Process & Decision Documentation

Project/Assignment Decisions

I learned how to use GenAI to brainstorm ideas and help build starting code for the multi-screen game in Side Quest 3.

GenAI Documentation:

Date Used: Feb 2, 2026

Tool Disclosure: ChatGPT 5.2

Purpose of Use: I used ChatGPT 5.2 to brainstorm, process, and structure my ideas to generate a starter code that helps support early ideation.

Summary of Interaction: The tool helped suggest different ideas for a multi-screen game that aligned with the assignment. It also outlined a basic structure with multiple game states and provided start code examples to get things going.

Human Decision Point(s): I made the final decisions for the game concept and theme, the visual design choices regarding the game, and which code suggestions to actually implement and rework that would be appropriate to the assignment and what we have learned in class thus far.

Integrity & Verification Note: I checked GenAI for accuracy, bias, and made sure it was appropriate to fit for this assignment. I made sure the output was aligning to the class materials and verified it by rereading the assignment instructions and comparing them to GenAI suggestions. Any content that was beyond what we learned in class was removed and replaced.

Scope of GenAI Use: The scope of using GenAI was for the ideation process, organization, and early scaffolding.

Limitations or Misfires: The tool is good for storing information and re visiting it. However, it over-generalized ideas, suggested certain features that were beyond the course material, and misunderstood specific constraints.

Summary of Process (Human + Tool)

- Initial ideation between drafts or versions
- Conversing with ChatPT 5.2
- Testing and revision

- Debugging and restricting code by myself
- Moments of rework: Using ChatGPT 5.2 too much to ideate the starter code led to extra revision of my work.

Decision Points & Trade-offs

- Options Considered: I considered implementing additional mechanics or branching logic, but chose a single meaningful choice structure to better align with the learning goals.
- What Changed: The early prompts given gave too overly broad ideas. Through iteration, I learned to be specific in the constraints and received more relevant output.

Verification & Judgement

- Rereading assignment criteria
- Testing the iterations manually
- Comparing code structure to example code
- Re-reading assignment criteria
- Ethical, representational, accessible, or global considerations
    - Treated as a support assistant for debugging and ideating, rather than taking full ownership.

Limitations, Dead Ends, or Open Questions

- Always be specific when sending in the prompts so it doesn't misunderstand

Appendix

My prompt:

I need help brainstorming a simple interactive game idea using multiple screens, but I want to stay within what we learned in class.

GenAI:

The Last Light: The lights are about to shut off. Decide whether to stay and finish your work or leave early.

After Hours: You're locked inside a campus building after closing. Choose whether to explore quietly or try to force an exit.

The Forgotten Desk: You notice something left behind on a desk. Choosing whether to look inside or ignore it changes the outcome.

Silent Library: You're studying in a library with strict quiet rules. One action risks drawing attention.

The Study Room: You’re alone in a quiet study room late at night. One object (drawer, bag, laptop, notebook) presents a single choice that leads to a win or lose outcome.

My prompt:

I want the game to use the existing files only. No extra JavaScript files and no bonus stats. How should the screens connect?

GenAI:

You can keep the same file structure provided in the starter code. Use the shared currentScreen variable to move between "start", "game", "win", and "lose". All transitions can be handled by changing the value of currentScreen inside mouse or key event handlers.

My prompt:

Can you show a very simple example of how a button in game.js could send the player to either a win or lose screen?

GenAI:

In gameMousePressed(), you can check whether the mouse is hovering over a button and then set currentScreen accordingly. For example, if the button is clicked, you could write:

```
if (isHover(gameBtn)) {
  currentScreen = "win";
}
```

My prompt:

I want the visuals across screens to feel consistent. What are some simple visual ideas I can use without advanced techniques?

GenAI:

You can rely on basic shapes and color to establish mood. A dark background with lighter text can suggest a quiet or serious environment. Soft glow effects can be created using semi-transparent ellipses. Objects in the environment, like a desk or lamp, can be drawn using rectangles and circles so they remain simple and readable.