

Санкт-Петербургское государственное бюджетное профессиональное образовательное учреждение

«Радиотехнический колледж»

**КУРСОВОЙ ПРОЕКТ**

Тема: « Разработка приложения “FitNess World”»

**Специальность:** 09.02 07 «Информационные системы и программирование»

**Выполнил(а):** обучающийся группы ИП2К-22 Удальцов Дмитрий Игоревич

**Руководитель:** преподаватель Финская Анастасия Константиновна

Санкт-Петербург 2024г

**СОДЕРЖАНИЕ**

[Введение 3](#_Toc1)

[1. Разработка эскизного и технического проекта программы 4](#_Toc2)

[1.1. Назначение и область применения 4](#_Toc3)

[1.2.1. Основные понятия 5](#_Toc4)

[1.2.2. Постановка задачи 6](#_Toc5)

[1.2.3. Создание UML диаграммы 7](#_Toc6)

[1.2.4. Выбор состава технических и программных средств 10](#_Toc7)

[2. Разработка рабочего проекта 11](#_Toc8)

[2.1. Разработка программы 11](#_Toc9)

[2.5. Тестирование программы 18](#_Toc10)

[Заключение 20](#_Toc11)

[Список использованной литературы 21](#_Toc12)

# **Введение**

С появлением новых технологий и ускорением процесса цифровизации, индустрия фитнеса и здоровья претерпевает значительные изменения. Фитнес-приложения, используя инновации, предоставляют пользователям способ поддержания здоровья и благополучия. В течение последних лет, мы наблюдаем более активное развитие этих приложений, что делает их неотъемлемой частью жизни многих людей.

Основной целью данного отчета является анализ процесса разработки Фитнес-приложения, определение целей и задач, разработку и тестирование продукта. Отчет описывает этапы создания приложения, а также рассматривает ключевые технологические инновации, которые могут улучшить пользовательский опыт и функциональность продукта. В эпоху, когда здоровье становится приоритетом важно предлагать решения, которые не только отвечают потребностям пользователей, но и способствуют их активному участию в процессе поддержания своей физической формы.

Примерный перечень функциональных возможностей интерфейса программы Фитнес-приложения включает:

1. Регистрация новых пользователей.
2. Вход под разными пользователями.
3. Начало тренировки.
4. Удобный переход между упражнениями в процессе тренировки.
5. Таймер отчитывающий время от начала каждого упражнения.
6. Возможность ставить на паузу упражнение.

Актуальность и необходимость создания такой системы обусловлена следующими аспектами:

1. Растущий интерес к здоровому образу жизни: Современное общество все больше осознает важность поддержания физической активности.

Цель: Стимулировать пользователей к регулярным занятиям спортом и поддержанию здорового образа жизни.

# **1. Разработка эскизного и технического проекта программы**

## **1.1. Назначение и область применения**

Фитнес-приложение является мобильным приложением, разработанным для личного использования. Оно предназначено для помощи людям в достижении их целей, поддержания здорового образа жизни. Приложение может использоваться в любом месте и в любое время, предоставляя пользователям удобный инструмент для проведения тренировки.

Приложение разработано с использованием языка программирования Kotlin и может использоваться на всех устройствах, поддерживающих Android версии 5.0 и выше. Оно оптимизировано для работы в среде Android и не требует установки дополнительных фреймворков.

**1.2. Технические характеристики**

Для запуска данного Фитнес-приложение необходимо устройство, Соответствующее следующим техническим характеристикам:

1. Android версии 5.0 и выше.
2. Оперативная память: минимум 1 ГБ.
3. Экран: Сенсорный, минимальное разрешение 800x480 пикселей.
4. Свободное место на устройстве: Минимум 50 МБ.

### **1.2.1. Основные понятия**

Перечень сокращений, символов и специальных терминов с их определением:

Kotlin – статически типизированный язык программирования, работающий поверх JVM (Java Virtual Machine), используемый для разработки Android-приложений и других платформ.

Android Studio – официальная интегрированная среда разработки (IDE) для Android, предоставляющая инструменты для разработки, отладки и оптимизации приложений, написанных на Kotlin и других языках.

Gradle – система автоматической сборки, используемая в Android Studio, которая управляет зависимостями и позволяет создавать различные варианты приложения.

Activity – компонент приложения, который обеспечивает пользовательский интерфейс (UI), с которым пользователи могут взаимодействовать для выполнения различных действий.

Intent – механизм в Android для взаимодействия между компонентами (например, Activity) внутри приложения или между приложениями.

View – основной блок для создания пользовательского интерфейса в Android, который может быть кнопкой, текстовым полем, списком и т.д.

Layout – файл XML, который определяет визуальную структуру для пользовательского интерфейса, например, расположение кнопок и текстовых полей.

Drawable – ресурс, который может быть использован как фон, иконка или для других графических целей. Drawables могут быть различных форм, например: Bitmap, Nine-Patch, Shape, или State List.

Firebase – набор инструментов для разработки приложений от Google, который предоставляет сервисы, такие как аутентификация пользователей, облачное хранение данных, реальное время баз данных, аналитика и многое другое. Особенно полезно для создания масштабируемых и хорошо интегрированных приложений.

### **1.2.2. Постановка задачи**

Целью программы заключается в создании уникального Фитнес-приложения для поддержания человека в хорошей физической форме. Доступ должен быть реализован для зарегистрировавшихся пользователей.

Разрабатываемое программное обеспечение должно выполнять следующие функции:

Пользователь:

1. Регистрировать пользователей.
2. Отображать почту пользователя в разделе аккаунта.
3. Начинать тренировку.
4. После начала тренировки должен появляться таймер до окончания упражнения.
5. Автоматический переход к следующему упражнению после окончания нынешнего упражнения.

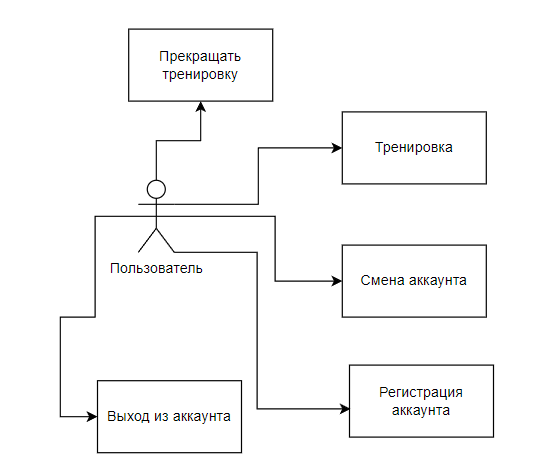
Для реализации данного проекта требуется:

1. Определение системных требований: Выбор устройства, соответствующего необходимым техническим характеристикам для оптимальной работы приложения.
2. Разработка программного обеспечения: Создание ПО с учетом заданных функциональных и технических требований..
3. Установка программного обеспечения: Установка программного обеспечения на выбранном устройстве.

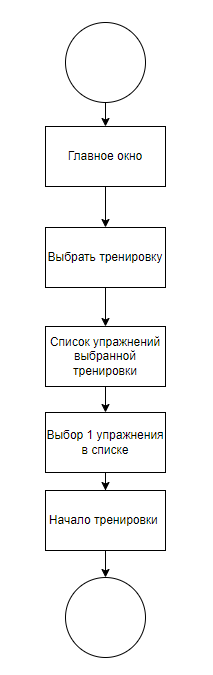
Модель предметной области для данного фитнес-приложения будет являться структурирование информации, разработку интуитивно понятного и удобного интерфейса, который позволяет пользователям легко взаимодействовать с приложением.

### **1.2.3. Создание UML диаграммы**

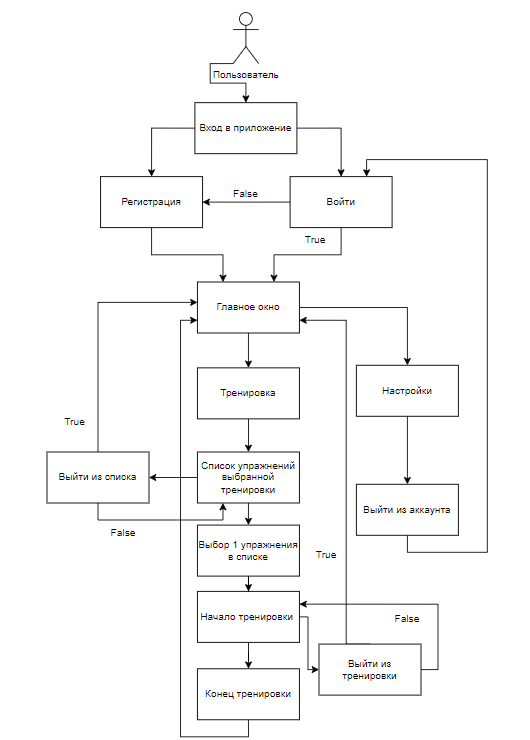
Проведя анализ предметной области я получил диаграмму диаграмму функциональных требований(рис. 1.1) и диаграмму последовательности (рис. 1.2):



**Рисунок 1.1 – UML Диаграмма функциональных требований**



**Рисунок 1.2 – Процесс начала тренировки**



**Рисунок 1.3 – UML Диаграмма последовательности**

### **1.2.4. Выбор состава технических и программных средств**

В качестве технических средств будут выступать:

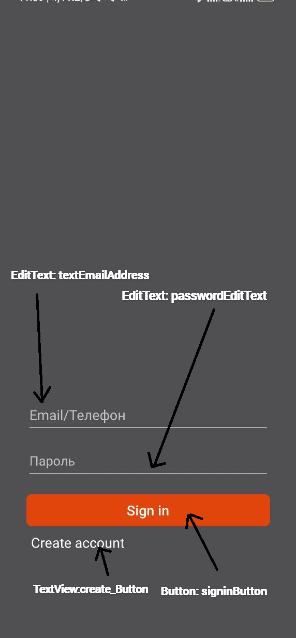
Программные средства:

* Язык программирования: Kotlin
* Среда разработки: Android Studio;
* Аппаратные средства:
* Windows 11.
* Видео память 8 ГБ.
* Оперативной памяти 16 ГБ.
* Объем свободного места на жестком диске 300 ГБ
* Манипуляторы: клавиатура, мышь и экран.

# **2. Разработка рабочего проекта**

## **2.1. Разработка программы**

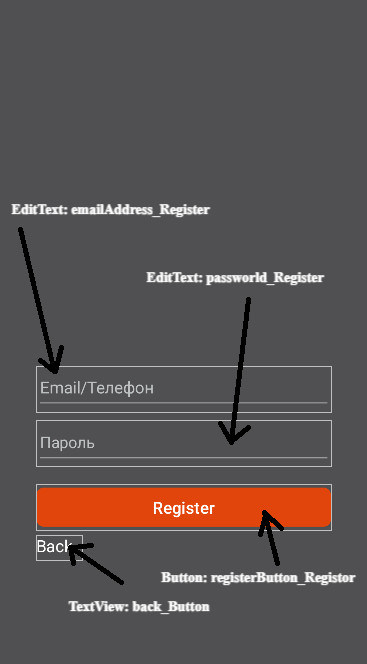
Для работы приложения были реализованы такие функции как: Вход в приложение (рис. 2.1), регистрация пользователя(рис. 2.2), Выход из приложения (рис. 2.3), Начало тренировки (рис. 2.4). Отчет времени до окончания выполнения упражнения (рис. 2.5).



**Рисунок 2.1 – форма для входа пользователя**

Окно, где мы вводим почту и пароль и заходив в приложение в свой аккаунт, или переходим к регистрации аккаунта.

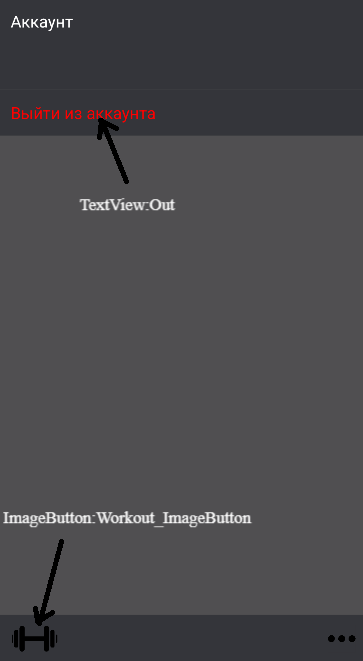
1. EditText: textEmailAddress – Поле ввода почты пользователя.
2. EditText: passwordEditText – Поле ввода пароля пользователя.
3. Button:signinButton – Кнопка для входа в приложение.
4. TextView:create\_Button – Кнопка для перехода к созданию аккаунта.



**Рисунок 2.2 – форма для регистрации пользователя**

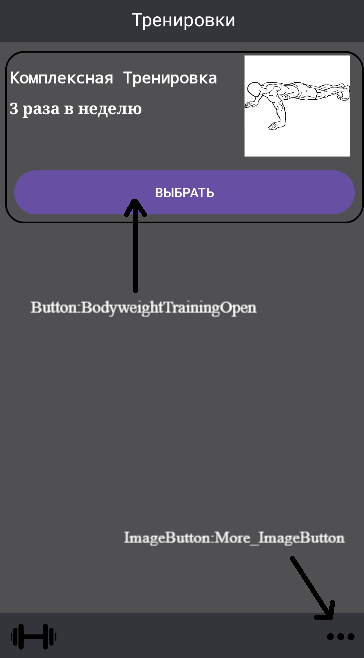
Окно, где мы вводим почту и пароль и регистрируем свой аккаунт, после регистрации автоматически заходит в приложение под наши аккаунтом. Так же можно вернуться в меню со входом нажав “Back”.

1. EditText:emailAddress\_Register – Поле ввода почты пользователя.
2. EditText:passworld\_Register – Поле ввода пароля пользователя.
3. Button:registerButton\_Registor– Кнопка для регистрации пользователя и входа в приложение после регистрации.
4. TextView:back\_Button – Кнопка для выхода из регистрации и перехода ко входу в аккаунт.



**Рисунок 2.3 – форма для выхода пользователя из приложения**

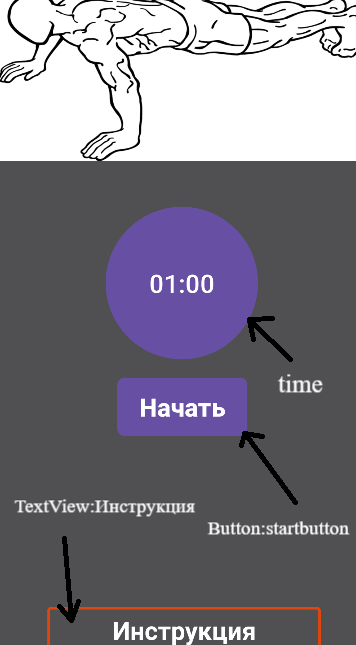
Окно, где мы можем выйти из аккаунта, посмотреть свою почту, и вернуться обратно в главное меню.



**Рисунок 2.4 – Начало тренировки**

Окно выбора тренировки и перехода к окну с почтой и выходом из аккаунта.

1. Button:BodyweightTrainingOpen – Кнопка что перекидывает нас к тренировке.
2. ImageButton:More\_ImageButton – Кнопка что открывает меню где можно посмотреть информацию о названии почты с которой мы регистрировались а также выйти из аккаунта.



**Рисунок 2.5– Отчет времени до окончания выполнения упражнения**

Окно выполнения упражнения, в нем после нажатия на кнопку startbutton начинается отчет до окончания выполнения упражнение, нажав повторно на кнопку тренировка по ставится на паузу.

Если пролистать вниз, то можно почитать инструкцию как правильно выполняется данное упражнение.

1. Button:startbutton – Кнопка что начинает выполнение упражнения.
2. time – Таймер отсчета до окончания тренировки.
3. TextМiew: Инструкция – Под надписью «Инструкция» находится описание как правильно выполняется упражнение.

**2.2. Описание программы**

Общие сведения: Разработанное приложение — является программой для устройств на базе Android, предназначенная для эффективного управления тренировками и поддержания здорового образа жизни.

Функциональное назначение: Программа дает пользователям инструменты для отслеживания процесса тренировки, а также мотивирует к регулярным занятиям спортом.

Для выполнения программы требуются:

Программные средства:

* Язык программирования: Kotlin
* Среда разработки: Android Studio;

Аппаратные средства:

* Windows 11.
* Видео память 8 ГБ.
* Оперативной памяти 16 ГБ.
* Объем свободного места на жестком диске 300 ГБ

Вызов: Программа запускается нажатием на иконку приложения.

## **2.5. Тестирование программы**

При разработке программы использовалась среда программирования o Android Studio. Эта среда представляет собой набор средств быстрой разработки и отладки приложений для Android. При разработке приложения применялся визуальный редактор, редактор программного кода.

Основными операциями в процессе разработки являлись: создание форм, размещение компонентов, редактирование свойств компонентов, формирование визуального пользовательского интерфейса, написание кода программы для обработки событий, сохранение и компиляция проекта.

В процессе выполнения тестирования требуется проверить программный модуль на работоспособность и устойчивость к некорректному вводу.

После чего сделать вывод о работоспособности приложения, и при необходимости принять меры к исправлению. В качестве вводимых данных указаны последовательный ввод и действия пользователя.

Порядок строк соответствует:

1. Ввод почты
2. Ввод пароля
3. Нажатие кнопки Register

Табл. 2.3 – Тестирование

|  |  |  |  |
| --- | --- | --- | --- |
| Номер тестирования | Вводимые данные | Ожидаемый результат | Итоговый результат |
| 1 | Null | Fields cannot be empty | Fields cannot be empty |
| 2 | Email- [testemail@gmail.com](http://testemail@gmail.com)  Passworld-qweqwe | Пользователь зарегистрирован | Пользователь зарегистрирован |
| 3 | Email- [testemail@g31m321ail.com](http://testemail@gmail.com)  Passworld-qweqwe | You have some Error | You have some Error |
| 4 | Email- [testemail@g31m321ail.com](http://testemail@gmail.com)  Passworld-Null | Fields cannot be empty | Fields cannot be empty |

При испытании программы ставилась цель проверить работоспособность программы, корректность расчетов и отсутствие логических ошибок. Также ставилась задача исключить «человеческий фактор» возникновения исключительных ситуаций и ошибок. В ходе выполнения тестирования не было выявлено значительных неполадок.

# **Заключение**

В заключение, данная курсовая работа привела к созданию фитнес-приложения на Kotlin, которое способствует улучшению здоровья и физической формы пользователей. Основные функции, такие как регистрация, таймер выполнения упражнений и автоматический переход к следующему упражнению, способствуют развитию и поддержанию стабильных тренировочных привычек. Интуитивно понятный интерфейс приложения мотивирует пользователей к регулярным занятиям спортом, обеспечивая при этом безопасное хранение данных. С потенциалом для дальнейшего развития и поддержки, это приложение обещает стать неотъемлемой частью повседневной жизни многих людей, а также найти применение в областях, связанных с фитнесом, здоровьем и общим благополучием.

# **Список использованной литературы**

1. <https://kotlinlang.org/docs/home.html> – Документация по Kotlin
2. <https://firebase.google.com/docs?hl=ru> - Документация по Firebase
3. <https://metanit.com/kotlin/tutorial/> – Руководство по языку Kotlin
4. <https://kotlinlang.ru/> – Руководство по языку Kotlin