

Guiding Light

Game Design Document

A 2D platformer prototype exploring the tension between sight and survival.

1. Design Argument

Guiding Light is built around a single design argument: every act of seeing must cost something.

In most platformers, vision is free. The player sees the level, navigates it, masters it through repetition. Guiding Light inverts that assumption. The world is dark by default. The player carries a torch that reveals everything—but it burns a finite resource. Lighting reveals the next obstacle; conserving keeps the player alive long enough to survive it.

This produces a specific emotional experience: anxious rationing. The player is constantly trading information for resources, and that trade compounds across the entire level. A reckless first flame in Level 1 changes how Level 2 feels. The game's tension is not in twitch difficulty—it is in the discipline of restraint.

Every system in the prototype exists to amplify that single tension.

2. Core Mechanic

The Light Loop

Holding Z lights the torch. The world becomes visible. Oil drains continuously while the torch is lit.

Releasing Z extinguishes the torch. The world goes dark. A small UI indicator (the Player Locator) shows the player's screen position, but environment, hazards, and pickups disappear from view.

Oil is finite. It carries continuously across all three levels. There is no auto-refill on death. The only way to recover oil is by collecting oil pouches placed throughout the levels.

Why Binary On/Off

An early design iteration explored two-intensity lighting (dim/bright with different drain rates). It was rejected. The binary choice—see or don't see—forces commitment. The player cannot 'mostly' see; they must decide to spend oil or not. This creates discrete decision points the player can feel.

The Three Pillars

- Light reveals the world (visibility = information).
- Oil is finite and consumed by light (vision = resource cost).
- Oil carries across levels (decisions compound).

Remove any one pillar and the design collapses. Without finite oil, light is free—no tension. Without continuous oil, each level resets the economy—no compounding consequence. Without darkness as default, the mechanic is invisible.

3. The Emotional Arc

Guiding Light is designed to produce a specific sequence of feelings across its three levels.

Level 1: Discovery

The player learns the basic loop. The first time they hold Z and the world fades in—that is the core emotional moment. They feel the value of light. Then they release Z and feel the cost of darkness. The cave becomes alien when they cannot see it.

By the end of Level 1, the player has internalized: lighting is for memorizing, darkness is for traveling.

Level 2: Vigilance

Stalactites introduce time as a hazard. The world is no longer just hazardous geometry—it is hazardous behavior. The player must now light to identify stalactite positions and time their passage.

The emotional shift: from 'memorize and traverse' to 'observe and time.' Vigilance replaces planning. Players report a noticeable increase in cognitive load.

Level 3: Calculation

Monsters introduce active threat. They chase when lit. The same light that reveals the cave now invites pursuit. Worse, monsters are solid platforms when dark—the player must learn to use them while resisting the instinct to flee.

The emotional arc completes: the player is now making layered calculations under pressure. Light to see, dark to be safe, but only briefly. Every torch press is weighed.

4. Decision Architecture

The prototype is built around three categories of player decision.

Macro: Oil Economy

How much oil to spend in this level vs. save for the next. There are no checkpoints. A profligate Level 1 makes Level 2 nearly impossible. This is the largest decision in the game and the only one that compounds across levels.

Meso: Section Strategy

How to approach each level section: light briefly to memorize, then traverse blind; or keep light steady for the entire section. Pouch placement is the design lever—pouches placed late in a level reward conservation, pouches placed early reward profligacy.

Micro: Frame-to-Frame Light

Tap or hold? Light during the jump or before it? At what point does the player release? These small decisions accumulate into the macro economy. The mechanic creates a kind of meditative rhythm at the micro level—the player learns to feel the cost per second.

5. Mechanics by Level

Each level introduces exactly one new mechanical layer. The Four-Beat Structure (Introduce, Develop, Twist, Test) is applied within each level to teach that mechanic.

Level 1: Foundation

- New mechanic: oil-rationed light + static hazards (spike rocks, pits).
- Introduce: First spike rock in clear view—light reveals it, player jumps.
- Develop: Pit + spike rock combination requires both lighting and platforming.
- Twist: A narrow path with hazards on both sides—lighting reveals the path; the player must commit.
- Test: Climax section requires precise movement under oil pressure to reach the beacon piece.

Level 2: Vigilance

- New mechanic: stalactites (proximity-triggered ceiling hazards with audio cue).
- Introduce: First stalactite in a safe position—player hears it crack, sees it fall, learns the pattern.
- Develop: Multiple stalactites along the lower path. Player must time passage.
- Twist: Beacon piece placed directly under a stalactite, over a pit. Three constraints stacked: time, platform, oil.
- Test: Upper path with three offset stalactites creating a wave pattern. The player must find the rhythm.

Level 3: Calculation

- New mechanic: monsters (active threats, become platforms when dark).
- Introduce: First monster placed where the player must encounter it. Light, see it chase, die or escape, learn 'lit = dangerous.'
- Develop: Multiple monsters force route planning under torch state management.
- Twist: A monster is the only path forward. Player must light briefly, see the gap, then dark-traverse across the monster's solid form.
- Test: Combined section with monsters, stalactites, oil pressure, and the beacon piece. Everything the player has learned is required.

6. Failure and Recovery

Death is brief. A short audio sting, the player freezes for 0.3 seconds, the scene reloads. The pause is functional: it lets the player register what killed them before the world resets.

Crucially, dying does not reset oil to full. The oil level at the start of the current scene-attempt is preserved. This means the player cannot grind their way through a section by dying repeatedly to refill. Death is a real cost.

This is the harshest of the three reasonable design positions on death-and-oil, and it was the right call in playtest. Earlier attempts with auto-refill removed all tension from the oil economy.

7. The Win Condition

Three beacon pieces, one per level, must be collected. They cannot be skipped—each level's EndLevel trigger gates on having collected that level's piece. The player cannot proceed without engaging the climax of each level.

The pieces converge at the Win Scene: a hollow beacon in a final cave area. Touching it with all three pieces displays the win text: 'The cave is lit. Move onto the next area.'

The framing is intentional: the prototype is positioned as the first chapter of a larger world. The player has lit one section. There are more to come. This sets up future content while keeping the prototype's scope contained.

8. Playtest Findings

Several iterations of playtesting shaped the final design. The most significant learnings:

Oil drain rate was reduced from 1.5/sec to 1.0/sec.

Initial playtest showed players consistently running out of oil in Level 2's stalactite sections. The mechanic was tight, but tight in a frustrating way—not in a tense way. Reducing the drain rate gave the player just enough headroom to feel pressure without consistent failure.

Stalactite warning duration tuned to 0.5 seconds.

The original 0.2-second crack-to-fall window was too tight for reactive play. Players reported deaths that felt unfair—they heard the sound but could not act in time. Extending the warning gave players a fair chance to dodge while still demanding alertness.

Monster speed was kept at 1.6x player speed.

Tested at 1.2x (felt avoidable), 1.6x (felt threatening), and 2.0x (felt unfair). The middle value preserved the design intent: monsters must be avoided through torch management, not outrun.

Platform-when-dark mechanic placed in Level 3's Twist beat.

Earlier sketches placed monster-as-platform earlier in the level. Playtesters had no framework to interpret it—they had not yet learned 'monster = threat,' so 'monster = platform' came too soon. Placing it in the Twist beat, after two encounters establishing threat, dramatically improved comprehension.

9. Scope and Constraints

The prototype is intentionally scoped to demonstrate the design argument cleanly. Three levels, one new mechanic per level, no checkpoints, no save system, no animation. Each cut was deliberate.

- Three levels was the minimum required to show the Four-Beat learning structure *between* levels—introducing oil rationing in L1, building on it in L2, climaxing in L3.
- No save system: a 15-minute prototype does not need persistence. Players experience it in one sitting.

- No animation: static sprites focus the player's attention on the mechanic, not the visual production value. The choice was deliberate and tested—players did not report wanting animation.
- Continuous oil across levels was the largest design risk. It punishes early mistakes harshly. Playtest confirmed it was correct: the cross-level consequence is what makes the oil economy feel real.

10. Design Argument Validation

The prototype's central question was: can a platformer make sight feel costly?

Playtest evidence suggests yes. Players consistently described the experience using language related to economy and decision-making: 'I was rationing,' 'I had to choose,' 'I was scared to light it.' This is the vocabulary of design success.

Players who failed Level 3 universally cited oil starvation rather than mechanical difficulty. They had spent too freely earlier. They understood the failure was their own—and wanted to try again. That is the design landing.

Guiding Light demonstrates that a single coherent design argument, executed across three layered mechanics, can produce a measurable emotional outcome without relying on narrative, art polish, or genre conventions.

—

Designed, Developed and in-game art by Pratik. Solo prototype, ~3 weeks. Unity 6.3 LTS, C#, WebGL.