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

1. Полное название учебно-научной мастерской:

«Разработка мобильных приложений»

1. Базовое подразделение (кафедра): ККМТ
2. Год создания мастерской: 2020
3. Месторасположение мастерской: Московская область, г. Королёв, ул. Пионерская, д.8
4. Заведующий мастерской (ФИО, должность)

В настоящий момент заведующий мастерской не закреплён. Данные о лаборатории были предоставлены заведующим мастерской «Программные решения для бизнеса», Поповым Вячеславом Николаевичем.

1. Специализация мастерской: проведение занятий по закреплённым за лабораторией дисциплинам.
2. Деятельность мастерской. Мастерской доступны следующие виды деятельности:

1) Образовательная. Проведение учебных дисциплин и курсов

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

1. Количество рабочих мест в лаборатории: 16

Планировка мастерской 105

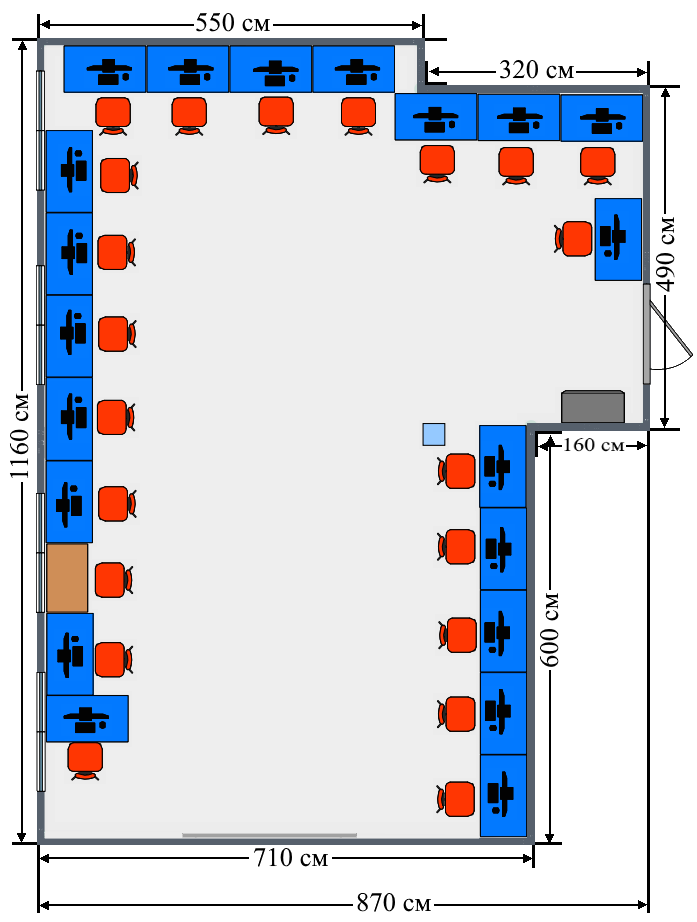


Рисунок . Общая схема

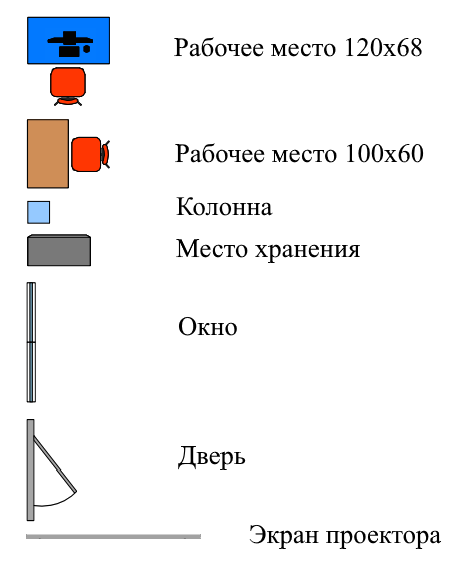


Рисунок . Легенда схемы

# Аппаратное обеспечение

Аппаратное обеспечение мастерской состоит из оборудования:

1. Автоматизированное рабочее место (16 шт.)

Характеристика:

ЦПУ: Core i7 9700 Soc-1151v2 (3.0GHz/Intel UHD Graphics 630)

ОЗУ: DDR4 16Gb

ПЗУ: SSD PCI-E x4 500Gb M.2 2280/ SATA-III 1Tb

Монитор: Dell 23.8" SE2416H черный IPS LED 16:9 HDMI матовая 250cd 178гр/178гр 1920x1080 D-Sub FHD.

2. Проектор Panasonic PT-VW360

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

# 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 . 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.

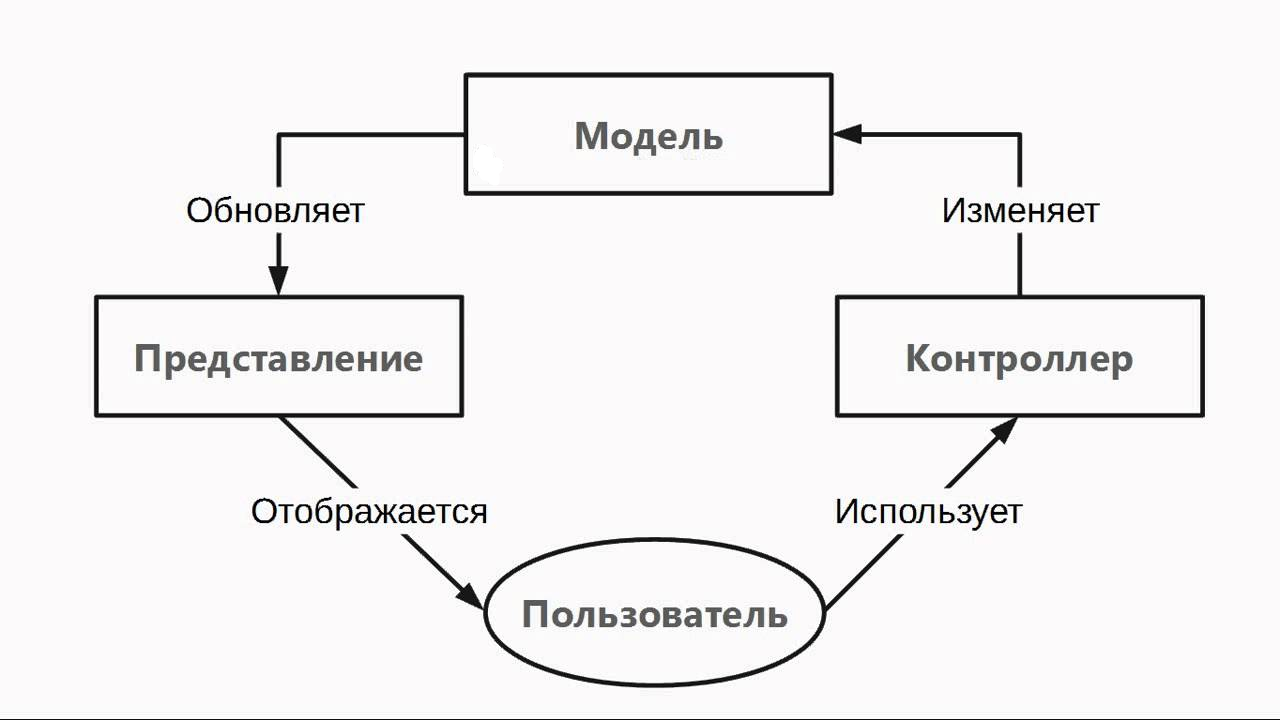


Рисунок 4. Пример использования метода проектирования MVC

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

Для защиты при обмене данными между приложением – клиентом и приложением - сервером, был использован алгоритм шифрования RSA

**Алгоритм RSA**

RSA (аббревиатура от фамилий Rivest, Shamir и Adleman) — криптографический алгоритм с открытым ключом, основывающийся на вычислительной сложности задачи факторизации больших целых чисел. Алгоритм RSA стал первым алгоритмом, пригодным и для шифрования, и для цифровой подписи.

В данном алгоритме имеется открытый ключ и закрытый ключ. Работа алгоритма происходит следующим образом:

Осуществляется генерация ключей: выбираются два достаточно больших случайных простых числа (желательно разрядностью 100-200 единиц или больше). Для большей безопасности ключи должны иметь равную длину.

p = 3557

q = 2579

Затем вычисляется произведение N = p \* q.

N = 3557 \* 2579

N = 9173503

После рассчитывается значение функции Эйлера по формуле:

φ(n) = (p-1)\*(q-1)

φ(n) = (3557-1)\*(2579-1) = 9167365

Далее выбирается открытый ключ(открытая экспонента) e (1 < e < φ(n)), взаимно простое со значением функции Эйлера

e = 3

Следом с помощью расширенного алгоритма Евклида вычисляется закрытый ключ шифрования D(секретная экспонента), удовлетворяющий условию:

**e \*** D≡1mod **(**φ(n))

D = 6111579

Заметим, что D и N также взаимно простые числа.

Числа E и N – это открытые ключи, а число D – закрытый.

Два простых числа p и q больше не нужны. Они могут быть отброшены, но не должны быть раскрыты.

При шифровании сообщение M сначала разбивается на цифровые блоки, размерами меньше N (для двоичных данных выбирается самая большая степень числа 2, меньшая N). Зашифрованное сообщение С будет состоять из блоков Ci такой же самой длины.

Предположим текст для шифрования M = 111111

Формула шифрования выглядит так:

Ci = E(Mi) = Mie mod(N)

C = 1111113 mod(9173503) = 4051753

При расшифровке сообщения для каждого зашифрованного блока Сi вычисляется по следующей формуле:

Mi = D(Ci) = Cid mod(N)

M = 40517536111579 mod (9173503) = 111111

# Программные решения

## Petralex Hearing Aid App

Petralex Hearing Aid App – мобильное приложение, доступное для установки на девайсы платформы Android и iOS. Предназначено для людей, теряющих слуховую способность. С помощью микрофона подключаемых к телефону наушников, приложение способно преобразовать поступающий в устройство звук в более громкий, на основе установленной пользователем настройки.

Сайт приложения в Google Play:

<https://play.google.com/store/apps/details?id=com.it4you.petralex&hl=en&gl=US>

Сайт прилолжения в App Store

<https://apps.apple.com/us/app/apple-store/id816133779>

Функционал, представленный на сайте:

* Автоматическая корректировка под специфику слуха;
* Коррекция слуха для каждого уха;
* Адаптация к различным типам окружающей среды;
* Усиление до 30 дБ с проводной гарнитурой;
* Встроенный тест уровня повреждения слуха;
* Динамическое сжатие. Усиление тихих звуков без потери общих.
* Встроенный 4 - недельный адаптивный курс для привыкания к приложению;
* Использование смартфона в качестве удалённого микрофона;
* Поддержка Bluetooth – гарнитур.

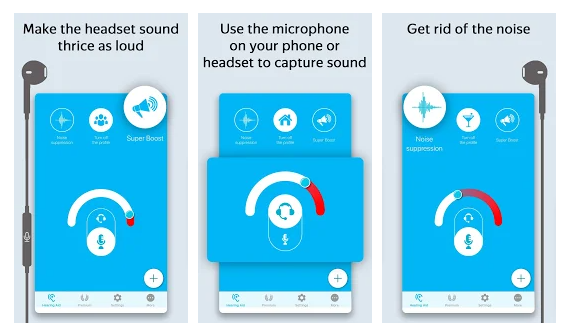


Рисунок 5. Интерфейс программы Petralex Hearing Aid App

## Hear Boost. Enhanced Microphone Volume & Recording.

Hear Boost – приложение с мощным функционалом, созданное для помощи слабослышащим людям. Принцип работы данного приложения схож с принципом работы вышеописанного приложения Petralex. Звук записывается через микрофон устройства, проходит усиление и поступает в динамик.

Доступно для бесплатного скачивания в магазинах:

Google Play: <https://play.google.com/store/apps/details?id=com.audiofix.hearboost&hl=en&gl=US>

App Store:   
<https://apps.apple.com/us/app/hear-boost-enhanced-recorder/id1437159134>

Функционал, представленный на сайте:

* Усиление чувствительности микрофона в 200 раз;
* Запись звука издалека;
* Запись интервью;
* Использование в роли слухового аппарата;
* При использовании человеческий голос звучит громче и чётче;
* Неограниченное время записи аудио.



Рисунок 6. Интерфейс программы Hear Boost

# Технологии обработки данных

Используемые в ходе разработке приложения программы для обработки данных представлены в таблице 3, сайты разработчика программного обеспечения приведены в таблице 4.

**Таблица 3.**

Программы для обработки данных

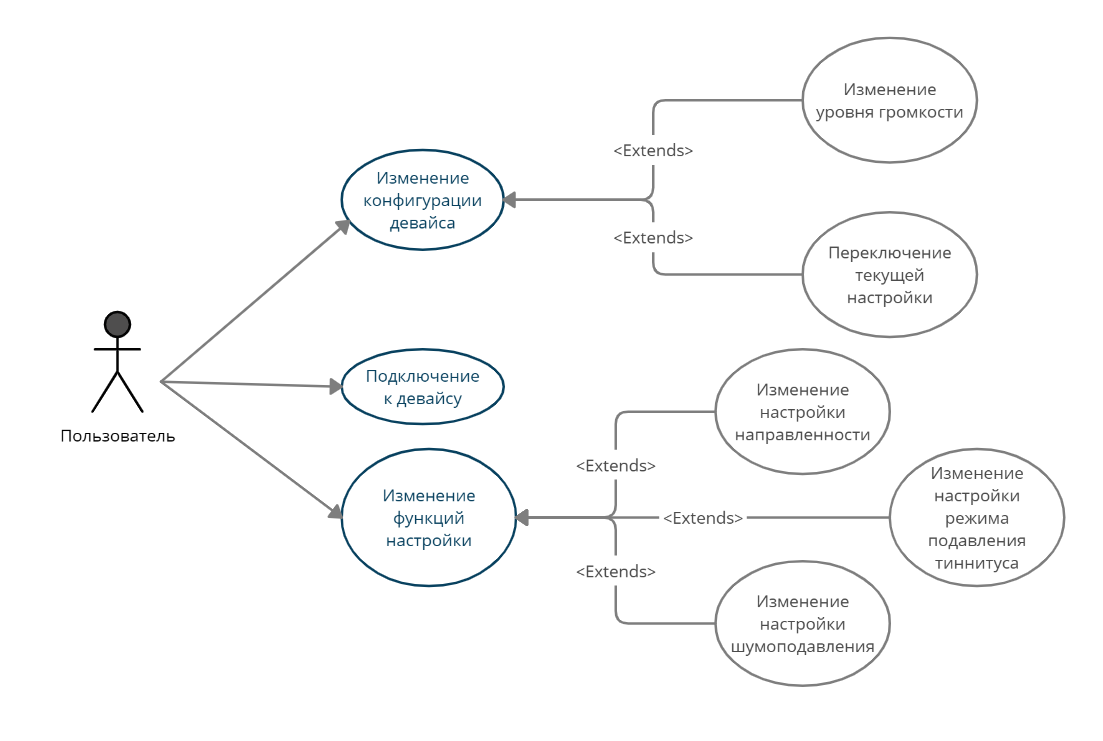
|  |  |  |
| --- | --- | --- |
| № | Наименование ПО | Предназначение |
| 1. | Adobe Acrobat Reader DC | Просмотр файлов типа PDF |
| 2. | Visio | Создание блок-схем |
| 3. | SSMS 18.6 | Управление базами данных. |
| 4. | Пакет приложений MS Office 2013 Pro Plus | Работа с текстовой информацией, таблицами, презентациями |

**Таблица 4.**

Сайты используемого программного обеспечения.

|  |  |  |
| --- | --- | --- |
| № | Наименование ПО | Сайт |
| 1. | Adobe Acrobat Reader DC | <https://get.adobe.com/ru/reader/> |
| 2. | Visio | <https://www.microsoft.com/en-ww/microsoft-365/visio/flowchart-software> |
| 3. | SSMS 18.6 | <https://docs.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver15> |
| 4. | Пакет приложений MS Office 2013 Pro Plus | <https://www.office.com/> |

# Алгоритм решения

На Рисунке 7 изображена Use Case диаграмма для программного продукта, который будет разработан в ходе выполнения дипломной работы:  Рисунок 7. Use Case диаграмма будущей программы

# 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.

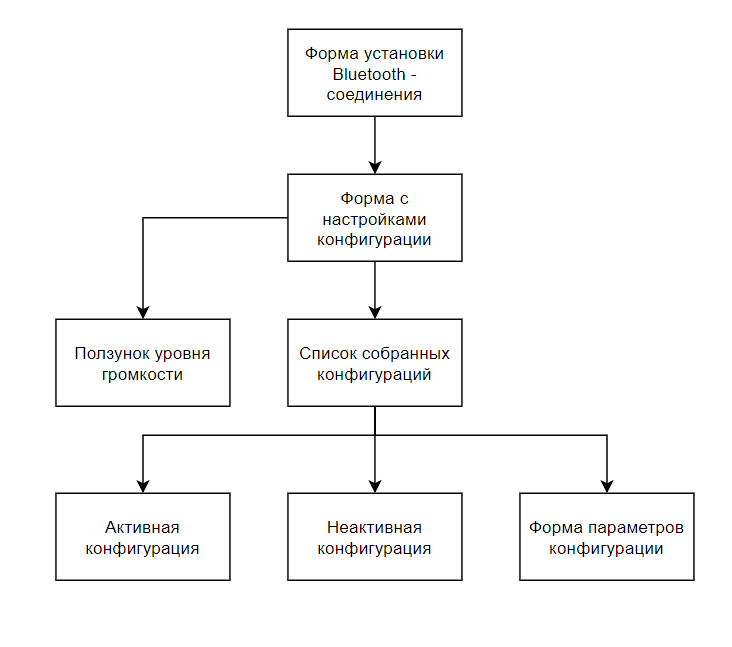


Рисунок 8. Структура главного модуля программы.

# 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.

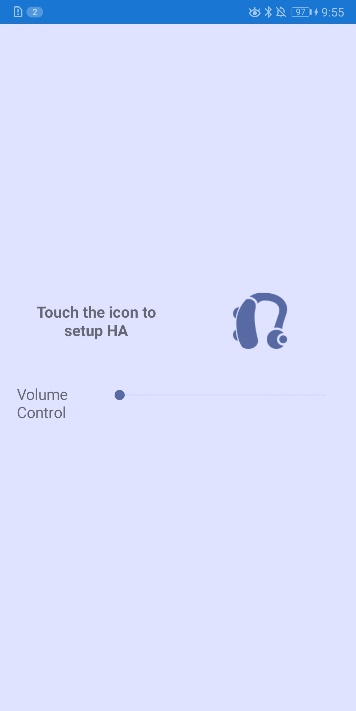


Рисунок 9. Главный экран до сопряжения

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

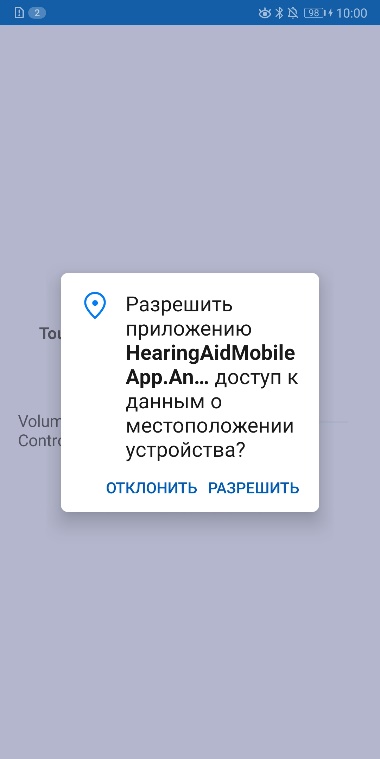


Рисунок 10. Разрешение доступа к местоположению

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

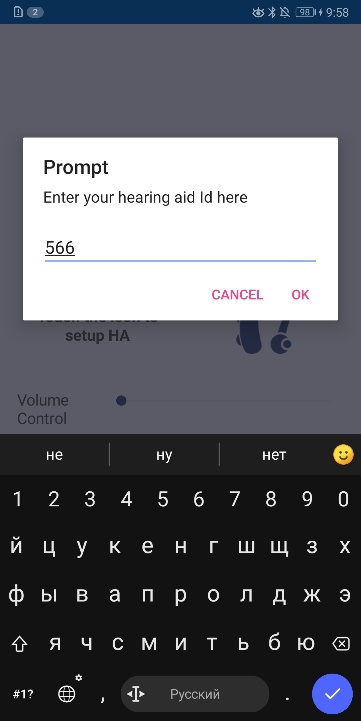


Рисунок 7. Ввод ID СА

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

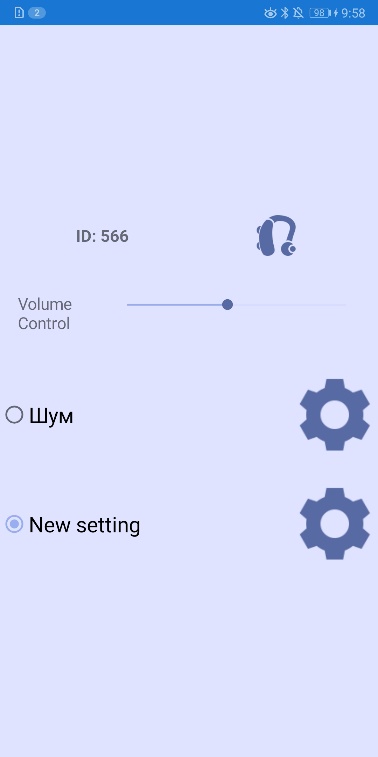


Рисунок 11. Главный экран после ввода ID

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»

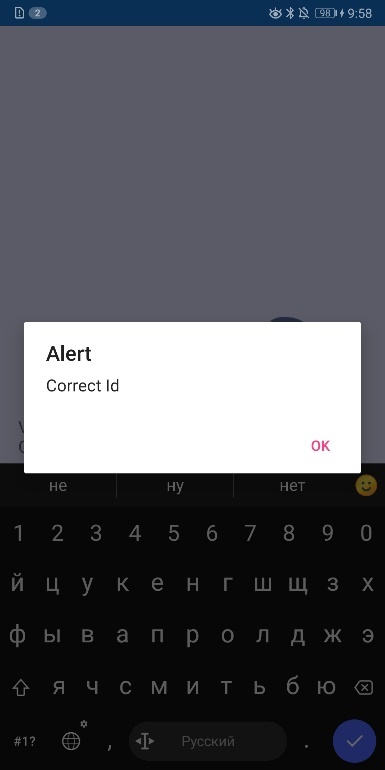
****

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.

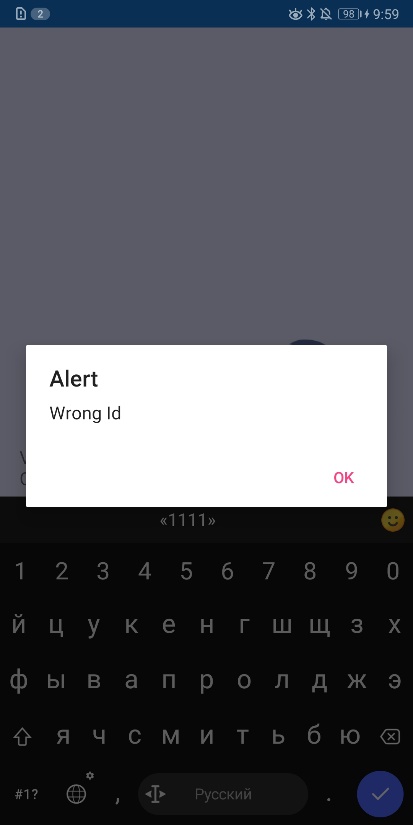
****

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 |  |