

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное  
образовательное учреждение высшего образования  
«Южно-Уральский государственный университет  
(национальный исследовательский университет)»

Высшая школа электроники и компьютерных наук  
Кафедра «Электронные вычислительные машины»

ДОПУСТИТЬ К ЗАЩИТЕ  
Заведующий кафедрой ЭВМ  
\_\_\_\_\_ Г.И. Радченко  
« \_\_\_ » \_\_\_\_\_ 2020 г.

Тренажер для коррекции зрения в игровой форме

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА  
К ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЕ

Руководитель работы,  
к.т.н., доцент каф. ЭВМ  
\_\_\_\_\_ Е.С. Ярош  
« \_\_\_ » \_\_\_\_\_ 2020 г.

Автор работы,  
студент группы КЭ-405  
\_\_\_\_\_ И.А. Плотников  
« \_\_\_ » \_\_\_\_\_ 2020 г.

Нормоконтролёр,  
ст. преп. каф. ЭВМ  
\_\_\_\_\_ 2020 г.  
« \_\_\_ » \_\_\_\_\_

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное  
образовательное учреждение высшего образования  
«Южно-Уральский государственный университет  
(национальный исследовательский университет)»  
Высшая школа электроники и компьютерных наук  
Кафедра «Электронные вычислительные машины»

УТВЕРЖДАЮ  
Заведующий кафедрой ЭВМ  
\_\_\_\_\_ Г.И. Радченко  
« \_\_\_ » \_\_\_\_\_ 2020 г.

### **ЗАДАНИЕ**

**на выпускную квалификационную работу бакалавра**  
студенту группы КЭ-405  
Плотников Игорь Александрович  
обучающемуся по направлению  
09.03.01 «Информатика и вычислительная техника»

1. **Тема работы:** «Тренажер для коррекции зрения в игровой форме»  
утверждена приказом по университету от 24 апреля 2020 г. №627
2. **Срок сдачи студентом законченной работы:** 25 мая 2020 г.
3. **Исходные данные к работе:** статьи, книги, техническое задание.
  - обеспечить коррекцию амблиопии;
  - пациентами являются дети дошкольного и младшего школьного  
возраста;
  - методика лечения должна соответствовать прибору Амблиотренер;
  - платформа Microsoft Windows не ниже 7.

Источники:

- Аветисов, Э.С. Дисбинокулярная амблиопия и ее лечение. М. – Медицина, 1968. –207 с.;
- Шамшинова, А.М. Амблиопия: патогенез, дифференциальная диагностика и основные принципы лечения / А.М. Шамшинова, Т.П. Кащенко, У. – Кампор // Клиническая физиология зрения. – М.: Науч. – мед. фирма МБН. 2002. – С. 473 – 480.

**4. Перечень подлежащих разработке вопросов:**

- анализ поставленной задачи, планирование ее решения;
- обзор и анализ аналогов;
- выбор средств разработки;
- архитектурное проектирование;
- программная реализация;
- тестирование разработанного программного продукта

**5. Дата выдачи задания:** 25 декабря 2019 г.

Руководитель работы \_\_\_\_\_ /Е.С. Ярош/

Студент \_\_\_\_\_ /И.А. Плотников /

## КАЛЕНДАРНЫЙ ПЛАН

Этап	Срок сдачи	Подпись руководителя
Введение и обзор литературы	01.03.2020	
Разработка модели, проектирование	01.04.2020	
Реализация системы	01.05.2020	
Тестирование, отладка, эксперименты	15.05.2020	
Компоновка текста работы и сдача на нормоконтроль	20.05.2020	
Подготовка презентации и доклада	25.05.2020	

Руководитель работы \_\_\_\_\_ /Е.С. Ярош/

Студент \_\_\_\_\_ /И.А. Плотников/

## Аннотация

И.А. Плотников. Тренажер для коррекции зрения в игровой форме. – Челябинск: ФГАОУ ВО «ЮУрГУ (НИУ)», ВШЭЖН; 2020, 77 с., 33 ил., библиогр. Список – 11 наим.

В рамках выпускной квалификационной работы производится изучение и анализ требований к программным продуктам для коррекции зрения, рассматриваются преимущества и недостатки существующих программных комплексов, описывается процесс разработки программного комплекса для тренировки и коррекции зрения детей в игровой форме и полученные результаты. Доказывается способность разрабатываемого программного продукта конкурировать с существующими программными продуктами.

## ОГЛАВЛЕНИЕ

1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ.....	9
1.1. ОБЗОР АНАЛОГОВ .....	9
1.2. ВЫБОР СРЕДЫ РАЗРАБОТКИ.....	12
1.3. ВЫВОД.....	20
2. ОПРЕДЕЛЕНИЕ ТРЕБОВАНИЙ И ОСНОВНЫХ КОНЦЕПЦИЙ .....	21
2.1 ФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ.....	21
2.2 НЕФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ .....	22
2.3 КОНЦЕПЦИЯ ПРОГРАММЫ .....	23
2.4 КОНЦЕПЦИЯ ИНТЕРФЕЙСА .....	24
2.5 ВЫВОД.....	28
3. ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ .....	29
3.1. ДИАГРАММА ПРЕЦЕДЕНТОВ.....	29
3.2. РЕАЛИЗАЦИЯ .....	35
3.3. ФАЙЛОВАЯ СТРУКТУРА .....	35
3.3. ОБЪЕКТЫ .....	37
3.4. ИСПОЛЬЗУЕМЫЕ КЛАССЫ .....	40
3.5. ПРОЦЕСС ТРЕНИРОВКИ .....	44
3.6. ВЫВОДЫ .....	47
4. ТЕСТИРОВАНИЕ .....	48
4.1. ФУНКЦИОНАЛЬНОЕ ТЕСТИРОВАНИЕ .....	48
5. ЗАКЛЮЧЕНИЕ .....	56
БИБЛИОГРАФИЧЕСКИЙ СПИСОК.....	57
ПРИЛОЖЕНИЕ А .....	58

## ВВЕДЕНИЕ

### **Актуальность темы**

В течение многих лет амблиопия является одной из самых актуальных проблем в детской офтальмологии. Данное заболевание является одной из частых причин слабого зрения у детей. По данным разных авторов, распространённость различных видов амблиопии среди детей дошкольного возраста достигает 6 %, а у детей школьного возраста составляет 2–3 %. Несмотря на то, что низкое зрение у школьников чаще всего обусловлено миопией, гиперметропическая рефракция занимает первое место среди аномалий рефракции в этой возрастной группе. При отсутствии рациональной коррекции гиперметропия ведёт к развитию рефракционной амблиопии, которая с возрастом может перейти в дисбинокулярную и осложниться косоглазием.

Основным методом лечения амблиопии является тренировочный. В результате тренировок происходит коррекция зрения.

Коррекция зрения с применением программных продуктов является весьма актуальной задачей, так как этот способ не требует больших затрат и доступен широкому кругу людей с проблемами в области зрительной системы.

На данный момент существует некоторое количество программ и программных комплексов для коррекции и поддержания остроты зрения у детей. Но у большей части из них имеется ряд недостатков, связанных с фокусировкой внимания детей на тренажере. Также большая часть из этих программ значительно устарела и не поддерживает современные операционные системы. Практически все имеющиеся программы распространяются на платной основе.

## **Цели и задачи**

Целью, представленной выпускной квалификационной работы, является разработка программного продукта, обеспечивающего коррекцию зрения детей.

Для достижения поставленной цели необходимо решить следующие задачи:

1. Провести анализ существующих программных продуктов для поддержания и коррекции зрения;
2. На основе рекомендаций врача-офтальмолога и результатов анализа п. 1 сформулировать подробные требования к разрабатываемому программному обеспечению;
3. Разработать архитектуру программного продукта, осуществить обоснованный выбор средств его реализации;
4. Выполнить программную реализацию в соответствии с предъявляемыми требованиями;
5. Выполнить тестирование разработанного программного обеспечения.



# 1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

Основные требования:

- лечение амблиопии;
- дошкольный и младший школьный возраст пациентов;
- игровой характер лечения;
- наличие информации о результатах прохождения тренировок;
- вывод информации о результатах пройденных тренировок в отдельный файл;
- совместимость с операционными системами не ниже Microsoft Windows 7.

На сегодняшний день существует множество программ и программных комплексов для коррекции и поддержания остроты зрения взрослых и детей.

Данные программы решают широкий спектр задач, связанных с коррекцией зрения, таких как: косоглазие, бинокулярные расстройства, амблиопия. Однако у большей части данных программ присутствует ряд недостатков:

- проблема фокусировки внимания пациента, который отвлекается от процесса тренировки;
- проблема отслеживания динамики пациента;
- отсутствие поддержки современных операционных систем;
- проприетарность.

## 1.1. ОБЗОР АНАЛОГОВ

На сегодняшний день существует несколько программ, обеспечивающих лечение и коррекцию амблиопии детей дошкольного и младшего школьного

возраста, использующихся на данный момент в клиниках и на дому на персональном компьютере.

Было решено провести анализ только тех программных продуктов, чьи цели и задачи схожи с целями и задачами предлагаемого программного продукта.

**Программа "Чибис"** – позволяет осуществлять тестовые и тренировочные процедуры для оценки состояния бинокулярного стереозрения и функционального лечения бинокулярных расстройств. Лечебное действие программы – это стимуляция координированной деятельности левого и правого зрительных каналов за счет использования чисто бинокулярных зрительных стимулов – стереограмм из случайных точек, которые могут успешно восприниматься только при согласованной работе двух глаз;

**Программа "ЦВЕТОК"** – интерактивная тренировочная программа, имеющая игровой характер. Программа предлагает пациенту серию усложняющихся, но однотипных зрительных упражнений, состоящих в поиске заданного объекта среди нескольких объектов, предъявляемых на лепестках цветка.

**Программа "еУе" ("Ай")** – предназначена для диагностики и лечения амблиопии и косоглазия, восстановления и развития бинокулярного зрения. В основу упражнений положены методы плеоптики, ортоптики и диплоптики. Разделение полей зрения осуществляется с помощью красно-синих очков.

**Программа "Контур"** – программа для лечения амблиопии, восстановления и развития бинокулярного зрения. В бинокулярных упражнениях пациент в красно-синих очках, видя опорный рисунок одним глазом, обводит или дорисовывает его "пером", видимым другим

глазом. Всего в программе 38 рисунков различной сложности. Для устранения функционального подавления и тренировки фузии предусмотрены многочисленные регулировки: изменяется соотношение контрастов объектов и толщина линий рисунка и "пера", объекты могут независимо переключаться из светлых в темные, включается мигание с управляемой частотой, регулируется сила стимула для периферической фузии.

**Программа "Крестики"** – это игровой паттерн-стимулятор для лечения амблиопии, в котором используется инвертирующееся шахматное поле. Этот стимул активизирует нейроны и восстанавливает межнейронные связи на всех уровнях зрительной системы. В ходе игры клетки уменьшаются до порога различения их игроком. Частоты инвертирования связаны с размером клеток, обеспечивая наибольший лечебный эффект. Контраст изображения можно выбирать в широких пределах: от максимального для стимуляции, до едва различимого для тренировки. Для воздействия на яркостной и цветооппонентные каналы зрительной системы используются черно-белые, красно-зеленые или желто-синие шахматные поля. Продолжительность одной игры — 5 минут.

**Программа "Паучок"** – еще одна игра для лечения амблиопии, где стимуляция производится структурированными динамическими изображениями. Игровая задача побуждает пациента удерживать взор вблизи центров подвижных радиальных или спиральных решеток. В результате макула и периферия одновременно возбуждаются стимулами оптимальных размеров: макула – мелкими, периферия – крупными. Также в процессе игры активизируются визомоторная деятельность, конвергенция и аккомодация. Различные настройки и цветовые переходы

позволяют стимулировать все типы фоторецепторов, а также on- и off-нейроны. Контраст изображения можно выбирать.

Кроме вышеперечисленных программ также существуют следующие: **Color EyesKeeper, DAZZLE, Eye Corrector, FLASHER, Калейдоскоп, RELAX, SafeEyes, STRABISMUS, VISUS, Головоломы, Доска**, но они значительно менее популярны и не относятся к рассматриваемому заболеванию.

Таблица 1.1 - Аналоги

	Фокусировка внимания	Отслеживание результатов	Поддержка современных ОС	Проприетарность
<b>Чибиc [5]</b>	-	-	-	-
<b>ЦВЕТОК [6]</b>	-	+	-	+
<b>eYe [7]</b>	-	-	-	-
<b>Контур [7]</b>	-	-	+	-
<b>Крестики [8]</b>	-	+	-	+
<b>Паучок [8]</b>	-	+	-	+

Из таблицы 1.1 видно, что ни одна из представленных на рынке программ не обладает заданным набором требований, в связи с этим актуальность данной разработки не вызывает сомнений.

## 1.2. ВЫБОР СРЕДЫ РАЗРАБОТКИ

Для достижения целей данной выпускной квалификационной работы необходимо выбрать среду разработки. При коррекции зрения используются

графические изображения. Также требуется удерживать внимание пользователя программного продукта таким образом, чтобы соблюдалась методика тренировки. Наиболее подходящим для этого является игровой подход. Поэтому рассматривались игровые движки и сопутствующее им системное программное обеспечение.

Рассматривались следующие варианты:

- Unity 2019.3.2f1 [1];
- CryEngine 3 [3];
- Unreal Engine 4 [2].

### **Unity 2019**

Unity 3D — это мощная среда для разработки 3D игр и приложений. Данная платформа создана в 2005 году в Дании. Главный плюс Unity 3D — простота разработки приложений. В данной среде разрабатывается огромное количество игр под различные платформы.

Unity очень тесно сотрудничает с Oculus, Apple и рядом других компаний, имеет очень мощный магазин Unity Asset Store, в котором есть абсолютно все: от текстур и 3D моделей до готовых проектов.

Unity поставляется с полностью интегрированными физическими движками NVIDIA® PhysX® и Box2D. При этом Unity сам выбирает, какой физический движок использовать в зависимости от контекста. Одним из 14 несомненных преимуществ использования платформы Unity является ее подробная документация с исчерпывающим описанием всех функциональных возможностей и примерами их применения. Также на сервисе Youtube есть официальный канал Unity с руководствами по применению функций Unity и огромное количество обучающих видео, записанных самими пользователями. Интерфейс платформы Unity приведен на рисунке 1.1.

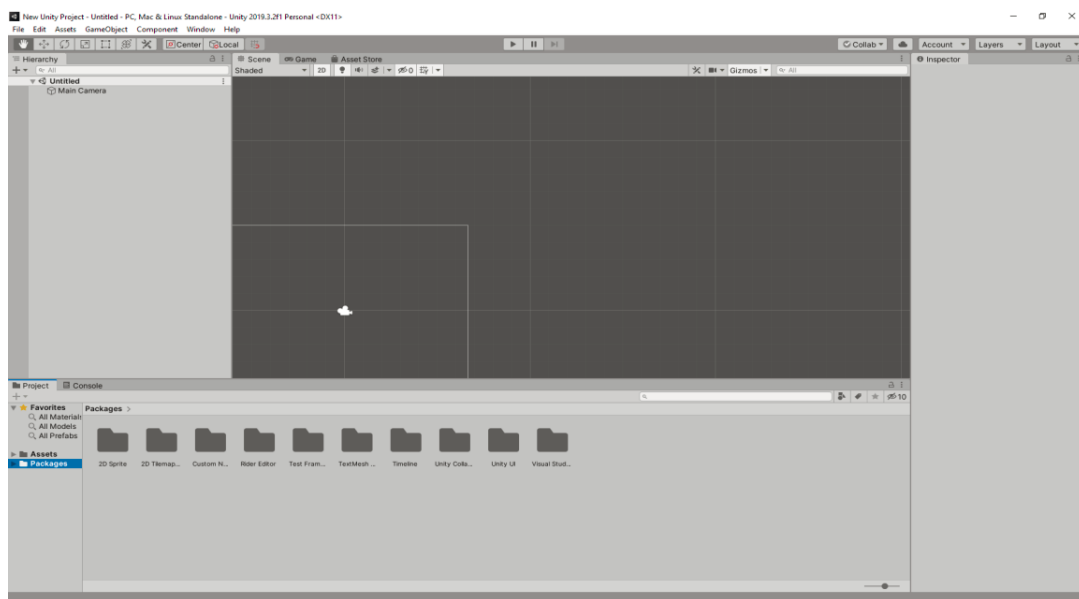


Рисунок 1.1 – Пример интерфейса Unity

#### Основные возможности и плюсы Unity 3D:

- доступный и понятный интерфейс;
- поддержка двух языков программирования: C# и JavaScript, на которых пишутся скрипты;
- большое сообщество;
- полная интеграция платформы со средой разработки;
- поддержка перетягивания объектов в редакторе;
- поддержка импорта большого количества форматов;
- поддержка физики ткани (PhysX Cloth);
- возможность дополнения функционала;
- возможность использования систем контроля версий.

#### Минусы:

- ограниченный набор инструментов;
- процесс изготовления игры отнимает много времени;
- не самая лучшая графика.

### CryEngine 3

CryEngine 3 –разработка компании Crytek. Cry Engine 3 создана в 2001 году, когда была анонсирована первая разрабатываемая на нем игра Far Cry. Разработчики этой платформы изначально преследовали цель не самим создавать на нем игры, а продавать его как технологию. CryEngine, в отличие от других платформ, нацелен на создание игр для PC и консолей. Это мощная платформа, которая выдает лучшую графику из всех трех выбранных сред разработки. Имеется динамический свет, затенение, затуманивание в реальном времени и управление уровнем детализации ландшафта. Лица и персонажи получаются по-настоящему захватывающими. С помощью данной среды были созданы такие игры как: Crysis, Far Cry, Warface, Star Citizen, Ryse: Son of Rome, Homefront: The Revolution, Evolve, Проект Армата, Kingdom Come: Deliverance.

Существует несколько способов использования платформы третьими студиями. Первый – для использования дома или для создания бесплатных игр, с целью изучения и без извлечения прибыли. Вторым вариантом – для коммерческого использования. В этом случае используется лицензия royalty only 20%. То есть необходимо отдать 20 процентов дохода. Третий – для инди и загружаемых игр. Четвертый вариант лицензии – для крупных игр, ее стоимость необходимо обговаривать индивидуально, она является коммерческой тайной.

CryEngine 3 ориентирован на создание кроссплатформенных игр, предназначенных для PC и консолей. В нем изначально присутствует поддержка глобальных мультиплеерных игр. CryEngine 3 обладает большим списком технологий визуализации. Физический компонент движка CryPhysics также работает независимо от физических API, таких как PhysX. Встроенная система анимации предлагает несколько отличных подсистем: индивидуализация персонажей, параметрическая скелетная анимация, процедурное деформирование движения.



В комплекте есть то, что необходимо для реализации AAA-класса игр, в том числе редактор уровней, который способен создавать карты в реальном времени. Интерфейс платформы CryEngine 3 представлен на рисунке 1.2.

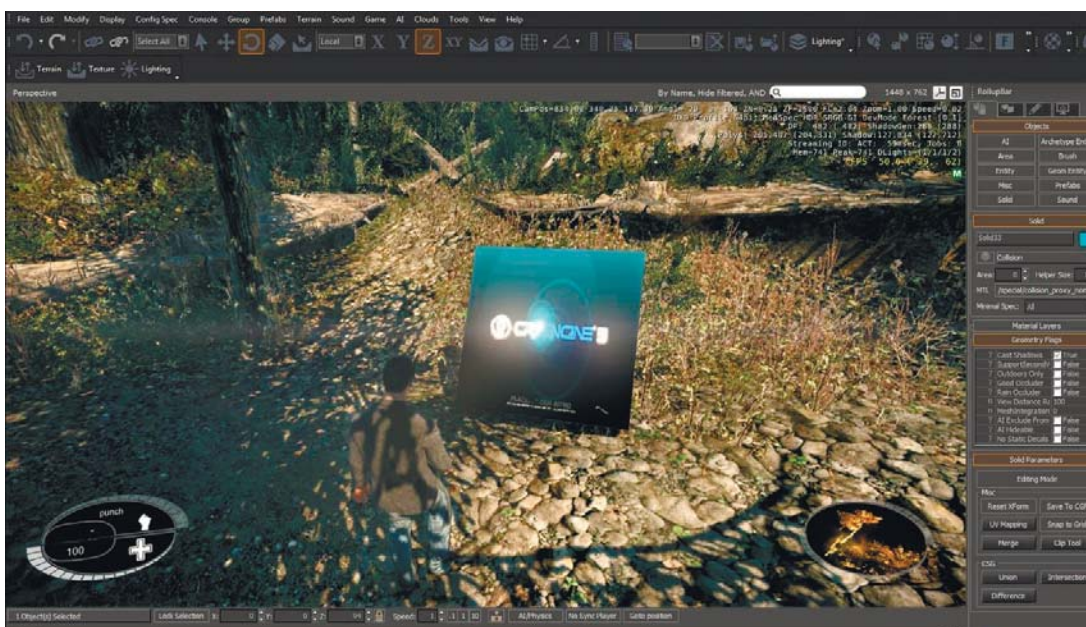


Рисунок 1.2 – Пример интерфейса CryEngine 3

#### Плюсы:

- возможность создание игры с отличной графикой;
- создание мощного звукового сопровождения;
- максимальная простота процесса создания искусственного интеллекта в сегменте;
- легкость создания пользовательского интерфейса начинающим разработчиком.

#### Минусы:

- слабая техническая поддержка бесплатной версии;
- поддержка только ОС Microsoft Windows;
- относительно высокий порог вхождения.



## Unreal Engine

Unreal Engine 4 – среда разработки, созданная Epic Games. Это самая популярная среда разработки для создания фильмов и AAA-проектов. Данная платформа обладает высокими графическими возможностями. С Unreal Engine 4 есть возможность разрабатывать игры под PC, Mac, консоли, IOS, Android.

В отличие от Unity, UE4 имеет мощный инструмент для дизайна игровых уровней прямо в сцене, достаточно удобную систему Blueprint, не имеющую аналогов, красивый дизайн самой платформы и интуитивность в использовании. Из всех сред разработки Unreal Engine 4 является самой инновационной. Он сочетает в себе высокую производительность, лучшую графику, простой язык программирования и удобство использования. Есть очень мощное сообщество, помогающее решить все проблемы. Epic Games раздает UE4 абсолютно бесплатно, весь исходный код открыт, за это разработчики просят лишь 5% с продажи одной копии игры. Команда Epic тесно сотрудничает с разработчиками VR технологий, а это значит, что уже сейчас можно создавать игры под PlayStation VR, Oculus и т.д.

На официальном сайте Unreal Engine есть огромное количество различной документации. Также есть раздел для тех, кто перешел с Unity3d. Магазин в Unreal Engine отстает от конкурентов. Если необходимо найти готовые 3D модели или сцены, то есть шанс, что нужные объекты не найдутся в фирменном магазине и придется их искать на различных интернет ресурсах или создавать самому. Кроме этого, если получится найти необходимые ресурсы, то придется отдать немалые деньги за возможность их загрузить, так как цены в официальном магазине достаточно высокие. Магазин Unity Asset Store в этом плане лучше, так как выбор в нем намного больше, а цены более демократичные.

На Unreal Engine 4 было сделано огромное количество AAA-проектов, таких как: Mortal Kombat X, Dead Island 2, Smite, Paragon, Infinity Blade, Gears of War, Batman: Arkham Asylum, Mass Effect. Данный движок написан на языке C++, разработан и поддерживается на протяжении 18 лет компанией Epic Games. Созданные игры и приложения на Unreal Engine работают на различных операционных системах и платформах. Интерфейс платформы Unreal Engine приложен на рисунке 1.3.



Рисунок 1.3 – Пример интерфейса Unreal Engine 4

#### Плюсы:

- структура ценообразования проста для понимания;
- большое сообщество;
- возможно, напрямую использовать в проекте файлы с исходным кодом на C++;
- отличная техническая поддержка и механизм апдейта;

- широкий ассортимент инструментов для различных целей;
- совместимость с различными платформами.

Минусы:

- сложно привыкнуть к определенным инструментам;
- небольшой выбор готовых инструментов в официальном магазине;
- относительно высокий порог вхождения.

### **Сравнение и выбор среды разработки**

При анализе средств разработки основными критериями были:

- новизна.
- даты рассматриваемых релизов и их практическое соответствие своему времени выхода;
- порог вхождения – оценка необходимых знаний для работы с выбранной платформой;
- поддерживаемые платформы и используемые языки;
- цена – хотя мы рассматриваем условно бесплатные среды разработки, у каждого есть полноценная коммерческая версия, цена на которую тоже имеет важное значение.

Исходя из данных параметров больше всего подходят платформы Unity и Unreal Engine.

Для написания скриптов было выбрано ПО Visual Studio 2019, поскольку оно идет в комплекте с выбранным ранее Unity 2019.

### **1.3. ВЫВОД**

Предлагаемый программный продукт должен выполнять задачи коррекции, тренировки зрения у детей с определенными отклонениями в зрительной системе, позволять осуществлять все традиционные процедуры аппаратного лечения, проводимого на синоптофоре. Программа должна быть построена по принципу имитации соответствующих процедур, но позволять расширить их временной и скоростной диапазон и использовать ряд новых режимов. Игровая форма тренировок позволит сделать процесс лечения интересным для детей, а, следовательно, и более качественным.

## **2. ОПРЕДЕЛЕНИЕ ТРЕБОВАНИЙ И ОСНОВНЫХ КОНЦЕПЦИЙ**

Для того чтобы перейти к этапу проектирования программного продукта необходимо сформулировать техническое задание с медицинской точки зрения. В качестве постановщика задачи выступает квалифицированный врач-офтальмолог, заведующая офтальмологическим отделением Челябинской областной детской клинической больницы Ядыкина Елена Владимировна.

### **2.1 ФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ**

Функциональные требования отражают возможности программного продукта в части выполнения основных целей и задач его создания. Они формулируются следующим образом:

- методика лечения должна быть основана на тренировках;
- в качестве методической основы взять прибор Амблиотренер;
- основной метод тренировки – прохождение лабиринта;
- конфигурация лабиринта должна меняться при неоднократном прохождении одного и того же уровня;
- общее количество уровней – не менее 5;
- при успешном прохождении уровня должны быть предусмотрены призы;
- возможность при загрузке приложения выбрать уровень, настроить сложность в соответствии со своей степенью амблиопии и узнать количество накопленных на данный момент игровых баллов,
- по запросу показ статистики за предыдущие сеансы тренировки и вывод её в отдельный файл;

- возможность в процессе работы настроить уровень сложности в соответствии со степенью своей амблиопии;
- возможность перемещаться между уровнями, проходить их, зарабатывать баллы, разблокировать новые уровни.

Приложение должно функционировать на ОС Microsoft Windows не ниже 7.

## **2.2 НЕФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ**

После загрузки программы пользователь должен иметь доступ в меню выбора уровня, настроек и статистики. Также пользователь должен иметь возможность выйти из приложения и вернуться в главное меню.

В сцене главного меню необходимо реализовать следующие кнопки:

- "Выбор уровня";
- "Настройки";
- "Статистика";
- "Выход".

В соответствии с п. 2.1, пользователь должен иметь возможность увидеть статистику за предыдущие сеансы тренировки, а также вывести её в файл.

Необходимо предусмотреть возможность изменения настройки разрешения, установки полноэкранный режима, изменения языка интерфейса. Языком по умолчанию является русский.

Программа должна быть адаптируема мониторами с различным соотношением сторон.

Должна быть обеспечена патентная чистота в отношении территории Российской Федерации.

Программа должна быть зарегистрирована в Едином реестре российских программ для электронных вычислительных машин и баз данных.

## 2.3 КОНЦЕПЦИЯ ПРОГРАММЫ

Решение проблемы фокусировки внимания и заинтересованности в максимально правильном прохождении тренировочных заданий будет достигаться за счет награждения пациента баллами, которые одновременно будут являться валютой для разблокировки новых тематических уровней, а также внешнего вида игрового персонажа. Это повысит заинтересованность пользователя в максимально качественном прохождении тренировки.

Программа будет представлять собой аналог существующего аппаратного комплекса под названием Амблиотренер.

Амблиотренер – настольный прибор, представляющий собой корпус, на верхней наклонной плоскости которого расположена панель из органического стекла и пульт управления. В углубление панели помещают сменные решетки, платы и рисунки с опто типами, под панелью размещено семнадцать ламп. На боковой панели прибора расположены гнезда для подключения трансформатора, головного телефона и токопроводящего стержня.

Принцип лечения амблиопии таким способом заключается в фокусировке обеих глаз на двух объектах: игровом персонаже и дополнительном объекте (линии, яблоке, сосиске и так далее).

Программный продукт будет представлять собой набор определенных уровней, каждый из которых оформлен в той или иной тематике и имеет определенную траекторию прохождения.

Чтобы попасть на новый уровень, пациенту будет необходимо пройти предыдущий на максимально возможное количество баллов.

Чтобы обеспечить лечебный эффект для пациентов с разными степенями амблиопии, в разделе "Настройки" находятся несколько параметров, изменение которых влечет за собой повышение или понижение сложности тренировки.

В программе есть несколько уровней.

Первый базовый уровень выглядит как минималистично оформленный процедурно генерирующийся лабиринт с проходящей по нему от места появления игрока до финиша ломаной линией, состоящей из отрезков прямых. Максимально точное движение по этой линии позволяет пациенту зарабатывать баллы.

Уровень «Кубач» – это один из разблокируемых по ходу зарабатывания баллов уровней, оформление которого похоже на стилистику популярной игры "Minecraft".

Уровень «Яблочный лес» – это один из разблокируемых по ходу зарабатывания баллов уровней, оформленный в виде тропинки в лесу, по которой разбросаны яблоки.

Уровень «Кошкин дом» – это один из разблокируемых по ходу зарабатывания баллов уровней, оформленный в виде дома, по которому, следуя за сосисками, бежит кошка.

Уровень «Охота за подарками» – это один из разблокируемых по ходу зарабатывания баллов уровней, оформленный в виде тропинки в новогоднем лесу, по которой разбросаны подарки.

Основная задача пользователя на всех уровнях – следуя указателю добраться до финиша, набрав максимум баллов.

## **2.4 КОНЦЕПЦИЯ ИНТЕРФЕЙСА**

Интерфейс пользователя – разновидность интерфейсов, в котором одна сторона представлена человеком (пользователем), другая – машиной/устройством. Он представляет собой совокупность средств и методов, при помощи которых пользователь взаимодействует с различными, чаще всего сложными, машинами, устройствами и аппаратурой.



На рисунке 2.1 представлено схематичное изображение создаваемой сцены "Главное меню" с требуемыми элементами.



Рисунок 2.1 – Макет главного меню

В сцене меню выбора уровня необходимо реализовать следующие элементы:

- перечень уровней;
- цену за разблокирование уровней;
- полотно для предποказа уровня перед его выбором;
- общее количество баллов на данный момент;
- выход в главное меню.

На рисунке 2.2 представлено схематичное изображение создаваемого меню "Выбор уровня" с перечисленными элементами.

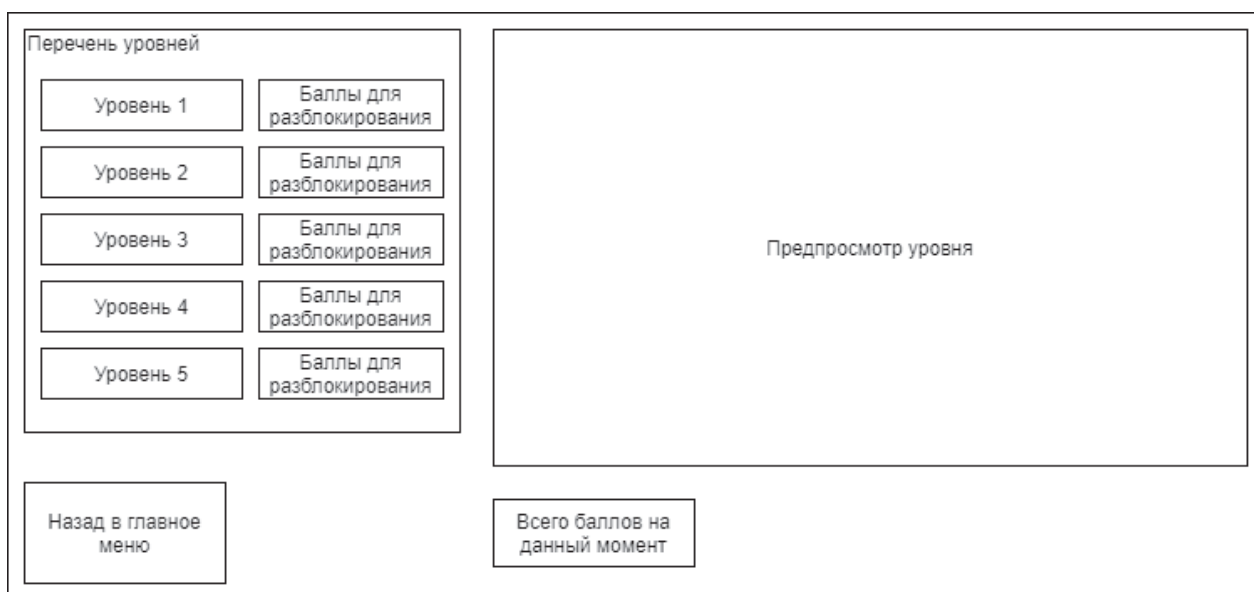


Рисунок 2.2 – Макет меню "Выбор уровня"

В сцене меню настройки планируется реализовать следующие элементы:

- настройку сложности;
- настройку полноэкранного режима;
- настройку разрешения;
- настройку языка;
- принятие настроек;
- выход в главное меню.

На рисунке 2.3 представлено схематичное изображение создаваемого меню "Настройки" с перечисленными элементами.

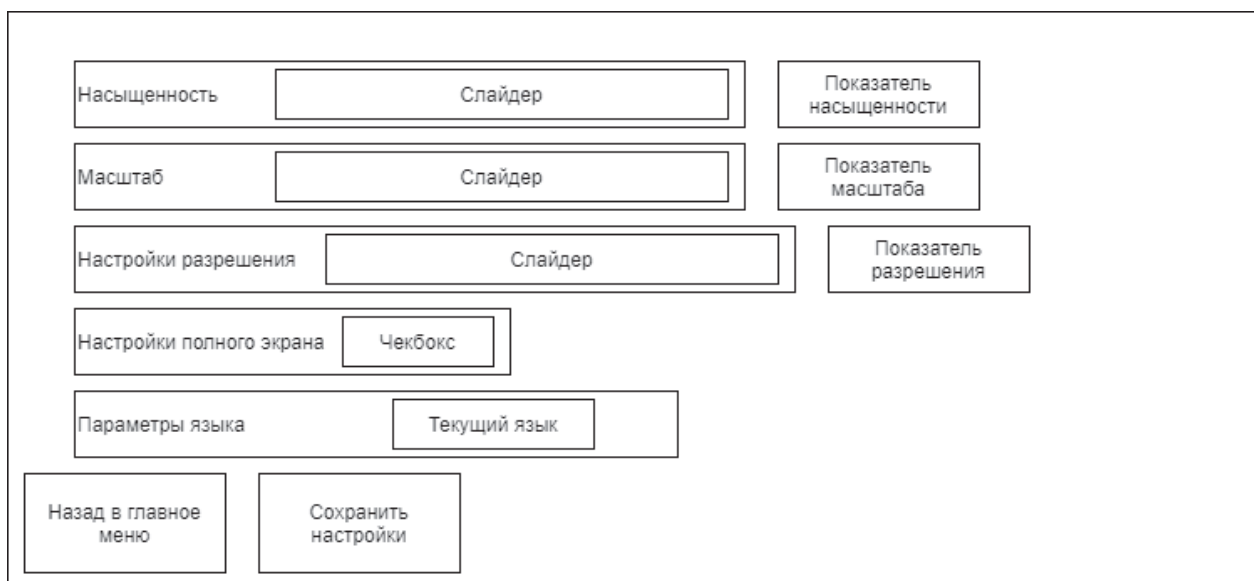


Рисунок 2.3 – Макет меню "Настройки"

В сцене меню статистики планируется реализовать следующие элементы:

- список результатов с указанием даты, времени и полученных за прохождение баллов;
- кнопку вывода результатов в отдельный файл;
- кнопку выхода в главное меню.

На рисунке 2.4 представлено схематичное изображение создаваемого меню "Статистика" с данными элементами.

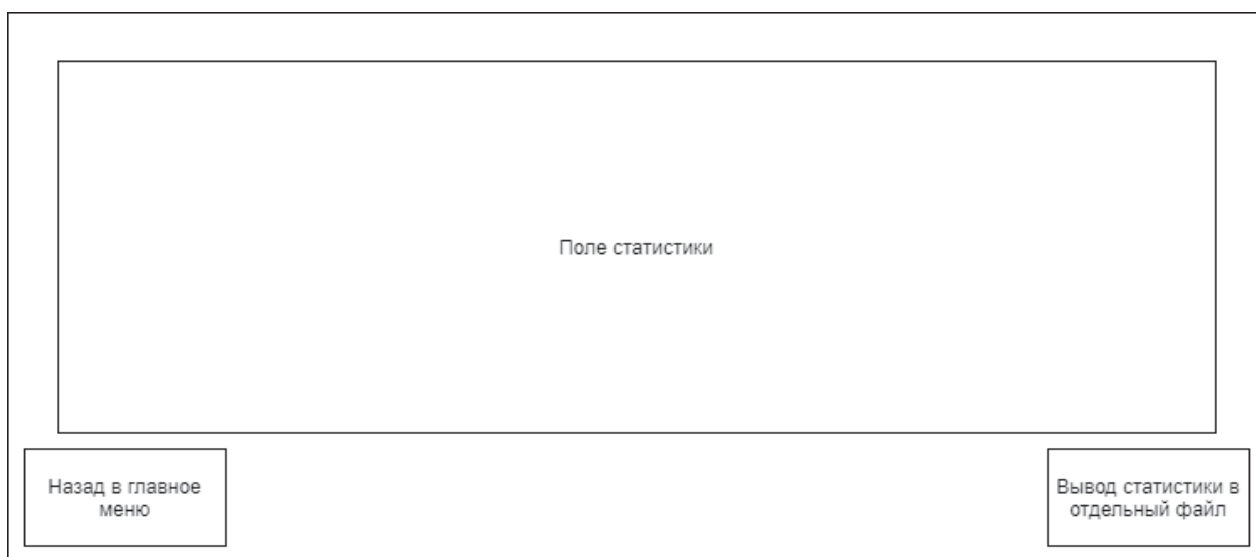


Рисунок 2.4 – Макет меню "Статистика"

## 2.5 ВЫВОД

В процессе постановки задачи были определены ключевые особенности процесса коррекции зрения при заболевании амблиопией, сформулированы требования к программному продукту функционального и нефункционального характера, составлена концепция программы, которую необходимо разработать, и концепция ее интерфейса.

## **3. ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ**

### **3.1. ДИАГРАММА ПРЕЦЕДЕНТОВ**

Диаграмма прецедентов UML, отражающая отношения между актёрами и прецедентами, позволяет описать систему на концептуальном уровне [9].

Прецедент – возможность моделируемой системы (часть её функциональности), благодаря которой пользователь может получить конкретный, измеримый и нужный ему результат. Прецедент соответствует отдельному сервису системы, определяет один из вариантов её использования и описывает типичный способ взаимодействия пользователя с системой. Варианты использования обычно применяются для спецификации внешних требований к системе.

Основным актером, взаимодействующим с системой, является «Пользователь», который взаимодействует с программой. Данный актер, находясь в разделе главного меню, может перейти в раздел настроек и настроить насыщенность и уменьшение объектов, также, вернувшись в главное меню и перейдя в раздел выбора уровней, пользователь может выбрать один из доступных ему уровней и начать тренировку, заработать баллы.

Диаграмма прецедентов меню программы приведена на рис. 3.1.

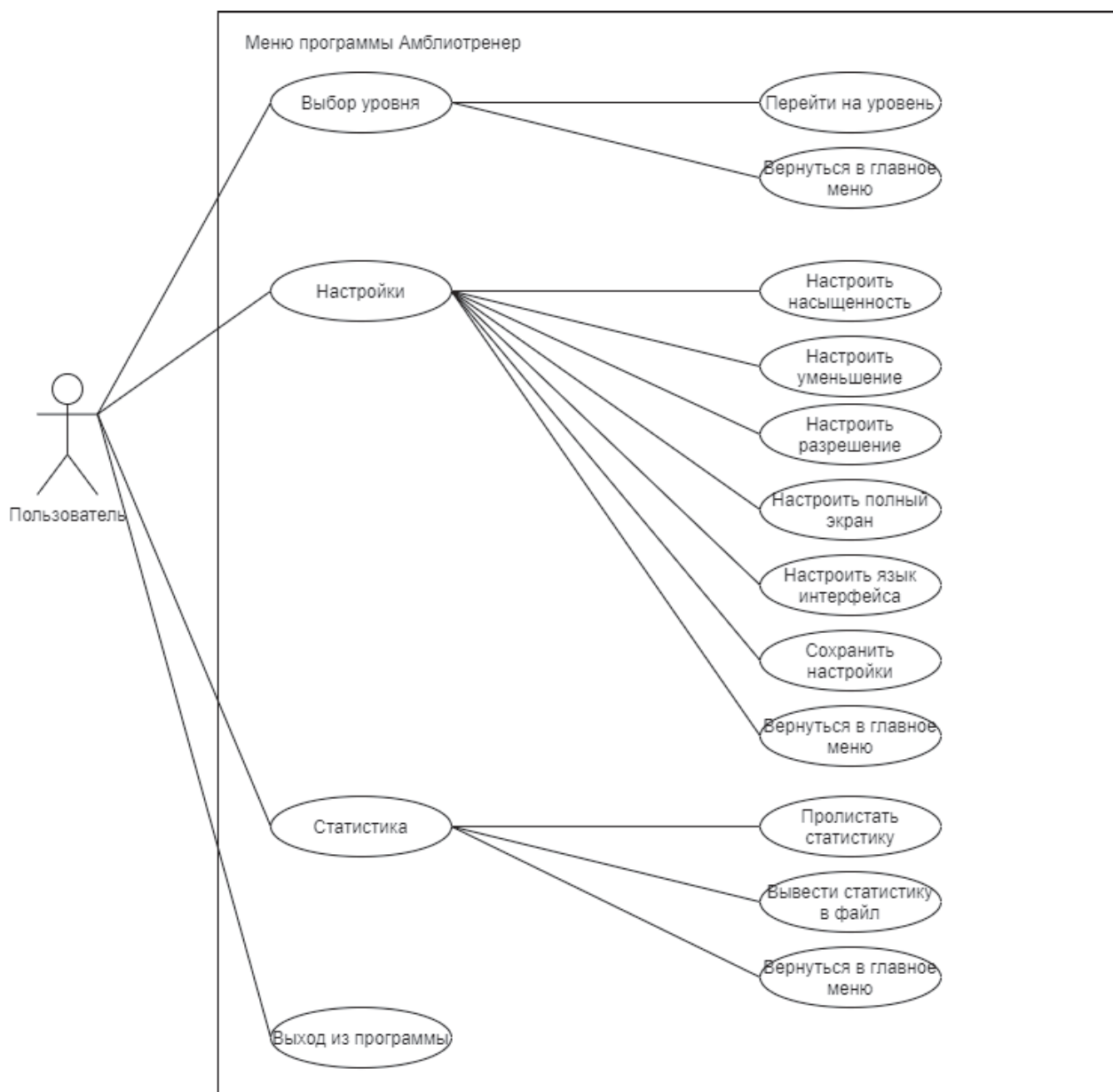


Рисунок 3.1 – Диаграмма прецедентов меню программы

На основе данных о потребностях пользователя выделены следующие варианты использования:

- выбрать уровень;
- перейти в настройки;
- перейти в меню статистики;

- выйти из программы;
- перейти на уровень;
- настроить насыщенность;
- настроить уменьшение объектов;
- настроить разрешение;
- настроить полный экран;
- настроить язык интерфейса;
- сохранить настройки;
- пролистать статистику;
- вывести статистику в отдельный файл;
- вернуться в главное меню.

#### **Прецедент «выбрать уровень»**

Позволяет пользователю выбрать уровень для тренировки. Этот вариант использования начинается, когда пользователь нажимает кнопку **«Выбор уровня»**.

#### **Прецедент «перейти в настройки»**

Позволяет пользователю перейти в меню настроек. Этот вариант использования начинается, когда нажимает кнопку **«Настройки»**.

#### **Прецедент «перейти в меню статистики»**

Позволяет пользователю перейти в меню статистики. Этот вариант использования начинается, когда пользователь нажимает кнопку **«Статистика»**.

#### **Прецедент «выход из программы»**

Позволяет пользователю выйти из программы. Этот вариант использования начинается, когда пользователь нажимает кнопку **«Выход»**.

#### **Прецедент «перейти на уровень»**

Позволяет пользователю перейти на выбранный уровень. Этот вариант использования начинается, когда пользователь нажимает кнопку с названием определенного уровня.

#### **Прецедент «настроить насыщенность»**

Позволяет пользователю настроить один из аспектов персонализации уровня сложности тренировки. Этот вариант использования начинается, когда пользователь передвигает слайдер в соответствующем разделе меню настроек.

#### **Прецедент «настроить уменьшение объектов»**

Позволяет пользователю настроить один из аспектов персонализации уровня сложности тренировки. Этот вариант использования начинается, когда пользователь передвигает слайдер в соответствующем разделе меню настроек.

#### **Прецедент «настроить разрешение»**

Позволяет пользователю настроить разрешения окна программы. Этот вариант использования начинается, когда пользователь передвигает слайдер в соответствующем разделе меню настроек.

#### **Прецедент «настроить полноэкранный режим»**

Позволяет пользователю изменять настройки полноэкранного режим для окна программы. Этот вариант использования начинается, когда пользователь нажимает на чекбокс в соответствующем разделе меню настроек.

#### **Прецедент «настройка языка интерфейса»**

Позволяет пользователю изменять язык интерфейса программы. Этот вариант использования начинается, когда пользователь нажимает на иконку с названием языка в соответствующем разделе меню настроек.

#### **Прецедент «сохранить настройки»**

Позволяет пользователю сохранить установленные им настройки. Этот вариант использования начинается, когда пользователь нажимает кнопку «Принять».



### Прецедент «вывести статистику в отдельный файл»

Позволяет пользователю сохранить статистику в отдельном файле. Этот вариант использования начинается, когда пользователь нажимает кнопку «Результаты».

### Прецедент «выйти в главное меню»

Позволяет пользователю вернуться в главное меню. Этот вариант использования начинается, когда пользователь нажимает кнопку «Назад».

Далее рассмотрим модель вариантов использования, когда пользователь находится непосредственно на тренировочном уровне (рис. 3.2). Пользователь, запустив уровень, может поставить его на паузу нажатием кнопки «escape», вернуться в главное меню и продолжить прохождение уровня.

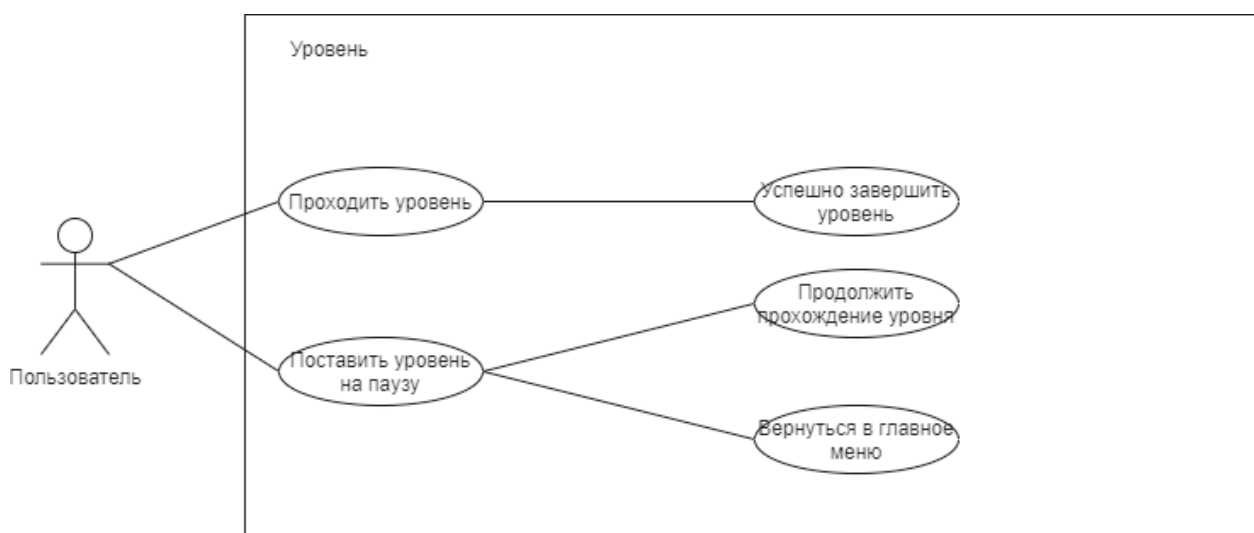


Рисунок 3.2 – Диаграмма прецедентов тренировочного уровня

На основе данных о потребностях пользователя выделены следующие варианты использования:

- проходить уровень;
- поставить уровень на паузу;
- успешное завершение уровня;

- продолжить прохождение уровня;
- вернуться в главное меню.

#### **Прецедент «проходить уровень»**

Позволяет пользователю пройти тренировочный уровень. Этот вариант использования начинается, когда пользователь двигается по заданному указателями направлению.

#### **Прецедент «поставить уровень на паузу»**

Позволяет пользователю поставить уровень на паузу. Этот вариант использования начинается, когда пользователь нажимает на клавиатуре кнопку «escape».

#### **Прецедент «успешное завершение уровня»**

Позволяет пользователю завершить прохождение уровня дойдя до выхода. Этот вариант использования начинается, когда пользователь проходит уровень до конца.

#### **Прецедент «успешное завершение уровня»**

Позволяет пользователю завершить прохождение уровня дойдя до выхода. Этот вариант использования начинается, когда пользователь проходит уровень до конца.

#### **Прецедент «продолжить прохождение уровня»**

Позволяет пользователю продолжить прохождение уровня после выхода из меню паузы. Этот вариант использования начинается, когда пользователь, находясь в меню паузы, нажимает на клавиатуре кнопку «escape» или в окне программы кнопку «Продолжить».

#### **Прецедент «выйти в главное меню»**

Позволяет пользователю вернуться в главное меню. Этот вариант использования начинается, когда пользователь нажимает кнопку «Меню».

## 3.2. РЕАЛИЗАЦИЯ

Для разработки приложения выбрана платформа Unity, имеющая низкий порог вхождения, большое количество обучающих материалов и активное сообщество разработчиков. Графическая составляющая создавалась с помощью редакторов Adobe Photoshop и Paint.NET. Программная часть разрабатывалась на языке программирования C#.

## 3.3. ФАЙЛОВАЯ СТРУКТУРА

Разработанный проект содержит каталоги, в которых хранятся:

- шрифты;
- шаблоны объектов;
- сцены;
- скрипты;
- изображения.

Файловая структура представлена на рисунке 3.3.

В директории Prefabs находятся готовые шаблоны объектов, при помощи которых можно составить уровень после его проектирования. Директория Scens содержит сцены уровней и главного меню, в которых происходят все действия. В папке Scripts находятся скрипты с описанием всех классов и взаимодействий. В каталоге Scens содержатся шесть подкаталогов:

- Main\_Menu содержит непосредственно сцену с главным меню;
- Random\_Maze\_Folder содержит сцену и все графические элементы уровня «Случайный лабиринт»;
- Cub\_Folder содержит сцену и все графические элементы уровня «Кубач»;

- Cat\_House\_Folder содержит сцену и все графические элементы уровня «Кошкин дом»;
- Apple\_Forest\_Folder содержит сцену и все графические элементы уровня «Яблочный лес»;
- NewYear\_Folder содержит сцену и все графические элементы уровня «Охота за подарками».

В каталоге UI содержатся подкаталог Menu, содержащий два подкаталога каталога с элементами интерфейса для двух языков.

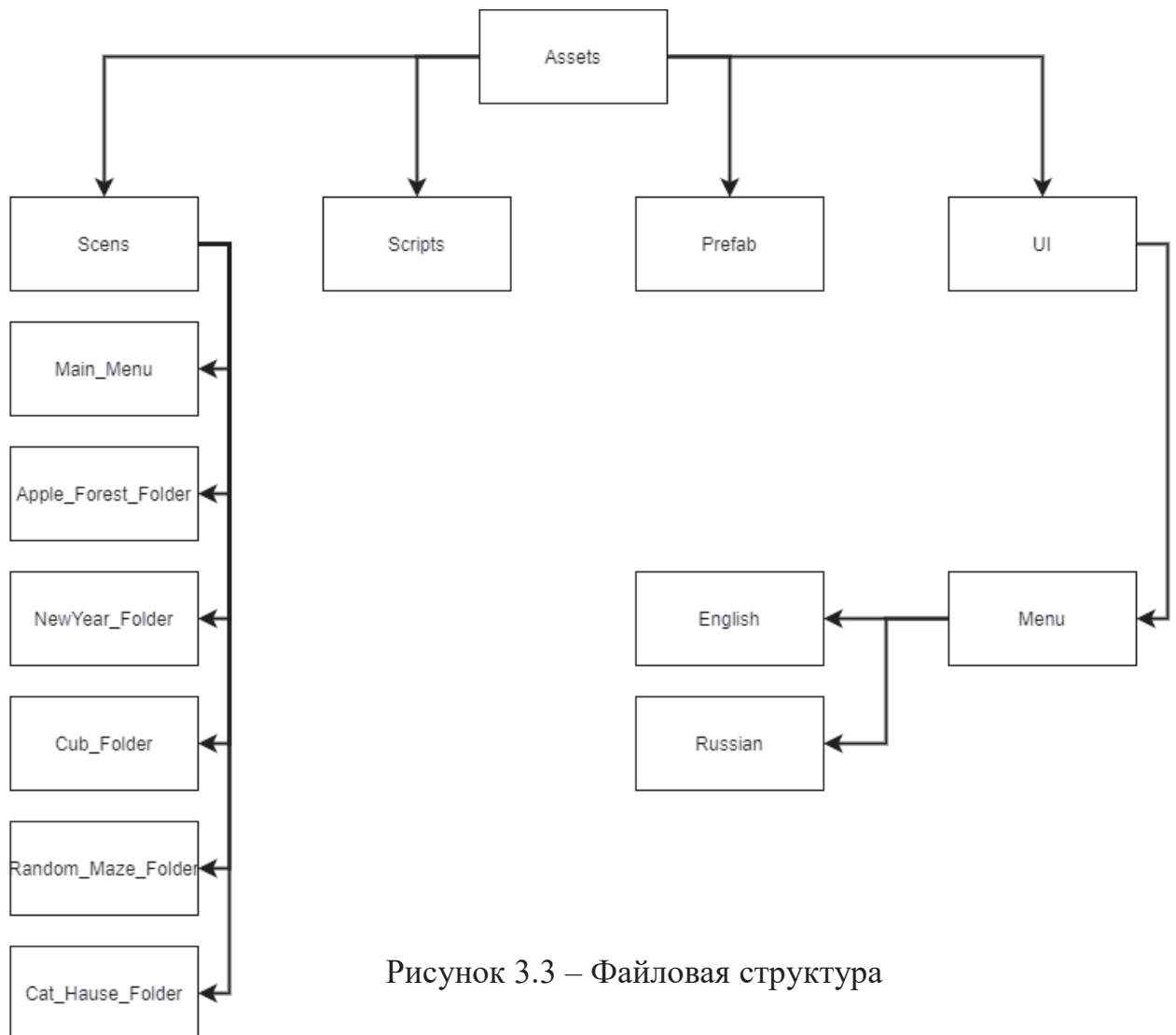


Рисунок 3.3 – Файловая структура

### 3.3. ОБЪЕКТЫ

Изображения, используемые для объектов, были скачаны в бесплатных наборах изображений с сайта Kennel [11]. Редактировались в графических редакторах Adobe Photoshop и Paint.Net.

#### Персонаж

Персонаж был нарисован в графическом редакторе Paint.Net. Варианты внешнего вида персонажа представлены на рисунке 3.4.



Рисунок 3.4 – Изображение вариантов вида главного героя

Для игрока заданы следующие характеристики:

- изображение;
- возможность перемещения;
- параметры скорости;
- коллизии с другими объектами.

#### Тренировочные уровни

Для оформления тренировочных уровней программы, создания визуального отличия и поддержания интереса пользователя был разработан и реализован тематический дизайн каждого уровня.

На рисунке 3.5 приведен дизайн одного из уровней.

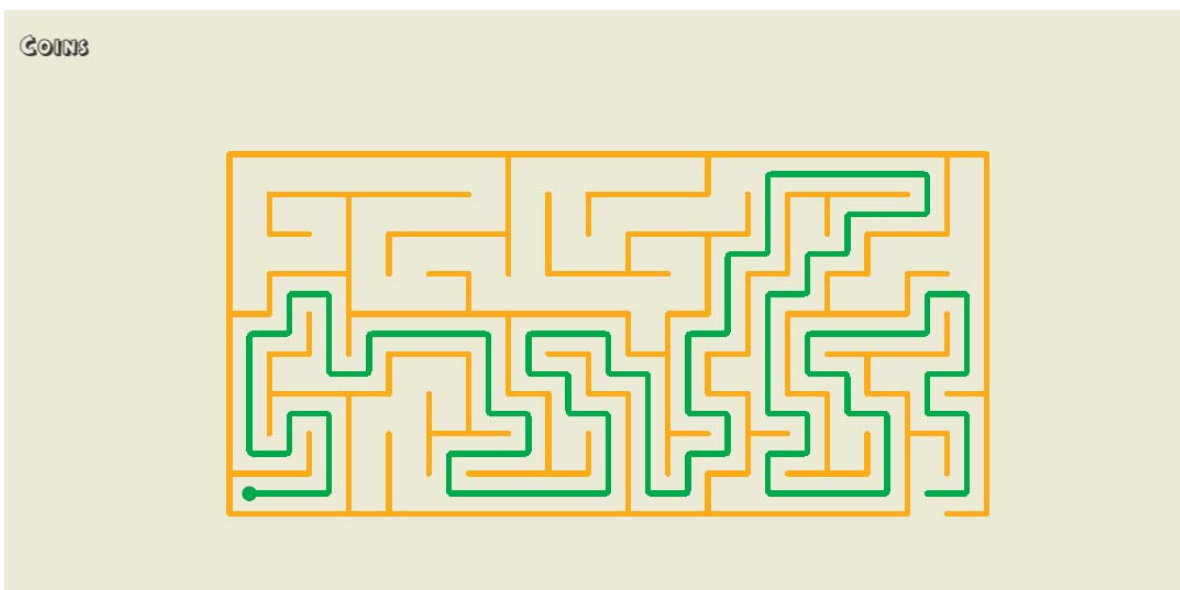


Рисунок 3.5 – Изображение уровня

### Указатели

На каждом уровне присутствует некоторое количество указателей, они же должны быть собраны для набора баллов. На рисунках 3.6 и 3.7 изображены указатели на уровнях «Яблочный лес» и «Охота за подарками».



Рисунок 3.6 – Указатель на уровне «Яблочный лес»



Рисунок 3.7 – Указатель на уровне «Охота за подарками»

## Финиш

На каждом уровне присутствует финиш, финальная точка до которой необходимо добраться, чтобы завершить уровень и получить баллы. На рисунках 3.8 и 3.9 изображены финишные точки на уровнях «Яблочный лес» и «Охота за подарками».



Рисунок 3.8 – Финишная точка на уровне «Яблочный лес»



Рисунок 3.9 – Финишная точка на уровне «Охота за подарками»

### 3.4. ИСПОЛЬЗУЕМЫЕ КЛАССЫ

Для реализации приложения были использованы скрипты, написанные на языке программирования C#.

В данной программе используется 24 скрипта: AllCoins, CameraControls, Cell, ChangeLanguage, Coin, CoinCounter, CoinView, DataSaver, Finish, Gammcfig, HintRenderer, Loader, Maze, MazeGenerator, MazeSpawner, MenuScript, Options, Pause\_Menu, PlayerComtrolls, Preview, PreviewImage, Scalecfig, LevelSelector, Statistics. Назначение каждого из них приведено в таблице 3.1.

Таблица 3.1 – Назначение скриптов

Название скрипта	Назначение скрипта	Листинг приложения А
AllCoins	Подсчет суммарного количества баллов.	Листинг А.1
CameraControls	Перемещение камеры за игроком.	Листинг А.2
Cell	Шаблон стены в процедурно генерируемом лабиринте.	Листинг А.3
ChangeLanguage	Перевод программы на английский язык.	Листинг А.4
Coin	Начисление баллов при соприкосновении игрока с объектом.	Листинг А.5
CoinCounter	Создание собираемых объектов в процедурногенерируемом лабиринте.	Листинг А.6
CoinView	Отображение набранных на уровне баллов.	Листинг А.7
DataSaver	Сохранение и загрузка данных в XML-файл.	Листинг А.8
Finish	Завершение уровня по его прохождению и начисление баллов.	Листинг А.9
Gammcfig	Конфигурация насыщенности изображения на тренировочном уровне.	Листинг А.10
HintRenderer	Поиск пути в процедурногенерируемом лабиринте.	Листинг А.11
Loader	Хранение данных.	Листинг А.12



Продолжение таблицы 3.1

Название скрипта	Назначение скрипта	Листинг приложения А
Maze	Один из составных скриптов, отвечающий за генерацию процедурногенерируемого лабиринта.	Листинг А.13
MazeGenerator	Один из составных скриптов, отвечающий за генерацию процедурногенерируемого лабиринта.	Листинг А.14
MazeSpawner	Один из составных скриптов, отвечающий за генерацию процедурногенерируемого лабиринта.	Листинг А.15
MenuScript	Отрисовка главного меню программы.	Листинг А.16
Options	Функционал меню настроек.	Листинг А.17
PauseMenu	Меню паузы.	Листинг А.18
PlayerCotrols	Управление персонажем.	Листинг А.19
Preview	Предпросмотр уровней.	Листинг А.20
PreviewImage	Скрипт, являющийся составным скрипту Preview	Листинг А.21
Scalecfg	Конфигурация масштаба на тренировочных уровнях.	Листинг А.22
LevelSelector	Перемещение между уровнями.	Листинг А.23
Statistics	Сохранение, загрузка и отображение статистики.	Листинг А.24

На рисунке 3.10 представлена диаграмма классов, ответственных за взаимодействие с XML-файлом.

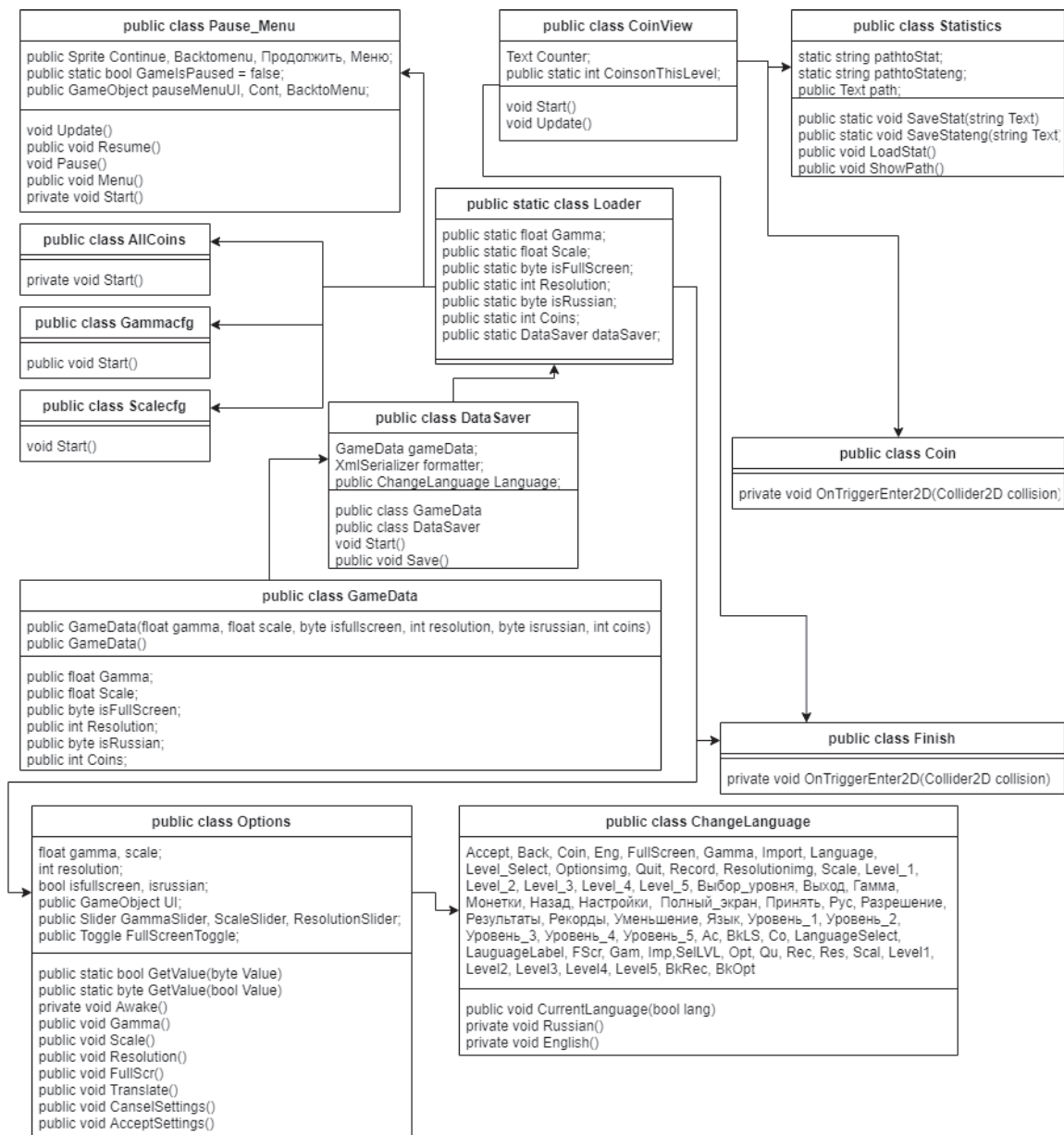


Рисунок 3.10 – Диаграмма классов, ответственных за взаимодействие с XML-файлом

На рисунке 3.11 представлена диаграмма классов, отвечающих за предыдущую генерацию одного из тренировочных уровней.

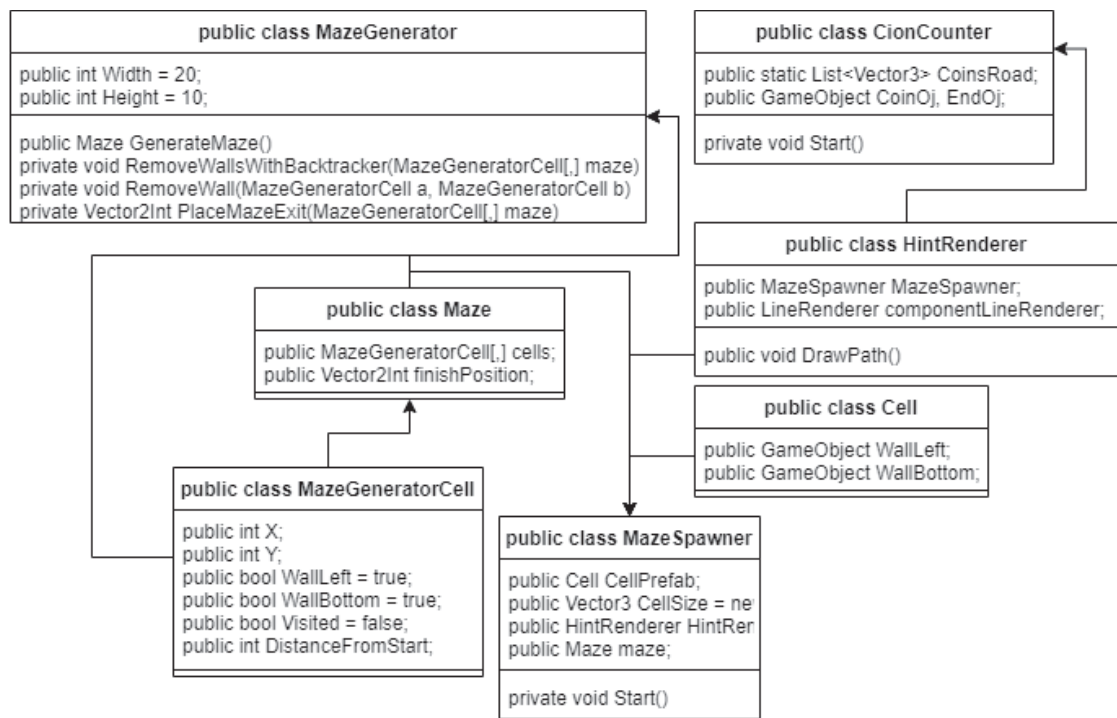


Рисунок 3.11 – Диаграмма классов, отвечающих за прцедурную генерацию уровня

На рисунке 3.12 представленна диаграмма классов, отвечающих за предпоказ уровней в меню выбора уровней.

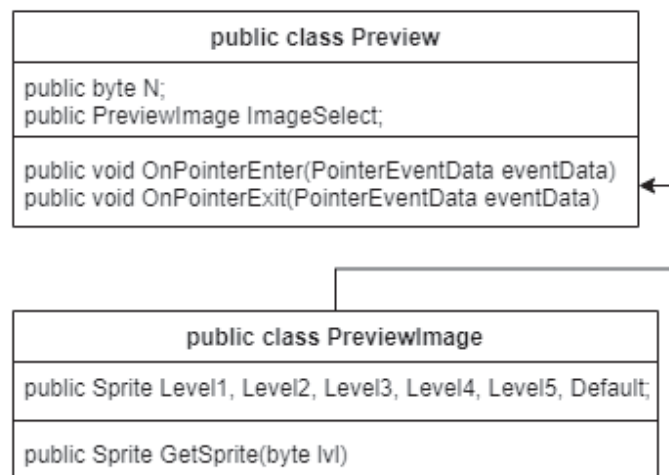


Рисунок 3.12 – Диаграмма классов, отвечающих за предпоказ уровня

На рисунке 3.13 представлена диаграмма служебных классов, отвечающих за исполнение функций внутри меню программы.

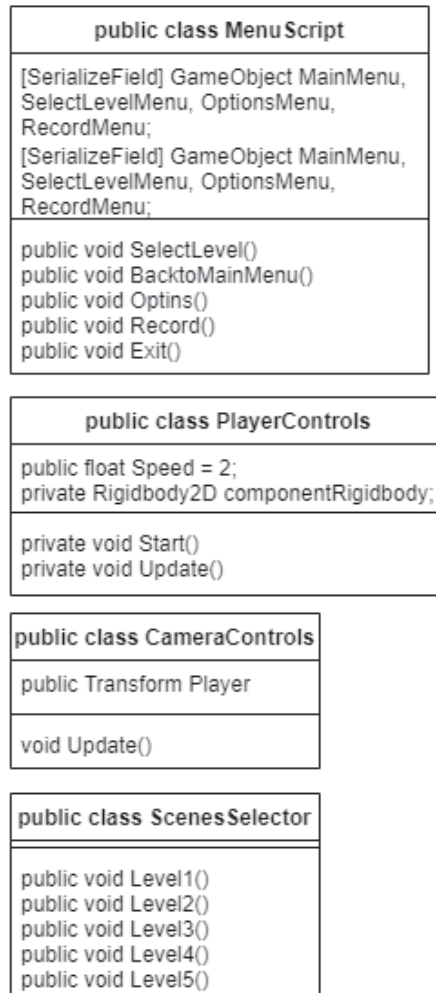


Рисунок 3.13 – Диаграмма служебных классов

Текст скриптов приведен в приложении А.

### 3.5. ПРОЦЕСС ТРЕНИРОВКИ

Проект состоит из сцен – файлов, содержащих всю визуальную и программную составляющую приложения, со всеми объектами, сценарием и настройками.

Объекты, в свою очередь, содержат наборы компонентов, с которыми взаимодействуют скрипты. У любого объекта на сцене обязательно присутствует компонент Transform, хранящий координаты местоположения, поворота и размеров объекта по всем трём осям. У объектов с видимой геометрией также по умолчанию присутствует компонент Sprite Renderer, делающий модель объекта видимой. На данный момент приложение состоит из главного меню и 5 тренировочных уровней. При запуске приложения пользователь видит экран главного меню, в котором доступны 4 кнопки: выбор уровня, статистика, настройки и выход. Реализация показана на рисунке 3.14.

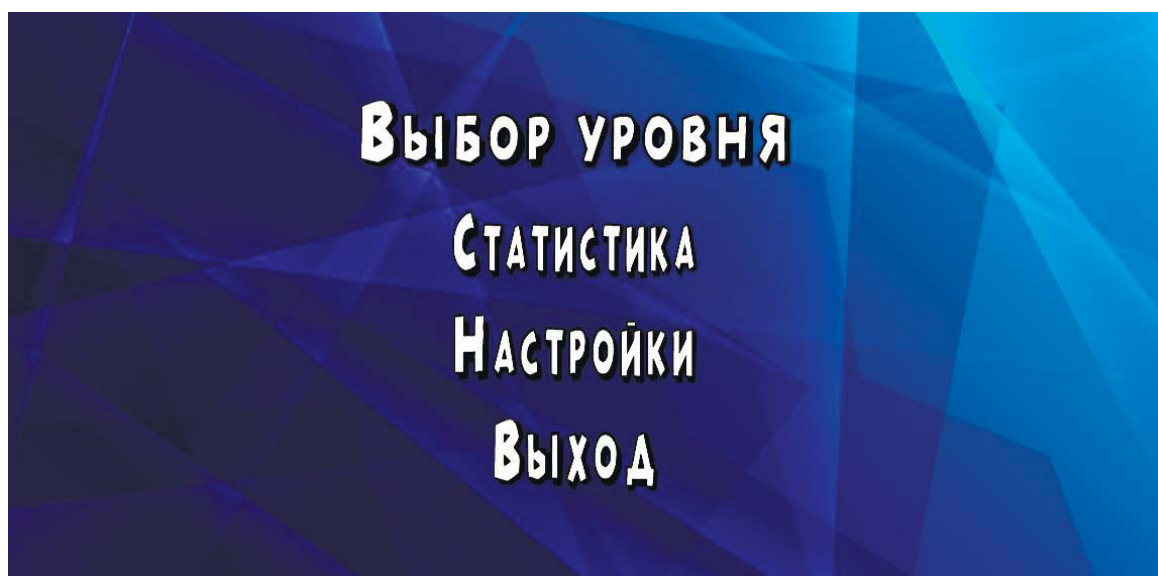


Рисунок 3.14 – Главное меню

При заходе в меню выбора уровня отображается список уровней с превью уровня. Выбор уровня отображен на рисунке 3.15.

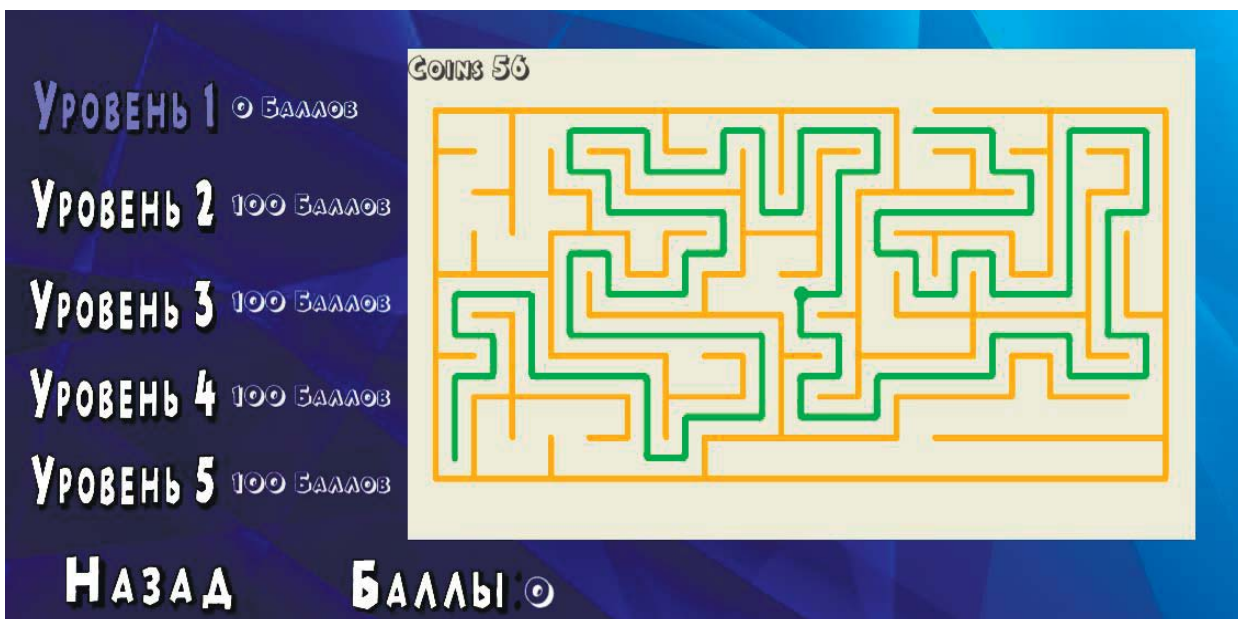


Рисунок 3.15 – Меню выбора уровня

Далее пользователь выбирает первый доступный уровень и после его прохождения получает баллы для разблокирования нового уровня. Один из уровней показан на рисунке 3.16.



Рисунок 3.16 – Уровень «Кошкин дом»

После прохождения уровня результаты пользователя записываются в раздел статистики.

### **3.6. ВЫВОДЫ**

В ходе проектирования программной системы были определены основные требования и разработаны UML-диаграммы вариантов использования в двух сценах: главное меню и непосредственно уровень для тренировки. Они определяют основные действия, которые необходимо реализовать в разрабатываемой программе.

Также в данном разделе были показаны некоторые принципы разработки предлагаемого программного продукта, приведены таблица скриптов и диаграммы классов.

В результате реализации проекта создана программа, имеющая возможности управления персонажем, настройки уровней сложности, переключения между русским и английским языками, управление через меню и соответствующие пользовательские интерфейсы. Разработан дизайн уровней и персонажей. Также реализовано начисление баллов, вывод статистики в специализированное программное подменю, вывод результатов тренировок в файл. Полный цикл работ проводился при помощи выбранных инструментов. Для проверки корректности работы программы, необходимо провести тестирование.



## 4. ТЕСТИРОВАНИЕ

### 4.1. ФУНКЦИОНАЛЬНОЕ ТЕСТИРОВАНИЕ

Для тестирования приложения использовался метод функционального тестирования.

Функциональное тестирование – это тестирование программного обеспечения в целях проверки реализуемости функциональных требований, то есть способности программного обеспечения в определенных условиях решать задачи, нужные пользователям [10].

Результаты тестирования:

#### Тест № 1. Переход в меню выбора уровня.

Входные данные: пользователь находится в главном меню.

Ожидаемый результат: при нажатии кнопки «Выбор уровня» пользователь переходит в меню выбора уровня.

Полученный результат: из рисунка 4.1 видно, что полученный результат совпадает с ожиданиями.



Рисунок 4.1 – Результаты теста №1



### Тест № 2. Переход в меню настроек.

Входные данные: пользователь находится в главном меню.

Ожидаемый результат: при нажатии кнопки «**Настройки**» пользователь переходит в меню настроек.

Полученный результат: из рисунка 4.2 видно, что полученный результат совпадает с ожиданиями.

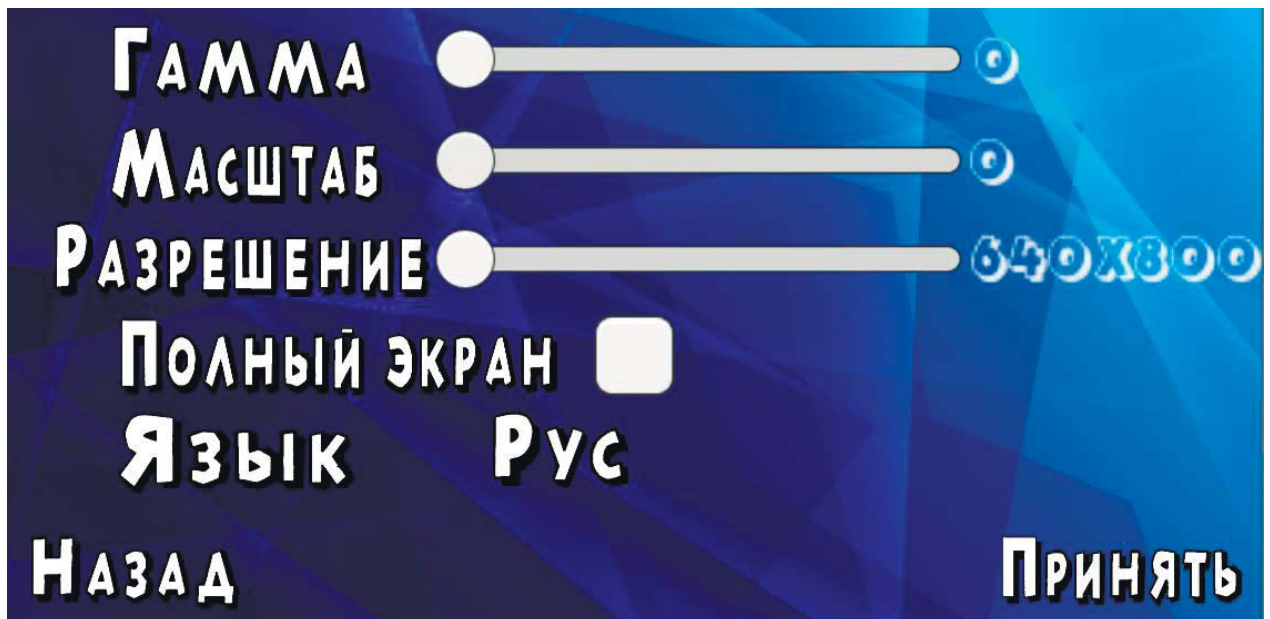


Рисунок 4.2 – Результаты теста №2

### Тест № 3. Переход в меню статистики.

Входные данные: пользователь находится в главном меню.

Ожидаемый результат: при нажатии кнопки «**Статистика**» пользователь переходит в меню статистики.

Полученный результат: из рисунка 4.3 видно, что полученный результат совпадает с ожиданиями.



Рисунок 4.3 – Результаты теста №3

#### Тест № 4. Переход на уровень 1.

Входные данные: пользователь находится в меню выбора уровня.

Ожидаемый результат: при нажатии кнопки «Уровень 1» пользователь переходит на уровень 1.

Полученный результат: из рисунка 4.4 видно, что полученный результат совпадает с ожиданиями.

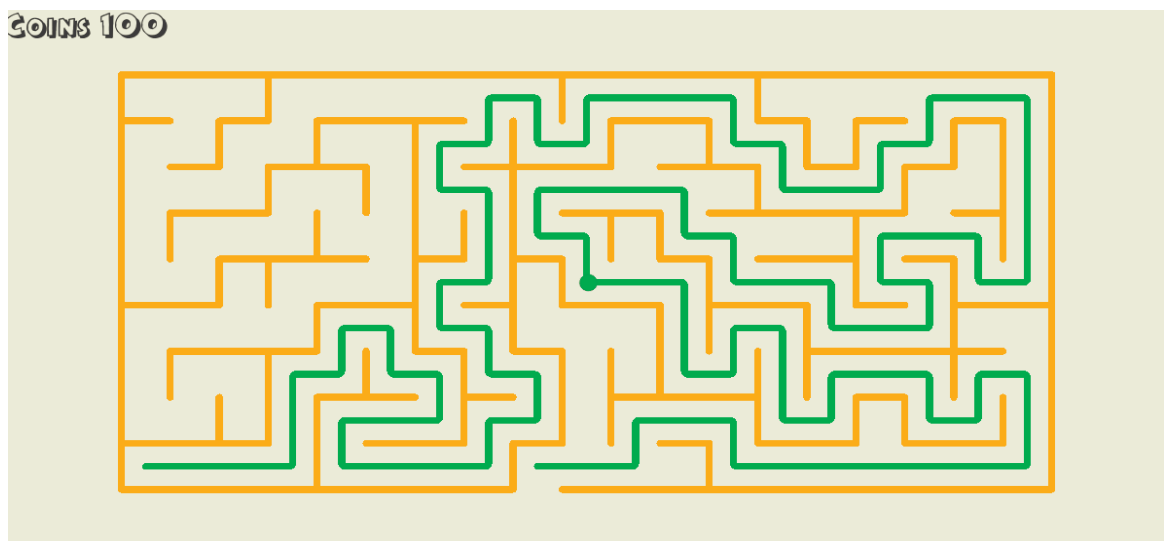


Рисунок 4.4 – Результаты теста №4

### **Тест № 5. Изменение настроек насыщенности.**

Входные данные: пользователь находится в меню настроек.

Ожидаемый результат: при изменении показаний слайдера «Гамма», насыщенность на всех уровнях изменяется в соответствии с настройками.

Полученный результат: из рисунка 4.5 видно, что полученный результат совпадает с ожиданиями.

Coins 97

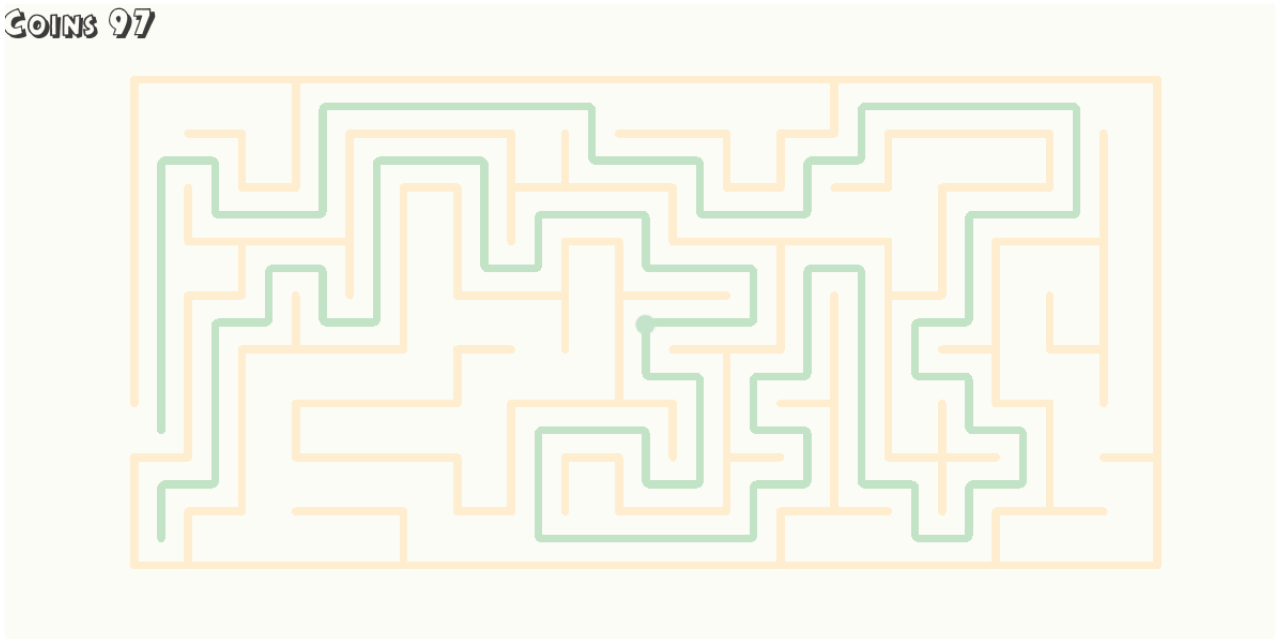


Рисунок 4.5 – Результаты теста №5

### **Тест № 6. Изменение настроек масштаба камеры.**

Входные данные: пользователь находится в меню настроек.

Ожидаемый результат: при изменении показаний слайдера «Масштаб», масштаб камеры на всех уровнях изменяется в соответствии с настройками.

Полученный результат: из рисунка 4.6 видно, что полученный результат совпадает с ожиданиями.

COINS ©

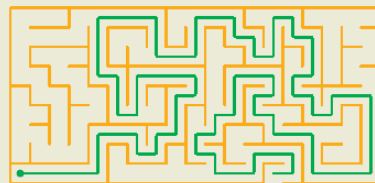


Рисунок 4.6 – Результаты теста №6

**Тест № 7. Начисление баллов.**

Входные данные: пользователь проходит тренировочный уровень с определенным результатом.

Ожидаемый результат: при переходе в меню выбора уровня счетчик баллов меняет своё значение.

Полученный результат: из рисунка 4.7 видно, что полученный результат совпадает с ожиданиями.



Рисунок 4.7 – Результаты теста №7

#### Тест № 8. Вывод статистики в файл.

Входные данные: пользователь находится в меню статистики с переключенным на русский язык интерфейса.

Ожидаемый результат: при нажатии на кнопку «Вывод в файл» вся статистика в виде файла выводится на рабочий стол.

Полученный результат: из рисунка 4.8 видно, что полученный результат совпадает с ожиданиями.

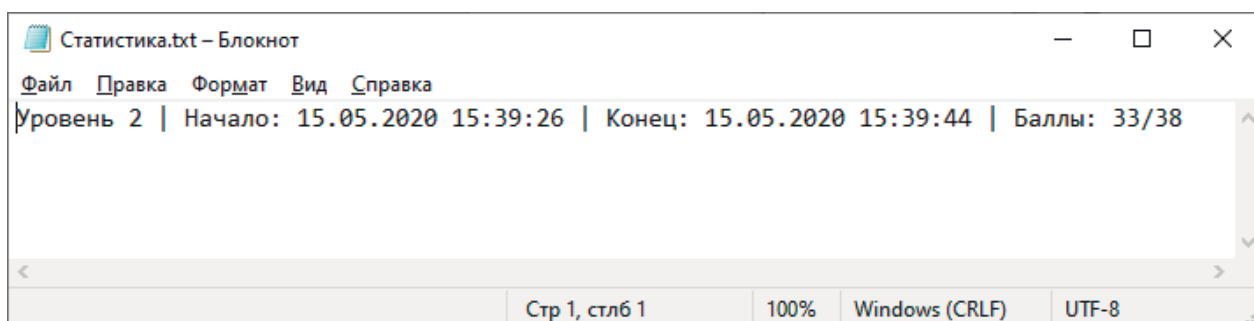


Рисунок 4.8 – Результаты теста №8

**Тест № 9. Изменение языка.**

Входные данные: пользователь находится в меню настройки.

Ожидаемый результат: при нажатии кнопки «Рус» язык интерфейса программы изменится на английский.

Полученный результат: из рисунка 4.9 видно, что полученный результат совпадает с ожиданиями.

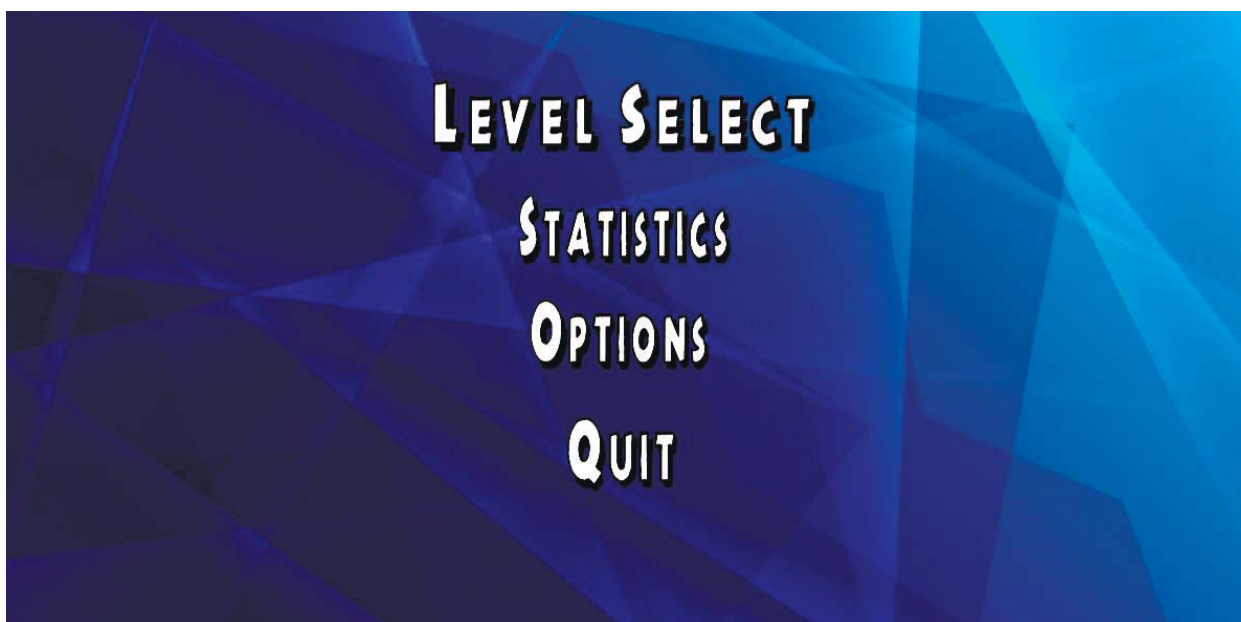


Рисунок 4.9 – Результаты теста №9

### Тест № 10. Вывод статистики в файл на английском языке.

Входные данные: пользователь находится в меню статистики с переключенным на английский язык интерфейса.

Ожидаемый результат: при нажатии кнопки «**Output to file**», вся статистика на английском языке, в виде файла, выводится на рабочий стол.

Полученный результат: из рисунка 4.10 видно, что полученный результат совпадает с ожиданиями.

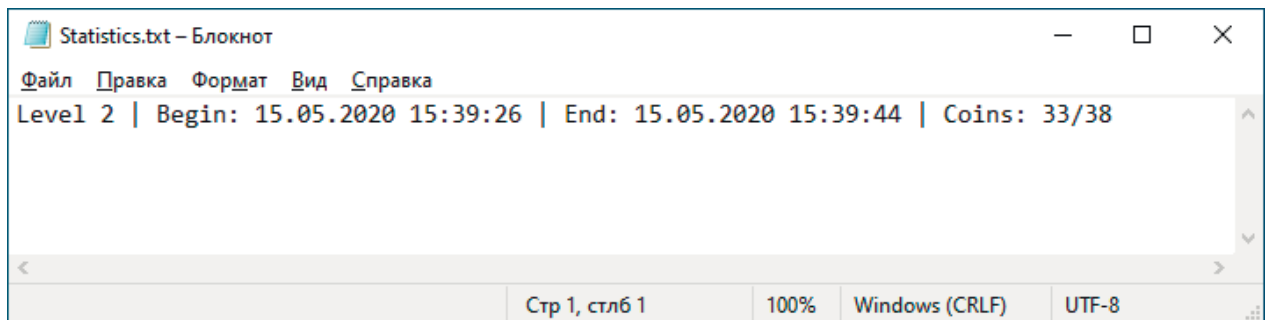


Рисунок 4.10 – Результаты теста №10

## 5. ЗАКЛЮЧЕНИЕ

В рамках данной работы была спроектирована и реализована программа для лечения амблиопии у детей методом тренировки зрительного аппарата. Тренировка заключается в прохождении уровней с меняющимися показателями контрастности и масштаба.

Для достижения поставленной цели были выполнены следующие задачи:

- проведен анализ аналогичных проектов;
- сформулированы функциональные и нефункциональные требования к программному продукту;
- проведен анализ и выбор средств реализации программы;
- описана концепция программы, выполнено ее проектирование и реализация;
- создана визуальная составляющая программы;
- проведено тестирование реализованной программы.

В рамках реализации программного продукта выполнен дизайн главного меню, меню выбора уровня, меню настроек, меню статистики программы, персонажа, предметов, тренировочных уровней. Написаны скрипты, отвечающие за весь функционал. Кроме этого, программа была переведена на английский язык для привлечения англоязычной аудитории.

К дальнейшим планам по развитию программного продукта относятся:

- добавление новых тематических уровней;
- добавление новых персонажей;
- расширение функционала;
- регистрация в Едином реестре российских программ для электронных вычислительных машин и баз данных;
- прохождение клинических испытаний.



## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Официальный сайт Unity [Электронный ресурс] URL: <https://unity.com/ru> (дата обращения: 28.02.2020).
2. Официальный сайт компании Epic Games и Unreal Engine. [Электронный ресурс] URL: <https://www.unrealengine.com/en-US/> (дата обращения: 28.02.2020).
3. Официальный сайт компании Crytek и CryEngine. [Электронный ресурс] URL: <https://www.crytek.com/cryengine> (дата обращения: 28.02.2020).
4. Сайт Всемирной организации здравоохранения/ Нарушения зрения и слепота. [Электронный ресурс] URL: <https://www.who.int/ru/news-room/fact-sheets/detail/blindness-and-visual-impairment> (дата обращения: 27.02.2020).
5. Сайт медицинского центра Мой Доктор [Электронный ресурс] URL: <https://moy-doktor.org> (дата обращения: 28.02.2020)
6. Сайт MED39.RU [Электронный ресурс] URL: <https://med39.ru> (дата обращения: 28.02.2020)
7. Сайт Комплекс Окулист [Электронный ресурс] URL: <http://astro.ru> (дата обращения: 28.02.2020)
8. Сайт компании Точка зрения [Электронный ресурс] URL: <https://tochka-zrenia.ru> (дата обращения: 28.02.2020)
9. Арлоу, Д. UML 2 и Унифицированный процесс / Д. Арлоу, А. Нейштадт – 2-е изд. – СПб: Символ Плюс, 2007. – 624 с.
10. Майерс, Искусство тестирования программ / Г. Майерс, Т. Баджет – М.: Изд-во Вильямс, 2012. – 272 с.
11. Сайт Kenney [Электронный ресурс] URL: <https://kenney.nl/> (дата обращения: 28.02.2020)