

Documentation

The interview task was a functional prototype for a simulation game in the style of "The Sims" and "Stardew Valley". The game allows players to interact with a top-down view of the game world, talking to a shopkeeper and buying/selling outfits with item icons and prices. Equipped outfits are also visible on the player character.

The clothes shop system consists of several classes that work together to provide the necessary features. The player character is controlled by the *CharacterController* class, which allows them to move and interact with the game world. The *EquippingManager* is responsible for managing the player's equipped items and updating the character's appearance accordingly.

The *GameManager* class is a singleton that keeps track of the game state, including the player's inventory and equipped items. Items are represented by the *Item* class, which is a *ScriptableObject* that stores the item's name, icon, price, and other relevant information. The *ItemUI* class is responsible for displaying items in the shop and the player's inventory.

The *PopupManager* class handles the display of pop-up messages, such as confirmation dialogs when buying items. The *ShopManager* class is responsible for managing the shop inventory, including buying and selling items, and updating the UI accordingly. The *UIManager* class handles the overall user interface, including the main menu, shop, and wardrobe screens.

The *WardrobeItemUI* class displays items in the player's wardrobe and allows them to equip or unequip them. The *WardrobeManager* class handles the player's wardrobe inventory and updates the *EquippingManager* accordingly.

During the interview, my thought process was to first understand the requirements and design a system that would meet them. I started by identifying the necessary classes and their responsibilities, and then worked on implementing each feature step by step. I used object-oriented design principles to ensure that each class had a clear responsibility and could be easily modified or extended.

Overall, I think I did well during the interview. I was able to create functional clothes shop with all the required features within the allotted time. I made use of design patterns such as the singleton pattern and the observer pattern to simplify the code and improve its maintainability. I also made sure to write clean, well-organized code with descriptive variable names and comments to make it easy for others to understand and modify.