System Explanation:

In developing "Little Sims World," I utilized several key asset packs: the 2D Mega Pack by Brackeys, the Mighty Heroes (Rogue) 2D Fantasy Characters Pack by Leaidan, and the Little Sprout Lands Asset Pack for the UI. The fonts used are Text Mesh Pro and Jacquard12. While sound implementation was not completed, I focused on interactive scripts. The core script is an abstract class used by both the Shopkeeper and Character Customization classes, allowing players to buy items with in-game currency, making them available in the Character Customization menu. I created a single item changer class that cycles through purchased items, updating the main player's clothing and weapon. Players can earn money through a simple dice game, winning by rolling a double six.

Thought Process During the Interview:

My primary goal was to demonstrate proficiency in Unity and asset integration, creating a cohesive and functional game. I focused on core gameplay mechanics to showcase my skills in scripting, UI design, and character customization. Ensuring a seamless player experience, I prioritized making purchased items easily accessible in the Character Customization menu. The dice game added engagement by providing a fun way to earn in-game currency. Balancing functionality with time constraints, I aimed to ensure core features were well-implemented, even if some aspects, like sound and additional minigames, were incomplete.

Personal Assessment of Performance:

Overall, my performance was good. I successfully integrated multiple asset packs and created a functional game with a working UI, character customization, and a mini-game. However, I wish I had more time to refine the store implementation and complete the clothing store, as currently, only the torso part is available. Additionally, adding another room of minigames would enhance gameplay. While my interaction code was decent, it had an error with multiple interactive objects nearby. With more time, I would implement a distance function to resolve this. Despite these limitations, I am proud of my Cinemachine function, coin handler, and use of actions, which were well-executed