Prototype Game Design Document

Game Prototype Overview:

The game prototype is a 2D top-down simulator game, where the player plays in a clothes shop, the core loop gameplay for this prototype is that the player can interact with the shopkeeper, buy & sell clothes, put clothes, and choose and change clothes anytime via the clothes inventory

How to Play:

Sprint :(WSAD/ Up-Arrow, Down-Arrow, Left-Arrow, Right Arrow) + left Shift = Walk

Interact: E

Inventory: TAB

Quit Game: Alt+F4

The Assets Used in this Project:

I bought 59.99$ asset for the customizable character, Link

For level Design I used those two assets from Itch.io this for 1.99$ Cloth Shop and Characters, Second Free Shop Level Design

For the UI I used the customizable character UI, Link and this free assets from Itch.io, UI Wooden Controls

Music & Sounds: Shop Music: Skies of Arcadia - Sailor's Island, Atmosphere: France: Large Clothing Department Store, Atmosphere , Freesound.org

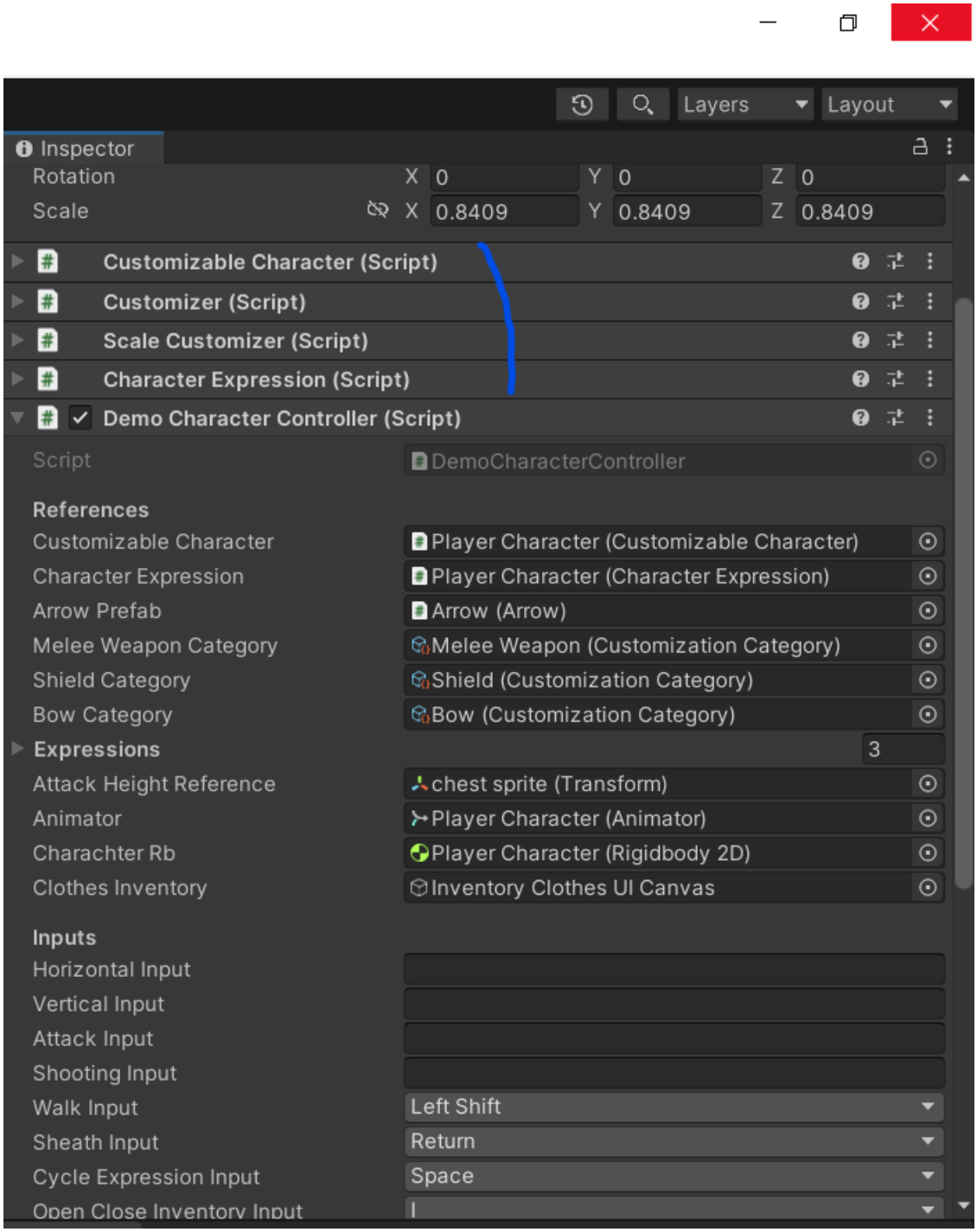
The Scripts that I wrote & the Script not wrote by me on that project:

The Scripts that I wrote:

For the clothes shop system and UI and Inventory I wrote all the code functionality and make the UI

For character movement I use the Character Customizable ***DemoCharacterController.cs,*** but I did modified and edited to use physics, and remove some unnecessary functions and add new inputs

The asset Character Customizable that used for this prototype, come with a cloth like system to edit the appearance and equipment of the character but I didn't use them but they are still assign to the character because they use them for Animation and Locomotion and I didn't remove them, to no brock the animations



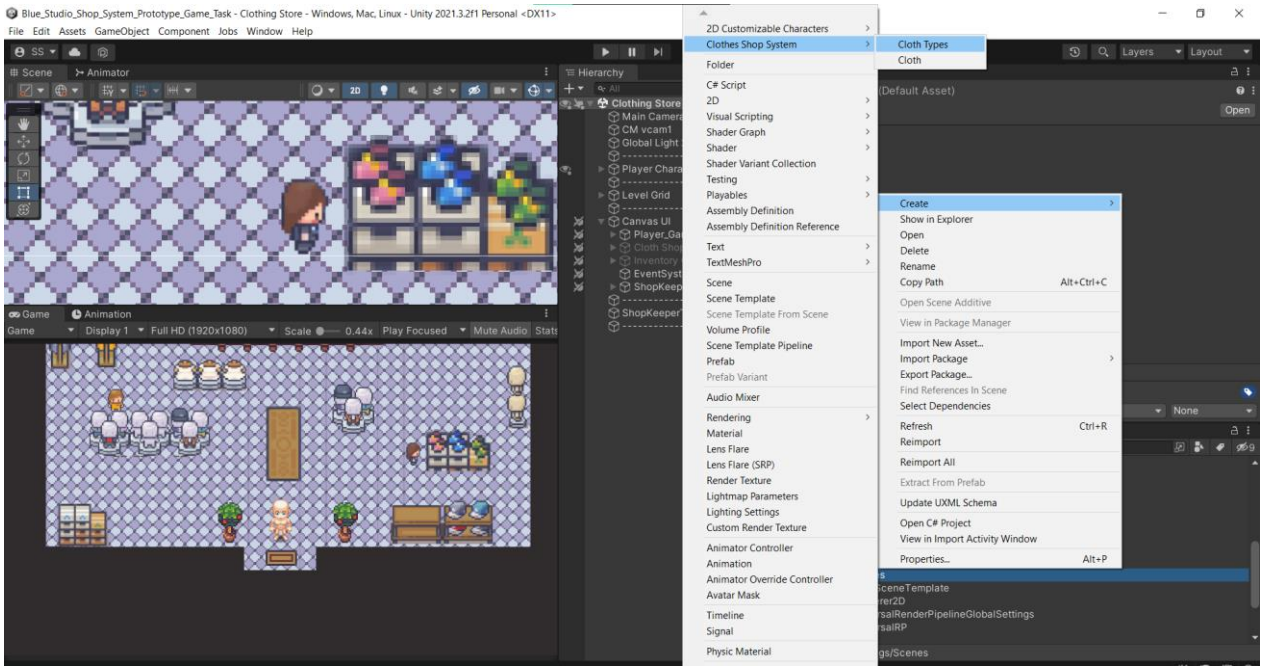
List of script I wrote and Edit:

- ***ClothCardUI.cs***
- ***ClotheSectionManager.cs***

- ***ClothesTypes.cs***
- ***ClothScriptableObject.cs***
- ***ClothSectionButtonUI.cs***
- ***ClothShopManager.cs***
- ***PlayerCurrancySystem.cs***
- ***ClothesInventoryManager.cs***
- ***ShopKeeperTriggerZone.cs***
- ***TriggerZoneBase.cs***
- ***DemoCharacterController.cs***

How Clothes Shop System:

To create a clothes that can the player chose to put you need first to create ***Clothes data container***, you can create it by clicking right-mouse click on assets menu, Create > Clothes Shop System > Clothes Types



The Clothes Types is a scriptable object data container contains, a list of Clothes Types that the player can buy and equip the list is type ***ClothType.cs*** class

```
[System.Serializable]
1 reference
public class ClothType
{
    public ClothTypeEnum ClothTypeValue;

    [Tooltip("List of clothes of that cloth type")]
    public List<ClothScriptableObject> ClothesList;
    public ClothCategory ClothCategory;


}
```

The ***ClothType class*** define what the cloth type is what the cloth category of that type of cloth is, and List of ***ClothScriptableObject.cs***

```
[System.Serializable]
1 reference
public class Cloth
{

    public ClothesTypes clothesType;

    [Range(0f, 99999f)]
    public float ClothPrice;

    [Tooltip("The Clothe Sprites Values ((Index 0)Front, (Index 1)Back, (Index 2)Side)) ")]
    public List<Sprite> ClothSprites;


}

[CreateAssetMenu(fileName = "Cloth", menuName = "Clothes Shop System/Cloth")]
Unity Script | 1 reference
public class ClothScriptableObject : ScriptableObject
{

    public List<Cloth> clothes;

}
```

The ***ClothScriptableObject.cs class*** is responsible of storing the list of sprites of that cloth type that will be assign to the player **Sprite Renders** this list is stored on ***Cloth.cs*** class, this class define the cloth price and a List<Sprites> sprites, in order Index 0 for Up sprite 1 for Down sprite, 2 for Side Sprite of that cloth, because we have a 2D top-down character, so it has 3 different side Down, Up, Side.

For the **Clothes Types** and **Clothes Category**, there is two type of **Clothes category, Appearance** and **Equipment,**

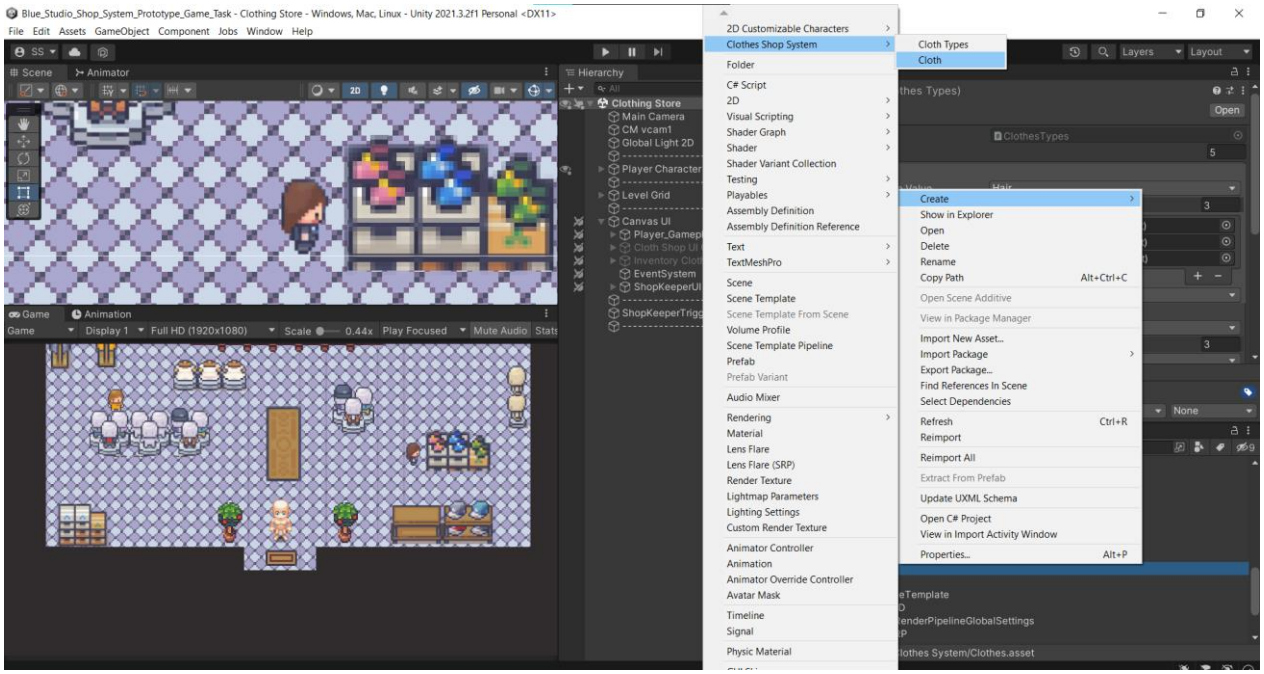
The Appearance category, is the cloth types that effect the appearance of the player, like hair, facial hair

The Equipment, is the clothes like, short, shoes, pants, hats, etc…

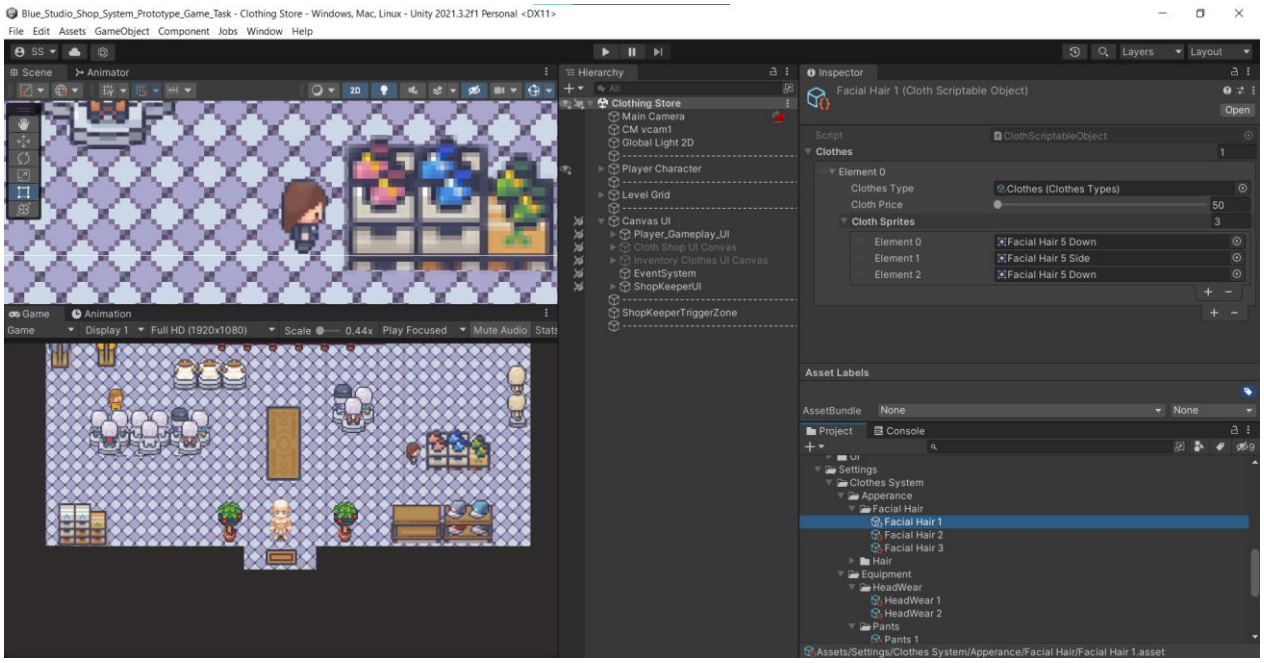
```
8 references
public enum ClothCategory
{
    Apperance,
    Equipment
}

7 references
public enum ClothTypeEnum
{
    Hair,
    FacialHair,
    Ears,
    Eyebrows,
    Eyes,
    Iris,
    Mouth,
    Makeup,
    Headwear,
    Top,
    TopOverlay,
    Belt,
    Pants,
    Handwear,
    Shoes,
    Glasses,
    Shoulders

}
```

The Cloth Types, can be varies depend on the character we have and what the arts provided by the 2D Designer, for the character I used for this prototype, there is 17 type of clothes, we will use the ***ClothTypeEnum*** to check when the player will equip a cloth we will know what type is, and assign it's List of Sprites to the correct Sprites Renders on the player character.

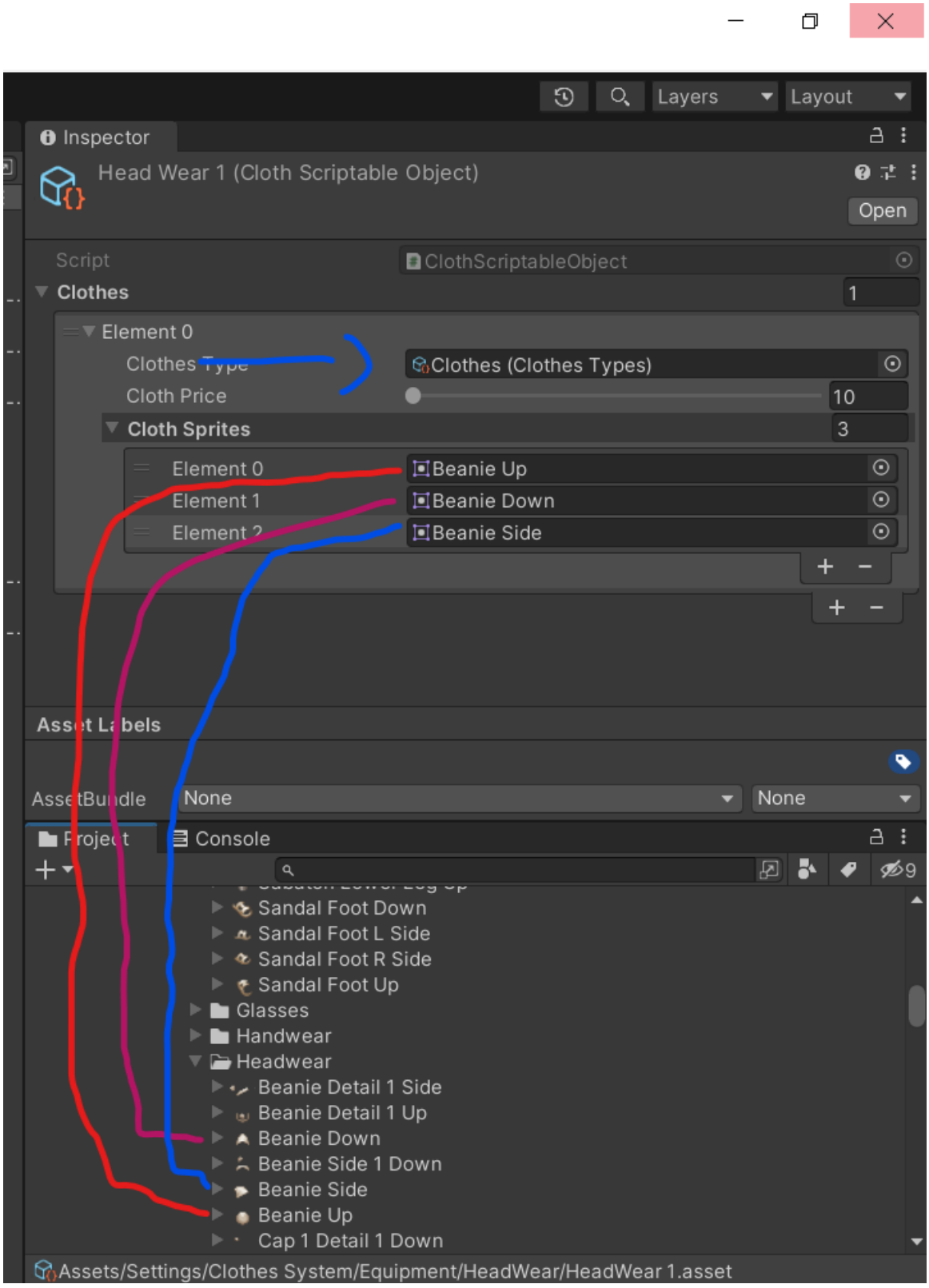
After we create The Clothes Types Data Container, we can create a cloth to assign it to the clothes types' data container, to do so we click right-mouse on assets folder, Create > Clothes Shop System > Cloth



As we mention previously the Cloth, is a Scriptable Object data container, contains the data of that cloth sprites (Up-Down-Side), the clothes types container, and the cloth price



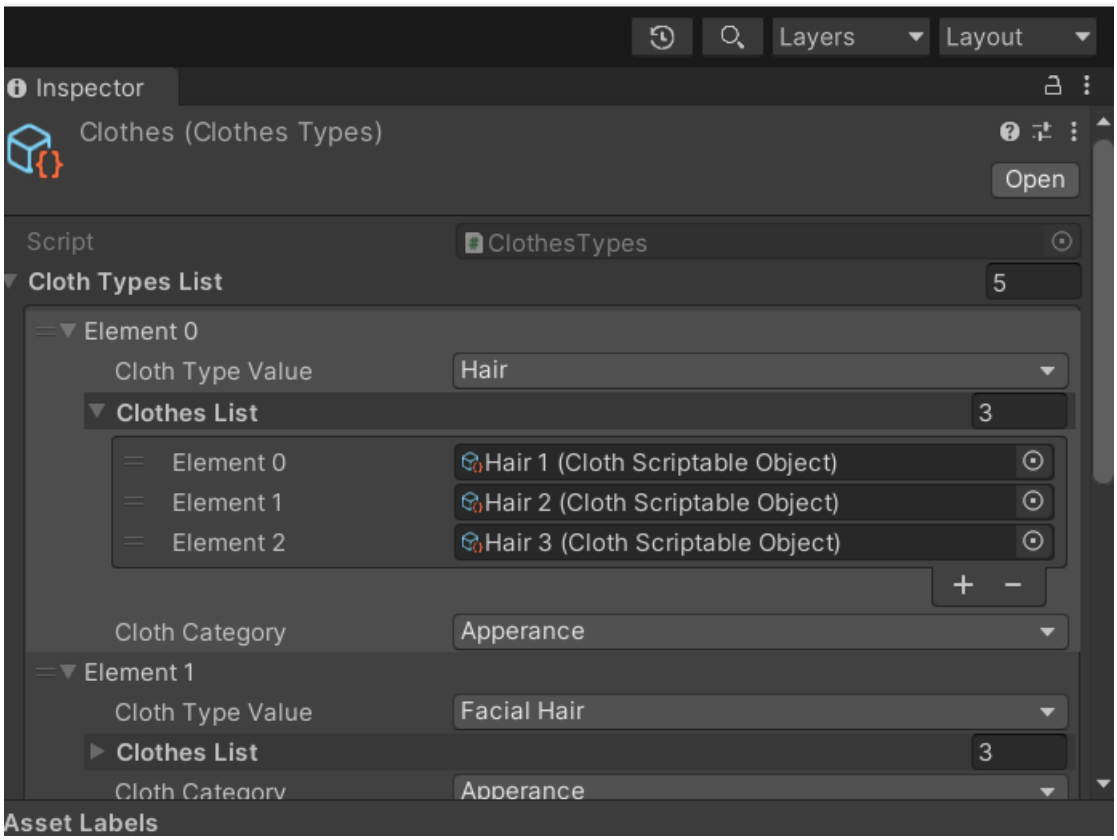
We assign the Clothe Types that we create first on the Clothes Type filed, and we add the Up - Down – Side by order (Index 0 – Index 1 – Index 2), to that cloth, like this example:



In some cases there is some clothes that don't have side or up or down sprites, like facial hair, this can be ignored, and is handled automatically by the script, just

add down or side to the missing part of that cloth, this depend on the character and designs

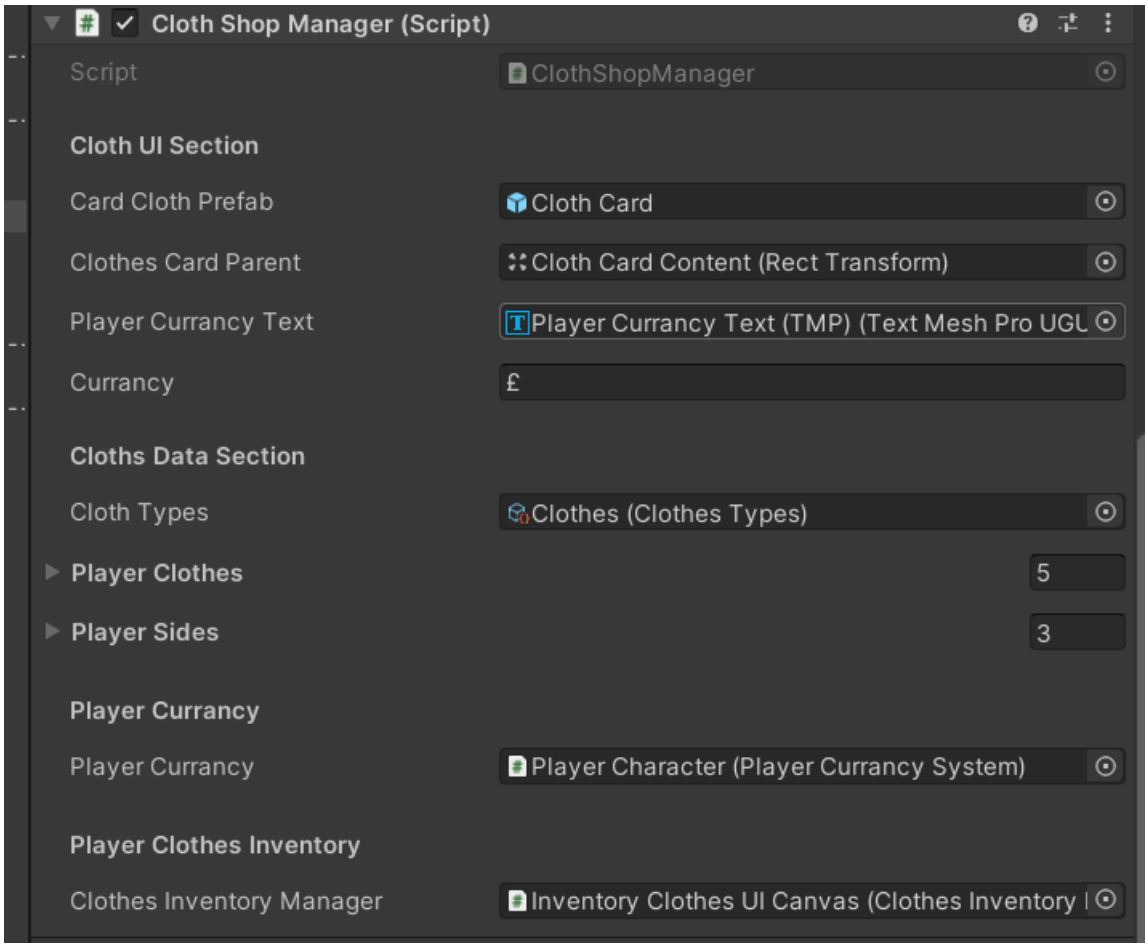
After we create our first cloth, we need to assign it to the clothes type data container



The Clothes Types Data Container, contains a list of Cloth Type, that has cloth Type value, and Cloth Category, and Clothes List is a list of Cloth of that that Type of clothes, like the example above, we created 3 different cloth of type hair, so we add new Cloth Type List to Cloth Types List, the list is of type Hair, and Cloth Category Appearance, and we assign the Clothes that we created To The Clothes List

After our first cloth type is ready and done and can be purchase via the shop store and equip via the player, we just need to handle the purchase and the equip and showing the clothes on the Clothing shop Store UI

And this the ***ClothShopManager.cs*** will come in, the script is responsible of handling the purchase of the cloth, sell, populate the clothes from the Clothes Type data container to the Cloth Shop UI, also getting the player currency from the ***PlayerCurrancyManager.cs*** class that has responsible of storing how much money the player has and adding or removing certain amount from the player currency



The Cloth UI Section

In this section, there is the **Cloth Card prefab,** this card is the holder of the cloth that contains the look of the cloth and price and button for Buying & selling & Equip & Unequipped the cloth

The Clothes Card Parent, is Content parent were the Clothe Cards Will spawn,

The Player Currency Text, is the text that visualize the current currency of the player,

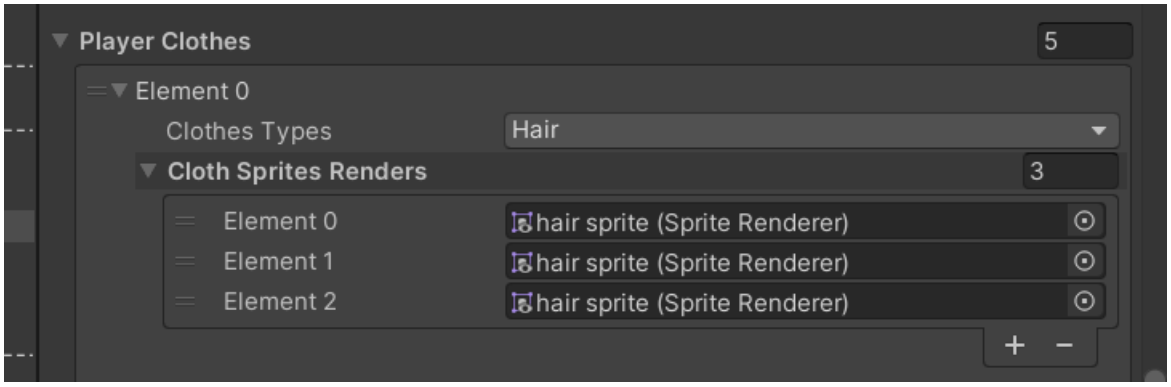
The Currency is the shop currency £, $ etc…

The Cloth Data Section

This section contains all the data necessary Cloth data for **Cloth Shop Manager** it to get the current clothes data available on the store, and use it to populate the Clothes Cards. And assign the values to the cards from the Clothes Types Data Container.

Player Clothes, is a class that store the player clothes sprite render and what type those sprites renders are Hair, Facial Hair etc…, and this type used by the Cloth Shop Manager to assign the Equipped cloth to the correct sprites render on the character, The sprites Renders List follow the same structure as the Cloth sprites list

(0->Up, 1->Down, 2->Side) and if any type don't have a side or up value can be ignored and add any one of the available sprite renders.



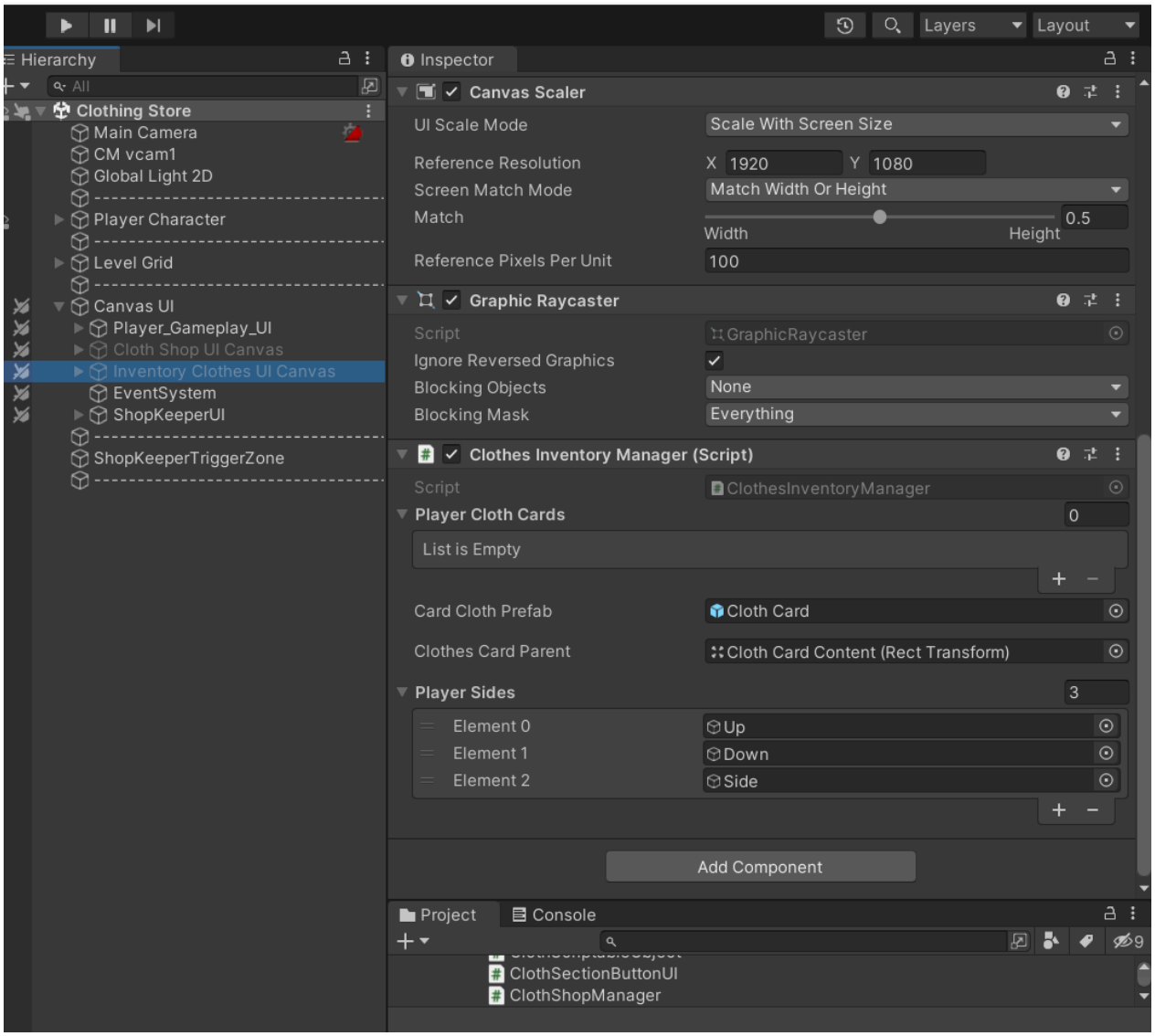

```
[System.Serializable]
1 reference
public class PlayerClothes
{
    public ClothTypeEnum ClothesTypes;
    public List<SpriteRenderer> _clothSpritesRenders = new List<SpriteRenderer>();


}
```

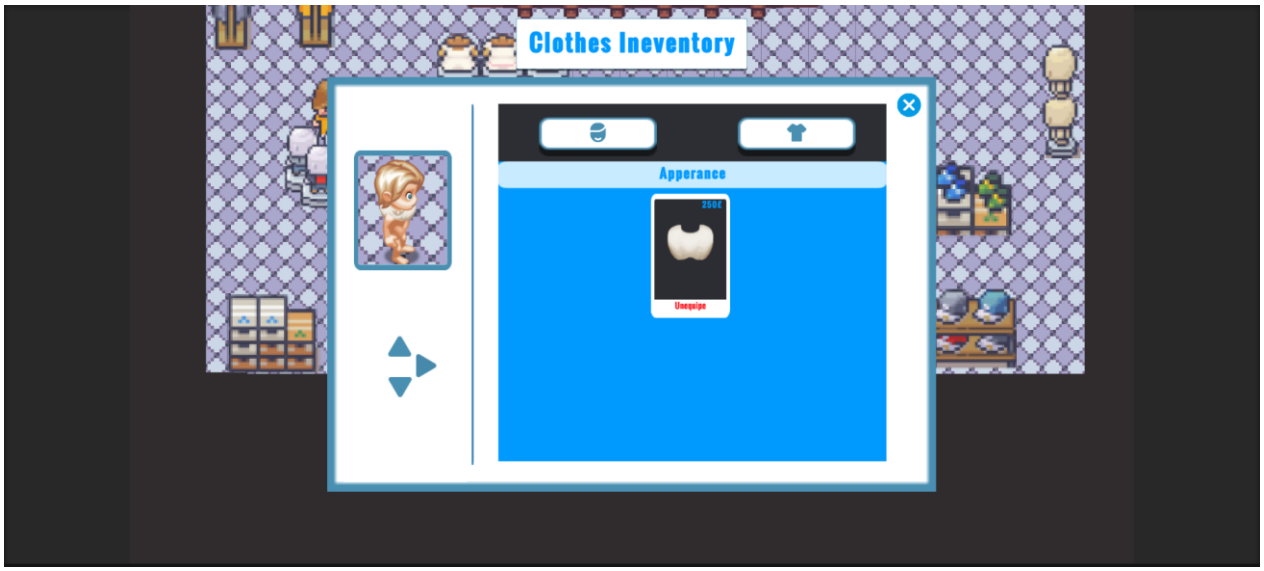
The Player Sides is the Sides of the Player character (Up, Down, Side), is used for rotating the player via the Cloth shop UI

Player Currency, is reference for the ***PlayerCurrancy.cs*** Class is referenced by the Cloth Shop Class to get the player curacy and call the Add or Remove Functions when player Buy or Sell Cloth

Player Inventory, is reference for the ***PlayerInventory.cs,*** this class is used to store the purchased clothes to the inventory or remove it if the player sell it, and also the player can equip or unequipped the clothes only from the inventory.

The **Player Inventory,** use the same structure as the **Clothes Shop Manager,** responsible of showing and storing the purchased clothes and Equip and unequipped the clothes, also rotate the player





My Thoughts about the system:

I think the system is pretty easy to use and reusable can be added to any 2D top-down, and is editor, so we will not need to access to code to add new clothes every time, so 97% of times, the designer can work on that adding new clothes without talking or asking for the developer to add new category or type, there clothes are already categorized and spear to types, that any type of 2D top-down may need.

If the clothes has animations, the system may need some editing to handle also the locomotion of the clothes, but I don't think well be in this case, because the clothes is like placeholders only the character bones and default sprites renders (Arms, legs, head, hips, chest) is animated and make the cloth look animated as well.

Finally I can say the system is out of the box and can be used by beginner or advanced developer or designer as well, just by following those introduction, and the Cloth UI handling can be take it from the Cloth Shop Manager to another class so that can be customizable depend on the UI of the game, and can be a good start for a fully functional system for ***Little Sims*** or any 2d-top down games in the futures

And I think I did good job of creating a reusable Cloth System and shop system for any case on short amount of time, with the limited art assets and hard to find something free or paid that can help make the system, but it was great experience and fun, and I looking forward to meet you and I have some great ideas for the **Little Sims** and maybe future projects! Hope I was as expectation and there is more I can show, and I hope you like the system and I wish for the good! Thanks for this great chance!