

**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук  
Департамент программной инженерии

**СОГЛАСОВАНО**  
Научный руководитель, приглашенный  
преподаватель департамента «Программной  
инженерии»

\_\_\_\_\_ Н. И. Веселко  
«\_\_» \_\_\_\_\_ 2024 г.

**УТВЕРЖДАЮ**  
Академический руководитель  
образовательной программы  
«Программная инженерия»

\_\_\_\_\_ Н. А. Павлочев  
«\_\_» \_\_\_\_\_ 2024 г.

**3D ИГРА НА UNREAL ENGINE  
«СКАЗКИ МРАЧНОГО ЛЕСА»**

**Пояснительная записка  
ЛИСТ УТВЕРЖДЕНИЯ  
RU.17701729.05.03-01 81 01-1-ЛУ**

Исполнитель:  
студент группы БПИ224  
\_\_\_\_\_/ Егорова Е.П. /  
«\_\_» \_\_\_\_\_ 2024 г.

Подп. и дата	
Инв. № дубл.	
Взам. инв. №	
Подп. и дата	
Инв. № подл	RU.17701729.05.03-01 81 01-1

УТВЕРЖДЕНО  
RU.17701729.05.03-01 81 01-1-ЛУ

3D ИГРА НА UNREAL ENGINE  
«СКАЗКИ МРАЧНОГО ЛЕСА»

Пояснительная записка  
RU.17701729.05.03-01 81 01–1

Листов 37

Инв. № подл	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата
RU.17701729.05.03-01 81 01-1				

## СОДЕРЖАНИЕ

<b>АННОТАЦИЯ .....</b>	<b>4</b>
<b>1. ВВЕДЕНИЕ .....</b>	<b>5</b>
1.1. Наименование программы .....	5
1.2. Документ, на основании которого ведется разработка .....	5
<b>2. НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ .....</b>	<b>5</b>
2.1. Назначение программы .....	5
2.1.1. Функциональное назначение .....	6
2.1.2. Эксплуатационное назначение .....	6
2.2. Краткая характеристика области применения .....	6
<b>3. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ .....</b>	<b>7</b>
3.1. Постановка задачи на разработку программы .....	7
3.2. Описание алгоритма работы и функционирования программы .....	9
3.2.1. Описание игрового процесса .....	9
3.2.2. Описание алгоритма создания врагов с ближними атаками «Волк» и «Скелет» .....	10
3.2.3. Описание алгоритма создания врагов с дальними атаками «Избушка» .....	12
3.2.4. Описание алгоритма создания системы урона для игровых персонажей .....	13
3.2.5. Описание алгоритма работы системы сохранений .....	14
3.2.6. Система создания Боссов «Баба Яга» и «Кощей» .....	14
3.3. Описание и обоснование выбора метода организации входных и выходных данных .....	16
3.4. Описание и обоснование выбора состава технических и программных средств .....	16
3.4.1. Состав технических и программных средств .....	16
3.4.2. Обоснование выбора технических и программных средств .....	17
<b>4. ОЖИДАЕМЫЕ ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ.....</b>	<b>18</b>
4.1. Ориентировочная экономическая эффективность .....	18
4.2. Предполагаемая потребность .....	18
4.3. Экономические преимущества разработки по сравнению с отечественными и зарубежными образцами или аналогами.....	18
<b>5. ИСТОЧНИКИ, ИСПОЛЬЗОВАННЫЕ ПРИ РАЗРАБОТКЕ .....</b>	<b>21</b>
<b>ПРИЛОЖЕНИЕ 1 .....</b>	<b>22</b>
<b>ПРИЛОЖЕНИЕ 2 .....</b>	<b>23</b>
<b>ПРИЛОЖЕНИЕ 3. ССЫЛКИ НА АНАЛОГИ .....</b>	<b>27</b>
<b>ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ .....</b>	<b>28</b>

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## АННОТАЦИЯ

Данный программный документ представляет собой пояснительную записку к программному проекту «3D игра на Unreal Engine "Сказки мрачного леса"», предназначенного для просмотра информации о свойствах химических веществ, а также ссылок на литературные источники и статей, связанных с химией.

Разделе «Введение» содержит наименование программы, условное обозначение темы разработки, документ, на основании которого ведётся разработка, а также организация, утвердившая данный документ.

В разделе «Назначение и область применения» указано функциональное и эксплуатационное назначение программы и краткая характеристика области её применения.

В разделе «Технические характеристики» содержатся следующие подразделы: постановка задачи на разработку программы, описание функционирования программы, описание алгоритма работы программы, описание и обоснование выбора метода организации входных и выходных данных, описание и обоснование выбора состава технических и программных средств.

В разделе «Ожидаемые технико-экономические показатели» указана предполагаемая потребность и экономические преимущества разработки по сравнению с отечественными и зарубежными образцами или аналогами.

Программный документ разработан в соответствии с требованиями:

1. ГОСТ 19.101–77 Виды программ и программных документов [1];
2. ГОСТ 19.102–77 Стадии разработки [2];
3. ГОСТ 19.103–77 Обозначения программ и программных документов [3];
4. ГОСТ 19.104–78 Основные надписи [4];
5. ГОСТ 19.105–78 Общие требования к программным документам [5];
6. ГОСТ 19.106–78 Требования к программным документам, выполненным печатным способом [6];
7. ГОСТ 19.404–79 Пояснительная записка. Требования к содержанию и оформлению [7].

Изменения к Пояснительной записке оформляются согласно ГОСТ 19.603–78 [8], ГОСТ 19.604–78 [9].

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## **1. ВВЕДЕНИЕ**

### **1.1. Наименование программы**

Наименование программы – «3D игра на Unreal Engine "Сказки мрачного леса"».

Наименование программы на английском языке – «3D Game "Tales of the Dark Forest" on Unreal Engine».

Краткое наименование программы - «Сказки мрачного леса».

### **1.2. Документ, на основании которого ведется разработка**

Основанием для разработки является учебный план подготовки бакалавров по направлению 09.03.04 «Программная инженерия» и утвержденная академическим руководителем тема курсового проекта.

Организация, утвердившая этот документ: Национальный исследовательский университет «Высшая школа экономики», Факультет компьютерных наук, образовательная программа «Программная инженерия».

## **2. НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ**

### **2.1. Назначение программы**

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Разработанная игра имеет развлекательный характер и предназначена для расслабления и эмоционального наслаждения.

#### **2.1.1. Функциональное назначение**

Данная программа - это игра, данный тип программ несет развлекательную и досуговую функции. Задачами игры являются: предоставление пользователю разнообразного игрового опыта, путем комбинирования различных систем. Также игра побуждает игроков к исследованию локаций, изучению новых навыков и запоминанию игровых правил.

#### **2.1.2. Эксплуатационное назначение**

Конечными пользователями данной программы являются люди, любящие игры на ПК, а также ценители древнерусской культуры и мифологии.

### **2.2. Краткая характеристика области применения**

«Сказки мрачного леса» – сюжетная 3Д игра, в сеттинге древнерусских сказок, изображает любимых персонажей детских страшилок: Бабу Ягу, Кощея, Лихо и многих других. В путешествии главному герою помогут волшебные путеводные клубки, которые будут оберегать игрока. Необычный боевой стиль и интересное представление всем известных сказок погрузит игрока в сюжет. Играть можно на клавиатуре.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

### 3. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

#### 3.1. Постановка задачи на разработку программы

Разрабатываемый игровой продукт должен выполнять следующие задачи:

1. Загрузка карты мира с расставленными на ней предметами, зданиями, ландшафтом и игровыми персонажами, а также вспомогательными объектами;
2. Реализовать передвижение игрока:
  - 2.1. Ходьба
  - 2.2. Бег
  - 2.3. Прыжки
3. Реализовать атаки и блок игрока
  - 3.1. Ближняя атака с помощью удара палки с клубком
  - 3.2. Дальняя атака с помощью прицела и кидания палкой клубка в сторону прицеливания
  - 3.3. Блок с помощью создания вокруг игрока сферы, дающей ему неуязвимость на фиксированное количество времени
4. Реализовать врага «Волк»:
  - 4.1. «Волк» должен патрулировать заданную ему территорию
  - 4.2. При обнаружении игрока «Волк» должен начать его преследовать
  - 4.3. При достижении радиуса своей атаки «Волк» должен напасть на игрока
  - 4.4. «Волк» преследует и атакует игрока, пока не будет убит
  - 4.5. «Волк» должен пропадать после своей смерти
5. Реализовать врага «Скелет»:
  - 5.1. «Скелет» должен патрулировать заданную ему территорию
  - 5.2. У «Скелета» в руках должен находиться меч, которым он будет атаковать
  - 5.3. При обнаружении игрока «Скелет» должен начать его преследовать
  - 5.4. При достижении радиуса своей атаки «Скелет» должен напасть на игрока
  - 5.5. «Скелет» преследует и атакует игрока, пока не будет убит
  - 5.6. «Скелет» должен пропадать после своей смерти
6. Реализовать врага «Избушка»:
  - 6.1. «Избушка» должна патрулировать заданную ей территорию

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

- 6.2. При обнаружении игрока «Избушка» должна начать атаковать персонажа или преследовать, если радиус атаки меньше того расстояния, что между ними
- 6.3. При достижении радиуса своей атаки «Избушка» должна атаковать игрока
- 6.4. «Избушка» преследует и атакует игрока, пока не будет убита
- 6.5. «Избушка» атакует врага издалека снарядами
- 6.6. «Избушка» должна пропадать после своей смерти

7. Реализация босса «Баба Яга и Избушка»:

- 7.1. Локация для борьбы с боссом ограничена по площади, чтобы игрок не мог уйти за ее пределы
- 7.2. Сражение с боссом разделено на 3 фазы
  - 7.2.1. Первая фаза - борьба с большой избой, у которой есть 2 вида атак. Одна из них дальняя, аналогичная атаке из п 6. про врага «Избушка», вторая же - атака по площади, которая наносит урон в определенном радиусе от себя. Чтобы победить этого босса, надо бить по его слабым местам, а именно - ногам. В это время над персонажем по заданной траектории летает «Баба Яга», которая запускает наводящиеся снаряды в сторону игрока.
  - 7.2.2. Вторая фаза начинается, когда игрок побеждает Избу, после этого наша задача - сбить «Бабу Ягу» на землю, для этого нам надо попасть по ней дальней атакой.
  - 7.2.3. Когда «Баба Яга» оказывается на земле, наступает третья фаза, в ней есть 2 атаки, первая аналогична атаке из фаз один и два, описанных в пунктах 7.2.1. и 7.2.2., персонажу нужно победить, и на этом битва закончится.

8. Реализация босса «Кощей»:

- 8.1. Локация для борьбы с боссом ограничена по площади, чтобы игрок не мог уйти за ее пределы
- 8.2. Сражение с боссом разделено на 3 фазы
  - 8.2.1. Первая фаза - сражение со «Скелетами», на игровой арене с промежутками во времени появляются враги, которых нужно победить за определенное количество времени.
  - 8.2.2. Вторая фаза начинается, когда вышло время, отведенное на битву со «Скелетами», «Кощей» спускается на арену, и используя два вида атак,

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата



нападает на игрока. Первая атака - атака по площади, которая наносит урон в радиусе фиксированной длинны, вторая атака происходит, когда он бежит к персонажу и наносит ему урон в ближнем бою, также надо будет добить оставшихся «Скелетов».

8.2.3. Когда значение здоровья босса становится меньше определенного значения, наступает третья фаза, которая состоит также из двух атак. Первая - рубящая атака, наносящая урон частицами, разлетающимися вокруг врага, а также обычная атака, которую босс совершает, преследуя персонажа.

#### 9. Реализация помощников «Клубков»:

9.1. По сюжету игрок спасает волшебные клубки от злых обитателей леса.

9.2. Игрок сражается с помощью «Клубков», которые появляются на его палке во время атак

9.3. «Клубки» слудуют за персонажем по всей игровой карте, умеют преодолевать препятствия, а также откатываются, когда игрок подходит слишком близко, чтобы не мешать перемещению

#### 10. Система восполнения здоровья:

10.1. В игре есть лечебные предметы, которые можно собирать и восполнять ими здоровье

#### 11. Коллекционные предметы:

11.1. В игре есть коллекционные предметы в виде «Клубков», которых охраняют враги. Персонаж может искать и собирать их, чтобы получить достижение на уровне.

11.2. На каждом уровне счетчик коллекционных «Клубков» свой

#### 12. Система сохранений:

12.1. Чтобы восстанавливать игровой прогресс между сессиями игры, существует система сохранений, которая фиксирует состояние игрока и позволяет начать игру с контрольной точки, которые расположены на картах.

### 3.2. Описание алгоритма работы и функционирования программы

#### 3.2.1. Описание игрового процесса

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

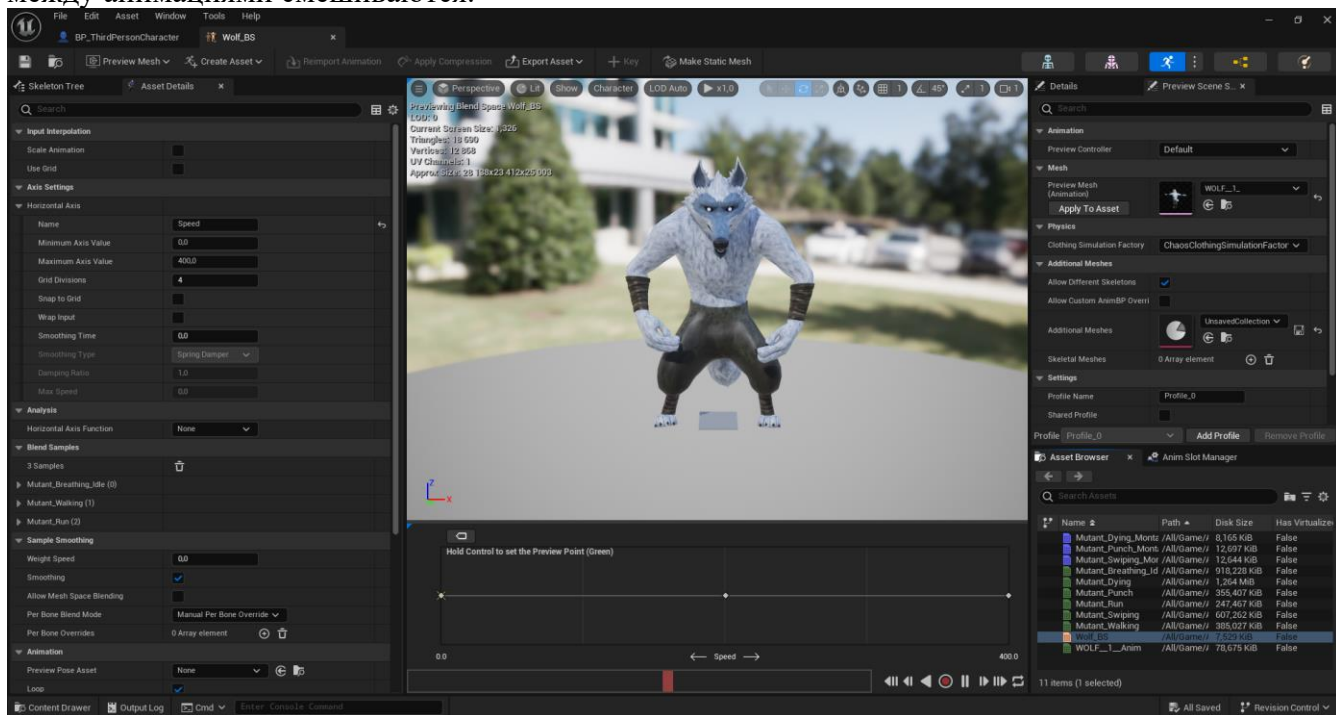
Игрок проходит сюжет за сына хозяина леса Лихо. Потеряв память при нападении на свой дом, в котором погибла его сестра, он оказывается глубоко в лесу и начинает путешествие, в котором найдет новых друзей и врагов. Сопровождать его будут волшебные путеводные клубки, которые помогут Лучезару (Имя персонажа - сокращенно Лучик) одолеть врагов и вспомнить свою историю, которая будет проявляться отрывками воспоминаний, рассказанных сестрой героя.

### 3.2.2. Описание алгоритма создания врагов с ближними атаками «Волк» и «Скелет»

Создание персонажа состоит из нескольких частей:

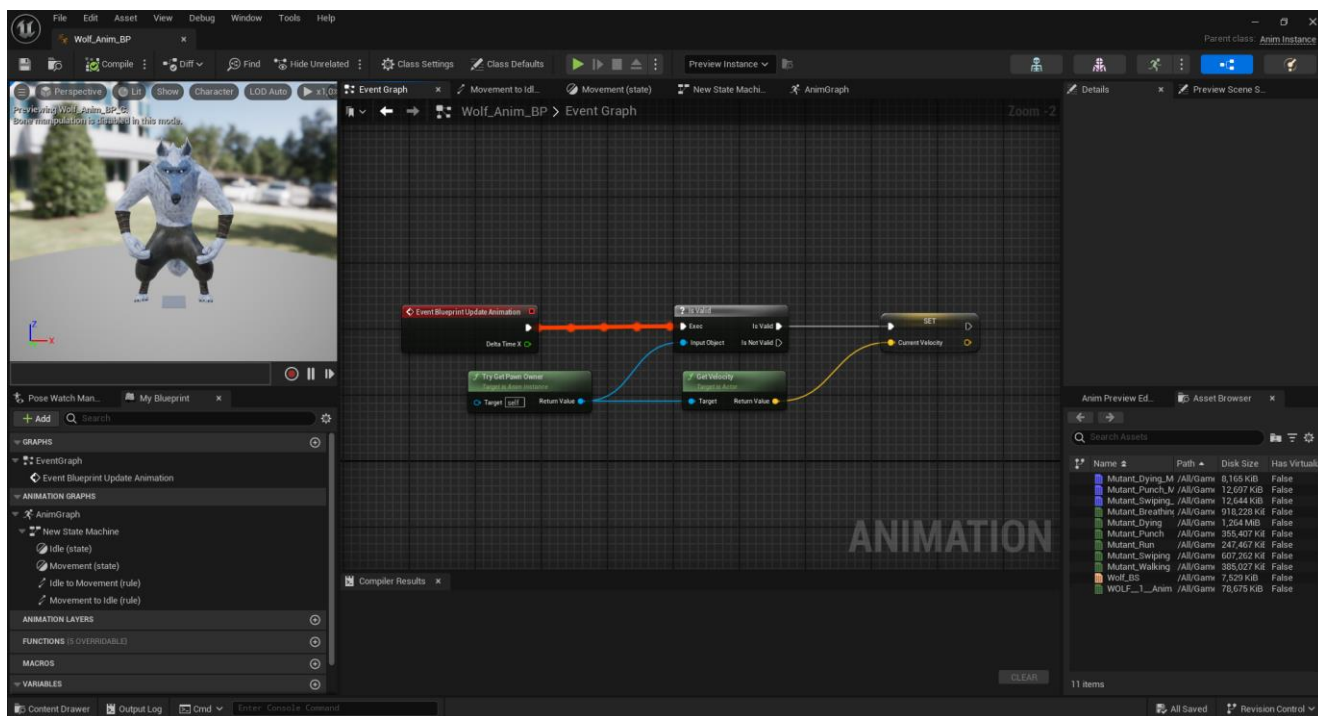
#### 1. Первый этап - Создание анимационного класса персонажа

Для того, чтобы создать систему движений персонажа и привязать к действиям, которые он совершает, я создала анимационный класс персонажа. В нем создали новую машину состояний, в которую добавила состояние Idle. Далее возник вопрос, как же реализовать передвижение: я выбрала вариант создать Blend Space анимаций, в который в качестве параметра передается скорость персонажа, благодаря которому переходы между анимациями смешиваются.



Чтобы скорость постоянно обновлялась, я обновляю её в Event Graph данного анимационного класса.

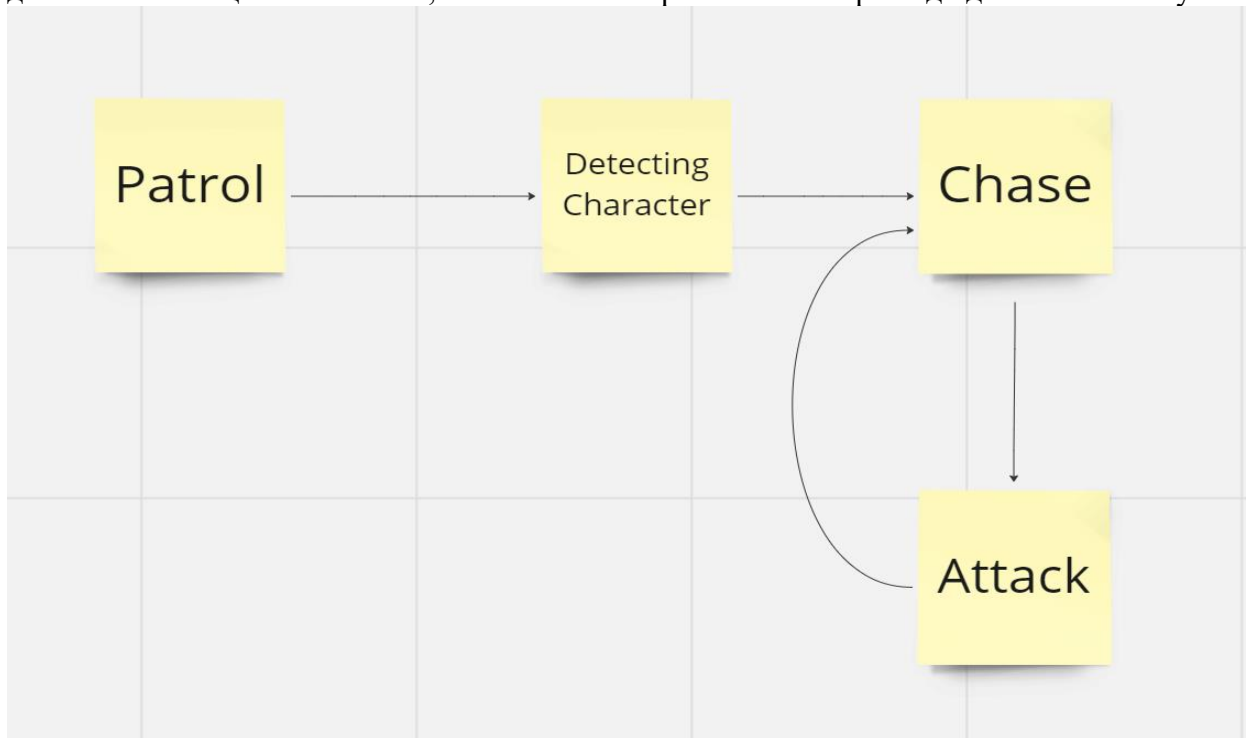
Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата



Теперь при передвижении персонаж меняет анимации в зависимости от скорости.

## 2. Второй этап - Создание основного класса персонажа

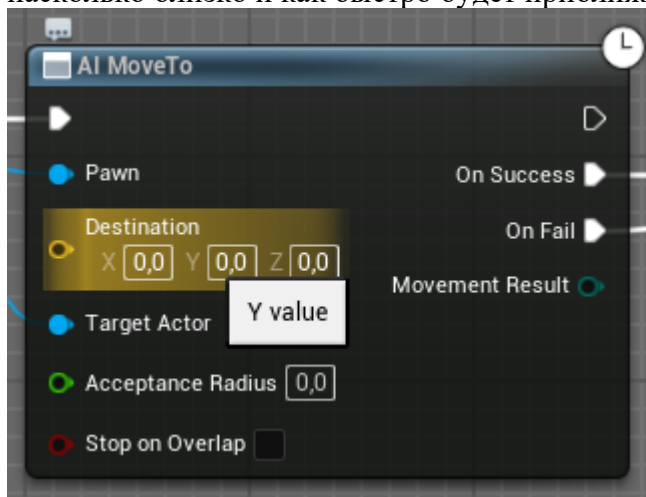
Чтобы реализовать логику персонажа я создала Character класс, в который добавила анимационный класс, а также mesh персонажа. Теперь надо добавить логику.



У каждого персонажа есть две переменные типа Actor, которые отвечают за стартовую и конечную позицию, они остаются публичными и при добавлении врага на

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

сцену, также расставляются два Actor, которые назначаются персонажу в данные точки, таким образом я добавляю переменную Direction, которая принимает значения 1 и -1, и с помощью нее определяю, в какую точку (стартовую или конечную) отправлять врага. В тот момент, когда враг замечает игрока (это происходит с помощью компонента Pawn Sensing, который добавляется в класс персонажа), он начинает следовать за персонажем, это осуществлено с помощью функции AI Move To, в которой можно настроить, куда, насколько близко и как быстро будет приближаться персонаж к заданному объекту.



При достижении заданного объекта, переходим в стадию атаки, она осуществлена очень простой логикой: мы проигрываем Montage[4], созданный из анимации атаки. Затем снова проигрываем атаку, если находимся в радиусе следования, если же игрок вышел из него, снова запускаем стадию Chase, и так пока персонаж не будет убит.

### 3. Замечание

Данную логику можно было бы реализовать в дереве поведения (встроенном инструменте Unreal Engine), она позволяет сделать более гибкую для расширений систему, но в моем проекте стадий поведения небольшое количество, поэтому было решено сделать логику в классе персонажа с помощью функций и Custom Event.

#### 3.2.3 Описание алгоритма создания врагов с дальними атаками «Избушка»

Аналогично врагам с ближними атаками, создание состоит из нескольких частей

##### 1. Первый этап - создание анимационного класса персонажа

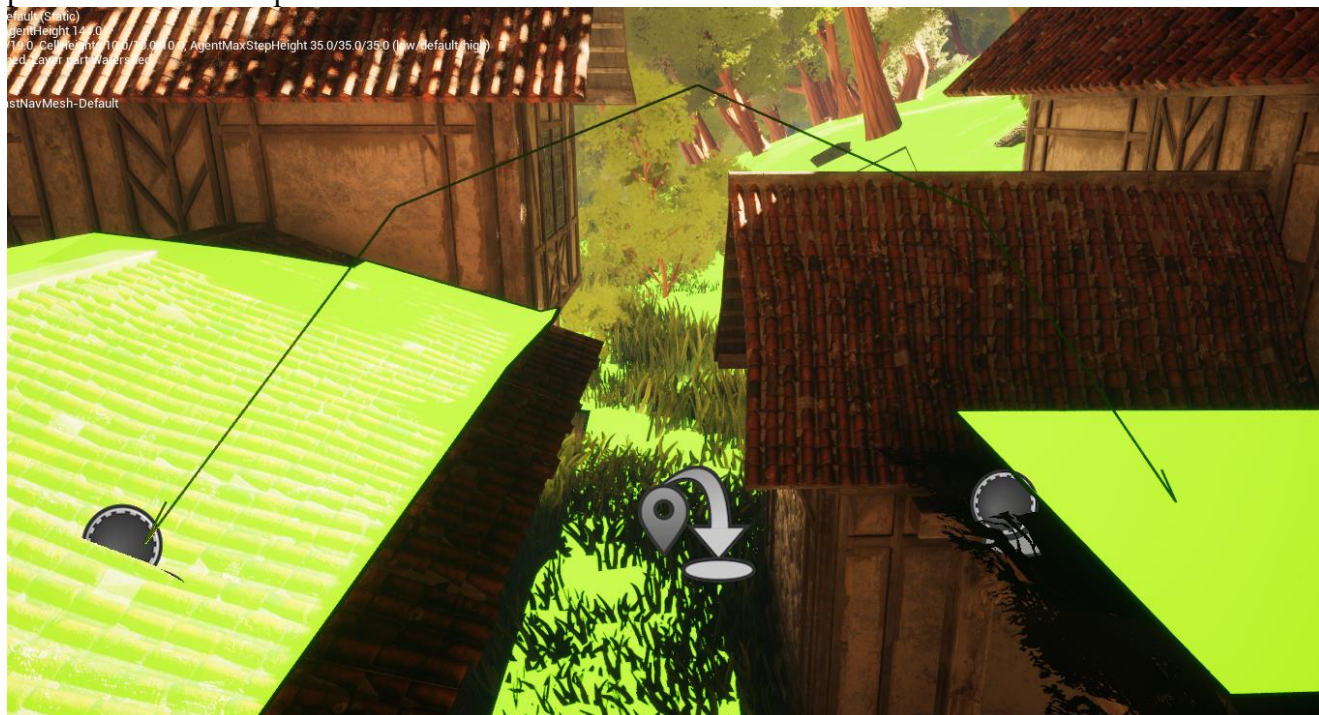
Для начала был создан анимационный класс персонажа, затем я добавила в него всё то же, что и в классе с персонажем ближних атак (см п 3.2.2.), а далее была добавлена логика прыжка, она добавлена потому, что персонажи с дальними атаками располагаются на возвышениях, для того чтобы Избушки могли преследовать персонажа, нужно было научить их перепрыгивать препятствия (в виде ям или разрывов между зданиями). Для этого был создан компонент Nav Link Proxy, в нем можно выставит, насколько высоко будет прыгать выбранный персонаж.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата



## 2. Второй этап - создание класса персонажа

Аналогично п 3.2.2 враг с ближней атакой использует стадии Patrol, Detecting Character, Chase. Так как «Избушка» враг, которого на карте я расставляла на возвышении, то нам важно, чтобы она могла достать персонажа, я использовала встроенную систему Nav Mesh, которая запекается только на горизонтальных и поверхностях с наклоном, стены и другие вертикальные объекты прерывают сетку и не позволяют перемещаться персонажам. Для этого я добавила Nav Link Proxy и расставила их на карте.



## 3. Третий этап - создание объекта снаряда

Так как «Избушка» враг с дальними атаками, то нам надо создать снаряды, которыми она будет атаковать игрока. Для этого я использовала встроенную Projectile System. Создаем класс снаряда и добавляем ему этот компонент, далее я добавила коллайдер, который даст нам обработать столкновение с другими объектами. Чтобы заставить снаряд пересекаться с персонажем, нужно было добавить функцию пересечения коллайдеров. Так как на данном объекте висит компонент Projectile System, то задав направление и скорость, мы получим снаряд, летящий в сторону игрока.

### 3.2.4. Описание алгоритма создания системы урона для игровых персонажей

Для создания системы урона я сделала интерфейс `BI_Enemy`. В нем я создала функцию `Damage`, она отвечает за получение урона. Это позволило при пересечении с любым Actor другого вида проверять, есть ли у него это интерфейс и вызывать функцию `Damage`, таким образом я унифицировала систему нанесения урона для всех врагов.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

### 3.2.5 Описание алгоритма работы системы сохранений

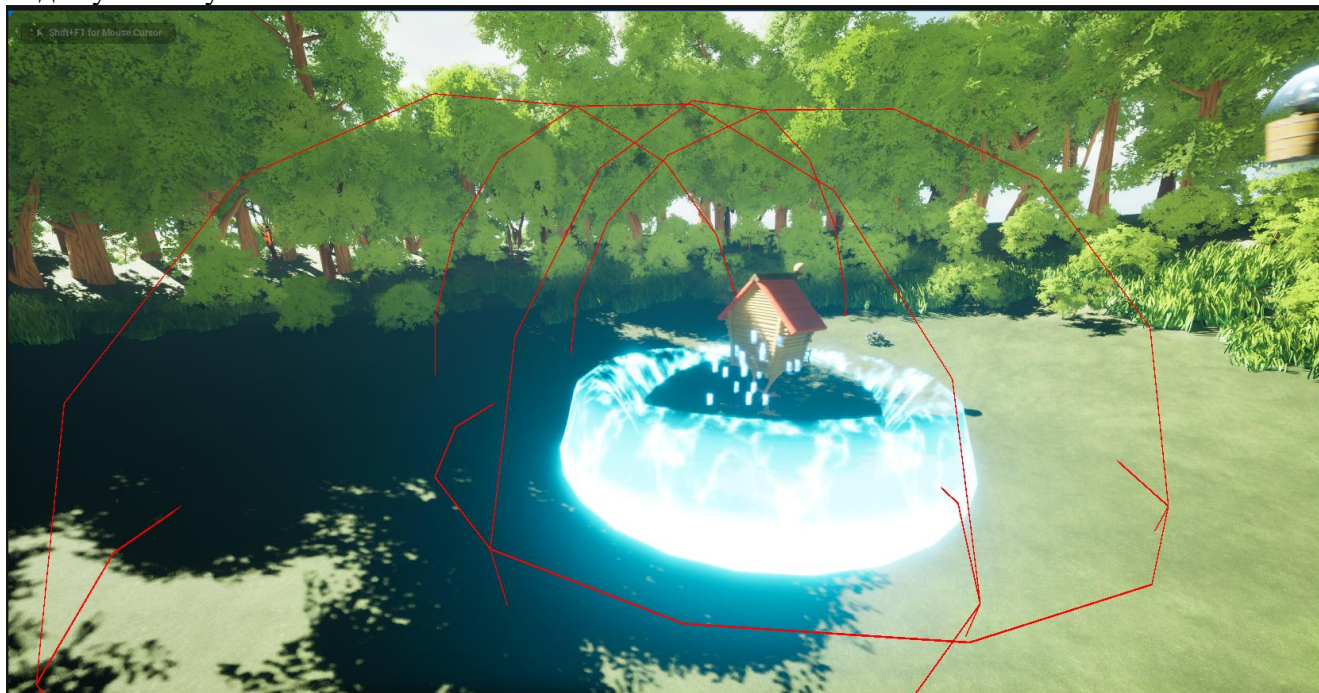
Для сохранения состояния игрока я создала 4 объекта. КАРТИНКА. S\_PlayerData - структура данных с информацией которую мы хотим сохранять об игроке. SaveGameData – blueprint класс со всеми переменными которые мы хотим сохранять, он содержит переменную типа S\_PlayerData для сохранения данных игрока и String для сохранения текущего уровня. SaveGameInstance - класс сохранений, он реализует интерфейс BPI\_SaveGame, определяющий функции SaveData и LoadData, для записи данных на диск и считывания их с него соответственно. Чтобы использовать данную систему в Игроке создаются функции GeneratePlayerSave, UsePlayerSave, которые сохраняют и записывают данные для сохранения.

### 3.2.6. Система создания Боссов «Баба Яга» и «Кощей»

Для создания босса я использовала уже готовую логику для дальних и ближних атак, разработанных в п 3.2.2. и 3.2.3. данного документа. Но также я добавила уникальные механики и поведения для этих персонажей.

#### 1. Первый босс - «Баба Яга»

Особенность данного босса в том, что кроме «Бабы Яги» в битве участвует большая изба, аналог «Избушек», которые я реализовала в п 3.2.3., однако я добавила атаку по площади, она срабатывает с вероятностью 25% в первой фазе битвы с «Бабой Ягой», чтобы реализовать данную механику, я видоизменила функцию нанесения урона у «Избушки», увеличив радиус атаки в стандартной нашей функции, чтобы игрок мог понять, как бьет враг, я также добавила Particle System компонент, кооторый создает видимую волну атаки.

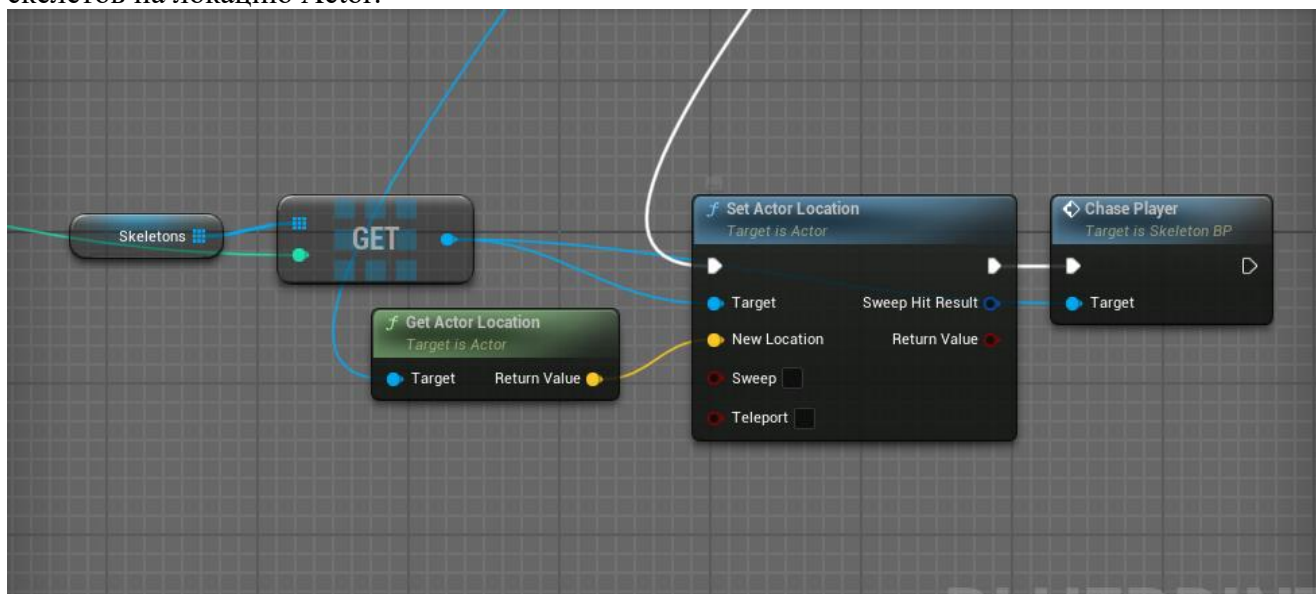


Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

После уничтожения «Избушки» начинается вторая фаза, в ней мы должны сбить «Бабу Ягу», которая следует по заданному пути и совершает дальние атаки по игроку. Чтобы реализовать следование по траектории в воздухе, я создала Spline объект и каждый раз телепортировала «Бабу Ягу» в точку на этой кривой.

## 2. Второй босс - «Кощей»

Данный босс также имеет три стадии. Первая стадия - битва со «Скелетами», чтобы сделать бой более сбалансированным, я добавляю «Скелетов» с разницей во времени, для этого использовала цикл. Сама система появления «Скелетов» на сцене сделана с помощью расставления на карте восьми Actor классов, а за картой восьми «Скелетов», чтобы переместить их в позицию Actor классов, в цикле я меняю локацию скелетов на локацию Actor.

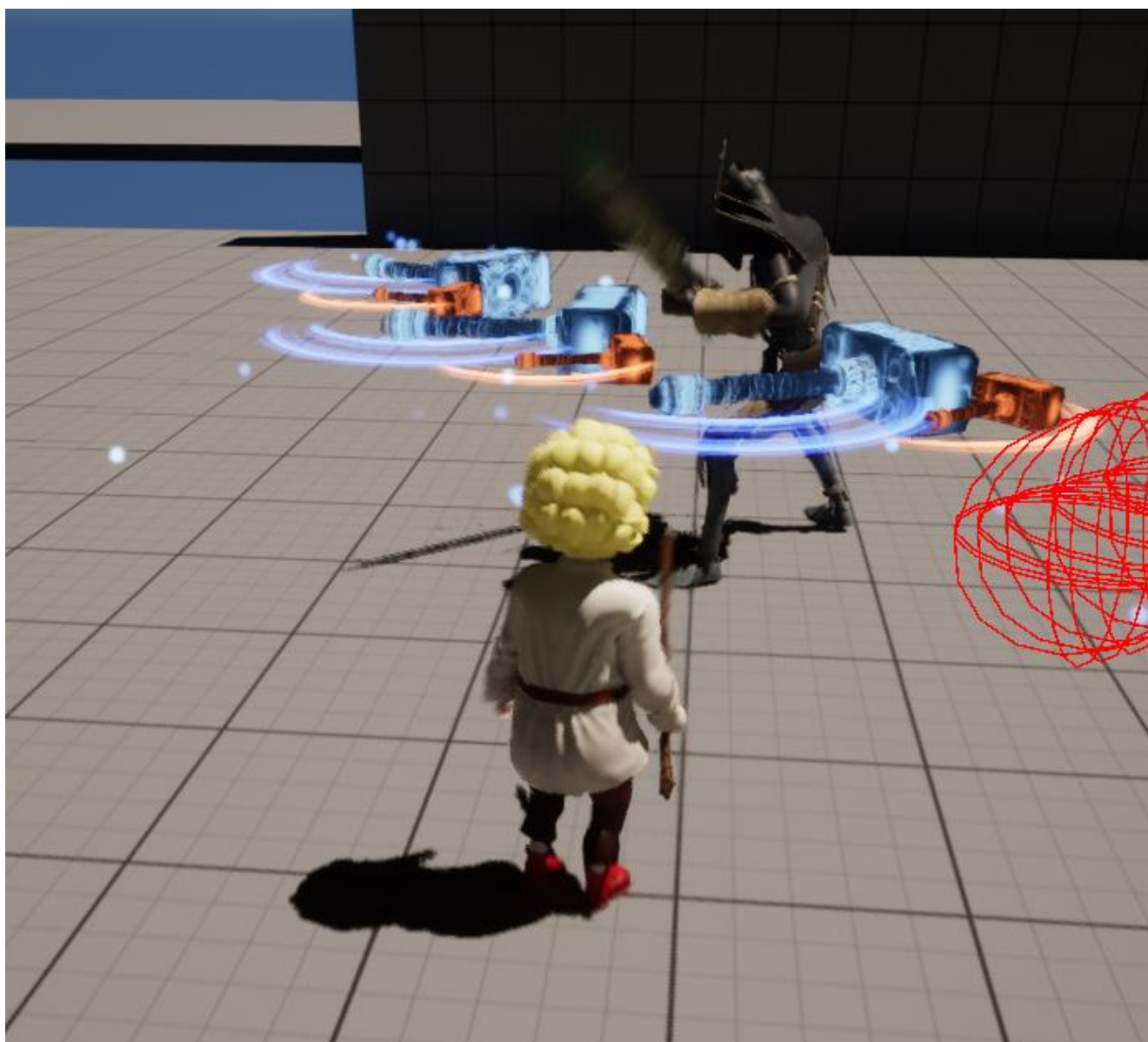


Вторая стадия состоит из ближней атаки, сделанной по принципу из п 3.2.2., а также атаки по площади, которая реализована, как в сражении с боссом «Баба Яга» на фазе с «Избушкой».

Когда «Кощей» теряет часть здоровья, начинается третья фаза. В ней также есть простая ближняя атака, но также я добавила вторую атаку, которая вызывается случайно. Атака представляет из себя перемещение в точку на карте, которая заданная Actor классом, а затем атаку Particle System в виде веера из частиц, разлетающегося вокруг «Кощей».

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата





### 3.3. Описание и обоснование выбора метода организации входных и выходных данных

Входными данными являются: сигналы нажатия на клавиатуру, а также сигналы от нажатия и поворота компьютерной мыши. Данный вид контроллера я выбрала, потому что он является наиболее понятным и часто используемым для игр жанра Action RPG.

### 3.4. Описание и обоснование выбора состава технических и программных средств

#### 3.4.1. Состав технических и программных средств

Для минимальной работы программы требуется следующий состав технических средств:

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата



1. Персональный компьютер/ноутбук с возможностью использования операционной системы Windows версии 10 и выше
2. Процессор Intel Core I7-6700K | AMD Ryzen 5 2600X
3. 8 ГБ ОЗУ
4. Видеокарта Nvidia GeForce GTX 970 (4GB) | AMD Radeon RX 480 (8GB)
5. 32 ГБ места на диске

Для надежной и бесперебойной работы программы требуется следующий состав технических средств:

1. Персональный компьютер/ноутбук с возможностью использования операционной системы Windows версии 10 и выше
2. Процессор Intel Core I5-12400f | AMD Ryzen 5 5500X
3. 16 ГБ ОЗУ
4. Видеокарта Nvidia GeForce RTX 3060ti (8GB) | AMD Radeon RX 6750 (12GB)
5. 32 ГБ места на диске

#### **3.4.2. Обоснование выбора технических и программных средств**

1) Если процессор не подходит под требования игра может работать с зависать, что приводит к плохому опыту.

2) Если оперативная память ниже требуемой, то программа может отображаться и работать не корректно.

3) Клавиатура и мышь необходимы для передачи ввода и управления программой.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

#### 4. ОЖИДАЕМЫЕ ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ

##### 4.1. Ориентировочная экономическая эффективность

В рамках данного проекта расчет экономической эффективности не предусмотрен.

##### 4.2. Предполагаемая потребность

В последние годы спрос на игры значительно возрос, привлекая все больше людей. Игры стали популярным способом отвлечься от повседневных забот и погрузиться в захватывающие миры с загадками и приключениями. Уникальный интерактивный опыт, который предоставляют игры, не имеет аналогов в других сферах развлечений, поэтому основными потребителями игр являются обычные пользователи.

##### 4.3. Экономические преимущества разработки по сравнению с отечественными и зарубежными образцами или аналогами

Продукт будет распространяться на рынке отечественной игровой индустрии, а также имеет необычную и интересную основу, связанную со славянской мифологией. Поэтому он будет востребован.

###### 4.3.1 Аналоги

	Blacktail	Black Book	Woolfe: The Red Riding Hood Diaries	Pikmin 4	Bramble: The Mountain King	Lost in Play	Kena: Bridge of Spirits	Skabma - Snowfall
Полноценный сюжет - история	+	+	+	-	+	+	+	+
3D графика	+	-	+	+	+	-	+	+

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

19  
**RU.17701729.05.03-01 81 01–1**

Мультипликационный стиль	-	+	-	+	+	+	+	+
Боевая система с магическими атаками	+	+	-	+	-	-	+	-
Помощники в путешествии по игровому миру	-	-	-	+	+	+	+	-
Славянская мифология в сюжете	+	+	-	-	-	-	-	-
Возможность настроить чувствительность мыши	+	-	-	-	-	-	-	+
Возможность играть на устройствах с разными операционными системами	+	+	-	+	-	+	+	-
Система сохранений автоматическая	+	+	-	-	+	+	+	+
Поддерживает русский язык озвучки	+	+	-	-	-	-	+	-
Музыкальное сопровождение в стиле мира игры	+	+	+	-	+	+	+	+
Возможность собирать коллекцию	+	-	+	-	-	+	-	-
<b>Итого</b>	<b>10</b>	<b>8</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>9</b>	<b>6</b>

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Игра "Сказки мрачного леса" станет прекрасной частью индустрии игр в стиле славянской мифологии, благодаря мультипликационному стилю и необычной механике боя, игра понравится и обычным пользователям, удобный интерфейс и возможность настроить игровой процесс станут ещё одной причиной играть.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## 5. ИСТОЧНИКИ, ИСПОЛЬЗОВАННЫЕ ПРИ РАЗРАБОТКЕ

1. ГОСТ 19.101–77 Виды программ и программных документов. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
2. ГОСТ 19.102–77 Стадии разработки. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
3. ГОСТ 19.103–77 Обозначения программ и программных документов. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
4. ГОСТ 19.104–78 Основные надписи. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
5. ГОСТ 19.105–78 Общие требования к программным документам. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
6. ГОСТ 19.106–78 Требования к программным документам, выполненным печатным способом. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
7. ГОСТ 19.201–78 Техническое задание. Требования к содержанию и оформлению. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
8. ГОСТ 19.603–78 Общие правила внесения изменений. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
9. ГОСТ 19.604–78 Правила внесения изменений в программные документы, выполненные печатным способом. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
10. ГОСТ 15150–69 Машины, приборы и другие технические изделия. Исполнения для различных климатических районов. Категории, условия эксплуатации, хранения и транспортирования в части воздействия климатических факторов внешней среды. – М.: Изд-во стандартов, 1997.
11. ГОСТ 19.301–79 Программа и методика испытаний. Требования к содержанию и оформлению. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
12. <https://dev.epicgames.com/documentation/en-us/unreal-engine/unreal-engine-5-3-documentation>  
(Дата обращения 31.03.2024)

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**ПРИЛОЖЕНИЕ 1**  
**ТЕРМИНОЛОГИЯ**

Терминология	Определение
Character	Blue Print class персонажа в Unreal Engine, который имеет Character Movement Controller
Mesh	Часть геометрической системы движка, объект который имеет полигональную структуру и может быть использован в качестве части составного объекта
Actor	Простейший Blue Print класс, в котором можно прописать любую кастомную логику
Custom Event	Кастомная (нешаблонная) функция, которую можно вызвать внутри класса Blue Print
Nav Mesh	Встроенная система прокладывания маршрутов в Unreal Engine
Коллайдер	Невидимый объект, который отвечает за пересечения с другими объектами и находится на объекте класса

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**ПРИЛОЖЕНИЕ 2**  
**ОПИСАНИЕ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ КЛАССОВ**

Описание и функциональное назначение классов.

<b>Класс</b>	<b>Функциональное назначение</b>
<b>BP_Jump_ai</b>	Класс, отвечающий за Nav Link Проху, который позволяет всем объектам с Smart Jump пересекать препятствия
<b>Clew_Anim_BP</b>	Класс, отвечающий за анимации и смешивание анимаций для объекта Клубок
<b>Clew_BS</b>	Класс, отвечающий за смешивание анимаций персонажа Клубок
<b>PB_Clew</b>	Основной класс персонажа Клубок, в котором прописана вся логика персонажа
<b>BP_CollectableActor</b>	Класс, отвечающий за коллекционных персонажей в игре
<b>BI_Enemy</b>	Интерфейс, унифицирующий функцию урона от атаки по врагам
<b>StartPatrolPoint</b>	Вспомогательный пустой класс для патрулирования территории врагами (Стартовая позиция)
<b>EndPatrolPoint</b>	Вспомогательный пустой класс для патрулирования территории врагами (Конечная позиция)
<b>SavePoint</b>	Класс, отвечающий за точку сохранения игры
<b>BP_HealActor</b>	Класс, отвечающий за лечебные предметы и их взаимодействие с игроком
<b>Koschei_anim_BP</b>	Класс, отвечающий за анимации и смешивание анимаций для объекта Кощей
<b>Koschei_BS</b>	Класс, отвечающий за смешивание анимаций персонажа Кощей
<b>Koschei_BP</b>	Основной класс персонажа Кощей, в котором прописана вся логика персонажа
<b>MaxiHut_BP</b>	Основной класс персонажа-босса Избушка, в котором прописана вся логика персонажа

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

<b>Hut_anim_BP</b>	Класс, отвечающий за анимации и смешивание анимаций для всех объектов Избушек
<b>Hut_BS</b>	Класс, отвечающий за смешивание анимаций всех объектов Избушек
<b>MiniHut_BP</b>	Основной класс персонажа Маленькая Избушка, в котором прописана вся логика персонажа
<b>BP_PrisonerClewActor</b>	Класс, отвечающий за создание и инициализацию персонажа Заключенный Клубок
<b>BP_PrisonerClewSimpleActor</b>	Класс, отвечающий за процесс спасения Клубка и дальнейшее его следование за игроком
<b>BP_BigHammerAttack</b>	Класс, отвечающий за нанесение урона атакой Избушки босса атакой по области
<b>BP_HammerAttack</b>	Класс, отвечающий за нанесение урона атакой Избушки босса дальней атакой
<b>BP_HutAttack</b>	Класс, отвечающий за снаряд, которым атакует избушка
<b>BP_Projectile</b>	Класс, отвечающий за снаряд, которым атакует главный персонаж
<b>BP_SplashAttack</b>	Класс, отвечающий за нанесение урона Splash атакой Кощея
<b>BP_YagaAttack</b>	Класс, отвечающий за создание обычной дальней атаки Бабы Яги
<b>BP_YagaEnergyAttack</b>	Класс, отвечающий за создание энергетической дальней атаки Бабы Яги
<b>BPI_SaveGame</b>	Интерфейс, создающий функции сохранения и загрузки игры
<b>S_PlayerData</b>	Структура, хранящая все данные для сохранения персонажа
<b>SaveGameData</b>	Класс, хранящий данные для инстанцирования
<b>SaveGameInstance</b>	Класс, хранящий в себе SaveGameData и иницирующий сохранение и загрузку игры
<b>Skeleton_amin_BP</b>	Класс, отвечающий за анимации и смешивание анимаций для объекта Скелет

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата



<b>Skeleton_BS</b>	Класс, отвечающий за смешивание анимаций персонажа Скелет
<b>Skeleton_BP</b>	Основной класс персонажа Скелет, в котором прописана вся логика персонажа
<b>Skeleton_NS</b>	Класс, отвечающий за Notify атаки персонажа
<b>BP_StoryVillageActor</b>	Класс, отвечающий за подсказки на локации с Деревней
<b>BP_StoryWolfsActor</b>	Класс, отвечающий за подсказки на локации с побегом от волков
<b>BP_Charapter0EndActor</b>	Актор окончания нулевого уровня
<b>BP_CharapterForestEndActor</b>	Актор окончания уровня Лес
<b>BP_CharapterGraveyardEndActor</b>	Актор окончания уровня Кладбище
<b>BP_CharapterVillageEndActor</b>	Актор окончания уровня Деревня
<b>BP_TutorialBlockActor</b>	Класс, отвечающий за воспроизведения обучения блокированию
<b>BP_TutorialHealActor</b>	Класс, отвечающий за воспроизведения обучения лечению
<b>BP_TutorialHitActor</b>	Класс, отвечающий за воспроизведения обучения ближней атаке
<b>BP_TutorialJumpActor</b>	Класс, отвечающий за воспроизведения обучения прыжкам
<b>BP_TutorialMoveActor</b>	Класс, отвечающий за воспроизведения обучения движению
<b>BP_TutorialRunActor</b>	Класс, отвечающий за воспроизведения обучения бегу
<b>BP_TutorialThrowActor</b>	Класс, отвечающий за воспроизведения обучения бросанию
<b>BP_Wolf</b>	Основной класс персонажа Волк, в котором прописана вся логика персонажа
<b>Wolf_NS</b>	Класс, отвечающий за Notify атаки персонажа
<b>Wolf_anim_BP</b>	Класс, отвечающий за анимации и смешивание анимаций для объекта Волк
<b>Wolf_BS</b>	Класс, отвечающий за смешивание анимаций персонажа Скелет
<b>AB_Yaga</b>	Класс, отвечающий за анимации и смешивание анимаций для объекта Баба Яга
<b>BP_Yaga</b>	Основной класс персонажа Баба Яга, в котором прописана вся логика персонажа
<b>BP_YagaPath</b>	Класс, отвечающий за путь патрулирования карты Бабой Ягой
<b>Yaga_BS</b>	Класс, отвечающий за смешивание анимаций персонажа Баба Яга
<b>BPI_Items</b>	Интерфейс, отвечающий за функцию поднятия предмета
<b>Blend_Boy</b>	Класс, отвечающий за смешивание анимаций персонажа игрока

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

<b>Boy_anim_BP</b>	Класс, отвечающий за анимации и смешивание анимаций для игрока
<b>BP_ThirdPersonCharacter</b>	Основной класс игрока, в котором прописана вся логика персонажа, а также система сохранений игры
<b>BP_ThitdPersonGameMode</b>	Класс настройки персонажа игрока
<b>NS_Attack_Detecting</b>	Класс, отвечающий за Notify ударов игрока
<b>AboutMenu</b>	Виджет класс, отвечающий за кладку об Авторах
<b>GameSettings</b>	Виджет класс, отвечающий за UI настроек
<b>MainMenu</b>	Виджет класс, отвечающий за Главное Меню
<b>VideoSettings</b>	Виджет класс, отвечающий за настройку расширения экрана

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**ПРИЛОЖЕНИЕ 3.  
ССЫЛКИ НА АНАЛОГИ**

Приложение	Ссылка (дата обращения 05.02.24)
Blacktail	<a href="https://blacktail.game/">https://blacktail.game/</a>
Black Book	<a href="https://blackbookgame.com/blackbook.html">https://blackbookgame.com/blackbook.html</a>
Woolfe: The Red Riding Hood Diaries	<a href="https://woolfegame.com/">https://woolfegame.com/</a>
Pikmin 4	<a href="https://pikminbloom.com/pikmin4-campaign-info">https://pikminbloom.com/pikmin4-campaign-info</a>
Bramble: The Mountain King	<a href="https://dimfroststudio.com/bramble/">https://dimfroststudio.com/bramble/</a>
Lost in Play	<a href="https://www.happyjuice.games/">https://www.happyjuice.games/</a>
Kena: Bridge of Spirits	<a href="https://www.playstation.com/ru-ua/games/kena-bridge-of-spirits/">https://www.playstation.com/ru-ua/games/kena-bridge-of-spirits/</a>
Skabma - Snowfall	<a href="https://store.steampowered.com/app/1665280/Skbma_Snowfall/?l=russian&amp;curator_clanid=9730205">https://store.steampowered.com/app/1665280/Skbma_Snowfall/?l=russian&amp;curator_clanid=9730205</a>

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ**

Лист регистрации изменений									
Номера листов (страниц)					Всего листов (страниц в докум.)	№ документа	Входящий № сопроводит ельного докум. и дата	Подп.	Дата
Изм.	Изменен ных	Заменен ных	Новых	Аннули рованных					

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата


Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.03-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата