

## Проектирование компьютерной игры

### Курсовая работа

Борисов Кирилл Игоревич,  
Специальность 09.02.07  
«Информационные системы  
и программирование»  
Курс III, группа ИСиП1-21  
Проверил: Цевун Евгений  
Геннадьевич,  
преподаватель  
профессиональных модулей  
высшей квалификационной  
категории

г. Холмск, 2024

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1. АНАЛИЗ ОБЪЕКТА И ПРЕДМЕТА ИССЛЕДОВАНИЯ	5
1.1 Общее представление о разработке игр	5
1.2 Жизненный цикл разработки компьютерных игр	7
1.3 Методологии проектирования компьютерных игр	8
1.4 Определение целевой аудитории	11
1.5 Анализ игрового жанра и его выбор	12
Вывод по первой главе	16
2. ПРОГРАММНЫЕ СРЕДСТВА И ТЕХНОЛОГИИ СОЗДАНИЯ ИГР	18
2.1 Проектирование игрового мира и персонажей	18
2.2 Разработка системы правил и поведения	20
2.3 Проектирование графического интерфейса	21
2.4 Анализ существующих игровых движков и языков программирования	22
2.5 Выбор оптимального игрового движка и языка программирования для реализации игры	26
ЗАКЛЮЧЕНИЕ	28
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	29

## ВВЕДЕНИЕ

Разработка игр является актуальной и постоянно развивающейся областью в современном мире. В настоящее время игры стали неотъемлемой частью культуры и развлечений многих людей по всему миру. Они предоставляют возможность погрузиться в увлекательные виртуальные миры, испытать новые эмоции и взаимодействовать с уникальными персонажами и сюжетами. Социальные платформы, мобильные устройства и компьютерные системы предоставляют все больше возможностей для разработки и распространения игр. Рынок игровой индустрии растет с каждым годом, привлекая миллионы игроков и генерируя значительные доходы. Это создает благоприятные условия для разработчиков, стимулируя их внести свой вклад в эту динамичную область.

Компьютерные игры относятся к числу актуальных проблем современной науки, так как представляют собой уникальный продукт развития техники и современной личности. Игры помогают нам смоделировать различные жизненные ситуации, проблемы и выдают некоторые возможные пути их решения. Игра содержит в себе все необходимые предпосылки для естественного развития личности и культуры общества.

С совершенствованием компьютеров совершенствовались и игры, привлекая все больше и больше людей. На сегодняшний день компьютерная техника достигла такого уровня развития, что позволяет программистам разрабатывать очень реалистичные игры с хорошим графическим и звуковым оформлением.

Цель курсовой работы: спроектировать модель компьютерной игры, отвечающую требованиям целевой аудитории и современным тенденциям игровой индустрии.

Для достижения цели определены следующие задачи:

1. Проанализировать предметную область.

2. Проанализировать программные средства и технологии создания игр.

3. Разработать концепцию игрового мира, персонажей, системы правил и поведения.

4. Проанализировать существующие игровые движки и языки программирования и выбрать оптимальный вариант для реализации игры.

Объект исследования: компьютерные игры.

Предмет исследования: технологический процесс проектирования компьютерной игры.

# 1. АНАЛИЗ ОБЪЕКТА И ПРЕДМЕТА ИССЛЕДОВАНИЯ

## 1.1 Общее представление о разработке игр

Разработка компьютерных игр — это сложный и многогранный процесс, который включает в себя множество этапов и аспектов. Основные элементы разработки игр включают игровой дизайн, программирование, создание графики и звукового сопровождения, которые совместно создают увлекательный и захватывающий игровой контент. Этот процесс требует творческого подхода, высокого уровня технических навыков и тесного сотрудничества между различными специалистами, чтобы создать полноценный игровой опыт для игроков.

Процесс разработки компьютерных игр может осуществляться как отдельными энтузиастами, так и крупными компаниями. Коммерческие игры обычно создаются командами разработчиков, нанятыми одной фирмой. Эти фирмы могут специализироваться на производстве игр для различных платформ, таких как персональные компьютеры, игровые консоли и планшетные компьютеры. Финансирование разработки может осуществляться крупной фирмой-издателем, которая после завершения разработки берет на себя затраты на распространение игры. В альтернативном подходе фирма-разработчик самостоятельно распространяет копии игр, например, через цифровую дистрибуцию.

В разработке игр, как правило, участвует множество людей, каждый из которых имеет свою конкретную роль. Разработка игры контролируется внутренними и внешними продюсерами. Внешние продюсеры назначаются издателем для контроля за прогрессом разработки и бюджетом. Внутренние продюсеры работают непосредственно в студии разработчиков, управляют командой, составляют графики, отчёты о ходе работы, нанимают и распределяют сотрудников.

В команду разработки игр входят следующие должности.

Геймдизайнер - специалист, который проектирует игровой процесс, разрабатывает правила и структуру игры. Геймдизайнеры создают концепции, которые определяют, как игроки будут взаимодействовать с игрой и что делает игру интересной.

Художник - человек, который отвечает за визуальную часть игры. Художники создают графику, включая персонажей, окружающую среду, предметы и интерфейс, что помогает сформировать уникальный стиль и атмосферу игры.

Программист - инженер, который разрабатывает программное обеспечение для игры. Программисты пишут код, который лежит в основе игрового процесса, обеспечивая функциональность и производительность игры.

Геймдизайнер уровней - специалист, создающий уровни, задачи или миссии для видеоигр. Используя инструменты разработки игр, они формируют различные игровые сценарии, которые бросают вызов игрокам и поддерживают их интерес на протяжении всей игры.

Звукорежиссёр - технический специалист, ответственный за создание звуковых эффектов и музыкального сопровождения. Звукорежиссёры работают над тем, чтобы аудио составляющая игры дополняла визуальный ряд и улучшала общий опыт игрока.

Тестировщик - человек, анализирующий игру для выявления дефектов и ошибок. Тестировщики документируют найденные проблемы, что является важной частью процесса контроля качества и помогает разработчикам улучшить игру до её выпуска.

Таким образом, разработка компьютерных игр — это коллективный процесс, требующий участия и координации множества специалистов. Успешная игра является результатом совместной работы различных профессионалов, каждый из которых вносит свой вклад в создание захватывающего и качественного игрового продукта.

## 1.2 Жизненный цикл разработки компьютерных игр

Понятие жизненного цикла (ЖЦ) является одним из базовых понятий методологии проектирования программного обеспечения (ПО).

Жизненный цикл программного обеспечения (ЖЦ ПО) представляет собой непрерывный процесс, начинающийся с момента принятия решения о создании программного обеспечения и заканчивающийся в момент полного изъятия его из эксплуатации.

Для того чтобы получить представление о возможной структуре жизненного цикла программного обеспечения, обратимся сначала к соответствующим стандартам, описывающим технологические процессы.

Международным стандартом, регламентирующим ЖЦ ПО, является IOS/IEC 12207 (ISO – International Standards Organization, Международная организация по стандартизации; IEC International Electrotechnical Commission, Международная комиссия по электротехнике).

Стандарт IOS/IEC 12207 определяет структуру жизненного цикла, включая процессы, действия и задачи, которые должны быть, выполнены во время создания программного продукта. Российским аналогом является ГОСТ Р ИСИ/МЭК 12207 «Информационная технология. Системная и программная инженерия. Процессы жизненного цикла программных средств».

Выделяют пять основных процессов жизненного цикла программного обеспечения:

- приобретение;
- поставка;
- разработка;
- эксплуатация;
- сопровождение.

### 1.3 Методологии проектирования компьютерных игр

Методологии проектирования компьютерных игр – это структурированные подходы и методы, используемые для создания, разработки и выпуска игр. Они помогают организовать процесс разработки, обеспечить эффективное взаимодействие между членами команды и достичь желаемого качества конечного продукта. Существует несколько основных методологий, каждая из которых имеет свои особенности и применимость в различных контекстах разработки игр:

1. Водопадная модель (Waterfall Model) — это одна из самых старых и наиболее традиционных методологий разработки программного обеспечения, включая компьютерные игры. Она была впервые предложена в 1970 году Уинстоном Ройсом и получила свое название из-за визуального сходства процесса с водопадом, где каждая фаза разработки плавно перетекает в следующую (Рисунок 1).

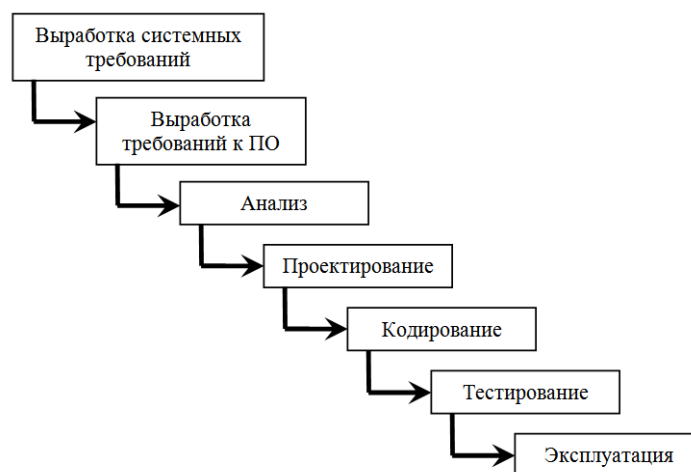


Рисунок 1 - Водопадная модель.

2. Гибкие методологии (Agile Methodologies) — это совокупность подходов к разработке программного обеспечения, которые акцентируют внимание на итеративном развитии, сотрудничестве между кросс-функциональными командами и быстрой адаптации к



изменениям. Эти методологии позволяют создавать более качественные продукты за счет постоянного взаимодействия с пользователями и заказчиками, а также регулярного выпуска обновлений (Рисунок 2);

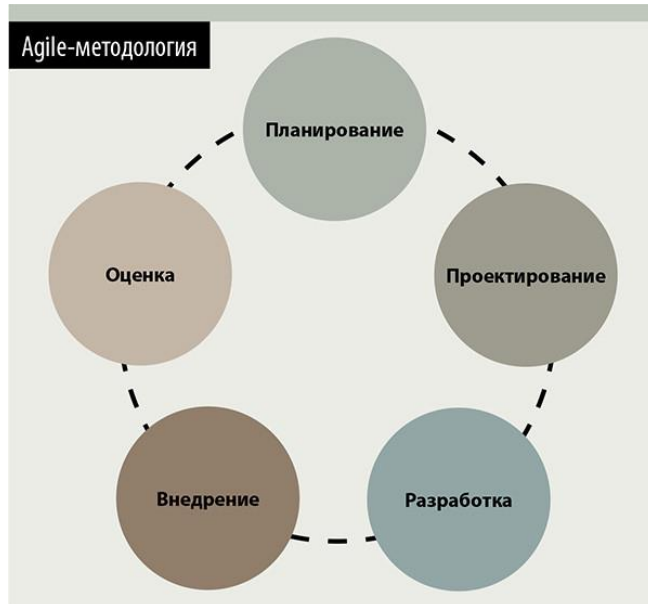


Рисунок 2 - Гибкие методологии

3. Scrum — это одна из наиболее популярных методологий в рамках Agile, предназначенная для управления проектами и оптимизации процессов разработки программного обеспечения. Scrum предоставляет четкую структуру и роли, которые помогают командам организовать свою работу и повысить эффективность (Рисунок 3);

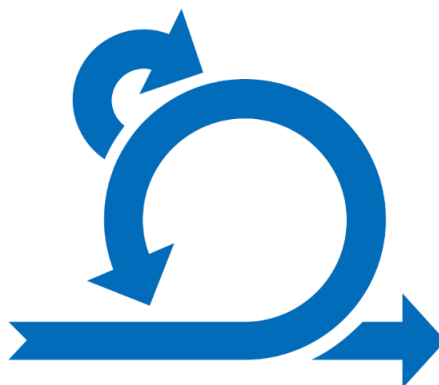


Рисунок 3 – Scrum

4. Kanban — это система постановки задач, при которой все этапы проекта визуализируются на специальной доске. Члены команды могут видеть текущее состояние задачи на любой момент времени. Это предполагает полную прозрачность работы (Рисунок 4);

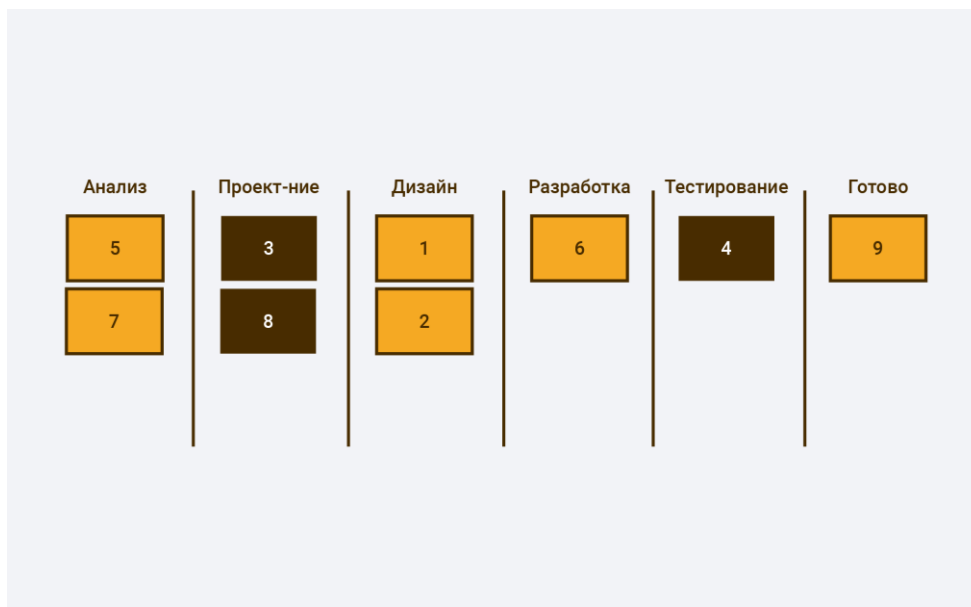


Рисунок 4 - Kanban

5. Lean-разработка — это методология бережливой разработки, которая подразумевает сокращение всех возможных потерь, экономичный и разумный подход к производственным ресурсам (Рисунок 5).



Рисунок 5- Lean-разработка

Исходя из представленных информации о методологиях, для данного проекта была выбрана Kanban модель, так как данная модель позволяет легко адаптировать свой рабочий процесс и вносить изменения в него по мере необходимости, что особо важно при выполнении работы. Все это делает Kanban отличным выбором для разработки проекта

#### 1.4 Определение целевой аудитории

Отбор целевой аудитории является ключевым этапом при разработке компьютерной игры в жанре арена-шутер на платформе Unreal Engine 5 для ПК. Целевая аудитория – это группа людей, которые будут наиболее заинтересованы в игре, её геймплее<sup>1</sup> и функциональных возможностях.

Исследования рынка и анализ конкурентов.

Первым шагом является исследование рынка арена-шутеров, созданных на платформе Unreal Engine 5 для ПК. Анализ популярных арена-шутеров, таких как «Quake Champions» и «Unreal Tournament», позволяет определить ключевые особенности, которые привлекают игроков:

- высокое качество графики: Игры, разработанные на Unreal Engine 5, известны своим фотореалистичным визуалом и сложными эффектами. Игроки ожидают высокой детализации окружающей среды и персонажей;
- динамичный геймплей: Быстрая и интенсивная игровая механика<sup>2</sup> является основной характеристикой арена-шутеров. Игроки ценят возможность быстро вступить в бой и применять различные стратегии для победы;

---

<sup>1</sup>Геймлей (gameplay)— это совокупность взаимодействий и механик, которые определяют, как игрок управляет игрой и взаимодействует с её миром и элементами.

<sup>2</sup>Механики — это правила и системы, которые определяют, как игрок взаимодействует с игрой, включая действия, которые можно выполнить, и способы достижения целей, например, прыжки, стрельба, сбор предметов и решение головоломок.

- разнообразие оружия и карт: Разработчики должны предоставить широкий выбор оружия и разнообразные игровые карты, чтобы поддерживать интерес и вовлеченность игроков.

Интересы и предпочтения потенциальных пользователей:

- целевая аудитория арена-шутеров включает в основном молодых людей в возрасте от 18 до 30 лет. Эти игроки активно интересуются киберспортом<sup>3</sup> и мультиплеерными онлайн-сражениями;
- соревновательный геймплей: Возможность соревноваться с другими игроками и участвовать в турнирах;
- Социальные взаимодействия: Поддержка голосового чата и систем дружбы для общения с друзьями и другими игроками;
- Обновления и дополнения: Регулярные обновления контента, новые карты, оружие и игровые режимы.

Определение целевой аудитории является критически важным этапом разработки. Понимание предпочтений и ожиданий игроков позволяет разработчикам создавать продукт, который будет наиболее интересен и привлекателен для своей аудитории. Это, в свою очередь, увеличивает шансы на коммерческий успех игры и её популярность среди игроков.

### 1.5 Анализ игрового жанра и его выбор

Компьютерные игры — компьютерная программа, служащая для организации игрового процесса (геймплея). Игры в свою очередь классифицируются на жанры:

Экшен (action)— жанр компьютерных игр, в котором делается упор на эксплуатацию физических возможностей игрока, таких как зрительно-моторная координация и скорость реакции.

---

<sup>3</sup>Киберспорт — соревнования в компьютерных играх.



Рисунок 6- Far Cry 3

Shooter (Шутер) - видеоигра, основу игрового процесса которой составляет взаимодействие с внутриигровыми объектами посредством стрельбы в них.

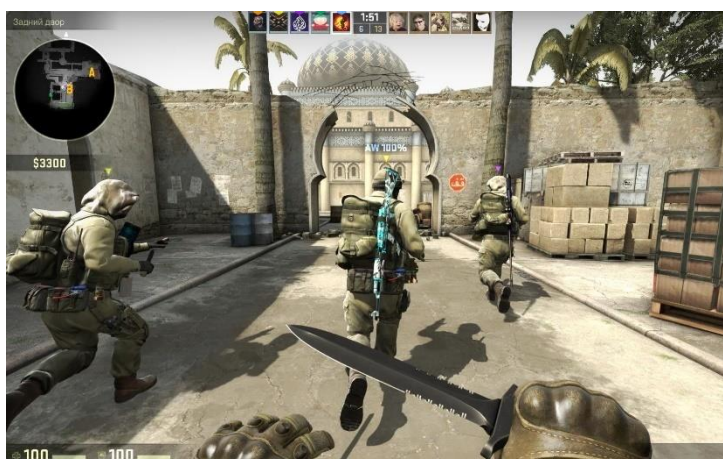


Рисунок7 - Counter-Strike: Global Offensive

Fighting (файтинг) жанр компьютерных игр, имитирующих рукопашный бой малого числа персонажей в пределах ограниченного пространства, называемого ареной.



Рисунок 7 - Mortal Kombat

Strategy (стратегия) жанр видеоигр, в которых пользователю предлагается решать военно-тактические, экономические, логистические и иные задачи, требующие планировать свои действия на несколько шагов вперёд, по возможности предусматривая ответные действия со стороны оппонентов (при их наличии).



Рисунок 8 – Age of Empires IV

Quest (Приключения) - элементами игры в жанре квеста являются повествование и исследование мира, а ключевую роль в игровом процессе играет решение головоломок и задач, требующих от игрока умственных усилий.





Рисунок 9 - Portal 2

Проведя анализ игровой площадки Steam<sup>4</sup>, было опубликовано 14 423 новых игр, что на 14,5 % больше, чем в 2022. Кроме того, в прошлом году было зафиксировано рекордное количество одновременно подключенных пользователей в Steam, а именно 33 598 520 игроков на 26 марта 2023 года. Число одновременно активных игроков в видеоигры в этом году впервые превысило 10 миллионов человек.

Среди самых популярных:

- Одиночная игра - 12 000 игр;
- Инди - 9 300 игр;
- Казуальная игра - 8 800 игр;
- Шутер - 8 500 игр;
- Приключения - 8 000 игр;
- 2D - 6 400 игр;
- Симуляторы - 5,500 игр;
- 3D игры - 5,300 игр;
- RPG - 5 200 игр;
- Стратегии - 4 800 игр.

---

<sup>4</sup>Steam – это онлайн-сервис цифрового распространения компьютерных игр и программ, разработанный и поддерживаемый компанией Valve.

Исходя из вышеописанного, можно сделать вывод, что создание игры в жанре экшен с поджаром арена шутер, данный жанр востребован у игроков.

Arena shooter — это многопользовательские шутеры с несколькими ключевыми характеристиками. Игроки начинают с базового снаряжения, которое можно улучшать с помощью усилителей, находящихся в определенных точках на карте. Арена-шутеры могут использовать механики движения, которые позволяют игрокам демонстрировать свои навыки, такие как стрейфинг для уклонения от выстрелов, использование взрывов ракет для более высокого прыжка или использование предметов для быстрого перемещения по карте. Эти механики часто сочетаются с относительно высокой скоростью передвижения. Карты в арена-шутерах структурированы таким образом, чтобы способствовать взаимодействию и бою между игроками, часто используя элементы, такие как порталы или трамплины, для предоставления дополнительных возможностей для перемещения.

Шутеры с арены восходят к ранним дням шутеров от первого лица, с Wolfenstein 3D (1992) заложившим базовые основы механики.

Doom (1993) отличался более быстрым игровым процессом и большим количеством экшена на экране.

Quake (1996) и другие игры, последовавшие за ним, также позволяли проводить многопользовательские матчи онлайн или по локальной сети.

Первыми настоящими шутерами на арене были Unreal Tournament (1999), Turok: RageWars (1999) и Quake III Arena (1999).

В последнее время появилось несколько игр с элементами «арены», таких как Doom (2016) и Doom Eternal (2020), но игр в этом жанре крупными издателями выпущено не было.

#### Вывод по первой главе

Создание игры включает множество этапов, от идеи до готового продукта, с решением специфических задач на каждом этапе для достижения успеха проекта. Жизненный цикл разработки игр включает этапы от



принятия решения о создании до полного изъятия из эксплуатации, описанные в международных стандартах. Разработка игр может следовать различным методологиям, включая водопадную модель, Agile, Scrum, Kanban и Lean-разработку, каждая из которых имеет свои особенности. Анализ целевой аудитории для арена-шутера на Unreal Engine 5 помогает разработчикам понять предпочтения и ожидания игроков, что увеличивает шансы на коммерческий успех игры. Жанр экшен с поджанром арена-шутер востребован у игроков, что подтверждается популярностью таких игр на платформе Steam и историей развития этого жанра.

## 2. ПРОГРАММНЫЕ СРЕДСТВА И ТЕХНОЛОГИИ СОЗДАНИЯ ИГР

### 2.1 Проектирование игрового мира и персонажей

Модели окружения, из которых выстроены локации игрового мира, показаны на рисунке (Рисунок 6).

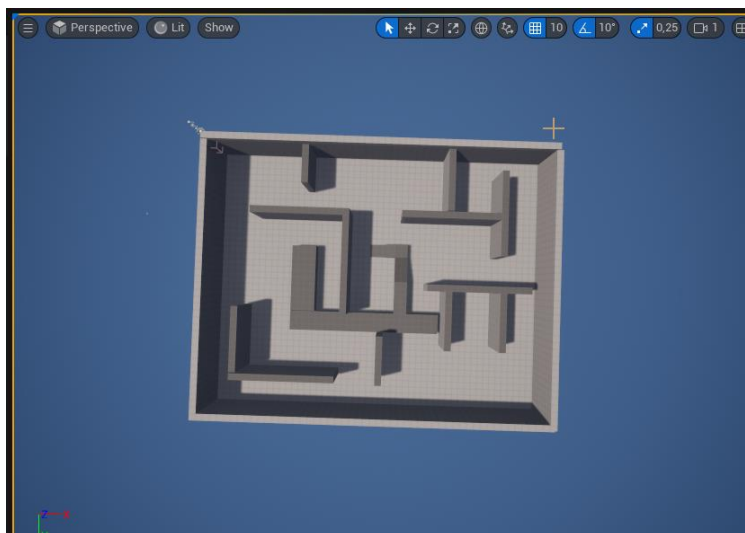


Рисунок 10 - Игровой мир

Таким образом, моделирование игровых объектов является важным этапом разработки видеоигр и позволяет создавать высококачественные и привлекательные игры, которые будут иметь успех у пользователей. Для успешной разработки игры необходимо уметь создавать 3D модели объектов, используя специальные программы для цифрового моделирования, а также обладать знаниями и навыками в области проектирования игровых объектов. Цифровое моделирование и проектирование позволяют ускорить процесс создания игры, улучшить качество и детализацию моделей, а также экономить ресурсы и сократить время разработки.

Процесс создания любой 3d модели включает в себя несколько общих этапов:

- концептуализация: определение общей идеи и формы модели;
- моделирование: создание базовой геометрии модели в 3D программном обеспечении;

- текстурирование: добавление цветовой и текстурной информации на поверхности модели;

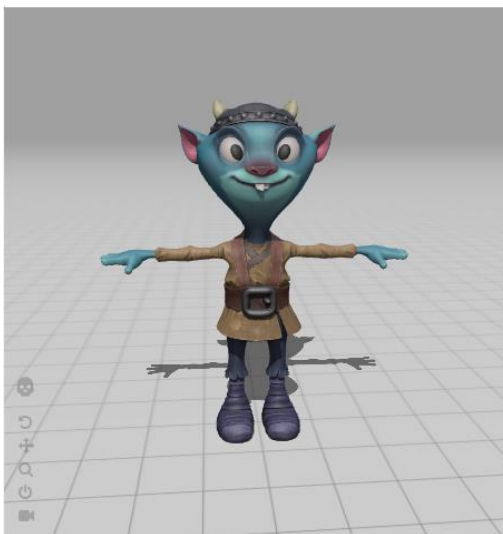


Рисунок 11 - Модель врагов



Рисунок 12 - Модель главного персонажа

- риггинг: создание внутренней структуры и контрольных точек, которые позволяют анимировать модель.

## 2.2 Разработка системы правил и поведения

Работа посвящена разработке системы правил и поведения, где игрок управляет моделью персонажа и сражается против волн мобов<sup>5</sup> в одиночном режиме. Целью является создание сбалансированной и увлекательной игровой механики, обеспечивающей интересный и динамичный игровой процесс.

Основная цель игры состоит в том, чтобы выжить как можно дольше, отбиваясь от волн наступающих мобов, и набрать максимальное количество очков за уничтожение врагов.

Игровое пространство.

Арена игры разделена на несколько зон, каждая из которых обладает различными условиями, такими как открытые пространства, укрытия и высоты. Игрок ограничен в возможности покидать пределы арены.

Волны мобов.

Мобы появляются на арене волнами с возрастающей сложностью. Каждая новая волна включает большее количество мобов, а также более сильных и разнообразных врагов.

Перемещение.

Игрок может ходить, бегать, прыгать, приседать и использовать укрытия. Параметры передвижения включают ограничения на высоту прыжка и скорость бега.

Оружие и снаряжение.

Игрок имеет доступ к разнообразному оружию с различными характеристиками, такими как урон, дальность и скорострельность. На арене предусмотрена возможность подбора и смены оружия. Существуют ограничения на количество носимого оружия и боеприпасов.

Здоровье и выносливость.

---

<sup>5</sup> Мобы - неигровые персонажи (NPC) в видеоиграх, обычно выступающие в роли противников, управляемый программой.

Индикаторы здоровья и выносливости показывают состояние игрока. Восстановление здоровья возможно через аптечки, время или специальные зоны. Выносливость влияет на скорость передвижения и выполнение действий.

Поведения противников и виды.

Мобы активно ищут игрока и атакуют его, Игра включает разные типы мобов с уникальными способностями и стилями атаки, такими как быстрые и слабые, медленные и сильные, а также стреляющие и ближнего боя. Через определенное количество волн появляются боссы.

Разработанная система правил и поведения противников обеспечивает сбалансированный и увлекательный игровой процесс. Применение данных правил позволит создать динамичную и интересную игру, удовлетворяющую требованиям целевой аудитории.

### 2.3 Проектирование графического интерфейса

Разработка пользовательского интерфейса (UI) играет важную роль в создании игры. Первым шагом в разработке UI является определение цели, которую должен достигать пользовательский интерфейс. Определение цели поможет разработчикам понять, какие элементы необходимы для достижения этой цели.

Перед нами стоит цель создать удобный и приятный пользовательский интерфейс, который будет помогать игроку взаимодействовать с игрой. Например, основное меню, в котором можно начать игру, настроить игру под себя или выйти. После того, как цель определена, следующим шагом является создание концепта UI. Это включает в себя определение общего дизайна и выбор цветовой схемы, шрифтов и стилей, которые будут использоваться в игре. Далее разрабатываем макеты UI представляющие собой прототипы пользовательского интерфейса, которые могут быть использованы для проверки функциональности.

Главным является то, что весь UI выдержан в одном стиле, а также достаточно удобен и понятен. После того, как макеты UI созданы, следующим шагом является реализация UI. Реализация происходит непосредственно в среде игрового движка, однако по мере внедрения также происходит тестирование. Она помогает обеспечить удобство использования игры пользователями

## 2.4 Анализ существующих игровых движков и языков программирования

Игровые движки:

Unity: Один из самых популярных игровых движков в индустрии. Поддерживает разработку игр для различных платформ, включая мобильные устройства, компьютеры и консоли. Имеет широкий спектр возможностей и большое сообщество разработчиков.

Особенности: Unity обладает широким набором инструментов для создания 2D и 3D игр, включая редактор сцен, анимацию, физику, свет, звук и многое другое. Он также поддерживает множество языков программирования, включая C#, JavaScript и Boo.

Плюсы:

- Гибкий и расширяемый движок. Множество компонентов для работы «из коробки»;
- Большое количество шаблонов и примеров работ;
- Большое количество уроков и курсов для начинающих;
- Огромная база знаний, большое сообщество;
- Простота в разработке за счёт C# и Bolt;
- Кроссплатформенность.

Минусы:

- Для разработки хорошей игры нужно глубоко погрузиться в аспекты движка;
- Много сырых версий, огромное количество багов;

- Отсутствие открытого кода для небольших команд разработчиков;
- Огромный размер;
- В приоритете разработки обновлений движка — мобильные платформы;
- Движок создан в первую очередь для бизнеса, поэтому его модель поддержки нацелена на бизнес, а не на удобство для разработчиков.

Unreal Engine: ещё один ведущий игровой движок, который предлагает мощные инструменты для создания игр AAA-класса. Имеет высококачественную графику, встроенные инструменты для создания и анимации персонажей, а также поддерживает виртуальную реальность.

Особенности: Unreal Engine предлагает впечатляющие графические возможности, включая визуальные эффекты, фотореалистичные графику, инструменты для создания и анимации персонажей, а также поддержку виртуальной реальности.

Плюсы:

- мощный редактор на все случаи жизни;
- гибкая архитектура игрового движка;
- нацеленность на масштабные проекты с реалистичной графикой и открытым миром;
- доступ к исходному коду;
- возможность создавать проекты без навыков программирования;
- поддержка Epic Games для разработчиков.

Минусы:

- более высокий порог входа;
- более закрытое и не такое многочисленное сообщество;
- акцент — на AAA-проекты;

- проблемы с оптимизацией мобильных игр, в частности из-за слишком большого размера файлов проекта;
- пятая версия движка пока не может похвастаться стабильной работой.

Godot Engine: Бесплатный и с открытым исходным кодом, Godot предоставляет среду разработки, которая подходит для создания 2D и 3D игр. Он обладает интуитивно понятным интерфейсом и хорошей документацией.

Плюсы:

- простота. Godot — сравнительно простой в освоении и использовании игровой движок. Работать с ним могут как опытные разработчики, так и начинающие энтузиасты;
- поддержка сторонних языков. В ядро Godot уже встроена возможность работы на GDScript, C# и C++. Также модуль GDNative позволяет привязать к ядру движка код, написанный на других языках программирования;
- гибкость. Система деревьев узлов и сцен позволяет разрабатывать игры на интуитивно понятном уровне;
- открытый исходный код. Godot — бесплатное программное обеспечение, распространяемое по лицензии MIT;
- кроссплатформенность. Godot существует в версиях для Windows, Mac, Linux, Android, iOS;
- обширная документация. По Godot есть множество информационных материалов от официальных разработчиков, сторонних профессионалов и энтузиастов-любителей.

Минусы:

- недостаточная проработка 3D. По этому параметру Godot уступает конкурентам вроде Unity или Unreal Engine;
- сложности с разработкой консольных приложений. Разработчики отмечают проблемы при создании или портировании Godot-игр



на консоли. Это требует использования сторонних инструментов и большого опыта;

CryEngine: изначально созданный для серии игр Crysis, CryEngine обеспечивает впечатляющую графику и физику. Он может быть сложным в освоении, но предлагает инструменты для создания крупных проектов.

Плюсы:

- масса инструментов для работы с графикой;
- постоянное развитие, обновления с новыми возможностями выпускают очень часто;

огромный магазин с инструментами для будущих игр;

Минусы:

- для создания качественной игры придётся учить программирование;
- очень сложно делать многопользовательские проекты из-за ограничений движка;
- скромное комьюнити<sup>6</sup>, которое состоит в основном из разработчиков известных студий, а они вряд ли помогут с первым проектом;
- следует отметить, что выбор игрового движка зависит от целей и амбиций разработчика;

Важно помнить, что у каждого движка есть свои особенности, поэтому перед использованием рекомендуется ознакомиться с обучающими материалами на официальном сайте.

Языки программирования:

C#: Язык программирования, который широко используется в Unity. Он отлично подходит для создания игр благодаря своей простоте и мощным возможностям, включая управление памятью и многопоточность.

C++: Этот язык часто используется в Unreal Engine и других игровых движках, где производительность имеет решающее значение. Он

---

<sup>6</sup> Комьюнити — это группа людей с близкими интересами, которые общаются через Интернет друг с другом

предоставляет более прямой доступ к аппаратному обеспечению компьютера, что позволяет разработчикам оптимизировать игры для высоких показателей кадров в секунду.

**GScript:** Язык программирования, специально разработанный для использования в Godot Engine. Он похож на Python и обладает простым синтаксисом, что делает его хорошим выбором для новичков.

**Lua:** часто используется для написания скриптов в различных игровых движках, включая Unity и CryEngine. Он известен своей простотой и гибкостью.

## 2.5 Выбор оптимального игрового движка и языка программирования для реализации игры

При разработке любой игры одним из самых важных решений является выбор игрового движка и языка программирования, которые будут использоваться. В данном проекте было решено выбрать игровой движок Unreal Engine и язык программирования Blueprint.

Unreal Engine, выпущенный в 1998 году, является мощным инструментом разработки игр, поддерживающим различные платформы. Он совместим с Windows, macOS и Linux. Unreal Engine стал популярным благодаря своему высококачественному графическому движку и обширным возможностям. Частые обновления и активное сообщество сделали Unreal Engine одним из ведущих игровых движков.

### Преимущества Unreal Engine.

Unreal Engine предлагает ряд возможностей, таких как моделирование физических свойств, динамические тени и освещение, а также поддержку VR и AR. Он обеспечивает высокую производительность и качественный визуальный редактор для удобства разработки.

Ключевое преимущество Unreal Engine заключается в его графическом движке, который обеспечивает потрясающую визуальную реалистичность и детализацию сцен.

Unreal Engine также предлагает систему компонентов для конструирования игровых объектов. В этой системе объекты создаются путем комбинирования компонентов, что делает подход к программированию более гибким.

Разработчики игр могут использовать язык программирования Blueprint в Unreal Engine для написания скриптов и программирования игровой логики. Blueprint представляет собой визуальный язык программирования, который позволяет создавать логику игры без необходимости в изучении кода.

#### Недостатки Unreal Engine.

Хотя Unreal Engine обладает множеством преимуществ, следует учитывать и его недостатки. Например, некоторые разработчики могут столкнуться с изучением нового визуального языка программирования Blueprint, что может потребовать времени и усилий.

Кроме того, в сложных сценах может быть сложно отследить и управлять всеми компонентами, особенно при создании больших проектов.

Тем не менее, в целом, Unreal Engine и язык программирования Blueprint предоставляют мощные инструменты для разработки игр любой сложности и жанра. Их использование позволяет создавать высококачественные игры с учетом современных технологий и требований рынка.

## ЗАКЛЮЧЕНИЕ

В данной курсовой работе проведен всесторонний анализ и рассмотрение ключевых аспектов разработки компьютерной игры в жанре арена-шутер. Исследование охватывает различные этапы создания игры, начиная от определения целевой аудитории и анализа игрового жанра, до выбора оптимальных программных средств и технологий. Первый раздел работы предоставил общее представление о разработке игр, включая жизненный цикл разработки, методологии проектирования и определение целевой аудитории. Важным этапом стало изучение различных игровых жанров и обоснованный выбор жанра арена-шутер, который сочетает в себе динамичность, соревновательный элемент и высокую вовлеченность игроков.

Второй раздел был посвящен техническим аспектам создания игры. Здесь были рассмотрены методы проектирования игрового мира и персонажей, разработка системы правил и поведения, а также проектирование графического интерфейса. Особое внимание уделено анализу существующих игровых движков и языков программирования, что позволило выбрать наиболее подходящий инструмент для реализации проекта.

Выводы данной курсовой работы подтверждают, что успешная разработка компьютерной игры требует тщательного планирования и выбора правильных инструментов и методологий. Применение полученных знаний и рекомендаций позволит создать качественный продукт, отвечающий требованиям целевой аудитории и современным стандартам игровой индустрии.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. ГОСТ Р ИСО/МЭК 12207 Дата введения 2012-03-01 «Информационная технология. Системная и программная инженерия. Процессы жизненного цикла программных средств»
2. Фёдорова, Г.Н. Осуществление в интеграции программных модулей: [учеб. пособие для студ. учреждений сред. проф. образования] / Г.Н. Фёдорова. - М.: Издат. центр «Академия», 2017. - 288 с. - (Проф. образование. Проф. модуль)
3. Попов, А.А. Эргономика пользовательских интерфейсов в информационных системах: [учеб. пособие для СПО по специальности]/ А. А. Попов. - М.: КноРус, 2023. - 305 с. - (Среднее профессиональное образование)
4. Захарченко К.С. «Компьютерные игры это искусство» [Электронный ресурс] – Режим доступа: <https://masters.donntu.ru/2019/fknt/zakharchenko/ind/index.htm>.
5. «Не только экшены, файтинги и платформеры. Полный гид по всем жанрам игр» [Электронный ресурс] – Режим доступа: <https://blog.eldorado.ru/publications/ne-tolko-eksheny-faytingi-i-platformery-populyarnye-zhanry-kompyuternykh-igr-32528>.
6. «Создание компьютерной игры» [Электронный ресурс] – Режим доступа: <https://school-science.ru/21/22/58228>.
7. «Гибкие методологии IT» [Электронный ресурс] – Режим доступа: <https://www.careerist.com/ru-insights/gibkie-metodologii-v-it>.
8. «Канбан» [Электронный ресурс] – Режим доступа: <https://www.unisender.com/ru/glossary/kanban/#anchor-1>.
9. «Жанры компьютерных игр» [Электронный ресурс] – Режим доступа: [https://vlab.fandom.com/ru/wiki/Жанры\\_компьютерных\\_игр](https://vlab.fandom.com/ru/wiki/Жанры_компьютерных_игр).
10. «Online gaming statistics 2023» [Электронный ресурс] – Режим доступа: <https://www.uswitch.com/broadband/studies/online-gaming-statistics/>.

- 11.«Жанры видеоигры, все что вам нужно знать» [Электронный ресурс] – Режим доступа: <https://habr.com/ru/articles/815409/>.
- 12.«Жанры создания игр» [Электронный ресурс] – Режим доступа: <https://gamegod.fandom.com/ru/wiki/Жанры>.
- 13.«Как выбрать жанр и тематику игры: советы для одиночного игрока» [Электронный ресурс] – Режим доступа: <https://rating-gamedev.ru/blog/vybor-zanra-i-tematiki-igry>.
- 14.«Выбор компьютерных игр» [Электронный ресурс] – Режим доступа: <https://cubiq.ru/zhanry-kompyuternyh-igr/>.
- 15.«23 жанра компьютерных игр» [Электронный ресурс] – Режим доступа: <https://gb.ru/blog/zhanry-kompjuternyh-igr/>.
- 16.«Разбираемся в жанрах компьютерных игр» [Электронный ресурс] – Режим доступа: <https://itproger.com/news/razbiraemsya-v-zhanrah-kompyuternih-igr>.
- 17.«Выбор жанра и тематики игр» [Электронный ресурс] – Режим доступа: <https://rating-gamedev.ru/blog/vybor-zanra-i-tematiki-igry>.
- 18.«Этапы создания компьютерной игры» [Электронный ресурс] – Режим доступа: [https://gamesisart.ru/game\\_dev\\_create.html](https://gamesisart.ru/game_dev_create.html).
- 19.«Action(экшен)» [Электронный ресурс] – Режим доступа: [https://ru.wikipedia.org/wiki/Экшен\\_\(жанр\\_компьютерных\\_игр\)](https://ru.wikipedia.org/wiki/Экшен_(жанр_компьютерных_игр)).
- 20.«Shooter (Шутер)» [Электронный ресурс] – Режим доступа: <https://bigenc.ru/c/shuter-99b2ad>.
- 21.«Fighting (файтинг)» [Электронный ресурс] – Режим доступа: <https://blog.eldorado.ru/publications/chtotakoe-faytingi-i-pochemu-stoit-poprobovat-v-nikh-poigrat-35392>.
- 22.«Strategy (стратегия)» [Электронный ресурс] – Режим доступа: <https://bigenc.ru/c/strategicheskie-videoigry>.
- 23.«Приключения» [Электронный ресурс] – Режим доступа: [https://ru.wikipedia.org/wiki/Приключенческая\\_игра](https://ru.wikipedia.org/wiki/Приключенческая_игра).

- 24.«Путеводитель по геймдеву. Не Unity едины. Большой обзор игровых движков для начинающих и профи» [Электронный ресурс] – Режим доступа: <https://dtf.ru/indie/966434-putevoditel-po-geimdevu-ne-unity-ediny-bolshoi-obzor-igrovyyh-dvizhkov-dlya-nachinayushih-i-profi>.
- 25.«Game Health Tracker» [Электронный ресурс] – Режим доступа: <https://newzoo.com/game-health-tracker>.
- 26.«Водопадная модель» [Электронный ресурс] – Режим доступа: <https://konspekta.net/studopedianet/baza5/198206424716.files/image001.png>.
- 27.«Гибкая модель» [Электронный ресурс] – Режим доступа: [https://www.osp.ru/FileStorage/ARTICLE/Direktor\\_informacionnoj\\_sluzhby/2017-10/08\\_17/13206923/Direktor\\_informacionnoj\\_sluzhby\\_029\\_\(3175\).jpg](https://www.osp.ru/FileStorage/ARTICLE/Direktor_informacionnoj_sluzhby/2017-10/08_17/13206923/Direktor_informacionnoj_sluzhby_029_(3175).jpg).
- 28.«Scrum» [Электронный ресурс] – Режим доступа: <https://bendesmedt.github.io/images/scrum.png>.
- 29.«Lean-разработка» [Электронный ресурс] – Режим доступа: [https://avatars.dzeninfra.ru/get-zen\\_doc/3323369/pub\\_6006afbda3a08c096fcc04c2\\_6006b9fae0a5593cf7b67f48/scale\\_1200](https://avatars.dzeninfra.ru/get-zen_doc/3323369/pub_6006afbda3a08c096fcc04c2_6006b9fae0a5593cf7b67f48/scale_1200).