Process & Decision Documentation

In Side Quest #3, I used ChatGPT-5.2 to help me build an interactive story that unfolds through multiple game states, branching like a small decision tree. The original code was a simple chance game that randomly allowed the player to win or lose. There was a start page, and once you clicked into it, there was a chance to win or lose.

I put this code into ChatGPT and asked it to build onto it while keeping the gameplay similar. I began this process by sharing the code and prompting it to “build an interactive story that unfolds through multiple game states and files, branching like a small decision tree.” I also wanted to include the bonus feature, so I prompted it to display three hearts representing the player’s health. The code began having issues, so I repeatedly asked the AI to retype it while ensuring the same template was used.

The game had many issues that I noticed, such as the player having only two hearts instead of three and never being able to lose (a 100% chance of winning). I also wanted the game to be more descriptive. Instead of simply saying “you win” or “you lose,” I wanted the story to take place in a forest. As a result, I asked the bot to add descriptions like “you have successfully escaped the forest.” I also wanted to include a set number of rounds (e.g., three rounds total), so that once the player reached three random choices, they would receive either a winning or losing ending.

I then realized that even though the game used random chances to lead toward a win or loss, the events did not align logically with losing a life. For example, “turning around and leaving the forest” always resulted in losing a heart. Due to this, I decided to be more creative by adding two options for each scenario. This allowed bad decisions to branch toward losing and good decisions to branch toward winning.

If the player lost a life from choosing a bad option, they could regain it (up to a maximum of three hearts) by choosing a good option afterward. For this system to work, I had to prompt ChatGPT to remove the fixed number of rounds. Instead, the game would end automatically once the player lost all three hearts. However, the player could continue for more than three rounds if they kept making good choices and regaining health.

Overall, I found that ChatGPT does not fully understand what works or does not work within code on its own, meaning it cannot produce a complete game without guidance, even with detailed prompts. AI requires constant training, trial and error, and development to meet expectations. It does not recognize issues in the code unless those problems are explicitly pointed out and it is prompted to fix them.

Project/Assignment Decisions

Throughout this side quest, I learned how to iterate on and refine a simple starter code through trial and error with AI, rather than expecting a perfect result from the start. I developed a greater understanding of how to analyze AI-generated code and make design decisions to improve gameplay. I also learned how to better use ChatGPT as a support tool, guiding it with specific prompts and corrections in order to achieve a desired result.

GenAI Documentation

Date Used: January 29th, 2026

Tool Disclosure: ChatGPT-5.2

Purpose of Use: I used ChatGPT as a learning and support tool to change a simple code/game into one that has more decisions and creativity.

Human Decision Point(s): I gave ChatGPT prompts to modify the code in order to keep code that worked and got rid of code that did not work.

Summary of Interaction: ChatGPT was used as a collaborative support tool to expand an existing game prototype. I provided the original code and prompted the AI to build upon it by adding branching game states, descriptive storytelling, and a health system represented by hearts.

Integrity & Verification Note: I opened a live demo in order to see if the code/game works. I constantly checked it whenever I updated the code and changed what was necessary.

Limitations, Dead-Ends, and Misfires: After asking ChatGPT to change another error in the code, it would be closer to forgetting the original code. This made it difficult as it would only be able to focus on one aspect at a time.

Decision Points & Tradeoffs

- **Random chance vs. player choice**: Deciding whether outcomes should be random or driven by player decisions.
- **Fixed rounds vs. health-based progression**: Choosing between a set number of rounds or ending the game based on the player's remaining hearts.
- **Simplicity vs. depth**: Deciding how complex the branching story should be while keeping the code manageable.

Appendix

Link to chat: ChatGPT Appendix