



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное
учреждение высшего образования

«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ
ГИДРОМЕТЕОРОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра Прикладной информатики

Направление: 09.03.03 Прикладная информатика

Профиль: Прикладные информационные системы и технологии

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
(квалификация – бакалавр)

На тему: «Разработка мобильного приложения для бега с оптимизацией маршрутов»

Исполнитель: Бабкова Валерия Дмитриевна

Руководитель: к.т.н., доцент, Колбина Ольга Николаевна

«К защите допускаю»

Заведующий кафедрой


(подпись)


(фамилия, имя, отчество)

«03» 06 2025 г.

Санкт-Петербург

2025

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	3
ГЛАВА 1. ПРЕДПРОЕКТНЫЙ АНАЛИЗ СИСТЕМЫ	7
1.1 Анализ предметной области. Описание концепции собственной системы	7
1.2 Анализ рынка устройств и приложений	9
1.3 Сравнительный анализ	10
1.4 SWOT-анализ	15
1.5 Требования к информационной системе	18
1.6 Информационные потоки. Методология DFD	20
1.7 Схема бизнес-процессов. Методология BPMN	24
ГЛАВА 2. ПРОЕКТИРОВАНИЕ ИС	31
2.1 Unified Modeling Language	31
2.2 Use case диаграмма	32
2.3 Сценарий	33
2.4 Диаграмма последовательностей	35
2.5 Создание схемы базы данных для реализуемой системы	36
2.6 Диаграмма компонентов	41
2.7 Диаграмма развёртывания	43
2.8 Создание макета приложения на основе анализа системы	44
ГЛАВА 3. РЕАЛИЗАЦИЯ ИНФОРМАЦИОННОЙ СИСТЕМЫ	46
3.1 Технико-экономическое обоснование реализуемого приложения	46
3.1.1 Обоснование целесообразности разработки проекта	46
3.1.2 Оценка уровня качества разрабатываемой информационной системы	48
3.1.3 Планирование работ по разработке информационной системы	50
3.1.4 Расчёт затрат на разработку мобильного приложения	52
3.1.5 Расчёт эксплуатационных затрат и оценка эффективности разрабатываемого продукта	55
3.2 Реализация мобильного приложения	58
ЗАКЛЮЧЕНИЕ	63
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ И ЛИТЕРАТУРЫ	65
ПРИЛОЖЕНИЕ А	68
ПРИЛОЖЕНИЕ Б	77
ПРИЛОЖЕНИЕ В	93
ПРИЛОЖЕНИЕ С	96

ВВЕДЕНИЕ

В современном мире человек и техника уже два неразрывно связанных понятия. На нашей планете проживает около 8 миллиардов людей¹, из них 5,4 имеют при себе мобильные телефоны, что в переводе в проценты составляет 67,5% от общего населения. Ежегодно на рынке техники появляются новые модели смартфонов.

Изучая оборот мобильных гаджетов в России, около 70% граждан пользуются смартфонами. По показателям с января по ноябрь 2024 года было продано 27 млн устройств. За аналогичный период 2023 года этот показатель составил 26 млн, в 2022 году — 21 млн.

Рассматривая увеличение объема производства телефонов, как средства для быстрого выполнения повседневных задач, также отслеживается тенденция расширения и перенасыщения онлайн-магазинов приложениями. Категории, представленные на различных торговых площадках, многообразны. Социальные сети, игры, приложения по работе с финансами, для заказа еды, осуществления шопинга, стриминговые сервисы и многие другие, несущие в себе разный характер потребления. Рынок настолько огромен, что уже на данный момент способен покрыть внушительный ряд базисных потребностей пользователей.

С развитием технологий люди нацелены на внедрение их практически во все аспекты своей жизни, в том числе в обеспечение комфортного поддержания здорового образа жизни. На сегодняшний день интерес к своему здоровью растёт с большой интенсивностью. Число приложений для спорта на рынке также неизменно увеличивается. В период с 2016 г по 2018 запрос на скачивание фитнес-приложений вырос на 22%[14]. В 2019 году общее количество пользователей составляло 302 млн. человек. В 2020 году был также заметен скачок по использованию приложений для фитнеса, по большей части из-за

¹ Статистика с численностью населения во всём мире [Электронный ресурс]: <https://www.worldometers.info/ru/>, 17.01.2025.

пандемии коронавируса. Число пользователей в период с 2020 по 2021 год достигло 400 млн., а в 2022 цифра дошла до 650 млн[14].

Также, если мы обратимся к статистике одним из востребованных видов спорта в России, по данным FitnessData[13] является бег, при чём его культура с каждым годом увеличивается по представленным показателям. Это складывается прежде всего из доступности, а также немало важную роль играет его продвижение через молодёжную политику, которая реализует мероприятия, направленные на привлечение всё большей доли людей и популяризации бега в России, например, через массовые забеги или благотворительные марафоны.

Информационные системы развиваются с высокой скоростью. Появляется огромное множество приложений в области здоровья и спорта, однако многие не отвечают потребностям пользователей в полной мере. Это указывает на такую проблему, как недоработанный продукт. Большой объём выпускаемых приложений не приравнивается к высокому коэффициенту их качества.

И на данный момент мы живем в эпоху, когда любой вид деятельности подвергается автоматизации и цифровым сопровождением, для комфорtnого и эффективного выполнения потребностей людей. В связи с этим появилась задача спроектировать систему, которая соберёт в себе ряд функциональных возможностей и покроет большинство требований пользователей, способствуют в формировании привычки заниматься бегом на постоянной основе, оставив в приоритете доступность и лёгкость в использовании.

В данной выпускной квалификационной работе будут рассмотрены и проанализированы особенности аналоговых приложений, на основе которых выявится уникальность и практическая значимость разрабатываемого приложения для бега.

Основной особенностью системы выступит возможность создавать свою тренировку с использованием карты и построения на ней индивидуального маршрута. Функциональными возможностями реализованного приложения являются: отслеживание активности пользователя, составление личной тренировки с интеграцией карт и построением маршрута, вывод статистики о

пройденном расстоянии и сожжённых калориях, система планирования, реализованная через заметки в календаре, для своевременного уведомления пользователя о намеченной тренировочной программе, дневник тренировок с выгрузкой данных о пройденных забегах.

Актуальность разработанной информационной системы: увеличенный спрос на фитнес-приложения, тенденция отслеживать свои физические показатели без использования дополнительных устройств, повышение коэффициента актуальности такого вида спорта, как бег, использование минимальных ресурсов для ведения здорового образа жизни и формирование новой спортивной привычки, а также интеграция в приложение маршрутизации для личного забега.

Объектом исследования в данной работе выступит разработка мобильного приложение для бега, с возможностью построения личных маршрутов, с целью повышения качества и удобства выполнения беговых программ.

Предметом исследования выпускной квалификационной работы являются методы построения маршрутов на основе существующих картографических сервисов.

Целью работы является спроектировать и реализовать мобильное приложение для бега с оптимизацией маршрутов исходя из выявленных возможностей аналоговых систем. Для достижения поставленной цели необходимо решить ряд вытекающих задач:

1. Анализ предметной области приложения. Выявление требований и рассмотрение уже существующих, аналоговых систем.
2. Выявление бизнес требований применимых в отношении разрабатываемого мобильного приложения.
3. Проектирование приложения, опираясь на проведенный анализ.
4. Реализация информационной системы, опираясь на технико-экономическое обоснование проводимых работ по разработке.

В основе проводимых работ для достижения поставленной цели было применено два подхода – теоретический и практический, включающие в себя:

- комплексный сравнительный анализ аналоговых систем на рынке;
- анализ, построенный на бизнес-аналитике – SWOT;
- анализ и проектирование, выполненные с использование ключевых нотаций: UML, DFD, BPMN;
- проектирование схем базы данных;
- реализация логики приложения через фреймворк Flutter, язык программирования Dart, среда разработки и сборки готового проекта - Android Studio, системы хранения данных с использование SQLite.

ГЛАВА 1. ПРЕДПРОЕКТНЫЙ АНАЛИЗ СИСТЕМЫ

1.1 Анализ предметной области. Описание концепции собственной системы

Здоровый образ жизни у людей набирает всю большую популярность и это складывается из ряда причин. Благодаря растущему осознанию людей о важности физического и психического благополучия в условиях современного стресса и быстрого темпа жизни. Социальные сети и медиа активно продвигают идеалы здоровья и фитнеса, вдохновляя пользователей делиться своими достижениями и мотивируя других следовать их примеру. Кроме того, научные исследования все чаще подтверждают, что активный образ жизни и правильное питание не только улучшают её качество, но и продлевают, что становится важным фактором для многих, стремящихся к долголетию. Обращая внимание на темп жизни, не всегда людям удается добавить в свой график целенаправленный поход в спортзал, фитнес-клуб, в первую очередь накладываются ограничения по времени, затраченного на дорогу до места тренировок, а также немаловажную роль играет финансовая составляющая. Средняя стоимость абонемента на получения возможности пользоваться услугами фитнес-залов составляет около 60 тысяч рублей в год, где ежемесячный платёж составляет 5 тысяч рублей.

Сравнивая эти показатели люди приходят к поиску альтернатив для поддержания здорового образа жизни с минимальными временными и финансовыми вложениями. Бег один из самых доступных видов спорта, так как для него не требуется специального оборудования или дорогих абонементов, достаточно лишь собраться, выйти на открытое пространство. Удобнее всего проводить забеги, когда есть план и то что поможет отслеживать свои успехи. Именно в этот момент первое, что приходит в голову людям – поиск готовых программ и реализация забегов через мобильные приложения.

Актуальность мобильных приложений для бега заключается в их способности отслеживать прогресс, предлагать тренировочные планы и

мотивировать пользователей через социальные функции и соревнования, при этом тратя на это минимум средств. Такие приложения делают занятия бегом более структуризованными и увлекательными, что способствует регулярности выполнения тренировок.

Рынок приложений для бега переполнен. И с одной стороны это большая привилегия для людей в выборе, но обращаясь ко многим из них пользователи зачастую забрасывают свои намеренья, оставляя желание уделить время здоровому образу жизни нереализованным. Многие мобильные приложения для бега на данный момент предлагают ограниченный набор функций, таких как: отображение показателей здоровья, статистика, готовые программы. Поэтому пользователи часто вынуждены искать альтернативные решения, чтобы получить недостающие возможности. Однако, сталкиваясь с новыми приложениями, они могут обнаружить, что в них отсутствуют те удобные и полезные функции, которые были реализованы в предыдущем продукте.

Безусловно ряд приложений имеют высокий уровень качества при подсчёте физических показателей пользователей, но они не реализовывают возможность создавать свою тренировку с использованием карт. Данная функция удобна тем, что пользователь может построить маршрут забега указав начальную и конечную точку, а система подберёт и сгенерирует маршрут исходя из внешних условий (качества дорожного покрытия), физических показателей, с учётом цели забега – экономичности и минимизации энергозатрат или же наоборот увеличения длительности маршрута.

На основе этого в рамках выпускной квалификационной работы была сформирована цель в реализации мобильного приложения для бега, основной функциональной особенностью которого выступит создание личной тренировки с использованием методов построения маршрутов и алгоритмов их оптимизации. Также будут реализованы немаловажные разделы для отслеживания своей активности: дневник тренировок, в котором будут отображаться выполненные программы; раздел с программами для забегов на фиксированные дистанции, а также с личной тренировкой с построением маршрута; календарь для

планирования тренировок. Кроме того, в мобильном приложении реализуется система регистрации и авторизации пользователя.

1.2 Анализ рынка устройств и приложений

Анализируя увеличение количественных показателей появления на рынке мобильных приложений и сделав вывод об актуальности разработки информационной системы «приложение для бега с оптимизацией маршрутов», следует также провести оценку лидирующих операционных систем для выявления дальнейших средств применяемых в реализации проекта с возможностью корректной работы на разных устройствах.

Выбор системы, под которую будет разработан конечный продукт напрямую влияет на уровень спроса среди потенциальных пользователей. На данный момент основным поставщиком мобильных телефонов является Китай. Он довольно долго держится на лидирующих позициях по экспорту смартфонов по всему миру. В настоящее время люди всё чаще стали отдавать предпочтение смартфонам на базе операционной системы(ОС) Android, в первую очередь из-за ценовой политики. Также немало важную роль играет и уровень защищённости устройств. По сообщению издания «Известия» от 9 августа 2023 года, резко сократились продажи дорогих моделей смартфонов из-за опасения взлома устройств. Модели Apple, хоть и славятся своим уровнем повышенной безопасности, отходят на второй план, что повышает спрос на модели среднего ценового сегмента на базе Android. Также на данный момент в силу принятых мер по ограничению набора функций смартфонов на базе IOS в связи с санкциями,ложенными на территорию Российской Федерации, но техника от компаний Apple всё также востребована, поэтому исключать данную категорию не рекомендуется.

ОС Android — это многофункциональная операционная система для мобильных устройств. Гибкость настроек, доступность многих маркетплейсов, технические характеристики, не уступающие лидирующим моделям высшего ценового сегмента, совместимость со множеством устройств, постоянное

обновление и доработка — всё это рождает с каждым годом спрос на смартфоны Android.

Опираясь на ряд характеристик, заложенных в операционные системы, люди делают выбор в первую очередь основываясь на: ценовую политику, объем предоставляемых функциональных возможностей системы и безопасность.

Мобильные устройства на базе операционной системы Android и iOS на данный момент выступают лидером по ряду характеристик, что отображается на основе статистических данных Statcounter (рисунок 1), выгруженных в виде графика. Это приводит к выводу, что необходимо разрабатывать кроссплатформенное приложение, для повышения охватов возможных пользователей системы.

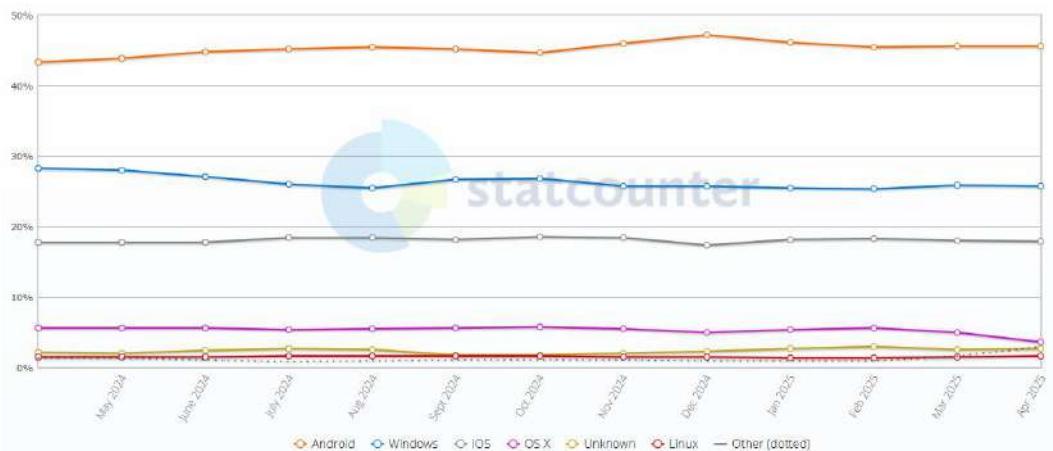


Рисунок 1 — Показатели рынка операционных систем во всём мире

1.3 Сравнительный анализ

Ранее упоминалось, что на данный момент рынок мобильных приложений переполнен трекерами, шагомерами, приложениями для бега, фитнеса и т.д. Многие обладают огромным функционалом, однако зачастую данные системы всё равно не способны удовлетворить необходимые запросы пользователей в рамках занятия таким видом спорта, как бег.

Осуществив анализ в сети Интернет готовых информационных систем из изучаемой предметной области, была выделена группа приложений для бега на сегодняшний день, занимающих лидирующие позиции на рынке [28]. В рамках

исследования рассмотрены не только прямые аналоги и их функциональные возможности, такие как —Nike Run Club, Adidas Running, Running Trainer (Бег), Бег для похудения, Начни бегать, но и косвенные:

- картографические сервисы (Яндекс карты, Google Map, 2ГИС);
- шагомеры (Pacer: шагомер и подсчёт шагов);
- приложения, позволяющие построить маршруты для туристов (Strava, Komoot, ActiveTrip.me).

Выделение прямых и косвенных аналогов даёт возможность проведения более развёрнутого анализа в рамках изучения потребностей пользователей.

Для наглядности примерной доли рынка по скачиванию указанных выше приложений, на основе открытых источников информации, новостных статей в интернете, была сформирована круговая диаграмма (рисунок 2):

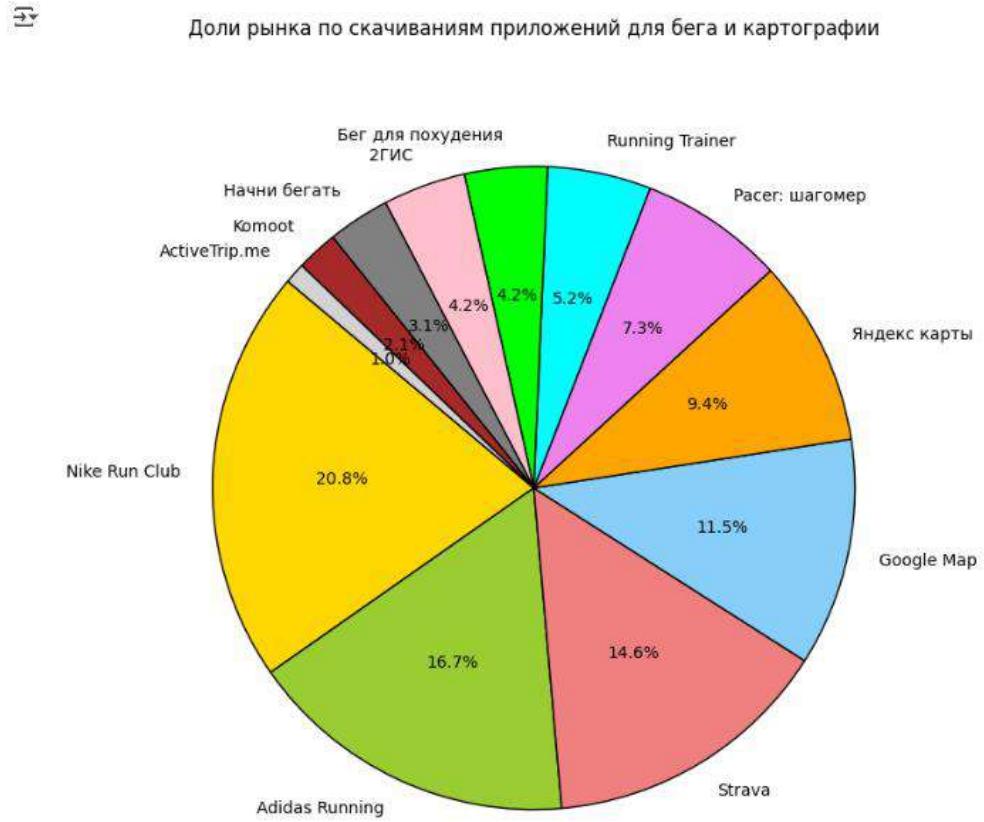


Рисунок 2 – Доли рынка по скачиванию косвенных и прямых аналоговых приложений

Таблица 1 – Аналоговые системы и их возможности

№	Название продукта	Основные возможности
Прямые аналоги		
1	Начни бегать	<ol style="list-style-type: none"> 1) Задать индивидуальные параметры — рост, вес; 2) Голосовое сопровождение; 3) Расписание тренировок по дням недели; 4) Система уведомлений; 5) Создание индивидуальной тренировки; 6) Выбор уровня сложности; 7) Вывод статистики; 8) Раздел с приложениями для силовых нагрузок.
2	Бег для похудения	<ol style="list-style-type: none"> 1) Указание индивидуальных данных (рост, вес, пол, длина шага); 2) Выбор плана тренировок для различных категорий(новичок/продолжающий/продвинутый); 3) Количество пройденных шагов внутри приложения; 4) Архив тренировок; 5) Система оповещений; 6) Сбросить результат; 7) Система достижений; 8) «Мой план» (отображение пройденных дней, и прогресс в %).
3	Running Trainer (Бег)	<ol style="list-style-type: none"> 1) Выбор программы с готовыми дистанциями; 2) История; 3) Настройка статистики тренировок; 4) Настройки профиля; 5) Достижения; 6) Режим просто бега; 7) Возможность поделиться тренировкой.
4	Nike Run Club	<ol style="list-style-type: none"> 1) Интеграция с внешними устройствами; 2) Базовая статистика – средний темп, дистанция, время; 3) Аудиотренировки и плейлисты с музыкой от известных тренеров, атлетов, музыкантов; 4) Виртуальные награды; 5) Лента друзей с просмотром и комментированием записей других пользователей, возможностью поделиться своими достижениями.
5	Adidas Running	<ol style="list-style-type: none"> 1) Собственная клубная система; 2) Отслеживание параметров пробежки (дистанция, темп, расход калорий);

№	Название продукта	Основные возможности
		3) Постановка целей; 4) Встроенный сервис для прослушивания аудиозаписей; 5) Выбор программы тренировок; 6) Голосовое сопровождение; 7) Синхронизация с My Fitness Pal, Suunto, Kinomap, Running Heroes, Keller Sports, Polar и Garmin Connect; 8) Интеграция с Apple Здоровье; 9) Чтение новостей от бренда и статей из блога; 10) Публикация своих постов.

Исходя из данных о возможностях, предоставляемых популярными прямыми аналогами, были выбраны те, что занимают лидирующие позиции на рынке, с большей долей установок, для проведения сравнительного анализа, целью которого является обоснование уникальности системы и целесообразности разработки. Ниже в таблице 2 представлены результаты проведенного исследования, выделены наиболее значимые функциональные возможности, реализуемые на первом спринте разработки – верхнего уровня, а также нижнего уровня – возможности при дальнейшем развертывании системы:

Таблица 2 – Сравнительный анализ аналоговых систем

№ п/п	Список характеристик	Название продукта №1 «Nike Run Club»	Название продукта №2 «Adidas Running»	Название продукта №3 «Начни бегать»	Название своей системы «Стремление»
Верхнего уровня					
1	Дневник пробежек (время, темп, км)	+	+	+	+
2	Календарь активности	+	-	-	+
3	Создание индивидуальных маршрутов	-	-	-	+

№ п/п	Список характеристик	Название продукта №1 «Nike Run Club»	Название продукта №2 «Adidas Running»	Название продукта №3 «Начни бегать»	Название своей системы «Стремление»
4	Составление статистики на основе активности	+	+	+	+
5	Выбор готовых планов занятий	+	+	+	+
6	Шагомер	-	-	-	+
	Итог:	4	3	3	6
Нижнего уровня					
1	Система достижений	+	+	-	+
2	Аудио помощник	-	-	+	+
3	Установка метрик	-	-	+	+
4	Новостной раздел	-	-	+	+
5	Взаимодействие с другими пользователями (совместные забеги)	-	-	-	+

Также в ходе проведенного анализа были рассмотрены потенциальные пользователи системы, с указанием возможностей внутри приложения.

Целевая аудитория:

- Люди с лишним весом - отслеживать сброшенные калории, голосовой помощник, статистика и цели;
- Профессиональные бегуны – создание индивидуальных тренировок, контроль за физическими показателями с отслеживающей статистикой;
- Любители бега, с малым или нулевым опытом - подготовленные варианты забегов и тренировок, статистика и ежедневная мотивация, система оповещений, система поощрений;

- Спортсмены из других видов спорта - контроль за физическими показателями с отслеживающей статистикой, создание индивидуальных тренировок.

Вывод

Большинство приложений закрывают одну область функционирования, но отстают в развитии смежных с ней. Например, такие приложения как — Strava, Komoot и даже наше российское приложения — ActiveTrip.me, выделяются на рынке проработанной навигацией, имеют возможность отображения слоёв, детализации маршрута, умная система корректировки линии маршрута. Однако они не имеют никакой мотивационной и программной составляющей для занятия бегом. Если рассматривать тематические приложения, такие как — «Начни бегать», «Бег для похудения», «Running Trainer (Бег)», «Adidas Running» и т.д., то в них можно наблюдать противоположную ситуацию. Многие из данных приложений обладают превосходной мотивационной базой, продуманной системой тренировок, но в них не предусмотрена работа с маршрутами. Также следует учитывать, что большинство прямых аналогов ушли с Российского рынка или наложили ограничения на использование комплекса полезных функций, что даёт определённое преимущество разрабатываемой системе на данный момент.

Исследуя предметную область, можно сделать вывод, что на сегодняшний день наличие мотивационной составляющей, вывод статистики и продуманной системы навигации внутри одной системы, не может предложить ни один проанализированный программный продукт.

1.4 SWOT-анализ

Для выявления эффективности проекта и целесообразности его реализации крайне важно применять комплексный подход при анализе, используя ряд методов, позволяющих разложить объект исследования на мелкие составляющие. Поэтому следующим и одним из значимых инструментов при

анализе разрабатываемой системы является SWOT-анализ (таблица 3). Это матрица, которая помогает наглядно отобразить сильные и слабые стороны программного продукта, позволяет детальнее углубиться в проект, наметить пути развития, а также выбрать подходящую стратегию для решения выявленных недостатков[29].

При этом сам SWOT-анализ сводится к формированию четырёх направлений для исследования:

S – сильные стороны продукта, дающие преимущество перед аналоговыми системами; W – минусы реализуемой системы; O – возможные пути развития системы в будущем, которые со временем перерастут в сильные стороны; T – угрозы, чаще всего выделяемые с целью реализации стратегического планирования и разработки сценариев по их своевременному предотвращению.

Для успешного функционирования мобильного приложения для бега с оптимизацией маршрутов необходимо учитывать выявленные угрозы и разрабатывать стратегии для их минимизации, чтобы обеспечить устойчивый рост и развитие приложения.

Таблица 3 – SWOT-анализ

S	Сильные стороны	W	Слабые стороны
	<ul style="list-style-type: none"> – Возможность отслеживать маршрут для забега в реальном времени – Возможность выбора параметров забега – Построение маршрута с учётом внешних факторов, с приоритетом на безопасность: покрытия дороги, ремонтные работы, погодные условия – Кроссплатформенность приложения 		<ul style="list-style-type: none"> – Высокий уровень конкуренции от аналоговых приложений, закрепившихся на рынке – Монетизация приложения – Зависимость при работе с картами от GPS – Постоянный доступ в интернет во время использования
T	Угрозы	O	Возможности

S	Сильные стороны	W	Слабые стороны
	<ul style="list-style-type: none"> – Сбой внешних карт, интегрированных в приложение – Угроза утечки данных – Ускоренное развитие технологий может привести к частой реализации новых функций внутри приложения, следовательно, возможно повышение уровня расходов на разработку 		<ul style="list-style-type: none"> – Подключение различных гаджетов для более детального анализа физического состояния пользователя – Создание и сохранение постоянных маршрутов, для повторного прохождения – Продуманная система мотивации пользователя, с возможностью привлечения спонсоров на реализацию отдельных заданий – Спонсоры в лице фитнес-организаций, деятелей спорта – Расширение функционал возможностей, например, отслеживание полезных привычек (водный баланс, режим сна и т.д.) – Повышение значимости здорового образа жизни у молодого поколения – Расширение функционал для лиц с ограниченными возможностями здоровья – Отображение результатов для других пользователей – Прохождение маршрутов в режиме «Совместный забег» – Делиться своим результатом и взаимодействовать дистанционно с личным тренером

Опираясь на данные полученные при составлении матрицы SWOT-анализа, были выявлены сильные и слабые стороны разрабатываемой информационной системы. Также особое внимание уделено угрозам системы. На сегодняшний день в рамках бизнеса важной задачей является поддержание должного уровня безопасности выпускаемого программного продукта. Утечка данных или атаки на серверы оказывают негативное воздействие на работу

приложения и безопасность пользовательских данных. Что в свою очередь, при успешном извлечении личной информации, приводит к потере доверия со стороны пользователей системы, негативная оценка продукта на маркетплейсах, а также может послужить поводом привлечения к юридической ответственности разработчика перед клиентом. Это не только способно существенно повредить репутацию приложения, но и снизить его конкурентоспособность.

Для предотвращения подобных проникновений в систему существуют множество решений, но можно выделить базовые, которые следует интегрировать при разработке системы:

- При передаче данных необходимо использовать протоколы HTTPS для защиты данных, передаваемых между приложением и сервером. Это позволит предотвратить нежелательный перехват данных из вне;
- Реализуя логику регистрации и авторизации пользователей следует интегрировать многофакторную аутентификацию для повышения уровня безопасности доступа к приложению. Это может включать использование паролей, SMS-кодов или биометрических данных (отпечатки пальцев, распознавание лиц). Также заранее необходимо сформировать требования к созданию сложных паролей и регулярной их смене. Что касается хранения паролей пользователей, следует применять методы хеширования;
- Пользоваться функцией резервного копирования данных, чтобы в случае утечки или потери данных можно было восстановить информацию.

1.5 Требования к информационной системе

В условиях проводимого анализа, необходимо сформировать чёткие требования под реализуемую систему, которые будут отвечать ожиданиям целевой аудитории. Данный этап крайне важен и формируется в рамках технического задания, составляемого между заказчиком и исполнителем, с целью выделения ключевых задач будущего мобильного приложения, на основе которых строится понимание и логика вытекающих бизнес-процессов.

Как правило требования разделяют на функциональные и нефункциональные. Первые, включают в себя конкретные действия, выполняемые информационной системой, при прямом взаимодействии пользователя с продуктом. Вторые, подразумевают под собой качественную оценку, то есть различные параметры и ограничения, начиная с описания производительности и заканчивая безопасностью будущей системы. Грамотно сформулированные требования упрощают процесс разработки выпускаемого продукта и помогают ответить в полном объёме на ожидания клиентов и целевой аудитории.

В условиях составления технического задания и выявления требований, предъявляемых к системе, необходимо пользоваться методами интервьюирование и анкетирование. В рамках выпускной квалификационной работы был проведён анализ отзывов представленных аналоговых систем, на основе выявленных недочётов и запросов пользователей были сформированы следующие требования к реализуемому мобильному приложению.

Функциональные требования:

1. Авторизация пользователя.
2. Регистрация пользователя.
3. Создание тренировки с построением маршрута.
4. Установка начальной и конечной точки.
5. Установка личных метрик.
6. Изменение начальных метрик.
7. Установка заметки о запланированной тренировки.
8. Выбор даты и времени заметки.
9. Удаление заметки.
10. Установка отметки о выполнении запланированной тренировки.
11. Контроль функции шагомера.
12. Сохранение результатов пробежки.
13. Удаление результатов пробежки.

14. Добавление комментария в дневники тренировок.

Нефункциональные требования:

1. Требования к операционным системам:

- Минимальная версия Android – 8.0 или выше;
- Минимальная версия iOS – 13 или выше.

Это связано с тем, что более ранние версии мобильных телефонов с операционной системой Android могут не поддерживать ряд библиотек или функций, реализованных через них. Аналогично для систем на базе iOS.

2. Повышенный уровень защиты данных пользователей.

3. Резервное копирование не реже двух раз в месяц, на случай сбоев на стороне сервера.

4. Нагрузка на систему не менее 1000-2000 пользователей.

5. Простота и удобство интерфейса для пользователей.

6. Быстрый отклик от картографического сервиса. Допустимая задержка до 10 секунд.

7. Переход между страницами внутри приложения без задержек, мгновенная загрузка выбранного раздела.

8. Обязательная регистрация пользователей.

1.6 Информационные потоки. Методология DFD

На данном этапе, исходя из выявленных участников бизнес-процессов, их роли внутри системы, описаны потоки данных между блоками системы, их начальное и конечное состояние. Для наглядного представления использовалась диаграмма потоков данных — DFD (Data Flow Diagram), а также подробное описание бизнес-процессов с помощью диаграммы BPMN.

Перед тем как начать составлять структуру схемы обмена данными внутри системы, следует выделить входные и выходные данные.

Входными данными являются:

- Настройки пользователя — язык, голосовой помощник, физические показатели, метрики и т.д.;
- Личные данные — имя, возраст, пол, рост и вес пользователя. Эта категория данных может быть использована для расчета калорий, темпа бега и других параметров тренировки;
- Данные для регистрации и последующей авторизации в системе.

Выходными данными являются:

- Отображение статистики, маршрут, изменение физических показателей. Статистика тренировок: информация о пройденном расстоянии, времени тренировки, сжигаемых калориях, темпе бега и других параметрах тренировки, которая может быть отображена в виде графиков или диаграмм.
- Рекомендации, планы тренировок: на основе входных данных приложение может предложить пользователю персонализированные тренировки и планы, учитывая его цели, уровень подготовки и предпочтения.
- Уведомления и напоминания: приложение может отправлять уведомления о предстоящих тренировках, достижении целей, напоминания о необходимости отдыха или выполнения разминки после тренировки.
- Система мотивации. Приложение может награждать пользователя за достижение определенных результатов, установку новых персональных рекордов или регулярность тренировок.

На рисунке 3 представлена контекстная диаграмма нулевого уровня. Она отображает общую картину взаимодействия пользователя с приложением. Являясь внешней сущностью, пользователь подаёт приложению входные данные описанные выше. Программа в дальнейшем использует их для расчёта и вывода статистики.

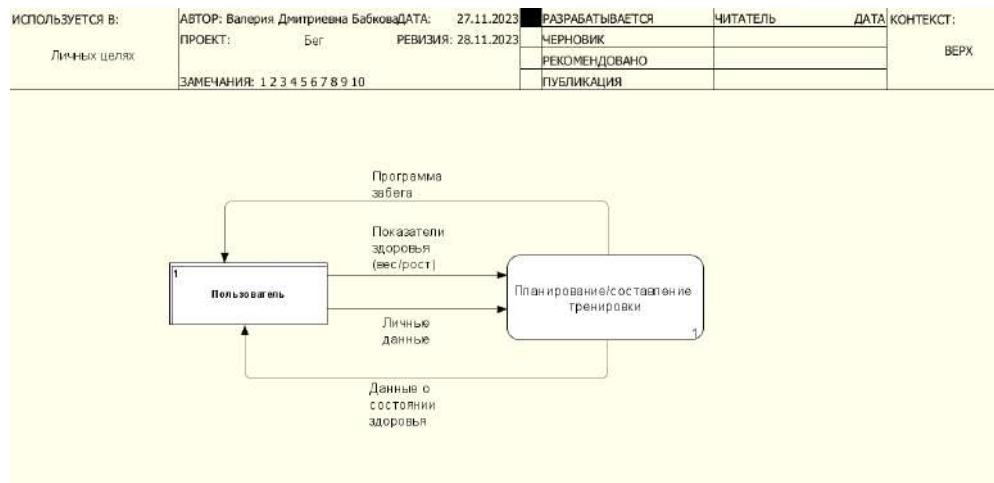


Рисунок 3 — Контекстная диаграмма

После мы опускаемся на уровень ниже (рисунок 4), декомпозирируя блок «Планирования и составление тренировок». Здесь подробно описывается вход пользователя в систему и его дальнейшее взаимодействие с ней. Вводимые данные пользователем в личную карточку, влияют и используются в нескольких процессах: работа со статистикой, составление программ для забега, система целей и мотивации.

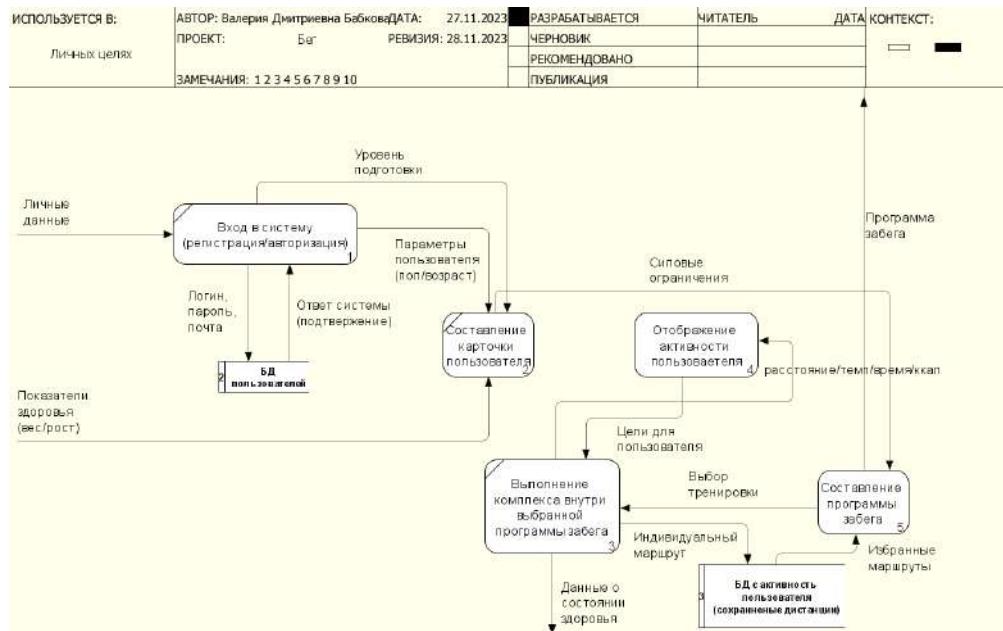


Рисунок 4 — Диаграмма потоков данных 1-ый уровень

Следующая декомпозиция (рисунок 5) составление программ забега. Выполнение данного процесса заключается в следующем. При выборе программ

для забега мы обращаемся к нашему разделу с программами. В нём происходит ветвление: выбрать готовые программы, выбрать из БД ранее пройденные маршруты или установить индивидуальный. Пользователь, как внешняя сущность заносит параметры маршрута, а программа, взаимодействуя с ГИС и БД прокладывает оптимальный маршрут.

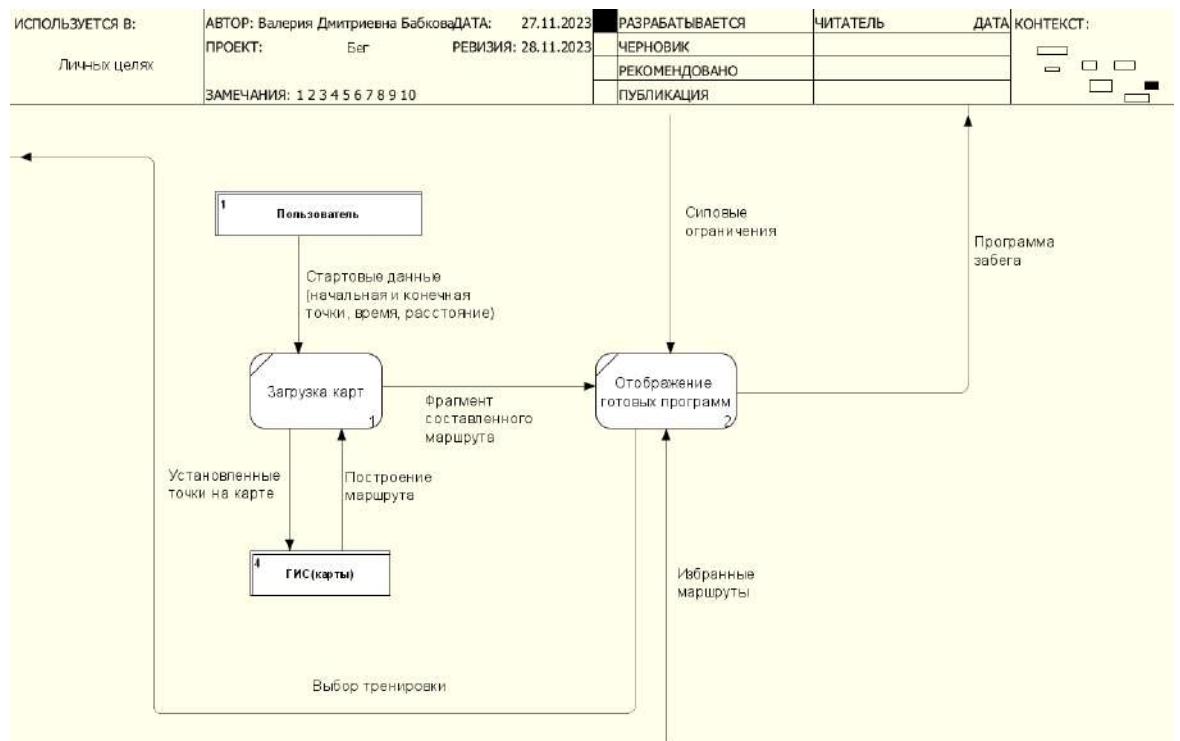


Рисунок 5 — Диаграмма потоков данных 2-ой уровень

Последний блок декомпозиции «Отображение активности пользователя» (рисунок 6) отвечает за сбор данных и отображение статистики. Анализирует процесс тренировки, пройденного маршрута, приложение сопоставляет введённые данные пользователя и усредненные значения по показателям здоровья, пройденному расстоянию и выводит приближенные. Вся активность пользователя хранится в БД откуда система подгружает информацию.

ИСПОЛЬЗУЕТСЯ В:	АВТОР: Валерия Дмитриевна Бабкова	ДАТА: 27.11.2023	РАЗРАБАТЫВАЕТСЯ	ЧИТАТЕЛЬ	ДАТА	КОНТЕКСТ:
Личных целях	ПРОЕКТ: Бег	РЕВИЗИЯ: 28.11.2023	ЧЕРНОВИК			
	ЗАМЕЧАНИЯ: 1 2 3 4 5 6 7 8 9 10		РЕКОМЕНДОВАНО			

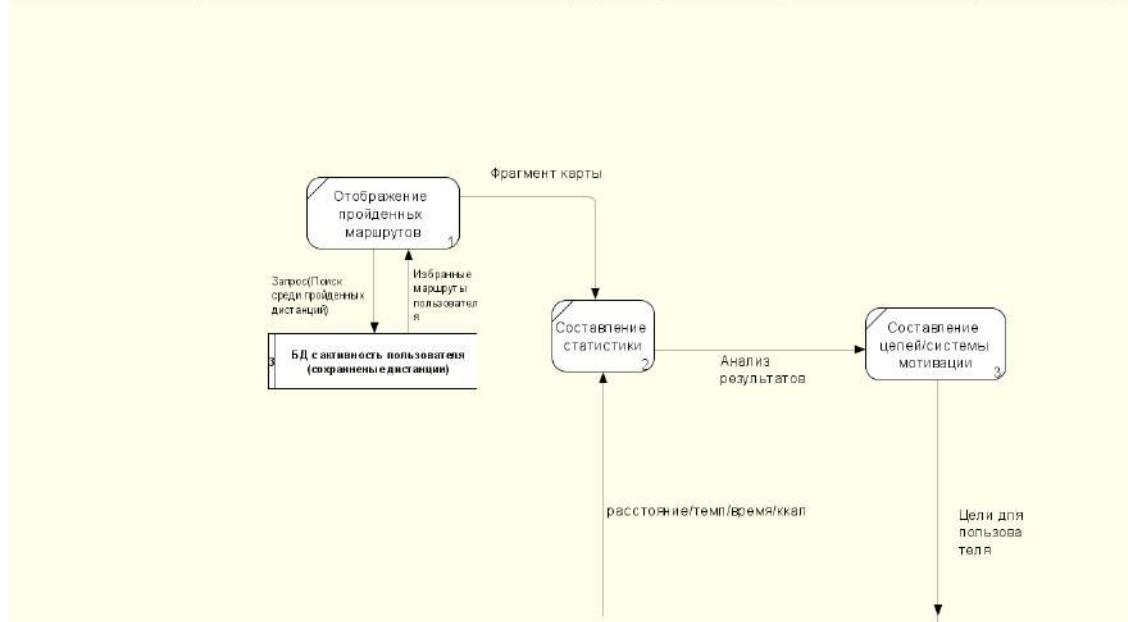


Рисунок 6 — Диаграмма потоков данных, декомпозиция

1.7 Схема бизнес-процессов. Методология BPMN

Выявить возможные бизнес-процессы разрабатываемого мобильного приложения — это понять структуру будущей системы, её сильные и слабые стороны. Анализируя систему на первых этапах, удалось рассмотреть следующие бизнес-процессы и описать их составляющие (таблица 4).

Таблица 4 — Бизнес-процессы и их составляющие

Процесс	Вход	Выход
Авторизация	Личные данные	Вход в систему/переход в раздел регистрации
Личный кабинет	Личные данные пользователя/медицинские данные	История забегов/сохраненные маршруты/статистика

Процесс	Вход	Выход
Анализ карт в соответствии с установленной начальной и конечной точкой	Отметка на карте	Перечень оптимальных маршрутов, варианты забегов, с указанием приближенного времени и расстояния. Отображение помех на пути
Изменение расстояния относительно вашего передвижения	Данные с карт о местоположении, скорость прохождения ед. расстояния за ед. времени	Уменьшение расстояния на экране или озвучивание внутри приложения голосовым помощником
Регулирование времени относительно вашего передвижения	Данные с карт о местоположении, скорость прохождения ед. расстояния за ед. времени	Фиксирование пройденного времени и оставшегося
Автоматическая перестройка маршрута при изменении темпа бега	Установка начальной и конечной точки маршрута на карте	Отображение на карте изменений, голосовое оповещение
Система сигналов/оповещений	Набор аудио сигналов (система оповещения)	Вывод сигнала через гарнитуру используемого устройства
Возможность сохранения маршрута и результата конкретной тренировки	Данные о результате пробежки	Вся информация о тренировки (время, расстояние, маршрут), занесено в раздел личного кабинета — «Дневник»
Работа с картами. Отображение карт внутри приложения	Загрузка карт с помощью API ключа, находящихся в общем доступе (например, 2ГИС, Яндекс)	Отображение выбранного города и своего местоположения на карте

Выделив процессы и их составляющие, следующим шагом стало их классификация на категории: верхнего — нижнего уровня, основные — вспомогательные. Ниже в таблице 5 отражены получившиеся результаты.

Таблица 5 — Классификация процессов

Процесс	Основной	Вспомогательный
Авторизация		+
Личный кабинет		+
Анализ карт в соответствии с установленной начальной и конечной точкой	+	
Изменение расстояния относительно вашего передвижения	+	
Регулирование времени относительно вашего передвижения	+	
Автоматическая перестройка маршрута при изменении темпа бега		+
Система сигналов/оповещений		+
Возможность сохранения маршрута и результата конкретной тренировки	+	
Работа с картами. Отображение карт внутри приложения	+	

Выявленные бизнес—процессы на этапе анализа системы, являются обобщенным представлением работы приложения. С помощью нотации BPMN (Business Process Model and Notation), языка моделирования бизнес—процессов, который является промежуточным звеном между визуализацией и воплощением бизнес—процесса, работа системы рассмотрена с точки зрения нескольких сущностей, систем управления одновременно.

Существует огромное разнообразие инструментов для работы с нотацией BPMN, приложенная схема разрабатывалась с помощью кроссплатформенного ПО — draw.io.

Диаграмма состоит из пяти дорожек: разработчик, ГИС(карты), приложение, сервер, пользователь. Описание процессов будет осуществляться от лица пользователя.

Запуская систему начинается ветвление (рисунок 7). Необходимо наличие аккаунта. При регистрации или авторизации пользователь заносит личные данные в систему, после чего процесс передаёт сигнал в сервер с запросом на проверку ведённых данных. В зависимости от ответа системы, мы либо получим доступ к функционалу приложения, либо останемся на начальном процессе авторизации/регистрации.

Внутри программы перед пользователем открывается набор разделов: просмотр статистики, просмотр дневника тренировки, спланировать тренировку, изменение настроек, дача обратной связи, запуск программы забега. Попадая в тот или иной раздел у нас есть выбор: покинуть приложение или вернуться в предыдущую точку.

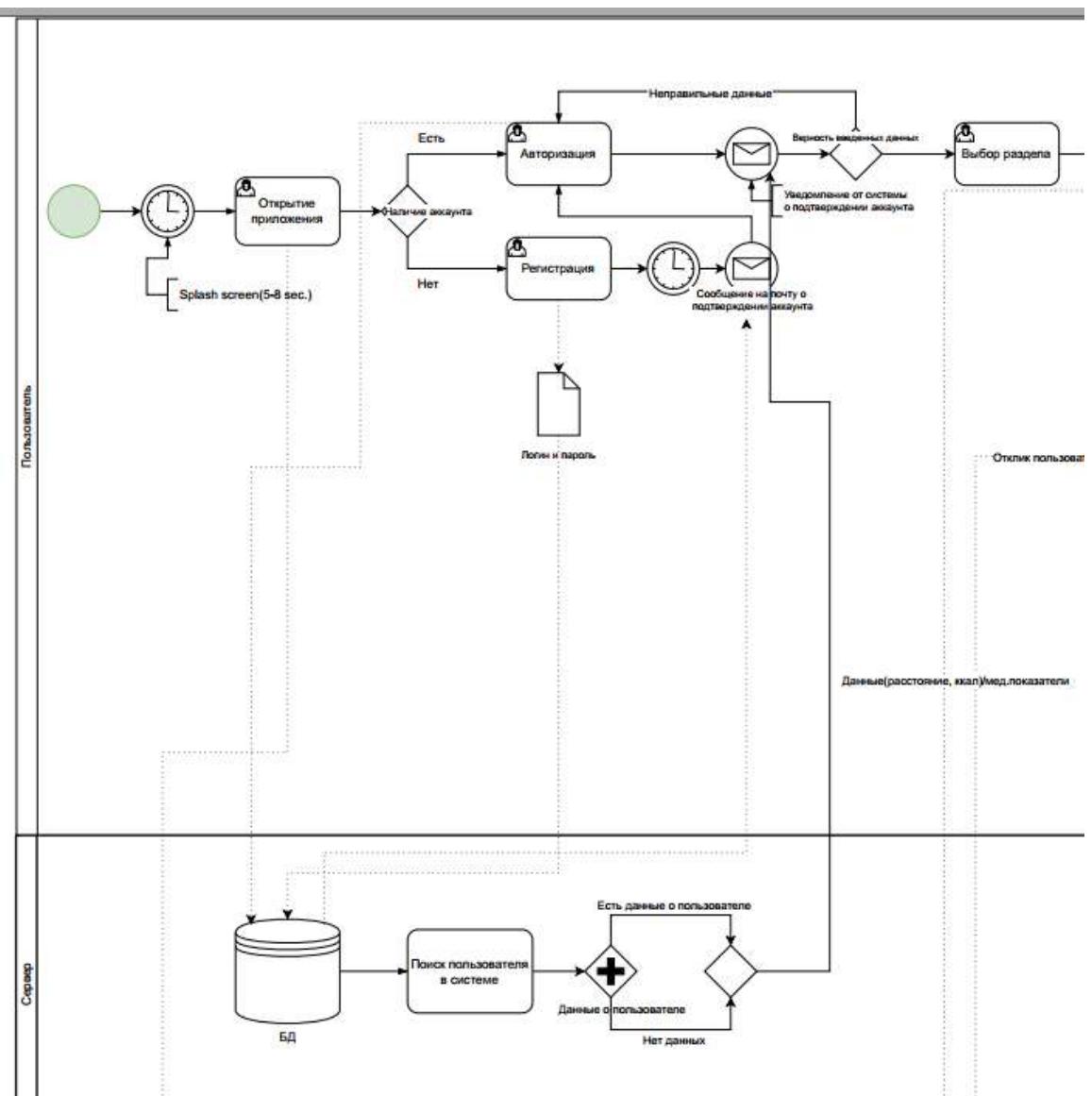


Рисунок 7 — Диаграмма BPMN, описание регистрации и авторизации

Раздел «Дача обратной связи» связан непосредственно с дорожкой разработчика (рисунок 8). Возникновение неполадок или предложения по улучшению отправляются в строку разработчика, где он обрабатывает заявку, выявляет проблему, даёт обратную связь в виде доработок программного кода.

В приложении присутствует система оповещений (рисунок 8), которая отправляет пользователю уведомления о запланированных тренировках. Голосовой помощник, тоже относится к системе оповещения и запускается во время активации одной из выбранных тренировок.

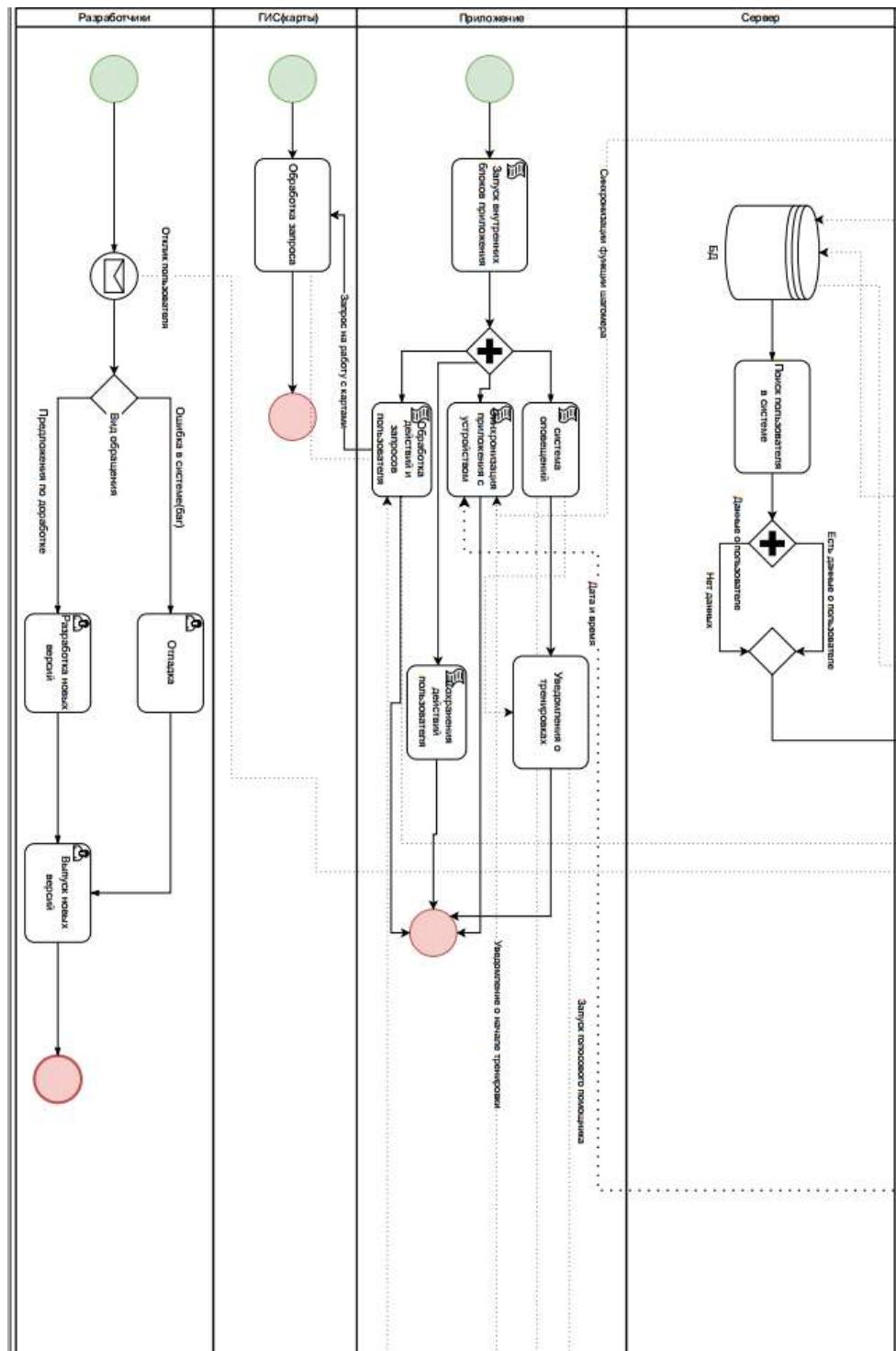


Рисунок 8 — Фрагмент диаграммы BPMN

Процесс в дорожке пользователя — «Запуск программы для забега» (рисунок 9), связан одновременно с дорожками — приложение, сервер и ГИС. При создании личного маршрута в ГИС отправляется запрос на отображение и работу с картами. Завершая тренировку маршрут и статистика переходят в БД, физические показатели отправляются в раздел с активностью внутри дорожки пользователя.

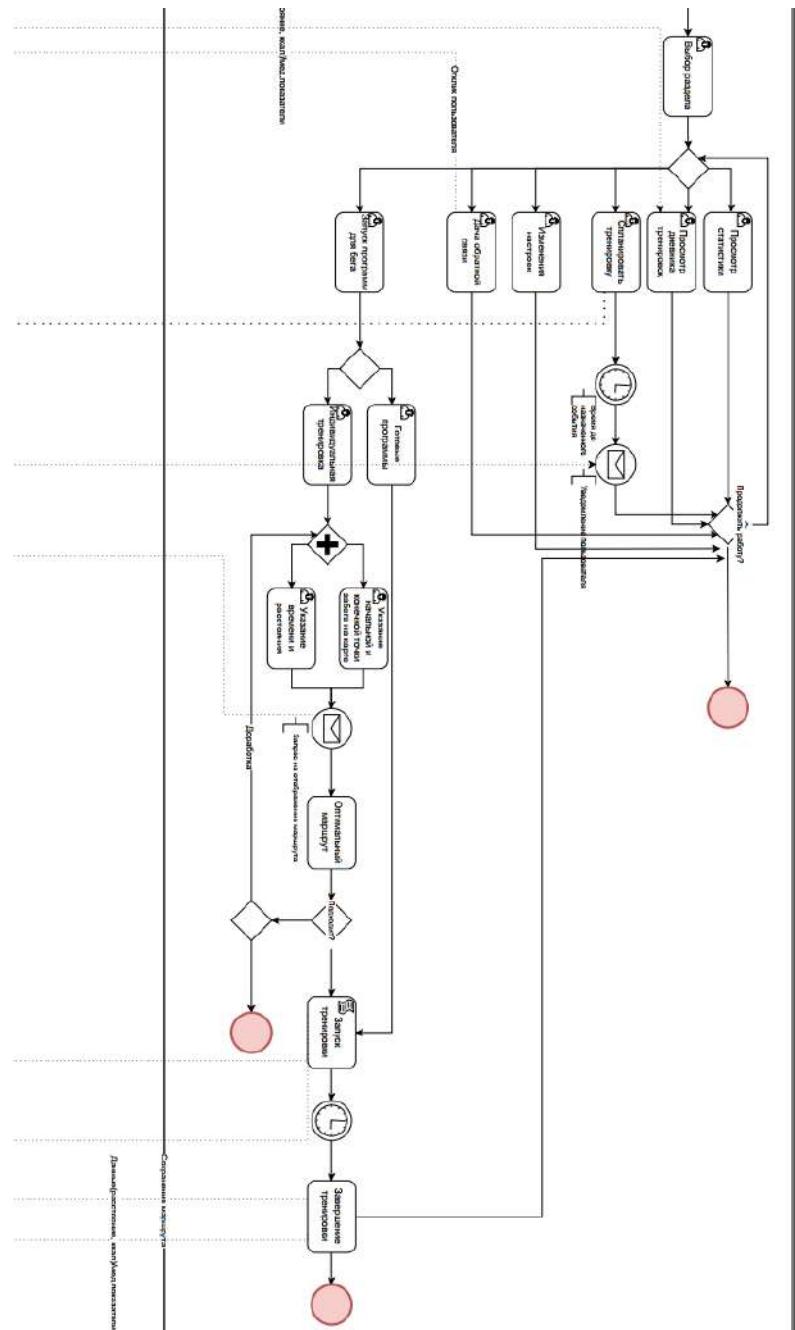


Рисунок 9 — Фрагмент диаграмма BPMN, запуск тренировочной программы

ГЛАВА 2. ПРОЕКТИРОВАНИЕ ИС

2.1 Unified Modeling Language

Unified Modeling Language (UML) — унифицированный язык моделирования, позволяющий графически описать процессы на этапе проектирования информационной системы. Способствует документированию структуры реализуемого программного продукта, отображает возможные варианты поведения, включенные в него объекты и связи между ними. [1][4]

Преимущество данного языка в том, что он облегчает понимание протекающих процессов внутри системы, её архитектуру, связи между компонентами, наполняющих приложение, что в дальнейшем упрощает работу со сборкой и развертыванием программного продукта. [6]

Язык моделирования преимущественно используется для описания логики сложных систем. Считается нецелесообразным применять нотацию UML для небольших проектов, так как это довольно затратный и ресурсоёмкий процесс.

Для описания операций, действий с разных сторон, выполняемых системой, в языке UML предусмотрено девять типов диаграмм:

- диаграммы классов;
- диаграммы объектов;
- диаграммы прецедентов;
- диаграммы последовательностей;
- диаграммы кооперации;
- диаграммы состояний;
- диаграммы действий;
- диаграммы компонентов;
- диаграммы развертывания.

Каждая из выше указанных диаграмм является способом детализации и конкретизации сложных систем и использует свои стандартизованные элементы графического представления.

В данной главе, по проектированию мобильного приложения для бега будут реализованы следующие диаграммы:

- диаграммы прецедентов или диаграмма вариантов использования;
- диаграммы последовательностей;
- диаграммы компонентов;
- диаграммы развертывания.

2.2 Use case диаграмма

Use case диаграмма — это первое, с чего предстоит начинать работу при проектировании системы. С её помощью мы выделяем:

- Актёров;
- Варианты использования;
- Связи.

Актёры — это совокупность объектов, которые буду взаимодействовать с программным продуктом извне и воздействовать на его работу. Обращаясь к анализу предметной области, описанного в главе 1, удалось выявить следующих актёров:

- Пользователь с аккаунтом;
- Пользователь без аккаунта;
- Картографический сервис.

У каждого актёра есть своя определённая последовательность действий внутри системы, которая в UML обозначается, как вариант использования в дальнейшем ВИ.

Основные ВИ:

- Авторизация (пользователь с аккаунтом);
- Регистрация (пользователь с аккаунтом/ пользователь без аккаунта);
- Запланировать тренировку (пользователь с аккаунтом);

- Выбор готового плана тренировки (пользователь с аккаунтом/пользователь без аккаунта);
- Просмотр статистики (пользователь с аккаунтом);
- Работа с дневником тренировки (пользователь с аккаунтом);
- Составление личной тренировки (пользователь с аккаунтом);
- Предоставление доступа к картам (картографический сервис).

Ниже на рисунке 10 представлена готовая диаграмма вариантов использования, которая отображает актёров проектируемой системы и их установленные связи с вариантами использования.

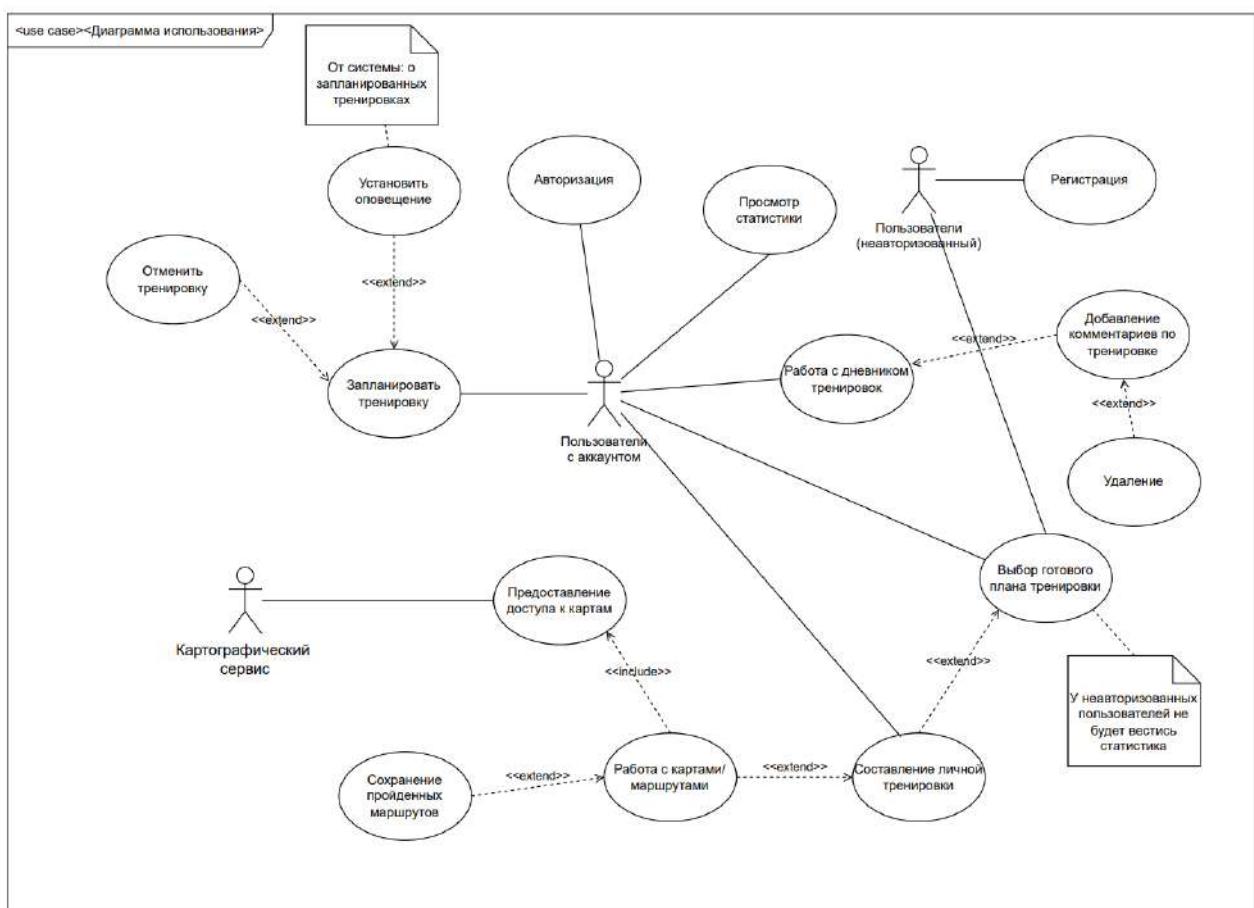


Рисунок 10 — Диаграмма Use case

2.3 Сценарий

Отобразив основные объекты на диаграмме прецедентов, следующим шагом необходимо дополнить её текстовым описанием, которое в UML принято

оформлять в качестве сценариев вариантов использования [5]. Он описывается в несколько этапов, основной — составление главного раздела, который будет отображать зависимости и общую информацию о ВИ. Проще всего реализовать все сценарии в едином файле, используя Microsoft Excel, с возможностью указания гиперссылок на исключения, а также при необходимости ссылки на другие варианты использования.

Рассмотрим ниже варианты использования:

- Просмотр статистики (рисунок 11—13);
- Работа с дневником тренировки (рисунок 14—15).

Главный раздел сценария выполнения варианта использования "Просмотр статистики"	
<u>Вариант использования</u>	Просмотр статистики
<u>Актеры</u>	Пользователь(авторизованный)
<u>Краткое описание</u>	Открытие раздела "Статистика"
<u>Цель</u>	Просмотр статистики за разные периоды времени(статистика за месяц, неделю, год). Выборка данных
<u>Тип</u>	Базовый вариант использования
<u>Ссылки на другие варианты использования</u>	отсутствуют

Рисунок 11 — Главный раздел сценария ВИ «Просмотр статистики»

Раздел Типичный ход событий сценария выполнения варианта использования "Просмотр статистики"	
Действия актёров	Отклик системы
1. Открыть раздел со статистикой	2. Запрос к БД на получение информации об активности пользователя
3. Выбрать период активности	
4. Нажать "Ok"	
	<u>1. Исключение. Выбор периода с нулевой активностью</u>
	5. Получение ответа от БД
	6. Отобразит активность в виде диаграммы с описанием

Рисунок 12 — Типичный ход событий «Просмотр статистики»

ИСКЛЮЧЕНИЯ	
Раздел Исключения сценария выполнения варианта использования "Просмотр статистики"	
Действия актёров	Отклик системы
	<u>1. Исключение. Выбор периода с нулевой активностью</u>
	5. Запрос к БД
	6. Получение ответа от БД
	7. Система сообщит, что за данный период не было активности
	8. Предлагает выбрать другой период для обзора статистики
<u>3. Выбрать период активности</u>	

Рисунок 13 — Исключения «Просмотр статистики»

Главный раздел сценария выполнения варианта использования "Работа с дневником тренировки"	
<u>Вариант использования</u>	Работа с дневником тренировки
<u>Актеры</u>	Пользователь(авторизованный)
<u>4 Краткое описание</u>	Открытие дневника тренировок. Просмотр пройденных дистанций (по месяцам). Открытие детального описания хода тренировки
<u>Цель</u>	Детальное изучение пройденной дистанции
<u>Тип</u>	Базовый вариант использования
<u>Ссылки на другие варианты использования</u>	Расширяется: Добавление комментариев/заметки

Рисунок 14 — Главный раздел сценария ВИ «Работа с дневником тренировки»

Раздел Типичный ход событий сценария выполнения варианта использования "Работа с дневником тренировки"	
Действия актёров	Отклик системы
1. Открытие тематического раздела	2. Запрос к серверу
	3. Загрузка данных
5. Выбор конкретной записи. Нажатие на запись	4. Отображение имеющихся данных
	6. Получение данных из БД
8. Нажатие кнопки назад	7. Переход на отдельную страницу с полной информацией по тренировке
	9. Отображение страницы "Дневник тренировки"

Рисунок 15 — Типичный ход событий «Работа с дневником тренировки»

Представленные выше сценарии ВИ описаны с точки зрения актёров, показывают их действия и ответную реакцию системы, а также отдельно были вынесены исключения — это ветвления, возникающее при выполнении определённых условий. Остальные фрагменты сценариев ВИ представлены в приложении А.

2.4 Диаграмма последовательностей

Данный тип диаграмм относится к диаграммам взаимодействия и отображает выполнение процессов внутри системы, то есть то, как они реализуются между объектами при их совместной работе. Взаимодействие показывается в виде потоков сообщений, без указания точного времени, но с отсылкой на него. [9]

Смысл времени на диаграмме: на ней не указывается четко время события или процесса, лишь показывается взаимосвязь раньше—позже. А отображается это расположением самих сообщений на диаграмме последовательности. Те что выше, передаются раньше тех, которые расположены ниже.

Каждый объект на диаграмме имеет свою собственную линию жизни и изображается она вертикальной пунктирной линией. Как раз между ними и происходит передача сообщений, которая показана на диаграмме как стрелочка.

Диаграммы последовательностей формируются на основе описанных ранее сценариев для обозначенных вариантов использования [7]. Ниже на рисунках приведены готовые диаграммы для таких вариантов использования, как: «Просмотр статистики» (рисунок 16), «Работа с дневником тренировок» (рисунок 17), «Выбор готового плана тренировок» (рисунок 18). Остальные фрагменты диаграмм последовательностей ВИ представлены в приложении Б.

2.5 Создание схемы базы данных для реализуемой системы

Опираясь на проведённый анализ системы, выявленные функциональные возможности пользователей, а также рассмотрев возможные потоки данных, были выделены блоки информации[17], которые необходимо хранить для планового функционирования системы. На основе чего спроектирована следующая диаграмма базы данных (рисунок 19):

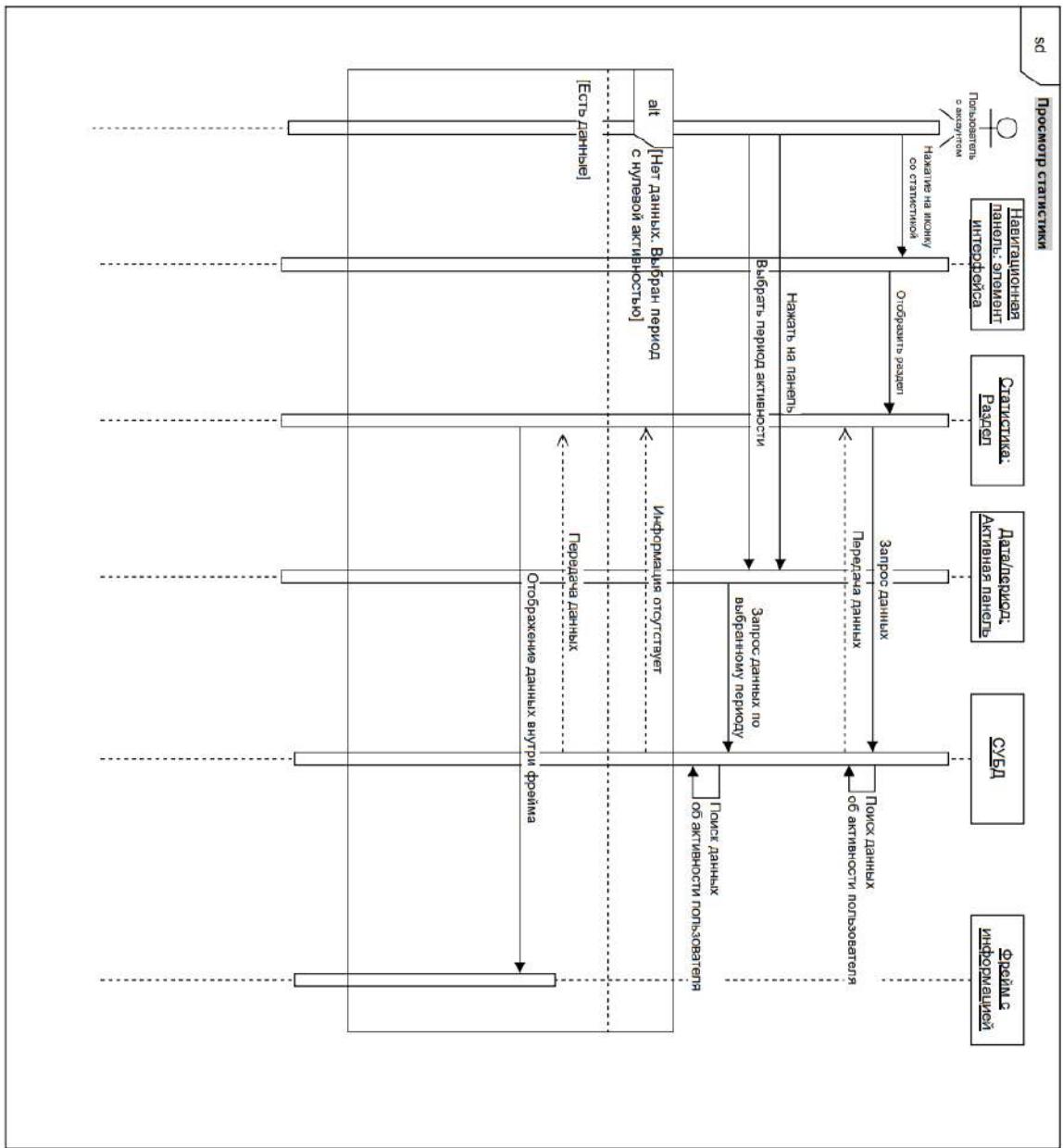


Рисунок 16 — Диаграмма последовательностей для ВИ «Просмотр статистики»

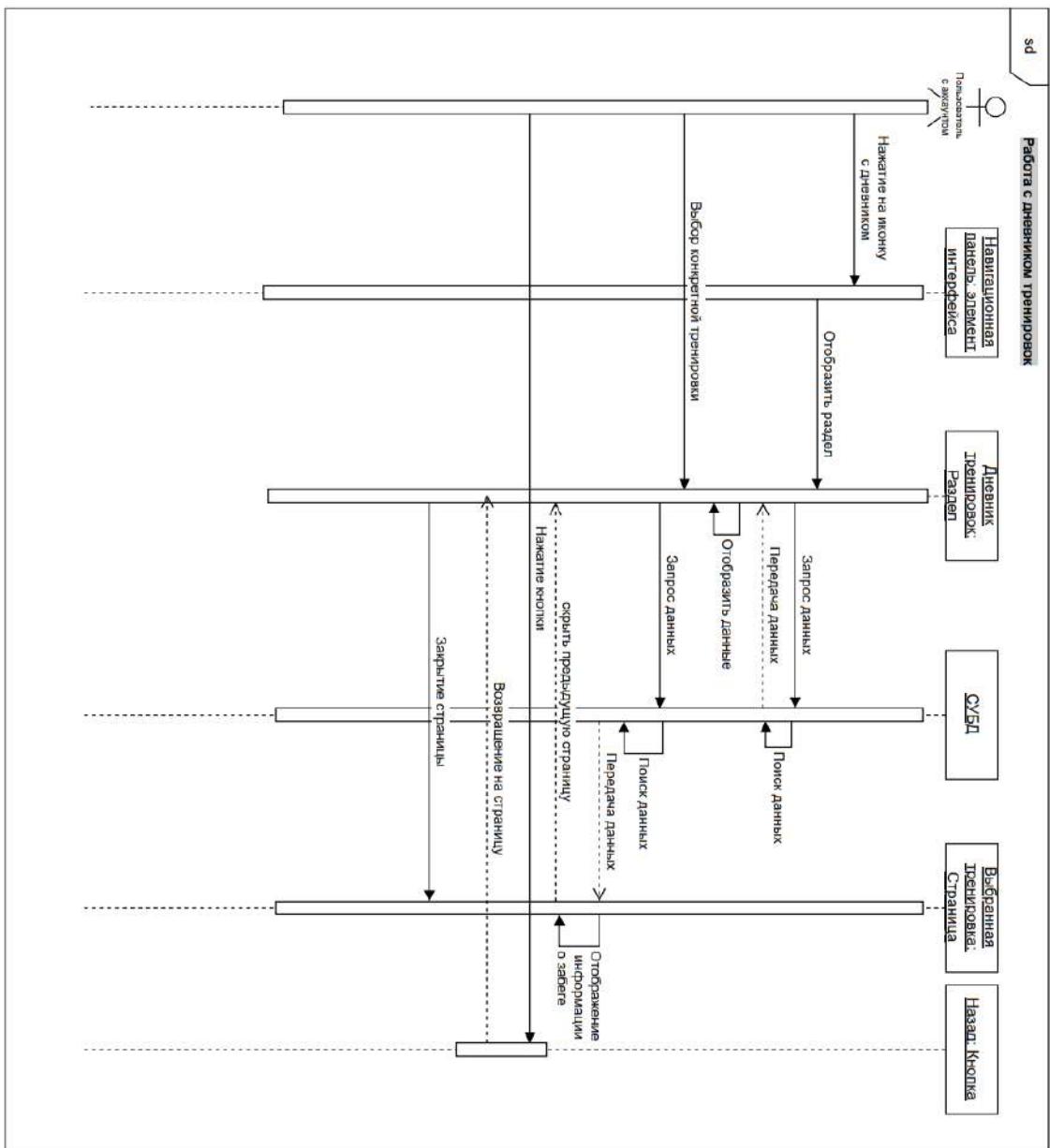


Рисунок 17 — Диаграмма последовательностей для ВИ «Работа с дневником тренировки»

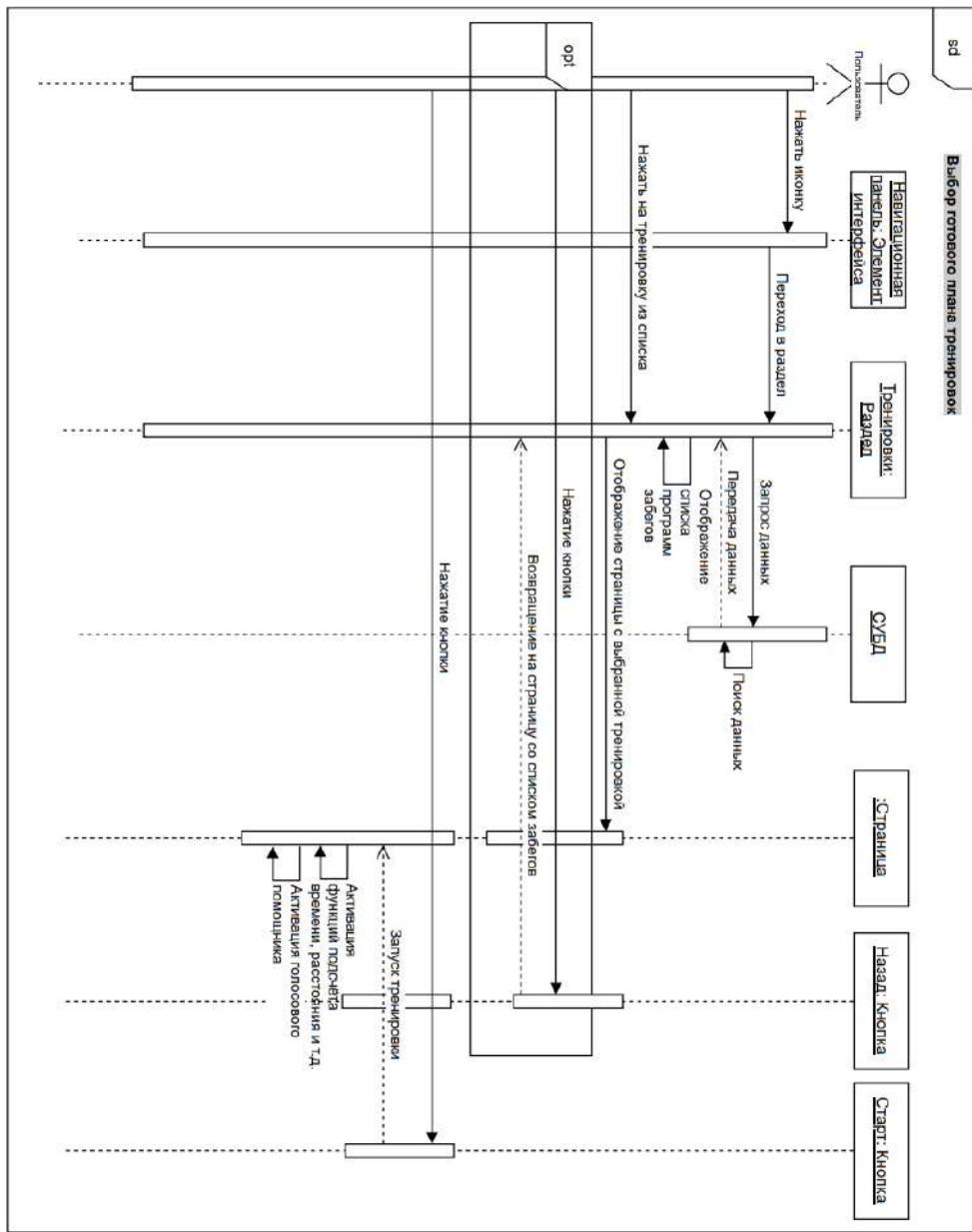


Рисунок 18 — Диаграмма последовательностей для ВИ «Выбор готового плана тренировок»

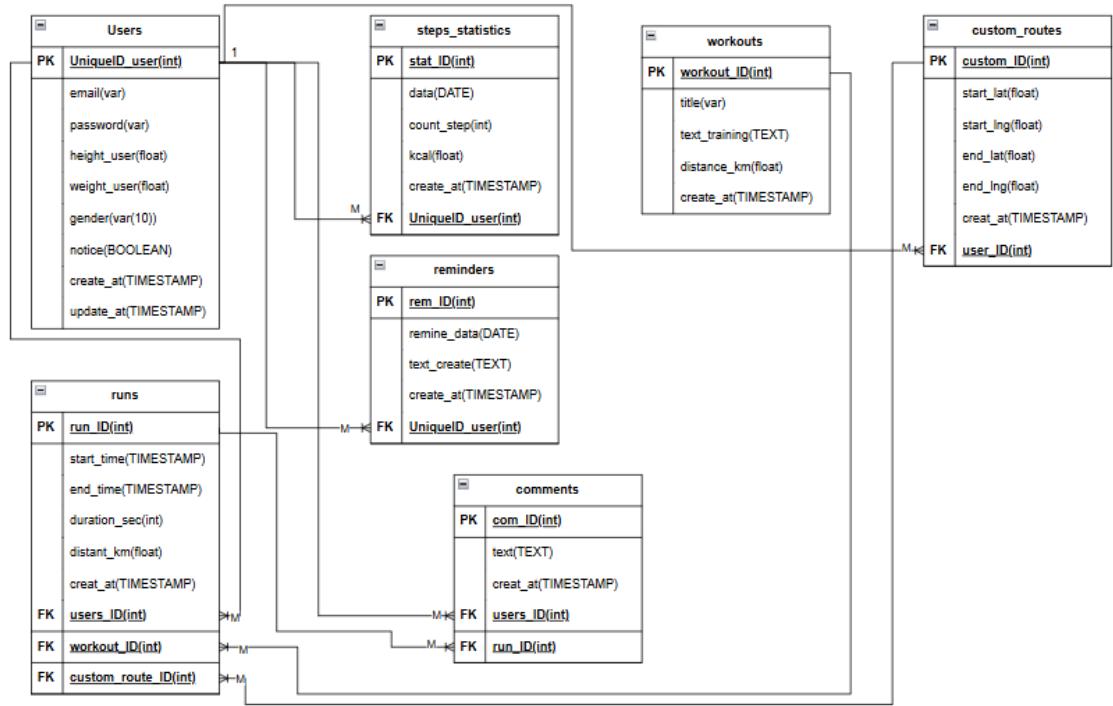


Рисунок 19 — Схема базы данных

Будущая система БД будет состоять из следующих таблиц:

1. Таблица **Users** – будет хранить в себе информацию о пользователях мобильного приложения, как личные данные, так и показатели здоровья, которые влияют на расчёты внутри системы.
2. Таблица **steps_statistics** – хранит в себе данные о статистике шагов и калорий.
3. Таблица **workouts** – готовые тренировки, где: title, text_training, distance_km – это заголовок, текст с описанием тренировки, расстояние дистанции, соответственно.
4. Таблица **custom_routes** – пользовательские маршруты, где: start_lat и start_lng – координаты широты и долготы начальной точки маршрута, end_lat и end_lng - координаты конечной точки маршрута.
5. Таблица **reminders** – напоминания о тренировках. Данные приходят со страницы календаря, где пользователь выбирает дату, в поле text_create

оставляет заметку, в поле create_at хранится информация, когда была создана заметка.

6. Таблица runs – пройденные тренировки (записи в дневнике). Duration_sec, distant_km, creat_at – длительность тренировки, дистанция, время запуска тренировки.
7. Таблица comments – комментарии к пройденным тренировкам. Text, create_at – комментарий к тренировке, дата создания комментария.

В таблицах первый ключ обозначается – PK, а внешний ключ/вторичный – FK. Через них осуществляются связи между данными хранящимися в таблицах.[10]

Спроектированная БД для разрабатываемой ИС играет ключевую роль в обеспечении эффективной работы приложения. На основании хранящихся данных приложение сможет выстраивать точную статистику и отображать её пользователю через интерфейс.

2.6 Диаграмма компонентов

Данная диаграмма позволяет рассмотреть архитектуру разрабатываемой системы, показать зависимости между программными компонентами, где в качестве компонентов выступают реальные физические части системы — библиотеки, модули, классы, исполняемые файлы и т.д. [12]

Ниже на рисунке 20 представлена диаграмма компонентов для разрабатываемой ИС:

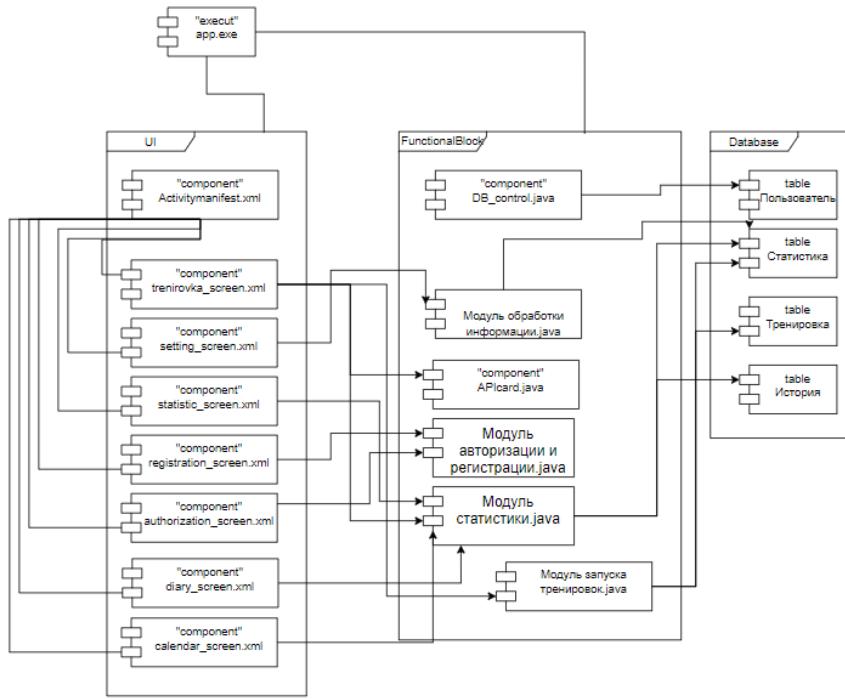


Рисунок 20 — Диаграмма компонентов

На диаграмме выделены три типа компонентов: пользовательские интерфейсы (UI), функциональность системы, компоненты, отвечающие за обработку и хранение данных.

Файл main.exe — это исполняемый файл, он запускает процесс выполнения всей программы.

Компоненты внутри блока UI являются страницами пользовательского интерфейса. Они связываются с блоком, где хранятся модули выполнения функций системы. Внутри данного блока модули, отвечающие за реализацию: функции шагомера, работу и подключение к картам, расчёта статистики, процесса авторизации и регистрации пользователя, работу алгоритмов запуска тренировок, подключение БД.

Компонент APIcard.java используется для взаимодействия с внешними API, для доступа к картам и построения маршрутов тренировок. По сути компоненты на представленной диаграмме — это блоки приложения, которые определяют его функционал.

2.7 Диаграмма развёртывания

В предыдущем блоке мы рассмотрели диаграмму компонентов, которая подробно отобразила нам архитектуру разрабатываемой информационной системы. Следующий тип диаграммы позволит описать физическое расположение компонентов системы на различных узлах и связи между ними.

Чтобы осуществлять связь между приложением и сервером, необходимо иметь общий формат представления данных. Передача данных между приложением и сервером осуществляется с помощью JSON (JavaScript Object Notation).

Приложение, реагирует на действия пользователя, отправляет данные на сервер используя сетевой протокол. Данные передаются в указанном JSON формате серверу, он их обрабатывает и возвращает в том же формате что и получил.

На диаграмме развёртывания мобильного приложения для бега (рисунок 21) представлен комплекс сетевых протоколов, используемых для передачи данных HTTP и WebSocket.

HTTP(HyperText Transfer Protocol) – это один из основных протоколов, позволяющий реализовывать общение между клиентом и сервером. Основной принцип заложенный в HTTP, что при новом действии пользователя отправляется запрос, а сервер в свою очередь получает запрос, обрабатывает и возвращает ответ[24].

Если говорить про WebSocket, то это технология непрерывных запросов между клиентом и сервером, то есть соединение происходит один раз, в отличие от HTTP, где при вызове запроса соединение открывается и закрывается, и данная цепочка действий повторяется до тех пор, пока поступают запросы. Но WebSocket позволяет установить соединение с сервером и оставлять его открытым. Данный подход часто можно встретить в чатах или онлайн играх.

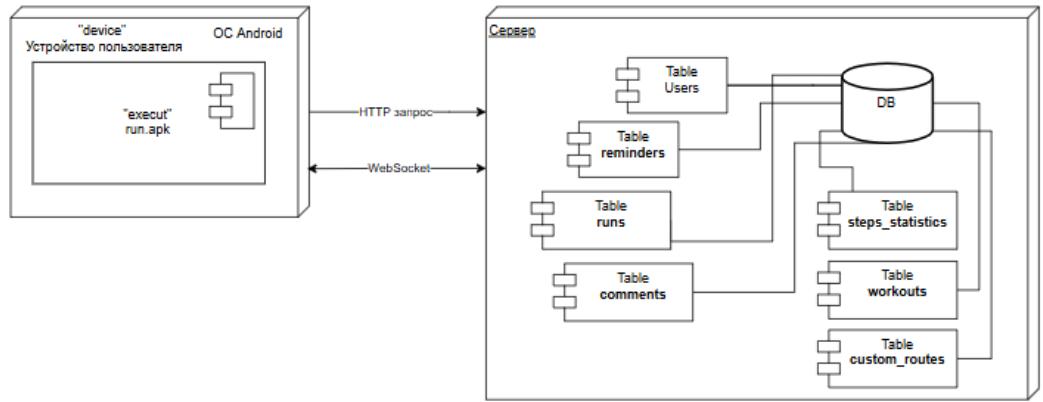


Рисунок 21 — Диаграмма развёртывания

2.8 Создание макета приложения на основе анализа системы

При разработке системы важным аспектом является реализация функционала, при этом с минимальной нагрузкой на графический интерфейс пользователя(ГИП). Когда речь идёт про ГИП, нужно понимать следующие понятия.

Графический интерфейс пользователя – это интерфейс, отображающий логику приложения, через графические элементы, скрывающие под собой заложенные на этапе разработки функциональные возможности системы[16]. При создании ГИП обращаются к UI/UX верстке. UI – это визуальная составляющая будущего продукта, UX – это тип верстки, отвечающий за удобство использования конечного продукта. Существует множество паттернов в UI/UX верстке, но хотелось бы подробнее уделить внимание принципу Якоба Нильсена.

Основная суть метода, изложенного в ряде трудов [19], заключается в том, все потенциальные пользователи наших приложений, веб-сервисов, прежде уже пользовались рядом схожих платформ. В связи с этим у них сформировалось ожидание как будет или должна выглядеть вновь открытая система. Опираясь на данный метод можно сделать вывод, что при разработке макета приложения, следует уделить внимание простоте и удобству восприятия элементов, учитывая опыт аналоговых систем.

В ходе проведенного анализа проектируемой системы в главе 1 были выявлены следующие системные разделы:

- Раздел со статистикой;
- Раздел — календарь активности;
- Раздел с готовыми программами тренировок, включающий в себя личную тренировку с построением маршрута;
- Раздел с настройками;
- Раздел с дневником тренировок.

Прототип приложения представлен ниже на рисунке 22 и включает в себя экраны статистики, отображение функции шагомера, готовые программы с забегами, календарь с планированием тренировок. Он является условной идеальной моделью того как должна выглядеть конечная система, и в рамках ВКР я буду ориентироваться на него в качестве шаблона для разметки функциональных блоков. Остальные макеты мобильного приложения представлены в приложении В.

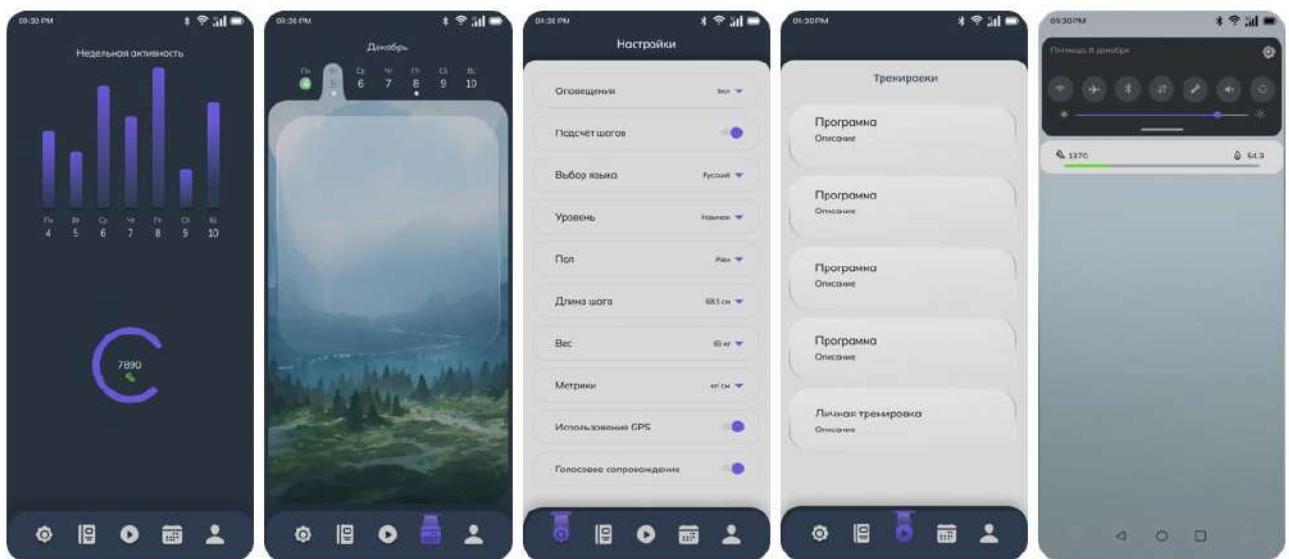


Рисунок 22 — Макет приложения

ГЛАВА 3. РЕАЛИЗАЦИЯ ИНФОРМАЦИОННОЙ СИСТЕМЫ

3.1 Технико-экономическое обоснование реализуемого приложения

Разработка мобильного приложения для бега направлена на удовлетворение большого спектра потребностей пользователей в отслеживании тренировок, показателей здоровья и создании индивидуальных планов с возможностью построения маршрутов, включающая в себя методы их оптимизации. В данном разделе представлено технико-экономическое обоснование проекта.

Технико-экономическое обоснование (ТЭО) проекта – это его неотъемлемая часть, которая важна в процессе разработки и планирования любых реализуемых систем. Это ещё один инструмент, необходимый для отображения показателя эффективности и актуальности разрабатываемого продукта. Оно включает в себя: оценку целесообразности, расчет трудозатрат, сроков разработки, ожидаемых затрат, приблизительных сроков окупаемости и анализ рисков[26].

3.1.1 Обоснование целесообразности разработки проекта

Именного с этого пункта стоит начинать описание технико-экономического обоснования проекта. Он отображает оправданность реализации системы, её дальнейшее сопровождение и наделение ресурсами в виде – людей, времени, финансов.

В условиях современного общества, где здоровье и физическая активность становятся приоритетами для многих людей, мобильные приложения для бега и фитнеса приобретают все большую популярность. С учетом растущего интереса к здоровому образу жизни, разработка мобильного приложения для бега с функцией оптимизации маршрутов представляется не только актуальной, но и необходимой.

Сегодняшний рынок мобильных приложений предлагает разнообразные решения для отслеживания физической активности, однако многие из них не предоставляют пользователям возможности оптимизации маршрутов, что

является важным аспектом для бегунов, стремящихся улучшить свои результаты и разнообразить тренировки. Также немало важно отметить их ограниченный функционал. Разработка мобильного приложения для бега с оптимизацией маршрутов будет направлена на удовлетворение потребностей целевой аудитории, состоящей из любителей бега, профессиональных спортсменов и людей, стремящихся к улучшению своей физической формы. Приложение будет включать функции, такие как создание индивидуальных маршрутов, анализ тренировок, в дальнейшем интеграция с носимыми устройствами.

Целесообразность разработки данного приложения также подтверждается тем, что на рынке существует недостаток специализированных решений, которые бы сочетали в себе функции отслеживания пробежек и оптимизации маршрутов. Это создает уникальную возможность для выхода на рынок с инновационным продуктом, который сможет привлечь внимание пользователей и занять свою нишу.

Кроме того, данная система будет способствовать популяризации бега и здорового образа жизни, что соответствует современным трендам и ожиданиям общества. В условиях увеличения числа людей, занимающихся спортом, приложение сможет стать важным инструментом для достижения их целей, что, в свою очередь, создаст дополнительные возможности для монетизации через подписки, платные функции и партнерские программы.

Таким образом, разработка мобильного приложения для бега с оптимизацией маршрутов является целесообразной и обоснованной, так как оно сможет покрыть большинство потребностей пользователей, заполнит существующий пробел на рынке и способствует популяризации здорового образа жизни. Приложение не только облегчит процесс тренировок, но и сделает его более эффективным и увлекательным для пользователей, проложит маршрут не только к конечной точке, но и к привычке заниматься спортом на постоянной основе.

3.1.2 Оценка уровня качества разрабатываемой информационной системы

Проведя анализ системы основываясь на сравнении с аналоговыми приложениями и их возможностями, необходимо также разобрать техническую составляющую мобильного приложения, выполнив оценку его эксплуатационных параметров. В рамках технико-экономического обоснования существует универсальный метод, позволяющий отразить численный показатель рациональности разработки проекта.

Эксплуатационно-технический уровень (ЭТУ) разрабатываемого продукта – это обобщенная экспертиза его возможностей, эксплуатационных свойств, степени новизны, которые в совокупности дают представление о качестве будущей системы. Для расчёта данного показателя чаще всего прибегают к использованию индекса эксплуатационно-технического уровня $J_{\text{ЭТУ}}$ (1), который появляется в результате суммы частных индексов, в которые входят стандартизованные показатели качества. Для выявления значимости ряда параметров применяется балльно-индексный метод.

$$J_{\text{ЭТУ}} = \sum_{j=1}^n B_j \times X_j , \quad (1)$$

где $J_{\text{ЭТУ}}$ - комплексный показатель качества продукта по группе показателей;

n – число рассматриваемых показателей;

B_j - коэффициент весомости j -го показателя в долях единицы, назначаемый в соответствии с потребностями организации заказчика программного продукта;

X_j - экспертная оценка j -го показателя качества по выбранной шкале оценивания.

Ниже в таблице 6 отражены полученные результаты, выполняемые на основе балльно-индексного метода по пятибалльной шкале оценивания. Все представленные показатели качества выбираются на основе международных стандартов (например, ГОСТ Р ИСО/МЭК 9126-93 [25]), экспертных оценок и отраслевого опыта, конкретных требований, предъявляемых к системе.

Таблица 6 – Сравнение коэффициентов весомости для реализуемого проекта и аналоговой системы

Показатели качества	Коэффициент весомости, B_j	Реализуемая ИС		Аналог	
		X_j	$B_j + X_j$	X_j	$B_j + X_j$
1. Удобство пользовательского интерфейса	0,16	5	0,8	3	0,48
2. Новизна (соответствие современным требованиям)	0,12	5	0,6	3	0,36
3. Соответствие профилю деятельности заказчика	0,15	5	0,75	4	0,6
4. Надёжность и стабильность работы	0,15	4	0,6	4	0,6
5. Безопасность данных	0,14	4	0,56	3	0,42
6. Скорость доступа к данным	0,10	5	0,5	4	0,4
7. Соотношение стоимость возможность	0,10	4	0,4	3	0,3
8. Ресурсная эффективность	0,08	4	0,32	4	0,32
Обобщенный показатель качества $J_{\text{ЭТУ}}$		$J_{\text{ЭТУ1}} = 4,53$		$J_{\text{ЭТУ2}} = 3,48$	

Отношение двух найденных индексов называют коэффициентом технического уровня A_k первого программного продукта по отношению ко второму:

$$A_k = \frac{J_{\text{ЭТУ1}}}{J_{\text{ЭТУ2}}} = \frac{4,53}{3,48} \approx 1,30$$

Проект по разработке мобильного приложения для бега по интегральному показателю качества на 30% превосходит аналог по выбранным критериям. Так как коэффициент технического уровня A_k больше 1, то это говорит нам об

оправданности и целесообразности разработки с учётом выбранных показателей качества.

3.1.3 Планирование работ по разработке информационной системы.

Проведя оценку разрабатываемого мобильного приложения и сравнение его с аналогом, был получен положительный коэффициент качества, что является благоприятными показателями в рамках реализации данной информационной системы.

Следующим шагом становится выделение основных этапов и задач в разработке мобильного приложения и установка ответственных за их выполнение. Планирование работ сводится к закономерности реализации и фиксировании чётких временных рамок, измеряемых в рабочих днях.

Весь комплекс работ над программным продуктом представлен ниже на рисунке 23:

Название этапа	Дата начала	Дата окончания	Длительность	Должность/исполнитель	ЗП (средняя)	Итоги
1. Организация проекта	10.01.2025	20.01.2025	10	Руководитель	3 850,00 ₽	38500,00
1.1 Сбор команды проекта	10.01.2025	20.01.2025	10	Руководитель	3 850,00 ₽	38500,00
2. Предпроектный анализ	21.01.2025	18.02.2025	28	Системный аналитик	5 268,33 ₽	146762,00
2.1 Системно-аналитическое обследование объекта автоматизации	21.01.2025	04.02.2025	14	Системный аналитик	5 161,00 ₽	72254,00
2.2 Анализ и обработка полученной информации	05.02.2025	12.02.2025	7	Системный аналитик	5 161,00 ₽	36127,00
2.3 Разработка технического задания	13.02.2025	20.02.2025	7	Системный аналитик	5 483,00 ₽	38381,00
3. Проектный анализ	21.02.2025	03.03.2025	10	Системный аналитик	5 161,00 ₽	51610,00
3.1 Анализ функциональных блоков(BPMN, DFD)	21.02.2025	03.03.2025	10	Системный аналитик	5 161,00 ₽	51610,00
4. Проектирование	04.03.2025	05.04.2025	30	СА+UX дизайнер	5 118,67 ₽	153560,00
4.1 UML	04.03.2025	14.03.2025	10	Системный аналитик	5 161,00 ₽	51610,00
4.2 ER	15.03.2025	25.03.2025	10	Системный аналитик	5 161,00 ₽	51610,00
4.3 Эскизное проектирование интерфейсов пользователя	26.03.2025	05.04.2025	10	UX дизайнер	5 034,00 ₽	50340,00
5. Реализация	06.04.2025	05.06.2025	60	Программист	5 000,00 ₽	900000,00
5.1 Разработка интерфейсов пользователя	06.04.2025	05.06.2025	60	Программист	5 000,00 ₽	300000,00
5.2 Разработка алгоритмов модулей	06.04.2025	05.06.2025	60	Программист	5 000,00 ₽	300000,00
5.3 Кодирование модулей	06.04.2025	05.06.2025	60	Программист	5 000,00 ₽	300000,00
6. Отладка	06.06.2025	27.06.2025	21	Программист	5 000,00 ₽	105000,00
6.1 Отладка реализованных модулей	06.06.2025	27.06.2025	21	Программист	5 000,00 ₽	105000,00
7. Тестирование	28.06.2025	05.08.2025	34	Тестировщик	4 773,80 ₽	166927,00
7.1 Альфа тестирование	28.06.2025	05.07.2025	7	Тестировщик	5 080,00 ₽	35560,00
7.2 Отчёт по итогу альфа тестирования	06.07.2025	09.07.2025	3	Тестировщик	3 789,00 ₽	11367,00
7.3 Ремонт	10.07.2025	20.07.2025	10	Тестировщик	5 000,00 ₽	50000,00
7.4 Бета тестирование	21.07.2025	28.07.2025	7	Тестировщик	5 000,00 ₽	35000,00
7.5 Отчёт по итогам бета тестирования	29.07.2025	05.08.2025	7	Тестировщик	5 000,00 ₽	35000,00
8. Оформление документации	06.08.2025	06.09.2025	31	Руководитель	3 850,00 ₽	119350,00
8.1 Работа с документами	06.08.2025	06.09.2025	31	Руководитель	3 850,00 ₽	119350,00
			Дней:	239		Итого:
						1 081 709,00 ₽

Рисунок 23 – Календарное планирование проекта

Все этапы, полученные на стадии календарного планирования, обычно принято отображать в виде графиков. Одним из распространённых типов визуализации данных о сроках работ принято считать диаграмму Ганта, представляющая временные ряды в качестве горизонтальных линий на

гистограмме. При таком подходе достаточно чётко видны границы всех запланированных этапов участникам проекта, в том числе заказчику. Также легко выделить набор задач, зависимых друг от друга, либо реализуемых параллельно. Данный вид диаграммы (рисунок 24) вносит ясность и организованность в процессе разработки, позволяет контролировать сдвиги по времени, не совершая существенных изменений в остальных блоках, из-за последовательного выполнения всех в проекте.

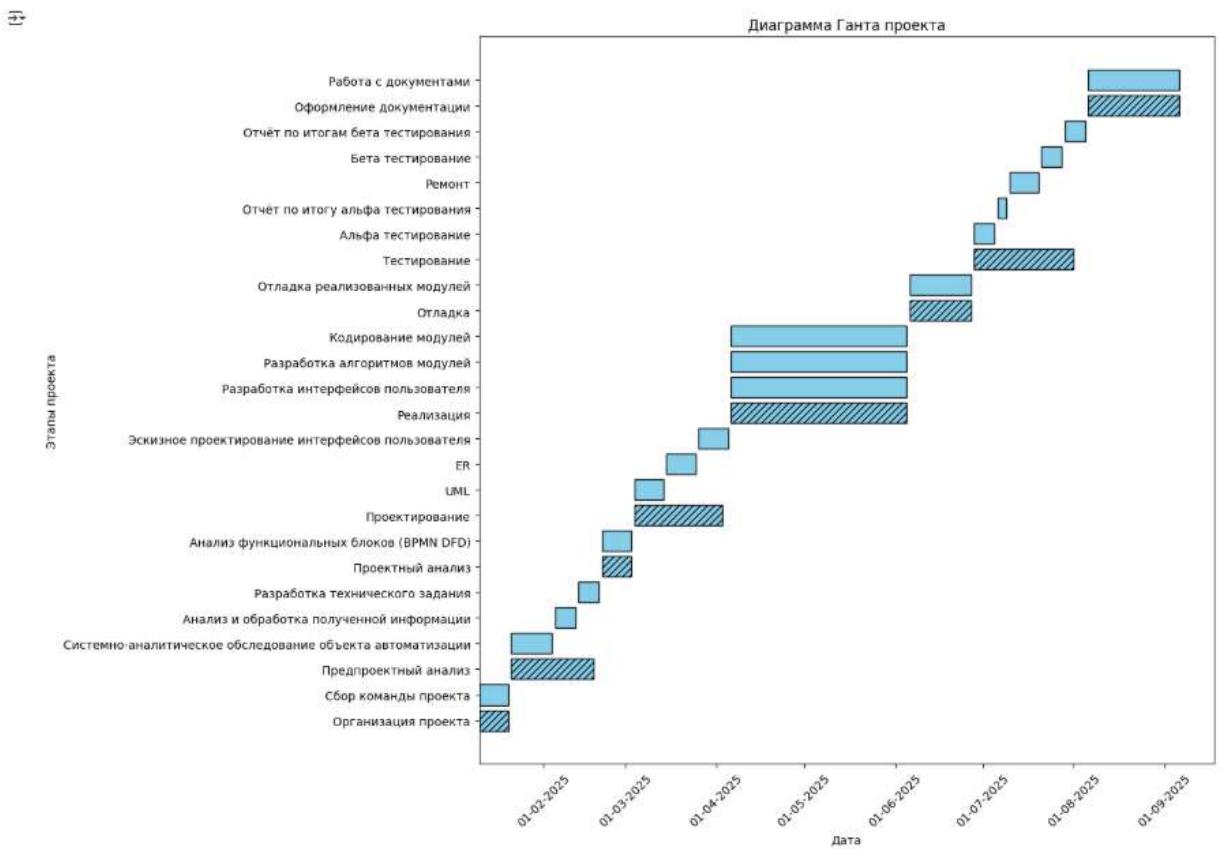


Рисунок 24 – Отображение календарного планирования через диаграмму Ганта

На основе полученных временных рамок были также выделены ответственные за их выполнение. Человеческие ресурсы:

- Руководитель;
- Системный аналитик;
- UX Дизайнер;
- Программист;
- Тестировщик.

В рамках выпускной квалификационной работы человеческие ресурсы ограничены, их составляют: руководитель и программист. Задачи руководителя проекта остаются неизменными, а программист становится ответственным за выполнение обязанностей следующих ролей: системный аналитик, тестировщик, UX дизайнер. Ниже на рисунке 25 представлена таблица с расчётомами человеко-часов, с учётом фиксированной трудовой нормы в виде восьмичасового рабочего дня.

Этап №	Этап	Длительность (дней)	Исполнитель(и)	Человеко-часы (дни × 8 × чел.)
1	Организация проекта	10	Руководитель	80
2	Предпроектный анализ	28	Системный аналитик	224
3	Проектный анализ	10	Системный аналитик	80
4	Проектирование	30	СА + UX дизайнер	240
5	Реализация	60	Программист	480
6	Отладка	21	Программист	168
7	Тестирование	34	Тестировщик	272
8	Работа с документацией	31	Руководитель	248
Итого				1792

Рисунок 25 - Таблица расчёта человека-часов в рамках этапов разработки

3.1.4 Расчёт затрат на разработку мобильного приложения

При расчёте фактических затрат на создание конечного продукта, включают следующие показатели:

- Вложения на проектирование ИС, руб.;
- Вложения на реализацию проекта, руб.

Рассмотрим из чего складывается каждый вид вложений.

Чтобы рассчитать суммарные затраты на проектирование [27] и реализацию необходимо вычислить следующие показатели:

- затраты на материалы;
- затраты на использование машинного времени.

Общая формула по расчёту итоговых затрат на реализацию проектных частей (2):

$$K_p = ((1 + W_d)(1 + W_c) + W_h) \sum_{i=1}^m Z_{oi} + C_M + M_b, \text{ где} \quad (2)$$

M_b – затраты на использование машинного времени;
 C_M – затрат на материалы;
 Z_{oi} – затраты на основную заработную плату работников;
 m – количество работников при процессе разработки ($m=2$);
 W_h – коэффициент, отражающий накладные расходы ($const=0,6$);
 W_c – коэффициент, на социальные нужды, накладываемый на основную и дополнительную заработные платы сотрудников. В соответствии с Федеральным законом 379-ФЗ от 03.12.2011г, страховые взносы в Пенсионный фонд в долях единицы – 0,22, страховые взносы в ФСС – 0,029, страховые взносы в ФФОМС – 0,051, страховые взносы на производственный травматизм – 0,002. Итого: $W_c = 0,302$;
 W_d – коэффициент, накладываемый на дополнительную заработную плату ($const = 0,4$).

Первый этап. Реализация мобильного приложения подразумевает под собой использование вычислительной техники, на которую также приходятся затраты на использование машинного времени. Обращаясь к составленному графику работ в разделе 3.1.3, вычислим M_b как (3):

$$M_b = t_{MB} S_{Mч} K_M, \text{ где} \quad (3)$$

t_{MB} – это машинное время, получаемое из временных затрат по графику планирования;

$$t_{MB} = 920$$

$S_{Mч}$ – стоимость одного часа машинного времени (средний показатель по городу Санкт-Петербургу равен 20 руб./час);

K_M – коэффициент мультипрограммности (отражает долю машинного времени, отводимого на реализацию системы; $const=1$).

$$M_b = 18400$$

Второй этап. Затраты на основную заработную плату (4) для всех работников проекта, с учётом календарного планирования, сформированных этапов и ответственных за них:

$$\sum_{i=1}^m Z_{oi} = Z_{dn} t_i, \text{ где} \quad (4)$$

Z_{dn} – среднедневная заработная плата работника;

t_i – количество дней, затраченных на работу.

Расчёты данного вида затрат приведены ниже на рисунке 26.

Z_{dn}	t_i			
5 023,31 ₽	183	1	Z_{oi}	919 265,73 ₽
3 850,00 ₽	41	2	Z_{oi}	157 850,00 ₽
		Итого:		1 077 115,73 ₽

Рисунок 26 – Таблица с расчётом суммарных расходов на заработную плату сотрудникам

Третий этап. Затраты на материалы включают в себя использование и приобретении ресурсов, необходимых при реализации проекта, но не включают в себя оплату труда и машинное время. Основные позиции и фактическая стоимость представлены в таблице 7.

Таблица 7 – Расходы на материалы

Материалы	Ед. измерения	Количество	Цена за единицу, руб. x Количество
Бумага	пачка	1	250
Тонер для лазерного принтера	шт.	1	750
Принтер	шт.	1	10000
Картриджи	шт.	1	1000
Сервер	шт.	1	58000
Выпуск на платформы App Store, Google Play	шт.	2	9920
Доступ к API-сервисам, Яндекс карты	шт.	1	195000
Ноутбук	шт.	1	70000

Итого	344920
-------	--------

Таким образом вложения на реализацию и проектирование равны:

$$K_p = ((1 + 0,4)(1 + 0,302) + 0,6) \times 1077115,73 + 18400 \\ + 344920 \approx 2972956 \text{ руб}$$

Также, опираясь на полученную таблицу расходов на материалы, следует рассчитать показатель амортизации. Амортизация – это стоимостная оценка затрат на оборудование, технику и иные средства используемые для поддержания работы конечного продукта, при их своевременном износе. Рассчитывается по формуле (5):

$$A_m = O_{\Phi\Pi} / T_{ce} \quad (5)$$

где $O_{\Phi\Pi}$ – это изначальная стоимость оборудования;

T_{ce} – срок эксплуатации устройства в месяцах.

Для реализации данной системы необходимо 1 сервер с примерной стоимостью 58000 руб. и персональный компьютер средней стоимостью в 70000 руб., а также принтер стоимостью 10000 руб. Примерный срок эксплуатации техники составляет 3-5 лет. Возьмём усредненное значение 4 лет и рассчитаем по формуле () затраты на амортизацию за период разработки равный 8 месяцев:

$$A_m = (58000 + 70000 + 10000) / 48 = 2875 \text{ руб.}$$

$$A_m = 2875 \times 8 = 23000 \text{ руб.}$$

Таким образом итоговая стоимость на стадии проектирования и реализации составит – 2995956 руб.

3.1.5 Расчёт эксплуатационных затрат и оценка эффективности разрабатываемого продукта

Как было выявлено из предыдущего параграфа, основными затратами при реализации приложения считаются:

- Фактические затраты (покупка оборудования и материалов, выплата заработной платы, страховые отчисления, мелкие непредвиденные расходы);
- Затраты на период эксплуатации (поддержание оборудования, выплата заработной платы, оплата сервисов для поддержания системы).

Сумма фактических затрат составляет - 2995956 рублей. Исходя из требований к системе ежемесячные расходы будут складываться из интеграции Яндекс Карт – 195000 руб./мес., оплата сервера – 58000 руб./мес.

Теперь приступим к расчёту окупаемости системы и для начала следует учесть возможные варианты получения прибыли:

- Поиск партнеров и интеграция в приложение рекламы и возможностей сервисов – ВК Музыка, Литрес, Додо;
- Подписка на функции внутри приложения;
- Реализация проекта по грантовой системе.

Основной вариант монетизации внутри системы – это:

- Разовая плата за покупку приложения. Учитывая среднюю стоимость продуктов конкурентов от 600 до 800 руб., принято решение установить цену ниже рыночной, что значительно отразится на притоке пользователей. Стоимость проекта составит – 299 рублей;
- Месячная подписка – 699 рублей, годовая – 4199 рублей.

Схема получения прибыли

Доход зависит от числа активных посещений и показов. Будем считать входным порогом при запуске системы - 1000 пользователей. Если говорить о получении прибыли от рекламы в приложении, то:

- 1000 пользователей, каждый видит среднем две рекламы в день, это приблизительно 60000 показов в месяц;
- Средняя цена на рынке за тысячу показов \approx 100 руб.

Отсюда получаем доход:

$$\text{Доход} = 60 \times 100 = 6000 \text{ руб./мес}$$

Партнерская система получения прибыли:

- Литрес: % от продажи книг по вашей ссылке;
- ВК Музыка: % с подписок/прослушиваний;
- Додо: % заказов по промокоду.

В среднем все выше представленные организации приносят 5–15% от суммы транзакции. Предположим, что через месяц внутри приложения пройдет:

- Литрес: 30 покупок \times 50 руб. комиссия = 1500 руб.
- ВК Музыка: 10 подписок \times 100 руб. комиссия = 1000 руб.
- Додо: 50 заказов \times 50 руб. комиссия = 2500 руб.

Итого, партнерская система принесёт доходов: 5000 руб./мес. Если суммировать все вышеизложенные варианты получения ежемесячной прибыли, то получается - 11000 руб.

Теперь, воспользовавшись классической формулой расчёта окупаемости проекта (6), получим приблизительный период.

$$T = FC/IC, \text{ где } FC - \text{фактические затраты, а } IC - \text{ежемесячный доход. (6)}$$

Допустим, к приложению имеют доступ 1000 пользователей по ежемесячной подписке (699 руб.):

- Доход от подписки: 699000 руб./мес.
- Доход от рекламы и партнёрских соглашений: 11000 руб./мес.
- Общий доход: 710000 руб./мес.
- Чистая прибыль: $710000 - 253000 = 457000$ руб./мес.
- Срок окупаемости: $2995956 / 457000 \approx 6,5$ месяца

Вывод

Проведённое технико-экономическое обоснование проекта является важным инструментом для оценки, планирования и управления проектом. Оно

помогает обеспечить успешную реализацию, минимизировать риски и максимизировать эффективность разработки информационной системы, рассчитать приблизительные сроки окупаемости, а также позволяет оценить потенциальную прибыльность проекта, привлечь инвестиции и обеспечить успешное внедрение мобильного приложения на рынок.

3.2 Реализация мобильного приложения

Ранее, анализируя лидирующие операционные системы, был сделан вывод о важности разработки кроссплатформенного мобильного приложения. Несмотря на то что на данный момент основными языками для мобильной разработки выделяют – Kotlin, Java, Python, но код написанный на этих языках обычно реализует логику работы приложения для определённой операционной системы. Поэтому, основным языком программирования для развёртывания проекта стал – Dart[18].

Dart – это современный, быстро развивающийся на данный момент объектно-ориентированный язык программирования с чистым и лаконичным синтаксисом. Позволяет разрабатывать приложения сразу для двух операционных систем, при этом используя один и тот же код. Одним из основных отличий является асинхронное программирование с поддержкой большого количества библиотек, отвечающих за надёжность и масштабируемость приложения. Frontend и backend части написаны на Dart. Данный язык позволяет реализовывать это одновременно. Работая над frontend частью был использован фреймворк Flutter[23], где каждая страница приложения представлена отдельным виджетом. Backend логика по работе с картами выполнена через библиотеку yandex_maps_mapkit[21], yandex_maps_navikit[22] и встроенные методы описанные в официальной документации. В качестве среды разработки была выбрана Android Studio.

При выявлении ведущей функциональной возможности будущей системы - оптимальное построение маршрутов во время забегов, необходимо было выполнить ряд действий:

- Выяснить по какому алгоритму рассчитывается и выстраивается маршрут на карте, проведя анализ уже существующие картографических сервисов;
- Проанализировать алгоритм, выявить основные типы данных и объекты, которые он обрабатывает.

В качестве готового решения для отображения карт был выбран сервис Яндекс Карты. Он работает на основе алгоритма Дейкстры и решает поставленную задачу в построении оптимальных маршрутов.

Основная суть заключается в том, что кратчайший путь из условной точки А в точку Б рассчитывается через анализ близлежащих соседних узлов, вычисление расстояния до них и выбора минимального значения. При этом в качестве значений в картографических сервисах могут выступать не только метрики расстояния (метры, километры), но и временные затраты, которые складываются из ряда ограничений - пробки, ремонтные работы, покрытие дороги и т.д. Если говорить простыми словами - это всеми известный метод графов, используемый в компьютерных сетях, при решении ряда алгебраических и геометрических задач, распределительных сетях (энергетике). Вершинами выступают ключевые точки/объекты на карте, перекрестки дорог.

Чтобы подробнее понять данный метод и углубиться в его обоснование в виде теоремы, понять вывод доказательств, иметь базовое представление о его работе необходимо знать фундаментальные основы алгоритмов для работы с графиками ([30] пункт 24.3, глава 25).

Если возвращаться к причине выбора сервиса Яндекс для отображения и работы с картами, то во-первых – это повышенная детализация, в отличии от того же OpenStreetMap, во-вторых вероятность сбоя меньше по сравнению с аналогами, которые в любой момент могут уйти с нашего рынка. Также следует учитывать нефункциональные требования, предъявляемые к проекту, именно надежность системы и бесперебойная работа стоят в приоритете при разработке конечного продукта.

Теперь об алгоритме выстраивания маршрута. Для более ясного понимания о взаимодействии пользователя и системы, действий, выполняемых в рамках функционирования мобильного приложения, составлена блок-схема (рисунок 27).

Если говорить кратко, то интерфейс Яндекс карт принимает на вход стартовую точку от пользователя и желаемое расстояние/время. Алгоритм внутри интегрированной карты уже реализует методы маршрутизации по принципу Дейкстры, однако было учтено что в него не заложена функция построения круговых маршрутов, поэтому была реализована модель данных описывающая класс с заданными параметрами:

```
class RouteData {  
    final Point startPoint;//инициализация переменной  
    final double targetDistance;//дистанция указанная пользователем  
    List<Point> routePoints = [];//лист для сгенерированных точек вокруг  
    //выбранной стартовой  
    RouteData(this.startPoint, this.targetDistance);//запись сгенерированных  
    //точек в routePoints  
}
```

Вся работа по настройке и интеграции сервиса в разрабатываемое мобильное приложение сводилась к установлению соединения с API Яндекс карт. Её инициализация через конфигурационный файл проекта, с установлением версии SDK[20], геолокатора, для определения местоположения пользователя, установка провайдера, для управления состоянием при передаче запросов из файла, описывающего алгоритм работы с картами и страницей где реализован frontend. Модель описанная выше взаимодействовала в связке рядом классов и методов, применяемые для работы с картами и вывода желаемого результата. Получение готовых маршрутов реализовывалось через запрос requestRoutes, внутри которого задавались параметры, такие как - DrivingOptions, позволил реализовать функцию генерирования нескольких маршрутов, через

параметр routesCount. Для отображения линий и установленных точек в пользовательском интерфейсе использовался метод MapObject.

Возвращение в стартовую точку при прохождении маршрута для бега имеет несколько логичных закономерностей. Во-первых, это обеспечивает удобство: бегун может начать и закончить тренировку в одном месте, что упрощает организацию времени и логистику. Такой подход позволяет лучше контролировать дистанцию. Кроме того, завершение пробежки в стартовой точке позволяет избежать потенциальных проблем, связанных с незнакомыми маршрутами, такими как потеря ориентации или необходимость искать альтернативные пути обратно. Это создает возможность для более безопасных тренировок, так как бегун может выбирать знакомые и проверенные маршруты, минимизируя риски.

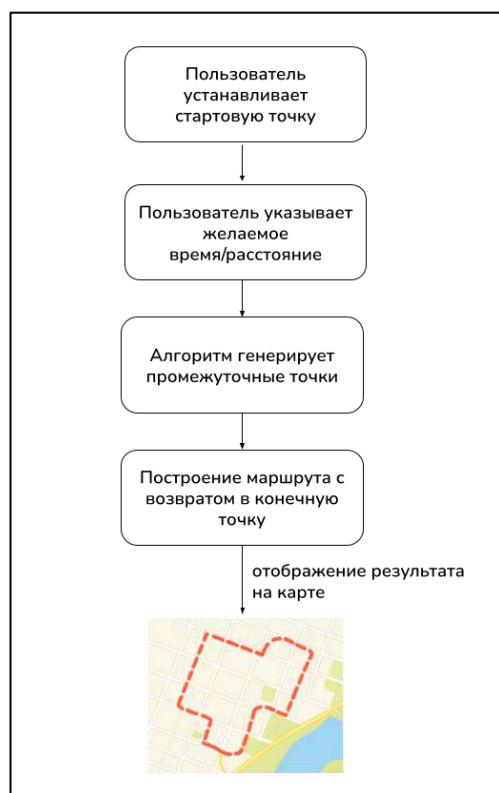


Рисунок 27 – Блок-схема алгоритма построения маршрутов

Хранение получаемых данных в системе было реализовано с помощью SQLite, которая, как и библиотеки по работе с картами, была инициализирована через конфигурационный файл - pubspec.yaml. Вся архитектура и связи созданы

опинаясь на схему базы данных спроектированную в главе 2. Работа с информацией, поступаемой и запрашиваемой от пользователя, осуществлялась, используя CRUD запросы. Также было реализовано шифрование паролей с помощью библиотеки `scrypt`, которая даёт возможность выполнять криптографические алгоритмы хеширования, например - SHA-256, MD5 и HMAC.

В рамках написания выпускной квалификационной работы спроектирована многоуровневая система со сложной логикой, которая была декомпозирована, выявлены уникальные функции на основе которых реализована практическая часть. Тестирование и отладка реализованных разделов и алгоритма проводились вручную. Разработаны следующие компоненты внутри приложения для бега, полученные результаты представлены в приложении С:

- Страница «Авторизации и регистрации»;
- Страница «Главная»;
- Раздел «Тренировки»;
- Раздел «Дневник»;
- Раздел «Календарь».

ЗАКЛЮЧЕНИЕ

В рамках выполнения выпускной квалификационной работы была достигнута поставленная цель и решены все связанные с ней задачи:

- Анализ предметной области приложения.
- Выявление бизнес требований применимых в отношении разрабатываемого мобильного приложения.
- Проектирование приложения, опираясь на проведенный анализ.
- Реализация информационной системы, опираясь на технико-экономическое обоснование проводимых работ по разработке.

В ходе проводимого анализа предметной области, были выделены аналоговые системы, их функциональные особенности. Применен сравнительный метод оценки конкурентных продуктов, позволивший определить уникальные возможности разрабатываемого приложения.

Проведённый анализ включал метод SWOT, позволивший выявить сильные и слабые стороны приложения, что отразилось на понимании путей развития будущей системы. Выявлена степень актуальности разрабатываемого программного продукта через анализ рынка мобильных приложений со смежной предметной областью на основе сравнения предлагаемых решений с концепцией будущей системы. Сформированы группы потенциальных пользователей, установлены функциональные и нефункциональные требования к приложению.

Выделены бизнес-процессы и потоки данных, необходимые для понимания архитектуры будущего программного продукта.

Опираясь на проведённый анализ, на этапе проектирования были созданы диаграммы последовательностей, компонентов, развертывания, описанные на языке UML. Также описаны сценарии вариантов использования, отражающие взаимодействие системы и претендентов. В рамках проектирования была сформирована схема хранения данных, основанная на функциональных возможностях системы, потоках данных отраженных в рамках анализа на диаграммах нотации DFD.

Было произведено технико-экономическое обоснование целесообразности разработки спроектированной системы, с использованием метода вычисления коэффициента весомости. Описаны фактические затраты на реализацию и сопровождение будущего продукта, рассчитаны сроки окупаемости.

На этапе реализации был рассмотрен алгоритм маршрутизации, сформированы методы и правила построения дистанций и интегрированы в проект. Архитектура спроектированного приложения многоуровневая и сложна для реализации в полном объеме на начальном этапе разработки, поэтому произведена сборка MVP (минимально жизнеспособный продукт) приложения, с основными функциональными возможностями описанной системы, отвечающий всем заявленным требованиям в рамках ВКР, с возможностью масштабирования в будущем.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ И ЛИТЕРАТУРЫ

1. Использование диаграммы вариантов использования UML при проектировании программного обеспечения [Электронный ресурс]: <https://habr.com/ru/articles/566218/> (дата обращения: 12.02.25)
2. Блок программиста. Основы UML — диаграммы использования (use—case) [Электронный ресурс]: <https://pro—prof.com/archives/2594> (дата обращения: 12.02.25)
3. Разница между INCLUDE и EXTEND диаграмма вариантов использования (use case) [Электронный ресурс]: <https://fkn.ktu10.com/?q=node/2236> (дата обращения: 13.02.25)
4. Моделирование на UML [Электронный ресурс]: <http://book.uml3.ru/content> (дата обращения: 16.02.25)
5. Use Case. Инструкция по работе со сценариями использования для молодого системного аналитика [Электронный ресурс]: <https://habr.com/ru/articles/699522/> (дата обращения: 17.02.25)
6. Нотация и семантика языка UML. Лекция 8: Элементы графической нотации диаграммы последовательности [Электронный ресурс]: <https://intuit.ru/studies/courses/32/32/lecture/1014?page=2> (дата обращения: 17.02.25)
7. Учебное пособие по диаграммам последовательностей: полное руководство с примерами [Электронный ресурс]: <https://creately.com/blog/ru/> (дата обращения: 17.02.25)
8. Справочник UML. Объектно—ориентированное проектирование [Электронный ресурс]: <https://openu.ru/Books/UML/Reference.asp> (дата обращения: 18.02.25)
9. Техноблог. Диаграмма последовательности (sequence diagrams) [Электронный ресурс]: https://itonboard.ru/analysis/394—diagramma_posledovatelnosti_sequence_diagrams.uml/ (дата обращения: 18.02.25)

10. Связи БД [Электронный ресурс]: <https://habr.com/ru/articles/488054/> (дата обращения: 24.02.25)
11. Основы UML. Краткое руководство по стандартному языку объектного моделирования [Электронная версия]: Фаулер М. UML. Основы, 3—е издание. — Пер. с англ. — СПб: Символ—Плюс, 2004. — 192 с., ил.
12. Диаграмма компонентов [Электронный ресурс]: <https://habr.com/ru/articles/756552/> (дата обращения: 21.02.25)
13. FitnesData. Рост популярности любительских забегов в России: тренды, перспективы, связь с фитнесом [Электронный ресурс]: <https://clck.ru/3LsnuW> (дата обращения: 26.04.25)
14. Мартыщенко С.О. Актуальность онлайн фитнес-приложений в России [Электронный ресурс]: <https://cyberleninka.ru/article/n/aktualnost-onlayn-fitness-prilozheniy-v-rossii/viewer> (дата обращения: 28.04.25)
15. ГОСТ 19.201-78 Единая система программной документации. Техническое задание. Требование к содержанию и оформлению.
16. Гультьяев, А.К. Проектирование и дизайн пользовательского интерфейса / А.К. Гультьяев, В.А. Машин. – СПб: КОРОНА прнт, 2010. – 352 с.
17. Основы технологий баз данных: учебное пособие / Б. А. Новиков, Е. А. Горшкова, Н. Г. Графеева; под ред. Е. В. Рогова. — 2-е изд. — М.: ДМК Пресс, 2020. — 582 с.
18. Документация по Dart [Электронный ресурс]: <https://dart.dev/docs> (дата обращения: 12.03.25)
19. Яблонски Дж. Законы UX-дизайна. – 2-е изд.: Пер. с англ. – Астана: АЛИСТ, 2025. – 176 с.: ил.
20. MapKit Mobile SDK Documentation [Электронный ресурс]: <https://yandex.ru/maps-api/docs/mapkit/index.html> (дата обращения: 25.04.25)
21. Yandex_Mapkit. Плагин Flutter для отображения карт Яндекса на iOS и Android. [Электронный ресурс]:https://pub.dev/packages/yandex_mapkit (дата обращения: 28.04.25)

22. Yandex_Maps_Navikit. Плагин Flutter для отображения карт Яндекса на iOS и Android [Электронный ресурс]: https://pub.dev/packages/yandex_maps_navikit/versions (дата обращения: 28.04.25)
23. Документация Flutter [Электронный ресурс]: <https://docs.flutter.dev/> (дата обращения: 20.04.25)
24. Учебник по протоколу HTTP [Электронный ресурс]: <https://code.mu/ru/internet/protocol/http/> (дата обращения: 11.03.25)
25. ГОСТ Р ИСО/МЭК 9126-93. Информационная технология. Оценка программного продукта. Характеристики качества и руководство по их применению.
26. Технико-экономическое обоснование выполнения проекта: методическое пособие / С.Л. Миньков. – Томск: ТУСУР, 2014. – 30 с. [Электронный ресурс]: https://asu.tusur.ru/learning/spec080801/d26/s080801_d26_work.pdf (дата обращения: 5.05.25)
27. Абрамичева, Т.В. Оценка стоимости автоматизированных информационных систем: методические указания / Т. В. Абрамичева, А. В. Павловская, Е. В. Истомина. – Ухта : УГТУ, 2014. – 56 с
28. Маркетинговые исследования: конкурентный анализ: учебное пособие / В.В. Еремин, Т.С. Селевич; Томский политехнический университет. – Томск: Изд-во Томского политехнического университета, 2009. – 200 с.
29. Что такое SWOT-анализ [Электронный ресурс]: <https://kaiten.ru/blog/swot-analiz/> (дата обращения: 19.03.25)
30. Кормен, Томас Х., Лейзерсон, Чарльз И., Ривест, Рональд Л., Штайн, Клиффорд. Алгоритмы: построение и анализ, 2-е издание. : Пер. с англ. — М. : Издательский дом “Вильямс”, 2011. — 1296 с. : ил. — Парал. тит. англ.

ПРИЛОЖЕНИЕ А
СЦЕНАРИИ ВАРИАНТОВ ИСПОЛЬЗОВАНИЯ

Главный раздел сценария выполнения варианта использования "Авторизация"	
Вариант использования	Авторизация
Актеры	Пользователь(авторизованный)
Краткое описание	Вход в приложение
1 Цель	Войти в систему. Получение доступа пользователю к расширенным функциональным возможностям системы.
Тип	Базовый вариант использования
Ссылки на другие варианты использования	отсутствуют

Рисунок А.1 – Главный раздел ВИ «Авторизация»

Раздел Типичный ход событий сценария выполнения варианта использования "Авторизация"	
Действия актёров	Отклик системы
1. Открывает приложение	2. Отображает страницу авторизации
3. Вносит личные данные в поля: пароль, логин	
4. Нажимает кнопку "Войти"	<u>1. Исключение. Внесены не все данные</u> 5. Запрос к БД 6. Проверка введённых данных
	<u>2. Исключение. Неверный логин или пароль</u> 7. Загрузка приветственного окна 8. Открытие главной страницы

Рисунок А.2 - Типичный ход ВИ «Авторизация»

ИСКЛЮЧЕНИЯ	
Раздел Исключения сценария выполнения варианта использования "Авторизация"	
Действия актёров	Отклик системы
1. Исключение. Внесены не все данные	
6. Ввод данных в пустые поля	5. Подсветит поля с пропущенными данными. Выведет инфостроки о пустых полях ввода
4. Нажимает кнопку "Войти"	7. Снятие подсветки с полей ввода и скрытие инфострек
2. Исключение. Неверный логин или пароль	7. Система выводит информационное окно об ошибке
<u>2.1 Исключение. Пользователь забыл пароль</u>	
8. Повторный ввод данных	
4. Нажимает кнопку "Войти"	
2.1 Исключение. Пользователь забыл пароль	
8. Нажмёт на кнопку "Восстановить пароль"	9. Открыть страницу для ввода нового пароля
10. Введёт почту	
11. Введёт новый пароль	
12. Введёт повторно новый пароль	
13. Нажмёт "Далее"	14. Проверка данных <u>Исключение. Данные введены некорректно</u>
	15. Отправка сообщения с кодом подтверждения
	16. Выведет страницу для ввода кода подтверждения
<u>2.1.1. Исключение. Код не пришёл</u>	
17. Введёт код подтверждения	18. Проверка правильности введённого кода
	<u>2.1.2. Исключение. Код неверный</u>
	19. Обновление данных в БД
	20. Перевод пользователя на главную страницу

Рисунок А.3 – Исключение ВИ «Авторизация»

2.1.1. Исключение. Код не пришёл
16. Запросить код повторно, нажать на кнопку
2.1.2. Исключение. Код неверный
19. Запросить код повторно
17. Отправка письма с кодом повторно
18. Вывод сообщения в инфостреке о неверном коде
20. Отправка письма с кодом повторно

Рисунок А.4 - Исключение ВИ «Авторизация»

Главный раздел сценария выполнения варианта использования "Регистрация"	
	Вариант использования
2	Актеры
	Краткое описание
	Цель
	Тип
	Ссылки на другие варианты использования
	Регистрация Пользователь(неавторизованный) Создание аккаунта пользователя Создание аккаунта. Получение доступа к аккаунту пользователя для использования функционала системы. Базовый вариант использования отсутствуют

Рисунок А.5 - Главный раздел ВИ «Регистрация»

Раздел Типичный ход событий сценария выполнения варианта использования "Регистрация"	
Действия актёров	Отклик системы
1. Нажать на кнопку "Регистрация"	2. Открытие страницы регистрации
3. Ввод данных для регистрации	
4. Подтверждение ввёденных данных. Нажать на кнопку "Далее"	5. Валидация данных <u>Исключение. Внесены не все данные</u> <u>1. Исключение. Данные введены некорректно</u> <u>2. Исключение. Пользователь с данной почтой уже существует</u> 6. Отображение страницы с кодом подтверждения 7. Отправка сообщения с кодом подтверждения на почту
Исключение. Код не пришёл	
8. Ввести код подтверждения	9. Проверка правильности введенного кода <u>Исключение. Код неверный</u> 10. Создание новой учётной записи в БД 11. Переводит пользователя на главную страницу

Рисунок А.6 - Типичный ход ВИ «Регистрация»

ИСКЛЮЧЕНИЯ	
Раздел Исключения сценария выполнения варианта использования "Регистрация"	
Действия актёров	Отклик системы
1. Исключение. Данные введены некорректно	6. Вывести инфострочку с допустимыми символами и правилами
7. Повторный ввод данных	
4. Подтверждение ввёденных данных. Нажать на кнопку "Далее"	2. Исключение. Пользователь с данной почтой уже существует 6. Вывести сообщение о наличии аккаунта с данной почтой
Исключение. Ввести новую почту(в одно действие)	
7. Вернуться на страницу с авторизацией .Нажать на кнопку "Авторизация"	
8. Нажать на кнопку "Восстановить пароль"	9. Открыть страницу для ввода нового пароля
10. Введёт почту	
11. Введёт новый пароль	
12. Введёт повторно новый пароль	
13. Нажмёт "Далее"	1. Исключение. Данные введены некорректно 14. Отправка сообщения с кодом подтверждения 15. Выведет страницу для ввода кода подтверждения
Исключение. Код не пришёл	
16. Введёт код подтверждения	17. Проверка правильности введённого кода Исключение. Код неверный

Рисунок А.7 - Исключение ВИ «Регистрация»

Главный раздел сценария выполнения варианта использования "Запланировать тренировку"	
	Вариант использования
5	Запланировать тренировку
	Актеры
	Пользователь(авторизованный)
	Краткое описание
	Создать заметку о запланированной тренировки в календаре
	Цель
	Оповещение/напоминание о запланированной пробежке, с указанием даты, времени и типа пробежки
	Тип
	Зависимый вариант использования
	Ссылки на другие варианты использования
	Расширяется: Отменить тренировку, Установить оповещение.

Рисунок А.8 - Главный раздел ВИ «Запланировать тренировку»

Раздел Типичный ход событий сценария выполнения варианта использования "Запланировать тренировку"	
Действия актёров	Отклик системы
1. Выбор раздела "Запланировать тренировку"	2. Отображение выбранной страницы
3. Добавить тренировку	4. Открытие формы для занесения данных о тренировке
5. Нажатие на активную строку "Дата и время", раскрывающийся список	6. Отображение фрейма с календарём и временным таблом
7. Установка даты и времени	
8. Скрытие раскрывающегося списка	9. Сохранение занесённых данных
11. Выбор типа тренировки	10. Скроет выпавший, в раскрывающемся списке фрейм
13. Указание типа тренировки	12. Открытие фрейма с доступными типами тренировок
14. Скрытие раскрывающегося списка	15. Сохранение занесённых данных
17. Подтверждение всех внесённых данных. Нажимает "Ok"	16. Закрытие фрейма
	1. Исключение. Если будет выбрана дата, которая уже прошла
	2. Исключение. Внесены не все данные
	18. Сохранение данных
	19. Отправка данных в БД
	20. Появление метки в календаре и данных о тренировке

Рисунок А.9 - Типичный ход ВИ «Запланировать тренировку»

ИСКЛЮЧЕНИЯ	
Раздел Исключения сценария выполнения варианта использования "Запланировать тренировку"	
Действия актёров	Отклик системы
	1. Исключение. Если будет выбрана дата, которая уже прошла
	18. Раскроет список "Дата и время"
	19. Выведет инфострочку об некорректно введённой дате
20. Выбор новой даты	
17. Подтверждение всех внесённых данных. Нажимает "Ok"	
	2. Исключение. Внесены не все данные
	18. Подсветит невыбранные/незаполненные поля. Не даст сохранить заметку
19. Дозаполнит поля	
17. Подтверждение всех внесённых данных. Нажимает "Ok"	

Рисунок А.10 - Исключение ВИ «Запланировать тренировку»

Главный раздел сценария выполнения варианта использования "Выбор готового плана тренировки"	
	Вариант использования
6	Выбор готового плана тренировки
	Актеры
	Пользователь, Пользователь(неавторизованный)
	Краткое описание
	Открыть перечень уже имеющихся тренировок и выбрать готовый комплекс
	Цель
	Запуск и выполнение уже сформированного системой комплекса забега
	Тип
	Базовый вариант использования
	Ссылки на другие варианты использования
	Расширяется: Составлением личной тренировки

Рисунок А.11 - Главный раздел ВИ «Выбор готового плана тренировок»

Раздел Типичный ход событий сценария выполнения варианта использования "Выбор готового плана тренировки"	
Действия актёров	Отклик системы
1. Выбор раздела с готовыми тренировками	2. Запрос к бд 3. Получение ответа от БД 4. Отобразит список программ забега
5. Выбор тренировки	6. Перейдет на новую страницу. 7. Загрузит информацию по тренировке
1. Исключение. Вернуться к списку с тренировками	
8. Запуск тренировки	9. Соединение с GPS 10. Запуск функций подсчёта времени, расстояния, калорий 11. Запуск голосового сопровождения

Рисунок А.12 - Типичный ход ВИ «Выбор готового плана тренировок»

ИСКЛЮЧЕНИЯ	
Раздел Исключения сценария выполнения варианта использования "Выбор готового плана тренировки"	
Действия актёров	Отклик системы
1. Исключение. Вернуться к списку с тренировками	
1. Нажать кнопку "Назад"	2. Отобразить страницу со списком программ забега

Рисунок А.13 Исключение ВИ «Выбор готового плана тренировок»

Главный раздел сценария выполнения варианта использования "Работа с картами/маршрутами"	
<u>Вариант использования</u>	Работа с картами/маршрутами
<u>Актеры</u>	Пользователь(авторизованный)
<u>Краткое описание</u>	Открытие карт в приложении, возможность установки маршрутов и использования всего функционала карт
<u>Цель</u>	Открытие карт , просмотр местности, просматривать другие маршруты
<u>Тип</u>	Зависимый вариант использования
<u>Ссылки на другие варианты использования</u>	Расширяется: Сохранение пройденного маршрута. Включает: Предоставление доступа к картам

Рисунок А.14 - Главный раздел ВИ «Работа с картами/маршрутами»

Раздел Типичный ход событий сценария выполнения варианта использования "Работа с картами/маршрутами"	
Действия актёров	Отклик системы
1. Нажатие на кнопку "Построить маршрут"	2. Отправка запроса на подключение к API карт
3. Картографический сервис получает запрос	
4. Картографический сервис даёт ответ системе	5. Получение ответа от картографического сервиса 1. Исключение. Отсутствие соединения с картографическим сервисом.
	6. Установление связи с картографическим сервисом
	7. Переход на страницу с картой
2. Исключение. Сообщение о включении распознавания геолокации	
8. Пользователь нажимает кнопку, отвечающую за режим установки начальной конечной точки забега	
9. Установка точек	
10. Нажать на "Подтвердить"	
3. Исключение. Изменить положение точек на карте. Переставить точку(одно действие)	
	11. Запускает алгоритм построения оптимального маршрута
	12. Отображает графически построенный маршрут
13. Запустить тренировку по данному маршруту. Нажатие кнопки "Старт"	

Рисунок А.15 - Типичный ход ВИ «Работа с картами/маршрутами»

ИСКЛЮЧЕНИЯ	
Раздел Исключения сценария выполнения варианта использования "Работа с картами/маршрутами"	
Действия актёров	Отклик системы
7. Нажимает на "Обновить карту"	1. Исключение. Отсутствие соединения с картографическим сервисом. 6. В окне с картой выдаётся ошибка загрузки 2. Отправка запроса на подключение к API карт
2. Исключение. Сообщение о включении распознавания геолокации	
8. Запрос о геолокации пользователя от картографического сервиса	9. Открытие информационного фрейма с сообщением "Разрешить приложению использовать данные геолокации?"
10. Нажатие на кнопку "Разрешить при использовании"	11. Подключение геолокации пользователя
2.1 Исключение. Запретить	12. Ответ картографическому сервису. Передача данных положения
13. Получение данных картографическим сервисом	14. Закрытие информационного фрейма
15. Отображение нынешнего положения пользователя на карте	
2.1 Исключение. Запретить	
10. Нажатие на кнопку "Запретить"	11. Закрытие информационного фрейма 12. Отображение карты пользователя без привязки к его положению

Рисунок А.16 - Исключение ВИ «Работа с картами/маршрутами»

Главный раздел сценария выполнения варианта использования "Добавление комментария/заметка"	
Вариант использования	Добавление комментария/заметка
Актеры	Пользователь(авторизованный)
Краткое описание	Комментирование пройденной тренировки внутри дневника тренировок
Цель	Оставление комментария о состоянии после тренировки
Тип	Расширяющий вариант использования
Ссылки на другие варианты использования	Расширяется: Составлением личной тренировки

Рисунок А.17 - Главный раздел ВИ «Добавление комментария/заметка»

Раздел Типичный ход событий сценария выполнения варианта использования "Добавление комментария/заметка"	
Действия актёров	Отклик системы
1. Нажатие на +	2. Отображение фрейма для оставления заметки
3. Оставить комментарий	4. Отображение заносимого сообщения
1. Исключение. Комментарий превышает допустимый объём текста	
5. Нажать "Ok"	2. Исключение. Отсутствует текст в поле 6. Сохранить введенной сообщение в БД 7. Прикрепить к записи о тренировке

Рисунок А.18 - Типичный ход ВИ «Добавление комментария/заметка»

ИСКЛЮЧЕНИЯ	
Раздел Исключения сценария выполнения варианта использования "Добавление комментария/заметка"	
Действия актёров	Отклик системы
1. Исключение. Комментарий превышает допустимый объём текста	
	1. Блокирует дальнейшее введение сообщения, обрывая заметку
2. Исключение. Отсутствует текст в поле	
	2. Выводит инфострочку о превышении объёма символов в заметке.
	1. Отобразить пустой блок

Рисунок А.19 - Исключение ВИ «Добавление комментария/заметка»

Главный раздел сценария выполнения варианта использования "Удаление комментария/заметка"	
Вариант использования	Удаление комментария/заметка
Актеры	Пользователь(авторизованный)
Краткое описание	Удаление уже имеющегося комментария по тренировке, очищение поля ввода комментария
Цель	Удаление введённой заметки ранее
Тип	Расширяющий вариант использования
Ссылки на другие варианты использования	Расширяет: Добавление комментариев/заметка

Рисунок А.20 - Главный раздел ВИ «Удаление комментария/заметка»

Раздел Типичный ход событий сценария выполнения варианта использования "Удаление комментария/заметка"	
Действия актёров	Отклик системы
1. Нажать на иконку удаления	2. Выводиться окно с подтверждением
	<i>Исключение: Пользователь нажимает "Отмена"</i>
3. Нажимает "Подтвердить"	4. Закрытие информационного фрейма
	5. Очищает поле
	6. Данные из БД удаляются
	7. Фрейм с заметкой сворачивается, на его месте появляется иконка +

Рисунок А.21 - Типичный ход ВИ «Удаление комментария/заметка»

ИСКЛЮЧЕНИЯ	
Раздел Исключения сценария выполнения варианта использования "Удаление комментария/заметка"	
Действия актёров	Отклик системы
	<i>Исключение: Пользователь нажимает "Отмена"</i>
1. Нажатие на кнопку "Отмена"	2. Закрытие информационного фрейма

Рисунок А.22 - Исключение ВИ «Удаление комментария/заметка»

Главный раздел сценария выполнения варианта использования "Установить оповещение"	
10	Вариант использования
	Установить оповещение
	Актеры
	Пользователь(авторизованный)
	Краткое описание
	Включить функцию оповещения заблаговременно о намеченной тренировке
	Цель
	Получение оповещения от приложения о запланированном забеге
	Тип
	Расширяющий вариант использования
	Ссылки на другие варианты использования
	Расширяет: Запланировать тренировку

Рисунок А.23 - Главный раздел ВИ «Установить оповещение»

Раздел Типичный ход событий сценария выполнения варианта использования "Установить оповещение"	
Действия актёров	Отклик системы
1. Кликнуть на иконку колокольчика	2. Сменит цвет иконки со статичного на подсвеченный
	3. Активирует функцию оповещения о запланированной тренировке

Рисунок А.24 - Типичный ход ВИ «Установить оповещение»

Главный раздел сценария выполнения варианта использования "Сохранение пройденных маршрутов"	
11	Вариант использования
	Сохранение пройденных маршрутов
	Актеры
	Пользователь(авторизованный)
	Краткое описание
	По завершении тренировки возможность сохранить снимок карты с пройденным маршрутом
	Цель
	Сохранение снимка карты в разделе "Дневник тренировок", выполнить позже повторно этот же маршрут
	Тип
	Расширяющий вариант использования
	Ссылки на другие варианты использования
	Расширяет: Работа с картами/маршрутами

Рисунок А.25 - Главный раздел ВИ «Сохранение пройденных маршрутов»

Раздел Типичный ход событий сценария выполнения варианта использования "Сохранение пройденных маршрутов"	
Действия актёров	Отклик системы
1. Нажать на кнопку "Сохранить тренировку"	2. Отправление данных в БД
	3. Сохранение данных в БД
	4. Сообщение об успешном сохранении тренировки

Рисунок А.26 - Типичный ход ВИ «Сохранение пройденных маршрутов»

Главный раздел сценария выполнения варианта использования "Отменить тренировку"	
12	Вариант использования Отменить тренировку
	Актеры Пользователь(авторизованный)
	Краткое описание Удалить тренировку из списка запланированных или поставить на стоп
	Цель Убрать тренировку из списка задач и параллельная отмена уведомлений по ней
	Тип Расширяющий вариант использования
	Ссылки на другие варианты использования Расширяет: Запланировать тренировку

Рисунок А.27 - Главный раздел ВИ «Отменить тренировку»

Раздел Типичный ход событий сценария выполнения варианта использования "Отменить тренировку"	
Действия актёров	Отклик системы
1. Открыть раздел с календарём тренировок	2. Запуск раздела
	3. Запрос к БД для получения данных о запланированных тренировках
	4. Ответ от БД
6. Выбор дня с запланированной тренировкой, нажатие на этот день	5. Отображение страницы раздела
	7. Открытие фрейма с описанием запланированной тренировки
	8. Отобразить в интерфейсе: Изменить или удалить
7. Выбор иконки "Удалить"	Исключение. Изменить
	8. Отобразить окно с подтверждением удаления
9. Подтверждение удаления	10. Закрытие фрейма с запланированной тренировкой
	11. Удаление запланированной тренировки из БД

Рисунок А.28 - Типичный ход ВИ «Отменить тренировку»

ИСКЛЮЧЕНИЯ	
Раздел Исключения сценария выполнения варианта использования "Отменить тренировку"	
Действия актёров	Отклик системы
	Исключение. Изменить
1. Выбор новых параметров тренировки	2. Проверка даты и времени
	Исключение. Выбрано прошлое
3. Нажать на сохранить изменения	4. Закрытие фрейма с изменениями
	5. Обновление данных в БД
	6. Отображение метки в календаре о запланированной тренировке
	Исключение. Отменить
1. Нажать отмена	2. Закрытие информационного фрейма
4. Выход из данного раздела	3. Отображение описания тренировки

Рисунок А.29 - Исключение ВИ «Отменить тренировку»

Главный раздел сценария выполнения варианта использования "Составление личной тренировки"	
13	Вариант использования Составление личной тренировки
	Актеры Пользователь(авторизованный)
	Краткое описание Выбрать вариант составления личной тренировки, с указанием времени, дистанции, построение маршрута на карте
	Цель Выполнение тренировки по личным показателям
	Тип Базовый вариант использования
	Ссылки на другие варианты использования Расширяющий: Выбор готового плана тренировок. Расширяется: Работа с картами/маршрутами

Рисунок А.30 - Главный раздел ВИ «Составление личной тренировки»

Раздел Типичный ход событий сценария выполнения варианта использования "Составление личной тренировки"	
Действия актёров	Отклик системы
1. Открытие раздела с планами тренировок	2. Отображение страницы с готовыми планами
2. Выбор категории "Личная тренировка"	4. Загрузка страницы для заполнения форм составления тренировки
5. Выбор типа тренировок в раскрывающемся меню	
6. Установка длительности из раскрывающегося меню	
7. Установка интенсивности тренировки	
8. Установка разминки и её длительности	
9. Нажатие кнопки "Далее"	10. Проверка введённых параметров
12. Нажать на "Выполнить тренировку"	11. Отображает составленную тренировку, её описание
14. Нажать "Нет"	13. Предложит пользователю составить маршрут к тренировке
	15. Обработает получившуюся программу
	16. Запуск тренировки

Рисунок А.31 – Типичный ход ВИ «Составление личной тренировки»

Главный раздел сценария выполнения варианта использования "Предоставление доступа к картам"	
<u>Вариант использования</u>	Предоставление доступа к картам
<u>Актеры</u>	Картографический сервис
<u>Краткое описание</u>	Выгрузка карт и отображение внутри приложения
14	Доступ на работу с картами(изменение масштаба, синхронизация с геолокацией пользователя, указание маршрута на карте)
<u>Цель</u>	Базовый вариант использования
<u>Тип</u>	
<u>Ссылки на другие варианты использования</u>	Включаемый: Работа с картами/маршрутами

Рисунок А.32 – Главный раздел ВИ «Предоставление доступа к картам»

ПРИЛОЖЕНИЕ Б
ДИАГРАММА ПОСЛЕДОВАТЕЛЬНОСТИ

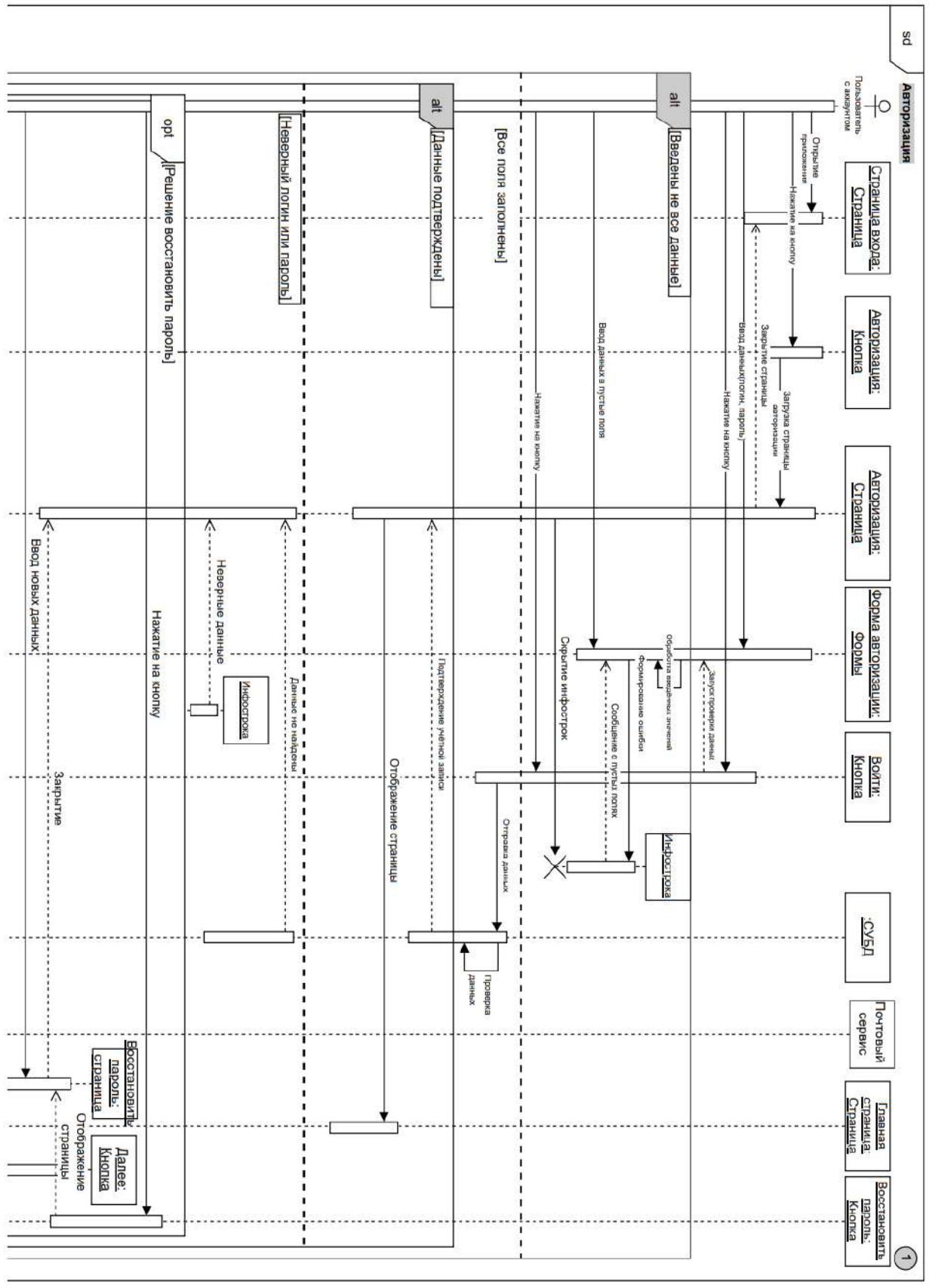


Рисунок Б.1 - Диаграмма последовательностей для ВИ «Авторизация»

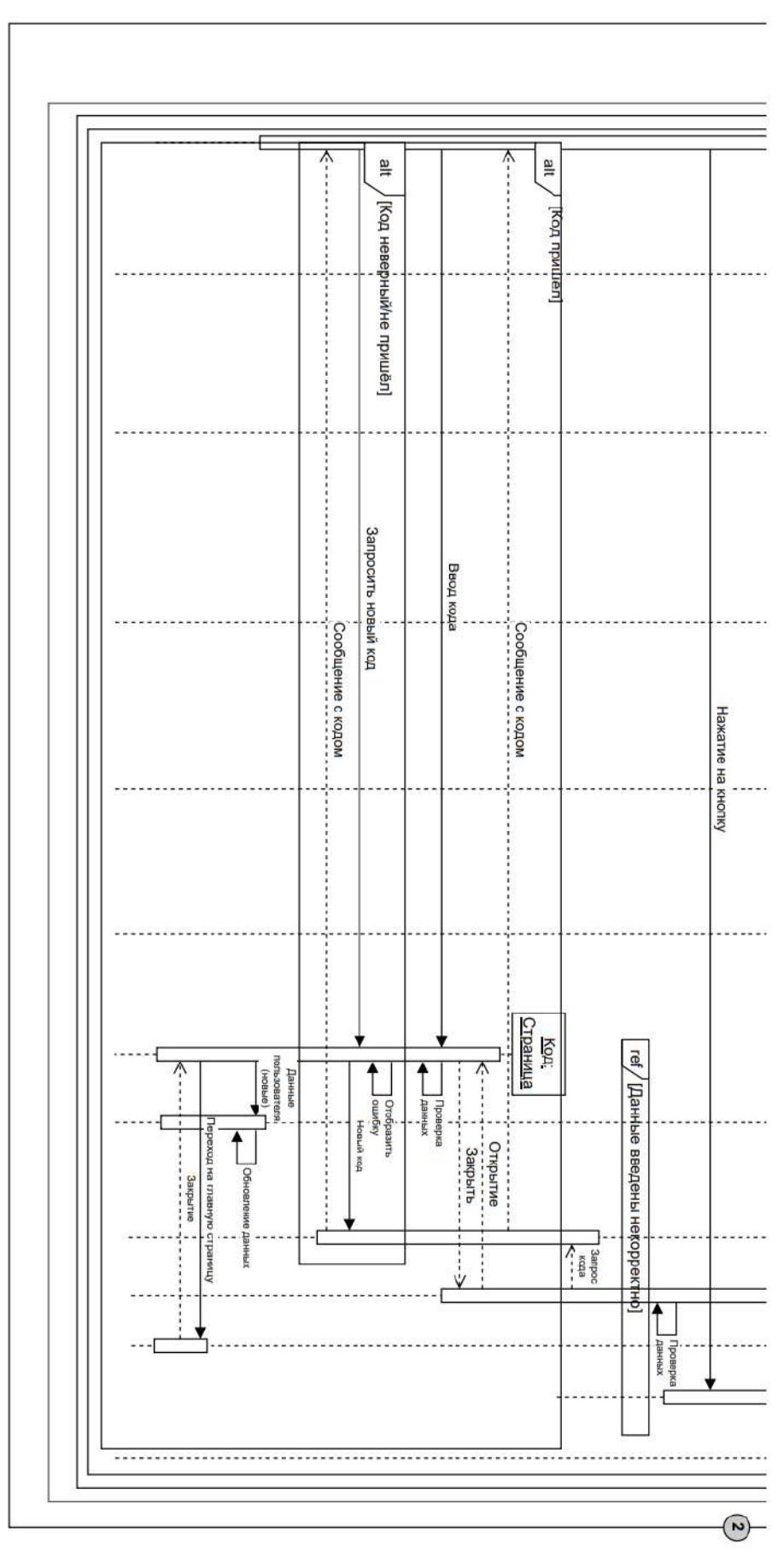


Рисунок Б.2 - Диаграмма последовательностей для ВИ «Авторизация»

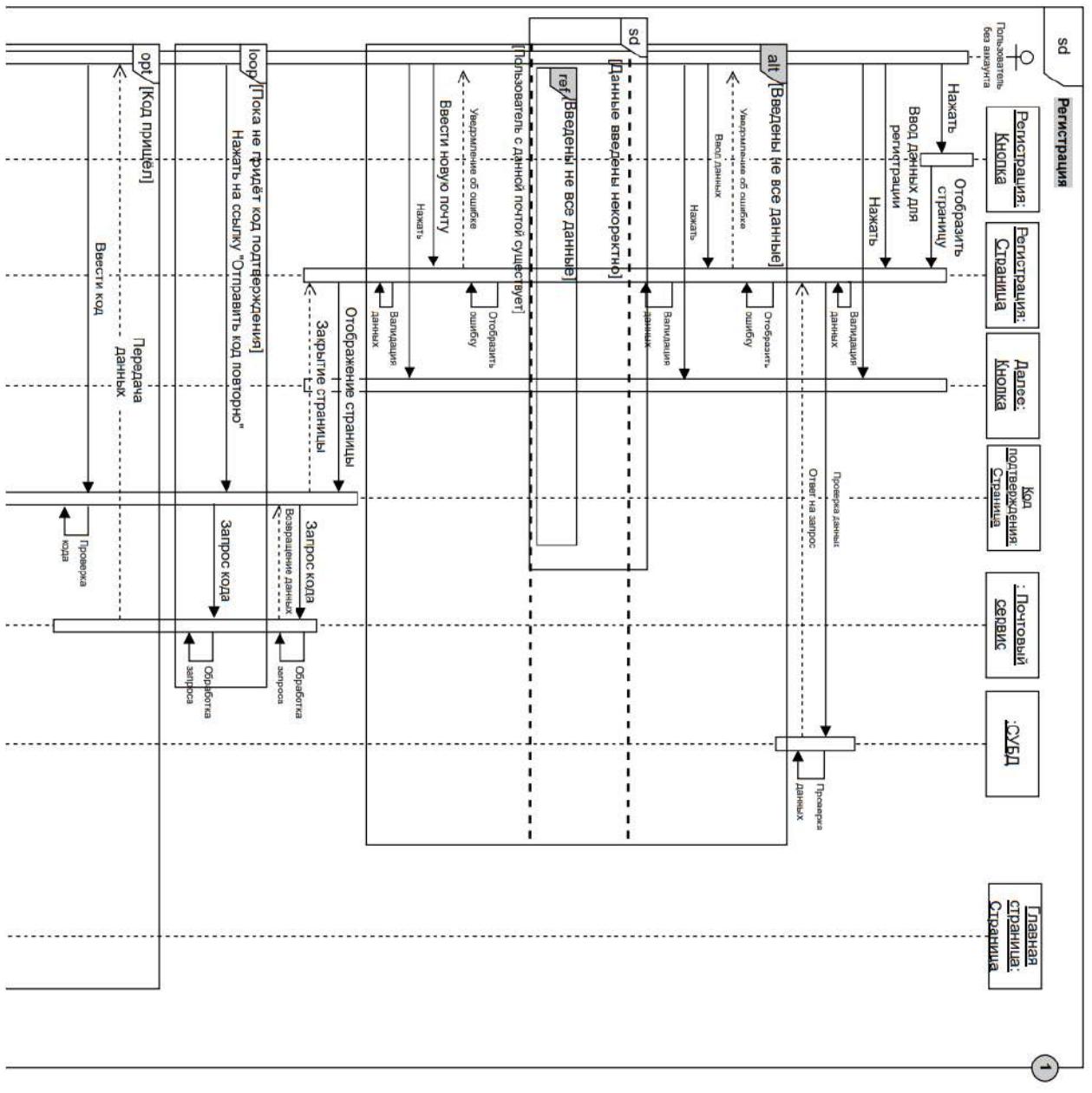


Рисунок Б.3 - Диаграмма последовательностей для ВИ «Регистрация»

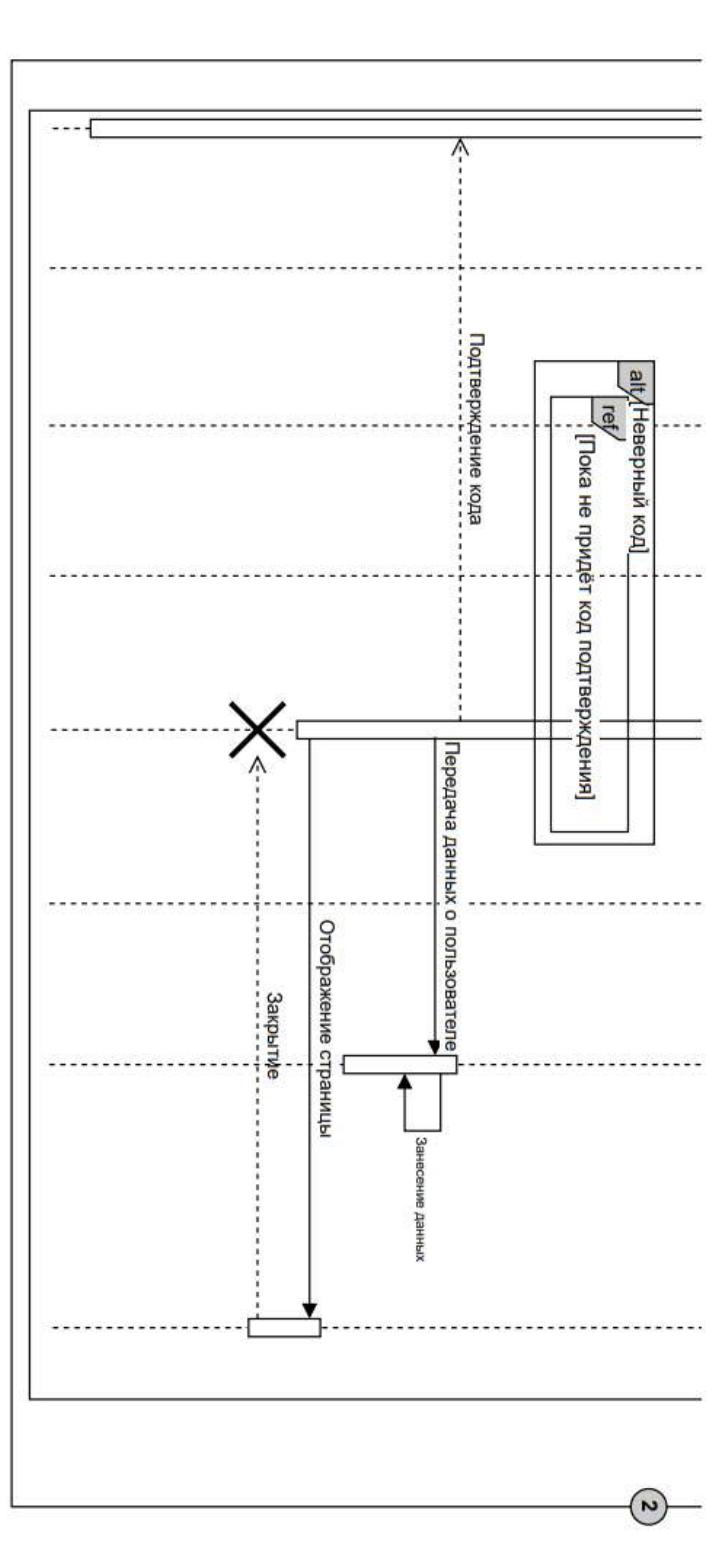


Рисунок Б.4 - Диаграмма последовательностей для ВИ «Регистрация»

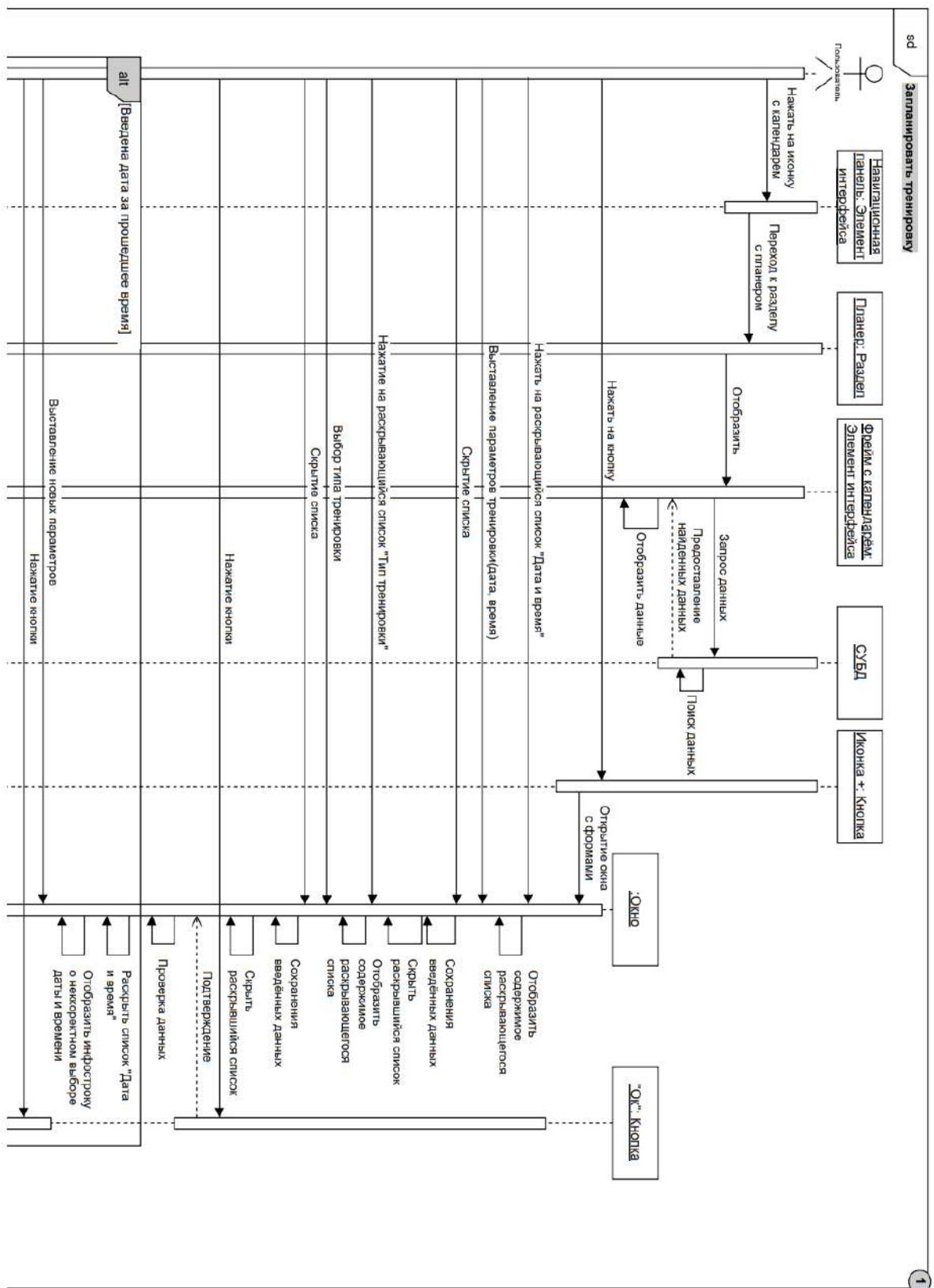


Рисунок Б.5 - Диаграмма последовательностей для ВИ «Запланировать тренировку»

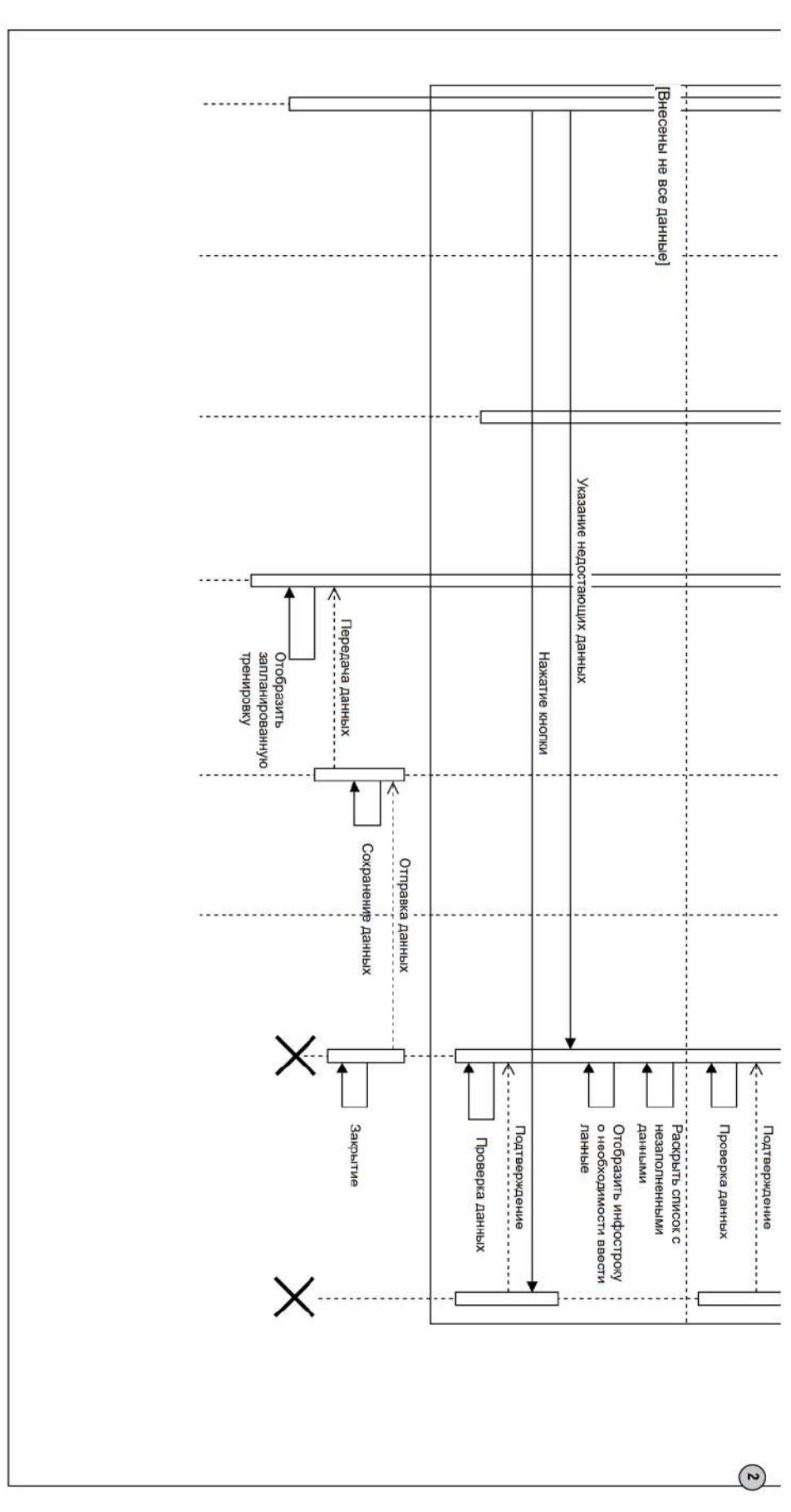


Рисунок Б.6 - Диаграмма последовательностей для ВИ «Запланировать тренировку»

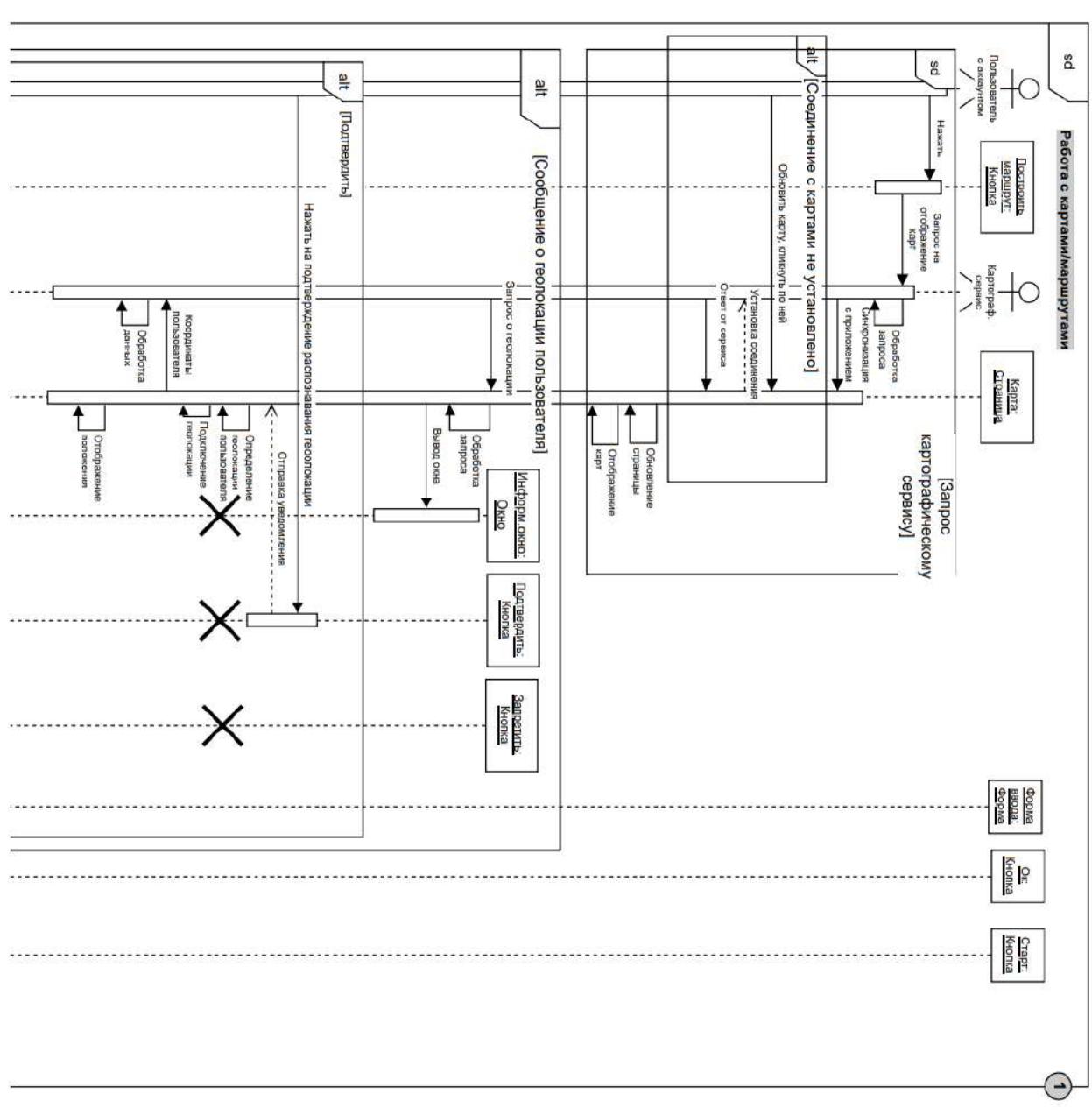


Рисунок Б.7 - Диаграмма последовательностей для ВИ «Работа с картами/маршрутами»

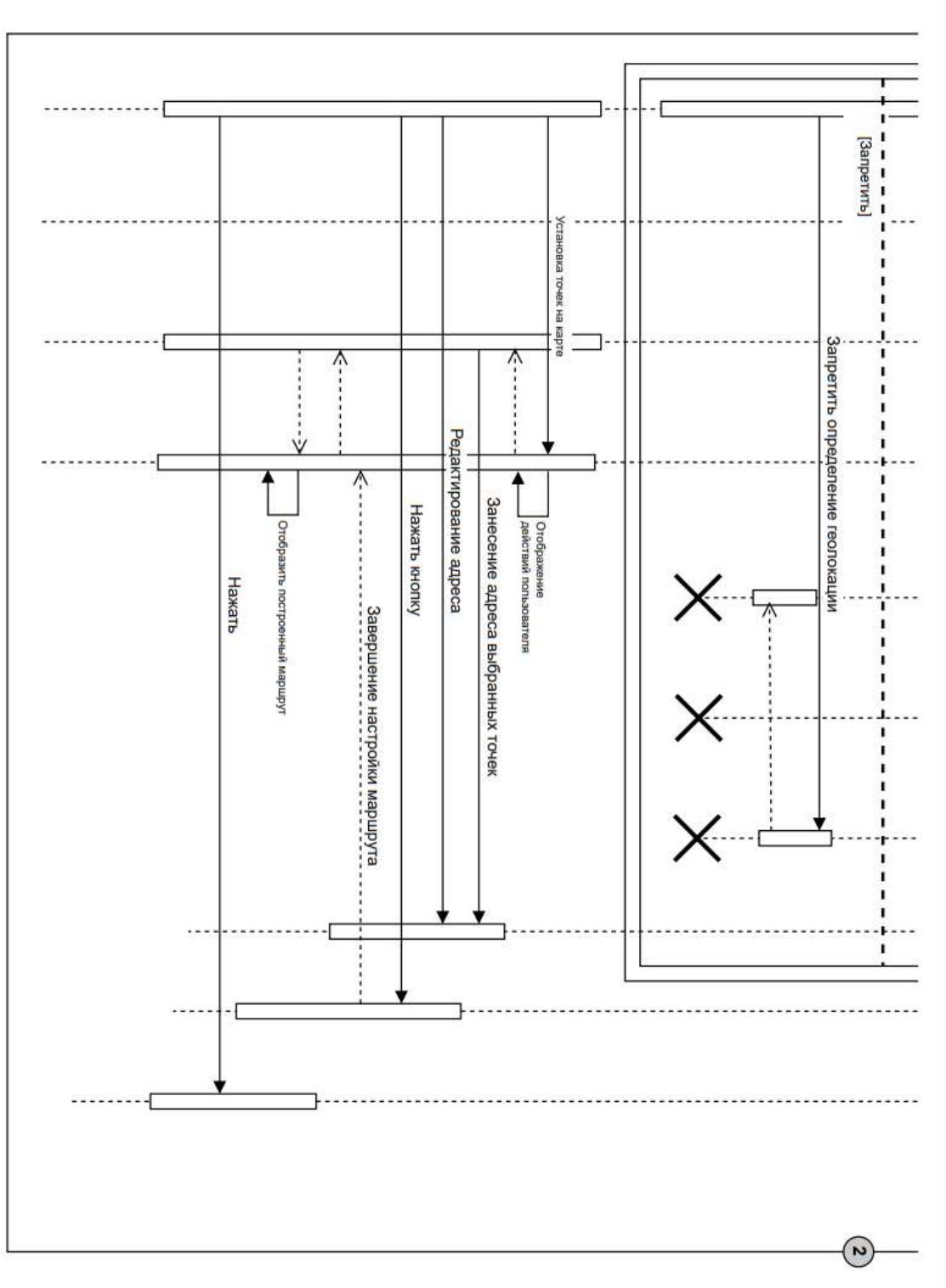


Рисунок Б.8 - Диаграмма последовательностей для ВИ «Работа с картами/маршрутами»

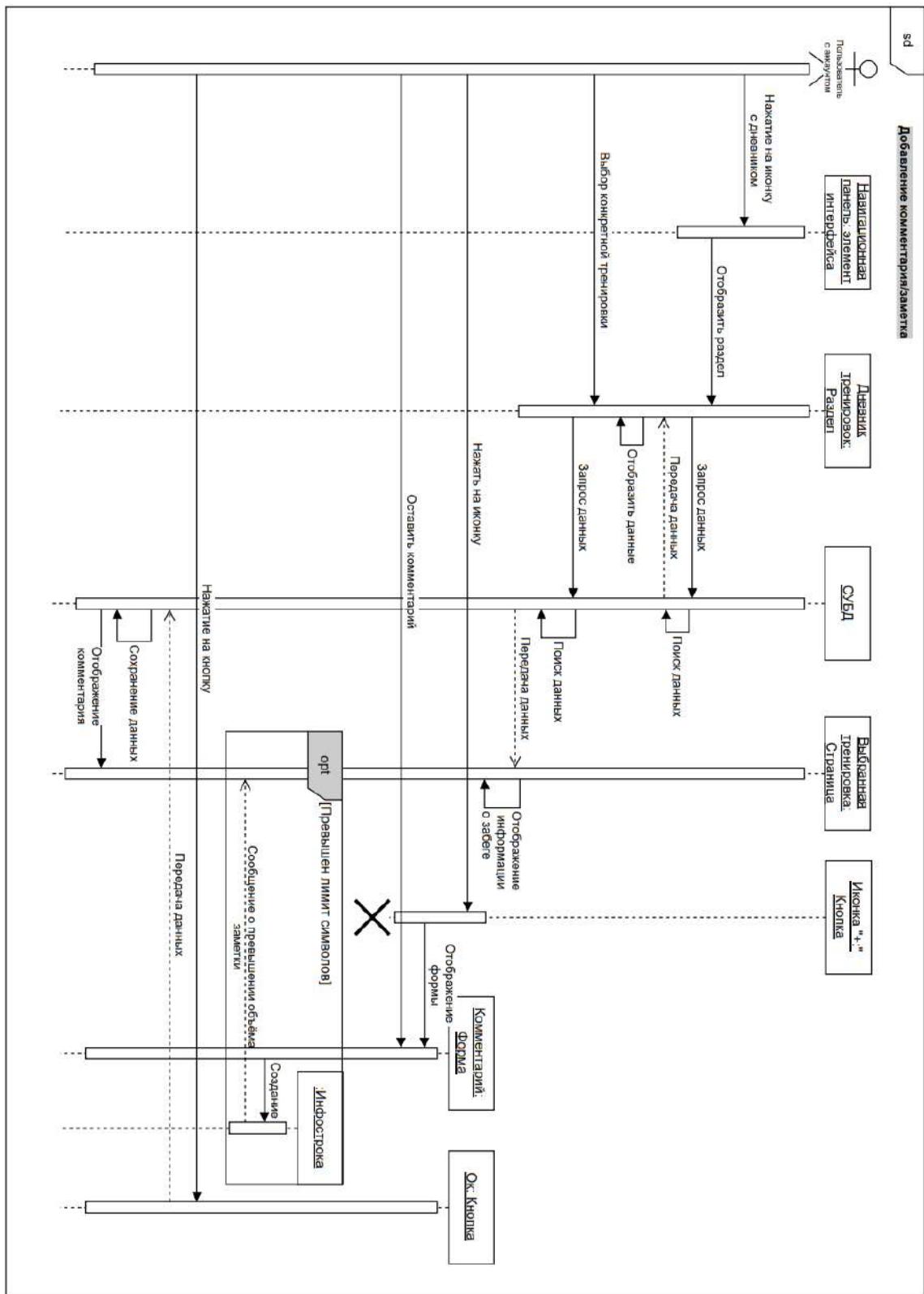


Рисунок Б.9 - Диаграмма последовательностей для ВИ «Добавление комментариев/заметка»

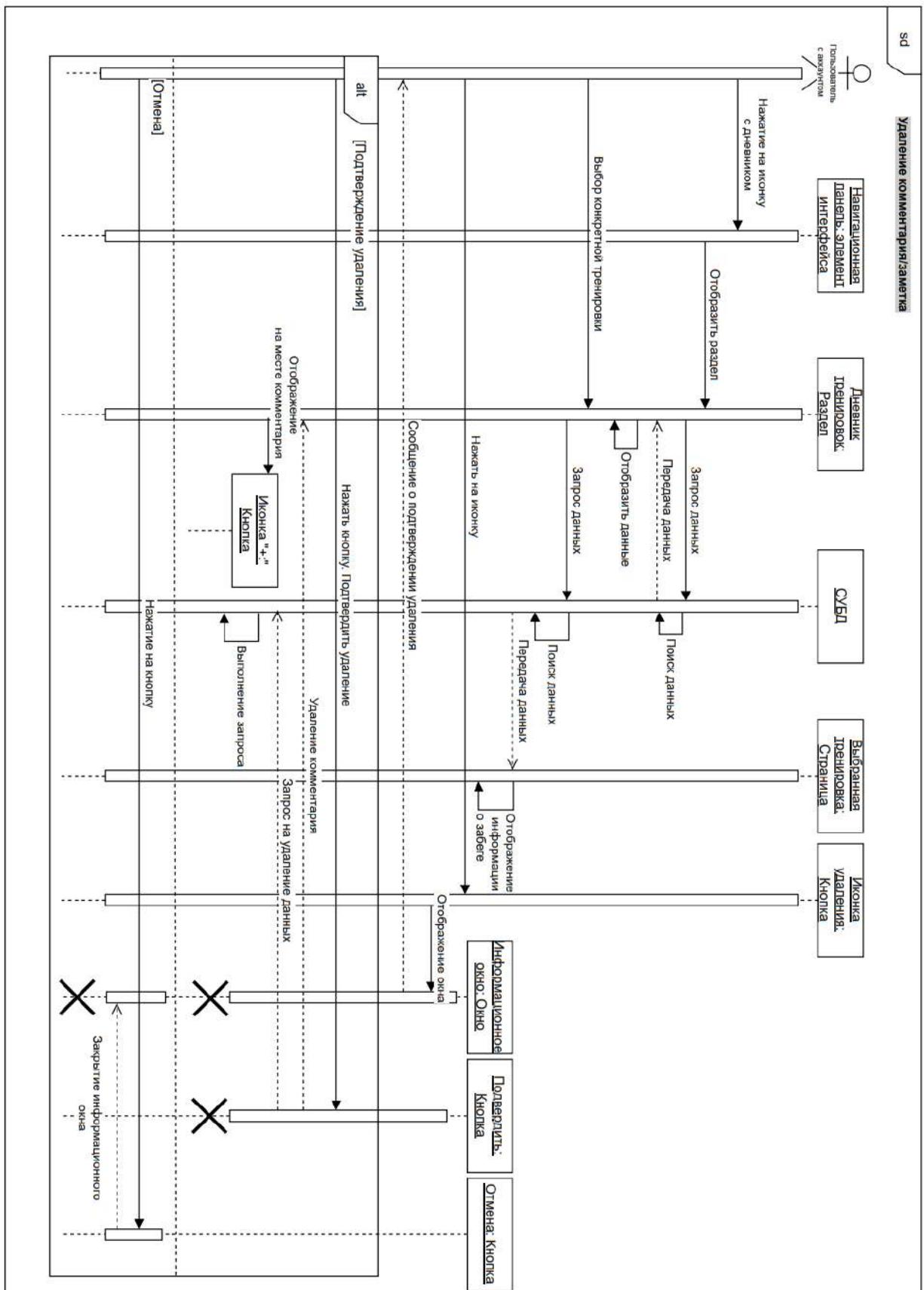


Рисунок Б.10 - Диаграмма последовательностей для ВИ «Удаление комментариев/заметки»

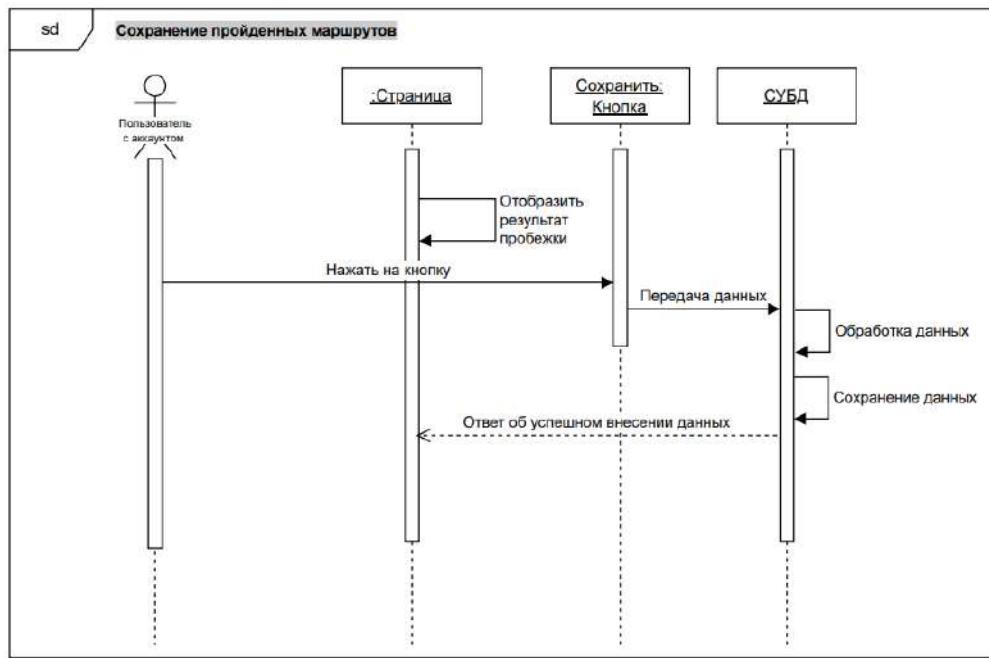


Рисунок Б.11 - Диаграмма последовательностей для ВИ «Сохранение пройденных маршрутов»

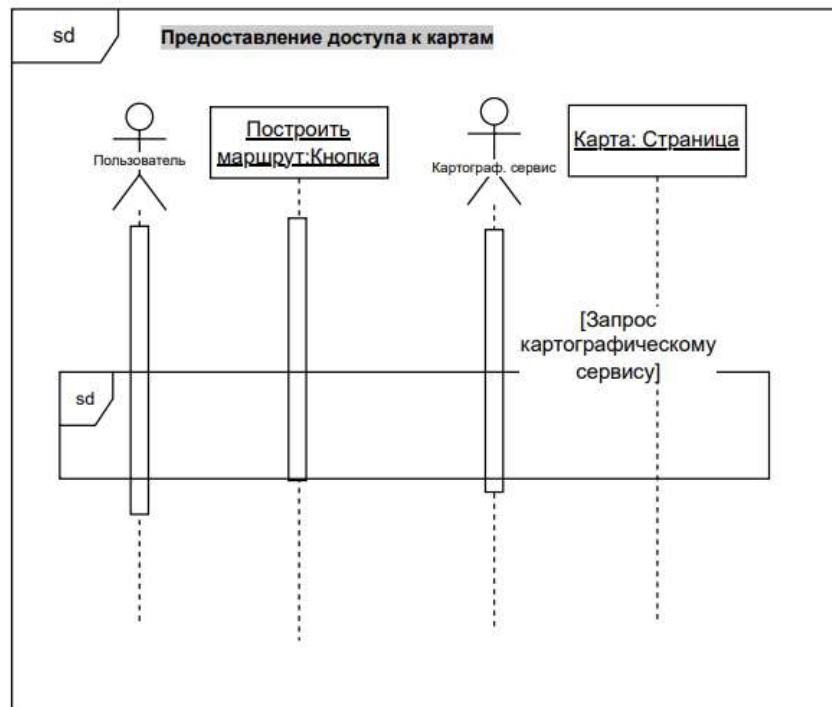


Рисунок Б.12 - Диаграмма последовательностей для ВИ «Предоставление доступа к картам»

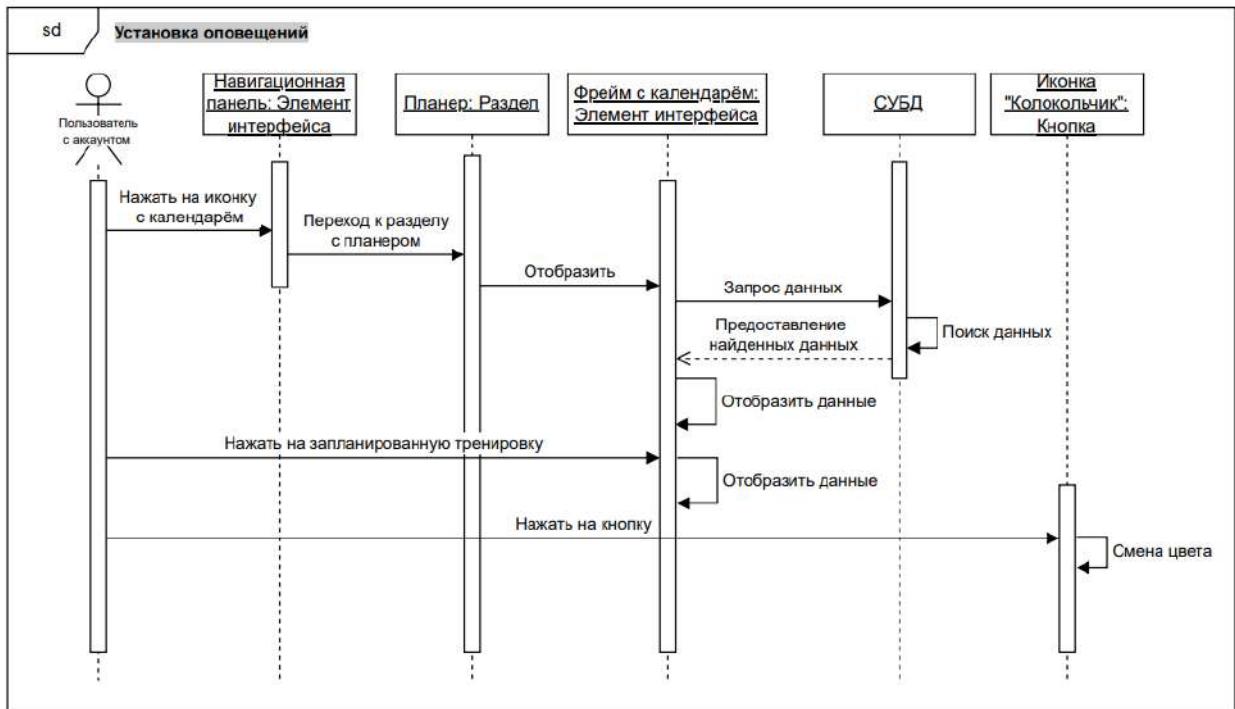


Рисунок Б.13 - Диаграмма последовательностей для ВИ «Установка оповещений»

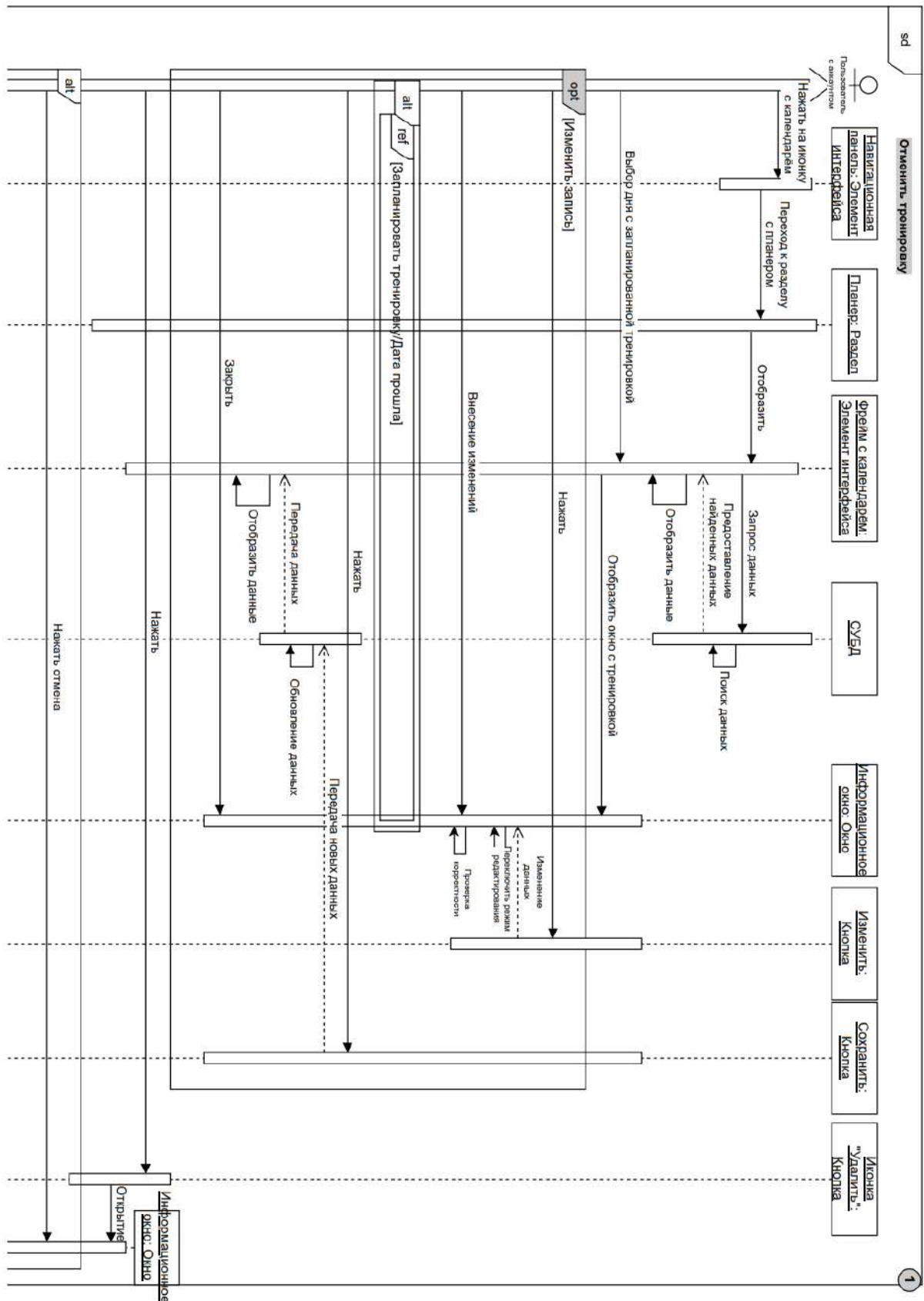


Рисунок Б.14 - Диаграмма последовательностей для ВИ «Отменить тренировку»

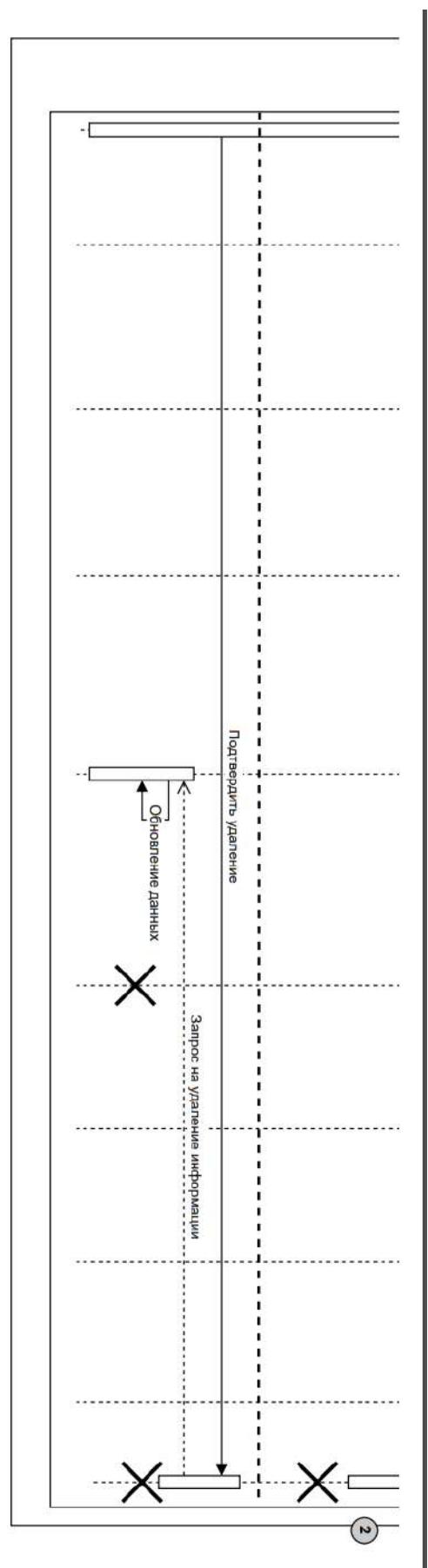


Рисунок Б.15 - Диаграмма последовательностей для ВИ «Отменить тренировку»

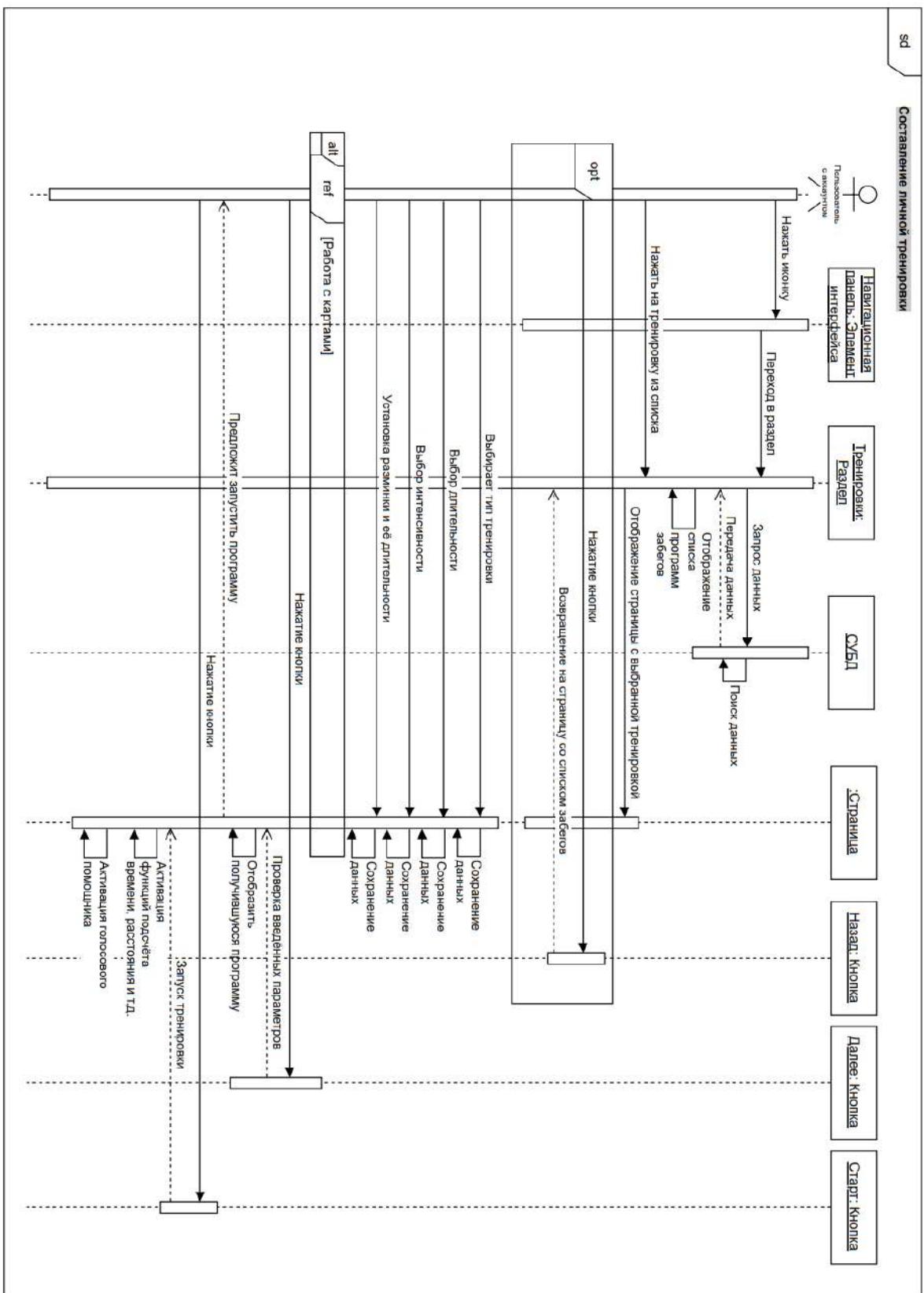


Рисунок Б.16 - Диаграмма последовательностей для ВИ «Составление личной тренировки»

ПРИЛОЖЕНИЕ В
МАКЕТ ИНФОРМАЦИОННОЙ СИСТЕМЫ

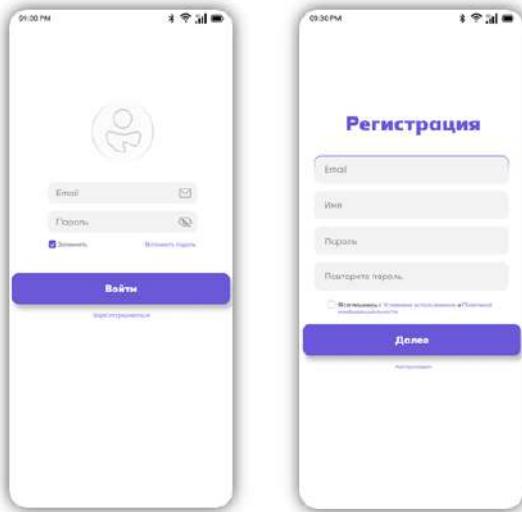


Рисунок В.1 – Экраны авторизации и регистрации



Рисунок В.2 – Экраны с разделами: Статистика, Календарь тренировок

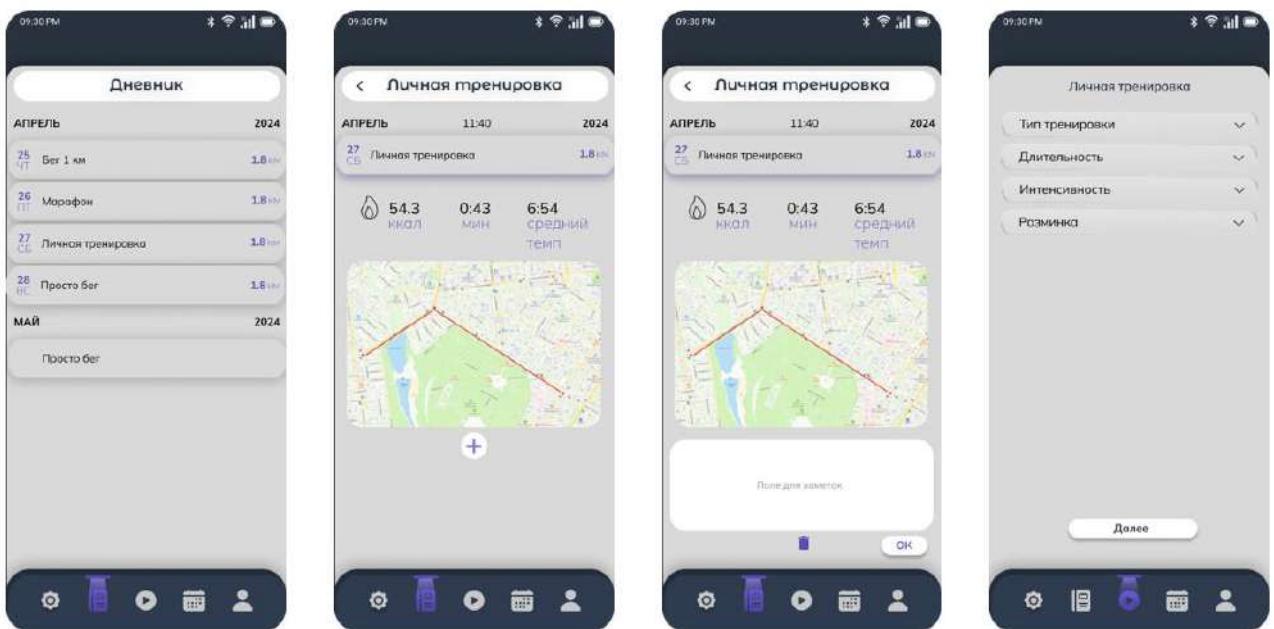


Рисунок В.3 – Экраны с разделами: Дневник тренировок, выбор составления личной тренировки

ПРИЛОЖЕНИЕ С
РЕАЛИЗОВАННЫЕ МОДУЛИ

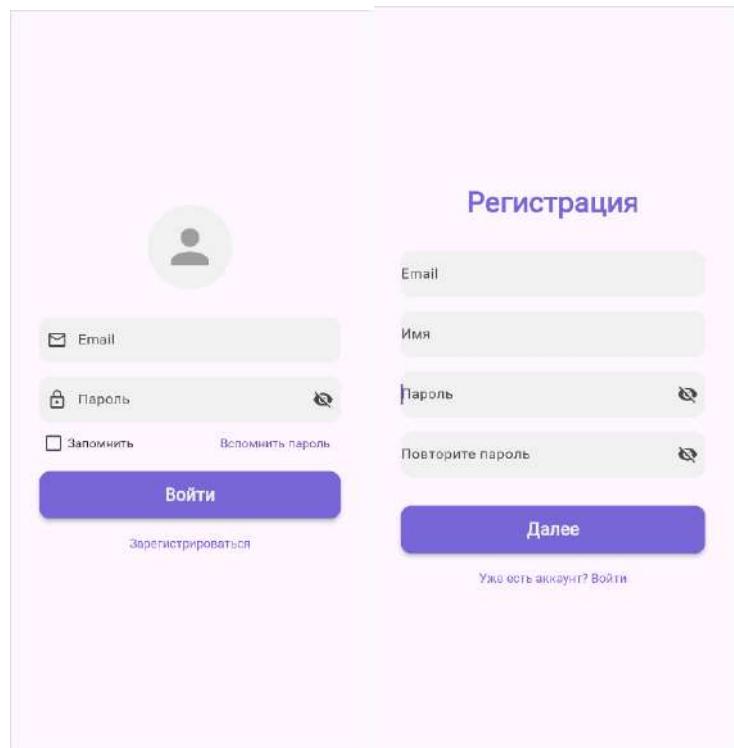


Рисунок С.1 – Авторизация и регистрация

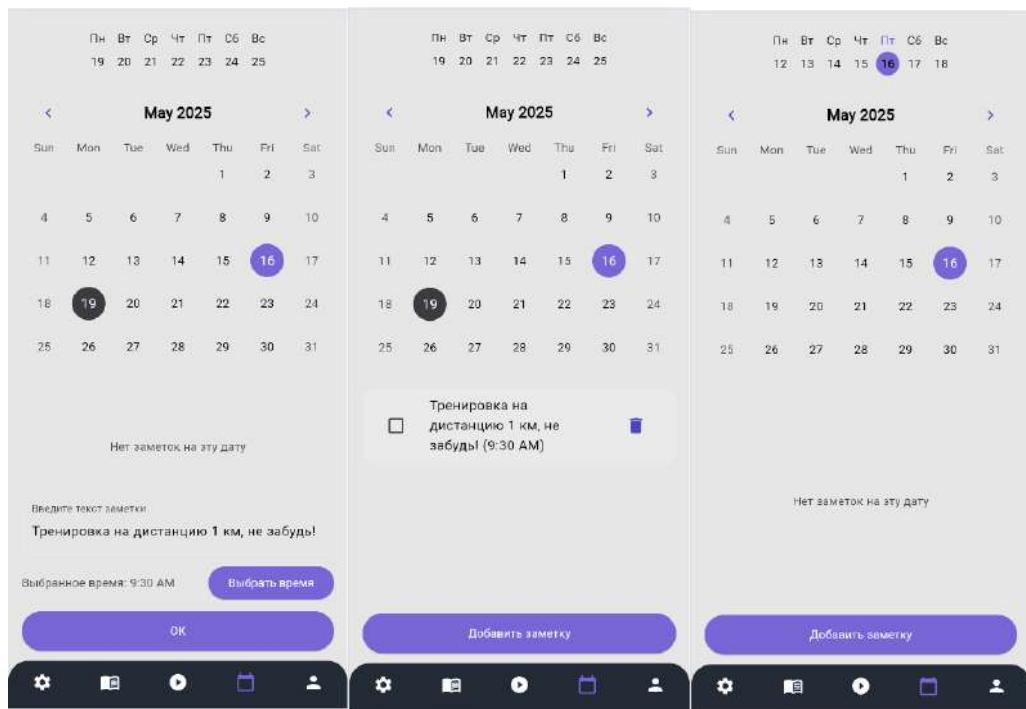


Рисунок С.2 – Модуль с календарём

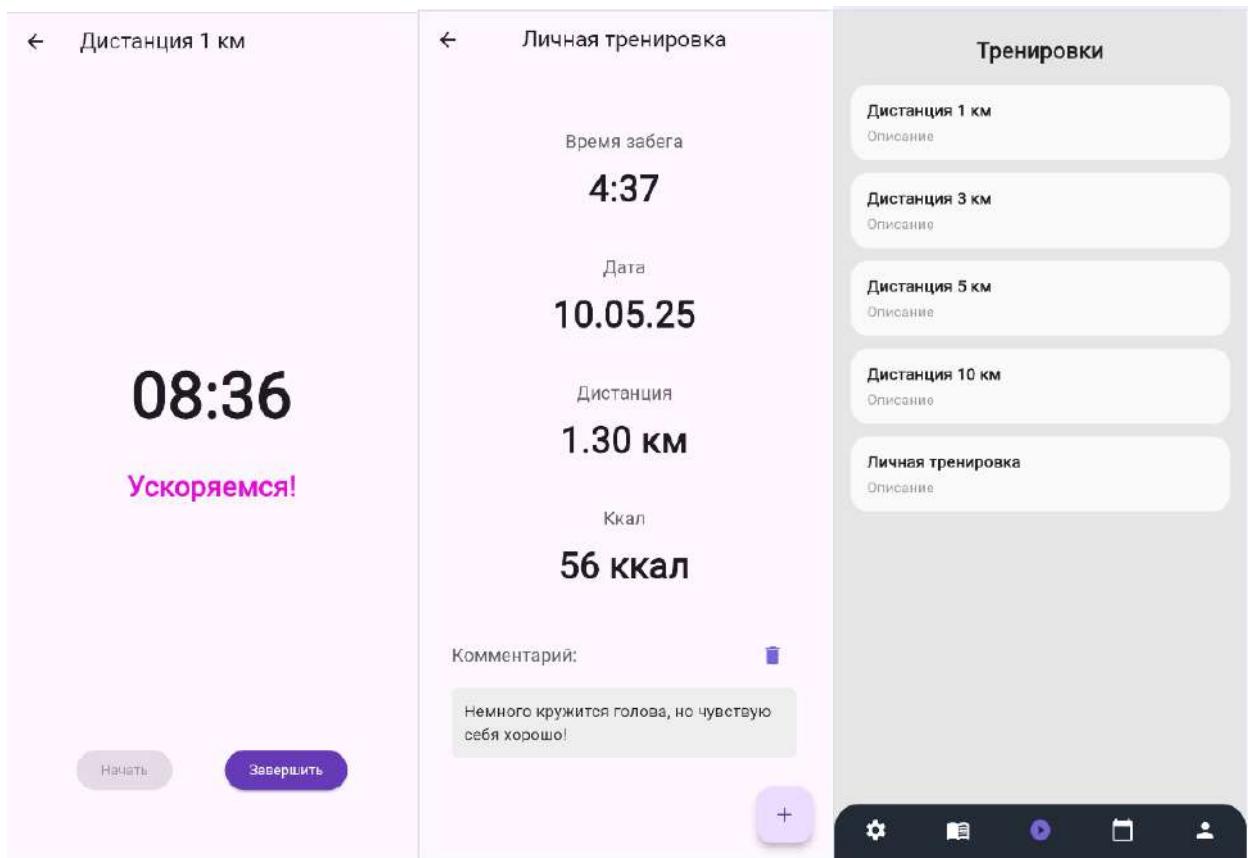


Рисунок С.3 – Выбор готовой тренировки

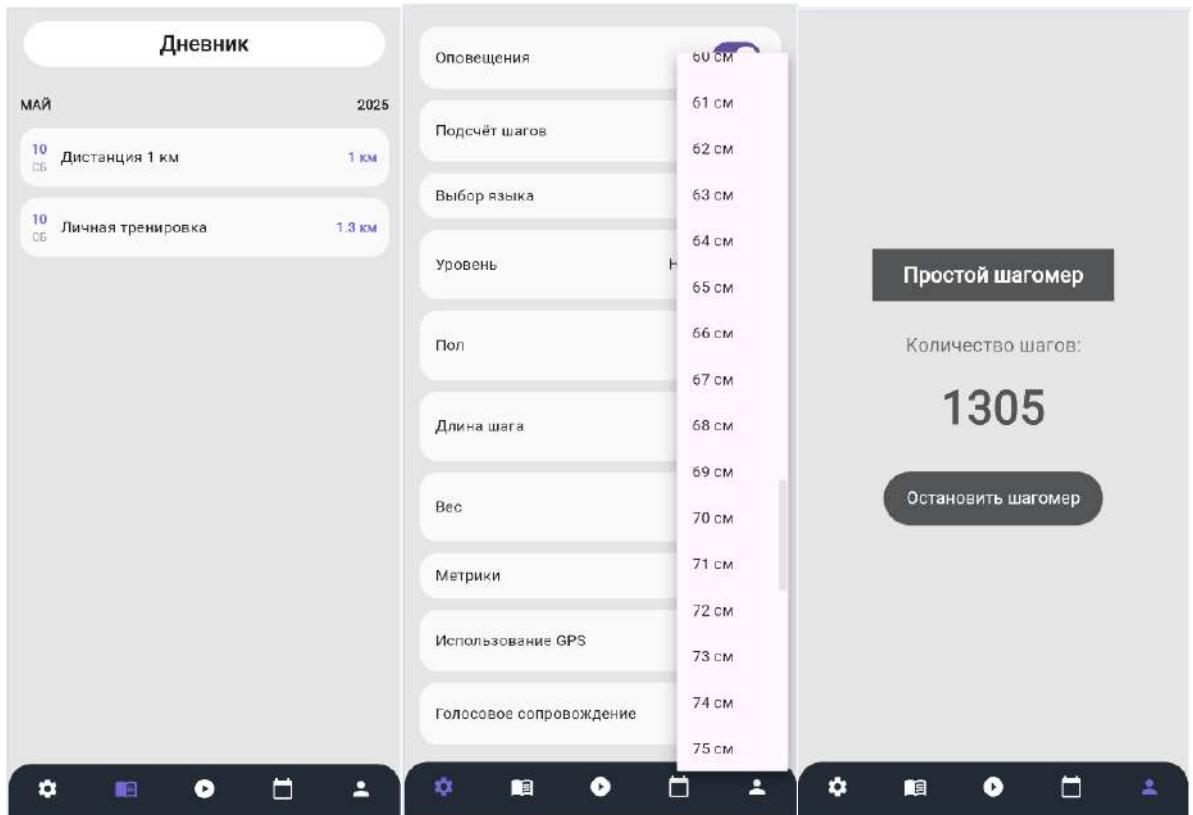


Рисунок С.4 – Дневник тренировок, настройки, шагомер