

Министерство образования и науки Республики Башкортостан
Государственное автономное профессиональное образовательное
учреждение

Уфимский колледж статистики, информатики и вычислительной техники

ОТЧЁТ ПО УЧЕБНОЙ ПРАКТИКЕ 04.01

Руководитель проекта

_____ О.В.Фатхуллова
«___» _____ 2025 г.

Студенты гр. 22П-2

_____ А.И.Файзуллин,
_____ А.И.Захаренко
«___» _____ 2025 г.

СОДЕРЖАНИЕ

	лист
Введение	3
1 Техническое задание	4
1.1 Назначение разработки	4
1.2 Требования к программе	4
1.3 Требования к функциональным характеристикам	5
1.4 Стадии и этапы разработки	6
1.5 Описание входной информации	7
1.6 Описание выходной информации	7
2 Разработка прототипа	11
3 Структура проекта	15
4 Описание запросов API	16
5 Результаты работы	33
Заключение	45

ВВЕДЕНИЕ

Настоящий отчет описывает процесс и результаты разработки веб-приложения «Skywhysales» — платформы, предназначенный для автоматизации продажи и покупки авиабилетов.

Актуальность данного проекта обусловлена стремительным ростом объемов воздушных перевозок и цифровизацией туристической отрасли. В современном мире путешествия стали неотъемлемой частью жизни миллионов людей, однако традиционные процессы поиска, бронирования и покупки билетов часто остаются неэффективными: они требуют значительных временных затрат, ручной обработки данных и взаимодействия с несколькими системами. Существующие решения, такие как агрегаторы или сайты авиакомпаний, не всегда обеспечивают полный цикл автоматизации — от управления каталогом рейсов и верификации клиентов до отслеживания продаж и отмен, — что приводит к ошибкам, задержкам и низкой удовлетворенности пользователей.

Целью проекта являлось создание полнофункционального веб-приложения, которое решает эти проблемы, предоставляя удобную и безопасную среду для взаимодействия менеджеров и клиентов: от регистрации и поиска рейсов до оформления билетов с учетом ограничений на данные и ролевого доступа.

В данном отчёте подробно рассматриваются реализованная архитектура проекта, ключевые функции готового продукта, использованные технологии, а также выводы о проделанной работе и потенциальных направлениях для дальнейшего развития платформы «Skywhysales».

1 Техническое задание

1.1 Назначение разработки

Разработка веб-приложения «Skywhysales» предназначена автоматизации процессов покупки авиабилетов. Система обеспечивает взаимодействие между менеджерами и клиентами, с учетом ограничений на данные и верификацией пользователей. Основная цель — оптимизация процессов ведения каталога рейсов и авиакомпаний, регистрации клиентов, поиска и покупки билетов, а также управления данными о продажах и отменах.

1.2 Требования к программе

Для реализации веб-приложения «Skywhysales» был применён следующий технологический стек и дизайн-решения:

1.2.1 Технологический стек:

- Frontend: Vue.js;
- Backend: C# ASP.NET Core, Entity Framework Core;
- база данных: PostgreSQL;
-

1.2.2 дизайн-решения и интерфейс:

- реализована светлая тема с акцентным цветом #605dec и текстовыми оттенками #a1b0cc;
- применён шрифт «Nunito-Sans» в соответствии с утвержденной иерархией;
- адаптивный интерфейс с минималистичным дизайном.

1.3 Требования к функциональным характеристикам

Разработанное веб-приложение «Skywhysales» реализует следующий набор функциональных характеристик, структурированных по ролям пользователей:

1.3.1 функциональные требования для неавторизованных пользователей;

1.3.1.1 просмотр перелетов;

1.3.1.2 поиск перелетов с применением фильтров по:

- города отправления;
- города прибытия;
- диапазона дат, в которые пользователь готов вылететь;

1.3.1.3 регистрация новой учетной записи в системе с использованием адреса электронной почты, имени, пароля;

1.3.2 функциональные требования для авторизованных пользователей (клиент);

1.3.2.1 требования к системе аккаунтов;

- авторизация в системе по адресу электронной почты и паролю;
- просмотр и редактирование информации в личном профиле пользователя.

- покупка авиабилета;
- отмена ранее купленного билета, если класс обслуживания позволяет.

1.3.3 функциональные требования для администратора системы;

1.3.3.1 управление справочной информацией;

- добавление новых рейсов;
- редактирование рейсов и атрибутов, существующих в каталоге.

Таким образом, в разработанном продукте полностью реализован весь

запланированный функционал, обеспечивающий удобный и полный цикл взаимодействия пользователя с платформой от поиска события до его организации и администрирования.

1.4 Стадии и этапы разработки

План-график по разработке веб-приложения «Skywhysales» составлен в соответствии с техническим заданием и устанавливает последовательность, сроки и содержание всех этапов проекта. Пример представлен в таблице 1.4.1.

Таблица 1.4.1 – Этапы разработки

Название этапа	Сроки выполнения	Отчётность
Анализ и проектирование	17.09 - 24.09	Предметная область Схема взаимодействия ПО Use-Case Диаграмма классов Входная и выходная информация с шаблоном

Продолжение таблицы 1.4.1

1	2	3
Проектирование базы данных	25.09 - 29.09	База данных Описание структуры БД
Создание прототипа	07.10 - 09.10	Прототип
Разработка API	10.10 - 25.10	API, документация API
Тестирование	26.10 - 27.10	Протоколы тестирования на вход в систему Модульные тесты
Финальное проектирование	02.11 - 03.11	Диаграмма развертывания Техническое задание (функционал, интерфейс)
Подготовка к тестированию	04.11 - 05.11	Контрольные примеры для тестирования проекта
Функциональное тестирование	06.11 - 15.11	Тестирование проектов (функциональное, юзабилити) Чек-лист Тест-кейс
Документирование	16.11 - 18.11	Руководство пользователя по проекту Отчет по проекту

1.5 Описание входной информации

Входные данные включают информацию о рейсах, пользователях, бронировании билетов и учетных данных для авторизации.

1.6 Описание выходной информации

В процессе работы системы выходной информацией будут являться отклонение записи на событие, отчёты о проведённых мероприятиях, отчёты

об активности пользователей. Описание выходных документов представлено в таблице 1.6.1.

Таблица 1.6.1 – Описание выходных документов

Наименование документа (шифр)	Периодичность выдачи документа	Кол-во экз.	Куда передаются
Уведомления по электронной почте	При регистрации	1	На электронную почту пользователя
Билет	При продаже билета	1	На электронную почту клиента

На рисунке 1.6.2 представлен шаблон выходного документа.

skywhysales

Электронный билет

Номер заказа: Номер_заказа
Класс обслуживания: Класс_обслуживания
Электронный билет: Электронный_билет

Пассажир: Пассажир

Место_вылета — Место_прилета

Место_вылета (Код_вылета)	Время_вылета	Номер_рейса	Самолет	Время_прилета	Место_прилета (Код_прилета)
Аэропорт_вылета	Дата_вылета	Продолжительность	Авиакомпания	Дата_прилета	Терминал

Номер билета для регистрации: Номер_билета
Всего в пути: Общее_время
Дата покупки билета: Дата_покупки_билета
Стоимость: Стоимость

Рисунок 1.6.2 – Шаблон выходного документа

2 Разработка прототипа

Веб-приложение состоит из нескольких страниц:

– «Главная». На странице находится форма параметрами поиска рейсов и навигация в шапке. На рисунке 2.1 показан интерфейс страницы. При нажатии на кнопку «Войти» открывается модальное окно и авторизацией. На рисунке 2.2 отображён внешний вид форм.

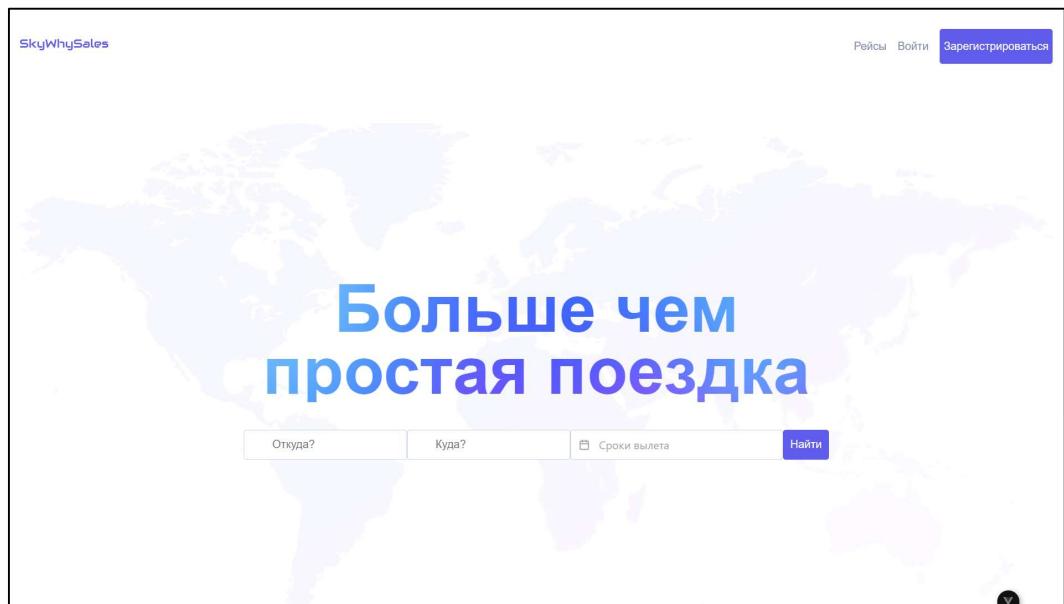


Рисунок 2.1 – Главная

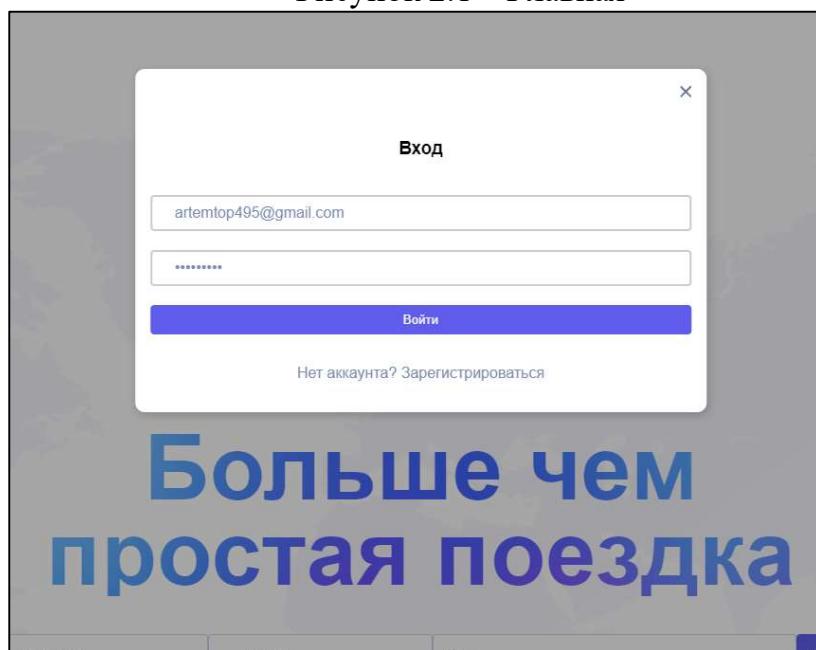


Рисунок 2.2 – Модальное окно с авторизацией

– «Рейсы». Страница с выводом всех доступных рейсов. На рисунке 2.4 представлены все доступные билеты. При нажатии на рейс появляется окно деталей рейса. На рисунке 2.5 представлено окно деталей рейса.

The screenshot shows a search interface for flights. At the top, there are input fields for 'Откуда?' (From), 'Куда?' (To), and 'Сроки вылета' (Flight dates), along with a 'Найти' (Search) button. Below this, a section titled 'Выберите рейс вылета' (Select flight departure) displays a list of seven flight options:

Авиакомпания	Время полета	Путь	Дата вылета	Цена
S7 Airlines	1ч 30 мин	Пулково - Домодедово	01.12.2025, 11:00 - 01.12.2025, 12:30	8200 ₽
Победа	2ч 30 мин	Домодедово - Сочи	02.12.2025, 14:20 - 02.12.2025, 16:50	7200 ₽
S7 Airlines	3ч 25 мин	Домодедово - Кольцово	03.12.2025, 07:15 - 03.12.2025, 10:40	12500 ₽
Utair	3ч 40 мин	Толмачёво - Домодедово	04.12.2025, 09:30 - 04.12.2025, 13:10	14000 ₽
Россия	2ч 20 мин	Сочи - Домодедово	05.12.2025, 18:00 - 05.12.2025, 20:20	6800 ₽
Nordwind	2ч 30 мин	Домодедово - Симферополь	06.12.2025, 10:10 - 06.12.2025, 12:40	9500 ₽
	2ч 10 мин	Казань - Домодедово		

Рисунок 2.4 – Страница всех рейсов

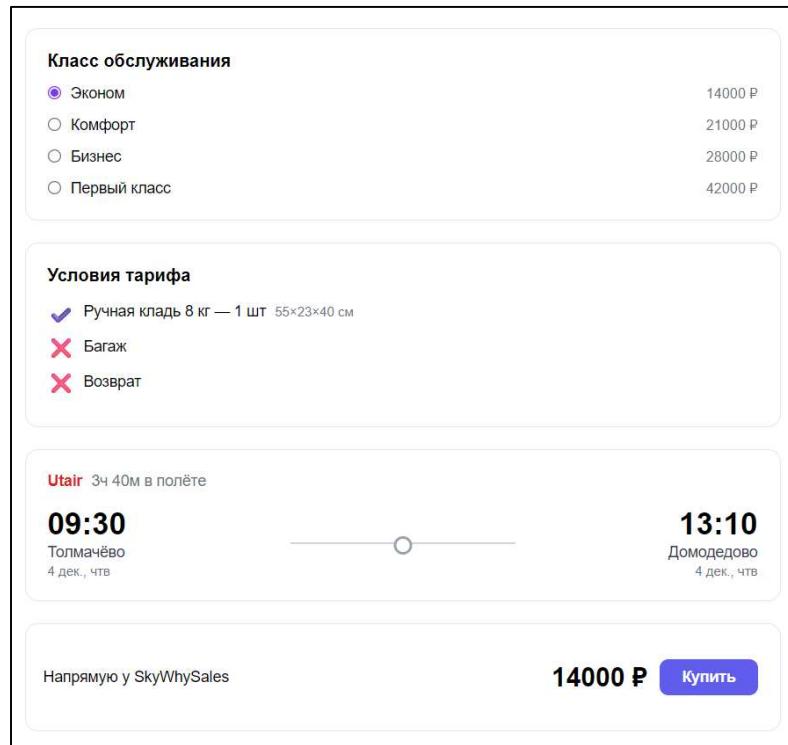


Рисунок 2.5 – Страница деталей рейса

– «Мой профиль». На странице отображается личная информация пользователя с возможностью изменений. На рисунке 2.6 приведен пример интерфейса.

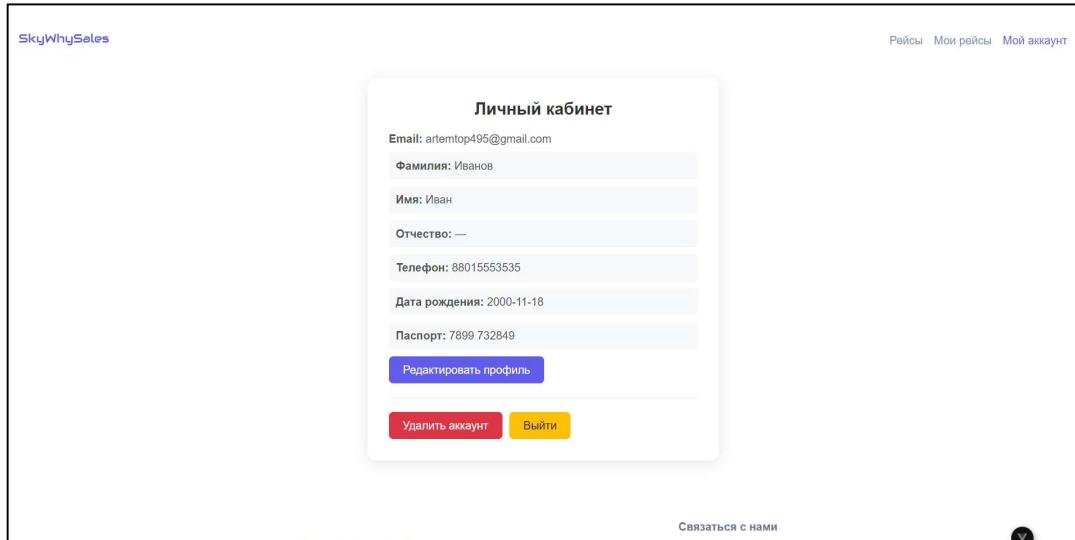


Рисунок 2.6 – Профиль

– «Админ-панель». В разделе находится шесть вкладок – краткий обзор, управление рейсами, пользователями, авиакомпаниями, аэропортами, просмотр купленных билетов. На рисунках 2.7-2.9 представлены примеры.

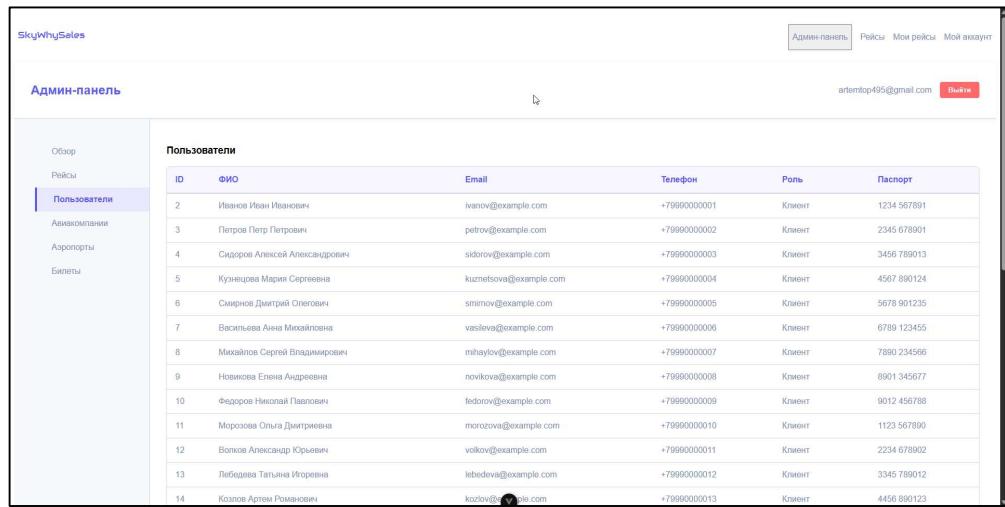
The screenshot shows the 'Admin-panel' section of the SkyWhySales application. On the left, a sidebar lists 'Обзор', 'Рейсы', 'Пользователи', 'Авиакомпании', 'Аэропорты', and 'Билеты'. The main area displays a welcome message 'Добро пожаловать, zxc!' and five summary boxes with counts: 20 Рейсов, 24 Пользователей, 22 Авиакомпаний, 21 Аэропортов, and 32 Билетов продано.

Рисунок 2.7 – Обзор

The screenshot shows the 'Рейсы' (Flights) section of the Admin Panel. The sidebar has 'Рейсы' selected. The main area shows a table of flight records with columns: ID, Откуда --> Куда, Вылет, Прилёт, Авиакомпания, Цена, Мест, and Действия. The table contains 11 entries, each with an 'Изменить' (Edit) button. The first entry is '2 Пулково --> String 01.12.2025, 11:00 12:30 S7 Airlines 8200 ₽ 150 Изменить'.

ID	Откуда --> Куда	Вылет	Прилёт	Авиакомпания	Цена	Мест	Действия
2	Пулково --> String	01.12.2025, 11:00	12:30	S7 Airlines	8200 ₽	150	<button>Изменить</button>
3	Домодедово --> Сочи	02.12.2025, 14:20	16:50	Победа	7200 ₽	189	<button>Изменить</button>
4	String --> Кольцово	03.12.2025, 07:15	10:40	string	12500 ₽	200	<button>Изменить</button>
5	Толмачёво --> String	04.12.2025, 09:30	13:10	Utair	14000 ₽	160	<button>Изменить</button>
6	Сочи --> Домодедово	05.12.2025, 18:00	20:20	Россия	6800 ₽	170	<button>Изменить</button>
7	String --> Симферополь	06.12.2025, 10:10	12:40	Nordwind	9500 ₽	180	<button>Изменить</button>
8	Казань --> String	07.12.2025, 06:45	08:55	Smartavia	7800 ₽	140	<button>Изменить</button>
9	Уфа --> Пулково	08.12.2025, 12:00	14:30	Azimuth	8800 ₽	150	<button>Изменить</button>
10	Самара --> String	09.12.2025, 15:25	17:50	Ямал	9200 ₽	160	<button>Изменить</button>
11	String --> Ростов-на-Дону	10.12.2025, 09:00	11:20	Red Wings	8400 ₽	170	<button>Изменить</button>

Рисунок 2.8 – Управление рейсами



The screenshot shows the 'Admin-panel' section of the SkyWingsSales application. On the left, there's a sidebar with navigation links: 'Обзор', 'Рейсы', 'Пользователи' (selected), 'Авиакомпании', 'Аэропорты', and 'Билеты'. The main area is titled 'Пользователи' and contains a table with 14 rows of user data. The columns are: ID, ФИО, Email, Телефон, Роль, and Паспорт. The data is as follows:

ID	ФИО	Email	Телефон	Роль	Паспорт
2	Иванов Иван Иванович	ivanov@example.com	+79990000001	Клиент	1234 567891
3	Петров Петр Петрович	petrov@example.com	+79990000002	Клиент	2345 678901
4	Сидоров Алексей Александрович	sidorov@example.com	+79990000003	Клиент	3456 789013
5	Кузнецова Мария Сергеевна	kuznetsova@example.com	+79990000004	Клиент	4567 890124
6	Смирнов Дмитрий Олегович	smirnov@example.com	+79990000005	Клиент	5678 901235
7	Васильева Анна Михайловна	vasilieva@example.com	+79990000006	Клиент	6789 123455
8	Михайлов Сергей Владимирович	mihaylov@example.com	+79990000007	Клиент	7890 234566
9	Новикова Елена Андреевна	novikova@example.com	+79990000008	Клиент	8901 345677
10	Федоров Николай Павлович	fedorov@example.com	+79990000009	Клиент	9012 456788
11	Морозова Ольга Дмитриевна	morozova@example.com	+79990000010	Клиент	1123 567890
12	Волков Александр Юрьевич	volkov@example.com	+79990000011	Клиент	2234 678902
13	Лебедева Татьяна Игоревна	lebedeva@example.com	+79990000012	Клиент	3345 789012
14	Козлов Артем Романович	kozlov@yandex.ru	+79990000013	Клиент	4456 890123

Рисунок 2.9 – Управление пользователями

3 Структура проекта

Проект имеет классическую клиент-серверную архитектуру, разделённую на два основных модуля: фронтенд и бэкенд.

- фронтенд реализован на фреймворке Vue.js и отвечает за пользовательский интерфейс, отображение данных и взаимодействие с пользователем;
- бэкенд построен на платформе ASP.NET Core и предоставляет REST API для обработки бизнес-логики, работы с базой данных и аутентификации.

Связь между модулями осуществляется через HTTP-запросы к заранее определенным endpoint API. На рисунке 4.1 представлена модульная схема проекта.

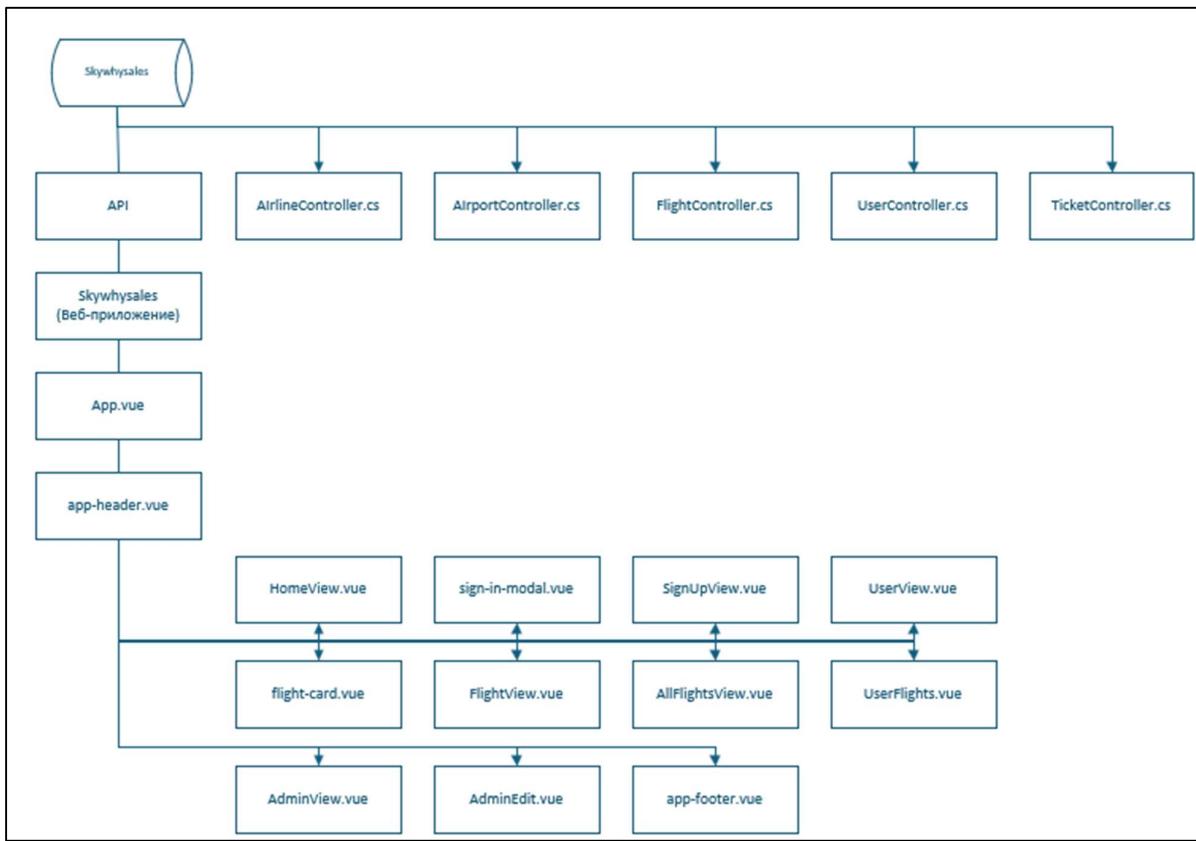


Рисунок 3.1 – Модульная схема

4 Описание запросов API

Разработанное API обеспечивает полный цикл взаимодействия с системой через эндпоинты. Взаимодействие построено вокруг основных сущностей: рейсов, билетов, пользователей и администрировании. Основные контроллеры описаны в таблице 4.1.

Таблица 4.1 – Описание запросов API

Метод	Curl (входные данные)	URL (Строка запроса)	Response body (возвраща- емый результа- тат)	Код результата
GetAirlines	curl -X 'GET' \ 'http://localhost:3000/api/ Airline/GetAirlines' -H 'accept: text/plain'	http://localhost:3000/api/ Airline/GetAirlines	[{ "alId": 1, "alName": "string", "alEmail": "string" }]	200

Продолжение таблицы 4.1

1	2	3	4	5
EditAirline	<pre>curl -X 'POST' \ 'http://localhost:3000/api/Airline/EditAirline' \ -H 'accept: text/plain' \ -H 'Content-Type: application/json' \ -d '{ "alId": 1, "alName": "qwe", "alEmail": "qwe" }'</pre>	http://localhost:3000/api/Airline/EditAirline	{ "alId": 1, "alName": "qwe", "alEmail": "qwe" }	200
AddAirline	<pre>curl -X 'POST' \ 'http://localhost:3000/api/Airline/AddAirline' \ -H 'accept: text/plain' \ -H 'Content-Type: application/json' \ -d '{ "alId": 0, "alName": "string", "alEmail": "string" }'</pre>	http://localhost:3000/api/Airline/AddAirline	{ "alId": 1, "alName": "string", "alEmail": "string" }	200

Продолжение таблицы 4.1

1	2	3	4	5
DeleteAirlin e	curl -X 'DELETE' \' http://localhost:3000/api/Airline/DeleteAirline/1 'http://localhost:3000/api/Airline/DeleteAirline/1' \ -H 'accept: */*'	http://localhost:3000/api/Airline/DeleteAirline/1		200
GetAirports	curl -X 'GET' \ 'http://localhost:3000/api/Airport/GetAirports' \ -H 'accept: text/plain'	http://localhost:3000/api/Airport/GetAirports	[{ "apId": 1, "apName": "string", "apCountry": "string", "apCity": "string", "apStreet": "string", "apBuilding": "string" }]	200
AddAirport	curl -X 'POST' \ 'http://localhost:3000/api/Airport/AddAirport' \ -H 'accept: text/plain' \ -H 'Content-Type: application/json' \ -d '{ "apId": 0, "apName": "string", "apCountry": "string", "apCity": "string", "apStreet": "string", "apBuilding": "string" }'	http://localhost:3000/api/Airport/AddAirport	{ "apId": 0, "apName": "string", "apCountry": "string", "apCity": "string", "apStreet": "string", "apBuilding": "string" }	200

Продолжение таблицы 4.1

1	2	3	4	5
GetAirport	curl -X 'GET' \ 'http://localhost:3000/api/Airport/GetAirport/1' \ -H 'accept: text/plain'	http://localhost:3000/api/Airport/GetAirport/1	{ "apId": 1, "apName": "string", "apCountry": "string", "apCity": "string", "apStreet": "string", "apBuilding": "string" }	200
EditAirport	curl -X 'POST' \ 'http://localhost:3000/api/Airport/EditAirport' \ -H 'accept: text/plain' \ -H 'Content-Type: application/json' \ -d '{ "apId": 1, "apName": "1", "apCountry": "1", "apCity": "1", "apStreet": "1", "apBuilding": "1" }'	http://localhost:3000/api/Airport/EditAirport	{ "apId": 1, "apName": "1", "apCountry": "1", "apCity": "1", "apStreet": "1", "apBuilding": "1" }	200

Продолжение таблицы 4.1

1	2	3	4	5
DeleteAirport	curl -X 'DELETE' \ 'http://localhost:3000/api/Airport/DeleteAirport/1' \ -H 'accept: */*'	http://localhost:3000/api/Airport/DeleteAirport/1		200
GetFlights	curl -X 'GET' \ 'http://localhost:3000/api/Flight/GetFlights' \ -H 'accept: text/plain'	http://localhost:3000/api/Flight/GetFlights	[{ "fId": 1, "fAirline": "string", "fDepartureAirport": "string", "fArrivalAirport": "string", "fDepartureTime": "2025- 10-25T00:00:00", "fArrivalTime": "2025-10- 25T00:00:00", "fSeatsCount": 100, "fPrice": 100 }]	200
GetFlight	curl -X 'GET' \ 'http://localhost:3000/api/Flight/GetFlight/1' \ -H 'accept: text/plain'	http://localhost:3000/api/Flight/GetFlight/1	{ "fId": 1, "fAirline": "string", "fDepartureAirport": "string", "fArrivalAirport": "string", "fDepartureTime": "2025- 10-25T00:00:00", "fArrivalTime": "2025-10- 25T00:00:00", "fSeatsCount": 100, "fPrice": 100 }	200

Продолжение таблицы 4.1

1	2	3	4	5
AddFlight	<pre>curl -X 'POST' \ http://localhost:3000/api/ 00/api/Flight/Add Flight' \ -H 'accept: text/plain' \ -H 'Content- Type: application/json' \ -d '{ "fId": 0, "fAirline": "string", "fDepartureAirpor t": "string", "fArrivalAirport": "string", "fDepartureTime": "2025-10-25", "fArrivalTime": "2025-10-25", "fSeatsCount": 10, "fPrice": 10 }'</pre>	http://localhost:3000/api/00/api/Flight/AddFlight	{ "fId": 0, "fAirline": "string", "fDepartureAirport": "string", "fArrivalAirport": "string", "fDepartureTime": "2025-10-25T00:00:00", "fArrivalTime": "2025-10-25T00:00:00", "fSeatsCount": 10, "fPrice": 10 }	200

Продолжение таблицы 4.1

1	2	3	4	5
EditFlight	<pre>curl -X 'POST' \ 'http://localhost:3000/api/ 00/api/Flight/Edit Flight' \ -H 'accept: text/plain' \ -H 'Content- Type: application/json' \ -d '{ "fId": 1, "fAirline": "string", "fDepartureAirpor t": "string", "fArrivalAirport": "string", "fDepartureTime": "2025-10-25", "fArrivalTime": "2025-10-25", "fSeatsCount": 100, "fPrice": 100 }'</pre>	<pre>http://localhost:3000/api/ Flight/EditFlight</pre>	<pre>{ "fId": 1, "fAirline": "string", "fDepartureAirport": "string", "fArrivalAirport": "string", "fDepartureTime": "2025-10-25T00:00:00", "fArrivalTime": "2025-10-25T00:00:00", "fSeatsCount": 100, "fPrice": 100 }</pre>	200

Продолжение таблицы 4.1

1	2	3	4	5
Register	<pre>curl -X 'POST' \ 'http://localhost:3000/api/ User/Register' -H 'accept: text/plain' \ -H 'Content- Type: application/json' \ -d '{ "uId": 0, "uSurname": "string", "uName": "string", "uPatronymic": "string", "uEmail": "ksebija@gmail.c om", "uPassword": "string", "uRole": "string", "uPhone": "string", "uBirthdate": "2025-10-25", "uPassportSerial": 0, "uPassportNumber": 0 }'</pre>	<p>http://localhost:3000/api/User/Register</p>	<pre>"uId": 1, "uSurname": "string", "uName": "string", "uPatronymic": "string", "uEmail": "ksebija@gmail.com", "uPassword": "AQAAAAIAAYagAAA AEJnn8QdUvop0BHeteah d2U3WPMpEz54Jy0oObp lc9mbDnW1v4eW+0Q1hy F0vAOJ1eg==", "uRole": "Клиент", "uPhone": "string", "uBirthdate": "2025-10- 25", "uPassportSerial": 0, "uPassportNumber": 0 }'</pre>	200

Продолжение таблицы 4.1

1	2	3	4	5
Auth	<pre>curl -X 'POST' \ 'http://localhost:3000/api/ User/Auth' -H 'accept: text/plain' \ -H 'Content- Type: multipart/form- data' \ -F 'login=ksebija@g mail.com' \ -F 'password=string'</pre>	<pre>http://localhost:3000/api/ User/Auth</pre>	<pre>{ "uId": 1, "uSurname": "string", "uName": "string", "uPatronymic": "string", "uEmail": "ksebija@gmail.com", "uPassword": "AQAAAAIAAYagAA AAEJnn8QdUvop0BHet eahd2U3WPMpEz54Jy0 oObplc9mbDnW1v4eW +0Q1hyF0vAOJ1eg==", "uRole": "Клиент", "uPhone": "string", "uBirthdate": "2025-10- 25", "uPassportSerial": 0, "uPassportNumber": 0 }</pre>	200

Продолжение таблицы 4.1

GetUsers	curl -X 'GET' \ 'http://localhost:3000/api/ User/GetUsers' -H 'accept: text/plain'	http://localhost:3000/api/ User/GetUsers	[{ "uId": 1, "uSurname": "string", "uName": "string", "uPatronymic": "string", "uEmail": "ksebija@gmail.com", "uPassword": "AQAAAAIAAYagAA AAEJnn8QdUvop0BHet eahd2U3WPMpEz54Jy0 oObplc9mbDnW1v4eW +0Q1hyF0vAOJ1eg==", "uRole": "Клиент", "uPhone": "string", "uBirthdate": "2025- 10-25", "uPassportSerial": 0, "uPassportNumber": 0 }]	200
----------	--	---	--	-----

Продолжение таблицы 4.1

1	2	3	4	5
EditUser	<pre>curl -X 'POST' \ 'http://localhost:3000/api/ User/EditUser' -H 'accept: text/plain' \ -H 'Content- Type: application/json' \ -d '{ "uId": 1, "uSurname": "qwe", "uName": "qwe", "uPatronymic": "qwe", "uEmail": "aidarka010@gmail.com", "uPassword": "AQAAAAIAAYagAA AAEJnn8QdUvop0BHet eahd2U3WPMpEz54Jy0 oObplc9mbDnW1v4eW +0Q1hyF0vAOJ1eg==", "uRole": "Клиент", "uPhone": "string", "uBirthdate": "2025-10- 25", "uPassportSerial": 0, "uPassportNumber": 0 }'</pre>	http://localhost:3000/api/User/EditUser	{ "uId": 1, "uSurname": "qwe", "uName": "qwe", "uPatronymic": "qwe", "uEmail": "aidarka010@gmail.com", "uPassword": "AQAAAAIAAYagAA AAEJnn8QdUvop0BHet eahd2U3WPMpEz54Jy0 oObplc9mbDnW1v4eW +0Q1hyF0vAOJ1eg==", "uRole": "Клиент", "uPhone": "string", "uBirthdate": "2025-10- 25", "uPassportSerial": 0, "uPassportNumber": 0 }	200

Продолжение таблицы 4.1

1	2	3	4	5
ChangePassword	<pre>curl -X 'POST' \ 'http://localhost:3000/api/User/ChangePassword' \ -H 'accept: text/plain' \ -H 'Content-Type: application/json' \ -d '{ "uId": 1, "uSurname": "string", "uName": "string", "uPatronymic": "string", "uEmail": "string", "uPassword": "qwe", "uRole": "string", "uPhone": "string", "uBirthdate": "2025-10-25", "uPassportSerial": 0, "uPassportNumber": 0 }'</pre>	http://localhost:3000/api/User/ChangePassword	{ "uId": 1, "uSurname": "qwe", "uName": "qwe", "uPatronymic": "qwe", "uEmail": "aidarka010@gmail.com", "uPassword": "AQAAAAIAAYagAA AAEBVvgeAoA1Tps8EsZUWRWwr/h7WPQmTR0r/CLQL1qs2Oh7KfXFo5XNDRpJC17Hr4ZQ=", "uRole": "Клиент", "uPhone": "string", "uBirthdate": "2025-10-25", "uPassportSerial": 0, "uPassportNumber": 0 }	200

Продолжение таблицы 4.1

1	2	3	4	5
DeleteUser	curl -X 'DELETE' \n 'http://localhost:3000/api/User/DeleteUser/1' \n -H 'accept: */*' \n	http://localhost:3000/api/User/DeleteUser/1		200
AddTicket	curl -X 'POST' \n 'http://localhost:3000/api/Ticket/AddTicket' \n -H 'accept: text/plain' \n -H 'Content-Type: application/json' \n -d '{ "tId": 0, "tFlight": 1, "tUser": "string string", "tBoughtDate": "2025-10-25T07:30:05.8840305+00:00", "tClass": "Эконом", "tTotalPrice": 0, "tStatus": "Куплен" }'	http://localhost:3000/api/Ticket/AddTicket	{ "tId": 1, "tFlight": 1, "tUser": "string string", "tBoughtDate": "2025-10-25T07:30:05.8840305+00:00", "tClass": "Эконом", "tTotalPrice": 0, "tStatus": "Куплен" }	200

Продолжение таблицы 4.1

1	2	3	4	5
GetTickets	curl -X 'GET' \ 'http://localhost:3000/api/Ticket/GetTickets' \ -H 'accept: text/plain'	http://localhost:3000/api/Ticket/GetTickets	[{ "tId": 1, "tFlight": 1, "tUser": "string string", "tBoughtDate": "2025-10-25T07:30:05.8840305+00:00", "tClass": "Эконом", "tTotalPrice": 0, "tStatus": "Куплен" }]	200
GetTicket	curl -X 'GET' \ 'http://localhost:3000/api/Ticket/GetTicket/1' \ -H 'accept: text/plain'	http://localhost:3000/api/Ticket/GetTicket/1	{ "tId": 1, "tFlight": 1, "tUser": "string string", "tBoughtDate": "2025-10-25T07:30:05.8840305+00:00", "tClass": "Эконом", "tTotalPrice": 0, "tStatus": "Куплен" }	200

Продолжение таблицы 4.1

1	2	3	4	5
ChangeTicketStatus	<pre>curl -X 'POST' \ 'http://localhost:3000/api/Ticket/ChangeTicketStatus' \ -H 'accept: text/plain' \ -H 'Content-Type: application/json' \ -d '{ "tId": 1, "tFlight": 1, "tUser": "string string", "tBoughtDate": "2025-10-26T07:32:32.195Z", "tClass": "string", "tTotalPrice": 0, "tStatus": "Отменен" }'</pre>	<p>http://localhost:3000/api/Ticket/ChangeTicketStatus</p>	<pre>{ "tId": 1, "tFlight": 1, "tUser": "string string", "tBoughtDate": "2025-10-25T07:30:05.8840305+00:00", "tClass": "Эконом", "tTotalPrice": 0, "tStatus": "Отменен" }</pre>	200

5 Результаты работы

Данное руководство содержит подробные инструкции и информацию, необходимую для самостоятельного и эффективного использования веб-сайта Skywhysales. Вы научитесь находить рейсы, покупать и отменять билеты, управлять своим аккаунтом и использовать все возможности платформы.

Skywhysales – это информационная система для автоматизации покупки авиабилетов. Платформа объединяет пользователей (клиентов), чтобы сделать процесс планирования путешествий простым, быстрым и удобным.

Веб-приложение ориентировано на следующие категории пользователей: – клиент, имеющий возможность покупать и отменять билеты, редактировать личные данные; – менеджер, добавляющий, удаляющий и редактирующий рейсы, аэропорты, авиакомпании, а также просматривающий билеты и пользователей. Каждая роль имеет собственный уровень доступа, от которого зависят функциональные возможности и доступ к разделам системы. Руководство пользователя для клиента

Для работы с системой выполните следующие шаги в указанном порядке:

1. Запуск серверной части:

- запустите докер;
- запустите api.

2. Запуск клиентской части:

- откройте папку с проектом;
- впишите в пути проводника cmd и нажмите enter;
- введите команду npm run preview;
- перейдите по адресу сайта <http://localhost:4173/>.

При первом посещении сайта пользователь видит «Главную страницу с формой поиска рейсов» представленную на рисунке 5.1.

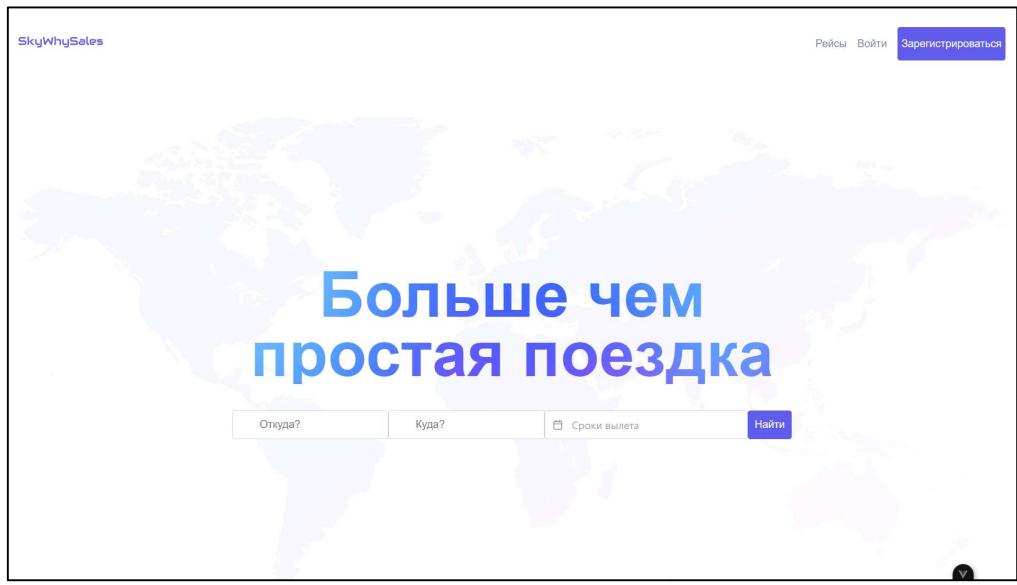


Рисунок 5.1 – Главная страница

В верхней части страницы находится основное меню навигации с элементами: логотип Skywhysales (ссылка на главную), «Рейсы» (переход к списку рейсов), «Войти» (модальное окно для входа), «Зарегистрироваться» (кнопка для перехода к регистрации) или «Мой аккаунт» и «Мои рейсы» (если пользователь авторизован). В центральной части расположена форма поиска рейсов с полями: «Откуда?» (выбор города отправления из списка), «Куда?» (выбор города прибытия из списка), «Сроки полета» (выбор диапазона дат с календарем), кнопка «Найти» для поиска. В нижней части страницы размещен раздел «Связаться с нами» он представлен на рисунке 5.2.

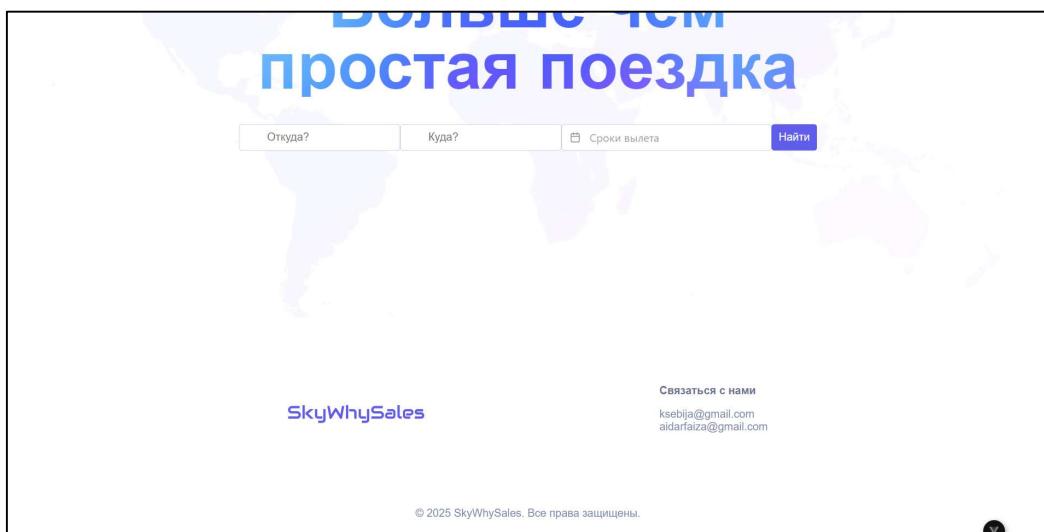


Рисунок 5.2 – Раздел «Связаться с нами»

Для доступа к полным возможностям платформы (покупка билетов, личный кабинет) необходимо зарегистрироваться или войти в аккаунт.

Для регистрации перейдите на страницу регистрации, представленную на рисунке 5.3, нажав «Зарегистрироваться» в шапке.

The screenshot shows the registration form for the SkyWhySales platform. At the top left is the logo 'SkyWhySales'. At the top right are links for 'Рейсы' (Flights) and 'Войти' (Log in). The main title 'Регистрация' (Registration) is centered above a series of input fields. Each field has a placeholder text: 'Введите фамилию' (Enter last name), 'Введите имя' (Enter first name), 'Введите отчество' (Enter middle name), 'Введите email' (Enter email), 'Введите пароль' (Enter password), 'Введите телефон' (Enter phone number), 'Введите дату рождения' (Enter date of birth), 'Введите серию паспорта' (Enter passport series), and 'Введите номер паспорта' (Enter passport number). Below these fields is a blue button labeled 'Зарегистрироваться' (Register). At the bottom left is the 'SkyWhySales' logo again. At the bottom right, there is a link 'Связаться с нами' (Contact us) followed by two email addresses: ksebija@gmail.com and aidarfaiza@gmail.com.

Рисунок 5.3 – Страница регистрации

Заполните форму:

- Фамилия (обязательно);
- Имя (обязательно);
- Отчество (опционально);
- Email (обязательно, формат: example@domain.com);
- Пароль (обязательно, минимум 8 символов, с заглавной буквой, строчной и цифрой);
- Телефон (обязательно, формат: +7XXXXXXXXXX);
- Дата рождения (обязательно, через календарь; пользователь должен быть старше 18 лет);
- Серия паспорта (обязательно, 4 цифры);
- Номер паспорта (обязательно, 6 цифр).

Нажмите кнопку «Зарегистрироваться». При успешной регистрации вы будете перенаправлены на главную страницу и авторизованы автоматически. на рисунке 5.4 представлено уведомление об успешной регистрации.

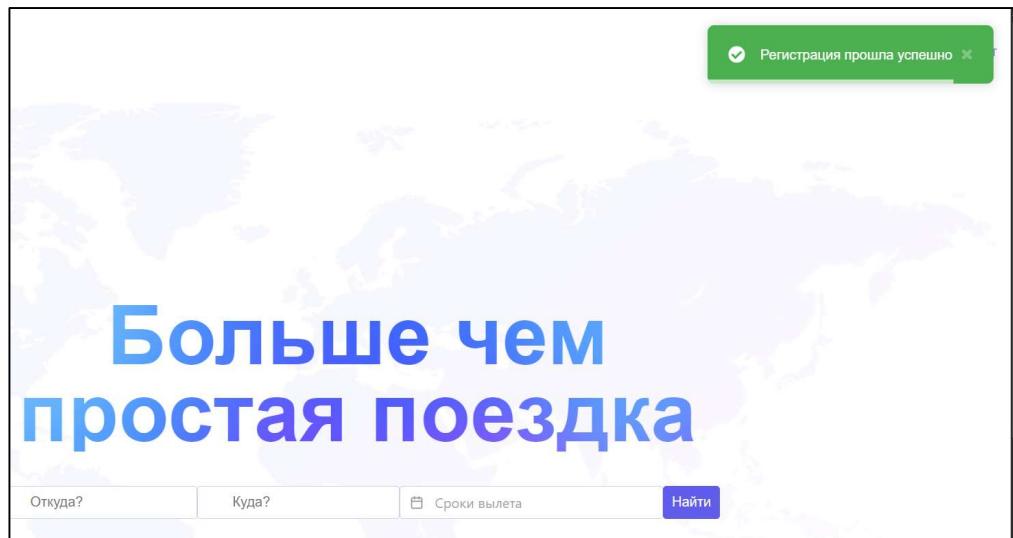


Рисунок 5.4 – Главная страница

На рисунке 5.5 представлено уведомление в случае ошибки (например, неверный формат).

A screenshot of a registration form titled "Регистрация". The form includes fields for first name ("Иванов"), middle name ("Иван"), father's name ("Введите отчество"), email ("аремтоп495@gmail.com"), password ("*****"), phone number ("88005553535"), date of birth ("11/18/2000"), and two additional fields with the numbers "7899" and "732849". A red error message at the top right states "⚠️ Указанный номер телефона занят" (The specified phone number is taken). At the bottom is a blue "Зарегистрироваться" button.

Рисунок 5.5 – Страница регистрации

Для входа нажмите «Войти» в шапке – откроется модальное окно, представленное на рисунке 5.6. Введите Email и Пароль, нажмите «Войти». Если аккаунта нет, кликните на ссылку «Нет аккаунта? Зарегистрироваться» для перехода к регистрации.

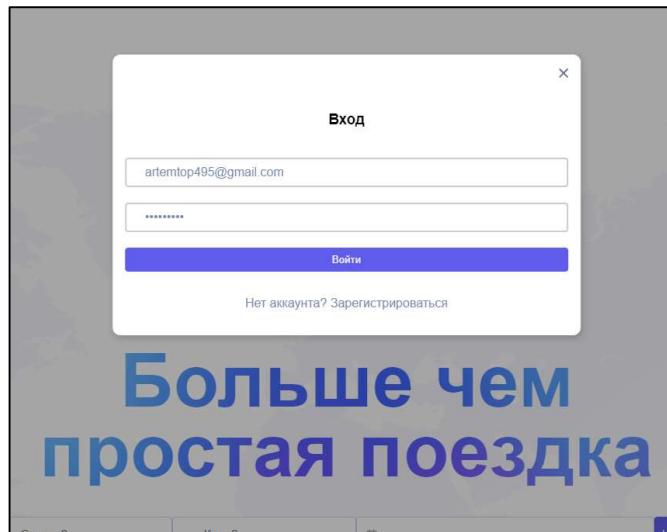


Рисунок 5.6 – Модальное окно входа

При успешном входе окно закроется, и в шапке появятся опции для авторизованного пользователя. На рисунке 5.7 представлены эти опции.

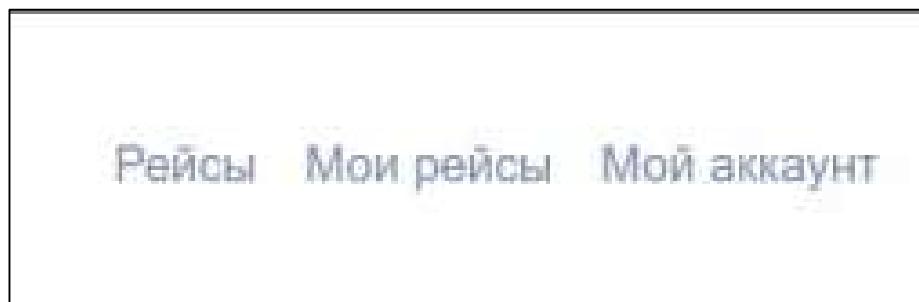


Рисунок 5.7 – Опции авторизованного пользователя

На главной странице или странице «Рейсы» используйте форму поиска, которая представлена на рисунке 5.8:

- Выберите город отправления из списка;
- Выберите город прибытия (не может совпадать с отправлением);
- Укажите диапазон дат, в которые готовы вылететь.

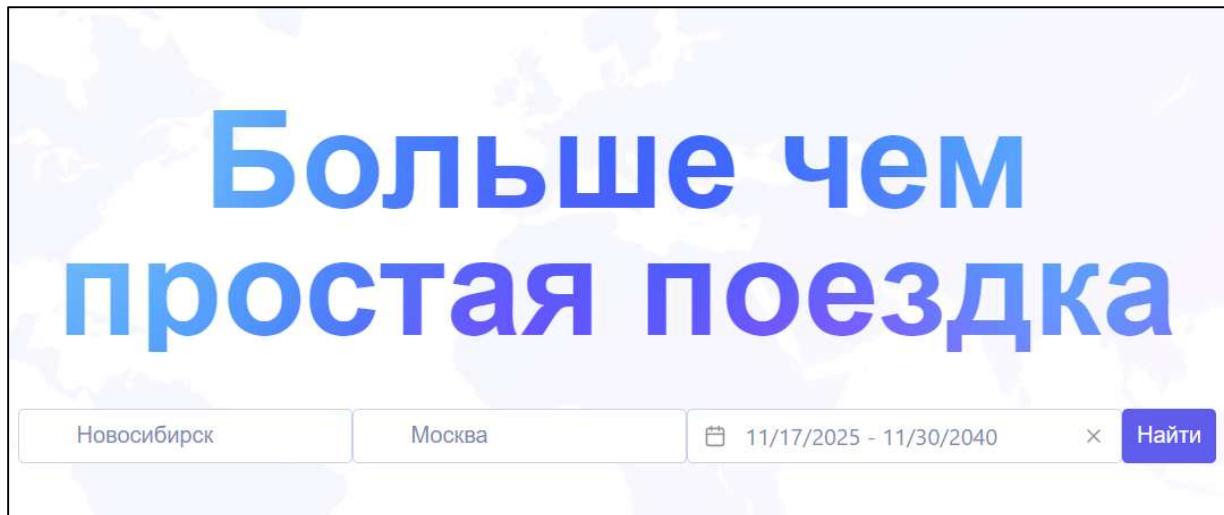


Рисунок 5.8 – Форма поиска

Нажмите «Найти» система перенаправит на страницу с результатами, представленную на рисунке 5.9.

Авиакомпания	Время вылета	Аэропорт вылета	Аэропорт прибытия	Время прибытия	Цена
Utair	3ч 40 мин	Толмачёво	Домодедово	04.12.2025, 09:30 - 04.12.2025, 13:10	14000 ₽

Рисунок 5.9 – Страница отфильтрованных рейсов

Результаты отображаются в виде карточек рейсов: каждая карточка содержит логотип авиакомпании, продолжительность полета, название авиакомпании, аэропорты отправления/прибытия, даты и время отправления/прибытия, цену. Можно показать все доступные билеты без фильтров нажав на кнопку посмотреть все. На рисунке 5.10 показаны все доступные билеты без фильтров.

The screenshot shows a search interface for flights. At the top, there are input fields for 'Откуда?' (From), 'Куда?' (To), and 'Сроки вылета' (Departure dates), followed by a 'Найти' (Find) button. Below this, a section titled 'Выберите рейс вылета' (Select departure flight) displays a list of flights:

Время	Авиакомпания	Путь	Цена
1ч 30 мин	S7 Airlines	Пулково - Домодедово 01.12.2025, 11:00 - 01.12.2025, 12:30	8200 ₽
2ч 30 мин	Победа	Домодедово - Сочи 02.12.2025, 14:20 - 02.12.2025, 16:50	7200 ₽
3ч 25 мин	S7 Airlines	Домодедово - Кольцово 03.12.2025, 07:15 - 03.12.2025, 10:40	12500 ₽
3ч 40 мин	Utair	Толмачево - Домодедово 04.12.2025, 09:30 - 04.12.2025, 13:10	14000 ₽
2ч 20 мин	Россия	Сочи - Домодедово 05.12.2025, 18:00 - 05.12.2025, 20:20	6800 ₽
2ч 30 мин	Nordwind	Домодедово - Симферополь 06.12.2025, 10:10 - 06.12.2025, 12:40	9500 ₽
2ч 10 мин		Казань - Домодедово	

Рисунок 5.10 – Страница всех рейсов

Для просмотра деталей рейса кликните на карточку рейса – откроется страница детали рейса. На рисунке 5.11 показана страница деталей рейса.

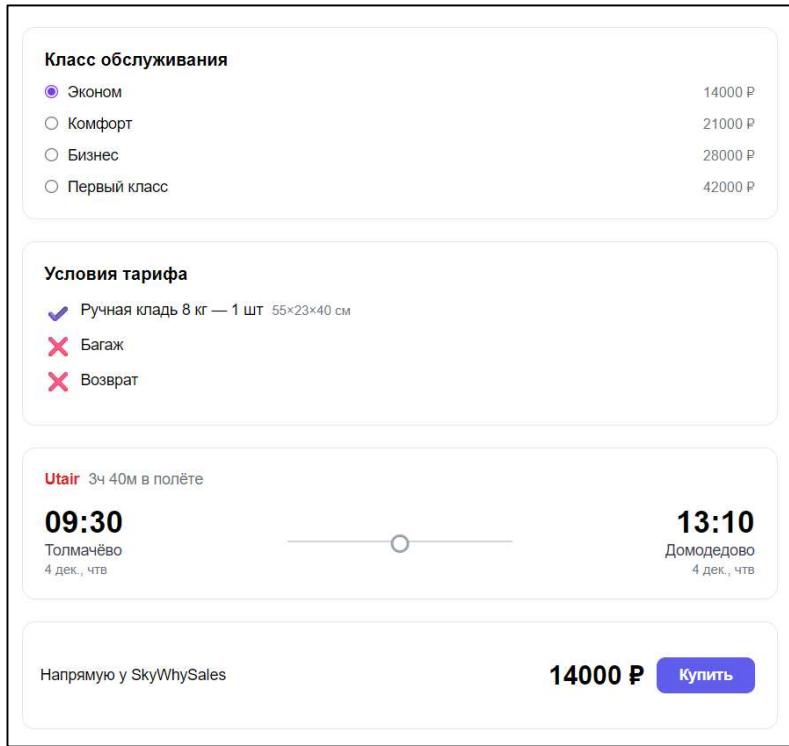


Рисунок 5.11 – Страница деталей рейса

Здесь отображается:

- Условия тарифа (ручная кладь, багаж, возврат);
- Маршрут (время отправления/прибытия, аэропорты, продолжительность);
- Цена и кнопка «Купить» (прямая покупка у Skywhysales).

По нажатию на кнопку купить оформляется билет. На рисунке 5.12 показано уведомление об успешном оформлении билета.

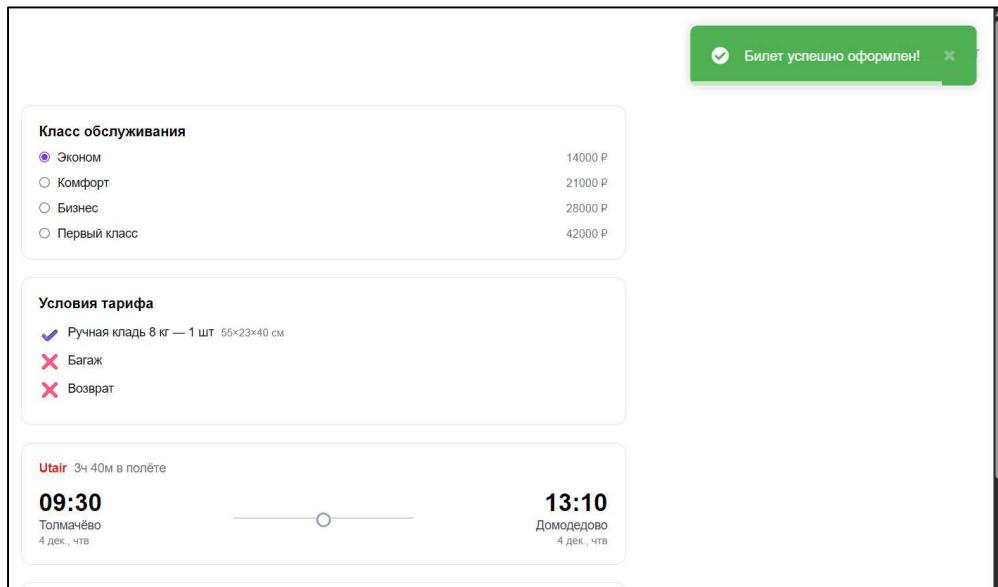


Рисунок 12 – Уведомление об оформлении билета

На рисунке 5.13 показано уведомление на почту об успешном оформлении билета.

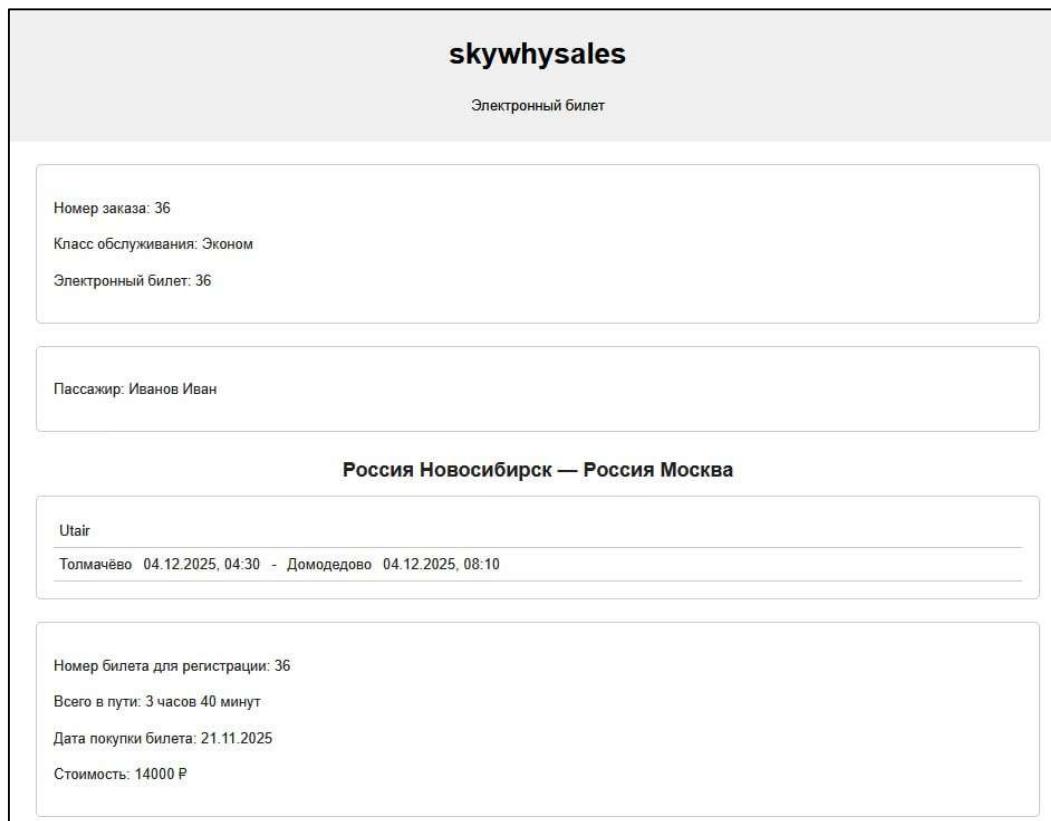


Рисунок 5.13 – Электронное письмо с подтверждением покупки

Для открытия личного кабинета после входа перейдите в «Мой аккаунт» (в шапке). На рисунке 5.14 представлено окно «Мой аккаунт».

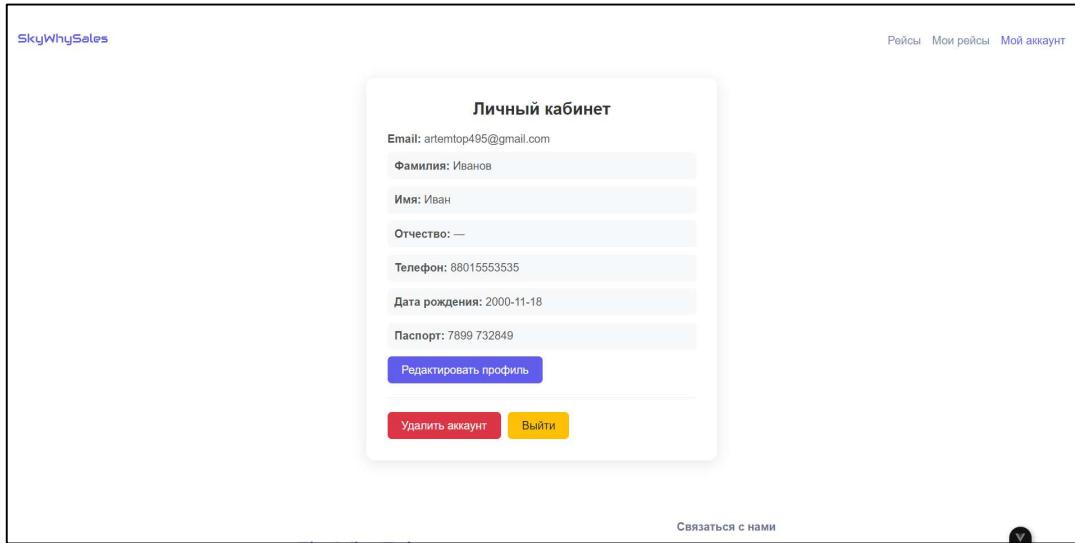


Рисунок 5.14 – Страница профиля

Здесь отображается:

- Email;
- Личные данные: Фамилия, Имя, Отчество, Телефон, Дата рождения, Паспорт.

Для редактирования нажмите «Редактировать профиль» — поля станут редактируемыми. на рисунке 5.15 представлены редактируемые поля.

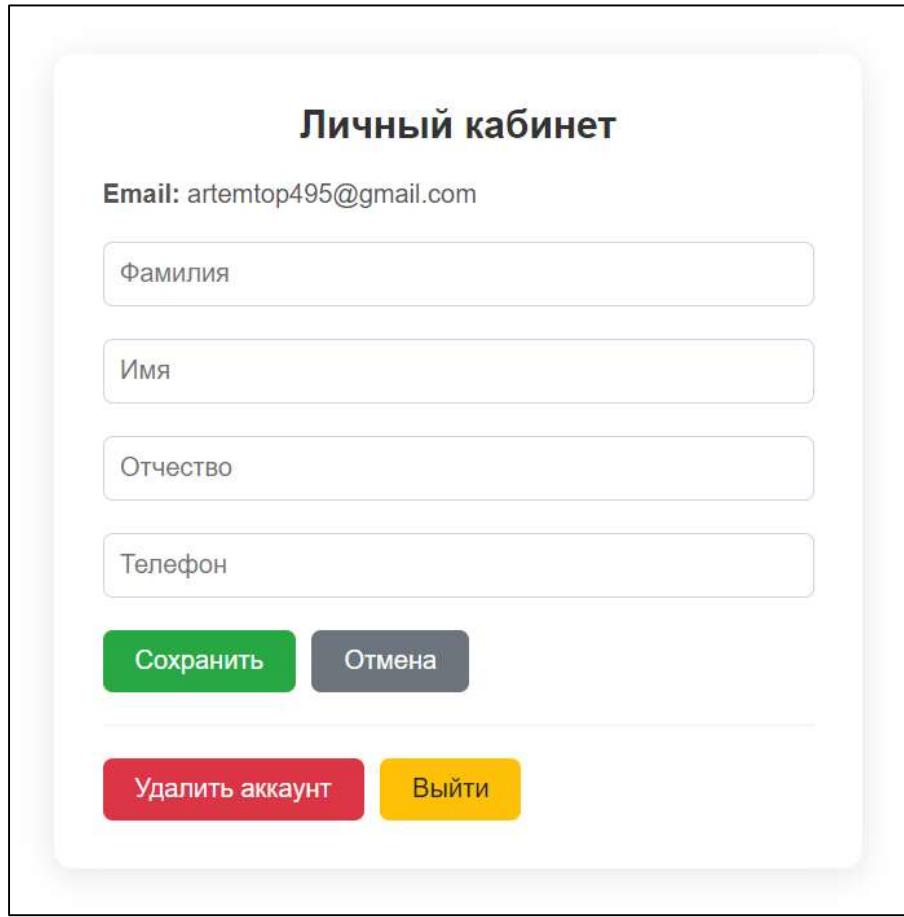


Рисунок 5.15 – Страница редактирования профиля

Заполните и нажмите «Сохранить». Для отмены – «Отмена».

В разделе также доступны:

- «Удалить аккаунт» (с подтверждением, необратимо. На рисунке 5.16 представлено подтверждение);
- «Выйти» (выход из аккаунта).

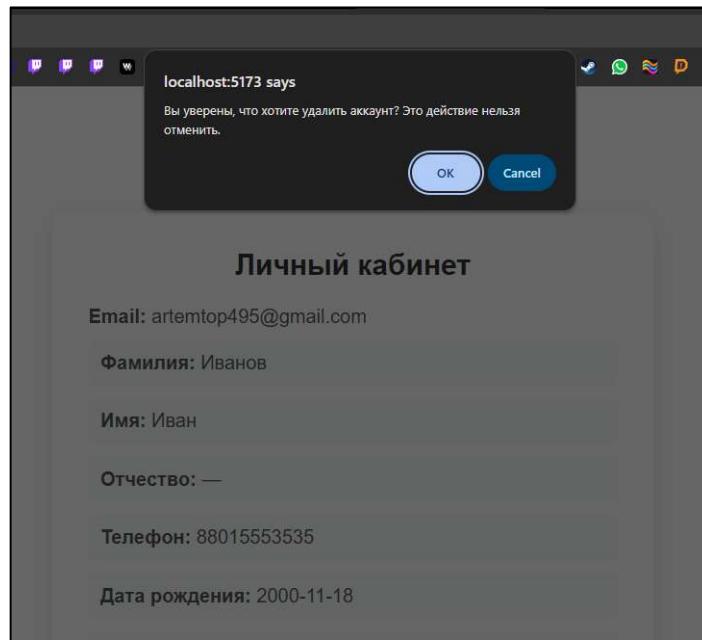


Рисунок 5.16 – Подтверждение при удалении аккаунта

Для просмотра оформленных билетов перейдите в «Мои рейсы» (в шапке). На рисунке 5.17 представлено окно «Мои рейсы».

SkyWhySales					Rейсы	Мои рейсы	Мой аккаунт
	Utair	Толмачёво - Домодедово 04.12.2025, 09:30 - 04.12.2025, 13:10	14000 ₽ Стоимость билета	Куплен Статус	<button>Отменить</button>		
	Победа	Домодедово - Сочи 02.12.2025, 14:20 - 02.12.2025, 16:50	21600 ₽ Стоимость билета	Куплен Статус	<button>Отменить</button>		
	Россия	Сочи - Домодедово 05.12.2025, 16:00 - 05.12.2025, 20:20	6800 ₽ Стоимость билета	Куплен Статус	<button>Отменить</button>		
	Red Wings	Домодедово - Ростов-на-Дону 10.12.2025, 09:00 - 10.12.2025, 11:20	25200 ₽ Стоимость билета	Куплен Статус	<button>Отменить</button>		
	Red Wings	Домодедово - Ростов-на-Дону 10.12.2025, 09:00 - 10.12.2025, 11:20	25200 ₽ Стоимость билета	Куплен Статус	<button>Отменить</button>		
	Red Wings	Домодедово - Ростов-на-Дону 10.12.2025, 09:00 - 10.12.2025, 11:20	25200 ₽ Стоимость билета	Куплен Статус	<button>Отменить</button>		
	Red Wings	Домодедово - Ростов-на-Дону 10.12.2025, 09:00 - 10.12.2025, 11:20	25200 ₽ Стоимость билета	Куплен Статус	<button>Отменить</button>		
	Red Wings	Домодедово - Ростов-на-Дону 10.12.2025, 09:00 - 10.12.2025, 11:20	25200 ₽ Стоимость билета	Куплен Статус	<button>Отменить</button>		

Рисунок 5.17 – Страница оформленных билетов

При нажатии на кнопку отменить появляется меню подтверждения. на рисунке 5.18 представлено меню подтверждения.

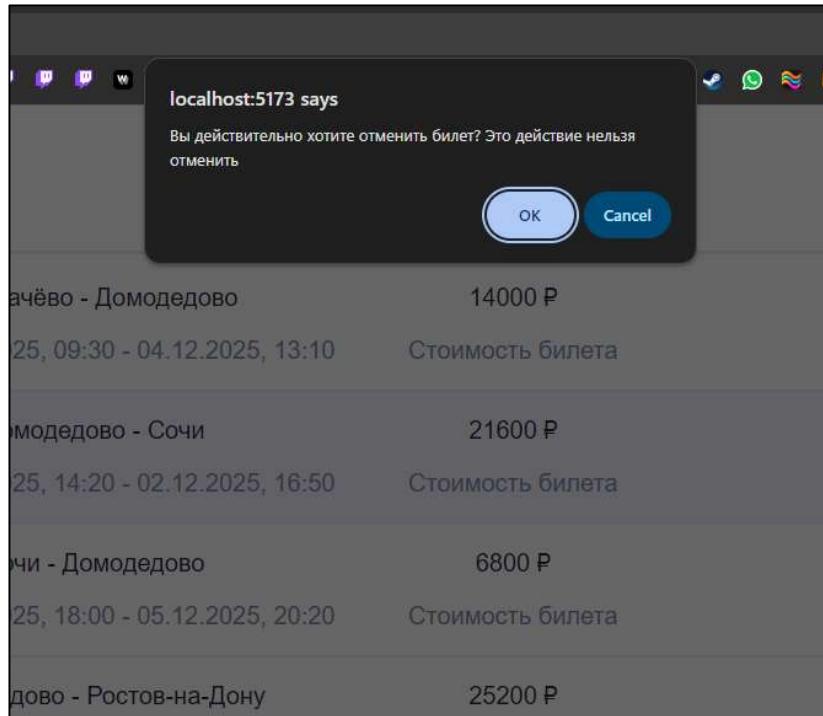


Рисунок 5.18 – подтверждение при отмене билета

После подтверждения статус билета меняется на отменен, и кнопка становится неактивной. На рисунке 5.19 представлено обновленное окно «Мои рейсы».

Авиакомпания	Направление	Дата вылета	Стоимость билета	Статус	Кнопка
Ulair	Толмачево - Домодедово	04.12.2025, 09:30 - 04.12.2025, 13:10	14000 ₽	Куплен	Отменить
Победа	Домодедово - Сочи	02.12.2025, 14:20 - 02.12.2025, 16:50	21600 ₽	Куплен	Отменить
Россия	Сочи - Домодедово	05.12.2025, 18:00 - 05.12.2025, 20:20	6800 ₽	Отменен	Отменен
Red Wings	Домодедово - Ростов-на-Дону	10.12.2025, 09:00 - 10.12.2025, 11:20	8400 ₽	Куплен	Отменить
Red Wings	Домодедово - Ростов-на-Дону	10.12.2025, 09:00 - 10.12.2025, 11:20	25200 ₽	Куплен	Отменить
Red Wings	Домодедово - Ростов-на-Дону	10.12.2025, 09:00 - 10.12.2025, 11:20	8400 ₽	Куплен	Отменить

SkyWhySales

Рейсы Мои рейсы Мой аккаунт

Связаться с нами
ksebija@gmail.com
aidarfaiza@mail.com

Рисунок 5.19 – Страница с отмененным билетом

Руководство пользователя для менеджера

Менеджеру доступны все те же возможности, что и пользователю. Только у него ещё появляется раздел «Админ-панель» в шапке веб-приложения. Необходимо авторизоваться в системе за пользователя с ролью «Менеджер». Перейти в раздел «Админ-панель». На рисунке 5.20 представлена кнопка открытия админ-панели.

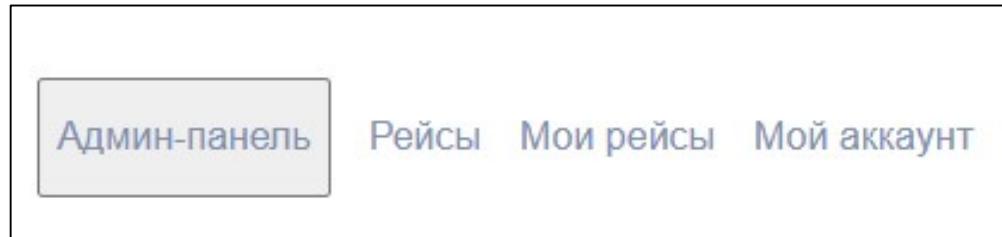


Рисунок 5.20 – Шапка приложения с админ-панелью

По умолчанию в админ-панели выбран раздел обзор. Он показывает количество записей для каждого из классов. На рисунке 5.21 представлен раздел «Обзор».

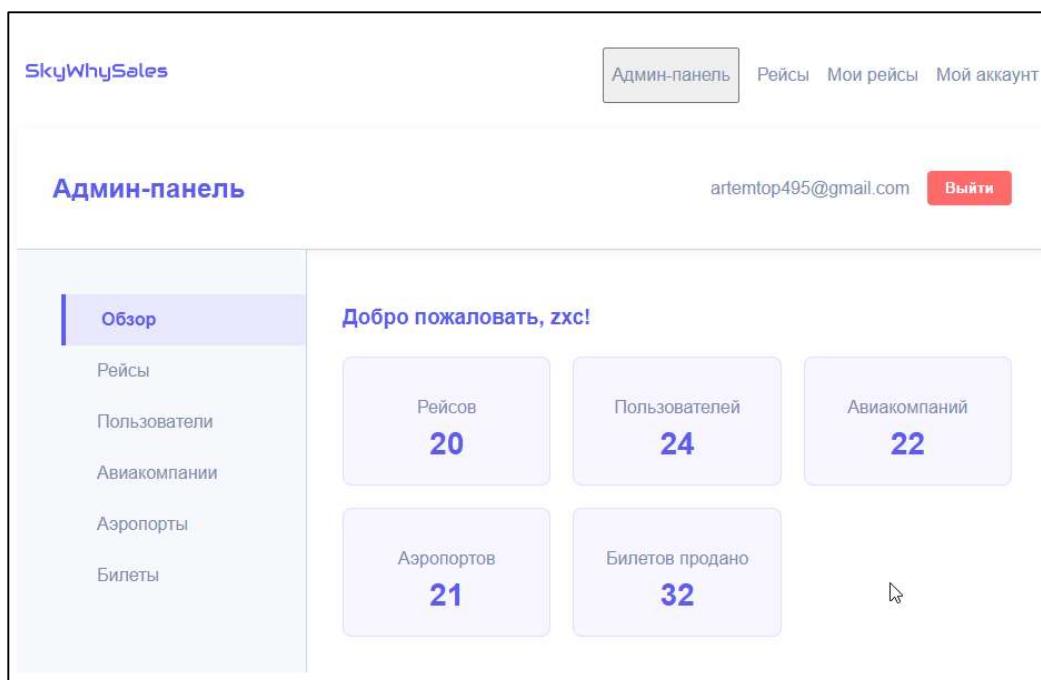


Рисунок 21 – Раздел обзор

В разделе рейсы можно просматривать, редактировать и добавлять рейсы. На рисунке 5.22 представлен раздел «рейсы».

Рейсы							
ID	Откуда → Куда	Вылет	Прилёт	Авиакомпания	Цена	Мест	Действия
2	Пулково → String	01.12.2025, 11:00	12:30	S7 Airlines	8200 ₽	150	<button>Изменить</button>
3	Домодедово → Сочи	02.12.2025, 14:20	16:50	Победа	7200 ₽	189	<button>Изменить</button>
4	String → Кольцово	03.12.2025, 07:15	10:40	string	12500 ₽	200	<button>Изменить</button>
5	Толмачёво → String	04.12.2025, 09:30	13:10	Utair	14000 ₽	160	<button>Изменить</button>
6	Сочи → Домодедово	05.12.2025, 18:00	20:20	Россия	6800 ₽	170	<button>Изменить</button>
7	String → Симферополь	06.12.2025, 10:10	12:40	Nordwind	9500 ₽	180	<button>Изменить</button>
8	Казань → String	07.12.2025, 06:45	08:55	Smartavia	7800 ₽	140	<button>Изменить</button>
9	Уфа → Пулково	08.12.2025, 12:00	14:30	Azimuth	8800 ₽	150	<button>Изменить</button>
10	Самара → String	09.12.2025, 15:25	17:50	Ямал	9200 ₽	160	<button>Изменить</button>
11	String → Ростов-на-Дону	10.12.2025, 09:00	11:20	Red Wings	8400 ₽	170	<button>Изменить</button>

Рисунок 5.22 – Раздел рейсы

При нажатии кнопки добавить открывается окно добавления рейса. Для добавления нужно нажать кнопку сохранить. На рисунке 5.23 представлено окно добавления рейса.

Добавление рейса

Аэропорт отправления

Аэропорт прибытия

Дата и время вылета

Время прилёта

Авиакомпания (название)

Цена (₽)

Количество мест

Рисунок 5.23 – Окно добавления рейса

При нажатии кнопки сохранить открывается главное окно админ-панели с уведомлением об успешном добавлении. На рисунке 5.24 представлено уведомление об успешном добавлении.

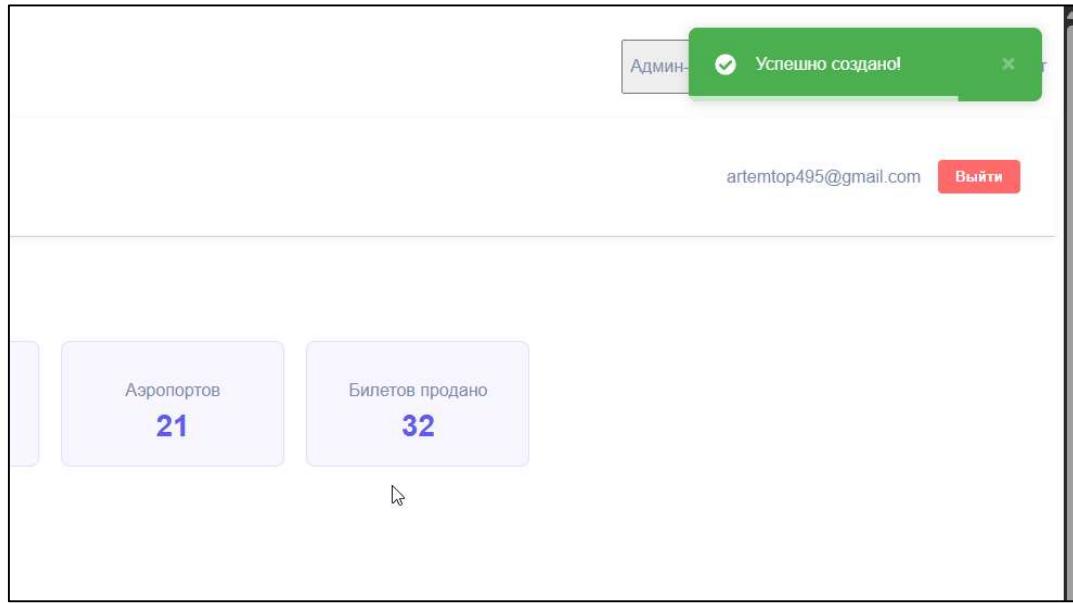


Рисунок 5.24 – Уведомление об успешном добавлении

При нажатии кнопки изменить открывается окно редактирования рейса. Для сохранения нужно нажать кнопку сохранить. На рисунке 5.25 представлено окно редактирования рейса.

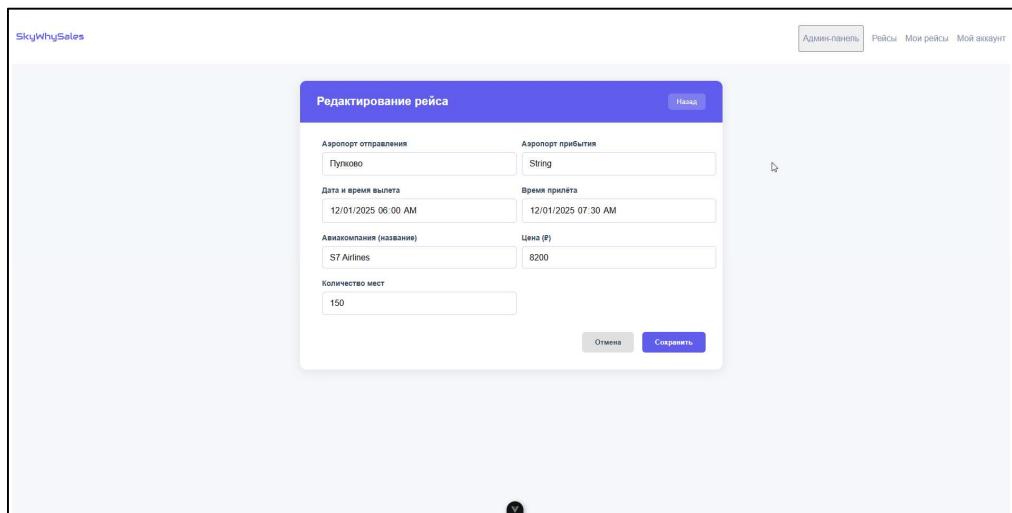


Рисунок 5.25 – Окно редактирования рейса

При нажатии кнопки сохранить открывается главное окно админ-панели с уведомлением об успешном добавлении. На рисунке 5.26 представлено уведомление об успешном обновлении.

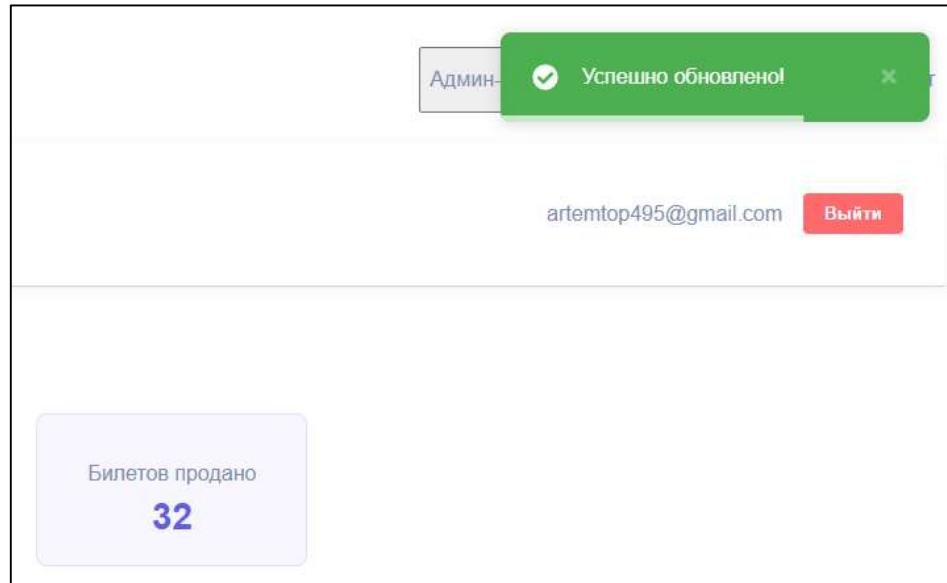


Рисунок 5.26 – Уведомление об успешном обновлении

В разделе пользователи можно просматривать всех пользователей системы. На рисунке 5.27 представлен раздел «Пользователи».

Пользователи					
ID	ФИО	Email	Телефон	Роль	Паспорт
2	Иванов Иван Иванович	ivanov@example.com	+79990000001	Клиент	1234 567891
3	Петров Петр Петрович	petrov@example.com	+79990000002	Клиент	2345 678901
4	Сидоров Алексей Александрович	sidorov@example.com	+79990000003	Клиент	3456 789013
5	Кузнецова Мария Сергеевна	kuznetsova@example.com	+79990000004	Клиент	4567 890124
6	Смирнов Дмитрий Степанович	smirnov@example.com	+79990000005	Клиент	5678 901235
7	Васильева Анна Михайловна	vasilieva@example.com	+79990000006	Клиент	6789 123455
8	Михайлов Сергей Владимирович	mihaylov@example.com	+79990000007	Клиент	7890 234566
9	Новикова Елена Андреевна	novikova@example.com	+79990000008	Клиент	8901 345677
10	Федоров Николай Павлович	fedorov@example.com	+79990000009	Клиент	9012 456788
11	Морозова Ольга Дмитриевна	morozova@example.com	+79990000010	Клиент	1123 567890
12	Волков Александр Юрьевич	volkov@example.com	+79990000011	Клиент	2234 678902
13	Лебедева Татьяна Игоревна	lebedeva@example.com	+79990000012	Клиент	3345 789012
14	Козлов Артем Романович	kozlov@example.com	+79990000013	Клиент	4456 890123

Рисунок 27 – Раздел пользователи

В разделе авиакомпаний можно просматривать, редактировать, удалять и добавлять авиакомпании. На рисунке 5.28 представлен раздел «Авиакомпании».

ID	Название	Email	Действия
1	string	string	<button>Изменить</button> <button>Удалить</button>
2	S7 Airlines	support@s7.ru	<button>Изменить</button> <button>Удалить</button>
3	Победа	help@pobeda.aero	<button>Изменить</button> <button>Удалить</button>
4	Utair	contact@utair.ru	<button>Изменить</button> <button>Удалить</button>
5	Россия	info@rossiya-airlines.com	<button>Изменить</button> <button>Удалить</button>
6	Nordwind	office@nordwind.ru	<button>Изменить</button> <button>Удалить</button>
7	Smartavia	smart@smartavia.ru	<button>Изменить</button> <button>Удалить</button>
8	Azimuth	azimuth@azimuth.ru	<button>Изменить</button> <button>Удалить</button>
9	Ямал	yamal@yamal.aero	<button>Изменить</button> <button>Удалить</button>
10	Red Wings	redwings@redwings.ru	<button>Изменить</button> <button>Удалить</button>
11	ИрАэро	iraero@iraero.ru	<button>Изменить</button> <button>Удалить</button>
12	Аэрофлот	airrosa@airrosa.aero	<button>Изменить</button> <button>Удалить</button>

Рисунок 5.28 – Раздел авиакомпаний

При нажатии кнопки добавить открывается окно добавления авиакомпании. На рисунке 5.29 представлено окно добавления авиакомпании.

Добавление авиакомпании

Название авиакомпании Email

Отмена Сохранить

Рисунок 5.29 – Окно добавления авиакомпании

При нажатии кнопки изменить открывается окно редактирования авиакомпании. На рисунке 5.30 представлено окно редактирования авиакомпании.

Редактирование авиакомпании

Название авиакомпании: S7 Airlines

Email: support@s7.ru

Отмена Сохранить

Рисунок 5.30 – Окно редактирования авиакомпании

В разделе аэропорты можно просматривать, редактировать, удалять и добавлять аэропорты. На рисунке 5.31 представлен раздел аэропорты.

Авиакомпании

ID	Название	Email	Действия
1	string	string	[Изменить, Удалить]
2	S7 Airlines	support@s7.ru	[Изменить, Удалить]
3	Победа	help@pobeda.aero	[Изменить, Удалить]
4	Ural	contact@uralair.ru	[Изменить, Удалить]
5	Россия	info@rossiya-airlines.com	[Изменить, Удалить]
6	Nordwind	office@nordwind.ru	[Изменить, Удалить]
7	Smartavia	smart@smartavia.ru	[Изменить, Удалить]
8	Azimuth	azimuth@azimuth.ru	[Изменить, Удалить]
9	Ямал	yamal@yamal.aero	[Изменить, Удалить]
10	Red Wings	redwings@redwings.ru	[Изменить, Удалить]
11	ИрАэро	iraeo@iraeo.ru	[Изменить, Удалить]
12	Алроса	alrosa@alrosa.aero	[Изменить, Удалить]

+ Добавить

Рисунок 31 – Раздел аэропорты

При нажатии кнопки добавить открывается окно добавления аэропорта. На рисунке 5.32 представлено окно добавления аэропорта.

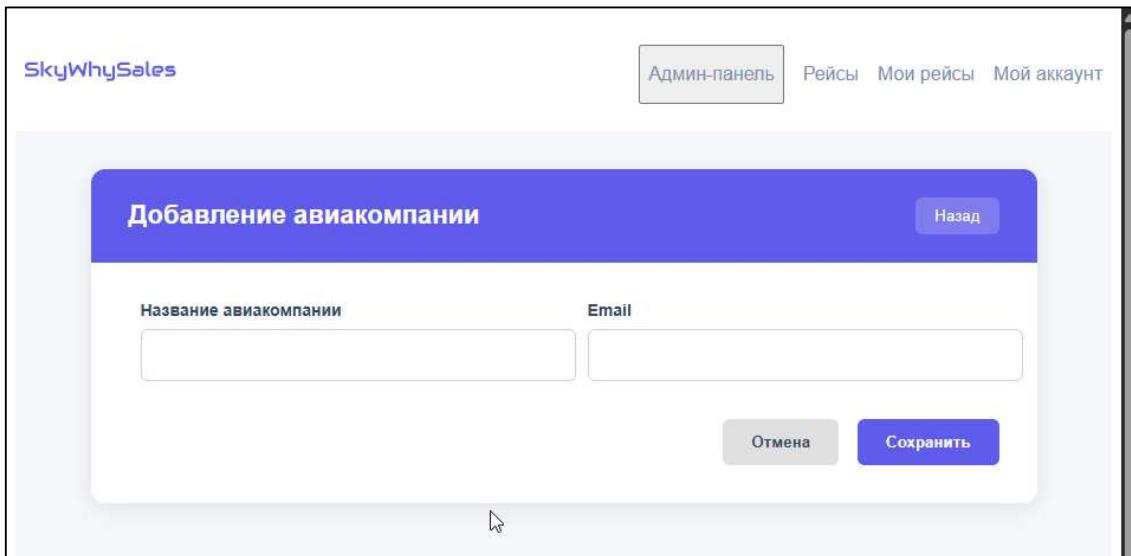


Рисунок 5.32 – Окно добавления авиакомпании

При нажатии кнопки изменить открывается окно редактирования аэропорта. На рисунке 5.33 представлено окно редактирования аэропорта.

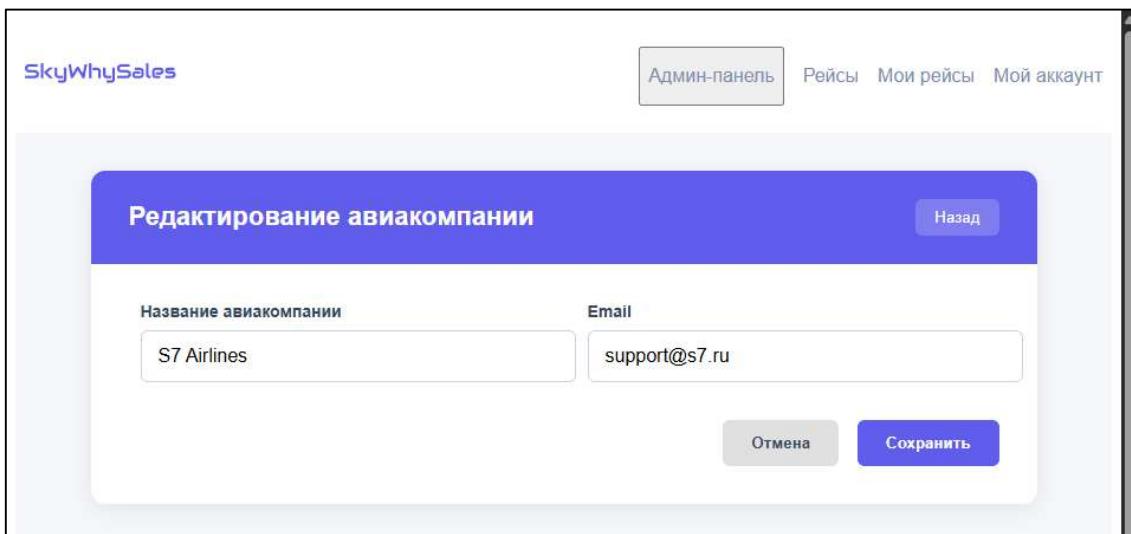


Рисунок 5.33 – Окно редактирования авиакомпании

При попытке удаления аэропорта или авиакомпании появляется окно подтверждения удаления. На рисунке 5.34 представлено окно подтверждения удаления.

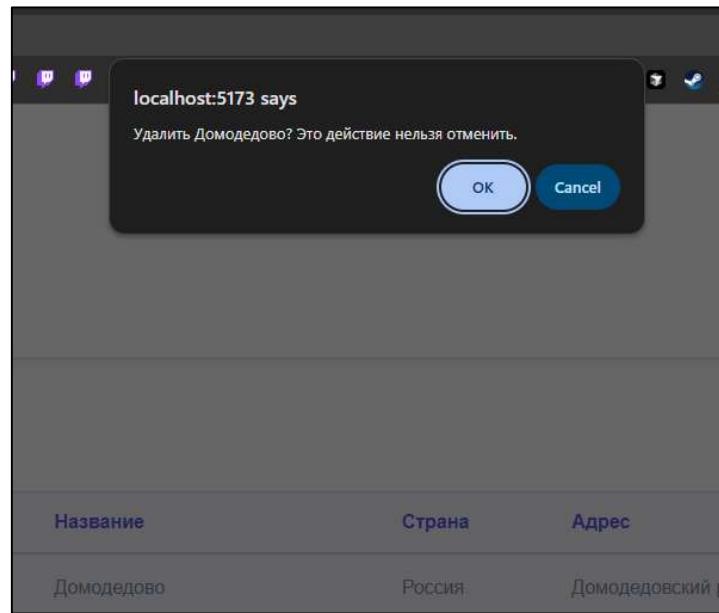


Рисунок 5.34 – Окно подтверждения удаления

В разделе билеты можно просматривать все билеты. на рисунке 5.35 представлен раздел «Билеты».

Проданные билеты						
ID	Пассажир	Рейс ID	Дата покупки	Класс	Цена	Статус
2	Иванов Иван Иванович	2	01.11.2025	Комфорт	12300 ₽	Куплен
3	Петров Петр Петрович	3	02.11.2025	Эконом	7200 ₽	Куплен
4	Сидоров Алексей Александрович	4	02.11.2025	Бизнес	37500 ₽	Куплен
5	Кузнецова Мария Сергеевна	5	03.11.2025	Эконом	14000 ₽	Куплен
6	Смирнов Дмитрий Олегович	6	03.11.2025	Комфорт	10200 ₽	Куплен
7	Васильева Анна Михайловна	7	04.11.2025	Эконом	9500 ₽	Куплен
8	Михайлов Сергей Владимирович	8	04.11.2025	Первый класс	31200 ₽	Куплен
9	Новикова Елена Андреевна	9	05.11.2025	Эконом	8800 ₽	Куплен
10	Федоров Николай Павлович	10	05.11.2025	Бизнес	27600 ₽	Куплен
11	Морозова Ольга Дмитриевна	11	06.11.2025	Эконом	8400 ₽	Куплен
12	Волков Александр Юрьевич	12	06.11.2025	Комфорт	10650 ₽	Куплен
13	Лебедева Татьяна Игоревна	13	07.11.2025	Эконом	8900 ₽	Куплен
14	Козлов Артем Романович	14	07.11.2025	Бизнес	55500 ₽	Куплен

Рисунок 5.35 – Раздел билеты

ЗАКЛЮЧЕНИЕ

В ходе учебной практики была успешно выполнена задача по разработке полнофункционального веб-приложения «Skywhysales», предназначенного для покупки авиабилетов. Все цели и задачи, поставленные в техническом задании, были достигнуты.

В результате проделанной работы:

- реализовано работоспособное веб-приложение с клиентской частью на Vue.js и серверной частью на ASP.NET Core, использующее базу данных PostgreSQL;
- разработан полный жизненный цикл взаимодействия пользователя с платформой от регистрации и поиска рейсов с помощью системы фильтров;
- создан интуитивно понятный и адаптивный пользовательский интерфейс, соответствующий современным стандартам UI/UX-дизайна;
- реализована система ролей, обеспечивающая разделение функционала между обычными пользователями и менеджерами. Менеджерам предоставлены инструменты для управления каталогом рейсов;

Практическая значимость проекта заключается в решении актуальной задачи автоматизации покупки авиабилетов, что делает процесс планирования перелетов простым, быстрым и удобным для всех.

Таким образом, в рамках учебной практики были закреплены и применены на практике знания по веб-разработке, проектированию баз данных, клиент-серверному взаимодействию и проектной деятельности. Проект «Skywhysales» считается завершенным и готовым к дальнейшему развитию. В качестве перспектив развития платформы можно выделить интеграцию платежных систем и разработку мобильного приложения.