Algorithms

Flamethrower Logic

The use of Animation Notify states was prevalent on the project, this extends to the flamethrower. Usually *Anim Notify States* have 3 events tied to them. The “On begin”, which starts at the beginning and is only executed once, “On End” which works exactly like the previous event but executes at the end of the loop, and finally “On Animation Tick” which runs once every frame while the animation is playing. These events are used in order to start key logic components and effects. The “On Begin” event is used to start the particle flame effects, and the end event is used to end them. The tick update is the really important one, its used to check if anything is hitting the Collider box for the flame attack, and, if that’s the case it will cause damage to that object, the damage of course is multiplied by delta so that the damage is consistent and doesn’t get affected by how many frames the game is running.

Flamethrower attack blocks movement from the player, however it allows the player to rotate to any direction he wants, this is done simply by using the on Begin and On End events from the notify state, the first one blocks movement by stetting a variable to false, and the final one enables movement by setting that same variable to true.

Flammable Objects

Flammable objects could have been done in one of two ways, the first one would be to do an interface and use that interface to have events like “Start Flame” and End Flame. But since that solution would require every flammable object to have the particle system all the same. So that’s why we opted out for the second option, creating a base class for flammables. This way every other object that can be lit on fire will just inherit the base “flammable” class and whenever we want to ignite something, we just check if the object in question can be cast to the base flammable class, if yes, then cast it on fire, if not, then nothing happens.

Ignition Spell

The ignition spell in this project works by launching a projectile of fire and, if it hits something, that thing will start to burn. This is done by using an animation notify, as soon as the animation hits a certain point specified in the animation, a projectile will be spawned on the hand socket of the character, this projectile is a class of its own and when spawned will automatically start moving towards the direction that the player was aiming at when starting the animation.

This is done the following way: \*EXPLAIN SHIT\*