**МИНОБРНАУКИ РОССИИ**

**федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

**«Балтийский государственный технический университет «ВОЕНМЕХ» им. Д.Ф. Устинова»**

**(БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова)**

**ОТЧЕТ**

**по курсовой работе**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Тема** | | Разработка минимально жизнеспособного продукта (MVP) | | | | | | | | |
| «Разработка фитнес приложения для мобильных устройств» | | | | | | | | | | |
|  | | | | | | | | | | |
| **Обучающегося группы** | | | О715Б | | | |  | Комаров Никита Валерьевич | | |
|  | | | группа | | | |  | Фамилия и инициалы | | |
| **Направление подготовки / специальность** | | | | 09.03.04 | | | | |  | Программная инженерия |
|  | | | | индекс | | | | |  | полное наименование направления подготовки / специальности |
|  | | | | | | | | | | |
| **Направленность**  **образовательной программы** | | | | | **Разработка программно-информационных систем** | | | | |
|  | | | | | профиль / специализация / магистерская программа | | | | |
|  | | | | | | | | | | |
| **Дисциплина (модуль)** | | | | | Управление программными проектами | | | | | |
|  | | | | | | | | | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | |  | | | | | | | |  | | **Руководитель:** | | | |  | | | | | | | | | | | |
|  | | |  | | | | | | | |  | |  | | | | подпись | | | | | | | | | | | |
|  | | | |  |  | | | | | | |  | |  | | | |  |  | | | | | | | | |
|  | | | | |  | | | | | | |  | | ученая степень, ученое звание | | | | | | | Фамилия ИО | | | | | | | | |
|  |  |  | | | |  |  |  |  |  | |  | | **Оценка:** | |  | | | | | | | | | | | | | |
|  |  |  | | | |  |  |  |  |  | |  | | « » |  |  | | | | | | |  | 20 |  | г. |  | | |
|  | | |  | | | | | | | |  | | **Обучающийся:** | | | |  | | | | | | | | | | | |
|  | | |  | | | | | | | |  | |  | | | | подпись | | | | | | | | | | | |
|  | | | |  |  | | | | | | |  | |  | | | | | |  | |  | | | | | | | |
|  | | | | |  | | | | | | |  | |  | | | | | | | Фамилия ИО | | | | | | | | |
|  |  |  | | | |  |  |  |  |  | |  | | « » |  |  | | | | | | |  | 20 |  | г. |  | | |

САНКТ-ПЕТЕРБУРГ

2025 г.

Содержание

[ВВЕДЕНИЕ 3](#_Toc196157939)

# ВВЕДЕНИЕ

С каждым днем поддерживать здоровый образ жизни становится сложнее. Загрязненный воздух, несбалансированное питание, вредные привычки, стрессовые ситуации, малая активность – факторы, которые влияют на каждого человека и пагубно сказываются на его здоровье.

Оздоровление происходит разными способами, наиболее естественный занятия спортом. Фитнес помогает нормализовать большую часть жизненных процессов в организме даже при минимальных усилиях.

В данной работе объектом исследования является фитнес-приложение как средство улучшения физического здоровья. Предметом исследования являются технологии разработки мобильного фитнес-приложения.

Целью данной курсовой работы является разработка и создание фитнес-приложения для мобильных устройств, которое будет способствовать улучшению физического развития.

Для достижения поставленной цели необходимо решить следующие задачи:

* произвести анализ предметной области;
* рассмотреть аналоги;
* сформулировать основные функциональные требования к приложению;
* спроектировать базу данных;
* спроектировать приложение;
* осуществить тестирование приложения.

В результате выполнения задач должно получиться готовое фитнес приложение.

# 1 Обзор предметной области

## Анализ предметной области

Фитнес – это систематическая физическая активность, которая способствует развитию и поддержанию физической формы организма. Он подразумевает под собой разные виды тренировок, правильное питание и восстановление.

Фитнес включает в себя:

* кардио-тренировки, к таким тренировкам можно отнести бег, езду на велосипеде, плавание;
* силовые тренировки, к таким тренировкам можно отнести упражнения с гантелями, подтягивания, отжимания;
* групповые тренировки, чаще всего это могут быть аэробика, йога, зумба;
* функциональный тренинг, включает в себя использование свободных весов и упражнения на равновесие;
* правильное питание, питание должно быть сбалансированно с соблюдением количества белков, жиров и углеводов;
* восстановление, отдых и сон, неотъемлемые составляющие восстановления организма.

## 1.2 Основные преимущества React Native

React Native (также известный как RN) – это популярная платформа мобильных приложений на основе JavaScript, которая позволяет создавать мобильные приложения с собственным интерфейсов для IOS и Android.

При выборе языков программирования для разработки мобильного приложения можно выделить несколько следующих преимуществ React перед другими:

* общая кодовая база, большая часть кода приложения под IOS и Android одинаковая, что приводит к минимизации количества багов и ошибок в ходе разработки;
* максимально похож на нативный, в отличие от других кроссплатформенных решений, которые имитируют среду браузера, React Native использует нативные API;
* большое комьюнити, возможность использовать репозиторий, в котором можно получить помощь, использовать различные библиотеки, которые упрощают разработку;
* поддержка TypeScript, статическая типизация позволяет сократить количество багов.

## Аналитика конкурентов

Рассмотрим ряд популярных фитнес приложений.

### 1.3.1 PUMATRACK

Одним из ключевых конкурентов является мобильное приложение для занятий спортом от компании Puma. Само приложение представлено на рисунке 1.

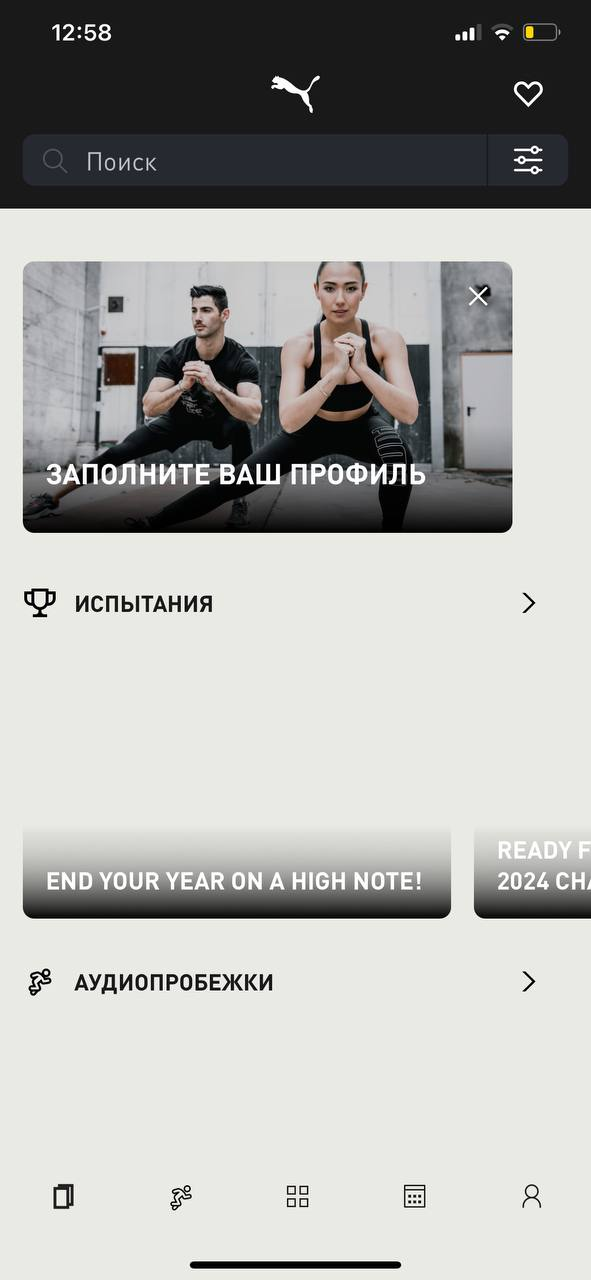


Рисунок 1 – Главная страница приложения

На главной странице присутствует поисковая строка для удобства нахождения нужной информации, возможность попробовать себя в спортивных испытаниях, выбрать понравившейся аудио для пробежки и не только, выбрать тренировки разной продолжительности или под руководством тренера.

На рисунке 2 представлена вкладка приложения для пробежки.



Рисунок 2 – Вкладка для пробежки

В данной вкладке указан вспомогательный функционал для комфортной пробежки, где есть поддержка местоположения, секундомер, поддержка аудио сопровождения. Помимо этого, в приложении находится новостной раздел и таблица рекордов, где пользователи могут делиться своими результатами и достижениями.

При анализе приложения PUMATRAC можно выделить следующие преимущества: возможность выбора тренировки, как с тренером, так и без него, разной продолжительности, удобный функционал для пробежки, планировщик, новостная лента с достижениями, возможность следить и соревноваться с другими пользователями, и отслеживать свой прогресс.

Из минусов можно выделить: нет возможности выбрать тренировку с определенными упражнениями, все тренировки на английском языке, тренировки с тренером без перевода, нет возможности выбрать короткие тренировки, в среднем все тренировки от 30 минут и более.

Целевая аудитория – англоязычные носители, люди, которые любят пробежки, марафоны.

### 1.3.2 Seven

Seven (https://seven.app) приложение для персональных тренировок в домашних условиях. Оно предлагает большой ассортимент тренировок для пользователей в зависимости от текущего физического состояния. Целевая аудитория – люди, которые хотят поддерживать форму в домашних условиях.

На рисунке 3 представлена вкладка планировщика.

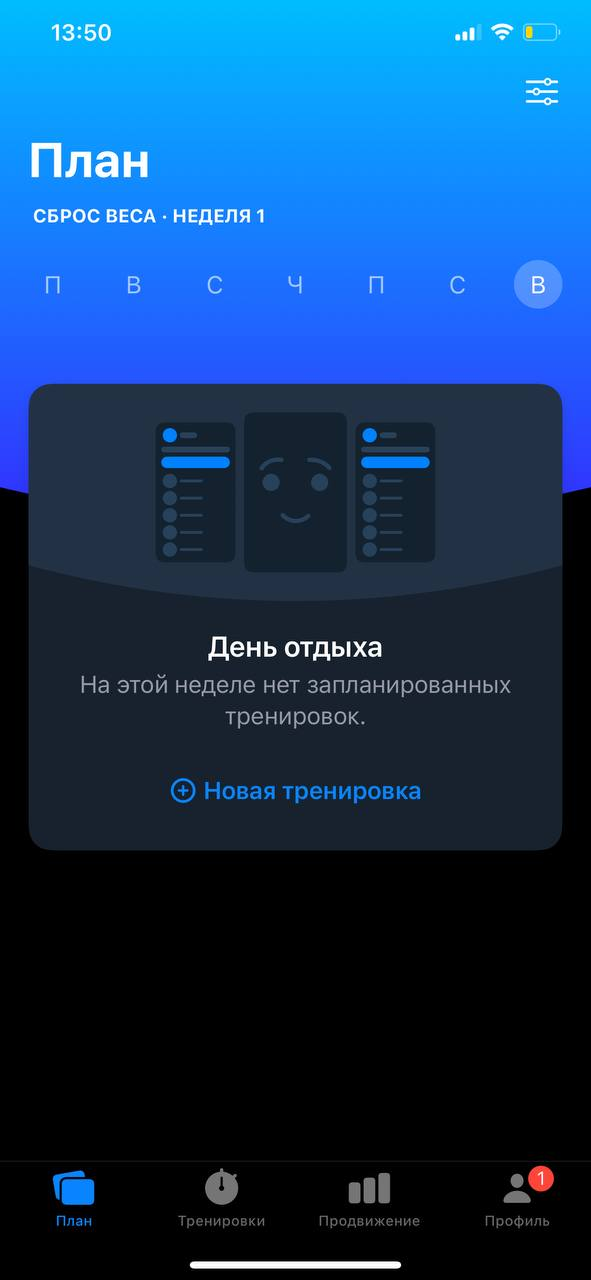


Рисунок 3 – Вкладка с планированием

На данной вкладке представлен план на неделю, указана цель занятий и продолжительность ваших занятий.

На рисунке 4 представлена вкладка с тренировками.

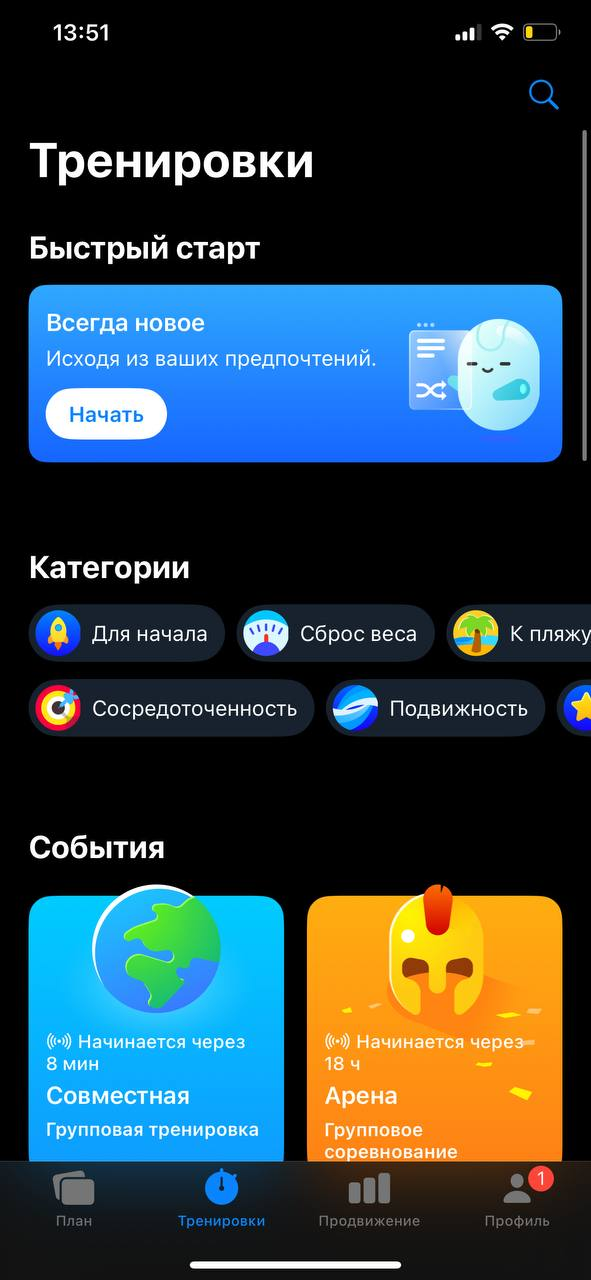


Рисунок 4 – Вкладка с тренировками

В данной вкладке есть большой выбор тренировок. Возможность создать собственную тренировку, выбрать тренировку только на определенную часть тела, разные программы, направленные на то, что вам необходимо, похудеть или набрать массу, стать более гибким или увеличить выносливость. Ежедневные события, как в виде групповых тренировок, так и в виде соревнований.

При просмотре приложения Seven можно выделить следующие преимущества: при запуске приложения опрос для формирования вашего персонального плана тренировок, большой выбор тренировок разного уровня сложности, упражнения не требуют сложного и дорогого оборудования, система состязаний и достижений, создание собственной тренировки, работа без интернета.

К некоторым минусам можно отнести: платная подписка, стоимость подписки выше средней среди фитнес-приложений, нет доступа ко всему функционалу без авторизации, однообразный контент тренировок, нет трекера питания, статистика сводится больше к регулярности занятий и количеству выполненных тренировок.

На основании вышеизложенного, Seven это приложение, имеющие большой выбор тренировок для всех нужд с возможностью создавать персональные тренировочные планы. Однако для пользователей, которые стремятся к более детальному анализу прогресса и продолжительному погружению в тренировочный процесс приложение может оказаться неподходящим.

### Fitness online

Fitness online (https://fitnessonline.app/ru) приложения для тренировок в домашних условиях и в зале. Целевая аудитория – люди, которые хотят заниматься спортом, с оборудованием или без.

На рисунке 5 представлена вкладка с тренировками.

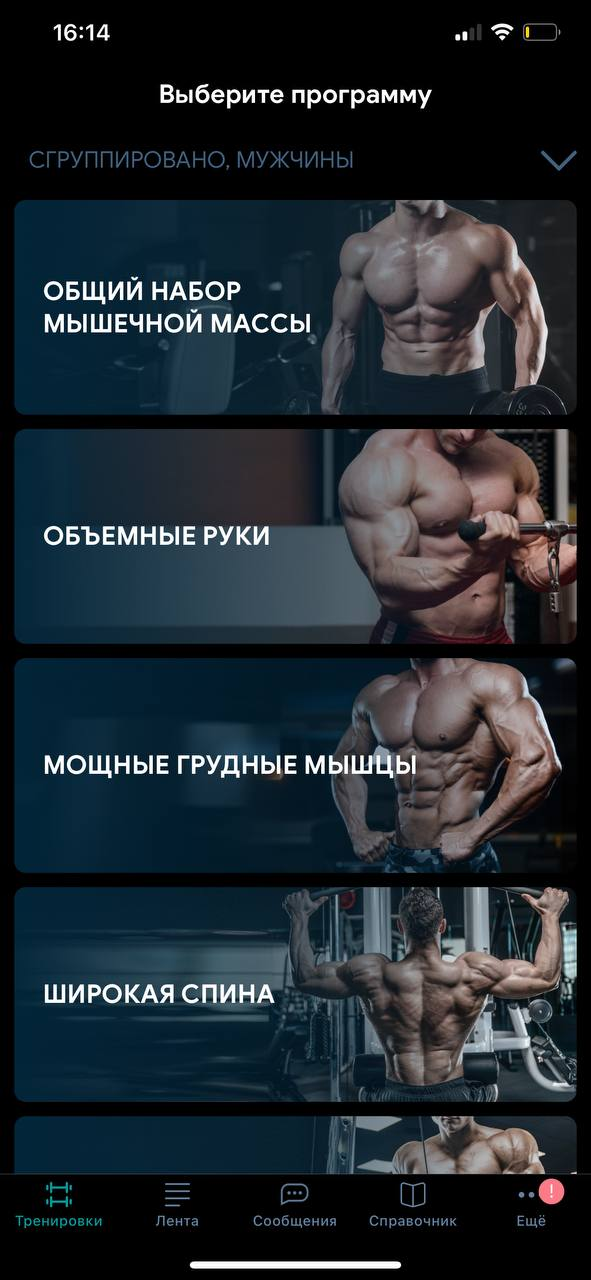


Рисунок 5 – Вкладка с тренировками

На вкладке с тренировками представлены разные программы, направленные на определенную группу мышц. Вкладка новостной ленты дает возможность общаться посредством публикаций в самом приложении, делить своими результатами, мнениями, смотреть дополнительную информацию у других пользователей.

На рисунке 6 представлена вкладка со справочной информацией.



Рисунок 6 – Справочная информация

В данной вкладке содержатся вся справочная информация про тренировки, питание, дополнительные исследования в сфере здоровья и спорта.

Во вкладке «Еще» можно посмотреть аналитику, добавить замеры тела, купить подписку, вступить в сообщества, просмотреть свой тренерский штаб. В случае происшествия, можно обратиться в техническую поддержку или попытаться настроить приложение.

Приложение Fitness Online включает в себя: выбор тренировок на разные группы мышц, возможность общаться внутри приложения, делиться своими результатами и смотреть их у других пользователей, большая справочная информация по ведению здорового образа жизни, возможность заниматься с тренером в самом приложении.

В этом приложении можно выделить следующие минусы: нет трекера питания, платная подписка, нет планировщика, некоторые разделы остаются пустыми, без регистрации и подписки функционал сильно урезан.

На основании вышеизложенного, Fitness online приложение для пользователей разной подготовки от новичка для профессионала для занятий в домашних условиях и в тренажерном зале, где можно посмотреть всю необходимую информацию и обсудить ее. Однако данное приложение не предоставляет возможности распланировать тренировки и питание в визуальном формате.

### 1.3.4Сравнение конкурентов

Таблица 1 – Результаты сравнительного анализа

|  |  |  |  |
| --- | --- | --- | --- |
| Аналог | PUMATRAC | Seven | Fitness online |
| Доступность | IOS/Android | IOS/Android | IOS/Android |
| Язык интерфейса | Русский | Русский | Русский |
| Целевая аудитория | Новички и любители | Люди, предпочитающие короткие тренировки | Пользователи разного уровня подготовки |
| Типы тренировок | Силовые, кардио, йога, с оборудованием и без, бег | Простые небольшие тренировки | Силовые, кардио, с оборудованием и без |
| Офлайн-режим | Есть | Есть | Есть |
| Поддержка языков | В основном английский | Поддержка русского языка и других | Поддержка русского языка |
| Трекер питания | Нет | Нет | Нет |

Продолжение таблицы 1.

|  |  |  |  |
| --- | --- | --- | --- |
| Платная подписка | Есть | Есть | Есть |
| Персонализация | Адаптация под цели | Базовая персонализация | Программы для дома и зала |
| Разнообразие упражнений | Широкий выбор | Ограниченный набор упражнений | Широкий выбор |
| Справочная информация | Нет | Нет | Есть |

PUMATRAC подходит для пользователей, которые больше увлечены пробежками и не любят однообразные и самостоятельные тренировки. Seven подходит для тех, у кого мало времени для занятий спортом. Fitness online подходит для пользователей, которые углубленно хотят погрузиться в тему тренировок и питания.

Исходя из анализа предметной области и сравнительного анализа популярных аналогов были выделены следующие критерии:

* возможность просмотра библиотеки тренировок для дома и зала не менее 50 упражнений;
* возможность выбора плана тренировок для новичков, любителей и профессионалов;
* возможность просматривать статистику по окончанию тренировки (сожженные калории, время выполнения тренировки);
* возможность использовать трекер питания, готовые планы питания;
* возможность просматривать раздел с обучающими материалами по технике выполнения упражнений.

# 2 Моделирование глобальных задач и подзадач проекта

## 2.1 Распределение задач проекта, распределение ресурсов

Для моделирования глобальных задач и подзадач проекта будет использоваться программное обеспечение ProjectLibre.

Процесс создания проекта изображен на рисунке 7.

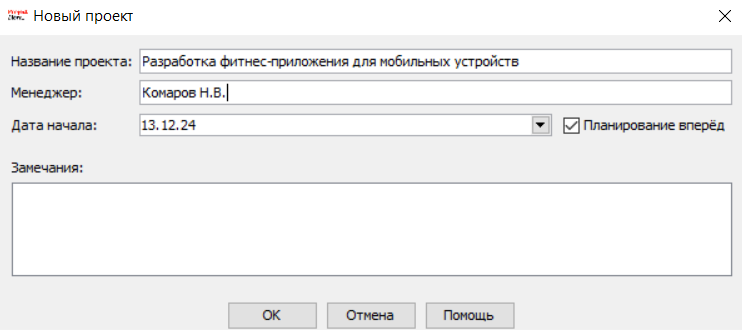


Рисунок 7 – Создание проекта

Далее, в соответствии с иерархической структурой работ, были выделены задачи, разделенные на фазы, в каждой из которых были обозначены вехи, созданы связи между задачами. На рисунке 8 отображены задачи проекта.

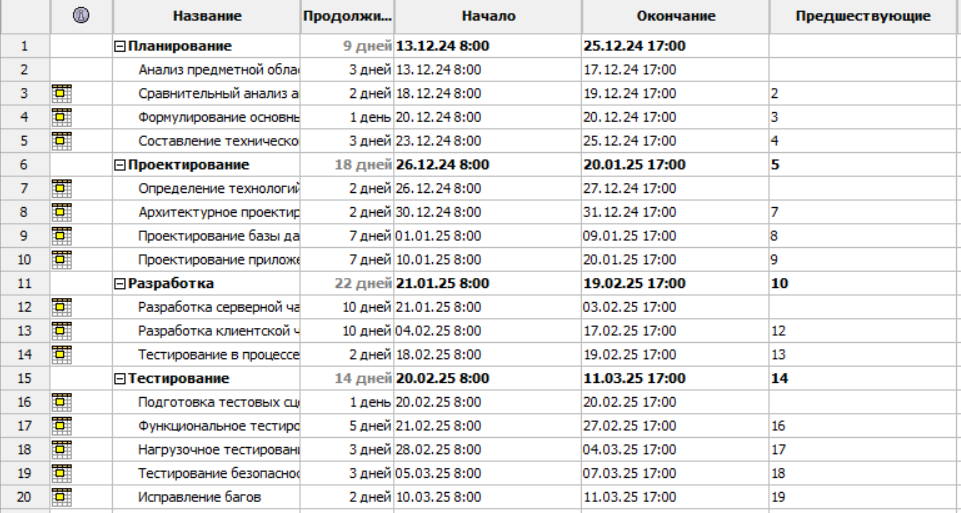


Рисунок 8 – Задачи проекта

Диаграмма Ганта представлена на рисунке 9.

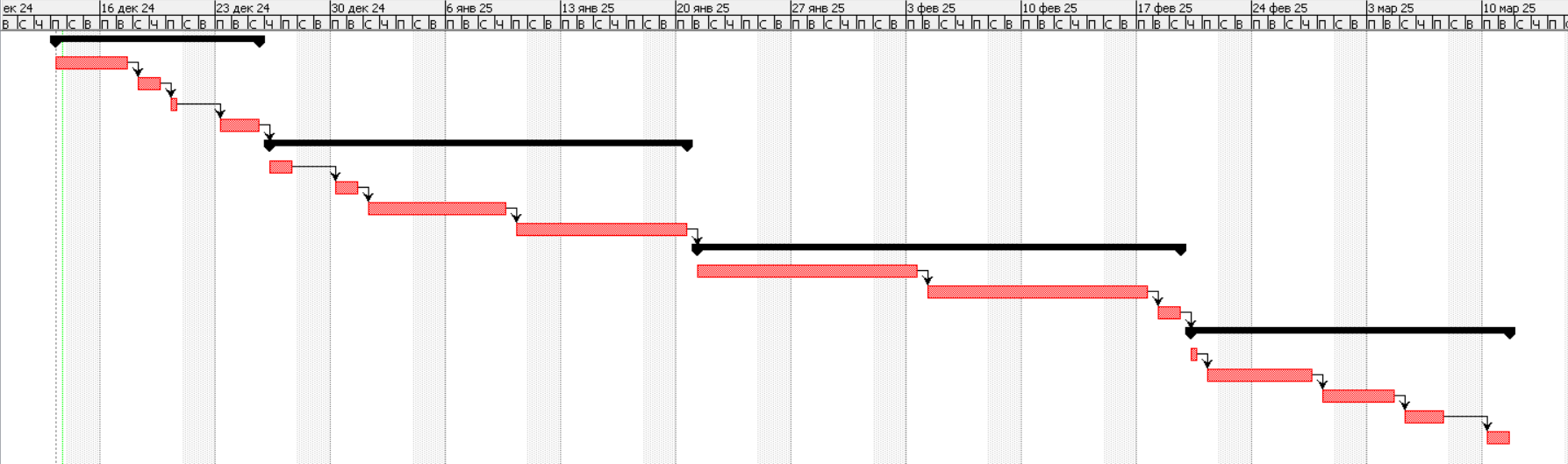


Рисунок 9 – Диаграмма Ганта

Далее необходимо описать и распределить доступные ресурсы. Список использованных ресурсов представлен на рисунке 10.

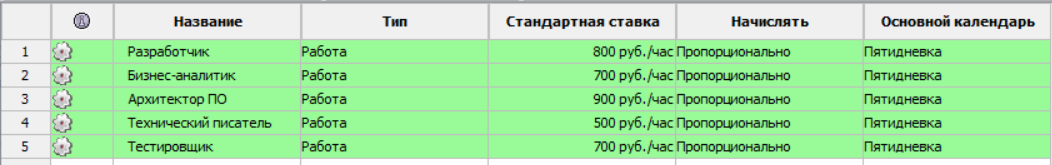


Рисунок 10 – Список ресурсов

На рисунке 11 изображено распределение ресурсов по задачам.

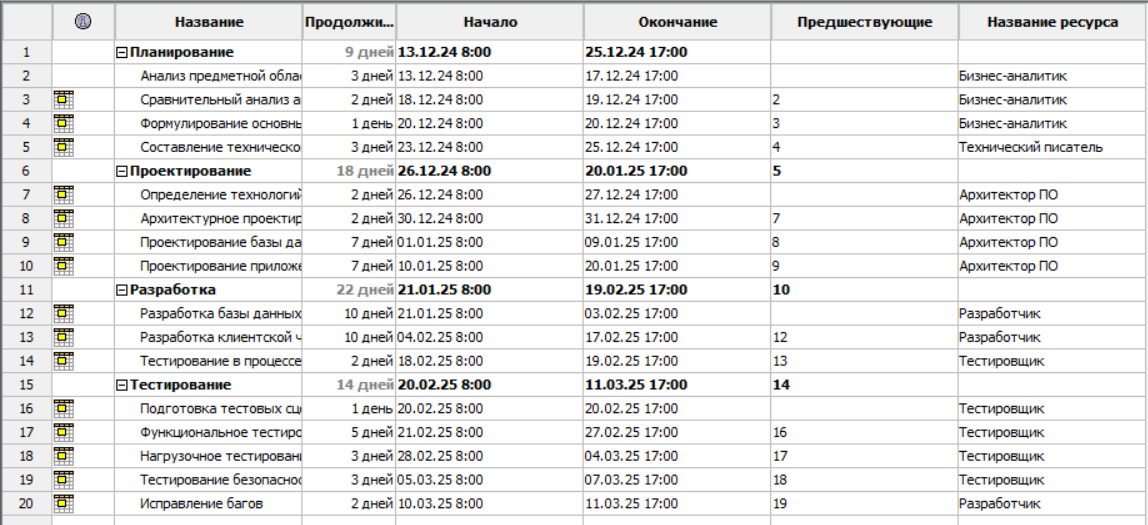


Рисунок 11 – Распределение ресурсов

На рисунке 12 представлена диаграмма Ганта с названиями используемых ресурсов.

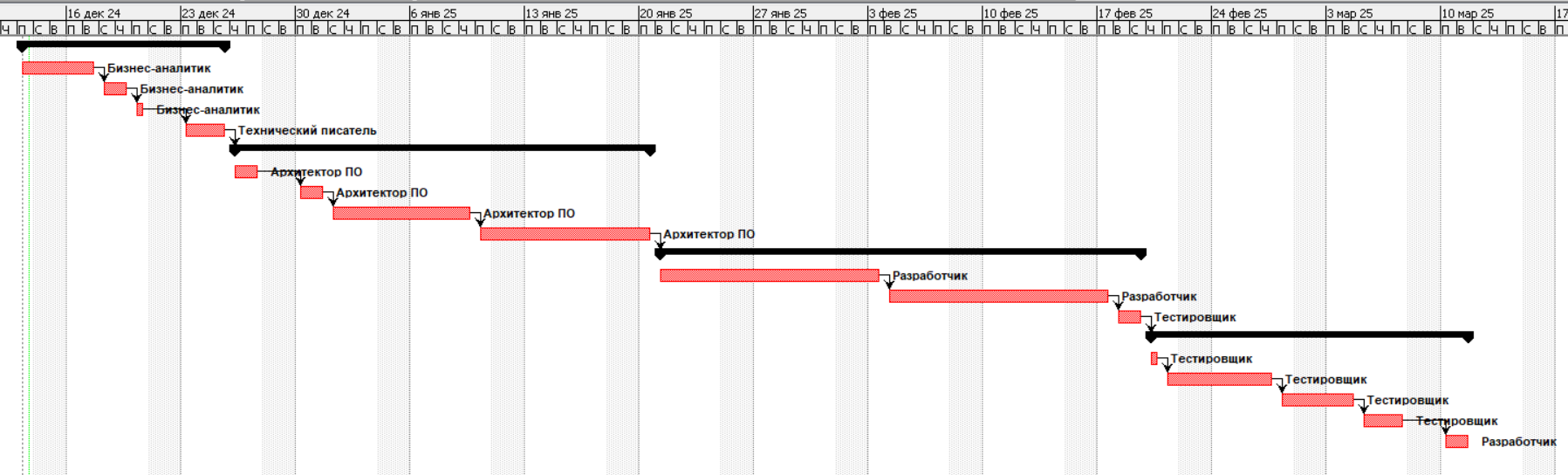


Рисунок 12 – Итоговая диаграмма Ганта

На рисунке 13 отображена таблица, где представлены уровни доступности каждого из ресурсов.

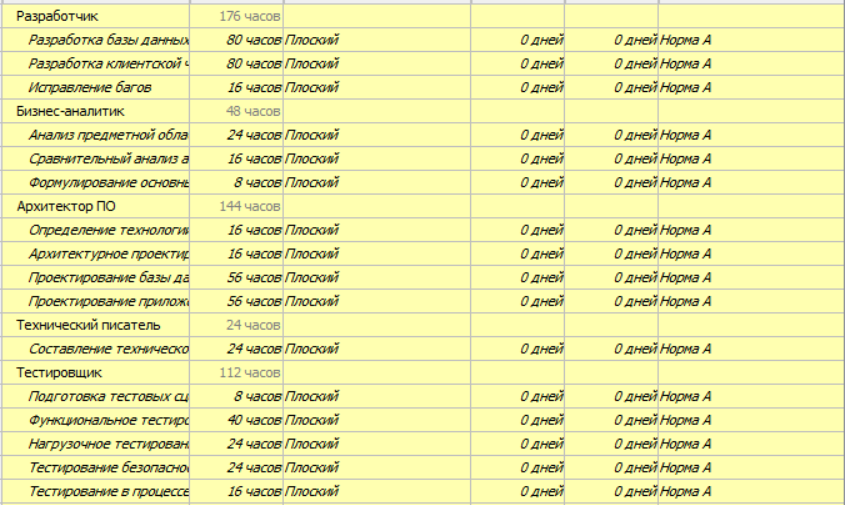


Рисунок 13 – Уровни доступности ресурсов

Итоговый отчет проекта отображен на рисунке 14.

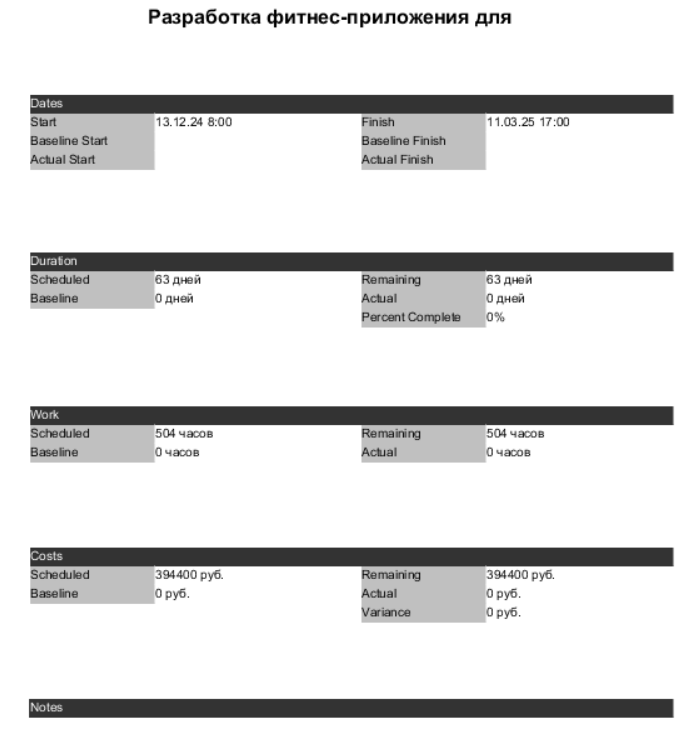


Рисунок 14 – Итоговый отчет проекта

## 2.3 Оценка рисков и угроз

Исходя из созданного во втором разделе проекта, были выделены следующие возможные риски, которые представлены в таблице 2.

Таблица 2 – Соотношение целей и рисков программного проекта

|  |  |  |
| --- | --- | --- |
| Этапы ЖЦ | Цели | Риски |
| Планирование | Анализ предметной области, сравнительный анализ аналогов, формулирование функциональных требований, составление ТЗ | Неполный или некорректный анализ требований, недооценка сроков и бюджета, недостаточная детализация ТЗ |
| Проектирование | Выбор технологий для реализации, выбор архитектуры приложения, проектирование БД, проектирование приложения | Низкая квалификация команды в выбранных технологиях, недостаточная гибкость архитектуры для дальнейших изменений |
| Реализация | Реализация БД, реализация приложения, тестирование в процессе разработки | Недостаточная функциональность модулей, задержки из-за нехватки ресурсов |
| Тестирование | Подготовка тестовых сценариев, функциональное тестирование, тестирование безопасности, нагрузочное тестирование, исправление багов | Нахождение критических багов, недостаточное покрытие тестами, необнаруженные уязвимости |

После определения рисков можно построить матрицу рисков. Матрица рисков представлена в таблице 3.

Таблица 3 – Матрица рисков

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| С  те  п  е  н  ь  у  щ  ер  ба | к  р  и  т  и  че  ск  ая |  | Необнаруженные уязвимости | Нахождение критических багов, неполный или некорректный анализ требований | Недостаточное покрытие тестами | Недооценка сроков и бюджета |
| о  че  н  ь  в  ы  со  ка  я |  | Низкая квалификация команды в выбранных технологиях | Недостаточная гибкость архитектуры для дальнейших изменений |  |  |
| в  ы  со  ка  я |  |  | Недостаточная функциональность модулей |  |  |
| ср  ед  н  яя |  |  | Недостаточная детализация ТЗ | Задержки из-за нехватки ресурсов |  |
| н  из  ка  я |  |  |  |  |  |
|  |  | 0-20% | 21-40% | 41-60% | 61-80% | 81-100% |

Зеленый цвет указывает на низкий уровень риска. Желтый цвет сигнализирует, что риск нуждается в внимании. Красный цвет обозначает высокую вероятность риска, который может привести к значительным последствиям.

После определения шкал оценки последствий рисков формируется таблица реестра рисков, в которой устанавливаются первопричина возможного риска, необходимое условие для появления этого риска, а также последствия в случае возникновения риска. В таблице 4 представлен реестр рисков.

Таблица 4 – Реестр рисков

|  |  |  |
| --- | --- | --- |
| Риск | Условие | Последствие |
| Неполный или некорректный анализ требований | Отсутствие понимания требований заказчика | Разработка функций, не соответствующих потребностям пользователей, необходимость внесения изменений в поздних этапах проекта |
| Недооценка сроков и бюджета | Недостаточная детализация планирования | Задержки и перерасход бюджета |
| Недостаточная детализация ТЗ | Ограниченное время на составление технического задания или недостаточное понимание требований | Сложности с реализацией и интеграцией, нарушение сроков проекта |
| Низкая квалификация команды в выбранных технологиях | Команда недостаточно знакома с выбранными технологиями или инструментами | Ошибки в реализации, снижение качества кода, возможные задержки |
| Недостаточная гибкость архитектуры для дальнейших изменений | Архитектура системы не позволяет легко добавлять новые функции или масштабировать проект | Затраты на переработку архитектуры, нарушение сроков внедрения новых функций |
| Недостаточная функциональность модулей | Команда не учла все функциональные требования на этапе проектирования | Необходимость доработки модулей в процессе или после завершения проекта, нарушение сроков и бюджета |
| Задержки из-за нехватки ресурсов | Неверная оценка трудозатрат, нехватка специалистов или технических ресурсов. | Затягивание сроков проекта, сдвиг этапов и нарушение план-графика выполнения задач |
| Нехождение критических багов | Большое количество ошибок или критические дефекты | Ошибки в работе приложения в реальной эксплуатации, отказ в работе приложения |
| Недостаточное покрытие тестами | Нехватка времени или тестирования, низкий приоритет тестирования | Проблемы в работе приложения, недовольство пользователей |
| Необнаруженные уязвимости | Неполное тестирование безопасности, недостаток экспертизы в области кибербезопасности | Утечка данных, компрометация системы, юридические последствия |

# 3 Проектирование

## 3.1 Построение иерархической структуры работ

Для построения иерархической структуры (WBS) проекта был выбран подход по элементам жизненного цикла.

В этом подходе каждый уровень иерархии WBS представляет собой последовательные этапы реализации проекта, начиная с его планирования и заканчивая тестированием готового продукта. Такой подход позволяет организовать работы по проекту в соответствии с временной последовательностью их выполнения и логической структурой этапов.

На рисунке 15 представлен WBS на основе этапов жизненного цикла для проекта по разработке фитнес-приложения.

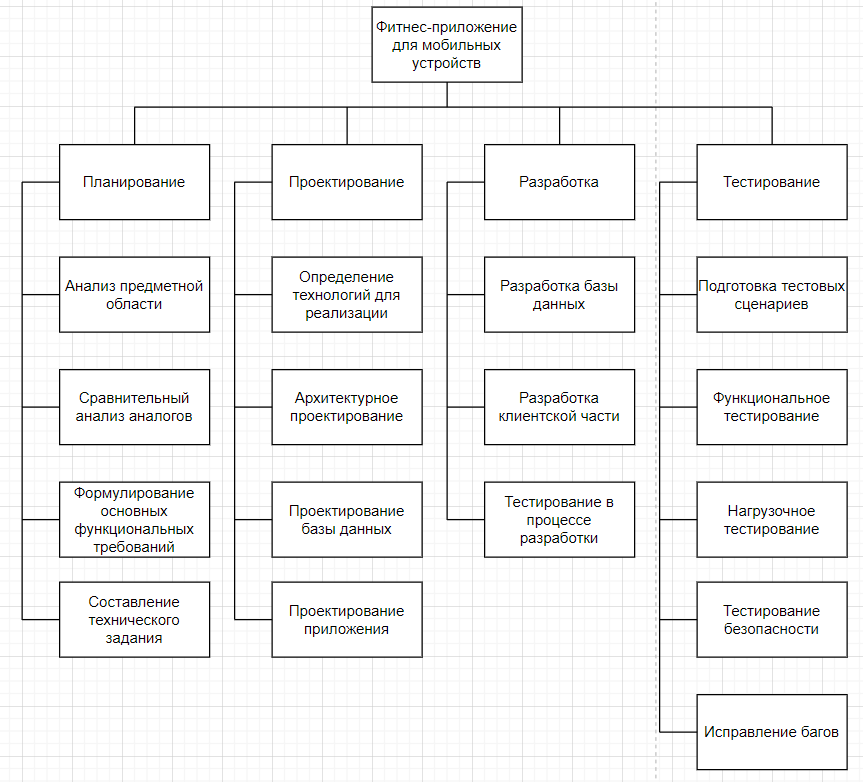


Рисунок 15 – WBS по этапам жизненного цикла

## 3.2 Построение UML диаграмм

### 3.2.1 Построение диаграммы использования

Диаграмма вариантов использования (прецендентов) играет основную роль в моделировании поведения системы, подсистемы или класса. Она показывает множество прецендентов, акторов и отношения между ними. Такие диаграммы используются для представления системы с точки зрения вариантов использования. Чаще всего это предполагает моделирование контекста системы, подсистемы или класса, либо моделирование требований к поведению указанных элементов.

На рисунке 16 представлена диаграмма вариантов использования.

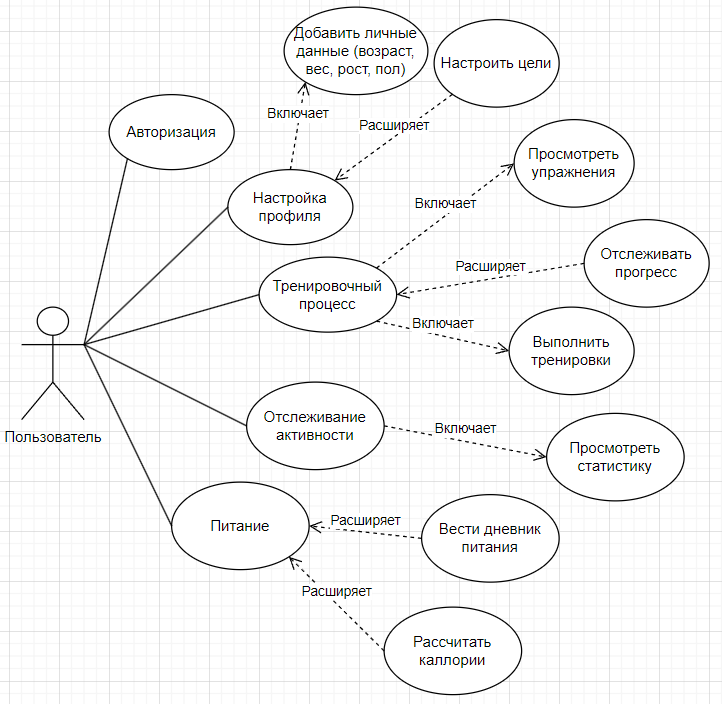


Рисунок 16 – Диаграмма вариантов использования

### 3.2.2 Построение диаграммы последовательностей

На диаграмме последовательности объекты изображаются прямоугольниками на вершине вертикальной пунктирной линии. Эта вертикальная линия называется линией жизни (lifeline) объекта. Она представляет собой жизненный цикл объекта в процессе взаимодействия. Такая форма была впервые предложена А. Джекобсоном. Каждое сообщение представляется стрелкой между линиями жизни двух объектов. Порядок следования сообщений устанавливается сверху вниз, т.е. так, как они показываются на диаграмме.

Диаграмма последовательностей представлена на рисунке 17.

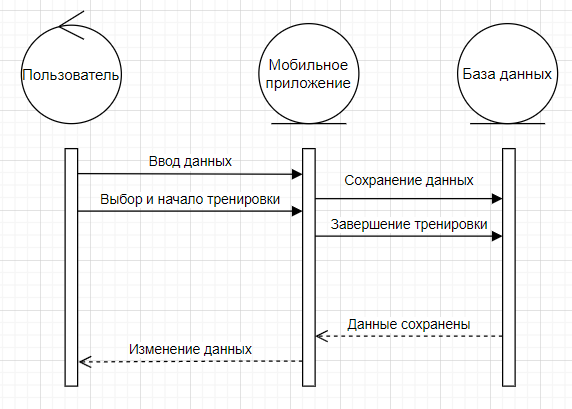


Рисунок 17 – Диаграмма последовательностей

### 3.2.3 Построение диаграммы классов

Диаграмма классов описывает типы объектов системы и различного рода статические отношения, которые существуют между ними. Имеется два основных вида статических отношений: ассоциации и подтипы. На диаграммах классов изображаются атрибуты классов, операции классов и ограничения, которые накладываются на связи между объектами.

На рисунке 18 отображена диаграмма классов.

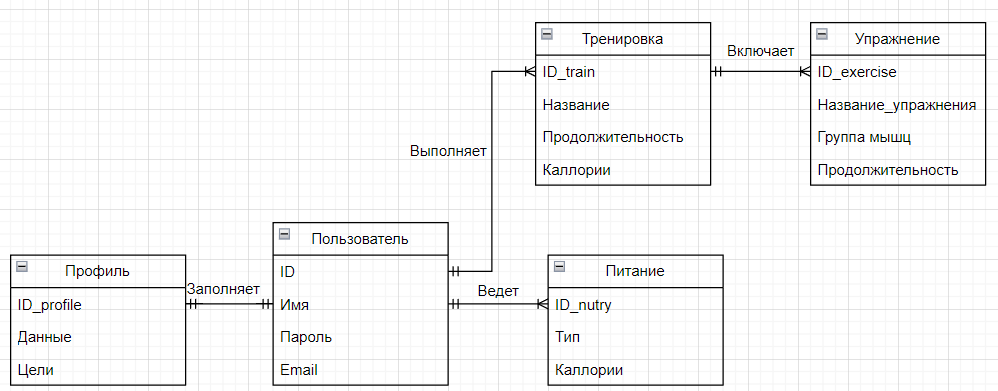


Рисунок 18 – Диаграмма классов

### 3.2.4 Построение диаграммы состояний

Диаграмма состояний является хорошо известным методом описания поведения систем. Она изображает все возможные состояния, в которых может находиться конкретный объект, а также изменения состояния объекта, происходящие в результате влияния некоторых событий на этот объект.

На рисунке 19 отображена диаграмма состояний.

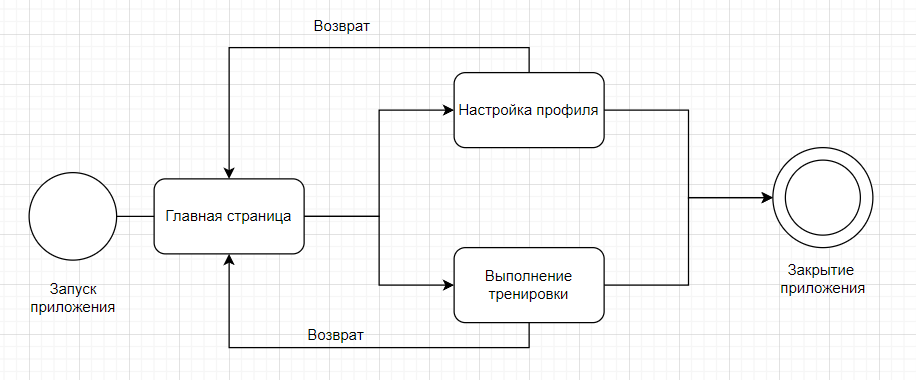


Рисунок 19 – Диаграмма состояний

## 3.3 Разработка прототипа продукта на основе анализа рынка

Исходя из проведенного анализа конкурентов на рынке, можно выделить следующие функциональные требования для фитнес-приложения:

* наличие авторизации и регистрации;
* создание и редактирование профиля;
* наличие базы упражнений с описанием и GIF-демонстрацией;
* наличие готовых программ тренировок для новичков, любителей, профессионалов;
* возможность создания индивидуальных тренировок;
* наличие дневника питания;
* наличие статистики прогресса;
* наличие уведомлений о тренировках;
* наличие обучающих материалов.

Приоритизорованные функции с использованием MoSCoW (Must have, Should have, Could have, Won’t have):

1. Must have:

* наличие регистрации;
* наличие авторизации;
* наличие базы упражнений с описанием и GIF-демонстрацией.

1. Should have:

* создание и редактирование профиля;
* наличие готовых программ тренировок для новичков, любителей, профессионалов;
* наличие статистики прогресса.

1. Could have:

* возможность создания индивидуальных тренировок;
* наличие дневника питания.

1. Won’t have:

* наличие уведомлений о тренировках.
* наличие обучающих материалов.

### 3.3.1 Прототип фитнес приложения

# 4 Анализ моделей ЖЦ

## 4.1 Характеристики модели ЖЦ в зависимости от процесса выявления требований

* являются ли проблемы предметной области проекта новыми для большинства разработчиков? — Нет;
* является ли технология предметной области проекта новой для большинства разработчиков? — Нет;
* являются ли инструменты, используемые проектом, новыми для большинства разработчиков? — Нет;
* изменяются ли роли участников проекта во время жизненного цикла? — Да;
* могут ли разработчики проекта пройти обучение? — Да;
* является ли структура программного продукта (ПП) более значимой для разработчиков, чем гибкость? — Да;
* будет ли менеджер проекта строго отслеживать прогресс команды? — Да;
* важна ли легкость распределения ресурсов? — Да;
* приемлет ли команда равноправные обзоры и инспекции, менеджмент/обзоры заказчика, а также стадии? — Да.

Вывод: для команды лучше подходит инкрементная модель, так как поддерживает обучение и гибкость распределения ролей.

## 4.2 Характеристики модели ЖЦ в зависимости от участия пользователей

* будет ли присутствие пользователей ограничено в жизненном цикле? — Нет;
* будут ли пользователи знакомы с определением системы? — Да;
* будут ли пользователи ознакомлены с проблемами предметной области? — Да;
* будут ли пользователи вовлечены во все фазы жизненного цикла? — Нет;
* будет ли заказчик отслеживать ход выполнения проекта? — Нет.

Вывод: Инкрементная модель подходит, поскольку участие пользователей ограничено, а заказчик не отслеживает все этапы.

## 4.3 Характеристики модели ЖЦ в зависимости от сложности проекта

* будет ли проект идентифицировать новое направление продукта для организации? — Да;
* будет ли проект системно интегрирован? — Да;
* будет ли проект являться расширением существующей системы? — Да;
* будет ли финансирование проекта стабильным на всем протяжении жизненного цикла? — Нет;
* ожидается ли длительная эксплуатация продукта в организации? — Да;
* должна ли быть высокая степень надежности? — Да;
* будет ли система изменяться, возможно, с применением непредвиденных методов на этапе сопровождения? — Да;
* является ли график ограниченным? — Да;
* являются ли «прозрачными» интерфейсные модули? — Да;
* доступны ли повторно используемые компоненты? — Нет;
* являются ли достаточными ресурсы (время, деньги, инструменты, персонал)? — Нет.

Вывод: Инкрементная модель соответствует требованиям гибкости и поэтапного добавления функциональности в условиях ограниченных ресурсов.

На основание анализа характеристик проекта, команды разработчиков и участия пользователей, инкрементная модель жизненного цикла является наиболее подходящей для разработки фитнес-приложения для мобильных устройств. Она предоставляет возможность поэтапного внедрения функционала, гибкость в управлении изменениями и эффективное распределение ресурсов.

# 5 План-график реализации проекта

План-график реализации проекта можно представить в виде диаграммы Ганта. На рисунке 20 отображены график реализации проекта и задачи, которые необходимо будет выполнить для создания приложения.

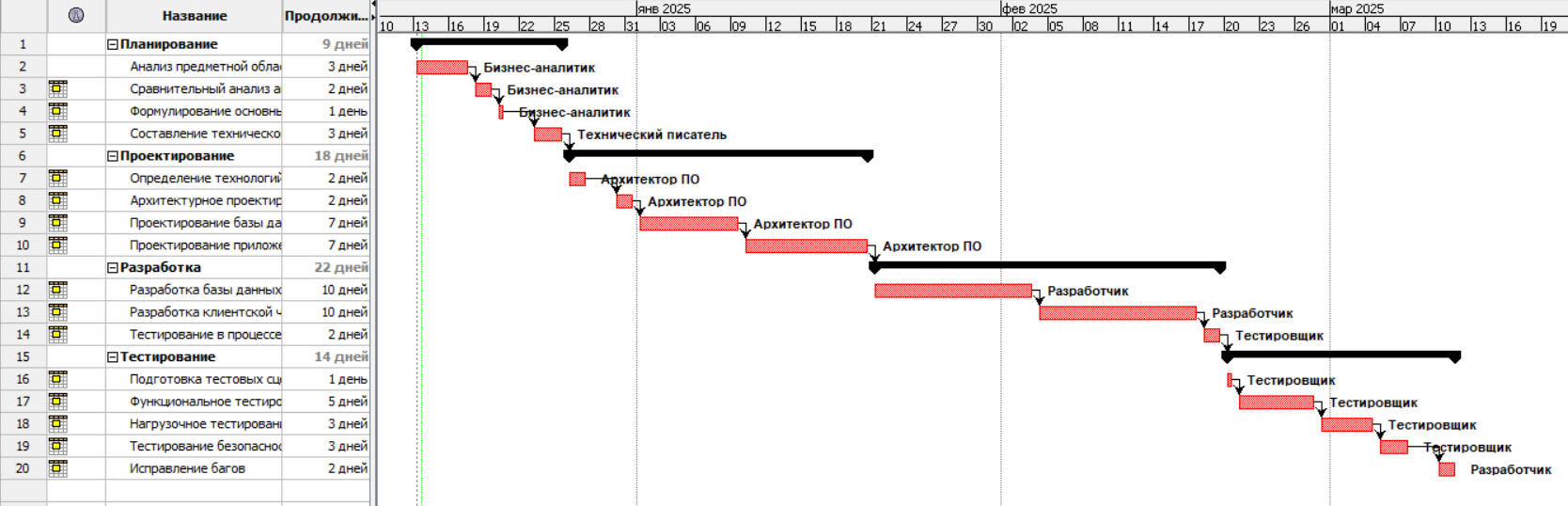


Рисунок 20 – План-график реализации проекта

## 5.1 Расчет финансово-экономических показателей

В таблице 5 представлены затраты на трудовые ресурсы.

Таблица 5 – Затраты на трудовые ресурсы

|  |  |  |  |
| --- | --- | --- | --- |
| Ресурс | Ставка | Часы | Общие затраты |
| Разработчик | 800 | 176 | 140800 |
| Бизнес-аналитик | 700 | 48 | 33600 |
| Архитектор ПО | 900 | 144 | 129600 |
| Технический писатель | 500 | 24 | 12000 |
| Тестировщик | 700 | 112 | 78400 |
| Итого |  |  | 394400 |

Также необходимо рассчитывать материальные ресурсы. Затраты на сервер составят 120000 рублей при условии использования в течение 1 года при стоимости 10000 руб./месяц. Так же стоит учесть затраты, например, на лицензии программного обеспечения. Прочие затраты будут составлять 10000рублей.

Теперь можно рассчитать себестоимость проекта = 394400 + 120000 + 10000 = 524400.

Для расчета срока окупаемости используется следующая формула: Срок окупаемости = Общие затраты / Ежемесячная прибыль. Расчетная ежемесячная прибыль составляет 100000 рублей, исходя из формулы срок, за который проект выйдет в плюс – 6 месяцев.

# 6 Реализация MVP

## 6.1 Внесение изменений в проект

В рамках процесса управления изменениями в разработке фитнес-приложения используется система контроля версий Git, что обеспечивает прозрачное отслеживание всех модификаций проекта. Для каждого нового функционала создаётся отдельная ветка, название которой соответствует вносимым изменениям. Это позволяет изолировать изменения, упростить тестирование и обзор кода.

После завершения разработки функционала ветка проходит процедуру код-ревью и интегрируется в основную ветку через pull request, что снижает вероятность конфликтов и повышает стабильность кодовой базы. Такой процесс помогает оперативно выявлять ошибки и своевременно вносить исправления.

## 6.2 Анализ изменений

### 6.2.1 Описание причины изменения

Первоначальная причина для внесения изменений связана с необходимостью реализации функции расчета количества калорий в день на человека с определенными параметрами по специальной формуле. Изначально не предполагалось использование формулы, но для лучшего понимания нормы пищи в день будет положительно сказываться на приложении.

### 6.2.2 Ожидаемые результаты

Ожидаемым результатом внедрения изменений является создание вкладки с формулой, где необходимо будет указать только 3 входных параметра: рост, вес и возраст, чтобы получить результат. Данный функционал позволит увеличить количество пользователей за счет пользы данной функции для пользователей, особенно новичков.

### 6.2.3 Предполагаемое воздействие на проект

Предполагаемое воздействие изменений на проект включает в себя расширение функционала. В дальнейшем предполагается внедрение ИИ-помощника, который поспособствует более лучшему разбору нормы калорий по формуле, а также сможет подобрать определенный рацион под вашу норму и ваши потребности. Это позволит увеличить количество новых пользователей в приложении за счет легкого использования и освоения информации о рационе на день.

### 6.2.4 Оценка сроков реализации

Оценочно, на реализацию изменений потребуется примерно 2 дня, где:

* 1 день уйдёт на проектирование вкладки с формулой расчета суточной нормы калорийности;
* 1 день займёт реализация функционала.

Данные сроки являются предварительными и могут быть скорректированы в зависимости от сложности интеграции и технических аспектов.

### 6.2.5 Задействованные ресурсы

Для реализации проекта помимо написания самого кода, необходимо провести анализ конкурентов, проверить техническую документацию и дополнительные источники информации, способствующие получению информации для разработки. Спроектировать приложение и походу разработки проверить работоспособность функционала. Каждый этап важен и требует определенного времени.

### 6.2.6 Оценка влияния на смежные задачи

Добавление дополнительной вкладки с формулой расчета калорийности не повлияет на разработку приложения, так как разрабатывается отдельно, и весь материал с ней помещается в отдельную ветку. После тестирования работы, будет добавлена к основному коду.

## 6.3 Внесение изменений

Для интеграции новой функциональности был создан отдельный сервис Formula, отвечающий за обработку запросов от фронтенда и взаимодействие с сервером. Такой подход позволил минимизировать изменения в других модулях, обеспечивая расширение функционала без нарушения работы системы.

На рисунке 34 представлен результат.

Рисунок 34 – Результат

**6.4 Оценка результатов**

В результате проведённой работы удалось успешно интегрировать новую вкладку с формулой расчета калорийности. Реализация была проведена в запланированные сроки, что позволило не отставать от графика разработки.

Работа с изменениями в проекте очень важная составляющая часть. Использование модульного подхода обеспечивает возможность эффективно вносить любые правки и дополнения в проект не нарушая целостность самого проекта.

## 6.5 Описание MVP

# ЗАКЛЮЧЕНИЕ

В результате проделанной работы был разработан проект фитнес-приложения для мобильных устройств.

Был произведен анализ предметной области и рассмотрены аналоги приложения. Проведенный анализ существующих решений позволил выявить их недостатки и определить основные требования к функционалу приложения, что легло в основу его архитектуры.

Также были выбраны технологии, спроектированы и разработаны приложение и база данных для него.

Ожидаемые результаты от реализации проекта:

* привлечение пользователей, заинтересованных в фитнесе и здоровом образе жизни;
* увеличение числа активных пользователей за счет качественного функционала и регулярных обновлений;
* возможность партнерства с брендами;
* быстрая и корректная работа на большинстве популярных платформ;
* защита пользовательских данных;
* интеграция с фитнес-трекерами и умными устройствами.

Таким образом, разработка данного приложения будет способствовать значительному повышению уровня физического здоровья, а также популяризации спорта.

Отчет по работе оформлен в соответствии ГОСТ 7.32-2017.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Мажайцев, Е. А. Управление программными проектами: практическое пособие / Е. А. Мажайцев, Г. С. Устиновский, А. К. Садырова. – Санкт-Петербург: Изд-во БГТУ «ВОЕНМЕХ» им. Д. Ф. Устинова, 2024. – 120 с.
2. ГОСТ 7.32-2017. Система стандартов по информации, библиотечному и издательскому делу. Отчёт о научно-исследовательской работе. Структура и правила оформления. — М.: Стандартинформ, 2018. — 16 с.
3. Документация по GitHub [Электронный ресурс]. — URL: https://docs.github.com/ru/get-started/start-your-journey (дата обращения: 04.12.2024).
4. Документация React. [Электронный ресурс]. URL: https://react.dev (дата обращения 04.12.2024).
5. Руководство по JavaScript [Электронный ресурс]. — URL: https://osintframework.com/ (дата обращения: 03.11.2024).