Содержание

[Введение 2](#_Toc71829453)

[Технико-экономическая характеристика 3](#_Toc71829454)

[1. Общие положения 3](#_Toc71829455)

[Аппаратное обеспечение 5](#_Toc71829456)

[Программное обеспечение отдела ПО 6](#_Toc71829457)

[Методы проектирования и разработки 8](#_Toc71829458)

[Метод Waterfall 8](#_Toc71829459)

[Метод MVC. 9](#_Toc71829460)

[Математическая постановка задачи 11](#_Toc71829461)

[Программные решения 14](#_Toc71829462)

[1. Petralex Hearing Aid App 14](#_Toc71829463)

[2. Hear Boost. Enhanced Microphone Volume & Recording. 16](#_Toc71829464)

[Технологии обработки данных 18](#_Toc71829465)

[Алгоритм решения 20](#_Toc71829466)

[Выбор инструментов 21](#_Toc71829467)

[Тестовые наборы к программе 23](#_Toc71829468)

[Описание главного модуля программы 24](#_Toc71829469)

[Руководство оператора 25](#_Toc71829470)

[Техника безопасности 30](#_Toc71829471)

[Источники 31](#_Toc71829472)

[Дневник практики 33](#_Toc71829473)

# Введение

На 4 курсе обучения в колледже космического машиностроения и технологий, мной была пройдена преддипломная практика. Было получено задание на разработку кросс платформенного мобильного приложения для использования в учебных целях для студентов ККМТ.

# Технико-экономическая характеристика

## Общие положения

**Характеристика мастерской**

1. Полное название учебно-научной мастерской: «Программные решения для бизнеса»

2. Базовое подразделение: ККМТ

3. Год создания мастерской: 2019

4. Месторасположение мастерской: Московская область, г. Королёв, ул. Пионерская, д.8

5. Заведующий мастерской - Попов Вячеслав Николаевич

6. Специализация мастерской: проведение занятий по закреплённым за лабораторией дисциплинам.

7. Кадровый потенциал лаборатории:

**Таблица 1**

|  |  |  |  |
| --- | --- | --- | --- |
| № | Ф.И.О. | Должность | Образование, ученая степень, ученое звание |
| 1. | Попов Вячеслав Николаевич | Заведующий мастерской | высшее |

8. Деятельность мастерской:

1) Образовательная. Проведение учебных дисциплин и курсов

2) Научная. Выполнение научных исследований студентами, аспирантами, докторантами, преподавателями и сотрудниками

9. Учебные дисциплины и учебные курсы, проводимые на базе мастерской:

**Таблица 2**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Название дисциплины (учебного курса) | Направление подготовки | Кол-во часов | Преподаватель |
| 1. | МДК 03.01 Технология разработки программного обеспечения | 09.02.03 | 392 | Попов В.Н. |
| 2. | Операционные системы | 09.02.04 | 588 | Попов В.Н. |
| 3. | Архитектура компьютерных систем | 09.02.03 | 268 | Попов В.Н. |
| 4. | Технические средства информатизации | 09.02.03 | 74 | Коптилин Р.М. |
| 5. | МДК 03.03 Документирование и сертификация | 09.02.03 | 276 | Бобкова Н.Ю. |

10. Количество студентов, прошедших обучение на базе лаборатории:

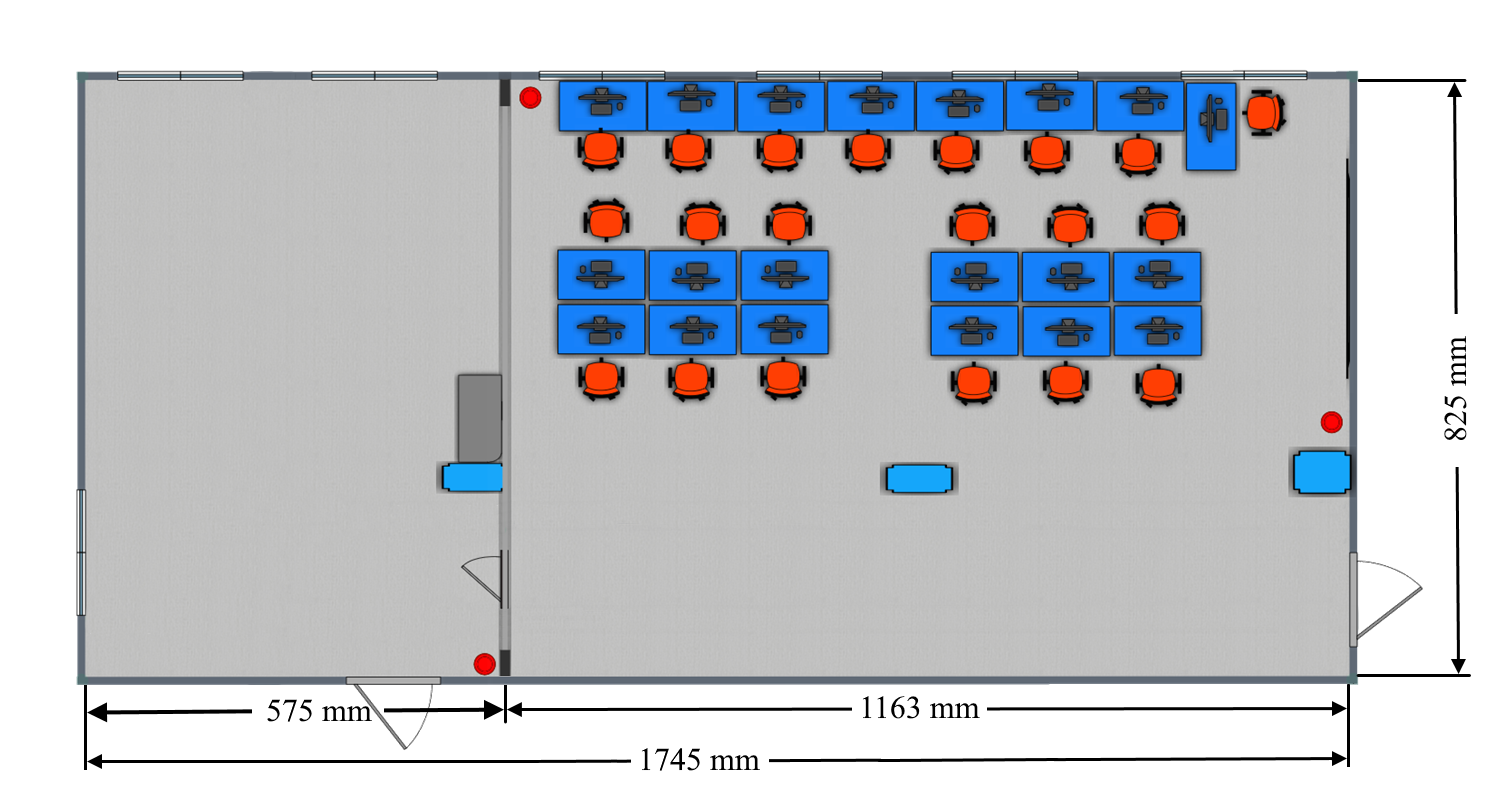
1. **Таблица 3**

|  |  |  |  |
| --- | --- | --- | --- |
| № | Направление подготовки | Учебный год | Количество студентов |
| 1. | 09.02.03 Программирование в компьютерных системах | 2019 – 2020 | 114 |
| 2. | 09.02.04 Информационные системы (по отраслям) | 2019 - 2020 | 60 |
| 3. | 09.02.03 Программирование в компьютерных системах | 2020 - 2021 | 133 |
| 4. | 09.02.04 Информационные системы (по отраслям) | 2020 - 2021 | 60 |
| 5. | 10.02.04 «Обеспечение информационной безопасности телекоммуникационных систем» (по отраслям) | 2020-2021 | 60 |

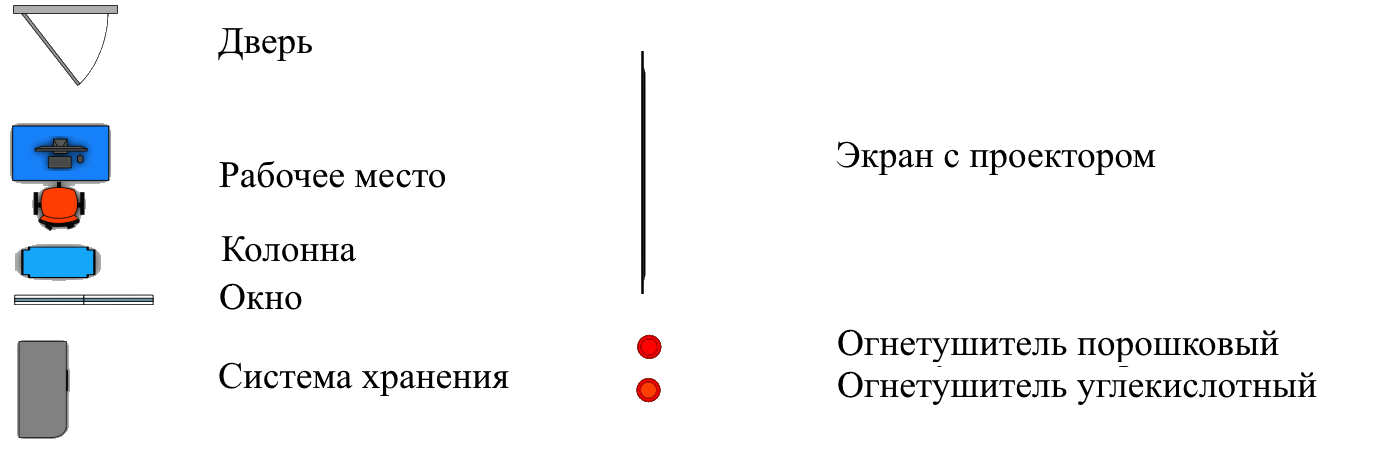
11. Количество рабочих мест: 20

12. Среднее количество пар в неделю, проводимых в данной мастерской(период март-апрель 2021): 22

**Планировка мастерской 313б**



1. Рисунок 1. Общая схема



1. Рисунок 2. Легенда

**Аппаратное обеспечение**

Аппаратное обеспечение мастерской состоит из:

1. Автоматизированных рабочих мест (20 шт.)

ЦПУ:

- Intel Core i7-9700;

- базовая тактовая частота 3.0 ГГц;

- количество физических ядер 8;

- количество потоков 8;

ОЗУ:

- 16 Гб;

ПЗУ:

- SSD объемом 500 Гб;

- HDD объемом 1000 Гб;

сетевого адаптера:

- технология Ethernet стандарта 1000BASE-T.

1. 20 мониторов ЖКД Dell p2419h 24"

2. Проектора Panasonic PT-VW360

3. Планшетного компьютера Lenovo TB-X605L (20 шт.)

4. Коммутатора catalyst 2960 si

5. Коммутатора AT-GS950/16

6. Сервера Team Server R2-E52 в комплектации:

- Процессор -  2хIntel Xeon Gold 5218 LGA 3647 22Mb 2.3Ghz

- Оперативная память - 256Gb RAM

- Жесткий диск - 6Tb SSD

Схема локальной сети мастерской соответствует топологии «звезда».

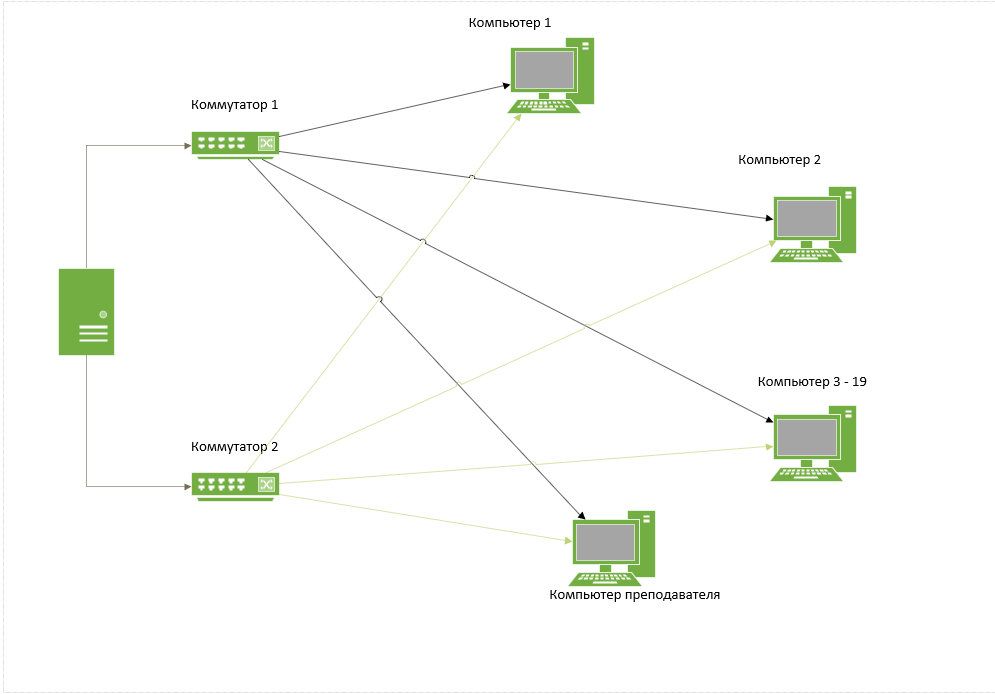


Рисунок 1. Схема локальной сети мастерской 313б

# Программное обеспечение отдела ПО

Программное обеспечение мастерской «Программные решения для бизнеса» состоит из перечня продуктов, приведенных в Таблице 1.

**Таблица 1. Программное обеспечение мастерской**

|  |  |  |  |
| --- | --- | --- | --- |
| № | Наименование | Количество лицензий | Вебсайт продукта |
| 1. 1 | MS Windows 10 Pro Edu | 20 | https://www.microsoft.com/ |
|  | 7-Zip 19.00 | 20 | https://www.7-zip.org/ |
|  | Пакет офисных приложений Microsoft Office 2019 | 20 | [Microsoft Office 2019 | Microsoft 365](https://www.microsoft.com/en-us/microsoft-365/get-started-with-office-2019) |
|  | XCode 12 | 10 | <https://developer.apple.com/xcode/> |
| 1. 4 | Adobe Acrobat Reader DC | 20 | https://www.adobe.com/ru/ |
| 1. 6 | Android Studio | 20 | https://developer.android.com/studio |
| 1. 10 | Git version 2.24.1.2 | 20 | https://git-scm.com/ |
| 1. 11 | Google Chrome | 20 | https://www.google.com/intl/ru/chrome/ |
| 1. 12 | IntelliJ IDEA Community Edition 2019.3 | 20 | https://www.jetbrains.com/ |
| 1. 13 | Java 8 Update 231 (64-bit) | 20 | https://www.java.com/ |
| 1. 14 | Java(TM) SE Development Kit 15 (64-bit) | 20 | https://www.oracle.com/ |

**Окончание таблицы 1**

|  |  |  |  |
| --- | --- | --- | --- |
| 1. 16 | Lightshot-5.5.0.4 | 20 | https://lightshot.ru.uptodown.com/ |
| 1. 17 | Microsoft .NET Framework 3.5 Targeting Pack (rus) | 20 | https://www.microsoft.com/ |
| 1. 23 | Microsoft Visual Studio Installer | 20 | https://visualstudio.microsoft.com/ru/ |
| 1. 27 | NetBeans IDE 8.2 RC | 20 | https://netbeans.apache.org/ |
| 1. 28 | Notepad++ (64-bit x64) | 20 | https://notepad-plus-plus.org/ |
| 1. 29 | OBS Studio | 20 | https://obsproject.com/ru |
| 1. 30 | Oracle VM VirtualBox 6.1.18 | 20 | https://www.virtualbox.org/ |
| 1. 32 | Ramus | 20 | http://ramussoftware.com/ |
| 1. 34 | Visual Studio Professional 2019 | 20 | https://visualstudio.microsoft.com/ru/vs/ |
|  | Android Studio 3.5 | 20 | <https://developer.android.com/studio> |

# Методы проектирования и разработки

## Метод Waterfall

При создании приложения был выбран метод разработки Waterfall (каскадная модель, или «водопад»). В этой модели разработка осуществляется поэтапно: каждая следующая стадия начинается только после того, как заканчивается предыдущая.



Рисунок 3. Схема использования Waterfall модели

Преимущества «водопада»

* Разработку просто контролировать. Заказчик всегда знает, чем сейчас заняты программисты, может управлять сроками и стоимостью.
* Стоимость проекта определяется на начальном этапе. Все шаги запланированы уже на этапе согласования договора, ПО пишется непрерывно «от и до».
* Не нужно нанимать тестировщиков с серьёзной технической подготовкой. Тестировщики смогут опираться на подробную техническую документацию.

Недостатки каскадной модели

* Тестирование начинается на последних этапах разработки. Если в требованиях к продукту была допущена ошибка, то исправить её будет стоить дорого. Тестировщики обнаружат её, когда разработчик уже написал код, а технические писатели — документацию.
* Заказчик видит готовый продукт в конце разработки и только тогда может дать обратную связь. Велика вероятность, что результат его не устроит.
* Разработчики пишут много технической документации, что задерживает работы. Чем обширнее документация у проекта, тем больше изменений нужно вносить и дольше их согласовывать.

## Метод MVC.

В роли метода проектирования был выбран паттерн MVC (Model View Controller). Основная идея этого метода проектирования в том, что и контроллер, и представление зависят от модели, но модель никак не зависит от этих двух компонент.

Метод проектирования MVC состоит из трёх частей:

1. Model (Модель) – часть, содержащая в себе функциональную бизнес-логику приложения. Модель должна быть полностью независима от остальных частей продукта. Модельный слой ничего не должен знать об элементах дизайна, и каким образом он будет отображаться. Достигается результат, позволяющий менять представление данных, то как они отображаются, не трогая саму Модель.

Модель обладает следующими признаками:

* Модель — это бизнес-логика приложения;
* Модель обладает знаниями о себе самой и не знает о контроллерах и представлениях;
* Для некоторых проектов модель — это просто слой данных (DAO, база данных, XML-файл);
* Для других проектов модель — это менеджер базы данных, набор объектов или просто логика приложения;

1. View (Представление) отвечает за отображение данных полученных от Модели. Однако, представление не может напрямую влиять на модель. Можно говорить, что представление обладает доступом «только на чтение» к данным.

Представление обладает следующими признаками:

* В представлении реализуется отображение данных, которые получаются от модели любым способом;
* В некоторых случаях, представление может иметь код, который реализует некоторую бизнес-логику.

1. Controller (Контроллер) определяет какое представление должно быть отображено в данный момент. События и представления могут повлиять только на контроллер, а контроллер может, влияя на модель, определить другое представление. Для одного контроллера возможно несколько представлений.

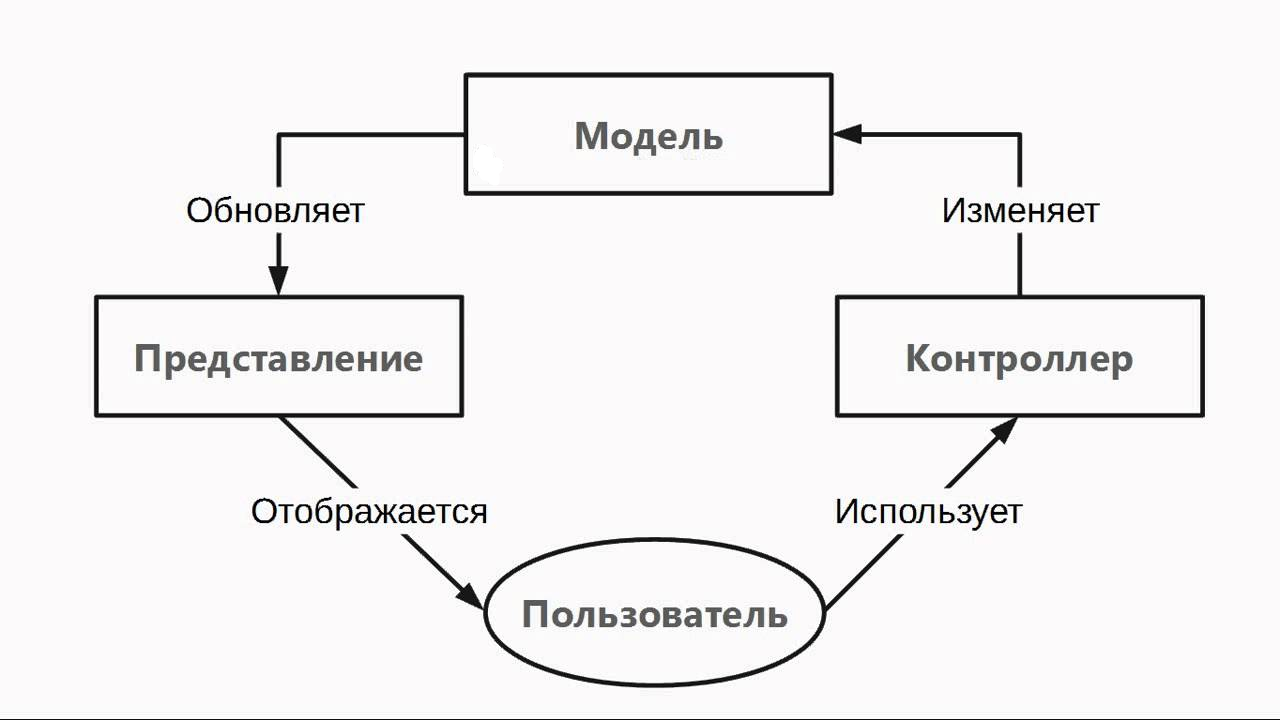


Рисунок 4. Пример использования метода проектирования MVC

При реализации данного метода, контроллер перехватывает событие извне и в соответствии с заложенной в него логикой, реагирует на это событие изменяя Модель, посредством вызова соответствующего метода. После изменения Модель использует событие о том, что она изменилась, и все подписанные на это события Представления, получив его, обращаются к Модели за обновленными данными, после чего их и отображают.

Источник - [Паттерны для новичков: MVC vs MVP vs MVVM / Хабр (habr.com)](https://habr.com/ru/post/215605/)

список будет выведен в интерфейс программы в удобном для восприятия виде.

# Математическая постановка задачи

**Формула передвижения игрового персонажа:**

**d = V \* S \* dltime \* r**

,где

d – расстояние пройденное игровым объектом,

V – вектор направления,

S – скорость объекта,

dltime– время между текущим в предыдущим кадром.

r – специальный коэффициент учета укореняя

**Алгоритм определения объекта взаимодействия**

В основе определения объекта для взаимодействия лежит система лучей и система игровых тэгов.

1. Первым шагом в алгоритме лежит запуск луча из центра камеры по формуле:

**s = start - h\*q**

где

**s** – длина луча;

**start** – специальный стартовый коэффициент по умолчанию;

**h** – коэффициент разрешения камеры;

**q** – коэффициент качества изображения.

1. определение первого объекта, перекрывший своим коллайдером данный луч и находящийся в доступном массиве слоев.
2. Определение игровой тэг данного объекта. Если тэг находится в области игрового взаимодействия, мы проецируем изменения интерфейса на UI и переходим в режим ожидания дальнейших действий.

**Алгоритм нахождения игрового персонажа не игровым**

В основе данного алгоритма лежит система лучей:

1. определяем на сцене игрового персонажа
2. запускаем луч из центра нашей позиции в центр позиции игрока
3. определяем первый объект в очереди на попадание. Если данный объект имеет тэг “Player” то запускается алгоритм движения к данной позиции.

# Программные решения

**Описание существующих разработок.**

В компьютерной индустрии существует огромное количество разнообразных жанров, под определение которых подпадает тот или иной проект.

1. **Valorant**



Рисунок 1. Запуск игры.

Valorant — это командный-соревновательный шутер.

У игры отличная атмосферная картинка и звук, никаких проблем с оптимизацией, интересный игровой процесс. Карты в игре – сбалансированные с кучей коридоров, укрытий и переходов.

Сюжет игры:

Игра является противостоянием двух команд по пять игроков, каждый из них играет в роли «агентов» с особенными способностями, которые либо покупаются перед началом раунда, либо зарабатываются в процессе матча. В ассортименте оружия игры есть нож, пистолеты, дробовики, пистолеты-пулеметы, пулеметы, автоматы и снайперские винтовки.

В основном режиме игры у атакующей команды есть бомба, которую они должны установить на точке закладки. Если атакующая команда успешно осуществляет подрыв, то она получает очко. Если команда защиты успешно обезвреживает бомбу или не даёт её установить атакующей команде по истечении 100-секундного таймера, то команда защиты получает очко. Уничтожение всех членов команды противника также приносит победу в раунде. Первая команда, выигравшая 13 раундов, побеждает в матче. После 12 раундов происходит смена сторон. 

Рисунок 2. Выбор агента



Рисунок 3. Командная работа.

Ссылка на скачивание игры:

https://playvalorant.com/en-us/download/

1. **Fall Guys Ultimate knockout**



Рисунок 4. Запуск игры.

Fall Guys — игра в жанрах королевской битвы, платформера и аркадной игры. В одном матче может принимать участие до 60 игроков. Каждый матч разбит на несколько раундов. В каждом раунде игроки в ходе участия в мини-играх должны выполнить определённую задачу, при этом часть игроков, которые не выполнили или не успели выполнить условия игры, выбывает. Данные мини-игры могут быть как индивидуальными, так и требующими выполнения задачи в командах из нескольких игроков, которых в зависимости от мини-игры может быть от двух до четырёх.

Мини-игры могут иметь разные условия победы и поражения, например: добраться до финиша раньше, чем определённый процент игроков, оставаться на уровне определённый промежуток времени, забить наибольшее количество мячей другой команде и т. д. Игра продолжается до тех пор, пока в финальном раунде не останется один участник.

Рисунок 5. Игровой процесс

Ссылка на скачивание: https://store.steampowered.com/app/1097150/Fall\_Guys\_Ultimate\_Knockout/

1. **Fortnite**



Рисунок 7. Окно запуска.

Fortnite — это кооперативная песочница на выживание, основными механиками которой являются исследование, сбор ресурсов, строительство укрепленных зданий и борьба с волнами наступающих зомби. Игроки кооперируются, чтобы собирать предметы, которые они могут использовать для строительства форта днём, а ночью защищаться от зомби. Строительство является основной механикой, при этом в игре огромное количество ресурсов для строительства. Игроки могут строить и редактировать каждую стену своего форта с сеткой 3х3, строить лестницы, крышу и окна, ваяя их в соответствии с той или иной потребностью.

Всего в Fortnite четыре основных режима игры, среди которых каждый игрок сможет выбрать тот, что подходит ему больше всего. «Королевская битва», «Королевская вечеринка», творческий режим и «Сражение с Бурей».

Режим Королевская битва.

Игроки в количестве до 100 человек сражаются в «королевской битве», где игроки воюют между собой, чтобы стать последним выжившим. Игроки могут выбирать, как войти в матч: в одиночку, в паре или отрядом из четырёх человек. В любом случае, последний человек или команда, оставшаяся в живых, выигрывает матч. Каждый матч начинается с того, что все игроки прыгают с дельтапланом из летящего синего автобуса над картой. Игроки начинают раунд с киркой, с которой можно разрушать постройки и предметы, тем самым добывая строительные материалы. Как только игроки приземляются, они могут осматривать различные здания, постройки и другие сооружения в поиске оружия, патронов, материалов, медикаментов и многих других вещей. Эти предметы распределяются по всей карте в начале матча, причём определённые зоны имеют повышенный шанс нахождения там лучшего снаряжения. Убитые игроки оставляют на месте своей смерти снаряжение. Строительные материалы нужны для строительства сооружений; для их добычи требуется разрушать разные здания, деревья и многие другие разрушаемые предметы.

Творческий режим

Творческий режим позволяет создавать игры и новые миры Fortnite, которые можно опубликовать в сети и предложить друзьям. Создайте Fortnite по своим правилам на личном острове. А ещё вы можете опробовать множество игр от участников сообщества вместе с друзьями. для этого введите код острова или посетите творческий центр в игре.

Режим Королевская вечеринка

«Королевская вечеринка» — это режим внутри «Королевской битвы». «Королевская вечеринка» — это совершенно новое, постоянно меняющееся пространство. Здесь не нужно сражаться, здесь вообще не нужно ни о чём заботиться — можно просто расслабиться и хорошо проводить время. Среди развлечений — воздушная полоса препятствий, гонки на катерах, фильмы и даже концерты звёзд музыкальной сцены.

Режим Сражение с бурей

В «Сражении с Бурей» вам предстоит исследовать огромный разрушаемый мир и вместе с другими игроками отбиваться от полчищ чудовищ. Сражаясь с этими монстрами — мозгляками — вы постепенно становитесь сильнее, собирая добычу, создавая оружие и расширяя вашу коллекцию героев. Стройте базу, укрепляйте её с помощью ловушек и сдерживайте натиск противников.



Рисунок 8. Королевская битва.



Рисунок 9. Творческий режим.

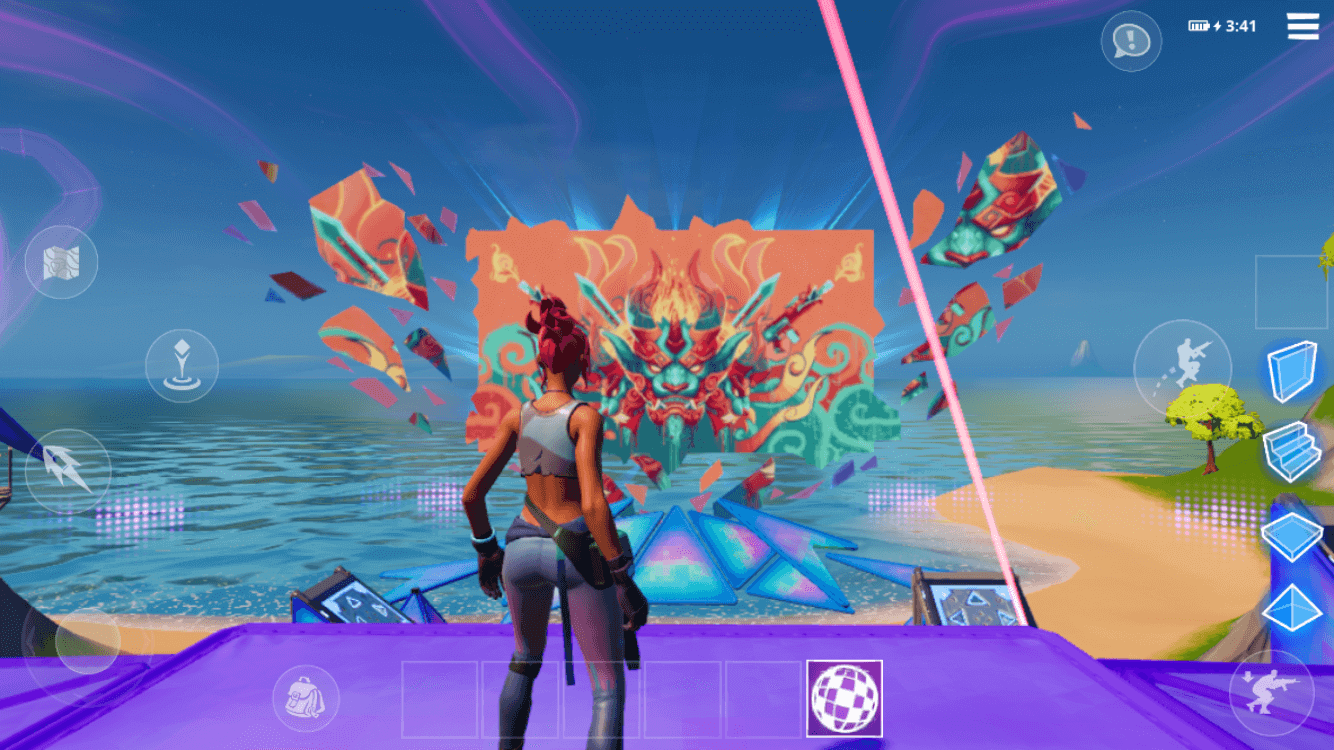


Рисунок 10 Королевская вечеринка.



Рисунок 11 Сражение с бурей.

Ссылка на скачивание:

<https://www.epicgames.com/fortnite/ru/download>

**Технология обработки данных**

Для обработки данных на компьютерах мастерской «Программные решения для бизнеса» было установлено следующее программное обеспечение, приведенное в Таблице 1 (Программное обеспечение для обработки данных), а также указаны ссылки на вебсайты разработчиков в Таблице 2 (Вебсайты используемого программного обеспечения для обработки данных):

**Таблица 1**

Программное обеспечение для обработки данных

|  |  |  |
| --- | --- | --- |
| № | Наименование ПО | Предназначение |
| 1. | MS Office 2013 Pro Plus | Работа с текстовой информацией, таблицами, презентациями |
| 2. | Adobe Acrobat Reader DC - Russian | Просмотр и работа с PDF-файлами |

**Таблица 2**

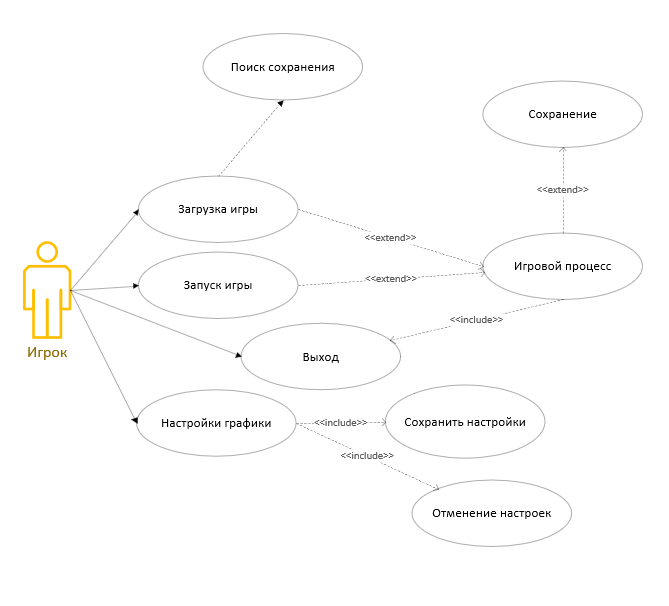
Вебсайты используемого программного обеспечения для обработки данных

|  |  |  |
| --- | --- | --- |
| № | Наименование ПО | Вебсайт разработчика |
| 1. | Образовательный портал Технологического университета (МГОТУ) город Королёв Московская область | https://ies.unitech-mo.ru/ |

# Алгоритм решения

**Алгоритм решения**

Для более наглядного представления алгоритма решения программы была создана диаграмма прецедентов, которая представлена ниже на рисунке 1(Диаграмма прецедентов), отражающая отношения между акторами (в данном случае это игроки) и прецедентами (действия игроков) и являющаяся составной частью модели прецедентов.

Рисунок 1. Диаграмма прецедентов.

**Таблица 1**

Описание прецедентов

|  |  |
| --- | --- |
| Прецедент | Краткое описание |
| Загрузка игры | Запускается игроком. Включает в себя расширяемый прецедент «поиск сохранения», «игровой процесс». |
| Запуск игры | Запускается игроком. Включает в себя расширяемый прецедент «игровой процесс». |
| Выход | Запускается игроком. |

# Выбор инструментов

Выбор инструментария разработчика должен быть четко обусловлен, для этого необходимо выделить ключевые требования, по которым будет подобран инструментарий.

Требования к продукту и его разработке:

* Кроссплатформенность. Приложение должно запускаться как на Android, так и на IOS.
* Время на разработку, отладку и тестирование. 1 месяц.
* Система контроля версий. Система контроля версий позволяет вести разработку более стабильно и безопасно.

Язык программирования и технология для разработки мобильных приложений.

Мной был выбран язык программирования C# и технология разработки кроссплатформенных мобильных приложений Xamarin.Froms. Разработка на языке C# крайне удобна из-за официальной интегрированной среды Visual Studio от Microsoft. Так же Xamarin Forms поддерживает только язык C#. Xamarin.Forms позволяет писать один код, который будет работать на всех выбранных платформах. Исключение составляют лишь платформозависимый код (работа с файлами манифеста девайса, системные настройки и т.д.) и комплексный интерфейс. За счёт небольшого количества платформозависимого кода время на разработку сокращается.

Среда разработки

Специальная среда для разработки приложений на C# это Visual Studio. Эта IDE, а именно Visual Studio Community 2019 является бесплатной в использовании. В этой среде очень удобно работать с различными библиотеками/фреймворками. Установить тот же Xamarin.Forms со всеми необходимыми инструментами для разработки и отладки можно прямо из меню установщика среды.

Система контроля версий

Выбор системы контроля версий тоже оказался не таким сложным. Выбор пал на github потому что функционал этой системы доступен каждому через любой браузер, без установки каких-либо дополнительных приложений.

# Тестовые наборы к программе

Тестовый набор №1.

Логин, введённый на девайсе – симуляторе: «1111»

Логин, введённый на основном девайсе: «1112»

Ожидаемый результат: неудачная авторизация через основное приложение и через приложение-симулятор. Отличие результата будет расценено как ошибка.2.

Тестовый сценарий работы: На устройстве отключают функцию Bluetooth, затем запускают тестируемую программу.

Ожидаемый результат: Системный запрос на требование о включении Bluetooth на устройстве от приложения.

# Описание главного модуля программы

На Рисунке 8 показана структура главного модуля будущей программы и взаимосвязь экранных форм программы между собой.

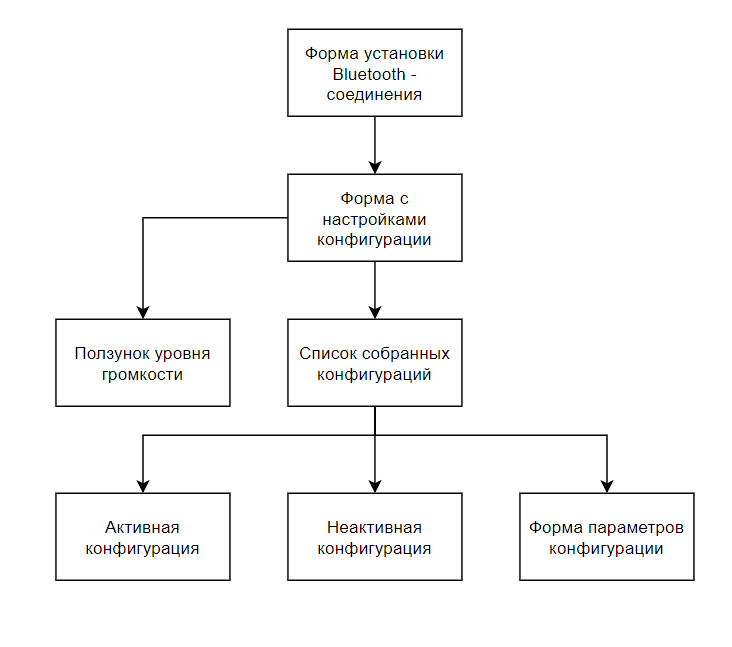


Рисунок 8. Структура главного модуля программы.

# Руководство оператора

1. Назначение программы.

Разрабатываемое программное средство предназначено для частичной настройки слухового аппарата посредством мобильного приложения с передачей данных на устройство через Bluetooth.

1. Условия выполнения программы.

Мобильная операционная система: Android 5.0 (Уровень API 21 - Lolipop) / IOS;

1. Выполнение программы.

При открытии приложения для соединения устройства со слуховым аппаратом необходимо нажать на кнопку.

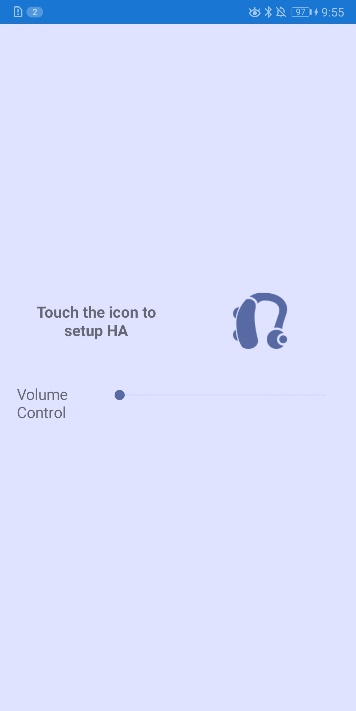


Рисунок 9. Главный экран до сопряжения

После чего приложение запросит разрешение на доступ к данным о местоположения устройства.

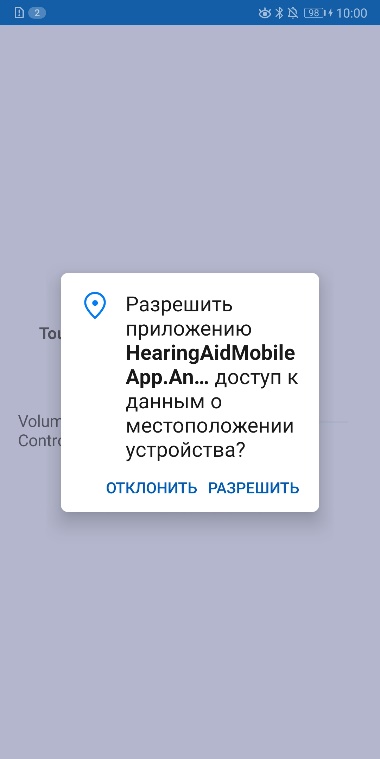


Рисунок 10. Разрешение доступа к местоположению

При нажатии на кнопку разрешить устройство синхронизируется со слуховым аппаратом и открывается окно для подтверждения ID СА.



Рисунок 7. Ввод ID СА

Если пользователь ввел корректный ID, то в интерфейс загружаются все данные из слухового аппарата.

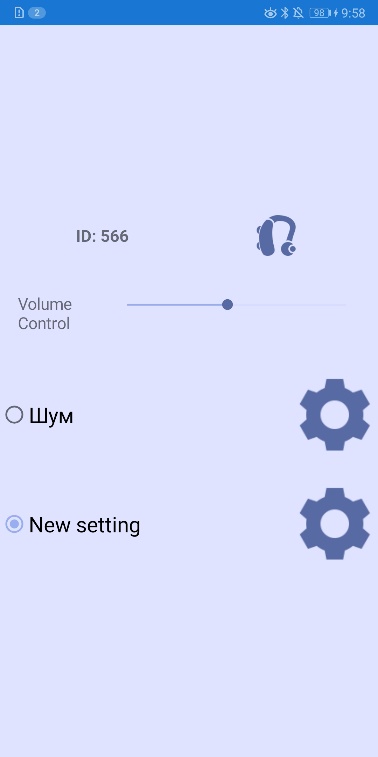


Рисунок 11. Главный экран после ввода ID

Оператор может настроить громкость СА с помощью слайдера громкости, выбрать нужную настройку, нажав на неё, и отредактировать её, нажав на кнопку «Шестеренка».

После нажатия на данную кнопку откроется окно для редактирования.



Рисунок 12. Окно редактирования настройки

В данном окне можно поменять название и режимы настройки. Нажав на строку с режимом, открывается список, из которого можно выбрать нужный.

1. Сообщение оператору

При верном вводе ID слухового аппарата приложение выводит сообщение «Correct Id»

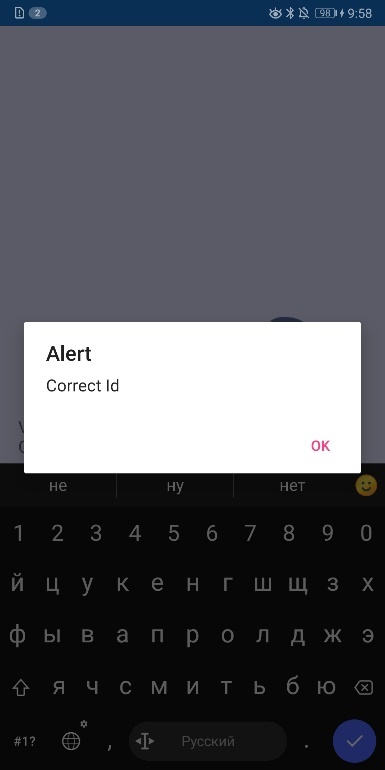
****

Рисунок 13. Сообщение о вводе правильного ID

Если оператор неправильно ввел ID, то приложение уведомит его об этом сообщением «Wrong Id». Интерфейс приложения не отобразит данных из слухового аппарата.

****

Рисунок 14. Сообщение о вводе неправильного ID

# Техника безопасности

Информация о технике безопасности приведена в файле «К16 – Раздел техники безопасности.docx».

# Источники

* Антамошкин, О. А. Программная инженерия. Теория и практика [Электронный ресурс] : учебник / О. А. Антамошкин. -

Красноярск: Сиб. Федер. ун-т, 2012. - 247 с. - ISBN 978-5-7638-2511-4. - Текст : электронный. -

URL: <https://znanium.com/catalog/document?pid=492527>

* Введение в программную инженерию : Учебник / В.А. Антипов, А.А. Бубнов, А.Н. Пылькин, В.К. Столчнев. —

Москва : КУРС: ИНФРА-М, 2019. — 336 с. - ISBN 978-5-16-103172-8. - Текст : электронный. -

URL: <https://znanium.com/catalog/document?pid=1035160>

* Гагарина, Л. Г. Разработка и эксплуатация автоматизированных информационных систем : учеб. пособие / Л.Г. Гагарина. —

Москва : ИД «ФОРУМ» : ИНФРА-М, 2019. — 384 с. — (Среднее профессиональное образование). -

ISBN 978-5-16-106202-9. - Текст : электронный. -

URL: <https://znanium.com/catalog/document?pid=1003025>

* Гагарина, Л. Г. Технология разработки программного обеспечения : учеб. пособие /

Л.Г. Гагарина, Е.В. Кокорева, Б.Д. Сидорова-Виснадул ; под ред. Л.Г. Гагариной. —

Москва : ИД «ФОРУМ» : ИНФРА-М, 2019. — 400 с. — (Высшее образование: Бакалавриат). -

ISBN 978-5-16-104071-3. - Текст : электронный. -

URL: <https://znanium.com/catalog/document?pid=1011120>

* Управление разработкой программного обеспечения

Альберт Нургалиев

URL: <https://stepik.org/course/56013/>

* Проектирование информационных систем

Владислав Тарасенко

URL: <https://stepik.org/course/63510>

# Дневник практики

|  |  |  |
| --- | --- | --- |
| **Дата** | **Содержание работ** | **Отметка о выполнении** |
| 21.04 | Изучение и обсуждение официальных документов |  |
| 22.04 | Проведение инструктажа по технике безопасности |  |
| 22.04 | Выдача заданий на преддипломную практику |  |
| 23.04 | Технико-экономическая характеристика объекта практики |  |
| 24.05 | Характеристика аппаратного обеспечения предприятия |  |
| 25.05 | Характеристика программного обеспечения предприятия |  |
| 26.05 | Методы проектирования и разработки |  |
| 27.05 | Математическая постановка задачи |  |
| 28.05 | Программные решения |  |
| 29.05 | Существующие технологии обработки данных |  |
| 30.05 | Консультация по описанию алгоритмов и решения |  |
| 03.05 | Консультация по основанию выбора инструментов, средств и языков |  |
| 04.05 | Консультация по описанию тестовых сценарию и тестовых наборов |  |
| 05.05 | Консультация по разработке и описанию структуры главного модуля |  |
| 06.05 | Консультация по разработке руководства оператора |  |
| 07.05 | Консультация по разделу техники безопасности |  |
| 11.05 | Консультация по общему оформлению записки |  |
| 12.05 | Консультация по созданию презентаций |  |
| 13.05 | Сдача отчётов по практике |  |
| 13.05 | Защита отчётов по практике |  |