RPG Character Classes

Enemy health is now being affected by Gameplay Effects (although it’s only a placeholder, since we want damage to be affected by the secondary stats for a real RPG combat experience)

Also, character is just using the default attribute value; in a real RPG different character types start with different attributes

These character types are often separated into different character classes

A screenshot of a video game

Description automatically generated

To properly initialise enemy attributes we should think about how we will do this; Aura will level up and get stronger; we also have enemy class level (a class variable) and for an enemy to spawn in at a different level that should **mean** something – higher level enemy = higher attribute values

So, how to initialise attributes for a character based on it’s character class?

Warrior: mainly melee attacks

Rangers: ranged attacks

Elementalist: magical spells

So characters should have a way to easily set their character class, which determines their starting attributes. Starting Level should also affect starting attributes

Loading in a character with a certain class and level is mostly related to enemy characters; Aura starts with base stats, unless being loaded from a save

Enum perhaps?

An asset to store data…a data asset

Gameplay effects can use curve tables to scale gameplay effect magnitudes based on level

Curve tables are a single asset that can hold multiple curves, like for Primary Attributes for example

A screenshot of a video game

Description automatically generated

So each character class, specified with character class Enum, can have a curve table with curves for each primary attributes storing the starting values that can scale up as level increases

Will also apply a Gameplay effect to initialise the attributes, then populate the secondary attributes based on the primaries, and an effect for Vitals once everything else is set

Finally any Abilities the enemy character class has should be in this data asset so they can granted at the beginning of the game

Some of the enemies may have the same abilities and effects eg Death or hit react as an ability and each enemy share those ability classes, so the data asset should also have all abilities and effects that all enemies should be given

A screen shot of a black and yellow text

Description automatically generated

6: could be in the AuraAbilitySystemLibrary, take in the Asset, ASC, and character level and apply all effects to initialise attributes

# Character Class Info

A screenshot of a computer program

Description automatically generated

Asset created, now need to store data for each character class and Enum to categorise classes

A screen shot of a computer

Description automatically generated

We can add constants for each character class

A screen shot of a computer program

Description automatically generated

Now we can distinguish between character classes

We also need a struct, with all the information for each class

New struct for the character class default info:

A screenshot of a computer program

Description automatically generated

In this struct we’ll need a gameplay effect to apply the primary attributes

A screen shot of a computer program

Description automatically generated

Now primary attributes can be set here

We could have Secondary and Vital gameplay Effects per class, or share the same Secondary and Vital among all classes

Let’s share them outside of the struct, in the main body

A computer screen shot of a program code

Description automatically generated

The asset now needs a way to store the structs, one for each class.

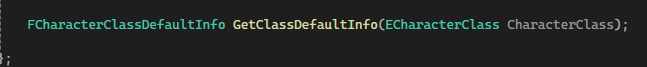
TArray or Map?

Map, so we can map the Enum to the struct

A screen shot of a computer

Description automatically generated

We want to retrieve the information for a given enum constant, so we’ll make a function to look up the info



A screen shot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

FindChecked is good because it will perform an assertion and if no enum we’ll get an assert

New Data Asset BP in Unreal based on the class:

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A black and white screen

Description automatically generated with medium confidence

Now we have a data asset we can create some new Gameplay effects to add to the pulldown menus

# Default Attribute Effects

We have already created default attributes for the Aura Character, but we will need them for the enemies as well

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

The Secondary Attribute Effects could just be the same some shared among all classes unless there was a reason to tweak the coefficients in the calculations made in the modifiers

A screenshot of a computer

Description automatically generated

So this could be set up to be different for each class with a separate GE so for example Warriors get more armor from their resilience or something, but we don’t have to do that

For this exercise I’ll use the same Secondary Attributes formulae and rename GE\_AuraSecondaryAttributes to GE\_SecondaryAttributes

A screenshot of a computer

Description automatically generated

Now we have some Gameplay effects they can be set in the Data Asset

A screenshot of a computer

Description automatically generated