Crit 3 Reflection

# Goals for Crit 3

For crit 3, I wanted my dungeon populated by gameplay and simple rules. To this end, I needed the following,

* A Character Controller
* Enemies with AI behaviours
* Dungeon rules
* Player Character Skills

For the character controller, I wanted to try and combine the gameplay of Hades and the Arkham combat style, plus a few aspects that would suit the game's environment. This includes a stamina system that dictates whether a player to dodge or sprint, the ability to attack enemies using an attack combo, attack multiple spread-out enemies using a special attack (endearingly named Lightning Flash), and dodge in and out of an enemy’s space using a directional dash.

The Arkham combat style is a very sophisticated system that aids Batman’s prowess by having him dash all over the combat space to hit an enemy on the other side of them. In building my own system, I found that I have completely hamstrung this aspect of Arkham combat as I decreased the range of the player’s “step” between attacks. Gideon’s attacks are also far too slow to be a good fit for this highly aggressive combat style now, however, this can be fixed by finding/making more aggressive/suitable animations like a thrust would be a good fit for the initial distance coverage due to its minimalistic movement in comparison to a swing.

The other issue in the pursuit of Arkham combat is that since the distance is considered as well as the enemy's direction in relation to an input direction, the system will always favour an enemy that is very close by over the distant enemy in the correct direction. As a result, I will try to remove the check to find the closest enemy as well as the directional enemy when an input is given but the system will choose the closest enemy within a given range if there is no directional input given.

In some initial playtesting, I found that players (including myself) would get lost in the dungeon’s layout –so I made a quick hint system that points to the closest open and incomplete room. However, in some cases that hint isn’t very helpful since the direction to the closest room might be in a completely different direction than the corridor the player is or can currently go. To solve this now I may need to make a waypoint system that will help the player navigate the dungeon from junction to junction – this will be very time-consuming though since the idea I currently have is to build a system of nodes and then have A\* traverse them in the same way it did when the dungeon is initially generated, and thus this is not a priority.

Next, the enemies, for the moment I only have 3 basic enemy types – melee, ranged and brutes. Melee enemies are the generic run-up-and-hit enemy, ranged enemies hang back and attack the player from range, and brutes are much tankier enemies with over 3x as much health as a melee enemy and do about double the damage of ranged enemies.

I want each of these enemies to have unique aspects from one another besides their weapon of choice and archetype. I began experimenting with this with the brute enemy, they have multiple attacks and can stagger unlike the other two. When out of range of the player character, they will close the distance with a jump attack, but will give chase when the player is close enough. If the player bates this chase long enough the brute will “Stagger” leaving him completely vulnerable to attack. Currently, the brute is the only enemy with a unique mechanic, however should there be time later, I would like ranged enemies to dodge out of the player’s attack range when they specifically are targeted, and more ambitiously to give the melee enemies hive mind capabilities to be able to coordinate simultaneous attacks or to surround the player.

Both challenges may be much simpler than I initially thought but I think that these added features will make each enemy more formidable which is needed, given the lack of unique enemy types.

Another thing to do with the enemies is some player-danger communication. Ranged enemies are usually placed throughout a room to surround the player, and so most of them will be off-screen, to combat this I made a line render when an enemy takes aim at the player and it gives a great “Oh Shit” moment when the player enters a room with seven ranged enemies and they simultaneously take aim. Brute enemies are also an almost constant threat, and their jump attack needs to have some warning to tell the player what is coming. To solve it a large warning circle is placed at the player’s feet and marks the radius of the area of effect the attack covers. I intend to focus on my in-world communication like this in the future.

Finally, Dungeon Rules (yes it does). The dungeon currently has some very simple rules – clear every room by killing every enemy. This is far too simple, and I would like a more sophisticated set of rules in the dungeon space and an actual end to the dungeon aside from the killing of the last enemy. To this end I want to have killing every enemy as an optional objective – and clearing the boss room is a major priority.

Given the dungeon’s design, the boss room is accessible from the beginning of the run and thus a player will be able to challenge the boss early and without the rewards[[1]](#footnote-1) dropped from each of the room encounters, which would hamstring the player’s general ability to deal with the boss. Giving the player the choice to take on the boss immediately can be a redundant choice any player that enters the boss room will be locked into the encounter until one is defeated.

This choice is a two-sided argument, however, yes the dungeon is completed faster than if you went through every room, but the player will miss out on all the experience[[2]](#footnote-2) points within the dungeon that they would have got if they had eliminated every enemy. However, given the current state of the game, this choice depends entirely on how much time the player *wants* to spend in the dungeon, but without the incentive of another currency or reward, there is little incentive to stay.

Player skills are the player’s main stats – Health, Strength, Vitality, and Endurance. Each of these controls the player’s physical attributes and increases alongside the player’s progress regardless of whether they clear the dungeon or not. By attacking and dealing damage, strength increases, by taking damage, Vitality increases, using stamina increases Endurance, and but increasing all of these skills, Health increases.

This system mimics something like Skyrim or Valheim in their implementation, however in the Rogue-lite environment, this carryover leads to the player becoming extremely strong from run to run. With the existence of special attacks, I want to include a new skill in the form of Dexterity which will control how often and how powerful the player’s special attacks are and how often they can use them since making them cost Stamina would place too much pressure on the Stamina gauge rather than the combat.

# Challenges

## Dungeon Lighting and Décor

With the dungeon hallways being a large part of the game, it is a problem that they are so bland, especially if they just had direct overhead directional light, and so I made a system that overlays the dungeon layout with a series of lines and positions to mark light positions along dungeon walls. The positions are evenly spaced out based on how dense I wish to make the lights through a value that represents the number of lights per meter I wish to have in the dungeon[[3]](#footnote-3). This has helped an issue I had in a slightly earlier system where the lights scaled to the length of the wall and thus appeared to have very different intensities.

However, now there are hundreds of lights in the scene and Unity is not jazzed about it, I had to develop another system that relies on line-of-sight to the player and distance from the player to dynamically switch off a majority of the lights in the scene and stop the constant flickering. This has brought about an interesting idea to deepen the theme of the game however as the lights can now appear to hide something, however, the hallway emission seems to ruin most surprises.

Apart from the lighting, the halls are empty. This makes being in the hallways a bit boring and that can cause a few players to become disinterested. To solve this I wish to do something similar to the lighting and place objects throughout the hallways procedurally so that the dungeon does not seem as empty.

## UI

Upgrading the UI from the default textures and images is always stressful for me, but I knew I needed to remove the potential walls of text and solid colour. To start, I made a radar graph that shows the relative levels of the player's skills – as a skill levels up, its point moves outward. I thought this would be a great way of displaying a player’s playstyle, a highly aggressive player will have a massive spike in their strength level while a player that isn’t as good at avoiding damage will have a larger spike in defence.

After that I needed to replace the health and stamina bars to make a more stylish and less obtrusive health bar. Unfortunately, this was not as simple as repurposing the radar graph code to make a bar. For this I made a custom shader that takes in the Red and Green Channels of the UI image and turns that into separate greyscale maps. These maps are used to designate the space each bar inhabits, and thus the bars are multiplied by a step function on a gradient to make the bars go up and down as needed.

While I am glad I figured out how to make a shader like this, it felt like there are much better ways to make a shader like this. The result is not perfect however, I just need to touch up the red and green on the original image to be pure red and pure green.

## Animation

The brute enemy presented a unique challenge from the other enemies, it needed to launch *itself* at the player. Meaning I needed to dip a toe in some very simple procedural animation, to that end I began using the built-in state machine Unity uses to drive the animator.

This in tandem with a timer class made the process much easier and customisable. The result of this new challenge was an intimidating attack that made the brutes stand out as a new challenge for players to overcome. I am glad I got this working and learned of the existence of the state machine.

From that lesson I remade the ranged enemies' behaviour to control how long they aim at the player, on top of that, there is now a time before they shoot when they stop tracking the player’s position. This made those situations where there are multiple enemies aiming at the player easier to survive.

I also finally made the first special attack for the player – Lightning Flash, where the player attacks 3 enemies in close succession, however, this needs to be tweaked as the ability drops the player at the centre between all the enemies and is disruptive to the player flow. The animation used for this attack is also too slow for an attack with the word “lightning” in the name, I will try to increase the animation speed to resolve this else I will need other animations.

# Crit Reflection

I was very confident going into this crit with a game that can actually be played by the lecturers, on top of that I feel that now my way forward is clearer with polish, enrichment, and visuals being the primary focuses.

I want the game to be more playable than it already is, it exists now without any real incentive to play it beyond increasing Gideon’s stats. I want the game to transition from a dungeon crawler to more of a roguelike, and for that, I want to plan and use the ideas given by the lecturers from the crit that could aid that end. Firstly, by punishing or softening player death/failure while rewarding player success, a way I want to do this is with a new currency gained from defeating enemies in the dungeon, initially, if the player fails, they will lose all these coins, and keep all should they complete the dungeon.

I really like this idea, however, given the nature of currency, the player needs to be able to buy things with that currency. This adds a whole new layer to what needs to be achieved before the exhibition, an entire store needs to be built with things to buy and equip. I want to focus on this element since it will get the game closer to being a roguelike with the carryover of a currency. A few ideas I have for the store are attacks for the player to use as I wish to overhaul that system to use the Trigger-Button combo we see in most action games[[4]](#footnote-4).

While I hopefully crush that problem, I will try to improve on other aspects to better match the simulation-cyberpunk theme I have chosen for myself, starting with new rooms based on the original set of large, medium and small rooms but with more variance to remove that feeling of samey-ness while playing more to the strengths of the enemy types such as putting ranged enemies in difficult places to reach and making massive arenas filled with goons and obstacles. This will also give an opportunity to improve the room entrances since they can now follow a convention, making it less likely for players to “Enter the Void”[[5]](#footnote-5).

Finally, thanks to some very helpful advice I will try to work on the game’s communication to the player such as when they hit an enemy, when enemies hit them, when a room has been completed and what the controls are. This is more UI and VFX stuff so I will try to do it in between the bigger fish as they are the polish that makes games so much better.

1. Rooms drop a random reward of Health, Damage Boost, or Defense Boost [↑](#footnote-ref-1)
2. Player’s upgrade skills as they do actions associated with that skill/stat. [↑](#footnote-ref-2)
3. A value of 0.2 will place a light every 5 meters. [↑](#footnote-ref-3)
4. By holding the trigger, it switches the player into a special attack mode where the next button press (AXBY) will initiate a different attack. [↑](#footnote-ref-4)
5. Some entrances are wider than the dungeon hallways, and so the player can see into the skybox and leave the dungeon space. [↑](#footnote-ref-5)