**Содержание**

[**Введение** 1](#_Toc498770321)

[**1. Теоретическая часть** 3](#_Toc498770322)

[**1.1 Что такое презентация и чем она не является** 3](#_Toc498770323)

[**1.2 Основные сведения об Unreal Engine 4** 4](#_Toc498770324)

[**1.3 Основные инструменты** 6](#_Toc498770325)

[**1.3.1 Режимы размещения** 6](#_Toc498770326)

[**1.3.2 Контент-браузер** 8](#_Toc498770327)

[**1.4 Инструмент визуального скриптинга Blueprint** 8](#_Toc498770328)

[**2. Практическая часть** 10](#_Toc498770329)

[**2.1 Создание уровня** 10](#_Toc498770330)

[**2.2 Создание PBR материала** 10](#_Toc498770331)

[**2.3 Установка освещения** 10](#_Toc498770332)

[**2.4 Привязка анимаций к скелету персонажа** 10](#_Toc498770333)

[**2.5 Добавление клавиш перемещения** 10](#_Toc498770334)

[**2.6 Добавление функционирующих дверей** 10](#_Toc498770335)

[**2.7 Добавление графического пользовательского интерфейса** 10](#_Toc498770336)

[**Заключение** 10](#_Toc498770337)

[**Список литературы** 10](#_Toc498770338)

[**Приложения** 10](#_Toc498770339)

# **Введение**

# **1. Теоретическая часть**

## **1.1 Что такое презентация и чем она не является**

Презентация – это практика показа и объяснений материала для аудитории. Другими словами, это способ более доступно и наглядно рассказать какой-либо материал аудитории. Любая презентация состоит из двух составляющих:

* Первая – это текст, с которым нужно выступать. Это главная часть презентации, так как весь смысл нужно передать устно;
* Второй – это набор слайдов, который помогает наглядно передать суть выступления.

Основные ошибки при выступлении и составлении презентации:

1. Повернутый спиной докладчик. Причиной поворота спиной может служить незнание своего текста, поэтому докладчик стремится поместить максимум текста на слайд и зачитать его оттуда;
2. Каждый новый слайд с новым эффектом смены слайда. При пояснении материала лишь отвлекает внимание от содержимого;
3. Чрезмерная анимация. Постоянно движущиеся объяты не дают сосредоточиться;
4. Разные фоны у каждого слайда;
5. Безудержная разноцветность и пестрота.

Главные правила хорошей презентации:

1. Никаких лишних деталей. Обобщение материала для краткости и ясности;
2. Единый стиль. Для создания единой работы;
3. Читаемые шрифты;
4. Адекватные цвета;
5. Использовать макеты расположения текста. Данные макеты позволят грамотно расположить нужное количество содержимого на слайде.

## **1.2 Основные сведения об Unreal Engine 4**

Unreal Engine 4 – это движок нового поколения для создания видеоигр. Этот движок предоставляет большой набор инструментария для создания 2D и 3D проектов. Это могут быть игры, архитектурная визуализация и различные демки или видеоролики. Unreal Engine 4 распространяемся бесплатно. Однако до тех пор, пока вы не выпустите свой первый коммерческий продукт на основе UE4.

Системные требования Unreal Engine 4:

* Настольный ПК или Mac
* Windows 7 64-bit или Mac OS X 10.9.2 или старше
* Четырехядерный процессор Intel или AMD, 2.5 GHz или лучше
* NVIDIA GeForce 470 GTX или AMD Radeon 6870 HD или лучше
* 8 Гб ОЗУ

Этот движок доступен для создания проектов для Windows PC, Linux, iOS, Mac, Android. Также есть встроенная поддержка виртуальной реальности для Oculus Rift. Помимо этого, UE4 поддерживает Xbox One и PlayStation 4.

Для того чтобы скачать Unreal Engine 4 нужно зайти на официальный сайт производителя и скачать лаунчер. После установки лаунчера нужно зарегистрироваться и подтвердить свою почту. В списке доступных приложений выбрать Unreal Engine 4 и место для скачивания, на данный момент самая актуальная версия это 4.18.

Для того чтобы начать создавать проект, нужно его создать. При открытии Unreal Engine 4 предлагает выбрать способ программирования BluePrint или С++, шаблон видеоигры или выбрать пустой бланк. После выбора шаблона, программа предлагает выбрать место хранения проекта, его название, платформу и начальный контент, включающий в себя стандартный набор 3D моделей, материалов и архитектурные модели. Проект представляет собой автономный блок, который содержит все содержимое и код, которые составляют индивидуальную игру и совпадает с набором каталогов на вашем диске. Например, на рисунке 1 изображено Древо иерархии браузера содержимого, который содержит ту же структуру каталогов, которая находится внутри папки Project на диске.

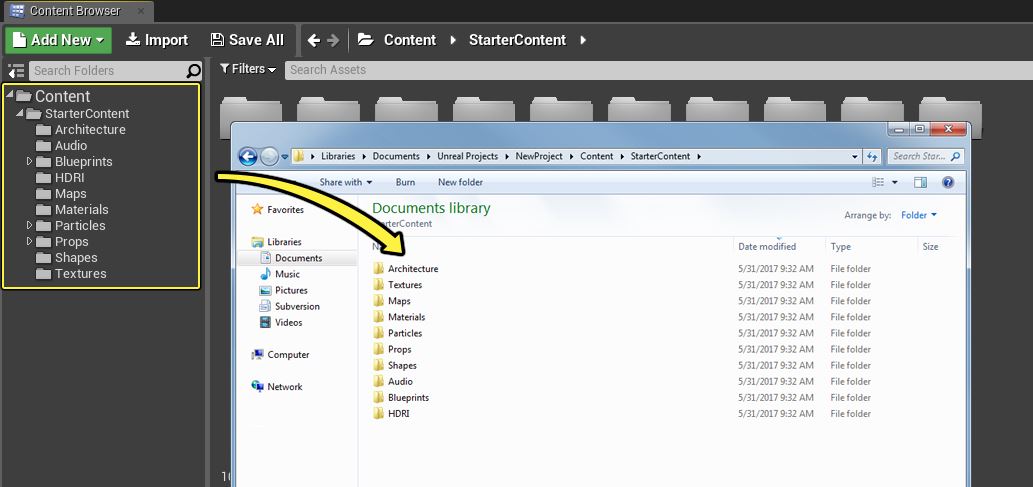


Рис 1. Древо иерархии

Возможно создать любое количество различных проектов, которые могут поддерживаться и развиваться параллельно. Unreal Engine может легко переключаться между ними, что позволит одновременно работать с несколькими играми или иметь несколько тестовых проектов. В Unreal Editor сцены, в которых создается игровой опыт, обычно называют уровнями. Любой объект, помещённый в мир, будь то свет или персонаж, считаются актером. Actor – это класс программирования, используемый в Unreal Engine для определения объект, который имеет данные о трехмерном положении, вращении и изменении масштаба. Для того чтобы видеть, что создается в Unreal Editor, существуют видовые экраны. Viewport – это окно для управления уровнем. Видовые экраны содержат множество инструментов и визуализаторов, которые помогут точно определить нужные данные.

## **1.3 Основные инструменты**

Для размещения любого объекта, в Unreal Editor есть специальная панель, которая называется “Modes” или режимы. Панель «Режимы» содержит ряд различных режимов работы редактора. Они изменяют основное поведение редактора уровней для специализированной задачи, например, размещения новых актёров в мире, создание геометрических кистей, рисование, создание растений, скульптурирование.

### **1.3.1 Режимы размещения**

Place Mode – это специализированный инструмент для ускорения и упрощения создания среды. Этот режим фокусируется только на тех активах, которые могу быть непосредственно размещены на уровне. Режим Place также облегчает добавление специализированных типов Актеров, таких как Свет. Кроме того, он упрощает создание специальных подтипов актеров, таких как создание субъектов физики из выбранных статических сеток. Для того чтобы разместить актера из режима Place, нужно открыть проект и в окне режимы выбрать Place Mode (1), далее нужно перейти на вкладку «Основные» (2), затем щелкнуть левой кнопкой мыши и перетащить актер (3) на уровень (рис 2).

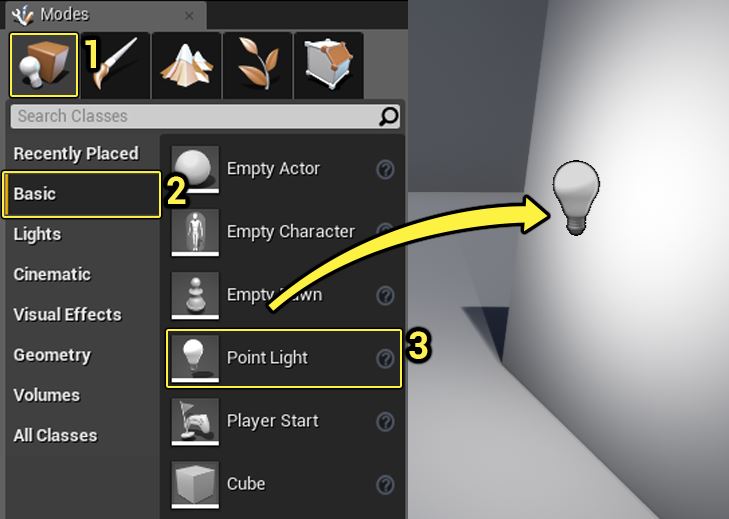


Рис 2. Размещение специального актера «Свет»

Режим Mesh Paint – обеспечивает быстрый и простой способ регулировки цвета и материала на актерах.

Режим Landscape tool – эта система позволяет создавать рельеф местности для уровня – горы, долины, неровную или наклонную землю, даже отверстия для пещер. В этом режиме существуют три инструмента для настройки пейзажа.

* Инструмент Manage – позволяет создавать новые ландшафты и изменять ландшафтные компоненты.
* Инструмент Sculpt – позволяет изменить форму ландшафта, добавить впадины, горы и пещеры.
* Инструмент Paint – позволяет изменить внешний вид частей пейзажа, нанести на него материал на основе слоёв.

Режим Foliage – эта система позволяет быстро нарисовать или стирать наборы статических ландшафтных актеров. Этот режим содержит пять инструментов.

* Инструмент Paint – этот инструмент предназначен для размещения мелких актеров на большой площади, например, травы.
* Инструмент Reapply Tool – позволяет выборочно изменить определенные параметры для актеров, уже размещенных на уровне.
* Инструмент Select – позволяет выбирать единичные размещенные объекты.
* Инструмент Lasso – позволяет одновременно выбирать количество большого количества актёров.
* Инструмент Fill – позволяет залить всю область статической сетки ландшафта различными актерами.

Режим Geometry Brush Actor – геометрические кисти являются самым основным инструментом для построения уровней в Unreal.

На базовом уровне создания уровней всё сводится к размещению элементов на карте. Эти предметы могут представлять собой геометрию, украшения в виде кистей, статических сеток, света, старта игрока, транспортных средств.

### **1.3.2 Контент-браузер**

Контент-браузер является основной областью в Unreal Editor для создания, импорта, организации, просмотра и изменения содержимого активов в Unreal Editor. Он также предоставляет возможность управлять папками содержимого и выполнять другие полезные операции с активами, такие как переименование, перемещение, копирование и просмотр.

## **1.4 Инструмент визуального скриптинга Blueprint**

Система визуальных сценариев Blueprints в Unreal Engine представляет собой полноценную систему сценариев игры, основанную на концепции использования интерфейса на основе узлов создания элементов игрового процесса. Эта система чрезвычайно гибкая и мощная, поскольку она позволяет дизайнерам использовать весь спектр инструментов, которые обычно доступны только программистам. Кроме того, специфичная Blueprint разметка, доступная в реализации Unreal Engine C++, позволяет программистам создавать базовые системы, которые могут быть расширены. Blueprints – это визуальное дополнение к вашей игре. При подключении узлов, событий, функций и переменных. С помощью связей можно создавать сложные элементы игрового процесса. Чертежи работают с использованием графиков и узлов для различных целей: построения объектов, отдельных функций и общих событий игрового процесса, которые специфичны для каждого чертежа. Есть два наиболее распространённых типа чертежа, это «Чертеж уровня» и «Чертеж класса»

* Чертеж уровня – представляет собой специализированный тип, который действует в качестве глобального графа событий в масштабах уровня. Каждый уровень в проекте имеет свой чертеж. Чертежи уровня также обеспечивают механизм управления потоками уровня и Matinee, а также привязки событий к актерам, размещенным на уровне.
* Чертеж класса – Классы Blueprint идеально подходят для создания интерактивных активов, таких как двери, переключатели, предметы коллекционирования и разрушаемые пейзажи. На изображении выше кнопка и набор дверей являются отдельными чертежами, которые содержат необходимый сценарий, чтобы реагировать на события перекрытия игрока, анимировать их, воспроизводить звуковые эффекты и изменять их материалы (кнопка загорается при нажатии, например).

В этом случае нажатие кнопки активирует событие внутри двери Blueprint, заставляя его открываться - но двери можно так же легко активировать с помощью другого типа Blueprint, либо с помощью последовательности Blueprint. Из-за самодостаточного характера чертежей они могут быть сконструированы таким образом, что вы можете понизить их до уровня, и они будут работать просто с минимальной настройкой. Это также означает, что редактирование Blueprint, который используется во всем проекте, обновит каждый экземпляр этого файла.

Сценарий Blueprint также можно использовать для создания HUD игры, который аналогичен классам Blueprint, поскольку он может содержать последовательности событий и переменные, но присваивается активу GameMode проекта, а не добавляется непосредственно на уровень. HUD можно настроить для чтения переменных из других чертежей и использовать их для отображения панели работоспособности, обновления значения оценки, отображения объективных маркеров и т. Д. Также можно использовать HUD для добавления хит-боксов для таких элементов, как кнопки, на которые можно щелкнуть, или, в случае мобильных игр, ответить на сенсорный ввод.

# **2. Практическая часть**

## **2.1 Создание уровня**

## **2.2 Создание PBR материала**

## **2.3 Установка освещения**

## **2.4 Привязка анимаций к скелету персонажа**

## **2.5 Добавление клавиш перемещения**

## **2.6 Добавление функционирующих дверей**

## **2.7 Добавление графического пользовательского интерфейса**

# **Заключение**

# **Список литературы**

# **Приложения**

## 

## 