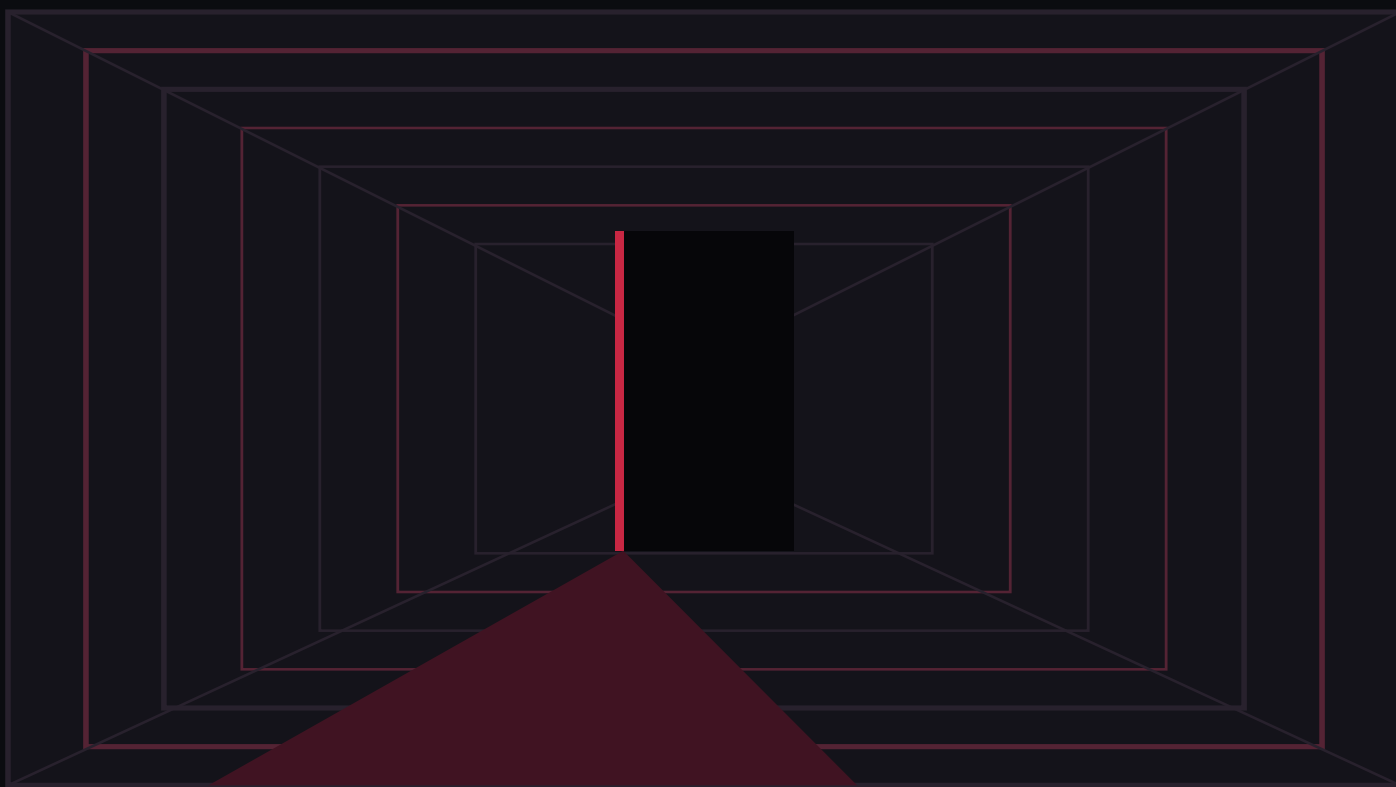


HOW TO MAKE A HORROR GAME

9 Lessons on Making Players Scared - in the Fun Way



HORROR & ATMOSPHERE

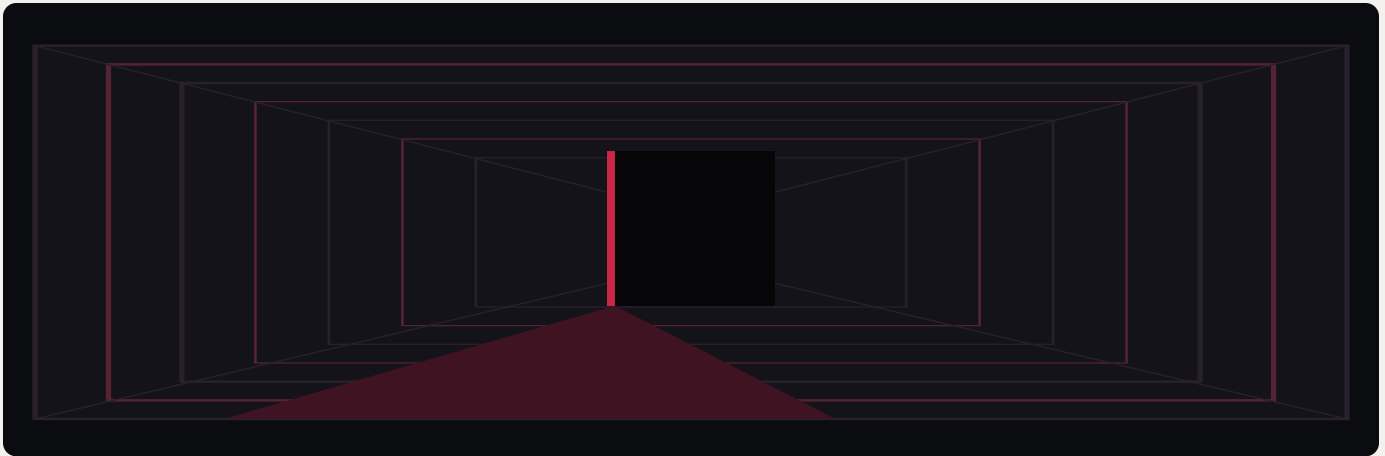
BUILD WHAT YOU IMAGINE.

FIELD GUIDE

A PRACTICAL FIELD GUIDE

How to Make a Horror Game

9 Lessons on Making Players Scared - in the Fun Way



Published by Vertex Arc

First edition / October 2026

Original text and procedural illustrations created for Vertex Arc. Copyright 2026 Vertex Arc. Free to download and read. All examples describe hypothetical projects; they do not claim client work or shipped results.

Use this guide as a set of practical experiments. Test advice in your own project and consult documentation for the engine version you ship. Game-design observations are design hypotheses, not clinical claims or guarantees. No affiliation with or endorsement by Unity Technologies is implied.

vertexarc.site/books

Inside this guide

Read in sequence, or choose the lesson that answers the next question in your project. Page numbers refer to the PDF page counter.

Introduction: fear with an invitation	4
01 Fear and traditional fun are different	5
02 The player can accidentally destroy your horror	6
03 Jump scares cannot carry the whole game	7
04 Too much power reduces vulnerability	8
05 Narrative gives fear meaning	9
06 The world needs believable rules	10
07 What you do not show can be frightening	11
08 Give the player a believable role	12
09 Player agency makes horror personal	13
Field checklist: a small horror playtest	14
Conclusion: earn the next step	15

About Vertex Arc / 16

Introduction: fear with an invitation

The player opens a maintenance hatch because they chose to, not because a film editor cut to the next shot. That small difference is the material of interactive horror. The game can make someone reluctant to continue while still giving them a compelling reason to take the next step. Your job is to shape that conflict without turning it into irritation or helpless waiting.

This guide uses an original hypothetical project called Night Maintenance. The player is a caretaker inspecting an isolated flood-control station after its automatic alarms stop reporting correctly. There is no existing game behind this example. It gives us a shared setting in which to test design choices: narrow service passages, familiar machinery, unreliable reports, and a job that must be completed before the water rises.

The nine lessons are design hypotheses, not universal claims about psychology. People respond differently to darkness, pursuit, uncertainty, sound, and loss of control. Test with the audience you intend to reach, ask about comfort boundaries, and distinguish an interesting fear from a confusing interface. A player who cannot read a prompt is not experiencing the same tension as a player who understands the danger and chooses to proceed.

Read one lesson, build its exercise in a small greybox scene, and observe another person playing it. Keep a short notebook with three columns: what you intended, what the player did, and what you will change. A corridor with one reliable interaction can teach you more than an elaborate environment whose rules are still undecided.

THE TENSION LOOP



Use the result to shape the next attempt.

01 / Fear and traditional fun are different

THE IDEA

Do not evaluate every moment of a horror game by whether the player feels powerful or immediately rewarded. A slow approach, a reluctant inspection, or a temporary retreat can be valuable if it creates anticipation and makes the eventual action meaningful. The experience still needs satisfaction, but that satisfaction may come from understanding, surviving, or resolving uncertainty rather than defeating an enemy.

Separate productive discomfort from friction. Productive discomfort asks a clear question: should I cross the exposed floor while the machinery is quiet? Friction asks the player to fight an unclear interaction prompt or repeat a long walk after a trivial mistake. The first creates a decision. The second consumes patience without developing the fear.

WHY IT WORKS

Players can enjoy the larger arc of tension and relief even when an individual moment feels unpleasant. They need enough trust in the game to believe that paying attention matters. If a cautious action produces understandable consequences, hesitation becomes part of play. If consequences seem arbitrary, the player may stop investing in the situation.

EXAMPLE

In *Night Maintenance*, a pump has to be restarted by hand. The controls are easy to operate, but each turn briefly silences the station's background hum. During that silence, the caretaker can hear movement beyond the service wall. The task is mechanically simple; the decision is whether to continue immediately or pause to listen. Completing it restores light to a safe route and gives the tension a useful resolution.

KEY TAKEAWAY

Make the action understandable and the situation unsettling. Difficulty operating the game is not a substitute for fear inside the game.

TRY THIS

Build one room with a three-step interaction and a nearby uncertain sound. Test it first with no threat at all. Ask the player what they were worried might happen and whether finishing felt worthwhile. If they cannot explain the task, repair its feedback before adding an enemy. Compare a short version with a longer one to find when anticipation becomes waiting.

DEVELOPER NOTE

Write down the intended emotional sequence: notice, hesitate, commit, recover. Use that sequence to judge changes instead of increasing intensity everywhere.

02 / The player can accidentally destroy your horror

THE IDEA

Players inspect boundaries. They jump on furniture, repeat sounds, stare directly at a creature, and test whether a door can trap it. These actions are normal attempts to understand a game. Horror weakens when the world responds in obviously mechanical ways that contradict its intended fiction. The answer is not to prevent every experiment, but to decide which experiments the game can support convincingly.

Look for interactions that expose the implementation rather than the world: an enemy endlessly turning against a chair, a threatening sound repeating on every trigger crossing, or a character calmly reusing the same line during a crisis. Fixing these details may strengthen atmosphere more than adding another scare.

WHY IT WORKS

A player who discovers a dependable exploit can move from uncertainty into analysis. That can be enjoyable, but it changes the emotional relationship. Preserve useful learning while avoiding loopholes that turn a supposed threat into a toy. The player should learn believable rules, not merely the weaknesses of a navigation setup.

EXAMPLE

The station's pursuer cannot fit through a narrow inspection opening. That is a valid refuge, but the opening is not an infinite victory button. The creature waits outside, reacts to sound, and eventually takes another route. The player gains time to plan without learning that crouching behind one crate disables the entire encounter.

KEY TAKEAWAY

Expect experimentation. Design the world's response so investigation reveals coherent rules instead of brittle scripting.

TRY THIS

Give a tester permission to behave badly in your scariest room for ten minutes. Ask them to repeat triggers, block paths, reverse direction, and approach the threat closely. Record the first three moments that become funny or obviously artificial. Fix the most damaging one, then repeat the same actions to verify the response. Do not remove a harmless player freedom simply because it was unexpected.

DEVELOPER NOTE

In Unity, draw navigation targets and state labels only in a developer view. Debug the failing transition directly; increasing enemy speed usually does not fix a stuck state.

03 / Jump scares cannot carry the whole game

THE IDEA

A sudden event can punctuate tension, but it needs a surrounding structure. Repeated surprises with the same timing become predictable. Random surprises with no relationship to the player's decisions can feel cheap. Ask what each scare reveals, changes, or teaches. If its only purpose is to increase volume, it may not deserve a place in the sequence.

Build anticipation through a recognizable situation, a reason to pay attention, and a period in which the outcome remains uncertain. The surprising event can then resolve or redirect that expectation. Silence and restraint are useful because they give a strong event room to stand out, not because every quiet moment must hide an ambush.

WHY IT WORKS

Players form expectations from repetition. You can vary those expectations meaningfully when the world has already established a pattern. If every cabinet produces a threat, opening cabinets becomes a routine tax. If one cabinet contains evidence that changes the player's understanding of a sound, the scene can stay relevant after the initial surprise is gone.

EXAMPLE

An inspection lamp flashes whenever a pressure valve opens. Early on, the player sees and hears the harmless mechanism. Later, the lamp flashes while the valve remains closed. No creature appears. The mismatch is the event: the player now has a reason to distrust the previously reassuring signal. A later encounter can build on that information instead of repeating a loud reveal.

KEY TAKEAWAY

A scare should leave something behind: knowledge, a changed space, a new risk, or a different expectation.

TRY THIS

List five planned scares and write the player's understanding immediately before and after each one. Remove any event whose two descriptions are identical. Replace one with an environmental change that can be noticed without a camera lock or loud sting. Test whether the player notices it and can explain why it matters. If not, improve the setup rather than making the reveal louder.

DEVELOPER NOTE

Provide independent sound controls and avoid relying on extreme volume spikes. Intensity should come from the situation and mix, not an uncomfortable playback level.

04 / Too much power reduces vulnerability

THE IDEA

Every reliable solution changes how threatening a situation feels. Unlimited ammunition, an effortless escape, or a stun that always resets the enemy may let the player control an encounter completely. That can support an action game, but it can undermine a design centered on vulnerability. Decide what the player can do well and where they must make a tradeoff.

Vulnerability does not require removing all tools. Tools can create tension when their benefits carry understandable costs: light improves navigation but reveals position, a loud machine opens a route but attracts attention, or a temporary barrier consumes a resource needed later. Give the player enough capability to make choices, then make those choices matter.

WHY IT WORKS

A limited tool encourages planning when the player can predict its effect. An unreliable tool with unexplained failures encourages distrust. Keep the difference clear. Resource scarcity also needs context: if players cannot judge what they may need later, they may hoard everything and miss the intended interactions entirely.

EXAMPLE

The caretaker carries a portable floodlight. It keeps the pursuer at a distance while powered, but drawing electricity from a damaged circuit shuts down the nearby door motor. The player can create a temporary safe zone or open a route, but cannot maintain both indefinitely. The light is useful because it creates a decision, not because it offers an automatic win.

KEY TAKEAWAY

Preserve agency by giving tools clear strengths and meaningful limits. Power becomes tension when using it changes another part of the situation.

TRY THIS

Take the strongest tool in your prototype and test the encounter with unlimited uses. Identify the point where the player stops caring about the threat. Add one readable cost, then test again without explaining it verbally. Watch whether players adapt or simply avoid the tool. If they never use it, make its benefit or cost easier to judge.

DEVELOPER NOTE

Tune one constraint at a time. Reducing duration, range, availability, and reliability together makes it hard to learn which limitation created the useful decision.

05 / Narrative gives fear meaning

THE IDEA

A threat matters more when the player understands what is at stake beyond a failure screen. Narrative can establish a responsibility, a relationship, or a consequence that makes a risky action worth taking. This does not require long cutscenes. A task, an object, or a change in a familiar space can carry the necessary meaning.

Connect story information to the decisions the player is already making. A note that explains why a door was sealed can change how someone approaches it. A page of unrelated history may enrich the world, but it should not be expected to create immediate tension on its own. Ask how each piece of information changes the next action.

WHY IT WORKS

Specific stakes give uncertainty a direction. The player is not merely avoiding a monster; they are deciding whether to restore a system that protects someone, retrieve evidence, or keep a promise. The design can then test that commitment through choices. Avoid assuming that a tragic backstory automatically creates emotional investment without time to understand it.

EXAMPLE

Night Maintenance begins with a routine request from the day caretaker: keep the lower pumps online because a nearby shelter depends on them. Later, the safest escape requires cutting power to that circuit. The story is now present in a mechanical choice. The player can understand the cost without pausing to read a long explanation while being chased.

KEY TAKEAWAY

Story strengthens horror when it gives the player's next decision a reason to matter.

TRY THIS

Choose one objective and write two versions: a purely functional instruction and an instruction with a concrete consequence. Build both with the same layout and threat. After testing, ask what the player believed would happen if they abandoned the task. If they invent a different stake, decide whether the fiction needs clearer evidence or whether their interpretation is useful.

DEVELOPER NOTE

Do not put essential understanding only in optional reading. Let the main interaction communicate the basic stakes, then use optional material to add depth and ambiguity.

06 / The world needs believable rules

THE IDEA

Believable does not mean realistic in every detail. It means the world behaves consistently enough that players can reason about it. If sound attracts a threat, define which sounds count. If darkness conceals the player, define how the enemy responds at the boundary. Exceptions can exist, but they need an explanation or a readable signal.

Write a short rules sheet before building complex encounters. Describe perception, safe spaces, interaction costs, and failure recovery in ordinary language. If two team members interpret a rule differently, players probably will too. Resolve that disagreement while the system is still small.

WHY IT WORKS

Fear can coexist with learning. A player who understands some rules can make a plan and worry about whether it will work. A world that changes rules without warning removes the basis for that investment. The player may still be surprised, but surprise alone is not a durable relationship with the game.

EXAMPLE

The pursuer responds to metal impacts carried through the station's pipework. Footsteps on rubber mats are relatively safe; dropping a tool near an exposed pipe is dangerous. An early harmless event demonstrates the connection. A later room combines the same rule with a useful shortcut, allowing the player to decide whether the noise is worth the time saved.

KEY TAKEAWAY

Keep the rules stable enough to support a plan. Make exceptions legible before asking the player to bear their consequences.

TRY THIS

Build three versions of a sound source: one clearly detectable, one clearly safe, and one near the threshold. Test enemy responses repeatedly from the same position. Then let a player describe the rule in their own words. Compare their explanation with your rules sheet. A mismatch may come from implementation, presentation, or an unnecessarily complicated rule.

DEVELOPER NOTE

Log why a threat changes state. A visible state transition and its trigger make fairness bugs easier to diagnose than watching only the final chase behavior.

07 / What you do not show can be frightening

THE IDEA

Withholding information can invite anticipation, but absence needs evidence around it. A moving shadow, an interrupted routine, or a sound from a sealed space gives the player something to interpret. An empty room with no context is simply empty. Choose what the player knows, what they suspect, and what remains unresolved.

Treat reveals as changes in information. Showing the creature clearly may answer questions about its size and movement while opening questions about its intention. You do not have to hide everything forever. The goal is to manage the player's model of the threat, not to delay a complete explanation by an arbitrary number of minutes.

WHY IT WORKS

Incomplete but coherent evidence encourages people to make predictions. Different players will imagine different possibilities, so test which evidence actually reaches them. Do not assume your intended implication is obvious just because you know the answer. A subtle cue that nobody notices contributes little to the experience.

EXAMPLE

The caretaker sees a wet handprint on the outside of a viewing window whose corridor has been dry for hours. Later, a chair behind that window changes position while the player services another room. There is still no clear sighting, but the evidence suggests movement and access. The next trip past the window now carries a question created by play.

KEY TAKEAWAY

Hide the answer, not every clue. Uncertainty needs enough information to become a meaningful expectation.

TRY THIS

Replace one direct enemy reveal with two separate pieces of evidence. Space them so the player has time to interpret the first. After the test, ask what they thought was present and which detail led them there. If they noticed nothing, improve contrast or timing. If they understood everything immediately, remove a redundant clue rather than adding unrelated mystery.

DEVELOPER NOTE

Make essential cues survive ordinary camera movement and display settings. A clue visible only when standing still at one exact angle is fragile unless searching for it is the intended task.

08 / Give the player a believable role

THE IDEA

The player's abilities, knowledge, and responsibilities should fit the person they are controlling. A maintenance worker reasonably understands machinery but may not know how to fight. A researcher may recognize instruments but lack access to security systems. This alignment makes the interface and objectives feel like parts of the world instead of arbitrary game rules.

Role also explains why the player stays involved. Avoid relying only on a locked exit when a more specific obligation can motivate action. The player can still want to escape, but they should understand why the character entered, what they can plausibly do, and why the next task is their responsibility.

WHY IT WORKS

A coherent role reduces the gap between player intention and character behavior. It becomes easier to accept specialized interactions and limited tools when they belong to the fiction. The design can also teach through familiar work: a routine inspection establishes normality before something interrupts it.

EXAMPLE

Before anything overtly threatening happens, the caretaker checks three gauges and tags a faulty valve. These actions establish competence and a physical route through the station. Later, the same valve has been reopened. The player recognizes the change because they previously handled it, not because a quest marker announces that the situation is now frightening.

KEY TAKEAWAY

Let the role explain the player's tools, knowledge, and reason to act. Familiar work can become the foundation of an unfamiliar threat.

TRY THIS

Write a one-paragraph job description for your player character. Compare every core interaction against it. Keep surprising abilities only when the game explains how they fit. Build a short routine that teaches two essential controls before danger appears, then alter one detail of that routine later. Test whether the player notices the violation without a tutorial message.

DEVELOPER NOTE

Do not mistake a believable role for mandatory realism. Simplify repetitive work into clear decisions; the goal is coherence, not an exact simulation of an occupation.

09 / Player agency makes horror personal

THE IDEA

Agency is the ability to make choices that have understandable consequences. It does not require a branching story at every turn. Choosing a route, deciding when to make noise, or spending a limited resource can make the player feel involved in the risk. The fear becomes connected to something they chose rather than an event they merely watched.

Avoid offering a choice that the game immediately invalidates without explanation. A scripted capture can work, but repeated unavoidable failures teach players that planning is irrelevant. If a scene temporarily removes control, make the transition clear and return meaningful action before passivity becomes the dominant experience.

WHY IT WORKS

Consequences are easier to care about when the player can connect them to a decision. This does not mean every outcome must be favorable or fully predictable. It means the game provides enough information for a reasoned choice, then responds in a way that can be understood afterward.

EXAMPLE

To reach the control room, the caretaker can cross a dark service bridge or restart a noisy lift. The bridge exposes the player to uncertain movement below; the lift reveals their position but offers a faster route. Both paths are viable and use established rules. The resulting encounter reflects the choice, so repeating the section can produce a different plan rather than the same compulsory surprise.

KEY TAKEAWAY

Let the player own a part of the danger. A small, consequential choice can create more involvement than a large scripted spectacle.

TRY THIS

Create two routes that differ in one meaningful cost. Keep their destination and approximate completion time comparable. Ask testers to explain their choice before entering and their interpretation afterward. If everyone picks one route because the other is obviously worse, adjust the tradeoff. If the consequences seem unrelated to the choice, strengthen the causal feedback.

DEVELOPER NOTE

Checkpoint placement shapes agency too. Preserve important choices when appropriate, but avoid forcing long solved sections to repeat after a short failure. Repetition can replace tension with impatience.

Field checklist: a small horror playtest

- Define the intended fear in one sentence and the player's reason to continue.
- Confirm that essential controls and objectives are readable without coaching.
- Write the threat's perception and response rules before testing exceptions.
- Test the strongest player tool with repeated use and look for trivial solutions.
- Observe one cautious player and one player who deliberately probes boundaries.
- Track what each scare changes in the player's understanding.
- Include quiet intervals that provide context or relief rather than empty delay.
- Test sound-off information access, captions, volume controls, and reduced motion.
- Ask players what they expected, what surprised them, and what felt arbitrary.
- Change one important variable, preserve the previous build, and compare again.

Conclusion: earn the next step

Horror design is a continuing negotiation between uncertainty and trust. You can make the situation disturbing while keeping the interaction clear. You can limit power while preserving meaningful tools. You can withhold a threat while showing enough evidence for the player to imagine it. These relationships matter more than the number of frightening assets in the project.

Return to one small scene and ask whether the player understands their role, their immediate task, and at least one consequence of acting. Then ask what remains uncertain. If both answers are clear, you have a useful foundation. Refine that foundation through observation before expanding the world around it.

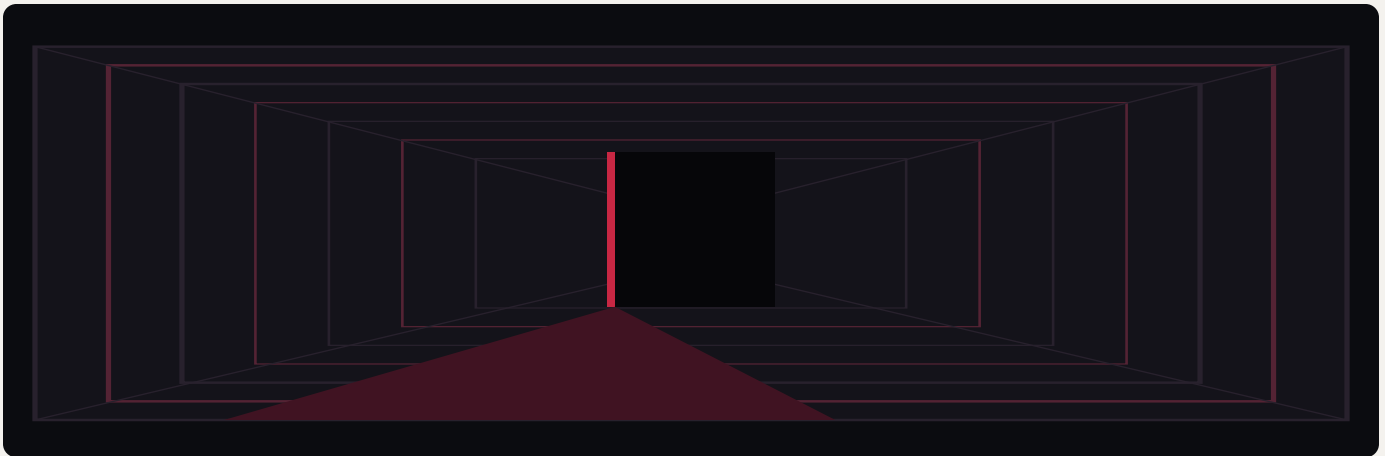
The next useful milestone is a short, complete sequence: a routine, a disturbance, a decision, and a recovery. Build it with temporary art and clear sound. Let a new player experience it without explanation. Their hesitation, curiosity, and choices will show you where the real horror is beginning to work.

About Vertex Arc

BUILD WHAT YOU IMAGINE.

Vertex Arc is an independent game-development studio and creative engineering partner. Its work spans playable prototypes, game systems, and interactive experiences across PC, mobile, and web. This publishing series shares practical ways to ask better questions, make focused experiments, and improve the experience players actually receive.

The guides are designed to be used beside a project. Keep the exercises small, record what you observe, and adapt the next step to the evidence. Finishing one useful experiment is a better result than collecting advice that never reaches a playable build.



[Browse the complete free library](#)

[Explore the studio and its work](#)

[Start a project with Vertex Arc](#)