

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ВГУ»)

Факультет Компьютерных наук
Кафедра программирования и информационных технологий

Техническое задание
на разработку мобильного приложения
«Мобильный фитнес-тренер. Персонализированное приложение для
домашних тренировок»

Исполнители

_____ Е. Ю. Мануковский
_____ Д. А. Колчин
_____ А. К. Быханов
_____ А.Е. Анисимова
_____ К.В. Курманов
_____ Д.В. Косьянов

Заказчик

_____ В.С. Тарасов

Воронеж 2025

Оглавление

1 Общие сведения.....	4
1.1 Наименование приложения	4
1.2 Заказчик	4
1.3 Исполнители.....	4
1.4 Перечень документов, на основании которых создаётся система.....	4
1.5 Основания для разработки	5
1.6 Плановые сроки начала и окончания работ	5
1.7 Возможные технические риски	5
1.8 Термины и сокращения	5
2 Назначение и цели создания	7
2.1 Назначение и цели приложения	7
2.2 Цели создания приложения	7
3 Функциональные требования.....	7
3.1 Возможности неавторизованного пользователя.....	7
3.2 Возможности авторизованного пользователя.....	8
3.3 Возможности пользователя с подпиской	8
3.4 Возможности администратора	8
4 Нефункциональные требования	9
4.1 Требования к системе в целом	9
4.1.1 Требования к структуре и функционированию системы.....	9
4.1.2 Показатели назначения системы	10
4.1.3 Требования к защите информации от несанкционированного доступа.....	11
4.1.4 Требования к аутентификации.....	12
4.1.5 Требования к численности и квалификации персонала.....	12
4.1.6 Требования к патентной чистоте	13
4.1.7 Требования к масштабируемости и открытости	13

4.1.8 Обработка ошибок.....	13
4.2 Требования к видам обеспечения	13
4.2.1 Требования к информационному обеспечению	13
4.2.2 Требования к программному обеспечению приложения	14
4.2.3 Требования к лингвистическому обеспечению	15
4.2.4 Требования к LLM компоненту	15
5 Структура приложения	15
6 Дизайн приложения	16
7 Навигация по приложению	17
8 Описание страниц приложения	17
8.1 Описание страниц.....	17
8.1.1 Окно входа	17
8.1.2 Окно регистрации.....	18
8.1.3 Восстановление пароля.....	18
8.1.4 Главный экран	18
8.1.5 Блок А.....	18
8.1.6 Блок Б.....	19
8.1.7 Личный кабинет.....	21
Приложение А	23
Приложение Б.....	24
Приложение В.....	25
Приложение Г	26
Приложение Д.....	27

1 Общие сведения

1.1 Наименование приложения

Полное наименование: «Trainova» – мобильное приложение для занятий спортом дома. Оно подойдёт и для тех, кто только начинает заниматься спортом, так и для тех, у кого уже есть опыт.

1.2 Заказчик

Старший преподаватель Тарасов Вячеслав Сергеевич, кафедра программирования и информационных технологий

1.3 Исполнители

Студент Мануковский Егор Юрьевич, кафедра информационных систем.

Студентка Анисимова Анастасия Евгеньевна, кафедра информационных систем.

Студент Быханов Артём Константинович, кафедра информационных систем.

Студент Колчин Даниил Алексеевич, кафедра информационных систем.

Студент Курманов Кирилл Владиславович, кафедра информационных систем.

Студент Косьянов Даниил Витальевич, кафедра информационных систем.

1.4 Перечень документов, на основании которых создаётся система

- Федеральный закон "Об информации, информационных технологиях и о защите информации" от 27.07.2006 N 149–ФЗ;
- Федеральный закон "О персональных данных" от 27.07.2006 N 152–ФЗ;
- Федеральный закон "О коммерческой тайне" от 29.07.2004 N 98–ФЗ;

— Закон РФ от 07.02.1992 N 2300 1 (ред. от 11.06.2021) "О защите прав потребителей";

1.5 Основания для разработки

Клиенту требуется мобильное приложение, которое будет предоставлять тренировки для занятий спортом дома. Пользователь сможет выбирать: заниматься бесплатно по тренировкам, предоставленным приложением, или же оплачивать подписку и заниматься по курсу тренировок понравившегося ему тренера. Помимо этого, Пользователь сможет фильтровать тренировки по включённым в тренировку группам мышц. Курсы тренировок можно будет оценивать и комментировать каждому пользователю

1.6 Плановые сроки начала и окончания работ

Срок начала работ – 13 февраля 2025.

Плановый срок окончания работ – 28 июня 2025.

1.7 Возможные технические риски

В ходе разработки мы можем столкнуться со следующими техническими рисками:

- Необходимость расширения базы данных;
- Возможное изменение структуры API в ходе разработки;
- Необходимость изменения дизайна окон, выявленные уже в процессе разработки.

1.8 Термины и сокращения

Блок А – обозначение, созданное командой разработки для удобства коммуникации. Обозначает часть приложения, в котором пользователи смогут просматривать и заниматься по тренировкам, выложенным администратором.

Блок Б – часть приложения, в котором пользователи смогут выкладывать свои курсы тренировок, а также смотреть курсы, выложенные другими людьми.

Администратор – авторизованный пользователь с особыми привилегиями, на эту роль назначается заказчиком.

PWA (Progressive Web Apps) – тип веб-приложения, который сочетает в себе преимущества веб-сайтов и нативных мобильных приложений.

Bottom-меню – элемент пользовательского интерфейса в мобильных приложениях, который обычно расположен в нижней части экрана. Данное меню включает в себя набор основных функций и разделов приложения, обеспечивая быстрый доступ к наиболее часто используемым действиям и страницам.

Backend – Серверная часть продукта, которая отвечает за внутреннюю логику и работу мобильного приложения.

Frontend – Пользовательский интерфейс, к нему относится всё, что пользователи видят в мобильном приложении, и с чем можно взаимодействовать.

PostgreSQL – Свободная объектно-реляционная СУБД.

СУБД – Система управления базами данных.

API – Программный интерфейс приложения. Описание способов, которыми одна компьютерная программа может взаимодействовать с другой программой.

Docker – платформа с открытым исходным кодом для автоматизации разработки, доставки и развёртывания приложений.

React – JavaScript-библиотека с открытым исходным кодом для разработки пользовательских интерфейсов.

Next.js – фреймворк на базе React для разработки клиент-серверных (fullstack) приложений как для мобильных платформ, так и для веба.

Nginx – программное обеспечение с открытым исходным кодом для создания веб-серверов.

GitHub – веб-сервис для хостинга IT-проектов и их совместной разработки.

FastAPI – современный и высокопроизводительный веб-фреймворк для создания API на Python.

2 Назначение и цели создания

2.1 Назначение и цели приложения

Целью приложения является создания платформы, на которой обычные пользователи смогут подобрать для себя тренировки согласно их целям и уровню подготовки, а также выкладывать свои курсы тренировок.

2.2 Цели создания приложения

Цели создания:

- Получение 10% от ежемесячных подписок пользователей на платные курсы тренировок.

3 Функциональные требования

Для создания тренировок, которые приложение будет предоставлять пользователю бесплатно, необходим как минимум 1 пользователь от заказчика, который будет выполнять роль администратора.

Для взаимодействия с данной системой выделяют следующие виды пользователей:

- Незарегистрированный пользователь;
- Зарегистрированный пользователь;
- Пользователь с подпиской;
- Администратор.

3.1 Возможности неавторизованного пользователя

Неавторизованный пользователь имеет возможность в процессе взаимодействия с данной системой выполнять следующие действия:

- Просматривать тренировки, но без возможности начать заниматься;
- Зарегистрироваться.

3.2 Возможности авторизованного пользователя

Зарегистрированный пользователь имеет возможность в процессе взаимодействия с данной системой, выполнять следующие действия:

- Возможности неавторизованного пользователя (кроме регистрации);
- Заниматься по бесплатным тренировкам, предоставленным платформой;
- Фильтровать тренировки согласно своим потребностям;
- Продолжить или начать заново тренировку, которую он не закончил;
- Смотреть бесплатные тренировки, предоставленные тренерами;
- Отслеживать частоту тренировок и свой вес на графике;
- Купить подписку на курс тренировок понравившегося тренера;
- Комментировать и ставить оценку тренировкам от тренеров;
- Возможность создавать через конструктор свои курсы тренировок.

3.3 Возможности пользователя с подпиской

Пользователь с подпиской имеет возможность в процессе взаимодействия с данной системой, выполнять следующие действия:

- Все возможности обычного зарегистрированного пользователя.
- Полный доступ к курсу тренировок, на который оформлена подписка.

3.4 Возможности администратора

Администратор имеет возможность в процессе взаимодействия с данной системой выполнять следующие действия:

- Создавать тренировки в специальном конструкторе, которые после будут видны пользователям, как бесплатные тренировки от приложения.
- Весь функционал, что и у зарегистрированного пользователя.

- Доступ к просмотру всех платных курсов тренировок от тренеров.
- Возможность модерации комментариев.

4 Нефункциональные требования

4.1 Требования к системе в целом

4.1.1 Требования к структуре и функционированию системы

Пользователь взаимодействует с серверной частью приложения при помощи клиентского приложения (PWA – Progressive Web Apps). Сервер, посредством REST API, возвращает пользователю необходимые данные. Администратор взаимодействует с серверной частью приложения при помощи клиентского приложения.

Серверная часть приложения включает в себя следующие микросервисы:

- Сервис авторизации и регистрации;
- Сервис курсов от тренеров (позволяет получать, создавать, редактировать тренировки и курсы);
- Сервис тренировок от приложения (позволяет получать, создавать, редактировать тренировки);
- Сервис управления профилем (имя, фамилия, пароли, способы оплаты, восстановление паролей, подписки, аватар);
- Сервис генерации мотивационной речи на основе LLM DeepSeek R1-7B и Ollama;
- Сервис комментариев для уроков.

Backend:

- Python – язык программирования для разработки серверной части;
- FastAPI – высокопроизводительный фреймворк для создания API на Python;

- PostgreSQL – система управления базами данных;
- Docker – платформа для автоматизации развертывания и управления приложениями;
- Nginx – веб-сервер и прокси-сервер.
- Ollama – это фреймворк для локального запуска крупных языковых моделей (LLM).
- Swagger – документация API с возможностью тестирования всех эндпоинтов прямо из браузера.

Frontend:

- React – JavaScript библиотека для создания пользовательских интерфейсов;
- Next.js – фреймворк для React с функциями серверного рендеринга и генерации статических сайтов;
- TypeScript – фреймворк для серверного рендеринга и улучшенной производительности.

4.1.2 Показатели назначения системы

- Реализация домашних тренировок;
- Реализация просмотра списка всех доступных тренировок и курсов;
- Реализация добавления и удаления своих тренировок и курсов после авторизации;
- Реализация краткого описания и рейтинга тренировок и курсов;
- Реализация просмотра рейтинга тренировок и курсов;
- Реализация поиска тренировок, курсов и тренеров;
- Реализация генерации мотивационной речи для пользователей;

- Реализация генерации советов для пользователей;
- Реализация системы комментариев для уроков.

4.1.3 Требования к защите информации от несанкционированного доступа

Система должна обеспечивать безопасность данных пользователей и включать в себя следующие механизмы защиты:

- Bearer Authentication с JWT (JSON Web Tokens): Использование стандартного механизма аутентификации Bearer с JWT токенами для безопасной аутентификации и авторизации пользователей;
- Безопасное хранение и передачу учетных данных;
- Поддержку access и refresh токенов;
- Валидацию JWT с проверкой срока действия;
- Возможность отзыва токенов через blacklist;
- Версионирование токенов при смене пароля;
- Хеширование паролей: хранение паролей пользователей в базе данных в хешированном виде с использованием алгоритма bcrypt;
- HTTPS: Обязательное использование защищенного протокола HTTPS для всех взаимодействий с API;
- CORS (Cross Origin Resource Sharing): Настройка политики CORS для предотвращения несанкционированных межсайтовых запросов;
- Валидация данных: Строгая валидация всех входящих данных на стороне сервера с использованием Pydantic моделей, что является встроенной функцией FastAPI.

4.1.4 Требования к аутентификации

- В системе должна быть реализована идентификация и проверка доступа при входе в систему по логину/почте и паролю длиной не менее 8 символов.
- Система защиты должна подвергаться проверке подлинности идентификации – осуществлять аутентификацию.
- Система не должна предоставлять доступ к защищённым данным неавторизованным пользователям.
- Система защиты должна обладать способностью надёжно связывать полученную идентификацию со всеми действиями данного пользователя.

4.1.5 Требования к численности и квалификации персонала

В системе предполагается наличие следующих ролей пользователей:

- Незарегистрированный пользователь;
- Зарегистрированный пользователь;
- Пользователь с подпиской;
- Администратор.

Для создания тренировок, которые приложение будет предоставлять пользователю бесплатно, необходим как минимум 1 пользователь от заказчика, который будет выполнять роль администратора.

Пользователь с ролью администратор должен обладать знаниями и навыками необходимыми для поддержания работоспособности системы и создания тренировок через специальный конструктор.

Использование системы подразумевает, что все пользователи имеют определённый уровень знаний и умений в обращении с веб приложениями и мобильными устройствами. Это включает в себя понимание основ навигации

по интерфейсу, способности к осуществлению касаний и свайпов для выполнения команд, а также использование виртуальной клавиатуры для ввода текста.

4.1.6 Требования к патентной чистоте

Проект не должен нарушать никаких патентных прав и лицензий. Все используемые технологии и библиотеки должны иметь соответствующие лицензии, разрешающие их использование в данном проекте.

4.1.7 Требования к масштабируемости и открытости

Проект должен предоставлять возможность добавлять новую функциональность с минимальным изменением существующего кода. Микросервисная архитектура должна обеспечивать независимое масштабирование отдельных компонентов системы в зависимости от нагрузки.

4.1.8 Обработка ошибок

В случае возникновения ошибок, пользователь должен получать соответствующие сообщение об ошибке. Приложение должно поддерживать обработку следующих основных ошибок:

- Некорректный ввод данных;
- Системный сбой;
- Ошибки авторизации и аутентификации;
- Ошибки при загрузке контента;
- Ошибки при работе с видео материалами.

4.2 Требования к видам обеспечения

4.2.1 Требования к информационному обеспечению

Требования по применению систем управления базами данных: PostgreSQL.

4.2.2 Требования к программному обеспечению приложения

Требования к программному обеспечению клиентской части:

- Приложение должно быть реализовано как PWA (Progressive Web App), что обеспечивает:
- Возможность установки на домашний экран устройства без необходимости загрузки из магазинов приложений;
- Push – уведомления для взаимодействия с пользователем;
- Приложение должно работать в веб-браузерах (Chrome, Firefox, Safari, Edge);
- Приложение должно работать без критических ошибок на мобильных устройствах под управлением iOS и Android.

Требования к программному обеспечению серверной части:

- Серверная часть приложения должна быть реализована на языке программирования Python с использованием фреймворка FastAPI;
- В качестве СУБД необходимо использовать PostgreSQL;
- Для контейнеризации микросервисов должен использоваться Docker;
- Для балансировки нагрузки и проксирования запросов должен использоваться Nginx;
- Для управления версиями кода должен использоваться GitHub.
- Ollama – это фреймворк для локального запуска крупных языковых моделей (LLM).

4.2.3 Требования к лингвистическому обеспечению

Приложение должно поддерживать русский язык. Язык ввода вывода данных русский. В дальнейшем возможно добавление поддержки других языков.

4.2.4 Требования к LLM компоненту

Сервис генерации мотивационной речи должен использовать модель DeepSeek R1 через Ollama для создания персонализированных мотивационных сообщений пользователям на основе их прогресса и целей тренировок.

5 Структура приложения

Приложение должно содержать следующие страницы/виды страниц:

- Страница регистрации;
- Страница входа;
- Страница восстановления пароля;
- Главная страница;
- Страница профиля пользователя (информация о пользователе);
- Страница безопасности;
- Страница финансов;
- Страница управлений подписками;
- Страница тренировок;
- Страница тренировок, составленных приложением;
- Страница настроек тренировки (у администратора);
- Страница пула упражнений (у администратора);
- Страница добавления упражнения (у администратора);
- Страница с курсами (блок Б);

- Страница профиля тренера (информация о тренере);
- Страница курса тренировок;
- Страница тренировки;
- Страница конструктора курсов (у тренера);
- Страница курса;
- Страница настройки курса (у тренера);
- Страница настройки тренировки (у тренера);
- Страница создания курса;

6 Дизайн приложения

Основные цвета приложения:

- Белый цвет (White #FFFFFF) – используется как цвет текста в приложении и в логотипе приложения;
- Оранжевый цвет (Vivid orange # F67B22) – используется как цвет рамок;
- Желтый (Yellow broom # FEA801) – используется в логотипе и названии приложения;
- Черный цвет (Black #000000) – используется как цвет рамок приложения, в логотипе приложения в качестве фона;
- Темно синий цвет (Very dark blue #2A2E31) – используется как цвет фона приложения;
- Серый цвет (Dark grayish blue #71717A) – используется как цвет иконок;

Вторичные цвета приложения:

- Оттенок оранжевого цвета (Moderate orange #BD7744) – используется в качестве дополнительного цвета;
- Оттенок серого цвета (Dark gray #999999) – используется как подсказка для текста на экранах;

При реализации приложения допускаются расхождения с данной цветовой палитрой, не нарушающие общий стиль приложения.

Основной шрифт приложения должен быть Lato (данный шрифт можно использовать в коммерческой и некоммерческой деятельности).

7 Навигация по приложению

Основным навигационным меню должно выступать нижнее меню (Bottom меню).

Нижнее меню должно включать в себя следующие кнопки:

- Главная страница;
- Тренировки;
- Курсы;
- Личный кабинет.

При переходе в личный кабинет незарегистрированного пользователя будет высвечиваться Рунет с предложением зарегистрироваться или войти в систему.

8 Описание страниц приложения

8.1 Описание страниц

8.1.1 Окно входа

Данная страница создана как для пользователей, которые ранее регистрировались на нашей платформе, так и для новых пользователей.

Содержит:

- Вход. Переход на главный экран;

- Регистрация. Переход на окно регистрации;
- Забыл пароль. Переход на страницу восстановление пароля;
- Начать без регистрации.

8.1.2 Окно регистрации

Данное окно создано для регистрации пользователя.

Содержит: Поле ввода логина и пароля пользователя;

8.1.3 Восстановление пароля

В данном окне вы можете заполнить форму по восстановлению пароля.

8.1.4 Главный экран

На этой странице пользователю предоставлены:

- Мотивирующие сообщение и советы от нейросети;
- График с изменениями веса и активностью пользователя;
- Все купленные тренировки и последние пройденные от приложения;

Помимо вышеперечисленного из главного меню пользователь, может используя нижнюю панель перейти в:

- Тренировки;
- Курсы;
- Личный кабинет.

8.1.5 Блок А

В данном блоке пользователь может выбрать тренировку, составленную приложением.

На этой странице представлен следующий функционал (для администратора):

- Пул упражнений. В данном окне администратор может посмотреть все упражнения на разные группы мышц, а также добавить новое упражнение;
- Настройки тренировки. В этом окне администратор может добавлять или удалять упражнения из тренировки и указывать количество повторений время выполнения, а также дать название и, при необходимости, полностью удалить тренировку;

Пользователь в данном блоке видит только готовые тренировки от приложения, при нажатии на которые он видит следующее окно:

- Тренировки из блока А. Здесь представлены тренировки, составленные приложением. На данном окне представлен следующий функционал:
- Видео упражнения для его визуализации;
- Сколько по времени или повторений делать упражнение;
- Возможность перехода между упражнениями (Предыдущее/Следующее упражнение);
- Оставшееся время тренировки.

8.1.6 Блок Б

В данном блоке пользователь может ознакомиться с разными курсами, выложенными тренерами. Помимо этого, он может перейти в свой профиль, а также все созданные им курсы.

Блок со следующим функционалом:

- Фильтр/Поиск. Пользователь может фильтровать курсы или найти тот, который ему нужен.
- Ваши курсы. На данном окне пользователь может просмотреть все созданные им курсы.

- Курсы от тренеров, где есть название курса, краткое описание, рейтинг, а также включённые группы мышц, помимо этого пользователь может перейти от курса на аккаунт тренера, создавшего этот курс.

Чтобы пользователю создать и опубликовать свой курс, он нажимает на кнопку «Ваши курсы», попадая в конструктор курсов. Здесь он может изменить уже существующие курсы (если они есть) или создать новый.

При создании нового курса пользователю необходимо ввести следующие данные:

- Название курса
- Описание
- Цена

Перейдя на конкретный курс, созданный ранее, пользователь видит все тренировки, включённые в этот курс, а также у него есть возможность создать новую, для этого он попадает в следующий конструктор, где у него есть возможность указать следующие параметры:

- Название тренировки;
- Описание урока;
- Настройки группы мышц;
- Ссылка на видео;
- Удалить урок;
- Опубликовать / Снять с публикации;
- Сделать бесплатным/ указать цену;

Помимо этого, во вкладке Курс тренеру доступны настройки курса, куда входит:

- Иконка курса;
- Название курса;
- Описание;
- Цена. С возможностью выбора: Платно/Бесплатно;
- Опубликовать/Снять с публикации;
- Удалить курс;

Также в блоке Б для пользователя доступен переход на Профиль тренера, который представляет из себя:

- Аватар профиля;
- Рейтинг тренера;
- Описание профиля;
- Курсы тренировок с описанием, ценой и общим рейтингом.
- Тренировка:
- Видео;
- Текстовое описание от тренера (прим. Какие группы мышц задействуются);
- Рейтинг;
- Комментарии.

8.1.7 Личный кабинет

В данном блоке пользователь может указать/отредактировать свои ФИО, аватар, написать описание профиля, управлять подписками и финансами или же выйти из профиля.

- Безопасность. Здесь пользователь может сменить электронную почту, а также поменять пароль;

- Управление подписками. Пользователь может отменять подписки на купленные им курсы;
- Финансы. Пользователь указывает свои реквизиты для списывания средств. Также здесь реализованы реквизиты для зачисления для пользователей, выложивших хотя бы одну платную тренировку;
- Выйти. Выход из аккаунта;

Приложение А

Все приложения также выложены на GitHub команды, где их легче разобрать.

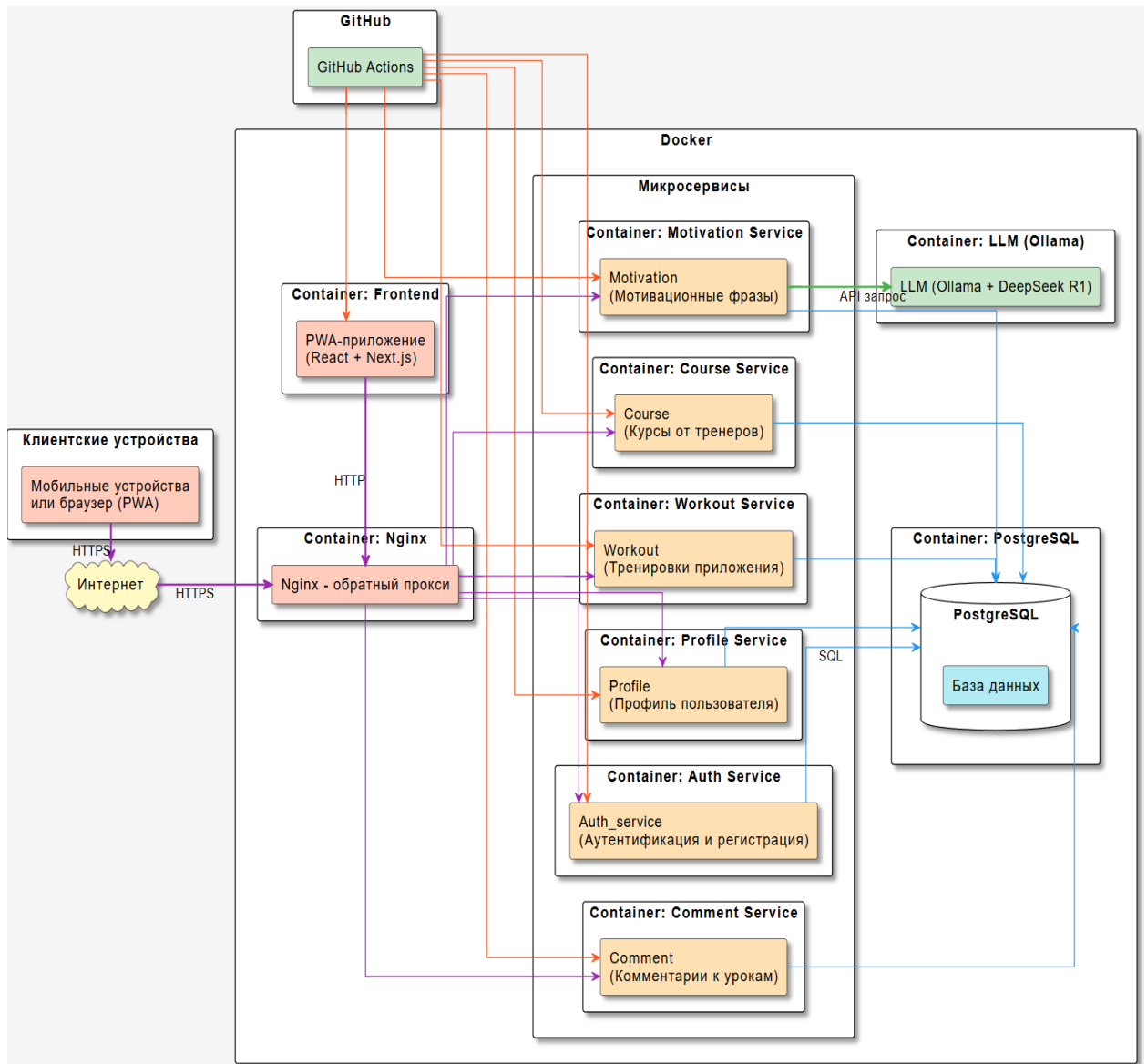


Рисунок 1 - Диаграмма развёртывания.

Приложение Б

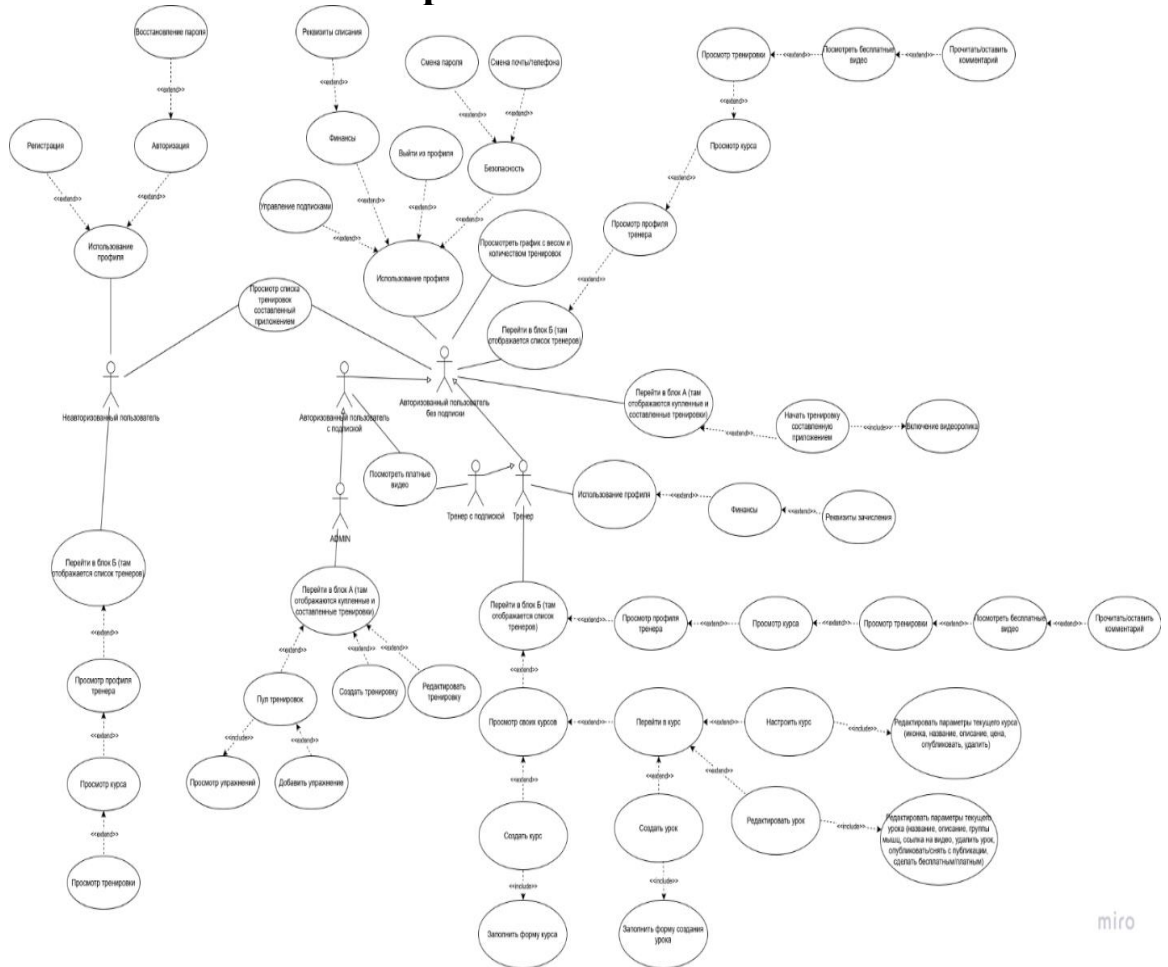


Рисунок 2 - UseCase диаграмма.

Приложение В

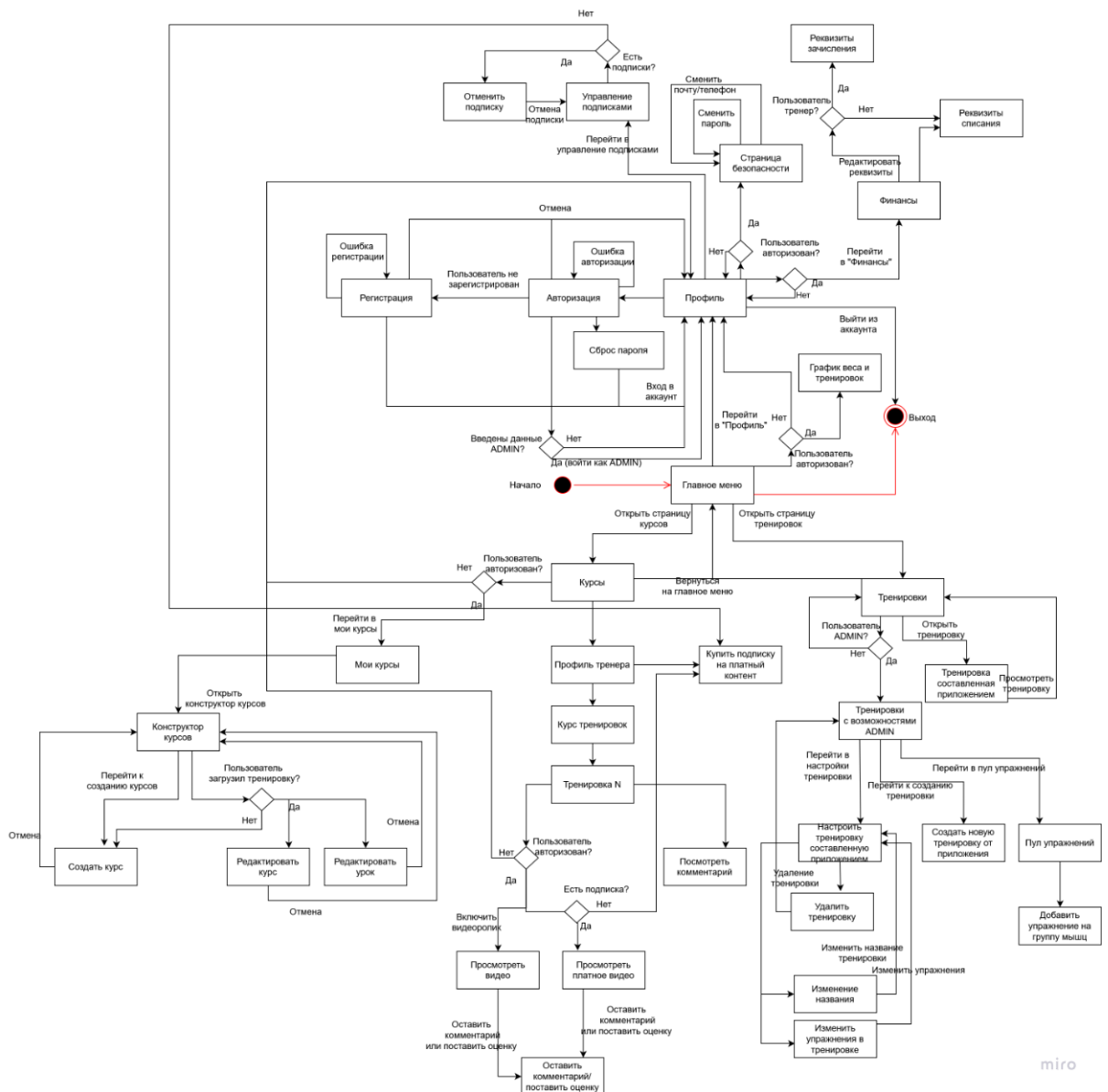


Рисунок 3 - Диаграмма состояний

Приложение Г

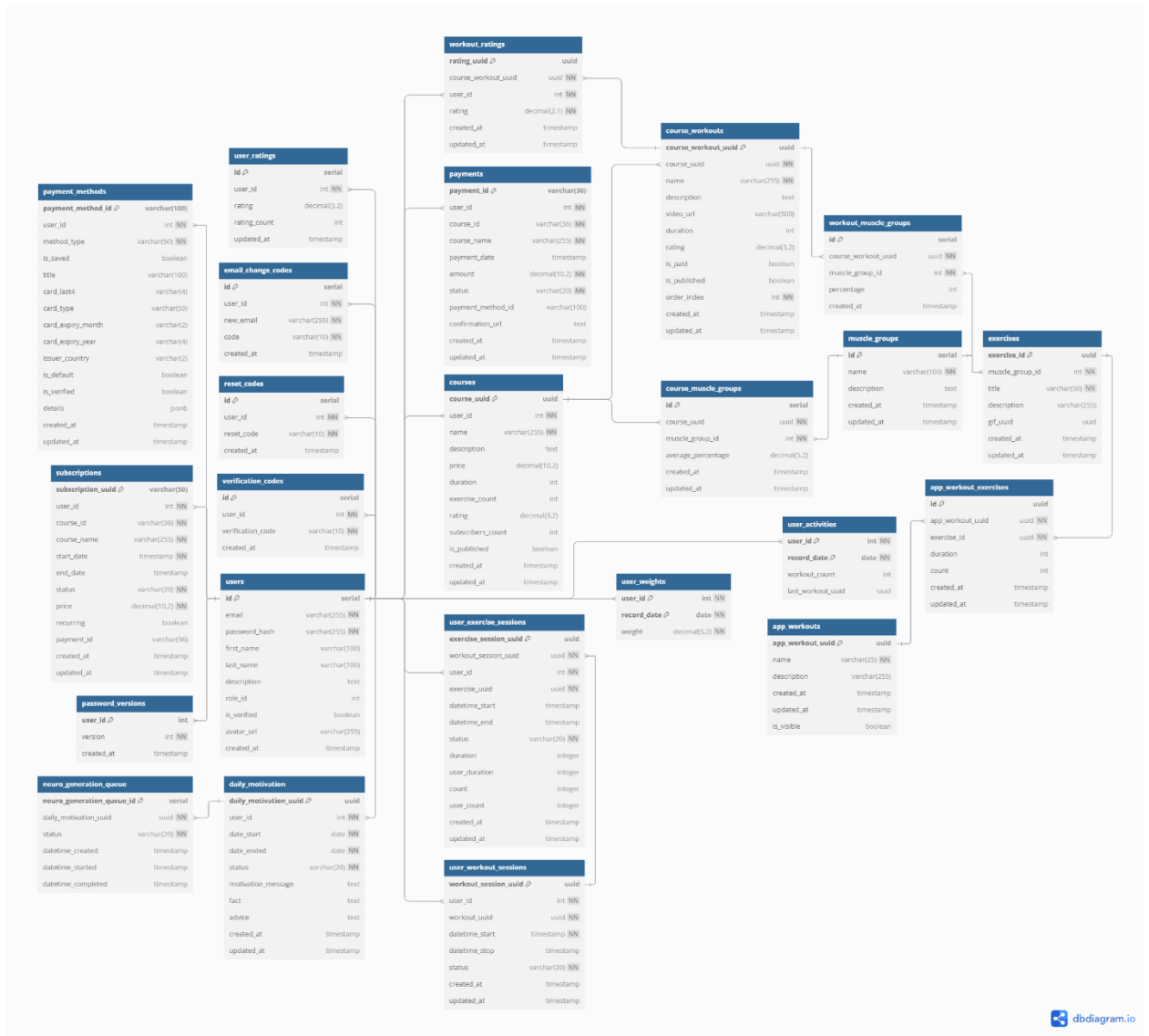
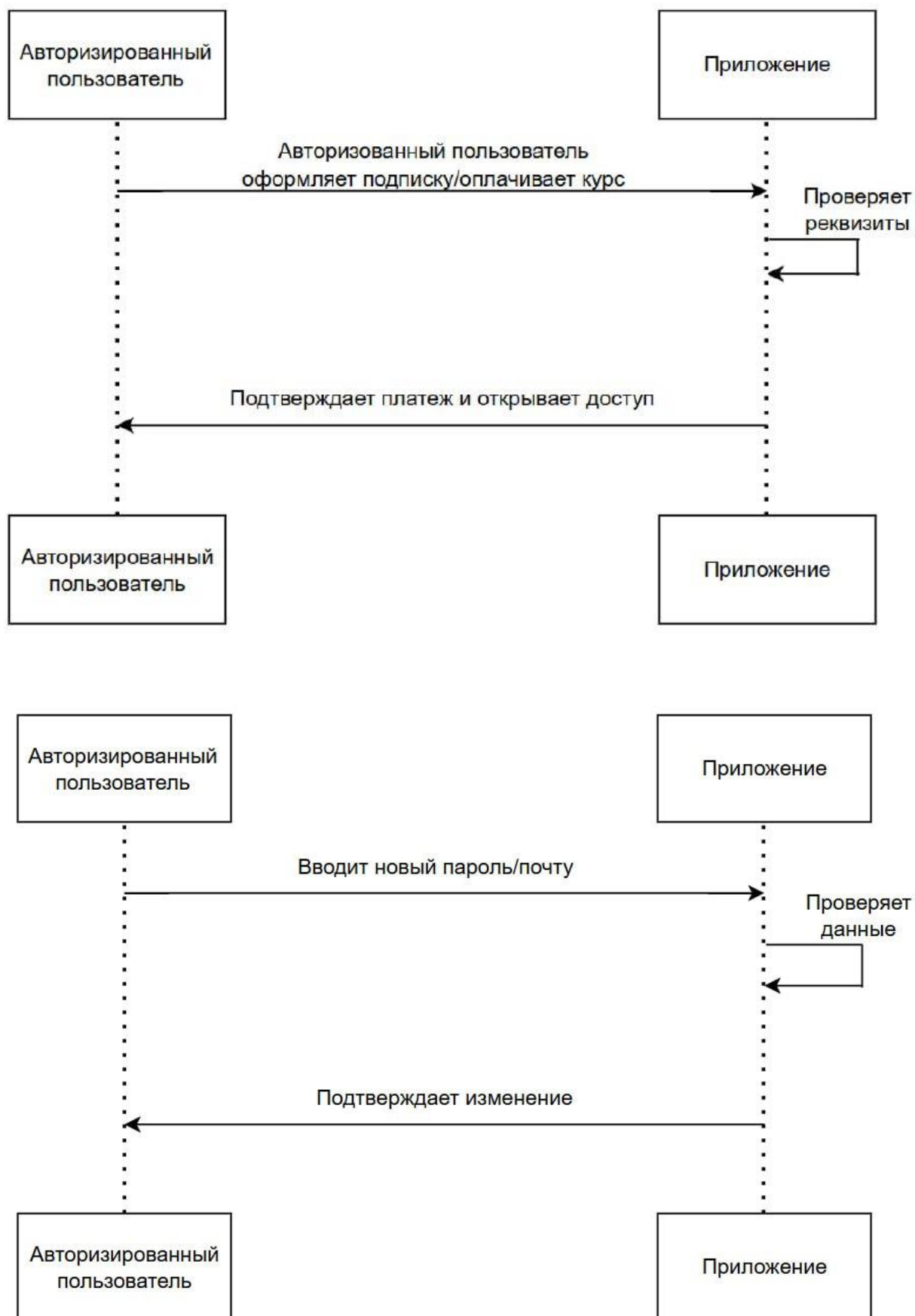


Рисунок 4 - ER диаграмма

Приложение Д





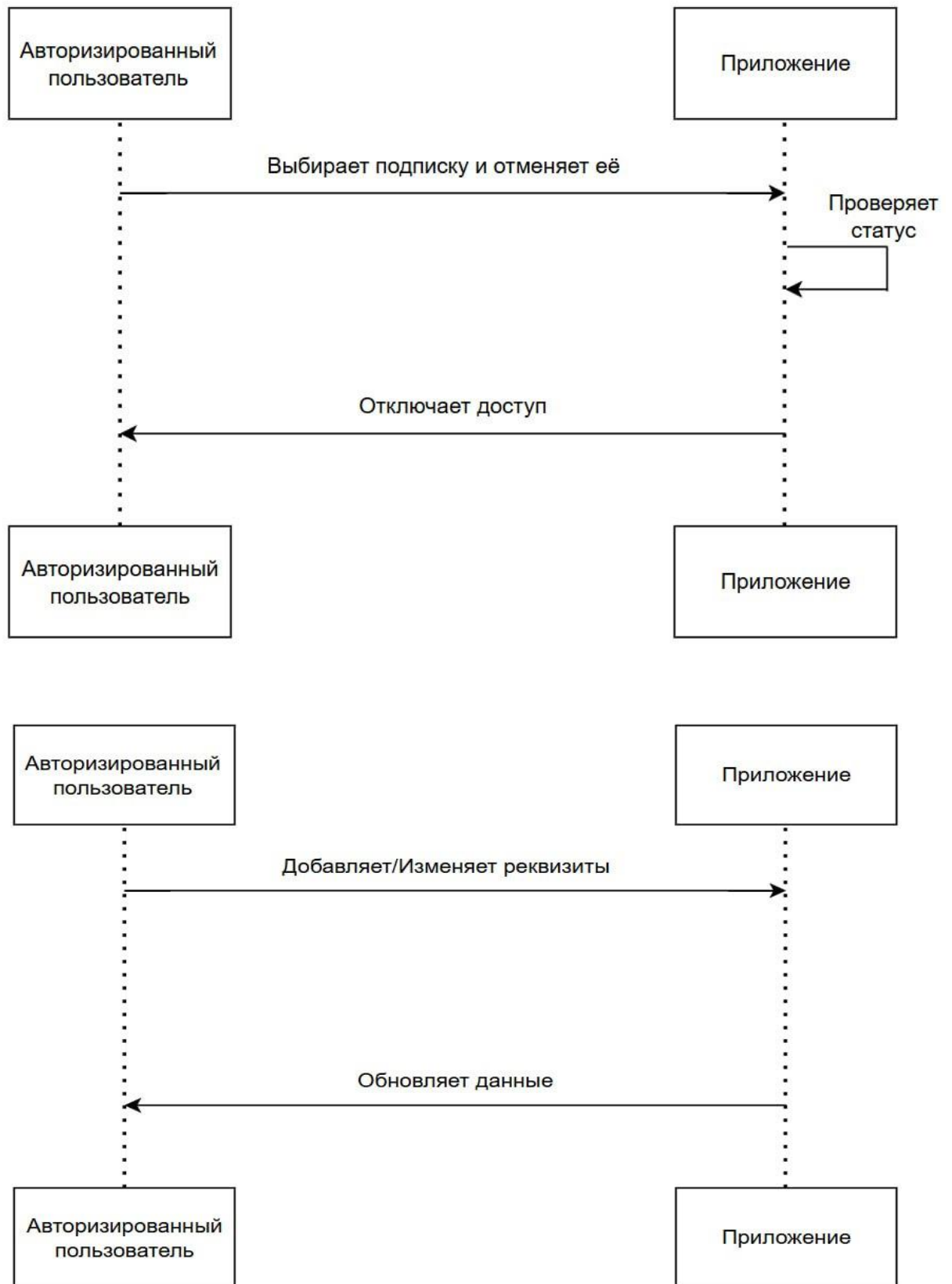


Рисунок 5 - Диаграмма последовательностей