

Process & Decision Documentation

Project/Assignment Decisions:

For Side Quest #3, I made small visual and narrative changes to the existing multi-screen game, such as updating colours, titles, button text, and on-screen messages to better demonstrate state changes and branching outcomes.

GenAI Documentation

Date Used: January 30, 2026

Tool Disclosure: ChatGPT 5.2

Purpose of Use: GenAI was used for wording support and to suggest small visual and narrative code changes.

Summary of Interaction: The tool suggested minor edits such as colour changes, text updates, and simple narrative framing to better demonstrate screen states and branching outcomes.

Human Decision Point(s): I selected only a few small changes from the suggestions and adjusted them to fit the existing code structure and assignment requirements.

Integrity & Verification Note: All suggestions were reviewed to ensure they aligned with course concepts, used correct p5.js syntax, and did not change logic of the assignment.

Scope of GenAI Use: GenAI did not write the full project or core game logic; the original structure, screen setup, and functionality were already provided and maintained.

Limitations or Misfires: The tool could not fully account for the course context or grading rubric, so suggestions needed to be simplified and selectively applied.

Summary of Process (Human + Tool)

I started with the provided multi-screen p5.js example and tested it to understand how screen states and input routing worked. I then used GenAI to help brainstorm small, appropriate changes and improve wording for on-screen text. After reviewing the suggestions, I manually implemented only minor visual and narrative edits, tested the game after each change, and adjusted details to ensure the code still functioned correctly.

Decision Points & Trade-offs

One key decision was to make only small visual and text-based changes instead of adding new mechanics or screens. GenAI suggested adjusting the positioning of some on-screen text to change the layout, but I chose to keep the original positioning to maintain visual consistency and avoid disrupting the existing screen hierarchy.

Verification & Judgement

I evaluated my changes by repeatedly running the sketch to confirm that screen transitions, buttons, and keyboard controls still worked as expected.

Limitations, Dead Ends, or Open Questions

The project was intentionally kept simple, which limited the depth of the narrative and player choice.

Appendix

Prompt:

Build an interactive story that unfolds through multiple game states and files, branching like a small decision tree. this is the start code // NOTE: Do NOT add setup() or draw() in this file ... [full start, game, and lose code shared]

Can you just add some things to it and show me where you add it, something like adding a different title and colour to start screen?

Output:

1) Start Screen:

- Changed background colour to a darker blue (background(120, 180, 200)).
- Updated title text to "One Choice Away" instead of "Win or Lose".
- Added a subtitle: "Every click decides your fate".
- Highlighted exactly where in the code these changes were added.

2) Game Screen:

- Updated button label to "TAKE THE RISK" to reflect decision-based gameplay.

- Added a short story prompt above the button: "You find a mysterious door. Do you open it?"
- Suggested small visual feedback for hover states remained as-is.

3) Lose Screen:

- Adjusted background colour to a darker red for visual impact.
- Changed main message text to: "The door was a trap."
- Added consequence text: "Your choice led to failure."
- Maintained existing click and key logic for returning to start.

1 START SCREEN — small visual + title change

What we'll add

- New **story title**
- New **background colour**
- One extra **subtitle line** (sets up the story)

Changes in **start.js**

```
function drawStart() {
  // ⚡ CHANGED: background colour
  background(120, 180, 200); // darker blue tone

  // ---- Title text ----
  fill(20, 40, 60);
  textSize(50);
  textAlign(CENTER, CENTER);

  // ⚡ CHANGED: title text
  text("One Choice Away", width / 2, 160);
```

```
// ● ADDED: subtitle for story context
textSize(20);
text("Every click decides your fate", width / 2, 215);

💡 Why this counts:
```

You're showing **visual state change** and **narrative setup** before gameplay even starts.

2 GAME SCREEN — add a *decision moment*

Right now the game is just random. Let's add **one small choice** that affects the outcome.

✓ What we'll add

- A **story prompt**
- Change button text to feel like a decision

🔧 Changes in `game.js`

Change button label

```
const gameBtn = {
  x: 400,
  y: 550,
  w: 260,
  h: 90,
  label: "TAKE THE RISK", // ○ CHANGED
};
```

Add story text in `drawGame()`

```
// ● ADDED: story prompt
textSize(22);
text(
  "You find a mysterious door.\nDo you open it?", 
  width / 2,
  260,
);
```

💡 Why this counts:

This turns the button into a **narrative choice**, not just UI.

3 LOSE SCREEN — add emotional feedback

✓ What we'll add

- A story-based lose message
- Slight colour tweak

🔧 Changes in `lose.js`

```
function drawLose() {  
    // 🟥 CHANGED: darker red background  
    background(220, 140, 140);  
  
    fill(30);  
    textAlign(CENTER, CENTER);  
  
    // 🟥 CHANGED: story-based message  
    textSize(38);  
    text("The door was a trap.", width / 2, 280);  
  
    // 🟢 ADDED: consequence text  
    textSize(20);  
    text("Your choice led to failure.", width / 2, 330);  
  
    text("Click or press R to return to Start.", width / 2,  
380);  
}
```

💡 Why this counts:

This reinforces the **branching outcome** of the decision made on the game screen.