

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
“ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ”

Факультет компьютерных наук

Кафедра программирования и информационных технологий

Приложение для аренды автомобилей «FarSharing»

Курсовой проект

09.03.02 Информационные системы и технологии
Программная инженерия в информационных системах

Зав. кафедрой _____ д. ф.-м. н., профессор С.Д. Махортов

Руководитель _____ В.С. Тарасов

Обучающиеся:

_____ В.И. Валеев

_____ Е.Р. Ивашечкин

_____ П.С. Мишин

_____ К.С. Бондаренко

Воронеж 2022

Содержание

Содержание	2
Введение	3
1. Постановка задачи.....	4
1.1 Функциональные требования	4
1.2 Технические требования	5
1.3 Требования к интерфейсу	5
2. Анализ предметной области	6
2.1 Анализ существующих решений.....	6
2.1.1 Яндекс.Драйв.....	6
2.1.2 Делимобиль	6
2.2 Анализ задачи	7
2.2.1 Варианты использования приложения.....	7
2.2.2 Взаимодействие компонентов системы	10
2.2.3 Взаимодействие компонентов системы	12
2.2.4 Модель базы данных	15
2.2.5 Развертывание приложения	16
2.2.6 Входные-выходные данные (IDEF0).....	17
2.2.7 Воронки конверсии	18
3. Реализация	20
3.1 Frontend	20
3.2 Backend	20
4. Интерфейс	21
4.1 Интерфейс неавторизованного пользователя	21
4.2 Интерфейс авторизованного пользователя	25
4.3 Интерфейс пользователя, авторизованного под ролью «Администратор»	27

Введение

Прокат автомобилей востребован в современном обществе. Рынок предъявляет высокие требования к уровню доступности, качеству и удобству использования сервиса. Немаловажную роль играют уникальные свойства продукта, включая, визуализацию, маркетинговые особенности и т.д.

Одним из видов проката автомобилей является каршеринг. Каршеринг (от англ. car sharing) — это автоматизированный почасовой или поминутный прокат автомобиля. Услуга предназначена для коротких поездок по городу. Доступные для аренды автомобили стоят в разных частях города, об их точном расположении можно узнать из мобильных приложений для смартфонов. Через приложение частично осуществляется управление автомобилем – бронирование, запуск двигателя, открытие дверей.

В настоящее время на рынке представлено более 20 компаний, предоставляющих услуги каршеринга. Для предоставления данных услуг используется более 10 мобильных приложений, основными из которых являются Яндекс.Драйв, Делимобиль, BelkaCar, YouDrive, Bumerang, Rentmee, Cars7, URАмобиль и Vi-Bi.car. Основными параметрами, которыми должны обладать данные приложения являются: функциональность, удобство использования, информационная безопасность, производительность и надёжность, правовая оценка, информация о приложении.

При разработке проекта используется каскадная модель, которая включает в себя следующие этапы:

1. Определение требований.
2. Проектирование.
3. Конструирование (также «реализация» либо «кодирование»).
4. Воплощение.
5. Тестирование и отладка (также «верификация»).
6. Инсталляция.
7. Поддержка.

1. Постановка задачи

Целью курсового проекта является создание мобильного приложения, выполняющего функции по аренде автомобилей.

1.1 Функциональные требования

В системе предусмотрены три уровня доступа:

1. Неавторизованный пользователь;
2. Авторизованный пользователь;
3. Администратор.

Неавторизованный пользователь обладает следующими возможностями:

- Регистрация;
- Авторизация;

Авторизованный пользователь обладает следующими возможностями:

- Просмотр доступных автомобилей;
- Аренда автомобиля;
- Просмотр личных данных;
- Связь с технической поддержкой;
- Изменение личных данных;
- Выход из системы;
- Завершение аренды автомобиля;
- Удаление аккаунта.

Администратор обладает следующими возможностями:

- Подтвердить бронь;
- Отклонить бронь;
- Добавить автомобиль;
- Выход из системы;
- Удалить автомобиль;
- Подтвердить регистрацию;
- Отклонить регистрацию;
- Просмотр данных пользователя;

- Просмотр списка автомобилей;
- Заблокировать пользователя;
- Разблокировать пользователя;
- Просмотр списка заблокированных пользователей.
- Изменение личных данных

1.2 Технические требования

Приложение должно обеспечивать:

- Авторизацию его пользователей посредством email и пароля;
- Автоматическую корректировку данных о арендованных автомобилях.

1.3 Требования к интерфейсу

Интерфейс должен обладать:

- Раздельным интерфейсом для пользователя и администратора;
- Отображать списки данных, перечисленные в функциональных требованиях.

2. Анализ предметной области

2.1 Анализ существующих решений

2.1.1 Яндекс.Драйв

Яндекс.Драйв является самой развитой и известной сетью на данный момент.

Достоинства:

1. Наличие личного кабинета;
2. С помощью фильтров легко найти интересующий автомобиль;
3. Наличие раздела «Отзывы»;
4. Наличие приложения для мобильных устройств;
5. Встроенный навигатор;
6. Минимальная цена 5 рублей за минуту аренды;

Недостатки:

1. Довольно сложный интерфейс;
2. Время ожидания ответа службы поддержки;
3. Непрозрачное ценообразование;
4. Блокировка пользователя за незначительное превышение скорости.

2.1.2 Делимобиль

Делимобиль является одним из ключевых игроков на российском рынке.

Достоинства:

1. Наличие личного кабинета;
2. Дают время на осмотр автомобиля;
3. Огромный автопарк;
4. Наличие скидочной системы;
5. Наличие раздела «Отзывы»;
6. Наличие приложения для мобильных устройств;

Недостатки:

1. Приложения не позволяют фиксировать повреждения, грязь, комплектацию.
2. Неудобно организована система заправки.

2.2 Анализ задачи

2.2.1 Варианты использования приложения

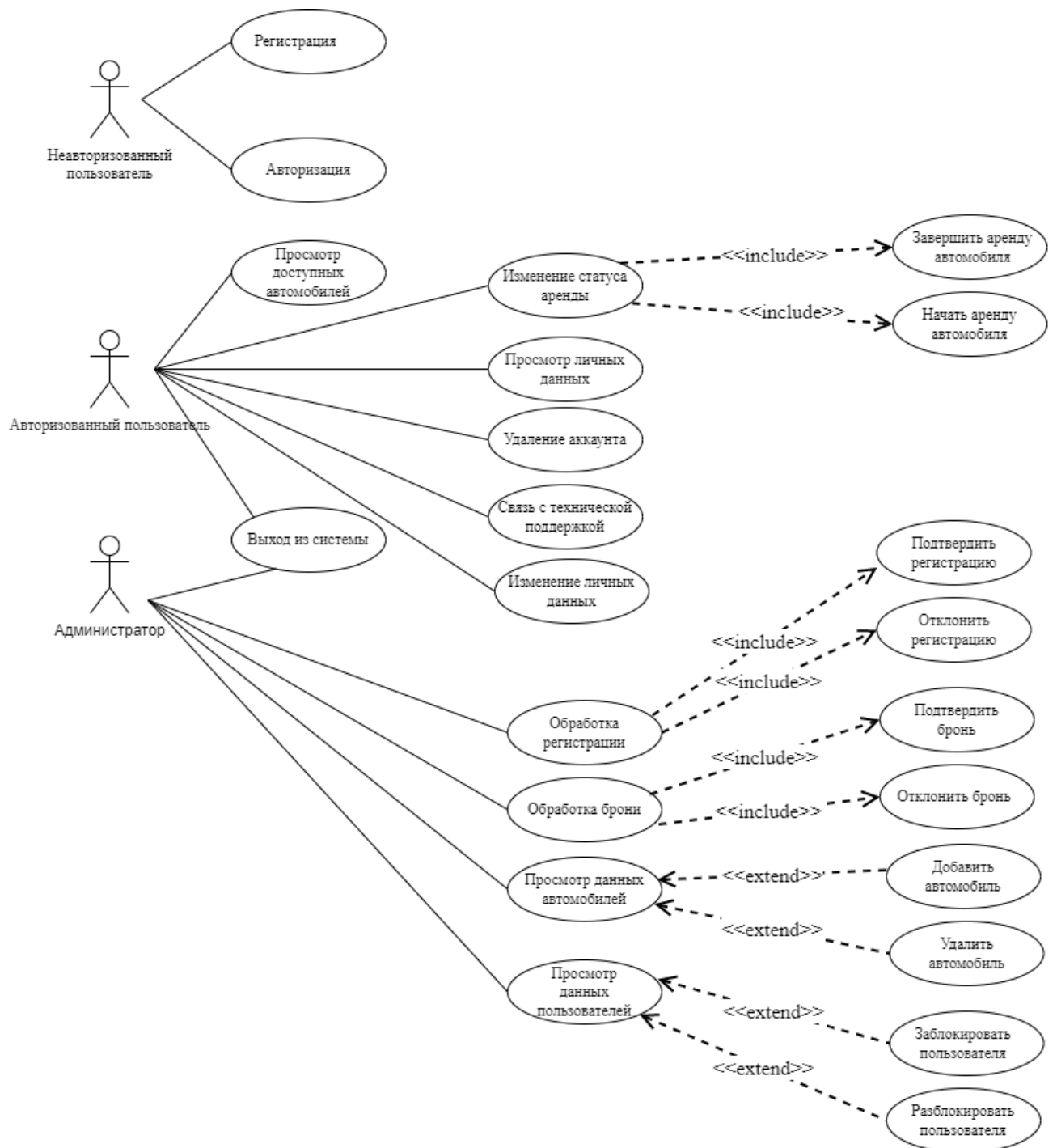


Рисунок 1. Диаграмма прецедентов.

Пользователь имеет возможность взаимодействовать с приложением без авторизации. При таком сценарии ему будут доступны следующие действия:

1. Регистрация;
2. Авторизация;

Также пользователь имеет возможность авторизоваться под ролью «Администратор» или ролью «Авторизованный пользователь». Если пользователь авторизуется под ролью «Администратор», то ему будут доступны следующие действия:

1. Обработка брони:
 - Подтвердить бронь,
 - Отклонить бронь.
2. Просмотр данных автомобилей:
 - Добавить автомобиль,
 - Удалить автомобиль.
3. Просмотр данных пользователей:
 - Заблокировать пользователя
 - Разблокировать пользователя
4. Обработка регистрации:
 - Подтвердить регистрацию,
 - Отклонить регистрацию.
5. Выход из системы.

Если же пользователь авторизуется под ролью «Авторизованный пользователь», то ему будут доступны следующие действия:

1. Просмотр личных данных;
2. Удаление аккаунта;
3. Связь с технической поддержкой;
4. Изменение личных данных;
5. Выход из системы;
6. Просмотр доступных автомобилей;
7. Изменение статуса аренды:
 - Завершить аренду автомобиля,
 - Начать аренду автомобиля.

2.2.2 Взаимодействие компонентов системы

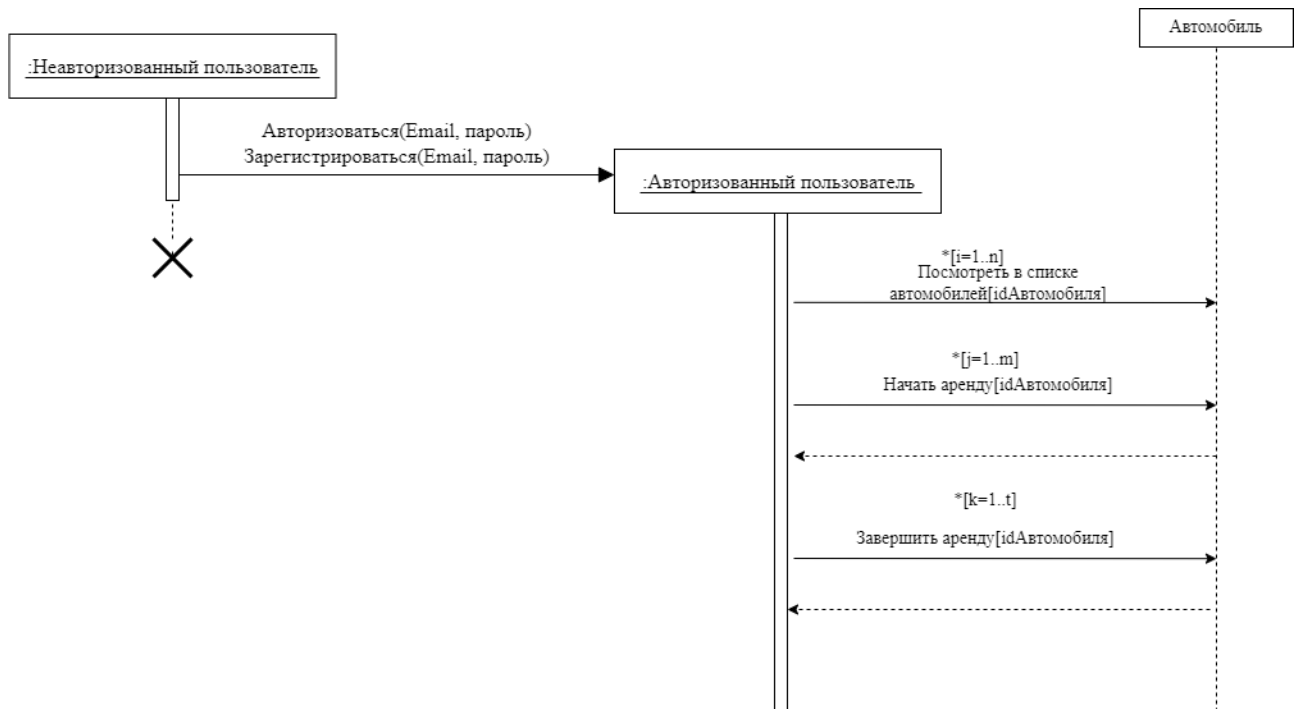


Рисунок 2. Диаграмма последовательностей для авторизованного пользователя.

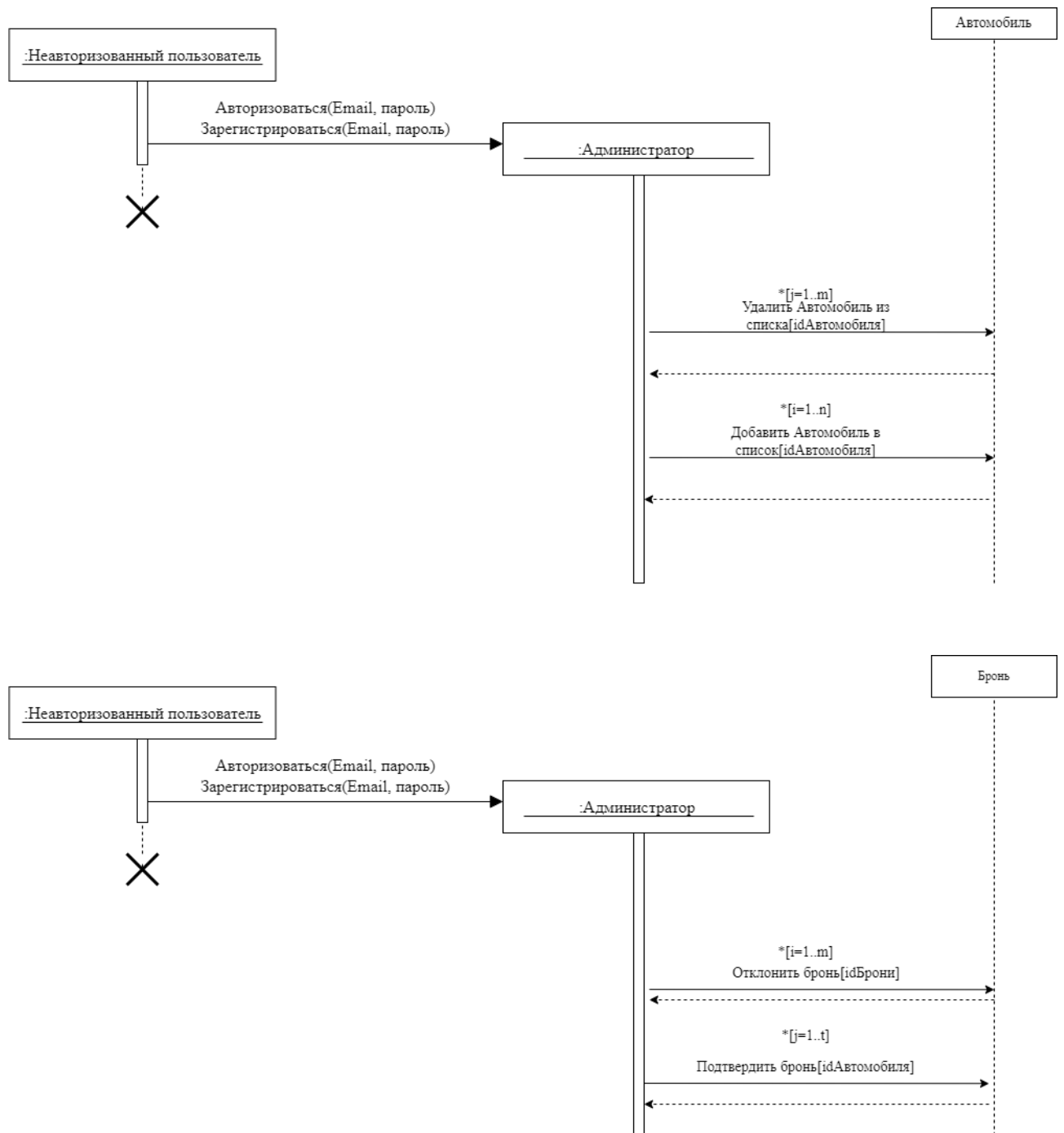


Рисунок 3. Диаграмма последовательностей для Авторизованного пользователя под ролью “Администратор”.

2.2.3 Взаимодействие компонентов системы

Возможные варианты взаимодействия компонентов системы, включающей неавторизованного пользователя, авторизованного пользователя и администратора, отражено на рисунках 4 – 6.

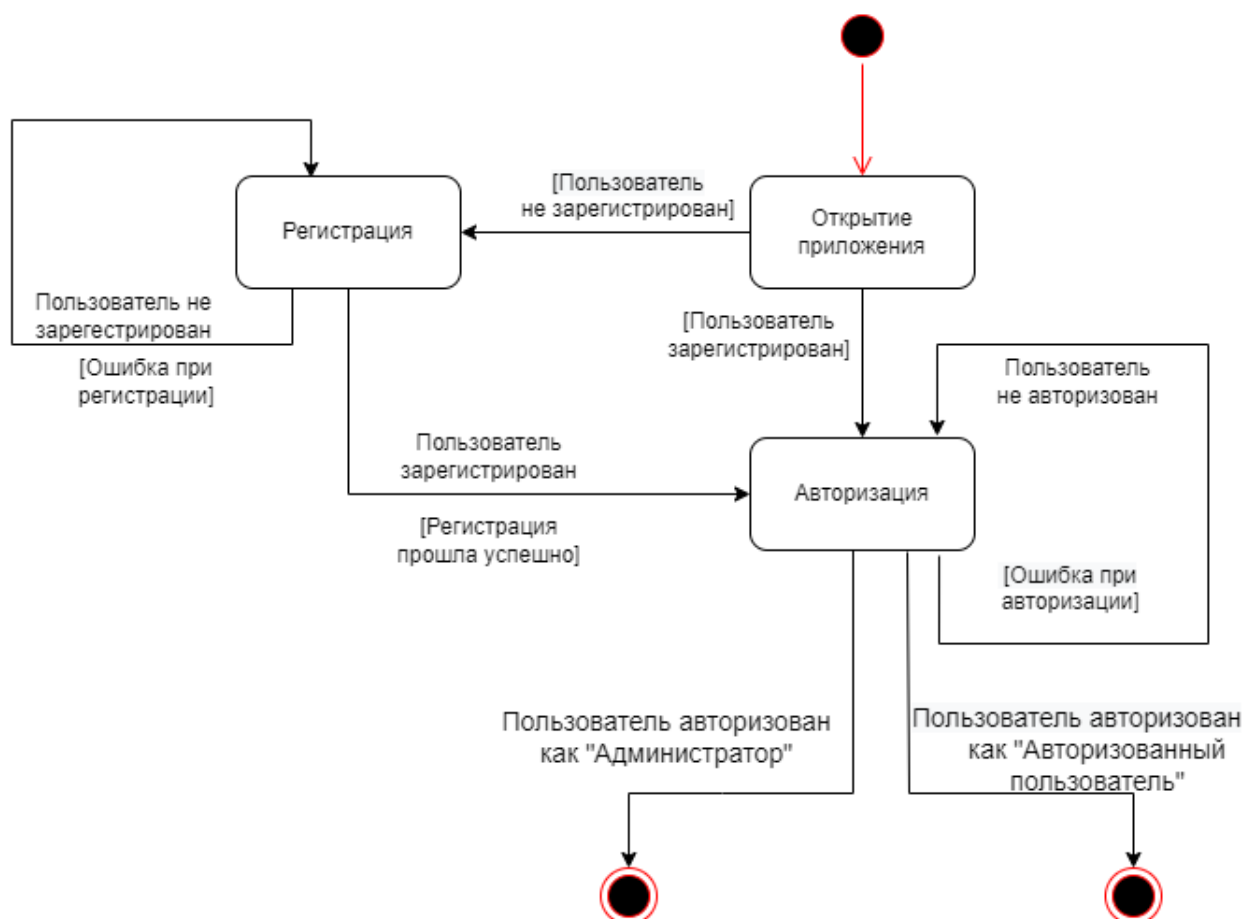


Рисунок 4. Диаграмма состояния неавторизованного пользователя.

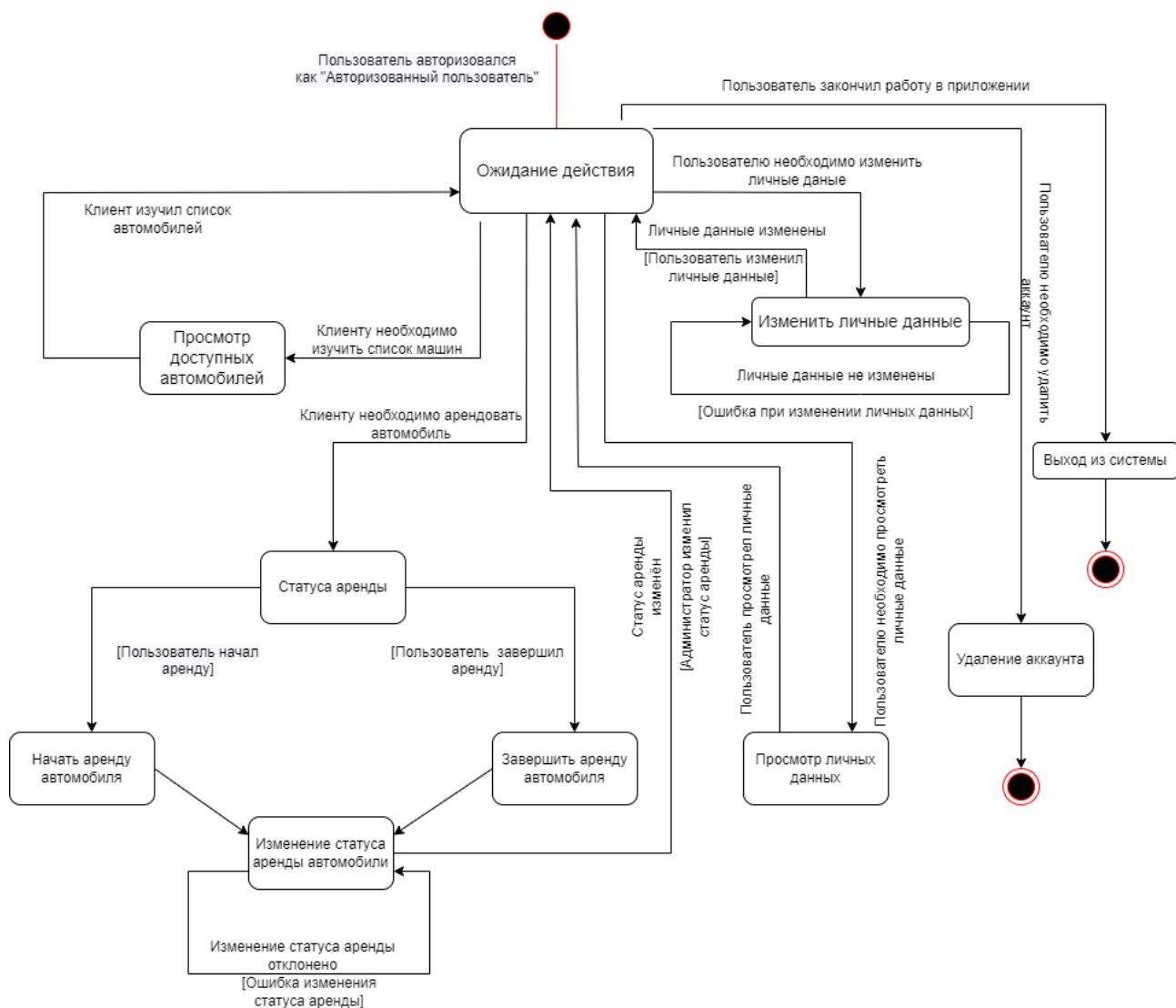


Рисунок 5. Диаграмма состояния авторизованного пользователя.

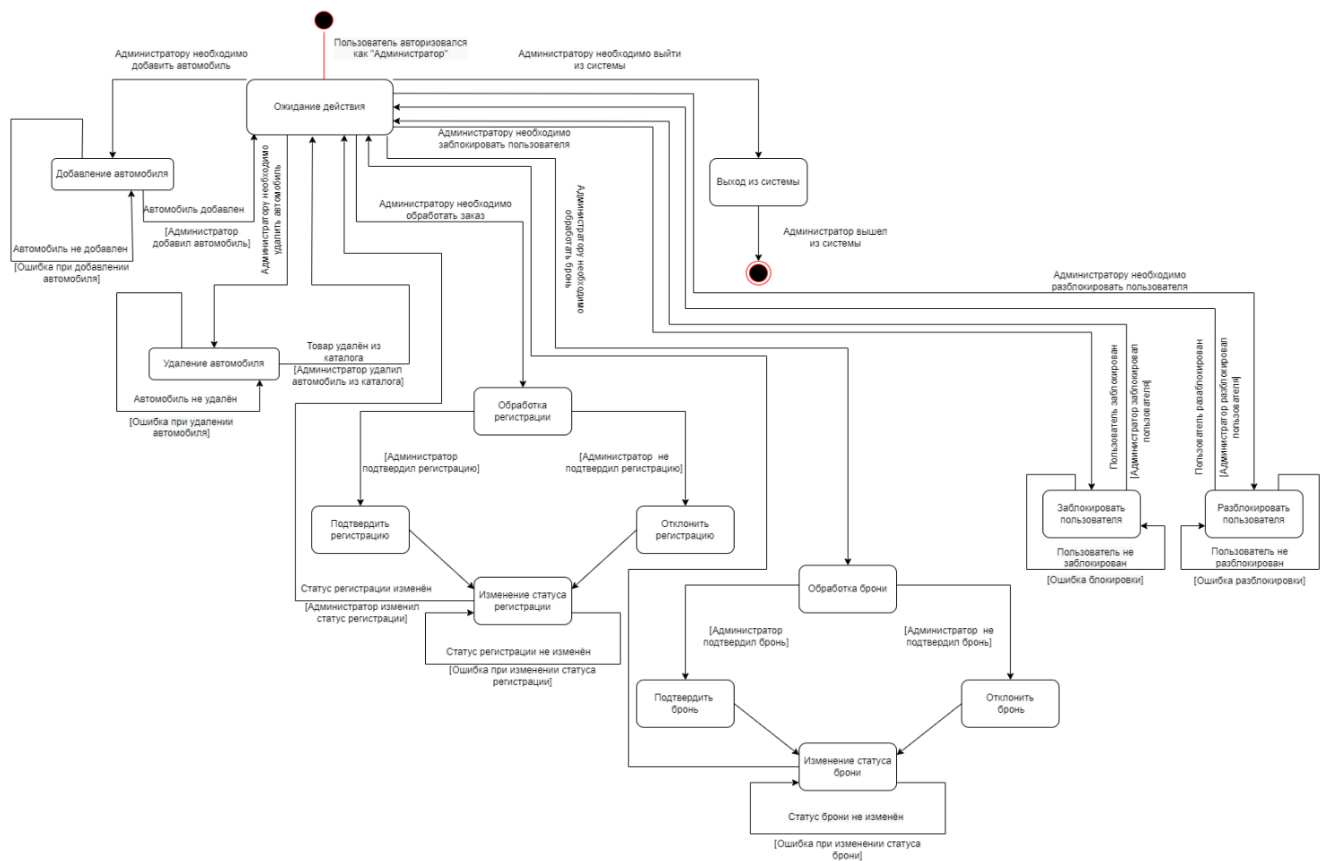


Рисунок 6. Диаграмма состояния администратора.

2.2.4 Модель базы данных

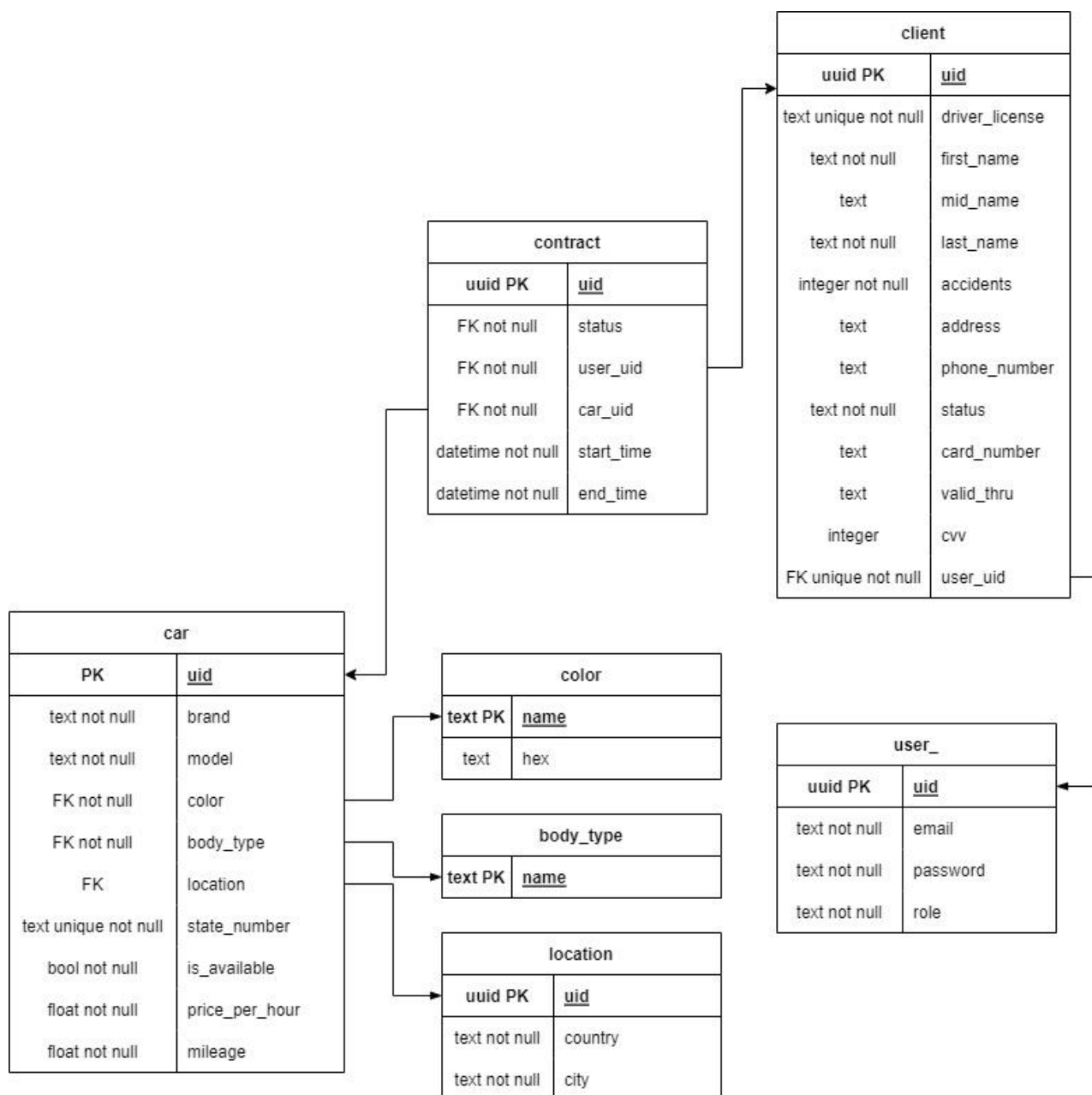


Рисунок 7. Схема базы данных.

На Рисунке 7 представлена схема базы данных для разрабатываемого приложения.

2.2.5 Развертывание приложения

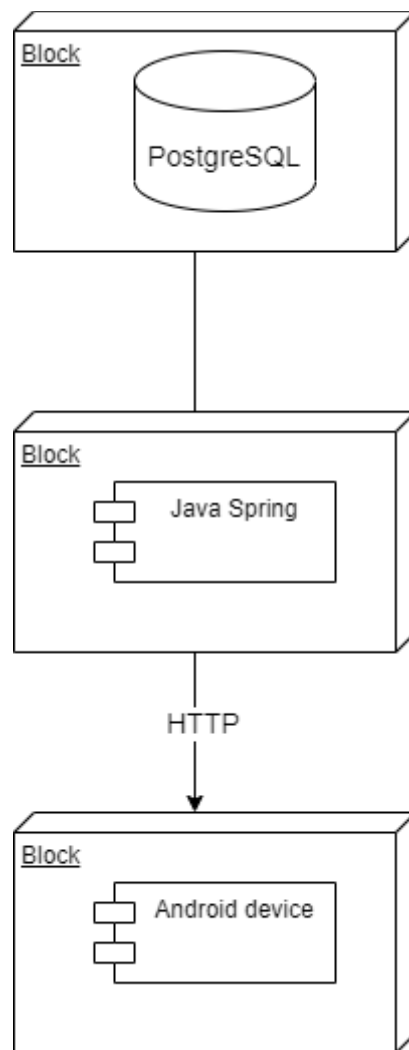


Рисунок 8. Диаграмма развертывания.

На Рисунке 8 представлена диаграмма развертывания, чтобы определить какие аппаратные компоненты (узлы) существуют, какие программные компоненты (артефакты) работают на каждом узле, и как различные части этого комплекса соединяются друг с другом.

2.2.6 Входные-выходные данные (IDEF0)

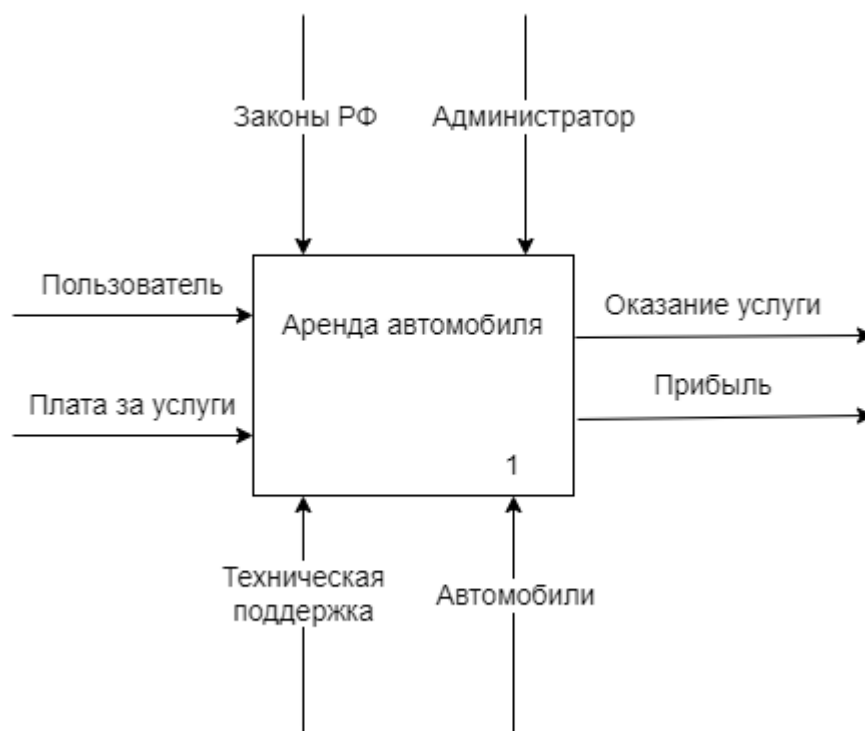


Рисунок 9. Контекстная диаграмма.

2.2.7 Воронки конверсии

Составная цель: Аренда автомобиля

Посетить главную страницу – Перейти на выбранный автомобиль –
Оплатить аренду.

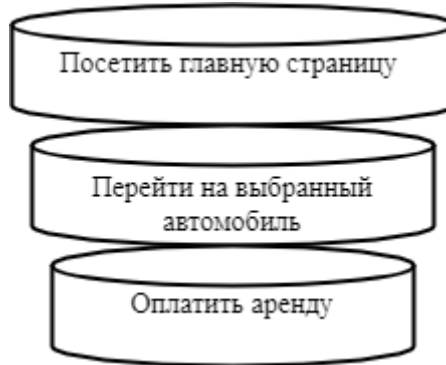


Рисунок 10. Воронка конверсии аренды автомобиля.

Составная цель: Удаление аккаунта

Посетить главную страницу – Перейти в меню - Перейти в личный кабинет
– Удалить аккаунт.



Рисунок 11. Воронка конверсии удаления аккаунта.

Составная цель: Завершить аренду автомобиля

Посетить главную страницу - Перейти в меню – Перейти в личный кабинет
– Завершить аренду.



Рисунок 12. Воронка конверсии завершения аренды автомобиля.

Составная цель: Обратиться в поддержку

Посетить главную страницу – Перейти в меню – Перейти в личный кабинет
– Написать сообщение в поддержку.

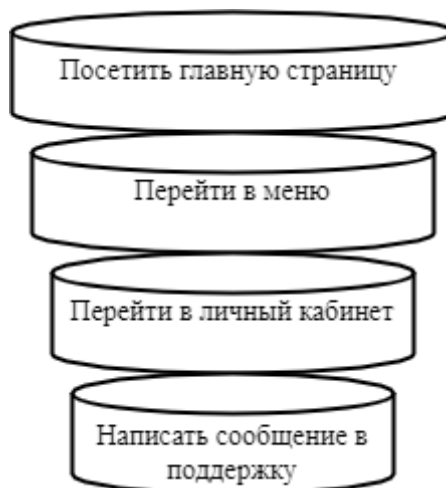


Рисунок 13. Воронка конверсии обращения в поддержку.

3. Реализация

3.1 Frontend

При создании клиентской части использовались набор инструментов Android SDK и REST-клиент Retrofit.

3.2 Backend

При создании серверной части использовался фреймворк Java Spring Boot. Для работы программы была выбрана Hibernate framework.

Документирование бэкенда приложения ведётся с помощью ПО с открытым исходным кодом Swagger. Этот инструмент позволяет разработчикам вести и предоставлять документацию REST веб-сервисов с возможностью тестирования запросов к методам API.

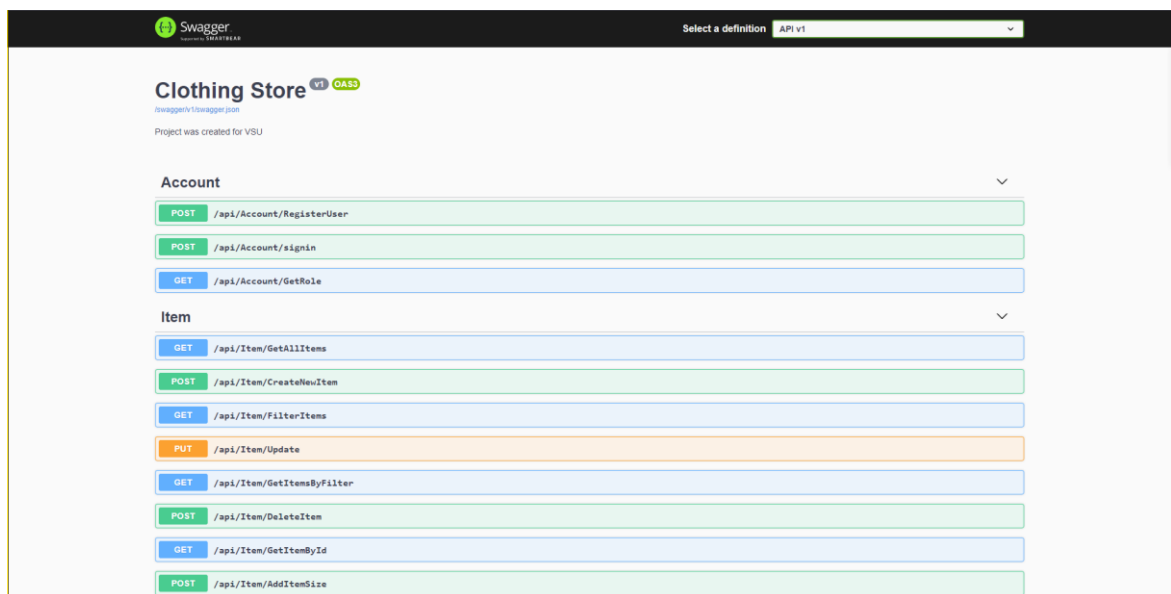


Рисунок 14. Пример документирования с использованием Swagger.

4. Интерфейс

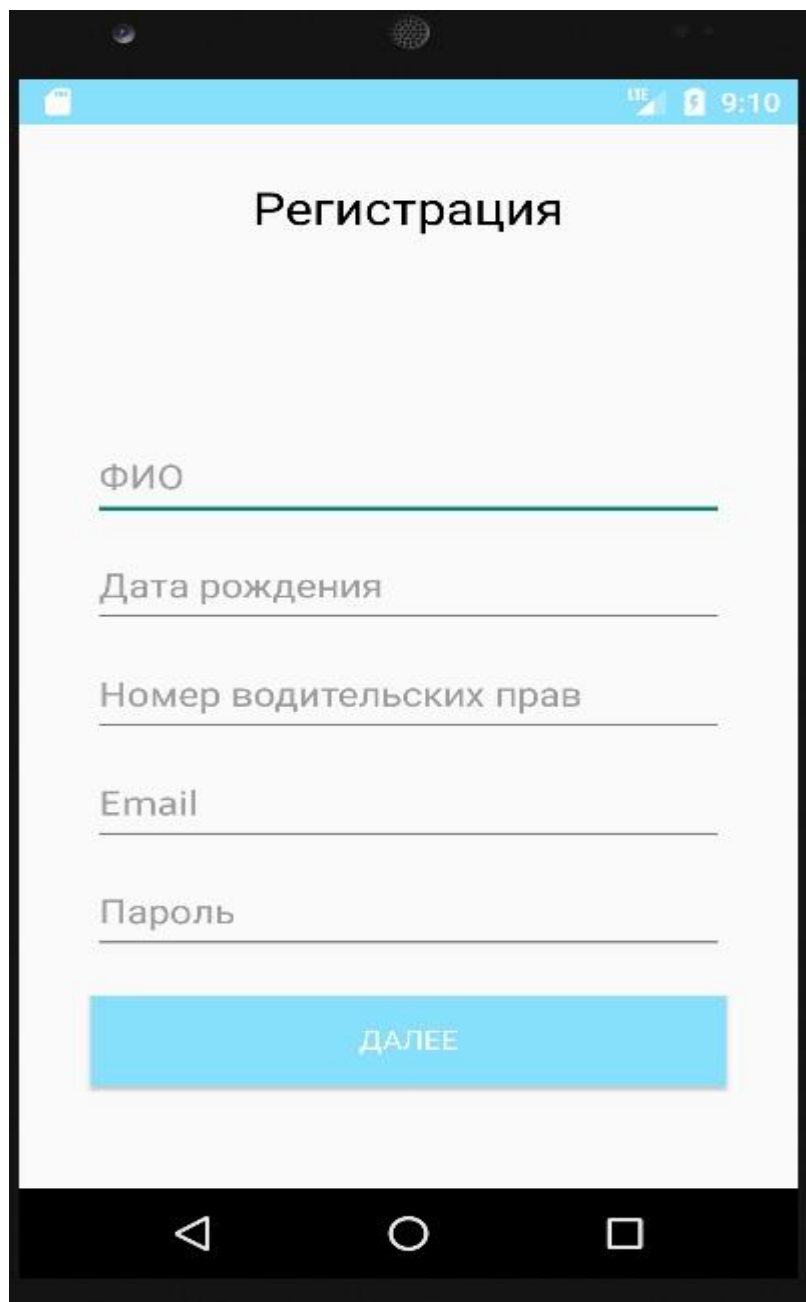
4.1 Интерфейс неавторизованного пользователя

При открытии приложения пользователь попадает на главную страницу приложения (Рисунок 15). На ней пользователь имеет возможность перейти на страницу авторизации или регистрации.



Рисунок 15. Стартовый экран.

Если по какой-то причине пользователь еще не зарегистрирован, то он имеет возможность нажать на кнопку «Регистрация» для перехода на страницу регистрации (Рисунок 16) и дальнейшей регистрации. Для того, чтобы зарегистрироваться пользователю необходимо заполнить поля «ФИО», «Дата рождения», «Номер водительских прав», «Email», «Пароль» и нажать кнопку «Далее».



The image shows a mobile application interface for registration. At the top, there is a status bar with a light blue background showing signal strength, battery level, and the time 9:10. Below this, the title 'Регистрация' is centered in a large, bold, black font. The main area contains five input fields, each with a light gray label above it: 'ФИО', 'Дата рождения', 'Номер водительских прав', 'Email', and 'Пароль'. Each field has a thin gray border. At the bottom of the form is a wide, rectangular blue button with the word 'ДАЛЕЕ' in white, uppercase letters. The entire screen is framed by a black border, and at the very bottom, there is a black navigation bar with three white icons: a triangle, a circle, and a square.

Рисунок 16. Экран регистрации.

При нажатии кнопки «Далее» пользователь перейдёт на экран Подтверждения почты (Рисунок 17).



Рисунок 17. Экран подтверждения почты.

После подтверждения электронной почты пользователь перейдёт на экран авторизации (Рисунок 18).

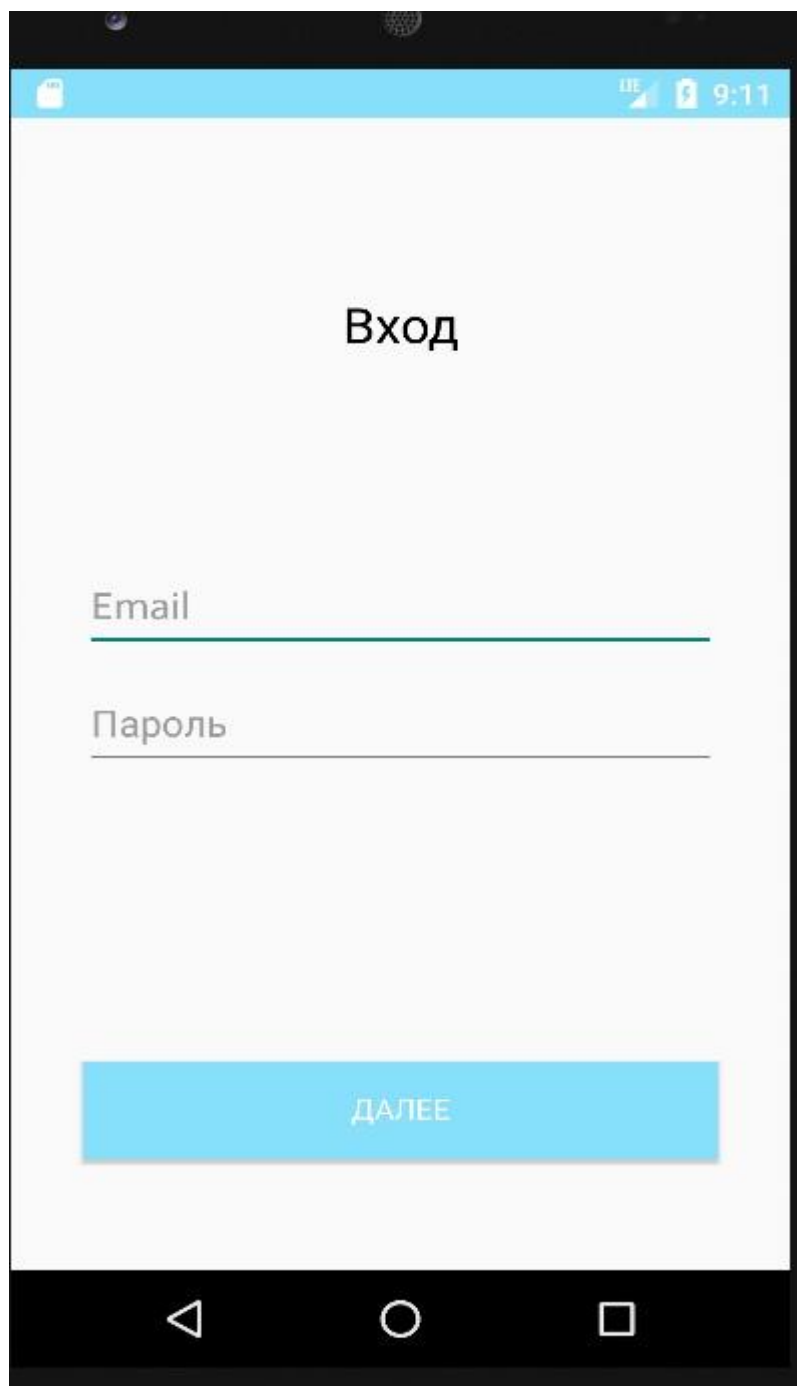


Рисунок 18. Экран авторизации.

4.2 Интерфейс авторизованного пользователя

Если пользователь авторизовался, то он попадает на главную страницу (Рисунок 19).

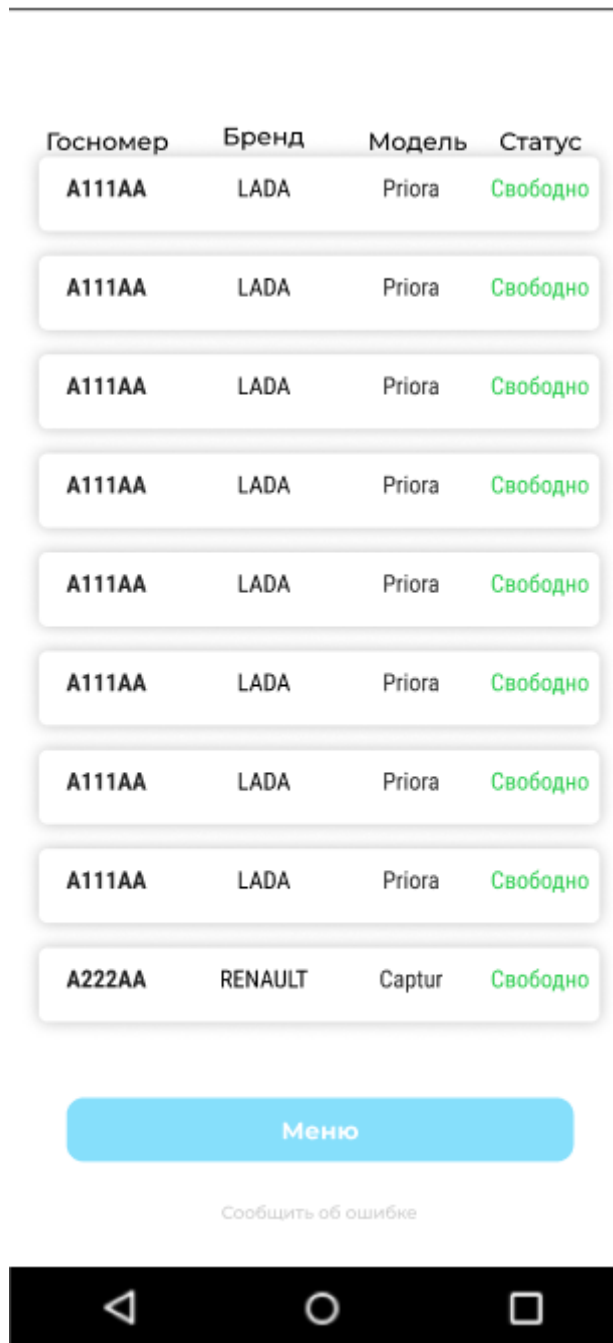


Рисунок 19. Главный экран пользователя.

На этой странице авторизованный пользователь может перейти на экран с его личными данными, нажав кнопку «Меню». После нажатия на эту кнопку пользователь переходит на страницу личного кабинета (Рисунок 19). На этом экране клиент может просмотреть свои личные данные и при необходимости их отредактировать, нажав на кнопку «Редактировать».

Также на главной странице авторизованный пользователь может, ознакомиться со списком доступных автомобилей. Выбрав автомобиль, мы перейдём на следующий экран, на котором сможем ознакомиться с более точными параметрами автомобиля и арендовать его (Рисунок 20).

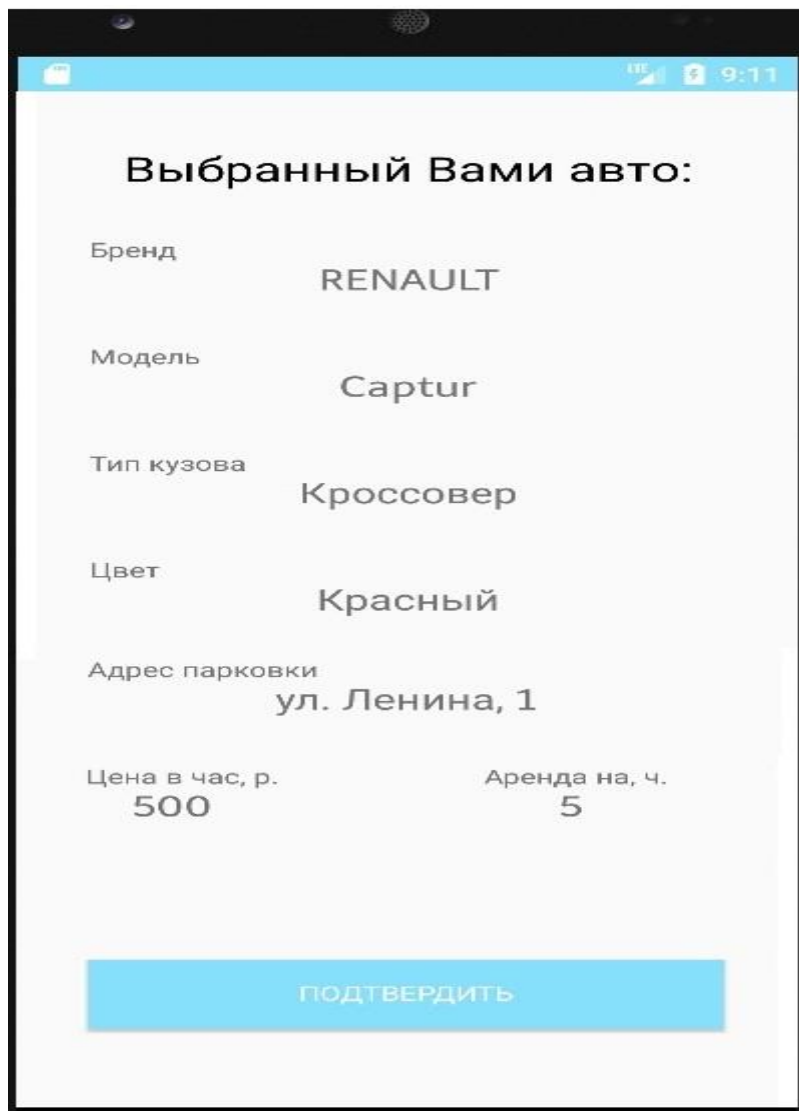


Рисунок 20. Экран – характеристик выбранного автомобиля.

4.3 Интерфейс пользователя, авторизованного под ролью «Администратор»

Если пользователь авторизовался под ролью «Администратор», то он попадает на главную страницу (Рисунок 21).



Рисунок 21. Главный экран администратора.

Администратор имеет возможность добавлять автомобили. Для этого ему необходимо перейти на страницу добавления автомобиля, нажав кнопку «Добавить автомобиль» в шапке страницы.