

# Blue Gravity Studios Interview Task

By Caio Flávio

“Create a functional clothes shop within a simulation game reminiscent of 'The Sims' and 'Stardew Valley,' featuring a top-down view similar to Stardew Valley. The player character should be capable of walking and interacting with the game world!”

## Concept Phase

The visual concept of the inventory and wardrobe system adhered to was based on the Little Sim World game itself. With a very intuitive and simple UI design, with buttons for categories aligned vertically and a grid of items on the side.

Buttons with larger images make it easier for the player to identify the item without the need for a lot of text.

Thinking about the dynamics of the game, I decided to incorporate the store into the game world, keeping the character alive and in contact with the game world, without the need to create a prefabricated closed environment, remembering MMORPGs.

The theme “chosen” was a medieval world in which the player is a ninja and needs to steal items from chests in caves and negotiate with the wizard.

## Development Phase

Before starting the development itself, I thought about all the possible and necessary features that the game would need to have. Obeying what was mandatory (player, store, wardrobe and interactions) and what was dispensable at the time (enemies, visual effects, art, etc).

Systems:

**Player:** The main character, can move and identify interactable objects in the scene, such as: chests, items and NPCs.

**Shop:** Responsible for purchasing and selling items. At the top, we have two important buttons, the hood meaning the wardrobe and the backpack, the inventory of artifacts stolen from chests by the player.

**Event Controller:** Through this, all game classes communicate, reducing direct dependence between codes.

**Game Controllers:** Class that manages other controllers. Through it we can manage the initialization queue so that there are no dependency errors between controllers.

**Cinemachine:** Confiner was used to delimit the area allowed for camera movement.

A second camera is used that is activated when entering the store, creating a blend with the character's camera.

**Database reader:** the database system used was a .xlsx spreadsheet.

Column structure: **code** | **name** | **type** | **price**.

Despite the existence of ScriptableObjects, I think it is always interesting to learn about new approaches that are not exclusively dependent on the game engine.

**UI animations:** The tool used was DOTween. Facilitating and circumventing possible problems with unnecessary updates in Canvas.

**Other animations:** Chest and NPC animations made by Unity's own animation component.

**Input System:** Used the new input system to control the characters and interactions with elements of the scene.

## Personal Assessment

I believe that this test not only validates the speed of delivery in a short space of time but also the ability to deal with a lot of work and a lot of pressure.

Despite having delivered the finished project before the deadline, this document was sent hours after the 48 hours. It was my decision to rest my mind and be more aware of what is being written here.

Regarding the difficulties encountered in development, I believe that despite having delivered all the required resources, everything can and should be improved as the days go by.

Screenshots:



