**10. NIAGARA MUZZLE EFFECT**

1. Какие у нас будут два эффекта – главное их отличие в настройках?

2. Какая логика будет в плане спауна для ниагара-системы лаунчера и винтовки (словесная логика)?

3. Начнем с базового класса оружия – что здесь создаем за проперти и функцию и какие два ЗФ подключаем? Какая логика у функции? Что принимает функция спауна системы? Что возвращает?

4. Какую логику реализуем в LauncherWeapon?

5. Какую логику реализуем в RiffleWeapon (проперти и две функции)? Где будем вызывать данные функции? Какая в них логика? Что останется настроить в BP обоих оружий?

1. У нас будет два эффекта – для винтовки и лаунчера. Важно – у лаунчера в нигара-системе в эмиттере значение Loop Behavior = Once – данная система проиграется один раз и уничтожится.

Ниагара-система для винтовки имеет тот же эмиттер, но Loop Behavior = Infinite, то есть данная система частиц будет проигрываться бесконечно долго, пока мы ее сами не удалим.

2. Запрограммируем спаун данных ниагара-систем. Обработка вспышек для лаунчера и винтовки будет отличаться.

Для лаунчера нам надо всего лишь заспаунить вспышку в точке дула, логика для винтовки сложнее – там стрельба происходит по таймеру. Однако ниагара-систему нам надо заспаунить всего один раз – когда мы начинаем стрелять мы устанавливаем Visibility данной систему в true, а когда заканчиваем – в false.

3. Начнем с логики базового класса BaseWeapon. Нам понадобится два forward declaration Для UNiagaraSystem и UNiagaraComponent. Указатель на данный класс (последний) возвращается при спауне системы.

Создаем проперти для ниагара-системы вспышки дула и функцию спауна ниагара-системы:

Изображение выглядит как текст

Автоматически созданное описание



Изображение выглядит как текст

Автоматически созданное описание

Для спауна импактов мы использовали SpawnSystemAtLocation, а в данном случае использовали SpawnSystemAttached – трансформация системы будет изменяться вместе с родительской.

Первый параметр – наша ниагара-система, потом компонент, к которому хотим приаатачить, дальше понятно. EAttachLocation – аналог EAttachmentRules.

Последний параметр AutoDestroy = true. Если ниагара-система не проигрывается бесконечно, то она будет автоматически удаляться из мира игры. Функция возвращает указатель на NiagaraComponent.

4. Переходим в LauncherWeapon – здесь нам надо всего лишь вызвать функции SpawnMuzzleFX() в MakeShot(). При этом нас не интересует возвращаемый указатель, так как у нас стрельба дискретная.

Изображение выглядит как текст

Автоматически созданное описание

5. В RiffleWeapon нам надо будет ввести проперти для возвращаемого указателя, потому что мы будем спаунить ниагара-систему один раз, а так же две дополнительные функции для улучшения читабельности кода:

Изображение выглядит как текст

Автоматически созданное описание

Первая будет спаунить ниагара-систему, если наш указатель нулевой, и устанавливать видимость системы в true. Вторая функция будет отвечать за Visibility нашей ниагара-системы.

Изображение выглядит как текст

Автоматически созданное описание

Вначале проверяем, если наш указатель нулевой, то мы должжны создать данную систему. Функцию будем вызывать каждый раз, когда начинается стрельба, поэтому нужно выставить видимость нашей системы в true.

Изображение выглядит как текст

Автоматически созданное описание

В функции установки видимости мы первым делом вызовем функцию SetPaused – данная функция останавливает или запускает ниагара-систему – то есть влияет на ее Tick. Вторая функция SetVisibility уже нам знакома.

Изображение выглядит как текст

Автоматически созданное описание

Каждый раз, когда мы начинаем стрелять, мы вызываем функцию InitMuzzleFX() – при этом система будет заспаунена всего один раз. Во все остальные разы будет меняться только Visibility. А когда заканчиваем стрелять – ставим Visibility в False.

Остается только установить соответствующие ниагара-системы для винтовки и для лаунчера.