5-2015

Experimental Boss Design and Testing

Joseph P. Mistretta
East Tennessee State University

EXPERIMENTAL BOSS DESIGN AND TESTING

By

Joseph Mistretta
The Honors College
University Honors Program
East Tennessee State University

Date 5/5/2015

Joseph Mistretta, Author

Todd Emma, Faculty Mentor

Dr. Scott Koterbay, Faculty Reader

Greg Marlow, Faculty Reader

ABSTRACT

Over the years, gaming has developed rapidly from simple pixel-based experiences to fully blown three-dimensional worlds. As developing technologies improve, so does the complexity and flexibility of what can be created. Encounters, along with all aspects of any gaming experience, have evolved along with the technologies that create them. These intense combat instances, often times referred to as "bosses", represent a chance for the developer to challenge player skill, cooperation, and coordination. In addition to being major challenges, encounters also allow players to feel a sense of progression as they learn and adapt to mechanics incorporated within an encounter's design. Eventually these mechanics are mastered, and surmounted to a lasting sense of accomplishment and success. This project details a personal process of encounter design from initial conception to eventual player testing, along with design choices, outside influences, and development methods. These were ultimately utilized in an attempt to create an engaging and successful boss encounter.

TABLE OF CONTENTS

INTRODUCTION

Like all aspects of gaming, encounters have gone through extreme evolution since their first appearance in the earliest video games. Not only has their complexity and scale evolved, but their definition has changed as well. “Encounter” can be used to describe many things in games, whether it be a common enemy, human opponent, or puzzle that needs solving. The term is subjective in many ways, and as a result is not always used to describe the typical “boss” experience. Some definitions can be as broad as to include any form of combat whatsoever. With so many different definitions for encounters, how do we know which one can be used to describe the encounter developed in this project? As stated before, the definition is highly subjective, and was therefore chosen by the developer. The specific classification of a game “encounter” chosen for this project falls in line with the recent MMO (Massive Multiplayer Online Game) definition. This definition describes a group of players, each with one of three traditional roles, teaming up to defeat a large enemy that would be impossible to defeat alone. The encounter developed in this project requires high levels of coordination and teamwork, as players deal with numerous hostile abilities and mechanics. Adhering to this definition, along with extensive personal design, a boss encounter was created and tested by groups of players. Ultimate success of the encounter, when compared to initial goals, was determined based on player feedback and performance. Please note that the focus of this project was on the design, creation, and testing of a boss encounter, rather than its visual appeal (Stout).

GOALS: WHAT DEFINES SUCCESS?

Success can be determined by multiple aspects of a boss encounter. Whether it be encounter integrity, player adaptation speed, performance or feedback, success is ultimately a term that needs definition in this project’s context. My goals as the encounter designer were to create a functional, engaging, and challenging encounter that multiple players, regardless of prior gaming experience, could defeat in a reasonable amount of time. In addition, players were intended to be occupied with encounter’s mechanics at all times, regardless of player role. For the sake of clarification, this project specifically defines “a reasonable amount of time” to be between ten and thirty boss attempts, based purely on personal preference and expectations based on the encounter difficulty. As a result, success for this project is defined as “player feedback that supports the encounter’s design goals”. The encounter is not intended to be defeated by a new group of players quickly, as recognition and mastery of the mechanics are needed to do so. Player performances and results (whether they defeated the encounter or not) have great influence on feedback tone. As such, the distinction between positive and negative feedback is not included in the definition of success, as its importance is diminished. In order to concretely support whether or not the encounter was a success, each player filled out a review sheet after their testing was complete. This review sheet allowed players to rate whether each design goal was supported in the playable encounter itself, as well as supply any personal feedback of their own. The testing and feedback processes are further elaborated on in their respective sections of the paper.

ENCOUNTER CONCEPT AND THEME

The concept of an encounter can be interpreted in many ways, as previously stated. An encounter can be a single event, a series of events, or something totally different. The concept of the encounter designed for this project focused on a tiered environment that ultimately led to a final confrontation of the boss entity. This environment is divided by large shields, which serve as both an advantage and disadvantage when dealing with the encounter's main ability (See Purge Protocol in the Mechanics Design section). This main ability was ultimately the focus point of the encounter, and the first mechanic created. This resulted in the mechanic becoming the base from which all other mechanics were designed around. Even the environment was created in a way to facilitate the full impact of the mechanic and its concept.

Players begin in an enclosed space, and must eventually complete a set of actions to move on to the next enclosed area. With each new area, the difficulty of mechanics increase, either by multiplying their instances, increasing their frequency, or simply making them do more damage. Players progress through four of these tiered areas to conclude the encounter in the final boss location, where the difficulty reaches its peak. In addition to this tier-oriented design, timing mechanics were also implemented to give players a "time limit" on how long they can linger in any certain area before they are overwhelmed. These mechanics are often called "enrage mechanics", and can be found in the Mechanics Design section.

Going hand-in-hand with the concept of the encounter, the encounter's theme is also an extremely important part in encounter design. The theme of an encounter can be anything from fighting a dragon atop a volcano to slaying a squid at the bottom of the sea, but in the end is an important part of the design process. Encounter theme often drives how the designer creates the mechanics of the fight. For example, a giant squid at the bottom of the ocean probably wouldn't shoot fireballs at players, and thus the mechanics are driven away from that aesthetic style and more towards one such a creature would (most likely and within reason) naturally have (Silva). That being said, the theme of the encounter created for this project is centered on mechanical elements. Players find themselves trapped in an AI-controlled security system that is intended to eliminate any organic material that enters it. As such, mechanics included in the encounter are designed around electricity, mechanical components, and advanced technology. Examples of these mechanics, which are explained in detail within the Mechanics Design section, include automatic gun turrets, flame vents, force fields, sawblades, and self-piloting death-spheres (Miozzi)!

MEDIUM: PROGRAMS USED

The programs used for the creation of all elements within this project include Epic Games' Unreal Engine, Autodesk Maya 2015, Adobe Photoshop, and the video/audio capturing program "Action!" by Mirillis. All modeling was done within Maya, while environment layout was done in Unreal. All coding for this project was done using Unreal Engine's built-in "Blueprint" system, which is a visual scripting method of coding rather than a traditional line coding system like C++. Even though I have experience with multiple coding languages and visual scripting programs, Blueprints were chosen because I felt it was the most user-friendly way to achieve what I was trying to accomplish in the shortest time possible.

MECHANICS DESIGN

Below are outlines detailing all designed boss and player mechanics in the project. Various information is given in the description of each individual mechanic. This information can include mechanic function, damage/healing values, occurrence frequency, importance level, and intended gameplay response from players. In addition, a short explanation on the reasoning behind a certain mechanic's design can be found directly after the mechanic's description. These explanations will be in italics. All mechanics in this section are listed as they were prior to post-testing balancing changes (Cameron).

"Cooldown" is a term used for how much time must elapse between usages.

"Damage Multiplier" is the value used to calculate damage taken by players.

Boss Mechanics

- 4 Sectors + Final Boss Sector
- ~70s / Sector + ~45s for final boss sector = ~5.42 min fight
- Tank Focused, DPS Focused, Healer Focused.
- Major Mechanics

Sector Barriers – Large barriers present between each sector that cannot be destroyed directly by players. Players cannot move or shoot through these barriers. When a barrier is destroyed, it will slowly start to regenerate. Players must move to the next sector quickly in order to not be trapped, as there is no way to destroy a regenerated sector barrier.

Tiers the encounter and sets up the distinguishable areas from one other.

Purge Protocol (Base Mechanic) – The boss purges all organic material in the sectors. Any players that touch the purge are vaporized. Players in Field-Entity Alcoves are not affected by this ability. Executes every 70 seconds.

Paces the fight. Sets time limit in each sector. Enrage Mechanic.

Pressure Plates – The security expels large amounts of pressure as it executes its numerous protocols, causing floor plates to rise up and down as pressure is released from the system. Every 10 seconds, a floor section of the current sector will raise or lower.

This is intended to act as cover during the encounter, as well as a means to deal with the Blue Soulburn Orbs mechanic.

Auto-Turrets – Turrets present on the walls in each sector. These turrets target and fire at players. Only turrets in the current and previous sectors are active. The number of turrets increase with each sector. These cannot be damaged or killed. Fire every 3 seconds. Every 80 seconds, all turrets become overcharged for 5 seconds. Overcharged turrets fire every 1 second.

> *Turrets are intended to be overcharged as players' progress to a new area, after the Purge Protocol passes. Turrets function as a constant damage threat, but can be avoided.*

Soulburn Orbs (+ DPS Focus) – Orbs that spawn in set locations within the current sector. Two different types of these orbs spawn, each with their own functions. One red and one blue orb will spawn in each sector every 45 seconds.

> Blue Orb - These orbs spawn suspended in the air. If a player touches them, they are destroyed and heavy damage is dealt to that player. If the orb is not absorbed in 30 seconds, it explodes and deals heavy damage to every player.
>
> > *Tanks are intended to soak the high damage from touching these orbs, due to their lower damage multiplier. Pressure Plates offers a means to reach these elevated orbs.*
>
> Red Orb – These orbs spawn at ground level and remain stationary, and mimic the Arete's Rage mechanic. These orbs deal 5 damage to every player, every 10 seconds. Red orbs can only be killed by damage.
>
> > *DPS players must kill these orbs by damaging them.*

Shred Saws – Saws that launch themselves towards player positions. When they reach the location, they become stationary and spin in place, slightly above the ground. Touching the edge of the saws deals damage. A saw is thrown every 25 seconds. Saws are not throw after the Final Protocol is executed. Saws will not be thrown into Field-Entity Alcoves.

> *Serves as obstacles that can be strategically placed based on player positioning.*

Bumpers – Additional enemies that spawn throughout the fight. These enemies bounce around towards players constantly, and deal damage to any players they come into contact with. These enemies cannot be destroyed by damage, and ignore all projectiles except the tank's primary fire. Bumpers spawn every 30 seconds, and automatically spawn every time a Sector Barrier is destroyed.

> *The tank player must use their primary fire to knock back these enemies. The only means of destroying them is by knocking them into open Flame Vents.*

Flame Vents – Vents present in each sector at certain locations on the ground. These vents begin closed, and open after the current sector has been active for a certain amount of time. These vents act like pits, killing anything that falls in them. Vents open/close every 30 seconds.

> *Environmental obstacle, and a means to permanently destroy Bumpers.*

Field-Entities – Additional enemies that are present in every sector's Field-Entity Alcoves. Two are present in each sector, and must be killed in order to progress to the next sector. When both of a sector's entities have been killed, that sector's barrier is destroyed.

Low-health enemies that must be killed by DPS players.

Field-Entity Alcoves – Small areas on each side of every sector that house the Field-Entities. These areas are blocked by alcove barriers that prevent anything from entering/exiting. These barriers can be destroyed by damage.

Arete's Rage – Enrage Mechanic. The boss continually pulses all sectors with energy, damaging all players in the encounter every 10 seconds. Cannot be avoided. Every time a Purge Protocol is executed, the pulse becomes 1 second faster.

Unavoidable damage source, and another timed factor in the encounter.

Boundless Rage – Pulsing damage that is dealt to all players in the encounter every 2 seconds, cannot be avoided. This damage is only dealt if a sector has only one Field-Entity alive.

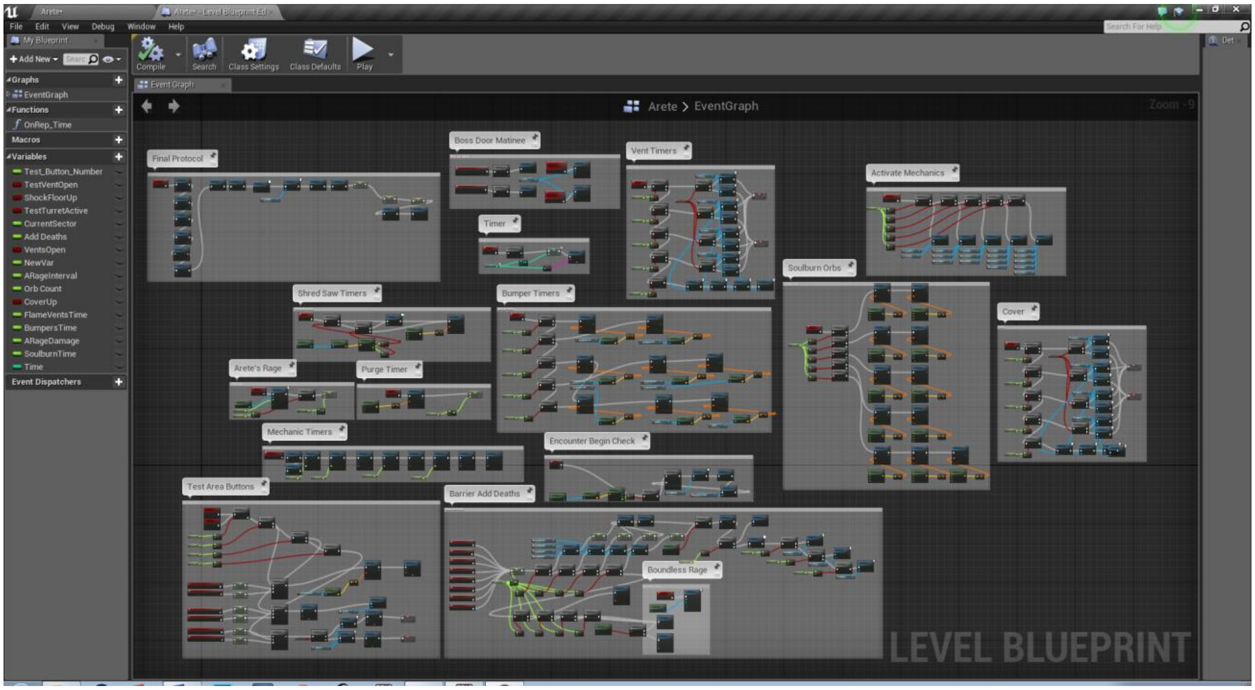
Requires players to kill each set of Field-Entities at roughly the same time.

The Final Protocol – Ability activated when the last sector's barrier is destroyed and the players access the boss sector. Several changes to previous mechanics go into effect.

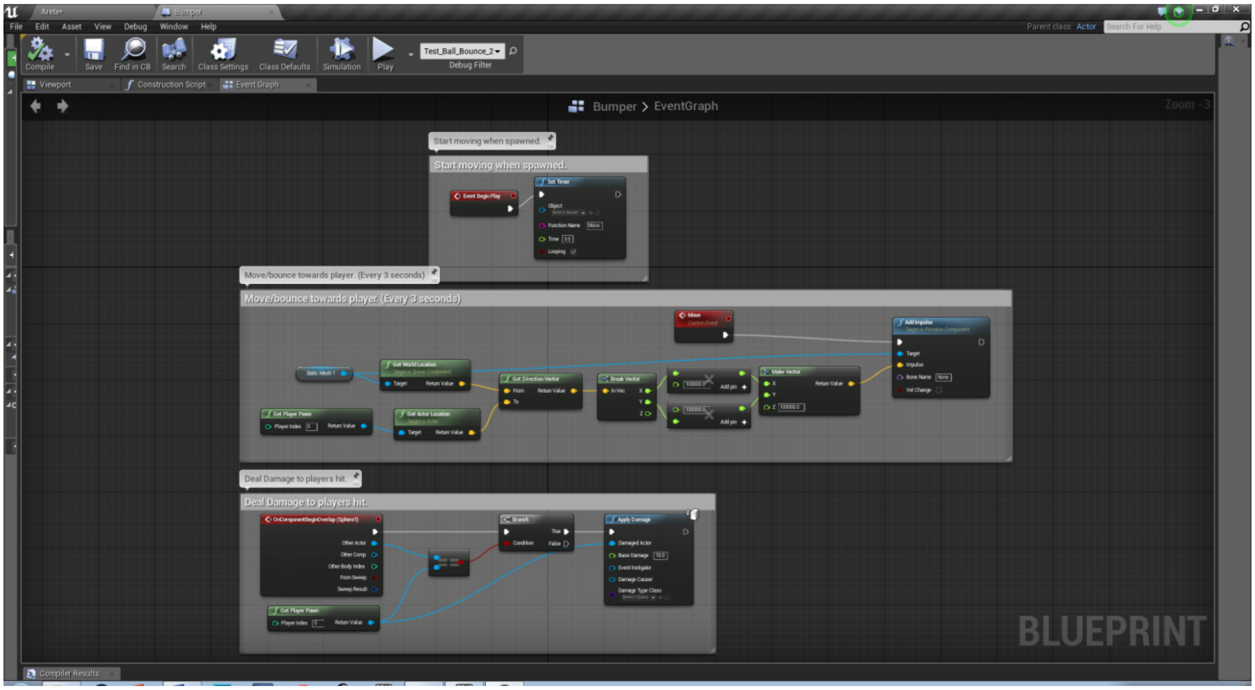
- A 15 second countdown is initiated, when completed the boss Sector Barrier is reformed. Keeping all players in the boss sector, or out of it permanently.
- A Bumper is spawned in the boss sector every 5 seconds.
- All Flame Vents are closed and all Pressure Plates are lowered.
- Arete's Rage pulses every 2 seconds, and increases in damage by 1 every pulse.
- The boss rotates a Purge Protocol at its location constantly, forcing players to move in a clockwise fashion constantly.
- Soulburn Orbs spawn in sets of 3 every 20 seconds, two blue and one red.

NOTE: The boss itself has a very small amount of health, and can be killed quickly by players. Thus the "overpowered" abilities enabled during The Final Protocol are not meant to be active for long, as this would result in a far too difficult final phase. The encounter is focused on the process to reach the boss, rather than a drawn out final confrontation.

An image of the main level blueprint, which contains all the visual scripting code for the main timers and processes of the encounter. The coding for individual parts of the encounter, like the code that runs Auto Turrets and Bumpers, all have their own blueprint file. Seen below.



The Bumpers specific blueprint file. This file tells the bumper to execute the “Move” code method every three seconds. The “Move” method applies a force on the Bumper in the direction of the player, as well as an upward force simultaneously. This mimics a jumping motion towards the player every three seconds. Finally, the code constantly checks if the Bumper is overlapping with another object. If it is overlapping with an object, and that object is a player, it applies damage to that player.



Player Mechanics

Damage Dealer (DPS) Role (Max Health: 100, Damage Multiplier: 1.0)

"DPS" (Damage per Second) is a common MMO term for a player whose main goal is to deal large amounts of damage to the enemy. This damage often times becomes the driving force that progresses players through the multiple stages of boss encounters.

Primary (Left Mouse) – Damage Bolt

- 5 Damage / Bolt

Alternative (Right Mouse) – Overcharge

- Increases damage and healing taken by 200% for 5 seconds.
- 45s Cooldown

Special (E Key) – Reincarnation

- One-Time use ability.
- Places an aura around the player that lasts for 2 seconds. If the player's health hits 0 within those 2 seconds, the player's health is set to 25 and their damage multiplier is set to 0.0 for 2 seconds. Afterwards, their damage multiplier is set to 1.2

Healer Role (Max Health: 100, Damage Multiplier: 1.5)

A "Healer" describes a player who supports the party through healing capabilities used to counteract damage taken. A healer enables the party to last longer by replenishing health points lost during the encounter. Though they usually do not focus on dealing damage, healers are an essential party to present-day MMO encounter progression.

Primary (Left Mouse) – Healing Bolt

- 5 HP/ Bolt. Each bolt passively heals the Healer 0.25 HP.
- Deals no damage.

Alternative (Right Mouse) – Healing Patch

- Places a patch on the ground that heals players inside it 5 HP/s for 10 seconds.
- 30s Cooldown

Special (E Key) – Healing Aura

- Passively heals all players 5 HP/s for 10 seconds.
- 65s Cooldown.

Tank Role (Max Health: 130, Damage Multiplier: 0.8)

"Tank" is a term used to describe a player whose goal is to take large amounts of damage so the rest of the party does not have to. Usually wielding higher defense and health values, tanks are often the ones that deal with the boss directly, and take large amount of damage that would be unmanageable by other player roles. Because of this, a tank's abilities are usually tailored to reduce personal damage.

Primary (Left Mouse) – Force Bolt

- Emits force physics on whatever it hits. (Used for Bumpers)
- Deals no damage.

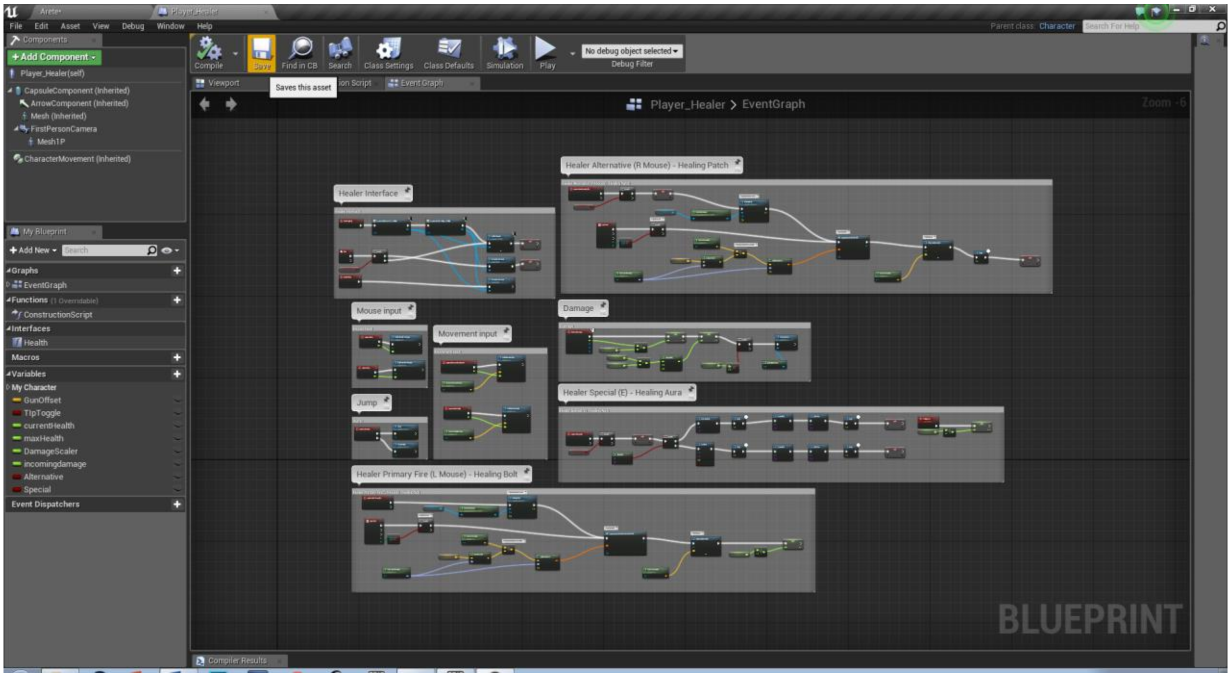
Alternative (Right Mouse) – Exoshield

- Places a shield that is immobile at the tank's location, above the tank. This shield blocks any projectiles it touches. Lasts 10 seconds.
- 55s Cooldown.

Special (E Key) – Personal Shield

- Places a personal shield on the tank that changes the tank's damage multiplier to 0.05 for 3 seconds.
- 65s Cooldown

Each player role has their own blueprint file, much like Bumpers and Auto-Turrets. This blueprint file (Healer Role file) contains all the code for keyboard and mouse input by the player, as well as all the code for the Healers three abilities.



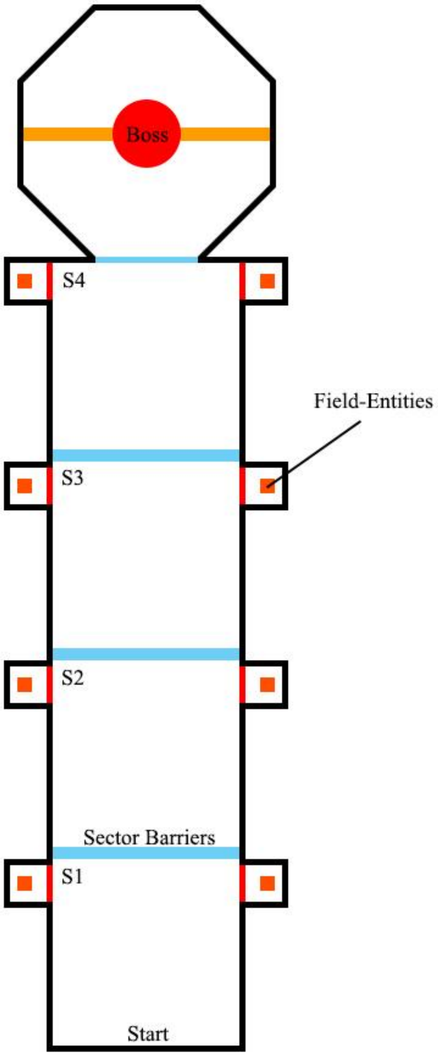
The design style of the player abilities coincide with the need for cooperation to defeat, or even progress, the encounter. Several boss mechanics were intentionally designed to only be handled by certain player role abilities. This ultimately sets an easily distinguishable set of jobs each player must do in order to keep the mechanics at a manageable level. Tank players need to eliminate Bumpers by knocking them into open Flame Vents with their primary fire, while absorbing the high-damaging blue Soulburn Orbs that spawn on the map. Damage dealing players are responsible for eliminating Field-Entities, their protective shields, and red Soulburn Orbs so the group can progress through each sector. Healers arguably have the most stressful job, having to deal with damage taken and healing the party so no player dies. The “shoot-to-heal” style of healing was chosen because it ultimately fit well with the other player abilities and the first-person format. This was also the easiest way to delegate all healing to one specific player role, and the easiest way to have a single player interact with other players in a way that was not obtrusive (Stout).

The encounter was designed in a way that facilitates groups of six players attempting the boss. The intended group composition included one tank, two healers, and three damage dealers. As such, the number of mechanics each player role was responsible for was tuned to accommodate the number of players fulfilling that role. Tanks deal with a moderate amount of mechanics, while DPS players must deal with a few more. Healers deal with mechanics based upon group performance and damage taken, resulting in a default safe number of two healers. In addition, relevant information such as Field-Entity shield health, red Soulburn Orb health, the amount of time Flame Vents are open, and other important aspects of the fight were tuned to values/intervals that could be handled by the number of players who must deal with them. In other words, three damage dealers have enough time to kill both Field-Entities before a Purge Protocol kills the group, and a tank player has enough time to eliminate several Bumpers while Flame Vents are opened so the group is not overwhelmed.

The rate at which players can fire their primary ability was not gated behind any limiting fire interval, as this felt like it restricted the output of each player. Instead players are able to output their primary ability as fast as they want, with the healing/damage values being relatively low per use to account for high frequency use, or “spamming”. This makes the players responsible for their own damage/healing output, avoiding mechanics being blamed for subpar gameplay. Each player role was given a unique ability that can be used to effectively deal with certain mechanics of the encounter in unconventional ways. For example, a damage-dealing player who cannot make it to safety before getting hit by the Purge Protocol can trigger their “Reincarnation” ability right before getting hit. This one-time-use ability will enable the player to revive and continue the fight with their teammates after taking fatal damage. These unique abilities offer both wiggle room for mistakes, as well as the means to develop alternative strategies to defeating the encounter. These abilities can also be very effective based upon group coordination, or can be forsaken all together based upon the groups preference and playstyle. These design choices were intended to give players options on how to deal with mechanics, rather than constrict them to a single style of play. In the end, the encounter was designed in a way that requires certain events to happen in order for players to achieve victory, yet offers multiple methods to execute such events.

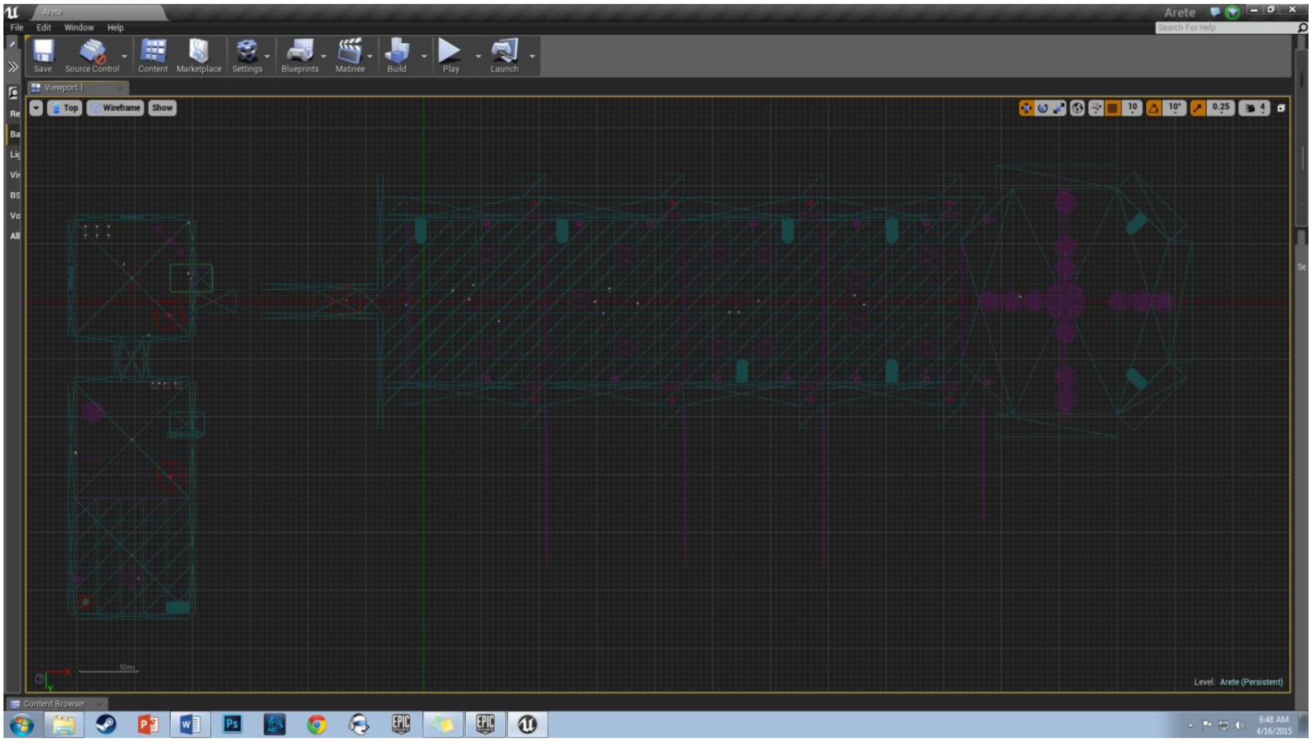
LEVEL LAYOUT

Below is a diagram that lays out the encounter environment, along with an outline of player progression through the environment. This includes all player-accessible areas during the encounter. This does not include the testing areas and role-selection hub present before the boss area.

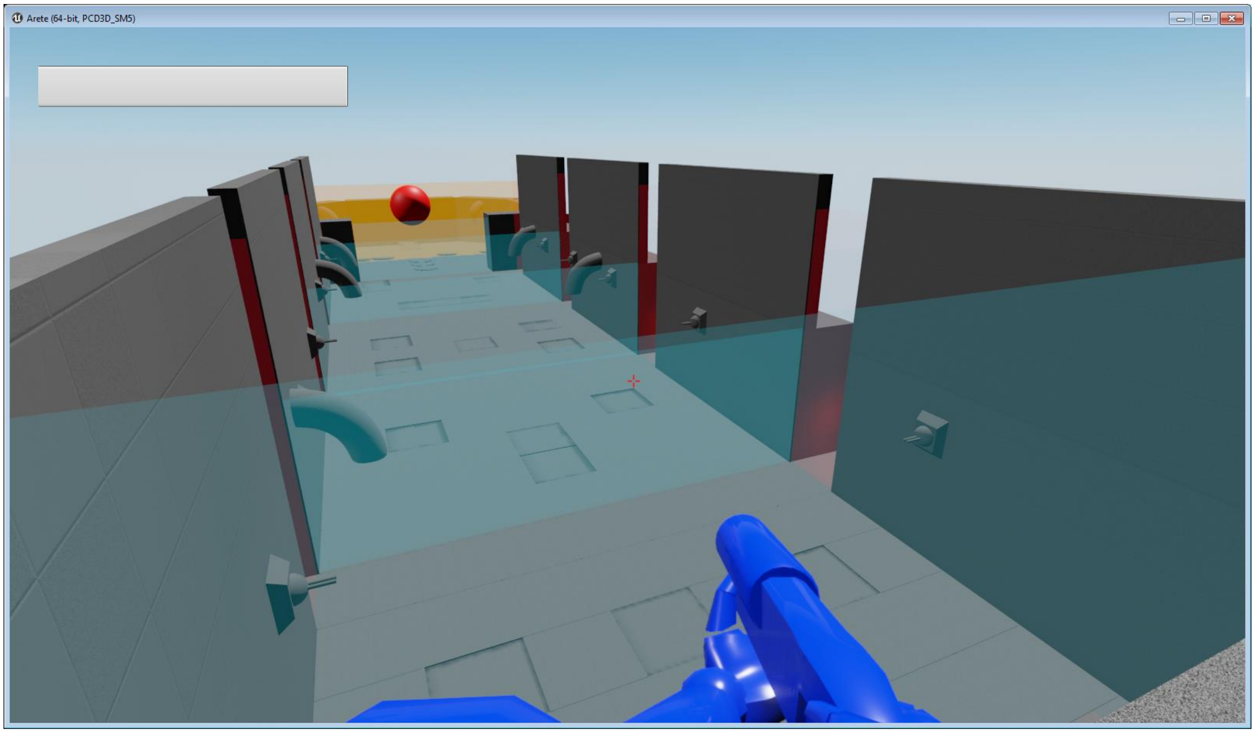


Players start at the southern-most point on an elevated platform, labeled START on the diagram. The encounter begins automatically when players jump down and land in sector one (S1). Each sector is headed by a sector barrier, through which players cannot shoot or move. On the northern end of each sector, there are two Field-Entity Alcoves that hold the Field-Entities inside, both on the west and eastern sides of the sectors. These alcoves are blocked by alcove barriers that require damage to destroy them, represented by the red sections of wall in the diagram. Once these alcoves are exposed, they offer a safe place to avoid the boss's Purge Protocol ability. The environment was specifically designed to offer no protection from this ability until a Field-Entity Alcove is exposed. This interestingly makes the boss's most deadly ability fluctuate between being extremely dangerous, and completely negligible. After the Field-Entities inside both alcoves are destroyed, the Sector Barrier of the current sector becomes inactive for a short amount of time, allowing the players to move on to the next sector. The Sector Barrier reforms after a short amount of time, preventing the players from utilizing previously cleared Field-Entity Alcoves. This process must be repeated three more times to access the final boss sector, all while dealing with increasingly difficult hostile mechanics. Once in the final boss sector, the players become trapped as constant damage ramps up until it becomes unbearable. Players must quickly destroy the boss while rotating in a clockwise fashion to avoid the stationary Purge Protocol the boss constantly rotates around himself, represented by the orange line on the boss.

Top view of the map layout (wire mesh), the areas on the left side of the image are the HUB area that players spawn into, and a testing area where players can see a few select boss mechanics in a controlled environment without starting the encounter:



View of the encounter area in game:



INFLUENCE: EXAMPLES OF INDUSTRY ENCOUNTER DESIGN

World of Warcraft's "Garrosh Hellscream" Encounter

Blizzard Entertainment's "World of Warcraft" franchise is a juggernaut in the MMORPG gaming scene. With eleven years, five expansion sets, and millions of subscribers under its belt, World of Warcraft is the most successful MMO game ever to be created. As such, Blizzard has defined many gameplay aspects that have become common, and almost expected, in any MMO game or experience. One of these aspects is the raiding gameplay niche that this project focuses on. With so many years of development, Blizzard has been able to master the process of raid encounter design through trial and error, experimentation, and polishing to create well designed, engaging encounters. One of the more recent additions, Garrosh Hellscream, can be used as a good example of top-tier encounter designs, and the many elements that go in to creating them.

Strictly focusing on mechanic and design oriented aspects of the encounter, rather than ascetics, Hellscream is an extremely long and epic encounter, usually ending shortly after the 10 minute mark. Being the "endboss" of the expansion in which he is set, all player efforts lead up to one final confrontation with this enemy. As such, the encounter's length is not necessarily unexpected, as major encounters in World of Warcraft usually contain more mechanics. The encounter is a multi-phase fight, with transition phases in between each main phase (Wowpedia).

The encounter yields a rather small amount of abilities for an expansion endboss, but remedies this with mechanic evolution, meaning the abilities gain new effects as the players progress. This simulates a gradual ramp in difficulty without actually adding any new base mechanics to the fray. In addition to main phases that ramp in difficulty, the encounter's transition phases are dynamic, in that the type of transition phase is randomly selected every time a transition phase is begun. This adds an element of surprise and randomness to the encounter even though there are a finite amount of possible transition phases. This allows players to learn each transition phase individually and master all possible outcomes without having to rely on a luck factor to achieve victory (Keithyw).

Finally, Hellscream utilizes the popular raid "final showdown" design in the last moments of the encounter. Nearing its end, the encounter drastically increases in difficulty instantly as all base mechanics transition to their most deadly variations. Players must defeat the encounter before they are overwhelmed, ultimately ending the encounter one way or another (Excitedsoup). This design style is an influencing factor in the design of this project's encounter. The gradual ramp of difficulty is seen through multiple mechanics of the Arete encounter. These mechanics, like Arete's Rage, increase in either damage or frequency as time goes on. In addition, the "final showdown" design makes an appearance in the encounter, simulated by The Final Protocol.

Destiny's "Templar" Encounter

Bungie's "Destiny" is unique because it is perhaps the first MMO first-person shooter to be released and rapidly popularized on the console gaming platform. In addition to its genre, Destiny also offers unique raiding content within its base game. This raiding content is special because it limits the player party size to six players, a size not common among raiding scenes. Prior to Destiny's release, raiding content was primarily seen within PC titles, where the content was usually a variation of a third-person camera setup and a 10+ player party. This content was also usually limited to fantasy-style combat (swords, shield, bows, magic, etc.), and certainly didn't include technology like guns, rocket launchers and energy shields. In a way, Destiny defined its own classification of raiding by designing it within these new formats. As a result, the content experienced the many advantages and disadvantages of such design ambition.

The first raid Destiny released was the "Vault of Glass". The first encounter of the raid is a large robot called "The Templar", who boasts devastating ranged attacks along with a shield that covers its entire frame, preventing him from taking any damage. The encounter, much like Hellscream, is a multi-phase encounter that ramps up in difficulty as players progress. Unlike Hellscream however, players do not actually combat the boss during all of the numerous phases. In fact, players are unable to even damage the boss until the final phase. Instead, the Templar encounter focuses highly on additional enemies, or "adds", that engage the players. These adds, which must be killed by players, are an extremely important part of the encounter, and teach players how they must divide their firepower in order to deal with them effectively before becoming overwhelmed.

These adds, along with a few more basic mechanics, are all that define the Templar encounter until its final phase, which is appropriate with the raid's six player limit. The final phase complies all the responsibility of add control the players learn, along with a final mechanic that allows a single player the ability to drop the Templar's shield for a short time. This final mechanic is imperative to defeating the encounter, and highlights the need for constant communication between players. The mechanic also places immense responsibility on a single member of the party, making their actions even more crucial to the group's success. This coordination between the efforts of a single player and the rest of the party defines the encounter, and leads to some fantastic gameplay experiences (Hamilton).

Unfortunately, Bungie is new to making raid content for a mass audience, and is therefore prone to hit bumps in the road. Much of the criticism the first-person shooter style raid received was only strengthened by the number of apparent bugs and exploits that were discovered and utilized after its release. These exploits allowed players to kill bosses without needing to obey or execute mechanics whatsoever, and Templar was no exception. In fact, these exploits became so popular that the mainstream method of defeating the bosses was to exploit them. In the end many of these exploits were fixed by Bungie, but many were not, and players continued to ignore all design aspects of the encounter. These issue persists today even in Destiny's newest raid, and ultimately result from poor AI coding and environment design. The Bungie raid style is still young though, and what weakens encounters today will serve as examples on how to strengthen encounters in the future (Hillier).

Influences from Destiny's Templar are obvious in this project's encounter. The first-person style was ultimately chosen because it fit best with the encounter's theme. It didn't feel right to have players fight energy shield and mounted turrets with swords and shields, which obviously geared from Destiny rather than World of Warcraft. This thought shaped the direction the encounter's design took and led to the players wielding guns. In addition, the idea of players working their way to the actual "boss" was influenced by the Templar as well. This style of design mimics both Hellscream's and Templar's multi-phase format by separating the environment into individual sectors, which in this case act like phases. Additional enemies also made a prevalent appearance in the encounter, represented by the Bumpers and Field-Entities, and even Soulburn Orbs.

Though many major design decisions were influenced heavily from the Templar encounter, many innovations were added onto the format as well. Unlike in Destiny, where players naturally regenerate their health, health regeneration was left to player responsibility in the Arete encounter. A support-oriented role was designed and labeled as the "Healer" class, which uses projectiles shot from its weapon to heal rather than damage. This was an interesting player design decision to make, because it seemed like an exciting way to implement healing into a first-person shooter environment.

PLAYER TESTING

For testing purposes, the encounter was designed to be multiplayer over a local internet connection. No support for non-local connections was built in due to time and technical difficulties. Players who participated in testing connected to one another through an IP address system, and all players were required to have the same version of Unreal Engine to ensure compatibility. To facilitate an easy connection process, I created a pair of shortcuts that players could use to easily join each other's games, as seen below.

The hosting player used the "Server" shortcut, and was the first player to load into the level. This player acted as the host, to which all other players connected to using the "Client" shortcut. Each shortcut's properties had to be appropriately modified based upon where the user had installed and stored the Unreal Engine program, as well as the game's .EXE file. This is easily done by right clicking on the shortcut, selecting properties, and changing the shortcut's target to the following lines.

Target for the Server/Host shortcut:

<.exe location address> ?listen –server

Target for the client shortcut:

<.exe location address> <Host's IP address> -game

This results in an easy way for players to be able to connect to one another, without having to use command line inputs within the game.

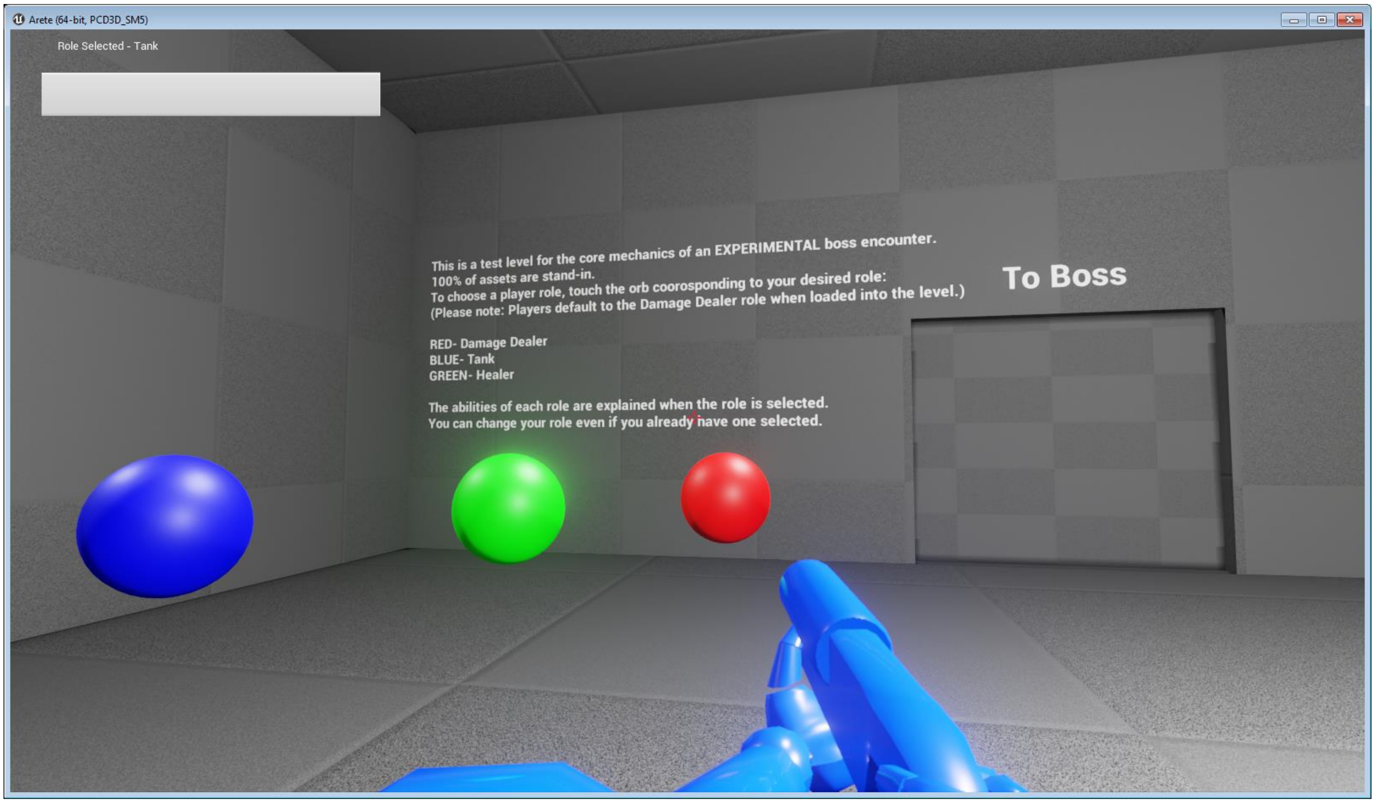
Once players load into the game, they selected their roles in a simple HUB/Role Selection area. A simple method of role-selection was chosen to facilitate any player desire to change roles multiple times during the role selection process. This led to colored orbs being placed in the HUB area, which changed the role of any player that touched them.

Red Orb – Change your role to "Damage Dealer (DPS)"

Blue Orb – Change your role to "Tank"

Green Orb – Change your role to "Healer"

Below is an image of this starting HUB/Role Selection area.



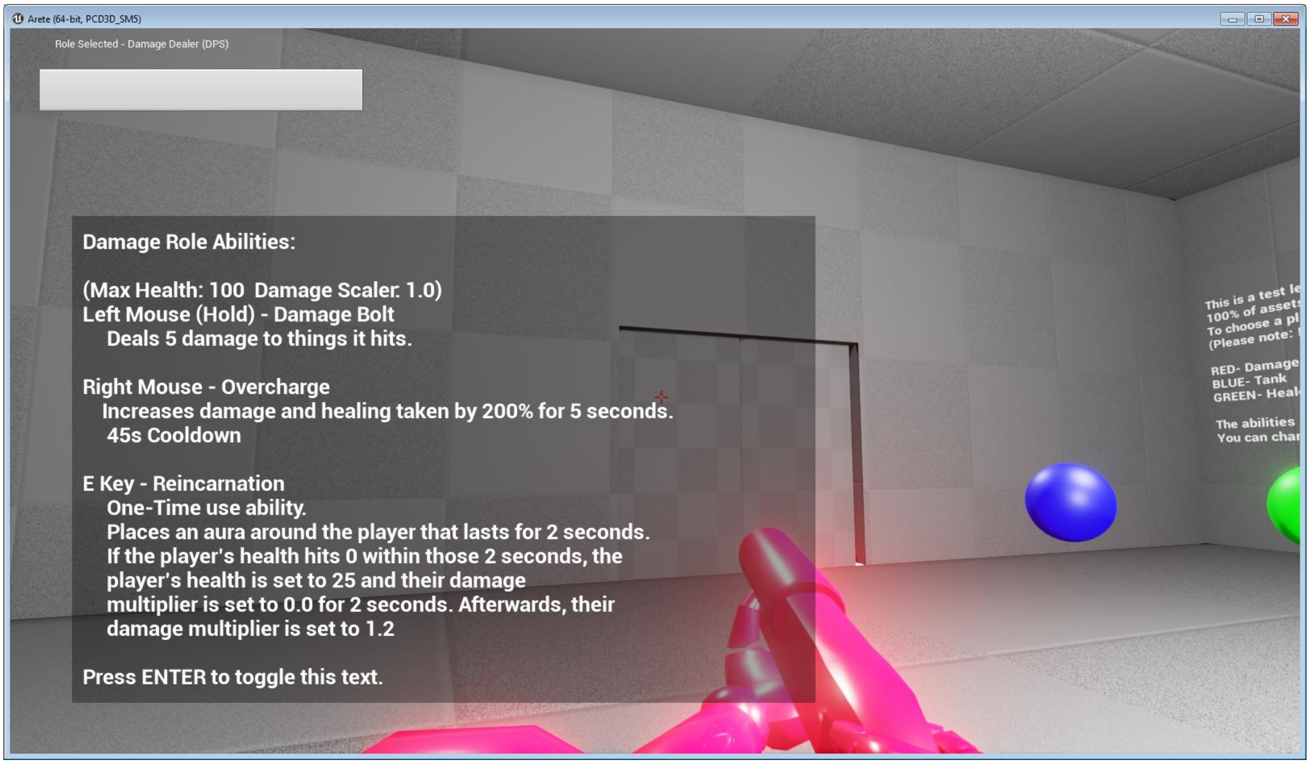
Role Selected - Tank

This is a test level for the core mechanics of an EXPERIMENTAL boss encounter.
100% of assets are stand-in.
To choose a player role, touch the orb coorosponding to your desired role:
(Please note: Players default to the Damage Dealer role when loaded into the level.)

RED- Damage Dealer
BLUE- Tank
GREEN- Healer

The abilities of each role are explained when the role is selected.
You can change your role even if you already have one selected.

Once a role is selected, a prompt appears giving the player more information on their selected role. Below is an image of the Damage Dealer's prompt.



Role Selected - Damage Dealer (DPS)

Damage Role Abilities:

(Max Health: 100 Damage Scaler. 1.0)
Left Mouse (Hold) - Damage Bolt
    Deals 5 damage to things it hits.

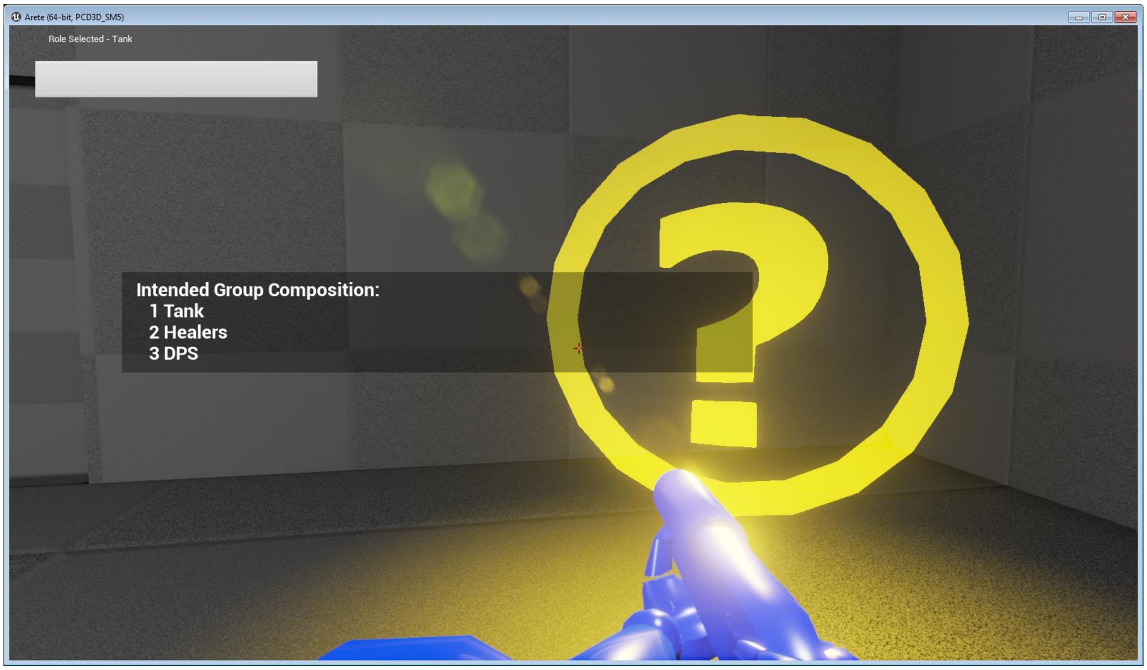
Right Mouse - Overcharge
    Increases damage and healing taken by 200% for 5 seconds.
    45s Cooldown

E Key - Reincarnation
    One-Time use ability.
    Places an aura around the player that lasts for 2 seconds.
    If the player's health hits 0 within those 2 seconds, the
    player's health is set to 25 and their damage
    multiplier is set to 0.0 for 2 seconds. Afterwards, their
    damage multiplier is set to 1.2

Press ENTER to toggle this text.

After roles were finalized, players proceeded through the door to the encounter area. In order to get results that were comparable and reasonable, all player groups adopted the suggested group composition of 1 Tank, 2 Healers, and 3 Damage Dealers.

An in-game prompt suggesting the suggested group composition:



Role Selected - Tank

Intended Group Composition:
1 Tank
2 Healers
3 DPS

As previously stated in the Mechanics Design section, this group composition was chosen as the default makeup based upon the number of mechanics each role is responsible for, damage output of boss mechanics, and health values of destroyable entities. In other words, the mechanics of the encounter were created and modified to a point where one tank can handle the tank abilities, two healers can handle the amount of damage players take, and three damage dealers can destroy the things they need to destroy in an acceptable amount of time (i.e. fast enough to where the boss's enrage mechanics do not kill the party).

A total of three player groups tested the boss, resulting in 18 individual feedback reports. After the players were finished with testing, they were then given a feedback response sheet that they filled out and returned to me via email. Below is a copy of the feedback response sheet, with values that represent the average response of all 18 players for each question.

PLAYER FEEDBACK/FEEDBACK ANALYSIS

Encounter Feedback Response (AVERAGES OF ALL 18 PLAYERS)

Respond to the questions below by bolding or underlining the option that represents your response. If the answer requires a specific response, fill in the parenthesis with your answer.

"X/18" Out of 18 Players "X/3" Out of 3 Groups

What was your Player Role?	**Tank (3/18)**	**Healer (6/18)**	**Damage (9/18)**
How many times did you try the boss?	**(19)**		
Did you defeat the boss?	**Yes (2/3)**	**No (1/3)**	
Did you read the encounter outline prior to attempting the boss?	**Yes (9/18)**	**No (9/18)**	
Did you use all of your abilities on your best attempt?	**Yes (16/18)**	**No (2/18)**	
If not, which ability did you not use?	**Primary (0/2)**	**Alt (0/2)**	**Special (2/2)**
I experienced bugs.	**Yes (5/18)**	**No (12/18)**	
I found exploits.	**Yes (2/18)**	**No (16/18)**	
I have played MMO "Raid" style content in the past.	**Yes (8/18)**	**No (10/18)**	

Read each statement below and supply a numeric value (0-5) that accurately represents your opinion.

0 – Strongly Disagree 1 – Disagree 2 – Slightly Disagree

3 – Slightly Agree 4 – Agree 5 – Strongly Agree

AVEREAGE = (Sum of all 18 values) / 18	
The encounter's difficulty was reasonable.	**(3.18)**
The encounter was too hard.	**(2.8125)**
The encounter was too easy.	**(1.3125)**
I was busy at all times during the encounter.	**(4.0625)**
My abilities were useful.	**(4.875)**
My role was easy to fulfill and execute.	**(3.0625)**
I had too many responsibilities.	**(2.25)**
Some boss mechanics were confusing.	**(2.0625)**

There were multiple ways to deal with obstacles.	**(3.00)**
Objectives were clearly defined.	**(3.75)**
Mistakes were easy to identify.	**(4.75)**
Boss mechanics synergized well with each other.	**(4.50)**
Player mechanics synergized well with each other.	**(4.3125)**
The encounter's difficulty ramp was smooth.	**(3.9375)**
I enjoyed the encounter.	**(4.4375)**
The encounter was well designed.	**(4.25)**
The encounter was engaging.	**(4.9375)**
The encounter encouraged cooperation.	**(4.25)**

Players were allowed to leave comments/suggestions at the end of the response sheet, which I have summarized into categories based on player role below.

General Player Comments based on Role

Tanks

There were no tank-specific comments supplied in the response sheets. All comments and suggestions tank players left were also left by players of other roles. These comments can be seen in the "Multiple Roles" category below.

Healers

Healer-specific comments mainly focused on the inability to see the health of other players, which made healing difficult. They also could not tell if they were actually healing other players successfully or not.

Damage Dealers

Damage Dealer-specific comments revolved around there being no indicator on how much health hostile entities had remaining. Comments also touched on the fact that damage dealing players could not tell if they were doing damage to destroyable enemies or not. Finally, DPS players reported that their Alternate ability "Overcharge" seemed useless, and rarely used.

Multiple Roles

Comments given by multiple roles included the player's inability to know when their abilities were active, on cooldown, or ready to be used again. Players also did not know when boss mechanics and abilities were going to execute next.

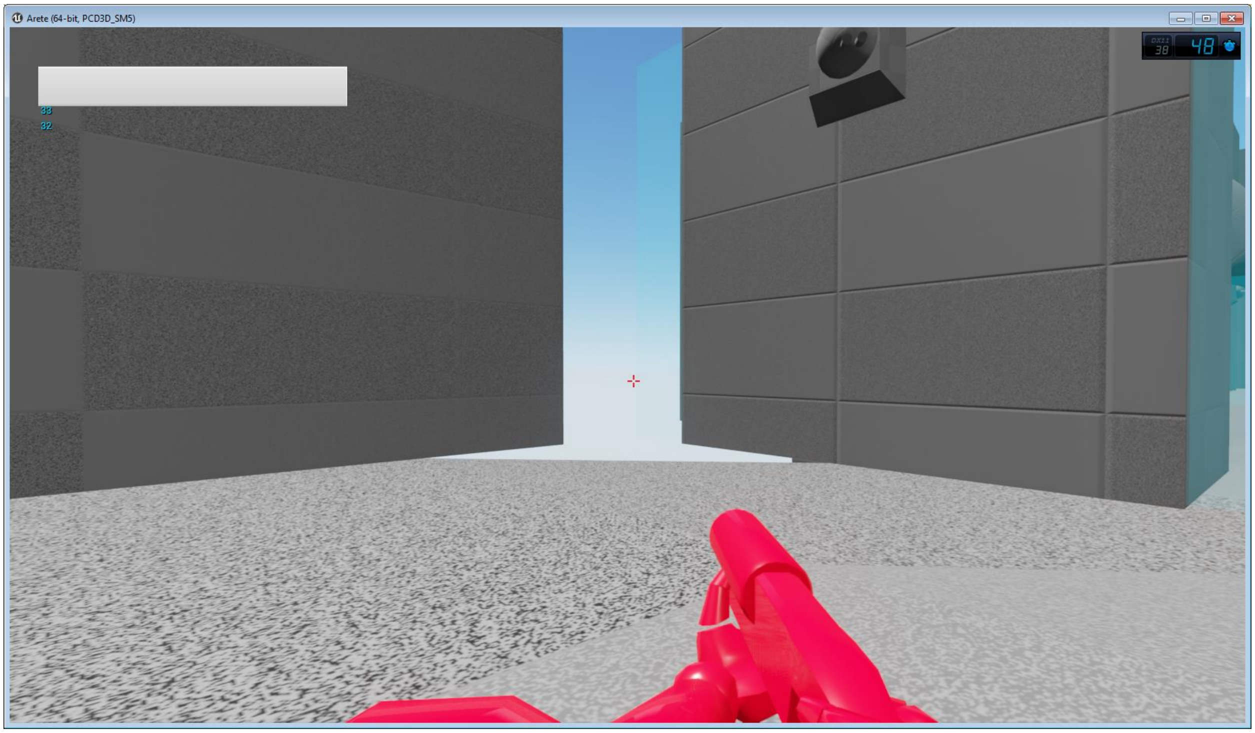
Before we analyze individual statistics, I want to briefly describe the makeup of each player group. Just to make things easier, I will refer to them as Group one, two, and three. It was my intention to include players in testing who had a varying range of prior raid content experience. Group one had absolutely no raid experience in its group, while group two had 2/6 players with World of Warcraft raid experience. All group three members had Destiny raid experience, with most of them having World of Warcraft raid experience as well.

Going through each statistic, I can get a general idea on how the player testing results stack up. As stated before, each group adopted the suggested group composition, so the player role spread is expected. The groups experienced different amounts of attempts before the conclusion of the testing, with 23, 20, and 14 attempts respectively. With the average attempt count being 19, the initial goal of expected attempts before completion (10-30) was satisfied. Groups two and three were able to defeat the encounter, while group one was not. Prior raid experience was an obvious benefit, as the group with the most experience in both World of Warcraft and Destiny style raiding defeated the boss in the fewest attempts. Group two, which had two of its six players being experienced, defeated the boss, but at a significant attempt increase. Another factor that seemed to play heavy into performance was whether or not the group looked over the encounter outline prior to testing. Each group was sent the encounter outline a few days before testing, but were not required to necessarily look at it. In an apparent attempt to handicap themselves, group one decided as a whole to not look at the encounter outline. Neither of the other groups fully ready the encounter outline, with 3 members of group two and 5 members of group three reviewing the outline before testing. Despite this, these groups that had some but not all of its members look over the outline performed better than the group that did not. It is my opinion that both prior experience and review of the encounter outline played significant roles in determining each group's performance.

All players except the tanks of group one and two used all of their abilities during the encounter. The two tanks in particular never used their special (E Key) ability during their best attempt. Arguably this ability is the least noticeable, and is just used to mitigate damage take for a short amount of time. Both players reported in their feedback comments that they simply forgot they had the ability. Visual HUD indications for all player abilities may help this issue.

Bugs reported from five of the eighteen players included minor instances that I discovered while creating the encounter. These included the Auto-Turret projectiles carrying players a distance if they hit the player's feet. This was caused by the projectile's physics properties, and could not be fixed without eliminating the projectile's arced movement path, which was not an intended design feature. Other reports of bugs were player misconceptions. These included players reporting red and blue Soulburn Orbs respawning as soon as one was killed or absorbed, when in fact the Soulburn Orbs are set on a fixed spawn time. The illusion of the orbs coming back to life was a result of players not dealing with mechanics fast enough, and stopped becoming a common occurrence once players gained more experience with the mechanic.

Exploits were something carefully avoided while the mechanics and environment of the encounter were being created. Two players reported the same exploit, which was found in the final phase of the encounter. Players utilized two holes in corners of the final area to eliminate Bumpers during The Final Protocol. This was an intended design point, as players needed some way to eliminate Bumpers in the final phase. The two players assumed that these holes were not supposed to exist, as they were always open (unlike the Flame Vents) and placed in extreme locals of the final area. The holes arguably looked like they did not belong there, which can be easily fixed. An image of one of these holes is supplied below:



In terms of the feedback where players supplied a numeric value as a response, the responses varied mostly based on player role, as many of the questions pertained to the difficulty of the fight and the execution of their roles. Since some roles were harder to play, this variance in responses makes sense and was expected. Generally, healers players had the hardest time during the encounter due to the amount of damage players were taking. They also had a particularly hard time because they could not see the health bars of other players, which was a common feedback comment from healers. This was a design choice that forced players to communicate well with each other. All players thought the encounter was at a decent difficulty level, with a few outliers saying the encounter was either too hard or easy. These outliers corresponded as one would expect with prior raid experience. Overall, players averaged “Agree” with the notions that the encounter was engaging, well designed, and encouraged cooperation.

Unexpected player feedback included players “Slightly Agree”ing that their classes were easy to play. Though since healers, which were the hardest role to play and the lowest raters of this category, made of 37.5% of the testing pool, it’s obvious that this value was dragged down by the default difficulty of their class. In addition to this statistic, players also rated low on

"Some boss mechanics were confusing" and "There were multiple ways to deal with obstacles". The first response I expect arose from their being no indicators on when boss mechanics were executing. This was a common comment from players in the feedback, and the only timer that was present during the fight was a rudimentary timer that kept track of the total number of seconds the players had been fighting the boss. The only way for players to know when mechanics would be executing would have been to know specific cooldown times from the encounter outline, and then do quick math during the boss attempts. Since half of the testers did not read the encounter outline, they had no idea when these executions would occur. This was an unfortunate take away from the encounter experience, but one that was foreseen with omission of a visual timer system. Mechanic functionality was prioritized over player warnings during development.

Players only "Slightly Agree"ing with the notion that multiple strategies could be used to deal with mechanics indicates that players were either focusing on a single aspect of the fight too intently, or alternative options were not obvious enough. As the designer, I think there were indeed multiple ways to do the encounter in terms of player positioning, ability stacking, and damage focus, but also agree that I could have made these situations more apparent to players through the development process.

Finally, in terms of common comments expressed by the players, most of the comments revolved around user interface or HUD features they felt should have been implemented into the game. The inability to see another player's health, regardless of role, was a common complaint by healers. In addition, many player suggested that boss/player timers be implemented so players can see when boss mechanics will happen and know when their abilities are ready for use. Finally, a few damage dealing players brought up the fact that they could not see enemy health values, or tell that they were dealing damage to enemies. The inability to see enemy health values was an intended design choice, forcing damage dealers to output their damage at a constant rate. The inability to see damage done to enemies, like boss and player timers, resulted from other parts of the development process being prioritized. Lastly, damage dealers reported that their alternate ability "Overcharge" was "useless" or "unneeded". This was followed up by stating that players pairing the "Healing Patch" and "Exoshield" abilities together was more than enough healing that was needed, rendering the "Overcharge" ability moot. Going back over the timeline of the encounter, I agree that this ability could be modified or replaced in order to be more useful.

POST-FEEDBACK BALANCING/CHANGES

Below is a list of several additions and balancing changes that were done to the game after player testing results were taken into account. Most of these changes have already been implemented into the game, with a few labeled as "future" changes/additions that will be made after the submission of this report. These changes are intended to fix any issues that were discovered through player testing, in an attempt to improve the overall experience quality of the encounter. A brief reasoning for each change is supplied in italics (Hazzikostas) (Sullivan).

Additions:

- (Future) Player-Friendly UI – including party health bars and ability notifications.
    - *Improved player experience, strong desire in feedback.*
- (Future) Boss Ability Timers – detailing times until next ability executions.
    - *Improved player experience, strong desire in feedback.*
- (Future) New DPS Alternative Ability "Sprint" – Increases the player's movement speed by 200% for 5 seconds. 45s Cooldown.
    - *Replaces irrelevant player ability.*
- Made the holes in the final phase area look like they belong there.
    - *Fixes confusion about last phase holes from player feedback.*

Balancing Changes:

Arete Defense System:

- The Purge Protocol now moves roughly 10% slower down the playing field.
    - *The ability seemed to move fast with no warning.*
- Pressure Plates now alternate between their raised/lowered positions every 8 seconds, down from 10.
    - *Tank players that missed a pressure plate had too small a window to absorb the blue orb.*
- Auto-Turrets now become Overcharged every 70 seconds, down from 80.
    - *Lines up overcharge with purge better.*
- Red Soulburn Orbs will no longer spawn in locations that already have a red Soulburn Orb, instead the existing red Soulburn Orb's maximum health will be doubled, and the orb will be set to 100% health.
    - *Fixes issue of overlapping red orbs without removing difficulty.*
- Shred Saws now deal 50% more damage.
    - *Saws seemed negligible in damage.*
- Bumpers are now 100% more bumpy.
    - *Aw yea.*

- Arete’s maximum health has increased by roughly 15%.
    - *Final phase seemed to last a bit shorter than expected.*

Players:

- Reincarnation’s invulnerability window has increased to 5 seconds, up from 2 seconds.
    - *Fixed issue with purge protocol double instant-kill on a DPS player. Accounts for purge movement speed change.*
- Decreased the Healer’s Damage Multiplier to 1.2, down from 1.5.
    - *Fixed an issue where healers were getting killed too quickly in some cases.*
- Decreased the healing “Healing Bolt” does to the healer that uses it to 2 HP / Bolt, down from 5 HP / Bolt.
    - *Diminishes healer bouncing “Healing Bolt” self-heals.*
- Increased the duration of “Barrier” to 5 seconds, up from 3 seconds.
    - *Makes the advantage of Barrier noticeable.*
- Renamed “Barrier” to “Personal Shield”.
    - *New name fits the ability feel better.*

CONCLUSION

After the analysis of the player feedback, it was determined that the project's specific definition of success was fulfilled by the project's testing results. Due to this fact, it was safely claimed that the project as a whole was a success within its own pre-defined definition. The initial goal of creating "a functional, engaging, and challenging encounter that multiple players, regardless of prior gaming experience, could defeat in a reasonable amount of time" was supported by player feedback agreeing that this was demonstrated in the final encounter. In addition, the targeted number of attempts before the boss's defeat (between 10 and 30) was satisfied, as the average number of player attempts resulted in 19. The satisfaction of these goals, coupled with a 66% player success rate (% of players that defeated the boss), conclude the project with an overall sense of accomplishment.

In addition to specifically defined project goals and terms of success, my personal goals were also achieved by the conclusion of the project. I was extremely happy with the finished product, especially when I consider that I started with little to no knowledge of the Unreal Engine's blueprint scripting system. Player feedback was better than I could have hoped for, and the player performance results were pleasing to say the least. I enjoyed the process of designing a complex boss encounter, as the challenges it presented were refreshing and vastly different from the challenges a simple player must confront in a game. Having to design an experience from all possible angles, while considering all possible player actions and responses to those design choices, was an experience I thoroughly enjoyed. With the addition of the features I plan to implement in the future, the encounter will become an even more successful experience, as well as one that respects the feedback of its testers. That being said, there is always room for improvement, as some of the feedback to certain aspects of the encounter was not as positive as I would have hoped. The design mistakes/omissions that resulted in this feedback has been taken into account, and learned from. With the experience in both software and design I attained during the development of this project, I look forward to creating even better work in the future.

LINKS

Damage Dealer Intro Video : < https://www.youtube.com/watch?v=xU54WFACpBQ >

Healer Intro Video: < https://www.youtube.com/watch?v=tnQoRROQIuo >

Tank Intro Video : < https://www.youtube.com/watch?v=3RUSl6fpb1A >

Encounter Run-through Video : <https://www.youtube.com/watch?v=s36XEtHHgI4>

REFERENCES/RESOURCES

Excitedsoup. "Raiding Review: The Siege of Orgrimmar Normal (Part 3)." *OMFG Soup*. WordPress, 25 Oct. 2013. Web. 16 Apr. 2015. <http://omfgsoup.com/2013/10/25/raiding-review-the-siege-of-orgrimmar-normal-part-3/>

"Garrosh Hellscream (tactics)." *Wowpedia*. Hydra, 27 Mar. 2015. Web. 16 Apr. 2015. <http://wow.gamepedia.com/Garrosh_Hellscream_%28tactics%29#Encounter_Design>

Hamilton, Kirk. "The Vault of Glass Is The Best Thing In Destiny." *Kotaku*. Kinja, 16 Sept. 2014. Web. 16 Apr. 2015. <http://kotaku.com/the-vault-of-glass-is-the-best-thing-in-destiny-1647214453>

Hazzikostas, Ion. "Developer Watercooler: Encounter Tuning by Watcher." *World of Warcraft*. Blizzard Entertainment, 17 Jan. 2013. Web. 16 Apr. 2015. <http://us.battle.net/wow/en/blog/8445111/>

Hamilton, Kirk. "Vault of Glass vs. Crota's End: The Destiny Comparison We Had To Make." *Kotaku*. Kinja, 26 Jan. 2015. Web. 16 Apr. 2015. <http://kotaku.com/vault-of-glass-vs-crotas-end-the-destiny-comparison-w-1681793216>

Hillier, Brenna. "Destiny Guide: How to Beat Atheon and Templar in the Vault of Glass." *VG247*. Disqus, 21 Oct. 2014. Web. 16 Apr. 2015. <http://www.vg247.com/2014/10/21/destiny-guide-how-to-beat-atheon-and-templar-in-the-vault-of-glass/>

Hillier, Brenna. "Destiny Hot Fix Ruins the Vault of Glass Templar Cheese Strategy." *VG247*. Disqus, 14 Oct. 2014. Web. 16 Apr. 2015. <http://www.vg247.com/2014/10/14/destiny-patch-vault-of-glass-templar-cheese-ps4-xbox-one/>

Keithyw. "World of Warcraft: Siege of Orgrimmar Review." *Kontroversial Keith*. WordPress, 03 Nov. 2013. Web. 16 Apr. 2015. <http://www.keithwatanabe.net/2013/11/03/world-of-warcraft-siege-of-orgrimmar-review/>

LockeZ. "BOSS DESIGN THEORY." *Boss Design Theory*. Django, 10 May 2011. Web. 16 Apr. 2015. <http://rpgmaker.net/articles/476/>

Miozzi, Cj. "6 Tips To Make Your RPG Boss Fights Memorable." *The Escapist*. The Escapist, 17 Apr. 2014. Web. 16 Apr. 2015. <http://www.escapistmagazine.com/articles/view/tabletop/features/11318-6-Tips-To-Make-Your-RPG-Boss-Fights-Memorable>

Stout, Mike. "A Boss Is a Test." *Gamasutra*. Gamasutra, 2015. Web. 16 Apr. 2015. < http://www.gamasutra.com/view/feature/134503/boss_battle_design_and_structure.php?print=1>

Silva, Marty. "THE BEST BOSS BATTLES OF 2014." *IGN*. Disqus, 2 Jan. 2015. Web. 16 Apr. 2015. <http://www.ign.com/articles/2015/01/02/the-best-boss-battles-of-2014>

Stout, Mike. "Evaluating Game Mechanics For Depth." *Gamasutra*. Gamasutra, 21 July 2010. Web. 16 Apr. 2015. <http://www.gamasutra.com/view/feature/5901/evaluating_game_mechanics_for_depth.php>

Sullivan, Lucas. "How to Build a Better Boss Fight." *GamesRadar+*. Future Plc, 8 Apr. 2013. Web. 16 Apr. 2015. < http://www.gamesradar.com/boss-dissecting-mmo-boss-encounters/>

"The Art of Boss Design." *Flark Design*. WordPress, 16 Jan. 2012. Web. 16 Apr. 2015. <http://www.flarkminator.com/2012/01/16/the-art-of-boss-design/>

Tutorial." *Game Development Tuts+*. Envato Pty Ltd., 7 Dec. 2012. Web. 16 Apr. 2015. <http://gamedevelopment.tutsplus.com/tutorials/designing-a-boss-fight-lessons-learned-from-modern-games--gamedev-2373>

"Unreal Engine 4 Documentation." *Unreal Engine*. Epic Games Inc., 2015. Web. 16 Apr. 2015. <https://docs.unrealengine.com/latest/INT/>

Vlad. "Garrosh Hellscream Detailed Strategy Guide (Heroic Mode Included)." *Icy Veins*. Vedatis S.A.S., 29 May 2014. Web. 16 Apr. 2015. <http://www.icy-veins.com/wow/garrosh-hellscream-strategy-guide-normal-heroic>

Cameron, Wood. "Designing a Boss Fight: Lessons Learned From Modern Games - Tuts+ Game Development Tutorial." *Game Development Tuts+*. Envato, 7 Dec. 2012. Web. 16 Apr. 2015. < http://gamedevelopment.tutsplus.com/tutorials/designing-a-boss-fight-lessons-learned-from-modern-games--gamedev-2373>

Software

Adobe Systems Inc. Adobe Photoshop. Computer Software. Photoshop.com. CC 2014. Web

Autodesk Inc. Autodesk Maya. Computer software. Autodesk.com. Version 2015. Web.

Epic Games Inc. Unreal Engine. Computer Software. Unrealengine.com. Version 4.5.1+. Web

Mirillis. Action!. Computer Software. Mirillis.com. Version 1.24.3. Web