**14. SHOW ANIMATIONS / meta = (BindWidgetAnim)**

1. Перепишем блюпринтовый запуск анимации для получения урона через C++. Что такое BindWidget на самом деле? Какой мы использовали элемент enum’а? Добавили проперти и определили новое поведение в функции STUPlayerHUDWidget – какое и где?

2. Зашли в эдитор, там предупреждение и ошибка. Из-за чего предупреждение и ошибка? Как пофиксили ошибку и что означает данный параметр?

3. Добавим во все основные виджеты в нашем проекте анимации старта. Что для этого создали и добавили туда проперти с функцией – какие? Далее у классов переопределили базовый класс – как? Что переопределили в MenuHUD (поведение в одной функции) и в GameHUD (два проперти исправили и поведение в двух функциях)?

4. Теперь везде у нас ошибки компиляции – почему? Какие добавили кому анимации?

1. Перепишем блюпринтовый запуск анимации для получения урона через C++. На самом деле, BindWidget – это одно из значений enum’а, в нем имеется значение BindWidgetAnim. Если мы его укажем, то сможем получить доступ к анимации:

Изображение выглядит как текст, внутренний, снимок экрана

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

При получении ущерба, мы вызывали ивент, на который подписывались в блюпринтах, и запускали анимацию. Теперь сделали это программно. Ивент нам теперь не понадобится. Однако не будем его удалять.

2. Если мы сейчас скомпилируем проект, то у нас выпадет ошибка и предупреждение. Предупреждение связано с тем, что мы не указали проперти для того, чтобы блюпринт мог прочитать нашу анимацию. Однако нам это уже не надо, чистим граф.

Ошибка же нам говорит, что проперти анимации не имеет спецификатора Transient. При указании данного спецификатора значение проперти не будет сохраняться на диск и на старте игры будет всегда иметь дефолтное значение. Добавим его:

Изображение выглядит как текст, оранжевый, закрыть, снимок экрана

Автоматически созданное описание

Смысл данного проперти в чем – если мы создадим актора с каким-нибудь проперти, дефолтное значение которому зададим в блюпринтах (или плюсах), а затем добавим экземпляр данного актора на сцену и изменим данное значение на другое, то с модификатором Transient у нас будет выводиться дефолтное значение (будет игнорироваться то, которое мы указали у экземпляра), в противном случае будет отображаться указанное значение.

3. Добавим во все основные виджеты в нашем проекте анимации старта. Для этого создадим базовый класс STUBaseWidget наших виджетов, в котором будет проперти анимации и функция для запуска анимации:

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

Далее надо у классов PlayerHUDWidget, PauseWidget, GameOverWidget и MenuWidget (так как у них будет проигрываться анимация) поменять класс на наш (от которого они наследуются), заменяя наследование и заголовочный файл:

Изображение выглядит как текст

Автоматически созданное описание

Так же везде, где мы генерировали виджеты, нужно поменять параметр в шаблонной функции на наш базовый класс – в MenuHUD:

Изображение выглядит как текст

Автоматически созданное описание

В GameHUD:

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

4. Теперь у нас везде ошибки компиляции у блюпринтов виджетов. Menu-виджету добавим картинку на весь экран, которой сделаем черный цвет и анимацию выцветания (уменьшая Opacity), а так же в Visibility картинке установили значение Not Hit-Testable – события мыши не будут срабатывать для данного UI-элемента – поэтому картинка не будет блокировать наши кнопки. Обязательно анимации давать такое же название, как в классе BaseWidget – ShowAnimation.

В PlayerHUDWidget добавим анимацию, при которой все элементы (за исключением прицела) будут выезжать из-за краев. Анимируем параметр Transform.

Далее добавили анимации в GamePause и GameOver виджеты. В них увеличиваем BlurStrength и меняем трансформацию.