

Разработка многопользовательской онлайн игры на игровом движке Unreal Engine 5

Звайгзне Алексей Юрьевич

Приамурский государственный университет имени Шолом-Алейхема

Студент

Аннотация

В данной статье рассматривается процесс создания сетевой игры на игровом движке Unreal Engine 5 с использованием технологии Blueprint, позволяющей сильно сократить время разработки будущего проекта. В статье описаны значения Graph, применяемых в работе узлов и прочие технические особенности процесса создания многопользовательской игры.

Ключевые слова: Unreal Engine 5, Graph, узел, Blueprint, многопользовательская игра.

Development of a multiplayer online game on the Unreal Engine 5 game engine

Zvaigzne Alexey Yurievich

Sholom Aleichem Priamursky State University

Student

Abstract

This article discusses the process of creating a network game on the Unreal Engine 5 game engine using Blueprint technology, which greatly reduces the development time of a future project. The article describes the Graph values used in the operation of nodes and other technical features of the process of creating a multiplayer game.

Keywords: Unreal Engine 5, Graph, Node, Blueprint, Multiplayer game.

Научный руководитель:

Лучанинов Дмитрий Васильевич

Приамурский государственный университет им.Шолом-Алейхема

Старший преподаватель

1 Введение

1.1 Актуальность

На данный момент процесс разработки игр требует большого объема ресурсов, времени, денег, знаний и навыков. Есть множество движков для игр, упрощающих процесс разработки, но сейчас, даже разработка движка игры отнимает много времени и ресурсов, поэтому большинство разработчиков пользуются уже готовыми решениями. Для получения лучшего результата большинство разработчиков покупает заранее ресурсы

для игр, нанимает третьих лиц для сжатия сроков, не отдает процессу отладки и устранения недостатков должного внимания и времени, поэтому множество игр выходит сырыми и с большим количеством проблем, так как, невозможно не проводя тщательный тест проекта добиться его завершения должным образом. Поэтому компания Epic Games [1], разработала для своего движка отдельную технологию Blueprint [2] с учетом минимальных знаний человека о программировании, создавать игры опираясь на логику и простые конструкции.

1.2 Обзор исследований

Б. Левченко в своей статье рассказали о проблемах синхронизации многопользовательских играх, созданных на движке Unreal Engine 4 используя модули искусственного интеллекта [3]. Р. Гордон в своей книге расписал подробности создания игр на Unreal Engine 4 для новичков с нуля, показал примеры игр, концепт создания игр на C++ и технологии Blueprint [4]. В научной диссертации Дж. Несса был реализован проект в жанре MOBA (Multiplayer Online Battle Arena) под названием NORS выполненный на движке Unreal Engine 4 [5]. Дж. Глейзер выпустил практическое руководство по созданию надежных сетевых многопользовательских игр, в которой рассказываются различные этапы через которые проходят движки игр и сами игры, примеры готовых продуктов и проблемы в их реализации [6].

1.3 Цель исследования

Создать прототип игры на игровом движке Unreal Engine 5 в которой игроки смогут подключаться друг к другу.

2 Материалы и методы

Для создания игры используется игровой движок Unreal Engine 5 с технологией Blueprint

3 Результаты и обсуждения

Разработка игры будет происходить на движке Unreal Engine 5 с использованием Blueprint. Для этого необходимо зарегистрироваться на сайте издателя движка Epic Games, скачать и установить фирменный Launcher – Epic Games Launcher, а после пройти процесс авторизации в Launcher'е через уже созданную учетную запись (рис. 1-3). Движок является бесплатным, но за использование созданного продукта на базе движка необходимо платить долю от продаж с расчетом по определенному критерию.

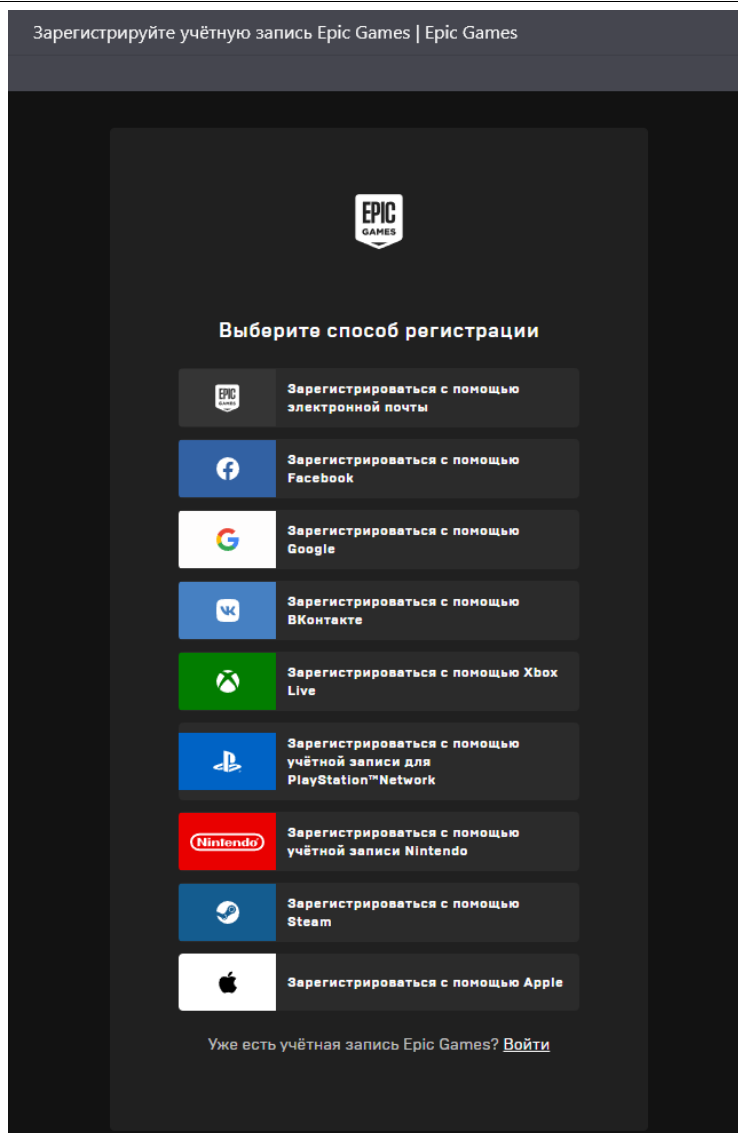


Рисунок 1. Вид окна регистрации на странице магазина

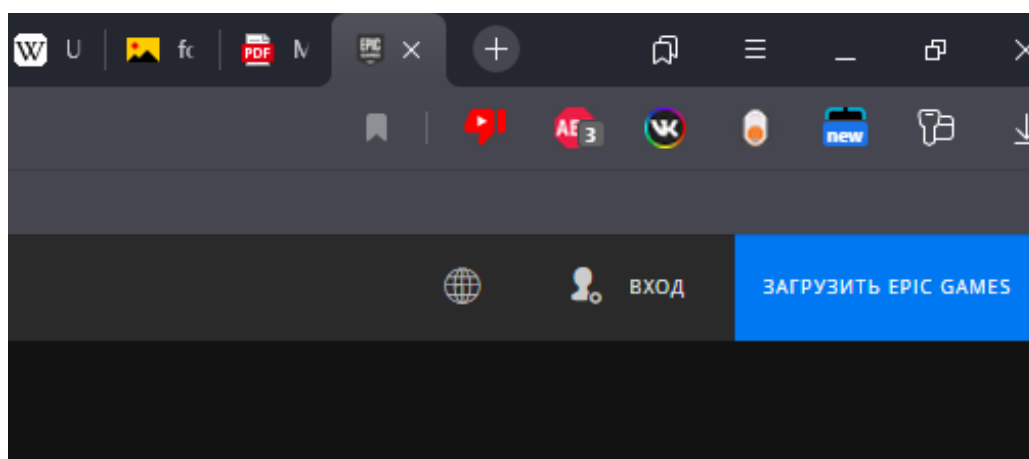


Рисунок 2. Кнопка загрузки клиента

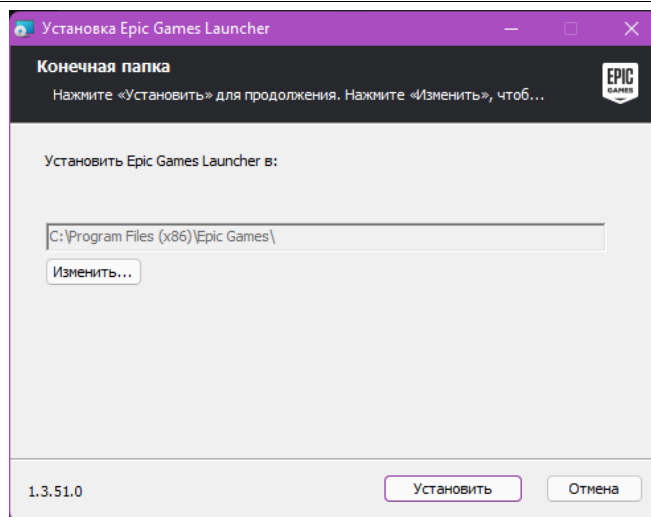


Рисунок 3. Окно установки Epic Games Launcher'a

Авторизация в Launcher'e происходит посредством браузера (рис. 4)

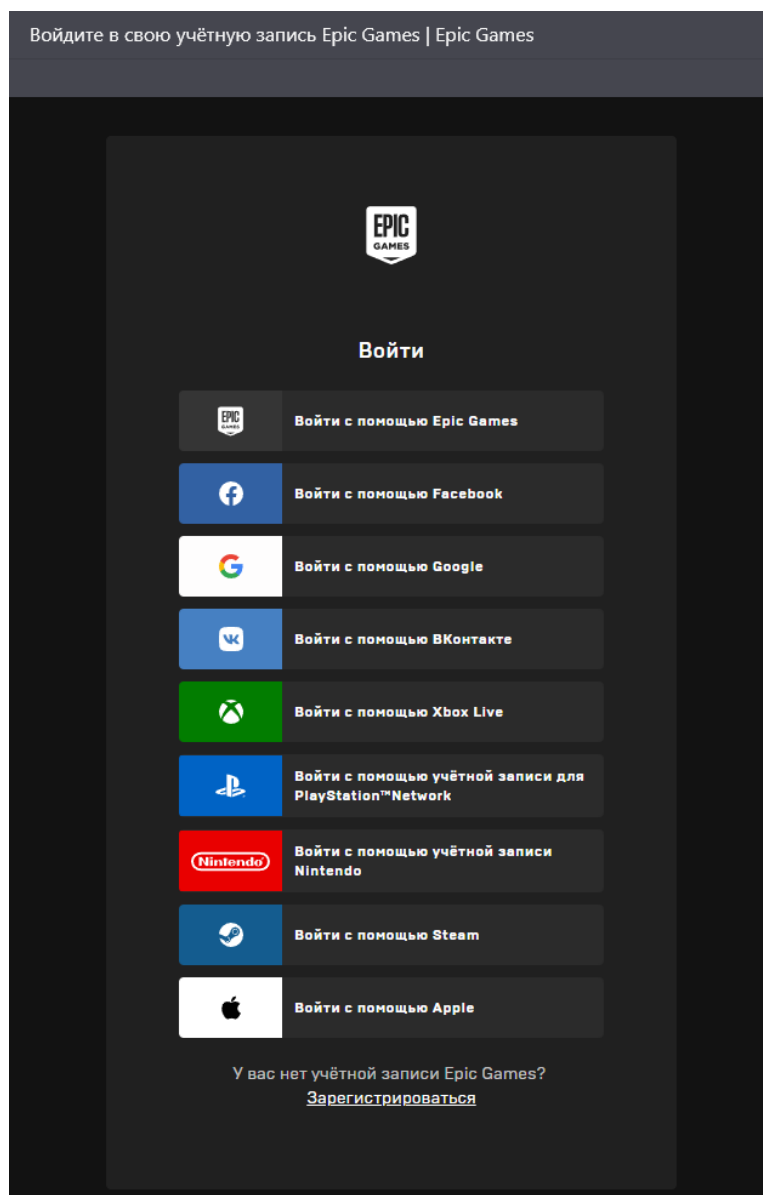


Рисунок 4. Окно авторизации

Далее необходимо скачать и установить сам движок. Попутно принять соглашение пользователя, в котором есть пункт, указывающий на необходимость отдавать часть прибыли с конечного продукта. Каждая версия движка занимает достаточно много места, порядка 60 Гигабайт свободного места. Для более быстрой работы движка рекомендуется устанавливать его на твердотельный накопитель (SSD) (рис. 5).

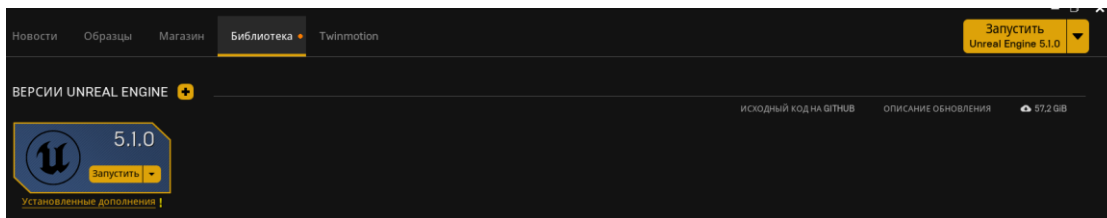


Рисунок 5. Вид окна с установленной версией движка

Для ускорения создания игры, будет использоваться готовый пресет для создания игр от первого лица (рис. 6).

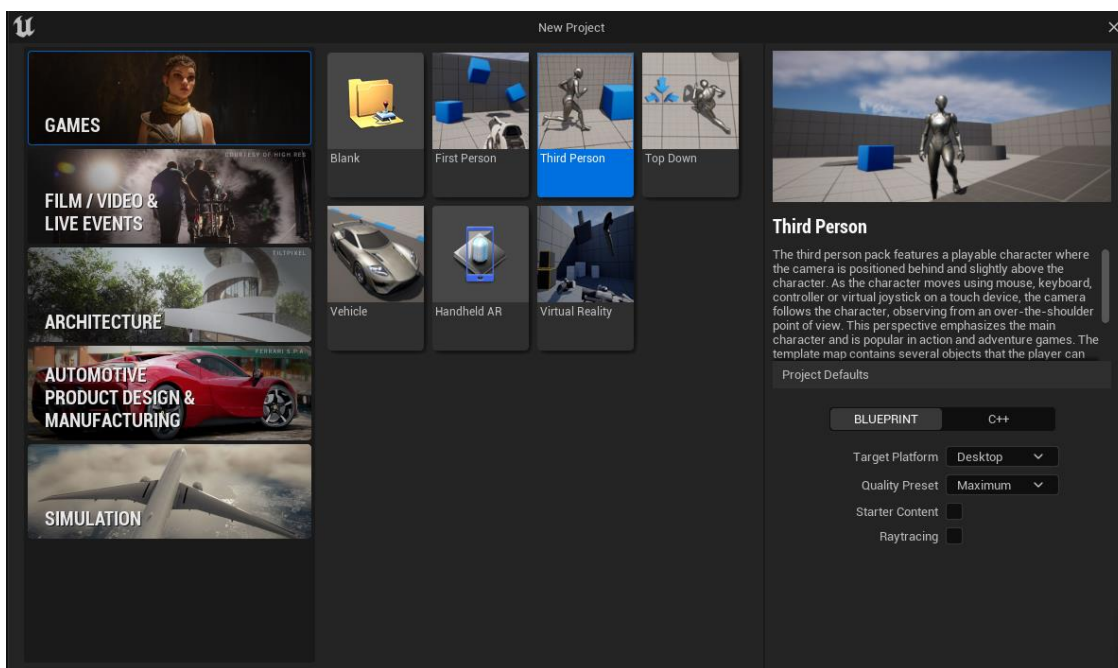


Рисунок 6. Вид окна создания проекта с выбранным пресетом «Third Person»

После создания проекта будет доступно следующее окно (рис. 7). Особенность движка состоит в том, что он «из коробки» поддерживает сетевую игру. Это можно проверить, нажав на троеточие рядом с компиляцией и запуском созданного проекта. В пункте «Number of players» можно указать желаемое число игроков. Из соображений оптимизации, в настройках проекта по умолчанию установлено максимальное количество игроков в размере 4 пользователей.

Также в «Net mode», желательно выбрать «Play As Client», так как игрок создавший сервер будет одновременно и держателем сервера, и его клиентом.

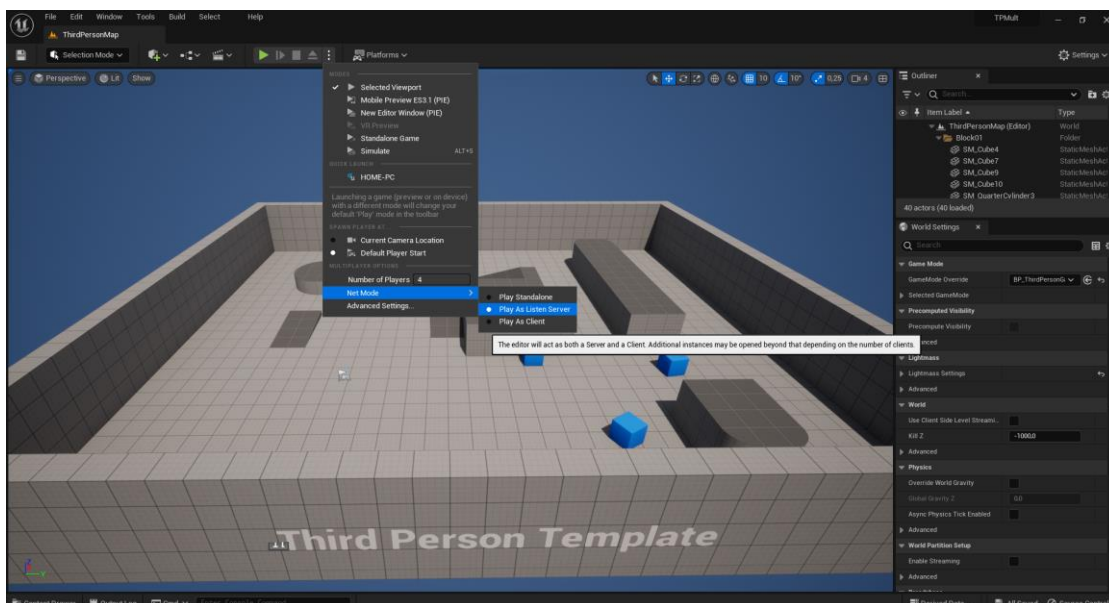


Рисунок 7. Вид окна созданного проекта открытыми настройками эмуляции

Далее нужно создать игровое меню для этого внизу окна в разделе «Content Drawer», правым кликом по свободному месту и создать Level и дать ему подходящее имя, например, «Menu» и открыть уже созданный уровень (рис. 8).

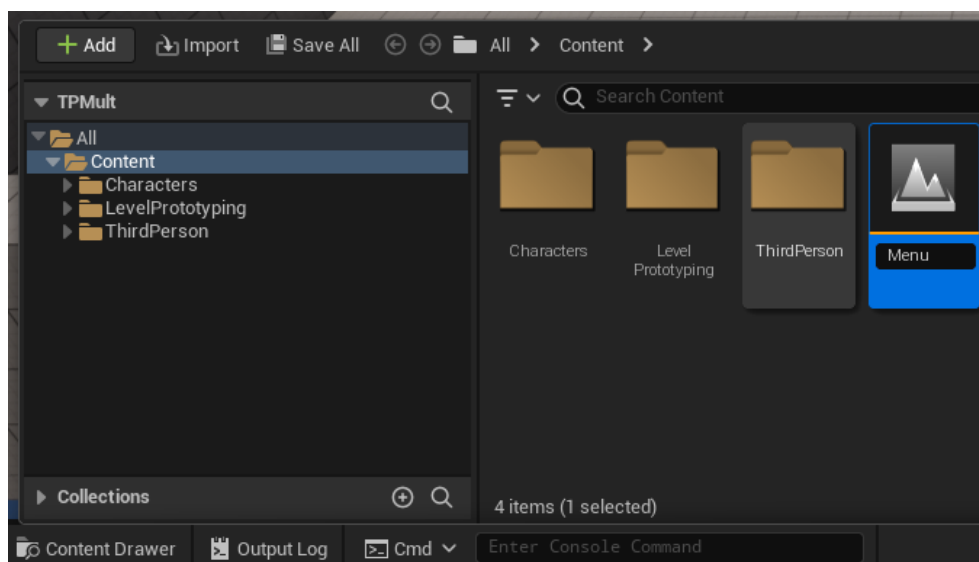


Рисунок 8. Вид панели контента с новым созданным уровнем

Рядом с уровнем нужно создать виджет пользовательского интерфейса для меню. Для этого также щелкнуть правой клавишей мыши в окне контент менеджера «User Interface» - «Widget Blueprint» (рис. 9).

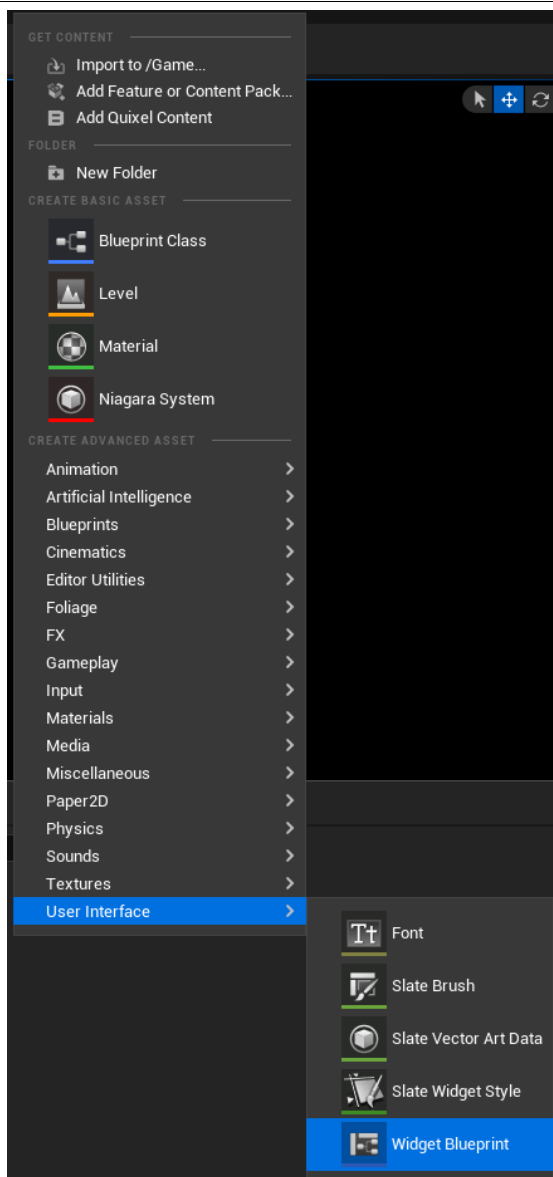


Рисунок 9. Создание виджета для меню

Далее его нужно сразу открыть (рис. 10). И добавить нужные элементы для главного меню, рамку, кнопки и блоки текста:

1. Рамка находится в разделе Pannel и называется Canvas Pannel.
2. Кнопка и текст лежат в разделе Common и имеют соответствующее название Button и Text.

Текст в блоках текста редактируется в левой части экрана в меню Details в пункте Text (рис. 11).

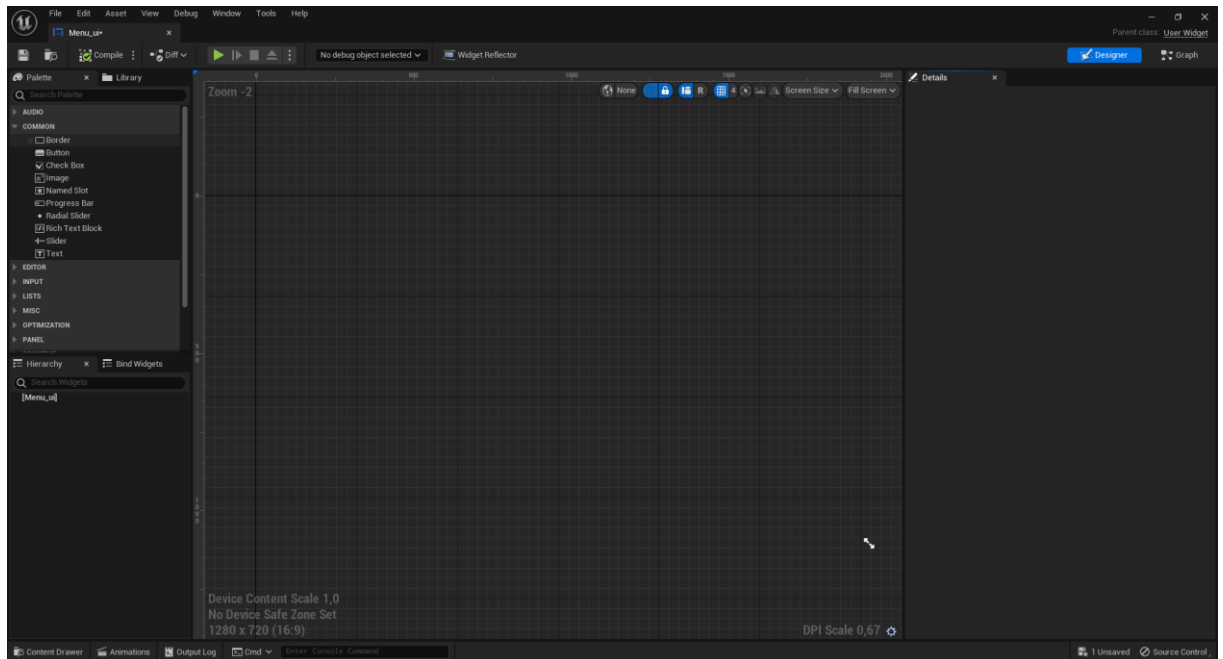


Рисунок 10. Окно с пустым виджетом

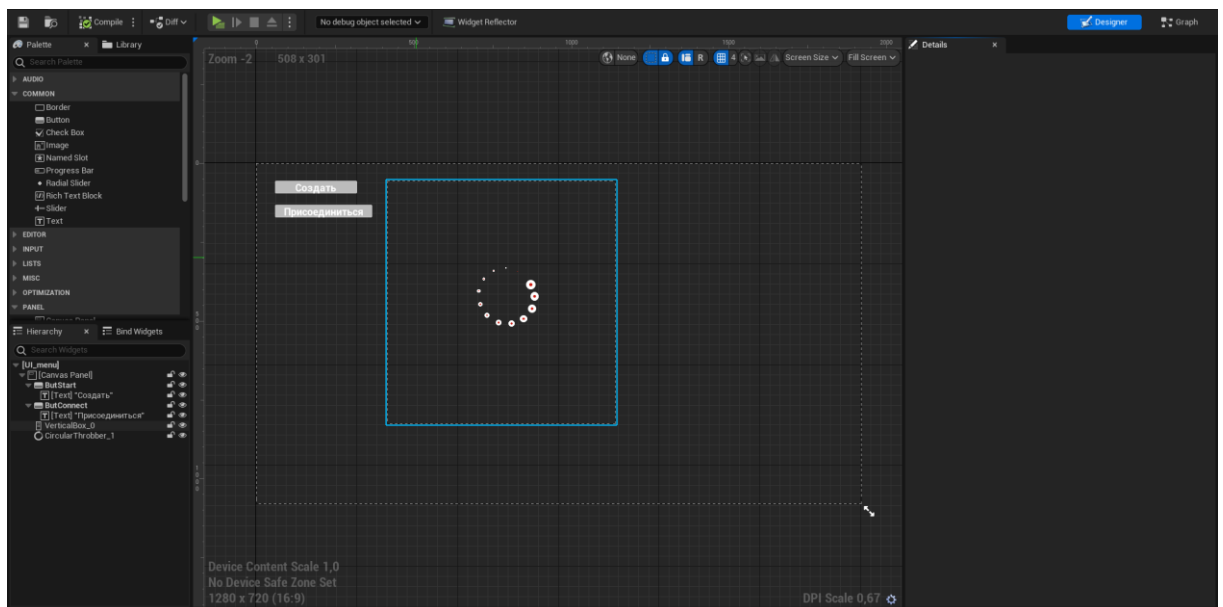


Рисунок 11. Готовая визуальная часть меню

После создания визуальной части, необходимо отработать Graph (логику кнопок и инструментов меню) (рис. 12).

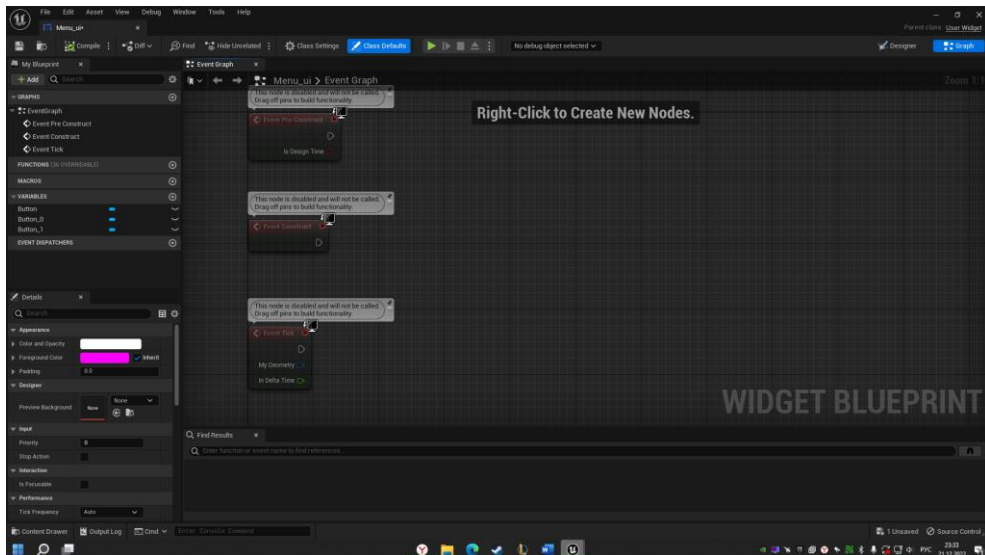


Рисунок 12. Вид окна Graph виджета

Желательно всем кнопкам заранее дать название, описывающее функционал, чтобы не запутаться.

Далее необходимо выбрать нужную кнопку и в левом нижнем меню задать действие по нажатию (OnClicked) (рис. 13).

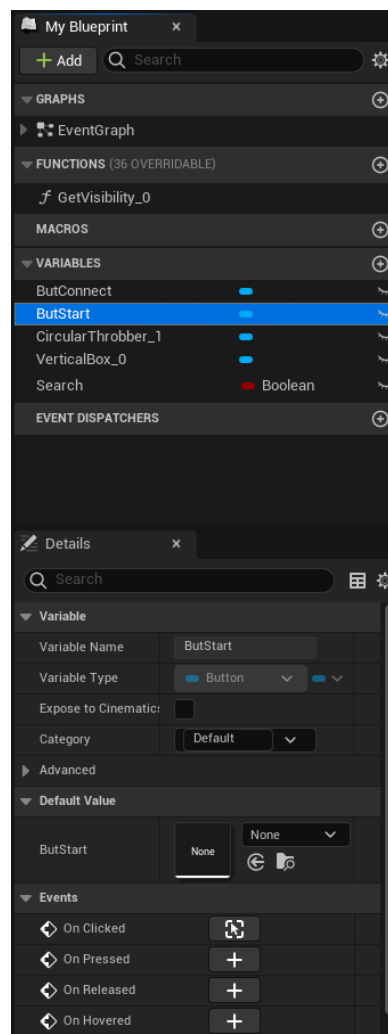


Рисунок 13. Левая панель элементов Graph

Нажатием правой кнопкой мыши по пустому месту необходимо создать указанные узлы (рис. 14). Get Player Controller позволяет передать управление выбранному игроку по параметрам в проекте. Create Session узел создающий игровую сессию. Соединить по ключам и в Public connections указать максимальное количество игроков на сервере равное 4.

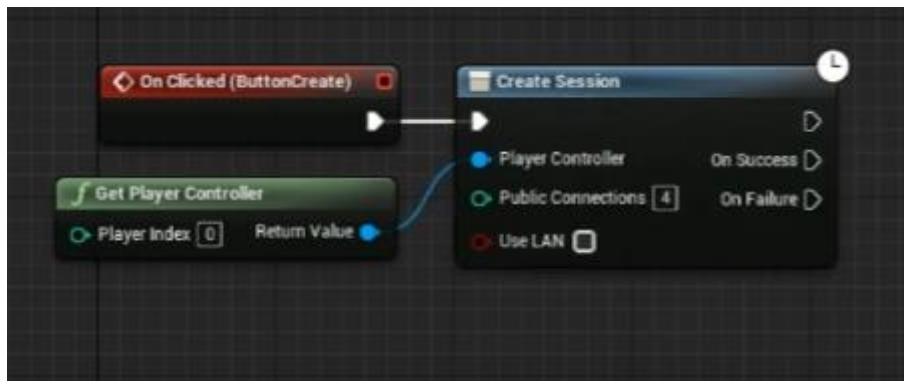


Рисунок 14. Узлы создания сессии

После этого надо добавить следующие узлы на экран и задать соответствующие параметры (рис. 15). Где ServerMap – это название вашей карты. Карта по умолчанию ThirdPersonMap. Open Level (by Name) открывает уровень по имени в проекте, путь указывать не обязательно, недостатком является, проблема дубликации файлов в проекте. Необходимо осторожно давать имена используемым объектам. Set Show cursor отвечает за отображение курсора. Здесь необходимо снять (не ставить) галочку в чек-боксе, чтобы указать его отдельно выше во всем виджете меню. Set Input Mode Game Only отвечает за установку игрового мода исходя из параметров уровня.

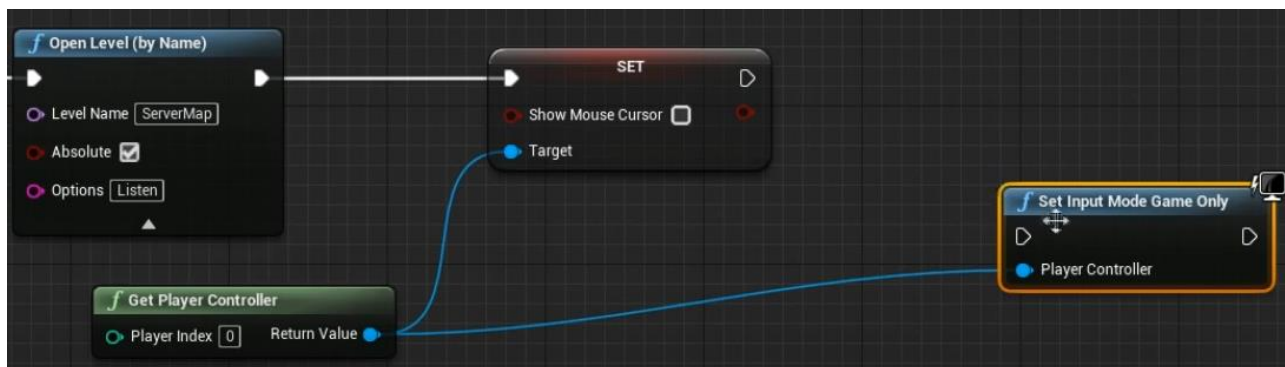


Рисунок 15. Вторая часть Graph для создания сервера

Далее сверху, где уже есть начальный узел Event Construct необходимо снова дать управление игроку и включить курсор мыши (рис. 16).

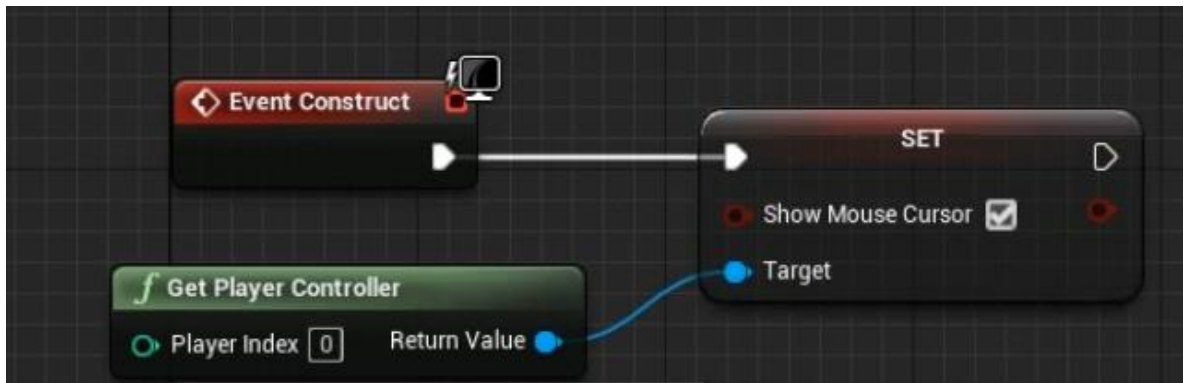


Рисунок 16. Передача управления курсором при использовании виджета

После этого нужно в контенте создать пустой Blueprint Class – GameMode Base. Дать ему имя, желательно связанное с меню и перенести его из контент-менеджера в открытый уровень, где находится меню, в панель World Settings в пункт GameMode Override и выставить следующие параметры в разделе GameMode, это необходимо, чтобы игрок не получал управление манипулятор или другими базовыми объектами (рис. 17)

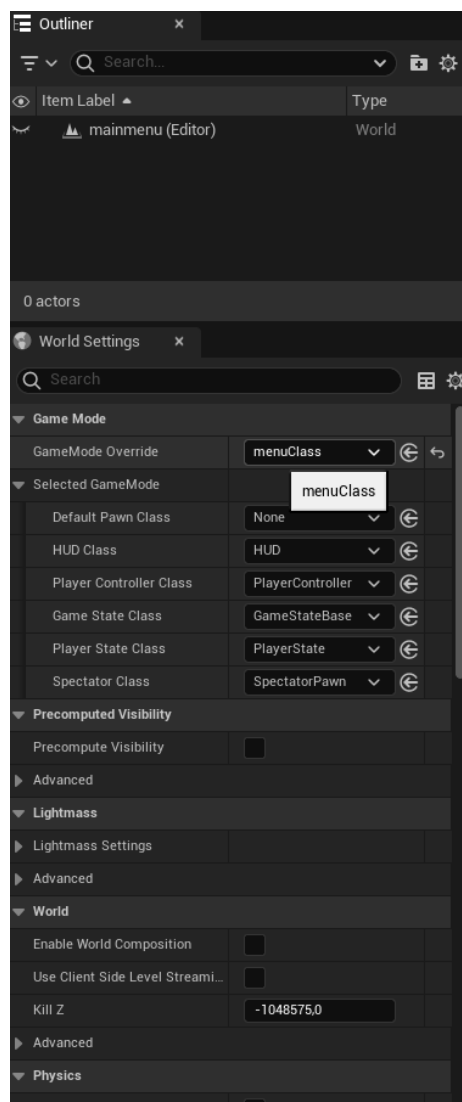


Рисунок 17. Вид панели World Settings

После этого можно открыть класс и начать редактировать. В нем нужно добавить 3 следующие узлы, Create Widget, Add to Viewport, Get Player Controller, которые запускают выбранный виджет на уровне меню (рис. 18).

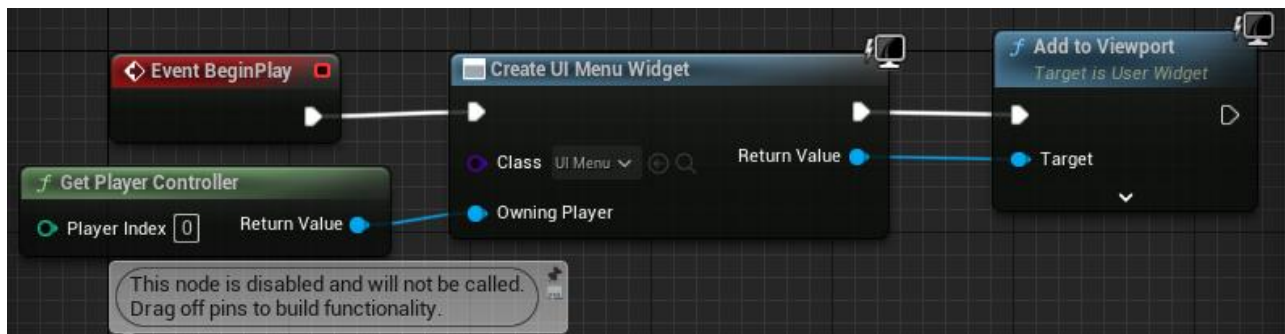


Рисунок 18. Graph, отвечающий за запуск игрового меню

После этого можно пробовать запустить и создать игру. На текущий момент только создаются сервера и нет возможности присоединиться.

Далее на виджет меню нужно добавить вертикальный контейнер (Vertical Box) и анимацию загрузки (CircularThrobber) (рис. 19), у вертикального контейнера в свойствах указать IsVariable (рис. 20).

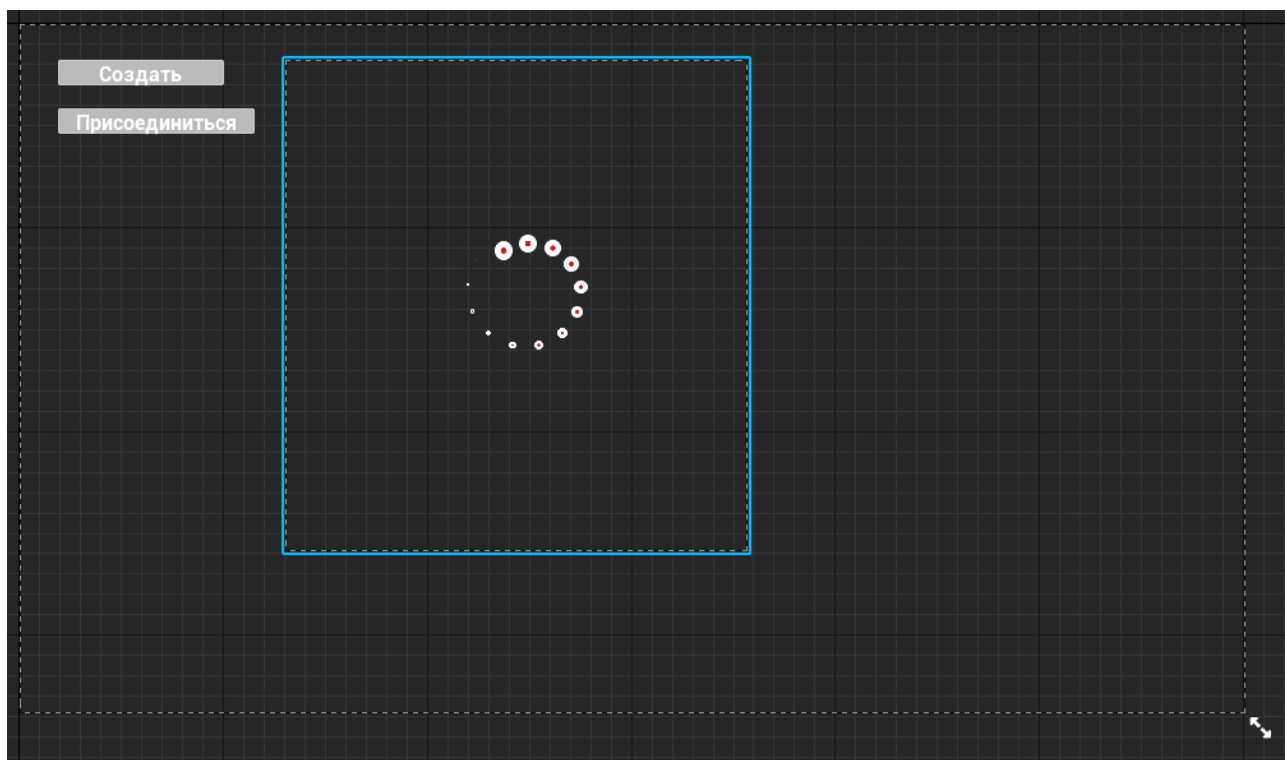


Рисунок 19. Новый вид виджета меню

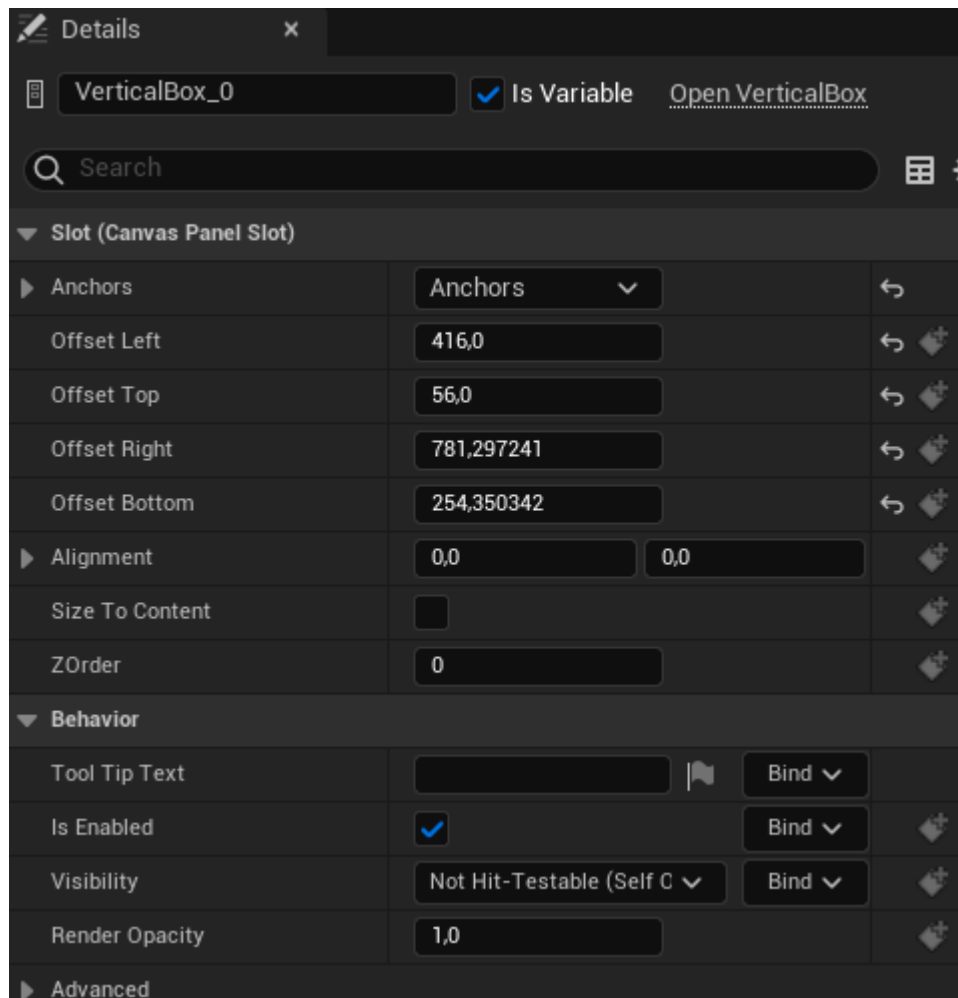


Рисунок 20. Вид окна свой вертикального контейнера

Далее необходимо создать ещё один виджет который будет выводить найденные сессии, сделать размером 600x100 пикселей и добавить туда sizebox, в который необходимо поместить кнопку и текст на кнопку (рис. 21).

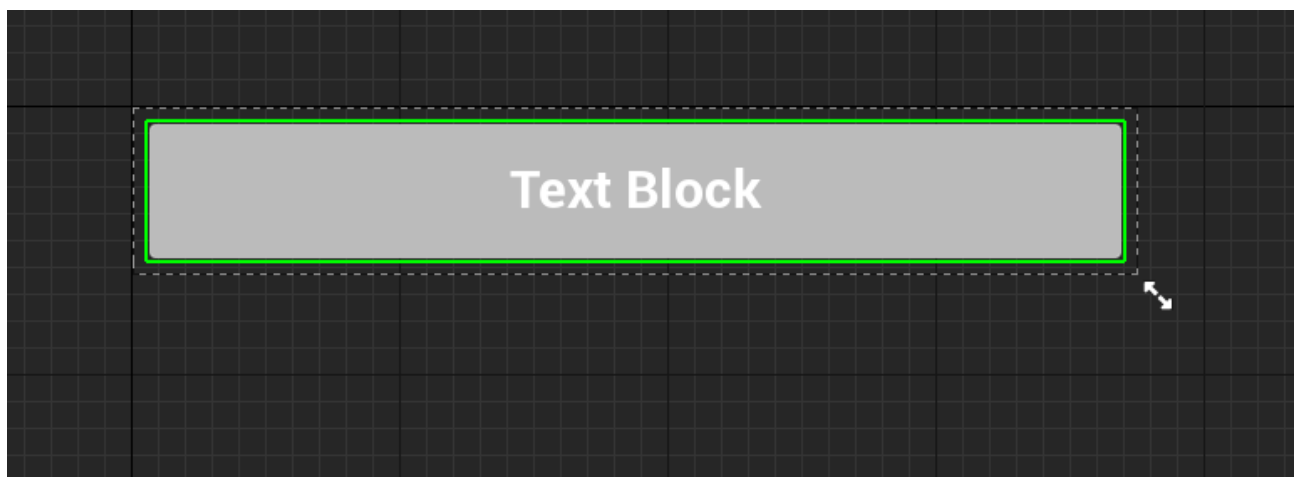


Рисунок 21. Вид виджета кнопки найденной игры

После этого необходимо доработать виджет, в Graph добавить новую переменную, которая будет отвечать за результаты найденных сессий. Для

этого создаем новую переменную в левой части Graph, в разделе переменных (VARIABLES). И указать в типе переменной Blueprint Session Result. В свойствах переменной указать Instance Editable и Expose on Spawn (рис. 22).

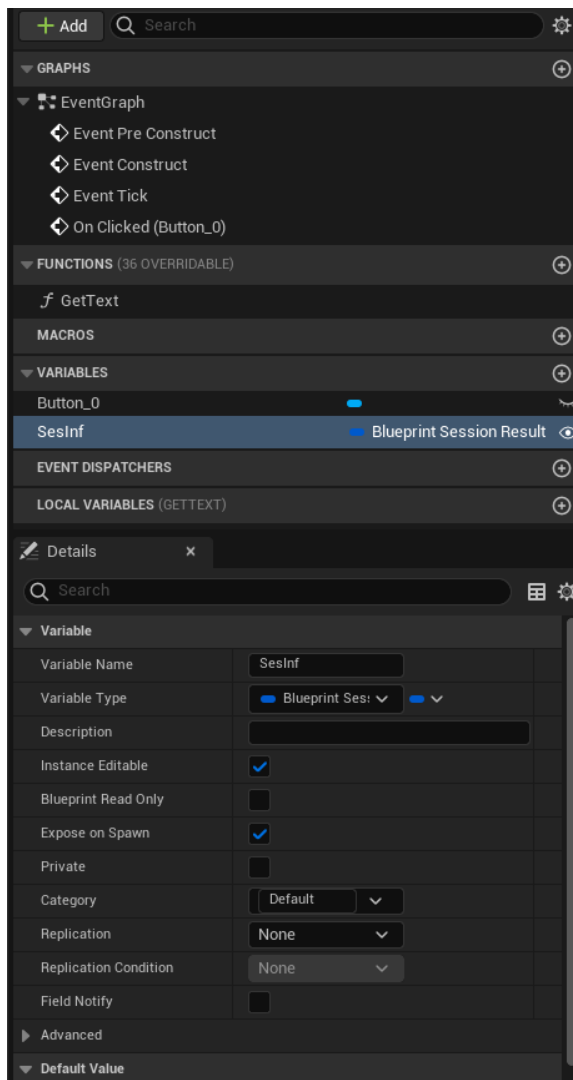


Рисунок 22. Свойства созданной переменной

Нужно создать следующие узлы по нажатию кнопки (OnClicked), созданную переменную через операцию Get, Get Player Controller, Join session после этого их необходимо соединить (рис. 23).

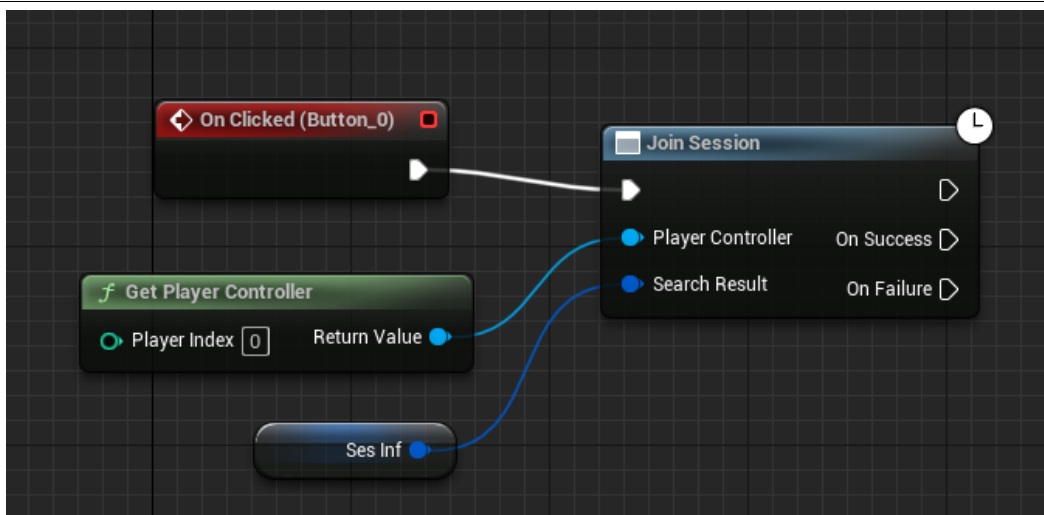


Рисунок 23. Graph отвечающий за поиск лобби

Для того чтобы кнопка отображала данные о сервере нужно через визуальный интерфейс (Designer), выделить текстовый блок и в правой панели свойств в пункте Text нажать Bind, откроется новое окно Graph. В новом окне Graph необходимо добавить следующие узлы, новую переменную через операцию Get, Get Current Players, Get Max Players, Get Ping In MS, Format Text. В узле Format text нужно указать:

«Игроков: {CurrPlayer} из {MaxPlayer} Пинг: {Ping(ms)} ».

И только после того, как появятся ссылки подключения узлов, можно их соединить (рис. 24).

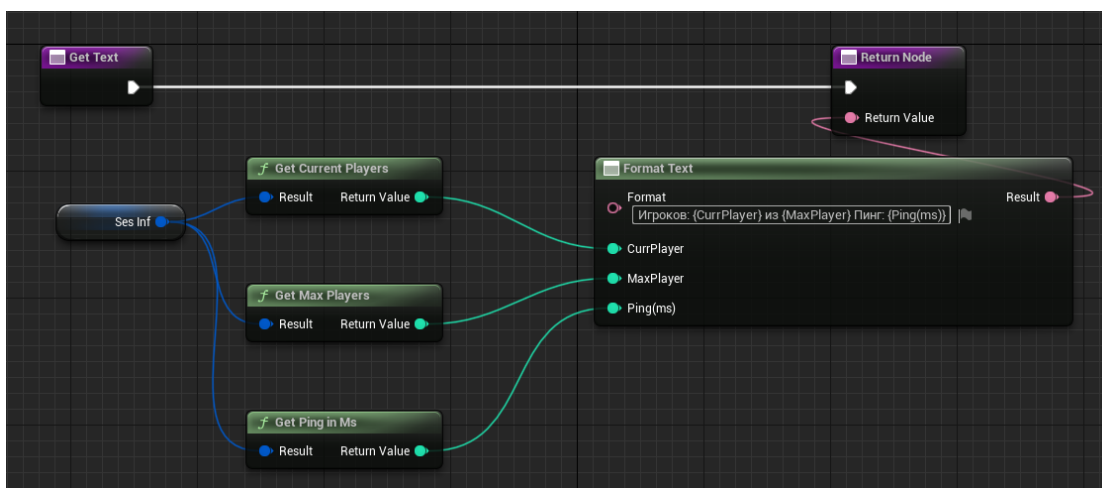


Рисунок 24. Вид Graph, генерирующего текст для кнопки подключения к серверу

На данный момент можно вернуться к редакции меню выбрав **CircularThrobber** в свойствах в пункте, отвечающем за видимость, нужно указать произвольные значения через следующий Graph, там же создать новую переменную, отвечающую за переключение элемента поиска в типе переменной указать Boolean (рис. 25).

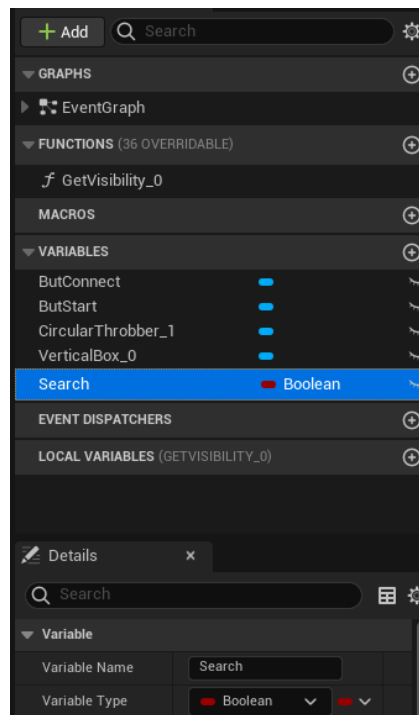


Рисунок 25. Новая переменная Search

В Graph добавить следующие узлы, новую переменную поиска через Get и узел Select, подключить и выставить параметры (рис. 26).

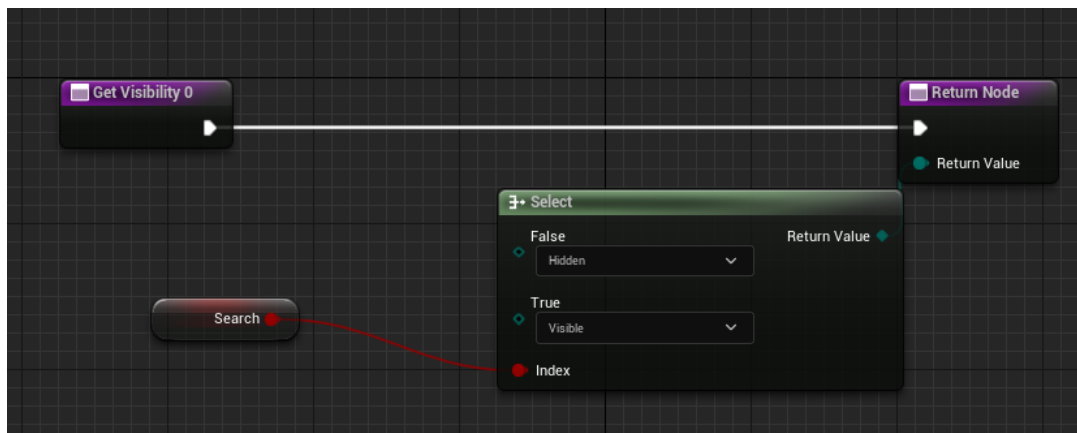


Рисунок 26. Graph, отвечающий за начальную видимость анимации загрузки

Теперь нужно вернуться в виджет меню и назначить действие по нажатию на кнопку подключения, для этого создаются следующие узлы, вертикальный контейнер через операцию Get, переменную вертикального контейнера через Get, Clear Children, Search через Set, Get Player Controller, Find Session, настроить и подключить (рис. 27).

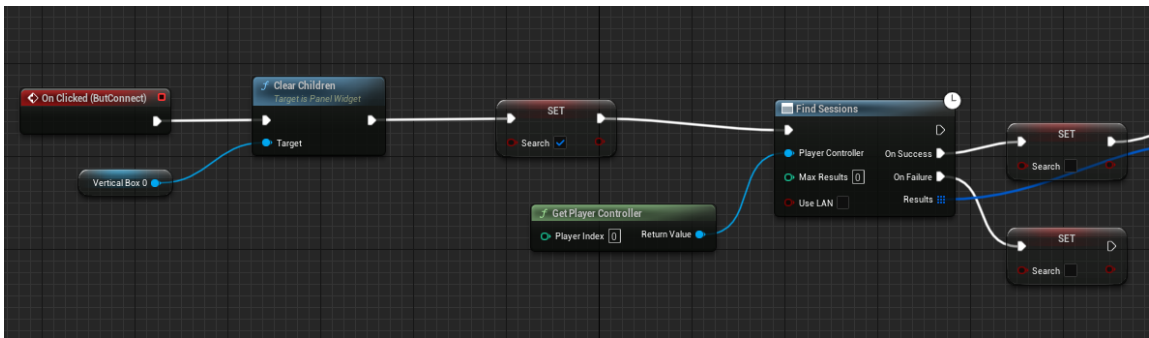


Рисунок 27. Настройка кнопки «Подключиться»

Осталось подключить оставшиеся узлы, For Each Loop, Create widget, переменную вертикального контейнера через Get, Add Child и Add to Viewport (рис. 28).

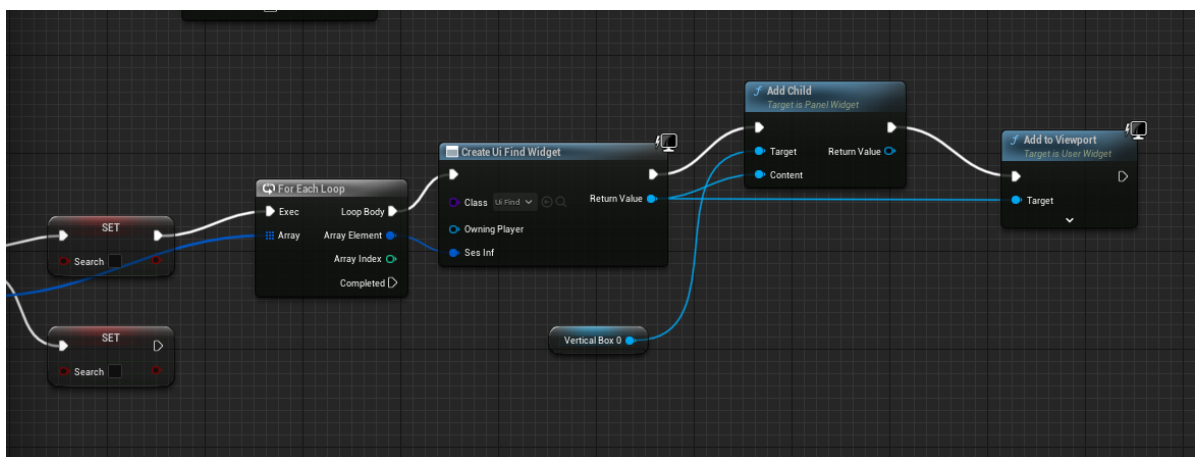


Рисунок 28. Оставшиеся узлы, отвечающие за подключение к сессии

Выводы

По окончании работ будет можно запускать эмуляцию игры, выставив нужное количество игроков. Каждый игрок может создать сессию, и она будет отобразиться в окне поиска сессии (рис. 29).

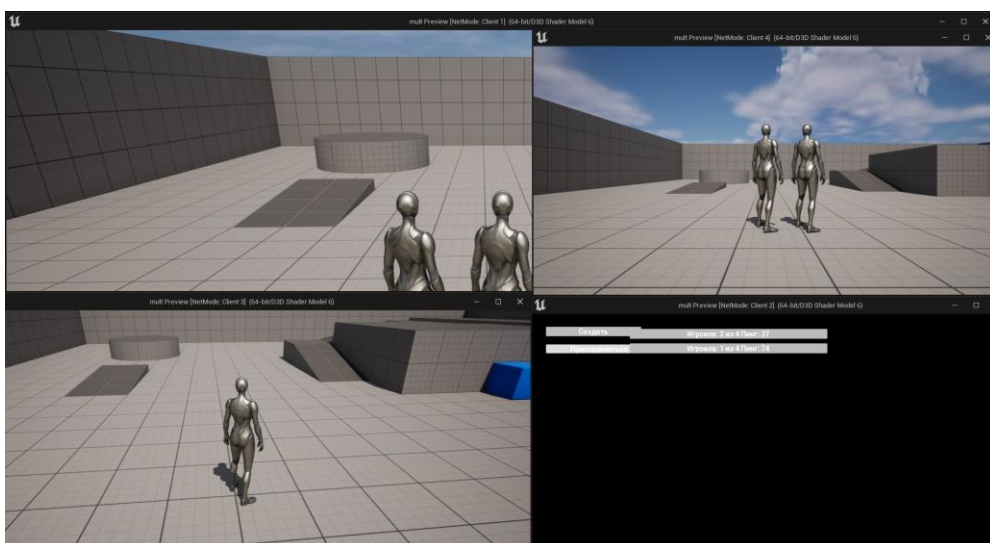


Рисунок 29. Готовый прототип игры с выводом игроков

Так как на начальном этапе, не уточнялось локальной ли будет игра, то сервер можно спокойно разместить в сети интернет на выделенном сервере. Целью не являлась синхронизация данных между клиентом и сервером, поэтому если какой-то игрок будет толкать объект, тогда как другой игрок этого скорее всего не увидит или положение объекта будет отличаться, так как в свойствах объектов в стандартном пресете отключена репликация, а TPS (Tick per second) отличается от FPS на стороне клиента, чтобы не нагружать сервер, но перемещение игроков будет видно в абсолютной позиции (фактической).

Библиографический список

1. Unreal Engine 5 URL: <https://www.unrealengine.com/en-US/unreal-engine-5> (дата обращения 22.12.2022).
2. Blueprint Basic User Guide URL: <https://docs.unrealengine.com/5.0/en-US/blueprint-basic-user-guide-in-unreal-engine/> (дата обращения 22.12.2022)
3. Levchenko B., Chukhray A., Chumachenko D. Development of Game Modules with Support for Synchronous Multiplayer Based on Unreal Engine 4 Using Artificial Intelligence Approach //Integrated Computer Technologies in Mechanical Engineering. Springer, Cham, 2020. С. 503-513.
4. Cordone R. Unreal Engine 4 Game Development Quick Start Guide: Programming professional 3D games with Unreal Engine 4. Packt Publishing Ltd, 2019.
5. Ness J. et al. Project NORs: a Multiplayer Online Battle Arena Game Implemented in Unreal Engine 4. 2015.
6. Glazer J., Madhav S. Multiplayer game programming: Architecting networked games. Addison-Wesley Professional, 2015.