

Bell

Horror Level Design Document



Version 1.3 03/06/2025



GXPLOERS
STUDIO

Level Overview

Title: Bell

Engine: Unreal Engine 5

Genre: First-Person Psychological Horror

Format: Single-Level Experience (Playable Horror Level Prototype)

Playtime: 10–15 minutes (depending on exploration and player pace)

Development Context: LCV College Project – Level Design II Class

Developer Role: Solo Designer – Luis Yibran Calderon Hernandez

Development Time: 3 weeks (from layout to playable prototype)

Platform: PC (Playable in Unreal Editor)

Input: Keyboard + Mouse

Camera: First-person

You wake up in the dusty, cold hallway of William Elementary, a place you once knew. The classrooms, the washrooms, the sounds of children... it all feels familiar. But something is wrong. There's no one here. The lights flicker. The old school bell echoes, even though no one rang it. And then you hear whispers.

In *Bell*, you must follow your daily “class schedule” to escape the abandoned school visiting seven haunting zones, recovering keys, and facing the spirits of the children who once studied here. But as you relive these twisted school-day rituals, fragments of memory surface. The truth begins to distort your reality.

By the time the final bell rings, you'll discover who you really are, not a student, but a former teacher, a predator, condemned to experience your victims' revenge in an eternal cycle of guilt, fear, and torment. This level is your punishment. Welcome to your personal hell.

Premise

Bell is a psychological horror level set in the haunted remains of William Elementary School. The player explores seven interconnected areas based on school classrooms and facilities, uncovering a disturbing story through exploration, environmental storytelling, and ghost encounters.

The narrative twist reveals the player is not a student, but a damned former teacher reliving their crimes in an eternal punishment loop, condemned to repeat their own personal hell every time the bell rings.

The experience blends exploration, dread, memory fragments, and narrative-based level progression using simple mechanics such as key collection, scare triggers, and a

final chase sequence. Built in Unreal Engine 5 using Blueprints, Level Sequencer, and Navigation AI, it demonstrates pacing, horror atmosphere design, and level structure.

Narrative & Structure

Narrative Overview

In *Bell*, the player wakes up in the abandoned William Elementary School, a decaying and silent structure haunted by shadows of the past. The game begins with a sense of mystery and fear, leading the player to believe they are a student lost in a nightmare. But as the story progresses, visual cues, ghost encounters, and memory triggers reveal the devastating truth, the player is not a student, but a former teacher, condemned to relive the horrors they inflicted.

This twisted psychological loop is designed to mimic a school day, with a corrupted class schedule leading the player from one traumatic memory to the next. The bell marks both progress and punishment, signaling the beginning and end of a cycle the player cannot escape. *Bell* is not just about fear, it's about guilt, denial, and facing what you've done.

Key Story Themes

- **False Innocence:** The game initially positions the player as a helpless student to mislead and heighten the eventual reveal.
- **Guilt and Memory:** Each zone represents a buried memory, resurfacing through ghostly manifestations, dramatic lighting, and symbolic props.
- **Judgment and Torment:** The spirits of the children do not speak much — their presence is enough. They judge through silence, jumpscares, and shadowed appearances.
- **Cyclical Punishment:** The class schedule structure is not guidance — it is ritual. Every bell restarts the torment.

Final Level Structure (Updated Gameplay Flow)

Section	Title	Story Purpose	Key Reveal or Moment
1	School Entrance & Lobby	Awakening in a haunted place.	Thunder reveals girl's shadow. Bell sound begins.

2	Principal's Office	Establish schedule, subtle memory triggers.	Key found. Bell sound triggers horror loop.
3	Hallway	First direct tension and haunting.	Child laughter. Girl's shadow at washroom door.
4	Pool Room	Escalation through supernatural danger.	Shark in pool. Hands pull player if they fall.
5	Ruined Classroom	Environmental horror and shock event.	Light explodes. Flashlight acquired.
6	Dark Classroom	Exploration under pressure.	Music rises. Key hidden in shadows.
7	Girls' Washroom	Sudden horror peak.	Ghost girl jumpscare in mirror.
8	Gym & Final Escape	Climax and chase sequence.	Screams, debris floats. Player chased to exit.

Key Narrative Tools

- **Ghosts as Triggers:** Appear suddenly or indirectly to shift mood or reveal guilt.
- **Environmental Storytelling:** Every zone includes signs of past life and trauma: ruined classrooms, blood stains, broken lights, scribbled notes.
- **Lighting and Audio Design:** Flickering bulbs, thunder, distant crying, silence before chaos.
- **Memory Events and Symbolism:** Events like the girl in the mirror or the floating gym debris represent emotional flashbacks.
- **The Bell:** Rings at beginning and end, tying the narrative to cycles of trauma and punishment.

Gameplay

Player Experience Goals

Disempowerment

The player is vulnerable: no weapons, no combat, only escape, hiding, and panic. This amplifies fear and tension, especially during ghost encounters and the final chase.

Narrative Curiosity & Unease

Through disturbing notes, sudden ghost appearances, and changing environments, the player uncovers dark truths. The story is non-linear and progressively more unsettling.

Psychological Tension

Familiar childhood spaces like classrooms and bathrooms are distorted into spaces of dread. The fear of returning to a corrupted past is central to the atmosphere.

Cinematic Horror Moments

Each section features a distinct horror sequence, flickering lights, floating objects, mirror ghosts, and ambient whispers. These moments are paced for maximum suspense and shock.

Reflection and Guilt

By the end, players don't just feel fear, they feel conflict. They pity the main character, but also despise him. This emotional twist lingers beyond gameplay.

Looped Hell Structure

The ringing bell marks a return to the start. The player is caught in an endless punishment loop, reinforcing the horror of inescapable guilt.

Design Vision & Pillars

These are the fundamental design values that define *Bell* as a horror experience. Each vision is a guiding principle that drives every aspect of the level, from layout and mechanics to sound and narrative. Without these, *Bell* would lose its identity.

1. Psychological Horror Over Physical Threat

The goal is to disturb the player's mind, not just startle them. Fear comes from atmosphere, implication, sound, memory, and internal dread, not gore or violence.

2. Narrative Discovery Through Exploration

The story is not told directly. The player must uncover it through environment, notes, sounds, and visual clues. Each new zone reveals a layer of truth.

3. Constant Disempowerment and Vulnerability

The player has no means to defend themselves. This helplessness intensifies fear and creates real tension, hiding, running, and uncertainty are the only tools.

4. Emotional Twist and Moral Conflict

The game leads the player to believe they are a child, only to twist expectations with the reveal that they are the predator. This recontextualizes the entire experience.

5. Timed Structure as Punishment Loop

The game mimics a school schedule. Each “class” is a level section. If the player fails to finish their schedule, they are punished and the loop begins again. The bell becomes a symbol of trauma.

Game Pillars (How Visions Are Implemented)

1. Atmosphere, Sound Design, and Randomized Events

To support psychological horror, the level uses low light, eerie ambience, whispers, flickering lights, and non-linear random scare triggers. The player never feels safe.

2. Environmental Storytelling and Collectibles

Instead of cutscenes or direct exposition, the game reveals its narrative through notes, room props, blackboard writings, memory cinematics, and ghost placements. Every object in the school contributes to the lore.

3. No Combat, Only Escape & Observation

Players can't fight back. Their only interactions are opening doors, picking up keys, reading notes, or running away. This design forces players to stay alert and fearful.

4. Shifting Perspective and Memory Reveals

Each section contains a “memory event”, a cinematic or ghost-triggered scene that reveals who the player is and what they've done. These moments are placed after tension peaks to amplify impact.

5. Class Schedule System and The Bell Loop

The player is guided by a distorted “class schedule” and must visit the seven rooms before the final bell. Failure or death restarts the loop, trapping them in their own guilt. The bell isn't just audio, it's a design mechanic.

Gameplay Mechanics

Bell focuses on minimalist, story-driven horror gameplay where tension and dread come from exploration, environmental storytelling, and trigger-based events. The player is not meant to fight, only to experience, survive, and uncover the truth. Mechanics are designed to reinforce the feeling of vulnerability and escalating fear.

Core Mechanics

1. Exploration

- First-person navigation using keyboard and mouse.
- Players explore linear but branching paths through 8 key school areas.
- Locked doors and progression require collecting specific keys.

2. Key Collection & Progression

- Each major zone requires a key to access. Keys are hidden or placed near scripted scare events.
- Keys are picked up via interaction (hover and click).
- The schedule acts as a soft guide to “which room comes next.”

3. Flashlight System

- Flashlight is picked up midway through the level.
- Needed to explore pitch-dark rooms (e.g., the second classroom).
- Allows visual navigation and creates horror pacing via limited light radius.

4. Triggered Horror Events

- Shadow girl appearing and vanishing in hallway.
- Shark in the pool, shadow hands when falling in.
- Flashbulb exploding in ruined classroom.
- Girl in the mirror jumpscare.
- Floating gym objects and final chase.
- Triggers activate via player position, time, or object interaction.

5. Chase Sequence (Final Event)

- After reaching the gym and walking forward, a scripted chase begins.
- Camera shifts, shadows scream, and debris floats.
- The player must sprint toward the exit, dodging chaos.
- If successful, the game ends, otherwise, restart.

Support Mechanics

6. Cinematics & Memory Fragments

- Triggered cinematics create key emotional beats (e.g., bell ring, girl chase).
- No UI cutscenes, all in-world transitions using camera blend and sound.

7. UI Elements

- Minimal UI. The flashlight icon may show when acquired.
- Class schedule shown briefly to guide player.
- Key pickup feedback via small text (e.g., “You found a Key.”)

8. Audio as Feedback

- Dynamic music escalates in later zones (e.g., during classroom and bathroom).
- Sudden stingers for jumpscares.
- Ghost sounds, whispers, and ambient tones enhance horror pacing.

Note: There is no combat, inventory system, puzzles, or dialogue. All gameplay is centered on movement, observation, and emotional tension through scares and progression.

Level Layout & Flow

Bell is structured around progressive emotional tension and horror pacing, guiding players through a linear but psychologically layered path. The level design relies heavily on environment cues, sightlines, signifiers, affordances, and scripted escalation to deliver an immersive narrative horror experience.

Overall Flow

The level follows an 8-part sequence representing a twisted school schedule. Each section builds tension and narrative progression while introducing varied spatial design, lighting, and sensory feedback.

Phase	Area	Mood Progression	Core Experience
-------	------	------------------	-----------------

1	Entrance & Lobby	Confusion & Curiosity	Intro, exploration, thunder scare
2	Principal's Office	Unease & Subtle Clues	Bell rings, first key found
3	Hallway	Suspense & Hints	Child laughter, first ghost shadow
4	Pool Room	Threat & Temptation	Shark encounter, danger trigger
5	Ruined Classroom	Shock & Reveal	Lightbulb explosion, flashlight
6	Dark Classroom	Anxiety & Dread	Tense music, key hunt with light
7	Girls' Washroom	Terror	Mirror jumpscare, ghost scream
8	Gym & Escape	Chaos & Climax	Debris floats, chase to exit

Game Objective & Core Loop

Main Objective:

Explore the haunted elementary school, collect keys, survive supernatural encounters, and uncover the truth about your identity and past. Escape the nightmare loop before the curse resets.

Core Gameplay Loop:

1. Explore a new area.
2. Observe environmental clues.
3. Collect a key or item.
4. Trigger a horror or memory event.
5. Use key to unlock next section.
6. Repeat, escalating tension and story progression.
7. Final event: survive chase and reach the exit.

This loop reinforces fear through repetition, escalation, and the illusion of progress within a doomed cycle.

Navigation Principles

- **Linear Core with Looped Access:** Players revisit the main hallway hub multiple times, reinforcing orientation without confusion.
- **Spatial Memory Reinforcement:** Familiar areas are re-encountered under different lighting/audio conditions, making the school feel alive and cursed.
- **Choke Points for Tension:** Narrow halls, cluttered rooms, and dark washrooms amplify vulnerability and direct player gaze.

Level Design Techniques Used

1. Orientation & Flow

- Landmarks: Principal's desk, pool ladder, classroom boards, mirror sinks.
- Sightlines: Long hallway views, framed doors, spotlighted items.
- Rhythm: Alternates between quiet exploration and high-tension events.

2. Wayfinding & Signifiers

- Signifiers: Exit signs, flickering lights, opened doors, destroyed debris paths.
- Light as Direction: Bright areas subtly guide player attention (e.g., pool trampoline, ruined desk).

3. Affordance

- Interactable elements are clearly readable: doors with handles, keys on desks, flashlight glow.
- Functional props like ladders or stairs invite movement and verticality.

4. Pacing & Intensity Mapping

The level uses a crescendo structure, starting quiet and escalating until the final chase. A heatmap of intensity shows:

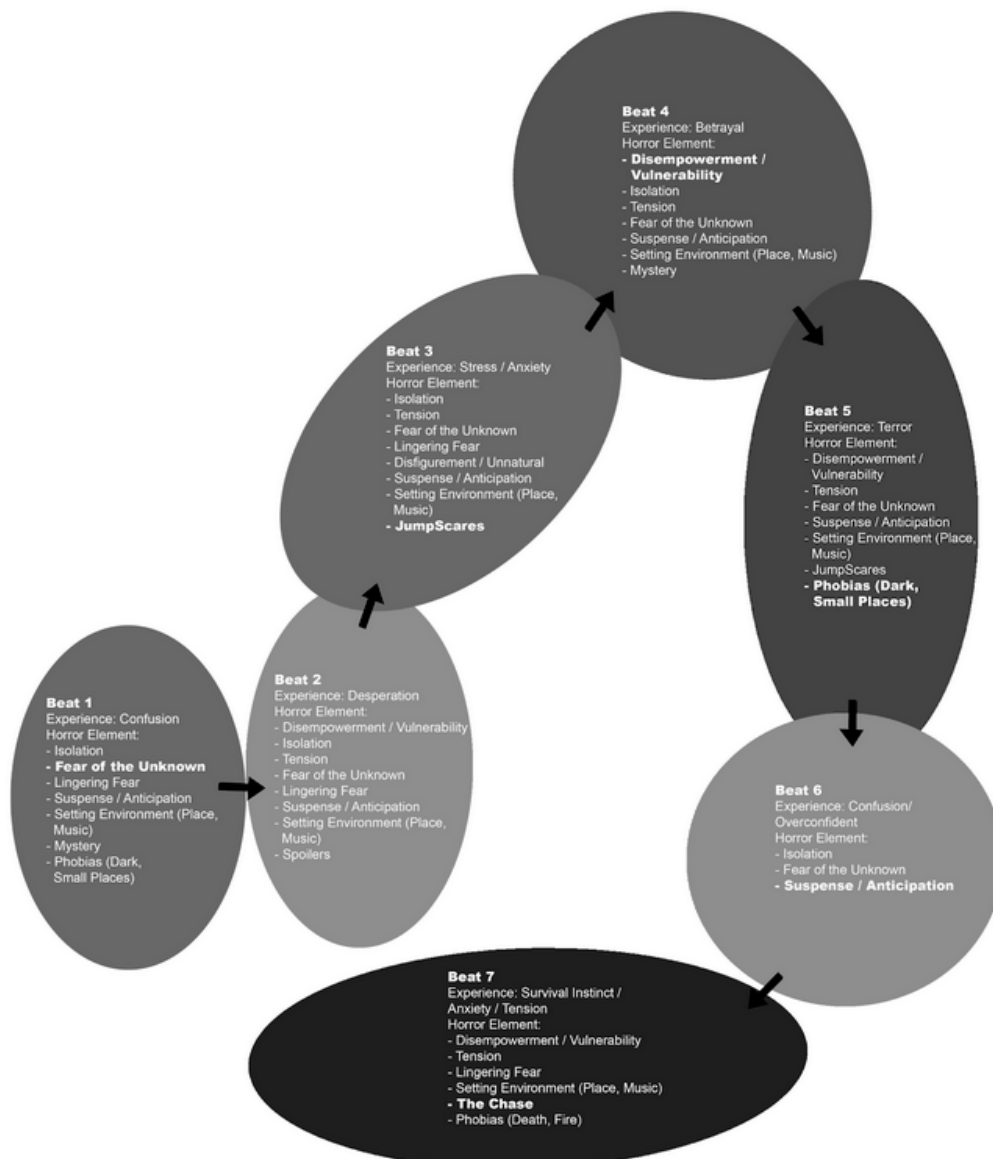
- Low → Thunder shadow, silence
- Medium → Pool room tension, hallway ghost
- High → Mirror scare, exploding bulb
- Peak → Floating gym chaos and chase

Key Layout Components

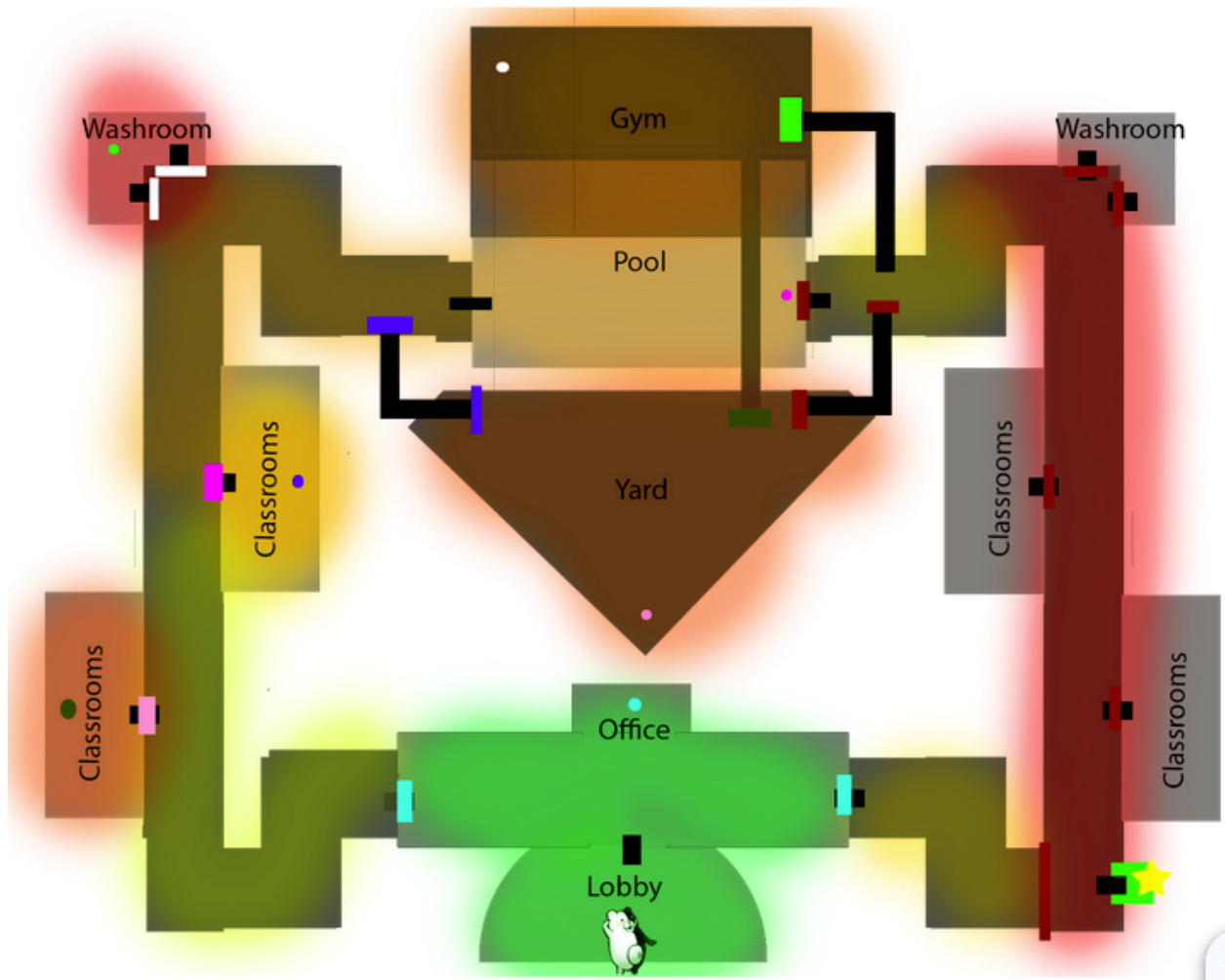
- **Lobby:** Player learns controls, pacing, and exploration logic.
- **Hallway:** Central connector for spatial orientation and returning segments.
- **Key Rooms:** Each serves narrative and horror function (classroom = memory, washroom = trauma).
- **Final Room:** Open space used for chaos and climax, with floating physics and chase scripting.

Visual Aids

Bubble Diagram:



Heat Map:



Implementation in Unreal Engine

The level *Bell* was developed using Unreal Engine 5, enhanced by the Horror Engine plugin and the Abandoned School environment asset pack. Built as a first-person psychological horror experience, the level integrates blueprint-driven systems for pacing, interaction, and atmosphere while following professional prototyping workflows.

Tools, Assets & Plugins

- **Engine:** Unreal Engine 5
- **Primary Plugin:** Horror Engine (for player controller, flashlight, keys, UI prompts, jumpscare triggers)

- **Environment Assets:** Abandoned School (Marketplace), Megascans (for modular dressing)
- **Lighting:** Combination of baked and dynamic lights, fog volumes, flickering fixtures
- **Audio:** Integrated ambient loops, stingers, whispering ghosts, and directional SFX cues
- **Blueprints:** All logic implemented using Unreal's Blueprint system (no C++)

Prototyping & Grayboxing

Before final assets were introduced, the full level was designed using a modular graybox approach:

- **Blockout geometry** defined classroom, hallway, and gym proportions
- **Scare timing** was playtested using dummy triggers and invisible events
- **Sightline testing** ensured scripted scares (e.g. shadow girl) were visible at precise angles
- **Basic interaction flow** (door unlocking, flashlight pickup) was iterated using Horror Engine blueprints

This phase was critical for planning intensity pacing and spatial rhythm before environmental detail was added.

Core Gameplay Systems & Blueprint Features

The following gameplay mechanics were implemented through custom Blueprints or Horror Engine integration:

Interaction & Progression

- Trigger boxes and area events (location-based scare logic)
- Inventory system using input key logic for door unlocking
- Flashlight pickup and toggle (usable only after acquiring it)
- Display and update of on-screen interaction messages
- Interactive notes (readable UI overlay)
- Checkpoint logic to reset player position if necessary

Movement & Flow

- Ladder climbing logic using trigger detection
- Teleport/position correction for cinematic sequences
- AI-based chase behavior and basic navigation logic in the final gym event
- Camera transitions for cinematic moments (e.g., final chase)

Horror Presentation

- Cinematic scripting with sound and visual FX coordination
- Mirror ghost jumpscare with camera shake, light burst, and sound stinger
- Post-process volume for underwater/shadow hand pool sequence
- Breakable and flickering lights as scare signifiers
- “Look-at” triggers to guide attention or trigger hidden events

Level Implementation Highlights

- **Entrance:** Shadow girl flash and thunder event using light triggers
- **Bell Ring:** Sound and schedule logic triggered upon interacting with desk object
- **Pool Room:** Shark visual + shadow hands scare using underwater overlay and sound
- **Classrooms:** Lightbulb explosion, darkness navigation, and flashlight-driven exploration
- **Girls’ Washroom:** Mirror scare sequence with shadow girl and sound stinger
- **Gym:** Full cinematic chase with floating debris, screams, camera control, and exit gate

Optimization

- All logic and systems implemented in Blueprints, no C++ or external scripting
- Full level loaded at runtime (no level streaming) to maintain scare timing
- Light baking for static areas (classrooms, hallway) to ensure performance

Development Challenges & Learnings

Key Challenges and Solutions

1. Limited Development Time

Creating a complete horror level within a few weeks required efficient planning. Prioritizing core horror beats and simplifying non-essential features helped manage time constraints without losing impact.

2. Large Multi-Zone Level Scope

Designing and implementing seven distinct sections with their own mechanics, lighting, and narrative pacing demanded fast iteration. Each area was first grayboxed, then gradually detailed based on priority and emotional impact.

3. Implementing Level Design Principles

Ensuring good navigation, orientation, affordance, pacing, and intensity required constant iteration. Sightlines, lighting, and sound cues were adjusted repeatedly through playtesting to align with horror flow.

4. AI and Blueprint Complexity

The final chase and dynamic horror sequences involved layered blueprint logic. Learning and modifying Horror Engine systems such as AI pathing, jumpscare triggers, and cinematic sequences was a technical challenge that required deep blueprint work.

5. Transitioning from Document to Prototype

The progression from concept to prototype, then graybox, and finally to polished level, was fast-paced. Each stage required clear versioning and a strong connection between design intent and in-engine execution.

6. Implementing Horror Systems Under Constraints

Flashlight mechanics, jumpscares, ambient sound triggers, and visual FX had to be implemented using modular and reusable blueprints. Balancing timing and emotional rhythm was refined via testing.

Summary of Learnings

- Effective scope control and planning enable strong results under time pressure.
- Grayboxing is essential for testing level flow, pacing, and tension early.
- Horror design depends on emotional rhythm, not just scares, but silence and buildup.
- Blueprint systems in UE5 can deliver complex gameplay with strong iteration if structured well.

