Process & Decision Documentation

Project/Assignment Decisions

Side Quests and A4 (Individual Work)

One significant decision I made was to design and implement custom logic for the moving yellow walls rather than treating them as simple timed obstacles. Instead of only making the walls impassable when they appeared, I added additional logic so that if a wall reappears while overlapping the player, it “kills” the player and restarts the level. This decision increased the importance of timing and positioning, making the mechanic more meaningful and consistent with player expectations. It also shifted the game from a basic navigation task to a system where player mistakes produce clear consequences, reinforcing learning through feedback and repetition.

Goal of Work Session

The goal of this work session was to redesign the example maze code into a more complete and playable game while learning how to use arrays and JSON data to generate levels and scale mechanics. In addition to expanding the map structure through data-driven levels, I focused on refining gameplay mechanics, improving visual feedback, and resolving usability issues discovered through repeated self-playtesting. This included debugging logic errors, improving layout and readability, and adding systems such as scoring, tutorials, and an ending screen to help players better understand the rules and evaluate their performance.

Tools, Resources, or Inputs Used

- GenAI tool: ChatGPT 5.2 (used for debugging, logic restructuring, and design reasoning)
- Starter code: Example maze code provided by Dr. Karen Cochrane and David Han
- Prior drafts or code
- Self-playtesting: Repeated playthroughs to identify bugs, confusion points, and pacing issues
- Browser tools: VS Code + Live Server for testing and iteration

GenAI Documentation

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

Date Used: Feb 8 - 9, 2026

Tool Disclosure: ChatGPT 5.2

Purpose of Use:
GenAI was used to support debugging and restructuring gameplay logic, refine UI and interaction design decisions, and summarize iteration points discovered through self-playtesting. It was also used to help articulate design decisions, process steps, and player-facing explanations for the README file and the process document, including the game description and interaction instructions.

Summary of Interaction:
GenAI assisted in explaining why certain bugs or layout issues occurred, suggesting alternative logic structures, and helping translate design intentions into working code patterns. It also helped summarize key action points and iterations into clear written documentation, making the development process and design rationale easier to communicate in the process document and README.

Human Decision Point(s):
All final decisions regarding mechanics, difficulty progression, visual feedback, and documentation framing were made by me. GenAI outputs were frequently modified, redirected, or simplified to better align with the actual gameplay experience, course concepts, and assignment requirements.

Integrity & Verification Note:
All GenAI-supported suggestions were tested through direct implementation and repeated self-playtesting. Written summaries were reviewed to ensure they accurately reflected actions I actually took and did not introduce unperformed steps or unsupported claims.

Scope of GenAI Use:
GenAI did not define the game concept, level structure, scoring logic, tutorial content, or final design direction. It did not independently write the full assignment or documentation without revision. All gameplay mechanics, visual decisions, and final written content were curated and approved by me.

Limitations or Misfires:
GenAI had difficulty understanding precise visual requirements, particularly when spatial relationships mattered. For example, when placing and rotating triangular shapes to represent keys and keyholes, GenAI could not accurately interpret my visual intent,

requiring me to manually calculate positions and adjust the shapes myself. In some cases, GenAI modified one triangle while leaving the corresponding triangle unchanged, which broke visual consistency and had to be corrected manually.

GenAI also struggled to account for player experience and game feel when suggesting gameplay logic. For instance, initial logic for the key and locked red wall did not include responsive feedback after the key was collected, and suggested mechanics lacked clear consequences for failure, such as punishment after being hit by moving yellow walls. These issues required human judgment to redesign interactions so they were readable, responsive, and challenging. Additionally, when implementing the scoring system, GenAI did not account for clearing or resetting score data between levels, which led to incorrect carryover and had to be resolved through manual logic adjustments.

Summary of Process (Human + Tool)

I began with the provided example code and gradually extended it through repeated cycles of modification, testing, and revision. As new mechanics were added, I frequently encountered layout, logic, and feedback issues that required reworking earlier decisions. GenAI was used as a support tool during this process, but all changes were iteratively tested and adjusted based on how the game felt during play. Much of the process involved identifying moments where the game failed to clearly communicate rules and then redesigning mechanics or visuals to address those gaps.

Decision Points & Trade-offs

One key decision was to add a scoring system tied to collectible green dots after introducing additional moving walls. While the moving walls increased difficulty, they initially discouraged exploration. Adding score-based incentives reframed risk-taking as rewarding rather than punishing. Another decision was to redesign the key–door interaction: instead of keeping a static symbol, I chose to add visual linkage (shared triangle shape) and a fade-out animation to clearly signal successful unlocking, prioritizing player understanding over mechanical simplicity.

Verification & Judgement

I evaluated each change through repeated self-playtesting, paying close attention to moments of confusion, visual ambiguity, or frustration during gameplay. In addition to assessing player experience, I reviewed my code structure to ensure that the redesign

meaningfully used arrays and JSON data to define levels and support expanding mechanics rather than relying on hardcoded values. I also revisited the assignment criteria to confirm that the added systems remained within scope while demonstrating an improved understanding of data-driven design, interaction feedback, and progression discussed in class.

Examples:

- Playtesting
- Comparison with course concepts
- Re-reading assignment criteria

Limitations, Dead Ends, or Open Questions

Some UI layout issues—particularly involving DOM elements layered over the canvas—required multiple iterations to resolve and highlighted limitations of mixing p5.js drawing with HTML elements. Due to time constraints, more complex mechanics (such as additional levels or advanced enemy behaviour) were not pursued and remain potential areas for future development.

Appendix

A comprehensive appendix documenting the use of Generative AI during the development process is available via the link below.

https://chatgpt.com/share/69898a91-5e1c-8005-a551-6ab57e9c9857