scenario

Style frame: GPT Image 2.5 Sunburst

HOW IT WAS MADE

Graveyard Waves

Making a Roblox wave survival game with Scenario assets. Every mesh, rig, motion clip, texture, sky, icon, sound effect and music track came from a Scenario model, run through the Scenario MCP by Claude Code with Opus 5.5.

5
waves of undead

20
textured meshes from concepts

38
Meshy Animation clips

12
Scenario models

121
successful runs

0
assets uploaded to Roblox



The same graveyard in Roblox Studio

See the exploration

Scenario MCP

A wave survival game, every asset from Scenario

The brief: a small Roblox wave survival game in a night graveyard. Five waves of undead rise from the graves and march on a shelter stone at the centre; players fight them off with four weapons that unlock one by one. One to eight players, on phone and PC.

THE RULE

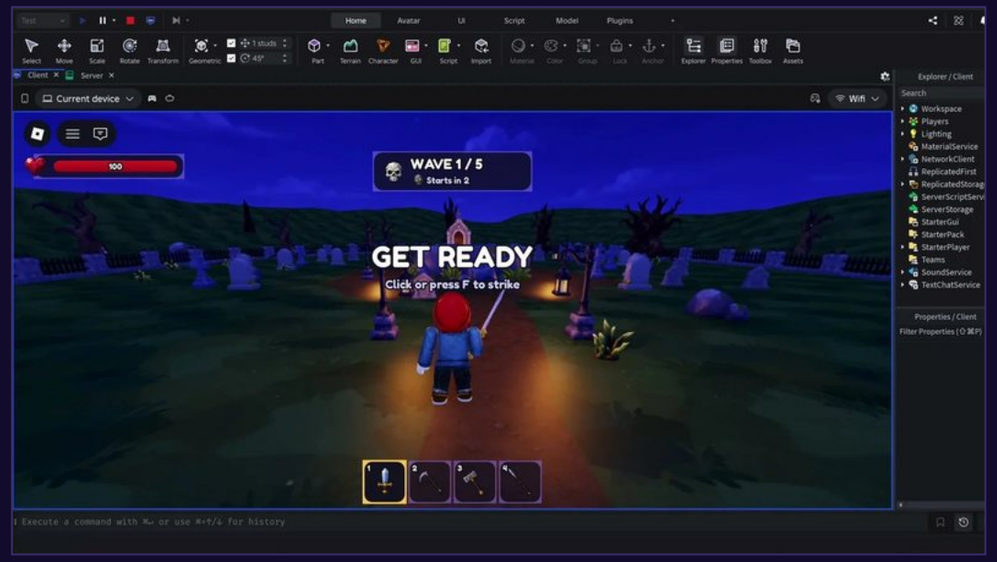
Every model, texture, sky, icon, VFX sheet, sound effect and music track is made with Scenario, through the Scenario MCP, in one stylized low-poly look, and saved in one Scenario project. Nothing is uploaded to Roblox: the game builds its assets at runtime.

THE TIMELINE

First Scenario run at 11:32 on 19 October 2026 (Paris time). Five waves played end to end at about 12:40, still with rigid zombies. The rest of the day went to reviews and fixes: rigs, weapons, sound, controls.

LOCAL TOOLS

Rojo (files to place), Lune and luau-lsp (tests, type checks) and headless Blender 5.2 (scale, pivot, triangle gate, texture size) convert and check. None of them makes an asset.



Roblox Studio with the game in Play. The agent drives it through the Roblox Studio MCP.

Claude Code

Opus 5.5

- Picks every model
- Writes the game in Luau
- Bakes the assets
- Runs the playtests
- Applies the review fixes

Scenario MCP

every asset

- `search`
- `recommend`
- `model_schema_get`
- `model_run`
- `jobs_wait`
- `asset_download`

Roblox Studio MCP

the live place

- Builds the place
- Runs Luau in Edit and Play
- Drives the playtests
- Captures each wave

14 Roblox skills

coming to skills.sh

`expert` `architecture` `gameplay` `animation` `vfx-audio` `ui` `environment` `assets` `networking` `performance` `luau` `tooling` `data` `monetization`

`scenario-roblox-*`

Search, recommend, read the schema, run

Before each new kind of asset, the agent asked the Scenario MCP for the best and newest model instead of a familiar default, then read that model's schema before the first run. The same four calls ran for every model in the build. Here they are for the meshes, with the session's real arguments.

1 SEARCH — 9 calls

`search`

```
target: "models", query: "3d",
public: true, sort_by:
["createdAt:desc"]
```

The newest 3D models first.

2 RECOMMEND — 7 calls

`recommend`

```
prompt: "Turn stylized low-poly game
prop and character concept images
into textured game-ready 3D meshes
for Roblox"
```

RANKED

1 Meshy T2.5 Smart Topology
2 Tripo P2
3 Meshy 7.1
4 Rodin Gen-2.5

3 SCHEMA — 19 calls

`model_schema_get`

```
model_id:
"model_meshy-t2-5-smart-topology"
```

image [array], targetPolycount 100 to 30,000, shouldTexture, textureResolution.

4 RUN — 121 runs

`model_run`

```
model_id:
"model_meshy-t2-5-smart-topology",
image: [the concept],
targetPolycount: 1500,
shouldTexture: true
```

A textured mesh at the budget, then jobs_wait and asset_download.

WHAT THE CALLS PICKED

Images, concepts	GPT Image 2.5 Sunburst	Ranked first; takes up to 10 reference images and returns transparent backgrounds. Runner-up: GPT Image 2.5 Flare.
Image to 3D	Meshy T2.5 Smart Topology	The newest (16 October 2026); builds at a chosen face count, textured, with an A-pose option. Runner-up: Tripo P2.
Sky	Scenario Skybox GPT	Takes reference images, so the sky matches the style frame. Runner-up: Scenario Skybox Flux.1.
Ground tiles	Ideogram V4 Tiling	The specialty pick for native tiling. Dropped after two runs wrote text into the tiles (page 9).
Rig and motion	Meshy Animation	Added after review, then a walk shoot-out against Cartwheel Text to Motion (page 6).
Sound	Mirelo SFX 2.0, ElevenLabs Music v2.5	From recommend and a newest-first search; the music won a shoot-out against Google Lyria 3.5 (page 10).

One style frame, twenty concepts

The first run made a single key art image. Every later image passed it as a reference, so the props, characters, weapons and icons share one palette and one faceted look.



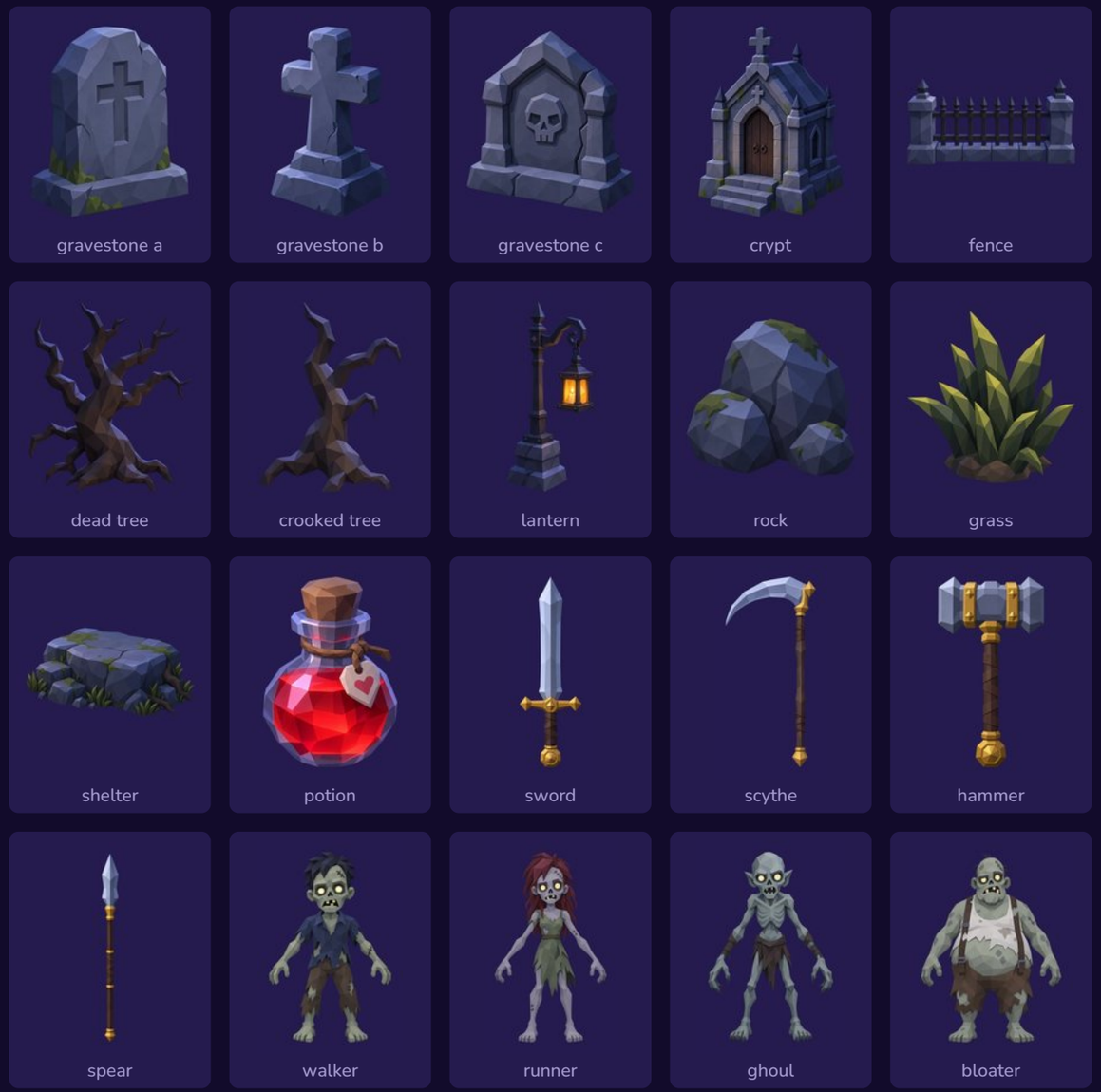
THE STYLE FRAME PROMPT (EXCERPT)

```
“Key art for a stylized low-poly 3D game: a small night graveyard arena
seen from a high three-quarter game camera. Flat-shaded faceted geometry
with large clean polygons and chunky, slightly exaggerated proportions
[...] Limited palette: indigo night, blue-grey stone, moss green, warm
orange accents. [...] no characters, no text.”
```

EVERY CONCEPT PROMPT OPENS THE SAME WAY

```
“Game asset concept for image-to-3D, in exactly the stylized low-poly
style, palette and moonlit lighting of the reference image. One single
isolated object [...] whole object in frame and centered, no ground, no
cast shadow, transparent background.”
```

Characters also passed the first zombie concept as a second reference, so the four kinds read as one family; the scythe, hammer and spear passed the sword.



The twenty concepts, GPT Image 2.5 Sunburst, transparent backgrounds.

Concepts to game meshes at a triangle budget

Each concept went to Meshy T2.5 Smart Topology with its triangle budget set at generation (targetPolycount), so the mesh came back textured at that face count. Characters started from A-pose concepts, ready for rigging.



Crypt — 6,000 triangles

Dead tree — 3,500 triangles

Lantern post — 1,800 triangles

Gravestone — 1,500 triangles

Zombie (A-pose) — 5,000 triangles

Fence — 1,500 triangles

Scythe — 1,500 triangles

Potion — 1,200 triangles

Left: the concept. Right: the Meshy T2.5 Smart Topology mesh after the bake, rendered in Blender.

EVERY BUDGET

Gravestones, sword and fence 1,500 triangles; lantern 1,800; rocks 1,400; grass 700; crooked tree 2,500; dead tree 3,500; first zombie 5,000; crypt 6,000; the three new characters 5,000 each; scythe, hammer and spear 1,500; potion 1,200; shelter stone 3,000.

THE BAKE

Blender 5.2, run headless, sized each mesh in studs, turned it to face the game's forward axis, set the pivot at the bottom centre and gated the triangle count. The lantern also got a glow mask from its warm bright pixels.

TWENTY MESHES, TWENTY-TWO RUNS

The potion and shelter runs failed once with an internal error; both were refunded and the reruns succeeded. Later, the heavy props were trimmed to fit the client's memory budget (page 11).

Rigs and motion from Meshy Animation

The first zombie was cut into six rigid parts and animated in code. That was a mistake. After the review (“we have animation and rigging tools on Scenario”), the agent switched to Meshy Animation: it rigs the mesh itself with 24 bones and returns library motions on that rig.

THE WALK SHOOT-OUT ON ONE ZOMBIE MESH, AND THE CHASE



Winner

Meshy Animation, Mummy Stagger

Upright shamble, arms reaching, readable face.

Runner-up

Cartwheel Text to Motion

Too hunched: the face is hidden and the head deforms.

Rejected

Meshy Animation, Monster Walk

Turned the body sideways.

Chase

Meshy Animation, Lean Forward Sprint

Blended with the walk by ground speed.

Blender previews of each run on the same mesh.

WHY A SPRINT TOO

The stagger covers 0.73 studs a second; zombies chase at 8 to 12. Played that fast it would skate, so a sprint run takes over by ground speed, each clip at the rate that keeps the planted foot still.

ONE BIND FOR EVERY CLIP

Every Meshy Animation run rigs the character again: same 24 bone names, but bones up to 4% longer and frames turned up to about 20 degrees. Clips are moved across in world space onto one bind.

SKINNED AT RUNTIME

A probe showed an EditableMesh can carry bones without an upload. The first build stretched the limbs; binding each bone with AddBone in mesh space fixed it. Each rig was later decimated to about 2,500 triangles, 24 bones kept.



The bind in Roblox: bones bound parent-relative (top), then in mesh space.

Four kinds, a clip for each use

Each kind has its own concept, mesh and Meshy Animation rig. The walker's six clips are the base set; a new kind gets its own clip only where the shared one did not fit. Durations are from the game's bake files.



Walker

Wave 1

The base stats.

walk	mummy stagger	4.83 s
run	lean-forward sprint	0.57 s
attack	two-fist lunge	2.13 s
hit	hit reaction	0.83 s
death	dying backwards	2.23 s
rise	zombie scream	2.23 s



Runner

From wave 2

Faster and frailer: health x0.75, speed x1.25.

run	head-down charge	0.5 s
death	death fall	2.93 s

Other clips: the walker's.



Ghoul

From wave 3

Crouched and fast: health x0.6, speed x1.4. The only kind that leaps onto the shelter stone.

walk	cautious crouch walk	1.13 s
leap	run and leap	1.03 s
death	blown back	2.0 s

Other clips: the walker's.



Bloater

From wave 4

Slow and heavy: health x3, speed x0.7.

attack	charged ground slam	2.43 s
death	collapse	1.9 s

Other clips: the walker's.

The slam's fist lands 1.27 s into the clip, so it plays at 1.27x to hit on the server's 1.0 s wind-up.



A walker rises with the zombie scream clip.



Ghouls leap onto the shelter stone.



Wave 5: a giant bloater (health 450, speed 9, damage 25, scale 1.7).

Four swings, one slash, a HUD

Each swing is a Meshy Animation library clip, retargeted onto the Roblox R15 avatar's upper body and played through its Animator as a KeyframeSequence built in code. It blends over the walk and replicates to other players.



Sword 1.17 s
`Right_Hand_Sword_Slash`
From the start.



Scythe 0.47 s
`Reaping_Swing`
After the first victory.



War hammer 0.93 s
`Heavy_Hammer_Swing`
After the second victory. Runner-up: Charged_Axe_Chop.



Spear 0.67 s
`Thrust_Slash`
After the third victory.

HOLDING THE WEAPON: THREE ROOT CAUSES

Library hold candidates (85, 89, 560, 335) held the weapon flat across the hips. A grip bench in Edit measured every vertex against the floor and the body. When angle tuning stopped helping, a “think deeper” pass found three causes:

1. **Mesh off the grip axis**
   The bake centred each mesh on its bounding box, so the scythe's blade pulled its shaft 1.06 studs off the axis the hand grips.
2. **A teleport at the swing start**
   The weapon jumped 4 to 6.6 studs from the carry in one frame. The swing now fades in over 0.05 to 0.15 s.
3. **Avatar proportions**
   The test avatar's arms sit 0.37 studs further in than a standard R15's. The carries are aimed at the measured avatar.



The grip bench in Edit: the four carries after the fixes.

VFX: THE SLASH

GPT Image 2.5 Sunburst



A 2x2 sprite sheet cut into four frames, played as a flipbook when the swing's Slash marker fires.



HUD



HUD sheet (GPT Image 2.5 Sunburst): health, wave, zombies left, attack. The weapon bar and potion icons are cut from the concepts.

A tiling shoot-out and one 360 sky



Grass



Dirt

IDEOGRAM V4 TILING, TWICE

Both runs wrote text into the grass and dirt tiles, so the model was dropped after the second failure.

THE SHOOT-OUT, SAME BRIEF

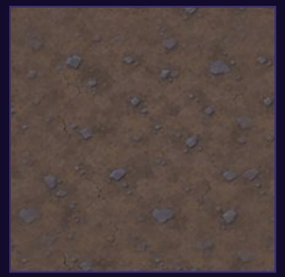
PATINA Material against GPT Image 2.5 Sunburst made seamless locally (offset and blend). PATINA won the grass, GPT Image 2.5 the dirt.



Grass: PATINA, picked



Grass: GPT, runner-up



Dirt: GPT, picked



Dirt: PATINA, runner-up



Grass in game



Dirt in game

GRADED FOR THE NIGHT

The first dressed captures showed the dirt too bright and orange, so both tiles were graded darker and cooler.

THE SKY

Scenario Skybox GPT



The equirectangular panorama, Scenario Skybox GPT.

One run of Scenario Skybox GPT, 2048 x 1024, with the style frame as its reference: an indigo night, low-poly clouds, a full moon, mountains, pines and a far ruined castle. Roblox's Sky object cannot take pixels built at runtime, so the game wraps the panorama on a self-lit inverted sphere that follows the camera.



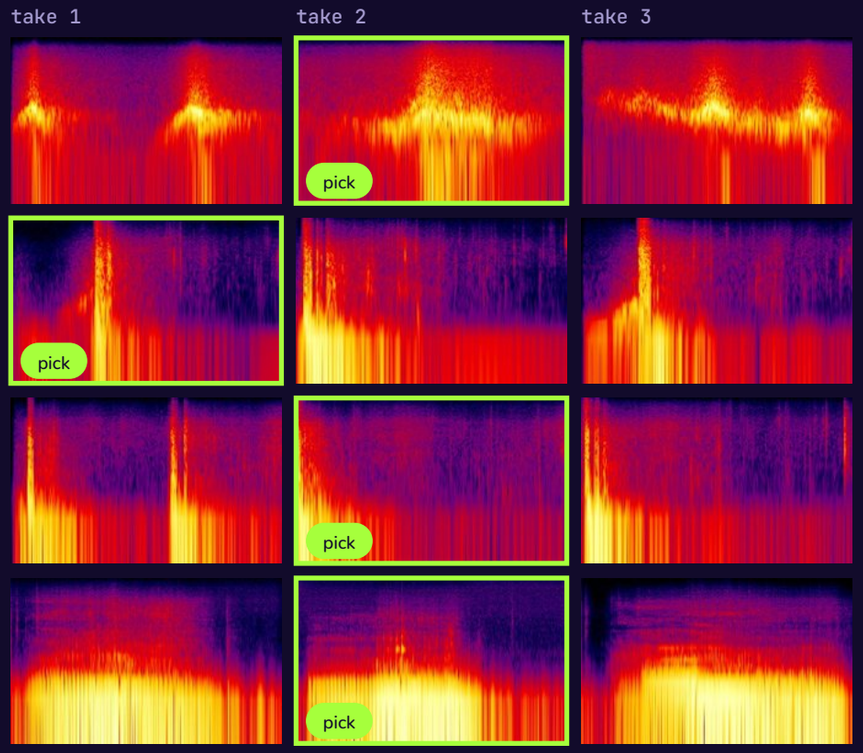
In game: the sky and the moon behind the player.

Three takes per sound, one hook across six tracks

SOUND EFFECTS

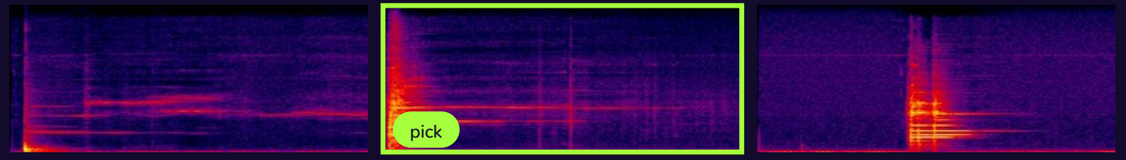
Mirelo SFX 2.0

Mirelo SFX 2.0 made three takes of each sound. Every take was checked without a listener: spectrogram, onset, length and loudness, and a whisper transcription of the voiced ones.



Zombie death take 1 was dropped: whisper heard “We're going away.” in it.

POTION PICKUP



Take 2 kept. Take 1 is a near-silent click; take 3 starts 0.38 s late.

MUSIC

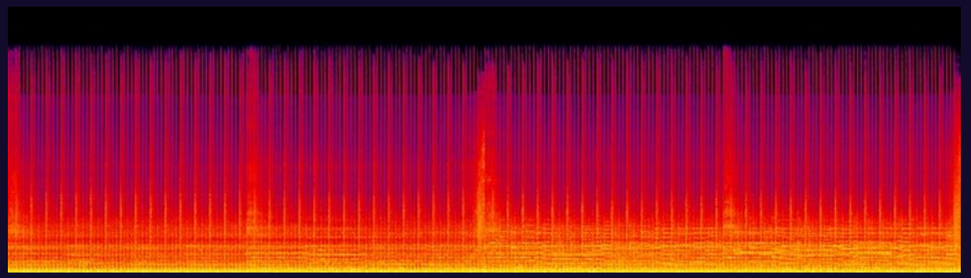
ElevenLabs Music v2.5

One brief for the whole score: a four-note riff on a detuned haunted organ, 128 BPM, A minor, instrumental, rising in intensity.

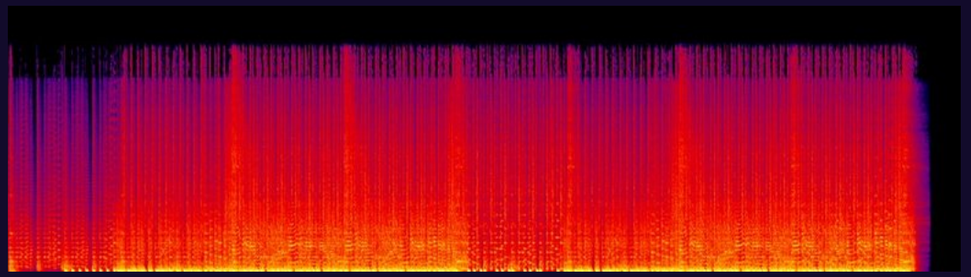
A Between waves	B Waves 1 and 2	C Waves 3 and 4
D The boss	E Victory sting, 8 s	F Game over sting, 6 s

Track B ran on two models. ElevenLabs Music v2.5 measured 129.2 BPM with no vocals and won. Google Lyria 3.5 came in at 89.1 BPM: it ignored the tempo. ElevenLabs then made A, C, D, E and F.

ElevenLabs Music v2.5, 60.02 s, 129.2 BPM



Google Lyria 3.5, 62.15 s, 89.1 BPM



Loops are cut on whole phrases, 16 bars between waves and 24 bars per wave tier, every seam within a few milliseconds of the bar line. Beds sit at -16 LUFS; whisper found no vocals in any track.

Built at runtime, played at every wave

Scenario `GLB and PNG` → **Bake** `Blender, numpy` → **Luau data** `zstd + base64` → **Client** `EditableMesh, EditableImage`

NOTHING UPLOADED

A mesh or image normally needs an upload to show in a Roblox game. Here every Scenario asset ships inside the place as Luau data, split into 180,000-character chunks (a script must stay under 200,000), about 10 MB in all. Each client decodes it and rebuilds the meshes and textures.

THE MEMORY BUDGET

The first playtest ran out of editable memory: about 80 MB per client, and every growable EditableMesh reserves a flat 10 MB. The game needed 21 meshes, about 210 MB. Fix: take a fixed-size copy of each mesh (about 1 MB per 1,000 triangles), drop the growable one, and build one at a time through a serial queue. With the rigs at 2,500 triangles and lighter props, 30 MB stays free.

PLAYTESTS THROUGH THE ROBLOX STUDIO MCP

A client autopilot plays all five waves and captures each one; the server logs every hit and frame.



WAVES 4 AND 5 STUTTERED

With 11 to 13 zombies, client frames hit 26.7 and 30.7 ms at the 95th percentile (gate 20 ms). The cost was a Highlight created on each hit flash; a 0.12 s red tint of the zombie's own SurfaceAppearance replaced it, and a later Studio playtest passed all five waves, wave 5 at 18.5 ms. The game's notes still record 2 to 4 dropped frames a second in waves 4 and 5.



The new hit flash, a tint of the zombie's own surface: none, red, bright.

NO KILLS, ENDLESS SWING

In the first hands-on test the zombies would not die: the hit box sat straight ahead. Each swing now turns the player toward the nearest zombie within 9 studs, and the server hits every zombie within 7.5 studs in an arc of about 80 degrees either side. F swings as well as a click; phones get a button. The swing no longer loops.



Controls check: the Get Ready prompt (click or press F), a swing that lands, the next wave.

Twelve models, 3,766 credits

KIND	MODEL	MADE	RUNS	CREDITS
Image	GPT Image 2.5 Sunburst	Style frame, 20 concepts, icons, slash, 2 tiles	25	391
3D	Meshy T2.5 Smart Topology	Every game mesh (20, plus 2 reruns)	22	1,200
Rig + motion	Meshy Animation	Rigs and 38 motion clips	38	1,216
Motion	Cartwheel Text to Motion	Walk shoot-out, runner-up	1	96
Sky	Scenario Skybox GPT	The night sky	1	32
Texture	PATINA Material	Grass, shoot-out winner	2	34
Texture	Ideogram V4 Tiling	Ground tiles, replaced	2	168
SFX	Mirelo SFX 2.0	Game and film sound effects	10	105
Music	ElevenLabs Music v2.5	Six game tracks	6	180
Music	ElevenLabs Music Advanced v2.5	The film score	2	120
Music	Google Lyria 3.5	Music shoot-out, runner-up	1	16
Voice	ElevenLabs v4	Film narration	13	208
Total	123 runs, 121 succeeded			3,766

Credits from Scenario usage for the project, for the game and both versions of its making-of film. Two Meshy T2.5 runs failed with an internal error and were refunded.

Make your own

The exploration page shows the game, its assets and the full prompt list. The skill lets your own agent run the same route through the Scenario MCP: pick each model with search and recommend, read its schema, run it, and bring the assets into a Roblox wave game.

```
npx skills add scenario-labs/skills
```

Installs Scenario's public skills. The 14 Roblox skills used here are coming to skills.sh.

- See the exploration
- Download the skill (ZIP)
- Scenario MCP
- Scenario Skills
- skills.sh
- Create on Scenario



Victory: all five waves survived.