Let's begin from the begginning. After reading the requirements of the task, i felt challenged, whereas I've never done a 2D top-down game before. However, I was really confidente on making a item/shop system, since i've done it several times. So I started working on the weak spots, and searched the internet for how i could code the movement and how to set the scenario.

Afterwards, I started thinking what game I should do. For me, one of the most importante things to have in mind while you are design a game is this simple question: "What feeling I want my player to feel while playing my game? One of the feelings that acrossed my mind was "progression". I liked it and immediatly merged this concept with the Itens/Shop system. I knew that i could develop a system that could be flexible and turn things easy to create new Itens with some variety. I Added to game the fact that I wanted to do shooter that I saw tutorials on YouTube. This way i decided to make a game that you are rewarded by shooting enemies in some kind of arenas, that gets harder and harder. In order to past all arenas, you should buy new and better equipment, that would makes you stronger.

With that said, i will explain the systems. First of all, the player. It was built with the component pattern to separate the logic module of it. Therefore, there is a script that handles movement, another one to handle shooting and a central one as a Controller with singleton pattern – there is even a Scriptable Object just to hold the player stats – all of them independent from each other. The enemies was maded following the exactly same logic, just without a SO for stats.

Now it's the item / shop system time. In this one, I maded a base Scriptable Object called Item. Then I separated it in two kinds: Armor and Weapon, because I wanted to change different values on player when I equip them. The other scripts would inherit this two, with unlimited possibilities. In this game case, each item type (Head, Left Arm, Right Arm, Torso and Weapon) are a class that is empty. This is kind of weird, but i leave it that way so in future, if I would like to extend the code for each type has a different behaviour, i could easily make it by overrinding methods. After that, i defined another two SO to hold and handle a List of Items, one for the Store and the other for Inventory. Next, I needed controllers for each of it: for inventory I maded a static class, since i only wolud have a single inventory with constant acess; for Store I maded a Salesman, once I would like to possibility to have many sellers. Finally, I just had to create the scripts that would handle UI and keep it updated using the Observer pattern.

In the arena system I maded the ArenaSettings ScriptableObject that would have a list of Challenges, witch in turn it is another SO (Yes, I like Scriptable Objects) that keeps information about how many enemies will have on the level, the time range that they spawn and the reard for completing it. Next step was adding a Interactor that would open the UI to select the challenges. And guess it? I would need a script to control this UI again. Finally, i did a controller to instantiante enemies on the scene based on active challenge and keep track of their death in order to reward player on completion.

As far as my opinion on Alien Arena is concerned, I think I did pretty well for a four-day game. The game is fun, it passes a good feeling of progression, and works pretty fine (there are just some few bugs, witch surprised me). Although, I wish I had polished it more. The game is definitely in need of animations for the player, also sprites for seller, arena interactor and for the recharges. I would love to add some NavMesh Agent on enemies; they would be a lot smarter and challenging. Lastly, I would balance better the game for sure.

In short, it was a fine experience making this Project. I learned lot of things, especially in 2D game development. I was very happy with some results of my code, but sad that I couldn't polish the game in time. The next steps on this is: polishing; creating a Editor script for help editing, creating and visualizing items, inventories, stores and challenges; add more enemies with more complex and smaterst behaviours; implementing a save system.