**7. Animation Blueprint**

1. Где содержится скорость персонажа (компонента)? Как называется параметр. Какая есть функция для получения скорости у любого актора? Что возвращает? Как получить длину этого вектора?

2. Как создать анимационный блюпринт (два способа)? Какие два графа есть в редакторе анимационного блюпринта, для чего какой и где мы что настраивали? Что содержат оба сразу по умолчанию?

3. Добавили нашу анимацию бега на анимационный граф и указали в меше нашего персонажа

4. У нас есть две анимации, мы хотим сделать так, чтобы они плавно переходили друг в друга – как это сделали? Что создали (дополнительный тип ассета)? Какая начальная анимация? Как изменили подписи у доп. ассета на оси главной?

5. Как настроили нашу анимацию после этого на анимационном графе?

6. Как настроили теперь передачу скорости в переменной на анимационном графе из ивент графа?

1. Скорость персонажа содержится в его CharacterMovement-компоненте, называется Max Walk Speed – измеряются в unit/s, то есть юнитах в секунды, они же см в с. По умолчанию она равна 600.

Для проверки этого перейдем на Event-граф и воспользуемся функцией Get Velocity, которая есть у каждого актора. Данная функция возвращает текущую скорость актора по каждой из осей (соответственно тип ВЗ – FVector).

Нас интересует длина данного вектора, для этого воспользуемся функцией VectorLength класса Vector:

В функции Tick актора выведен значение данной величины.

Изображение выглядит как текст, электроника, зеленый

Автоматически созданное описание

2. Проверили – удалили. Приступи к созданию **анимационных блюпринтов**. Создаем папку animations.

Создать анимационный BP можно несколькими способами: найти Skeletal Mesh нашего персонажа и через ПКМ выбрать Create->Anim Blueprint, либо же в ContentBrowser кликаем ПКМ и выбираем Animation->AnimationBlueprint и во всплывающем окне выбираем наш SkeletalMesh.

Называем ABP\_BaseCharacter. И открываем его. Открывается специальный редактор Animation Blueprint Editor, всю логику будем настраивать тут.

Здесь существует два графа, первый – Event Graph, он такой же, как и в классическом блюпринте. Здесь мы будем обрабатывать различные ивенты от анимации, так же у него существует функция Update, которая является, по сути, функцией Tick блюпринта.

Второй граф – анимационный граф, отвечает за обработку всех анимаций. В ноду Output Pose попадает финальная анимация, которая будет представлена в текущем фрейме.

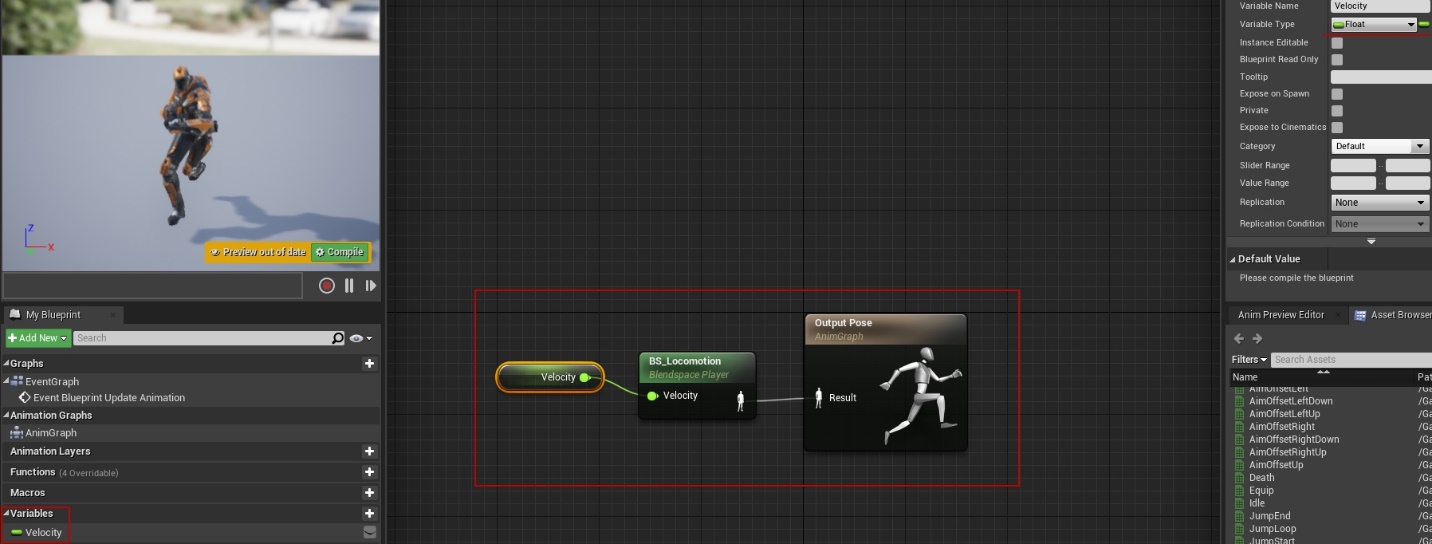
3. В asset browser’e находим нашу анимацию бега вперед и перекидываем drag’n’drop в AnimGraph и выводим в OutputPose. Укажем данный анимационный BP в Mesh’e нашего персонажа.

4. У нас есть анимация Idle (состояние покоя) и анимация Run\_Fwd. Мы хотим сделать так, чтобы при скорости 0 был Idle, а при 600 – Run\_Fwd и был плавный переход между ними.

Для этого мы создаем дополнительный тип ассета с названием Blend Space 1D и назовем ее BS\_Locomotion. Данный ассет служит для смешивания анимаций – для плавного изменения одной анимации к другой. Он интерполирует состояние и выдает промежуточное значение между анимационными позами. Внизу в layout’e для блендинга (смешивания) анимаций у нас ось называется None и мин. и макс. значения равны 0 и 100. Справа в окне Axis Setting меняем значения на нужные нам.

А затем драг-н-дропаем наши две анимации на начало и конец. Мы можем перенести анимацию в любую доступную секцию, добавить еще одну секцию, добавить больше анимаций, у нас нет ограничений в этом плане. С этим закончили.

5. В анимационном BP драг-н-дропаем наш blend space на анимационный граф и подключаем его в Output Pose. Однако нам надо передать текущую скорость актора, для этого создаем новую переменную Velocity, меняем тип на float, выкидываем на граф и подключаем к blend space.



6. Последний шаг – присвоить данной переменной текущую скорость актора. Для этого в Event графе будем устанавливать данную переменную в функции Update Animation (в тике данного BP). Здесь уже есть функция Try Get Pawn Owner, она возвращает ссылку на текущий инстанс пауна. Мы воспользуемся аналогичными функциями Get Velocity и Vector Length и присваем выходное значение нашей переменной.

