The first step was to acquire all the sprites needed for the game. I went with these ones:

Character and clothes:
https://craftpix.net/product/rpg-character-sprite-sheet-generator/?num=1&count=43&sq=clothes&pos=4

Scenario
https://opengameart.org/content/lpc-modified-base-tiles

Keyboard keys:
https://dreammix.itch.io/keyboard-keys-for-ui

I paid attention to the pixel density of the sprites so all of them look nice without much aesthetic discrepancy (All sprites were configured to be 32 PPU)

After creating the project, i imported some assets to help the development process, all of them created by me:

https://github.com/RotcivOcnarb/RotsLib - A repository full of helper functions and standard base structures to help the workflow

https://github.com/RotcivOcnarb/RotsLibPopups - A popup library that uses the previous helper lib, and Addressables to create a lightweight, simple popup system

https://assetstore.unity.com/packages/tools/gui/textmarkupparser-187825 - A lib that can parse tagged text in a more robust way to use in Dialog Systems

Also, I imported the following Built-in packages from unity:

- Cinemachine
- Input System
- Addressables

The first thing I implemented was the Character controller. I imported all the sprites of the character and the scenario and put up the character running. Used some Motion Blends to make the 4 direction animations work

[Basic character controller commit on github]

Then, I needed to make the clothes to be dynamic and interchangeable within the character sprite, and to animate correctly over the sprites.

Since all the sprites have the same sprite sheet structure, instead of creating animation objects manually for each of the items (that would take me too much time), I coded an editor only script that basically copies the main player animation asset, and changes the texture and sprites to the clothes texture. I do that by searching by the GUID of the clothes asset using the AssetDatabase, and replacing the string text within the raw .asset object of the original with the new one.

Then I can simply create an AnimationOverride Controller, and swap all the original animations with the clothes animations, for each one of the clothes.

Then, for each clothe currently equipped, I can create an overlay object over the sprite that mimics the animator of the character, and put up the clothe animator override.

[Skin renderer commit on github]

The next step was to create the player inventory, also with the currently equipped clothes inventory.

The rendered skin will be based on the clothes that are equipped, but there also should be an overall inventory of clothes that the player owns. They can be changed by accessing the Wardrobe UI.

I needed to create a Scriptable Object for each clothing item, so I could embed more information into it, like price, display name, description, thumbnail, type (head, torso, hair, etc).

I also went with the script based approach to create each SO, so i can do it quickly and not bother to manually create 90+ files. Not only that but if the game scales as to put 200 more clothes, it wont get out of control

For now, i'll give the player some clothes (but not all of them, to check if the inventory is working properly), to later implement the store functionality

[Inventory base commit]

Now for the Wardrobe UI, first i created an wardrobe object that enables a key to be pressed when near it, and for that to open the Inventory UI

For the UI sprites, i went with this one from Craftpix:
https://craftpix.net/product/game-ui-pixel-art/?num=2&count=381&sq=pixel&pos=4

Ok, first problem:
If you equip an item like a hood over a big hair, the hair will stick out of the hood, looking ugly.

To fix that, I noticed that I can do a trick in which I can select the hair object, and activate the "Render Inside Mask" option, and then add a Sprite Mask component into the hood. That way, part of the hair can be visible inside the hood, but not outside it

I embedded some more information into the SOs so i can check which of the clothes needs to be masked in over another clothe

[Wardrobe UI commit]

To do the NPC that sells you new clothes, i can use the already made code to render skin to create the character, since the component is detached from the player code

I took an already made Dialog System that I use in my other games (made by me) and imported into the game. I don't have a public link to it, so I just copy/pasted all the files from another game into this one. (DialogPopup.cs, DialogObject.cs, SingleDialogDrawer.cs) It uses the popup system i imported, so i needed to set it up (create the Loading scene, setup the transition manager and popup manager, as also the Addressables configuration)

Since i was already dealing with popup windows, i just turned everything needed into a Popup to be used by my Popup system (the wardrobe UI, and later, i will make the shop UI as a popup too)

Then, I stumbled upon a bug where I can't select the accessories properly, since I can't unequip them.
I fixed this issue by unequipping the item when i click the slot again, but just for the Accessories

Before creating the Shop UI, i needed to sketch some layouts to structure all the things i needed for it to be available, so, i came up with these:

A tab system that looks like the wardrobe one, in which it will show all clothes available.
For each clothe slot, i need the following info:
- If the player already have it (so i can prevent them from purchasing again)
- If it is currently selected (and to be shown on the player character as a preview)
- The price
These infos are nice to be shown in the slot itself, so it can be easier for the player to take an overall look at all the slots and see what he wants, instead of having to click one by one to see the prices and availability of the item.

Also, wanted to make a new area in the shop that shows more info about the selected clothe, like a bigger sprite picture, the description text, and the button to purchase it (that needs to be disabled if there is not enough money, or if the player already have it)

One thing I should pay attention to, is that the shop shouldn't sell skin tones, as they are not a thing that you usually buy at a shop. All the skins should be available freely to the player at the start. Having to buy skin tones can be perceived as a racist behavior, since you have to PAY to choose the skin tone of your own character

Since the same could be said for hair, I just got this idea where besides only having a clothing shop NPC, I could create another barber shop NPC, which is focused mainly on hair.
The UI can be exactly the same as the shop, but I just put hair options instead of clothes.

[Shop UI and purchases commit]

To create the barber shop, it was quite easy, I just needed to clone the NPC, and create a copy for the ShopUI, just adjusting the Skin Type list to only show Hair objects. The way that the store was structure enables me to create new types of stores with easy adjustments

[Barber Shop commit]

Now, the player has to be able to sell the already owned clothes, and there should be an UI for it. I currently have 2 options, the first one is to embed that functionality in the Shop UI, or I can put it in the Wardrobe UI.
Selling clothes in the shop UI:
        Pros: It makes more sense in the lore, if you want to sell your clothes, you would probably go to a shop to do it
        Cons: it's not practical to be navigating over many shops just to free your inventory, since different skin pieces are sold in different shops.

Selling clothes in the Wardrobe:
        Pros: it's much easier and practical to be able to sell all your stuff into one single place, and since the Wardrobe don't have much UI complexity, there is plenty of space to embed a selling functionality
        Cons: It doesn't really make sense to be able to sell your clothes in your wardrobe, inside your own house, people might miss the point and be lost by not knowing where is the place to sell the clothes

I went with the shop UI, because even though it isn't that practical, it is much more immersive and intuitive, and it's not like all the player experience will be destroyed just because it needs to go to some other place to sell their stuff

To have the items be able to be sold, I just changed the "purchase" button in the UI to "sell" in outfits that you already own, instead of having the text "owned". When the player sells an item, if it is the one currently equipped, it unequips from the player

[Selling equips commit]

Now that everything is working fine, the next step is to polish the scenario so the mechanics can happen outside of that prototype room. My idea was to create a little city that has an outside area as a main hub, from there you can go to your own house (where the wardrobe is), the clothing shop (to buy clothes) and the barber shop (to buy and change hairstyles).

I began searching for sprites and assets that can fulfill my needs, as I have no knowledge in art and drawing, I have a heavy dependance in premade assets, although I can modify current ones to fix some needs.

The searching of the sprites took me too long and I was not finding the assets I needed, like a city tileset, and things themed as I needed, for example the interior and exterior of a clothes shop, and the barber shop.

Because time was running out, and all of the mechanics were already functional, I ended up leaving the project as is, so as to not overwhelm myself into creating a map that I'm not familiar with, and to risk for the time run out, and the project to be half done.

So i focused on testing the final build and to check for any more bugs i may encounter