

## Task Assessment

This prototype is a very simple vertical slice of a top down 2D game in which equipable clothes are the main focus. The player is able to move freely, talk to a shopkeeper in order to buy/sell clothes which they can equip from their inventory.

After creating the Git Repo I first started by importing some assets I use on a regular basis while developing projects, most notably DOTween as it makes it easier to Lerp variables and set them independently from Unity's timescale which is useful to display or hide UI elements while the game is paused.

Firstly I focused on creating the player movement by creating a static InputManager class so it can be referenced by any script that will need to read user input, then I created the Clothes class as is the primary element of this task I decided to make it an scriptable object so it's easier to configure it's elements like price, sprite and attachment point (head, torso or legs).

The player Controller was setted in order to have a controllable player avatar. I moved on to create the shop interaction and the inventory system as an scriptableObject, the idea is that both the player and the shop have their own configurable inventories. The shop allows the player and itself to peek into each other's inventory and exchange clothes.

After it was possible to exchange inventories I created the UI for the game and binded all events to the corresponding buttons, to give the game more flavor I created a SoundManager, like the input one is a static class that can be referenced in order to play whatever sounds it's needed.

The last two adjustments I made to the project were to design a quick environment using tilemaps as it's a common and flexible way to create and edit levels and Implemented a save system that allows the game to save both the player and the store inventory so that the purchases and the outfit last equipped are persistent between different game sessions.

Overall I think the game is a solid prototype for a 2D top-down store, I consider that the weakest aspect is the Sound manager, it works but is unsafe since every sound must be searched via strings which a misspell is prone to happen although it's handled so while no sound will play a null reference will not occur.

