

# Process & Decision Documentation

## Side Quests and A4 (Individual Work)

I added a second level, accessible when pressing “N” on the keyboard. I made the blob jump amount higher for a “high jump” effect and changed the colours of the background, platforms, and blob to visually signify that it’s a new level.

## Entry Header

Name: Ellis Byun

Primary responsibility for this work: Editing code for side quest week 4

### *Goal of Work Session*

Enabled the next level keypress function and added a second level to the existing JSON based level system.

### Tools, Resources, or Inputs Used

- ChatGPT
- Assignment instructions on LEARN
- Prior code (Week 4 Example 5)

### *GenAI Documentation*

**Date Used:** Tue. Feb. 10, 2026

**Tool Disclosure:** ChatGPT 5.2

**Purpose of Use:** Clarification of programming concepts, identifying what change is needed to create a second level.

**Summary of Interaction:** GenAI helped me understand that the keypress logic for “N” was already implemented, but only one level existed in the file. Because the system cycles through available levels, pressing “N” appeared to do nothing since there was no second level to load. GenAI also explained how the level array worked and how adding a second level entry would allow the existing function to work properly. It also clarified basic concepts like arrays in simple terms so I could understand how the file was organized.

**Human Decision Point(s):** I steered my focus on understanding how the system worked rather than copying large code blocks without comprehension, I decided to add the second level code manually instead of asking GenAI to generate entirely new code,

redirecting the interaction towards explanation rather than full code generation to ensure I understood what I was actually doing.

**Integrity & Verification Note:** To verify the output, I reviewed the existing code and checked that the info it was telling me was accurate to the task at hand. I ensured I understood how the array index changed and could explain the logic independently.

**Scope of GenAI Use:** GenAI did not design the original world system or make any direct changes to the code. It only supported my concept and task clarification.

**Limitations or Misfires:** GenAI initially provided explanations that assumed more programming knowledge than I had, so I had to ask for simpler explanations. It also introduced terminology that I did not initially understand, which required further clarification. Additionally, the tool attempted to give me full code blocks of what I should change, which I had to redirect to explain to me in words instead.

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

Describe what you did, focusing on process rather than outcome. This may include: Iteration between drafts or versions, Discussion with teammates, Testing and revision, Moments of uncertainty, failure, or rework

I began by testing the next level feature and noticed that pressing “N” did not change anything. I initially assumed the keypress function was broken. After reviewing the code, I saw that the keypress logic existed but was unsure how multiple levels were stored. I consulted GenAI to clarify how arrays and JSON data structures worked within this context. After understanding that the system cycles through available levels, I realized that only one level existed in the dataset. I then added a second level and tested the functionality, which worked as intended after this modification.

### *Decision Points & Trade-offs*

The first decision I made was choosing whether to rewrite the whole code for the next level keypress function using GenAI, or using the tool to help me understand how the existing system worked. I chose the latter option in thinking ahead for future side quests, I figured if I take the time to understand now, it would help me later on. Another decision I made (which is similar to the first decision) was asking GenAI to write me replacement code for what I should change/add to it, or just requesting an explanation of what I can add instead. Again, for the same reason, I chose the explanation to ensure comprehension.

## *Verification & Judgement*

I verified the appropriateness of the changes by testing the game directly in the live server and reviewing the assignment instructions to ensure my changes met the requirements.

## 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.

Me:

using the existing code, can you make it so that when I press “n” on the keyboard, there’s a new level?

ChatGPT:

Your code already supports switching levels with the “N” key. The reason nothing happens is likely because there is only one level defined. The function cycles through available levels, so if there’s only one, it reloads the same level.

Me:

oh i didn’t realize that. where are the levels actually stored?

ChatGPT:

The levels are stored in your levels.json file as an array. Each level is one object inside the levels list.

Me:

wait can u like clarify in simple terms what is an array or JSON data? idk if im remembering the terms correctly lol

ChatGPT:

LOL that’s okay. Let’s strip this down **very simply**, no jargon, and tie it directly to **your game**. An array is just a list of things. In your case, the platforms and levels are lists. JSON is a structured way of writing data so your program can read it. Your levels.json file already uses both — you’re just adding items to a list.

Me:

so i don’t need to change the keypress code at all ?

ChatGPT:

Exactly. The keypress logic is already implemented. You only need to add a second level entry to the levels array so the system has something new to load.

If you want, I can:

- add second level entry code to the system
- refactor your existing platform array into a hybrid system
- add randomized obstacles

Just tell me what you need.

Me:

can you explain what i should add instead of writing all the code for me?

ChatGPT:

Yes, of course! You would duplicate the existing level object in the JSON file and modify values like platform positions, start location, or colors. As long as the structure matches, the system will load it automatically.

Me:

ok that makes sense. i'll add the second level and test it

ChatGPT:

That's a good approach. After adding it, press "N" and confirm the level name or layout changes. If it does, the system is working as intended.