Top-Down Game – Description

***All the code used in the project was developed during the interview**

General:

The project was designed to maintaining in mind the separation of concerns, being the main categories:

Player: responsible for Keyboard Input, triggers other functions though interfaces, Character Customization and Animation.

Mechanics: provides the models to logical operations, such as inventory and money manipulation. Acts as the model.

UI: Shop, Inventory, Acts as the view.

Misc: Multiple autonomous scripts to provide common utilities.

INPUT

"WASD" – Move

"E" – Interact

"I" – Open inventory

"Esc" - Exit

Components:

Player:

The player scripts act as a controller, triggering the other components via Interfaces.

Interfaces:

ICollectable: Every loot in the map is an ICollectable.

IInteractable: When the player enters the trigger of an object with a possible action, the interface calls the UI show "Press 'E' To Interact", when pressed it will execute the object internal function.

Items:

Every item in the game is of the type of "Item", including the clothing, this decision was made to universalize the items, meaning that every item can be sold at the

market or looted. This decision weights the common actions of many items, such as equip, consume, and hold. Having a common interface makes easier for future features such as the use of tools (Pickaxe, Axe, Shovel).

All those items are Scriptable objects, and contain a field for an Icon, which is used in the UI.

Shop and Inventory menus:

All menus follow almost the same logic, they start by getting a list of an Items of an inventory. For the buying menu the list is from the available items of the clerk, for Selling/Inventory is the list of items are the items owned by the player.

With that list the code can create a table of items to interact in the UI, the Items in the UI automatically update on changes, and are removed if an item is out of stock.

The table also filters elements by types, making it easier for the player to navigate.

These decisions were made to make it possible to automatically any type of item to the game without additional configuration.

Animations and Clothes

The animations are fully controlled by the animator, with the animator receiving the X and Y values from the Player.

The Clothes are controlled by the script Skin Manager, and it uses Animation Override Controllers to switch the Clothes. This was made to easy the processes of switching sprites when equipping Clothing items.

Assessment

Looking back on the development of this prototype there are some aspects that I personally would like to improve, specifically the Shop management, I my point of view it could be more abstracted, and it could have a strong separation of concerns.