МИНОБРНАУКИ РОССИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

(ФГБОУ ВО «ВГУ»)

Факультет компьютерных наук

Кафедра технологий обработки и защиты информации

Курсовой проект

По дисциплине: Технологии программирования

Веб-приложение «Cowёr»

09.03.04 Программная инженерия

Информационные системы и сетевые технологии

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](#_Toc169815592)

[ВВЕДЕНИЕ 6](#_Toc169815593)

[1 Анализ предметной области 7](#_Toc169815594)

[1.1 WORKKODE 7](#_Toc169815595)

[1.2 GrowUp 8](#_Toc169815596)

[1.3 Ясная поляна 8](#_Toc169815597)

[1.4 Итог анализа 9](#_Toc169815598)

[2 Постановка задачи 10](#_Toc169815599)

[2.1 Требования к функциональной части 10](#_Toc169815600)

[2.2 Требования к программному обеспечению веб-сервиса 11](#_Toc169815601)

[2.3 Общие требования к оформлению и верстке страниц 12](#_Toc169815602)

[3 Графическое описание работы системы 13](#_Toc169815603)

[3.1 Диаграммы прецедентов 13](#_Toc169815604)

[3.2 Диаграмма развертывания 15](#_Toc169815605)

[3.3 Диаграммы последовательности 16](#_Toc169815606)

[4 Реализация 19](#_Toc169815607)

[4.1 Реализация базы данных 19](#_Toc169815608)

[4.2 Реализация серверной части приложения 20](#_Toc169815609)

[4.3 Реализация клиентской части приложения 23](#_Toc169815610)

[5 Тестирование 30](#_Toc169815611)

[ЗАКЛЮЧЕНИЕ 31](#_Toc169815612)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 32](#_Toc169815613)

# ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ

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

Таблица 1 - 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 |
| 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 |
| 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 |
| TypeScript | Язык программирования, позиционируемый как средство разработки веб-приложений, расширяющее возможности JavaScript |

# ВВЕДЕНИЕ

C 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 разработки за основу были взяты уже имеющиеся решения, одной из задач стало сохранение преимуществ этих решений и исправление недостатков.

Целью данного курсового проекта является разработка веб-приложения «Cowёr», предоставляющего пользователям возможность выбирать рабочие места на карте коворкинга и бронировать их, оплатить бронь онлайн, студентам и волонтерам получать скидки на бронирование.

# Анализ предметной области

Анализ существующих решений будет проводиться на основе сервисов, перечисленных в таблице Таблица 2.

Таблица 2 - 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 день, месяц и год.

Преимущества:

* перечисление характеристик рабочих мест;
* наличие информации о коворкинге на главной странице.

Недостатки:

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

## GrowUp

Веб сервис для аренды рабочих мест в коворкинге на час, день, 10 дней и месяц.

Преимущества:

* возможность забронировать рабочее место на час;
* наличие возможности заказать организацию мероприятия.

Недостатки:

* отсутствие карты коворкинга с выбором рабочего места на ней;
* отсутствие скидок для студентов;
* отсутствие информации о возможном времени бронирования на сайте;
* отсутствие подробной информации о рабочих местах;
* отсутствие возможности осуществить и оплатить бронь на сайте.

## Ясная поляна

Веб сервис для аренды рабочих мест в коворкинге на час, день и месяц.

Преимущества:

* наличие подробной информации о каждой зоне коворкинга;
* возможность забронировать рабочее место на час;
* наличие скидок для студентов.

Недостатки:

* отсутствие карты коворкинга с выбором рабочего места на ней;
* отсутствие информации о возможном времени бронирования на сайте;
* 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;
* наличие информации о возможном времени бронирования на сайте.

# Постановка задачи

Целью данного курсового проекта является разработка веб-сервиса «Cowёr», предназначенного для бронирования рабочих мест в коворкинге.

## Требования к функциональной части

Приложение должно содержать следующие бизнес-роли пользователей:

* неавторизованный пользователь;
* авторизованный пользователь;
* администратор.

Для неавторизованного пользователя должны предоставляться следующие возможности:

* регистрация и авторизация на сервисе;
* просмотр карты коворкинга по этажам;
* выбор даты бронирования;
* выбор времени начала и окончания бронирования;
* выбор рабочего места на карте и просмотр описания этого места;
* восстановление пароля.

Для авторизованного пользователя должны предоставляться следующие возможности:

* просмотр карты коворкинга по этажам;
* выбор рабочего места на карте и просмотра описания этого места;
* выбор даты бронирования;
* выбор времени начала и окончания бронирования;
* бронирование выбранного места;
* оплата заказа;
* просмотр заказов, созданных пользователем;
* отмена заказа, при условии, что время, на которое совершена бронь еще не наступило;
* просмотр и редактирование имени, фамилии, номера телефона в личном кабинете.

Для администратора должны предоставляться следующие возможности:

* просмотр карты коворкинга;
* добавление, перемещение, удаление рабочих мест на карте коворкинга по этажам;
* изменение описания рабочего места;
* изменение цены на рабочее место за час;
* просмотр, добавление и отмена заказов пользователей.

## Требования к программному обеспечению веб-сервиса

Для реализации серверной части приложения будут использоваться следующие средства:

* язык программирования C# версии 12;
* фреймворк ASP.NET Core версии 8.0;
* СУБД PostgreSQL версии 16.2.

Для реализации клиентской части приложения будут использоваться следующие средства:

* HTML версии 5;
* CSS версии 3;
* язык программирования TypeScript версии 5.2.2;
* библиотека React 18.2.0.

## Общие требования к оформлению и верстке страниц

Сайт должен быть оформлен в одной цветовой палитре с использованием ограниченного набора шрифтов. У страниц сайта должен быть единый стиль.

Основные цвета веб-сайта:

* бежевый используется как цвет фона веб-сайта;
* розовый используется как цвет верхней панели и рабочих мест на карте;
* бордовый используется как цвет основного текста и иконок.

Основные шрифты веб-сайта:

* Saira;
* Montserrat;

В оформлении приложения должны присутствовать разработанные название и логотип.

Необходимо корректное и одинаковое отображение страниц сайта в следующих браузерах:

* Google Chrome версии 122.0.6261 и новее;
* Mozilla Firefox версии 123.0 и новее;
* Opera версии 108.0.5067 и новее;
* Яндекс Браузер версии 23.9.0 и новее.

Необходимо корректное отображение интерфейса на персональных компьютерах с операционными системами Windows версии 7 и выше, с разрешением окна браузера от 1280 x 720 пкс.

# Графическое описание работы системы

В данном разделе содержится 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.

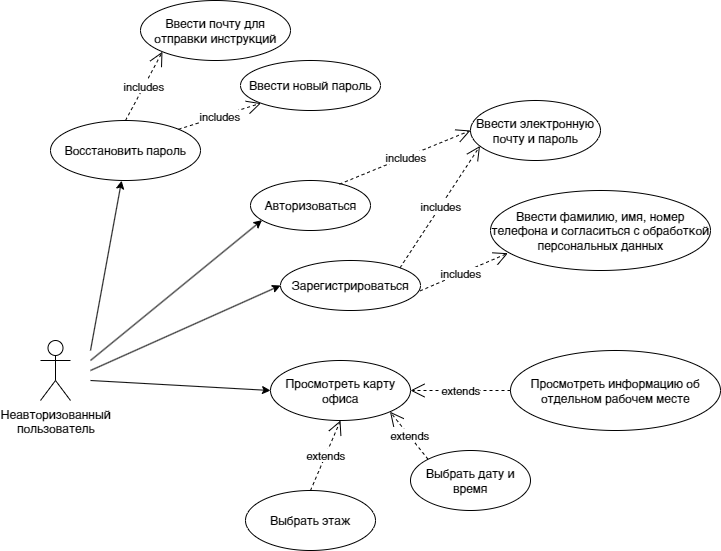


Рисунок 1 - Use-case диаграмма для неавторизованного пользователя

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

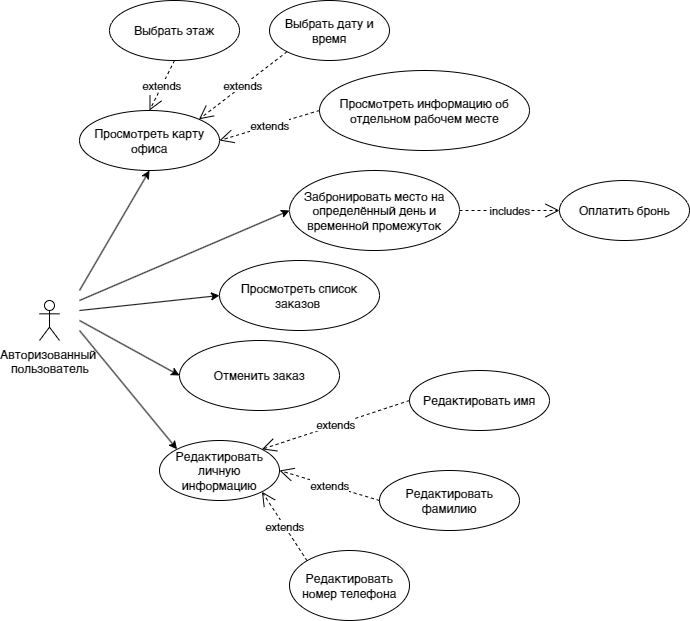


Рисунок 2 - Use-case диаграмма для авторизованного пользователя

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

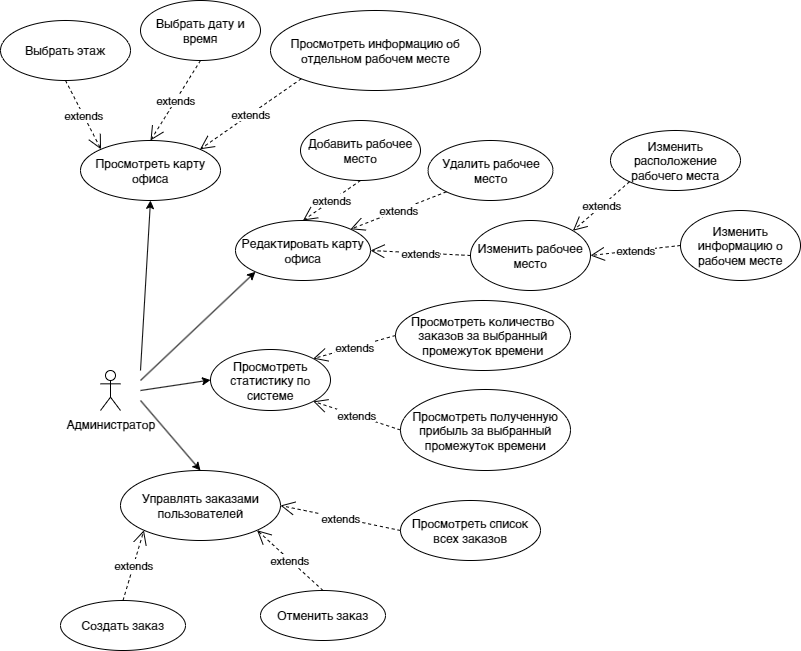


Рисунок 3 - Use-case диаграмма для администратора

## 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.

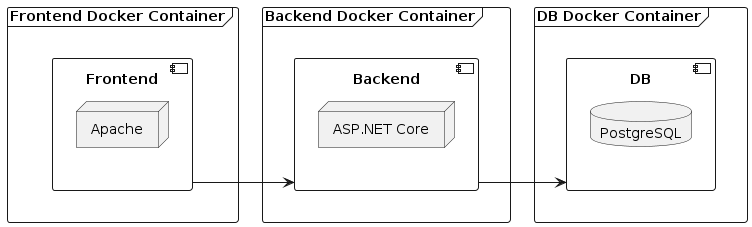


Рисунок 4 - Диаграмма развертывания

## 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.

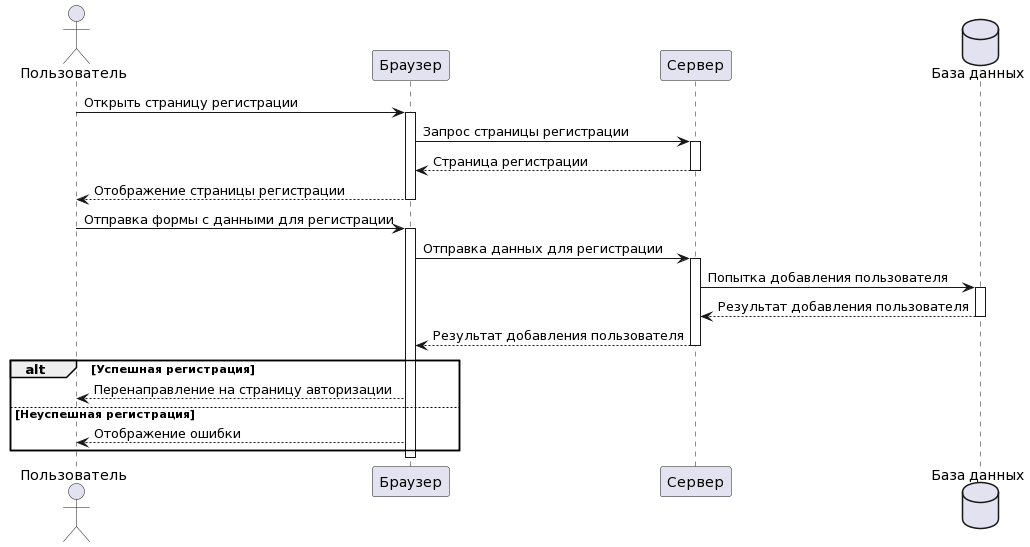


Рисунок 5 - Диаграмма последовательности для процесса регистрации

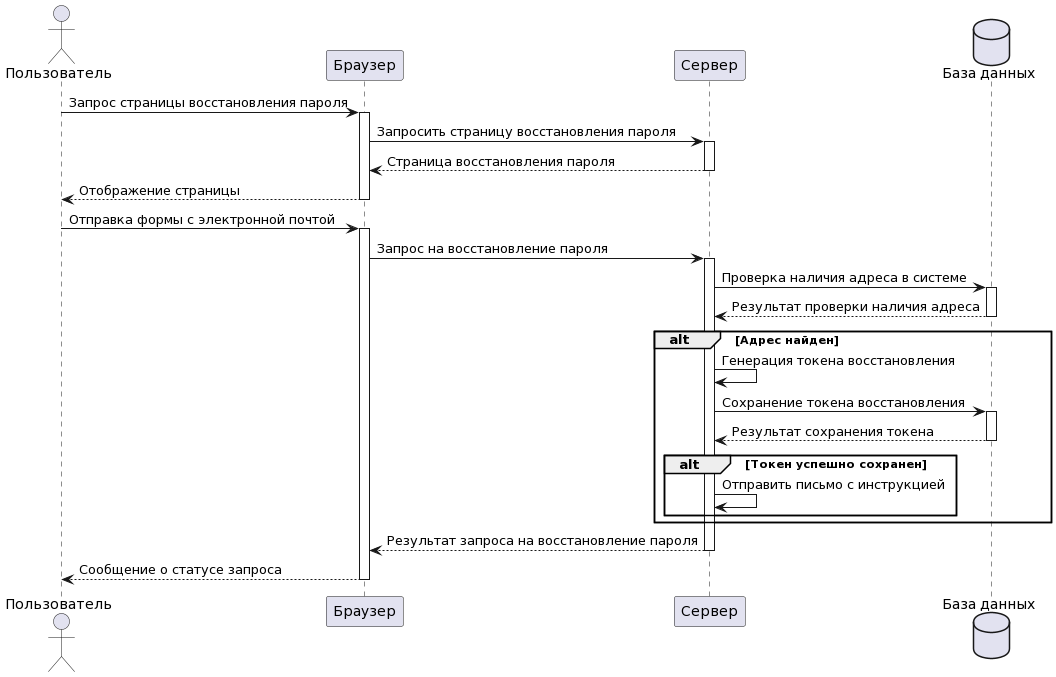


Рисунок 6 - Диаграмма последовательности для процесса запроса восстановления пароля

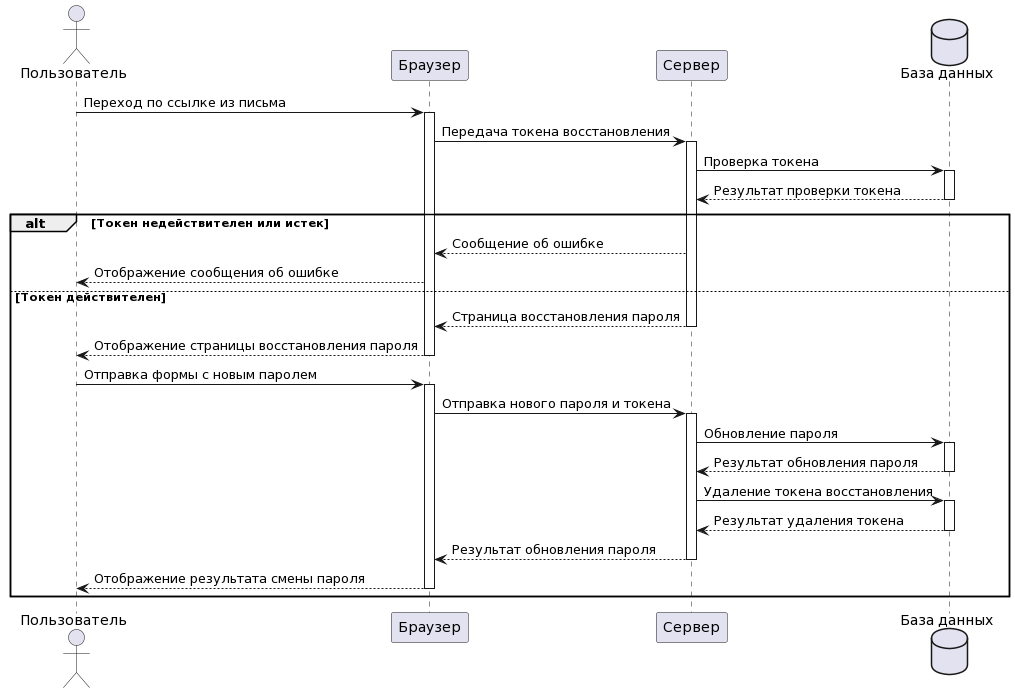


Рисунок 7 - Диаграмма последовательности для процесса восстановления пароля

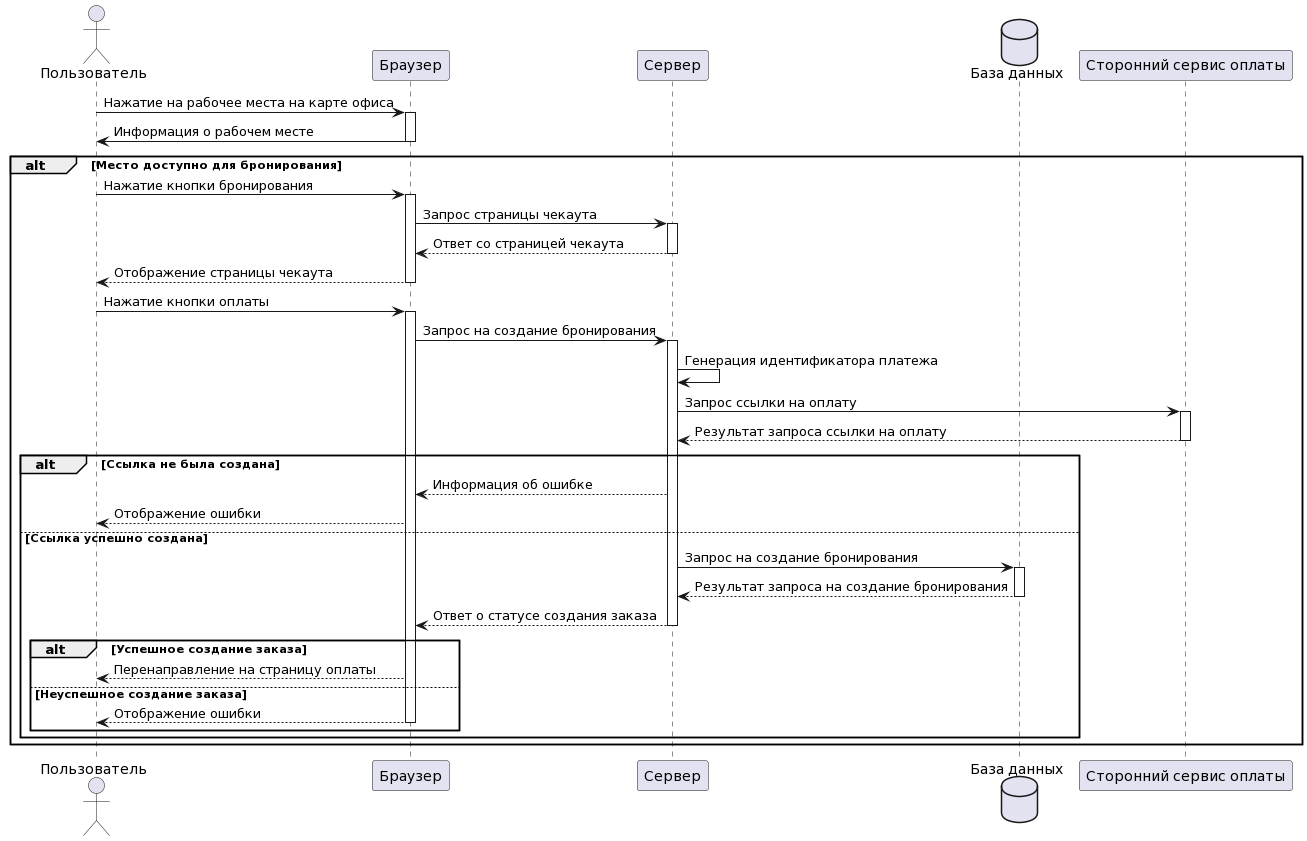


Рисунок 8 - Диаграмма последовательности для процесса создания бронирования

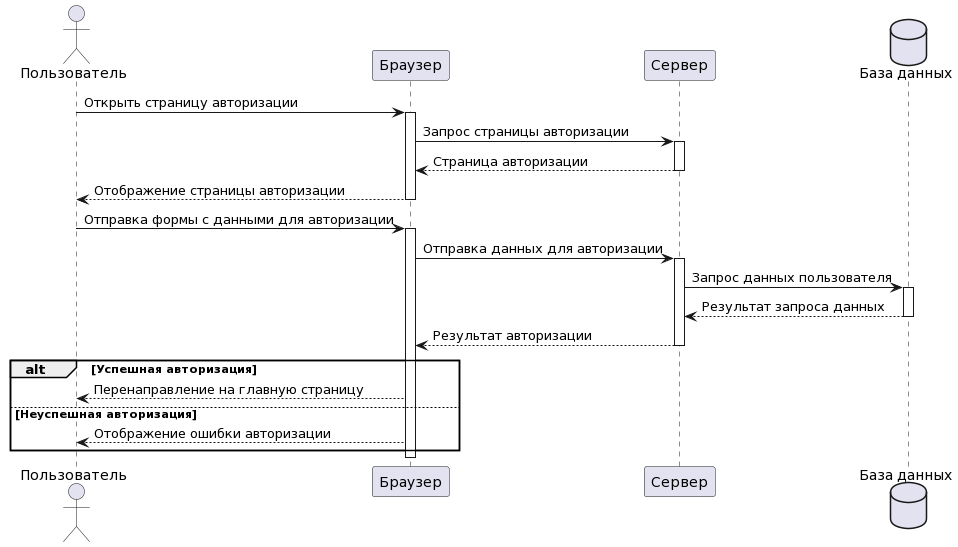


Рисунок 9 - Диаграмма последовательности для процесса авторизации

# 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 рисунках Рисунок 10 и Рисунок 11 представлена ER-диаграмма используемой базы данных

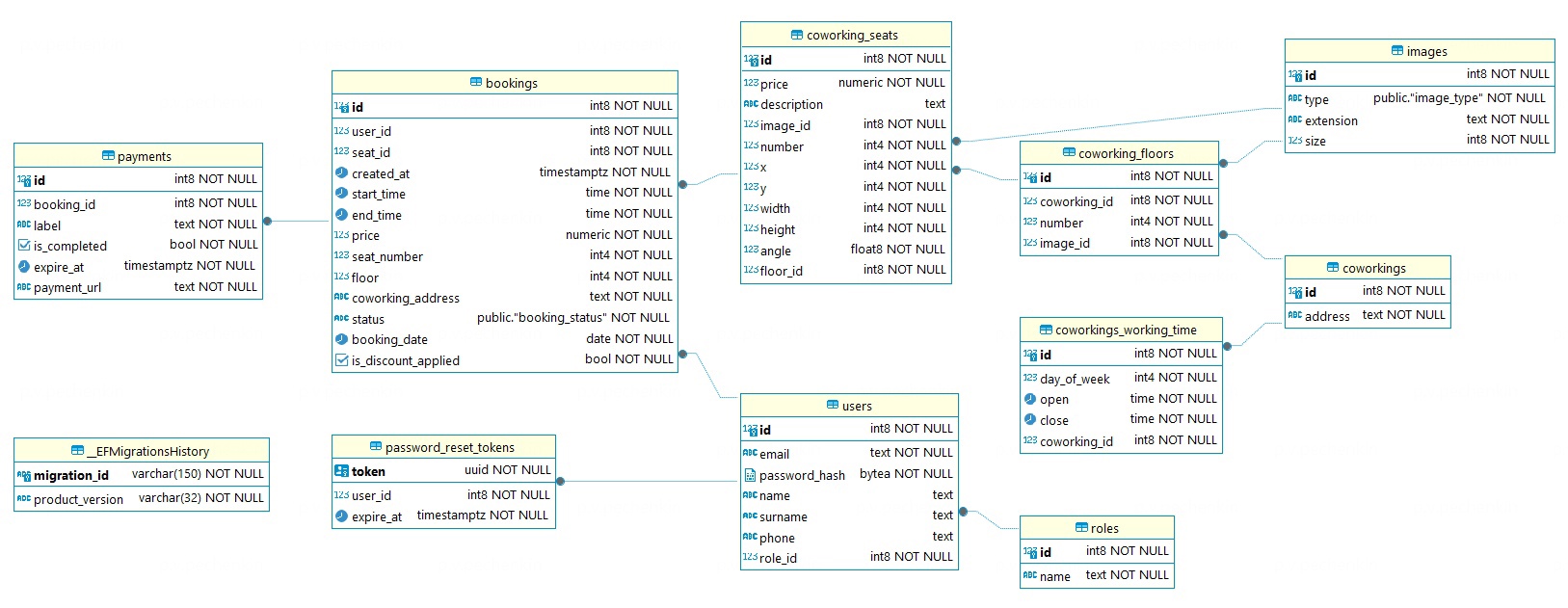


Рисунок 10 – ER-диаграмма, 1 часть

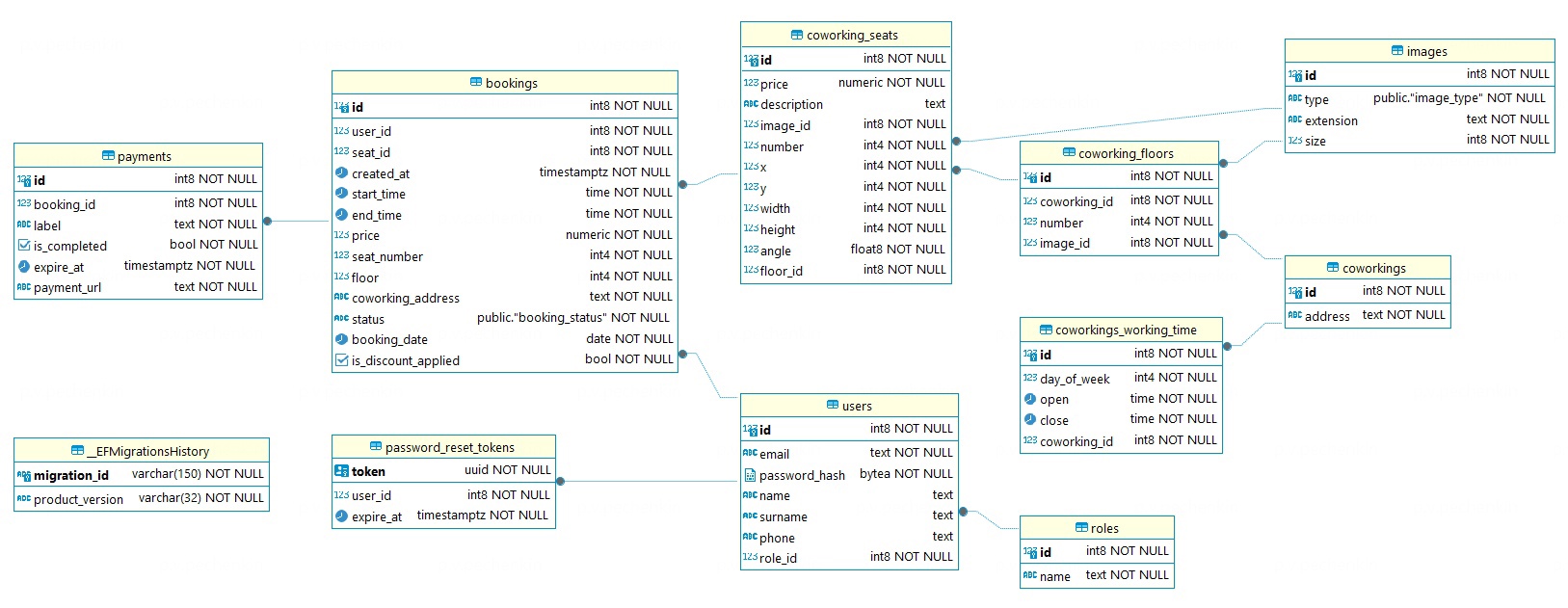


Рисунок 11 - ER-диаграмма, 2 часть

## Реализация серверной части приложения

Серверная часть приложения состоит из следующих компонентов:

* Проект Cower.Domain, состоящий из доменных моделей, которые используются в нескольких слоях приложения (диаграмма классов доменной модели представлена на рисунке Рисунок 12).
* Проект Cower.Data, являющийся слоем доступа к данным. Содержит в себе репозитории и Dal объекты (репозитории представлены на рисунках Рисунок 13 и Рисунок 14).
* Проект Cower.Service, являющийся слоем бизнес логики. В нем находятся сервисы, которые используются слой доступа к данным, и вызывается из слоя презентации (сервисы представлены на рисунках Рисунок 15 и Рисунок 16).
* Проект Cower.Web, являющйися слоем презентации. Внутри него находятся REST контроллеры, которые используются клиентской частью.

Настройки приложение берет из переменных среды при запуске.

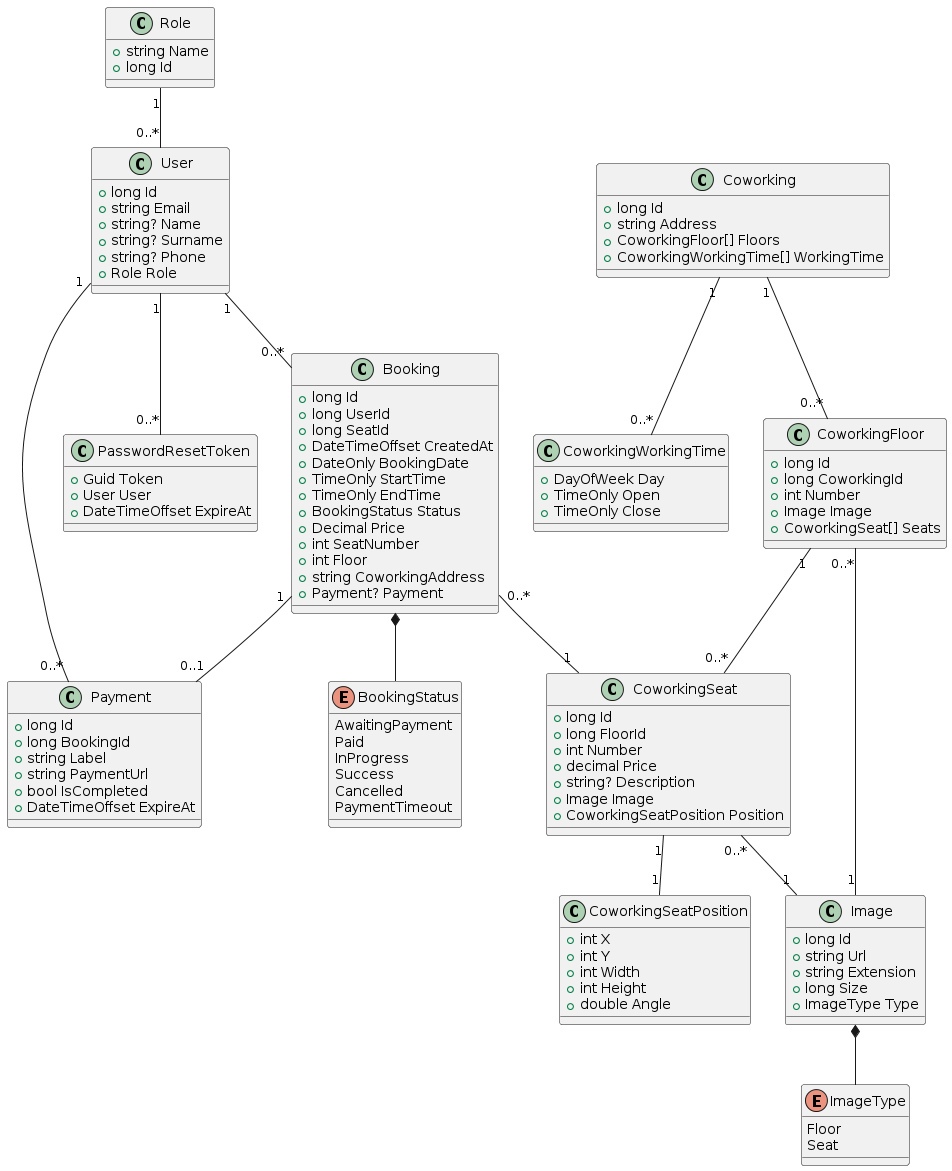


Рисунок 12 – Диаграмма классов доменной модели

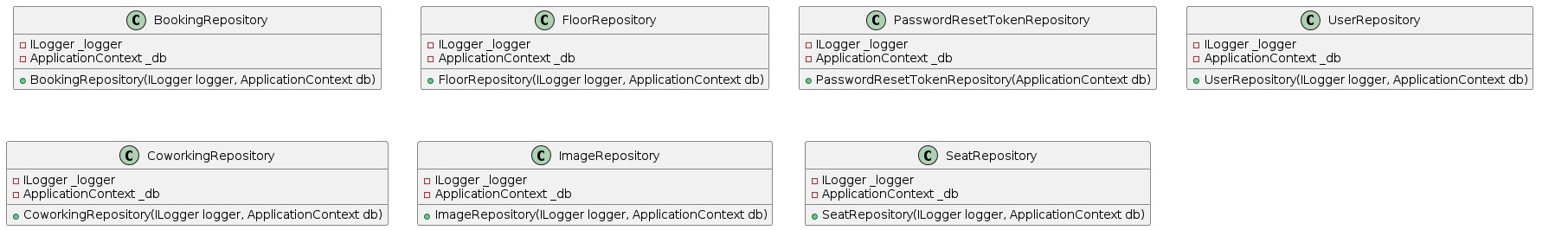


Рисунок 13 – Репозитории, часть 1

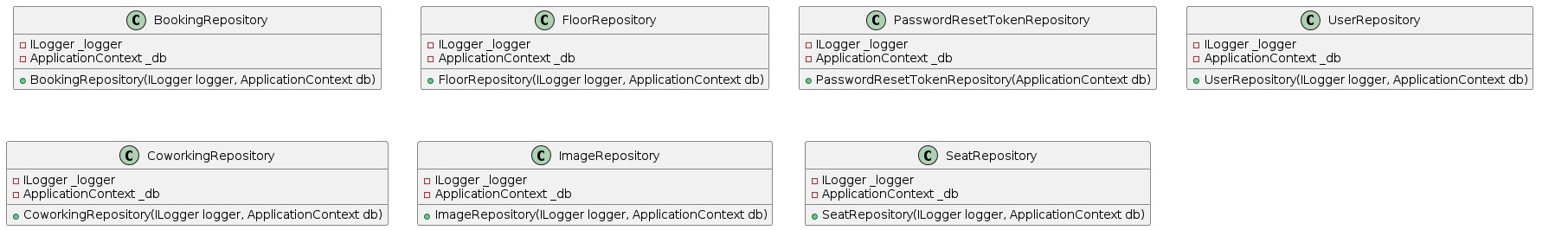


Рисунок 14 – Репозитории, часть 2

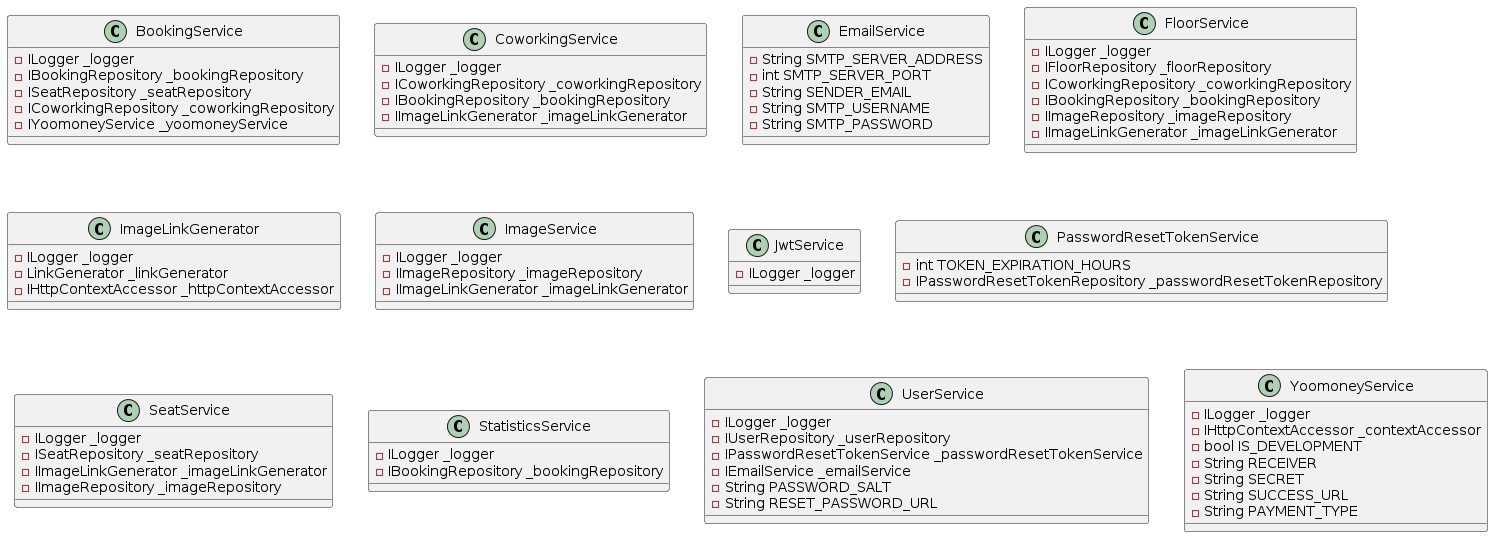


Рисунок 15 – Сервисы, часть 1

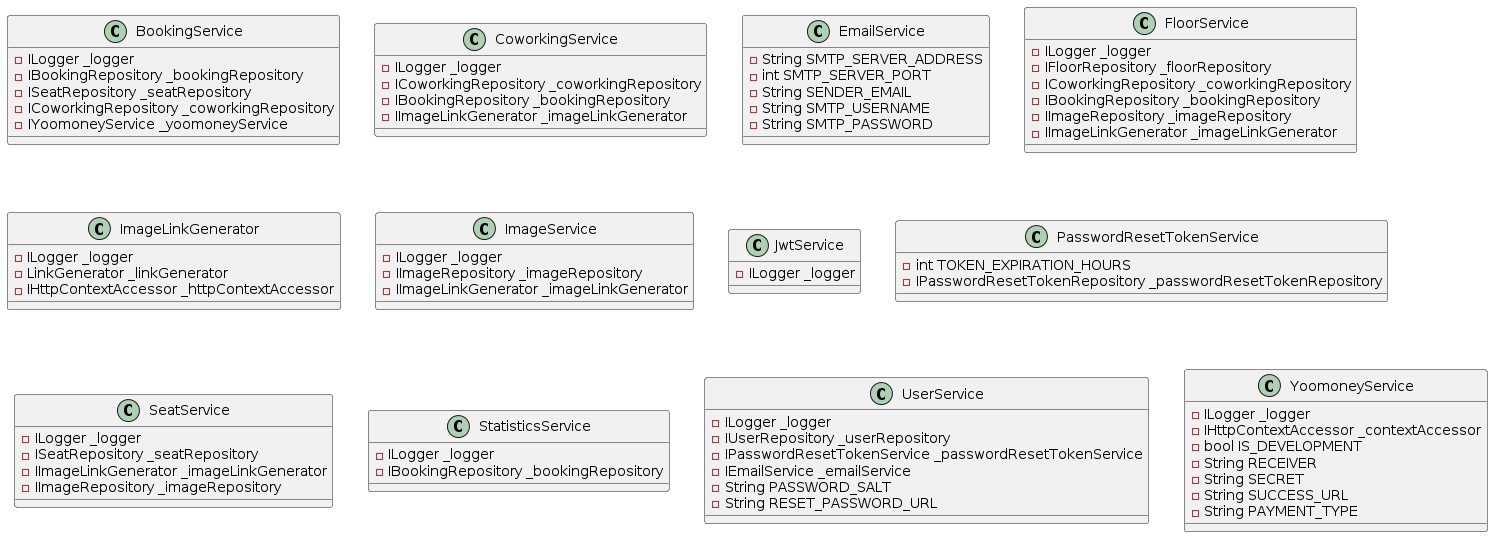


Рисунок 16 – Сервисы, часть 2

## Реализация клиентской части приложения

В данном разделе содержится описание реализации клиентской части приложения.

### Общая информация

Клиентская часть приложения представляет собой веб-сайт, написанный при помощи языка TypeScript с использованием библиотеки ReactJS и инструмента сборки проекта Vite.js. Пользовательский интерфейс формируется из так называемых компонентов - отдельных логически обособленных частей, которые содержат в себе стили для отображения на странице и логику поведения (примеры компонентов: модальное окно, форма регистрации, навигационная панель). Для стилизации элементов интерфейса использовалась библиотека StyledComponents. Для управления состоянием приложения использовалась библиотека Redux. Для клиентской навигации в приложении использовалась библиотека React Router. Для анализа и приведения javascript кода к единому стилю использовался программный инструмент ESLint.

### Графический интерфейс

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

На рисунке Рисунок 17 представлена страница авторизации. Пользователь может ввести почту и пароль, чтобы войти в аккаунт. При нажатии на кнопку «Регистрация» пользователь переходит на страницу регистрации.

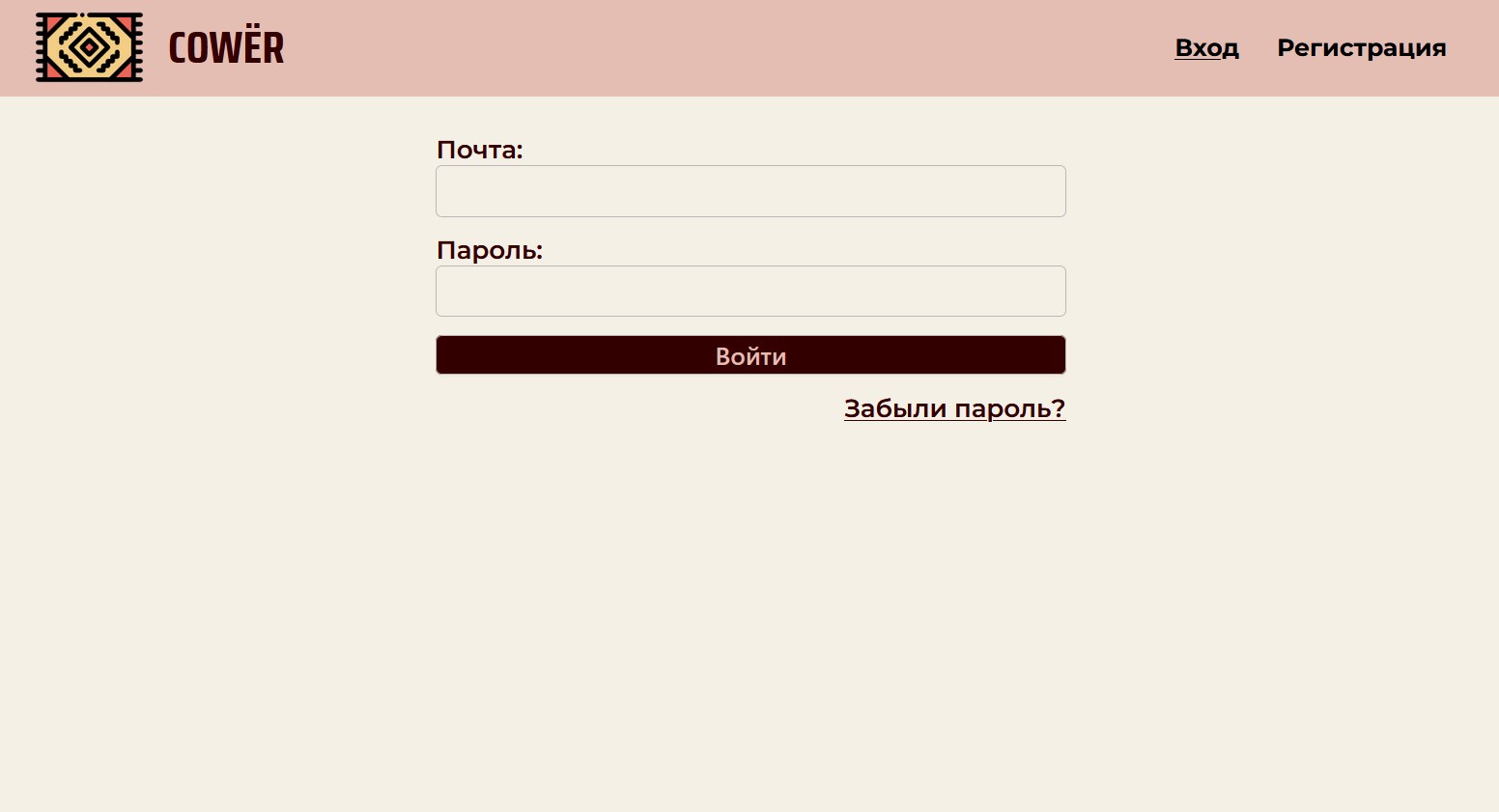


Рисунок 17 – Страница авторизации

На рисунке Рисунок 18 представлена страница регистрации. Пользователь может ввести почту и пароль (обязательно), имя, фамилию, номер телефона (опционально) и нажать на кнопку «Зарегистрироваться», чтобы зарегистрироваться.

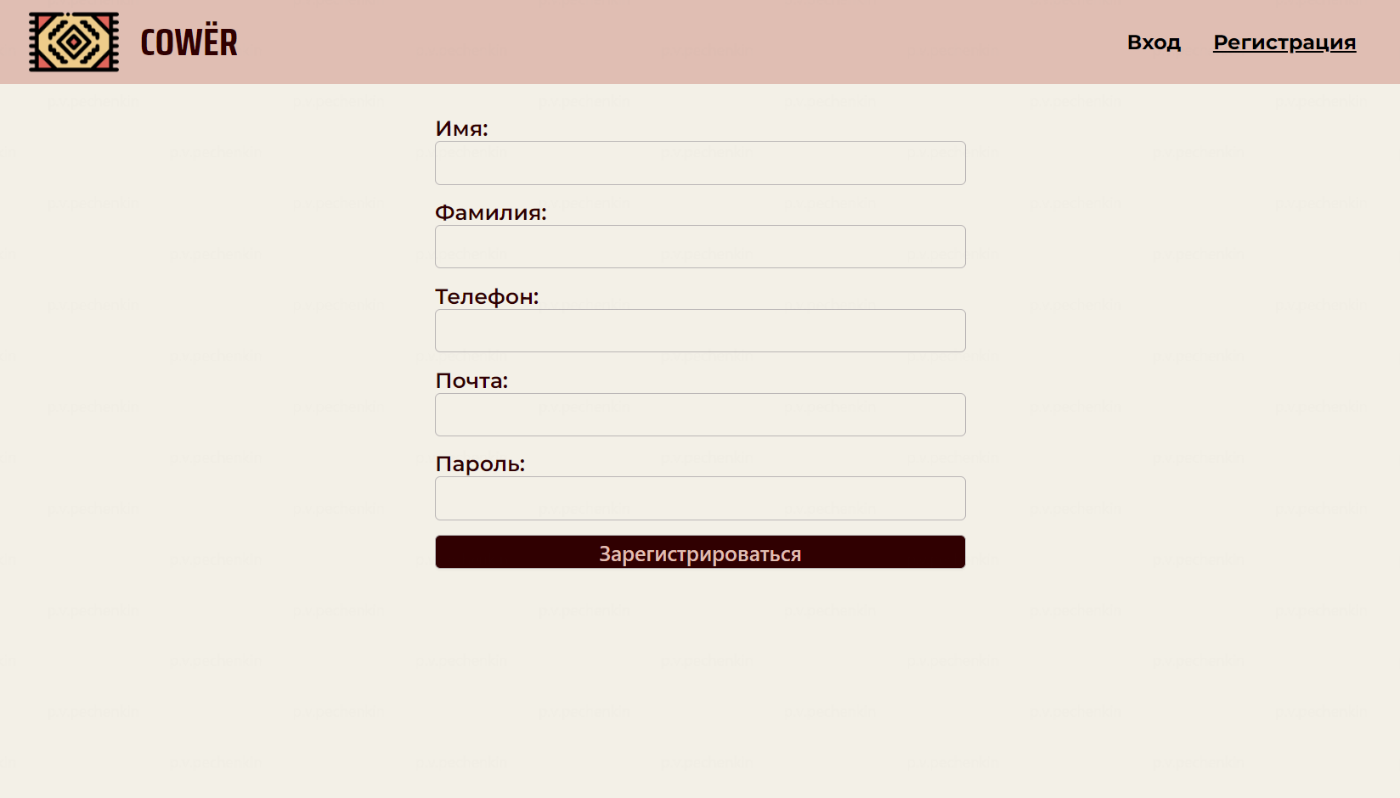


Рисунок 18 – Страница регистрации

На рисунке Рисунок 20 представлена страница личного кабинета. После нажатия на кнопку «Редактировать» пользователь может ввести имя, фамилию, номер телефона и электронную почту в соответствующие поля, чтобы изменить данные своего личного кабинета (рисунок Рисунок 19). Также пользователь может нажать на кнопку «Сбросить пароль», чтобы изменить пароль от своего личного кабинета.

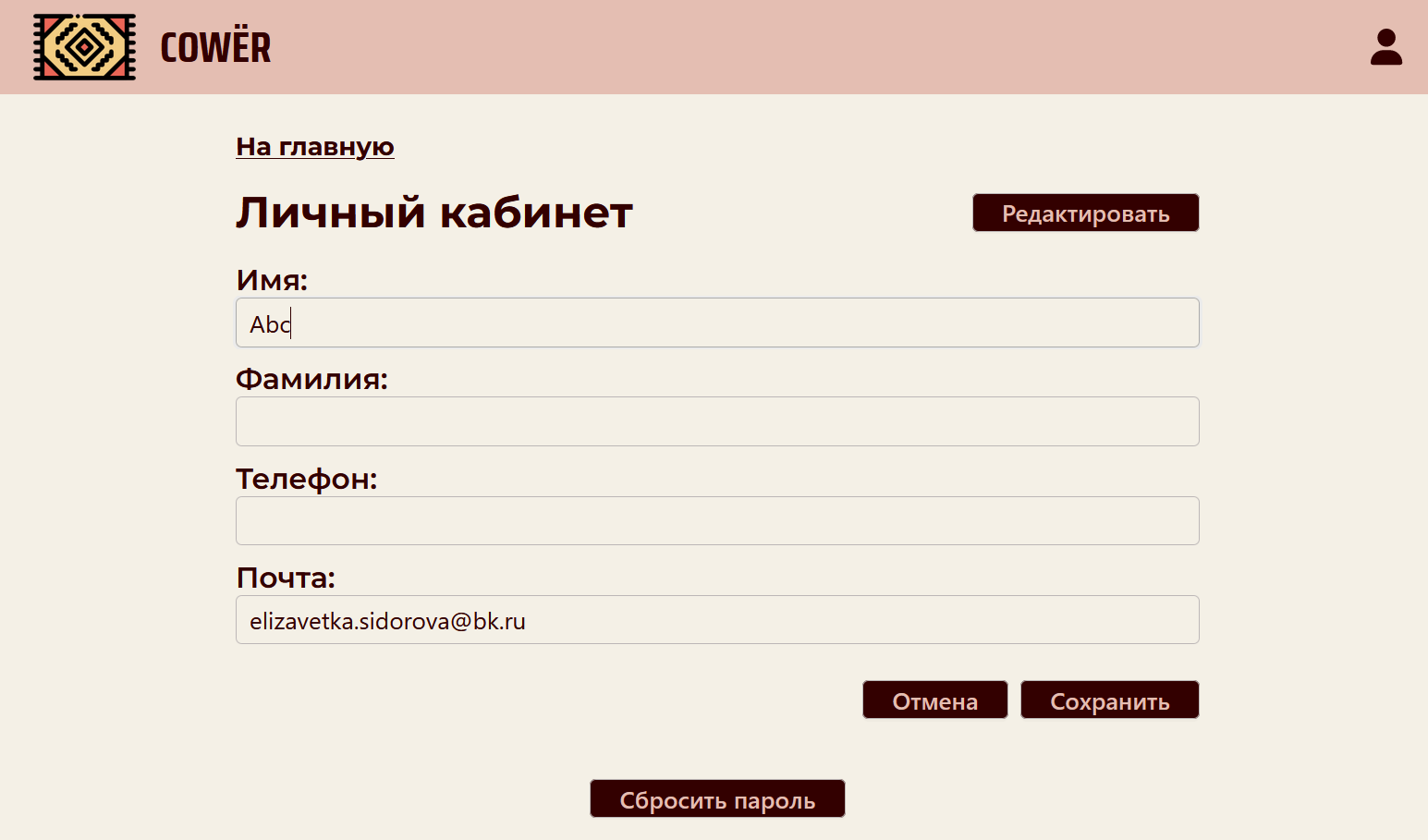
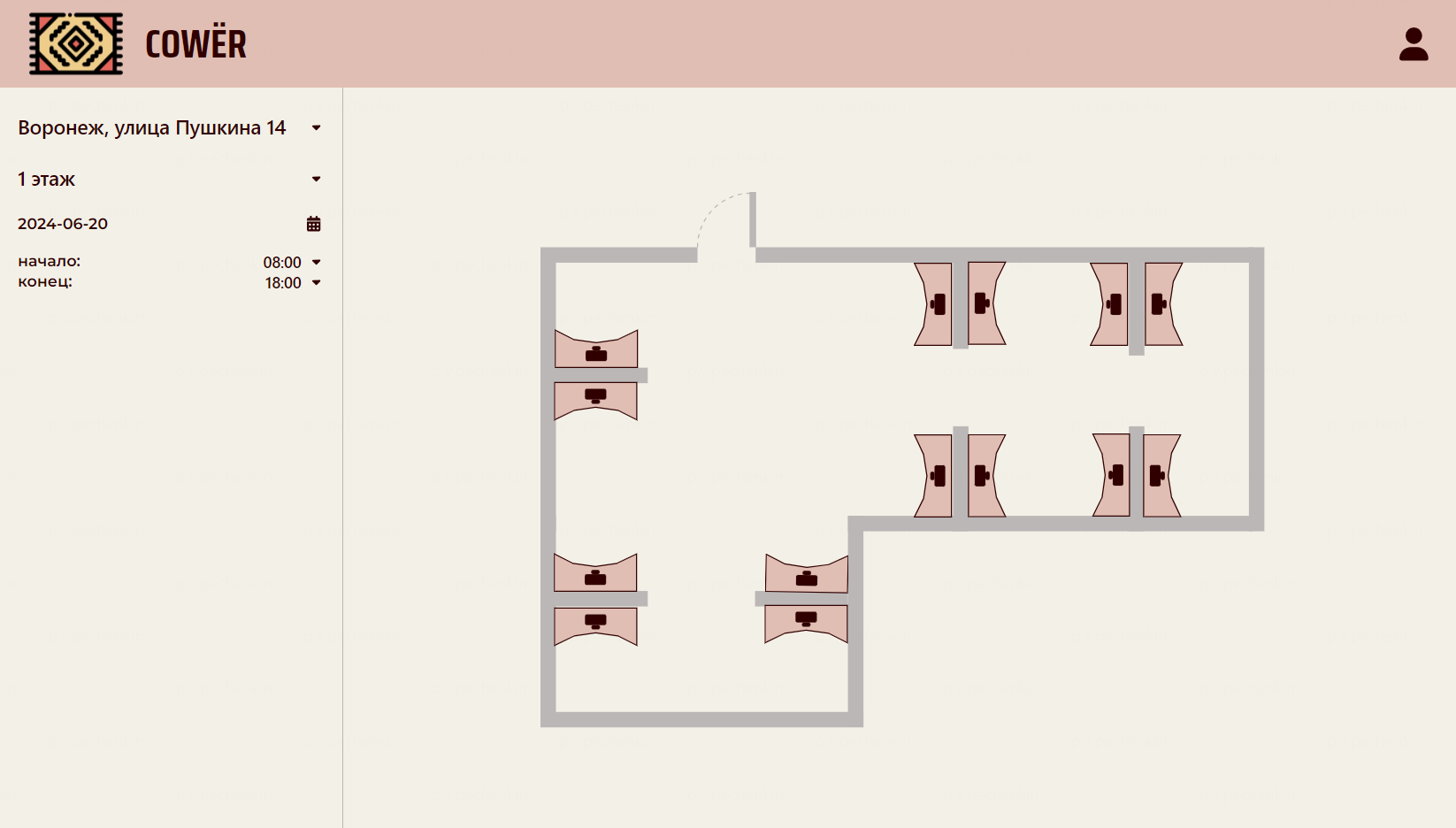


Рисунок 19 – Страница редактирования профиля

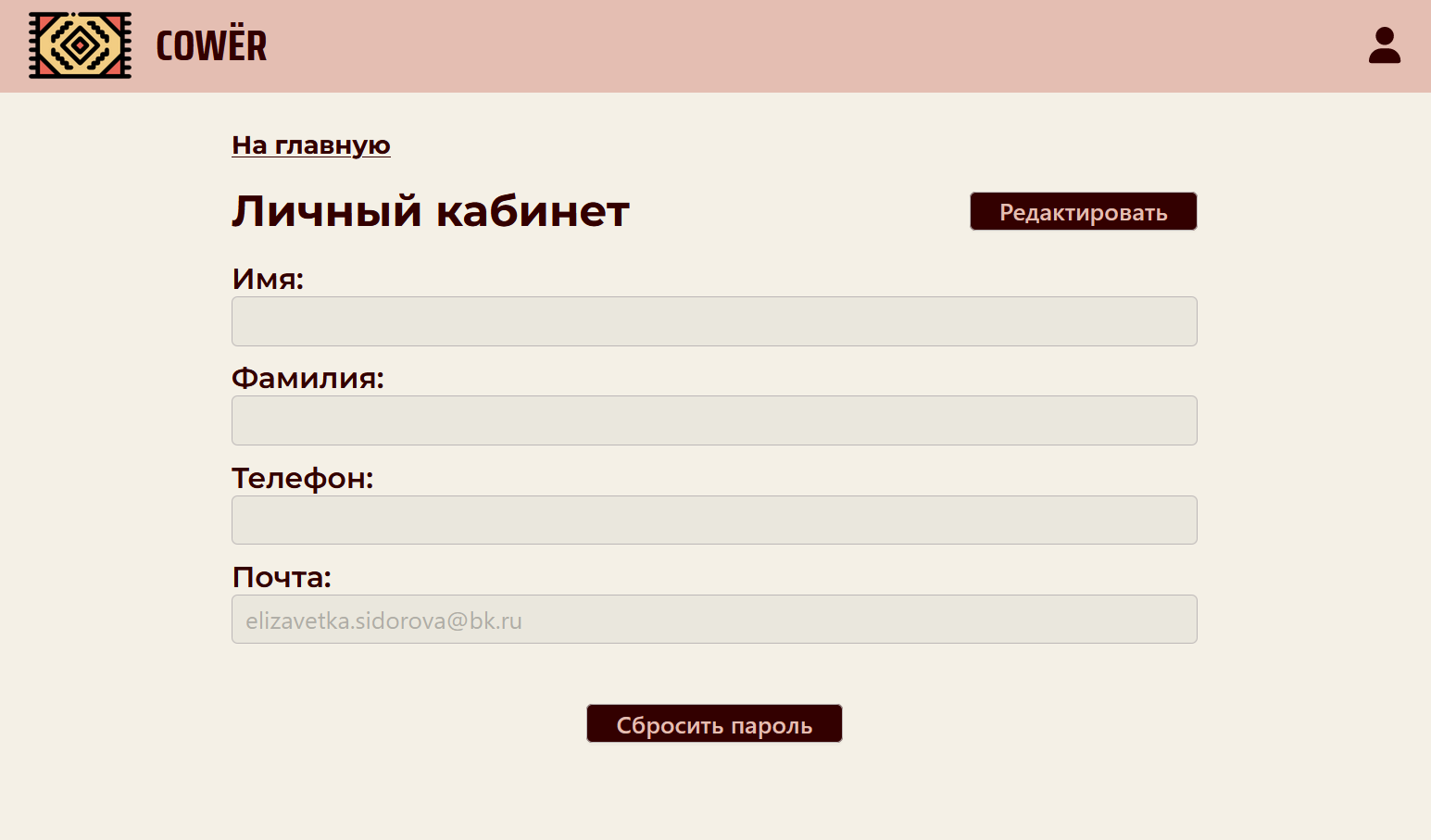


Рисунок 20 – Страница личного кабинета

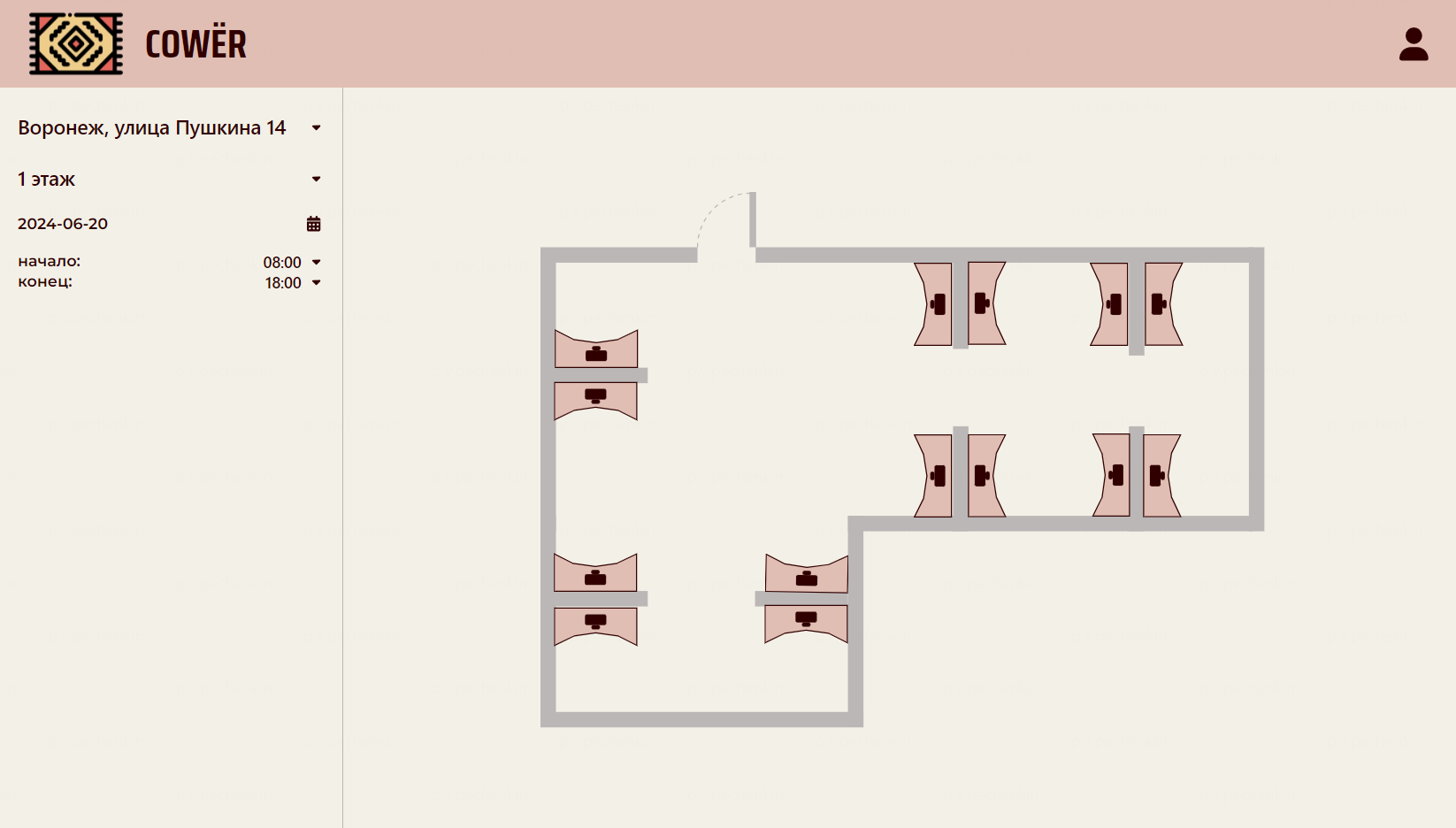
На рисунке 

Рисунок 21 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.

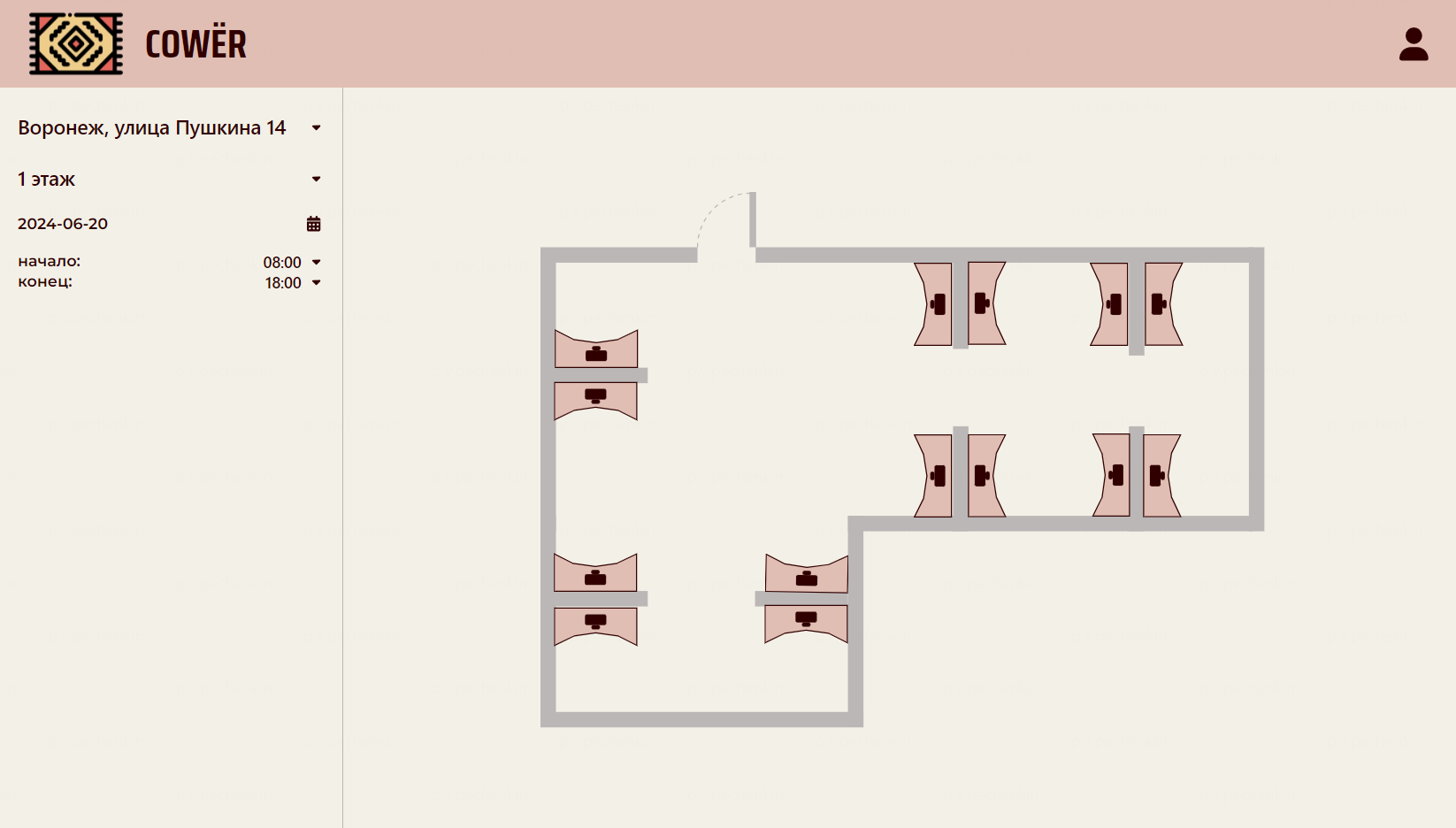


Рисунок 21 – Главная страница

На рисунке Рисунок 22 представлена страница чекаута с оплатой, на которую пользователь переходит после нажатия на кнопку «Забронировать» на главной странице.

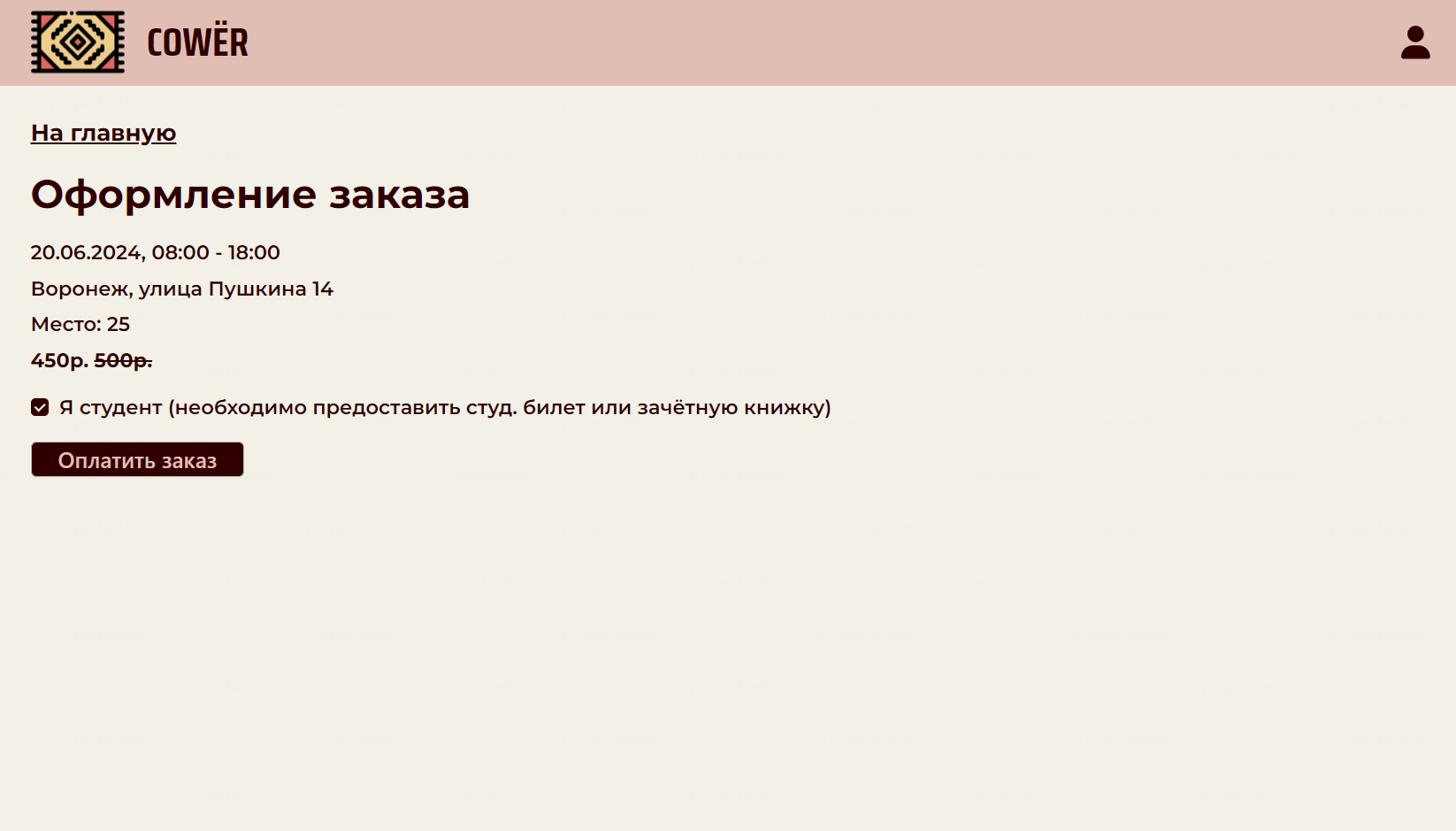


Рисунок 22 – Чекаут с оплатой

На рисунке Рисунок 23 представлена страница со списком заказов. Пользователь может оплатить неоплаченный заказ или отменить заказ, если время, на которое совершена бронь, еще не наступило.

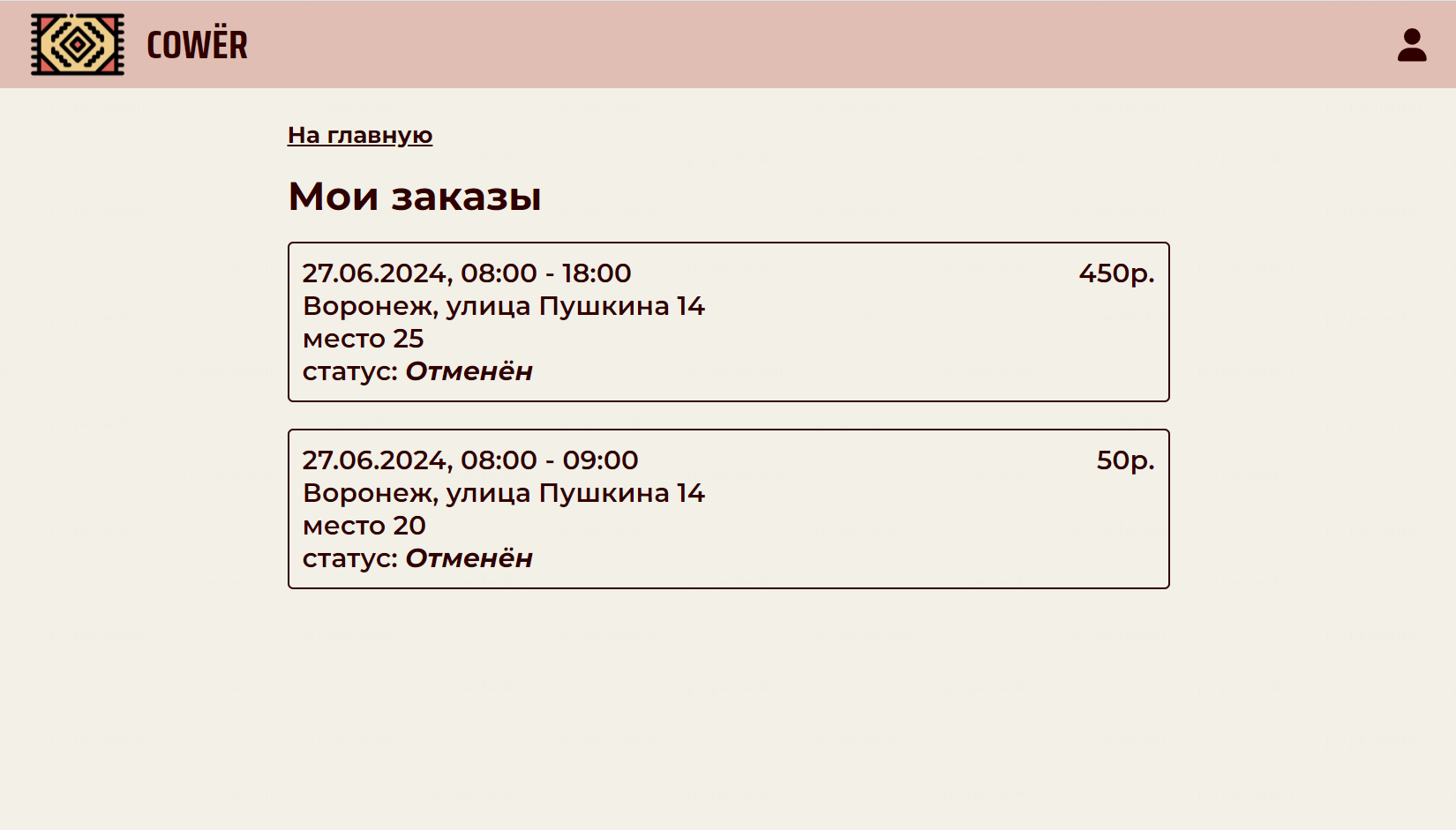


Рисунок 23 – Страница со списком заказов

На рисунке Рисунок 24 представлена страница запроса на восстановление пароля. Пользователь может ввести электронную почту и нажать на кнопку «Отправить письмо на почту», после чего пользователю на электронную почту придёт письмо со ссылкой, перейдя по которой пользователь попадет на страницу восстановления пароля.

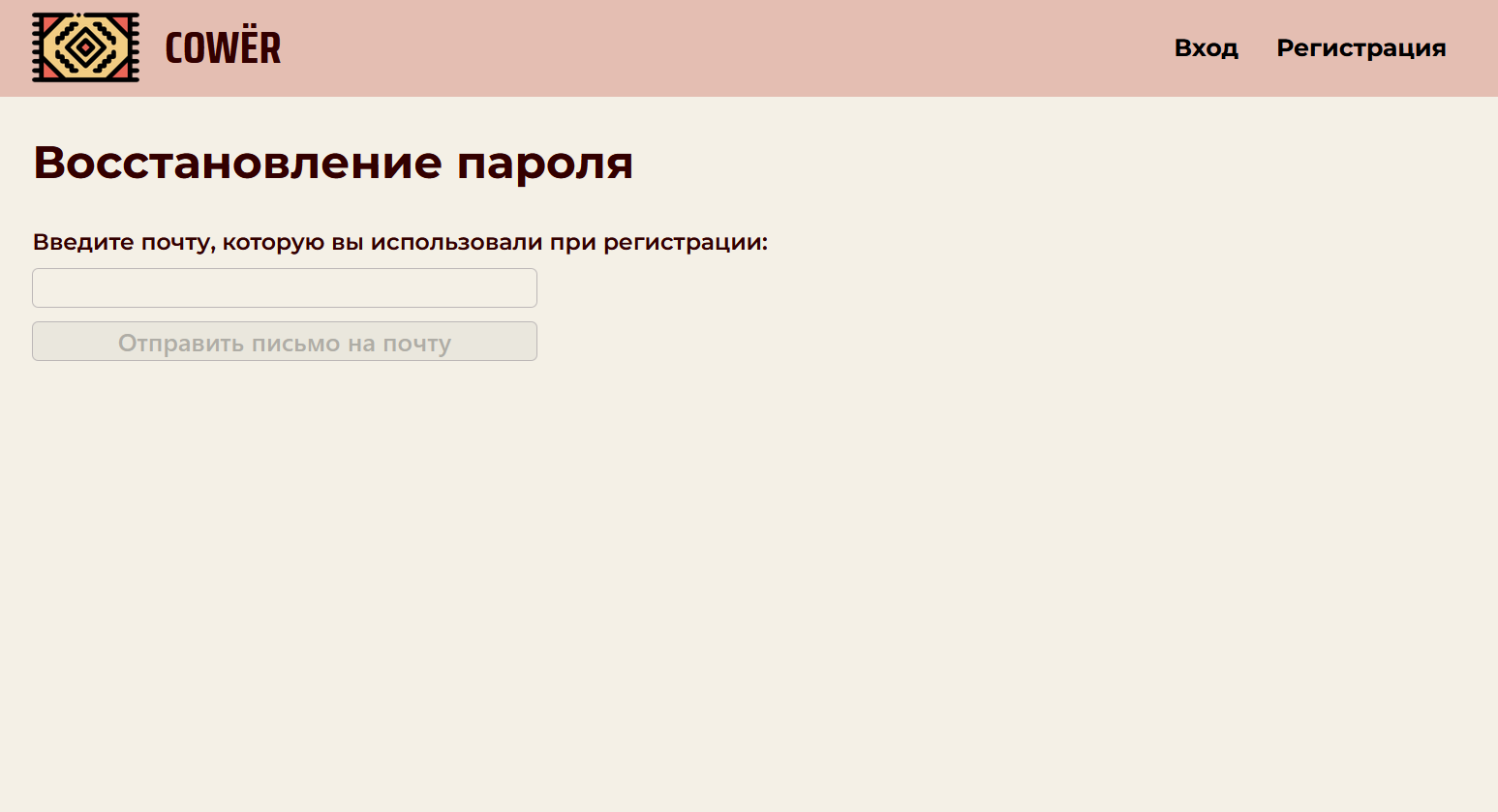


Рисунок 24 – Страница запроса на восстановление пароля

На рисунке Рисунок 25 представлена страница администратора для редактирования планировки офиса. Администратор может добавить, удалить, переместить рабочее место, изменить его описание. Также администратор может добавлять и удалять коворкинги и этажи в коворкингах.

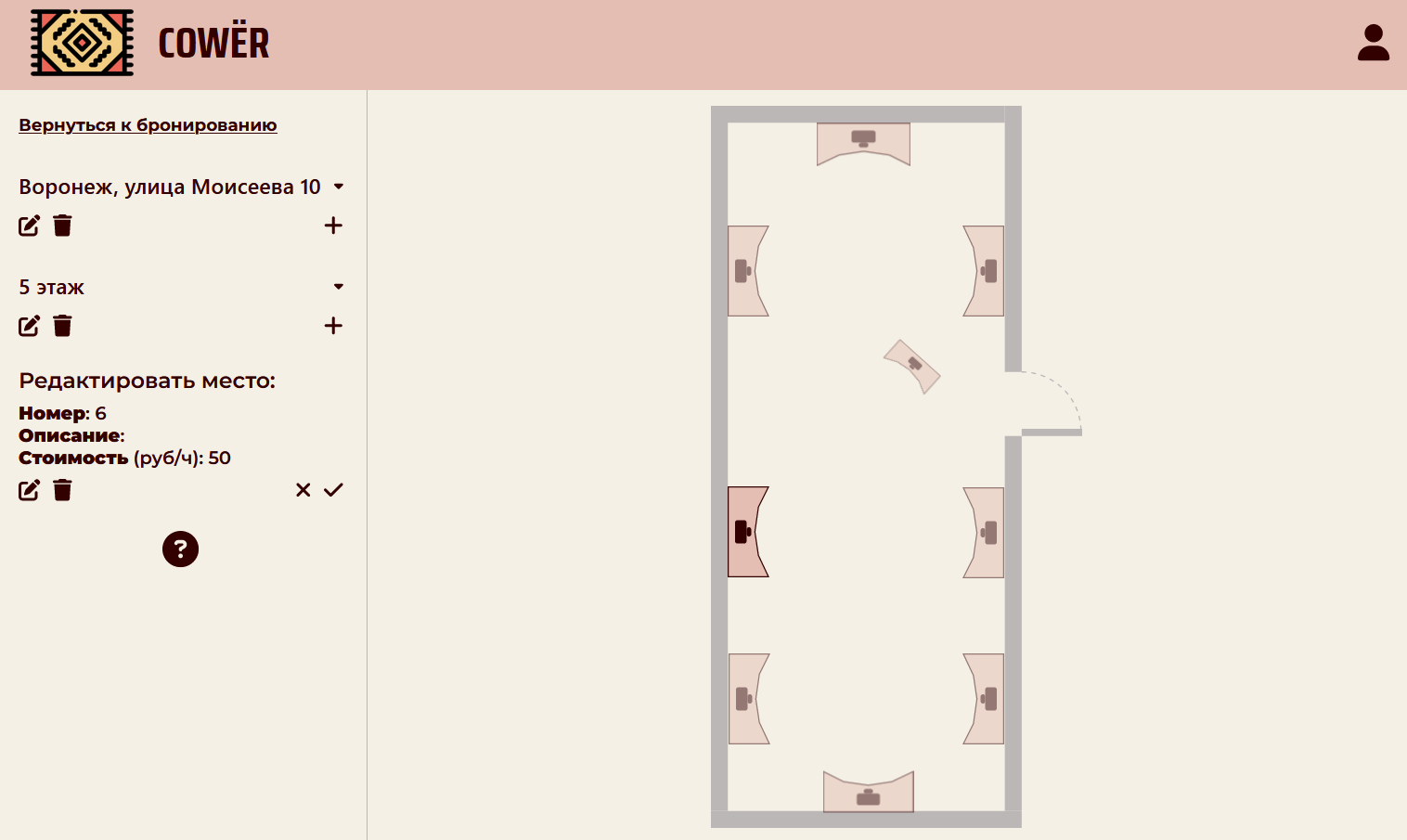


Рисунок 25 – Страница администратора для редактирования планировки офиса

# Тестирование

Перечень проводимых испытаний и их результаты приведены в документах «Программа и методика испытаний» и «Протокол проведения испытаний».

# **ЗАКЛЮЧЕНИЕ**

В ходе реализации данного проекта были достигнуты поставленные задачи. Был разработан веб-сервис «Cowёr», 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

1. Исследуем, как растет рынок коворкинга в России [Электронный ресурс]. – Режим доступа: https://bankspace.ru/blog/koworking\_research (Дата обращения: 20.03.2024).
2. WORKKODE [Электронный ресурс] – Режим доступа: https://wkode.co/ (Дата обращения: 29.04.2024).
3. GrowUp [Электронный ресурс] – Режим доступа: https://growup-coworking.ru/ (Дата обращения: 20.03.2023).
4. Ясная поляна [Электронный ресурс] – Режим доступа: https://coworkingspb.ru/coworking#openspaces (Дата обращения: 20.03.2023).