

I initially programmed a simple sprite swap mechanism as a sort of proof of concept. I set up a ScriptableObject based system where each ScriptableObject is a “category” of clothing (hats, upper body clothing, legs clothing, etc.), containing a list of possible clothes, with position and layer offsets to ensure the sprites appear in the proper locations.

To expand the clothing choices, I added the option of changing the color of each piece in the categories. To achieve this, the base clothing sprites had to be white-ish, as Unity changes sprite colors multiplicatively.

The customization is achieved by parenting the clothing pieces to the character and sorting the layers properly so that they’re always drawn on top of the “naked” character.

Afterwards, I decided on a pixel-based aesthetic and downloaded the assets (<https://mythril-age.itch.io/mythril-age-sprites-volume-1>). I had to modify most of them by hand to use in the customization system. I set up Unity’s Pixel Perfect camera system and went to work on Animation.

I decided to **not** use Unity’s base animation system, as I believe it isn’t adequate for top-down sprite based animation, and also wouldn’t play well with with customizable character system. Instead, I heavily modified a sprite animation system found on Gamasutra

(https://www.gamasutra.com/blogs/JoeStrout/20150807/250646/2D_Animation_Methods_in_Unity.php), so that it would properly animate the layered sprites.

My next step was using Unity’s tile system to build a (very simple) environment for the player to walk around in. I used tiles from Open Game Art (<https://opengameart.org/content/lpc-modified-base-tiles>). This was set up rather quick.

Afterwards, I went to work on a real Inventory for the player, and an UI for both the Inventory and the shop. The shop has a list of items, with descriptions, prices, icons, etc., and this automatically populates each category the player can select in the shop’s UI. When an Item is bought, it is removed from the shop’s inventory and added to the player’s.

The player’s inventory UI is similar to the shop’s UI, but has more slots.

To stay consistent with the aesthetic, I used UI elements from this pack:

<https://bakudas.itch.io/generic-rpg-pack> (slightly edited one), and this font:

<https://www.dafont.com/04b-03.font>

Finally, I added small polishing touches: A button prompt when near the vendor, a money counter, ensuring the UI refreshes when the player closes and opens it and better messages in the shop’s UI.

Things I’d do better given more time:

- Set up an MVC scheme for the UI
- A more open-ended shop/inventory UI system that could adjust to any number of items
- An easier to use Animation system
- Ensure the game can adapt to multiple resolutions

I typically also use Odin Inspector in my projects, which would make both the animation and the customization system easier to use, but since that’s a paid asset and I have to make the project’s source is available on Github, I decided to skip it.