

# **PROJECT**

## **DOCUMENTATION**

### **Character Creation System**

Hugo Garelha

# 1 Overview

## 1.1 - What is included

This is a Small Game Dev Toolkit that may help you with Character generation, recruitment and so on, especially for RPG games, Tactis and similar games.

This system Includes:

- Class system that allow you to create a class and their stats for combat Ex: Soldier, Mage ,Thief and so on. The system already includes Scriptable Objets easy to edit or change with new values.
- Random Name selector The Random name selector selects a name from a List of names. Filters for Race and Gender are included but can be changed and added at will. The System Keeps track of Already in use names in order to not repeat.
- Character Saving The ToolKit has a built in character save system that allows to save the created Character with all their informations,auto-save on empty slots and delete or replace characters.
- Random Selector of stats within Range It includes a system that will pick a random value between a range for each stats this allow slightly different characters to be created even if they share the same class.
- Built in Lists The Toolkit already contain example lists that can be used as they are or easily changed without the need to edit any code.
- Demo and Display Example The project includes a Demo scene that allow you to test the system the demo also includes functionalities for displaying Character Informations that can be re-used in your own project.

## 1.2 - How to Use

The project includes ready to use prefabs and already set up values that you can use and adapt to your project easily.

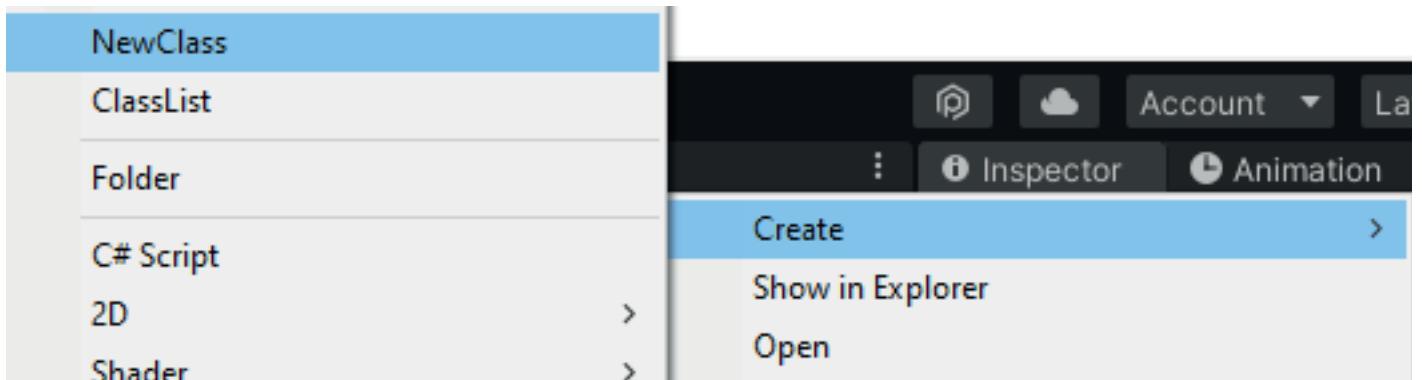
For that see the Demo content folder.

# 2 Class System

## 2.1 - Character Classes

The Character Classes is a Scriptable Object that already includes all the main stats that you may find in an RPG game.

In order to create a new Custom Class, you can create a new Scriptable Object "New Class".



By default when creating a new random Character the stats will be randomized between a min and max value.

If you don't want these values to be randomized you can simply leave the max empty and the system use the stat Min value as default.

| Class Initial Values Range |     |
|----------------------------|-----|
| Class HP Min               | 90  |
| Class HP Max               | 100 |
| Class SP Min               | 15  |
| Class SP Max               | 20  |
| Class ATK Min              | 88  |
| Class ATK Max              | 98  |
| Class DEF Min              | 84  |
| Class DEF Max              | 94  |
| Class MATK Min             | 60  |
| Class MATK Max             | 70  |
| Class MDEF Min             | 72  |
| Class MDEF Max             | 77  |
| Class RES Min              | 55  |
| Class RES Max              | 65  |
| Class ACC Min              | 50  |
| Class ACC Max              | 60  |
| Class LUCK Min             | 15  |
| Class LUCK Max             | 20  |
| Class SPD Min              | 57  |
| Class SPD Max              | 59  |
| Class Raw CRITDMG          | 25  |
| Class Raw CRITDRES         | 10  |

If you want to change the class script you can do that by editing the ClassDefaultValues Script, but keep in mind that other changes may be required in other scripts.

If you want to use only some of the stats but not all you can simply ignore those stats that you do not need all values will be kept to 0 and can be ignored in your project.

The class system also includes other values that are related to another system (Leveling System) not released yet at the time of posting.

## 2.2 - Class Lists

In order to Keep track of the actual available classes another Scriptable object is used containing lists of the available classes, by default it includes three lists but you can add as many as you need for your project

# 3 Character Stats

## 3.1 - Character Stats

The character Stats is a Superclass that allows the creation of child Scripts that will contain all the required information about the character Stats/Name/race etc.

It also includes all required methods to use the stats in battle.

To do so you can simply call the SetCurrentStats and it will create the currentValues without the need of adding any more code.

```
0 referências
public float CurrentHp
{
    get { return currentHp; }
    set { currentHp = value; }
}

float currentAttack;
0 referências
public float CurrentAttack
{
    get { return currentAttack; }
    set { currentAttack = value; }
}

float currentDefence;
0 referências
public float CurrentDefence
{
    get { return currentDefence; }
    set { currentDefence = value; }
}
```

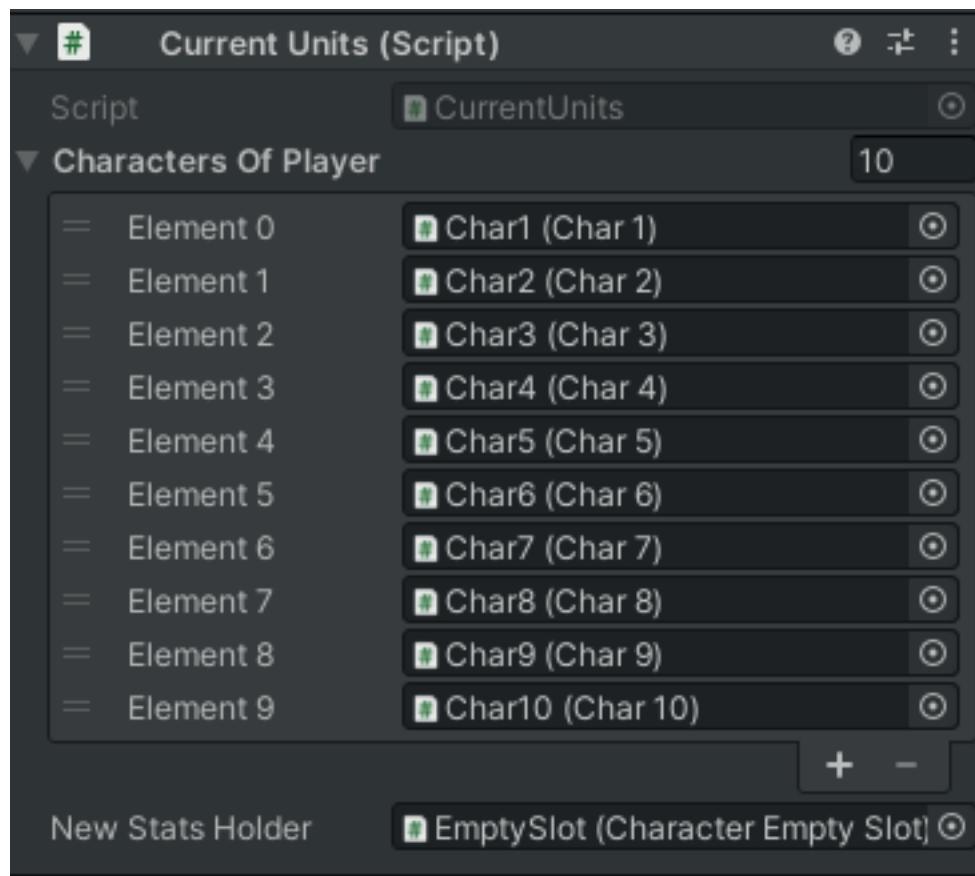
```
0 referências
public void SetCurrentStats() //At b
{
    currentSpeed = speed;
    currentHp = hp;
    currentSp = sp;
    currentAttack = attack;
    currentDefence = defence;
    currentMatk = matk;
    currentMdef = mdef;
    currentRes = res;
    currentAcc = acc;
    currentLuck = luck;
    currentCritRate = critChance;
    currentCritRes = critRes;
}
```

In order to Create new characters you must Create Scripts of type CharacterStats to contain all individual information that can be used, changed and so on.

## 3.2 - Characters Lists

The system requires a Character Units Script to work, this script contains references to all character Stats scripts.

The system also requires a Character Stats script to be used as a Holder, this script will contain all newly created information before it is either saved or deleted.

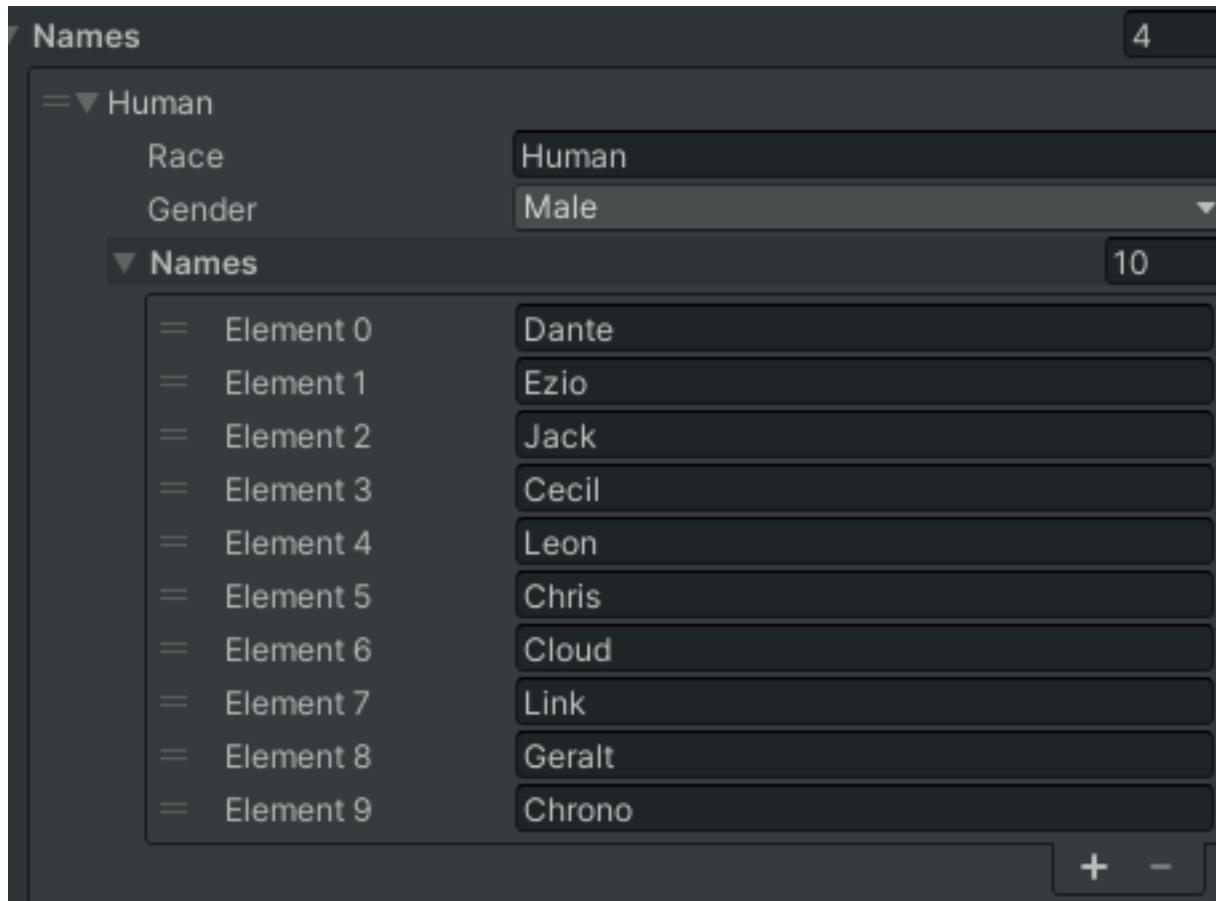


The list can contain as many Characters as required.

# 4 Names List

## 4.1 - Name And Race List

You can create as many Races and Names as you want by adding to the List.



By default when deciding a name the system will randomize between all the created lists but the system comes with an option for ignoring gender and race.

On the other hand the system allows you to easily create a specific race and gender with very small changes (you can see how to replicate this in the demo content).

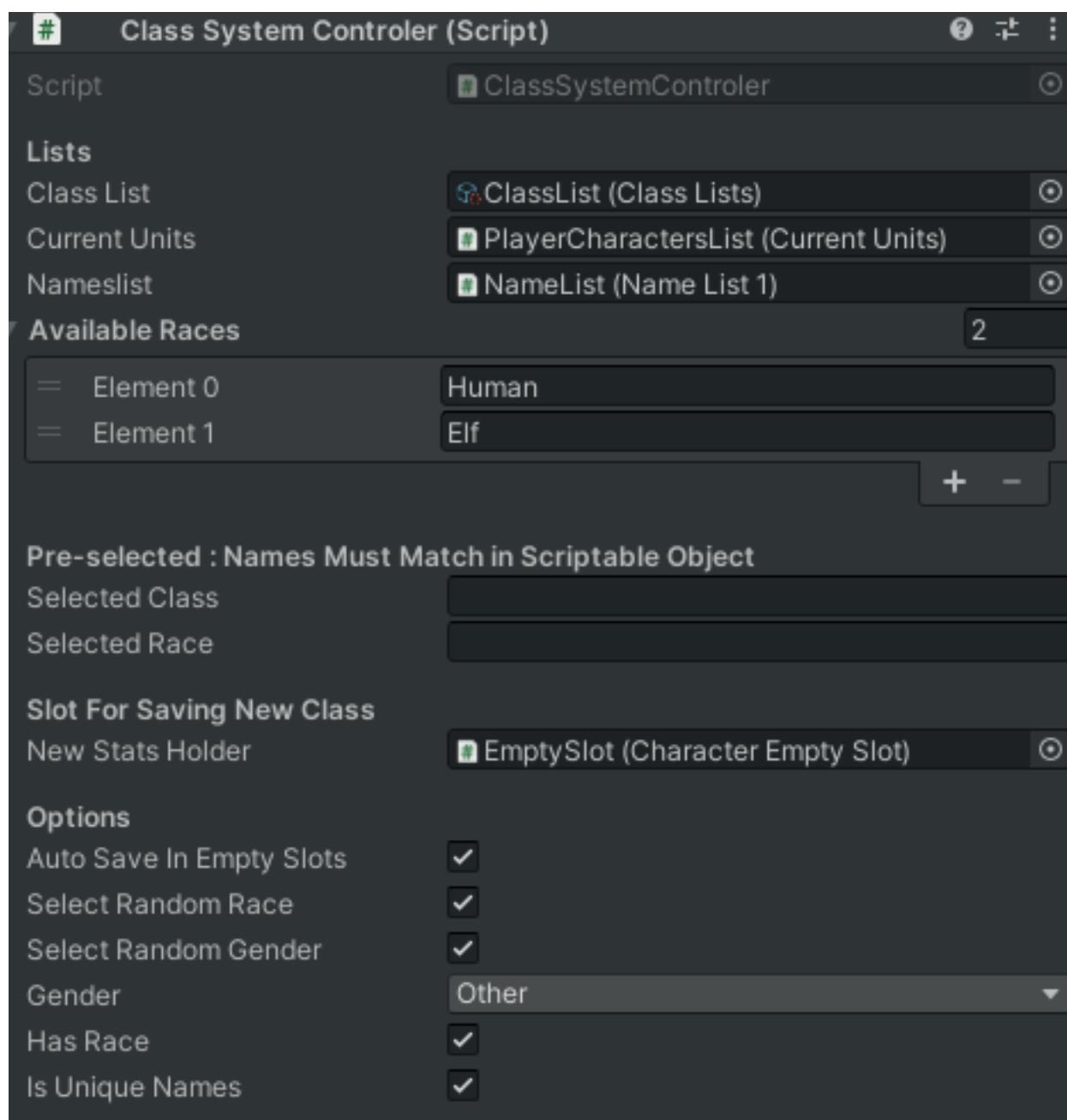
# 5 Controller

## 5.1 - Settings

You can use the system by using the ClassSystemManager Prefab already available.

An important note is that the Available Races Names MUST MATCH the races in name in the race List.

The system is prepared to be used with minimal changes but can be easily expanded.



# **6 Demo content**

## **6.1 - Demo**

A demo is included and you can adapt it to your own project, it includes a character save system, a stats display and how to create specific characters.