

Market, Shop and Retail Game Creator – Unity Asset

- Documentation -

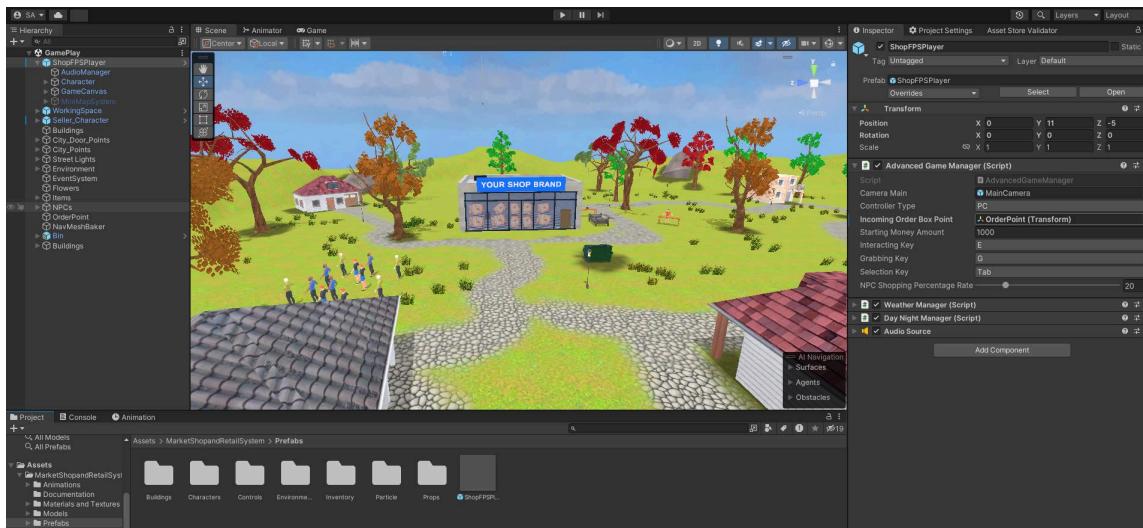
1. Installation

You can purchase the package from Unity Asset Store and download it by using Package Manager in Unity. After downloading the package, import it into your project and “MarketShopandRetailSystem” folder will appear in your project.

Important Note: This asset is using Text Mesh Pro. After installation of the asset, please import Unity’s Text Mesh Pro as well.

You can find the GamePlay Demo Scene in the Scenes folder. Everything is prepared as Prefab in the Prefabs folder. It is very easy to use. If you have further questions, you can contact me via queendeveloper95@gmail.com.

I advise you to run the “**GamePlay**” scene. Run and Play it. While you are playing, you will be able to explore all the components and capabilities of the package and imagine your own game.



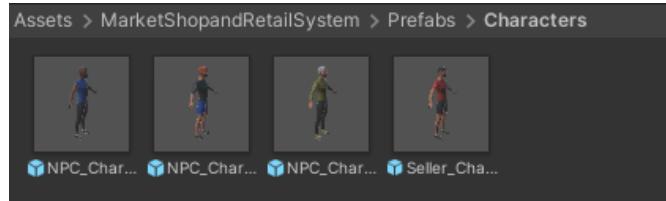
2. Prefabs to Use

The package has got lots of prefabs ready to use. You can find all of them archived carefully in the `Assets\MarketShopandRetailSystem\Prefabs` directory. You only need to drag and drop from the directory into your scene or you can find all of them prepared in the GamePlay scene and copy- paste it from there to your scene (That's what I prefer ☺).



2.1. Prefabs \ Characters

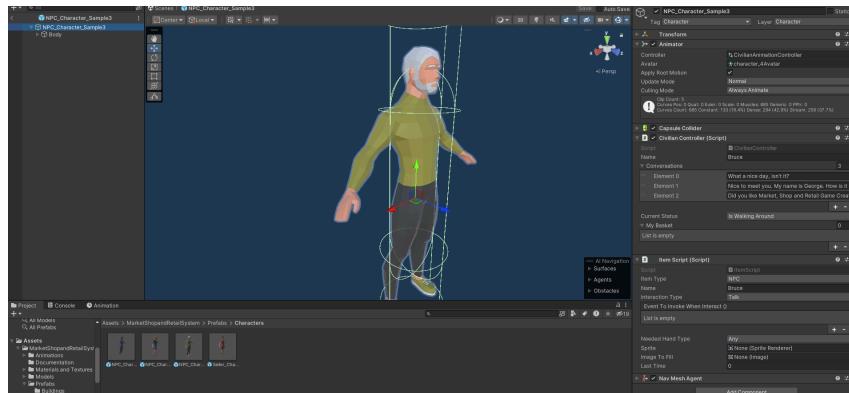
There are four main character prefabs in the package. They are ready to use and they need baked Navmesh in order to move.



How can you bake your nav mesh? It is easy to do. Window > AI > Navigation. Go to the Bake tab on the opened window. Click to Bake and Unity will figure out all walkable areas on your Scene and bake it. This NPC characters have many functionalities like:

- Walking Around,
- Waiting in Idle,
- Talking with Player Character
- Shopping (Go to the Shop, grabs sellable objects as much as he/she want)
- Making Payment
- Going to Home When It is Dark.

- **NPC Character Samples:** These are our citizens who can wait idle, walk around, go shopping, make payment and talk with you. They have their own animations. You don't need to do anything. Just spread them to your scene and determine their conversation texts on their inspector. Let's explore their components:



Civilian Controller:

Name: Your NPC's name.

Conversations: They will pick a random conversation from this list and talk with you.

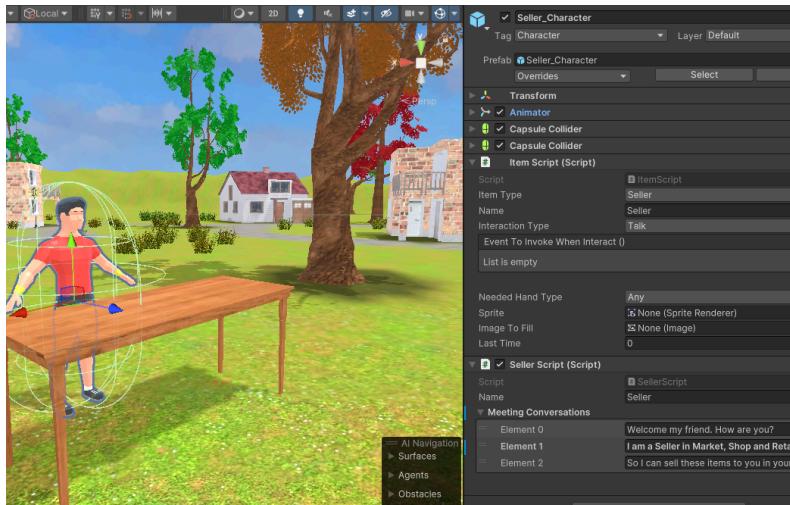
Replacing the NPC Models:

You can replace these NPC models with your own models easily. You can follow these steps:

1. Import your own NPC Model into Unity.
2. Make the model's rig system Humanoid.
3. Make sure its rig is imported without any warning.
4. Open the NPC prefab.
5. Drag and Drop your own Model into the Prefab. Position it correctly.
6. Change the Avatar of Animator with your new model's avatar.
7. Remove the old model from Prefab and it is ready to go. ☺

We will talk about NPC Manager later in the documentation. You can determine your NPCs' walking targets, what hour they will appear or they will go home. You can even determine their houses' doors. So they will go there when it gets dark.

- **Seller Character:** This is our merchant (item seller). You can place them where your market is in your scene. You can talk with him and check his items in order to buy. After your first conversation, he won't have the same conversation again and again with you. Instead of that, he will show his items (by UI) directly next time:



Seller Script:

Name: Your Seller's name.
Conversations: He will pick conversations one by one from this list and talk with you in order.

Replacing the Seller Models:

You can replace these Seller model with your own models easily. You can follow these steps:

1. Import your own Seller Model into Unity.
2. Make the model's rig system Humanoid.
3. Make sure its rig is imported without any warning.
4. Open the Seller_Character prefab.
5. Drag and Drop your own Model into the Prefab. Position it correctly.
6. Change the Avatar of Animator with your new model's avatar.
7. Remove the old model from Prefab and it is ready to go. ☺

2.2. Prefs \ Inventory

The package has 6 sample Inventory objects. You can copy them and replace the meshes, the names and properties in order to create your own buyable – buildable – repairable objects in your game.



OK! This is very important. Before learning how to create our own objects, let's explore some of them. What we have:

- Shelves** (Where you can put all your Sellable Objects). It has 75 Placeable points. You can relocate these placeable points or add/remove as you want. But please don't forget to rearrange the list on Arrayable Area script on the Shelves object. Floors are important as well!
- Table** (Where you can put all your Sellable Objects). It has 27Placeable points. You can relocate these placeable points or add/remove as you want. But please don't forget to rearrange the list on Arrayable Area script on the Shelves object. Floor is important as well!
- Corner Shelves** (Where you can put all your Sellable Objects). It has 42 Placeable points. You can relocate these placeable points or add/remove as you want. But please don't forget to rearrange the list on Arrayable Area script on the Shelves object. Floors are important as well!
- Cash Register** (Where your customers (npc) come to make the payment). All customers will wait in front of the cash register. Once we interact (sit in front of it in order to use) with the cash register, customers will start to put their items in their basket.



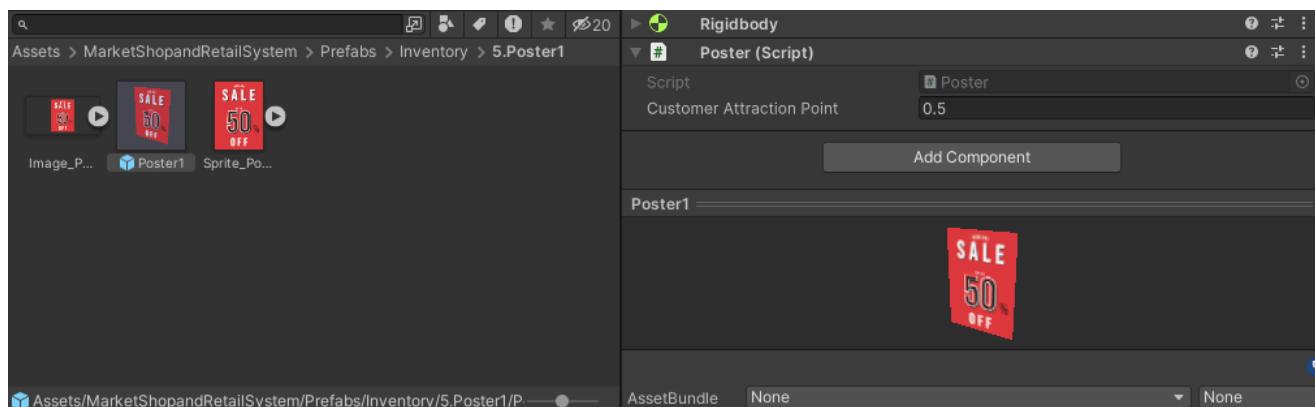
There are few mechanics here. First of all, you need to scan all the sellable objects in the customer's basket one by one. After that, Customer will give you money. You need to calculate the change correctly and press the digit buttons to give the change. Your players will love it. Easy to scan :) and easy to calculate.



5. **Poster1** (You can stick this kind of object type on your shop walls. Sample 1 Discount Announcement Poster)
6. **Poster2** (You can stick this kind of object type on your shop walls. Sample 2 Discount Announcement Poster)

WHAT POSTERS DO?

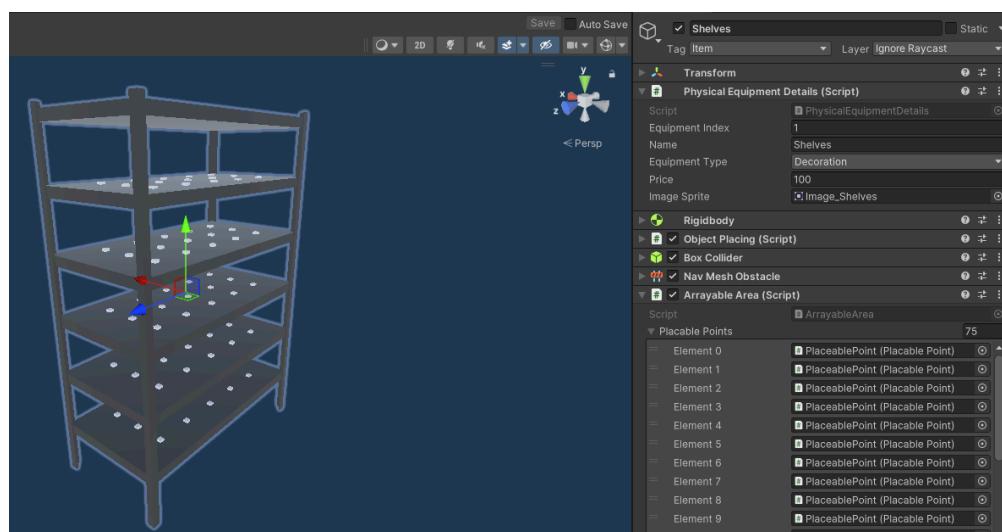
Posters have got “CustomerAttractionPoint” value on their Poster script. You can set this value. This value effects the Customers Shopping Decision Rate. For example, you have 10 Posters1 on your Store Walls. This means $10 \times 0.5 = 5$ Points + 25 Default Customer Shopping Rate (Main Setting). So, Your customers will decide to do shopping with %30 possibility 😊



These are Object prefabs for building and interacting (It will also appear on Seller's Item List and your Inventory). And we need this object's sprite in order to visualize it in the UI:



We have got couple of components on it and let's talk about all of them one by one:



Object Placing: This component gives you the ability to move and rotate it while you are building it. It is essential that this script must have Collider and Rigidbody attached.

Box Collider: We need to collide.

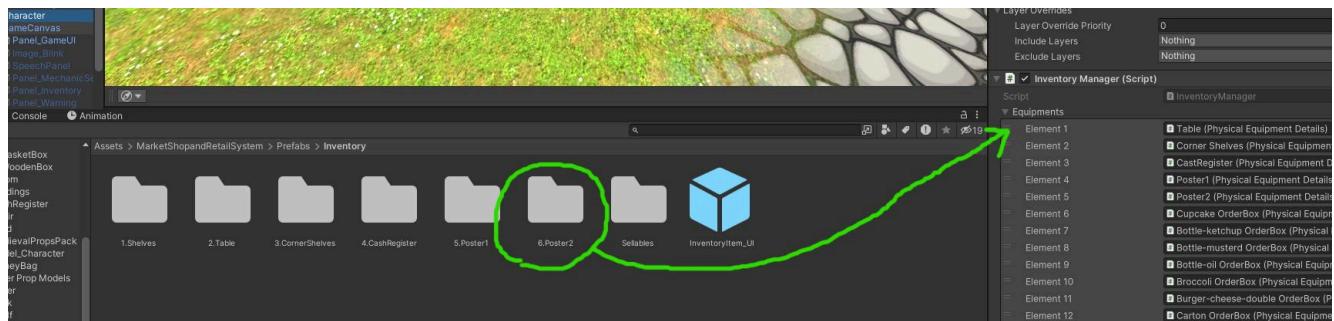
NavMesh Obstacle: After we build this object on our map, we don't want NPCs to walk through it, right? By this component, they will realize it as an obstacle and walk around it.

Physical Equipment Detail component is one of the important components. Let's explore its properties one by one:

- Equipment Index: Unique ID of the equipment. It is important for saving the current positions and rotations of the built objects on the scene. So when the player starts to play the game the next day, he/she will find all previously built objects in the game.
- Name: Name of the Object. It is very important to give a unique name for each object.
- Equipment Type: You can define as Decoration (Can be built on ground), Sellable (Can put on the PlaceablePoints on Decoration Objects) or Poster (Can be built on wall)
- Price: The Object's price to buy.
- Image Sprite: The image of the Object for Inventory and Seller UI.

Arrayable Area: If this object will be an area like shelves or tables, you need to add this script and assign all the placeable points (we locate sellable objects on these points) to Arrayable Area's PlaceablePoints list. DON'T WORRY. 😊 Once you review the Shelves prefab, you will understand.

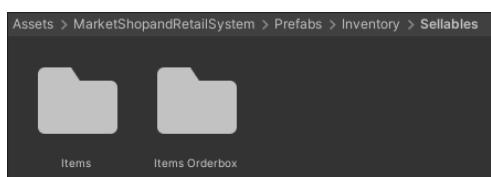
When you complete a new Inventory Object, please don't forget to add it's prefab to the Equipment list in the Inventory Manager (ShopFPSPlayer > Character object):



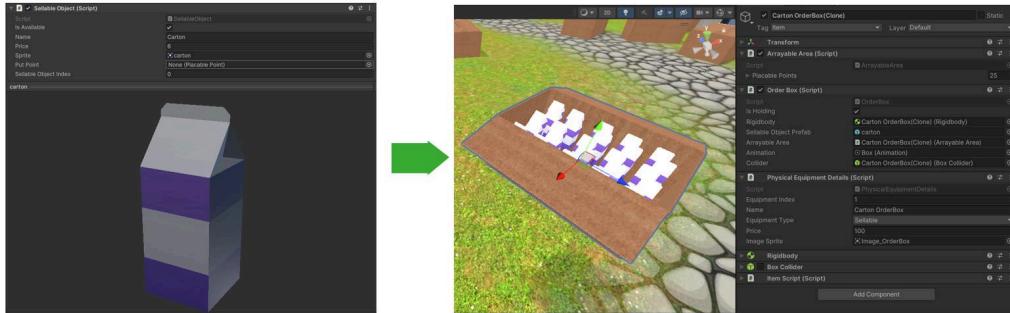
Ok. We have learnt how Buildable objects are working. We can use Shelves, Corner Shelves and Tables which have Placeable Points. Or we can create our own prefabs with new models if we want to customize them.

Now, it is time to explore Sellable Objects. Sellable Object means an item which can be bought by Customers. We need to place these sellable objects on the Shelves. NPC characters (Customers) will check these conditions in order to start shopping:

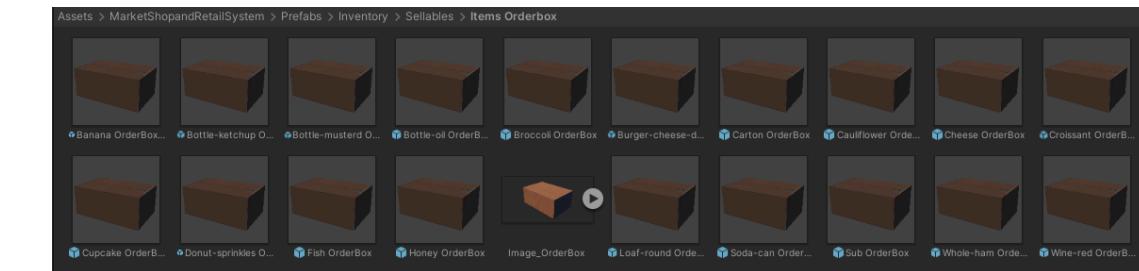
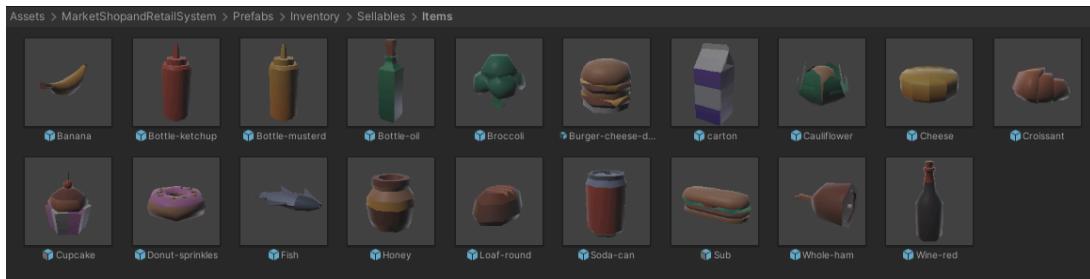
- Is there any Sellable Object located in the Scene that we can buy?
- Is there any Cash Register that we can make our payment?
- Are these Sellable Objects reachable? (Is the door of the shop open?)



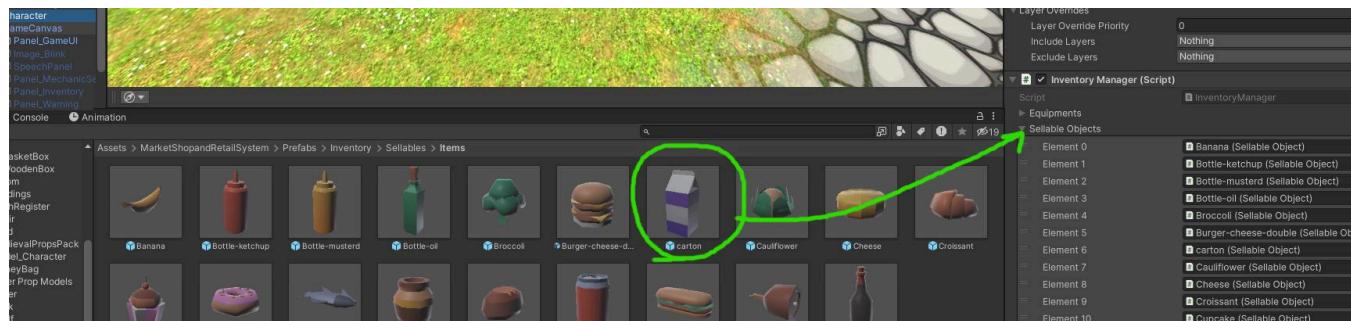
Each Sellable Object needs two prefabs. One is for the object itself (where we can take it out from the box and place it on a placeable point of Shelves). Second is Orderbox prefab which contains a certain amount of the Sellable Object. We buy Orderbox from Seller and one it is delivered in front of our shop, we carry it and place all the sellable objects on shelves:



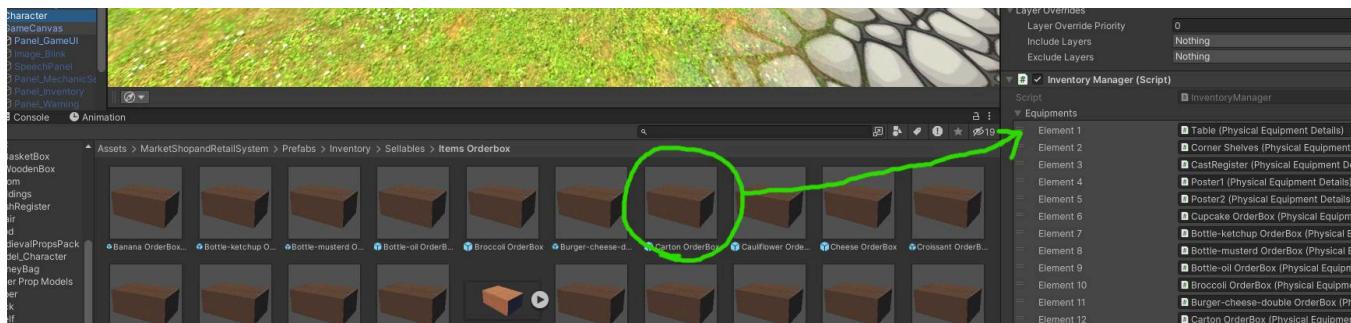
I have created 19 sample Sellable Prefab and their Orderbox Prefab for you. You can just explore them and have a better understanding. Easy! Create your Sellable Object, create your Orderbox for this Sellable Object. Assign the sellable item to Orderbox script's "Sellable Object Prefab" and it will create the box for you automatically with full of that sellable object 😊



When you complete a new Sellable Object, please don't forget to add it's prefab to the Sellable Objects list in the Inventory Manager (ShopFPSPlayer > Character object):

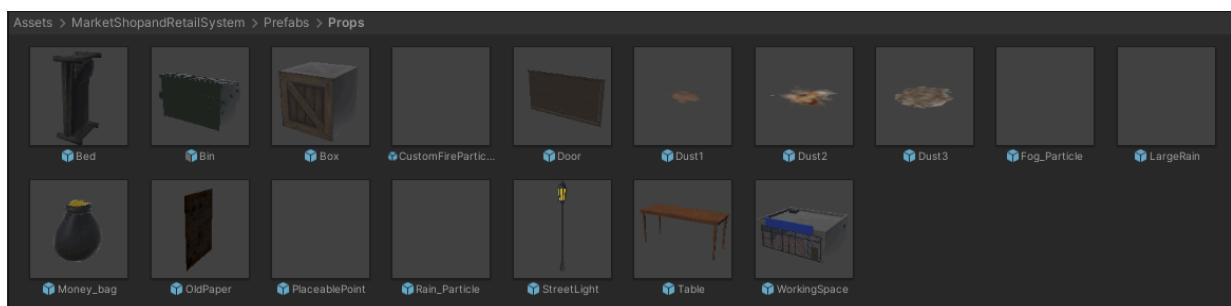


When you complete a new Order Box Object, please don't forget to add it's prefab to the Equipments list in the Inventory Manager (ShopFPSPlayer > Character object):

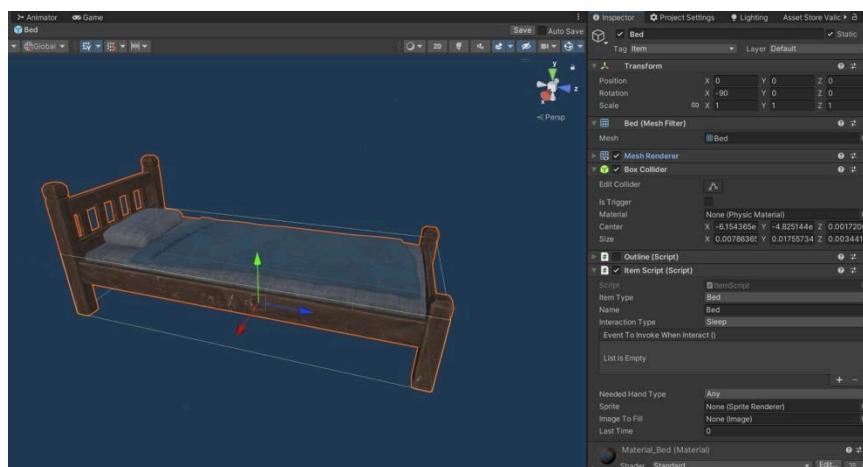


2.3. Prefs \ Props

These props are ready to use and you can see many of them already in the GamePlay Demo scene. While you are designing your own Scene, you can use them by simply Drag and Drop. Please explore them one by one.



All these prefabs have got **ONE COMMON COMPONENT** which is called **ItemScript**. This script mostly decides how to be interacted by the Player. Let's explore one them together:

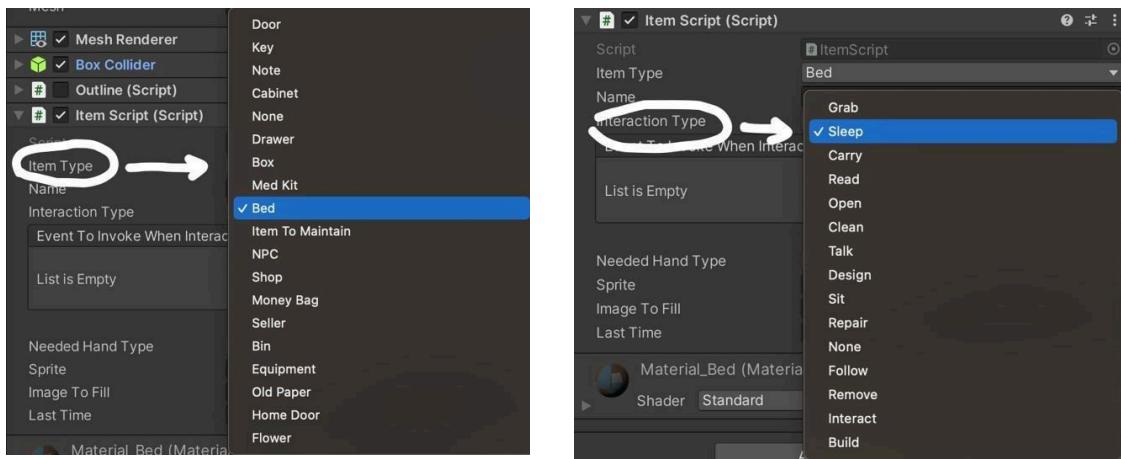


Item Script:

Item Type: You can select the item type or define new ones for your own game.

Interaction Type: How the player will interact with this object and what he/she will see as a text on the scene.

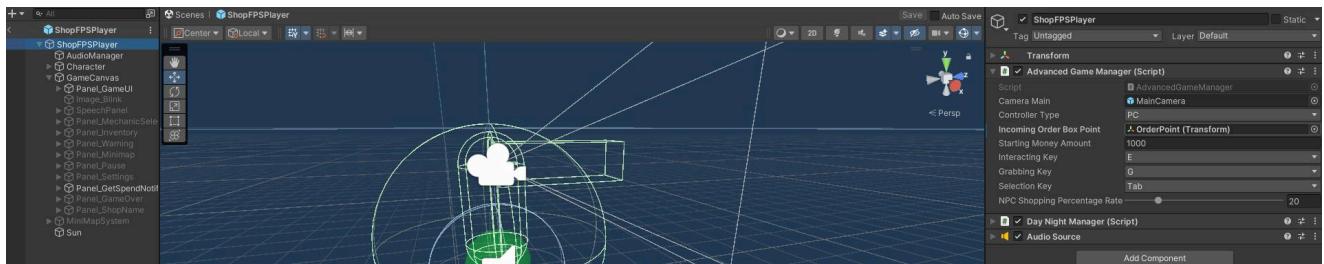
We have got various ready to use Item Types and Interaction Types. They are listed below:



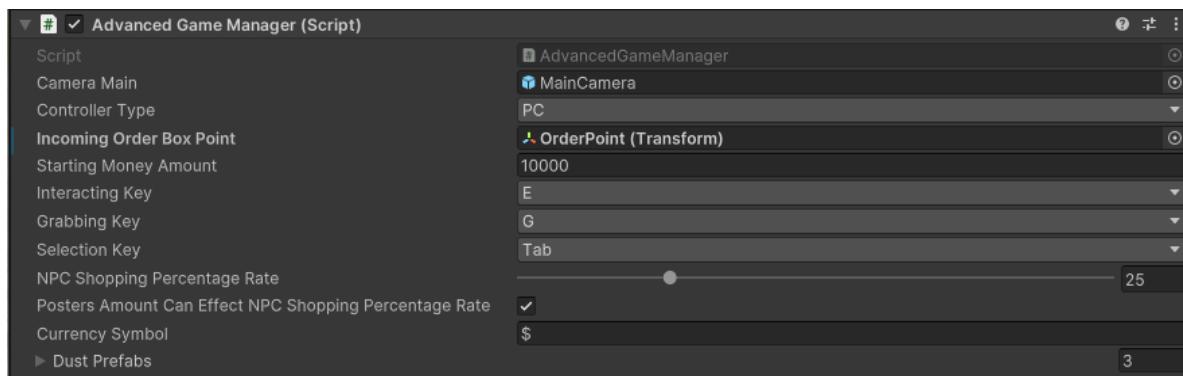
2.4. Prefabs \ ShopFPSPlayer 並 並 並

Guys! This is the most important prefab in the game. It is the center of everything. This prefab determines the Key Inputs, Manages the Canvas UI Panels, Game Conditions, Weather, Day – Night Cycle , Mini Map System.

This prefab manages the audio operations. It contains our FPS Character and all the mechanics like Building, Cleaning. Let's check together.



You should always have this prefab in your Scene. We will go through all the components in detail so you can configure them easily for your own game.



Starting Money Amount: When the player starts the game for the first time, how much money he/she will have.

Interacting Key: Key code for Interaction Grabbing Key: Key code for Grabbing objects.

Selection Key: Key code for Mechanic Selection.

Incoming Order Box Point: When we call Orderbox items from our Inventory, they spawn at this transform's position.

Controller Type: You can select PC or Mobile as controller type. If you select PC, your game will be playable by using Mouse and Keyboard. If you select Mobile, Touch Buttons, Joystick and Touchpad will appear during the game and players will be able to play the game by using these controls.

NPC Shopping Percentage Rate: You can configure this global shopping rate for your game. Basically it controls the possibility of a customer's shopping decision. If you make it 100 😊 they will always want to go shopping (like women). If you make it 20, they will decide to go shopping with %20 possibility. They will walk around %80. (It is random)

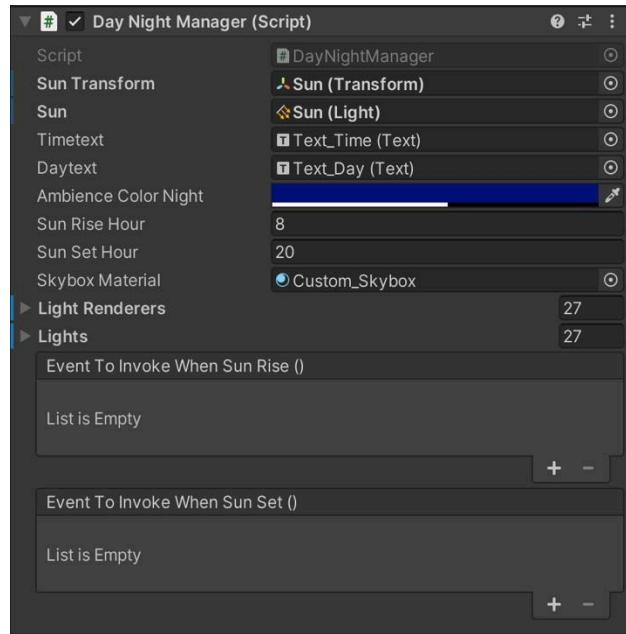
Posters Amount Can Effect NPC Shopping Percentage Rate: This setting determines how the number of posters (Posters Amount) impacts the percentage rate at which NPCs engage in shopping activities. In essence, the more posters there are, the higher the likelihood (or percentage rate) of NPCs shopping.

Currency Symbol: What symbol will appear on the price tag and Cash Register Screen.



Rain Particle System: Attach your Rain particle object on the Scene.

Weather Change Interval: Every x Period, Weather Manager will select a condition (Sun or Rain) randomly and executes it.



Sun Transform and Sun: Attach your Directional Light on the Scene to both. Light Intensity will be decreased or increased during Day-Night Cycle to make it dark or light.

TimeText: Text UI for showing the exact time in the game (Exp 14:53)

Daytext: Text UI for showing the current Day in the game (Exp: Day 4)

Ambience Color Night: When it gets dark, the ambience color of the environment.

Sun Rise Hour: What time the sun will rise in the cycle.

Sun Set Hour: What time the sun will set in the cycle.

Skybox Material: You have to assign your Skybox of the Scene in order to decrease or increase the exposure of the skybox for night.

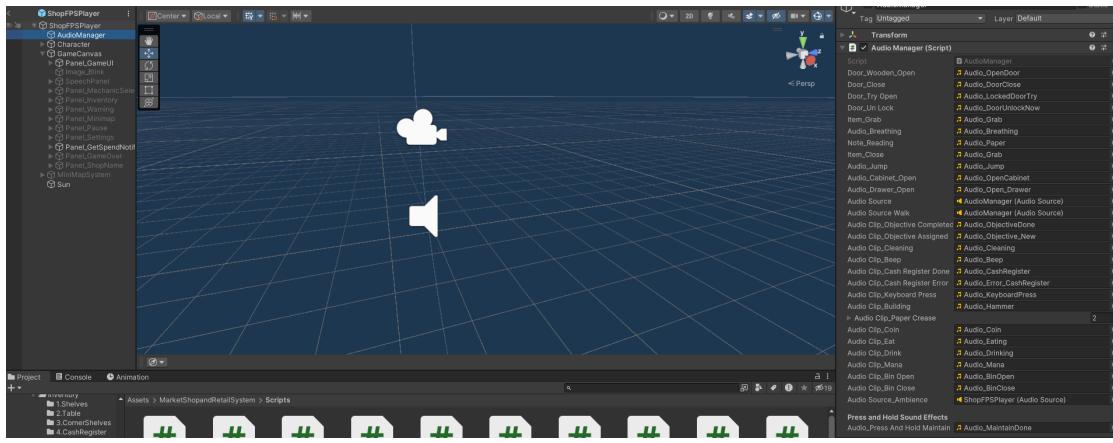
Light Renderers: Attach your street lights' materials.

Lights: Attach your point or spot lights of your scene to this list. When it gets dark, Day Night Manager will light up your lights. When it is day time, Day Night

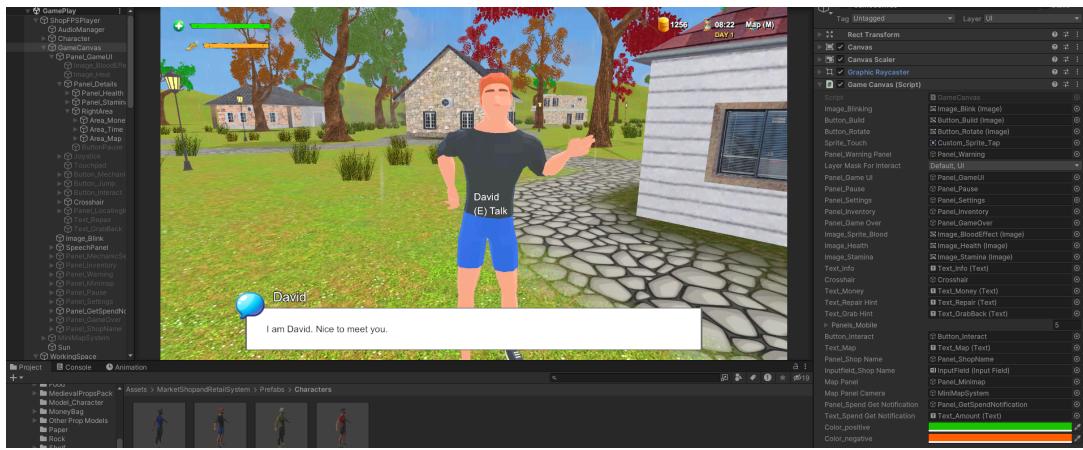
Manager will switch off your lights automatically ☺

AudioManager has got all the required Audio Clips and you can call any of them by easily calling

AudioManager.Instance.PlayXXX() method:



GameCanvas has got Game Canvas Script and it manages all the UI Elements, opening – closing UI Panels, Interaction UI elements, Speech Management, etc. You can explore them easily.



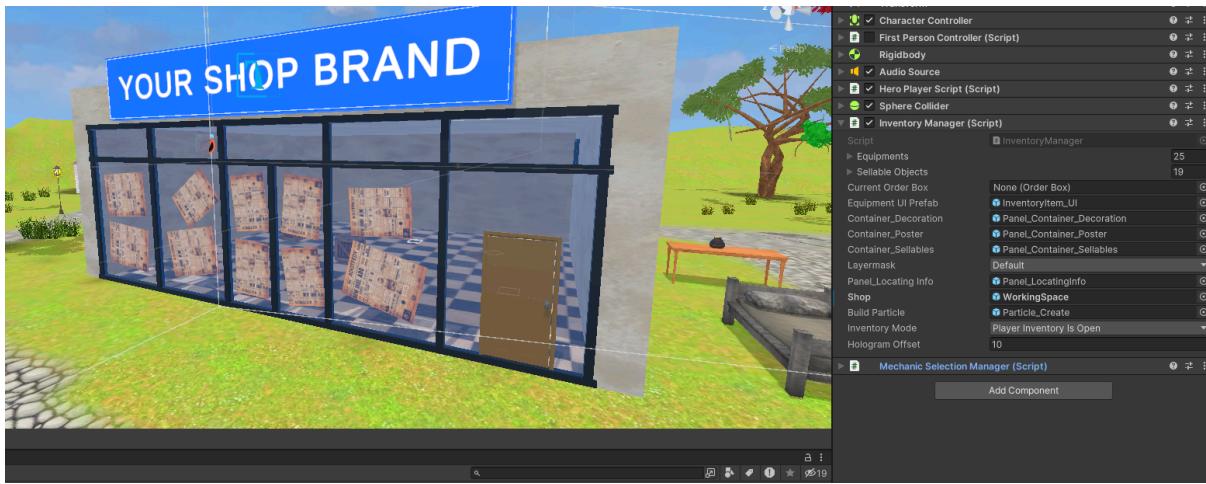
The Minimap System is the map of the scene. It follows our player from top. There are two buttons: Zoom In and Zoom Out. The camera's orthographic camera size is being increased or decreased by these buttons. You can change the MiniMap System's following object by simply Selecting MinimapCamera Object > Assign your Player to Player attribute of Minimap Script.



Character Object is our player character. It has got 3 main components. Our main camera is also attached to this object.

1. *First Person Controller Component:* You can configure our character's speed, jump height, sprint speed here.
2. *HeroPlayerScript Component:* You can configure our character's health and stamina here. When you are developing your own game, these helper methods can help you. You can call them from anywhere:
 - a. HeroPlayerScript .Instance.GetDamage(int Damage)
 - b. HeroPlayerScript .Instance.Heal(int Damage)
 - c. HeroPlayerScript .Instance. ActivatePlayer()
 - d. HeroPlayerScript .Instance. DeactivatePlayer()
3. *Inventory Manager:* Our Seller Character's Items and our Inventory system is being managed by Inventory Manager. There are 3 important properties to focus here:
 - a. Equipments: When you create your Objects to buy, build and repair, please don't forget to add their prefabs here Equipments list.
 - b. Shop: When you buy an item from Seller and build it to your Working Space, this Shop transform is its parent. So, All your built objects are being contained in this Shop transform.
 - c. Sellable Objects: You can create your own sellable objects like Chips, Potato, Cola, etc. Once you created them, please don't forget to add them into your Sellable Objects list in the Inventory Manager.

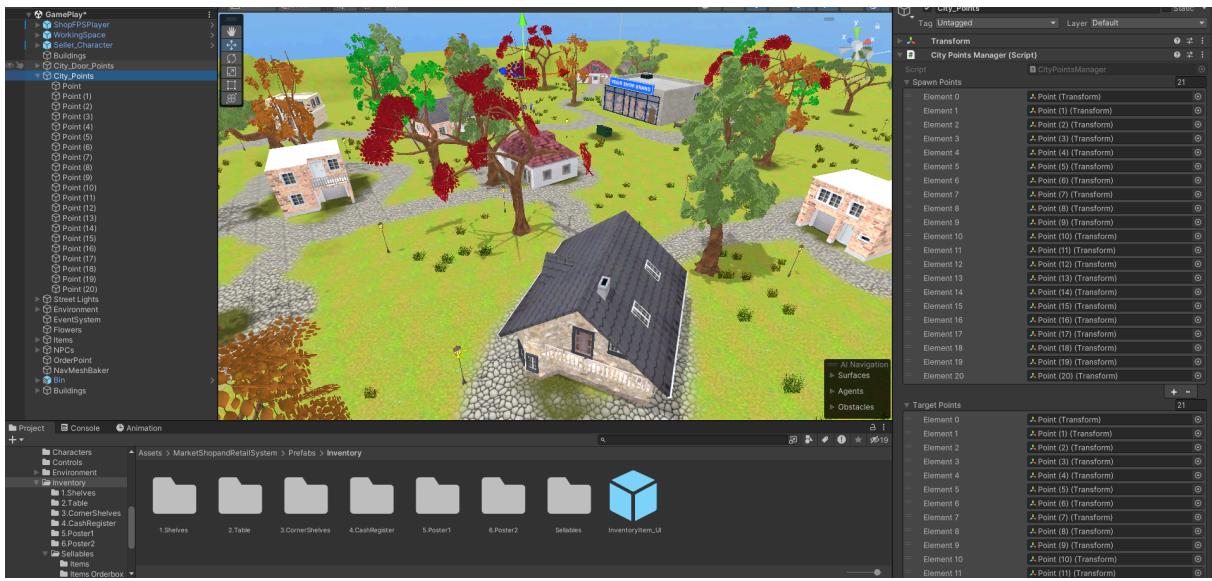
IMPORTANT NOTE: You should know that you can only decorate the area triggering with the "Collider_DecorablesArea" object in the Scene. This collider has a "Shop" tag and you can scale, rotate and reposition it anywhere you want.



[3. Other Helper Managers:](#)

There are two important Components help to make your town more lively and dynamic. These are City Points Manager and NPC Manager.

Let's start with the **NPC City Points Manager**. This Manager contains NPCs' Spawn Points when the scene is started, their random Target points to walk and Doors list for going there and disappear when it gets dark. They don't appear until the sun rises and when the sun rises, they appear from the same location

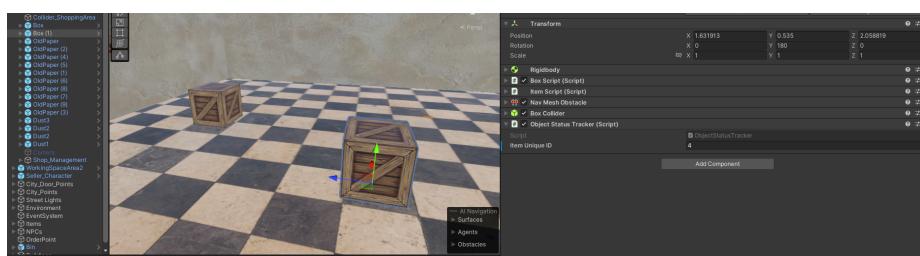


Secondly, we have an **NPC Manager** which manages the NPC characters' moves and contains all of them as a parent. There are some important configurations you can configure for your own game:



Civilians list contains all the NPC characters on the map. Going out Hour is to determine the Hour when NPC characters will appear (for example 8 am in the morning) and Going Home Hour is to determine the Hour when NPC characters will take a random door as a target and go there to disappear until Going out Hour.

Object Status Tracker is an important script as well. If an item like box (throwable), dust (cleanable), money bag (grabbable) has this script with a unique id, then its status will be tracked by the System. (If you cleaned and removed it, next time it won't appear):



I hope you like my asset and create successful Market, Shop and Retail games with it. ☺

If you need any help or if you have any questions, please don't hesitate to ask me guys. You can send me an email anytime (queendeveloper95@gmail.com)

Kind Regards,
Queen Developer