# Process & Decision Documentation

## Project/Assignment Decisions

### Side Quests and A4 (Individual Work)

Redesigned the example p5.js game (“Panic Blob”) by iterating with ChatGPT to add more complexity, clearer interaction rules, and a more complete user flow. Key scope decision: instead of only changing visuals, I added structured game states (intro → gameplay → end screen), a win condition (steal all coins), and restart/end navigation to make the experience feel like a finished mini-game rather than a single mechanic demo.

## Role-Based Process Evidence

### Entry Header

Name: Lynette Shen

Role(s): game designer

Primary responsibility for this work: Redesign the example code by adding my own creation and complexity; prompt and iterate with ChatGPT to generate improved code; test and debug changes in VS Code.

#### *Goal of Work Session*

Briefly describe what you were trying to accomplish during this phase of the assignment.

Examples:

- Turn the starter p5.js blob example into a more complete game loop (start → play → end).
- Improve UI layout in the intro (title + button alignment, hierarchy, centered composition).
- Add a clear win condition and an end screen that matches the intro style.
- Add restart and “end to intro” options so the game is replayable and user-friendly.
- Debug logic issues (state switching, button click detection, counters).

## Tools, Resources, or Inputs Used

- **GenAI tools:** ChatGPT 5.2 (code iteration + debugging + feature additions)
- **Code environment:** Visual Studio Code, p5.js runtime
- **Course/assignment inputs:** Side Quest Week 2 instructions + A4 documentation requirements
- **Peer/TA check:** Quick confirmation that GenAI usage was allowed for this task
- **Testing method:** Run-and-check testing (visual + interaction testing after each code change)

### *GenAI Documentation*

**If GenAI was used (keep each response as brief as possible):**

**Date Used**: Jan 27th, 2026

**Tool Disclosure**: ChatGPT 5.2

**Purpose of Use**: as assignment instructed and approved by professor; use chatGPT to redesign the side quest week2 (“game blob”).

**Summary of Interaction**: generated 5 times of different codes, so the game is visually,logically, overall upgraded and more creative.

**Human Decision Point(s)**: I was able to modify the prompt with chatGPT after running the code on visual studio code, and test it, come up with things so chatGPT can add. Making the game better.

**Integrity & Verification Note**: double checked with my peers and TA, made sure that GenAI is allowed in this assignment.

**Scope of GenAI Use**: GenAI was used throughout the assignment,

**Limitations or Misfires**: The tool will misunderstand some of my instructions, i will have to go back and past the code(into my prompt). Instead of describing it without the actual code.

### *Summary of Process (Human + Tool)*

- **Baseline review:** Started with the provided p5.js code and identified that it behaved more like a mechanic demo than a full “game” (no clear start/end structure).
- **UI iteration (intro):** Adjusted title size and alignment issues by rewriting the intro text placement to align with the button center. Tested visually and tweaked spacing.
- **Feature expansion:** Added game-state structure so the player must start from an intro screen (with a minimum wait time), then enter gameplay.
- **Win condition design:** Defined success as “steal all coins” instead of “steal anything,” because it creates a clear objective and ending.
- **End screen implementation:** Created a finishing screen that reuses the intro’s visual language (white background, blob on right, centered title + buttons).
- **Navigation + replayability:** Added two buttons: **Restart** (immediate replay) and **End** (returns to intro).
- **Testing + debugging:** After every major change, ran the sketch and tested interactions: button clicks, state transitions, coin counting, and whether the end triggers correctly.

### *Decision Points & Trade-offs*

**Added structured screens (intro/end) instead of only adding more hazards or art**

**Options considered:**

Option A: Only add more objects/animations and keep it as one continuous loop；Option B: Build a complete user flow (intro → game → end)

**What changed:** I chose Option B and implemented game states and UI screens.

**Why:** A complete flow makes it feel like an actual designed interaction, not just a sketch. It also improves usability: users understand when the game starts, what the goal is, and when it ends.

### *Verification & Judgement*

**Playtesting:** Repeatedly ran the game in VS Code and checked:

1. Title/button alignment and readability
2. Button click detection and state switching

3. Whether coins count correctly and the end triggers only when intended
4. Whether “Restart” resets counters/objects and “End” returns to intro + resets intro timer

**Criteria check:** Confirmed the redesign still fits Side Quest constraints (using p5.js and building on example code while adding original design decisions).

### *Limitations, Dead Ends, or Open Questions*

**Misinterpretation from GenAI:** Sometimes layout feedback like “more centered” wasn’t implemented correctly until I pasted the specific code block and gave exact coordinates to change.

**Scope constraint:** I prioritized a complete game loop and clear UX over advanced polish (sound effects, difficulty progression, animations for UI transitions).

**Open question (future improvement):** Add clearer feedback to the user during gameplay (e.g., a small “coins remaining” indicator that stands out more, or end-screen stats like time taken / panic average).

## Appendix

Please include a full transcript of your conversation with the GenAI. GenAI transcripts are used for transparency and verification only and are not assessed for writing quality or completeness.

**Note**: If GenAI is used in another language, you must include both the original language and a translated transcript in the appendix

**Prompt 1: “**instruction: Redesign the blob’s movement and environment to express a specific emotion (e.g., joy, frustration, panic). Bonus: Add a “mischief” mechanic, your blob steals or bumps objects oh a small map
The code:
// Object representing a soft animated blob let blob = { // Position of the blob (centre of the shape) x: 240, y: 160, // centre of the canvas // Base size and shape resolution r: 28, // Base radius of the blob points: 48, // Number of vertices around the circle (higher = smoother) // Shape deformation settings wobble: 8, // Maximum amount the edge can move in or out

wobbleFreq: 0.8, // Controls how lumpy or smooth the blob looks // Time values for animation t: 0, // Time input for noise() tSpeed: 0.01, // How fast the blob "breathes" }; function setup() { createCanvas(480, 320); noStroke(); // Text settings for on-screen instructions textFont("sans-serif"); textSize(14); } function draw() { background(240); // --- Animate over time --- // Increment time so noise() changes smoothly every frame blob.t += blob.tSpeed; // --- Draw the blob --- // We draw a circle made of many points, // then push each point in or out using Perlin noise fill(20, 120, 255); beginShape(); // Loop once around the circle for (let i = 0; i < blob.points; i++) { // Angle around the circle (0 → TAU) const a = (i / blob.points) * TAU; // Sample Perlin noise using: // - direction (cos/sin of angle) // - time (blob.t) for animation const n = noise( cos(a) * blob.wobbleFreq + 100, sin(a) * blob.wobbleFreq + 100, blob.t, ); // Convert noise value (0–1) into a radius offset const r = blob.r + map(n, 0, 1, -blob.wobble, blob.wobble); // Convert polar coordinates (angle + radius) // into screen coordinates (x, y) vertex(blob.x + cos(a) * r, blob.y + sin(a) * r); } // Close the shape to form a solid blob endShape(CLOSE); // --- On-screen tip for experimentation --- fill(0); text("Blob breathing via noise(). Try wobble and tSpeed.", 10, 18); }”

**Prompt 2: “** change the blob's face expression once it is dMouse=<120, to a sad face. same eyes, but the mouth needs to be upside down”

**Prompt 3: “**for each square, make it randomly pops tentacle, make it pop tentacle every 5 second, but not each square at the same time. and if the blob touches tentacle, or <30, panic mode starts. “

**Prompt 4:** “can u make a 5 second begin scene, where it includes. a white background with the blue blob design on the right side(when the mouse touches it or gets closer it panics) BUT it does not need to move, it is just an opening. with the title: welcome to PANIC BLOB. and under the title, there is a rounded corner rectangle that says game start. Now if only the user presses the game start, the game starts.”

**Prompt 5:** “can u make a 5 second begin scene, where it includes. a white background with the blue blob design on the right side(when the mouse touches it or gets closer it panics) BUT it does not need to move, it just a opening. with the title: welcome to PANIC BLOB. and under the title, there is a rounded corner rectangle says game start. now if only user press the game start, the game starts.”