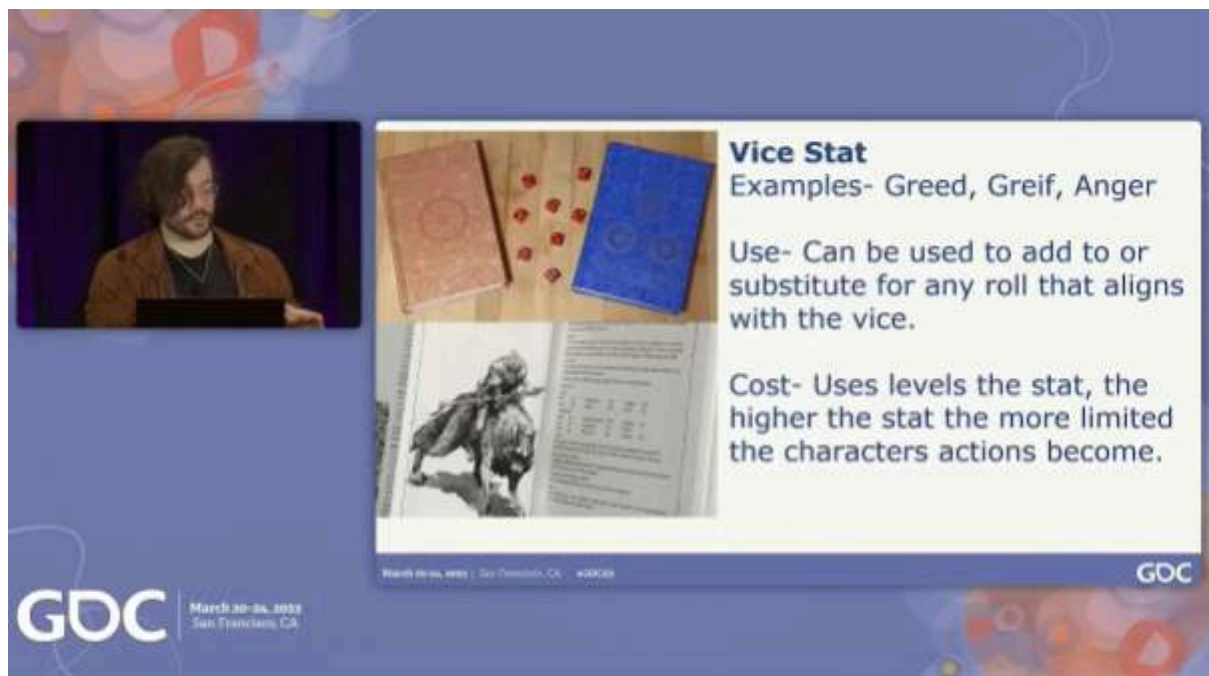
Weekly objective: The main objective for this week is to have a playable vertical slice. That includes basic attacks such as sword attack, spells attack, health gain, being able to defeat the boss, so we can have our game tested for the playtest.

Research Source:

- [The Best and Most 'Stealable' Mechanics from Tabletop RPGs](#)

Research Analysis:

The research I have made this week is based on a talk given at the Game Developer Conference in 2023, given by Evan Hill who worked for Naughty Dog (The Last Of Us Part 2) and Obsidian Entertainment (The Outer Worlds 2). Even though our game is much more combat oriented now I think it is still valuable to not lose the Dungeon & Dragon feeling associated with the module assignment. Particularly when we keep playing the paper prototype every week.



Evan Hill explaining how to use a "greed" check and how it impacts player's actions.

The interesting part of the talk was about the use of resources and stats: he recommends using them all at once. And how you use it naturally creates dynamic characters.

Let's take as an example the "greed" stat: if you have a 4 stat in "greed" and think of doing a bargain to acquire an item you maybe want to use your "charisma" check. If we have a system that would always increase your stats the more you are using them, the "greed" stat of your character would increase and reduce the player action on some specific actions. If the player wanted to share a pie with other players, his character would take a bigger slice than the rest, without even asking the player.

The idea behind this is that you are presenting the player with an easy road of good bonuses, moving towards an alignment , a style or a class that you see as valuable and helpful but they are not limited to that.

The players are constantly given this choice to boost their roles to try and succeed but with that balancing of the cost later, the character impact.

Weekly Summary:

During this week we finally were able to add the cathedral blackout in the scene, which was fundamental for the playtest. It is essential at this stage of the production we have a sense of the dark fantasy atmosphere throughout the world, and as a game designer it gave me an idea of the level design of our game.



Battle scene in the blackout.

Considering the actual digital prototype, I managed to implement different spell attacks, such as thunder, ice, and healing system. On top of that, I added a sword attached to the mannequin, and imported a few animations from Mixamo, instead of the animations from Unreal Engine 5 that was used so far.

Because it was mainly melee and kick attacks, I thought it was necessary to the gamefeel: I wanted the player to feel the game is set in a dark and fantasy world.



Battle scene from the player point of view.

I ran into a few issues this week but the main one was the turn system stopping when Lucien was healing himself. Isbah (Tech Art) has helped me on this. Basically when Lucien is healing himself, the montage was never completed, which causes the turn to not end. The fix was a tick that needed to be set as true in the "Play Montage" node.



I have started using another method of collecting data after the playtest and much more accurately than the feedback form we have been using in first and second year. It goes through a Google Form and the particularity is that we can use linear scale. As you can see in the image below, we submit statements to the player and they can rate from 1 to 5, from Strongly Disagree to Strongly Agree. These are required answers, and then they can explain their responses if they want to.

On top of the that, we use the Net Promoter Score (NPS), 3 fundamental questions:

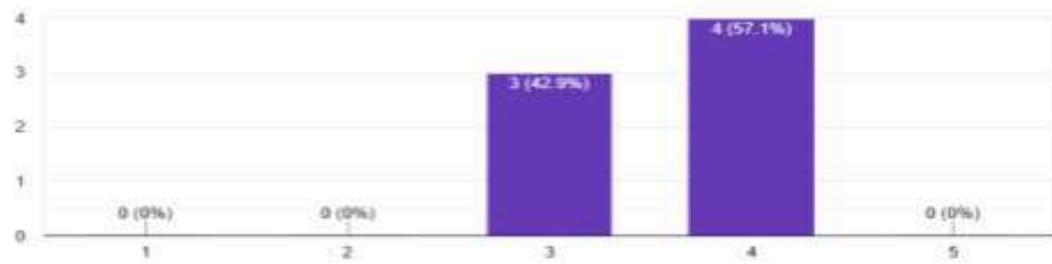
- Was this game fun?
- Would you play it again?
- Would you recommend it to a friend?

These are essential data collected that validate the pertinence of the project.

I found the game fun

 Copy chart

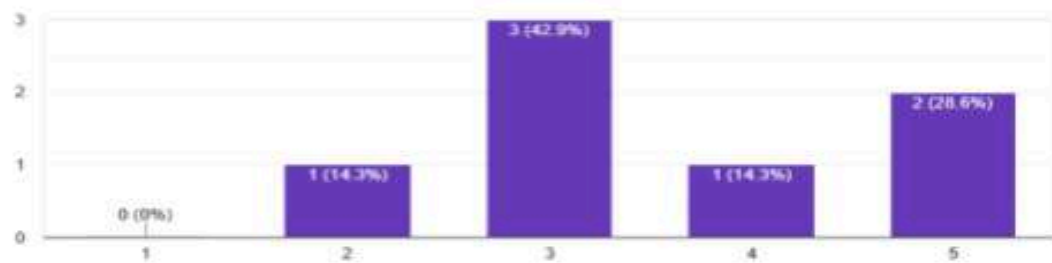
7 responses



I would play this game again

 Copy chart

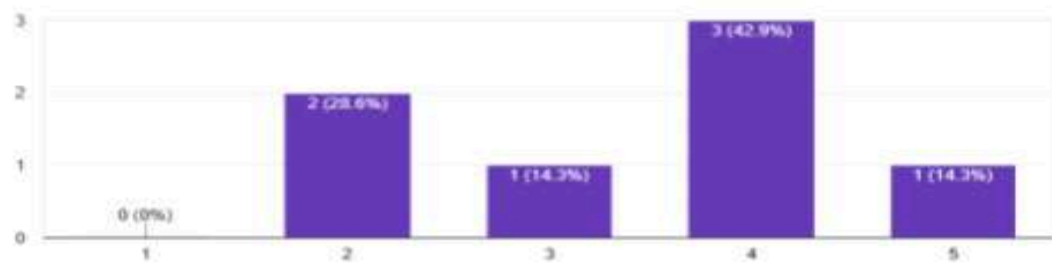
7 responses



I would recommend this game to a friend

 Copy chart

7 responses



Collecting data from our first playtest.

Weekly Reflection:

It has been a very constructive week. I am relieved that Isbah came to help me many times with blueprints, her logic is much appreciated and I finally feel like working in a team, one of the aspects of the job I like the most.

Having the blackout coming in the actual scene feels like we are moving forward, even though the attendance in meetings online or commitment on sprint review/planning is not what I would expect from people ready to work in the industry.

It was also really helpful to have help from other team members to organize the playtest. For the foreseeable future, I will get ready with better questions in order to collect better information, and have a few PC stations ready with the forms, so we can avoid having playtesters queueing to submit their answers.

WEEK 7 : Having a new system to implement attacks

Weekly objective: Regarding the last feedback, the players want a wider variety of attacks. To do so, I need to re-work my system as it is not very suitable for making it easy.

Research Source:

- [*The Right and Wrong Way To Make Turn Based Combat*](#)

Research Analysis:

On the MangaKamen Youtube channel we could find this rather sharp analysis on turn-based combat. Although I would preferably name this video

“Introduction to...”, there are some noticeable aspects that I will highlight in my research.

First, and I think I might have already talked about that in one of my previous logs but it is something that appears a lot when we look at turn-based combat games: the grinding. One of the main downsides of that game genre (I personally strongly dislike when I must grind to defeat an enemy and/or progress the story. It just put me out of the narrative), correlated with random encounters. Some games have found some interesting solutions for the player to avoid hours of grinding. As an example, *Earthbound* (Nintendo, 1994) allows the player to kill with a dash when you randomly encounter an enemy. *Persona 5* (Atlus, 2016) offers the Insta-Kill mechanic, by sprinting through the enemy. Not only does it avoid the random fight, but it also rewards the player with XPs. *Baldur's Gate 3* (Larian Studios, 2023) gives the possibility for the player to talk your way out of some situation, while he still gains XPs as rewards.

The most used solution is level-scaling, by definition the anti-grinding: we adjust the enemy level to the player level.

The battlefield is an aspect of combat in turn-based games that is often put aside whereas it would benefit the immersion to capitalize on it. We want to use the environment at our advantage and be able to either push enemies, throw items, activate hazardous locations



Mana Khemia turn order based on a cards system (Gust Co.Ltd, 2007)

Once again, we always have to ask ourselves how do we keep the player entertained? The easiest way to answer this question would be to have fancy graphics in the game, stunning animations but in my opinion, and this research on the MangaKamen channel comforts me on this, it would have to be through quick-time events and playing with the turn economy.

Slider check, combination of keys is a great way to keep the player involved, and the player stats could benefit from additional effects or damage.

As seen in Mana Khemia (Gust Co.Ltd, 2007), the turn order is based on a cards system: the actions taken by the player previously will define which position he will occupy.

Weekly Summary:

Regarding the feedback we had last week, the addition of various attacks would become necessary to improve the fun of our game. Unfortunately The way I implemented a simple attack at the start of the production was not developer friendly at all, unlike my magic system. So I took an example on this later system and I was expecting issues as it is never a drag and drop or copy and paste.



AddMP function in BP_UnitBase

I started by adding the possibility to gain MPs (which wasn't the case so far) and implemented that function in my BP_UnitBase. As you can see in the image below, both Mana Gain Table and Mana Cost Table variables allow me to give specific values to any attack (in the details panel).



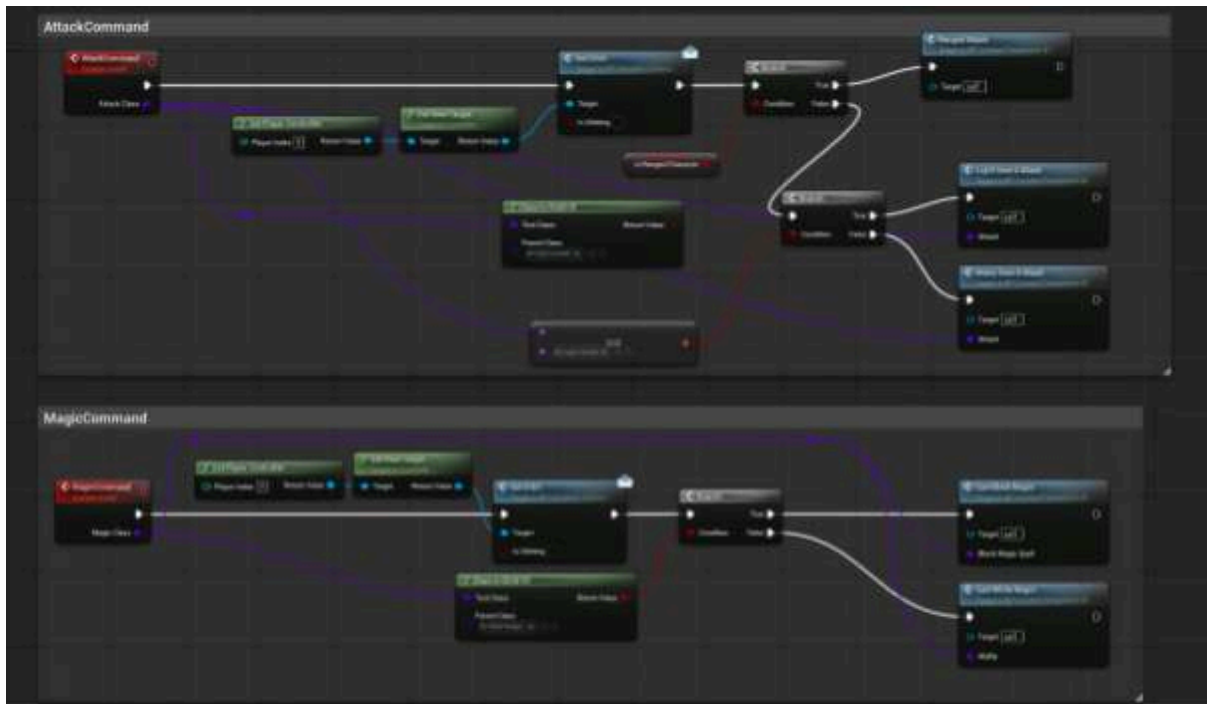
Mana Gain Table variable allow me to easily adjust value (same goes for Attack Power or Mana Cost)

Once this was done, two issues appeared to me: first the damage was not working on these attacks anymore, and the list of attacks wasn't shown properly on the widget.

Since we are not using the BP_DealingDamageNotify anymore, the way to calculate damage has to be changed and set as the one I am using for the Magic system.

Isbah has been a great help on this, with debugging everywhere we could until we found where the calculation was not working and the value of 0 sent. I have spent a lot of time on this, maybe too much, but it makes much more sense to have this system working through classes now.

I still have a little issue where the attack list does not show exactly how I would like: right both light and heavy attacks give the same list of attacks. I am currently trying to fix this now, and the two pictures below show where I am at precisely.



Testing different ways to call the attacks

Right now I am trying to copy and use the same system I used for magic: AttackCommand and MagicCommand should both check which class they are a child of, then send the other class.



Creating the widgets by calling a specific class of attacks

Same reflection on the widget: I am currently working on a way to send a list of AttackBase class variables through an array and create the specific widget. It is not successful for now, I will look for more options next working day.

Weekly Reflection:

This week has been quite difficult on a personal level as I have been terribly sick and could not really think clearly for a day or two. It feels like I have lost some time, and the fact that I had to re-arrange my system took me longer than expected (and still have one issue to fix). That was one of those bugs fixing week...

Despite all that, I had some really helpful feedback from my tutor Phil: I think it is time to bring the fun in the game. Let's introduce skill checks, sliders, and key combinations.

Finally we had our playtest on Thursday and the overall feeling was close to Phil's feedback: the players want a bit more diversity with actions, such as regenerating MPs, being able to Skip Turn, a gimmick type mechanic.

WEEK 8 : Starts implementing the fun in the game

Weekly objective: Tweaking gameplay elements, affining what is there and adding key combinations, skill check.

Research Sources:

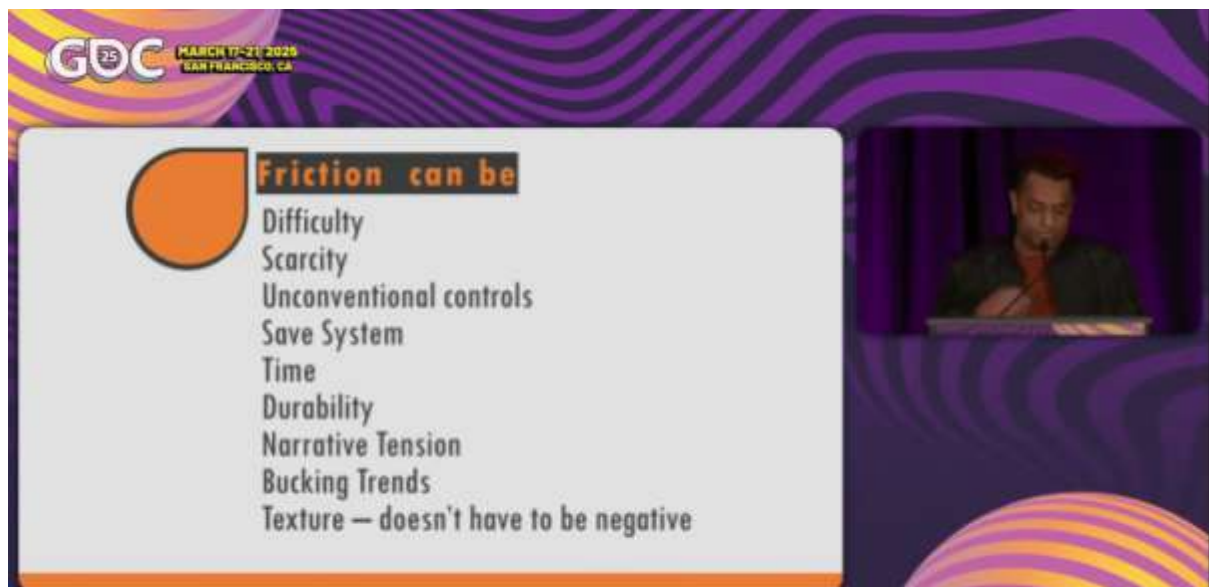
- [Rules of the Game 2025: Uncommon Techniques from Insightful Designers](#)

Research Analysis:

For the talk named "Rules of the Games 2025: Uncommon Techniques from Insightful Designers" (GDC, 2025), 5 designers were invited on stage to share

their experience and useful advice. I will highlight some particular thoughts that have been relevant to my knowledge.

First, Chandana Ekanayake, creative director at Outerloop Games gives a brief speech about friction and resonance. According to him, us as designers have to create intentional friction to build emotional resonance.



Chandan Ekanayake explaining what friction can be (GDC, 2025)

Friction can be many different things, as we can see in the image above: from difficulty to save system, from durability to narrative tension. On the other hand, resonance is when game mechanics enhance themes and emotion.

Alisha Thayer is a design director at Crystal Dynamics and has been working on titles like Max Payne 3 (Rockstar, 2012), Tomb Raider (Crystal Dynamics, 2013) or Marvel's Avengers (Crystal Dynamics, 2020).

She points out 3 mistakes encountered during the development of these games:

- Perfect gameplays are better scripts

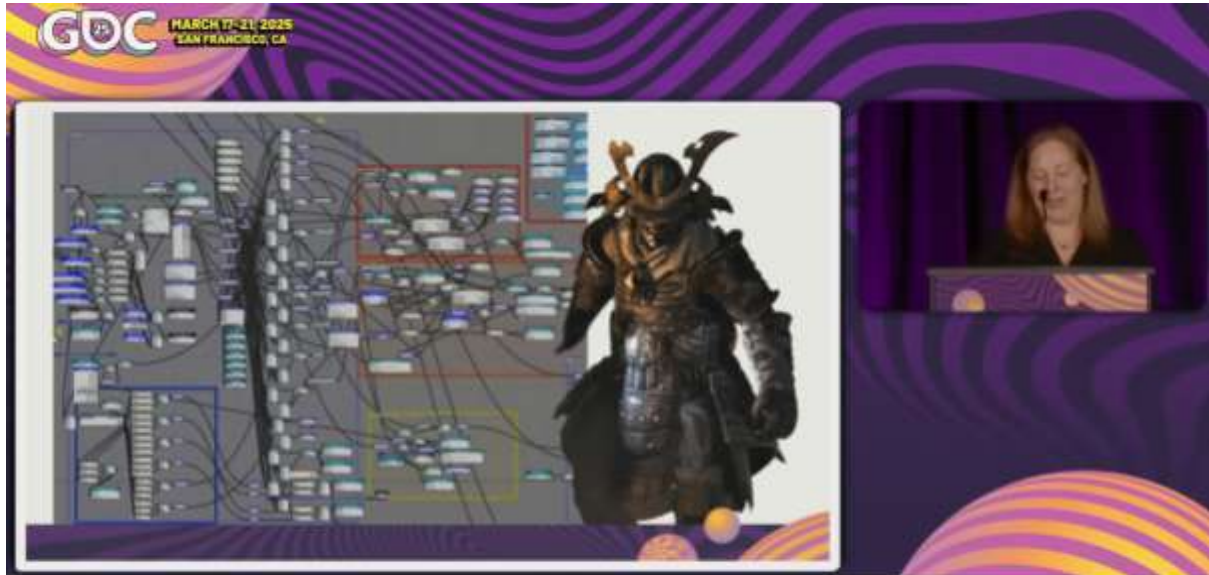
Players won't remember your beautiful, orderly code; they will remember how the game made them feel.

- Structure always improves the creative process

Iteration is messy, and mess is a part of the process of realizing great gameplay.

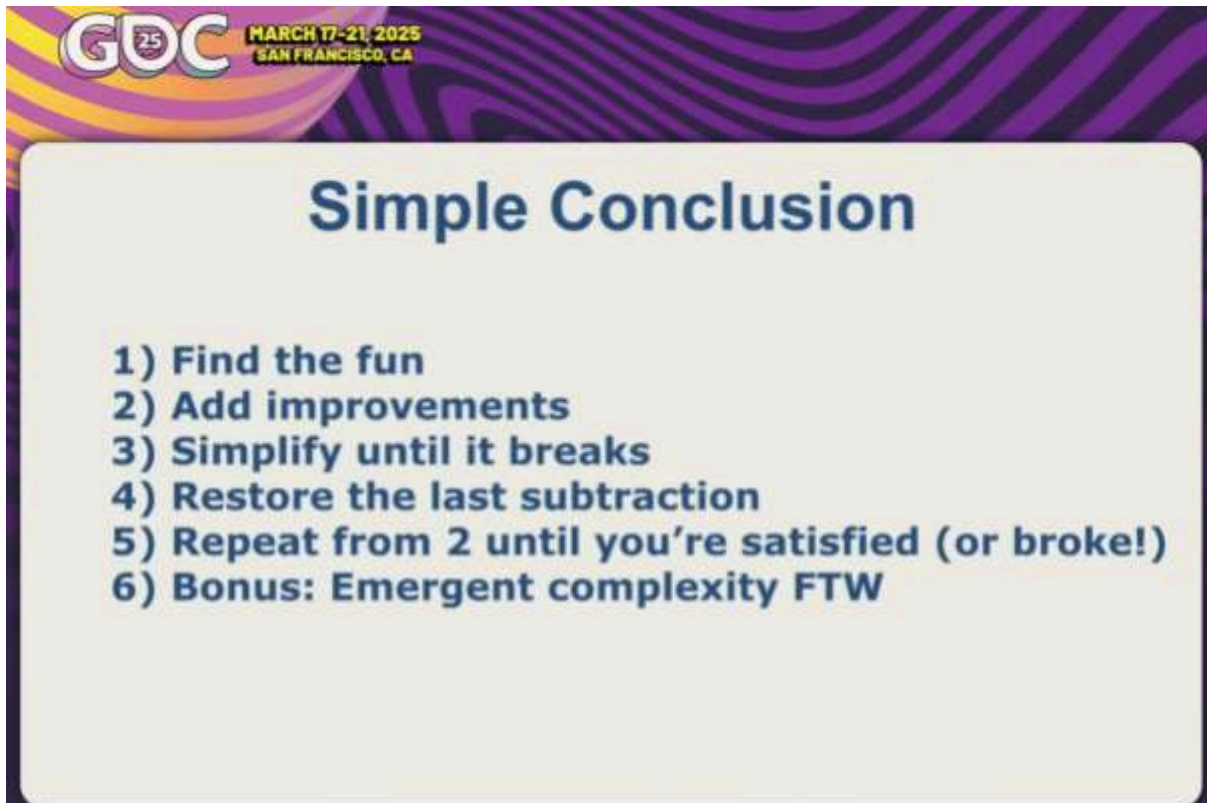
- Content patterns always make better content

If your patterns make realizing the experience harder, they are bad patterns.



Boss fight flow chart in Tom Raider (Crystal Dynamics, 2013)

Noah Falstein, founder of The Inspiracy, has been working on games such as Sinistar (Williams Electronics, 1983) and Chaos Island: The Lost World (DreamWorks Interactive, 1997). He focuses on the simplicity in production: we have to make everything as simple as possible, but not simpler. As an example, he uses his work on AI in Chaos Island: The Lost World. In this game you fight against dinosaurs and bandits; the AI on dinosaurs was not fun and very predictable. His solution was to simplify his system and remove layers of complexity until it broke. Then, he only added a bit of randomness in the AI and it worked greatly: that was a very simple solution but to achieve it, he had to simplify as much as possible.



Noah Falstein's advice on how to keep everything simple (GDC, 2025)

Finally, Harvey Smith worked on the Deus Ex franchise and Dishonored 2 (Arkane Studios, 2016); his talk is focused on the building of a thoughtful interconnection between gameplay systems, level design and narrative. He rather focuses on a game he has not worked on, Prey (Arkane Studios, 2017). This game is a great example of combining non linear, explorable space with reactive physics: the glue gun allows the player to harden foam and create barriers or new paths to areas.

The sensory based AI implemented on mimics is another example of layered system: enemies can be anything in the level, trash cans or books and the player never knows if he is walking past an enemy or just an ordinary common object.

And finally, as explained in the interview below, that interconnection between layered systems also serves a narrative purpose: telling "the story through the player's actions".

Raphael Colantonio
Creative Director, Prey (2017)

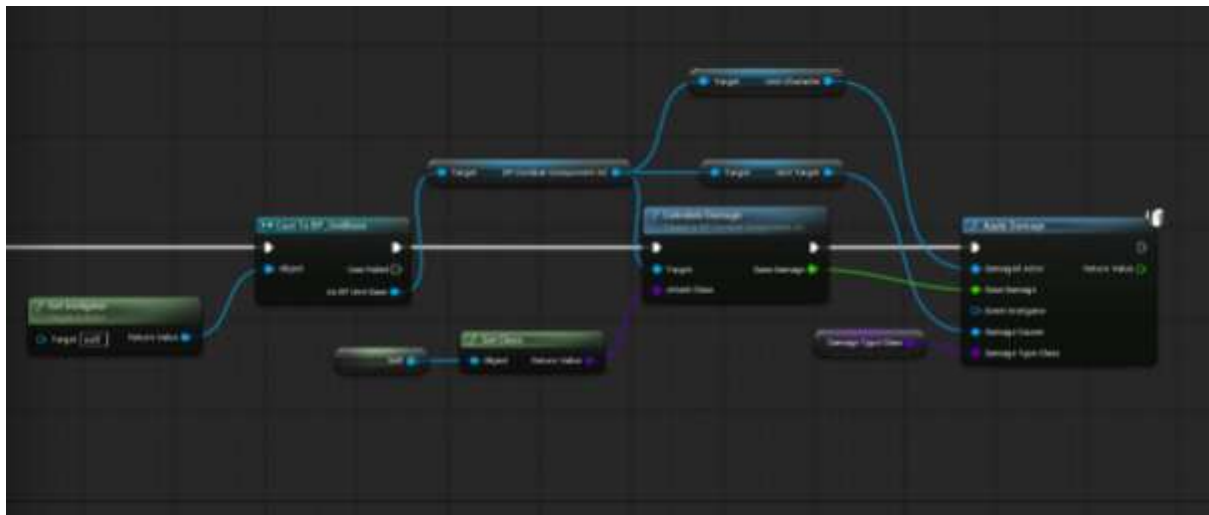
One of the things I love in our games is when people go, "Oh shit, that worked?!" It's important to tell the story through the player's actions, not through cinematics. Breaking the glass near the start [of Prey] was powerful because it happened through player action, at the player's pace. And for that to work, you have to have an environment full of narrative-rich gameplay interactions.

Extract from a conversation between Harvey Smith (Dishonored 2, 2016) and Raphael Colantonio (Prey, 2017)

Weekly Summary:

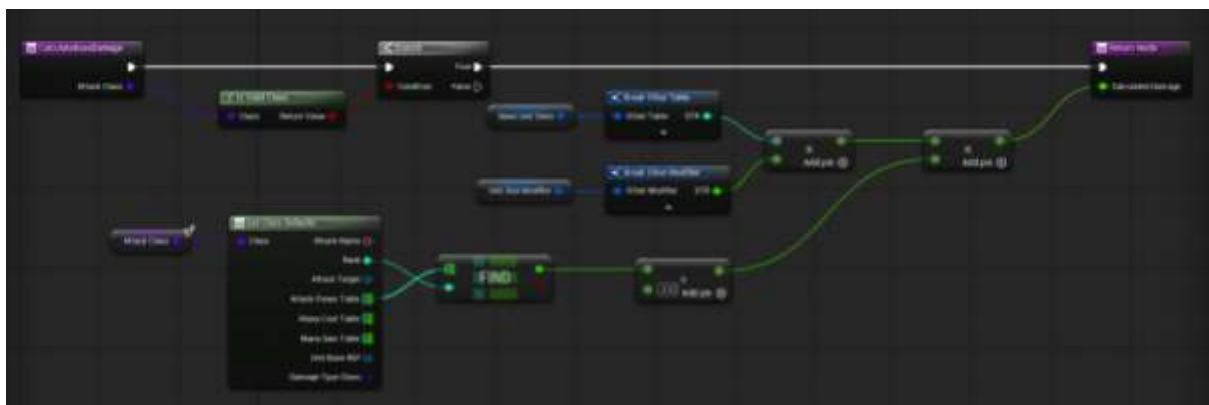
Since I had to re-think my way of implementing the attacks in my game, it came with more bugs than originally planned. As an example, the mana power was not recharging when using physical attacks, it worked only for abilities.

Besides, the enemy was not damaging the players anymore. To correct this issue, I had to add a Unit Base reference in the "damage actor" pin in the "Apply Damage" node, for the specific attacks used by BP_Kael.



Unit Character and Unit Target variable needed to be swapped

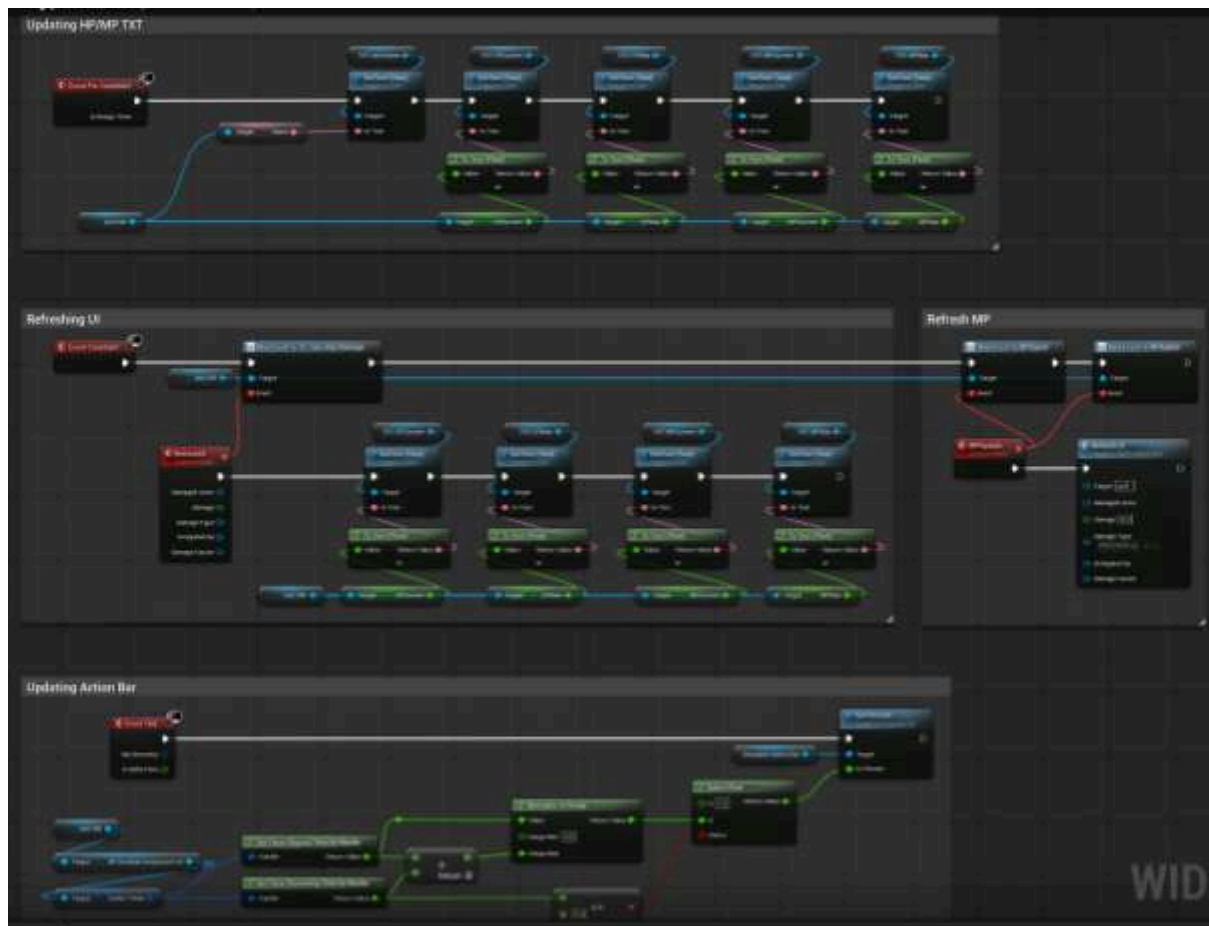
When I was checking the damaged values applied to BP_UnitBase , I found out they were not the ones set in the combat component. As an example, the light sword attack is meant to do an -10HP damage, but the value I had was around 12 or 14. The issue was in the CalculateBaseDamage function after the “Find” node: with Isbah help, I have debugged each step to see where the calculation was wrong. I originally copied that function from “CalculateMagicBaseDamage” and it contained a “SQRT” node that would drastically change the output values in my new function. By removing it then dividing by 2 it allowed me to have the correct damage number.



I had to rethink the maths here and debug at every step to understand what was happening.

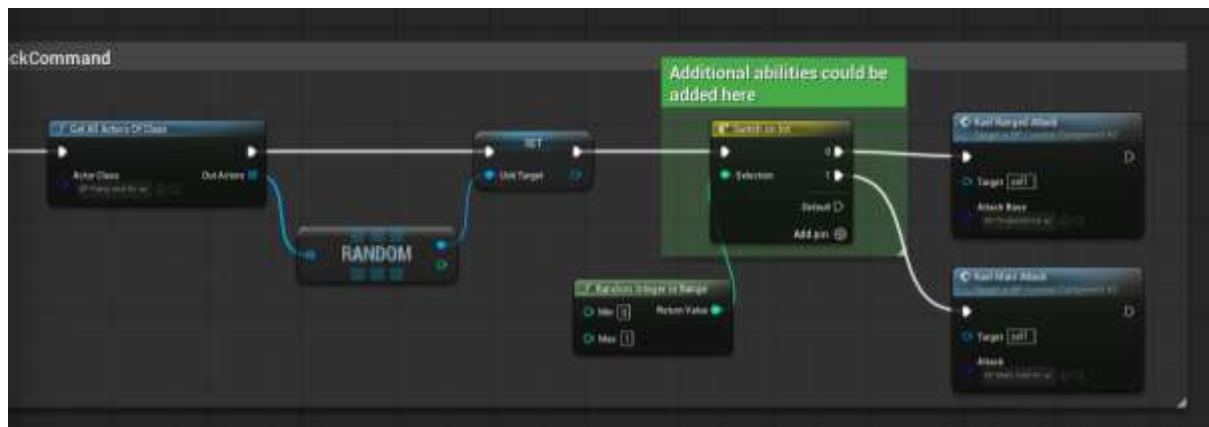
I finally found a way to have all the available attacks showing in the hud: that was a simple mistake on my side. In effect, I forgot to add the attack class in the specific character, through the combat component details function.

I added the health bar for the boss.



The functionalities of the boss health bar in the widget.

I added a projectile class for both the party and enemy, following the new attack system in place. To do so, I had to re-think once again the projectile BP I originally implemented: I wanted those to be included in the main attacks class base, so I could use damage, MPs gain or loss at my convenience and have this system working for both playable characters or the boss.



The boss attacks are randomized on a Switch for now.

Weekly Reflection:

It was great to work with Isbah, she has been a great help and because of the time spent working with her, I guess I am getting better at debugging blueprints.

We had the usual playtest session on Thursday afternoon and the main output was that the game feels too repetitive at the moment. Also the QTE we started implementing was not clear enough for the players, and they did not know what keys to use to complete it.

A great output we had was to need for the player to see what attack is actually doing, what kind of attack it is and what it does.

My response to this feedback session was to bring action and unexpected events in the game such as the parry, or start working on the boss AI. At this very moment his attacks are randomized but I think the game would benefit to have these designed.

During the online review on Friday morning, my tutor had a different analysis of the feedback: instead of adding QTEs or a parry system, why not think about improving the core gameplay? I have to come up with a system where the player has to think about his next move.

That seems to be strongly related to my “Seize and Shatter” mechanic, so I know what is on my plate for the next week.

WEEK 9 : “Seize and Shatter”

Weekly objective: Working on the first part of that mechanic: “Seize”

“Seize” is the Valeria part of that mechanic.

- Create a system that would check Kael's state before each attack: is he floating or grounded?
- Having the supercharged Valeria's attack (Eidolon) grounding Kael.
- Create a system that would change damage applied to Kael depending on his current state
- Show in the UI those specially modified values

Research Source:

- [Building a Paper Prototype For Your Narrative Design](#)

Research Analysis:

Jamie Antonisse is a game designer and a writer for Disney and during his talk at the Game Developer Conference in 2014 focuses on how to build a narrative paper prototype. Even though our game is not based on narrative, I found that research useful as I am using a prototype for our project and I want to specialize in narrative design.

The fundamental question we can ask ourselves is: what is a prototype?

- It is focused on a core aspect of the experience
- It has to be playable

- It has to be made quickly
- It is a disposable practice run of a grander project

It has to be differentiated from a digital sandbox prototype, which is quite often the only prototype. The digital sandbox prototype doesn't address the larger experience and doesn't include the progression, choices or the game's narrative.

Narrative in game has its own specificities: the player is performing actions that influence events, he has to learn to master systems to achieve their goals. Finally, and the most important, the player is the hero.



Jamie Antonisse during his talk at GDC in 2014.

Then we can observe different narrative prototyping practices. For example, prototyping narrative in books can result in the use of notes, outlines or drafts. Films will use scripts, storyboards, read-throughs whereas animated films would use animatics or pre-vis on top of those previously cited. The biggest risk in games is the player: we have to put the hero into the narrative as cheaply as possible, as early as we can.

The making of the prototype ideally goes along with a prototyping toolbox (including an idea, a deadline, prototypers if needed, simple supplies, narrative

reference points, a personal storytelling experience and a focus on the central question) and an initial write-up.

Regarding the write-up, some interesting points highlighted here by Jamie Antonisse: he recommends to “go nuts” for 10mn and see what is happening. That is in that part of the process we can establish the player role, goals, conflict, choices, actions or resources.

After building the story, we should be very likely to cut story points that don't:

- Reinforce or showcase goals
- Call the player to action
- Give the player feedback on their choices
- Provide a break/reward after heavy action

And finally Jamie Antonisse strongly recommended practicing the prototype following three steps: first completing a solo run, then with trusting friends that would not be judgmental and only after with people he called “conspirators”.

He goes very quickly then on recording the playtest session, if possible on video, collecting and responding to feedback, steps that I am much more familiar with.

Weekly Summary:

Once again I played the paper prototype with a friend of mine and came up with different new ideas. I have changed my perspective and wanted the player to think about his next move, or even his next moves.

As a result of this, buffs and debuffs have made their apparition with the attacks. As an example, let's take the new ability called Chrono-Sanctification: Lucien can call Valeria right after his turn, skipping the enemy turn. It is a strong move that also applies the “Strength” status to Valeria (she can deal +50 damage for one turn) but it costs -10HP to Lucien.

As you can see, those new ideas are meant to give more choice to the player, so he can elaborate the best strategy to kill the boss.

Action Types	Name	Enemy DMG	MP	Function	Buff	Debuff	Elemental Effect
Light Attack Dark Magic	Ichor Rain	5	-1 MP	Pour holy water on Kael	Afflicts 'Holy Curse' on Kael for 3 turns.	-5 HP on Lucien	Water
Light Attack Dark Magic	Rite of Cinders	10	-1 MP	Fire spell		Afflicts 'Searched' on Kael for 1 turn, or for 2 turns if Kael is Grounded.	Fire
Heavy Attack Dark Magic	Glacial Tomb	15	-2 MP	Ice spells.	Gets rid of 'Holy Curse' debuff.	If Kael is already 'Grounded', deals +10 damage. If Kael is afflicted by 'Holy Curse', Kael is 'Grounded' for 2 turns.	Ice
Defense White magic	Prayer of Mending		-2 MP	Heals Target (+30HP). If Valeria is picked, +2 AP.	Applies 'Heavenly Blessing' to chosen character for 1 turn.		
Defense White magic	Consecrating Cleanse		-3MP	Cleans up to 2 debuffs from the party member it is applied to.			
Defense	Chrono-Sanctification		-3 MP	Valeria gets her turn next, skipping Lucien and Kael's turns.	Applies 'Strength' to Valeria for that one turn.	-10 HP on Lucien	
Skip Turn	Brace		+2 MP				
Supercharged defense	Divine Blessing		-5MP	(at max. MP) Strengthens party member it is applied to	Applies 'Godly Favour' to player for 3 turns.		

Example of Lucien new skills after paper prototyping

Status effects have all been given names, with help of Isbah, and their design came from playtesting the paper prototype.

Status Effects

General Status Effects

Name	Type	General Effect	Player Specific Effect
Weak	Debuff	Reduces character's total damage by 5.	
Scorched	Debuff	Character becomes invulnerable and take any action.	
Strength	Buff	Character does 10% more damage.	

Party Status Effects

Name	Type	General Effect	Player Specific Effect
Holy Curse	Debuff	Deal 5 Damage to Kael at the start of each turn.	
Scorched	Debuff	Character takes an extra 5 damage each time they take damage.	
Heavens Flaming	Buff	Character takes 30% less damage.	
Godly Power	Buff	Character deals 10% more damage, takes 10% less damage, all (30%) are automatically successful.	
Invulnerability	Buff	Character takes no damage.	

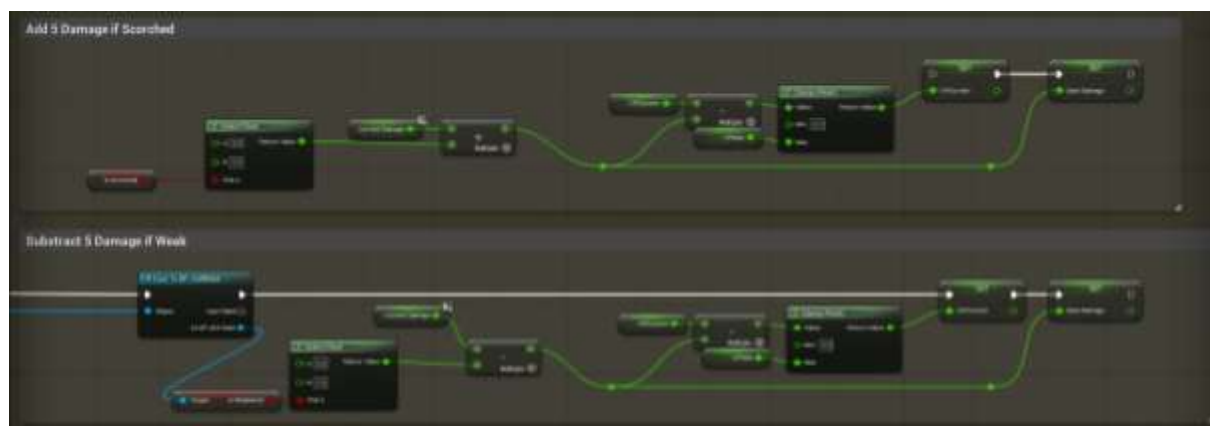
Kael Status Effects

Name	Type	General Effect	Player Specific Effect
Life Steal	Debuff	Any damage done to the affected character leaks Kael for that amount.	
Frightened	Debuff	Character takes 10% more damage.	If Kael is frightened, also gains 10% less AP.
Weakness	Debuff	Character does 10% less damage.	
Spectral Regen	Buff	Gives character a shield that reduces the next attack deal 5 damage but breaks the shield.	

Buffs and debuffs have made their apparition in the combat

On top of the core concept of the game, I started working on the "Scorch" status: when the enemy is hit by Rite Of Cinder, the players deal 5 extra damage for one turn.

I have created a BPI_CombatUnit interface, the event being called in BP_UnitBase.



Currently working on a buff and debuff system, adding or subtracting HP from the BP_UnitBase

In the UpdateStatusEffects function, I am setting how long I want a status to be effective.



Checking how long we apply a status, depending on how many turns left.

Weekly Reflection:

I have not fulfilled any of the objectives set at the beginning of the week: the paper prototyping testing has had us reconsidering many aspects of the attacks. Now we have buffs and debuffs, statuses.

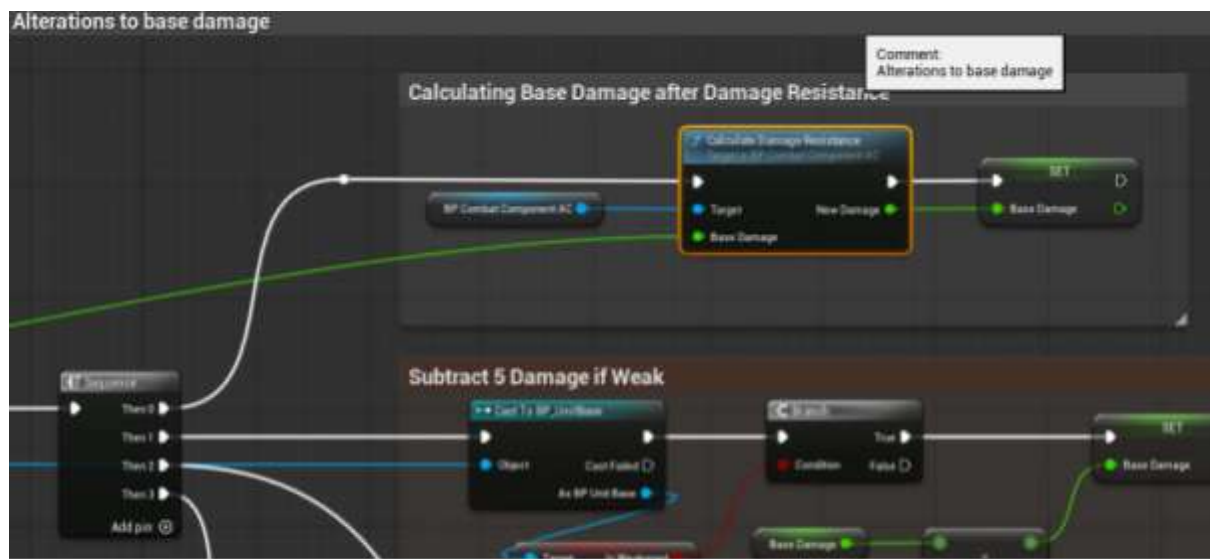
I am considering the possibility of getting rid of the grounded mechanic: I am not sure it can be a central aspect of combat anymore.

WEEK 10 : Refactoring the combat system

Weekly objective: Since new attacks, buffs and debuffs are coming in the game, I might need to think of a new optimized way of implementing them.

Weekly Summary:

Early this week we had an issue with the project, due to a “redirectory update” done by a studio assistant. This has caused a hard reset of the combat component values: fortunately I had a copy of the project at home so I could compare and restore them.



Setting New Damage as Base Damage

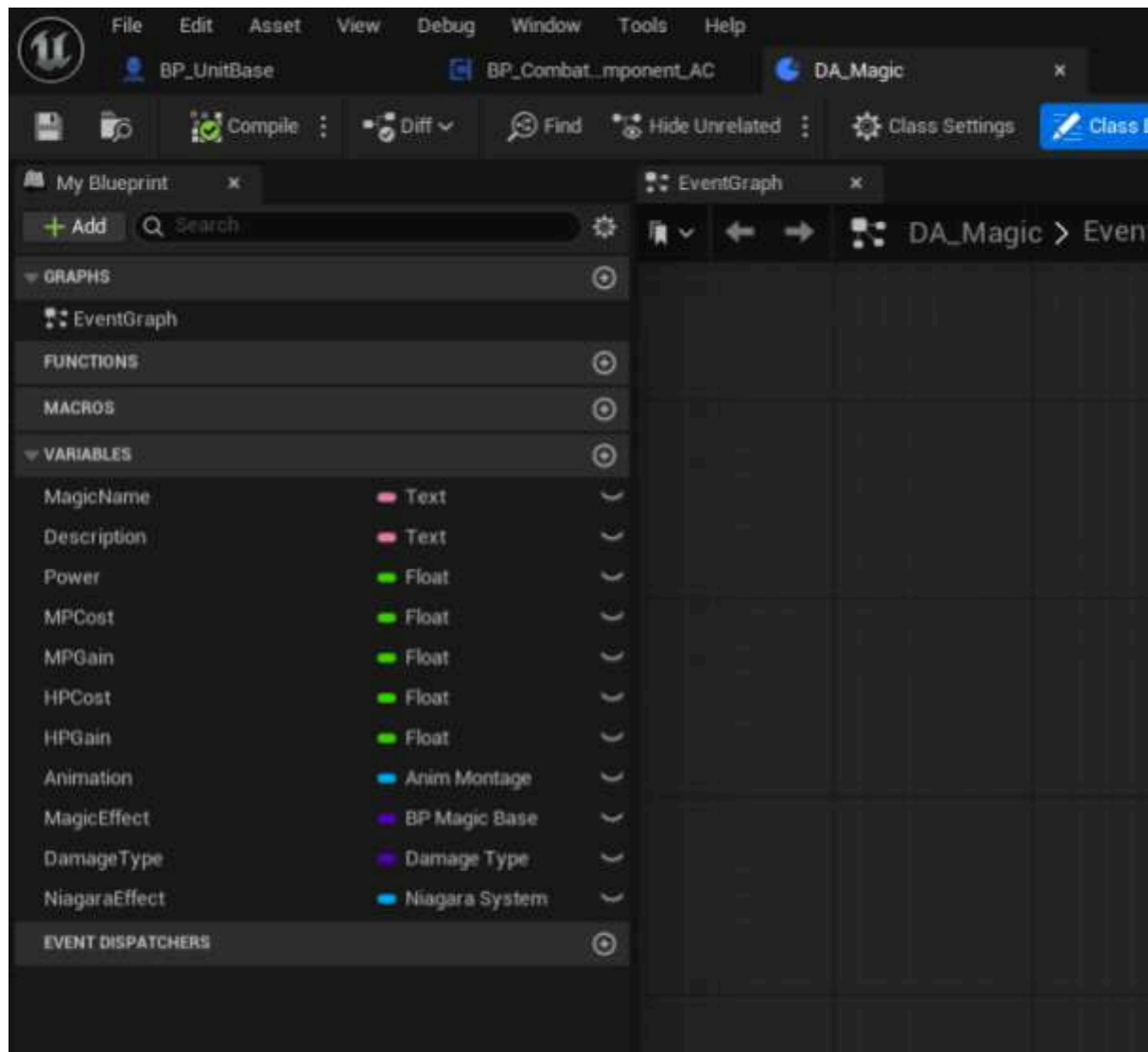


New "Scorched" Damage

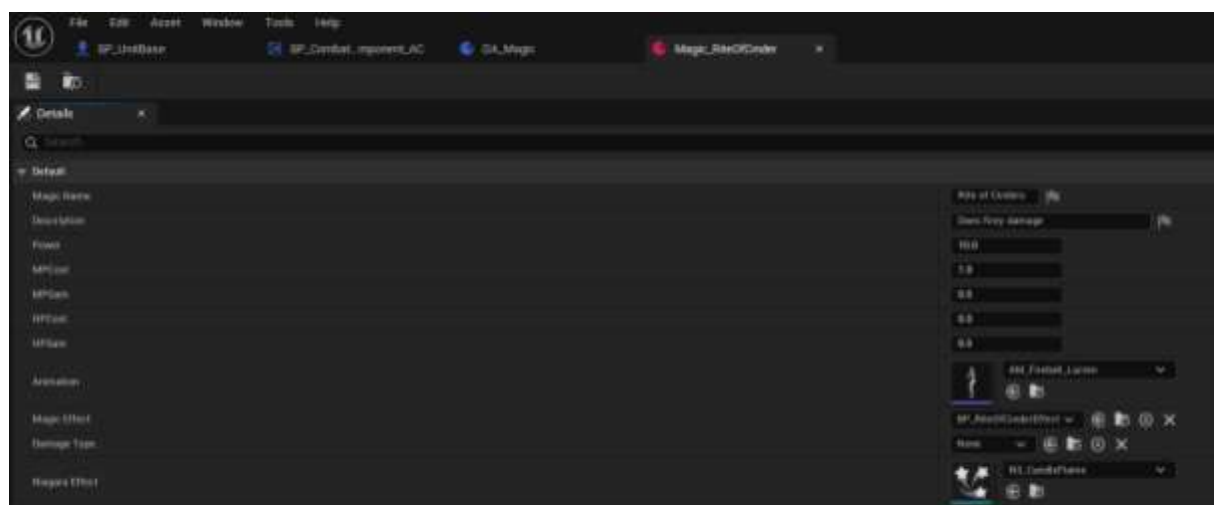
When I implemented the "Scorched" status, I could see the damage was being counted twice. Isbah helped me to fix this: the solution was, as always with me, to make it simpler. Once the base damage has been set after the CalculateDamageResistance function, it is just a matter of adding the desired damage through a float variable and setting this as the total damage.

The biggest work of this week has been done refactoring the combat system: it now runs through data assets and not classes anymore. The purpose of this change is to give much more flexibility when it comes to adding new attacks, spells, projectiles or even vfx, animations.

The following images go through the main steps of the process. Regarding the HUD, it was just a matter of updating the ability class references to the new selected magic data asset variable.



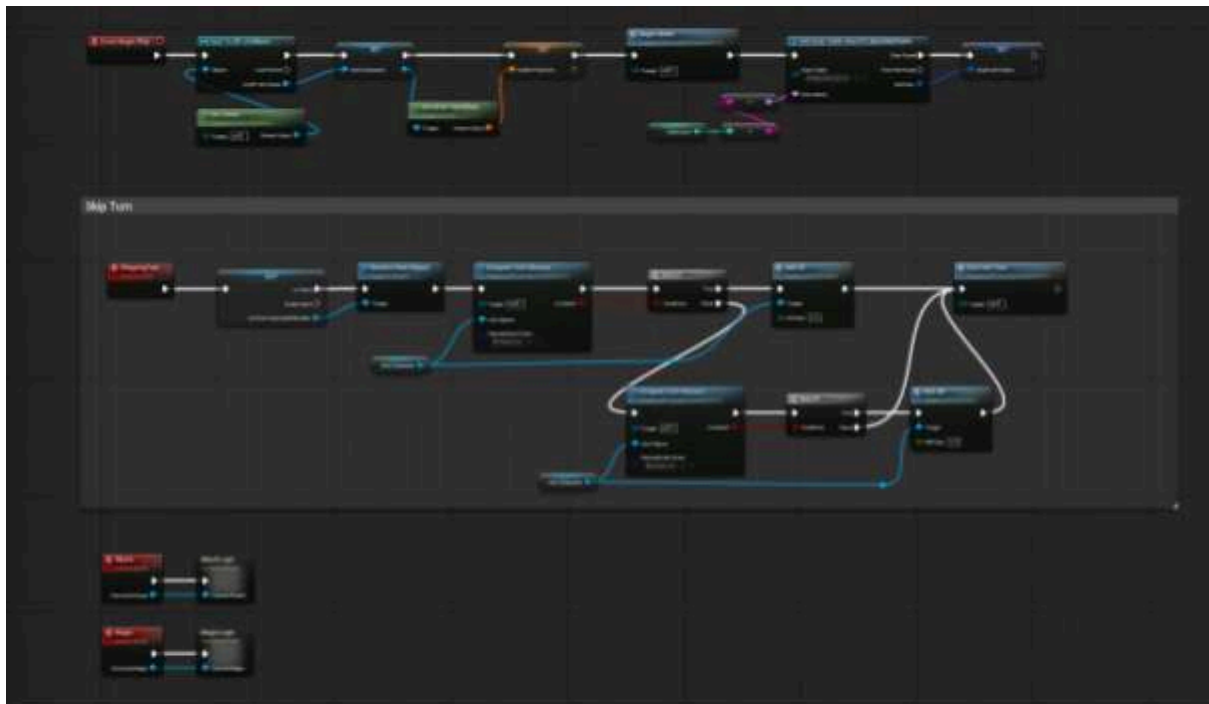
Defining the base variables needed in DA_Magic



For each specific attack, I can easily adjust the values



I can create a specific BP_Effects, where a desired behaviour can be implemented.



Event Graph of the Combat Component after using functions and collapsed nodes to clean it up

Weekly Reflection:

A very satisfying week even though not much had been added in the game but the time spent with my tutor refactoring the system was priceless.

Regarding the team, same old. Barely no one online during the online review feedback, poor attendance at Uni.

WEEK 11 : Implementing more statuses

Weekly objective: The objective for this week is to add more statuses to the attacks or spells.

Weekly Summary:

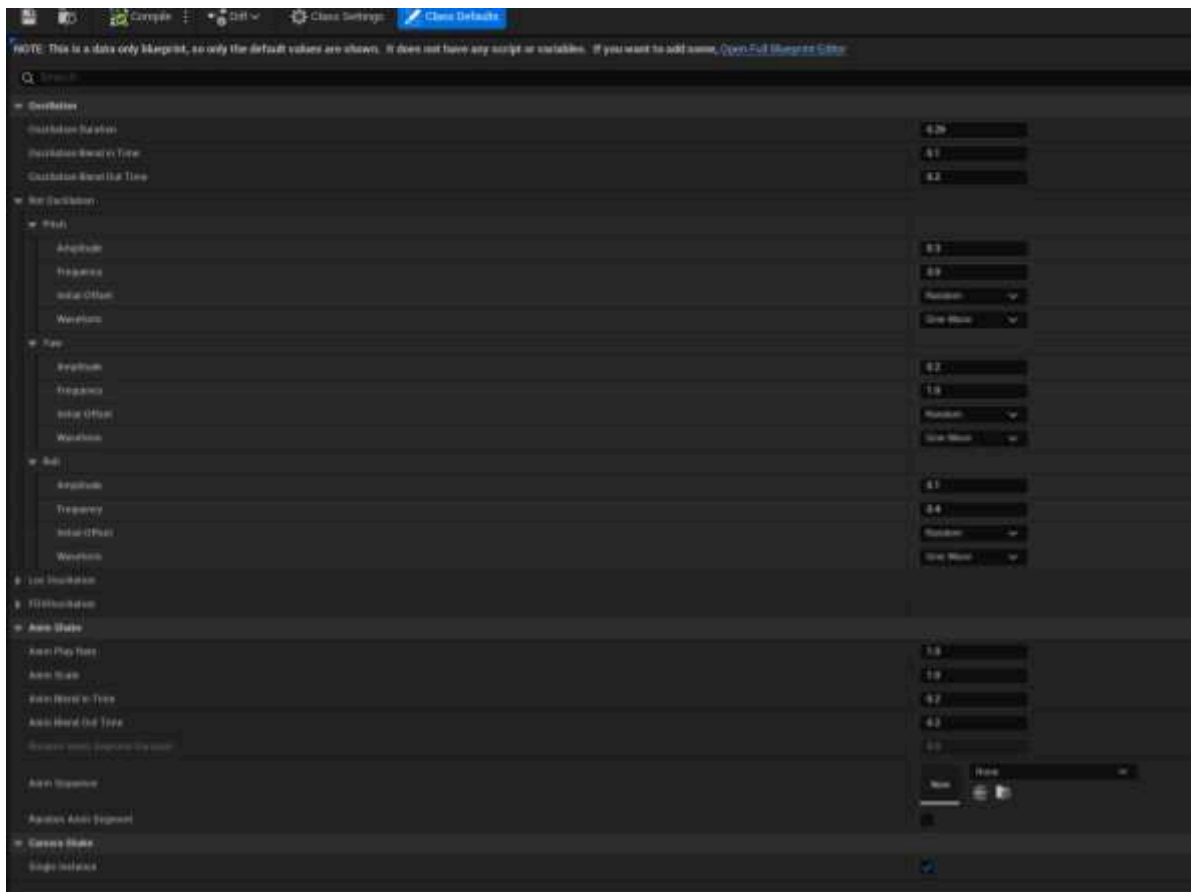
From the feedback we had about our camera angles, two different points of view:

- Metafor: Refantazio style (camera slightly inclined, from a low perspective, close to the characters and acts as a disorienting high energy director)
- Expedition 33: Clair Obscur (camera moves aggressively to frame attacks, aiming for a movie-like feel during battle)

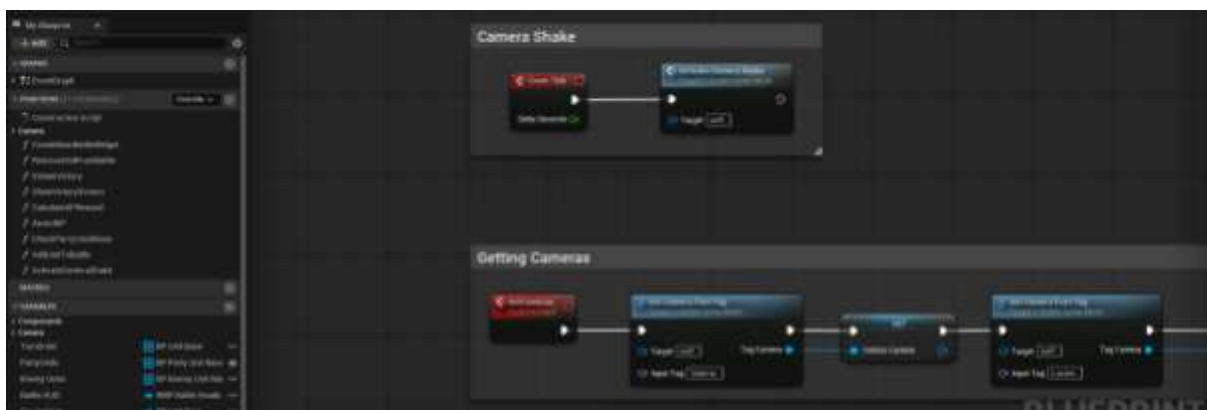
One of our tutor did not appreciate the Metafor style, so I adjusted the camera to have something more stable. Another aspect that counted much is the possibility to appreciate the environment, to show the lore.



New camera angle gives that cathedral feeling



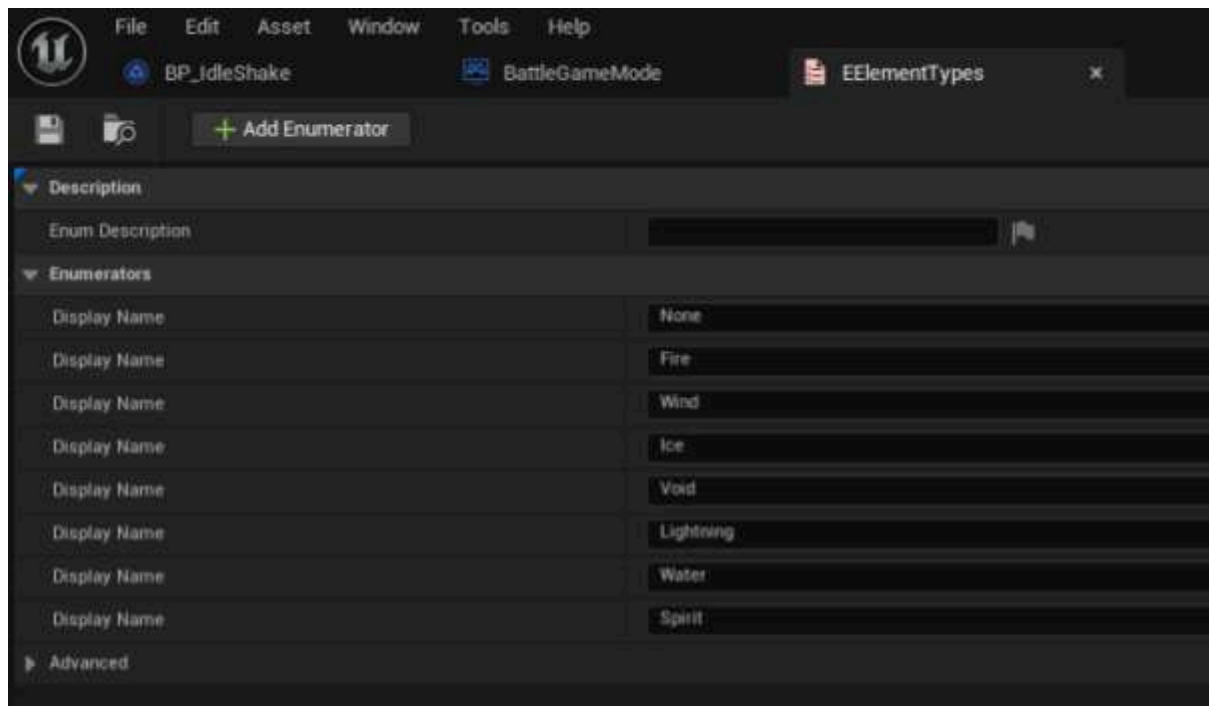
Reducing the Pitch and the Yaw (Amplitude and Frequency) for a smoother feeling (BP_IdleShake)



ActivateCameraShake function set in BP_BattleGameMode

Additionally, I started working on elemental combos, and how the player can be immune, resistant or weak. I have created an Enum where I listed the

element types we defined in our game design document such as water, fire, spirit or void.

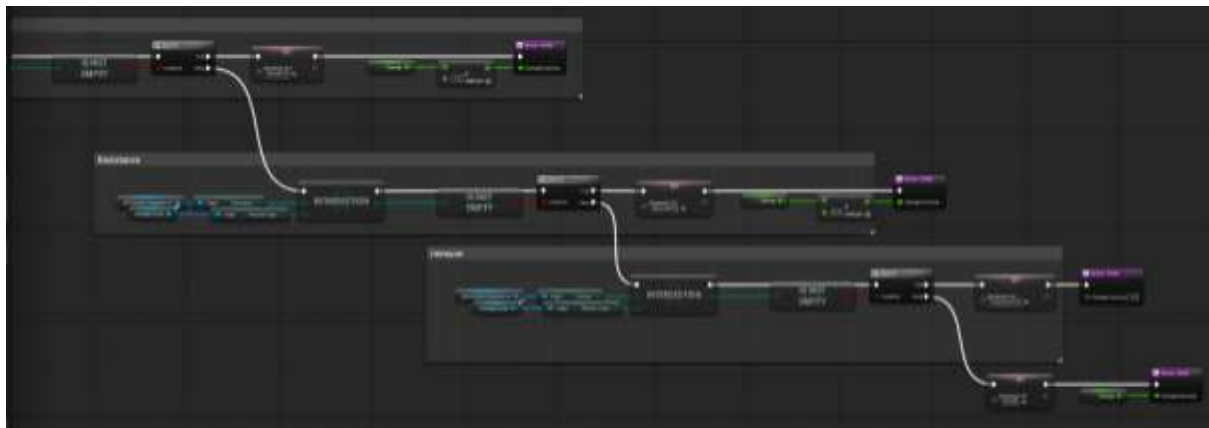


Element types listed in the Enum

Then in a function called ElementalAffinities, in the BP_UnitBase, I am calling those element types and defining 3 different behaviours: weak, resistant and immune, with specific damage for each one of these.



ElementalAffinities function part 1



ElementalAffinities function part 2

This is a work in progress: at the moment this function does not fire in the HandleDamage function, in the BP_UnitBase.

Finally, I have been working on these two statuses, Holy Curse (Kael takes 5 extra damage for 3 turns when Lucien uses Ichor Rain) and Heavenly Blessing (unit chosen take 30% less damage for 1 turn).



Checking if the enemy is "HolyCursed" before removing HP at the start of the turn



If Heavenly Blessing is on, base damage is multiplied by 0.7 to reach 30% less damage

Weekly Reflection:

Overall good week, great exchange with my teammates, and finally we have almost everyone during scrum meetings, which actually got everyone involved and more motivated.

I spent a lot of my time with animators, we might have to do some cuts. They have a huge amount of animations to work on, and more animations means more specific behavior, vfx, UI... I do not think we need a crazy amount of spells or attacks for the game to be fun.

On my end, I still have some annoying bugs that I want to get rid of. I might ask for help.

WEEK 12 : Playtesting on Itchio

Weekly objective: The objective for this week is to add controller compatibility, have a build of the game ready to be played, Itchio page.

Research Sources:

- [Boss Battle Design and Structure](#)
- [Keys to Rational Enemy Design](#)

Research Analysis:

I have been assigned by my tutor to study a couple of research documents he provided in order to improve my boss combat and its design.

The first piece of research is a post article written by Mike Stout, game designer at Activision. I found this article very interesting and every bit of information is priceless but nevertheless I have summarized the big inputs.

1. The Boss is a test

The player wants to demonstrate the mastery of his skills. To help with that, we can make a list of the skills we want to test the player on, make a list of attacks or challenges that will test those skills, decide how we want to theme the attacks, and decide how the boss defends himself.

This last point is important because the boss needs defensive capabilities, we have to allow him to attack without being interrupted, and give himself space to attack the player.

2. The Boss is a story

Boss battles tend to be structured based on a series of “story” beats. Mike Stout identified 8 beats: build up, intro/reveal, “business as usual”, escalation, midpoint, It’s on!, “kill” sequence, victory sequence.

This provides a strong backbone when it comes to breaking down boss battles.

The second piece of research is a post article written by Nicolas Krag and relates more to RPGs enemies. It contains some very resourceful information but less focused on boss design. Nicolas Krag defines 5 core keys to reach what he calls a “rational enemy design”:

1. Make a list of all your player’s abilities and associate all your enemies to each of them.
2. Identify the pillars of your game and the core skills it challenges in your players.
3. Different combinations of metrics let you create widely different enemies.
4. Identify the behaviors you want, or do not want, your player to show.
5. What emotions do you want your players to feel?

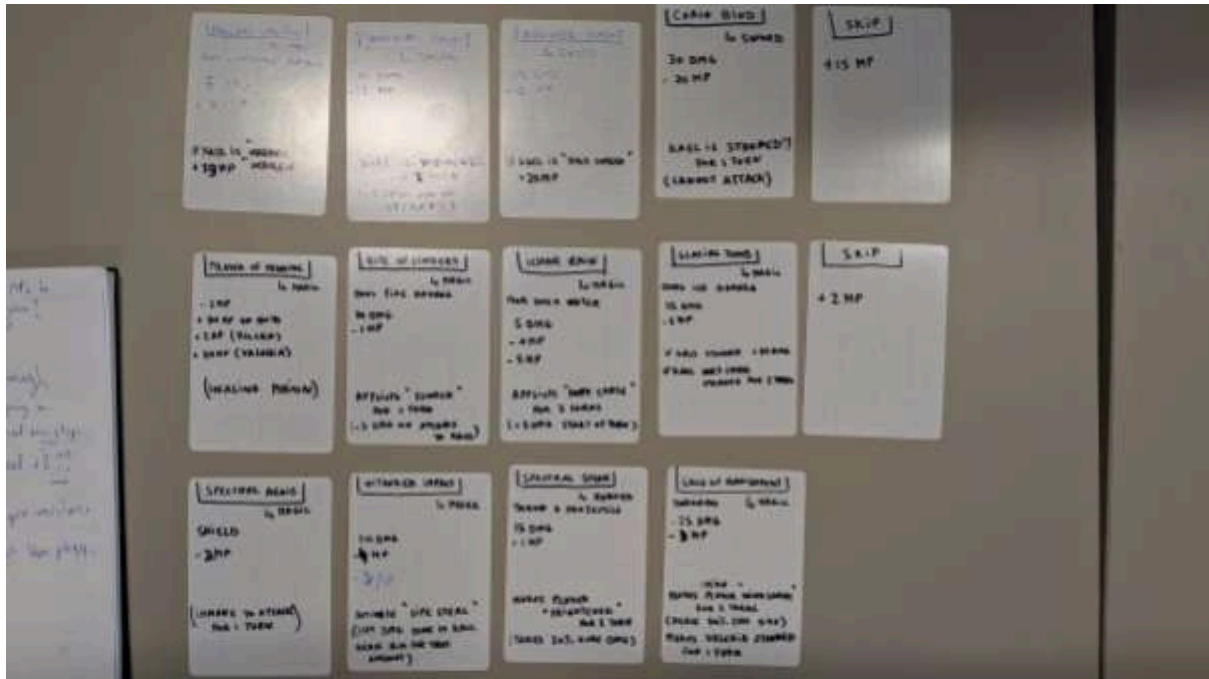
Weekly Summary:

Early this week we decided to reduce the scope of the game, due to the lack of time. We have 8 weeks left, and it feels like we will never have all the animations originally planned. The plan is to have 4 attacks or abilities per character, adjusting statuses, buffs or debuffs after playtesting.

It means I had to go back to paper prototyping the game to balance these attacks together.

- Fulgur Strike gives +25MPs if Kael is *weaken*
- Vanguard Slash gives +20MPs if Kael is *holy cursed*
- Chain Bind is not 100 MPs cost anymore but 70, afflicts 30 DMG
- If Valeria skips her turn she gains +15 MPs

- Weak status up to 2 turns
- Prayer of Mending heals both characters (+30HP) and gives +20 MPs to Valeria
- Any MPs cost related to Kael has been reduced by 1
- Kael HP set to 400



Refining the paper prototype



THE PRICE ON KAEI

This is a university team project, currently in development !

OVERVIEW

THE PRICE ON KAEI is a turn-based boss-battle experience set in a dark fantasy world.

In the region of Moria, the land and its people are left butchered by the wickedness of the Lich King, Kael. The encounter pits Valeria the Tracker and her acolyte the Priest Lucien, both summoned by the guild of Saint Voxel to vanquish the sinister figure.

GAMEPLAY

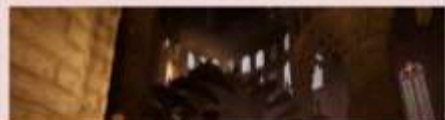
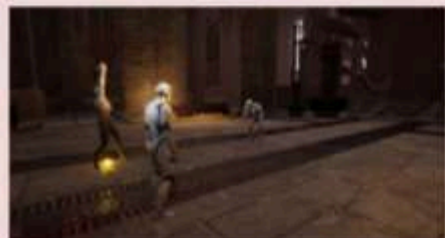
Play as Valeria and Lucien and learn how to make the best of their abilities. Valeria is a knight that deals massive damage with her sword whereas Lucien can cast offensive or defensive spells. But there is more to it. Let's see if you can build on statuses and coordinate attacks to have more impact...

FEEDBACK :

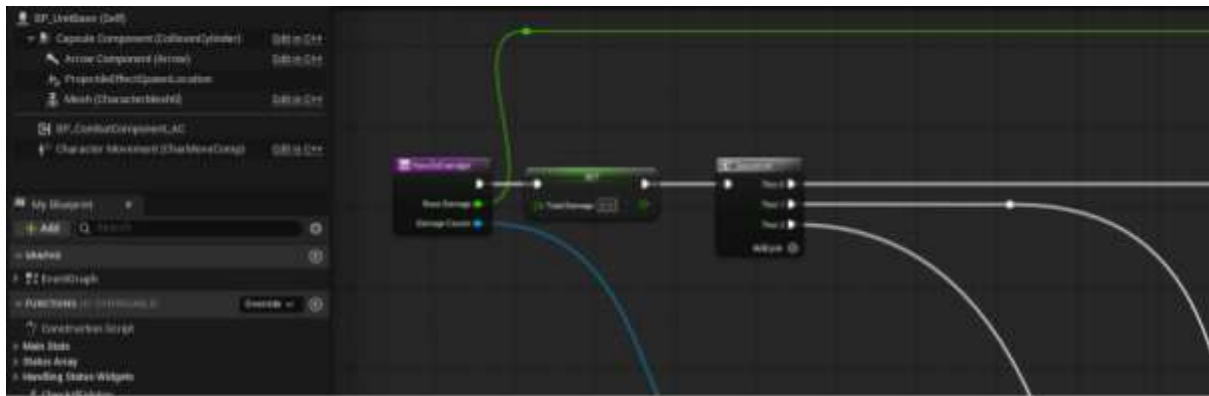
Please share what you think about the gameplay. Also, tell us about the mechanics. What was your overall experience like?

This will help us make the game better.

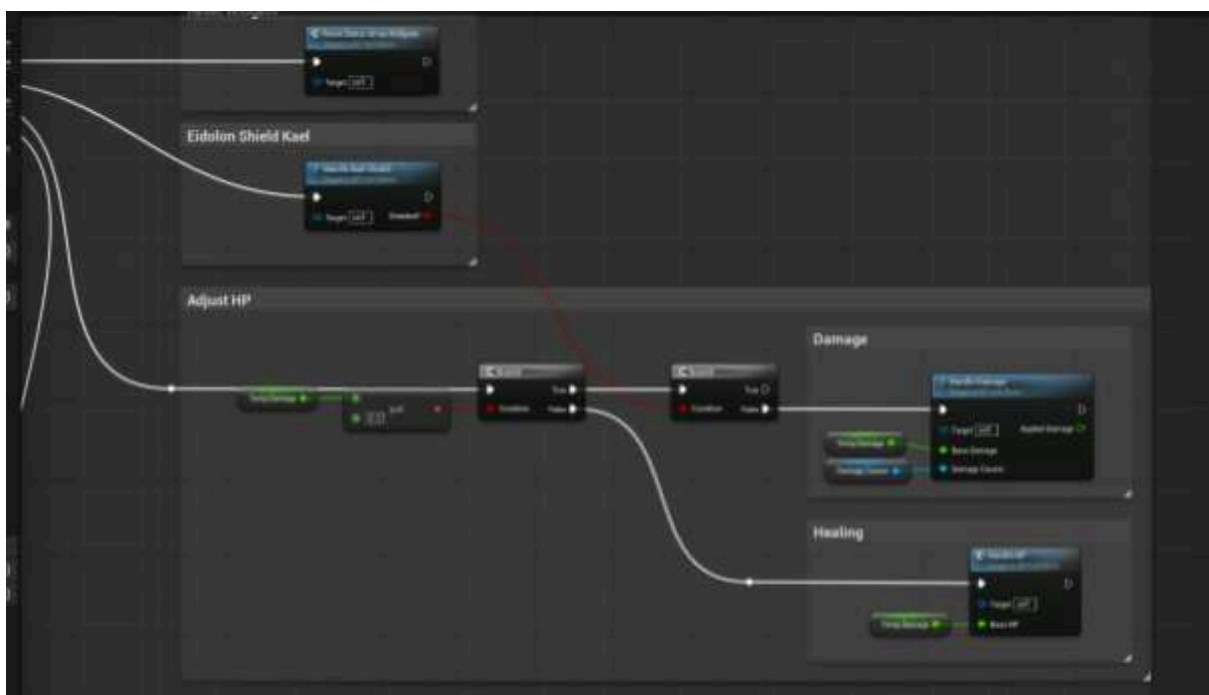
Link: [https://docs.google.com/forms/d/e/1FAIpQLSe3QD7cPonkCsJ2\(Aakm_QeTQcGjIBgDy_AN...c...](https://docs.google.com/forms/d/e/1FAIpQLSe3QD7cPonkCsJ2(Aakm_QeTQcGjIBgDy_AN...c...)



Working on the Itchio page



Setting the total damage to 0 before dealing any new damage



Checking if shield is active or not before dealing any damage

Weekly Reflection: This week has been quite positive as we finally have the game playable on Itchio. Working again on the paper prototype has been quite interesting as it allowed me to scrap what was not essential and focus on the core gameplay elements. I am a bit worried about my teammate Marie who has not been in class for a while and after 3 months we barely have any UI in the game except what I implemented originally. Same worry regarding the characters...

WEEK 13 : Mentorship session

Weekly objective: Fix the build issue, finalise boss design and start implementing it, have the documentation ready for industry feedback review.

Weekly Summary:

This week I have been working on the “Frightened” status. When Kael hits a unit with his Spectral Spear attack, the unit is frightened, which means any attacks would deal an extra 20% damage.



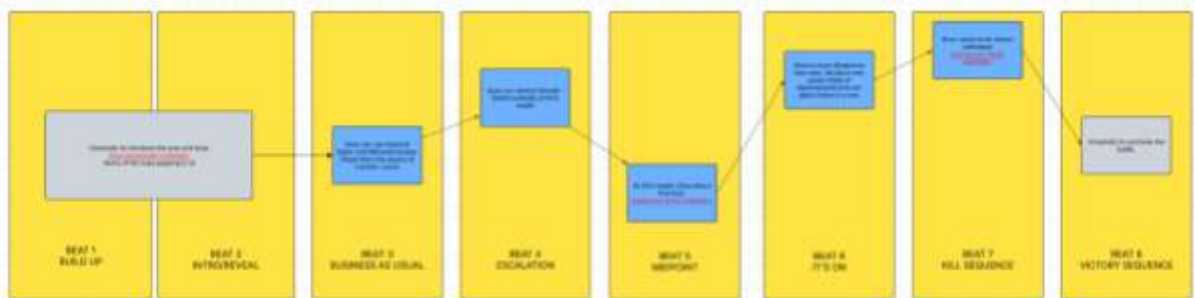
Frightened damage deals an extra 20%

The “Life Steal” status is a defensive buff that allows Kael to himself. When he uses Withered Impact, for 2 turns any attempts to damage him would heal him for the same amount of damage.



LifeSteal function allow Kael to heal himself

Looking back at the research, I designed an emotional chartflow of the combat, thinking about the intensity, the story and how could I test the player in building his own strategy to defeat Kael.

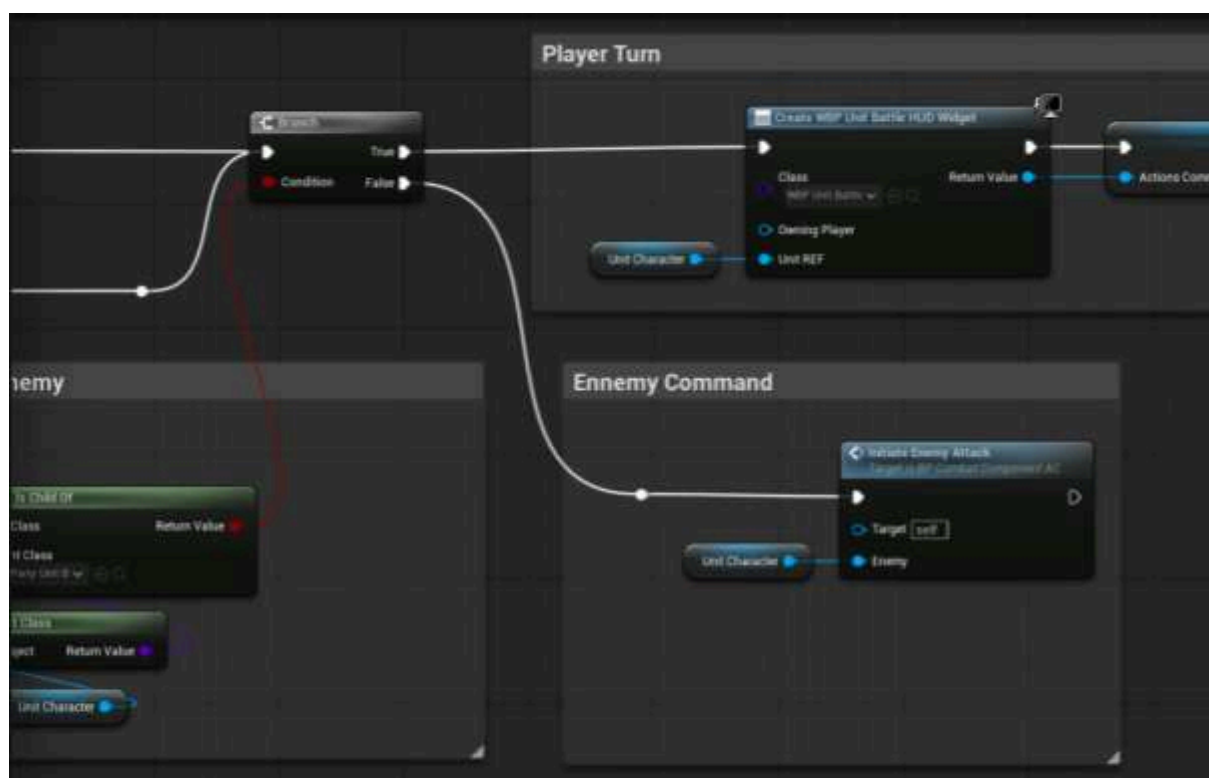


Boss behavior design

I originally started working on some of the documentation regarding the mentorship session planned on Friday, before being told there is no need for it as the only thing they need is playable build.



Refining the Game Treatment before the Industry review session



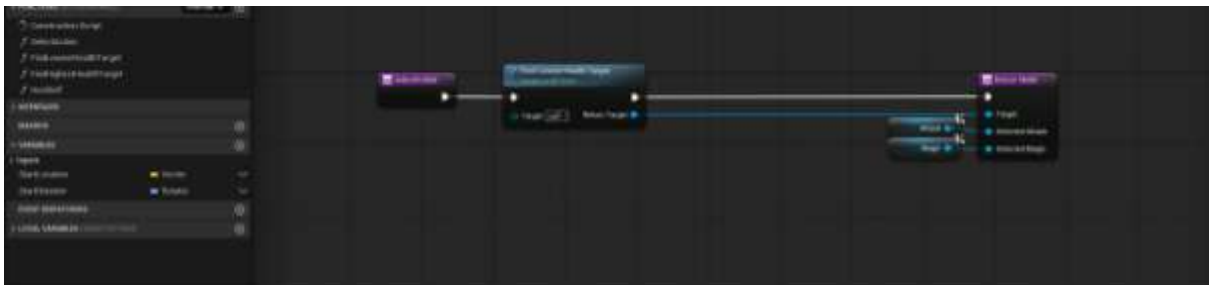
Initiate Enemy Attack function is replacing the random behaviour we had before.

With the help of my tutor, I started working on the implementation of the boss's behaviour. After analysing what systems would be best (I thought about Behavior Trees or State Trees) I decided to go with my own custom

logic and blueprints. For the enemy command, I originally had a switch on integer that would randomly pick through a list of commands. Since I designed Kael's behavior, I created an *InitiateEnemyAttack* function where I could implement the logic inside.



In this function we can implement anything related to Kael.



In the Select Action function, we can determine specific behaviours for specific attacks.

At the moment, I am working on a *SelectAction* function and see how I can use this system and make the best of it.

Weekly Reflection:

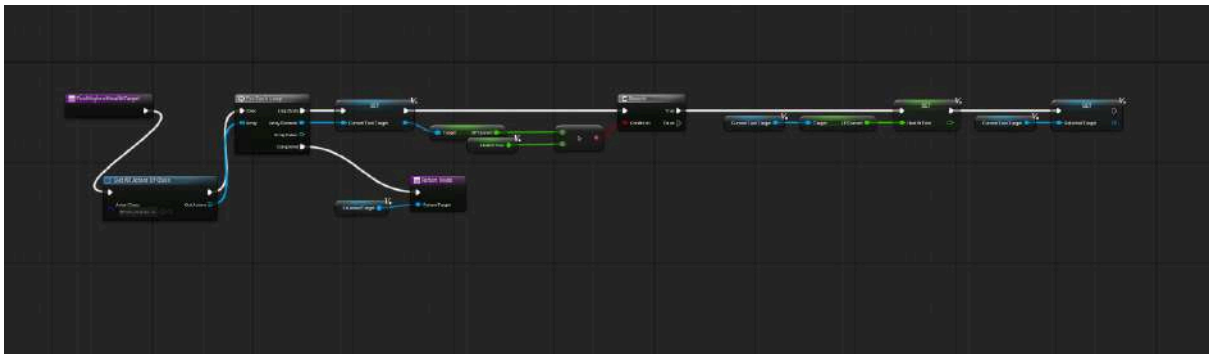
I feel like I am starting the hardest part of this: implementing the boss behavior. The mentorship feedback was not too bad, almost encouraging: we know what we have to focus on to finish our game...

WEEK 14 : Boss AI

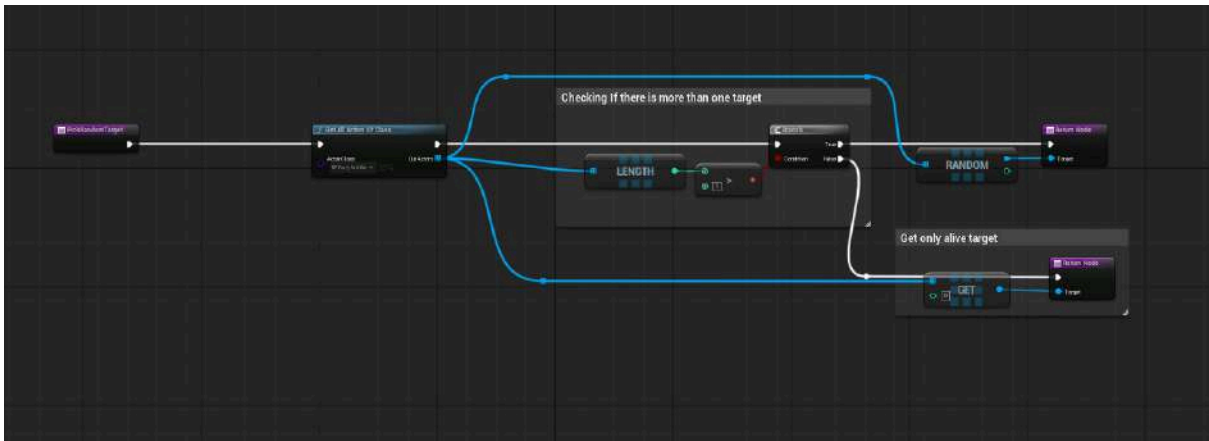
Weekly objective: The objective for this week is to fix the new build issue we are having right now (due to circular dependencies) and keep on implementing the boss AI.

Weekly Summary:

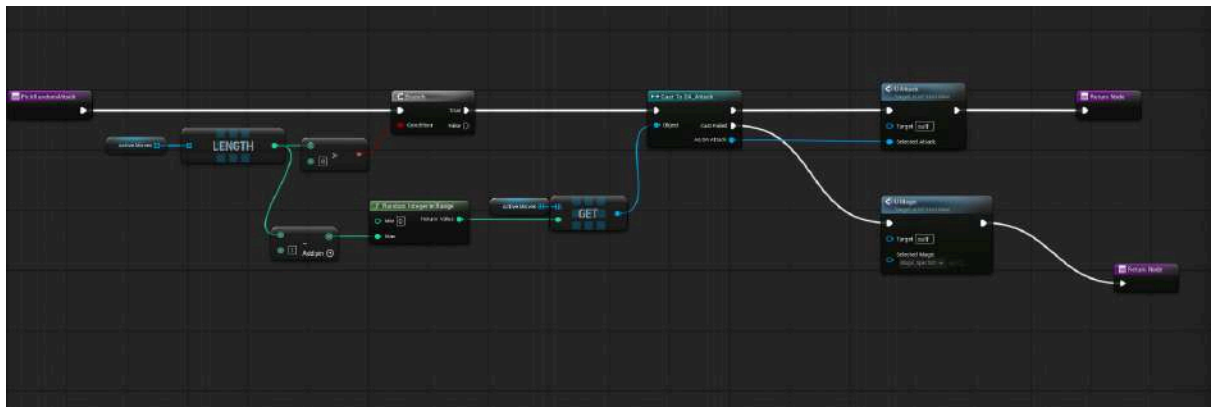
I am working on a bunch of functions that could allow the boss to have a specific behavior in some particular situations.



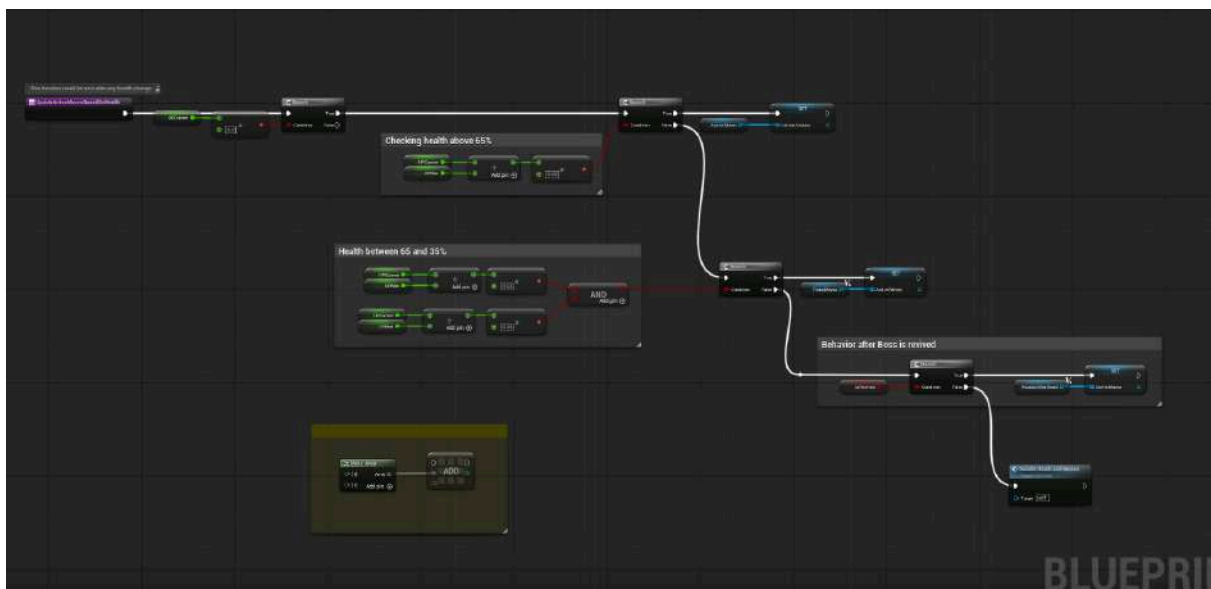
Following the feedback of the game being too hard and the enemy kept attacking Lucien, I implemented a *FindHighestHealthTarget* function where Kael is attacking the strongest unit.



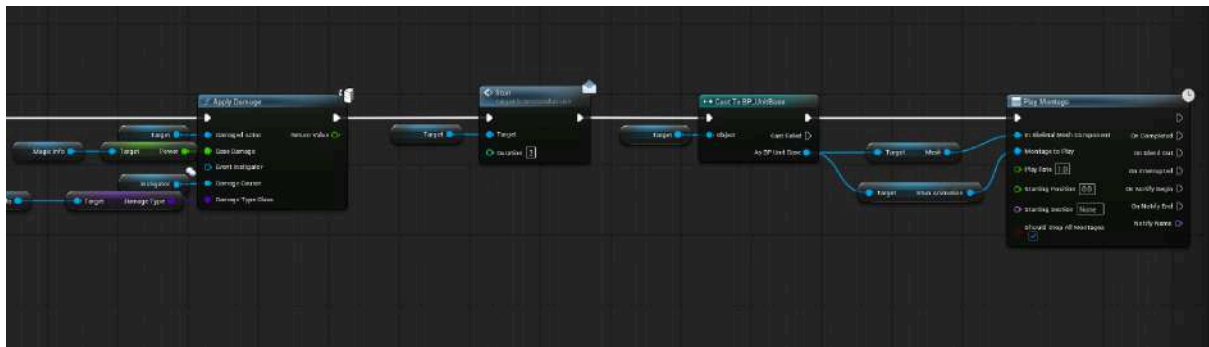
Here is the *PickRandomTarget* function that could be used in some specific circumstances.



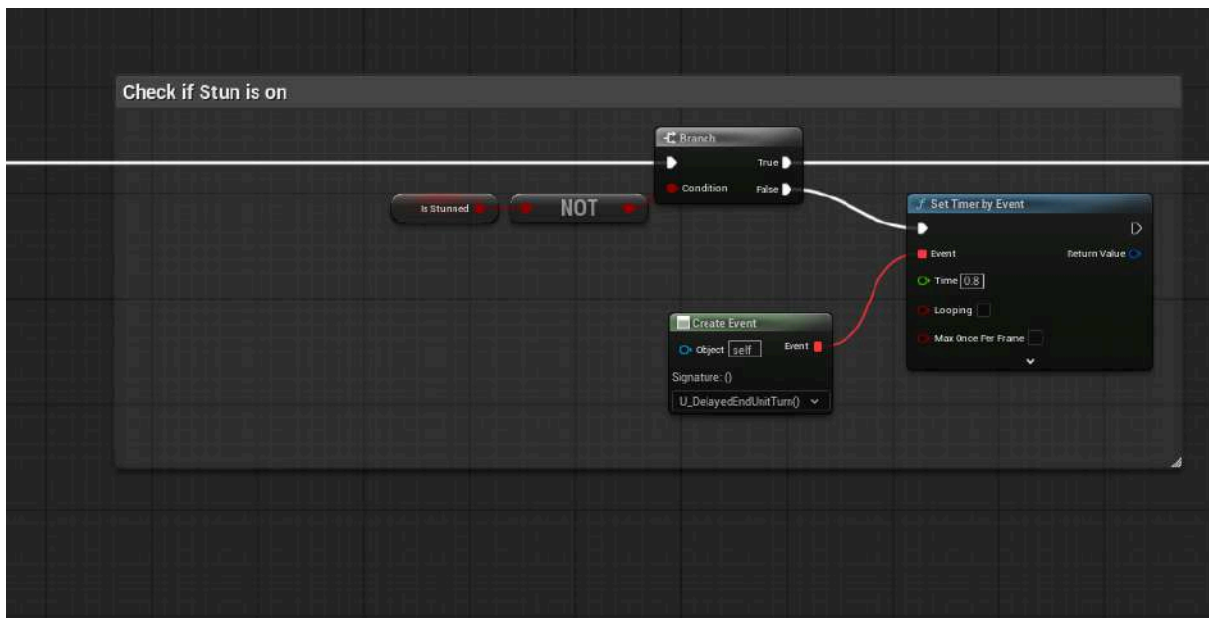
PickRandomAttack could also be used in some specific situations.



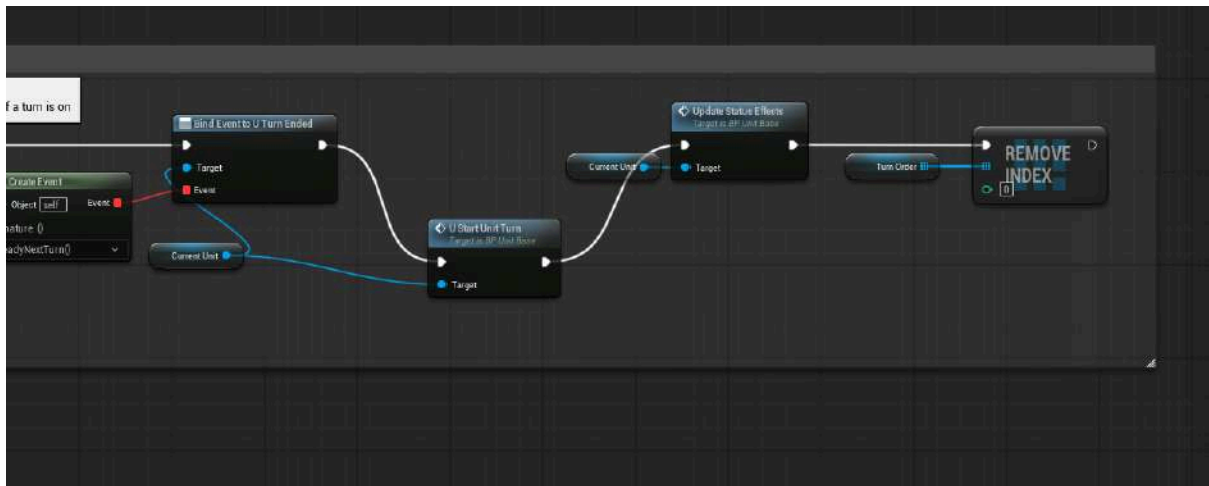
Here is the *UpdateActiveMovesBasedOnHealth* function where I am trying to have Kael using particular attacks or spells depending on his health. The target is to have him using more skills as his health goes low.



In the BP_ChainBindEffect, I am applying the damage then the Stun status for 2 turns, and playing the stun animation.



In the *StartUnitTurn* function I am checking if the Stun status is on. If that is the case, I am calling the *EndUnitTurn* function with a short delay to allow the cameras and and next unit abilities to load in properly.



In the BattleGameMode, I am calling the *StartUnitTurn* function after binding to *TurnEnded*.

Weekly Reflection: I spent some time this week with Isbah trying to fix the circular dependencies that were preventing us from making a build. We had to fix it the hard way: she basically re-created a new combat component actor. Anything that was originally related to the UnitBase has been re-implemented in the UnitBase. It is working now!

My work on boss AI is going slow..I know what I want to do, it is my technical limitations that prevent me from going faster.

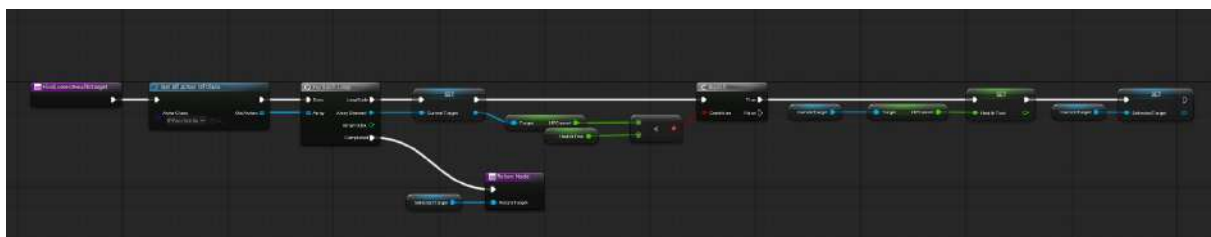
WEEK 15 : Catch up on Easter break + last implementation objectives

Weekly objective: Catching up on production logs, Individual contribution list and professional practice module (done during Easter break). This week I plan to keep on implementing the boss behaviour in game, now that we clearly designed who we want him to target and which moves he can use.

Weekly Summary:

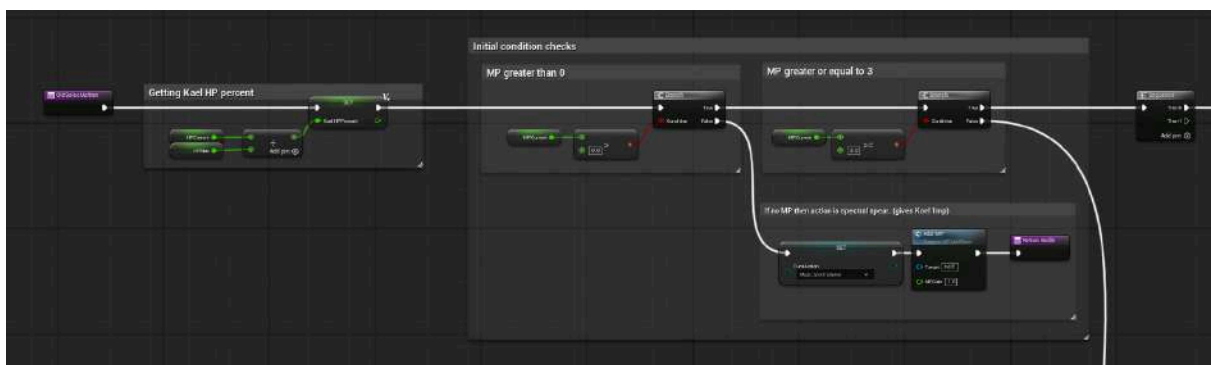
In order to do so, I decided to implement this logic in BP_Kael, instead of being originally set in BP_GameMode. Here is a list of functions made to achieve this.

This function aims Kael to target the party units with the lowest HP value.

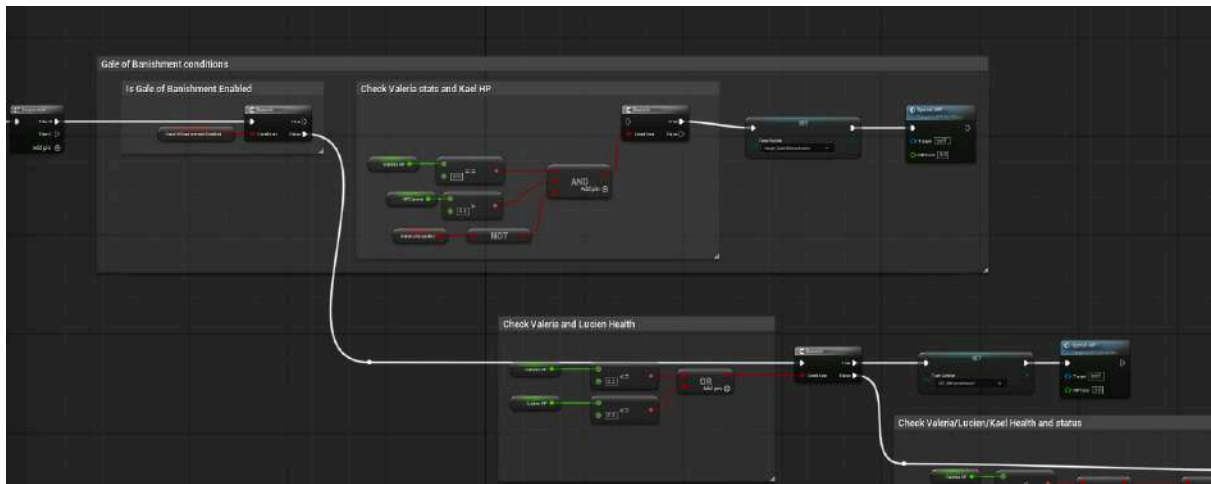


FindLowestHealthTarget

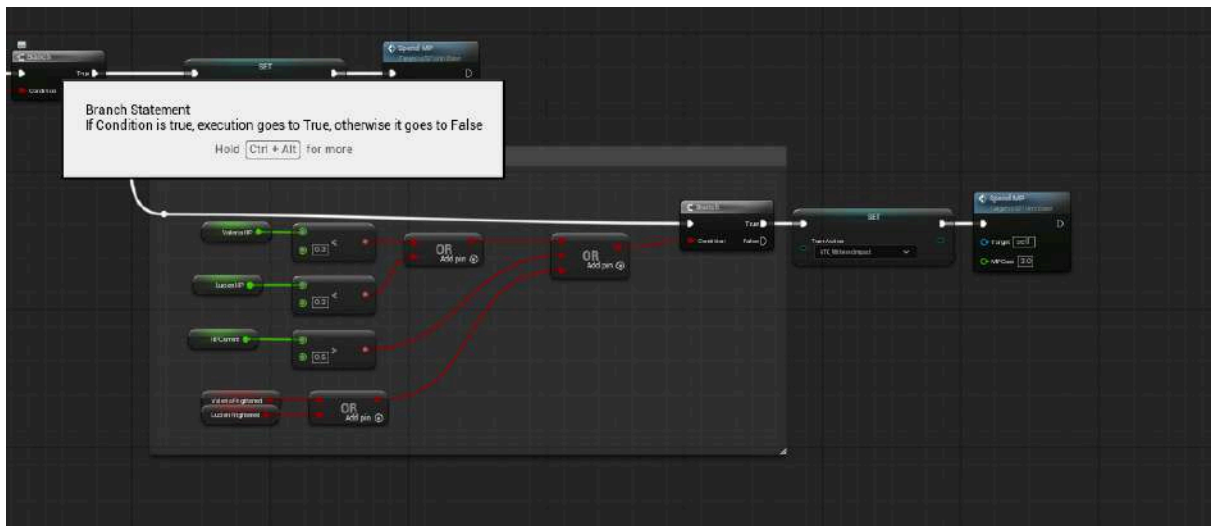
In SelectAction, we want to enable specific moves for Kael depending on his MP values.



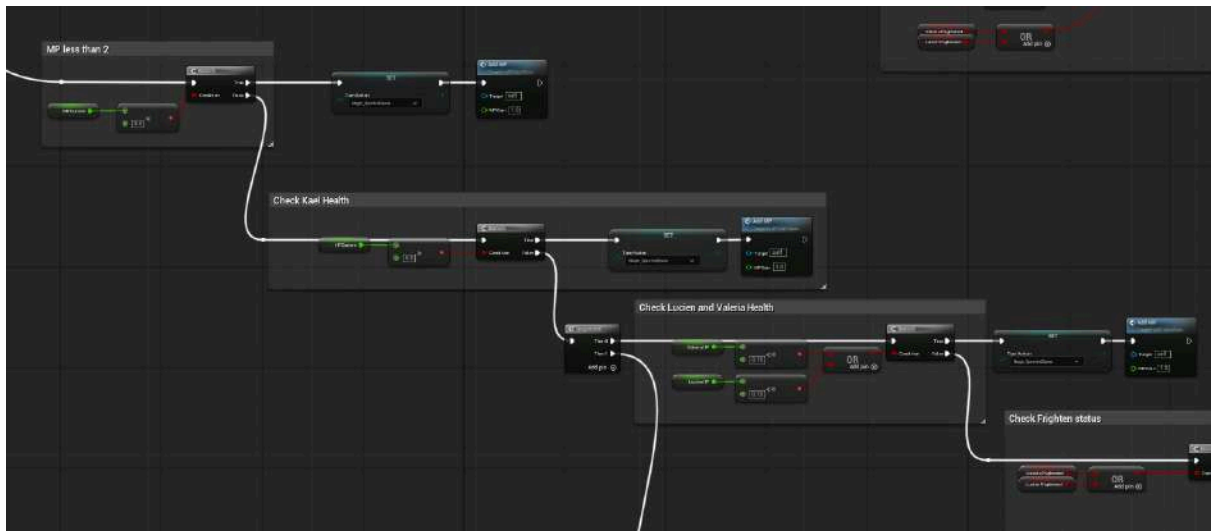
SelectAction (Part1)



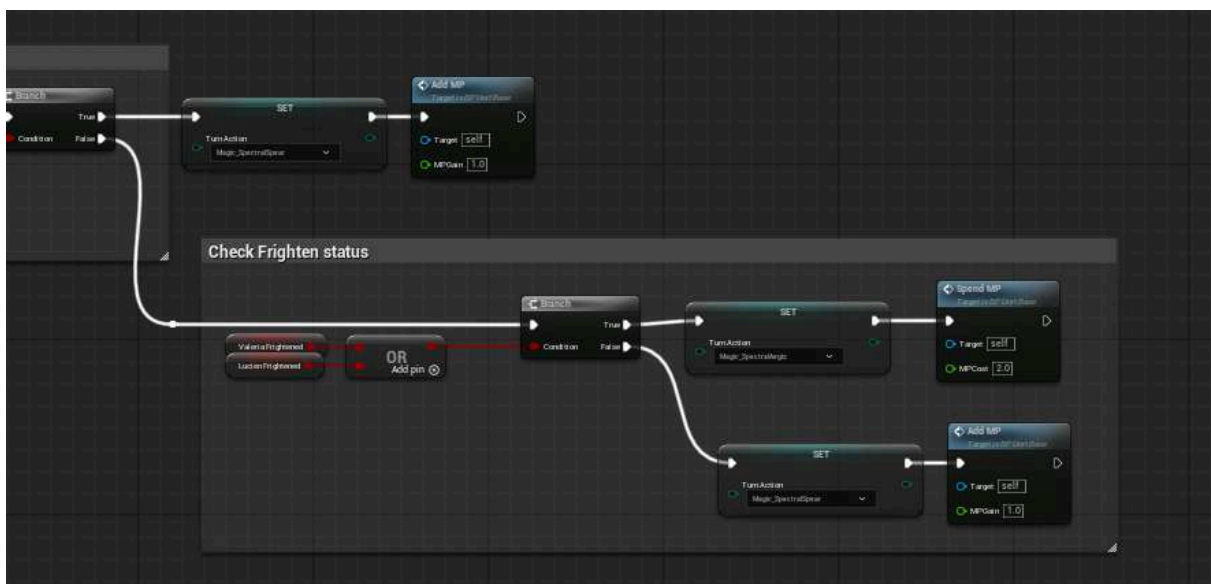
SelectAction (Part 2)



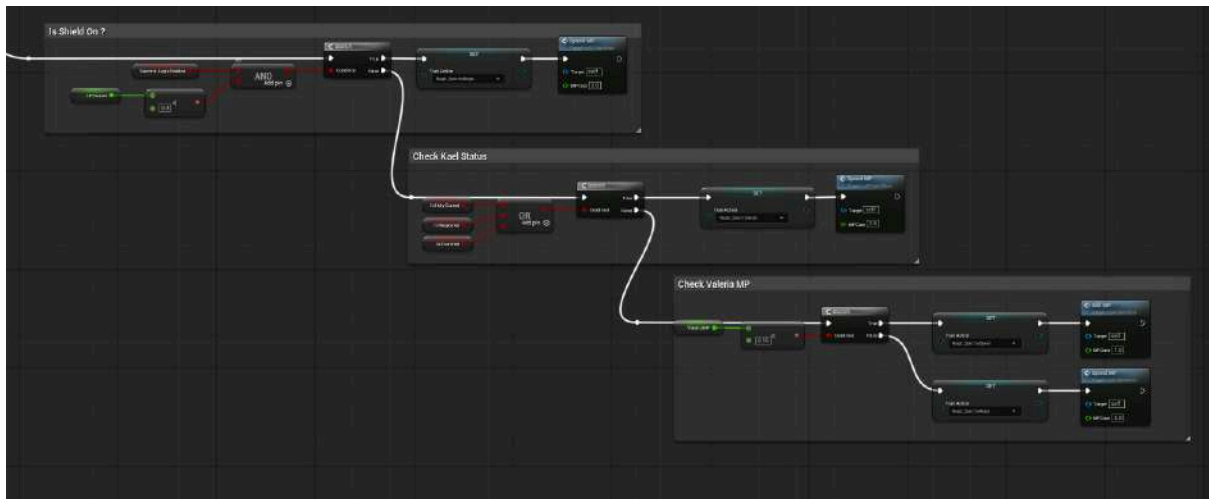
SelectAction (Part 3)



SelectAction (Part 4)



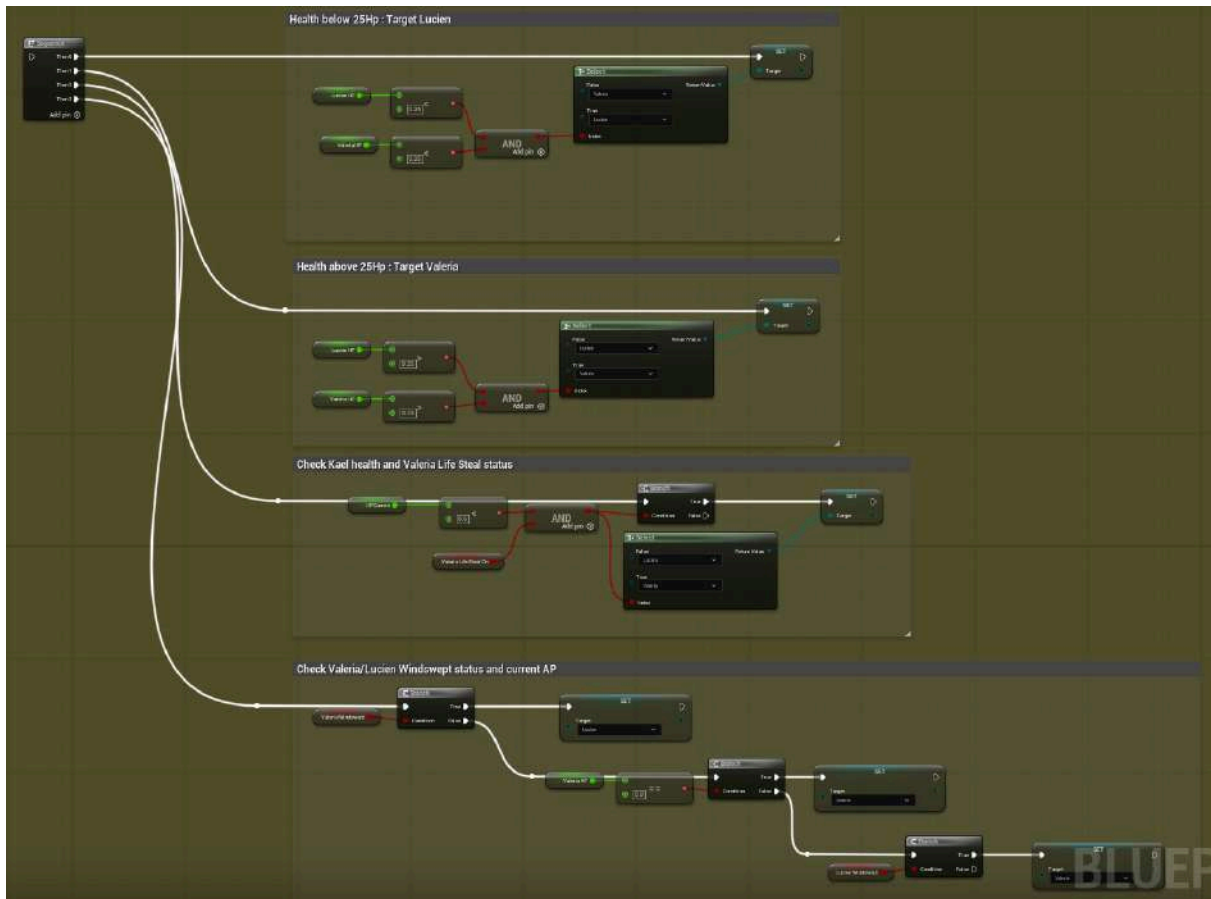
SelectAction (Part 5)



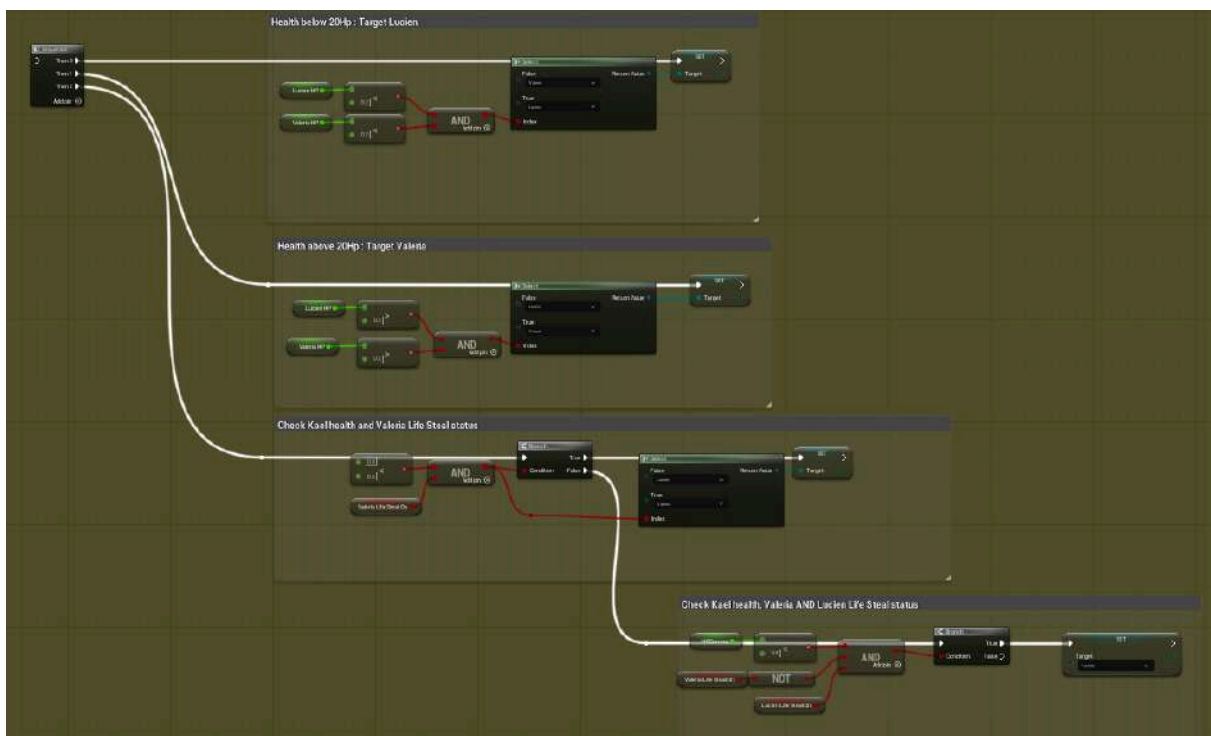
SelectAction (Part 6)

As notified in my individual contribution list, my teammate Isbah re-organised this SelectFunction (and the SelectTarget below) in order to make it more usable, making functions regrouping to avoid repeating patterns.

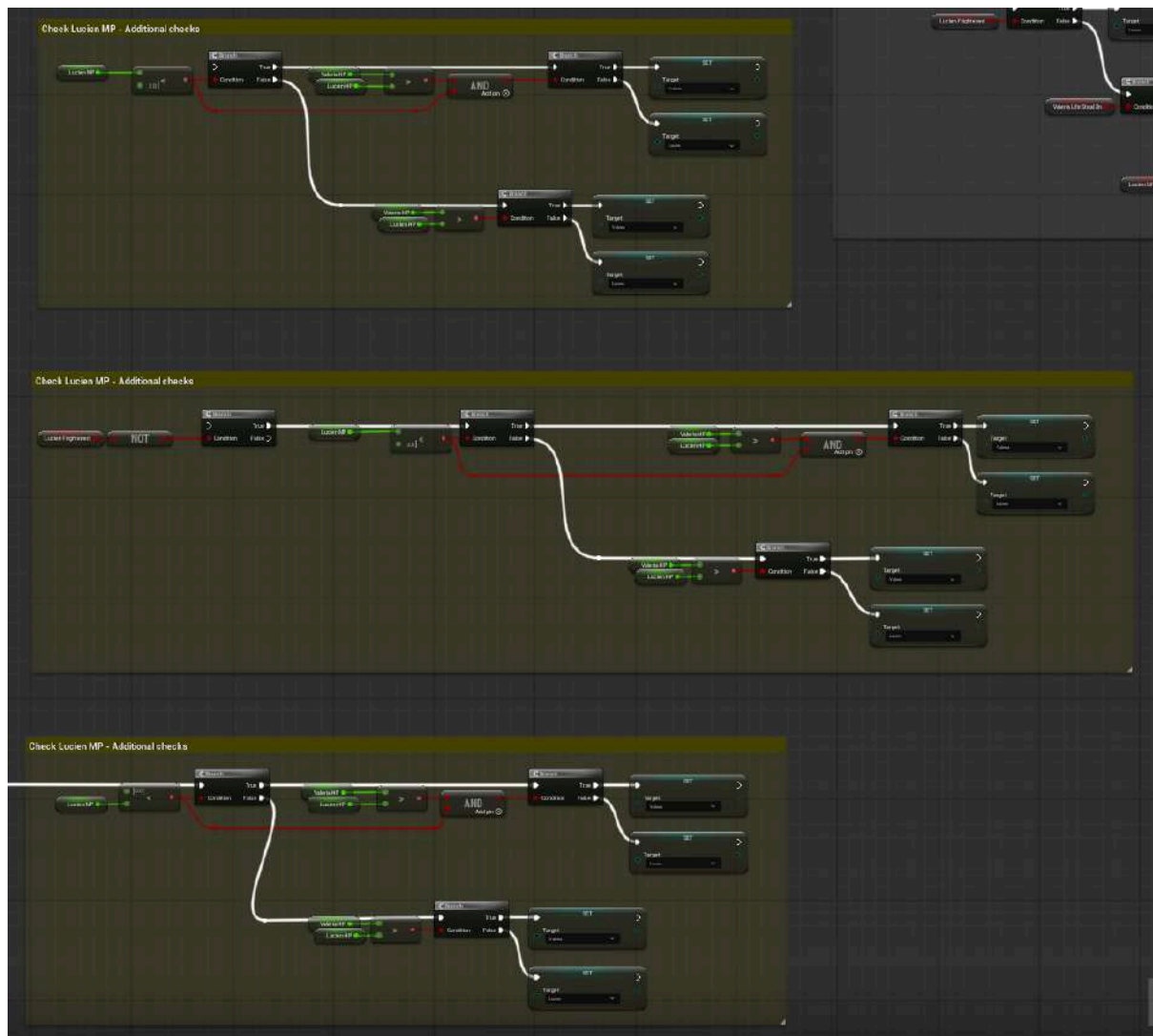
With the SelectTarget function, I want to tell Kael who to target (for each move) between Lucien or Valeria, depending on several variables: Party Units HP values, what statuses and if they are applied or not.



SelectTarget (Gale Of Banishment move)



SelectTarget (Withered Impact move)



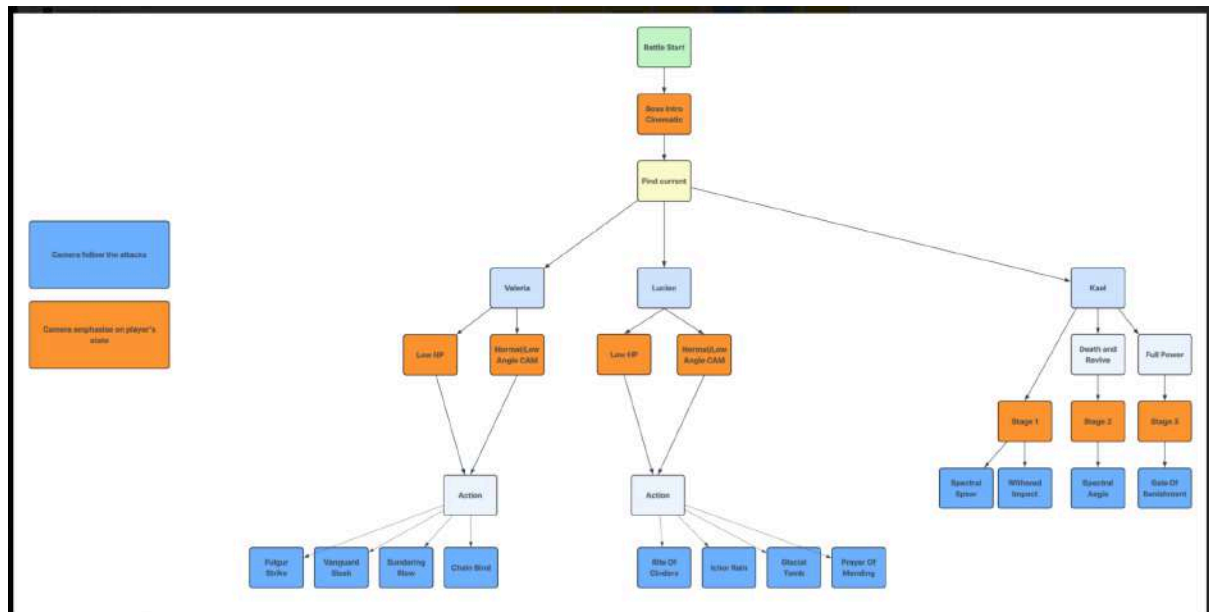
SelectTarget (Additional checks)

Following the feedback of my tutor, it is time to work on some camera movements to bring some punch during the gameplay. I have looked at two games in my research *Metafor ReFantasio* (Atlus, 2024) and *Expedition 33: Clair Obscur* (Sandfall Interactive, 2025). Both games use a turn-based combat system, but with different camera uses.

What inspired me from *Metafor ReFantasio* is the way they build up intensity before a decisive battle: I found their camera movements quite classic but very efficient (long range shot, then getting closer to the enemy with some dialogue).

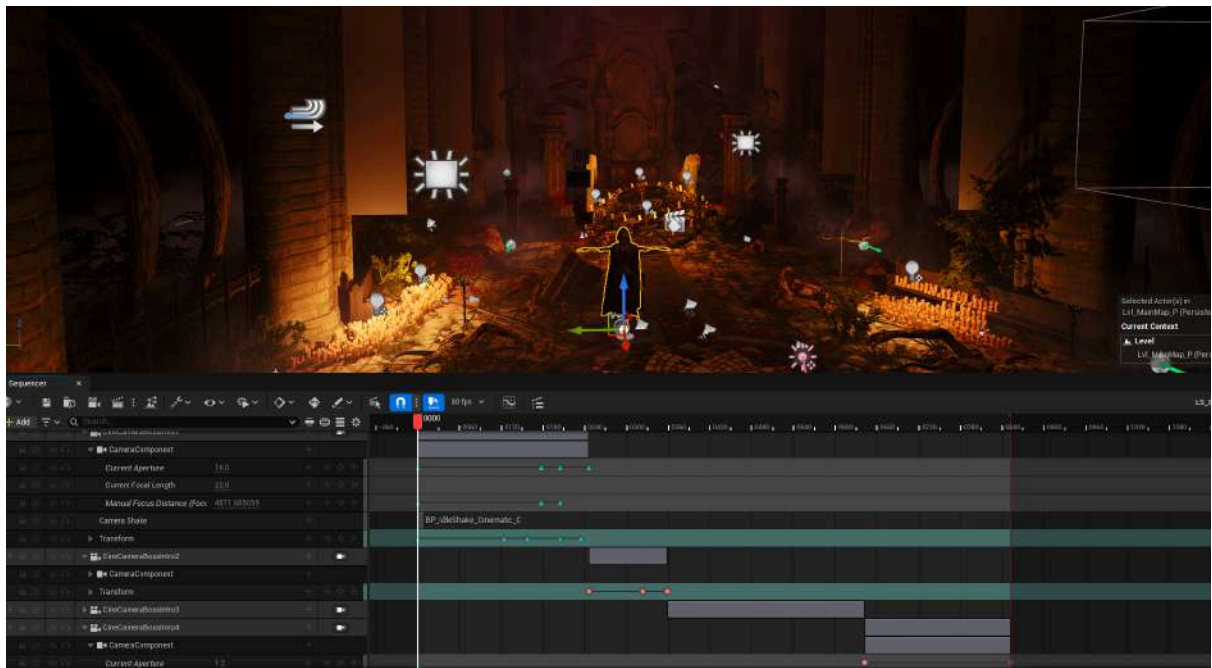
On the other hand, Expedition 33: Clair Obscur is all about punch and vivacity. Their camera shots during gameplay are short, fast, very responsive and dynamic.

But before working on the actual sequences, I made of plan of what I would need:

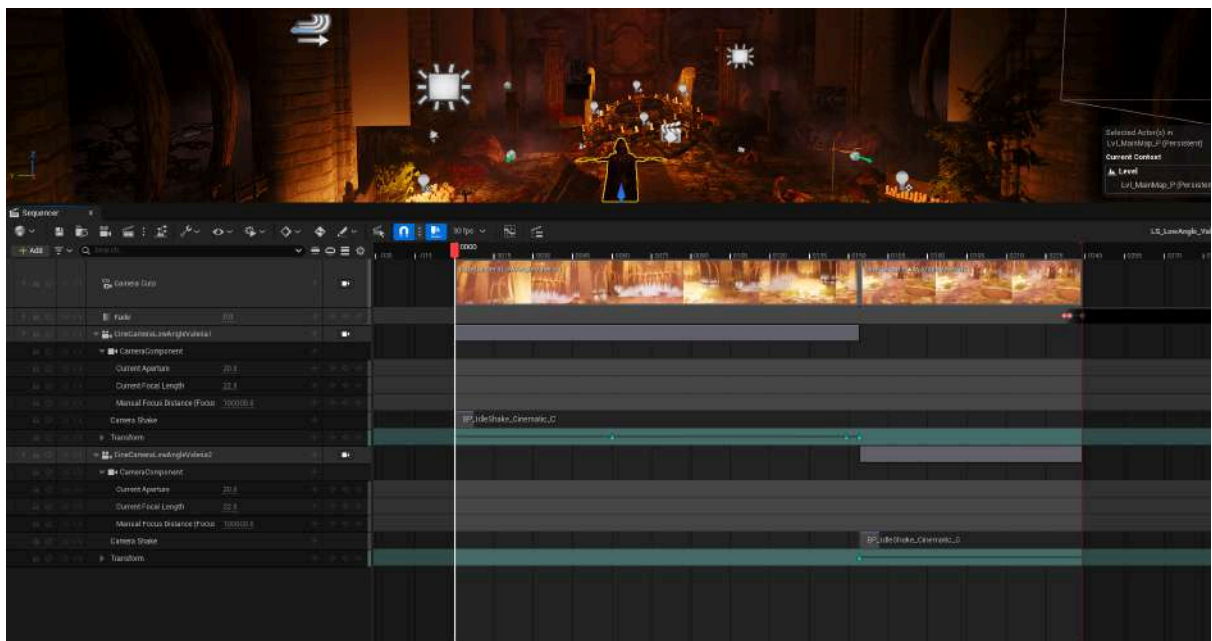


Camera plan

And below you can see different pictures of level sequences I have worked on this week:



BossIntro cinematic



Valeria Low Angle

Weekly Reflection: It has been a busy week. I am quite happy with work accomplished regarding the boss logic during the combat (even though it had been simplified and more accurate by Isbah). I am also happy with the sequences, I think it is going to bring a lot to the game.

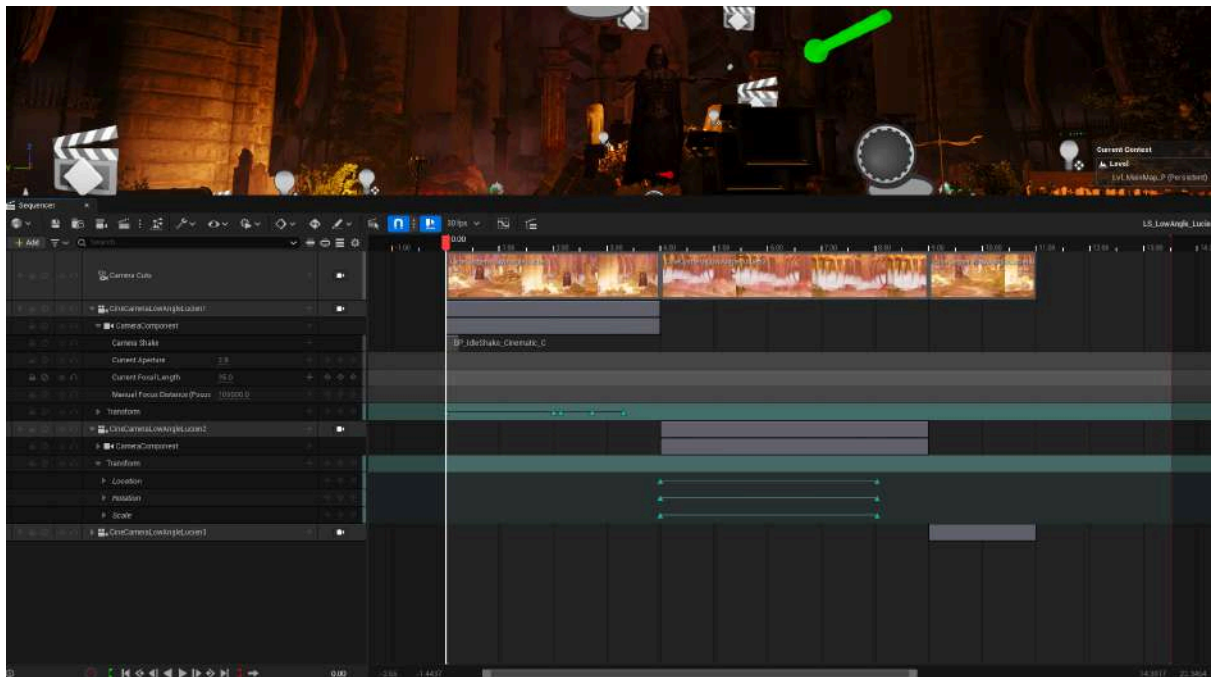
My biggest regret is the lack of playtest we have been doing during the project: we (I) had the game not really working or with so little to test for too long. Now that we are getting closer to the submission, it really pains me that we were that late on the project.

WEEK 16 : Working on attack camera movements

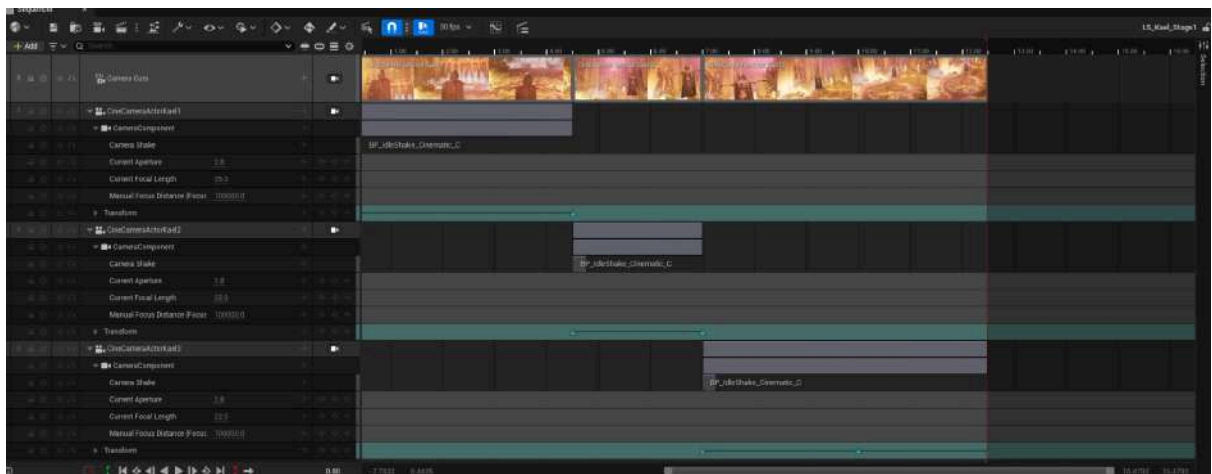
Weekly objective: What I am aiming to achieve for this week is to finalize all camera shots and sequences for the game.

Weekly Summary:

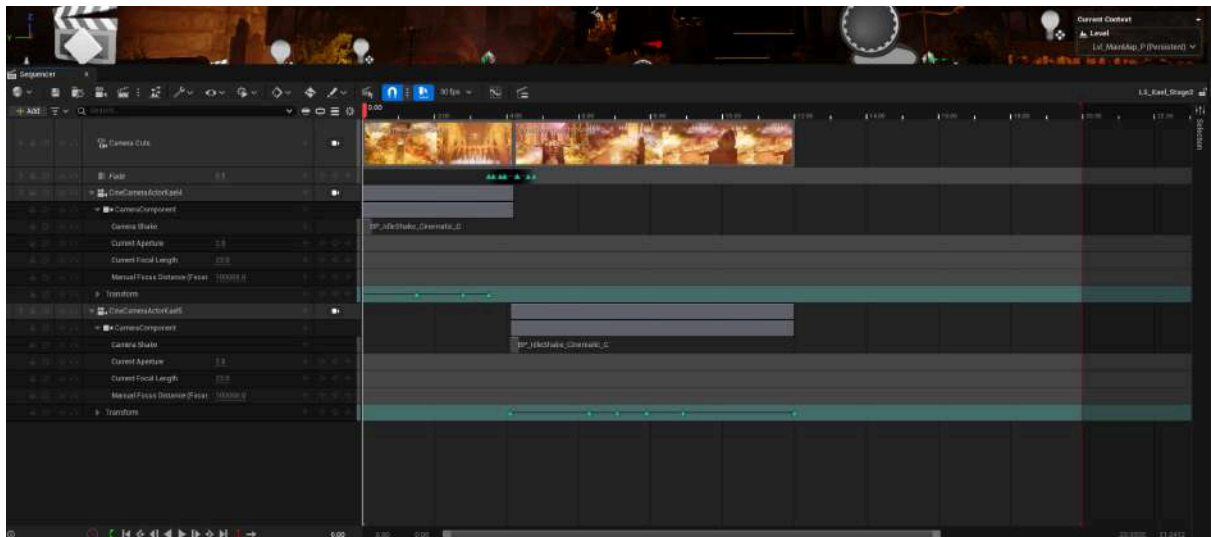
You can find below pictures on the progress I have made on the sequences:



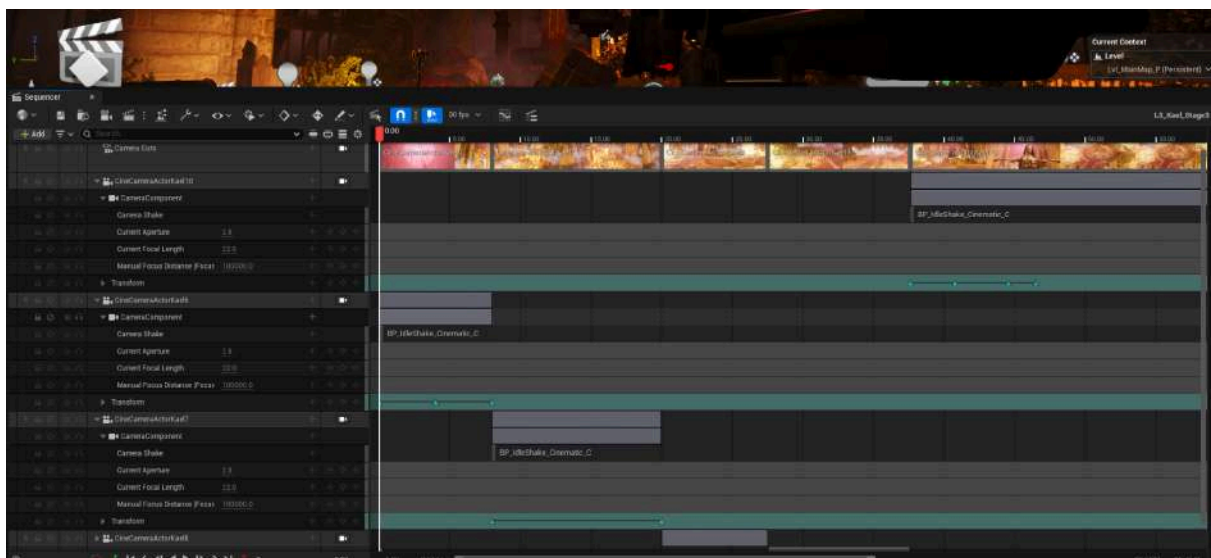
Lucien Low Angle



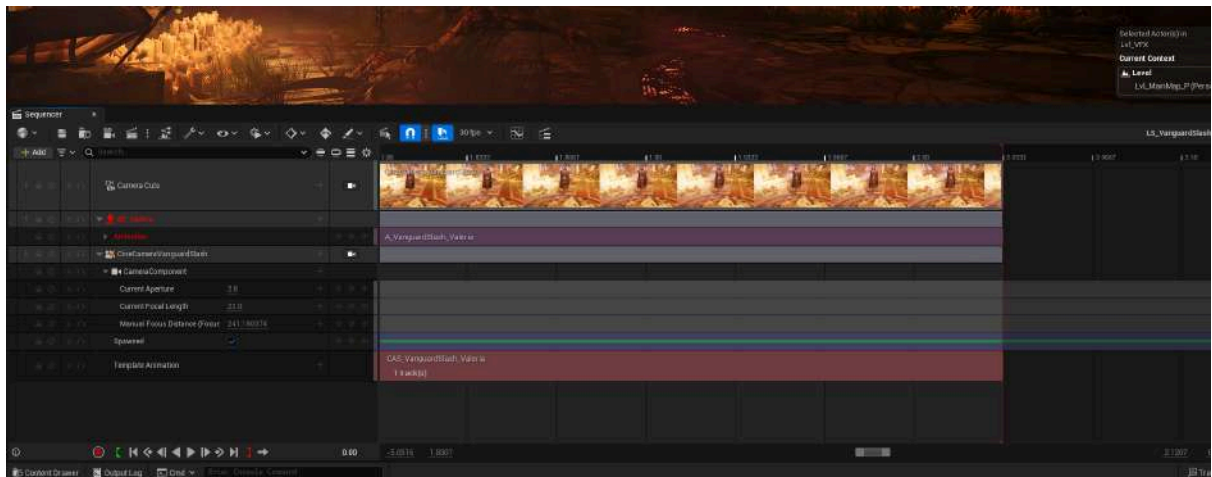
Kael Stage 1 (Before his first attack)



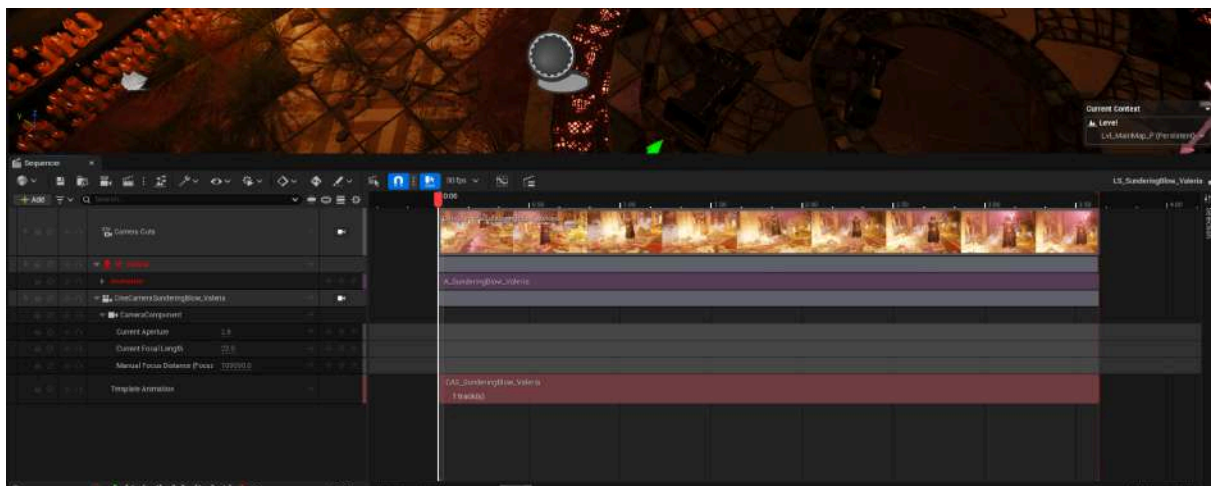
Kael Stage 2 (when 65% HP, shield activated)



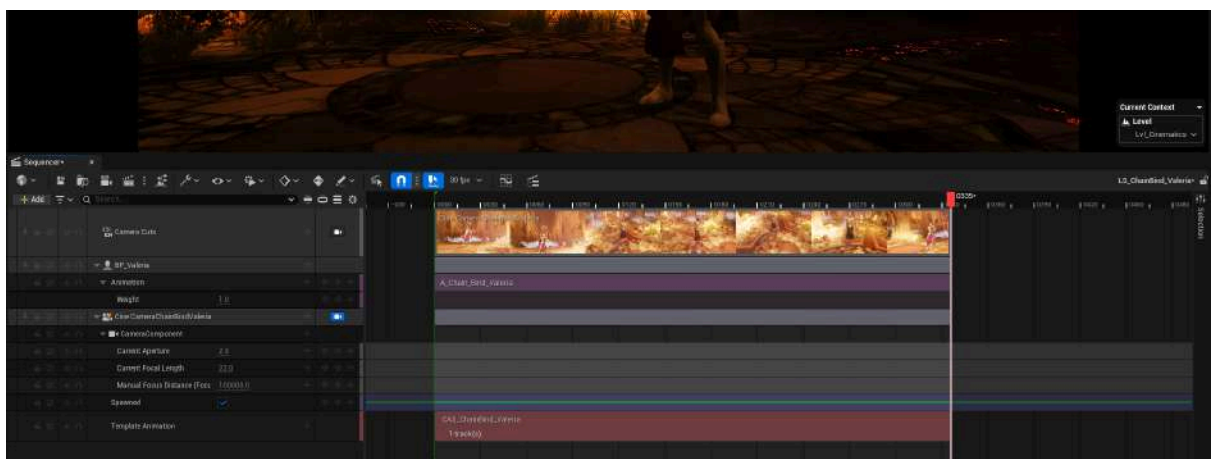
Kael Stage 3 (when 35% HP, Gale Of Banishment activated)



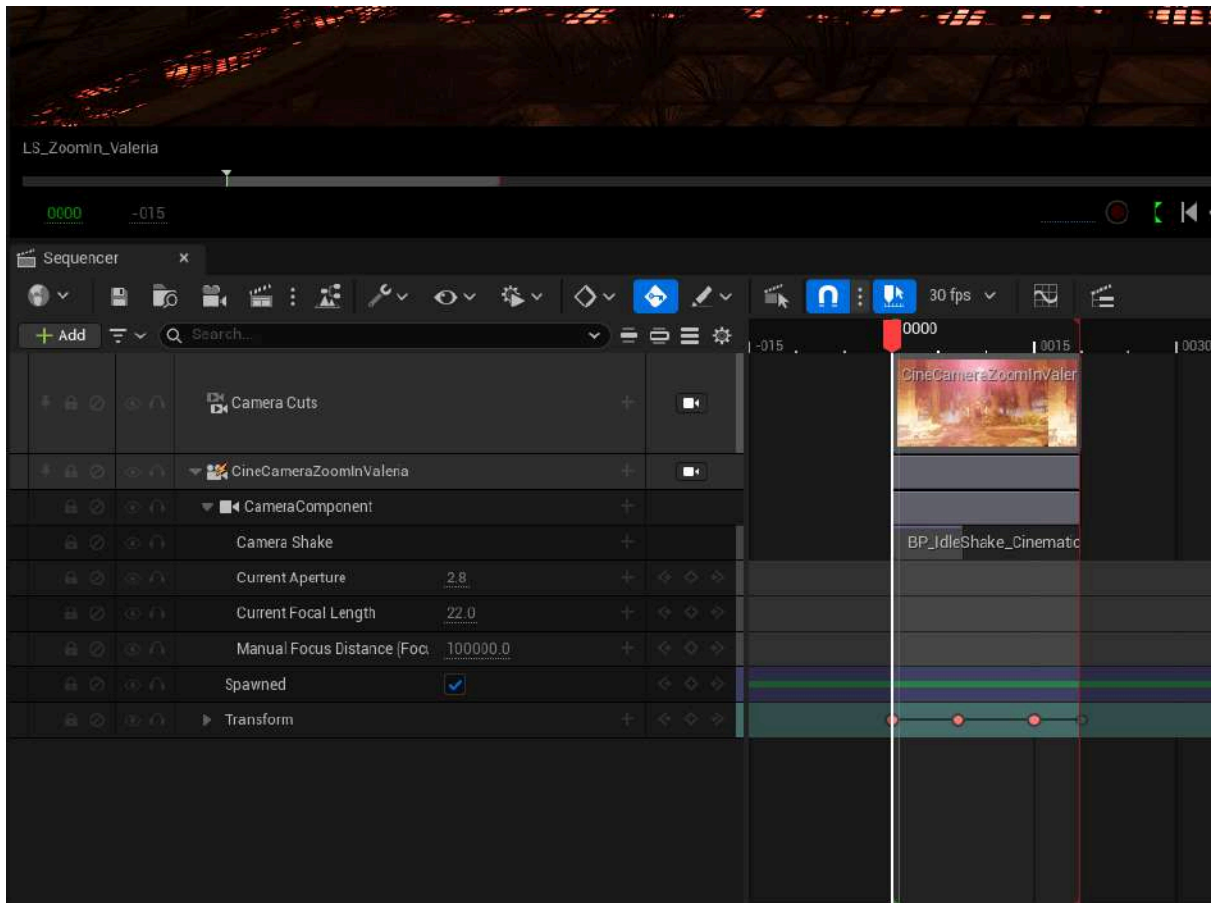
Valeria VanguardSlash camera movement



Valeria SunderingBlow camera movement

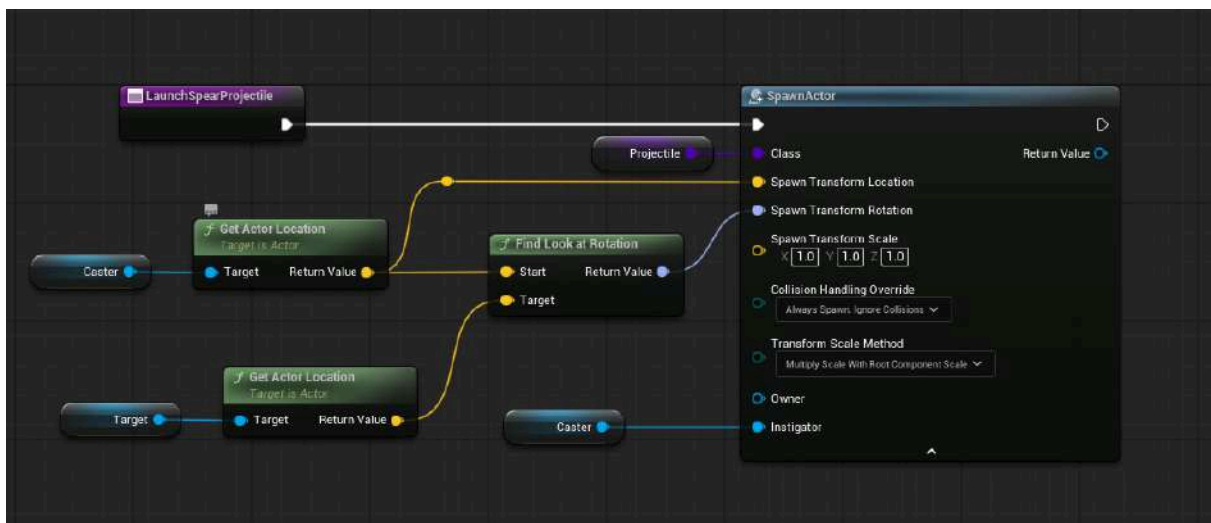


Chain Bind Camera Mov



ZoomIn Valeria

Since we finally had the final spear in the game, I needed to implement its functionality differently than before.

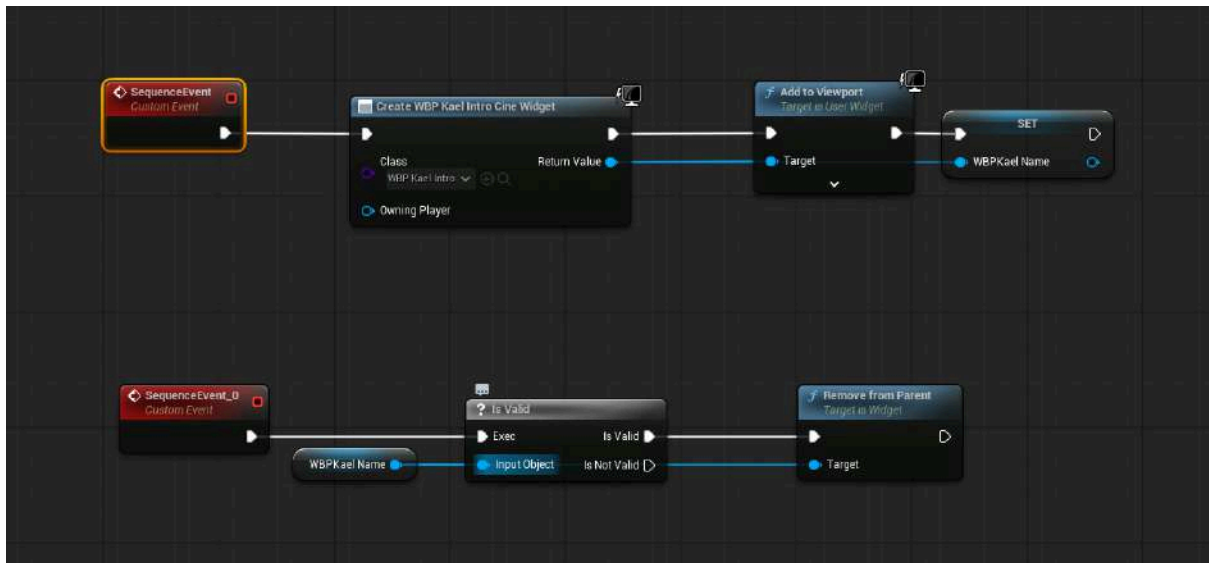


LaunchSpearProjectile

I also attached the Spear on Kael's Skeleton:



I created a widget with small animation containing the name of the enemy, and have it coming during the BossIntro cinematic, through an event in the level sequence:



The animation team were not really satisfied with hands positioning on Valeria's skeleton, so they decided to correct this so I could attach the sword properly.



Sword being repositioned in BP_Valeria

Weekly Reflection: It has been an intense week, a lot has been done but some tensions have appeared within the team. It is true that we do not have much regarding the characters, and I can understand the frustration. It looks like it was too much to handle for one of the animators and a strong argument occurred between him and Razvan, our character creation artist. It was not constructive feedback and apologised shortly after, but that gives an idea of the stress we are enduring at the moment.

WEEK 17 : Implementing Cinematics in game and final tweaks

Weekly objective: In this final week, I am aiming to implement cinematic in game with audio and text, and make some final tweaks on animations.

Weekly Summary:

Here is an example of the scripts I have written for the cinematics. The idea is to bring more narrative, set the personality of the characters so the player is having more context when playing the game and to build up tension.

DIALOGUES

Boss Intro cinematic

Kael

The living usually have the courtesy to die before entering my sanctuary. You are... an untidy exception.

Kael

Lucien... the boy who plays with shadows. And Valeria... the girl who thinks steel can stop the inevitable.

Valeria

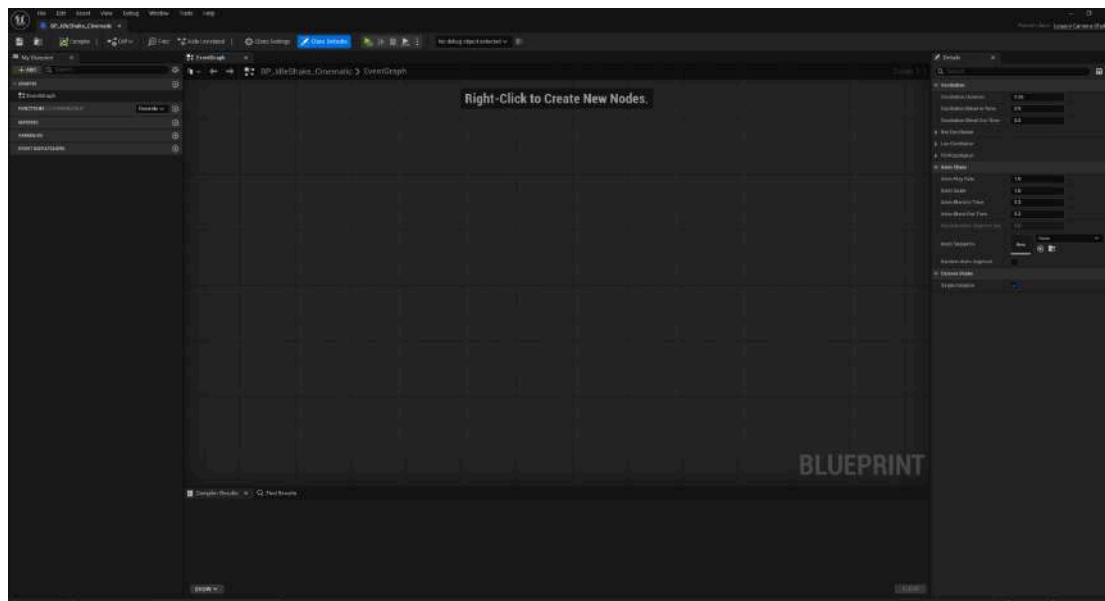
Save the poetry, Kael. We aren't here for your hospitality.

Valeria Low Angle

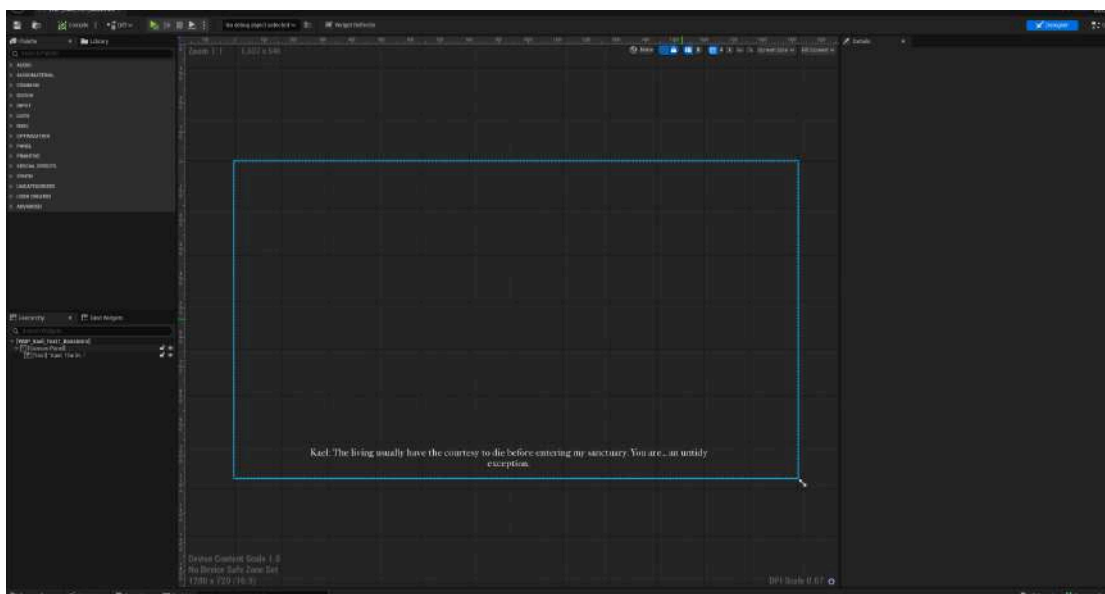
Valeria

We've tracked the rot back to this nest. It ends here, or we bury you under it.

I have created this IdleShake camera behaviour, slightly different from the one used in game. The values are increased, to give the idea someone is holding the camera while filming.



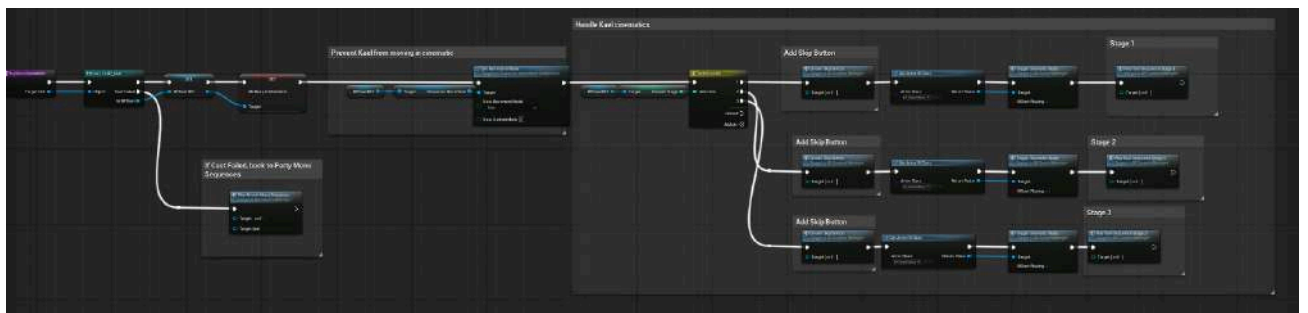
When it comes to adding text during the cinematic, I chose to do this the easiest way possible for 2 reasons: time is running short, and there are just few lines that need to be shown during the cinematics. It is simply showing through a widget, activated or removed within the level sequence blueprint editor, attached to keyframes.



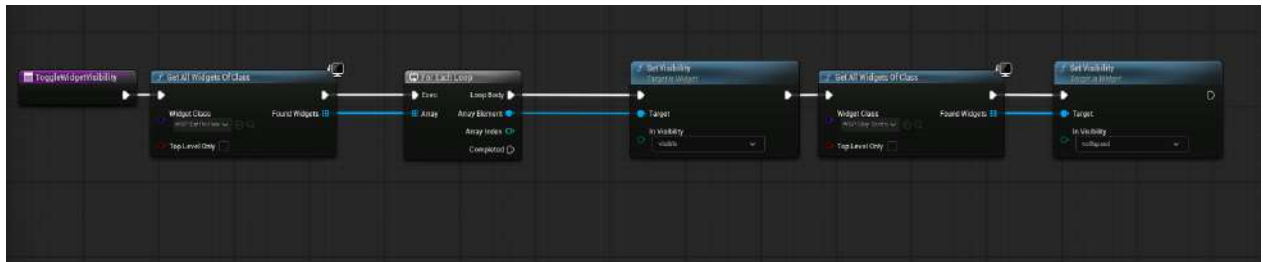
Here is an example of a level sequence blueprint editor, where I am activating or removing the desired text included in the widget.



I have been working on this function for some time, PlayBossCinematics. It plays the different cinematics depending on the boss current stage, adding the skip button on the screen and playing the dedicated music.



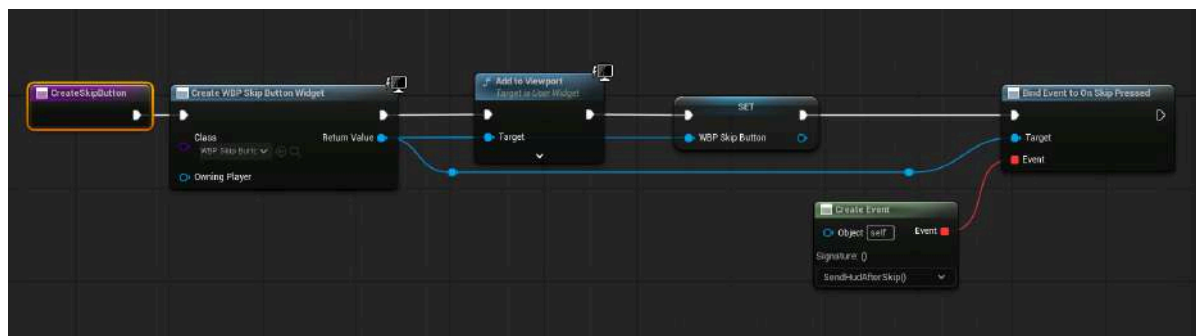
ToggleWidgetVisibility is a simple function that allows me to hide the HUD and how the skip button. Just to avoid repeating the same line of code too often.



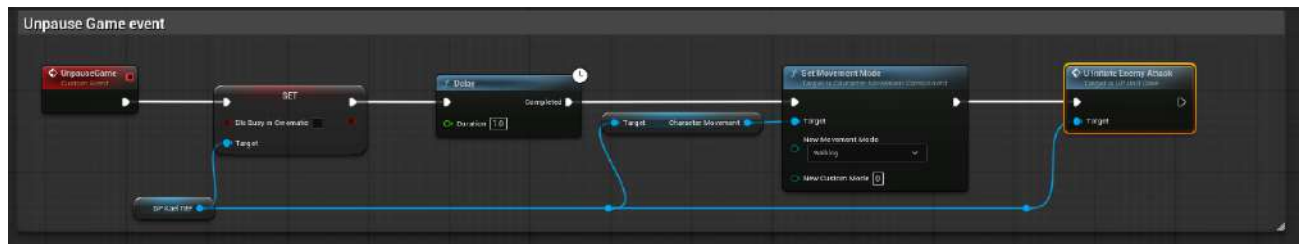
PlayKaelSequences is the function where I am creating the level sequence player, but also include custom events: ShowHudAfterStage and Unpausegame.



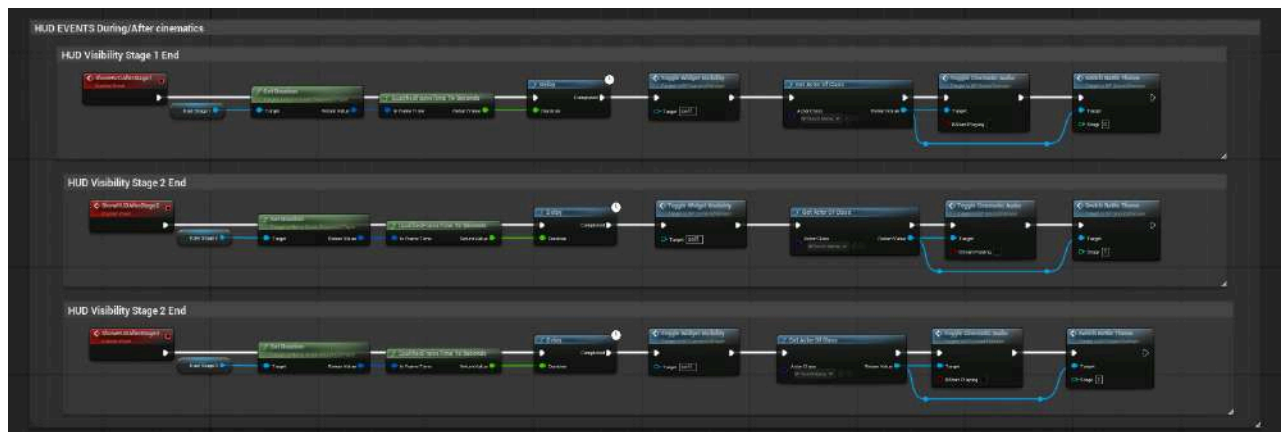
The CreateSkipButton function allows me to call the Skip button widget and call back the HUD on screen when it is pressed.



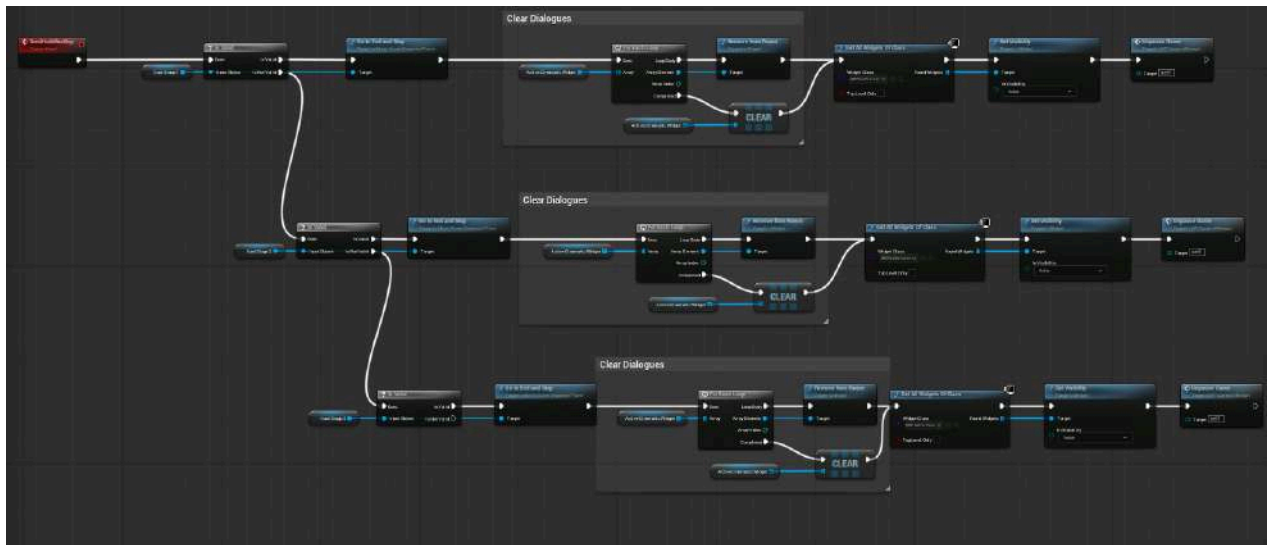
The UnpauseGame custom event sets Kael BusyInCinematics variable to false, allowing him to move and attack again.



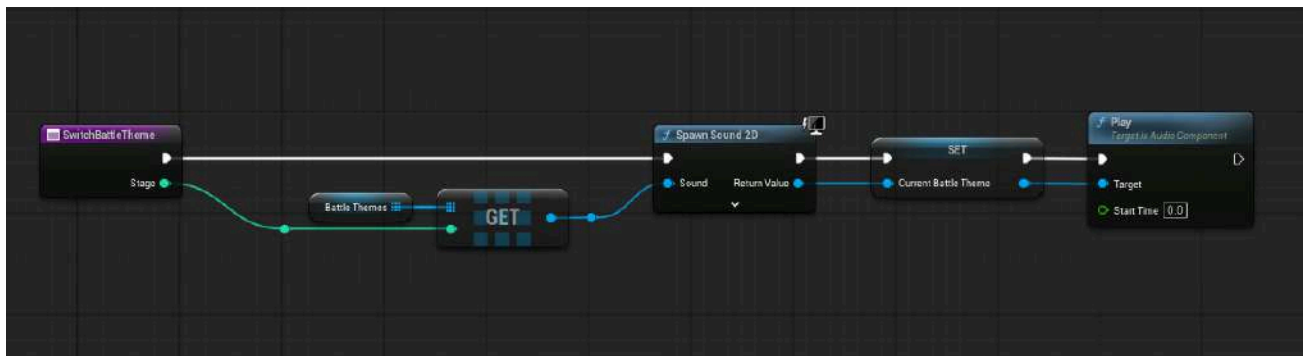
ShowHUDAfterStage is hiding the main HUD when the cinematic is on, it calls the ToggleWidgetVisibility, ToggleCinematicAudio and SwitchBattleTheme functions.



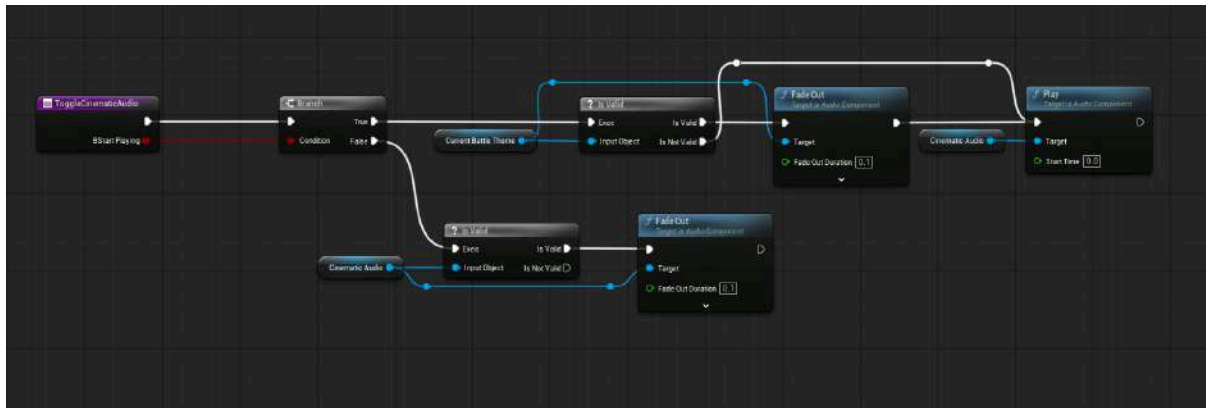
SendHudAfterSkip contains different behaviour that needs to be handled once you press skip the event: it is ending the cinematic, clearing all dialogues, activating back the main HUD and calling the UnPauseGame custom event.



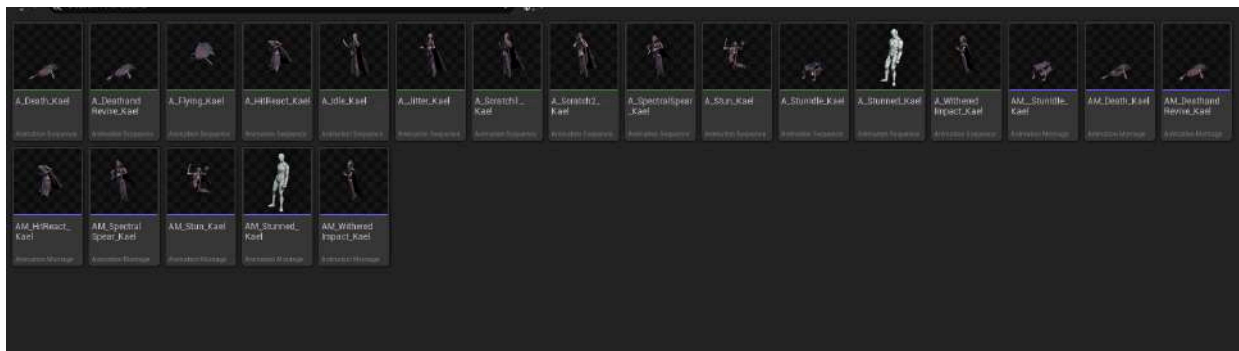
SwitchBattleTheme function checks the current boss stage and plays the dedicated music theme.



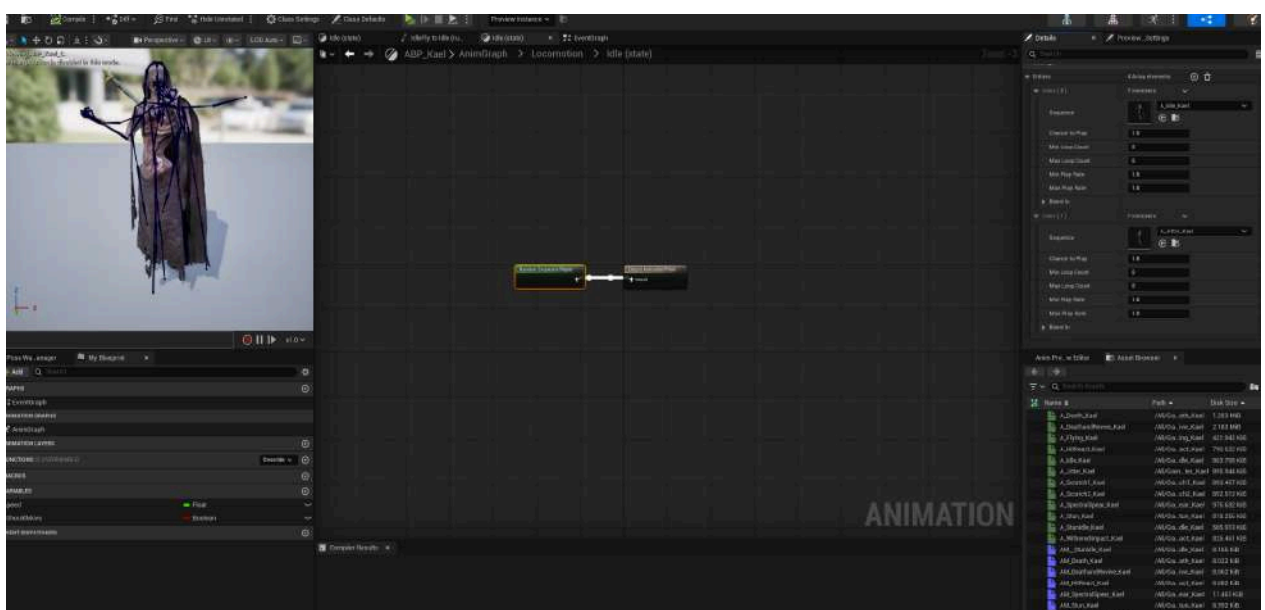
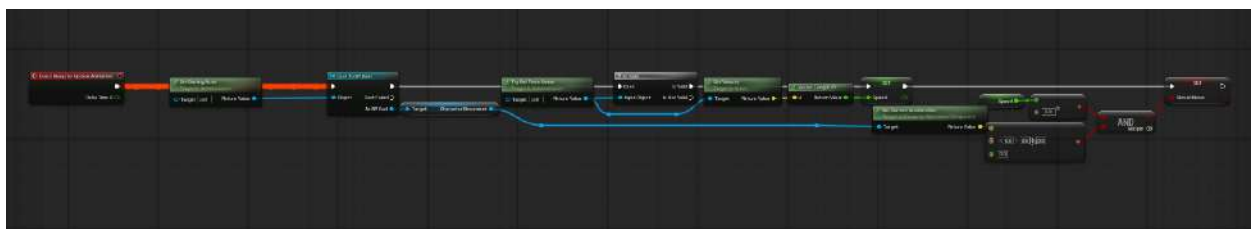
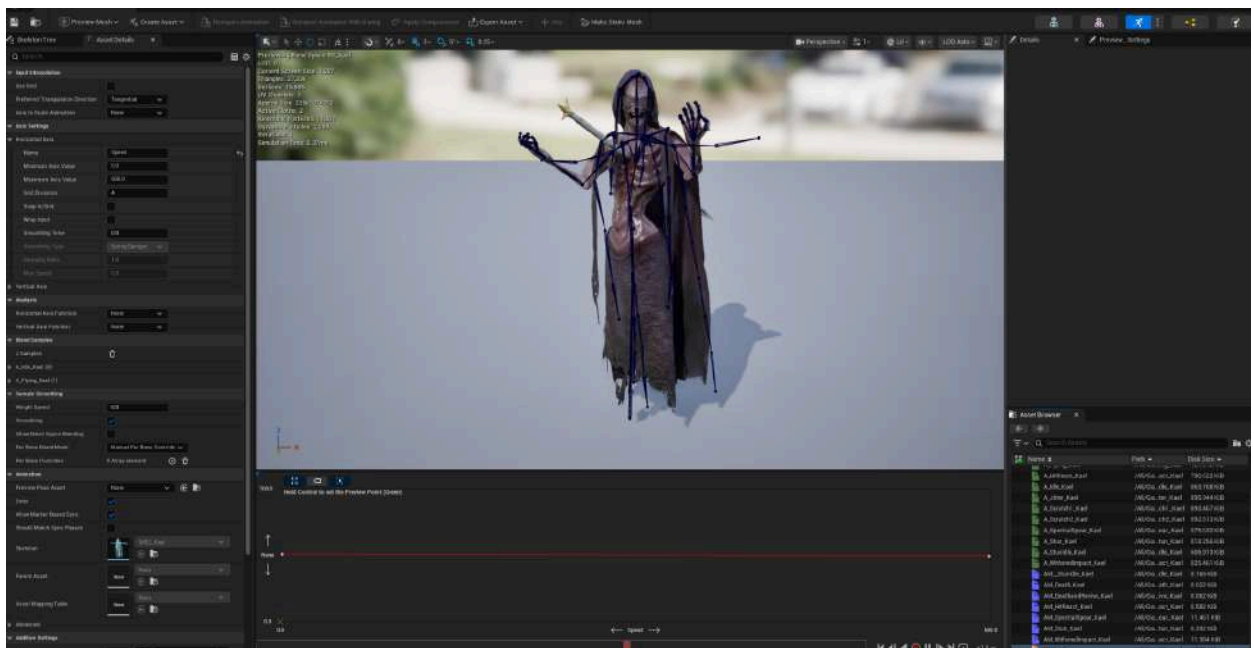
ToggleCinematicAudio checks if a current battle theme is playing, and then it plays the correct cinematic audio.



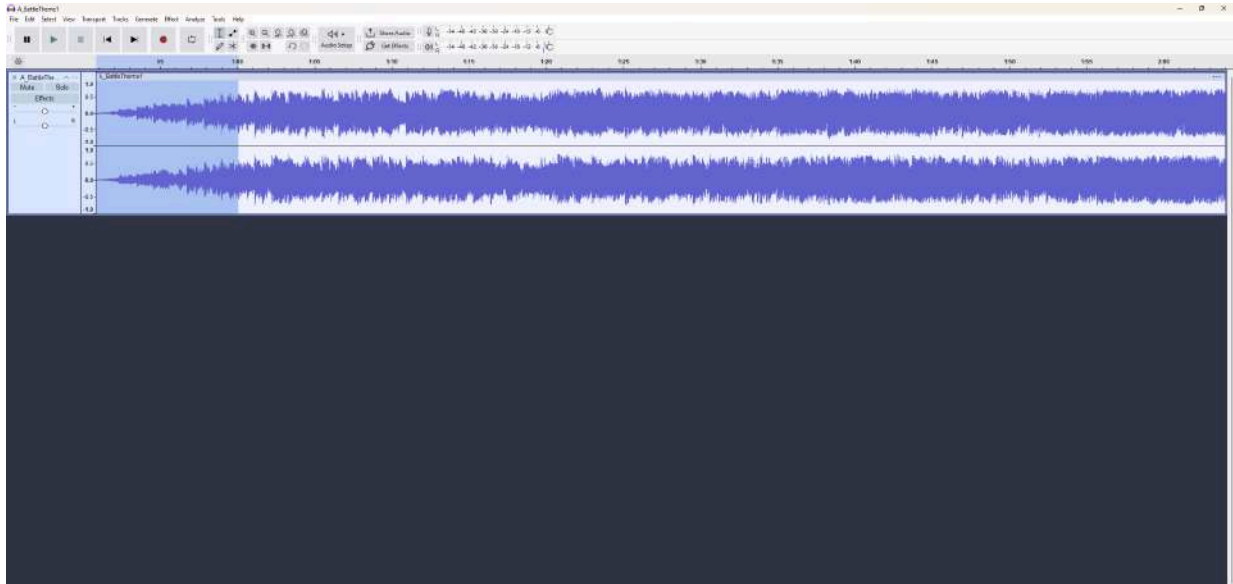
This week I also spent much time with the animators as they flooded me with the latest animations, updates on old ones. They were sending their updates on a shared drive so it was my task to import each one of these tests in the project and testing them in game. And since we started the project.



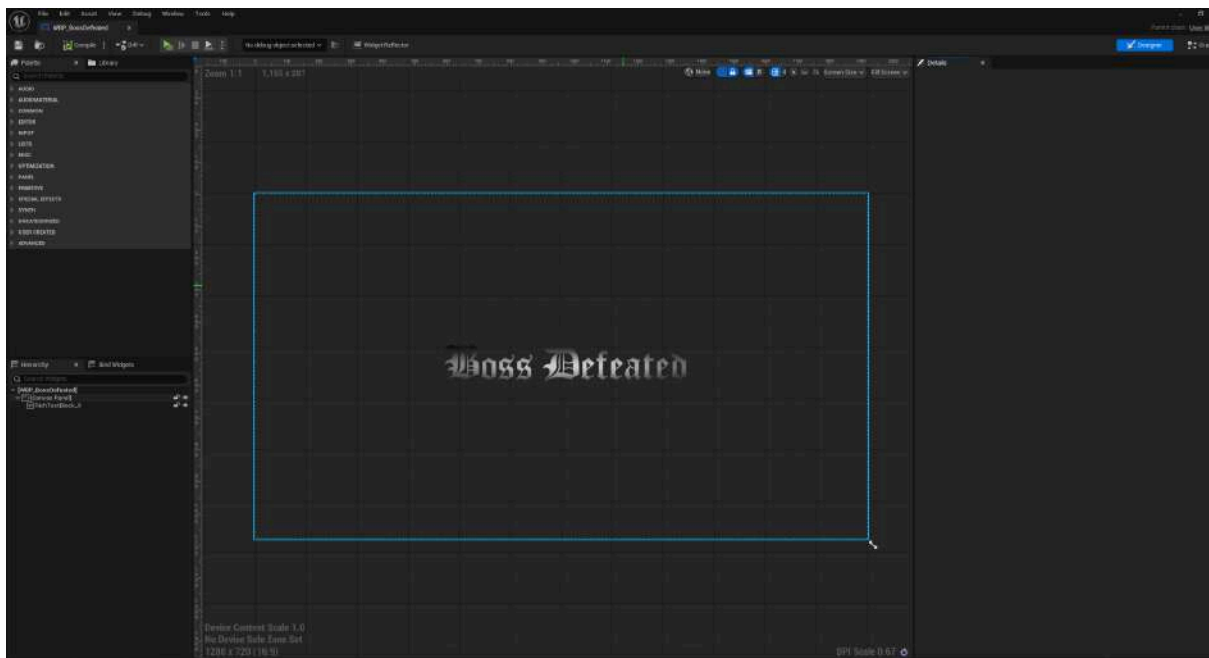
Below are a few pictures of me setting up all the systems to have the animations in game, from animation blueprints, blend spaces, or blending animations in state machines.

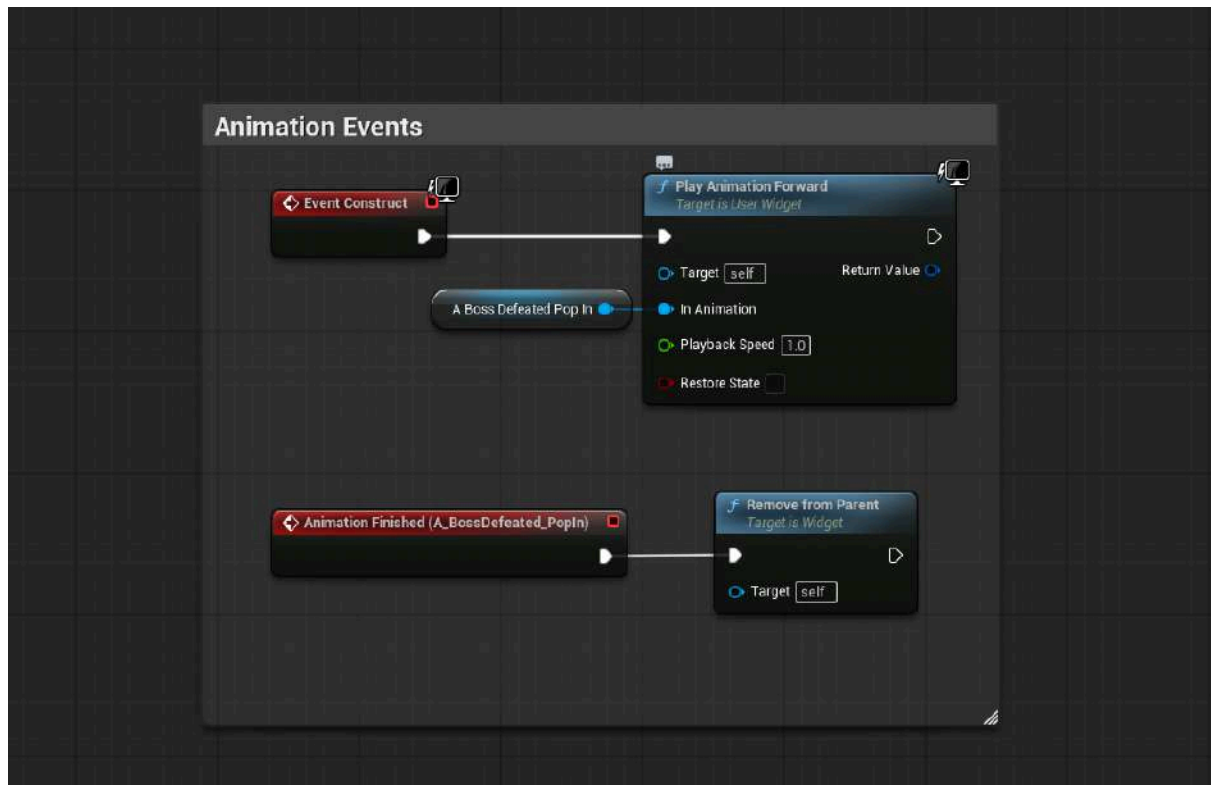


All the audio I have implemented in the game has been edited in Audacity. It allowed me to pick specific sounds I liked, isolated them, and made a loop out of it. I also equalized and compressed some of them, while others have been faded out or faded in.

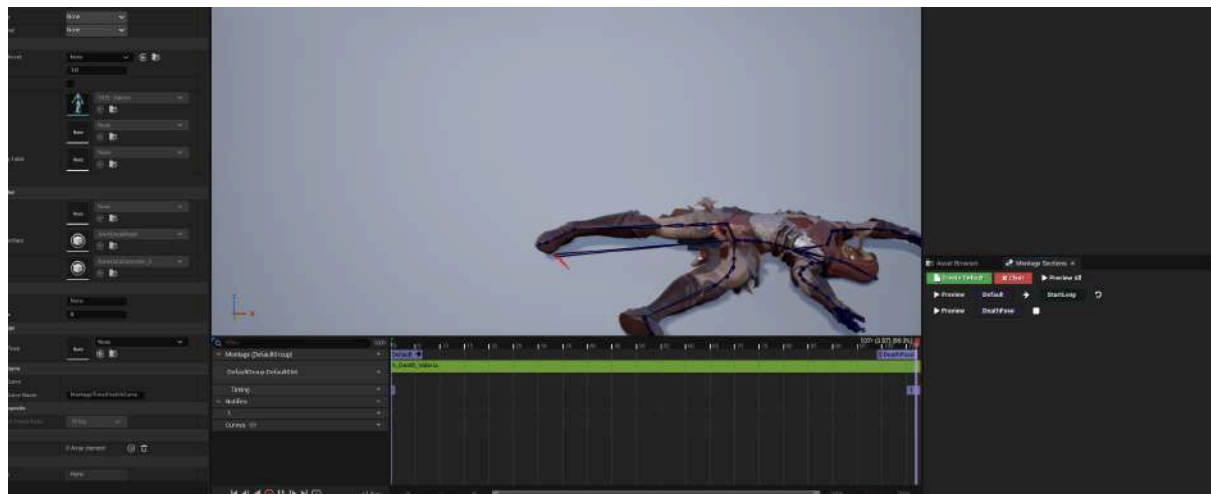


I have created a new victory and defeat screen with a short animation. After a short delay, the player is sent to the main menu.





For all characters in the game, I have created a death pose loop so they will stay in that position as long as the game is still running.



Weekly Reflection:

It has been an intense week. Almost all animations needed reimport and testing in game as animators were polishing their work, and we had an issue on Valeria's rig where the run, impact and death pose were not working anymore. I anticipated this issue and had a few Mixamo animations we could use, but it was very time consuming as we had to apply again and again VFX, camera movements, or tweak some camera angles in cinematics.

To be honest I am relieved to see the game in that state now as I was not very confident a few weeks ago. On the other hand I feel frustrated too because I can't help thinking how close we were to make a very good game with an extra couple of weeks, or if we did not waste a bit of time at the start of the production.