

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
для организации совместных поездок
«TripShare»

Исполнители

_____ А.А. Тарарыков

_____ Е.А. Рогачев

_____ В.Е. Платон

Заказчик

_____ В.С. Тарасов

Воронеж 2024

Содержание

1	Термины, используемые в техническом задании	4
2	Общие сведения	5
2.1	Полное наименование системы и название приложения	5
2.2	Наименование исполнителя и заказчика приложения	5
2.2.1	Наименование заказчика	5
2.2.2	Наименование исполнителя	5
2.3	Перечень документов, на основании которых создается приложение .	6
2.4	Плановые сроки начала и окончания работ	6
2.5	Состав и содержание работ по созданию приложения	6
2.6	Порядок оформления и предъявления заказчику результатов работ по созданию приложения	7
3	Назначение и цели создания приложения	8
3.1	Назначение приложения	8
3.2	Цели создания приложения	8
3.3	Задачи, решаемые при помощи приложения	8
4	Требования к приложению и программному обеспечению	10
4.1	Требования к архитектуре системы	10
4.2	Требования к функциям, выполняемым приложением	10
4.3	Требования к технологиям, используемым при создании приложения	11
4.4	Требования к программному обеспечению приложения	11
4.5	Требования к внешним интерфейсам	12
4.6	Общие требования к оформлению и верстке страниц	12
4.7	Требования к численности и квалификации персонала, обслуживающего приложение	12

4.8	Требования к форматам и размеру данных в приложении	13
4.9	Требования к системе администрирования	13
5	Структура приложения.....	14
5.1	Динамические страницы	15
5.2	Статические страницы.....	16
6	Языковые версии приложения.....	17
7	Группы пользователей	18
8	Дизайн приложения	19
9	Навигация по приложению	20
10	Описание страниц приложения	21
10.1	Страница авторизации	21
10.2	Страница профиля.....	22
10.3	Страница редактирования информации о себе	23
10.4	Страница редактирования предпочтений	24
10.5	Страница добавления автомобиля.....	25
10.6	Страница настроек аккаунта	26
10.7	Страница поиска поездки.....	27
10.8	Страница создания поездки	28
10.9	Страница запланированных поездок.....	29
10.10	Страница чатов.....	30
11	Функциональные возможности приложения	31
12	Контент и наполнение приложения	40
13	Источники разработки.....	41
14	Порядок контроля и приемки работ.....	42
15	Реквизиты и подписи сторон	43
	Приложение.....	44

1 Термины, используемые в техническом задании

Термин	Определение термина
FastAPI	Современный веб-фреймворк для разработки API на языке Python.
HTTPS (Hypertext Transfer Protocol Secure)	Защищенная версия протокола HTTP, который используется для передачи данных в Интернете.
JWT (JSON Web Token)	Компактный и самодостаточный способ представления информации об аутентификации и авторизации в форме структурированного JSON-объекта.
ORM (Object-Relational Mapping)	Техника программирования, которая позволяет разработчикам использовать объектно-ориентированный подход при работе с реляционными базами данных.
Pydantic	Библиотека для проверки и валидации данных в Python, которая работает на основе аннотаций типов данных и предоставляет инструменты для создания схем данных (data schemas).
Vue.js	Прогрессивный JavaScript фреймворк для создания пользовательских интерфейсов и одностраничных приложений.
Yandex Maps Геосаджест API	Сервис, предоставляемый компанией Яндекс, который позволяет веб-разработчикам интегрировать в свои приложения функциональность предложения автозаполнения (автокомплита) и подсказок поиска на основе географических данных.
Гарантированное получение оплаты	Процесс, предоставляемый платежным сервисом, который обеспечивает поставщикам уверенность в том, что они получают оплату за свои услуги.
ГОСТ 34.602-2020	Стандарт, который определяет правила для организации и функционирования систем обработки информации (СОИ).
Минималистичный стиль	Эстетический подход в дизайне и других областях, характеризующийся простотой, ясностью форм и ограниченным использованием декоративных элементов.
Динамическая страница	Страница, содержание или элементы которой могут быть изменены или обновлены без необходимости полной перезагрузки страницы.

2 Общие сведения

2.1 Полное наименование системы и название приложения

Приложение для организации совместных поездок, включающее профили пользователей, систему отзывов и расчет стоимости поездки.

Наименование приложения: TripShare.

2.2 Наименование исполнителя и заказчика приложения

2.2.1 Наименование заказчика

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

2.2.2 Наименование исполнителя

Исполнитель: студент Тарарыков Антон Алексеевич. Воронежский Государственный Университет, Факультет компьютерных наук, кафедра Программирования и информационных технологий.

Исполнитель: студент Рогачев Егор Андреевич. Воронежский Государственный Университет, Факультет компьютерных наук, кафедра Программирования и информационных технологий.

Исполнитель: студент Платон Василий Евгеньевич. Воронежский Государственный Университет, Факультет компьютерных наук, кафедра Программирования и информационных технологий.

2.3 Перечень документов, на основании которых создается приложение

Данное приложение будет создаваться на основании настоящего Технического Задания, составленного в соответствии с ГОСТ 34.602-2020.

2.4 Плановые сроки начала и окончания работ

Плановый срок начала работ – февраль 2024 г.

Плановый срок окончания работ – 10 июня 2024 г.

2.5 Состав и содержание работ по созданию приложения

Состав и содержание работ по созданию приложения включают в себя следующие этапы:

- Предпроектное исследование, сбор необходимой информации.
Результат: определение целей, задач системы, которые в дальнейшем должны быть решены;
- Анализ предметной области. Результат: подробный анализ системы и введение организационных требований к решению задач и целей;
- Разработка ТЗ. Результат: документация на разрабатываемую систему, в которой указаны сроки реализации, кем будет реализована, для кого, описаны все необходимые организационные требования к разработке;
- Разработка модели программы. Результат: описание спецификаций данных, определение связей между сущностями, построение концептуальной модели БД, построение логической модели БД;

- Разработка рабочего проекта, состоящего из написания, отладки и корректировки программы;
- Проведение тестирования и доработка информационного программного обеспечения по замечаниям и предложениям;
- Сдача системы в эксплуатацию с выпуском описания алгоритмов и технологической документации.

2.6 Порядок оформления и предъявления заказчику результатов работ по созданию приложения

Результаты работ предъявляются Заказчику в следующем виде:

- Настоящее Техническое Задание по Гост 34.602-2020;
- Документация к приложению;
- Презентация в формате видео с демонстрацией функциональности приложения;
- Курсовой проект на основе настоящего Технического Задания
- Защита проекта;
- Работающее, согласно настоящему Техническому Заданию, мобильное приложение;
- Исходный код приложения.

3 Назначение и цели создания приложения

3.1 Назначение приложения

Приложение для организации совместных поездок предназначено для упрощения процесса планирования поездок и совместного использования транспорта. В приложении предусмотрена возможность поиска совместных поездок, создание профилей пользователей, система отзывов и рейтингов, расчёт стоимости поездки.

3.2 Цели создания приложения

К целям создания приложения относятся:

- Уменьшение транспортных пробок и загрязнения окружающей среды;
- Экономическая выгода для пользователей;
- Улучшение социальной связанности и общения пассажира и водителя;
- Возможность организовать совместную поездку.

3.3 Задачи, решаемые при помощи приложения

Приложение должно позволять пользователям следующее:

- Поиск подходящих попутчиков;
- Поиск подходящей поездки;
- Управление графиком поездок;
- Предварительное планирование поездки;
- Оценка надёжности и безопасности пользователей;

- Расчёт стоимости поездки;
- Гарантированное получение оплаты за поездку водителем;
- Гарантированный возврат средств пассажирам за несостоявшиеся поездки.

4 Требования к приложению и программному обеспечению

4.1 Требования к архитектуре системы

Пользователь взаимодействует с серверной частью приложения при помощи клиентского приложения (мобильное приложение). Сервер посредством REST API возвращает пользователю необходимые данные. Администратор взаимодействует с серверной частью приложения при помощи клиентского приложения (веб-браузер).

Приложение также должно иметь базу данных, для хранения данных зарегистрированных пользователей, профилей пользователей, запланированных и завершённых поездок, отзывов, расходов, связанных с поездкой.

4.2 Требования к функциям, выполняемым приложением

Функции, выполняемые приложением:

- Возможность регистрации и авторизации пользователей. Пользователи могут создавать учётные записи и входить в систему с помощью логина и пароля;
- Возможность создания и редактирования профилей пользователей;
- Возможность поиска совместных поездок;
- Возможность оценивания и написания отзывов. Пользователи могут оставлять отзывы и оценки другим участникам поездки после завершения поездки для обеспечения безопасности и надёжности;
- Возможность расчёта стоимости поездки. Приложение должно рассчитывать стоимость поездки на основе заданных параметров;

- Возможность управление поездками;
- Возможность получения уведомлений. Приложение должно уведомлять пользователей о новых поездках, обновлениях в существующих поездках и отзывах.

4.3 Требования к технологиям, используемым при создании приложения

Код приложения должен быть написан на языке программирования Kotlin 1.9.20, чтобы обеспечить совместимость с Android 10. Серверная часть должна использовать язык Python с SQL базой данных. Рекомендуется использовать fastapi admin для административной панели сервера, однако можно рассмотреть и другие варианты.

4.4 Требования к программному обеспечению приложения

Требования к программному обеспечению клиентской части:

- Приложение должно быть совместимо с мобильными устройствами на базе Android 10 и выше.

Требования к программному обеспечению серверной части:

- Сервер должен быть реализован на Python с использованием FastAPI.
- Для базы данных необходимо использовать PostgreSQL.
- Для ORM следует использовать sqlalchemy.
- Все JSON запросы и ответы должны проходить валидацию с помощью pydantic. Документация должна быть предоставлена в формате Swagger.

4.5 Требования к внешним интерфейсам

В приложении должна быть реализована поддержка следующих API:

— Yandex Maps Геосаджест API для поиска и проверки адреса.

4.6 Общие требования к оформлению и верстке страниц

Все страницы приложения должны соответствовать общему стилю. Цветовая палитра должна включать в себя цвета, используемые на логотипе приложения.

Мобильное приложение должно быть оптимизировано для экранов с диагональю от 5 дюймов.

Панель администратора (веб-приложение) должна поддерживать разрешение экрана от 1270 x 720 пикселей.

Для разработки дизайна мобильного приложения будет использоваться язык программирования Kotlin 1.9.20.

Для разработки дизайна панели администратора будут использоваться следующие технологии:

- HTML 5;
- CSS 3;
- JavaScript ES2022;
- Vue.js 3.0.0.

4.7 Требования к численности и квалификации персонала, обслуживающего приложение

Приложение должно поддерживаться как минимум 3 администраторами, ознакомленными с правилами работы с приложением. Специальной подготовки, кроме умения работы с таблицами и панелью администратора, не требуется.

4.8 Требования к форматам и размеру данных в приложении

Приложение и сервер должны взаимодействовать друг с другом посредством REST API, поэтому основным форматом данных будет JSON. Дополнительно, необходимо обеспечить возможность загрузки изображений в формате PNG с максимальным разрешением 2000 x 2000 пикселей и размером, не превышающим 2 МБ.

4.9 Требования к системе администрирования

Через систему администрирования администраторы должны иметь возможность просмотра, добавления, редактирования и удаления профилей пользователей и отзывов, иметь возможность обработки жалоб на нежелательное поведение или контент, возможность управления системой рейтинга, возможность просмотра и возврата платежей, возможность предоставления технической поддержки.

5 Структура приложения

Приложение требует архитектуры, соответствующей модели клиент-серверного взаимодействия, основанной на REST API, с обменом запросами и ответами через HTTPS. Сервер должен взаимодействовать с базой данных для сохранения и получения данных, как показано на диаграмме развертывания приложения на рисунке ниже.

Аутентификация клиента должна происходить с использованием токена JWT. Клиентский запрос должен содержать всю необходимую информацию о состоянии сессии, что позволит серверу не сохранять эту информацию.

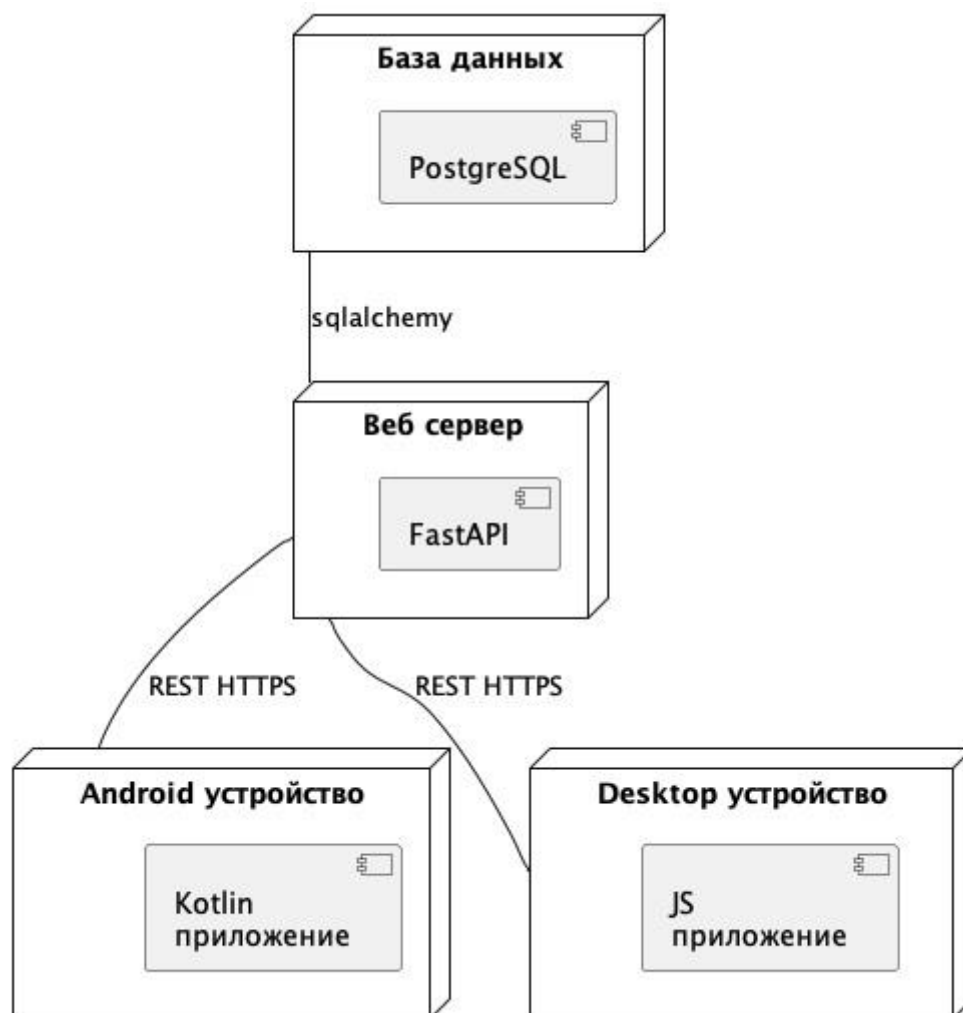


Рисунок 1 - Диаграмма развёртывания приложения

Приложение должно содержать следующие страницы и виды страниц:

5.1 Динамические страницы

- Приветственная страница для первого запуска;
- Страница поиска поездок;
- Страница с результатом поиска поездок;
- Страница деталей поездки;
- Страница проверки деталей поездки;
- Страница создания поездки;
- Страница авторизации;
- Страница регистрации;
- Страница профиля;
- Страница редактирования информации о себе;
- Страница редактирования предпочтений;
- Страница профиля другого пользователя;
- Страница всех чатов;
- Страница чата;
- Страница подтверждения бронирования;
- Страница редактирования поездки;
- Страница всех запланированных поездок;
- Страница настроек аккаунта;
- Страница оплаты;
- Страница просмотра отзывов;

- Страница создания отзыва;
- Страница добавления автомобиля.

5.2 Статические страницы

- Страница с информацией о приложении.

6 Языковые версии приложения

Приложение должно быть предоставлено для взаимодействия с пользователем на русском языке.

7 Группы пользователей

Приложение должно содержать следующие группы пользователей:

- Неавторизованные пользователи;
- Администраторы;
- Авторизованные пользователи.

8 Дизайн приложения

Приложение должно быть выполнено в минималистичном стиле.

Основные цвета приложения:

- Оттенок серого цвета (#F2F2F2) - используется как цвет фона приложения;
- Белый цвет (White - #FFFFFF) – используется как цвет фона компонентов приложения и навигационной панели;
- Мятный цвет (Mint - #3EB489) - используется как цвет активных иконок на нижней навигационной панели и кнопок, а также используется в логотипе приложения.

Вторичные цвета приложения:

- Оттенок фиолетового цвета (#B984EE) - используется как акцентный цвет;
- Оттенок синего (#33B6FE) - используется как акцентный цвет и используется в логотипе приложения;
- Оттенок красного цвета (#F62123) - используется как цвет уведомлений;
- Чёрный цвет (#000000) – используется как цвет рамок и текста;
- Оттенок серого (#737373) – используется как цвет текста и неактивных иконок.

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

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

9 Навигация по приложению

Навигация по приложению осуществляется с помощью навигационного меню внизу экрана.

Меню должно содержать следующие кнопки:

- «Поиск поездки» - открытие страницы поиска поездок;
- «Организовать поездку» - открытие страницы создания поездки;
- «Запланированные поездки» - открытие страницы всех запланированных поездок;
- «Чаты» - открытие страницы всех чатов;
- «Профиль» - открытие страницы профиля.

Прототип навигационной панели представлен на рисунке ниже.

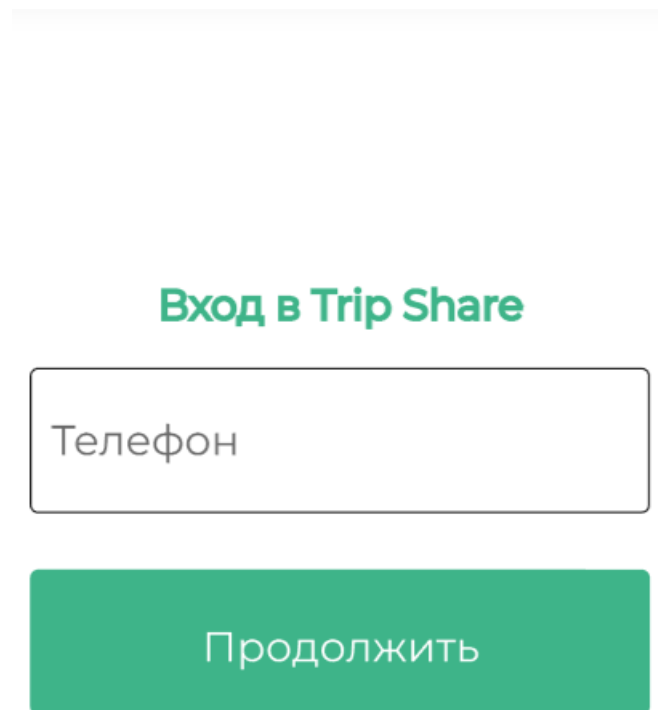


Рисунок 2 - Навигационная панель

10 Описание страниц приложения

10.1 Страница авторизации

Страница авторизации позволяет неавторизованным пользователям получить доступ к своему аккаунту или создать новый. Прототип страницы авторизации представлен на рисунке ниже.



Прототип страницы авторизации приложения Trip Share. В центре экрана находится заголовок "Вход в Trip Share" зеленого цвета. Ниже него — белое текстовое поле с закругленными углами и серой рамкой, содержащее текст "Телефон". Под полем расположен крупный зеленый прямоугольный button с закругленными углами и белым текстом "Продолжить".

Рисунок 3 - Страница авторизации

10.2 Страница профиля

Страница профиля позволяет авторизованным пользователям ознакомиться с информацией о себе и управлять своим аккаунтом. Прототип страницы профиля представлен на рисунке ниже.



Рисунок 4 - Страница профиля

10.3 Страница редактирования информации о себе

Страница редактирования информации о себе позволяет авторизованным пользователям обновить профиль, чтобы он всегда оставался актуальным. Прототип страницы редактирования информации о себе представлен на рисунке ниже.

Редактирование информации о себе

Имя

АНТОН

Фамилия

Тарарыков

Телефон

89514960549

Почта

anthonytararykov@yandex.ru

Дата рождения

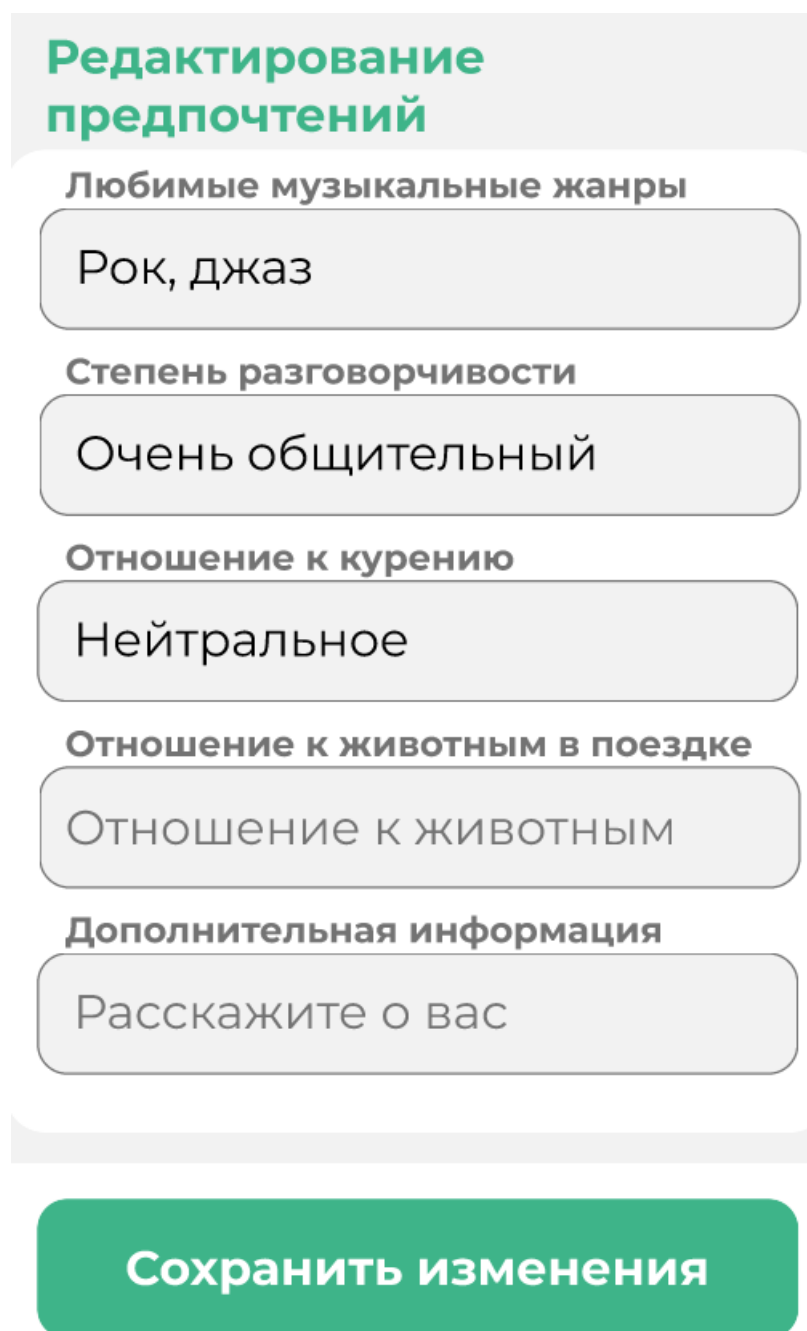
31.02.2020

Сохранить изменения

Рисунок 5 - Страница редактирования информации о себе

10.4 Страница редактирования предпочтений

Страница редактирования предпочтений позволяет авторизованным пользователям настроить профиль для наилучшего соответствия их предпочтениям и потребностям в поездках. Прототип страницы редактирования предпочтений представлен на рисунке ниже.



Прототип страницы редактирования предпочтений. Визуально это представляет собой серию вертикально расположенных элементов. В начале находится заголовок «Редактирование предпочтений» в зеленом цвете. Далее следуют пять секций, каждая с заголовком и полем ввода: «Любимые музыкальные жанры» (Рок, джаз), «Степень разговорчивости» (Очень общительный), «Отношение к курению» (Нейтральное), «Отношение к животным в поездке» (Отношение к животным) и «Дополнительная информация» (Расскажите о вас). В конце страницы находится большая зеленая кнопка «Сохранить изменения».

Редактирование предпочтений

Любимые музыкальные жанры
Рок, джаз

Степень разговорчивости
Очень общительный

Отношение к курению
Нейтральное

Отношение к животным в поездке
Отношение к животным

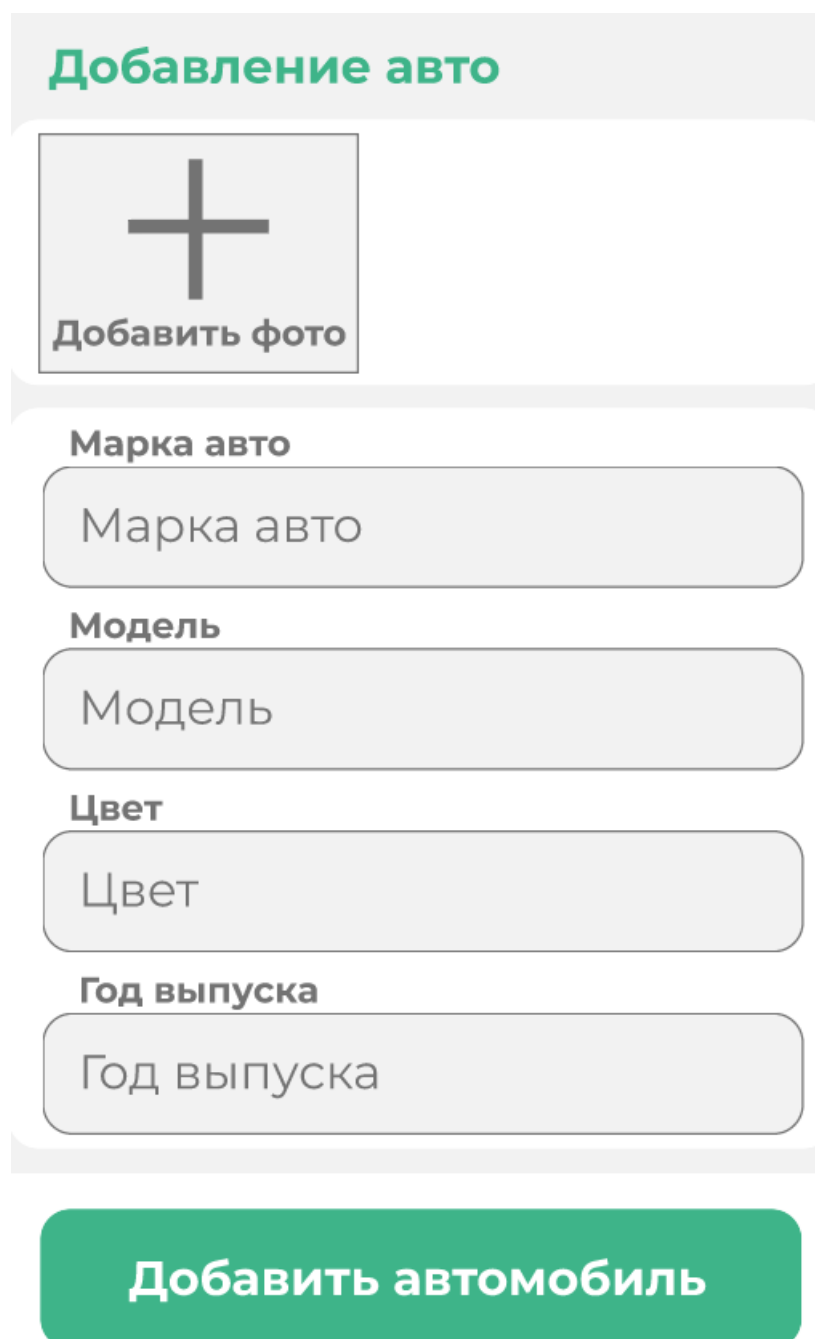
Дополнительная информация
Расскажите о вас

Сохранить изменения

Рисунок 6 - Страница редактирования предпочтений

10.5 Страница добавления автомобиля

Страница добавления автомобиля позволяет авторизованным пользователям добавить информацию об автомобиле. Прототип страницы добавления автомобиля представлен на рисунке ниже.



Добавление авто

+

Добавить фото

Марка авто

Марка авто

Модель

Модель

Цвет

Цвет

Год выпуска

Год выпуска

Добавить автомобиль

The image shows a UI prototype for adding a car. It features a light gray background with a white rounded rectangle containing the form. At the top, a green header bar with white text reads 'Добавление авто'. Below this is a square button with a large gray plus sign and the text 'Добавить фото'. The form then consists of four labeled input fields: 'Марка авто', 'Модель', 'Цвет', and 'Год выпуска', each with a light gray placeholder text. At the bottom, a wide green button with white text reads 'Добавить автомобиль'.

Рисунок 7 - Страница добавления автомобиля

10.6 Страница настроек аккаунта

Страница настроек аккаунта позволяет авторизованным пользователям сменить пароль от аккаунта, добавить или редактировать способ оплаты, посмотреть историю платежей, связаться с технической поддержкой, выйти из аккаунта, а также удалить аккаунт. Прототип страницы добавления автомобиля представлен на рисунке Рисунок 8 -ниже.

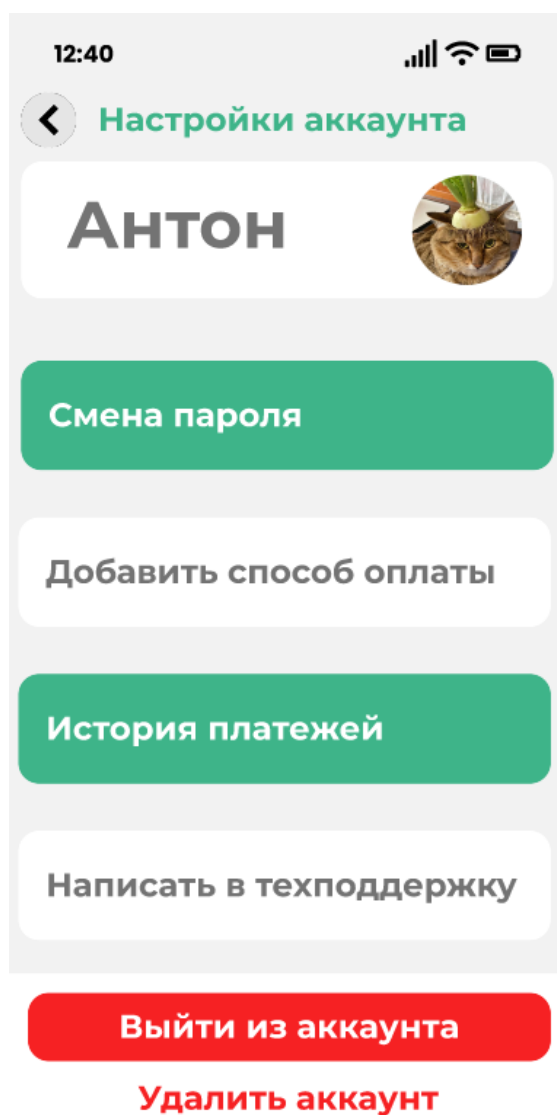


Рисунок 8 - Страница настроек аккаунта

10.7 Страница поиска поездки

Страница поиска поездки позволяет пассажирам указать адрес отправления, указать адрес прибытия, указать дату отправления, указать время отправления, указать необходимое количество пассажиров, выбрать необходимые удобства и нажать на кнопку «Поиск». Прототип страницы поиска поездки представлен на рисунке ниже.

12:40

Поездки по самым низким ценам

Откуда

Куда

Сегодня

15:10 - 16:00

1

☐ Максимум 2 пассажира на заднем сидении

☐ Можно курить ☐ Свободный багажник

☐ Можно перевозить животных

Поиск

Поиск поездки

Организовать поездку

Запланированные поездки

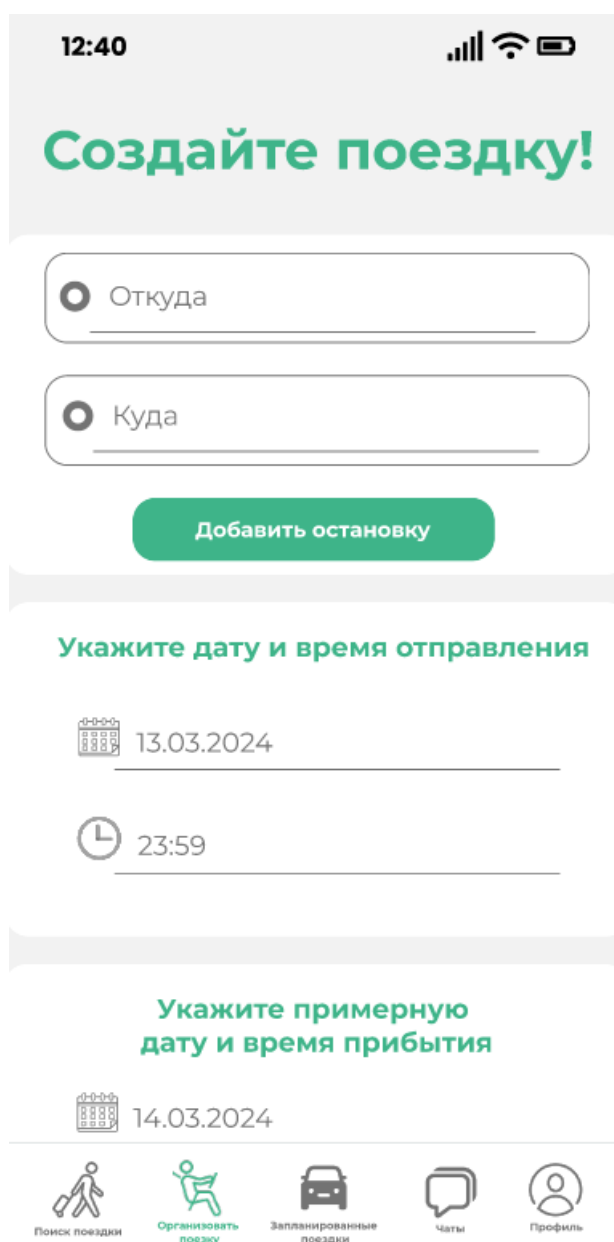
Чаты

Профиль

Рисунок 9 - Страница поиска поездки

10.8 Страница создания поездки

Страница создания поездки позволяет водителям указать точку отправления, точку прибытия, добавить остановку, указать дату и время отправления, указать примерную дату прибытия и нажать на кнопку «Создать поездку». Прототип страницы создания поездки представлена на рисунке ниже.



12:40

Создайте поездку!

Откуда

Куда

Добавить остановку

Укажите дату и время отправления

13.03.2024

23:59

Укажите примерную дату и время прибытия

14.03.2024

Поиск поездки

Организовать поездку

Запланированные поездки

Чаты

Профиль

Рисунок 10 - Страница создания поездки

10.9 Страница запланированных поездок

Страница запланированных поездок позволяет авторизованным пользователям увидеть список запланированных поездок с краткой информацией о поездках. Прототип страницы запланированных поездок представлен на рисунке ниже.

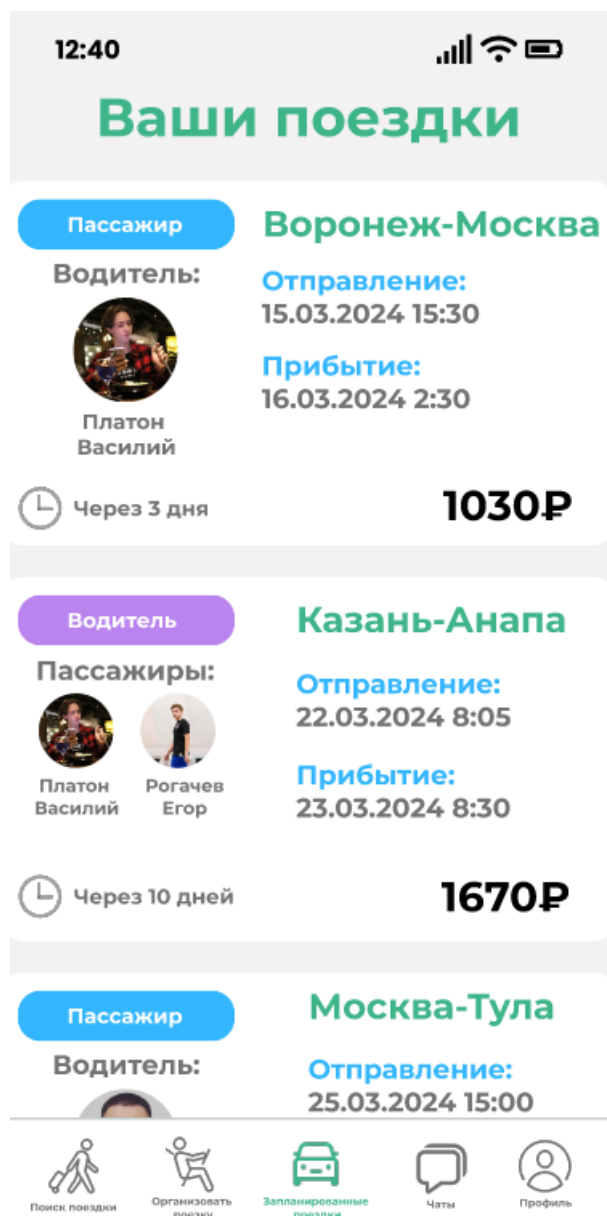


Рисунок 11 - Страница запланированных поездок

10.10 Страница чатов

Страница чатов позволяет авторизованным пользователям увидеть список чатов, в которых участвуют пользователи. Прототип страницы чатов представлен на рисунке ниже.

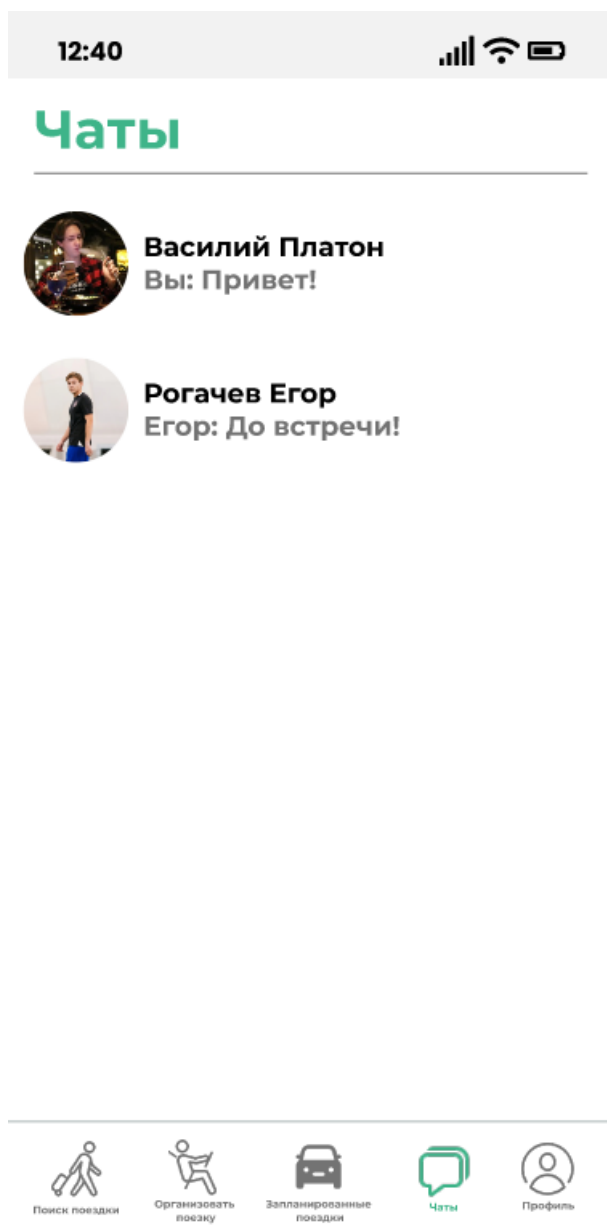


Рисунок 12 - Страница чатов

11 Функциональные возможности приложения

Разработаны отдельные функциональные возможности для каждой группы пользователей.

Для пассажира:

- Найти поездки;
- Отфильтровать поездки;
- Просмотреть список подходящих поездок;
- Посмотреть детали поездки;
- Выбрать сортировку поездок;
- Выбрать удобства, предоставляемые в поездке;
- Выбрать конкретную поездку.

Карта пользовательских историй для пассажира представлена ниже.

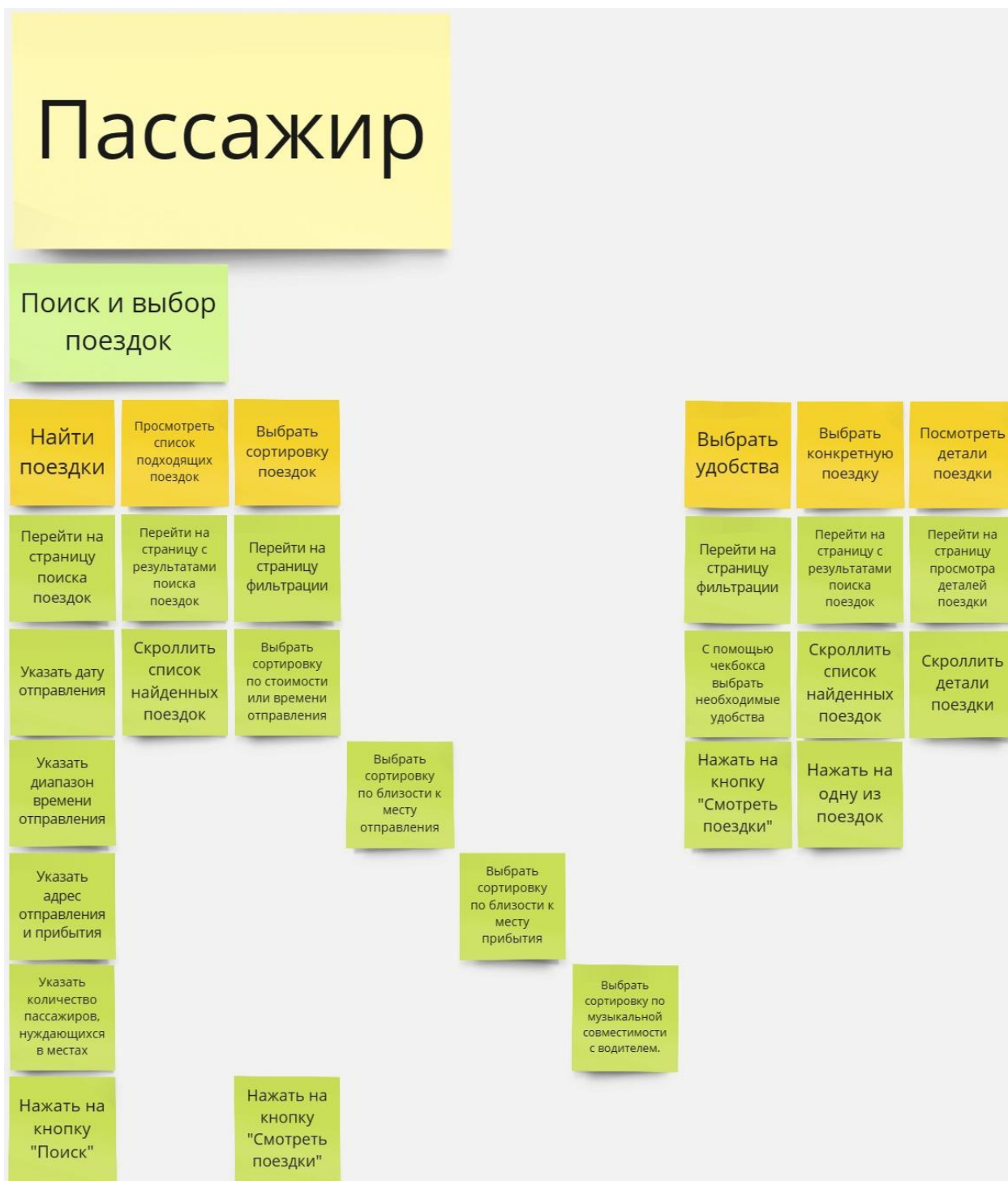


Рисунок 13 - Карта пользовательских историй для пассажира

Функциональной возможностью для водителя является создание поездки. Карта пользовательских историй для водителя представлена на рисунке ниже.

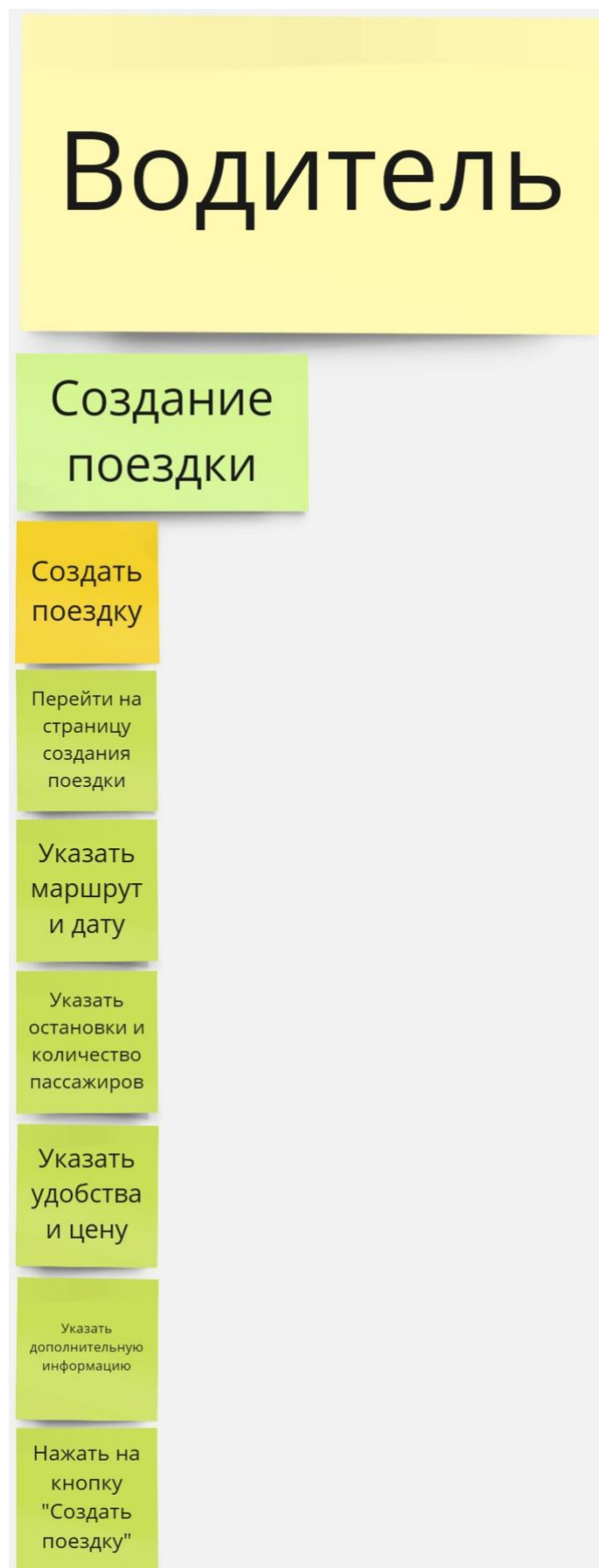


Рисунок 14 - Карта пользовательских историй для водителя

Функциональной возможностью для авторизованного пассажира является отправка запроса на бронирование поездки. Карта пользовательских историй для авторизованного пассажира представлена на рисунке ниже.

Авторизованный пассажир

Отправить
запрос на
бронирование

Отправить
запрос на
бронирование

Перейти на
страницу с
детальными
поездки

Нажать на
кнопку
"Забронировать"

Перейти на
страницу
проверки
деталей
поездки

Проверить
информацию
о поездке

Нажать на
кнопку
"Отправить
запрос"

Рисунок 15 - Карта пользовательских историй для авторизованного пассажира

Функциональные возможности для неавторизованного пользователя:

- Авторизоваться;
- Зарегистрироваться.

Карта пользовательских историй для неавторизованного пользователя представлена на рисунке ниже.



Рисунок 16 - Карта пользовательских историй для неавторизованного пользователя

Функциональные возможности для авторизованного пользователя:

- Посмотреть запланированные поездки;
- Посмотреть информацию о запланированной поездке;
- Изменить информацию о себе;
- Изменить свои предпочтения;
- Добавить автомобиль;
- Написать сообщение пользователю;
- Прочитать сообщение от пользователя;
- Просмотреть профиль пользователя;
- Поставить оценку пользователю;
- Изменить пароль;
- Изменить способ оплаты;
- Посмотреть историю платежей;
- Написать в техподдержку;
- Получить ответ от техподдержки;
- Выйти из аккаунта;
- Удалить аккаунт.

Карта пользовательских историй для авторизованного пользователя представлена на рисунке ниже.



Рисунок 17 - Карта пользовательских историй для авторизованного пользователя

12 Контент и наполнение приложения

Все изображения, загружаемые в приложение, могут быть изменены по размеру, обрезаны или сжаты для оптимального отображения на его страницах.

13 Источники разработки

[1] ГОСТ 34.602-2020. ИНФОРМАЦИОННАЯ ТЕХНОЛОГИЯ. КОМПЛЕКС СТАНДАРТОВ НА АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ. ТЕХНИЧЕСКОЕ ЗАДАНИЕ НА СОЗДАНИЕ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ [В Интернете]. Доступно: <http://swrit.ru/doc/gost34/34.602-2020.pdf>

[2] ФЗ "О персональных данных" от 27.07.2006 N 152-ФЗ [В Интернете]. Доступно: https://www.consultant.ru/document/cons_doc_LAW_61801/

14 Порядок контроля и приемки работ

В течение текущего учебного семестра проверку приложения будет проводить команда преподавателей курса "Технологии программирования".

Окончательная версия проекта, включая готовое приложение и полную документацию, будет передана заказчику в сроки, указанные в настоящем Техническом задании. Заказчик несет ответственность за прием и проверку соответствия приложения требованиям, изложенным в Техническом задании и внутренних документах проекта.

Документация будет подготовлена и опубликована на GitHub, а также предоставлена заказчику в печатном и электронном виде.

15 Реквизиты и подписи сторон

Исполнители:

_____ Тарарыков А.А.

_____ Рогачев Е.А.

_____ Платон В.Е.

Заказчики:

_____ Тарасов В.С.

Приложение

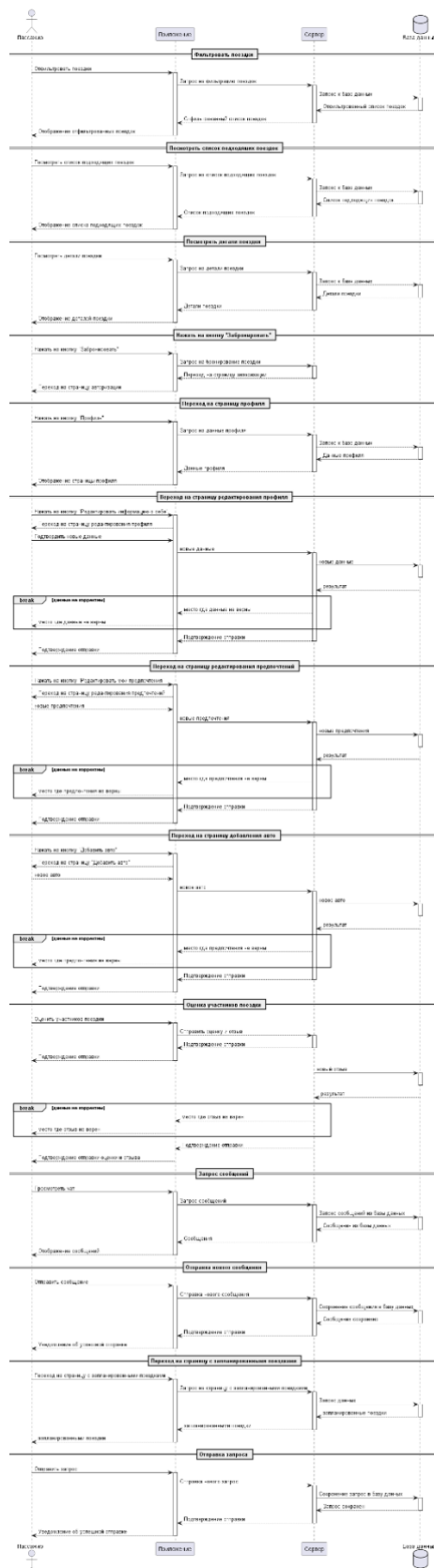


Рисунок 18 - Диаграмма воздействия для пассажира (часть 1 из 2)

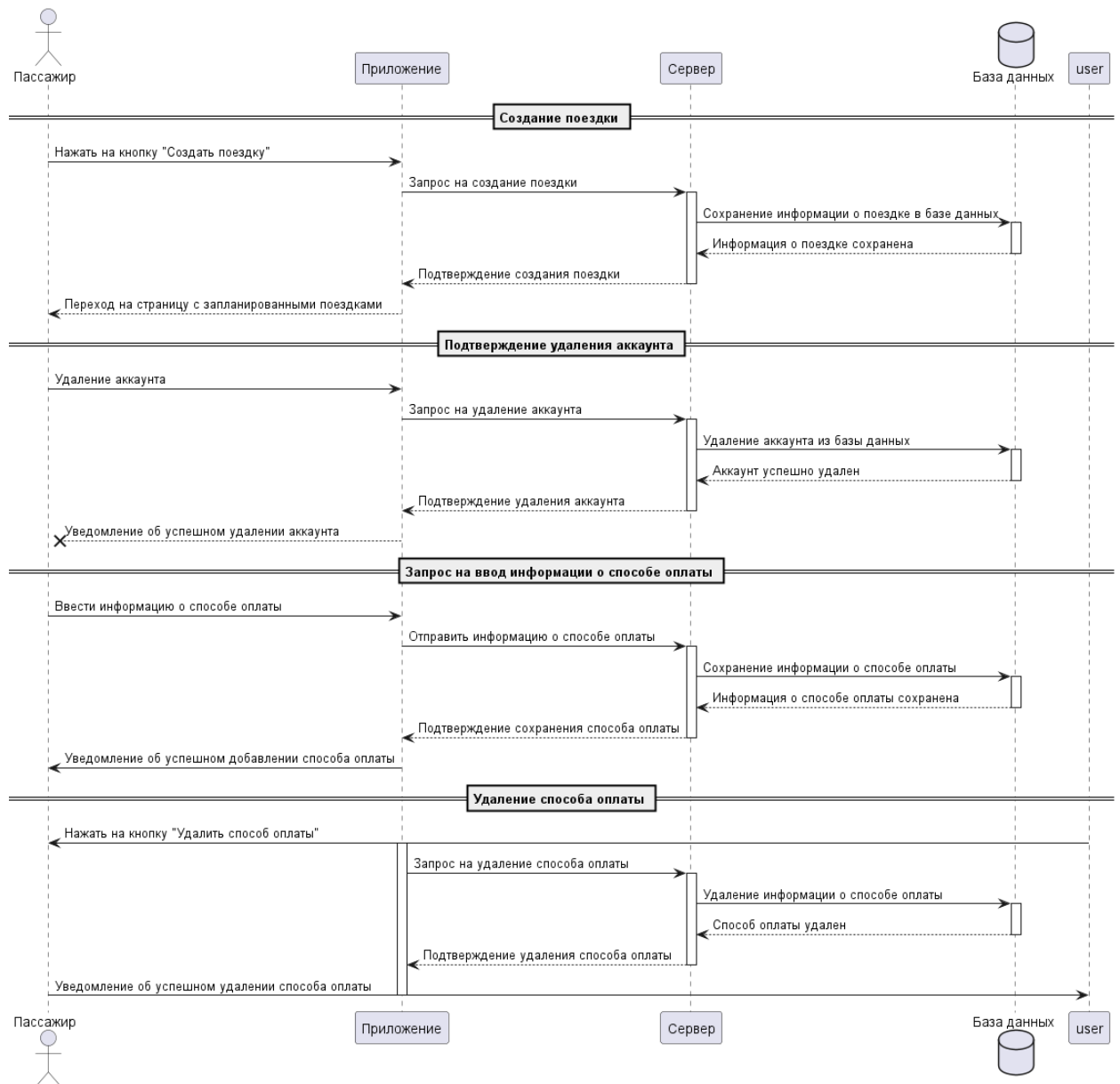


Рисунок 19 - Диаграмма воздействия для пассажира (часть 2 из 2)

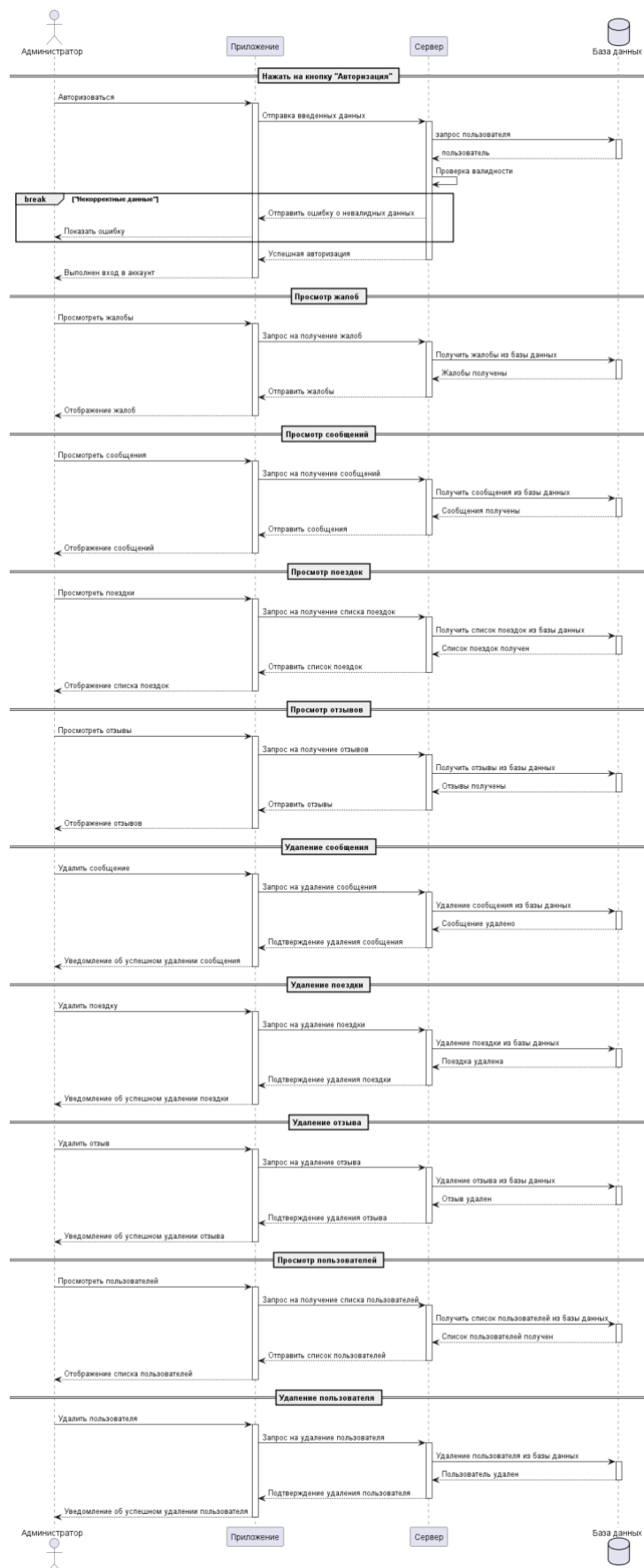


Рисунок 20 - Диаграмма воздействия для администратора

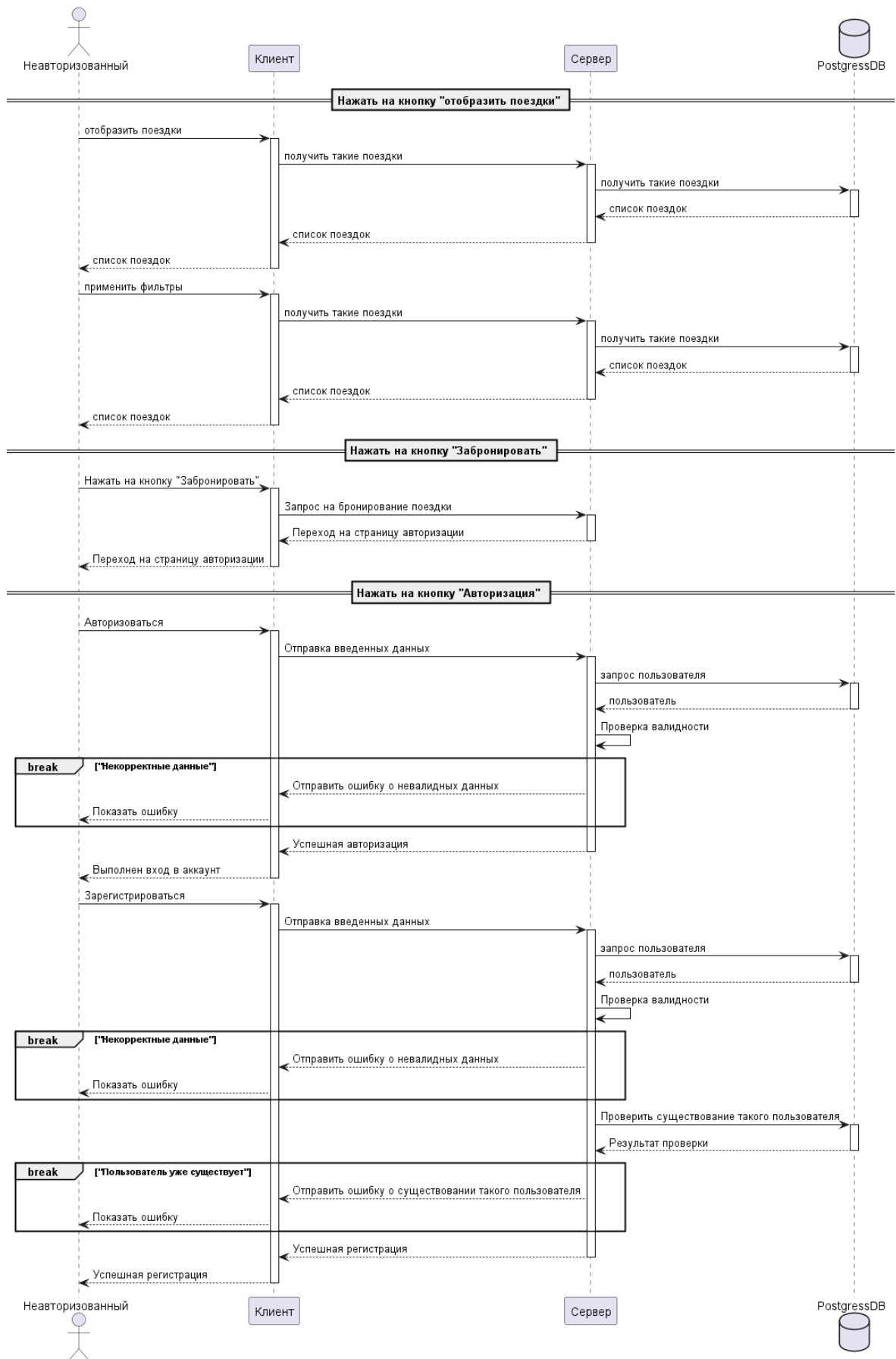


Рисунок 21 - Диаграмма воздействия для неавторизованного пользователя

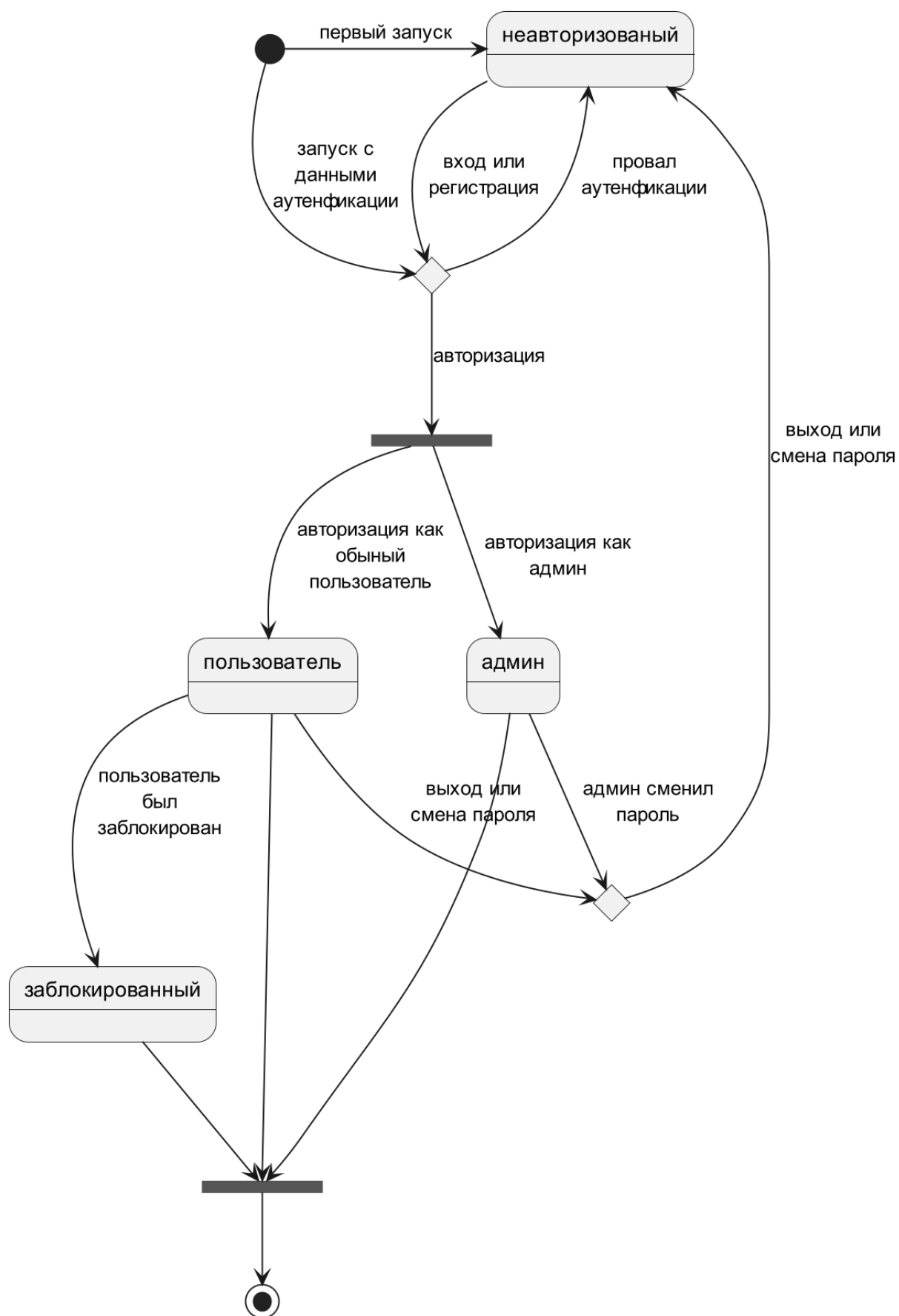


Рисунок 22 - Архитектура Spring приложения

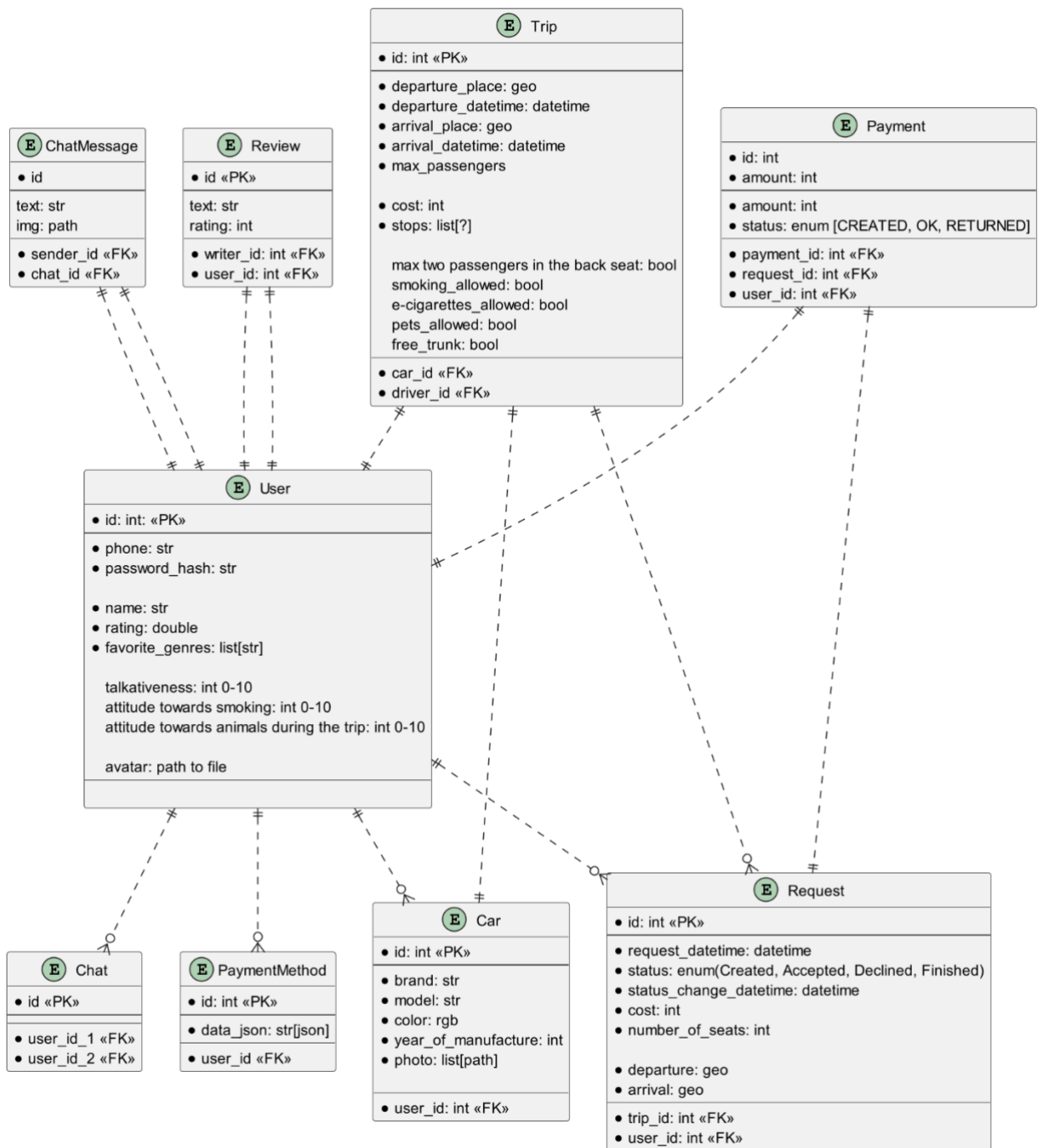


Рисунок 23 - ER диаграмма

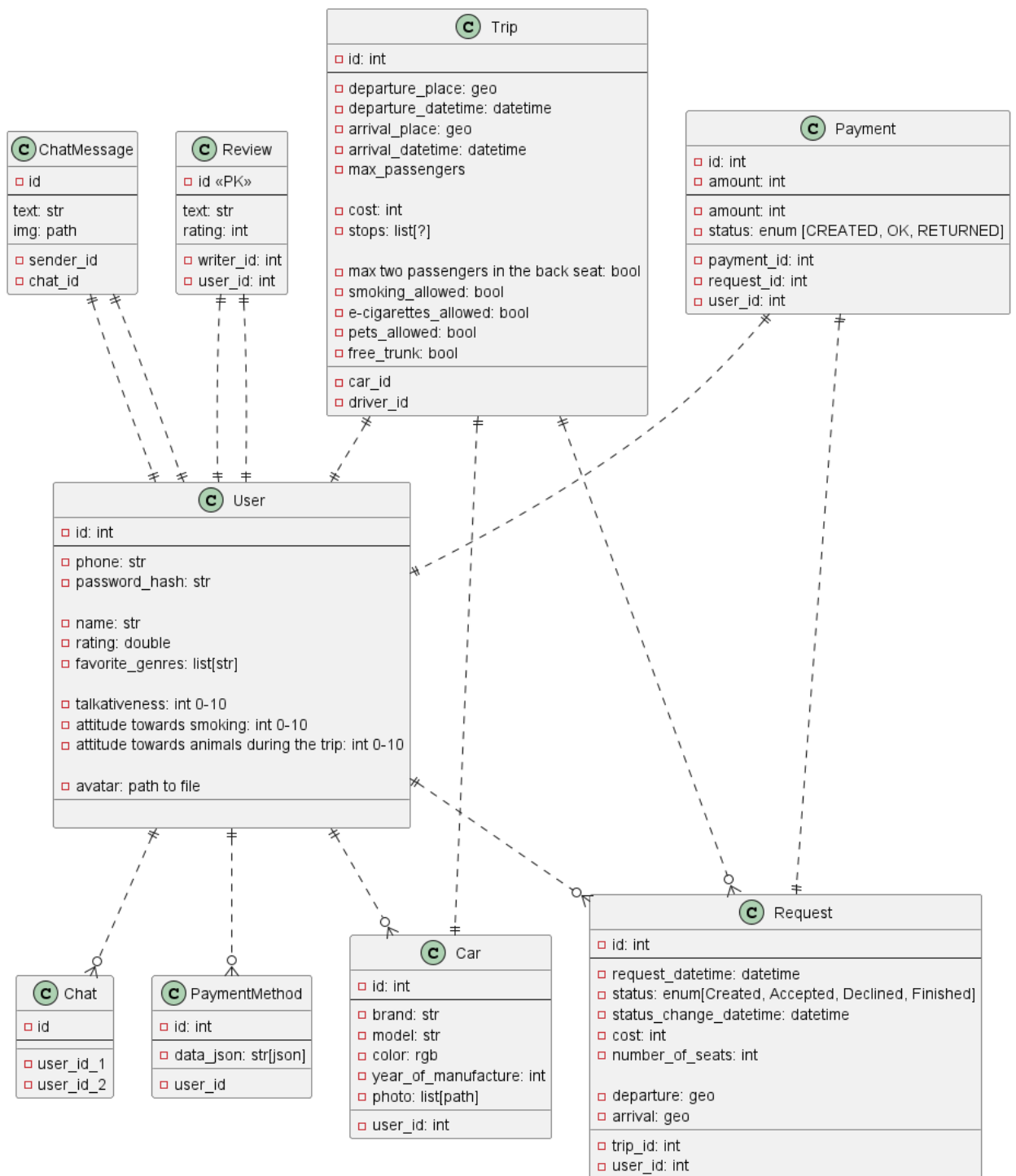


Рисунок 24 - Диаграмма классов