

# Yu (David) Bao

## GAME DESIGNER · LEVEL × NARRATIVE × SYSTEMS

Los Angeles, CA · yubao9527@gmail.com · (213) 449-0511  
davidyubao.com · baoy.itch.io · linkedin.com/in/yudavidbao

USC Games sophomore who builds **levels where the space carries the story**: hand-crafted 3D spaces in which gating, sightlines, and pacing deliver the narrative beats. I implement my own designs in **Unity/C#**, and ran 50+ playtests as Usability Lead on a **Steam-shipped** 50+-person production. **CRPG player**: 200+ hrs in Baldur's Gate 3, 170+ in Divinity: Original Sin 2, 80+ in Solasta.

### EDUCATION

**University of Southern California — School of Cinematic Arts (USC Games)** Los Angeles, CA · Expected May 2029  
B.A. in Game Development and Interactive Media (Interactive Media & Games Division) · Sophomore · GPA 3.92/4.0

### EXPERIENCE

**Usability Lead** — *Sisyphus's Worst Day (again)* · USC AGP Aug 2025 – May 2026 · **Shipped on Steam, Jul 2026**

- Ran **50+ playtests** and the studio-wide "Everybody Tests" sessions on USC's year-long, 50+-person Advanced Games Project; turned observations into prioritized issue reports for the design and engineering leads on **level readability and difficulty pacing**, onboarding, camera, and fail states.
- Proposed **level and character-design changes** from that data and worked with the designers on the tutorial and level revisions—zoomed-out default camera, closed tutorial gaps, flattened late-level difficulty spikes—before gold master.

### SELECTED PROJECTS

**Room Numbers** — Solo · level design, C# gameplay, Blender, lighting, audio · first-person puzzle Unity 6 (URP), C# · 2026

- Designed the **two-circuit core mechanic** and the level around it: a backup generator powers only two rooms at a time, so reading the next door code means choosing what to leave in the dark. Laid out the **critical path, glass-partition sightlines, and one-way gating** across eight rooms so every power decision has a cost.
- **Taught the rule through failure** instead of text (overload the generator once and the panel tells you why). Ran a **~20-player playtest**, threw out the first extension, and rebuilt it as the Annex: a second level that escalates the same verbs rather than adding mechanics.

**Traveler** — Co-designer · story, level design, scene building, scripting · 2-person team · 3D narrative Unity 3D, C# · 2026

- Wrote the story and designed and built all **eight scenes from blackout to set dressing and lighting**: a criminal's path toward heaven or hell, with the moral choice carried by exploration and staged story beats rather than a dialogue menu.
- **Scripted the story triggers** in C# so beats fire from where the player walks and looks, not from cutscenes.

**Glimmer** — Co-designer & developer · 2-person team · narrative puzzle Unity 2D, C# · 2026

- Designed the **mood-driven visibility system** at the core of the game: a lower mood darkens and narrows the player's view but raises speed and attack, so emotional state decides what can be seen and touched.
- Wrote the **GDD** and story, built scenes 1–4 with lighting, and implemented player movement/animation and the mood UI; partner owned ghost AI, brightness, cutscenes, and dialogue code.

**Fishitting** — Design & programming · 3-person team · CIGA Game Jam 2026 · 48 hours Unity 2D, C# · 2026

- Designed an eco-satire arcade loop where upgrading is both the win condition and the trap; programmed and **tuned the economy, projectile combat, and upgrade curve** in 48 hours.

**After They're Gone** — Solo · design, environment art, implementation · environmental storytelling Unity 3D, C# · 2025

**Blocked out and dressed** a city humans abandoned so its story is told entirely by the spaces the robot explores.

Also: **Overtale** — solo 2D platform shooter, all code and pixel art self-made (Unity, 2024) · **Picnic Panic!** — co-op tabletop game; rules, balance, playtesting (2024) · **CAMI** — AI-companion concept, solo UX/UI case study in Figma (2024).

### SKILLS

|           |                                                                                                           |
|-----------|-----------------------------------------------------------------------------------------------------------|
| ENGINES   | Unity (2D & 3D, Unity 6 URP), Blender, Git                                                                |
| SCRIPTING | C#, C++, Java                                                                                             |
| DESIGN    | Level design (blockout, gating, sightlines, pacing), narrative & systems design, GDDs, playtesting, UX/UI |
| TOOLS     | Figma, Google Docs/Sheets; AI prototyping (Claude Code, Codex, Unity MCP; multi-agent orchestration)      |
| LANGUAGES | English & Mandarin Chinese (bilingual)                                                                    |