Approach:

My initial approach was to focus on building a functional interactable item inventory system. I started by designing inventory slots to equip items of clothing, trying to create a good design the player would be familiarized with. To optimize space and streamline the UI, I decided to integrate the shop UI within the inventory interface. This approach enabled me to neither have to move my inventory items around nor my equipment.

As I continued developing this, my aim was to create a project that would be easily scalable, for example, by adding scriptable objects for items. With this in mind, I shifted my design to incorporate a modular UI shop system. Basically, each NPC would their items, dialogue and head sprite saved in them and when the player interacted with them they would send this information to the shop script. This would then also allow for the NPC to keep in itself their inventory after trading with the player.

Thought Process:

My main focus with this project was creating ease of use and scalability, as well as independence between different systems. Having worked on a big project, I know how imperative it is to write code that is not only easily understandable but also make it easy to be changed.

I began by breaking down the task into manageable components and establishing a clear roadmap for implementation. First by animating the player, then by creating the inventory. Followed by the equipment and shop.

Personal Assessment:

Overall, I am satisfied with my performance during the interview. I demonstrated strong problem-solving skills, adaptability, and creativity in developing the inventory and shop UI system. However, I acknowledge areas for improvement. My approach to equipping the items was not the best, and very hard to scale upon. If I had had more time I would have used some sort of sprite swap feature. But as I'd never used anything like that I was afraid the complications would prove too time consuming for this task. And when I was particularly engaged in coding I tended to forget to commit my tasks to GitHub.

Most of the code was also developed for this project within the 2 days so I'm very satisfied with my performance in terms of time management.