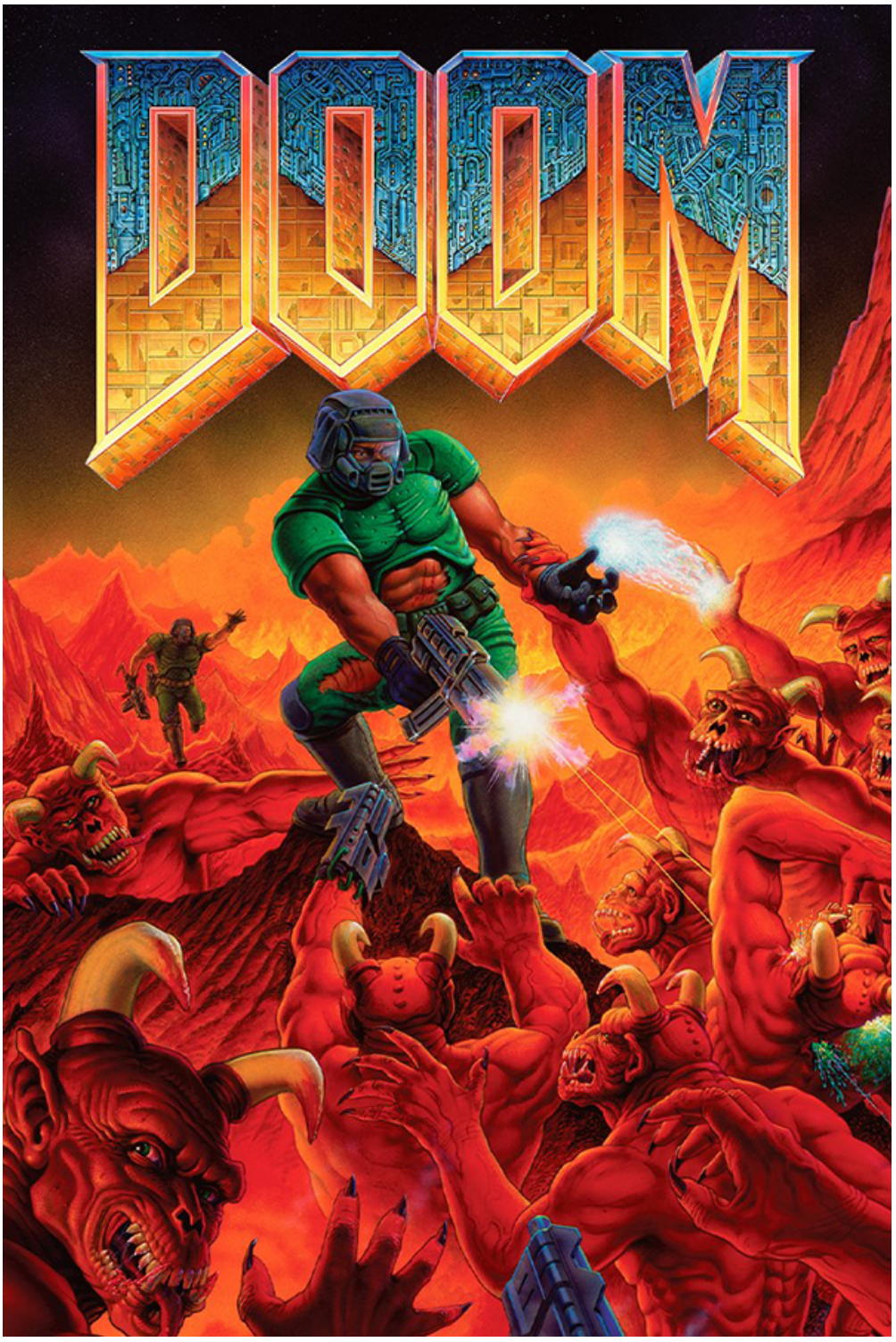
QUUK: A Doom inspired level



Index

Level Briefing

Title:	QUUK: A Doom inspired level
Authors:	Àlex Melenchón, Jose Luís Redondo, Oscar Pérez, Ferran-Roger Basart, Adrià Serrano
Overview:	This is supposed to be a level in *Doom*, therefore it's a level on a Action FPS thought to be found in the early stages of a story mode.
Key events:	Refer to Gameplay Summary for a detailed explanation
Target:	Doom nostalgics
Platform:	PC
Mechanics:	Move, Run, Shoot (3 differnt weapons), Health & Armor, Secret Doors, Keys & Locked doors, Loot & Powerups (Power Sphere).

Methodology

The pipeline for this project was **divided into 5 stages**. The intention behind this decision is so the team is aligned in the status of the project and where the focus should be put on. Each stage would take from a few days to a week (depending on how time consuming it was). Let's go trough them:

1) Team Alignment

The team was separated between **design** (Adri, Àlex, Ferran) **and programming** (Jose, Oscar) roles, to better suit the capabilities of each member. The plan was to have the **programming team iterate the project while setting the basis** in *Unreal Engine*. Concurrently, the **design team would**

do the research, providing information to the programming team to start implementing content, to later support them.

2) Design Phase

The design phase was progressed in **short cycles of feedback loops**, the three members of the team having **periodic meetings to ensure flexible iterations**, and that everyone was up to date.
The content to be discussed evolved with the project's state. We **began defining internal roles**: Adri would be responsible for level layout, Àlex would take care of enemy design, and Ferran would design player movement and interactions. With this distribution, and an initial approximation being inspired by the original *Doom*'s combat, the first proper part of the design phase would begin: research.
With enough information gathered, and through various iterations and roundtables, the team decided on the aspects that would better fit our level idea. Such **first draft was discussed and shared with the programming part of the team, adapted to the limitations we faced with a new engine**, and pushed forward towards implementation.

3) Execution Phase

For the execution phase, the **programmer team's prior research would result invaluable**, as they found a convenient way to have **multi-user editing in *Unreal Engine***. Without any restriction of users to co-develop, each member could take turns with the list of assigned prioritised tasks. Because of that, execution phase was quicker than expected and a **first iteration of the level was completed in a few days**,

4) Iteration

Having implemented the first approximation of the level, it was time to **play through it, detect where the weaker points were (or improvable points) and iterate** over them. This step took almost all time left (about a week and a half) in which all the members of the team would suggest changes, try them out and implement them permanently if found to be an improvement.

5) Post Mortem

Final phase: the level is done and all needed mechanics are implemented. Here, we **look back at the whole process and discuss what did we learnt**, what could've been done differently and try to identify the greatest pain points in the pipeline.

Literated Summary

The level is structured as a **classic *Doom* level**, where the player is presented with a **layout they have to scan, recognize, and learn to traverse**, in order to unlock all the **shut doors** that will eventually lead them to the **climatic challenge**. To remain engaging, the level makes use of **verticality and framing windows**, in hopes of **raising questions that incites exploration** and clear goals for informed decisions.

Gameplay

The player spawns at a predetermined point in the map, and has to traverse it, **gathering all the necessary keys** to reach the exit while taking down any enemy that stands in their path. To do so, they will be able to **move around** (walking or running), turning their camera (horizontally only), **collecting objects** and **shooting** their weapons.

The **player starts only with a basic pistol** with infinite ammunition. As they advance, they might **collect the Shotgun, or the Rocket Launcher**, which have limited ammunition. The Shotgun has a wide, low damage spread, while the Rocket Launcher has a big AoE damage that affects the player too. **Managing their ammunition and the contextual fights is where the level's elements of triangularity are found**.

Every time the **player kills an enemy, there is a chance of ammunition drop**. Throughout the level there are also **objects that restore health and armor** (a defensive value), which are useful to guide the player and provide tension relief at the resting spots between challenges. The **player**

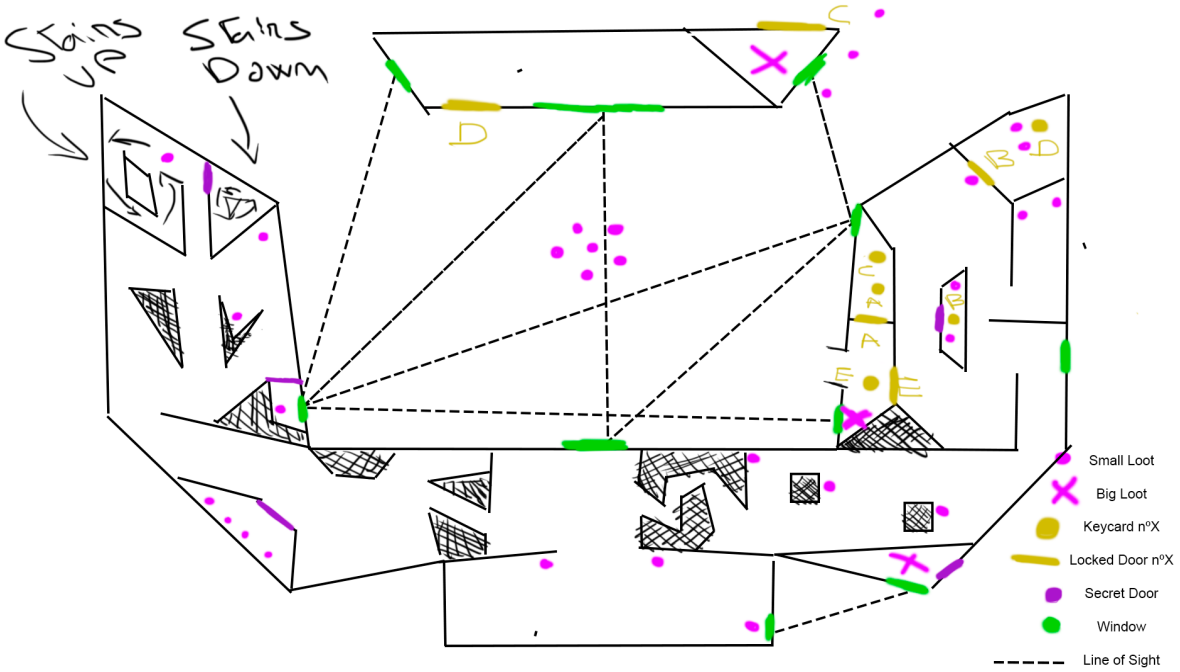
might also find a better armor, or a power-up that increases health and armor to the maximum value.

Hotkeys:

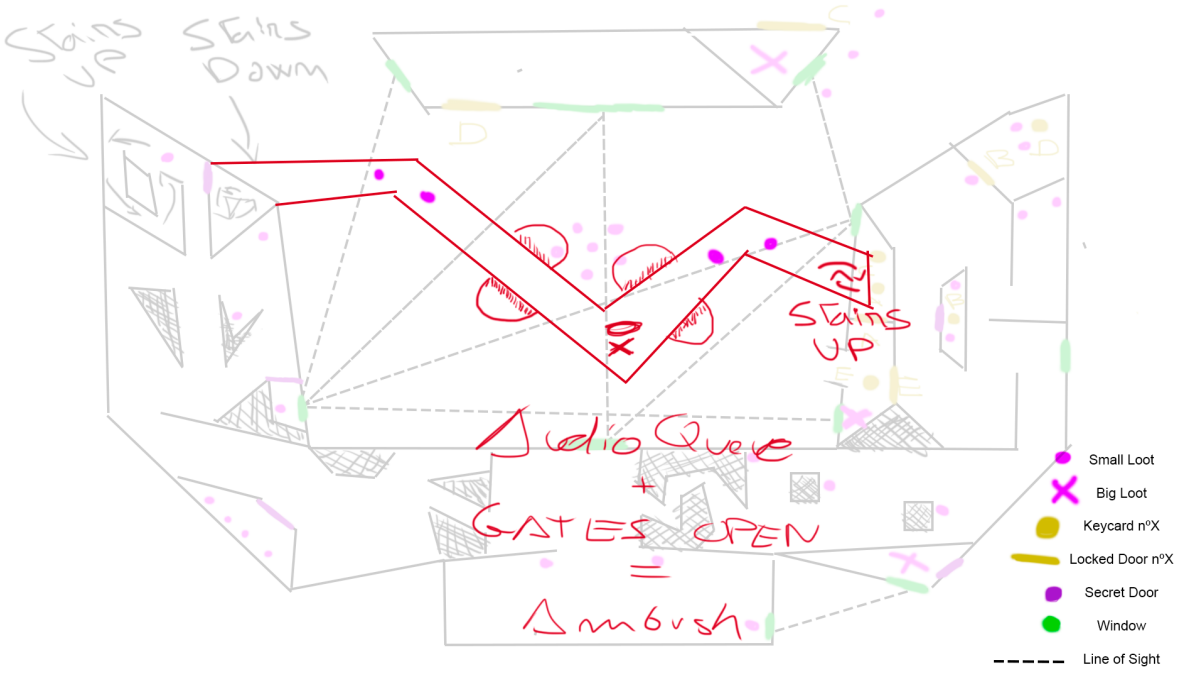
- WASD: movement
- Shift: run
- QE / Mouse horizontal movement: player turn (can be combined for greater speed)
- Left click: shoot (if it is kept pressed, player keeps shooting)
- Right click: pick-up keys / open doors

Drawing of the Map

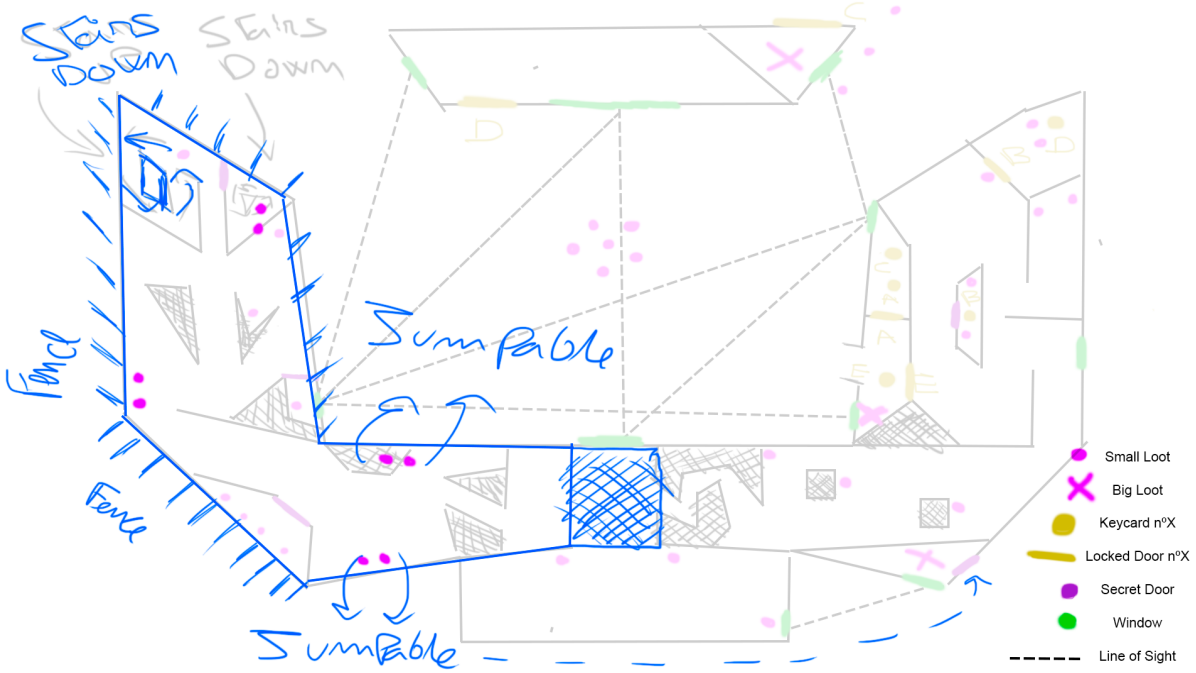
Floor



Underground



Ceiling



Level Summary

The level is a classic **"horseshoe level"** (a concept that comes up in John's Romero discussion about his approach to designing Doom's levels). A horseshoe level (called thus because of the form it takes when looked upon in a top-down perspective) revolves around one idea: **there should be multiple lines of sight from inside of the horseshoe to the outside**. Those lines of sight should **inform the players of different points of interest and/or danger**. Additionally, those lines should connect points of the horseshoe itself (traversing empty space in the middle). Finally, it is a **promise to the player: it should be possible to go outside** the horseshoe at some point, and re-enter it from a different point.

In conclusion, a horseshoe structure is a classic "level design pattern" of Doom levels in which then **fun comes from the promises of the lines of sight and the resolution and rewards of those promises** (apart from the funt hat comes from the combat). Thus, we can say that the **intrinsic fun comes from a desire of exploration** (and narrative implication, should this level be a point of progress in the overarching narrative). This exploration is manifested twofold: **discovering all the different secret rooms and doors (a Doom staple) and discovering all the different weapons in the level (upgrades)**. About player choice in Doom levels (Critical vs Golden path) we find two different approaches: small nooks and crannies surrounding the path (secret doors) or a different path hidden mid level that rewards the player with more loot and a harder challenge. **We choose to implement both.**

Regarding the critical vs golden path decision, we decided to **use vertical level design** to solve the problem. The inherent promis of a horseshoe level is that the player eventually will get outside. In combining both statements, we ended up with the following proposal:

Critical Path

Player takes a left after exiting the playground area. We used the level layout in the right to convey more danger (the structure at 5 creates no line of sight and therefore is perceived as more

dangerous), so it's clear that the low-effort path (critical) is on the left. After player makes this decision, it clears 2 combat encounters (13 and 17) and gets to the bifurcation. Ignoring the secret door at 19, player climbs the ladder at 18, getting to the rooftop.

After clearing the encounter in the rooftop, player can explore the ceiling, which will make it obvious one can just jump off to the inner area to 20. From there, and after exploring for a bit, player gets into 29, which opens the door back to 8 and the first weapon. After clearing that encounter and getting the key in 9, player can open the door at 10, getting access to the key to 22 which is the final boss. Since player has unlocked the door at 8, player can just go outside, go to 22, open the door and kill the boss.

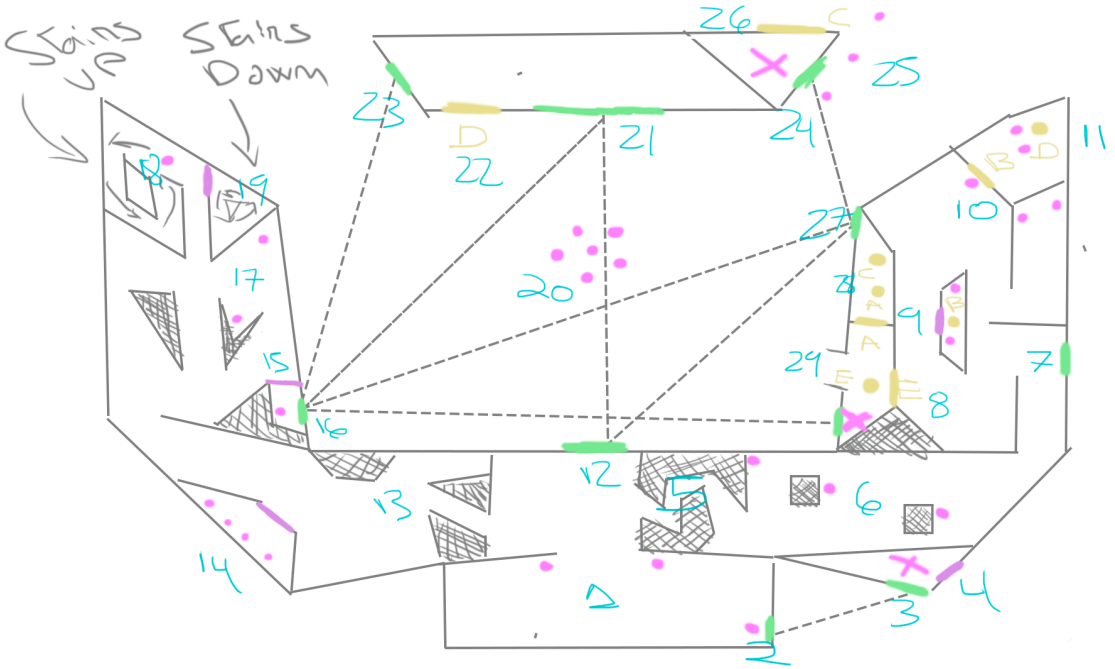
Golden Path

We will consider the **golden path as the path that gets to collect all 3 big loot points.**

Player starts by traversing the critical path until getting to the bifurcation at 19. There, player brakes the secret door and drops down to the underground (we consider that in vertical layouts, going up is always considered safer and more direct that going down, just by psychological perception; so it's clear which way is the critical and which one is the golden). In the underground, player clears out the guantelet at 34, ending up at 28 which will reward the Key that opens 26. From there, player can access directly the room next to that, which gets it in part with the critical path and therefore the first weapon. From there, player repeats the beats on the critical path until getting out for a second time. At that point, the player can go all around the complex until getting at 4 (which was the first carrot we planted on their mind at the window on 2). Here, player gets the second weapon. After going to 22 to kill the boss, player can take the key that they got front he golden path and take it to 26, which will reward them with a good powerup: the MegaSphere. After that, armed with a powerup and 2 premier weapons, the golden path player can finally face the boss at 22.

Map Drawing and beat by beat Analysis

Let's take a look at the map and see what we did:



1. Player starts here. Intended "empty" open space to serve as a playground of movement / camera / controls, etc.

2. Carrot the player with some small goodie and lure it to the window. From there, player can see into n3 window, which makes player spot the rocket launcher. First instance of line of sight usage to create premises.

3. From this window (which connects LoS from 1) Player can see Big Loot. This is not seeable from ground level, only from Window 2 (therefore, both points have a little altitude).

4. Player will get here once they get outside, since the room is not accessible from the inside. Therefore, players which remember the existence of this room once they get outside could surround the complex and try to look at the level from behind, which would make them spot the hidden door that rewards them with a new weapon.

5. Corridor that separates the two zones (ample - narrow - ample, a technique described by Doom's level creator as one of the go-to patterns to create tension). Here, player can face a combat encounter.

6. Combat opportunity. Pillars + corridor offer some tactical depth since they can be used as cover + LoS blocker for melee enemies.

7. This window is here so player knows it is possible to get outside, to the “outter” ring of the building. This hints that trough that part of the map, it is possible to get to n3 or n4.

8. Combat zone + visible door on E, locked though (opens from the other side). This makes players look at the zone trying to find the key, which leads us to...

9. Semi-obvious hidden door, with a switch or breakable wall. Small loot inside + a switch that opens B door when clicked. Note that the intention is that they find this while looking for a key to 8, which is not found here. This creates another promise (what's behind that door and how do I open it?).

10. Some small loot to lure player to check the door in case player hasn't noticed 9.

11. Small loot + key that opens D. Players can now access the final boss once they get outside.

12. Observation spot. From here, player can check 20,27,21,22 and 25. It lets players plan out the first steps of the run, since this is the first thing they find when stepping outside the playground area.

13. Combat zone, with enemies coming from the 2 sides from behind + possible from the front.

14. Some goodies (health, armor, ammo) behind hidden door. Probably the first instance of a hidden door the player will find, it should be presented clearly as such so player gets introduced to the concept.

15. Hidden door that leads into 16. Important Window since it gives premier LOS into 27 and 21 (from here, you can see the boss! Which was not seeable from 12).

16. Great observation spot as a reward to players. From here,player can see “hidden” info such as 23, 29 and 27

17. Sequence to lead players coming from 13 to the following: check the upper loot, rotate camera to get back on track, which lets them see the lower loot on 17 and maybe notice something weird on 15’s secret door (player guidance).

18. Critical path with small loot lure. Staircase goes up.

19. Golden path behind secret door. Staircase goes down.

20. Cluster of goodies in middle of map, spottable from first observation spot at 12. Clearly indicates players that at some point, they are getting outside.

21. Window which only shows partial information (upper half). Carrot seeable from 12, while threat could only be seeable from 23.

22. How to get inside the complex. Inside, climactic battle + end level.

23. Observation spot, which is in the player's mind if spotted trough 16. From here, the threat inside 22 can be seen and a plan can be made.

24. Window which let's players see the Big Loot inside, spottable from outside or directly from 27: a shotgun!

25. Line of candies to guide players to 26. We want this to be obvious, since we want them to find 26 (which its key is the reward of the golden path).

26. Door which can be unlocked if player checked the golden path, inside there's the Big Loot.

27. Window to indicate players than it is possible to access 28, even if no apparent entrances exist. Intentionally placed there so no information about 28 can be seen from 16 or 12.

28. From here, player collects the keycard to 26 + can use a switch that opens the adjacent door to 29.

29. Switch which opens path back to 8 + leads players outside. This way, player connects inside to outside regardless if the player took the critical path or the golden path

30. Staircase goes up to the ceiling, where player can walk and later fall off to the outside (vertical level design).

31. Combat in open space, opportunity to have something different (less obstruction from walls and things, open space combat)

32. Opportunity to jump to open space, leading players to 29 and all other outside points of interest

33. Player guidance should indicate players that they can jump tot he back directly, signalling the presence of something good there (in this case, 3 & 4)

34. Hard combat for players on the golden path. Upon pressing a pressure plate on 34, a audio queue lets them know that an alarm went up (opr something akin to that) and all the adjacent walls open up, revealing the monster closets.

Constraints

Theme

From the beginning, we decided to limit ourselves theme-wise by **adopting a very *Doom*-like style.** We considered this a positive constraint, because it would give us a **focused objective and frame** from which to begin working, since the main point of the project was to develop notions on level design.

Time

Time, as it is the case in many projects, **was a problem for all members**, since all of them either had other assignments to fulfil, or were occupied with their internships. Nonetheless, with proper planning and scope-limiting, we managed to distribute work as to see the end of the level's iteration with satisfactory results.

Team

We consider the team that worked on this project to be a very capable one, **relatively experienced in iterative projects and with a fine awareness of each other's capabilities** and work dynamics. Additionally, the size of the team was adequate for the scope of this project, and the differences between each member helped complement the different tasks that had to be performed. It is, in fact, this range of expertise what managed to take the project towards success despite the constraints found in the rest of areas.

Tools

In terms of development tools, we faced our **greatest limitation by having to work with a completely new engine** (which is not as simple to learn as *Unity*). This resulted problematic not only because of the inherent time loss of learning something new, but due to the usage of visual programming, and the mixed quality of free teaching content. All of those added to the time

constraint, complicating the development especially for our programmers, who had to endure most of the hassle.

The lack of assets, most of which were taken from the engine's default first-person shooter pack, also **added a feeling of unpolished results**, especially due to the lack of audiovisual variety, but we considered this secondary hurdle a lesser one, since the **purpose of the level prototype is to provide a solid basis for future iteration**.

Because of that, **we decided that our level should "end" in a whiteboxing stage.** Just the minimum so **level intentionality is clear and playable.** Focusing on that instead of battling the engine to have complete art & light effects we hope will enable us to create a better design.

Level Goals

Since the level was designed as an exercise to develop design skills, it lacks a game-level context that could dictate what role it should take, or what kind of mechanics it should introduce / develop. Consequently, when defining the extrinsic goals of the level we settled with **being a fun combat map, being engaging and interesting to traverse**. Should this level be implemented into *Doom*, the motivations are those described into Level Summary, which boil down to a hard focus on exploration and combat thrill.

From the designer's perspective, since not all weapons in *Doom* are found in every level, we thought it could serve as a level in which either the Shotgun or the Rocket Launcher are introduced. As such, the **level design reflects this decision with distinct combat scenarios that makes the player take decisions** on what kind of approach is better suited for the situation.

Enemies

Enemy Patterns (AI)

Justification

We decided to use the **original DOOM simplified AI as a basis**, as it has been proven that it works and will aid us with our limited experience with Unreal Engine. We will build upon it with more modern techniques, like for example, Unreal's pathfinding system instead of more raw movement options of the original DOOM.

In addition to the technological advances, we will also adapt the AI to the needs of the project, adding or subtracting AI behaviours as we see **fit with the design of the level.**

Common Pattern

Enemies share a **common AI pattern** between each other, this simplifies things, as their differentiation is on stats, timings and types of attacks.

Enemies are idle and do **not move until they detect the player,** this can happen in multiple ways:

- Enemies have a vision frustum; if the player is inside it they will start chasing him.
- If the player is not inside the frustum but it is at 64u or less to the enemy.
- If the player damages the enemy.

Once they are triggered, the enemies will start to chase the player down. Chasing the player works as follows:

- (ON ENTER ONLY) When the enemy starts chasing the player down, it will set up a reactionTime counter. This will be a timer that will vary between enemies and, only once it reaches 0, the enemy will be allowed to attack.
- The enemy will move towards the player. It will always rotate itself to look at the player.

- Every time it moves, it will check if the player is at a range for an attack. If the player is, the enemy will attack; melee range (if exists) takes priority, then the range check is done.
    - In order to determine if a player is in range for an attack, the enemy must check:
        - If its reaction time is 0
        - Line of sight w/ the player (raycast)
        - The range of the attack, as we said, with melee is just 64u. For ranged attacks it gets a little more complicated:
            - We will calculate the distance from the Player to the Enemy; then we will subtract 64u from that number (This has a cap where it cannot go lower than 200u). Then we generate a random number between 0 & 256; if the calculated number is lower than the distance, do not shoot; if it is not shoot.
            - If the enemy is flagged with the 'Attack Back' flag, ignore all checks and fire a projectile, then remove the flag.
- Once an enemy attacks, it will be flagged as 'just attacked'; this will be a timer that will not allow the enemy to shoot again until it reaches 0, forcing it to move.

An enemy will repeat this behaviour until they die, stop chasing the player or get hit.

Stop Chasing The Player Pattern

- An enemy will stop chasing the player and return to idle if:
    - The player is dead
    - Is above a certain distance threshold and they are not able to see him.

Getting Hit

- Once an enemy is hit, we subtract the damage dealt from its health and if it is lower or equal to 0, it dies. If it is not we do the following:
    - Generate a random number between 0 & 100. If that number is lower than the enemy's pain chance, the enemy exits its regular pattern and enters an 'on hit' state that lasts as long as its pain time and then goes back to its regular pattern.

- ➢ Once the state is over, the enemy will be flagged with a 'Attack Back' flag, which will guarantee it to attack with a range attack (if they have it)
  - ■ This state cannot be stacked; meaning that an enemy cannot reset its 'get hit' state if it gets hit during it.
- ➢ This sets the reactionTime of the enemy to 0 (if it was not already)

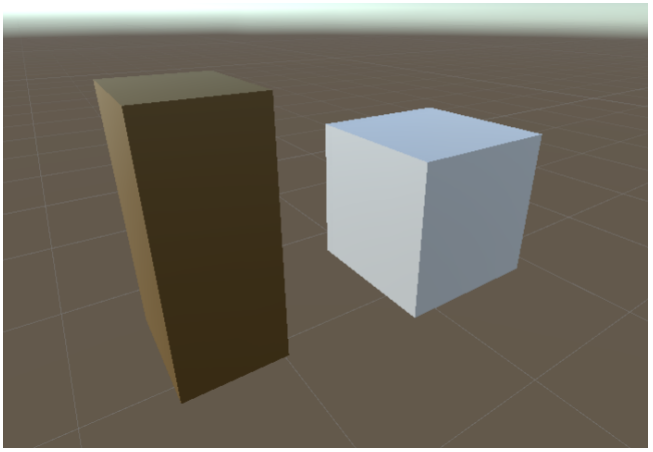
Additional Rules

- ❖ Enemies will always have the player as a target and will not be able to hit each other.
- ❖ Enemies' projectiles are not able to hurt any other enemy but will disappear on impact.
- ❖ Enemies projectiles can still make explosive barrels explode.

Enemy 1: **Imp**

Sprite in Doom

Suggested MVP Look (compared to a '1x1' cube)



Why?

The 'base' enemy of our level and the most basic one: mainly teaches you to dodge projectiles. As an initial enemy, it might seem confusing that it has both a ranged and melee attack but it will mainly use the ranged attack and the melee just when the player tries to get too close. Plus, taking into account the way we created the AI for enemies should not be a problem, since it will be using its ranged attack most of the time, being its melee a resource when the player tries to get very close to an Imp, a thing that is not encouraged by the rest of the design.

Description

Base enemy, relatively **low health and a projectile range attack** that is easy to dodge / cover from. Its speed is average and has a melee attack mostly used when the player gets too close, as a sign to keep them afar.

Attack Patterns

- Melee Attack
- Ranged Attack (Projectile)

Stats

Thread Level	1
HP	50
Speed	83 u/s
Width (hitbox)	40u
Height (hitbox)	80u
Mass (knockdown resistance)	100
Pain Chance	70%
Pain Time	133 ms
Melee Attack Dmg	3-10
Range Attack Dmg	5-15
Reaction Time	264 ms
Time Between Attacks	1125 ms

Projectile Stats

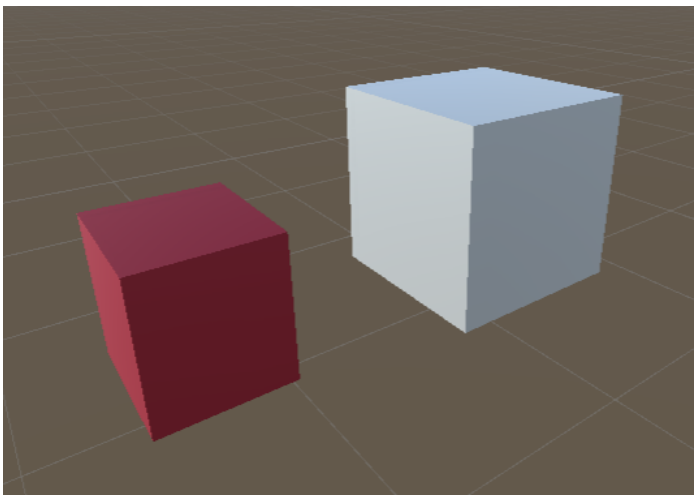
Speed	350 u/s
Width (hitbox)	10u

Height (hitbox)	10u

Enemigo 2: **Demon (Pinky)**

Sprite in Doom

Suggested MVP Look (compared to a '1x1' cube)



Why?

A clear opposite of the Imp: it is fast and charges at the player; forcing the player to move in a more aggressive way though the scenario.

Description

Act as a **fast melee sweeper** that forces the player to move through the scenario in a different way than dodging projectiles. It is faster and has more HP than the IMP.

Attack Patterns

- Melee Attack

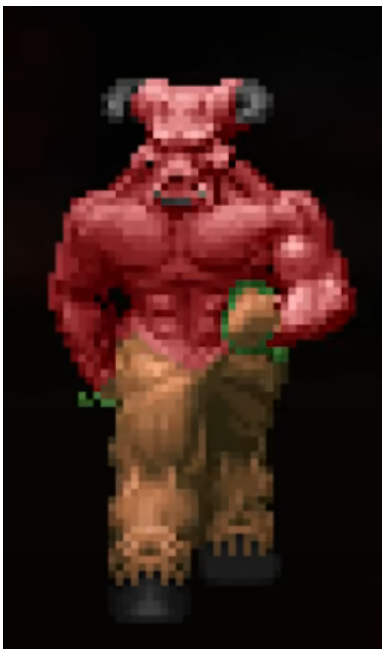
Stats

Thread Level	**2**
HP	120

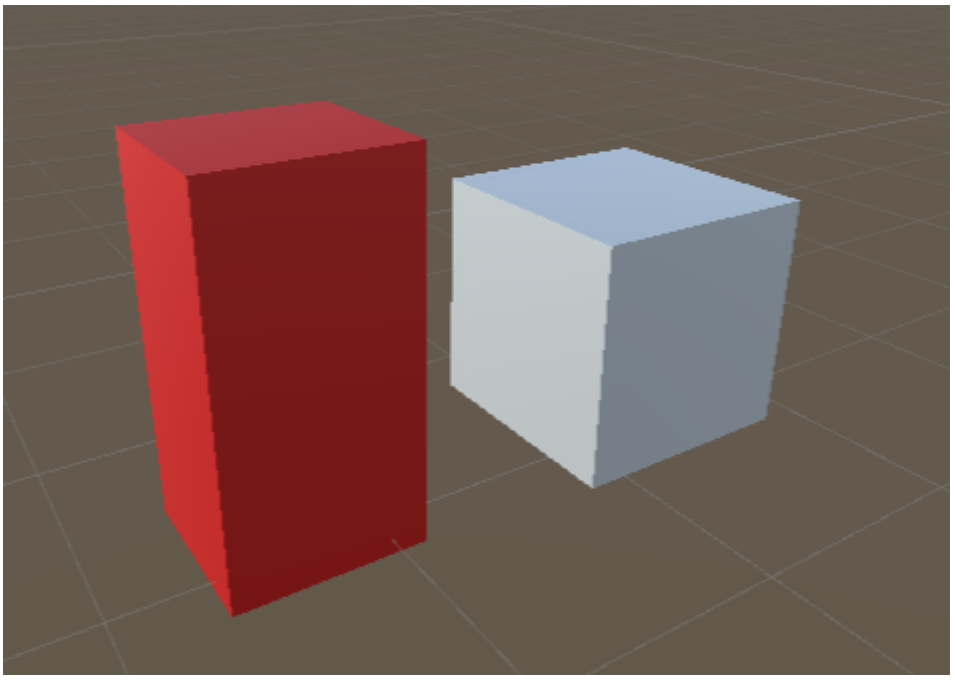
Speed	175 u/s
Width (hitbox)	60u
Height (hitbox)	60u
Mass (knockdown resistance)	400
Pain Chance	55%
Pain Time	100 ms
Melee Attack Dmg	8-20
Range Attack Dmg	N/A
Reaction Time	264 ms
Time Between Attacks	1125 ms

Boss; **Baron of Hell**

Sprite in Doom



Suggested MVP Look (compared to a ‘1x1’ cube)



Why?

Our **boss enemy**: it is a combination of the past two enemies, only more bulkier and dangerous.

Description

It can launch projectiles and perform melee attacks and it's faster than the Imp. Its attacks do significantly **more damage and its HP is way higher too.** It tests the player skills and its exploration of the level, since with the additional weapons it becomes way easier.

Patterns

- Melee Attack
- Ranged Attack (Projectile)

Stats

Thread Level	**7**
HP	350
Speed	120 u/s
Width (hitbox)	50u
Height (hitbox)	70u
Mass (knockdown resistance)	1000
Pain Chance	15%
Pain Time	116 ms
Melee Attack Dmg	22-40
Range Attack Dmg	12-31
Reaction Time	464 ms

Time Between Attacks	825 ms

Projectile Stats

Soeed	550
Width (hitbox)	15u
Height (hitbox)	15u

Environmental Prop; Explosive Barrel

Why?

Explosive barrel that acts as an environmental tool and danger: it can be used to quickly kill enemies but the player has to be careful not to get caught in the blast. Enemies can trigger its explosion too.

Stats

HP	20
Width (hitbox)	20u
Height (hitbox)	70u
Explosion Damage	1-128
Explosion Range	127u

How does the explosion work?

The closer you are to the barrel, the more damage you take. If the receiving entity is 0 to 1 units close to the barrel it will take full damage; if it is far from it, we will subtract the damage for every unit the positions are separated, being +128u units taking 0 damage.

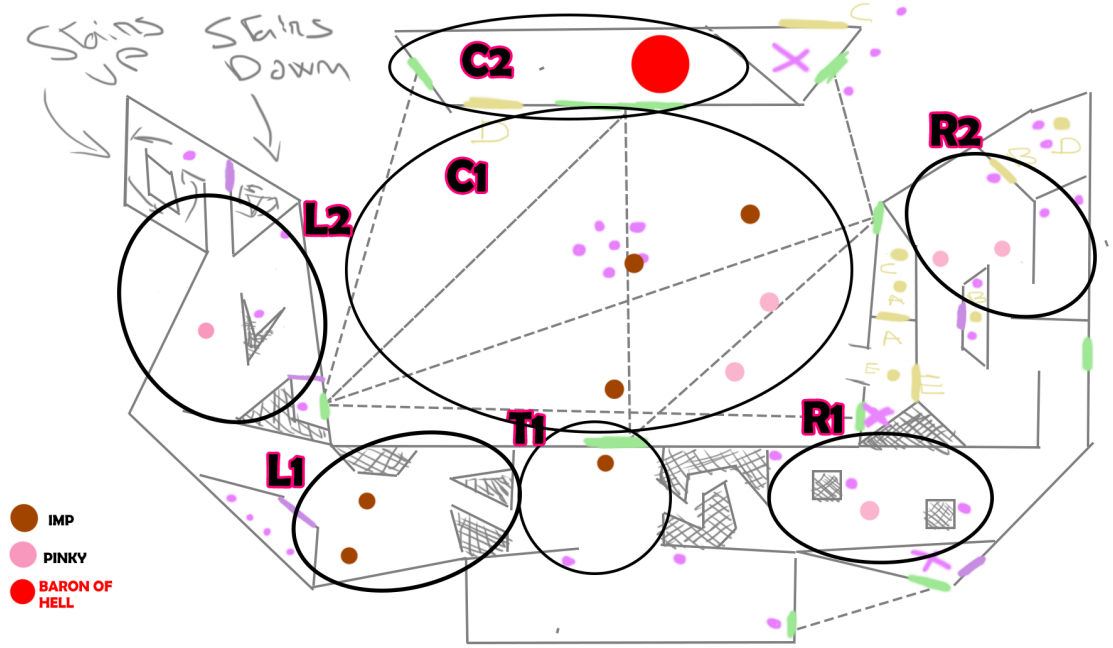
Encounters

For encounters, we will divide the map in different sections and place encounters taking into account the level layout, enemies and leaving room to breathe between them.

We will assign to each encounter a Threat Level, this will be composed of the enemies thread level as well as some extras for the dynamics between enemies, environment or the player's situation when reaching it.

Now, we are going to explain the encounters designed for our Combat Level, as well as their intentionality and place in the overall difficulty curve of the level.

Encounters at the Floor Level



T1

	Description	Threat Level
Enemies	X1 IMP	1
Environment	Open, plenty of room to navigate	0

	Total Threat Level	1

As a tutorial, the player will see a single IMP, which is the weakest enemy overall, facing the ground, so he/she can freely start shooting and have an advantage in the battle. The tutorial IMP will still have time to turn and shoot back but the challenge will be considerably reduced, making it a perfect way to introduce an enemy.

The intention behind this is for it to be a tutorial for the player showing him that he/she can shoot enemies and that they can shoot back.

L1

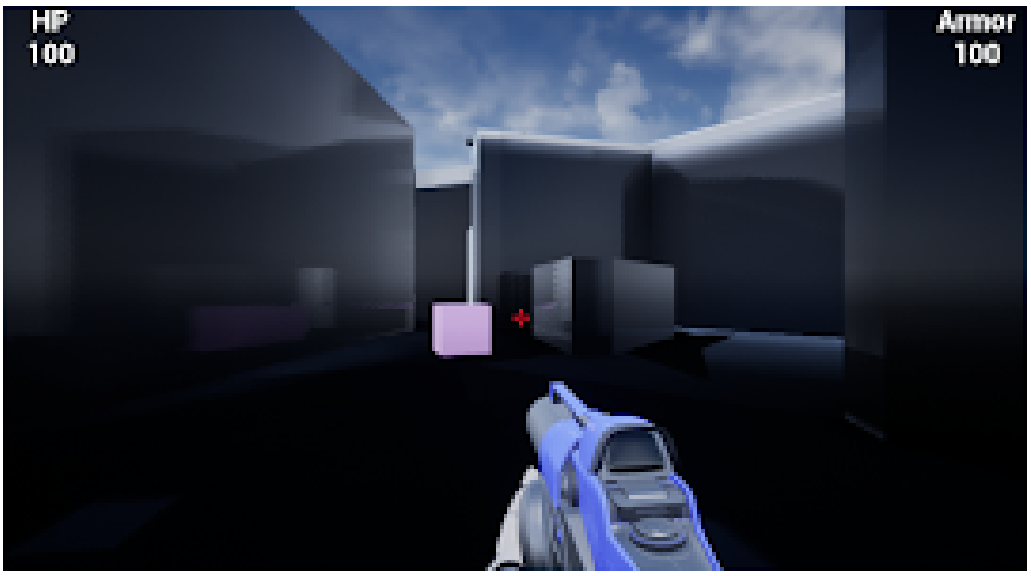
	Description	Threat Level
Enemies	X2 IMP	2
Environment	Open, but restraining enough for a 2 enemy fight	0.5
	Total Threat Level	2.5

The first fight we encounter after the tutorial if we decide to go to the left, is an expansion of the tutorial as we make the player fight against two of the same enemy as before in a reduced space.

L2

	Description	Threat Level
Enemies	X1 Pinky	2
Environment	Space divided in the middle (which is good, since you can manoeuvre the Pinky around)	0
	Total Threat Level	2

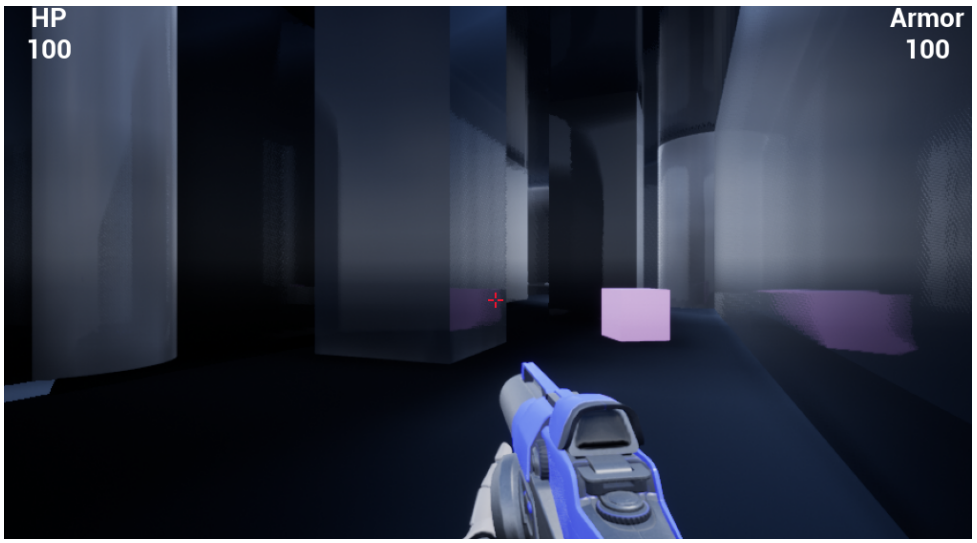
Here we reduce the difficulty a little bit, since we are introducing the player to a different enemy: the Pinky. This enemy is way different than the previous two encounters so we want the player to experience it in a controlled environment.



R1

	Description	Threat Level
Enemies	X1 Pinky	2
Environment	Space divided by columns. Teaches the player to move the Pinky around	0.5
	Total Threat Level	2.5

A challenge relatively similar to L2 and for a good reason: it is, on one hand, it acts as the same as the L2 if the player decides to go to the right instead to the left and acts as a different challenge. On the other hand, if the player goes through here (following the Golden Path) will find a challenge that is familiar and, ence, its perceived difficulty will be lower and will work as a 'resting zone' to reduce the difficulty curve. Regardless of that, the challenge will be different, as the environment where you face the enemy is completely different and more challenging than in L2.

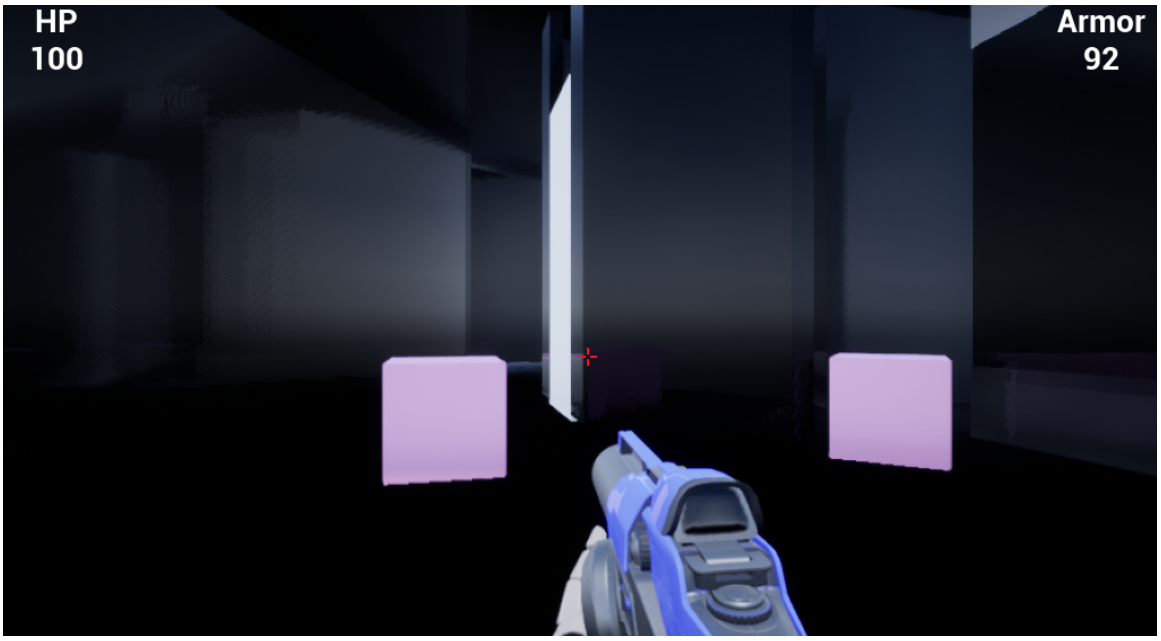


R2

	Description	Threat Level
Enemies	X2 Pinky	4
Environment	Space divided by columns. Teaches the player to move the Pinky around	0.25
	Total Threat Level	4.25

The difficulty peak of the right wing. We offer a challenge where the space and environment takes centre stage. Most of the difficulty comes from playing around the enemies using the environment, either for your benefit or not.

Here the enemy placement is quite important, as we want the player to face the challenge whether he/she is coming from the right wing start or from the outside patio. For that reason, we have introduced 2 pinkies, both facing each door separated by a column. This way, the player will always enter facing an enemy and, when moving around the space to kill it, will find him/herself with the other one, adding into the challenge.



C1

	Description	Threat Level
Enemies	X2 Pinky, X3 IMP	5
Environment	Open space with plenty of room to roam around	0
	Total Threat Level	5

The combat before the boss is the most difficult encounter the player will face before the boss fight in terms of the number of enemies but not the environment. As the player has plenty of room to roam around.

We want to make sure that the combat feels 'epic' in scale and check that the player has all the learnings done before going into the actual boss fight but not make it 'too excessive' in difficulty, as it would water down the actual boss fight. Along these lines, it might seem counterintuitive to place a

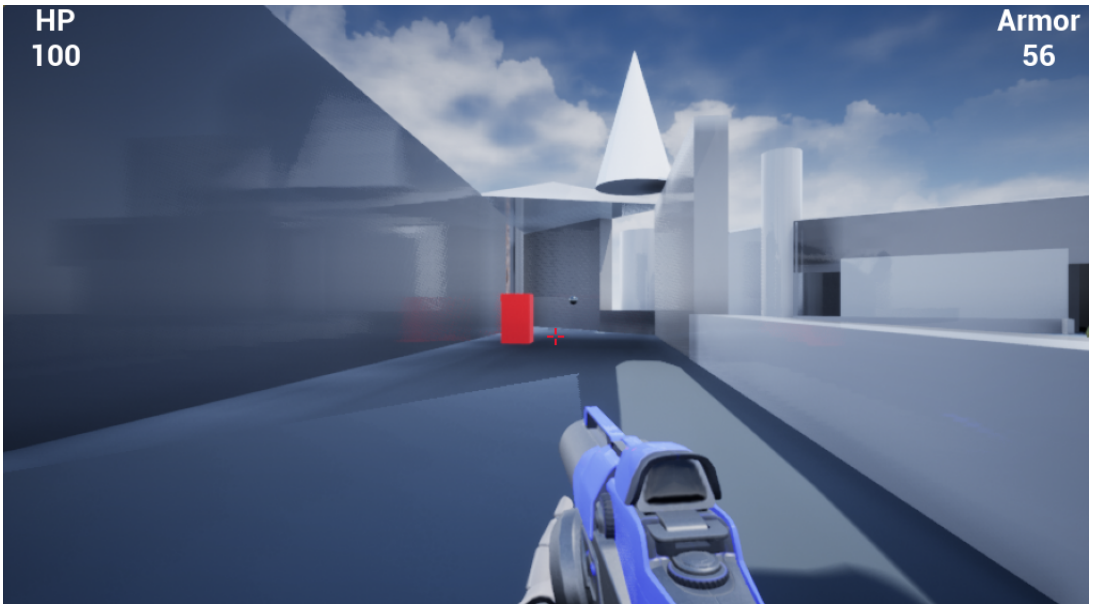
fairly difficult encounter before the boss but it works in this case as the player has plenty of room (and is incentivised to) roam around finding loot, working as a rest area.

C2

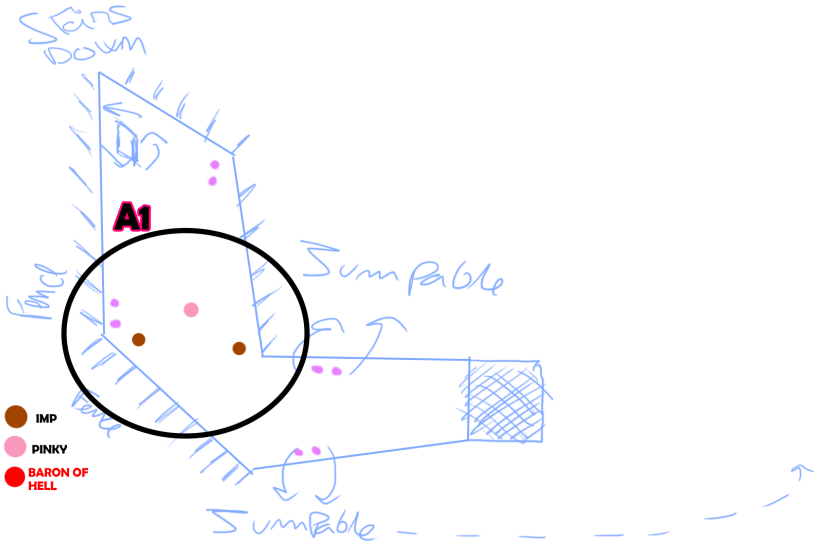
	Description	Threat Level
Enemies	X2 Baron of Hell	7
Environment	Very small claustrophobic space	2
	Total Threat Level	9

The boss fight, here we face the player with its most difficult challenge yet: a very difficult enemy with projectile and melee patterns, facing it in a very narrow space.

Here, the player will have a tough time but we will reward the exploration as having the different weapons scattered across the map will enormously facilitate the boss fight.



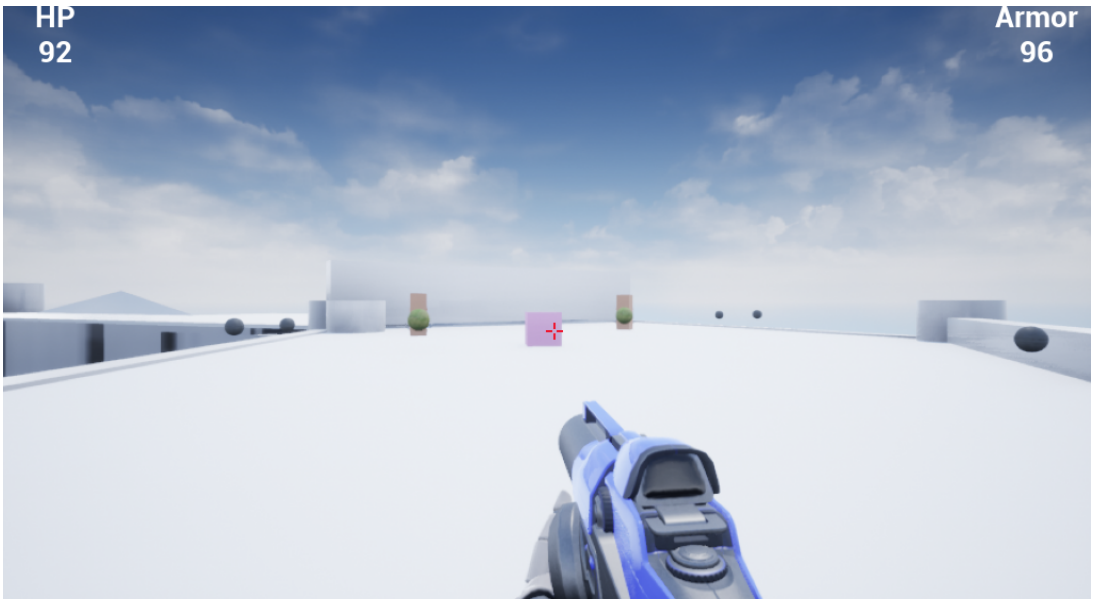
Encounters at the Ceiling



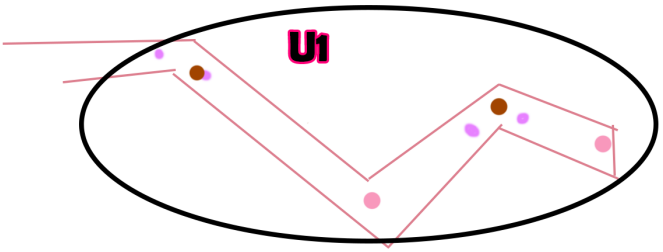
A1

	Description	Threat Level
Enemies	X2 IMP, X1 Pinky	4
Environment	Open, plenty of room to navigate	0
	Total Threat Level	4

On the top floor, we find a single encounter. Used as a mid-peak difficulty spike. It offers a combination of previous challenges combined: 2 IMPs and 1 Pinky. Adding a new dimension to moving through the level avoiding attacks. Since this is a difficult and challenging new encounter, the arena is quite open so the player can dodge attacks without worry.



Encounters at the Underground

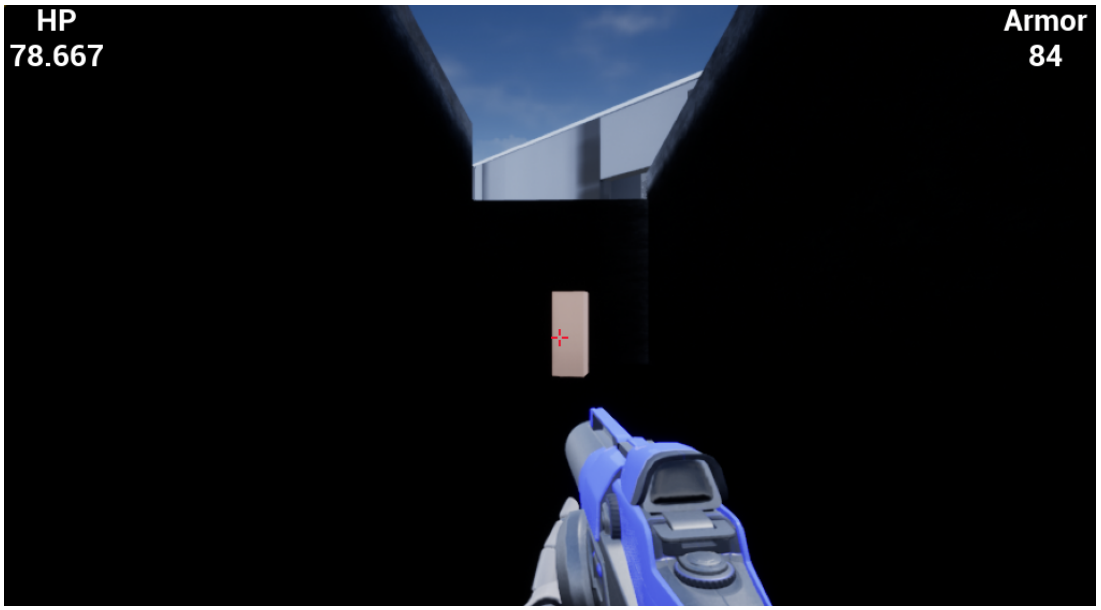


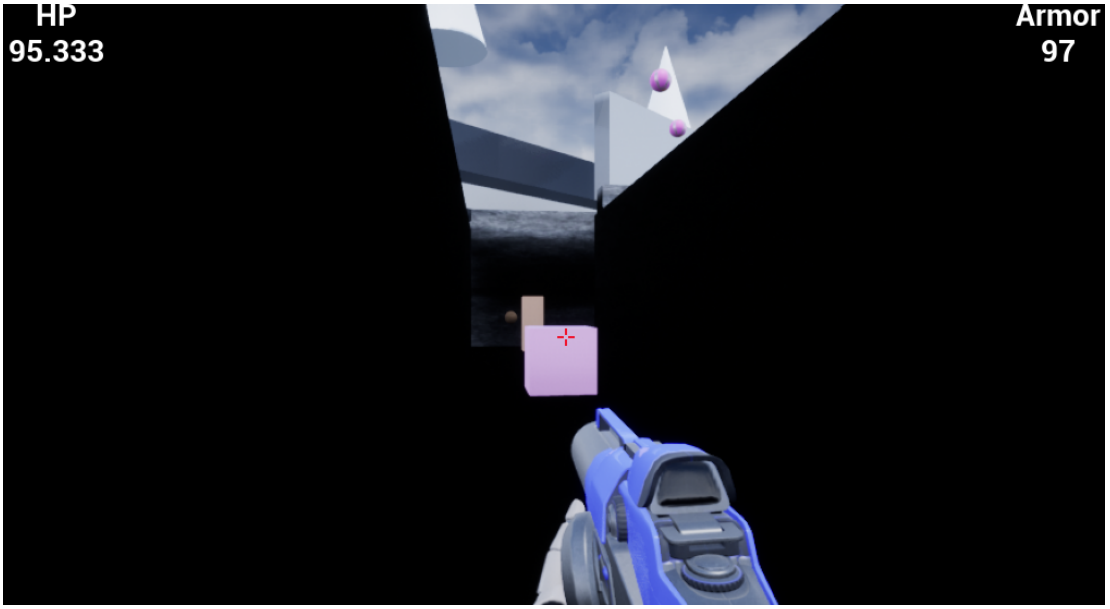
U1

	Description	Threat Level
Enemies	X2 IMP, X2 Pinky	6
Environment	A hallway with plenty of room to dodge projectiles. Enemies get introduced 1 or 2 at a time.	-2
	Total Threat Level	4

In the same way as A1, it acts as the mid difficulty peak of the level, also merging together the Pinky and Imp challenges, but in a special way. If A1 was all about dealing with enemies in a open space area, U1 is about little challenges that build the whole challenge: since it is placed in a halfway, the player still experiences the Pinky + Imp combination but more dosified, as he/she will not deal with all the enemies at the same time.

It also introduces the idea of space constraints in a more extreme way, thing that will be useful for the boss fight down the line.

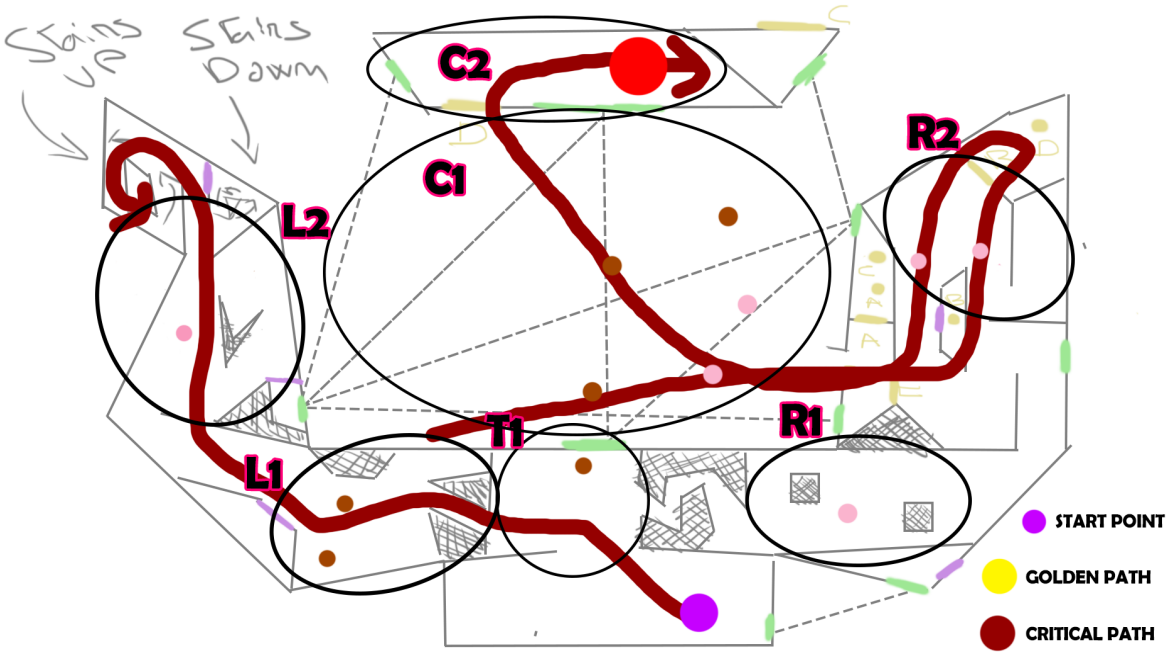


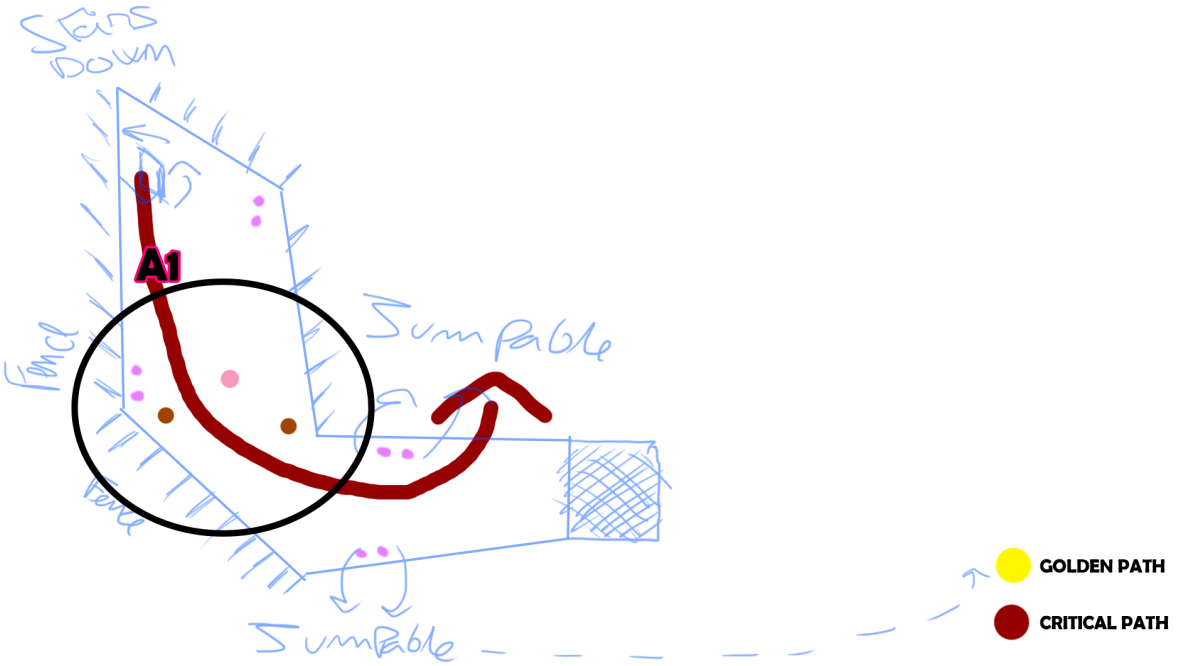


Routes & Difficulty

Critical Path

As mentioned in the Level Summary section, the critical path consist of the following steps:

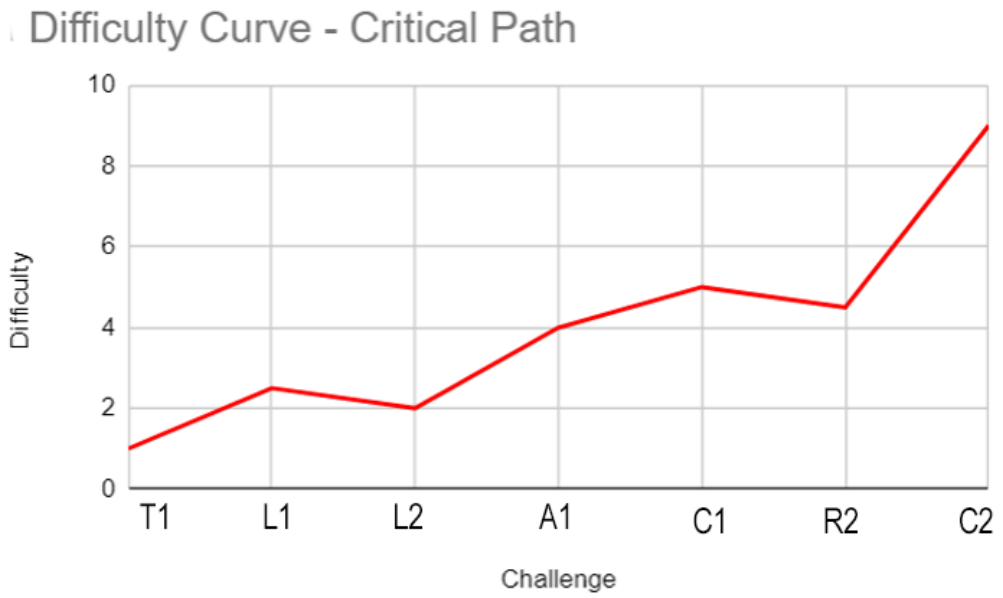




This forces the player to go through the following encounters in this order:

T1 -> L1 -> L2 -> A1 -> C1 -> R2 -> C2

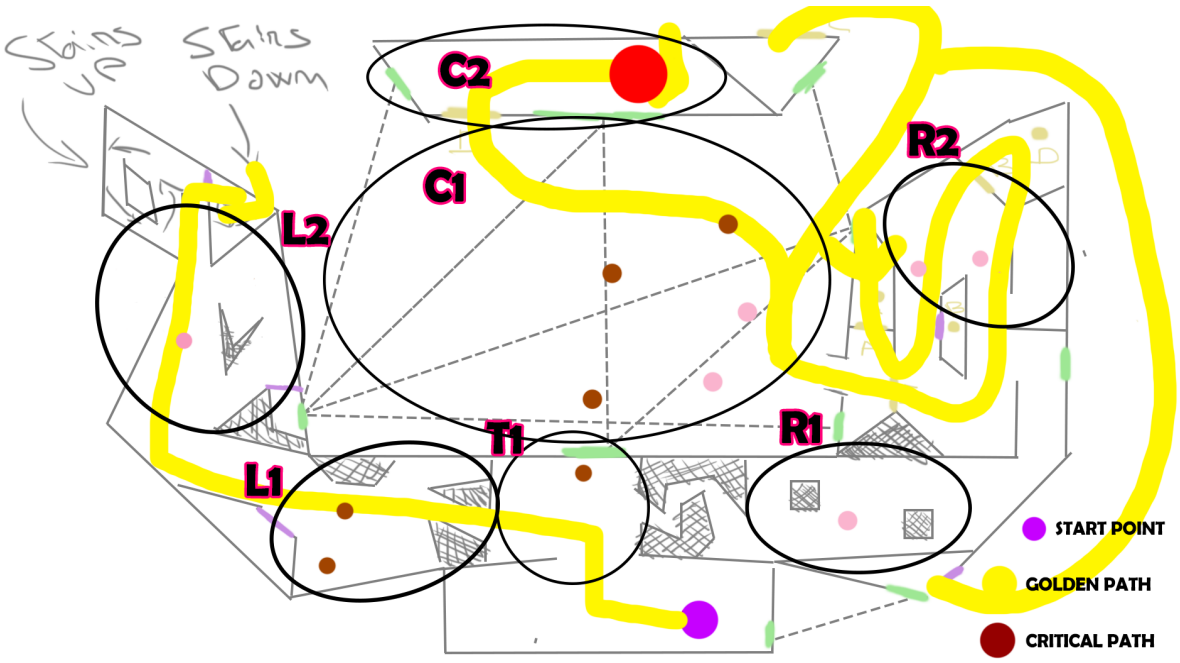
This generates the following **difficulty curve chart**:

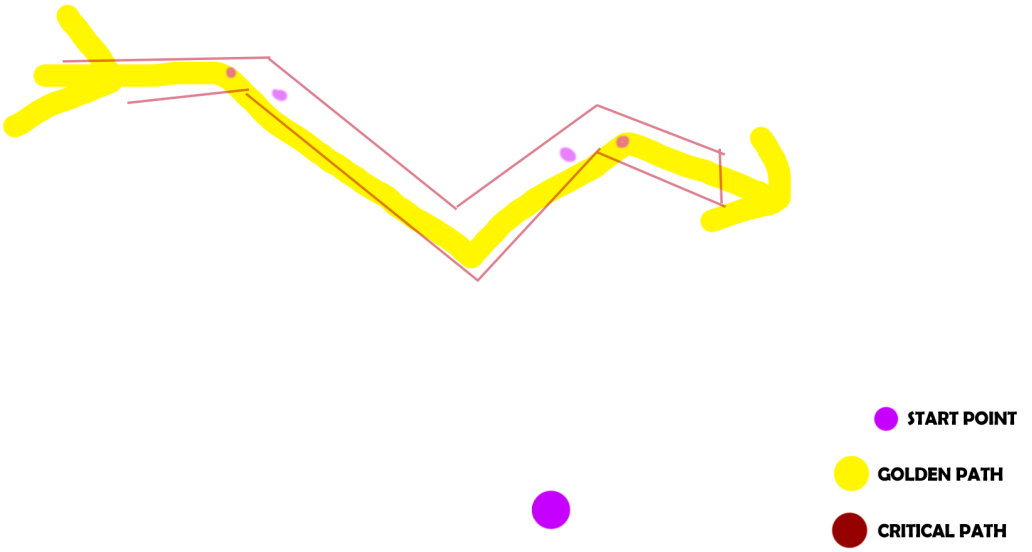


You can check the chart calculations here

Golden Path

As mentioned in the Level Summary section, the critical path consist of the following steps:

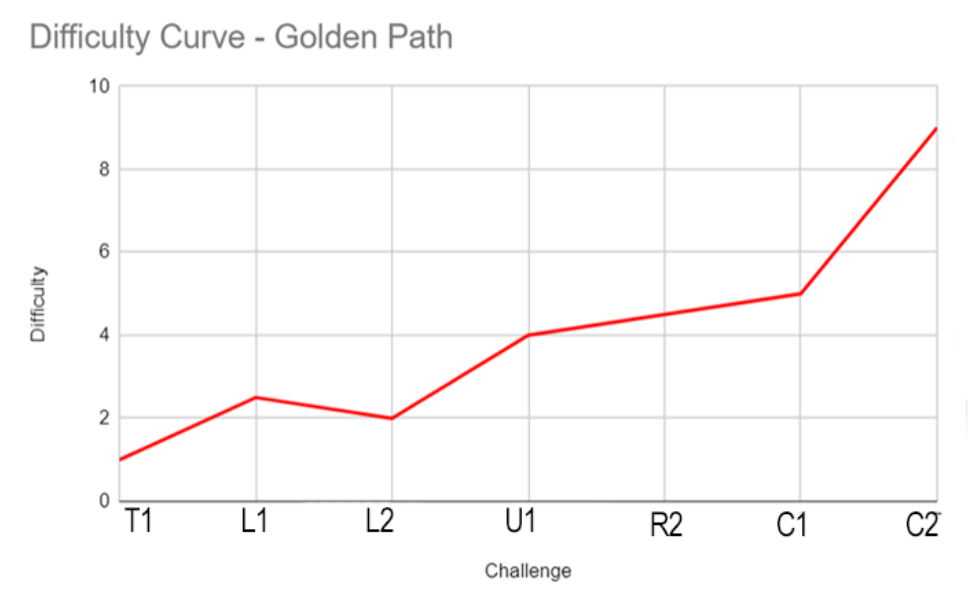




This forces the player to go through the following encounters in this order:

T1 -> L1 -> L2 -> U1 -> R2 -> C1 -> C2

This generates the following **difficulty curve** chart:



You can check the chart calculations here

About Doom

Having taken *Doom* as our main point of reference, **extensive research was done on it**, the fruits of which are shared here, to better contextualise where our decisions came from:

Doom is a first-person shooter where the player has to traverse a series of levels. Throughout them, they will confront a series of enemies that they can defeat with the weapons they start with, or with the ones they might find as they progress. As they do, **utility items, power-ups and keys will be found to facilitate level traversal**, all of which will be lost once the level is completed, or a game over is reached by losing all health points.

The avatar exists as an entity of 32 units of width, 56 of height (later on, making the conversion, **we calculated that 1 *Doom* unit is equivalent to 3.75 *Unreal* units**, a very useful conversion to develop the implementation of our level). The **basic movement is composed of a camera rotation to the sides, with the horizontal arrow keys; and forward - backward movement, with vertical arrow keys** (there is no jumping nor vertical camera movement).

Mouse can be used in the same way as the directional keys, which increases the speed of either camera or player. Horizontal movement can be achieved by pressing alt together with the horizontal arrow keys. Player can increase its speed by running, with no real cost (pressing shift); running is around 30% faster when performed diagonally than vertically, due to X and Y components being calculated independently. To open doors, the "Space" key is used.

The **player can move while shooting** or attacking, at least with most weapons (the chainsaw, for example, is an exception, as it keeps the player latched onto the enemy while the animation is going on). This makes it so that **most of the times, the player can reposition and look for vantage points while taking care of enemies**. To interchange weapons, each weapon has a slot number assigned.

Weapons

Originally, *Doom* used pistol and brass-knuckled fists (infinite attacks, since they don't need ammunition) as level-starting weapons, and then expanded with chainsaw, shotgun, chaingun, rocket-launcher, plasma gun and a BFG9000 distributed throughout the level, as the designers saw fit. All of those weapons, when used, will alert enemies. The attack button can be held to continuously use whatever weapon is equipped, and no weapon switching can be done while a shot weapon is on recharging cooldown. Once all their ammunition is used, they stay on the player's inventory, but are useless.

- **Fist weapon**s: 2 frames start-up with a range of 64 units.
    - **Fists**: similar to the gun, but working only at melee range.
    - **Chainsaw**: melee weapon that substitutes brass-knuckled fists if available (can be toggled in later instalments of the series). Moves the player slightly forward. Has approximately 4 times the fist's speed, damage of 2-20 points per attack, 525 RPM, has infinite duration, and causes mutilation. It is especially useful to minimise risks

at melee range, but because a lot of enemies have ranged attacks, it doesn't provide as much survivability when there is a lot of distance between player and enemies, or when there are ranged enemies within a hord.

- **Fire weapon**: all weapons have a random angle deviation nearing an average of 5° when fired. Each weapon has different ammunition that is consumed separately, and except for the pistol, each are only obtainable from enemies or found in the world, meaning the designers can easily influence how each level is played based on the weapons that are made available..

    - **Pistol**: default weapon that starts with 50 bullets, each dealing 5-15 damage, with a rate of fire of 150 RPM. The first shot is made at the centre of the screen, but if the trigger is kept, the random deviation is applied. A very poor weapon that is used out of necessity or to save resources.

    - **Shotgun**: having 4-16 shells depending on how it is acquired and difficulty mode, deals the same damage as the pistol (5-15 damage per shot), but because each shell shoots 7 pellets, if all land, it can do from 35 to 105 damage. Has 56.8 RPM, thus roughly having an animation of one second. Very effective on the close and even mid-range, especially against hordes of weaker enemies, although it will hardly be able to keep up against larger packs or more resilient opposition due to its low cadence.

    - **Chaingun**: 20-40 ammunition (based on difficulty), 5-15 damage per shot, 525 RPM. Each shot, however, fires two bullets, so damage can potentially be 10-30. The first pair of bullets will not suffer from random deviation. This weapon is all about being fast at the cost of lots of ammunition. It is not particularly strong, but it quickly builds stun, and can dispatch large quantities of weak enemies without problem.

    - **Rocket launcher**: 2-4 rockets (based on difficulty), 56.8 RPM, and a projectile type of damage, meaning that unlike many other weapons, it is not limited to the hitscan

limit of 64 units. The projectile is unaffected by gravity. Its damage can go from 20-160 in multiples of 20; additionally, it can do from 0-128 damage from blast damage, depending on distance (units) to explosion (if an enemy is hit directly, damage has 128 directly added to it). High-speed projectiles (20 map units per tic; 700 per second) and damage make it the best long-ranged weapon, but nefarious at short-range, since the player is affected by the blast damage.

- Can be used for rocket jumping. Also has a horizontal auto-aim, but it seems more like a developer limitation than an intended feature.

- **Plasma gun**: 40-80 energy units (based on difficulty), with each burst dealing 5-40 damage points. 700 RPM, with 3 ticks (0.086 seconds) between shots, and a recovery time of 21 ticks (0.6 seconds) when the trigger is released. A jack-of-all trades that does high damage and is effective at all distances. At a long range, however, it is easy that enemies can move away from the projectile (25 map units per tic; 875 per second), or that even if hit, the pushback will move them to a side, making the following projectiles miss their target.

  - Can be used to light up dark places. Suffers from the same horizontal auto-aim problem as the rocket launcher.

- **BFG9000**: coming with 40-80 energy cells, it deals 100-800 damage per hit, in multiples of 100. The start takes 30 ticks (0.857 seconds) to begin, consuming 40 plasma cells. After impacting, enemies hit by a ray trace within a cone shape roughly at 45° of the impact point will receive 49-87 damage points, within 1024 units (those rays affect cyberdemons and spiderdemons, usually immune to blast damage). Since the dispersion damage is not considered a blast, this weapon can be used at close range, generating very risky but rewarding damage scenarios, but only having a few chances to succeed before ammunition runs off.

Power-ups:

- **Health**: objects scattered around the level, which increase health in different percentages.
    - Health potion: increases health in 1%, to a maximum of 200%.
    - Stimpack: increases health in 10%, to a maximum of 100%.
    - Medikit: increases health in 25%, to a maximum of 100%.
    - Soul sphere: increases health a 100%, to a maximum of 200%.
- **Armor**: objects scattered around the level, which increase armor in different percentages (armor acts as a health substitute).
    - Armor bonus: increases health in 1%, to a maximum of 200%.
    - Armor: increases armor to 100, and can't be picked up if it's already at least that high. Reduces all damage by a third until it is depleted.
    - Megaarmor: increase armor to 200, and can't be picked if it's already that value. Reduces damage by half until it is depleted.
    - Soul sphere: increases health for 100%, to a maximum of 200%.
- **Power-ups:** items scattered around the level, with various effects.
    - **Backpack**: doubles ammunition storage; up to 400 bullets, 100 shotgun shells, 100 rockets and 600 energy cell units. It also contains the lowest acquirable ammunition commented in the weapon section.

- **Berserk**: increase fist damage by 10 until the end of the level, dyeing the screen red for around 20 seconds. Additionally, it puts player health to 100% if it was any lower.

- **Computer area map**: displays areas not explored, unless they are flagged otherwise.

- **Invulnerability**: makes player invincible for 30 seconds. It inverts the color on-screen, and blinks before ending.

- **Light amplification visor**: shows area in full brightness for 120 seconds, blinking back to normal light before disappearing again.

- **Megasphere**: puts player to 200% health and armor (implemented in *Doom II*).

- **Partial invisibility:** adds invisibility to the player, which affects the deviation angle of monster projectile attacks.

- **Radiation shielding suit:** gives immunity to damaging floors for 60 seconds, adding a green haze on screen, blinking 5 seconds before wearing off.

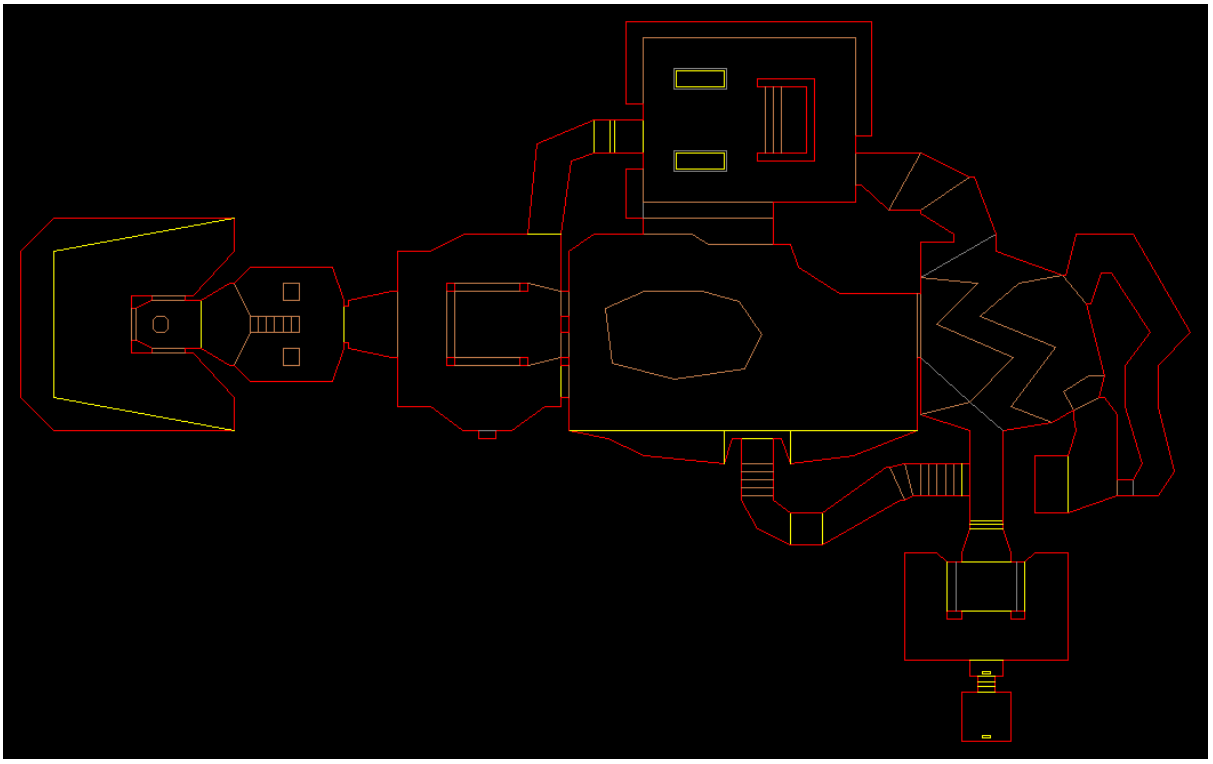
- **Ammunition**: drops from enemies or found in weapons. While already discussed, dropped ammunition can be small or large; the large kind gives five times the amount found in weapons.

- **Keys**: items found on the ground which are used to unlock zones of the level (mandatory or not). While there are 3 color keys, multiples of the same color can be found within one same level.

Map References

As explained before, we're **building our map to fit as seemingly as possible in Doom**. Because of that, we dug up and studied multiple Doom levels before designing our own. Here are the ones which ended up being the main inspirations for us.

- **Doom Map: E1M1 Hangar**



- Extensive use of the horseshoe structure

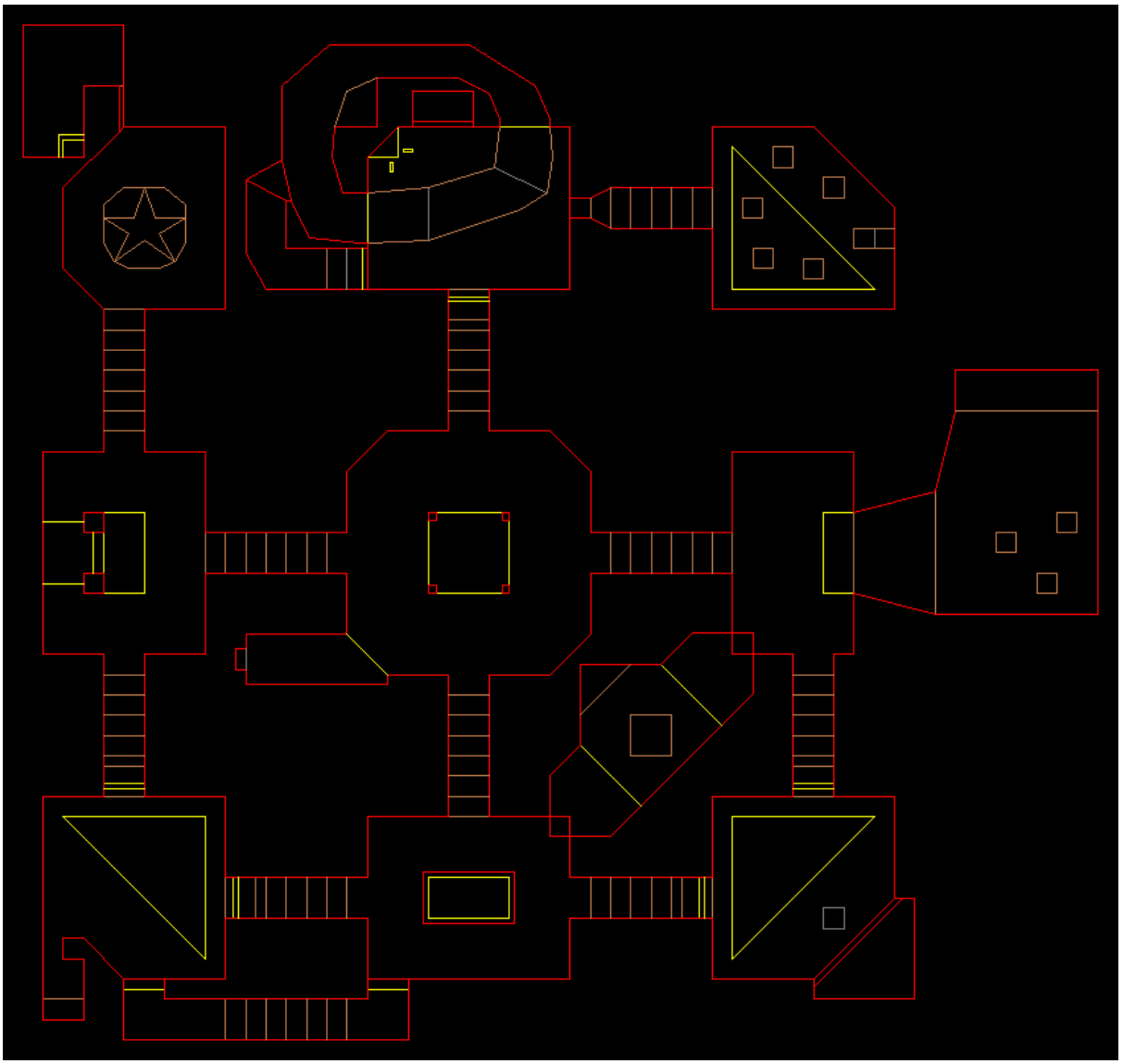
Here we see the **"horseshoe" technique** used to its maximum. It's used multiple times (at the start, bottom right; in the top and in the end, upper left). While the structure presented does not open multiple lines of sight, it is used to create points on **interest in the center and create decision at the start of the structure (right vs left)**.

- Offshoot paths for golden path

Back in Level Summary we talked about the two main approaches in which Doom implements Golden Path elements, one of which being **little offshoot paths**. Here we can see that technique

put to use. The level is rather straightforward, with 3 main sections linked together, but we can find coves and structures that connect them in alternative ways.

- **Doom Map: E1M8 Military Base**



- 2 loops, with exit on one of the branches

We liked how the **map incentives traversing the same paths multiple times**: going trough the loops collecting keys that open other loops. This effect of chaining keycards works really well to create long paths in a small amount of space, which is exactly what we are after.

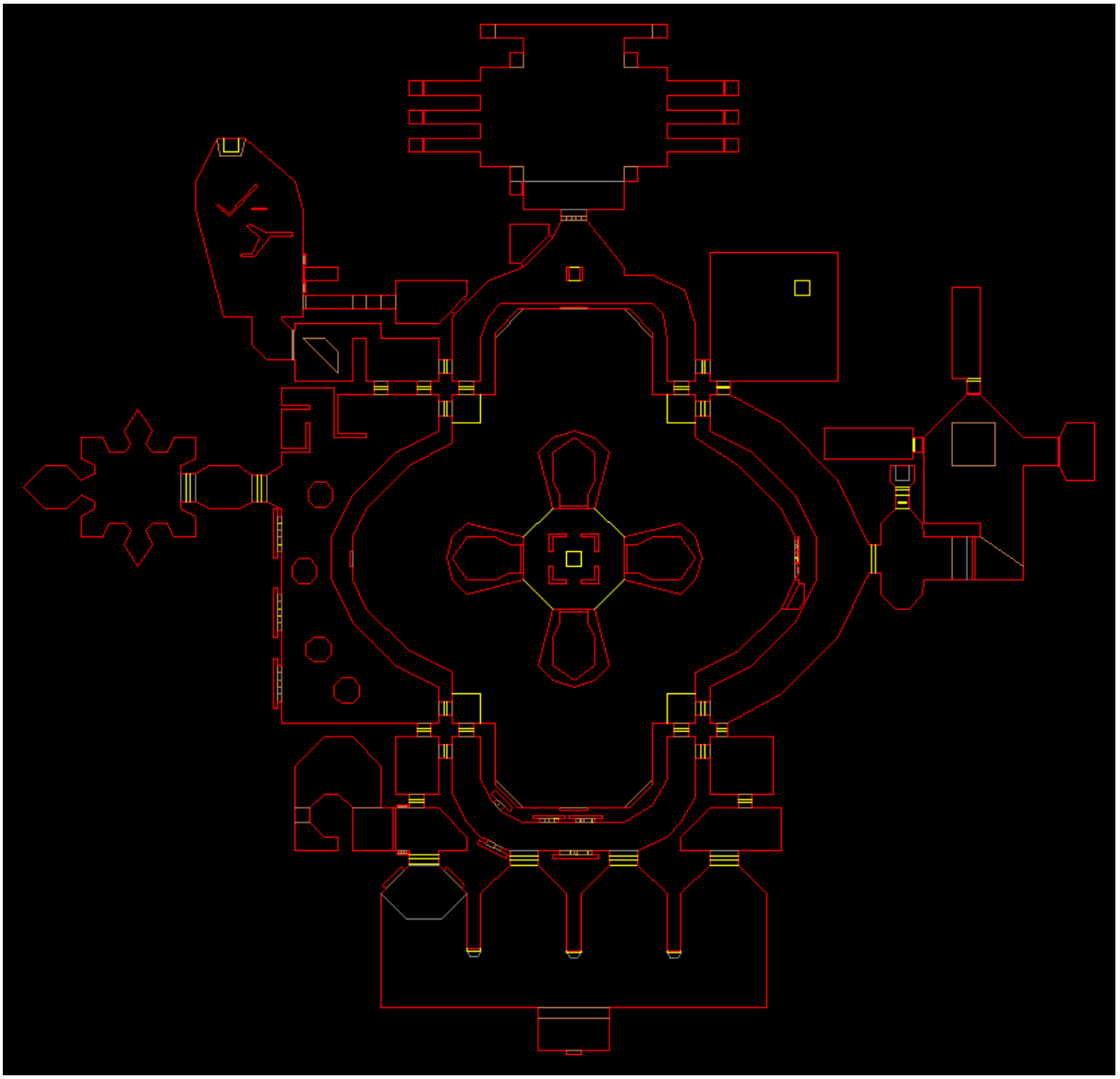
- Ofshoot for golden path

We see here how **Doom implements clear and aggressive golden paths proposals:** it's a complete separate path from the main proposal (loop structure) which the player con go trough to get some goodies. After that, they come back to the critical path. This is the approach we took in our level.

- Secret chambers / closet enemies everywhere

The **"closet enemy"** is a technique Doom uses extensively to create amushes and increase the difficulty of combat situations without the need of creating new enemies or IA patterns. We liked how simple it was, yet effective.

- Doom Map: E3M5 Unholy Cathedral



- One big loop with tons of offshoot paths

We got the idea of a **big central section** from here. The level is structured so a player goes multiple times trough the center, while exploring all the offshoot paths and rooms that encircle it.

- Scouyting for keycards to new paths

Once again, we find a **"linked key" approach to the level layout**, starting different key chains from these offshoot paths which are not mandatory for our player.

Future Iterations

Some things we didn't do because they fell out of scope, while some things we couldn't do because we ran out of time.

These are all the things we wish could be in the level, but are not:

1) **Closet enemy encounter on underground section**

We wanted to have closet enemies, and we wanted it bad. **We even knew how to program it** (we know how to make doors disappear and how to turn on/off enemy aggro AI). The **problem was everything else:** how to mask sound before the trigger, how to make it so it is understandable what is going on, how to properly set the event trigger...
In the end, we decided against it: **cutting this functionality clears a lot of development time for a feature that is seen only once** in all the map and thus is not priority. However, we'd love to have it at some point and implement the underground encounter as intended.

2) **More verticality**

We really liked how the Underground-Base-Ceiling structure ended up. However, it is **only used one** and it's all concentrated in one section of the map. More time (or noticing the issue earlier) would allow us **more design iterations** in this direction. Some ideas were already being discussed (having the right part of the map work as an extension to the golden path, adding something on the ceiling;

having some sort of trapdoor on the outdoor leading to some cellar; connecting the boss room with an underground tunnel to the chamber with the rocket launcher).

3) **Level culmination implemented**

Right now, the intent of the level is killing the final boss, the Baron of Hell. Upon doing that... nothing. **We really wanted to add something there, even a simple fade to black with a win screen** or something like that. However, we were running out of time and, while highly desired, **it does not add to the design of the level in any way** (just to the player experience). We had to greet our teeth and move past it, but we felt we needed some closure there.

These are the thing that fell out of scope, but would be the next steps should we come back to this assignment at some point.

1) **Going to Greyboxing**

While whiteboxing is the minimum that allows us to create the level itself, going to **Greyboxing would allow us to inform extra meaning trough shape**. We didn't have the time and we knew it, so we decided to focus on a good whiteboxing rather than in a "meh" greyboxing, but still: a clear next step is this.

2) **More secrets**

This is a Doom level! **It should be full of secret doors, enemies, weapons, powerups and easter eggs!** With the intended expansion we would do on a "Quuk 2.0", we would surely go all out on this.

3) **More feedback on player actions (sfx and audio both in combat and traversing)**

While the intent of the delivery was on the level design, we're still delivering a playable level. If it can be played it should feel good to do so, out of principle. Having a week (or at least a couple of days) to polish game feel, audio, controls, camera, etc. would be in order.0

Post Mortem

So, after all is said and done, what did we learn? Here is a recollection of all learnings we take from this experience:

- **Focus on gameplay vs focus on level design**

We wanted to do a Doom level. This meant two things: we needed to spent some time developing the level and some time developing the Doom mechanics. Without noticing it too much, **we went out of scope with the Doom features**: multiple weapons, secret doors, multiple enemies... We should've realized earlier that we could've just "faked" the mechanics and spent more time on the design iteration, but alas... we realized too late.

- **Unreal 4 tutorials are subpar**

We were **too used to the Unity tutorials and got "cocky"**. Getting everyone together in a UE4 project, needing to set it up from the beginning and having to develop all the mechanics from scratch took way more time than expected (which compounds on the impact when combined with prior point).

- **Blueprints problems**

We found **blueprints to be a double edged sword**. Yes, we could start iterating faster because of it, but it also made the developing of mechanics slower because of it. From here, we take the following lesson: if this was made just as a level (no gameplay, no need to develop enemies and weapons) blueprints all the way. **If a noticable amount of new code needs to be implemented, it's better to just start from scratch**.

- **Interaction AI - Level**

Enemy **AI interaction with level layout proved to be more complicated than we thought**. Enemies can see players trough the walls, all the different checkboxes on the detection system, making it so all enemies on a room alert at the same time (so you can't just bait them one by one...).

Looking back, we should've started looking into the enemy AI much sooner (having it be part of the research itself, not only in intentionality, but also in how to implement it related to UE4 technology).

- **LoS problems**

A big part of the level is how the differen**t LoS between windows** create player expectation. The fact is that some **LoS are present in the level which were not intended**, but are just consequence of the geometry of the level itself. The lesson? **Take into account the degrees (or radiants) that the cone of vision of the player has**. When designing a level, overlapping that cone to the corners of each window and see how does the LoS expand because of that. Doing so would've prevented the unintended LoS.

- **Need of valves / gates**

Designing combat encounters in a level without valves or gates is rather challenging. In a Doom setting, there's some **soft "solution", since the enemies come at you so far than trying to retreat and slamming into a wall** easily converts into death. However, looking back, the implementation of a simple door (nothing fancy needed) would go miles to making combat encounters easier to design (and more effective).

- **The importance of interaction of movement & level design in combat**

This is something we found in our level and **immediately recognized as one of the Doom core experiences**: it's really satisfying to dash across the objects, using them as cover or to create obstacles between oneself and melee enemies. The fact that we got to feel that in our level (altough in a much less refined implementation than in Doom) made it enjoyable in the combat segments.

- **Hiding secrets based on player camera and intended path**

This is something we learned from Doom and we found out **it works wonders**. Taking into account camera angle + intended path **let's you hid secrets "in plain sight"**, which creates really rewarding moments for the player when they find out (because you've guided them there, of course... but they don't know that).

- **Doors, keys and secret doors: an absolute hit**

They are **easy to understand, fast to implement** and allow designers to **up the map's complexity and richness** up four notches. They are an elegant way to make the player go trough the same places numerous time, create expectation, allows for mysterious rewards in the form of keys... all around, we were impressed on how much better the level felt upon implementing them (specially taking into account how fast they came to be when compared with other features).

- **Lights in Unreal 4**

We felt we needed to comment on lights as their own bullet point. We've mentioned multiple times the complications that come with UE4 on the technical side. Lights take the crown here, by a landslide. **They were problematic since the beginning and still are, even with the level finished**. If done again, this should be moves to prio 1 in the development pipeline.

References

Doom Wiki:

https://doomwiki.org/wiki/Entryway

Decino's How Doom's Enemy AI Works:

https://youtu.be/f309P9x1eCE

Doom (1992)'s Source Code:

https://github.com/id-Software/DOOM/blob/master/linuxdoom-1.10/p_enemy.c

Doom level Analysis:

https://www.youtube.com/watch?v=ptHurafdCoQ&ab_channel=Chubzdoomer

Doom 2 level Analysis

https://www.youtube.com/watch?v=w9oG2LBuMwY&ab_channel=Chubzdoomer

Doom gameplay Analysis:

https://www.youtube.com/watch?v=yu00bGjCA7Q&ab_channel=GameMaker%27sToolkit

Doom correlation between level and gameplay Analysis:

https://www.youtube.com/watch?v=-hbp_sXnnHI&ab_channel=AndrewYoder

Doom gameplay by his level designer:

https://www.youtube.com/watch?v=YUU7_BthBWM&ab_channel=IGN

Doom maps database:

http://www.classicdoom.com/doommaps.htm

Link to Presentation

Presentation - LD-P2-QUUK-AdriàÀlexJoseOscarFerran

Link to Release

QUUCK: Release 1.0

Link to Video Walkthrough

QUUK: A Doom inspired level | Level Design Walkthrough*

*In the delivery it was stated that all the members of the team must participate in the video walkthrough but we decided to just do it with 2 members. This decision was made because the video with 5 people was too chaotic and we were not happy with the result, we decided to cut down and have a more friendly/podcast/sports commentary approach to it, where there is a main caster and a secondary one with more specific and technical analysis.