Unity Programmer Task - Development Report - Luiz Felippe

Asset Usage

For this project, all art assets were sourced from the Unity Asset Store.

External Code

The only code that was not written during the given time frame was the `GameObjectPool` class. This is a utility class that I previously developed and used in another project.

Inventory System Implementation

The inventory system is structured around a **Scriptable Object** that serves as the inventory container. It includes a serialized list of items that is updated at runtime to track the player's inventory. Any system that needs to modify the inventory (such as adding or removing items) does so through this **Scriptable Object**.

Any interested system (such as inventory slot UI elements) subscribes to an Action event. This allows everyone "interested" on inventory changes to respond automatically whenever changes occur.

Additionally, the runtime inventory list is persistently saved and loaded using a `.json` file. The system writes the inventory state to a file when the game exits and reads from it when the game starts, ensuring that the player’s inventory remains consistent across sessions.

Development Process

The development process was somewhat chaotic due to time constraints. I was only able to start working on the task at 3:00 PM (BRT) on Saturday, with the deadline set for 11:00 AM (BRT) on Sunday. Given this limited window, I had to prioritize rapid implementation over extensive planning.

My initial approach was to define a general structure, choosing to make the inventory and items **Scriptable Objects** to work as described below, as it was suggested by the instructions but also something that I had already done and was familiar with.

Final Thoughts

While I successfully implemented all the core requirements of the task, I acknowledge that the final result does not fully showcase my potential. Given more time, I would have focused on polishing certain aspects, improving code, general structure and organization. Additionally, I would have introduced more gameplay features such as enemies and a proper gameplay loop to make the experience more complete. Nevertheless, I am satisfied with having delivered a functional prototype that meets the basic requirements, considering the tight schedule I had due to extra obligations during the task period.