Trade System

by Paulo André Pimenta Aragão

System Workflow

When the system starts, it will display a main menu for the player, which has three options: play, exit and settings (gear on the upper right corner). The only setting implemented for this prototype is volume configuration. After the player hits the play button, the game will load (asynchronously) the main scene, which is composed by the trader, some coins (total of 10 coins) for performing a trade and some hearts (total of 5 hearts) for health recovery.

In this scene, the player can move freely as a top-down view 2D game, as requested by the interview task. For performing a trade, the player can collect the coins, placed on the bottom of the map, and then talk to the NPC in the middle of the room. All items can be bought and sold, using the price variable. After buying an item, it will be moved to the player's inventory, and can be equipped or unequipped by the player anytime. For equipping the item, the player must open the inventory and click on the desired item. For unequipping an item, the player must click on the equipped item slot. Furthermore, the inventory system supports more than one unit of each item. With that in mind, the inventory works as following:

- If the item to be **added** to inventory **does not exist** in it, a new inventory slot will be created and added to inventory;
- If the item to be **added** to inventory **exists** in it, the amount is increased;
- If the item to be **removed** from inventory has **more than one** unit, the amount is decreased;
- If the item to be **removed** from inventory has **only one** unit, the inventory slot is removed from inventory and destroyed.

Also, there is a portal in this room that leads to a battle arena, where the player must kill enemies to earn coins for buying new items. In this arena, the player must survive 5 waves of enemies to go back to the main scene. These waves get bigger every round, adding the number of the current level of enemies to the next round. For example, if the round number is 3 and the previous number of enemies is 7, this new round will

have 10 enemies to be killed. If the player dies in combat, a game over screen is displayed, allowing the player to choose rather go back to the main menu or exit the game. It is important to observe that this combat system was made only for a more dynamic way for players to get coins and buy items.

Considering that the target platform of this prototype is PC (Windows), the following inputs were used:

- WASD – Moves the player around the map;
- E – Inventory;
- Escape – Pause menu;
- F – Interact;
- Left Mouse Button – Shot a skill.

For this prototype, no pre-written code was used. Also, the free Unity assets used for this task were: Brackeys 2D Mega Pack, Thalea Pixel Font. All the sounds and the item icons were taken from the internet **only for this task, and may not have commercial license for redistribution**.

Thought Process

For develop this task, I had the following line of thoughts, which guided me through the whole process:

1. Requirements elicitation – First I pointed out which were the requirements for this task, for example, trade system, inventory system, etc.;
2. Character movement – After getting all the requirements, I started to think about the character movement, considering the top-down view. With that, the player controller and character controller 2D where needed, which manages all the player aspects;
3. Item and inventory system – With the character moving, it was time to define the item class, considering its attributes. It is important to note that the item is a scriptable object, which facilitates adding new items to the game. After that, the inventory system was developed, considering adding, removing and using the items requirements;
4. Trade system – After the item and inventory system were developed, the trade system implementation began. For this, I thought that the trade should only be made if the player has enough coins for the item. Although it works for buying, there are no coin restriction for selling an item;

5. Combat – For making this prototyping funnier than just giving coins to the player and trading items with the shopkeeper, I had the idea of adding a battle arena, where the player can kill enemies and earn some coins. So after completing the trade system development, all the combat aspects were implemented;
6. Item attributes – Considering the battle arena, I decided to add some attribute buffs to the player if an item is equipped. These buffs are limited to additional health (for armor and helmet) and additional damage (for weapon);
7. Menus – After the item buffs, it was time to create both main and pause menu, to facilitate the game navigation;
8. Interface – At last, but not least, all the interfaces were properly improved, using the free assets cited above.

Personal Opinion

I am very satisfied with this prototype, since I was able to develop a workful inventory and trade system that can be used for items that have and have not attribute buffs. Also, I'm very happy that I was able to implement a combat arena for earning coins, which makes the prototype funnier to use. I also know that the system has an issue with audio, that gets higher or lower randomly during gameplay, but I'm currently working on it. Furthermore, I was able to show the way that I develop games with Unity and how I organize the files. I know that there might be better ways to do so, but this is the best I can do now. Finally, I want to thank you for this amazing opportunity of doing this test and for all the patience during the interview process so far!