МИНОБРНАУКИ РОССИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ**

**ВЫСШЕГО ОБРАЗОВАНИЯ**

**“ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ”**

Факультет *компьютерных наук*

Кафедра *технологий обработки и защиты информации*

*Веб-приложение «Cowёr»*

*Курсовой проект*

*по дисциплине*

*Технологии программирования*

09.03.04 Программная инженерия

6 семестр 2023/2024 учебного года

Зав. Кафедрой \_\_\_\_\_\_\_\_\_\_\_\_\_\_д. ф.-м. н., профессор С. Д. Махортов

Обучающийся \_\_\_\_\_\_\_\_\_\_\_\_\_\_П. В. Печенкин, 3 курс

Обучающийся \_\_\_\_\_\_\_\_\_\_\_\_\_\_Т. А. Улезько, 3 курс

Обучающийся \_\_\_\_\_\_\_\_\_\_\_\_\_\_Е. И. Сидорова, 3 курс

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_ В.С. Тарасов, ст. преподаватель \_\_.\_\_.20\_\_*.*

Воронеж 2024

**СОДЕРЖАНИЕ**

[ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ 3](#_Toc167939245)

[ВВЕДЕНИЕ 7](#_Toc167939246)

[1 Анализ предметной области 8](#_Toc167939247)

[1.1 WORKKODE 8](#_Toc167939248)

[1.2 GrowUp 8](#_Toc167939249)

[1.3 Ясная поляна 9](#_Toc167939250)

[1.4 Итог анализа 9](#_Toc167939251)

[2 Постановка задачи 11](#_Toc167939252)

[2.1 Требования к функциональной части 11](#_Toc167939253)

[2.2 Технические требования 12](#_Toc167939254)

[3 Графическое описание работы системы 13](#_Toc167939255)

[3.1 Диаграммы прецедентов 13](#_Toc167939256)

[3.2 Диаграмма развертывания 15](#_Toc167939257)

[3.3 Диаграммы последовательности 16](#_Toc167939258)

[4 Реализация 19](#_Toc167939259)

[4.1 Реализация базы данных 19](#_Toc167939260)

[4.2 Реализация серверной части приложения 20](#_Toc167939261)

[4.3 Реализация клиентской части приложения 21](#_Toc167939262)

[ЗАКЛЮЧЕНИЕ 25](#_Toc167939263)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 26](#_Toc167939264)

# ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ

Перечень терминов, используемых в курсовом проекте, представлен в таблице Таблица 1.

Таблица - Термины и сокращения

| Термин | Определение термина |
| --- | --- |
| Авторизация | Предоставление определённому лицу прав на выполнение определённых действий, а также процесс проверки (подтверждения) данных прав при попытке выполнения этих действий |
| Авторизованный пользователь | Пользователь, прошедший процесс авторизации |
| Администратор | Человек, имеющий доступ к расширенному функционалу веб-сервиса |
| Бэкенд | Часть программного обеспечения, отвечающая за обработку данных, бизнес-логику и взаимодействие с базой данных. Поддерживает работу фронтенда |
| Валидация | Процесс проверки данных, на соответствие заданным критериям |
| Веб-сайт | Набор связанных по смыслу страниц, которые находятся в сети Интернет. На них размещается информация, текстовая или графическая |
| Веб-сервис | Идентифицируемая уникальным веб-адресом (URL-адресом) программная система со стандартизированными интерфейсами, а также HTML-документ сайта, отображаемый браузером пользователя |
| Коворкинг | Офисное пространство, в котором человек может арендовать рабочее место |
| Модальное окно | Такое окно, которое появляется поверх родительского окна |
| Неавторизованный пользователь | Пользователь, не прошедший процесс авторизации |
| Пользователь | Лицо, которое использует систему для решения стоящих перед ним задач |
| Профиль | Учетная запись пользователя в веб-приложении, вход в которую осуществляется с помощью логина, номера телефона или электронной почты и пароля. В учетной записи содержится информация о пользователе |
| Регистрация | Раздел веб-приложения, предназначенный для создания профиля пользователя, в котором человек указывает свои данные и придумывает пароль |
| СУБД | Система управления базами данных. Комплекс программ, позволяющих создать базу данных (БД) и манипулировать данными |
| Фреймворк | Заготовка, готовая модель в программировании для быстрой разработки, на основе которой можно дописать собственный код. Он задает структуру, определяет правила и предоставляет необходимый набор инструментов для создания проекта |
| Фрилансер | Человек, занимающийся удалённым способом заработка без трудоустройства в штат |
| Фронтенд | Пользовательский интерфейс компьютера или любого устройства |
| Чекаут | Страница или набор страниц, которые покупатели видят, когда переходят к оплате покупки |
| Чекбокс | Элемент графического пользовательского интерфейса, позволяющий пользователю управлять параметром с двумя состояниями - включено и отключено. Во включённом состоянии внутри чекбокса отображается отметка |
| Apache | Программное обеспечение с открытым исходным кодом, которое позволяет создавать веб-сервер |
| API | Программный интерфейс приложения. Описание способов, которыми одна компьютерная программа может взаимодействовать с другой программой |
| C# | Высокоуровневый язык программирования общего назначения |
| CSS | Формальный язык, служащий для описания оформления внешнего вида документа, созданного с использованием языка разметки (HTML, XHTML, XML) |
| HTML | Стандартизированный язык гипертекстовой разметки документов для просмотра веб-страниц в браузере |
| JavaScript | Язык программирования, который используют разработчики для создания интерактивных веб-страниц |
| PostgreSQL | Свободная объектно-реляционная система управления базами данных |
| React | JavaScript-библиотека для создания пользовательских интерфейсов |
| REST | Архитектурный стиль взаимодействия компонентов распределенного приложения в сети |
| YooMoney | Сервис электронных платежей, входящий в экосистему Сбера. |
| YouTrack | Коммерческая система отслеживания ошибок и программное обеспечение для управления проектами, разработанное компанией JetBrains |
| TypeScript | Язык программирования, позиционируемый как средство разработки веб-приложений, расширяющее возможности JavaScript |

# ВВЕДЕНИЕ

В связи с ростом популярности удаленной работы в современном мире увеличивается популярность коворкингов [1]. В связи с этим возникает потребность в решениях для бронирования рабочих мест в коворкингах онлайн.

В процессе общения с заказчиком были выявлены идеи, которые помогут оптимизировать бизнес процессы и увеличить клиентскую базу.

В процессе разработки за основу были взяты уже имеющиеся решения, одной из задач стало сохранение преимуществ этих решений и исправление недостатков. Был сделан упор на функциональность и простоту использования: приложение должно решать ряд задач, но при этом быть простым, интуитивно понятным для всех пользователей и не перегруженным лишними функциями.

# Анализ предметной области

Анализ существующих решений будем проводить на основе сервисов, перечисленных в таблице Таблица 2.

Таблица - Примеры существующих решений

|  |  |
| --- | --- |
| WORKKODE | https://wkode.co/ |
| GrowUp | https://growup-coworking.ru/ |
| Ясная поляна | https://coworkingspb.ru/coworking#openspaces |

## WORKKODE

Веб сервис для аренды рабочих мест и деловых пространств на час, месяц и день.

Преимущества:

* перечисление характеристик рабочих мест;
* фотографии рабочих мест.

Недостатки:

* отсутствие карты коворкинга с выбором рабочего места на ней;
* отсутствие скидок для студентов;
* отсутствие возможности забронировать место меньше, чем на час;
* отсутствие возможности осуществить и оплатить бронь на сайте.

## GrowUp

Веб сервис для аренды офисов для 3-50 человек на час, день, 10 дней и месяц.

Преимущества:

* перечисление характеристик рабочих мест;
* фотографии рабочих мест.

Недостатки:

* отсутствие карты коворкинга с выбором рабочего места на ней;
* отсутствие скидок для студентов;
* отсутствие возможности забронировать место меньше, чем на час;
* отсутствие возможности осуществить и оплатить бронь на сайте.

## Ясная поляна

Веб сервис для аренды рабочего места в одном из деловых пространств компании.

Преимущества:

* перечисление характеристик рабочих мест;
* фотографии рабочих мест.

Недостатки:

* отсутствие карты коворкинга с выбором рабочего места на ней;
* отсутствие информации о возможном времени бронирования на сайте;
* отсутствие скидок для студентов;
* отсутствие возможности забронировать место меньше, чем на час;
* отсутствие возможности осуществить и оплатить бронь на сайте.

## Итог анализа

В процессе анализа предметной области было установлено, что при разработке приложения следует придерживаться следующих аспектов:

* наличие карты коворкинга с выбором рабочего места на ней;
* наличие скидок для студентов и волонтеров;
* возможность бронирования рабочего места на сайте;
* возможность оплаты брони на сайте;
* наличие информации о рабочих местах;

# Постановка задачи

Целью данного курсового проекта является разработка веб-сервиса, предназначенного для бронирования рабочих мест в коворкинге, в разрабатываемом сервисе должна быть предусмотрена возможность просмотра карты офиса, выбора на ней рабочего места и последующего его бронирования.

## Требования к функциональной части

Приложение должно содержать следующие бизнес-роли пользователей:

* неавторизованный пользователь;
* авторизованный пользователь;
* администратор.

Должна быть разработана следующая функциональность для неавторизованного пользователя:

* возможность регистрации и авторизации на сервисе;
* возможность просмотра карты коворкинга по этажам;
* возможность выбора даты бронирования;
* возможность выбора времени начала и окончания бронирования;
* возможность выбора рабочего места на карте и просмотра описания этого места;
* возможность восстановления пароля.

Должна быть разработана следующая функциональность для авторизованного пользователя:

* возможность просмотра карты коворкинга по этажам;
* возможность выбора рабочего места на карте и просмотра описания этого места;
* возможность выбора даты бронирования;
* возможность выбора времени начала и окончания бронирования;
* возможность бронирования выбранного места;
* возможность оплаты заказа;
* возможность просмотра заказов, созданных пользователем;
* возможность отмены заказа;
* возможность просмотра и редактирования имени, фамилии, номера телефона в личном кабинете.

Должна быть разработана следующая функциональность для администратора:

* возможность просмотра карты коворкинга;
* возможность добавления, перемещения, удаления рабочих мест на карте коворкинга по этажам;
* возможность изменения описания рабочего места;
* возможность изменения цены на рабочее место за час;
* возможность просмотра, добавления и отмены заказов пользователей.

## Технические требования

Приложение должно удовлетворять следующим техническим требованиям:

* возможность авторизации пользователей с помощью электронной почты и пароля;
* хранение данных для входа в сервисы moodle и BRS;
* хранение необходимых данных в базе данных;
* интеграция с сервисом YooMoney;
* наличие панели администратора.

# Графическое описание работы системы

## Диаграммы прецедентов

Далее будут приведены диаграммы Use-Case (диаграммы прецедентов) для разных ролей пользователей. Эти диаграммы демонстрируют различные сценарии, возникающие при использовании сервиса.

На рисунке Рисунок 1 представлена Use-case диаграмма для неавторизованного пользователя.

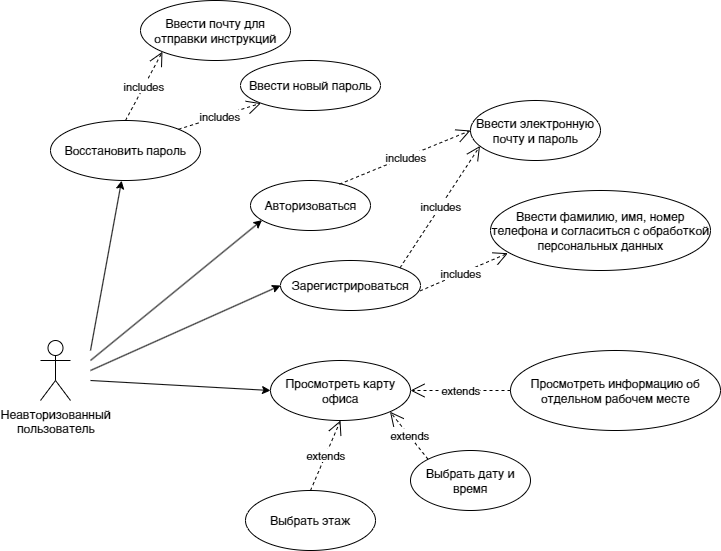


Рисунок - Use-case диаграмма для неавторизованного пользователя

На рисунке Рисунок 2 представлена Use-case диаграмма для авторизованного пользователя.

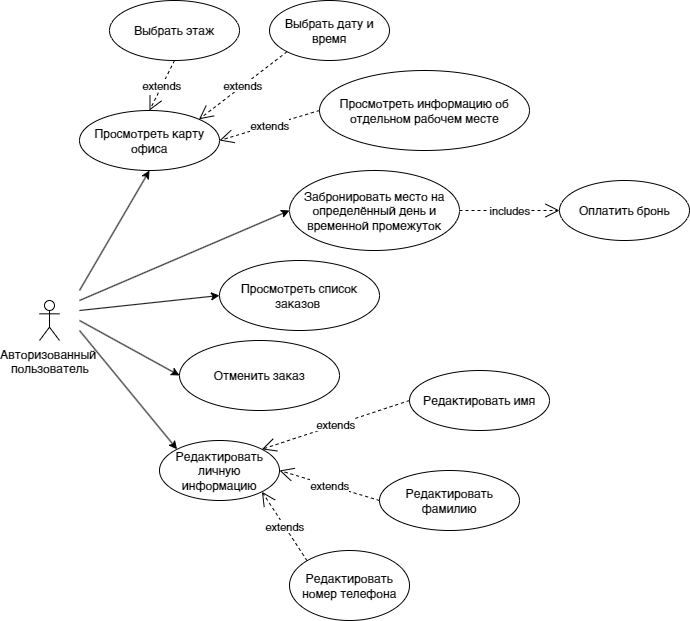


Рисунок - Use-case диаграмма для авторизованного пользователя

На рисунке Рисунок 3 представлена Use-case диаграмма для администратора.

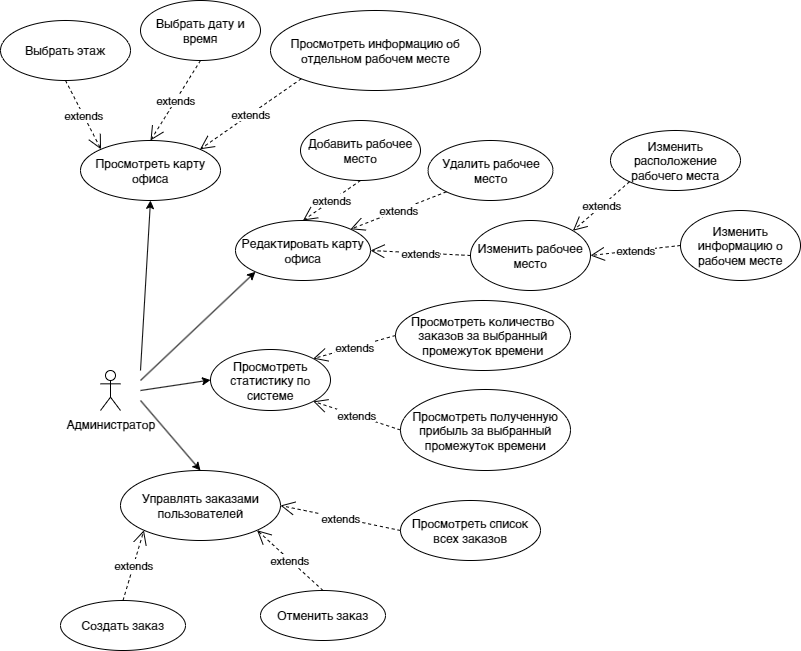


Рисунок - Use-case диаграмма для администратора

## Диаграмма развертывания

На рисунке Рисунок 4 представлена диаграмма развёртывания. Она используется для моделирования архитектуры системы, ее компонентов. Она показывает структуру системы, то есть как ее компоненты взаимодействуют между собой.

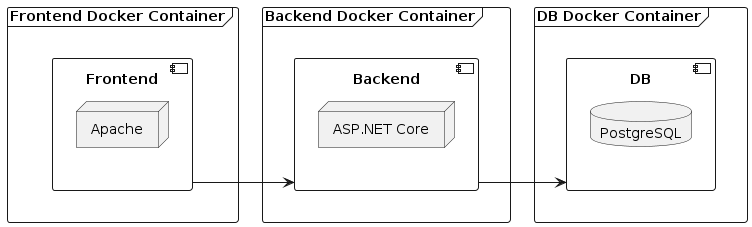


Рисунок - Диаграмма развертывания

## Диаграммы последовательности

На рисунках Рисунок 5, Рисунок 6, Рисунок 7, Рисунок 8, Рисунок 9 представлены диаграммы последовательности, они позволяют описать последовательность происходящих событий при взаимодействии пользователя с системой для различных процессов.

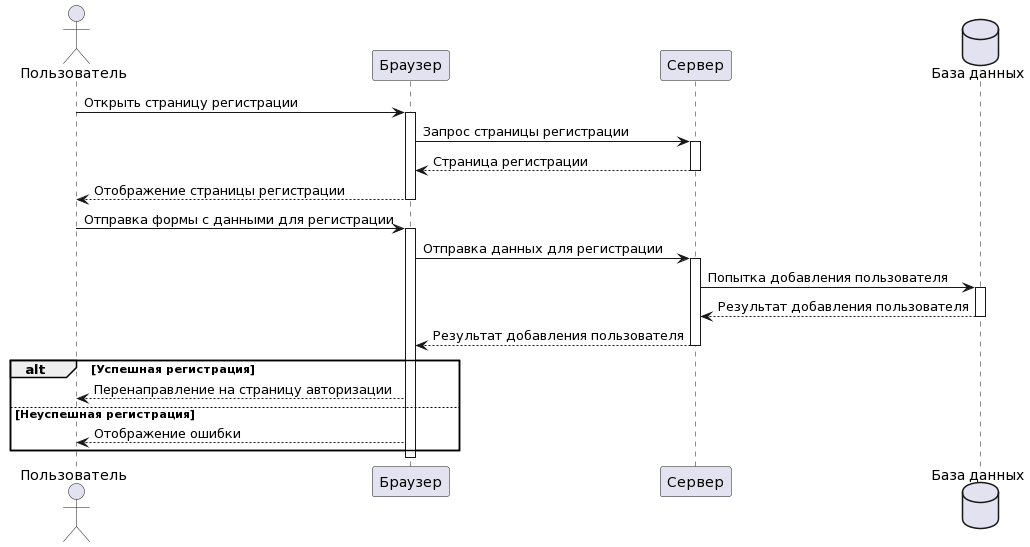


Рисунок - Диаграмма последовательности для процесса регистрации

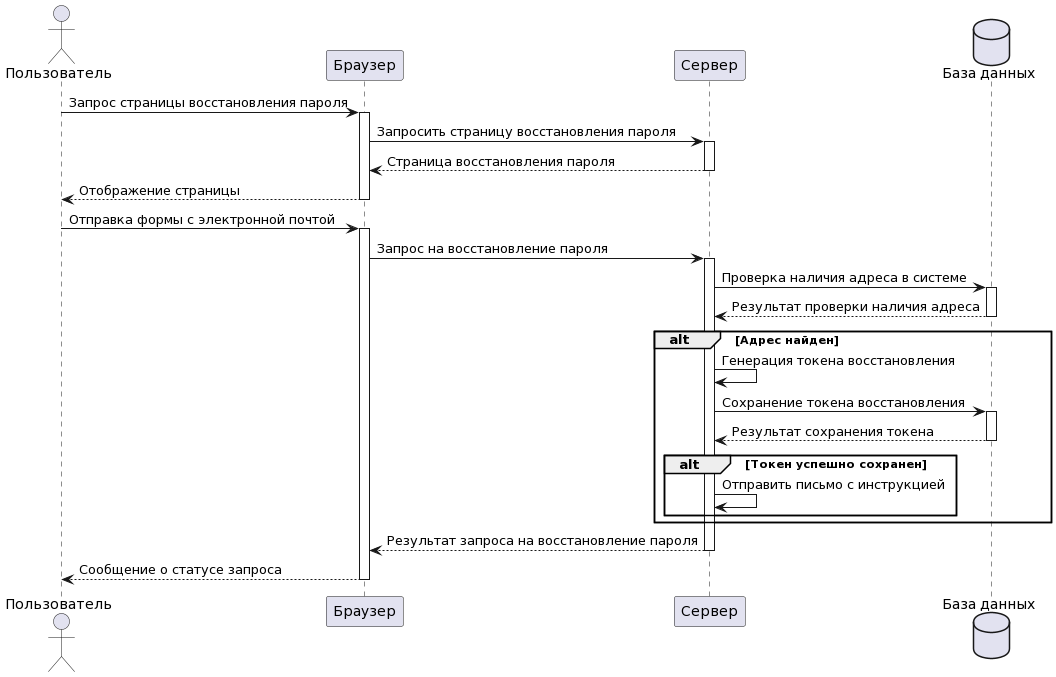


Рисунок - Диаграмма последовательности для процесса запроса восстановления пароля

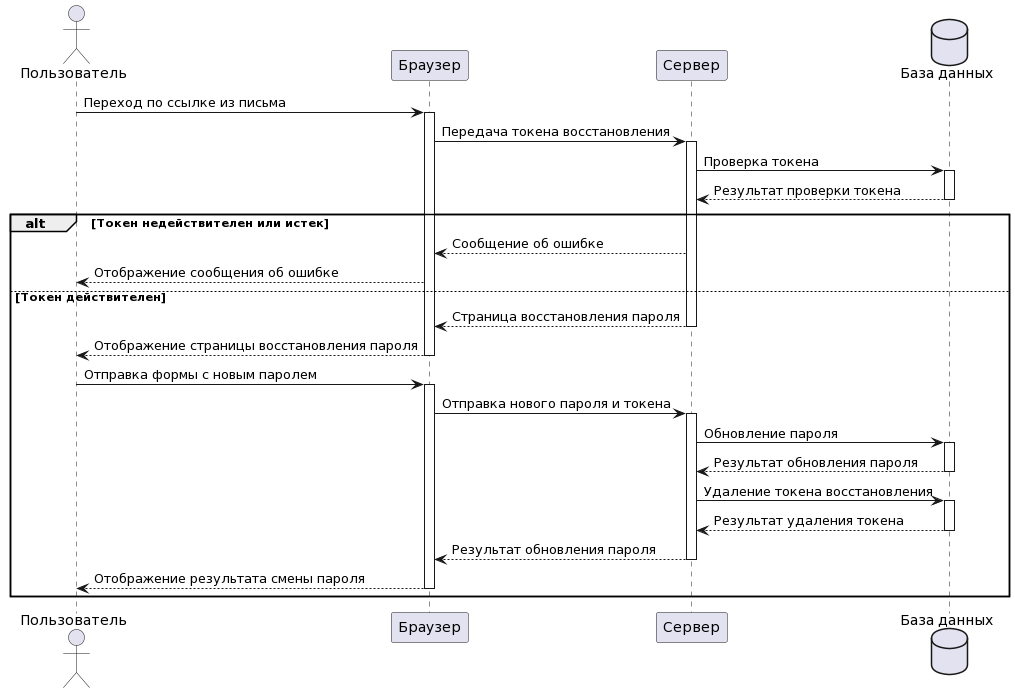


Рисунок - Диаграмма последовательности для процесса восстановления пароля

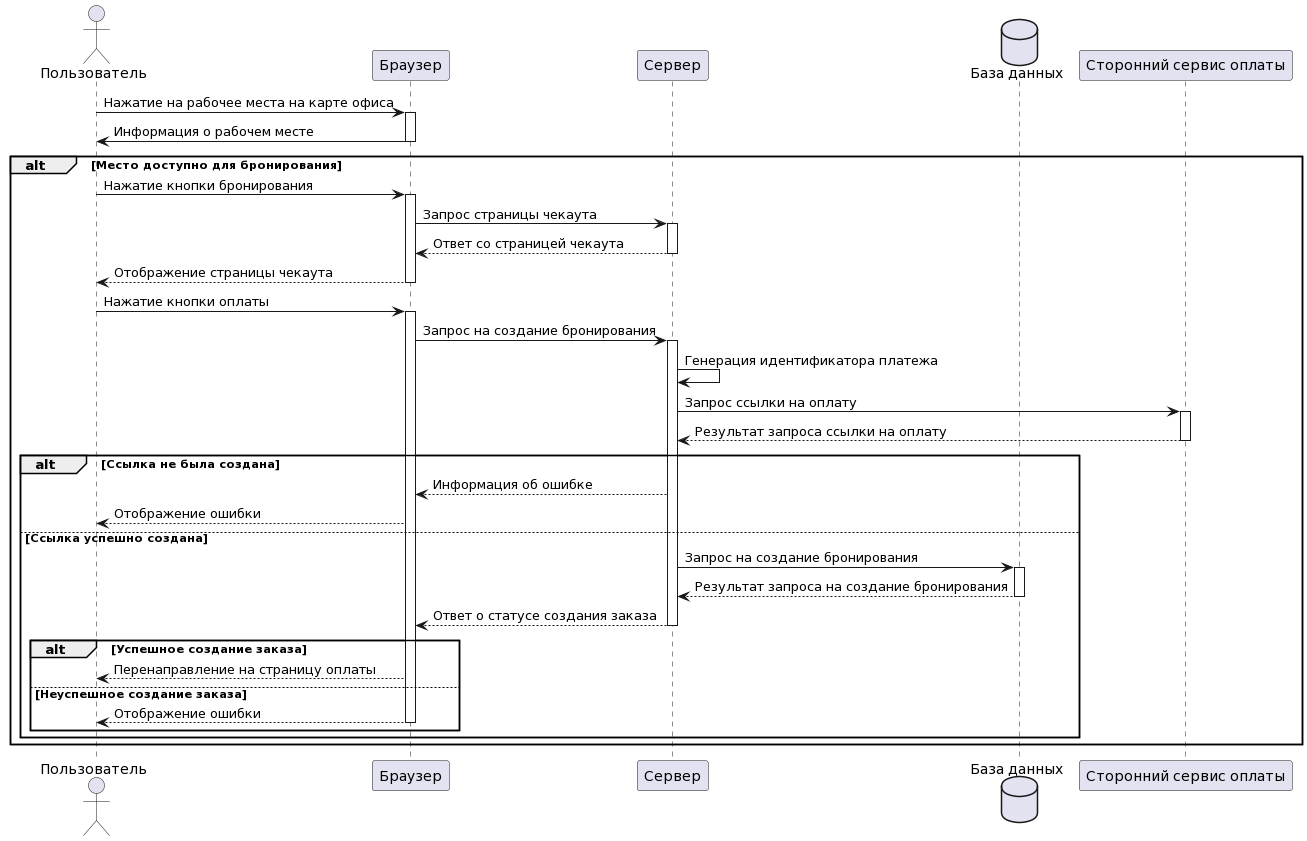


Рисунок - Диаграмма последовательности для процесса создания бронирования

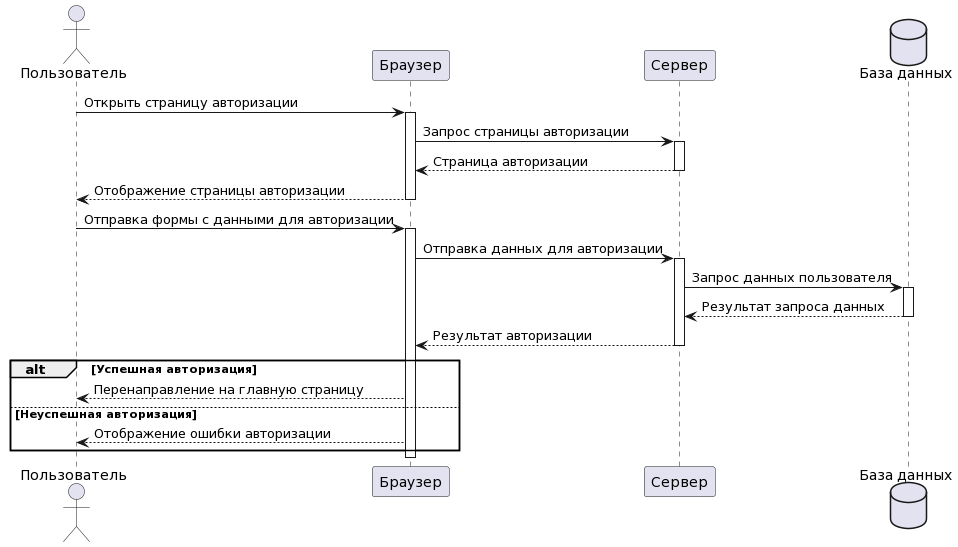


Рисунок - Диаграмма последовательности для процесса авторизации

# Реализация

Приложение должно быть реализовано с использованием трехуровневой клиент-серверной архитектуры, состоящей из клиента (веб-приложение), сервера и базы данных.

Для реализации серверной части были выбраны следующие средства:

* язык программирования C#;
* фреймворк ASP.NET Core;
* СУБД PostgreSQL.

Для реализации серверной части были выбраны следующие средства:

* HTML;
* CSS;
* язык программирования TypeScript;
* библиотека React.

## Реализация базы данных

На рисунке Рисунок 10 представлена ER-диаграмма используемой базы данных

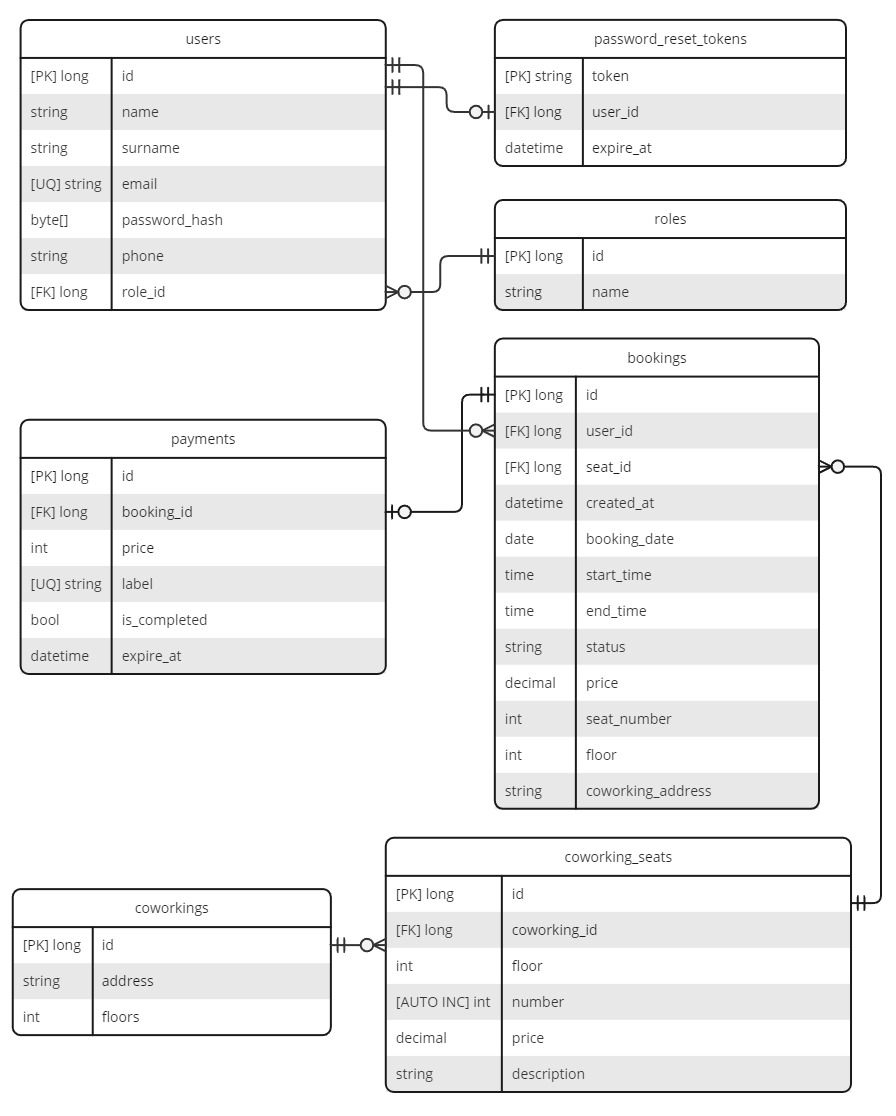


Рисунок – ER-диаграмма

## Реализация серверной части приложения

Серверная часть приложения состоит из следующих компонентов:

* Проект Cower.Domain, состоящий из доменных моделей, которые используются в нескольких слоях приложения.
* Проект Cower.Data, являющийся слоем доступа к данным. Содержит в себе репозитории и Dal объекты.
* Проект Cower.Service, являющийся слоем бизнес логики. В нем находятся сервисы, которые используются слой доступа к данным, и вызывается из слоя презентации.
* Проект Cower.Web, являющйися слоем презентации. Внутри него находятся REST контроллеры, которые используются клиентской частью.

Настройки приложение берет из переменных среды при запуске.

## Реализация клиентской части приложения

### Общая информация

Клиентская часть приложения отвечает за пользовательский интерфейс и взаимодействие пользователя с данными.

### Графический интерфейс

Далее будут представлены основные страницы веб-сайта и приведена информация о них.

На рисунке Рисунок 11 представлена страница авторизации. Пользователь может ввести почту и пароль, чтобы войти в аккаунт. При нажатии на кнопку «Регистрация» пользователь переходит на страницу регистрации.

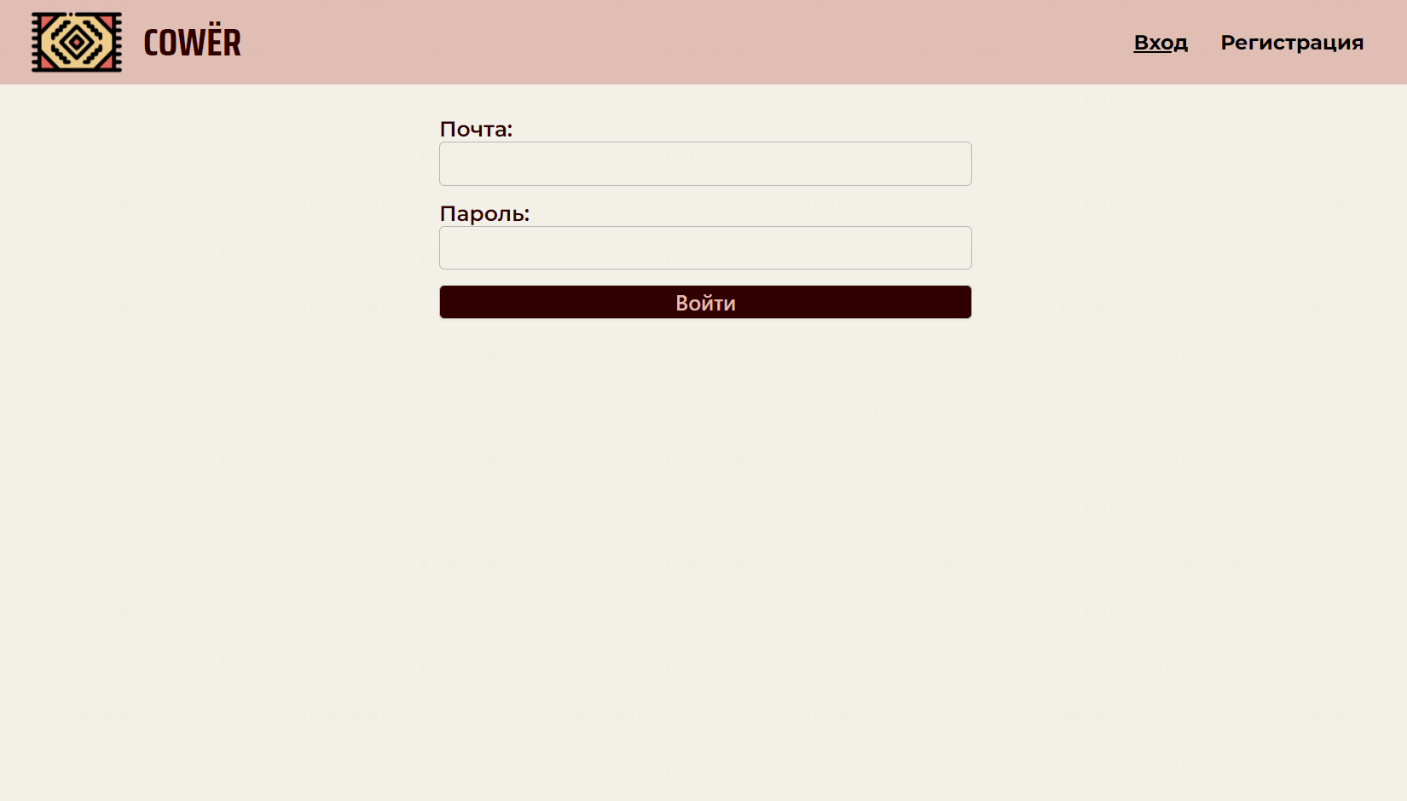


Рисунок – Страница авторизации

На рисунке Рисунок 12 представлена страница регистрации. Пользователь может ввести почту и пароль (обязательно), имя, фамилию, номер телефона (опционально) и нажать на кнопку «Зарегистрироваться», чтобы зарегистрироваться.

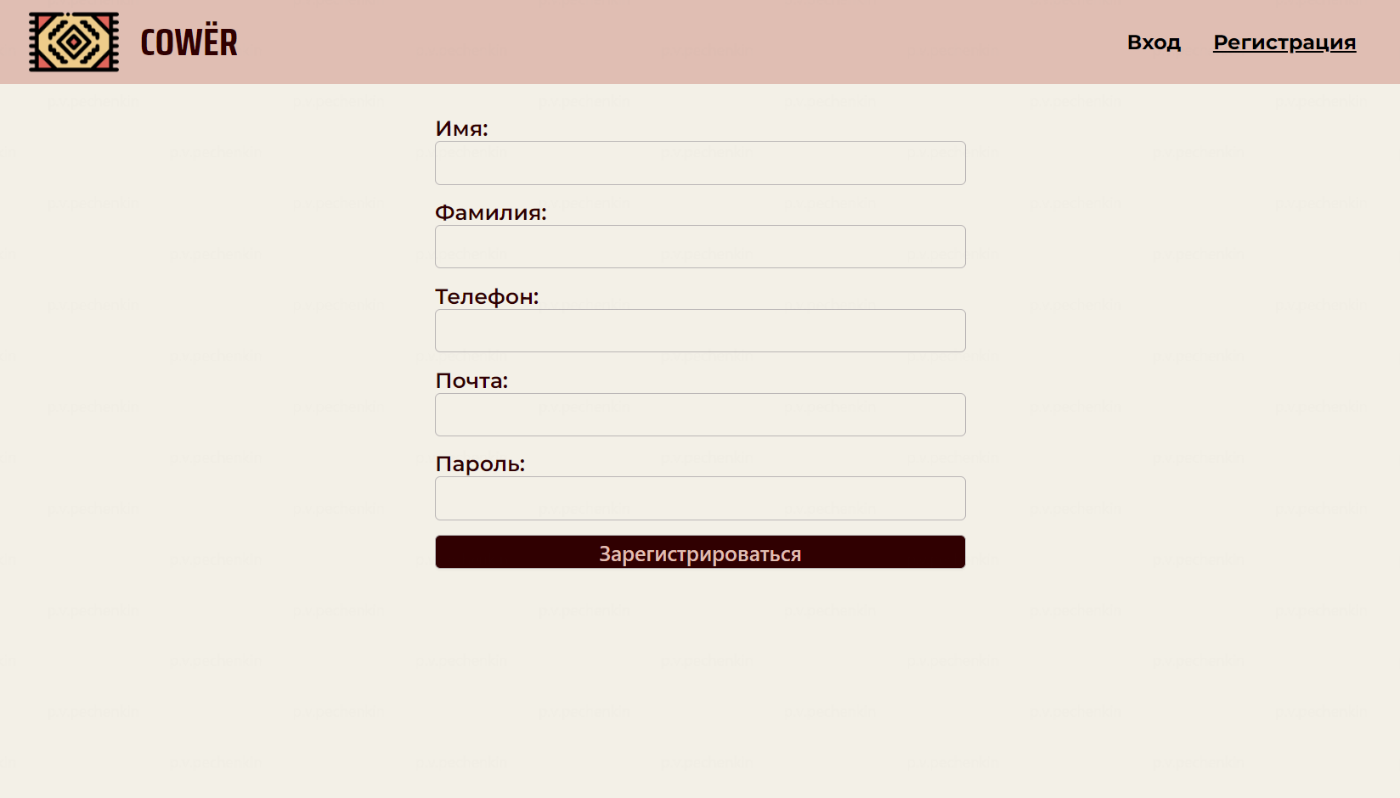


Рисунок – Страница регистрации

На рисунке Рисунок 13 представлена главная страница с картой офиса. Пользователь может выбрать адрес офиса, дату и время бронирования, а также нажать на рабочее место на карте офиса и посмотреть информацию о нем. При нажатии на иконку в правом верхнем углу экрана пользователь может перейти на станицу личного кабинета, на страницу со списком заказов или выйти.

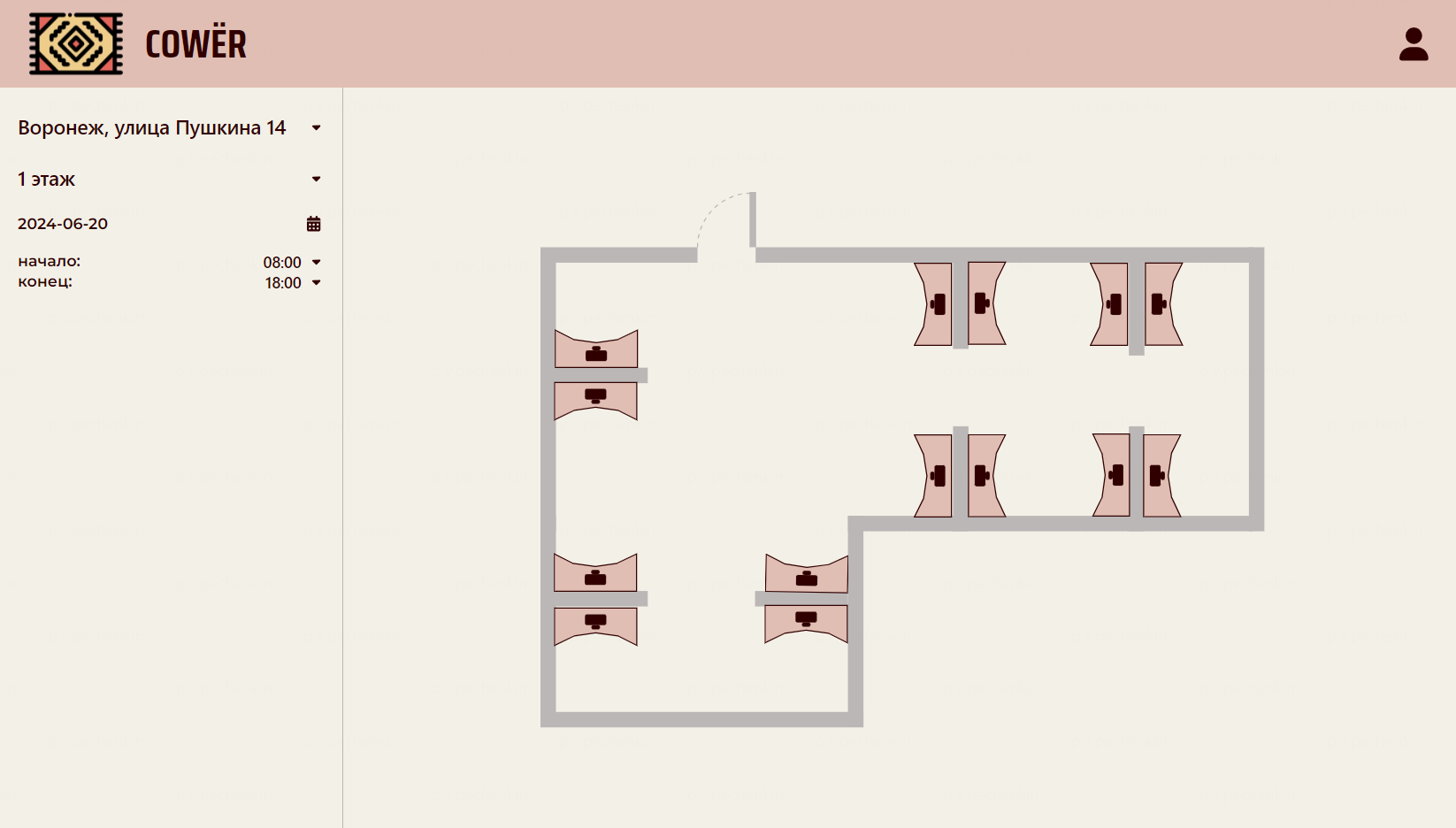


Рисунок – Главная страница

На рисунке Рисунок 14 представлена страница чекаута с оплатой, на которую пользователь переходит после нажатия на кнопку «Забронировать» на главной странице.

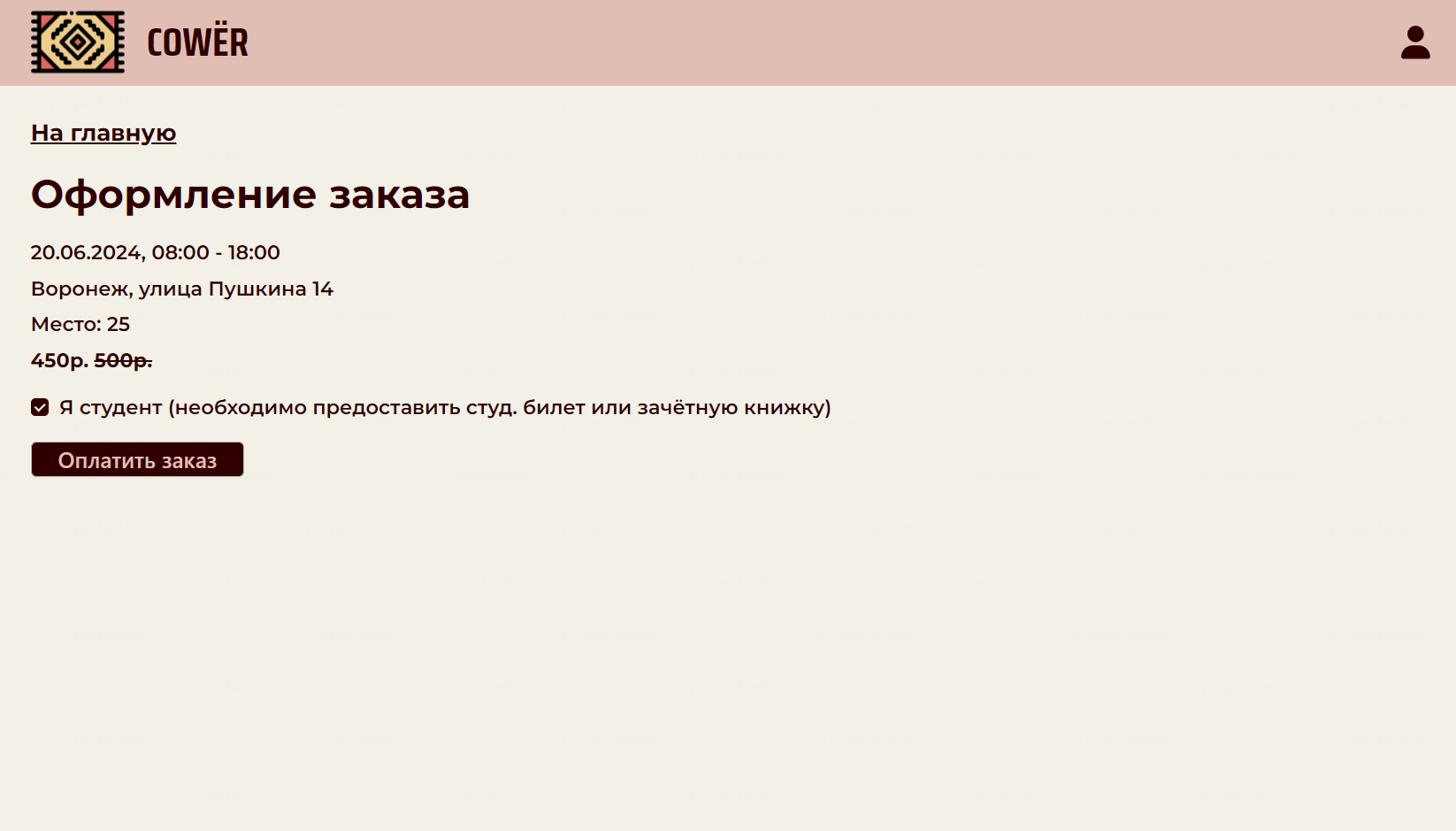


Рисунок – Чекаут с оплатой

На рисунке Рисунок 15 представлена страница со списком заказов. Пользователь может оплатить неоплаченный заказ или отменить заказ, если время, на которое совершена бронь, еще не наступило.

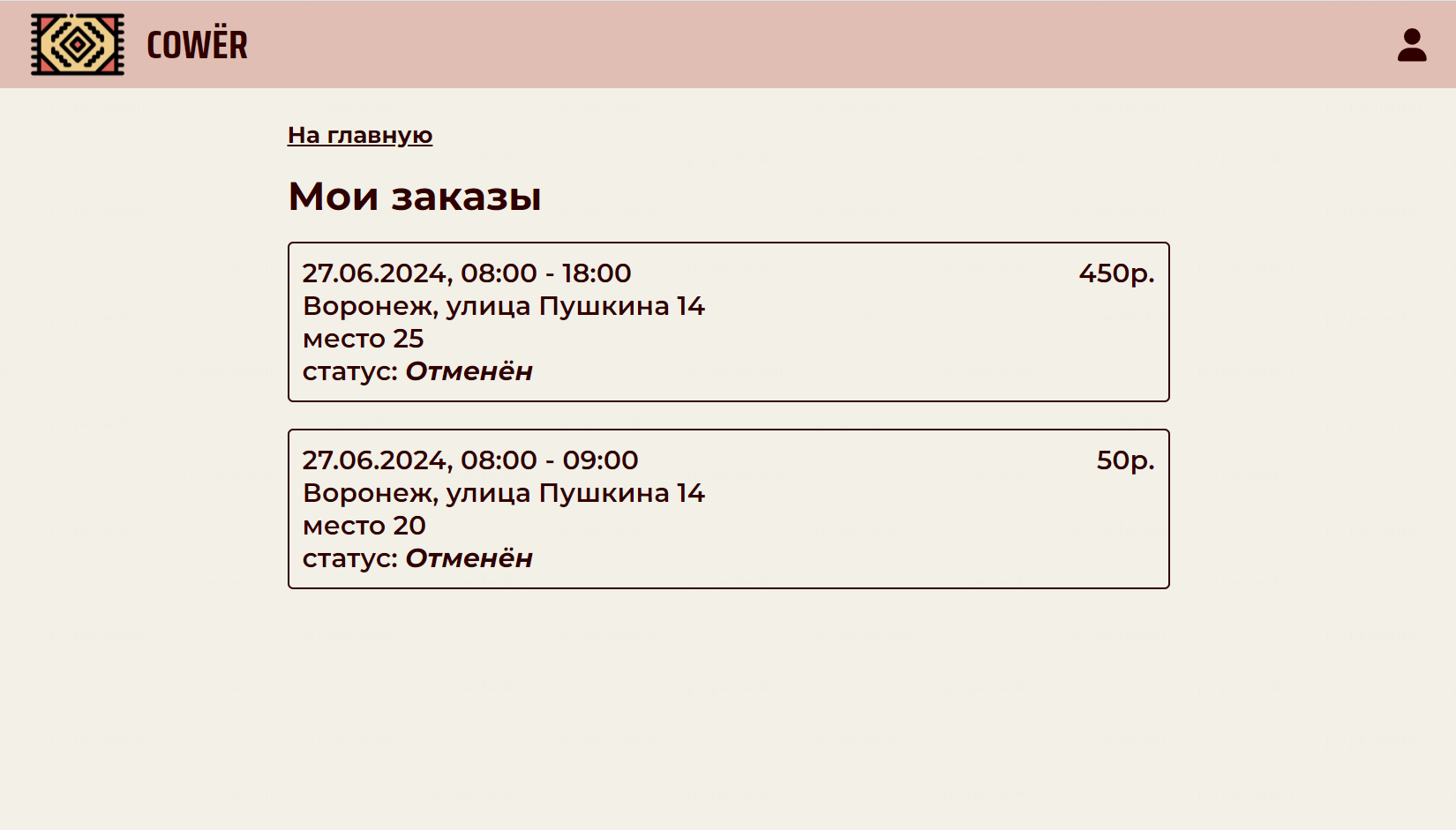


Рисунок – Список заказов

# **ЗАКЛЮЧЕНИЕ**

В ходе реализации данного проекта были достигнуты поставленные задачи. Был разработан веб-сервис, предназначенный для бронирования рабочих мест в коворкинге. В разработанном сервисе предусмотрена возможность просмотра карты офиса, выбора на ней рабочего места и последующего его бронирования.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

Исследуем, как растет рынок коворкинга в России [Электронный ресурс]. – Режим доступа: https://bankspace.ru/blog/koworking\_research (Дата обращения: 20.03.2024).

WORKKODE [Электронный ресурс] – Режим доступа: https://wkode.co/ (Дата обращения: 29.04.2024).

GrowUp [Электронный ресурс] – Режим доступа: https://growup-coworking.ru/ (Дата обращения: 20.03.2023).а

Ясная поляна [Электронный ресурс] – Режим доступа: https://coworkingspb.ru/coworking#openspaces (Дата обращения: 20.03.2023).