

Воронежский Государственный Университет
наименование организации – разработчика ТЗ на АС

УТВЕРЖДАЮ

Тарасов Вячеслав Сергеевич (Старший преподаватель, Воронежский Государственный Университет)
Руководитель (должность, наименование предприятия – заказчика АС)

Личная Расшифровка
подпись подписи

Печать

Дата 13.03.2024

УТВЕРЖДАЮ

Таранцов Иван Юрьевич (Студент, Воронежский Государственный Университет)
Руководитель (должность, наименование предприятия – разработчика АС)

Личная Расшифровка
подпись подписи

Печать

Дата 13.03.2024

Мобильный фитнес-тренер
наименование вида АС

Приложение для индивидуальных домашних тренировок
наименование объекта автоматизации

Fitobochka
сокращенное наименование АС

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

На 2-24 листах

Действует с 13.03.2024

СОГЛАСОВАНО

Тарасов Вячеслав Сергеевич (Старший преподаватель, Воронежский Государственный Университет)
Руководитель (должность, наименование предприятия – заказчика АС)

Личная Расшифровка
подпись подписи

Печать

Дата 13.03.2024

Содержание

1 Общие сведения.....	4
1.1 Наименование системы.....	4
1.1.1 Полное наименование системы и название приложения	4
1.1.2 Краткое наименование системы	4
1.2 Основания для проведения работ	4
1.3 Наименование исполнителя и заказчика приложения.....	4
1.3.1 Наименование заказчика	4
1.3.2 Наименование исполнителя	4
1.4 Плановые сроки начала и окончания работ.....	5
1.5 Сведения об источниках и порядке финансирования работ	5
1.6 Порядок оформления и предъявления заказчику результатов работ по созданию приложения.....	5
2 Назначения приложения	6
2.1 Назначение проекта.....	6
2.2 Цели проекта.....	6
2.3 Задачи проекта.....	6
2.3.1 Для незарегистрированного пользователя.....	6
2.3.2 Для зарегистрированного пользователя.....	6
2.3.3 Для администратора.....	6
3 Требования к приложению	7
3.1 Общие требования.....	7
3.1.1 Требования к архитектуре	7
3.1.2 Требование к эргономике и технической эстетике.....	7
3.1.3 Требования к защите информации	7

3.1.4 Требования по патентной части.....	7
3.1.5 Требования к используемым технологиям	7
3.2 Требования к функциям приложения.....	9
3.2.1 Макет экрана поиска музыки.....	9
3.2.2 Макет экрана плеера	10
3.2.3 Макет экрана авторизации	11
3.2.4 Макет экрана регистрации	12
3.2.5 Макет экрана восстановления пароля.....	13
3.2.6 Макет экрана личной страницы.....	14
3.2.7 Макет экрана редактирования профиля.....	15
3.2.8 Макет экрана добавленной музыки.....	16
3.2.9 Макет экрана загруженной музыки.....	17
3.2.10 Макет экрана добавления музыки	17
3.2.11 Макет экрана редактирования музыки.....	18
4 Состав и содержание работ по созданию приложения	19
5 Порядок контроля и приемки приложения	21
6 Требования к составу содержания работ по вводу системы в эксплуатацию	22
7 Требования к документированию	23
8 Источники разработки	24

1 Общие сведения

1.1 Наименование системы

1.1.1 Полное наименование системы и название приложения

Полное наименование: приложение для индивидуальных домашних тренировок.

1.1.2 Краткое наименование системы

Краткое наименование музыкального приложения: Fitobochka.

1.2 Основания для проведения работ

На основании следующих документов создается данное приложение:

- Техническое задание
- Дизайн-макет приложения созданный в Figma

1.3 Наименование исполнителя и заказчика приложения

1.3.1 Наименование заказчика

Заказчик: Старший преподаватель Тарасов Вячеслав Сергеевич. Воронежский Государственный Университет, Факультет Компьютерных Наук, кафедра Программирования и Информационных Технологий.

1.3.2 Наименование исполнителя

Исполнитель: студент Таранцов Иван Юрьевич. Воронежский Государственный Университет, Факультет Компьютерных Наук, кафедра Программирования и Информационных Технологий.

Исполнитель: студент Тернавский Даниил Олегович. Воронежский Государственный Университет, Факультет Компьютерных Наук, кафедра Программирования и Информационных Технологий.

Исполнитель: студент Путилин Максим Дмитриевич. Воронежский Государственный Университет, Факультет Компьютерных Наук, кафедра Программирования и Информационных Технологий.

1.4 Плановые сроки начала и окончания работ

Плановый срок начала работ – март 2024 г.

Плановый срок окончания работ – июнь 2024 г.

1.5 Сведения об источниках и порядке финансирования работ

Финансирование работ осуществляется из личных средств Исполнителя.

1.6 Порядок оформления и предъявления заказчику результатов работ по созданию приложения

Результаты работ предъявляются Заказчику в следующем виде:

- Работающее, согласно настоящему Техническому Заданию, мобильное приложение
- Сервер мобильного приложения
- Документация к приложению
- Презентация в формате видео с демонстрацией функциональности приложения
- Презентация по Техническому Заданию
- Защита проекта

2 Назначения приложения

2.1 Назначение проекта

Создание мобильного фитнес-тренера.

2.2 Цели проекта

- Разработать приложение для индивидуальных домашних тренировок

2.3 Задачи проекта

2.3.1 Для незарегистрированного пользователя

- Просмотр базовых курсов
- Выполнение базовых курсов

2.3.2 Для зарегистрированного пользователя

- Просмотр упражнений
- Выполнение упражнений
- Отслеживание личного прогресса
- Использование калькулятора калорий
- Составление индивидуального тренировочного плана пользователем
- Автоматическое составление индивидуального тренировочного плана

2.3.3 Для пользователя с премиум подпиской

- Просмотр упражнений
- Выполнение упражнений
- Отслеживание личного прогресса
- Использование калькулятора калорий
- Составление индивидуального тренировочного плана пользователем
- Автоматическое составление индивидуального тренировочного плана
- Уникальные курсы

3 Требования к приложению

3.1 Общие требования

3.1.1 Требования к архитектуре

Приложение должно быть построено на трехуровневой архитектуре:
клиент (мобильное приложение) – сервер – база данных.

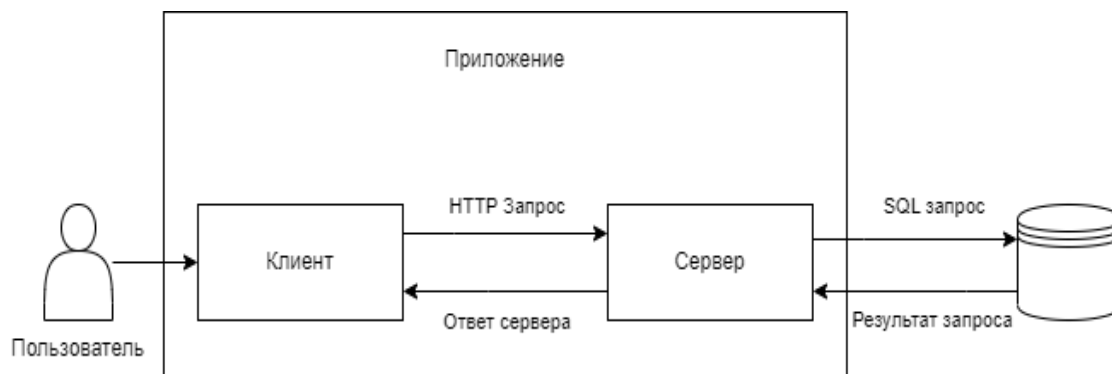


Рисунок 1 - Трехуровневая архитектура приложения.

3.1.2 Требование к эргономике и технической эстетике

Внешний вид приложения должен соответствовать дизайн-макету. Дизайн-макет будет разработан после утверждения настоящего Технического Задания.

3.1.3 Требования к защите информации

- Обеспечение авторизации и аутентификации пользователей
- Использование механизмов защиты от SQL-инъекций
- Использование протокола передачи данных HTTP

3.1.4 Требования по патентной части

Приложение должно не нарушать никаких лицензий и патентов. В случае нарушения данного пункта всю ответственность несет сторона Исполнителя.

3.1.5 Требования к используемым технологиям

Ниже приведен перечень используемых технологий, который в ходе разработки может расширяться.

Backend:

- Python – динамически типизированный, высокоуровневый, объектно-ориентированный язык программирования. Он был выбран в качестве основного, поскольку продолжает удерживать позиции одного из самых популярных языков программирования в современной разработке благодаря своей простоте, читаемости и гибкости. Python обладает обширной стандартной библиотекой и огромным количеством дополнительных библиотек и фреймворков

для различных направлений. Имеет большое и активное сообщество разработчиков, что обеспечивает обширную документацию, учебные материалы и поддержку

- Django – мощный веб-фреймворк с открытым исходным кодом для языка программирования Python. Он был выбран в качестве основного инструмента для разработки, так как обеспечивает высокую степень совместимости с множеством библиотек и фреймворков, что делает его подходящим для использования в разнообразных проектах и на различных платформах. Использует архитектурный шаблон MVC (Model-View-Controller), который помогает разделить логику приложения на три основных компонента: модели (бизнес-логика и работа с базой данных), представления (отображение данных пользователю) и контроллеры (управление потоком данных между моделями и представлениями). Известен своей высокой производительностью благодаря встроенным оптимизациям, таким как ORM (Object-Relational Mapping) для работы с базами данных
- PostgreSQL – Данное СУБД было выбрано т. к. является продуктом с открытым исходным кодом, который поддерживается многими серверами. Поддержка различных типов данных. PostgreSQL поддерживает множественные типы данных, такие как числа разной точности, тексты с различными кодировками, изображения, звуки, видео, XML–документы, JSON–объекты и многие другие
- Docker - инновационная платформа для создания, развертывания и управления контейнеризированными приложениями. Он позволяет упаковывать приложения и их зависимости в легковесные контейнеры, которые могут быть запущены на любой платформе, где установлен Docker. Это упрощает развертывание и масштабирование приложений, снижает вероятность конфликтов между зависимостями и обеспечивает консистентность окружения как на этапе разработки, так и в продакшне. Docker также позволяет

эффективно использовать ресурсы сервера, улучшает процесс разработки и управления приложениями, а также ускоряет процесс развертывания новых версий приложений.

Frontend:

- Flutter - современный кроссплатформенный фреймворк для разработки мобильных, веб- и настольных приложений. Он предоставляет универсальный набор инструментов для разработки приложений с красивым дизайном и плавной анимацией, которые могут одновременно работать на различных платформах, включая Android и iOS, без необходимости изменения исходного кода. Flutter использует язык программирования Dart и включает в себя мощный рендеринговый движок, а также богатую библиотеку предварительно созданных виджетов, что позволяет разработчикам быстро и эффективно создавать визуально привлекательные и функционально насыщенные приложения
- Dart — это современный объектно-ориентированный язык программирования, который часто используется в разработке веб- и мобильных приложений, а также для создания серверных приложений. Язык позволяет разработчикам создавать высокопроизводительные приложения благодаря своей быстрой виртуальной машине и компилятору AOT (Ahead-of-Time), который преобразует код Dart в машинный код для более эффективного выполнения. Кроме того, Dart обеспечивает простую и понятную систему типов, что способствует разработке более надежного и поддерживаемого кода. Одним из ключевых преимуществ данного языка является его интеграция с фреймворком Flutter. Благодаря этой интеграции, разработчики могут создавать приложения, которые могут работать на различных платформах, включая Android, iOS и веб-браузеры, без необходимости изменения исходного кода

Инструменты для ведения документации:

- Miro – платформа для совместной работы распределенных команд
- Swagger – это фреймворк для спецификации REST API
- Draw.io – Бесплатное кроссплатформенное программное обеспечение для рисования графиков с открытым исходным кодом. Его интерфейс можно использовать для создания диаграмм, таких как блок-схемы, каркасы, диаграммы UML
- Ramus – графическая среда для проектирования и моделирования сложных систем широкого назначения, который может быть использован для создания диаграмм в формате IDEF0
- Figma – онлайн-сервис для дизайнеров, веб-разработчиков и маркетологов. Он предназначен для создания прототипов сайтов или приложений, иллюстраций и векторной графики

Дополнительный инструментарий:

- Git – распределённая система управления версиями.
- GitHub – платформа разработки программного обеспечения с открытым исходным кодом, представляющая систему управления репозиториями кода для Git
- Firebase
- YouTrack – визуальный инструмент, обеспечивающий эффективность командной работы на любом проекте.

3.2 Требования к функциям приложения

3.2.1 Макет экрана главная страница

Пользователь (авторизованный и не авторизованный) имеет возможность найти упражнение для выполнения. Пользователь может нажать на упражнение, чтобы выполнить его.

Зарегистрированный пользователь имеет возможность отслеживать свой прогресс, использовать калькулятор калорий и автоматически составлять индивидуальный тренировочный план на основе своих физических данных.

Снизу расположена панель с кнопками, при нажатии на соответствующую кнопку пользователь переходит на страницы: главная,

прогресс, уведомления, профиль пользователя.

На данном экране расположены: тренировочный план на день, список упражнений.

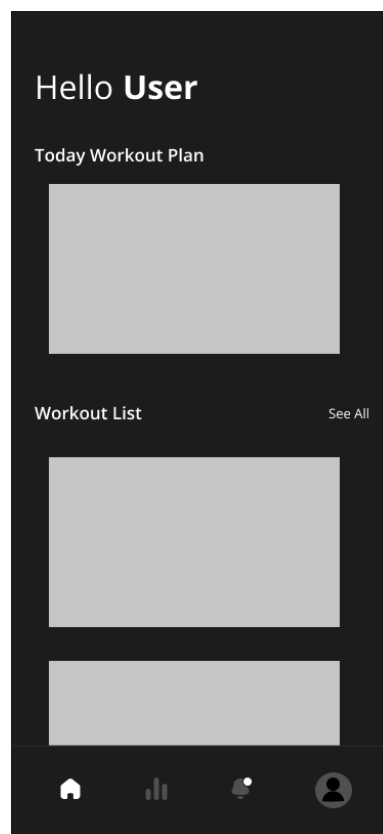


Рисунок 2 - Макет экрана главная страница

3.2.2 Макет экрана прогресс

Снизу расположена панель с кнопками, при нажатии на соответствующую кнопку пользователь переходит на страницы: главная, прогресс, уведомления, профиль пользователя.

На данном экране отображены поля со сожженными калориями пользователя, временем, затраченным на выполнение упражнений и с выполненными упражнениями.

Сверху расположена панель с датами (дни недели и числа месяца). Нажимая на даты, можно посмотреть свою статистику в выбранный день.

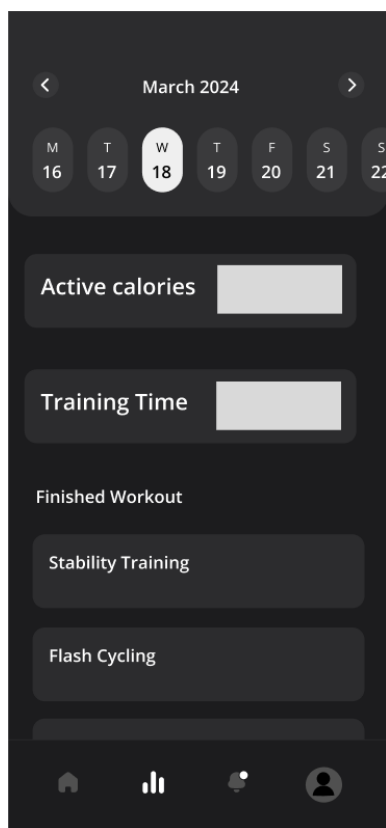


Рисунок 3 - Макет экрана прогресса

3.2.3 Макет экрана авторизации

На данном экране отображены поля с вводом логина и пароля к аккаунту, кнопка для входа, кнопка для регистрации, а также кнопка для восстановления пароля.

Снизу расположена панель с кнопками, при нажатии на соответствующую кнопку пользователь переходит на страницы: главная, прогресс, уведомления, профиль пользователя.

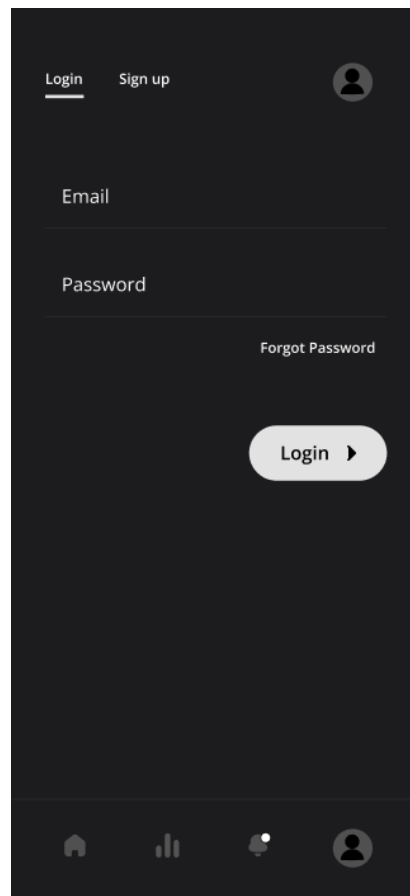
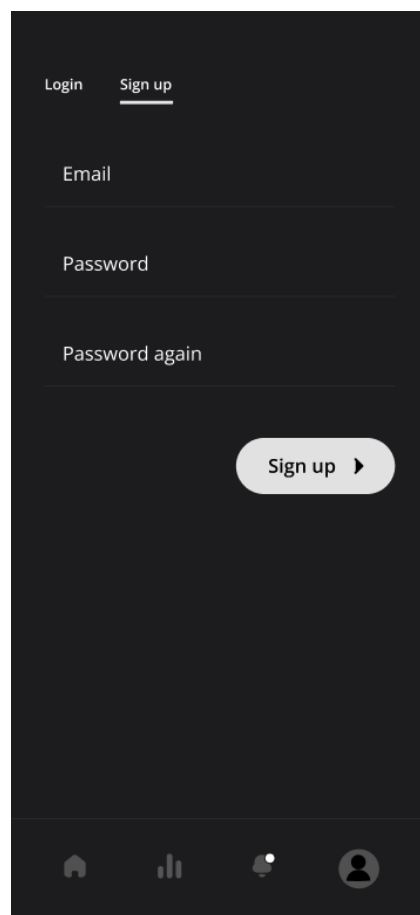


Рисунок 4 - Макет экрана авторизации

3.2.4 Макет экрана регистрации

На данном экране отображены поля для ввода почты, пароля и еще раз пароля. Ниже расположена кнопка регистрации.



The image shows a mobile application interface for registration. At the top, there are two tabs: "Login" and "Sign up", with "Sign up" being the active tab. Below the tabs are three input fields labeled "Email", "Password", and "Password again". A "Sign up" button with a right-pointing arrow is located to the right of the "Password again" field. At the bottom of the screen is a navigation bar with four icons: a home icon, a bar chart icon, a speech bubble icon, and a user profile icon.

Рисунок 5 - Макет экрана регистрации

3.2.5 Макет экрана восстановления пароля

На данном экране отображены поля для ввода адреса электронной почты и кнопка отправки кода. После введения данных появляется форма для ввода полученного кода, и кнопка для подтверждения кода. При успешной проверке кода, появляется форма для заполнения нового пароля, и кнопка подтверждения нового пароля. В случае неудачи пользователь будет возвращен на начальную форму восстановления пароля.

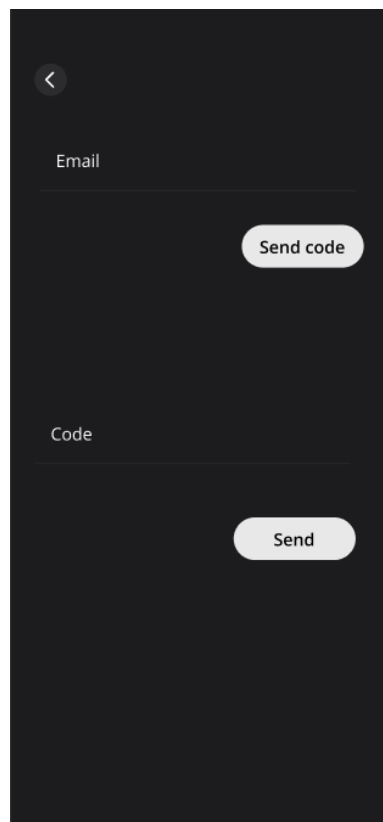
The image shows a vertical mobile app screen with a dark background. At the top left is a white back arrow icon. Below it is a text input field labeled "Email" in white. To the right of this field is a white button with rounded corners labeled "Send code". Below the "Email" field is another text input field labeled "Code" in white. To the right of this field is a white button with rounded corners labeled "Send".

Рисунок 6 - Макет экрана восстановления пароля

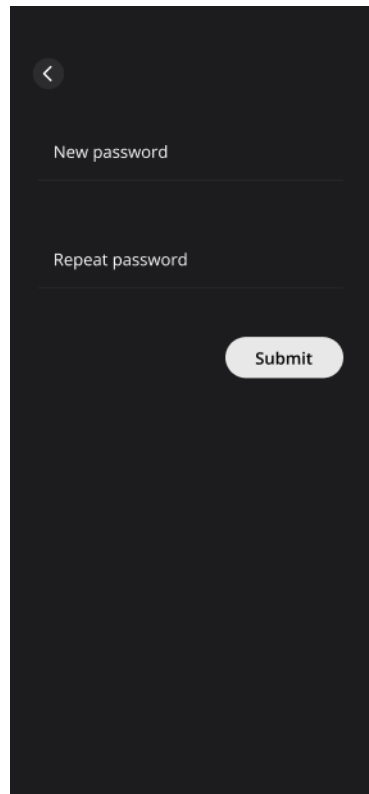


Рисунок 7 - Макет экрана ввода нового пароля

3.2.6 Макет экрана уведомления

На данном экране отображаются уведомления. После нажатия на уведомление, открывается экран с полным текстом уведомления.

Снизу расположена панель с кнопками, при нажатии на соответствующую кнопку пользователь переходит на страницы: главная, прогресс, уведомления, профиль пользователя.

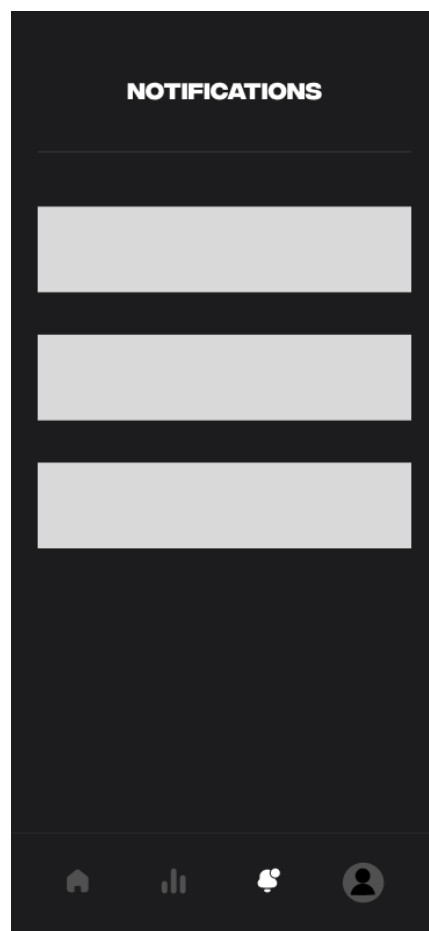


Рисунок 8 - Макет экрана уведомления

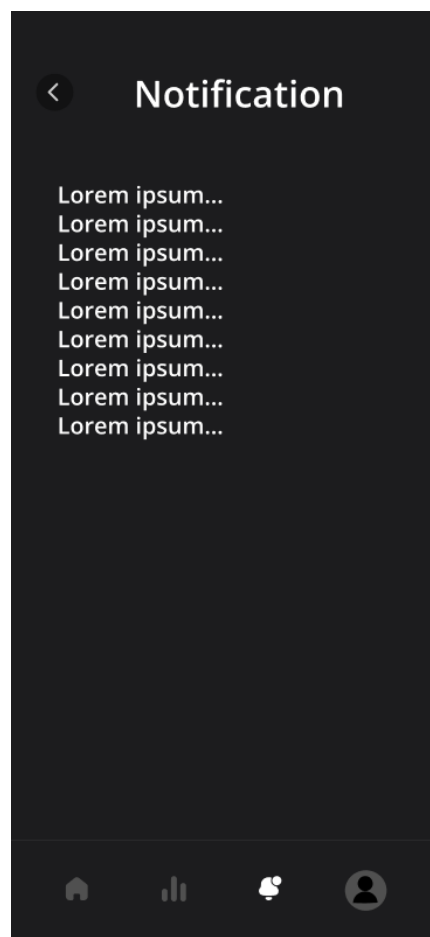


Рисунок 9 - Макет экрана уведомление

3.2.7 Макет экрана профиля

На данном экране указана информация о пользователе (имя, фамилия) рядом расположена кнопка редактирования пользователя и выхода из аккаунта.

Снизу расположена панель с кнопками, при нажатии на соответствующую кнопку пользователь переходит на страницы: главная, прогресс, уведомления, профиль пользователя.

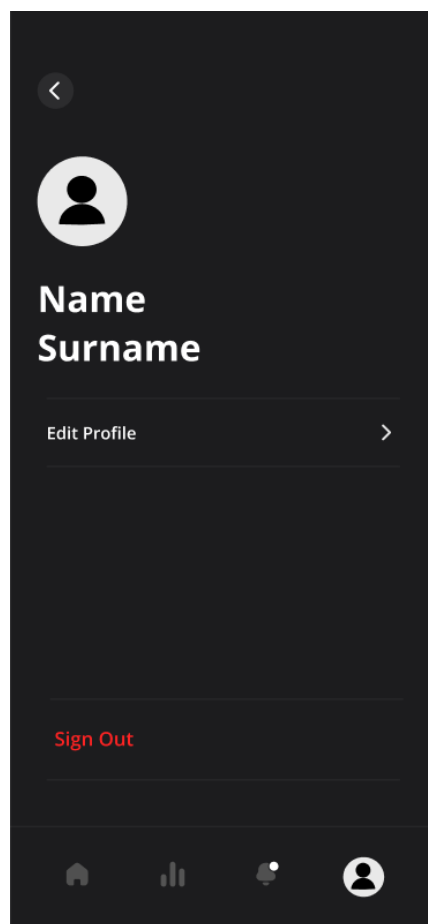


Рисунок 10 - Макет экрана личной страницы

3.2.8 Макет экрана редактирования профиля

На данном экране отображены поля для ввода имени, фамилии, почты. Ниже расположена кнопка для сохранения данных.

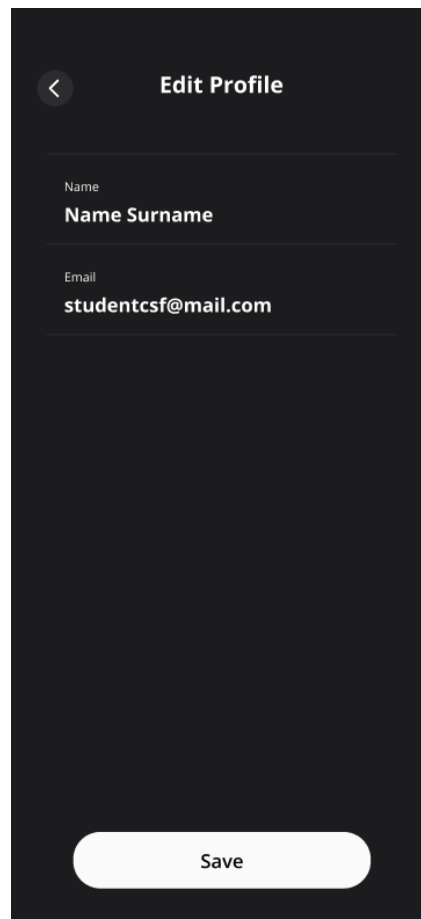


Рисунок 11 - Макет экрана редактирования профиля

3.2.9 Макет экрана индивидуальный тренировочный план

На этой странице отображается список упражнений, кнопка начала выполнения этих упражнений, название тренировки, время (которое займет тренировка), количество сожженных калорий после выполнения тренировки.

Снизу расположена панель с кнопками, при нажатии на соответствующую кнопку пользователь переходит на страницы: главная, прогресс, уведомления, профиль пользователя.



Рисунок 12 - Макет экрана индивидуальный тренировочный план

3.2.10 Макет экрана выполнения упражнения из тренировки

На данном экране расположены название упражнения, время выполнения, количество сжигаемых калорий, описание (как правильно выполнять и т.д.), таймер и кнопка перехода к следующему упражнению.

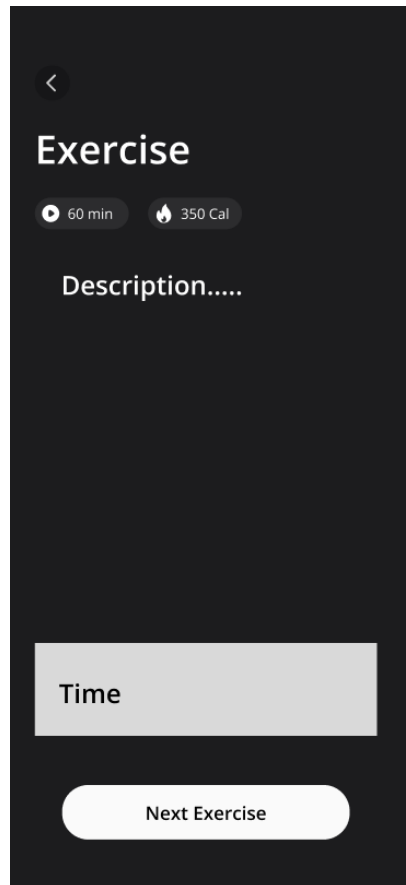


Рисунок 13 - Макет экрана выполнения упражнения

3.2.11 Макет экрана завершения тренировки

На данном экране расположены название тренировки, поздравления и количество затраченного времени и сожженных калорий, кнопка завершения.

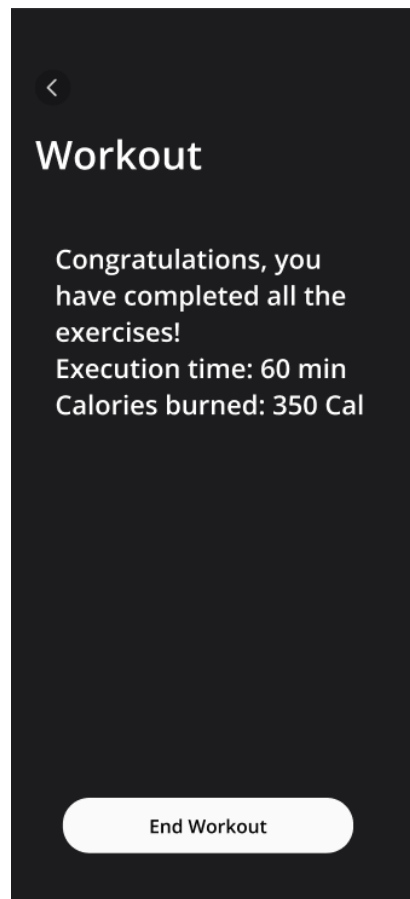


Рисунок 14 - Макет экрана добавления музыки

3.2.12 Макет экрана список упражнений

На экране расположены поля со всеми упражнениями и выбор уровня физической подготовки.



Рисунок 15 - Макет экрана список упражнений

3.2.13 Макет экрана выполнения одного упражнения

На экране расположены поля со всеми упражнениями и выбор уровня физической подготовки.

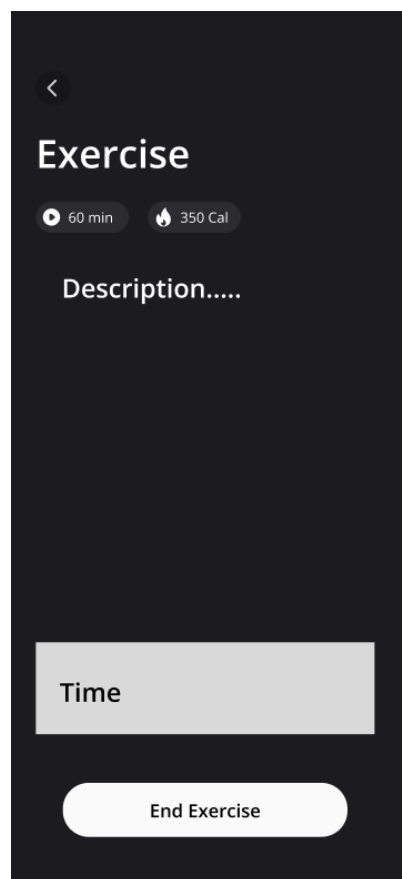


Рисунок 16 - Макет экрана выполнения одного упражнения

4 Состав и содержание работ по созданию приложения

Основные этапы работ, их содержание и приблизительные сроки приведены в Таблице 1.

Таблица 1 - Состав и содержание работ по созданию приложения.

Этап	Содержание работ	Порядок приемки и документы	Сроки	Ответственный
Составление технического задания	Разработка функциональных и нефункциональных требований к системе	Утверждение Технического Задания	13.03.2024 г. 21:00 по МСК	Разработка - Исполнитель. Согласование - Заказчик
Техническое проектирование	Разработка дизайн- макета мобильного приложения	Описание функциональной схемы в сервисе Miro	13.03.2024 г. 21:00 по МСК	Исполнитель
	Проработка сценариев мобильного приложения	Создание UML диаграмм	13.03.2024 г. 21:00 по МСК	Исполнитель
Разработка программной части	Разработка серверного модуля, модуля хранения данных	Приемка осуществляется в процессе испытаний, документация Swagger	Конец мая 2024г.	Исполнитель
	Развертывание серверной части приложения.	Приемка осуществляется в процессе испытаний	Конец мая 2024г.	Исполнитель
	Разработка мобильного Android- приложения и	Приемка осуществляется	Конец мая 2024г.	Исполнитель

	интеграция с серверной частью	я в процессе испытаний		
Тестирование функциональности приложения	Проверка на соответствие функциональности Техническому Заданию Проверка комплекта документации Доработка и повторные испытания до устранения недостатков	Согласованность с Техническим Заданием Обеспечение тестирования функциональности	Во время и после Разработки	Исполнитель
Эксплуатация	Эксплуатация с привлечением небольшого количества участников (несколько аукционов среди знакомых). Доработки и повторные испытания до устранения выявленных ошибок	Согласованность с Техническим Заданием Сбор отзывов участников эксплуатации	Июнь 2024г.	Исполнитель
Разработка Курсового Проекта	Разработка Курсового Проекта, содержащего информацию о проекте на основе Технического Задания, отчет о проделанной работе	С начала утверждения Технического Задания и до защиты проекта	Июнь 2024г.	Исполнитель

5 Порядок контроля и приемки приложения

Контроль разработки приложения осуществляется путем обсуждения текущего этапа разработки среди исполнителей, а также путем встреч с заказчиком.

Готовое приложение со всей необходимой документацией предоставляется заказчику в обозначенный им срок.

За заказчиком остается осуществление проверки соответствия функциональности приложения согласно данному документу и приемки приложения.

Исполнитель также обязан предоставить заказчику следующий комплект поставки необходимый для защиты проекта:

- Техническое Задание
- Курсовой Проект
- Демонстрационная версия приложения со всеми основными сценариями
- Исходный код приложения
- Аналитику проекта
- Видео-презентацию работы Приложения
- Презентация по Техническому Заданию

6 Требования к составу содержания работ по вводу системы в эксплуатацию

При вводе системы в эксплуатацию необходимо выполнить следующий перечень работ:

Осуществить набор персонала в лице:

— минимум одного главного администратора системы

— тестировщика основных сценариев приложения

Обеспечить возможность эксплуатации приложения на любом мобильном устройстве с операционной системой Android версии 8.0 и выше, IOS версии 11.0 и выше.

7 Требования к документированию

Документирование системы ведется в рамках настоящего Технического Задания, составленного в соответствии с ГОСТ 34.602-2020.

Документ об отчете проделанной работы над системой будет представлен в рамках Курсового Проекта на основе настоящего Технического Задания.

Документирование основных сценариев работы приложения осуществляется в сервисе Miro.

Документирование серверной части приложения обеспечивается с помощью инструмента Swagger.

8 Источники разработки

1. Документация по использованию Spring boot [Электронный ресурс]. – Режим доступа: <http://spring-projects.ru/> - Заглавие с экрана. – (Дата обращения: 19.02.2023).
2. Документация по использованию PostgreSQL 14[Электронный ресурс]. – Режим доступа: <https://www.postgresql.org/docs/14/index.html> - Заглавие с экрана. – (Дата обращения: 19.02.2023).
3. Документация по использованию Docker [Электронный ресурс]. – Режим доступа: <https://wiki.dieg.info/docker> - Заглавие с экрана. – (Дата обращения: 19.02.2023).
4. Документация по использованию Android Studio[Электронный ресурс]. – Режим доступа: [Электронный ресурс]. – Режим доступа: https://translated.turbopages.org/proxy_u/en-ru.ru.d67149fc-641b3b64-d5fd4077-74722d776562/https/www.geeksforgeeks.org/a-complete-guide-to-learn-android-studio-for-app-development/ - Заглавие с экрана. – (Дата обращения: 19.02.2023).

СОСТАВИЛИ

Наименование организации, предприятия	Должность исполнителя	Фамилия, имя, отчество	Подпись	Дата
Воронежский Государственный Университет	Студент	Таранцов Иван Юрьевич		24.03.2024
Воронежский Государственный Университет	Студент	Тернавский Даниил Олегович		24.03.2024
Воронежский Государственный Университет	Студент	Путилин Максим Дмитриевич		24.03.2024

СОГЛАСОВАНО

Наименование организации, предприятия	Должность	Фамилия, имя, отчество	Подпись	Дата
Воронежский Государственный Университет	Старший преподаватель	Тарасов Вячеслав Сергеевич		24.03.2024