Interact with Shopkeep

Dialogue Options

The DialogueStuff method basically rotates around the different arrays of dialogue, once it gets to the end of that dialogue, it opens the Menu.

Open the ClothingMenu

Just open the menu

ClothingMenu

The shopping menu is filled at start with buttons

The preview window in the shopping menu is controlled via MenuNavigation script

Get all clothing items from the Resources_Prefabs folder

This allows for modularity. You can put as many items in the folders as you want and the system will dynamically adapt, something important to games with a lot of customization options.

Create buttons out of the data in each one of the prefabs

Instantiate the button, then clone the stats from the prefab onto the GridItem script on the buttons to access later. Also check the CFM for ownership, and change the button sprite accordingly

Fill each of the tabs in the menu with the buttons created

All of the buttons created go under parents called "ItemGrid" which automatically sort them into grids using Unity's UI system.

When a button on the grid is pressed, call SelectItem in MenuNavigation

SelectItem also has logic for changing the customization preview on the right, so that you can see what the character will look like before buying and display the price under the "Cost" text

After selecting an item, you can press "Buy" to call BuySelected in MenuNavigation

This will also display a message (Handled in PlayerCoinCount) letting the player know if they were successful or not

Once an item has been bought, the script then updates ownership of the item in ClothingFileManagement

CFM is an incredibly important script, holding data for each item's ownership and default preview sprites. This is how we communicate between the Shop and the Wardrobe about what items have been bought and which have not. This script also has dummy code (not used) for a file system, which I did not prioritize (so it is not used) but is easily created in the future.

Interact with Wardrobe

No dialogue (the wardrobe sadly cannot talk)

Open the WardrobeMenu

Loads the Prefabs in the Resources folder just like the ShoppingMenu

Creates buttons and assigns them to the grids

Have you already bought this item? Is it a default item?

Yes

No

The Preview works the same as in the ShopMenu

Do nothing (the button will stay in the grid)

Destroy the button (we can't equip something we don't own)

PlayerInteraction

This is the script that triggers when you interact with the Shopkeep or wardrobe. It also handles the Dialogue options

PlayerCoinCount

This is the script that handles the coins that you have, and also handles displaying both positive messages and warnings from anywhere in the game. I know the warnings don't have anything to do with coins but I didn't want to clog up Interaction or movement and I didn't want to make a new script.

OutfitHolder

This script is just used to change the appearance of the character when an item is equipped.

https://drive.google.com/file/d/1zEvqtZT-Dz3SKtZz5puPS2viTLAnHTJY/view?usp=sharing

This is a basic flowchart of the major systems in the game. Interacting with the Shopkeep, interacting with the Wardrobe, and 3 of my player scripts explained in short detail. The link is a direct link to view the flowchart natively.

The system all flows around the MenuNavigation Script. When you open either the WardrobeMenu or the ShopMenu, the Menu Navigation script will fill the appropriate grids with buttons by calling FillItemGrids. These buttons are created out of a template found in "Resources_Prefabs\MenuButtonPrefabs" and filled with information found in the "ClothingPrefabs" folder in the same directory. Each of the clothing prefabs has a ClothingItem script attached, which holds the stats for each of the prefabs, be it name, price, sprite, etc. The data from that ClothingItem script is transferred to the GridItem script on each button created, seamlessly allowing for the designers to simply create new clothing items, dump them into the Prefabs folder, assign stats there and watch as they appear on the shop grid without hassle.

This system is modular, and has the potential to be endlessly expanded upon, something that is important for Sim games with many customization assets. I also designed the idea of the Wardrobe (something not required on the initial assignment) which allows you to view only the items that you have bought before.

All of the sprites (except the UI sprites) were done by me, which is why they look so rushed and shoddy, though I had a fun time doing it.

As for overall performance during the project, I had a lot of fun with this challenge. I am used to prototyping in a very action-oriented sense, so having to think about backend first and build up from that was new and pretty fun as an exercise. I think I could have done better, but that was mostly due to not being able to work on it for most of the day Thursday due to external issues. Still, I put the best work that I could do in the time frame out there, and delivered something that I think is modular, expansive, and solid enough to provide a great basis for something better.

Even if this interview falls through, I still want to thank you for the challenge prototype. Also, feel free to use my code as you see fit. I don't really care if you use it officially in the game.
-Noah Schultz