
SPACE SHOOTER EDITOR

User's Manual

Developed by Germán López Gutiérrez

CONTENT

FIRST STEPS.....	3
GAME INSTANCE CONFIGURATION	3
NEW GAME INSTANCE CREATION.....	4
LEVEL.....	4
GAME MODE CONFIGURATION	5
NEW GAME MODE CREATION.....	5
ENEMY CREATION	7
GENERATE NEW ENEMY.....	7
CUSTOM AI.....	8
CUSTOM BULLET CREATION	9
GENERACIÓN DE NIVELES	10
APPENDIX.....	11

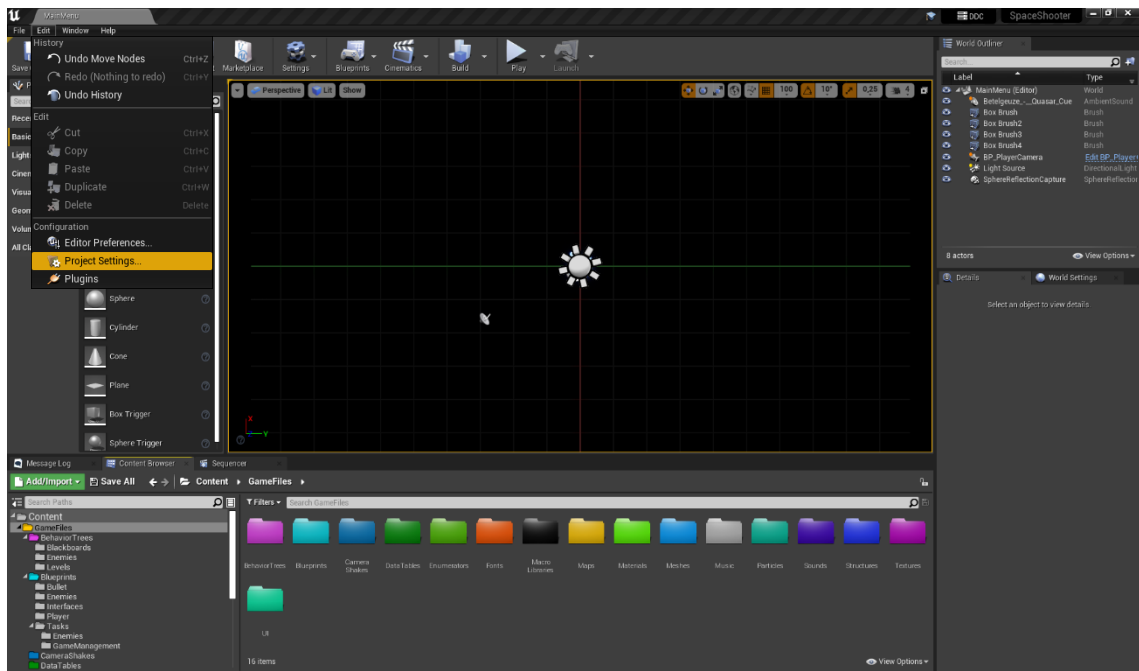
FIRST STEPS

To make the files found in this project work in a different project, the following steps must be followed:

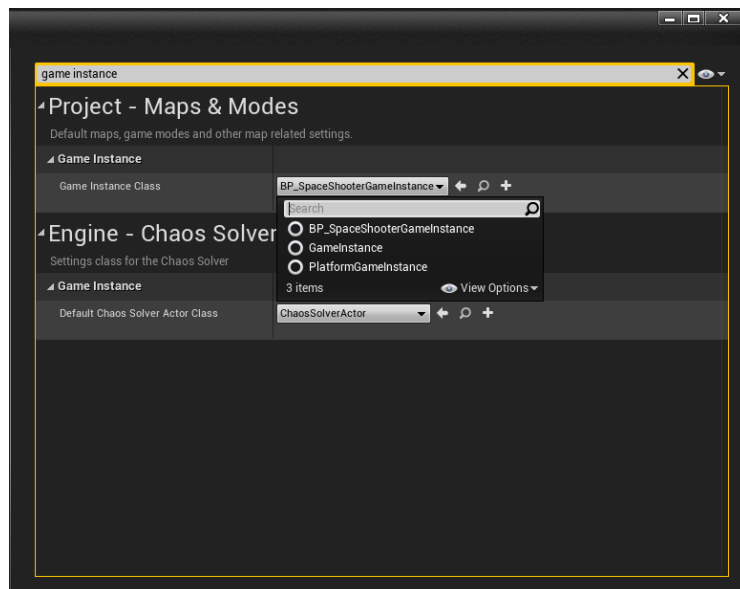
- Change the **Game Instance** of the project / Add interface to an existing one
- Choose the level where you will work
- Choose the **Game Mode** of the scene / Add interface to an existing one

GAME INSTANCE CONFIGURATION

You must access the **Edit/ProjectSettings** tab.



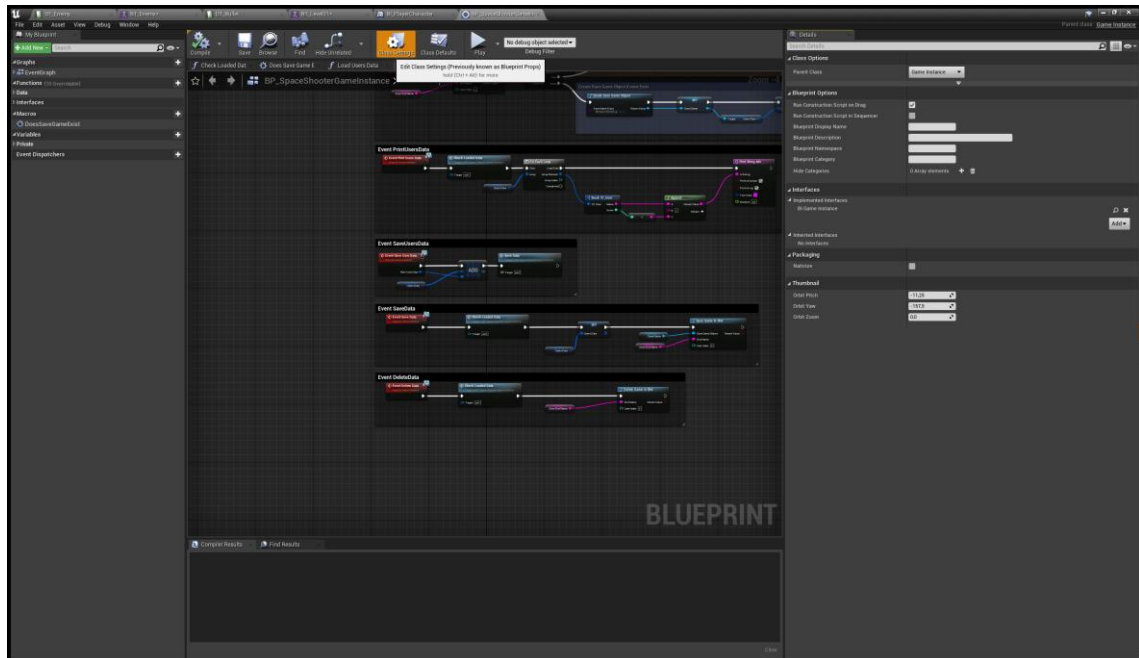
Once this is done, type **"Game Instance"** in the explorer and change the **Game Instance Class** to **BP_SpaceShooterGameInstance**.



NEW GAME INSTANCE CREATION

In case you want to use your own **Game Instance**, the only thing you have to make sure is to include the **BI_GameInstance** interface in yours.

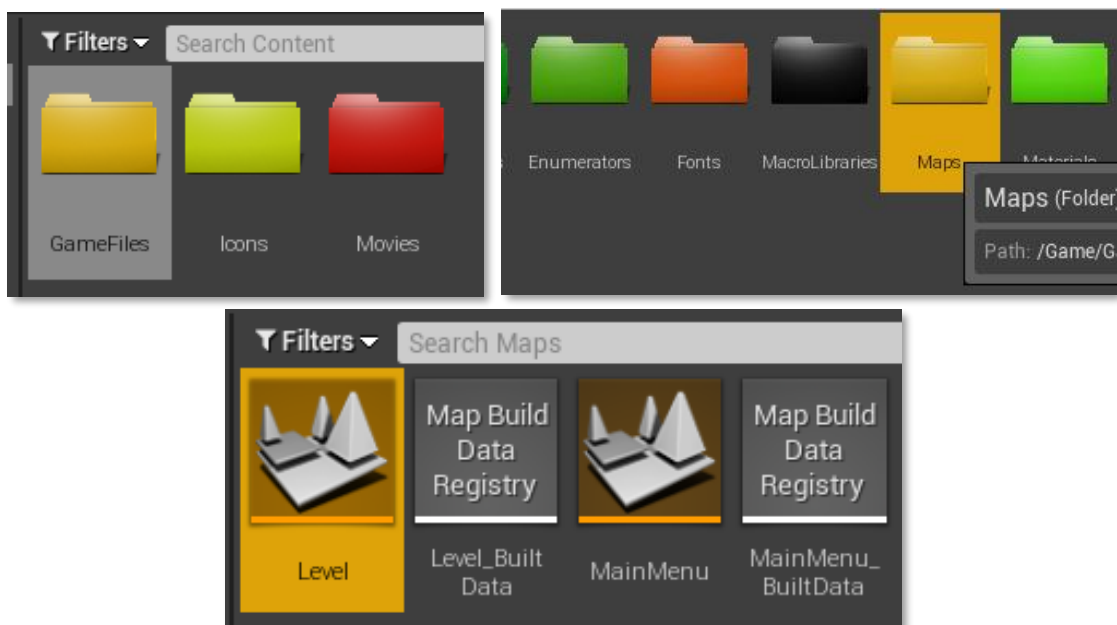
For it, inside your **Game Instance**, open the **Class Settings** tab, and where it appears **Interfaces**, add the **BI_GameInstance** interface.



Now, you can modify the content of the methods of this interface to calibrate it to your liking.

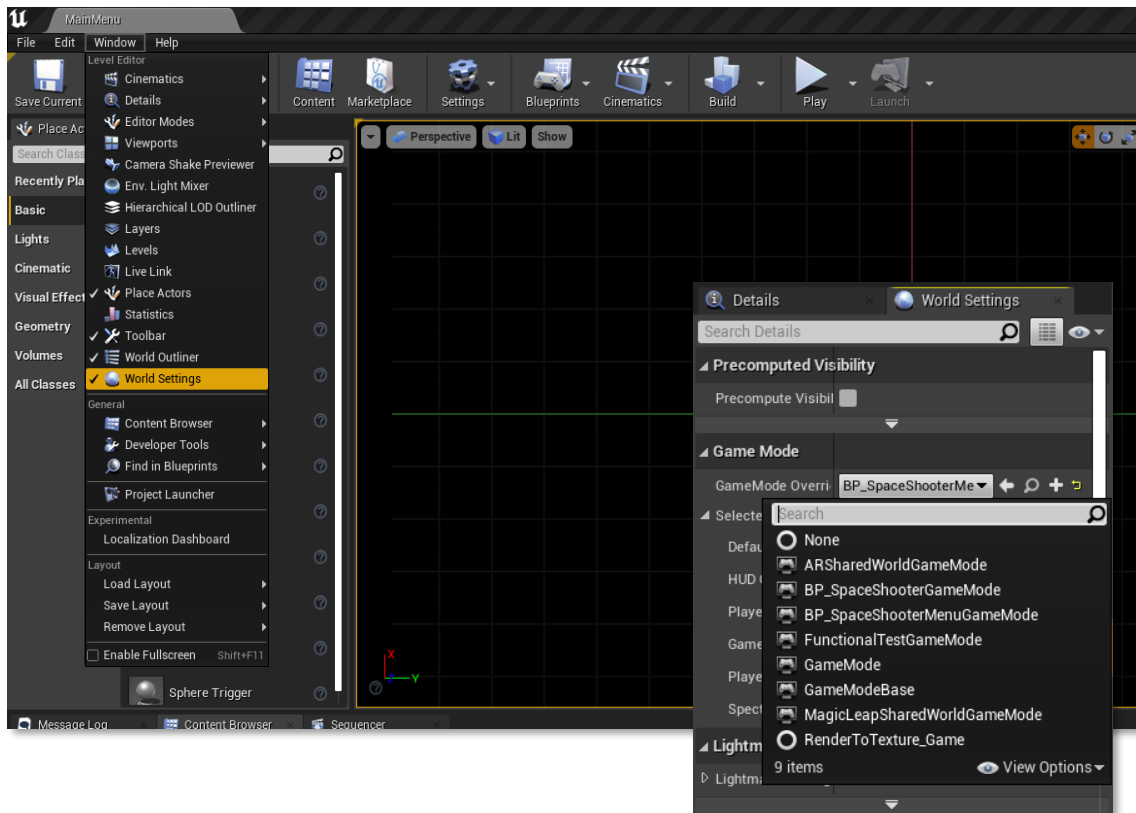
LEVEL

The map to be worked on will be the one named **Level** located in the **GameFiles/Maps/Level** directory.



GAME MODE CONFIGURATION

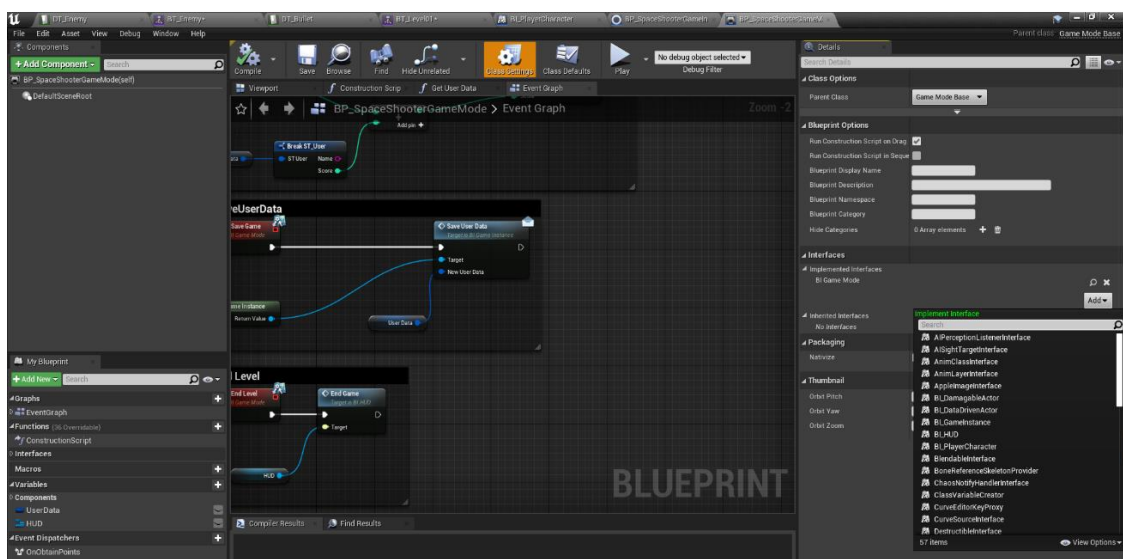
On the map where the game will be played, check that the **Game Mode** selected in the **World Settings** window is the one named **BP_SpaceShooterGameMode**.



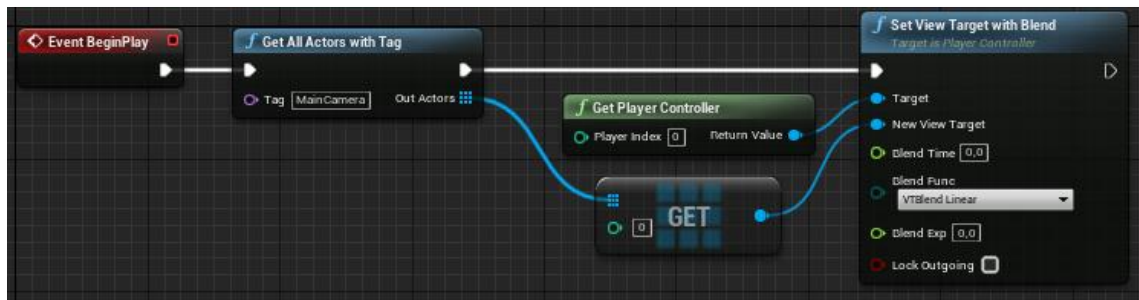
NEW GAME MODE CREATION

In case you want to use your own **Game Mode**, the only thing you have to make sure is to include the **BI_GameMode** interface in yours.

To do so, inside your **Game Mode**, open the **Class Settings** tab, and where **Interfaces** appears, add the **BI_GameMode** interface.



In order to configure the camera in the new Game Mode, add the following in the **Begin Play** event:



Despite the existence of this possibility, I do not recommend creating your own Game Mode, as it may cause several problems related to the user interface and the implemented Dispatchers.

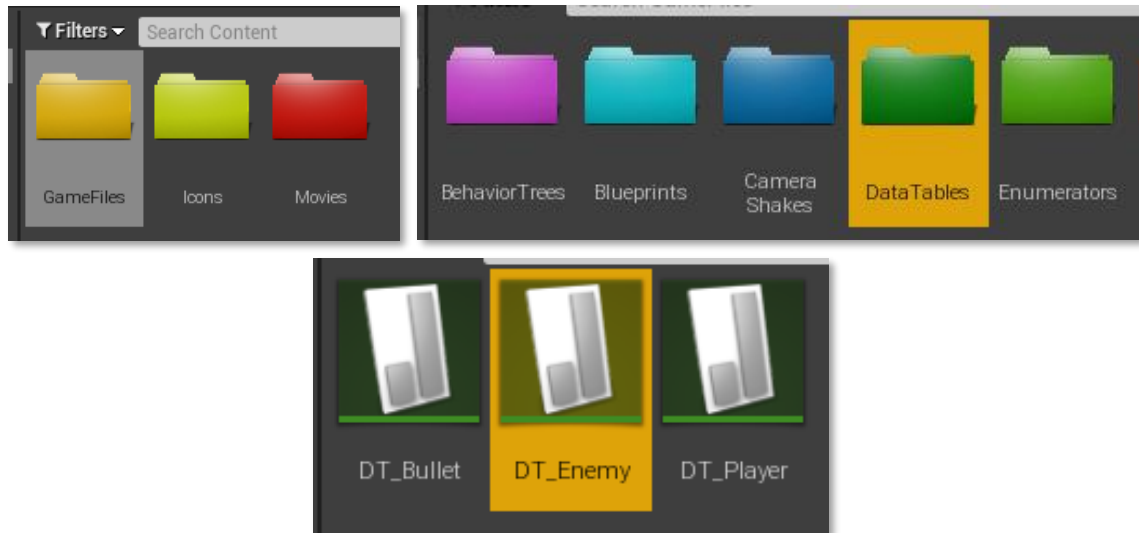
ENEMY CREATION

GENERATE NEW ENEMY

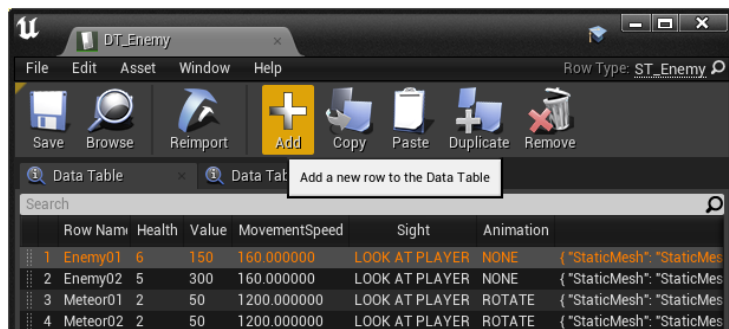
To create new enemies, the first thing to do is to import a **Static Mesh** of the desired appearance.

Once the **Static Mesh** is imported, you must access the directory and open the file **DT_Enemy**:

GameFiles/DataTables/DT_Enemy



After opening the corresponding **Data Table**, you can create a new enemy by clicking on the **Add** button:



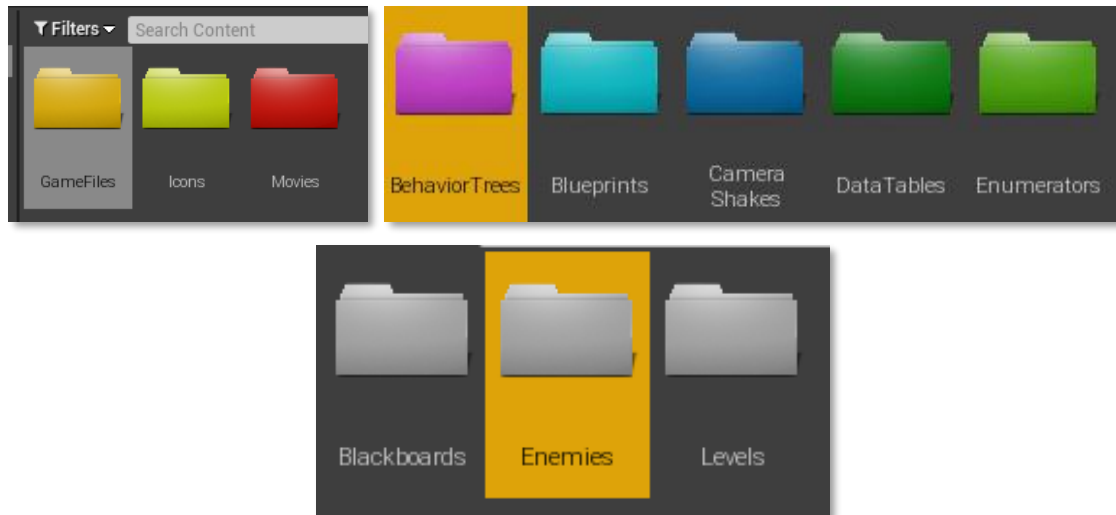
After that, you will have to fill in all the data to create your enemy:



CUSTOM AI

To create a custom AI, you will make use of the **Behavior Trees** located in the following directory:

GameFiles/BehaviorTrees/Enemies

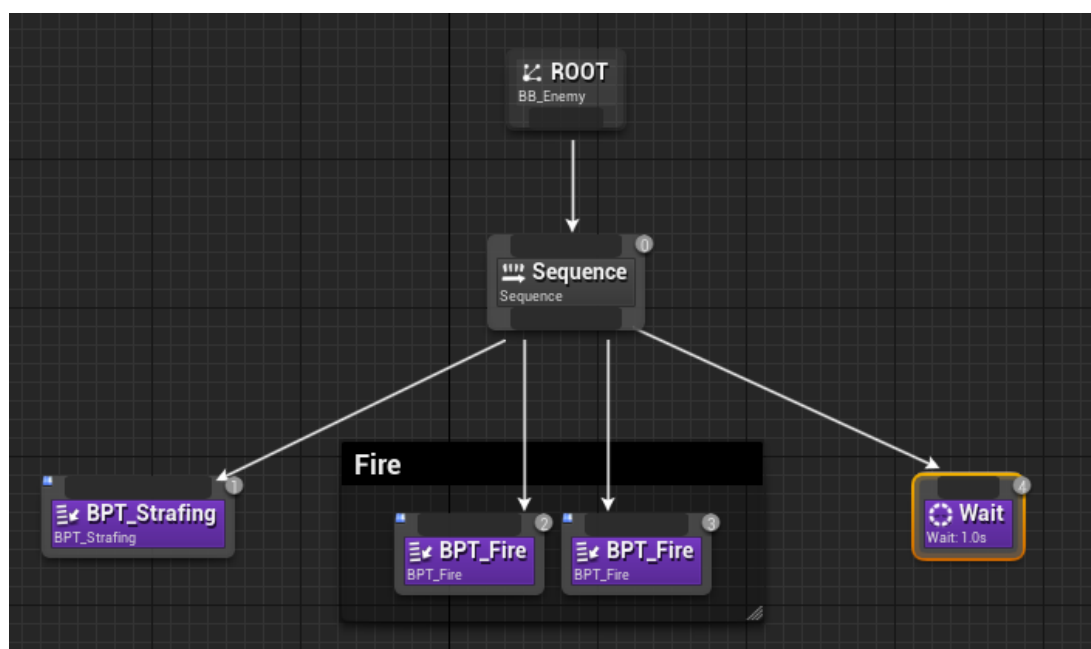


In this folder, you will duplicate any of the existing **Behavior Trees**. In the **Behavior Tree** created, the **Tasks** will be located. Currently there are the following ones:

- **BPT_Fire**: Fires a series of bullets receiving a total number of bullets, the duration of the attack, the type of bullet and both the initial and final angle of the sweep.
- **BPT_MoveForward**: The enemy moves forward (following its **Forward Vector**).
- **BPT_Strafing**: The enemy moves in a random diagonal avoiding walls.

In the *GameFiles/Blueprints/Tasks* directory you can find existing **Tasks**, which can be used as a reference to create custom ones.

With these **Tasks** you can create your own AIs for the enemies following the same logic required for a **Behavior Tree** of any other project.

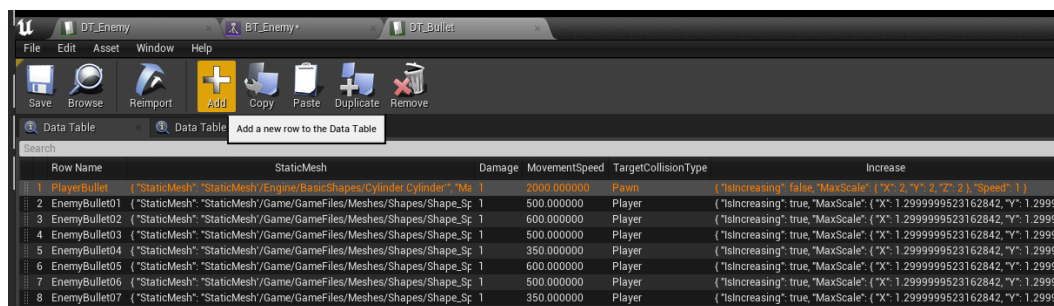


CUSTOM BULLET CREATION

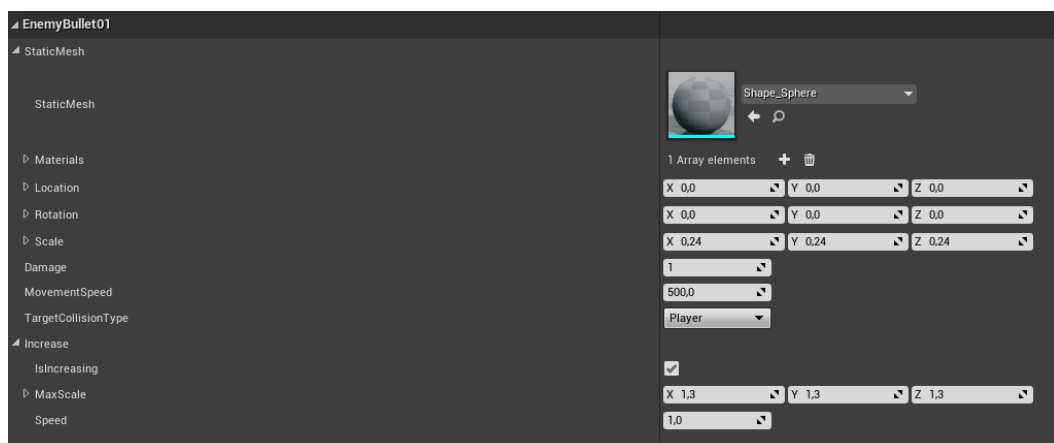
The *GameFiles/DataTables/DT_Bullet* directory contains all the bullets that can be fired by both the enemies and the player.



To create a custom bullet in this Data Table, the procedure is the same as for the enemies. You will have to click on the **Add** button or duplicate one of the existing ones.



Now you only have to fill in the variables required for its operation.



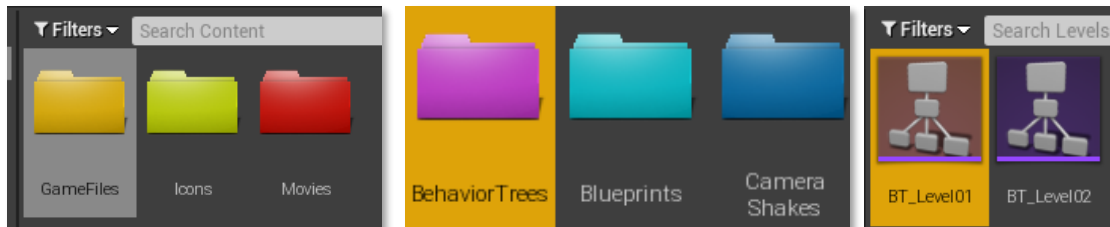
In case you want to modify the player's bullet, change the parameters of the row named **PlayerBullet**.

GENERACIÓN DE NIVELES

One of the most interesting points of this system is the possibility of creating customized levels. To do this, as with the enemies, there is a **Behavior Tree** that manages it.

To create a level, access the following directory::

GameFiles/BehaviorTrees/Levels



Within this folder there are two levels of examples. You can duplicate either one to create your own level.

In these **Behavior Trees** there are three **Tasks** to consider:

- **BPT_SpawnEnemy**: Makes an enemy of your choice appear at a position on the stage. In addition, you can select if it is necessary to eliminate it to advance to the next node.
- **BPT_SpawnWave**: Works the same as the previous **Task**, but with more than one enemy. You can choose the speed of appearance of this list of enemies.
- **BPT_EndLevel**: Ends the level.

The stage is divided in a 20x20 grid, being the central position the {0,0}. Therefore, the grid starts from position {-10,-10} to {10,10}.

APPENDIX

All data structures, enumerators, macros and interface elements are easily configurable in their respective files. The code is commented and following the logic of the code, the game can be customized for a completely personalized experience.