Содержание

[Аннотация 3](#_Toc198058106)

[1. Введение 4](#_Toc198058107)

[1.1. Наименование программы 4](#_Toc198058108)

[1.2. Документы, на основании которых ведется разработка 4](#_Toc198058109)

[2. Назначение и область применения 5](#_Toc198058110)

[2.1. Назначение разработки 5](#_Toc198058111)

[2.1.1. Функциональное назначение 5](#_Toc198058112)

[2.1.2. Эксплуатационное назначение 5](#_Toc198058113)

[2.2. Краткая характеристика области применения 5](#_Toc198058114)

[3. Технические характеристики 6](#_Toc198058115)

[3.1. Постановка задачи на разработку программы 6](#_Toc198058116)

[1. Главное меню: 6](#_Toc198058117)

[2. Меню настроек: 6](#_Toc198058118)

[3. Окно достижения: 6](#_Toc198058119)

[4. Окно статистика: 6](#_Toc198058120)

[5. Меню уровней – «Карта»: 6](#_Toc198058121)

[6. Игровой уровень: 6](#_Toc198058122)

[3.2. Описание и обоснование алгоритма и функционирования программы 8](#_Toc198058123)

[3.2.1. Описание архитектуры 8](#_Toc198058124)

[3.2.2. Описание алгоритма и функционирования программы 11](#_Toc198058125)

[3.2.2.1. Описание главного меню 11](#_Toc198058126)

[3.2.2.2. Описание меню выбора сохранений 11](#_Toc198058127)

[3.2.2.3. Описание меню выбора уровней 11](#_Toc198058128)

[3.2.2.4. Описание меню настроек 12](#_Toc198058129)

[3.2.2.5. Описание окна достижений 12](#_Toc198058130)

[3.2.2.6. Описание окна статистики 12](#_Toc198058131)

[3.2.2.7. Описание здоровья 12](#_Toc198058132)

[3.3. Описание и обоснование метода организации входных и выходных данных 13](#_Toc198058133)

[3.4. Описание и обоснование выбора состава технических и программных средств 14](#_Toc198058134)

[3.4.1. Состав технических и программных средств 14](#_Toc198058135)

[3.4.2. Обоснование выбора состава технических и программных средств 14](#_Toc198058136)

[4. Технико-экономические показатели 15](#_Toc198058137)

[4.1. Предполагаемая потребность 15](#_Toc198058138)

[4.2. Ориентировочная экономическая эффективность 15](#_Toc198058139)

[Приложение 1. 16](#_Toc198058140)

[Список используемой литературы 16](#_Toc198058141)

[ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ 17](#_Toc198058142)

# 1. Введение

## 1.1. Наименование программы

Наименование программы – 2D игра «Ночные сны»

Наименование программы на английском языке – 2D game «Night dreams»

Краткое наименование программы – «Ночные сны»

## 1.2. Документы, на основании которых ведется разработка

Программа выполняется в рамках темы курсового проекта в соответствии с учебным планом подготовки бакалавров по направлению 09.03.04 «Программная инженерия» Национального исследовательского университета «Высшая школа экономики», факультет компьютерных наук, департамент программной инженерии.

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

# 2. Назначение и область применения

## 2.1. Назначение разработки

### 2.1.1. Функциональное назначение

«Ночные сны» представляет собой однопользовательскую 2D игру. Gameplay состоит из последовательного сражения с боссами в фиксированных различных локациях, а также исследования окружающего игрового сеттинга с целью поиска новых противников, снаряжений и навыков по мере продвижения по сюжету.

### 2.1.2. Эксплуатационное назначение

Игра предназначена для досуга и развлечения. «Ночные сны» нацелена на любителей хардкорных игр и поклонников жанра «run and gun». Игра охватывает аудиторию широкого возрастного диапазона (от 6 до 60 лет), заинтересованную в прохождении платформеров.

### 2.2. Краткая характеристика области применения

«Ночные сны» представляет собой 2D игру в жанре "run and gun", выполненную на движке Unity. Игра погружает игрока в атмосферу ночного мира с мистической тематикой и интенсивными боями, где основная задача — выжить, уничтожая врагов и продвигаясь через уровни, наполненные опасными боссами. Игровой процесс сочетает динамичное движение, сражения с врагами, коллекционирование усилений и использование разнообразного оружия.

# 3. Технические характеристики

## 3.1. Постановка задачи на разработку программы

2D игра «Ночные сны» должна реализовывать следящие функции:

### Главное меню:

* 1. Новая игра – создает новое сохранение и переводит в меню уровней
  2. Продолжить игру – переводит в меню уровней последнего сохранения
  3. Настройки – переводит в меню настроек
  4. Выход – закрывает приложение
  5. Статистика – демонстрация статистику сохранения
  6. Достижения – демонстрация полученных достижений

### Меню настроек:

* 1. Микшер громкости - управление громкостью
  2. Микшер яркости - управление яркостью
  3. Кнопка «Back» - выход назад в главное меню

### Окно достижения:

* 1. Экран со списком всех достижений, которые возможно получить в игре с отображением статуса получения.
  2. Кнопка «Back» - выход назад в главное меню

### Окно статистика:

* 1. Количество смертей за игру.
  2. Лучшее время прохождения каждого уровня.
  3. Кнопка «Back» - выход назад в главное меню

### Меню уровней – «Карта»:

* 1. Демонстрация всех уровней игры.
  2. Выбор доступного уровня текущего сохранения.
  3. Открытие доступа к уровням по мере прохождения игры.
  4. Кнопка «Back» - выход назад в главное меню
  5. Окно выбора доступного оружия.
  6. Окно выбора доступного усиления.

### Игровой уровень:

* 1. Игровой процесс происходит на 2D поле с видом сбоку.
  2. Поле представляет собой прямоугольную арену, на которой находятся платформы и ловушки.
  3. Игровой процесс на каждом уровне начинается с появления главного персонажа, противника (Босса), ловушек и платформ.
  4. Цель каждого уровня – победить Босса локации.
  5. Окно «Пауза»:
     1. Вызывается нажатием кнопки Esc.
     2. Кнопка «Exit» - выход в «Главное меню»
     3. Кнопка «Retry» - начинает уровень заново
     4. Кнопка продолжить игру.
  6. Окно «Победа»:
     1. Вызывается после победы над Боссом.
     2. Демонстрация сообщения о победе
     3. Отображение времени прохождения уровня
     4. Кнопка «Exit» - выход в «Главное меню»
     5. Отображение в окне «Карта» статуса прохождения уровня и открытие следующего.
  7. Окно «Поражения»:
     1. Вызывается при уменьшении очков здоровья до 0 у главного персонажа.
     2. Демонстрация сообщения о поражении
     3. Кнопка «Exit» - выход в «Главное меню»
     4. Кнопка «Retry» - начинает уровень заново
  8. Противники (боссы):
     1. Каждый босс имеет свой Hit Box (Box Collider), по которому должен проходить урон.
     2. Различные анимации атак босса
     3. Босс должен умирать при количестве очков здоровья равному 0
     4. Атаки босса должны наносить урон главному герою, который от этого теряет очки здоровья

## 3.2. Описание и обоснование алгоритма и функционирования программы

### 3.2.1. Описание архитектуры

Игра реализована на игровом движке Unity, так как для большинства 2D игра в данном жанре используют именно его в силу удобства и простоты. Язык программирования выбран C# в соответствии с требованиями Unity.

Структура проекта:

1. В Attacks хранятся все противники, которые могут нанести урон главному персонажу
2. В папке Img хранятся спрайты для кнопок, бэкграундов
3. В папке Player хранятся префаб, скрипты и спрайты главного героя
4. В папке Scenes хранятся сцены уровней, меню, достижений.
5. В папке Settings хранятся настройки сцен.
6. В папке TextMesh Pro хранятся пакеты Unity для текста.
7. В папке Scripts хранятся скирпты окон.

Изображение выглядит как текст, Шрифт, снимок экрана, дизайн

Контент, сгенерированный ИИ, может содержать ошибки.

Виды сцен:

Main Menu

Изображение выглядит как текст, снимок экрана, меню

Контент, сгенерированный ИИ, может содержать ошибки.Изображение выглядит как текст, рождественская елка, снимок экрана, книга

Контент, сгенерированный ИИ, может содержать ошибки.

1. Main camera – камера сцены
2. Canvas – содержит UI окна такие как: Создание новой игры (New Game), Продолжение текущего сохранения (Continue), меню выбора уровня и снаряжения (Levels selector), меню настроек (Settings), кнопку выхода (Exit), меню достижений (Achievements).
3. Audio Manger – объект, ответственный за звук в меню

Все объекты хранятся на одной сцене, что обеспечивает моментальный переход между окнами без каких-либо задержек. Также факт того, что все находится в главном меню, облегчает навигацию пользователя в игре.

Level XX:

Изображение выглядит как текст, программное обеспечение, Мультимедийное программное обеспечение, снимок экрана

Контент, сгенерированный ИИ, может содержать ошибки.

1. Main camera – камера сцены
2. Global Light 2D – объект, отвечающий за свет.
3. Platformes – объект, отвечающий за границы арены и находящиеся на ней платформы
4. Player – префаб главного героя
5. First Boss – префаб Босса уровня
6. Attack manager – объект, отвечающий ща фазы атак босса и его здоровье
7. Attack – объект, хранящий в себе все атаки, которые запускает босс уровня
8. Canvas – объект, отвечающий за появление окна по результатам уровня победа/поражение, пауза.
9. GameWin, GameOver, VictoryCanvas – появляющиеся объекты по результатам уровня.

Структура уровня однотипная, т. е. все уровни имеют такую же структуру, но другие механики и спрайты боссов, расположение платформ. Такая архитектура позволяет быстро масштабировать количество уровней в игре.

### 3.2.2. Описание алгоритма и функционирования программы

### Описание главного меню

Главное меню является первой экранной сценой, с которой взаимодействует игрок при запуске игры. Оно обеспечивает доступ ко всем ключевым разделам игры такие как: меню выбора сохранений, меню выбора уровней, меню настроек, окно достижений, окно статистики.

### Описание меню выбора сохранений

Меню выбора сохранений является панелью на сцене главного меню. Оно позволяется выбрать игроку, куда будет сохраняться прогресс прохождения игры.

### Описание меню выбора уровней

Меню уровней является панелью на сцене главного меню. Оно позволяет выбрать игроку доступные в текущем сохранении уровни, которые открываются по мере прохождения игры.

Также на этой панели внизу отображаются доступные усиления, которые пользователь выбирает для упрощения прохождения уровня.

Дополнительно на этой панели должна отображаться панель выбора доступного оружия.

### Описание меню настроек

Меню настроек является панелью на сцене главного меню. Оно позволяет изменять параметры такие как: громкость в игре и яркость игры.

### Описание окна достижений

Окно достижений является панелью на сцене главного меню. Оно позволяет игроку просмотреть все достижения и их условия получения. Достижения, полученные по мере прохождения игры пользователем, подсвечиваются.

### Описание окна статистики

Окно статистики является панелью на сцене главного меню. Оно позволяет просматривать время и количество попыток (смертей) потраченных на прохождение всех уровней.

### Описание здоровья

Отображение количества очков здоровья происходит с помощью UI объектов Unity. Каждый имеющийся Hit Point обозначается полным сердцем. Их можно восполнить, убив соответствующего персонажа в процессе прохождения уровня. Игрок теряет очки здоровья при получении урона от объектов, наносящих урон.

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

**Входные данные**

Программа читает прогресс прохождения игры (доступ к уровням) из внешнего файла.

Для перехода между окнами главного меню, изменения настроек, для выбора уровня в меню уровней, для выбора действий при завершении уровня нужно нажать на мыши ЛКМ.

**Выходные данные**

Программа сохраняет прогресс прохождения игры во внешний файл.

Программа отображает прогресс прохождения игры в меню выбора уровня.

Программа отображает доступные усиления в меню выбора уровня.

Программа отображает доступные виды оружия в меню выбора уровня.

Программа отображает процесс перемещения и нанесения урона главным героем.

Программа отображает процесс нанесения урона Боссу уровня, а также отображает различные фазы атак и перемещения Босса уровня.

**Обоснование метода организации входных и выходных данных**

Текущий способ организации данных является распространённым для игр, который соответствует требованиям ТЗ.

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

### 3.4.1. Состав технических и программных средств

Для нормальной производительности игры требуется устройства

* CPU: Intel Core i3 8100 или AMD Ryzen 1600
* RAM: 8 ГБ и выше
* GPU: NVIDIA GeForce GTX 960 или AMD RX 460
* DirectX: Версия 9.0 и выше
* ОС: Windows 10 или выше
* HDD: 20 ГБ свободного пространства**.**
* **Экран:** приложение адаптировано для экранов с разрешением 1920x1080.
* **Периферия**: компьютерная мышь и клавиатура.

Программные средства:

* Unity (версия 6000.1.1f1) – последняя версия.
* Среда разработки с поддержкой C# (Rider/Visual Studio)

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

В качестве операционной системы выбрана Windows, поскольку она является наиболее популярной платформой для ПК-игр.

При несоблюдении минимальных PC, во время игры могут возникать проблемы с производительностью.

В качестве IDE был выбран Rider JetBrains, т. к. есть интеграция с Unity, которая обеспечивает инструментами автозаполнения, отладки, анализа кода и навигации по проекту.

# 4. Технико-экономические показатели

## 4.1. Предполагаемая потребность

Предполагаемая потребность Пользователи, заинтересованные в сложных 2D играх жанра "run and gun, получат новую игру с уникальной атмосферой и механиками. Игра также может привлечь внимание speedrun-сообщества.

## 4.2. Ориентировочная экономическая эффективность

В рамках данной работы расчет экономической эффективности не проводился.

# Приложение 2

## Список используемой литературы

1. ГОСТ 19.101-77 Виды программ и программных документов. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
2. ГОСТ 19.301-79 Программа и методика испытаний. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
3. ГОСТ 19.401-78 Текст программы. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
4. ГОСТ 19.505-79 Руководство оператора. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
5. ГОСТ 19.404-79 Пояснительная записка. //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. Unity, [Электронный ресурс] / Unity. Режим доступа: <https://docs.unity.com/>, свободный. (дата обращения: 10.01.2025)
9. GitHub, Inc. [Электронный ресурс] / GitHub. Режим доступа: <https://github.com/>, свободный. (дата обращения: 10.01.2025).
10. C#, [Электронный ресурс] / C#. Режим доступа:

[https://learn.microsoft.com/ru-ru/dotnet/csharp/,](https://learn.microsoft.com/ru-ru/dotnet/csharp/) свободный. (дата обращения: 10.01.2025)

# Приложение 2

**Терминология**

|  |  |
| --- | --- |
| Термин | Значение |
| Босс | Особенно сильный и сложный противник, который обычно появляется в ключевых моментах игрового процесса, таких как завершение уровня. |
| OS | Операционная система |
| Стриминг | Технология передачи данных в реальном времени, позволяющая воспроизводить аудио, видео или другой контент на устройстве пользователя без необходимости полной загрузки. |
| Speedrun | Прохождение видеоигры или её части с максимально возможной скоростью, обычно с целью установить рекорд времени. |
| Стример | Человек, который ведёт прямые трансляции (стримы) через интернет, демонстрируя игровую сессию. |
| СPU | Центральный процессор |
| RAM | Оперативная память |
| SSD | Твердотельное запоминающее устройство, которое использует интегральные схемы для постоянного хранения данных |
| Ячейка сохранения | Это отдельная область данных, в которую сохраняется текущее состояние игры. Каждая ячейка позволяет игроку начать, продолжить или перезаписать своё прохождение, не затрагивая другие сохранения. |
| Canvas | Компонент Unity, отвечающий за отображение элементов пользовательского интерфейса на экране. Все UI-элементы размещаются внутри Canvas. |
| Prefab | Шаблон объекта, который можно многократно использовать в сценах. Содержит заранее настроенные компоненты, поведение и внешний вид. |
| Scene | Отдельный уровень, содержащий игровые объекты, окружение и логику. |
| Collider | Компонент, определяющий физические границы объекта. |
| Script | Скрипт — это файл с кодом, который управляет поведением объектов в игре. |
| ЛКМ | Левая кнопка мыши |
| IDE | Приложение для разработки кода |
| PC | Персональный компьютер |
| Canvas | Оболочка для всего UI |

**Описание и функциональное назначение полей, методов, свойств и классов**

|  |  |  |
| --- | --- | --- |
| Класс Lvel Menu | | |
| Метод | Тип | Назначение |
| OpenLevel | Void | Загрузка сцен с уровнями |

|  |  |  |
| --- | --- | --- |
| Класс Pause Menu | | |
| Метод | Тип | Назначение |
| Update | Void | Проверка нужно ли ставить на паузу игру |
| Resume | Void | Продолжить игру |
| Pause | Viud | Поставить игру на паузу |
| Load Menu | Void | Загрузка окна Паузы |

|  |  |  |
| --- | --- | --- |
| Класс Victory manager | | |
| Метод | Тип | Назначение |
| Show Victory | Void | Загрузка окна победы |
| Exit | Void | Выход из уровня и загрузка главного меню |

|  |  |  |
| --- | --- | --- |
| Класс GameOverManager | | |
| Метод | Тип | Назначение |
| Show Game Over | Void | Загрузка окна поражения |
| Exit | Void | Выход из уровня и загрузка главного меню |
| Retry | Void | Повторная загрузка уровня |

|  |  |  |
| --- | --- | --- |
| Класс Achievement Manager | | |
| Метод | Тип | Назначение |
| Unlock Achievement | Void | Регистрации достижения |
| Show Achievement Panel | Void | Демонстрация окна со всеми достижениями |
| IsUnlocked | Void | Проверка получения достижения |
| Collected Achievment | List | Список полученных достижений |

|  |  |  |
| --- | --- | --- |
| Класс Stats Manager | | |
| Метод/Поле | Тип | Назначение |
| Total Time | float | Подсчет общего времени прохождения |
| Total death | int | Подсчет общего количества смертей |
| Start | Void | Начало подсчета времени |
| Update | Void | Изменение времени |
| RegisterDeath | Void | Изменение числа смертей |

|  |  |  |
| --- | --- | --- |
| Класс Settings Manager | | |
| Метод/Поле | Тип | Назначение |
| brightness | float | Сохранение уровня яркости |
| Volume | Float | Сохранение уровня громкости |
| Start | Void | Начало работы меню настроек |
| Set brightness | Void | Изменение яркости |
| Set Volume | Void | Изменение громкости |
| Instance | Settings Manager | Объект настроек |

# ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Лист регистрации изменений | | | | | | | | | |
| Изм. | Номера листов | | | | Всего листов (страниц) в документе | № документа | Входя- щий № сопро- водит. докум. и дата | Под- пись | Дата |
| изменен- ных | заменен- ных | новых | аннулиро- ванных |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |