УДК 004.9

Кузнецов Владислав Сергеевич, магистрант
Савинцев Артем Сергеевич, магистрант
Научный руководитель — Колокольникова Алла Ивановна, доцент, к.т.н.
(КузГТУ, г. Кемерово)
Vladislav S. Kuznetsov, master's student
Artem S. Savintsev, master's student
Scientific supervisor — Alla I. Kolokolnikova, associate professor,
candidate of engineering sciences
(KuzSTU, Kemerovo)

РЕАЛИЗАЦИЯ ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ ОРГАНИЗАЦИИ СОВМЕСТНЫХ ПОЕЗДОК В ГОРОДСКОЙ СРЕДЕ

IMPLEMENTATION OF A WEB APPLICATION FOR SHARED RIDES ORGANIZING IN URBAN AREAS

Аннотация. Описано веб-приложение для организации совместных поездок по городу: как создаются и ищутся маршруты, как водитель подтверждает бронирования, как участники общаются между собой. Отдельно говорится о планах по мобильным версиям.

Abstract. The paper describes a web application for arranging shared rides within a city: how routes are created and searched, how a driver confirms bookings, and how participants communicate. Plans for mobile versions are also outlined.

Городские дороги с каждым годом загружены сильнее. При этом многие водители ездят в одиночку, хотя в салоне есть свободные места, а кому-то из горожан надо в ту же сторону. Если сводить такие поездки вместе, можно задействовать машины, которые и так уже едут [1].

Решение известное — карпулинг: водитель берёт попутчиков, которым с ним по пути [2]. Но в городе всё устроено мельче. Расстояния короткие, садиться люди хотят в разных местах, время выезда приходится уточнять. Текстового объявления тут не хватает; нужны карта, поиск по условиям, запрос на место и чат.

Задача работы — собрать действия водителя и пассажира в одном приложении, от публикации маршрута до бронирования и переписки. Ниже описана веб-версия и то, как с ней работают. Над проектом мы трудимся поэтапно: веб-версия уже есть, мобильные пока впереди.

Создание и публикация поездки

Начинается всё с регистрации. В личном кабинете пользователь заполняет контакты и настройки, смотрит историю своих поездок. Постоян-

ной роли у него нет: сегодня он предлагает поездку как водитель, завтра ищет место как пассажир. Первый шаг водителя — опубликовать маршрут (рисунок 1).

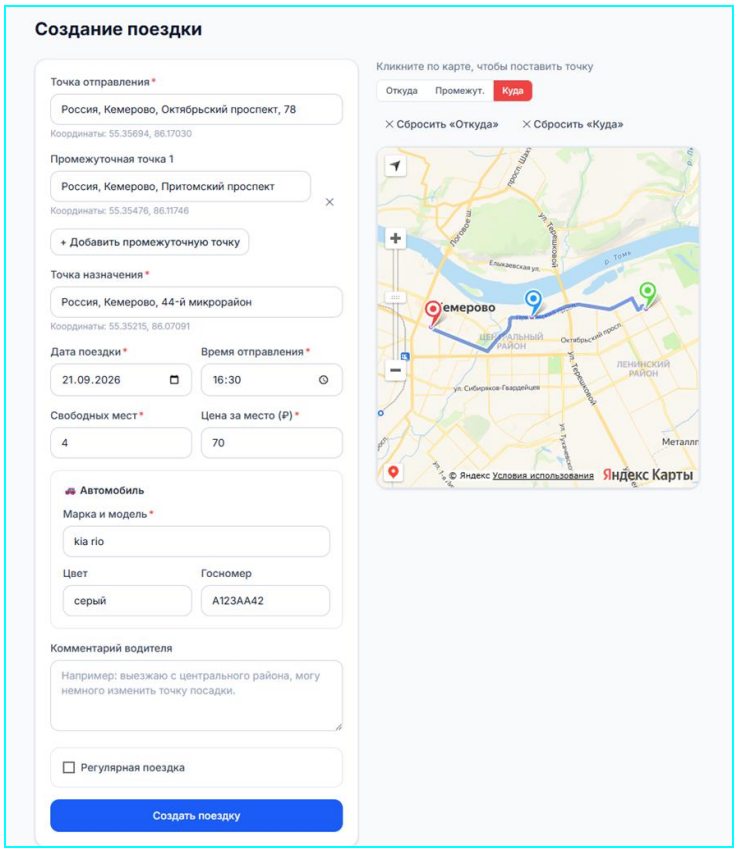

Рисунок 1 – Создание поездки с промежуточной точкой и отображением маршрута

Водитель вводит адреса отправления и назначения или ставит точки на карте. Если по пути кого-то нужно подобрать, добавляется промежуточная остановка. Маршрут рисуется на карте сразу, так что ошибку в направлении видно до публикации. Пассажиру карта понятнее перечня улиц: он смотрит, далеко ли остановка от его пути.

Дальше водитель задаёт дату, время, число мест и цену за место и описывает автомобиль: марка, модель, цвет, номер. В комментарии можно предупредить, скажем, о багаже. Поездки, которые повторяются, помечаются как регулярные.

Сохранённая поездка сразу видна другим. Пассажира интересует не только конечная точка: ему важно, можно ли сесть где-то по дороге. Промежуточные остановки отвечают на этот вопрос, и писать водителю с уточнениями приходится реже.

Поиск и выбор подходящего маршрута

В поиске пассажир выбирает направление, дату и время. Можно отфильтровать по числу мест, цене и рейтингу водителя. Результаты выво-

дятся списком: время отправления, цена, водитель, автомобиль. Рядом на карте показаны маршруты всех найденных поездок.

Цена и время — не всё. Пассажиру важно, как далеко проходит маршрут от места, где ему удобно сесть [3], и особенно когда предложения почти одинаковы. Сравнивать их удобно на самом экране поиска (рисунок 2).

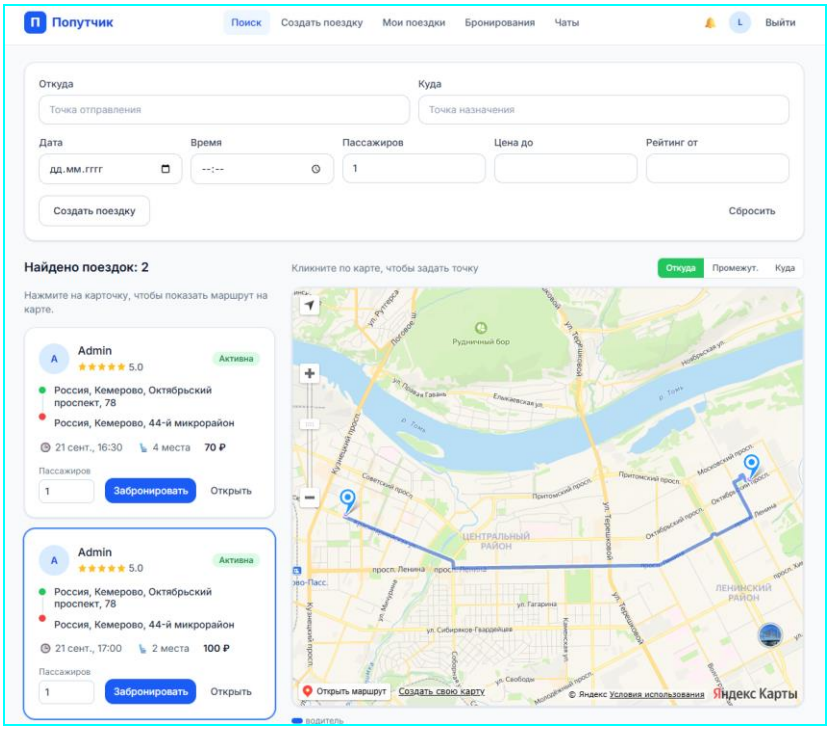

Рисунок 2 – Результаты поиска поездок и маршруты на карте

В городе до точки посадки бывает добираться дольше, чем потом ехать, и тогда подходящее по времени предложение оказывается бесполезным. Поэтому в списке учитываются два параметра: время отправления и расстояние от посадки до указанного пассажиром места. Карта рядом со списком показывает, где маршрут проходит на самом деле.

Бронирование мест и управление поездкой

Открыв поездку, пассажир видит маршрут, водителя, автомобиль и число свободных мест. Место за ним не закрепляется: сначала он отправляет заявку, а решение принимает водитель на странице поездки (рисунок 3).

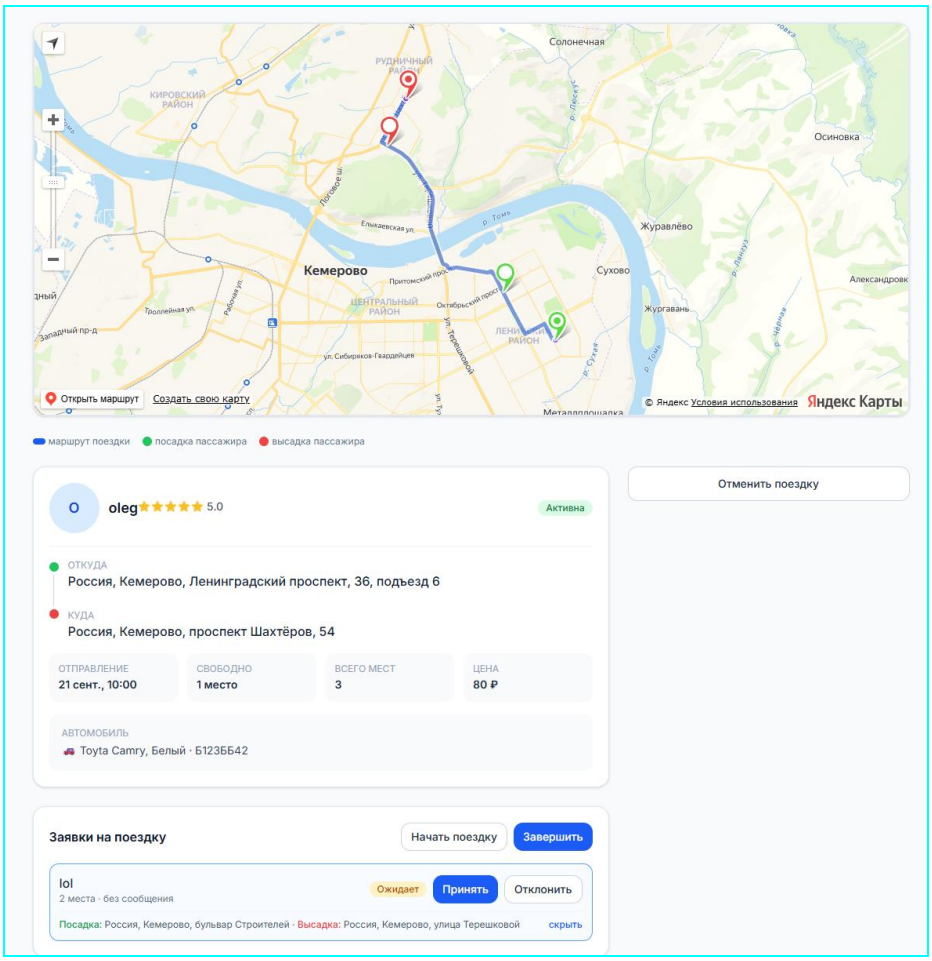

Рисунок 3 – Страница поездки и подтверждение заявки пассажира водителем

Попутчика мало найти, с ним нужно договориться [4]. Пассажир выбирает число мест и, если нужно, уточняет, где сядет и где выйдет. Заявка относится к одной поездке и сперва «ждёт ответа»; это ещё не бронь. Водитель видит заявку и принимает или отклоняет её, а пассажир узнаёт результат в своём интерфейсе.

Принятая заявка уменьшает число свободных мест на количество пассажиров. Если её отменяют до выезда, места освобождаются. Чтобы водитель не принял больше, чем у него есть, остаток проверяется ещё раз прямо при обработке заявки. Статус и число мест меняются одной операцией в PostgreSQL [5].

На странице поездки у водителя есть список заявок. У каждой виден статус и две кнопки: «Принять» и «Отклонить». Запрос на участие не путается с уже согласованной поездкой.

Пассажир на той же странице видит маршрут, водителя, условия и статус своей заявки, поэтому переспрашивать ничего не нужно. Договорившись, водитель отмечает начало и конец поездки либо отменяет её, если планы поменялись.

Три шага — поиск, заявка, подтверждение — разделены не случайно, особенно когда мест мало. Пока один пассажир ждёт ответа, поездка остаётся в выдаче, и другой тоже её увидит. Поэтому судьба места решается в момент подтверждения, а не при показе результатов поиска.

Последовательность взаимодействия водителя и пассажира

Возьмём типичный случай: поездка по городу, мест несколько. Водитель публикует маршрут, добавляет остановку по пути и время выезда. Пассажир открывает карту, решает, что остановка ему подходит, и просит нужное число мест. Пока водитель молчит, заявка висит в ожидании, и считать место своим пассажир не может.

Водитель смотрит, сколько мест просят и где человек хочет сесть и выйти. Устраивает — подтверждает; статус меняется, остаток мест пересчитывается, и следующие пользователи видят уже новое число. Не устраивает — отклоняет, и пассажир ищет другую поездку. Число мест, получается, зависит от ответа водителя, а не от самого запроса.

Общение участников, отзывы и личный кабинет

Когда заявка принята, открывается чат: в нём договариваются о месте встречи и времени подъезда (рисунок 4). На рисунке пассажир спрашивает, можно ли сесть у остановки на Ленинградском проспекте, а водитель отвечает, что подъедет к 16:25. После поездки участники могут оценить друг друга.

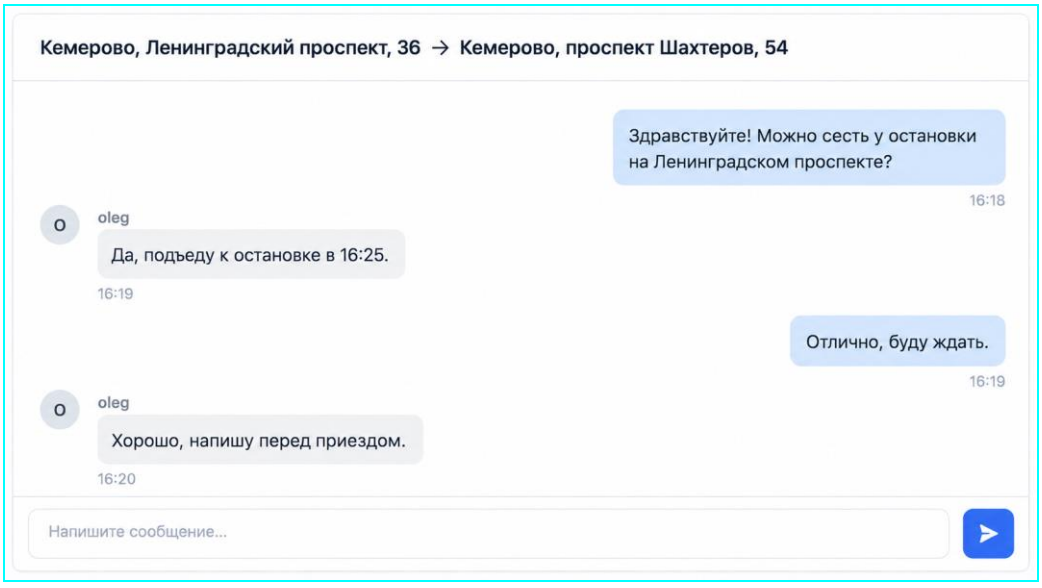

Рисунок 4 – Переписка водителя и пассажира перед поездкой

Переписка принадлежит конкретной поездке. Из-за этого не приходится уходить в сторонние мессенджеры или оставлять номер телефона в общем списке объявлений.

Чаты разных поездок не смешиваются. Новые сообщения приходят через WebSocket, без обновления страницы, а о событиях по поездке со-

общают уведомления. Для коротких городских маршрутов это существенно, ведь место посадки часто уточняют за считанные минуты до выезда.

После поездки участники ставят друг другу оценку от одной до пяти звёзд и при желании пишут комментарий. Оценки видны в профиле. Там же можно указать предпочтения: музыка, разговоры в дороге, перевозка животных. Историю поездок можно посмотреть и как водителю, и как пассажиру.

Веб-версия и перспективы развития для мобильных платформ

Сейчас приложение существует как веб-сервис. Интерфейс сделан на React, сервер — на Django, данные хранит PostgreSQL. Адреса вводятся и маршруты строятся через Яндекс Карты [6]. В веб-версии доступны создание и поиск поездок, заявки, чат и личный кабинет. Клиент и сервер разделены, поэтому готовую серверную логику можно будет использовать и дальше.

Следующий шаг — приложения для Android и iOS на основе веб-версии. Пока их нет. Придётся адаптировать интерфейс и проверить на телефонах карты, уведомления и остальное.

В дальнейшем планируется расширение функциональности приложения, включая дополнительные механизмы проверки личности пользователей, внедрение системы оплаты поездок и программы лояльности. Эти изменения направлены на повышение доверия между участниками, упрощение расчетов и создание более удобных условий для регулярных совместных поездок.

Заключение

Разработанная веб-версия приложения объединяет публикацию городских маршрутов, поиск попутчиков по заданным условиям, оформление и подтверждение бронирований, общение участников и систему отзывов. Использование карты помогает учитывать расположение мест посадки, а разделение заявок и подтвержденных мест обеспечивает понятный порядок участия в поездке. Реализованные функции и серверная часть создают основу для последующей подготовки мобильных версий приложения. Таким образом, работа демонстрирует применение веб-технологий для организации совместных городских поездок в рамках единого цифрового сервиса.

Список литературы

1. Дорофеева, Л. В. Влияние урбанизации на инфраструктурное развитие пространства регионов / Л. В. Дорофеева, Н. А. Рослякова, Р. С. Фесенко. – Санкт-Петербург : ООО "Скифия-принт", 2020. – 200 с. – ISBN 978-5-98620-494-9. – EDN OGIIID.

2. Кузнецов В. С. Разработка веб-сервиса для поиска попутчиков в пределах города // Россия молодая : сборник материалов XVII Всероссийской научно-практической конференции с международным участием, Кемерово, 22–25 апреля 2025 года. – Кемерово : КузГТУ, 2025. – С. 31418.1–31418.3. – EDN IWBQPL.
3. Кузнецов В. С. Веб-сервис для оптимизации совместных городских поездок // Студенческий. – 2025. – № 19-2(315). – С. 41–43. – EDN NBILSW.
4. Кузнецов, В. С. Веб-сервис для поиска попутчиков в пределах города / В. С. Кузнецов, Д. В. Булочников // Информационно-телекоммуникационные системы и технологии (ИТСиТ-2025) : Материалы Всероссийской научно-практической конференции, Кемерово, 22–24 октября 2025 года. – Кемерово: Кузбасский государственный технический университет им. Т.Ф. Горбачева, 2025. – С. 249–252. – EDN OOFVJE.
5. PostgreSQL 18 Documentation // PostgreSQL : [сайт]. – URL: https://www.postgresql.org/docs/18/ (дата обращения: 28.10.2026).
6. О JavaScript API Yandex Maps // Yandex Maps API : [сайт]. – URL: https://yandex.com/maps-api/docs/js-api/index.html (дата обращения: 28.10.2026).