

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.

Студент гр. 22П-2

_____ А.А.Мухитова

«___» _____ 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 «PlayTime» - готовой платформы, предназначенной для поиска единомышленников и организации совместных игр в настольные игры.

Актуальность данного проекта обусловлена растущей популярностью настольных игр как формы живого социального взаимодействия. Несмотря на этот тренд, потенциальным игрокам и организаторам часто не хватает удобного и специализированного инструмента для быстрого поиска компании, согласования времени и места проведения игр. Существующие решения, такие как социальные сети или мессенджеры, не предоставляют целостного сервиса, объединяющего календарь событий, систему бронирования и тематическое сообщество.

Целью проекта являлось создание полнофункционального веб-приложения, которое решает эти проблемы, предоставляя пользователям удобную и безопасную среду для организации своего досуга.

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 «PlayTime».

1 Техническое задание

1.1 Назначение разработки

Разработка веб-приложения «PlayTime» инициирована с целью решения конкретной проблемы – сложности в поиске компании и организации игровых встреч для любителей настольных игр. Существующие средства коммуникации (социальные сети, форумы) не предоставляли специализированного инструмента для этого, приводя к неэффективной трате времени и разрозненности сообщества.

Назначением проекта стало создание централизованной цифровой платформы, которая:

- автоматизирует процесс поиска игроков и организации событий;
- структурирует информацию о играх и предстоящих встречах;
- объединяет разрозненных пользователей в тематическое сообщество с удобными инструментами для взаимодействия.

Таким образом, разработанный продукт закрывает существующий пробел на рынке досуговых услуг, предоставляя целевой аудитории законченное и удобное решение для планирования своего хобби.

1.2 Требования к программе

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

1.2.1 технологический стек разработки:

- frontend: vue.js с использованием функциональных компонентов и хуков;
- backend: c# с фреймворком asp.net core для rest api, JavaScript для методов на клиентской стороне;
- база данных: mysql 8.0 с использованием entity framework core;

1.2.2 дизайн-решения и интерфейс:

- реализована светлая тема с акцентным цветом #ff596f и текстовыми оттенками #292a57;

- применён шрифт «Montserrat Alternates» в соответствии с утвержденной иерархией;

- адаптивный интерфейс с закругленными элементами и минималистичным дизайном;

- разработан логотип и иллюстрации, тематически связанные с настольными играми.

1.3 Требования к функциональным характеристикам

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

1.3.1 функциональные требования для неавторизованных пользователей;

1.3.1.1 просмотр календаря предстоящих публичных событий;

1.3.1.2 поиск событий с применением фильтров по:

- дате проведения;

- наименованию настольной игры;

- количеству игроков;

- ключевым словам в названии или описании события.

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.

1.3.2.2 требования к работе с календарем и поиску событий;

- использование расширенного поиска событий с применением фильтров;

- подача заявки на участие в выбранном событии;
- отмена ранее поданной заявки на участие в событии.

1.3.2.3 требования к организации событий;

- создание нового события через веб-интерфейс;
- отмена созданных пользователем событий.

1.3.3 функциональные требования для администратора системы;

1.3.3.1 модерация пользовательского контента:

- просмотр и отмена созданных пользователями событий;
- управление пользователями: просмотр, смена ролей.

1.3.3.2 управление справочной информацией;

- добавление новых наименований настольных игр в общедоступный каталог;
- редактирование описаний и атрибутов существующих в каталоге игр.

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 «PlayTime» 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 – это информация, поступающая в систему, чтобы она могла с ними работать.

К входной информации относятся:

- данные о пользователях;
- информация о настольных играх;
- заявки на участие в мероприятиях;

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
1	2	3	4
Подтверждение записи на событие	Сразу после подачи пользователем заявки на участие	1	Пользователю на почту
Отклонение записи на событие	При отклонении заявки пользователя организатором	1	Пользователю на почту
Отчёт о проведённых мероприятиях	При необходимости	1	Администратору системы
Отчёт об активности пользователей	При необходимости	1	Администратору системы

Шаблоны выходных документов представлены в таблицах 1.6.2 – 1.6.6.

Таблица 1.6.2 – Шаблон выходного документа «Подтверждение записи на событие»

Подтверждение записи на игротеку _____
Здравствуйте, _____!
Это автоматическое подтверждение того, что Ваша заявка на участие в игре была принята. Ждём Вас, чтобы разделить радость от любимых игр в отличной компании!
Подробная информация:
Событие: _____
Дата и время: _____, с: _____ до: _____

Продолжение таблицы 1.6.2

Место проведения: _____
Важный момент:
Количество мест ограничено. Если Вы понимаете, что не сможете прийти, пожалуйста, отмените запись через сайт или напишите напрямую организатору, чтобы Ваше место мог занять другой игрок.
До скорой встречи за игровым столом!
С уважением,
Команда PlayTime

Таблица 1.6.3 – Шаблон выходного документа «Отклонение записи на событие»

Отмена записи на событие « _____ » на __.__.____
Здравствуйте, _____!
К сожалению, событие « _____ » от __.__.____ было отменено.
На нашем сайте Вы также можете ознакомиться с другими мероприятиями. До новых встреч!
С уважением,
Команда PlayTime

Таблица 1.6.4 – Шаблон выходного документа «Отчёт о проведённых мероприятиях»

Отчёт о проведённых мероприятиях на платформе PlayTime		
Период отчета: с __.__.____ по __.__.____		
Дата формирования отчета: __.__.____		
Показатель	Значение	Динамика
Общее количество мероприятий	___	___%
Среднее кол-во игроков	___	___

Продолжение таблицы 1.6.4

Статусы мероприятий		
Показатель	Значение	Динамика
Успешно проведено	___ (%)	___
Отменено	___ (%)	___%
Топы и география		
Показатель	Значения	Комментарий
Топ 5 городов	1. 2. 3. 4. 5.	
Топ 5 игр	1. 2. 3. 4. 5.	

Таблица 1.6.5 – Шаблон выходного документа «Отчёт об активности пользователей»

Отчёт об активности пользователей PlayTime		
Период отчета: с ____:____:____ по ____:____:____		
Дата формирования отчета: ____:____:____		
Регистрация и активность		
Показатель	Значение	Динамика
Новые пользователи	___	___%
DAU (ежедневные)	___	___
WAU (еженедельные)	___	___
MAU (ежемесячные)	___	___

2 Разработка прототипа

Веб-приложение состоит из нескольких страниц:

– «главная». На странице находится баннер с общей информацией о проекте и навигация в шапке. Интерфейс страницы показан на рисунке 2.1. При нажатии на кнопку «Начать играть» открывается модальное окно с регистрацией и авторизацией. Внешний вид форм отображён на рисунках 2.2 – 2.3.

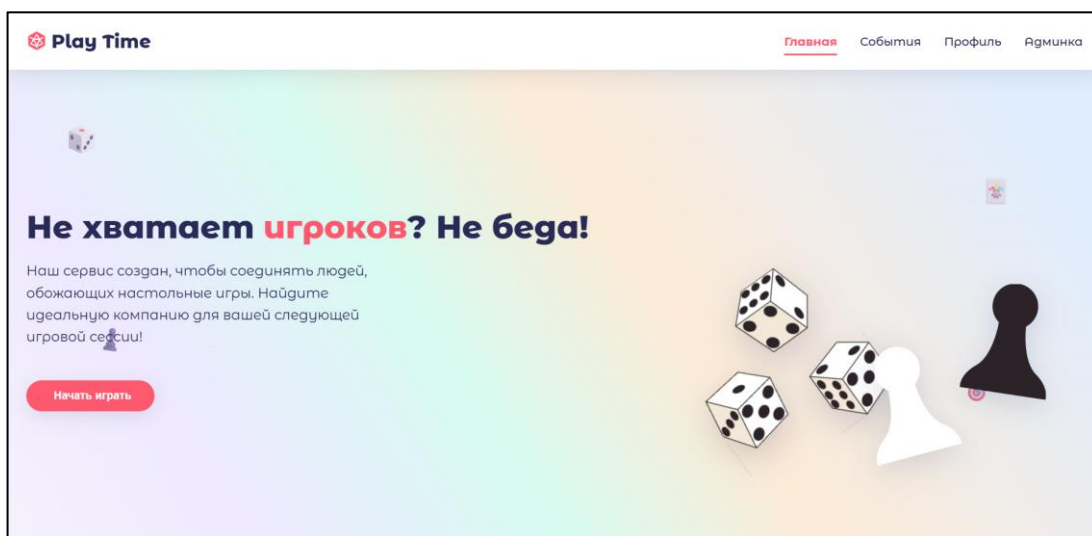


Рисунок 2.1 – Главная

The image shows a registration modal window with a dark blue border. The title is 'Присоединиться' in bold. Below the title, it says 'Создайте аккаунт чтобы начать играть'. The form contains four input fields: 'Имя' (Name), 'Email' (containing 'qwerty@yandex.ru'), 'Пароль' (Password, masked with dots), and 'Подтвердите пароль' (Confirm password). A red button labeled 'Создать аккаунт' is located below the password fields. At the bottom, there is a link 'Уже есть аккаунт? Войти в аккаунт'.

Рисунок 2.2 – Модальное окно с регистрацией

С возвращением!

Войдите в свой аккаунт

Email

qwerty@yandex.ru

Пароль

.....

☐ Запомнить меня [Забыли пароль?](#)

Войти

Еще нет аккаунта?

[Зарегистрироваться](#)

Рисунок 2.3 – Модальное окно с авторизацией

– «события». Страница с выводом всех активных событий и фильтрацией по городу, игре, дате, количеству игроков и названию/описанию. Пример представлен на рисунке 2.4. При нажатии на кнопку «Создать событие» открывается модальное окно с формой. Внешний вид формы отображён на рисунке 2.5.

Play Time Главная События Профиль Админка

Каталог событий

Найдите идеальное игровое событие для себя

[Создать событие](#)

город игра дата количество игроков

Все города Все игры gg.мм.гггг Любое количество

Поиск по названию события или описанию...

События не найдены

Попробуйте изменить параметры фильтрации

Рисунок 2.4 – События

Создание нового события

Название события (мин. 10 знаков) * Игра *

Дата и время * Место проведения *

Максимальное количество участников *

Описание события

Отменить Создать событие

Рисунок 2.5 – Модальное окно для создания нового события

– «профиль». На странице отображается личная информация пользователя с возможностью изменений, созданные им события и те, в которых он сам принимает участие. Пример интерфейса приведён на рисунке 2.6.

Play Time Главная События Профиль Админка

Личная информация
Основание данные профиля

Имя * Email *

Дата рождения * Пол *

Контактная информация *

О себе
Расскажите о своих интересах и увлечениях

Редактировать профиль

Мои созданные события 0 событий

Не удалось загрузить ваши созданные события

Предстоящие события 0 событий

Вы еще не записаны ни на одно событие
Найдите интересные события и присоединяйтесь!

Выход

Рисунок 2.6 – Профиль

– «админка». В разделе находится три вкладки – управление играми, событиями, пользователями. Примеры представлены на рисунках 2.7-2.9.

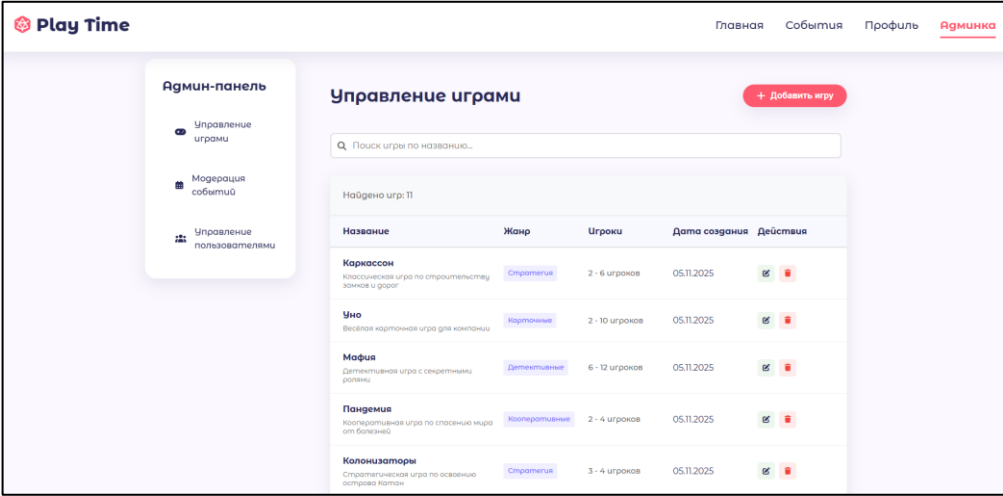


Рисунок 2.7 – Управление играми

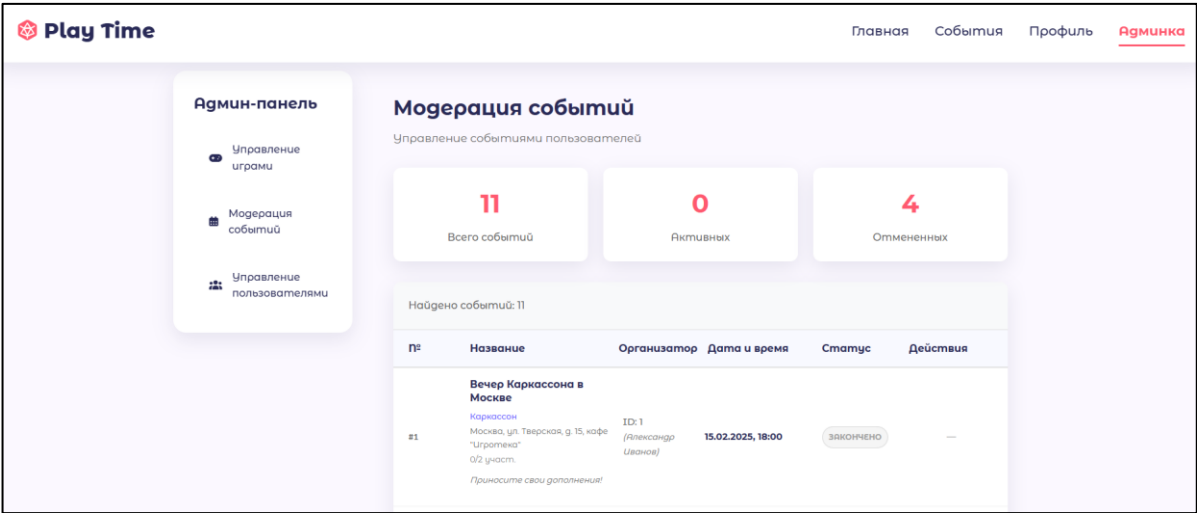


Рисунок 2.8 – Управление событиями

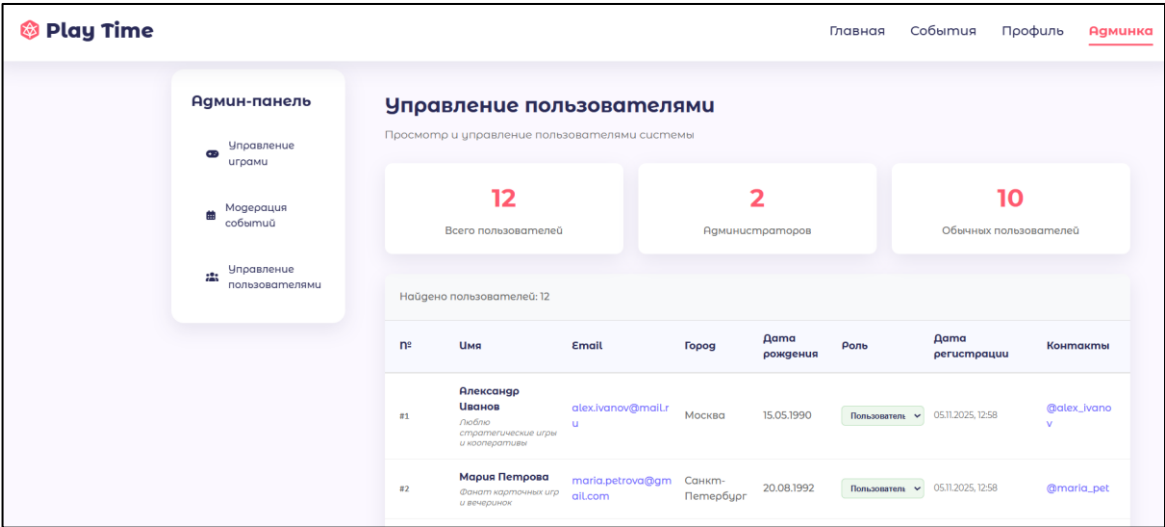


Рисунок 2.9 – Управление пользователями

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. Модульная схема проекта представлена на рисунке 4.1.

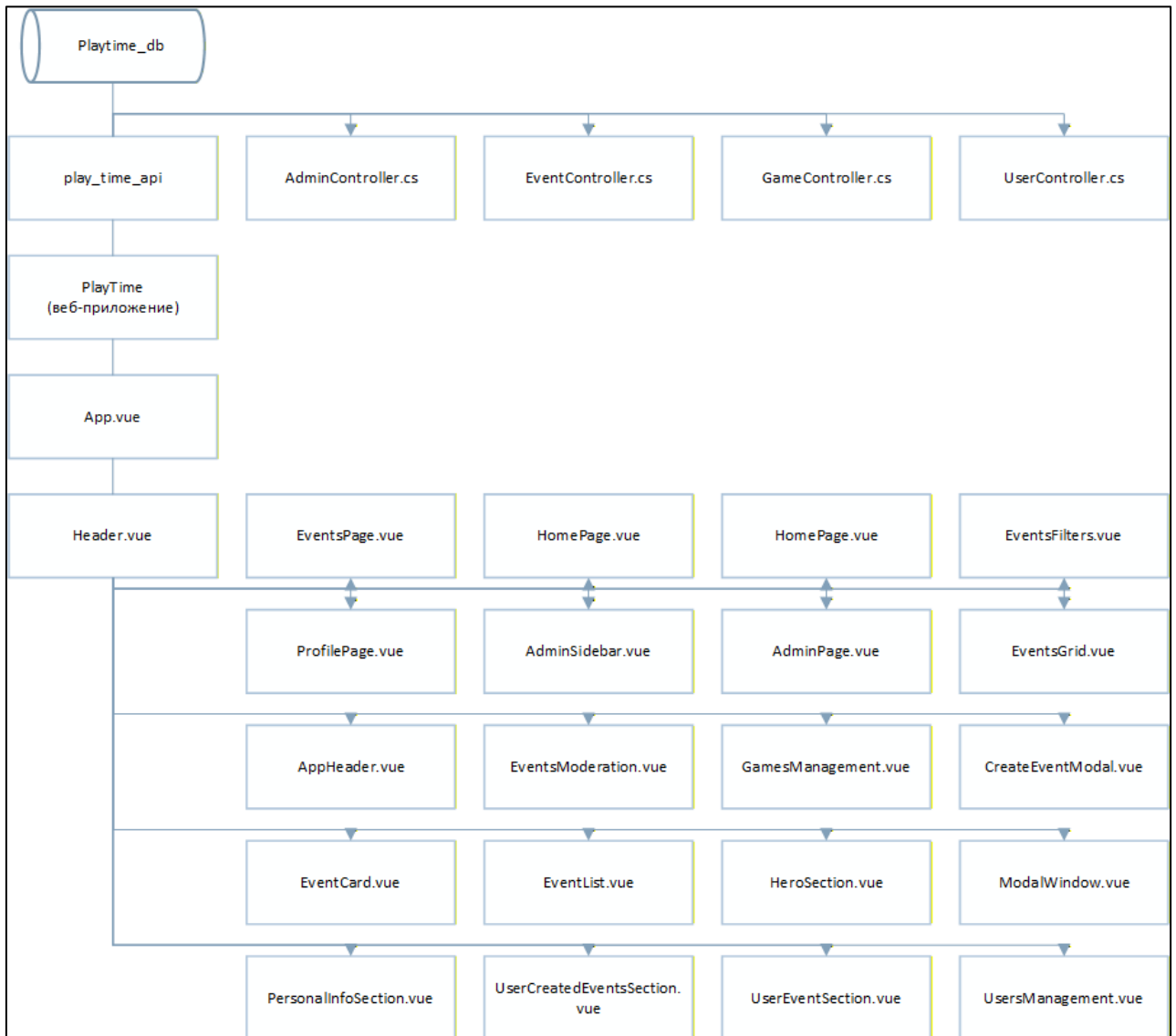


Рисунок 3.1 – Модульная схема

4 Описание запросов API

Разработанное API обеспечивает полный цикл взаимодействия с системой через эндпоинты. Взаимодействие построено вокруг основных сущностей: событий, игр, 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 (возвращаемый результат)	Код результата
Получение всех событий	-	GET /api/Event /GetEvent	[{ "eventId": 3, "organizerId": 1, "gameId": 1, "eventName": "имя", "eventDatetime": "2025-10-15T17:31:42", "location": "локация", "maxParticipants": 1, "description": "описание", "status": "Активно", "createdAt": "2025-01-01T00:00:00" }]	200

Продолжение таблицы 4.1

1	2	3	4	5
Редактирование события	idEvent, "eventDatetime": "2025-10-17T15:20:13.525Z", "eventName": "string", "location": "string", "description": "string", "maxParticipants": 0	PUT /api/Event/EditEvent?idEvent=3	«Событие успешно обновлено»	200
Редактирование события	idEvent, "eventDatetime": "2025-10-17T15:20:13.525Z", "eventName": "string", "location": "string", "description": "string", "maxParticipants": 0	PUT /api/Event/EditEvent?idEvent=10	«Событие не найдено.»	400
Присоединение к событию	eventid, userId	POST /api/Event/JoinEvent?eventId=4&userId=2	«Пользователь успешно записан на событие»	200

Продолжение таблицы 4.1

1	2	3	4	5
Присоединение к событию	eventId, userId	POST /api/Event /JoinEvent ?eventId=10&userId=2	«Событие не найдено.»	404
Присоединение к событию	eventId, userId	POST /api/Event /JoinEvent ?eventId=4&userId=2	«Пользователь уже записан на это событие.»	400
Присоединение к событию	eventId, userId	POST /api/Event /JoinEvent ?eventId=4&userId=10	«Пользователь не найден»	404
Присоединение к событию	eventId, userId	POST /api/Event /JoinEvent ?eventId=3&userId=2	«Достигнуто максимальное количество участников»	400
Присоединение к событию	eventId, userId	POST /api/Event /JoinEvent ?eventId=5&userId=2	«Нельзя записаться на неактивное событие.»	400

Продолжение таблицы 4.1

1	2	3	4	5
Создание события	"eventDatetime": "2025-10-17T15:24:23.591Z", "eventName": "string", "location": "string", "description": "string", "maxParticipants": 0, "gameId": 1, "organizerId": 3	POST /api/Event /AddEvents	«Событие создано»	200
Создание события	"eventDatetime": "2025-10-17T15:24:23.591Z", "eventName": "string", "location": "string", "description": "string", "maxParticipants": 0, "gameId": 50, "organizerId": 3	POST /api/Event /AddEvents	«Игра не найдена.»	400

Продолжение таблицы 4.1

1	2	3	4	5
Выход из события	eventId, userId	DELETE /api/Event /RemoveParticipant? eventId=4 &userId=2	«Пользователь успешно вышел из события»	200
Выход из события	eventId, userId	DELETE /api/Event /RemoveParticipant? eventId=10&userId=2	«Событие не найдено»	404
Выход из события	eventId, userId	DELETE /api/Event /RemoveParticipant? eventId=10&userId=15	«Пользователь не найден»	404
Выход из события	eventId, userId	DELETE /api/Event /RemoveParticipant? eventId=3 &userId=3	«Пользователь не записан на это событие»	400

Продолжение таблицы 4.1

1	2	3	4	5
Получение всех игр	-	GET /api/Game /GetGame s	[{ "gameId": 1, "genreId": 1, "gameName": "Каркассон", "minPlayers": 2, "maxPlayers": 5, "description": "Классическая игра по строительству замков и дорог", "createdAt": "2025-10-06T14:28:12", "events": [], "genre": null }]	200
Добавление игры	{ "gameName": "новая игра 2", "minPlayers": 1, "maxPlayers": 4, "description": "новое описание", "genreId": 1, "createdAt": "2025-10-17" }	POST /api/Game /AddGame	«Игра добавлена!»	200

Продолжение таблицы 4.1

1	2	3	4	5
Добавление игры	{ "gameName": "новая игра 2", "minPlayers": 1, "maxPlayers": 4, "description": "новое описание", "genreId": 1, "createdAt": "2025-10-17" }	POST /api/Game /AddGame	«Такая игра уже есть в базе данных.»	400
Поиск игры	name	GET /api/Game /SearchGame?name= %D0%B D%D0%B E%D0%B 2%D0%B 0%D1%8 F%20%D 0%B8%D 0%B3%D 1%80%D 0%B0%2 02	[{ "gameId": 6, "genreId": 1, "gameName": "новая игра 2", "minPlayers": 1, "maxPlayers": 4, "description": "новое описание", "createdAt": "2025-10-17T20:27:57", "events": [], "genre": null }]	200

Продолжение таблицы 4.1

1	2	3	4	5
Поиск игры	name	GET /api/Game /SearchGa me?name =%D0%B 8%D0%B 3%D1%8 0%D0%B 0'	«Игра не найдена.»	400
Получение всех пользовате лей	-	GET /api/User/ GetUsers	[{ "userId": 1, "name": "Алексей Иванов", "email": "alex@mail.com", "passwordHash": "hashed_password_1", "cityId": 1, "birthDate": "1995-03-15", "description": null, "contactInfo": "@alexey", "role": "Пользователь", "createdAt": "2025-10-06T14:28:05", "city": null, "eventParticipants": [], "events": [] },]	200

Продолжение таблицы 4.1

1	2	3	4	5
Авторизация	Email. password	POST /api/User/ Authorization?email =qwe&password=qwe	{ "userId": 4, "name": "qwe", "email": "qwe", "passwordHash": "qwe", "cityId": null, "birthDate": null, "description": null, "contactInfo": null, "role": "Пользователь", "createdAt": "2025-10-15T15:56:11", "city": null, "eventParticipants": [], "events": [] }	200
Авторизация	Email. password	POST /api/User/ Authorization?email =qwe&password=123	«Неверный логин или пароль»	404
Регистрация	{ "name": "string", "email": "string@yandex.ru", "password": "string", "confirmPassword": "string"	POST /api/User/ Registration	«Пользователь успешно зарегистрирован»	200

Продолжение таблицы 4.1

1	2	3	4	5
Регистрация	{ "name": "string", "email": "string@yandex.ru", "password": "string", "confirmPassword": "string" }	POST /api/User/ Registration	«Такая почта уже существует.»	400
Регистрация	{ "name": "string", "email": "asdasd@yandex.ru", "password": "string", "confirmPassword": "123" }	POST /api/User/ Registration	«Пароль и подтверждение пароля должны совпадать.»	400

Продолжение таблицы 4.1

1	2	3	4	5
Обновление данных	<pre> userId, { "name": "string", "email": "string", "cityId": 1, "birthDate": "2025-10-17", "description": "string", "contactInfo": "string", "currentPassword": "string", "newPassword": "string", "confirmNewPassword": "string" } </pre>	<pre> PUT /api/User/ UpdateProfile/6 </pre>	«Профиль успешно обновлён»	200

Продолжение таблицы 4.1

1	2	3	4	5
Обновление данных	<pre> userId, { "name": "string", "email": "alex@mail.co m", "cityId": 1, "birthDate": "2025-10-17", "description": "string", "contactInfo": "string", "currentPasswo rd": "string", "newPassword ": "string", "confirmNewP assword": "string" } </pre>	<pre> PUT /api/User/ UpdatePro file/6 </pre>	«Пользователь с таким email уже существует»	400

Продолжение таблицы 4.1

1	2	3	4	5
Обновление данных	<pre> userId, { "name": "string", "email": "asdasx@mail. com", "cityId": 1, "birthDate": "2025-10-17", "description": "string", "contactInfo": "string", "currentPassword": "123", "newPassword": ": "string", "confirmNewPassword": "string" } </pre>	<pre> PUT /api/User/ UpdateProfile/6 </pre>	«Текущий пароль неверен»	400

Продолжение таблицы 4.1

1	2	3	4	5
Обновление данных	<pre> userId, { "name": "string", "email": "asdasx@mail. com", "cityId": 1, "birthDate": "2025-10-17", "description": "string", "contactInfo": "string", "currentPassword": "string", "newPassword": "123", "confirmNewPassword": "string" } </pre>	<pre> PUT /api/User/ UpdateProfile/6 </pre>	«Новый пароль и подтверждение не совпадают»	400

Продолжение таблицы 4.1

1	2	3	4	5
Получение данных по идентификатору пользователя	userId	GET /api/User/ GetProfile /6	{ "userId": 6, "name": "string", "email": "string", "passwordHash": "RzKH+CmNunFjqJeQiVj3wOrnM+Jd LgJ5kuou3JvtL6g=", "cityId": 1, "birthDate": "2025-10-17", "description": "string", "contactInfo": "string", "role": "Пользователь", "createdAt": "2025-10-17T20:32:57", "city": null, "eventParticipants": [], "events": [] }	200
Получение данных по идентификатору пользователя	userId	GET /api/User/ GetProfile /15	«Пользователь не найден»	404
Получить события, в которых участвует выбранный пользователь	userId	GET /api/Event /GetEvent ByUser/1 9	«Пользователь не найден»	404

Продолжение таблицы 4.1

1	2	3	4	5
Получить события, в которых участвует выбранный пользователь	userId	GET /api/Event /GetEvent ByUser/1 3	«Пользователь не записан ни на одно активное событие»	404
Получить события, созданные конкретным пользователем	userId	GET /api/Event /GetEvent sCreatedB yUser/13	«Пользователь не создал ни одного события»	404
Смена роли пользователя	userId, role	PUT /api/Admin/Change Role/4534 ?role=Ад министратор	«Пользователь не найден»	404

Продолжение таблицы 4.1

1	2	3	4	5
Смена роли пользовате ля	userId, role	PUT /api/Admin/ Change Role/4534 ?role=Ад министра тор /13?role= Админис тратор	«Роль успешно сменена»	200

5 Результаты работы

Настоящее руководство пользователя предназначено для ознакомления с основными функциями веб-приложения «PlayTime» и последовательностью действий, необходимых для его корректной и эффективной эксплуатации. В руководстве описаны интерфейсы, действия и процессы, доступные для различных ролей в системе.

Веб-приложение ориентировано на следующие категории пользователей:

- клиент, имеющий возможность вступать или создавать собственные события, редактировать личные данные;
- администратор, модерирующий события, игры и пользователей.

Каждой роли соответствует собственный уровень доступа, от которого зависят функциональные возможности и доступ к разделам системы.

Руководство пользователя для клиента.

Для корректной работы веб-приложения необходимо выполнить следующие шаги в указанной последовательности:

а) запуск серверной части (API);

- откройте решение `play_time_api` в среде разработки Visual Studio;
- запустите проект на <http://localhost:5119/swagger/index.html>;

б) запуск клиентской части;

- откройте среду разработки Visual Studio Code;
- в открывшейся среде откройте папку проекта PlayTime;
- для открытия встроенного терминала используйте сочетание клавиш `Ctrl + ``;
- в терминале выполните команду `npm i` для установки всех необходимых зависимостей проекта;
- после успешной установки выполните команду `npm run dev` для запуска клиентского приложения в режиме разработки;
- при завершении сборки перейдите по ссылке <http://localhost:3000/>;

После запуска веб-приложения происходит переход на главную страницу. В верхней части интерфейса располагается навигационное меню, включающее ссылки на разделы: «Главная», «События», «Профиль». Страница изображена на рисунке 5.

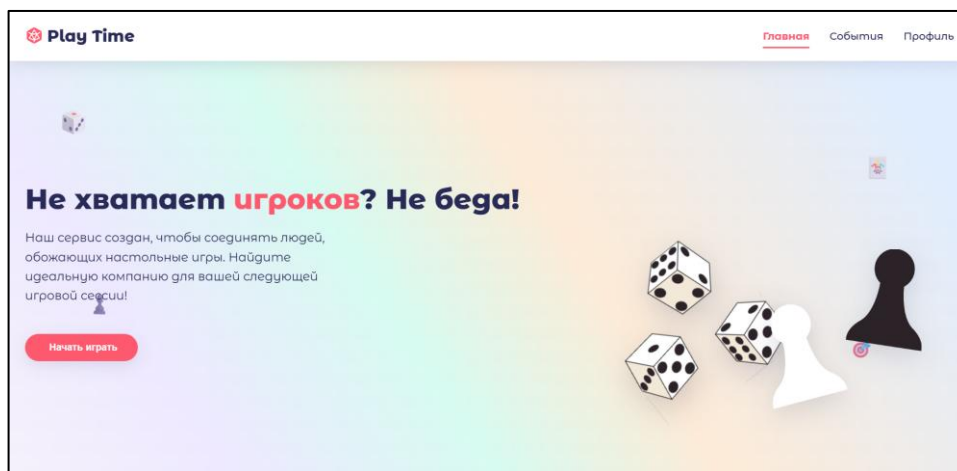


Рисунок 5.1 – Главная страница

Для начала работы с сервисом требуется регистрация. Для регистрации необходимо на главной странице нажать на кнопку «Начать играть», ввести корректные данные во все поля и нажать на кнопку «Создать аккаунт». Пример представлен на рисунке 5.2.

The screenshot shows a registration form on a dark blue background. At the top, a black notification box displays the message "Сообщение с localhost:3000: Пользователь успешно зарегистрирован" (Message from localhost:3000: User successfully registered) with an "OK" button. Below the notification, the form has a heading "Создайте аккаунт чтобы начать играть" (Create an account to start playing). The form contains four input fields: "Имя" (Name) with the value "Руслан", "Email" with the value "ruslan@yandex.ru", "Пароль" (Password) with masked characters "*****", and "Подтвердите пароль" (Confirm password) with masked characters "*****". A red button labeled "Создать аккаунт" (Create account) is located below the password fields. At the bottom of the form, there is a link "Уже есть аккаунт? Войти в аккаунт" (Already have an account? Log in to account).

Рисунок 5.2 – Регистрация с корректными полями

После успешной регистрации необходимо войти в аккаунт. Для этого требуется ещё раз нажать на кнопку «Начать играть», перейти во вкладку «Войти в аккаунт», ввести данные, с которыми происходила регистрация, и нажать на кнопку «Войти». Пример формы представлен на рисунке 5.3.

Сообщение с localhost:3000:
Вход выполнен успешно!

С возвращением!
Войдите в свой аккаунт

Email
ruslan@yandex.ru

Пароль
.....

☐ Запомнить меня [Забыли пароль?](#)

Войти

Еще нет аккаунта?
[Зарегистрироваться](#)

Рисунок 5.3 – Успешная авторизация

Все активные события доступны в разделе «События». Чтобы туда попасть, надо нажать на соответствующую ссылку в шапке веб-приложения. Интерфейс страницы показан на рисунке 5.4.

Play Time Главная **События** Профиль

Каталог событий
Найдите идеальное игровое событие для себя

[Создать событие](#)

город: Все города игра: Все игры дата: dd.MM.yyyy количество игроков: Любое количество

Поиск по названию события или описанию...

Игра	Участников	Место	Дата и время проведения	Организатор
Каркассон	1/2 участников	Москва, ул. Тверская, д. 15, кафе "Уютная"	15 февраля 2025 г. в 18:00	Александр Иванов
Уно	4/12 участников	Санкт-Петербург, Невский пр., д. 25, коворкинг "Пространство"	20 февраля 2025 г. в 19:00	Мария Петрова
Мафия	5/15 участников	Новосибирск, ул. Ленина, д. 10, антикафе "Время игр"	25 февраля 2025 г. в 20:00	Дмитрий Сидоров

Рисунок 5.4 – Страница «События»

Можно отфильтровать события по городу, игре, дате и количеству игроков. Для этого надо поставить нужные фильтры. Пример представлен на рисунке 5.5.

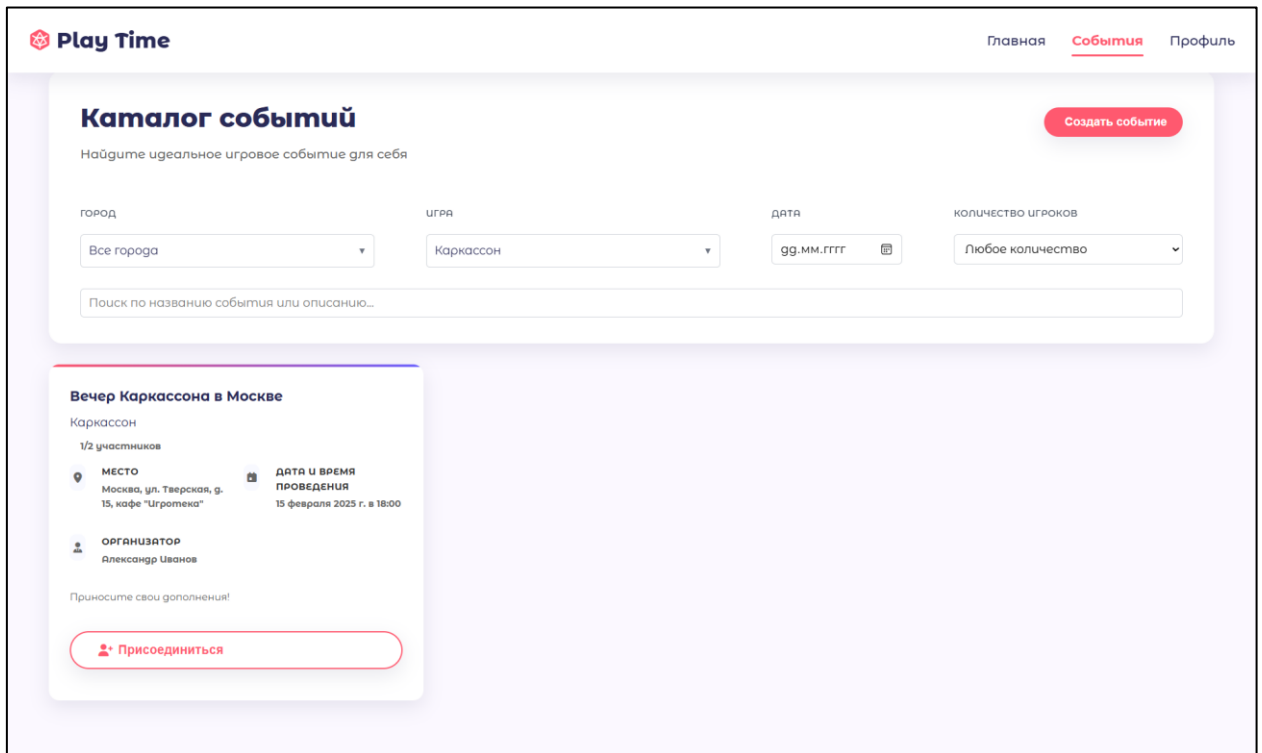


Рисунок 5.5 – Вывод событий с фильтром

Для участия в событии необходимо нажать на кнопку «Присоединиться» в карточке понравившегося события. Далее открывается уведомление с подтверждением присоединения. После нажатия кнопки «Ок» производится запись на событие. Пример интерфейса приведён на рисунке 5.6.

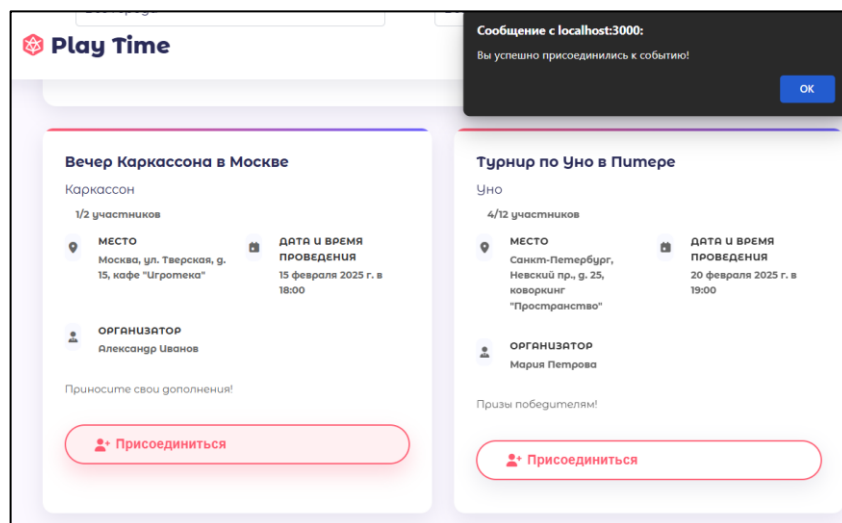


Рисунок 5.6 – Успешное присоединение к событию

Также предусмотрено создание нового события. Для этого на вкладке «События» надо нажать на кнопку «Создать событие», заполнить все поля корректными данными и нажать на соответствующую кнопку, после чего появится уведомление об успешном создании. Пример представлен на рисунке 5.7.

The image shows a web application interface for creating an event. At the top, a dark modal dialog box displays the message "Сообщение с localhost:3000: Событие успешно создано!" (Message from localhost:3000: Event successfully created!) with an "OK" button. Below the dialog, the event creation form is visible. It includes the following fields:

- Название события (мин. 10 знаков) ***: Text input with "Игра 1".
- Игра ***: Dropdown menu with "Каркассон" selected.
- Дата и время ***: Date and time picker showing "21.12.2025 12:32".
- Место проведения ***: Text input with "Уфа, Кирова".
- Максимальное количество участников ***: Text input with "6".
- Описание события**: Text area with placeholder text "Опишите ваше событие, правила, особенности...".

At the bottom right of the form, there are two buttons: "Отменить" (Cancel) and "Создание..." (Create...).

Рисунок 5.7 – Успешное создание события

Для просмотра всех событий, связанных с пользователем, необходимо перейти в раздел «Профиль» в верхней шапке веб-приложения. Демонстрация интерфейса приведена на рисунке 5.8.

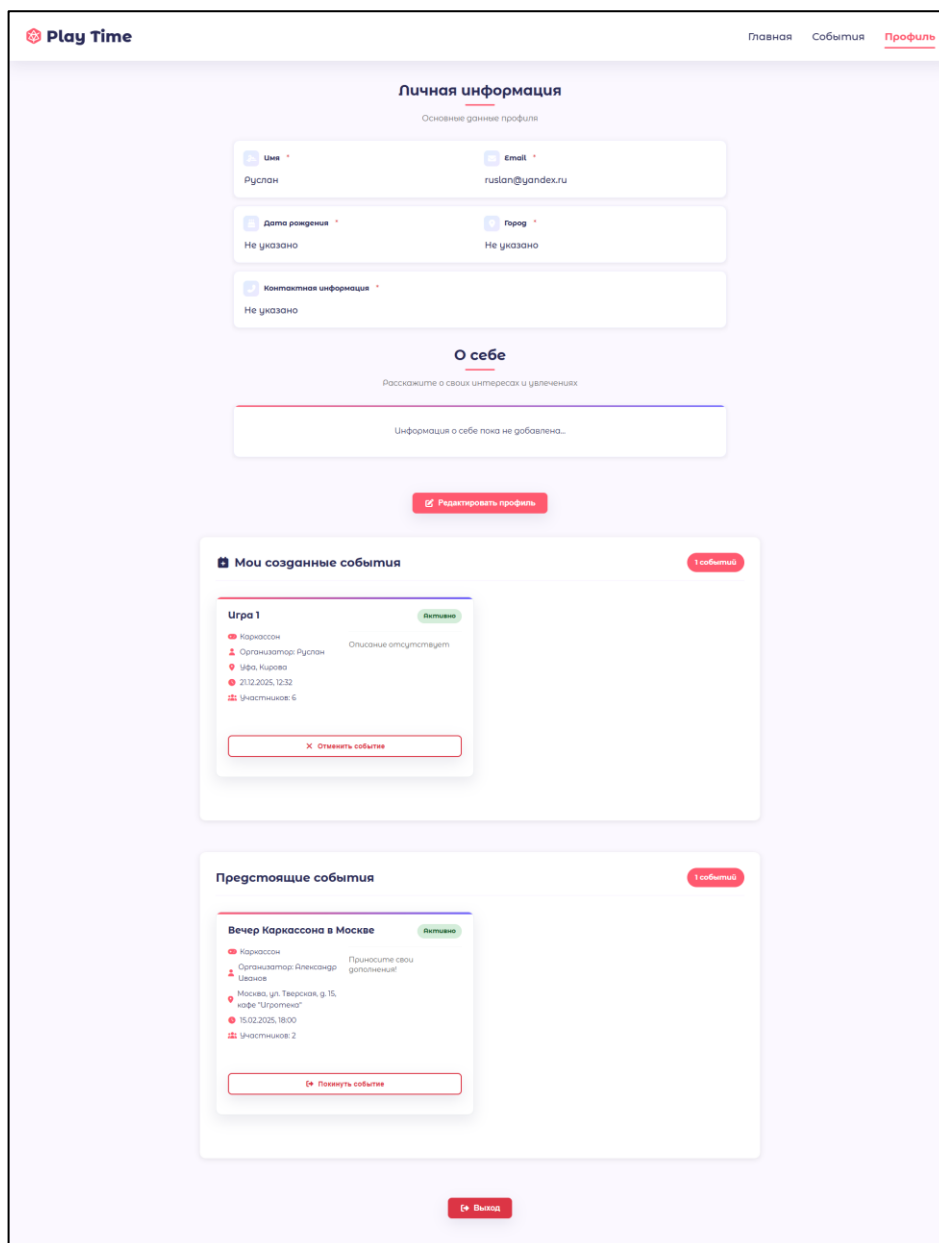


Рисунок 5.8 – Раздел «Профиль»

Созданное событие можно отменить нажатием на кнопку «Отменить событие» в списке «Мои созданные события». Появится окно с подтверждением, и при согласии событие поменяет свой статус с «Активно» на «Отменено». Пример окна показан на рисунке 5.9.

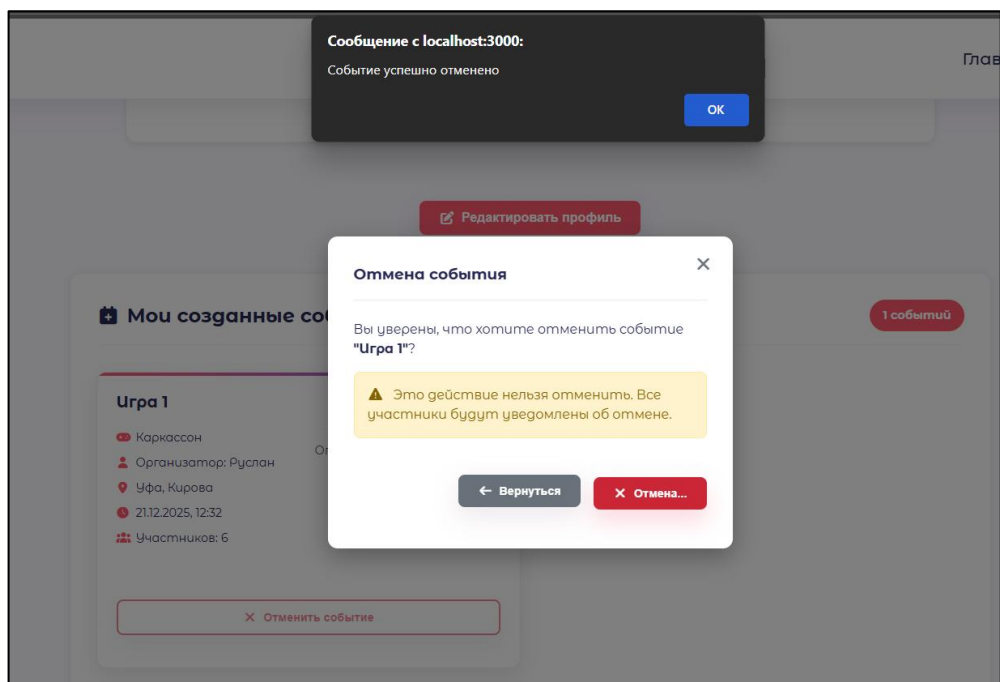


Рисунок 5.9 – Отмена созданного события

Также предусмотрен выход из события нажатием на кнопку «Покинуть событие» в списке «Предстоящие события». После подтверждения событие так же пропадёт из списка. Пример представлен на рисунке 5.10.

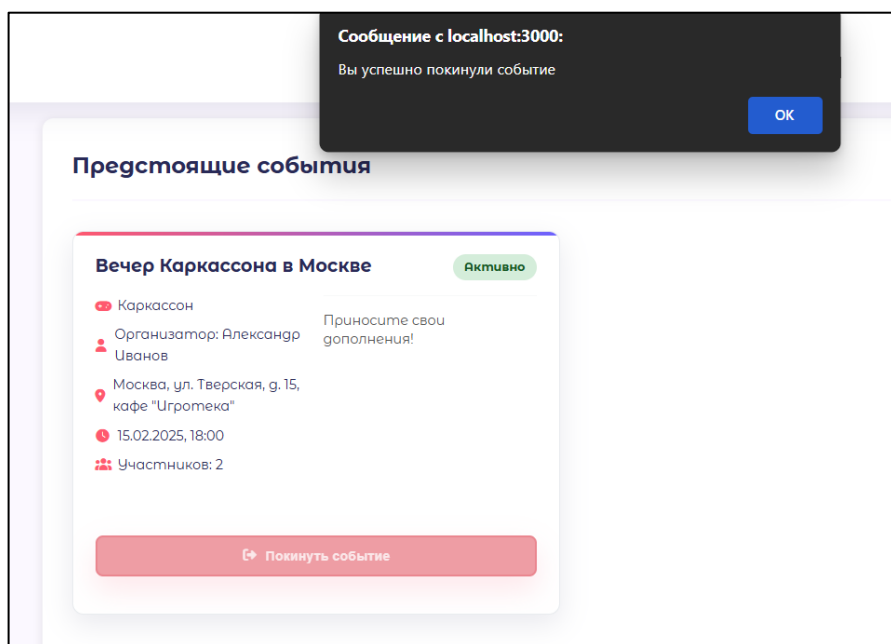


Рисунок 5.10 – Выход из события

В данном разделе доступно редактирование личных данных. Для редактирования необходимо нажать на кнопку «Редактировать профиль»,

ввести новые данные и нажать на «Сохранить изменения». Внешний вид профиля показан на рисунке 5.11.

Личная информация

Основные данные профиля

Имя *
ruslan

Email *
ruslan@yandex.ru

Дата рождения *
16.09.2000

Город *
Уфа

Контактная информация *
Мой номер телефона: +79864839211

О себе

Расскажите о своих интересах и увлечениях

Люблю играть в настольные игры и спать !

Рисунок 5.11 – Изменение личной информации в профиле
Руководство пользователя для администратора

Администратии доступны все те же возможности, что и пользователю. Только у него ещё появляется раздел «Админка» в шапке веб-приложения.

Требуется авторизация под учетной записью с ролью «Администратор». Перейти в раздел «Админка». Пример представлен на рисунке 5.12.

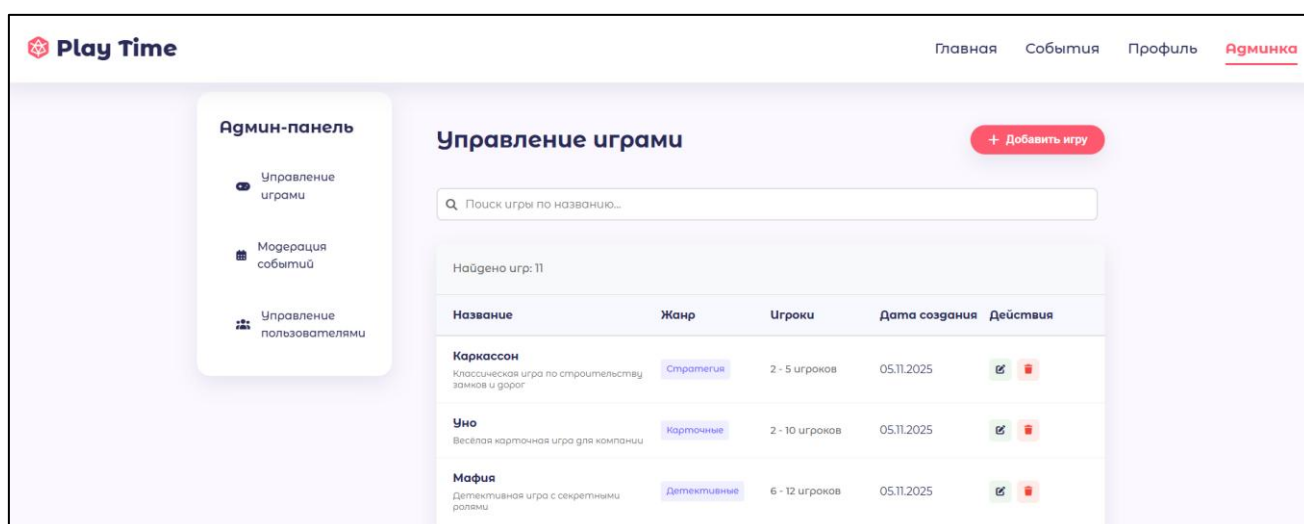
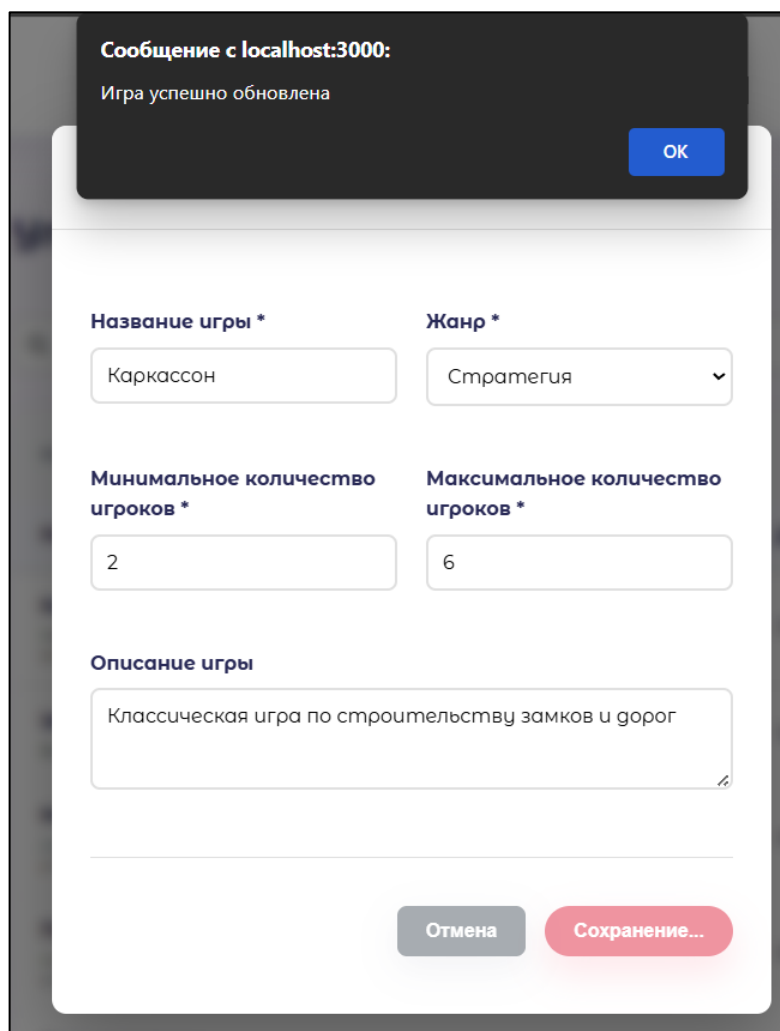


Рисунок 5.12 – Раздел «Админка»

Первой вкладкой раздела является «Управление играми». Здесь можно просматривать все игры, редактировать их, удалять и создавать новые. Для редактирования необходимо нажать на соответствующую кнопку в карточке игры, изменить нужные поля и нажать на кнопку «Сохранить изменения». Внешний вид формы отображён на рисунке 5.13.



The image shows a web application interface for editing a game. At the top, a dark notification box displays the message "Сообщение с localhost:3000: Игра успешно обновлена" (Message from localhost:3000: Game successfully updated) with an "OK" button. Below this is the main form for editing the game. It contains several input fields: "Название игры *" (Game name) with the value "Каркассон" (Carcassonne), "Жанр *" (Genre) with a dropdown menu showing "Стратегия" (Strategy), "Минимальное количество игроков *" (Minimum number of players) with the value "2", and "Максимальное количество игроков *" (Maximum number of players) with the value "6". There is also a text area for "Описание игры" (Game description) containing the text "Классическая игра по строительству замков и дорог" (Classic game about building castles and roads). At the bottom right of the form are two buttons: "Отмена" (Cancel) and "Сохранение..." (Save...).

Рисунок 5.13 – Редактирование игры

Для удаления игры необходимо нажать на соответствующую кнопку и подтвердить действие. Пример представлен на рисунке 5.14.

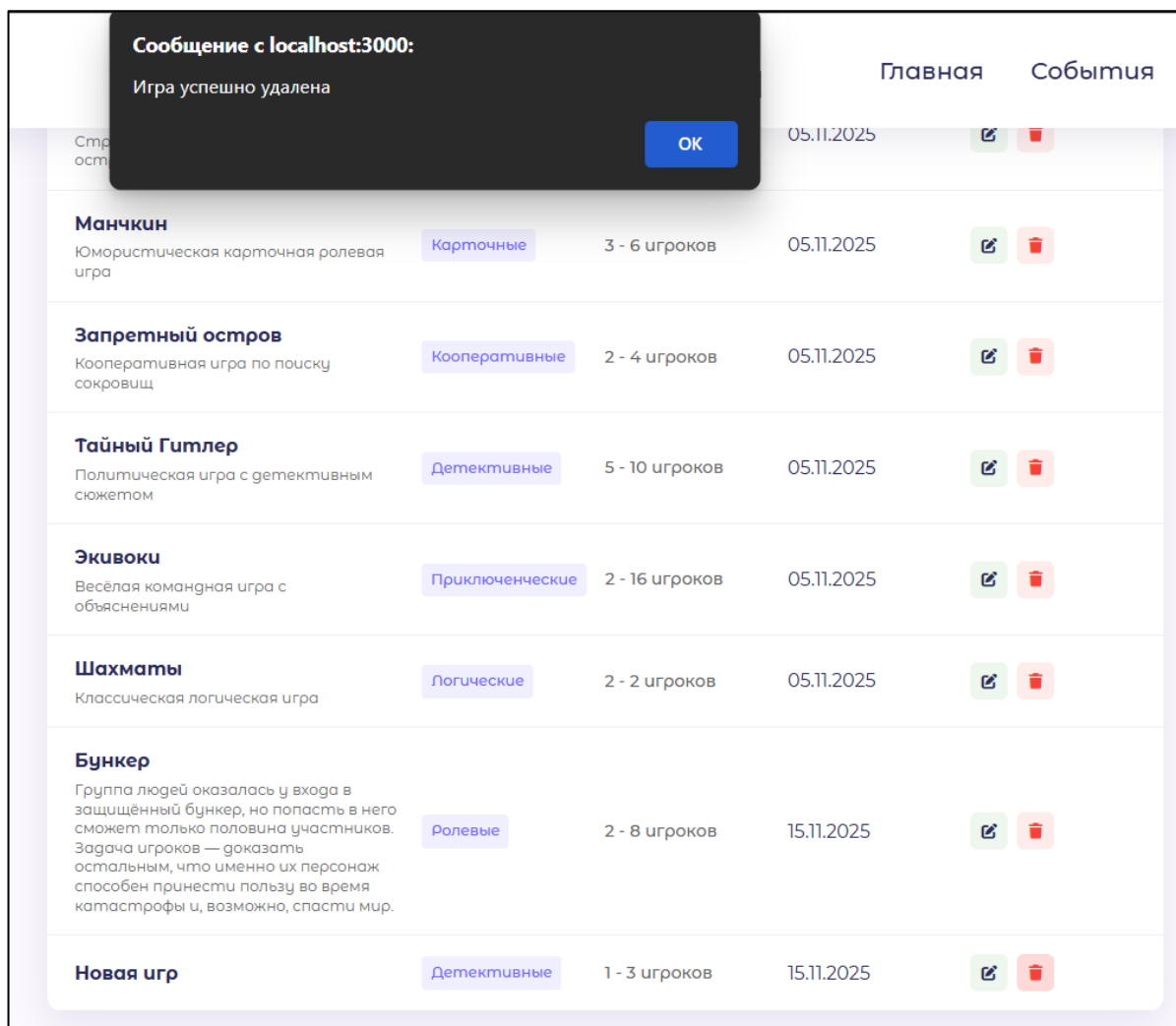


Рисунок 5.14 – Удаление игры

Во вкладке «Модерация событий» отображается небольшая статистика. Интерфейс страницы показан на рисунке 5.15.

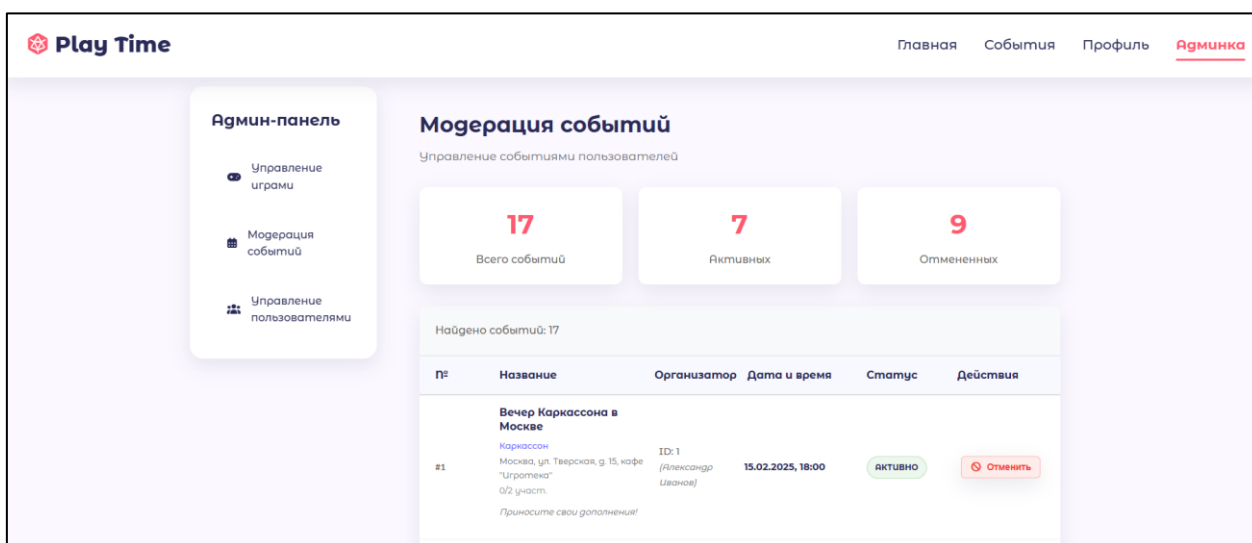


Рисунок 5.15 – вкладка «Модерация событий»

При нажатии кнопки «Отменить» у события и подтверждении действия статус события меняется с «Активно» на «Отменено». Пример представлен на рисунке 5.16.

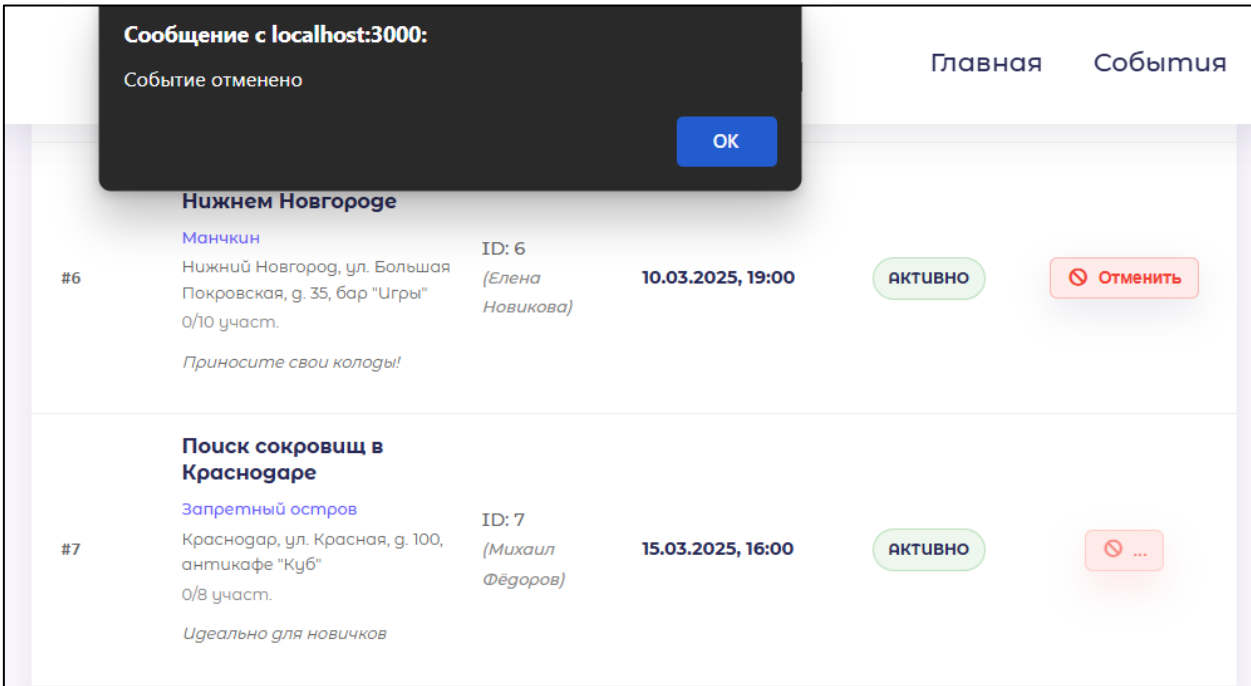


Рисунок 5.16 – Отмена события во вкладке «Модерация событий»

Последней вкладкой является «Управление пользователями». В данном разделе отображается небольшая статистика и список всех пользователей системы, а также их роли. Пример страницы показан на рисунке 5.17.

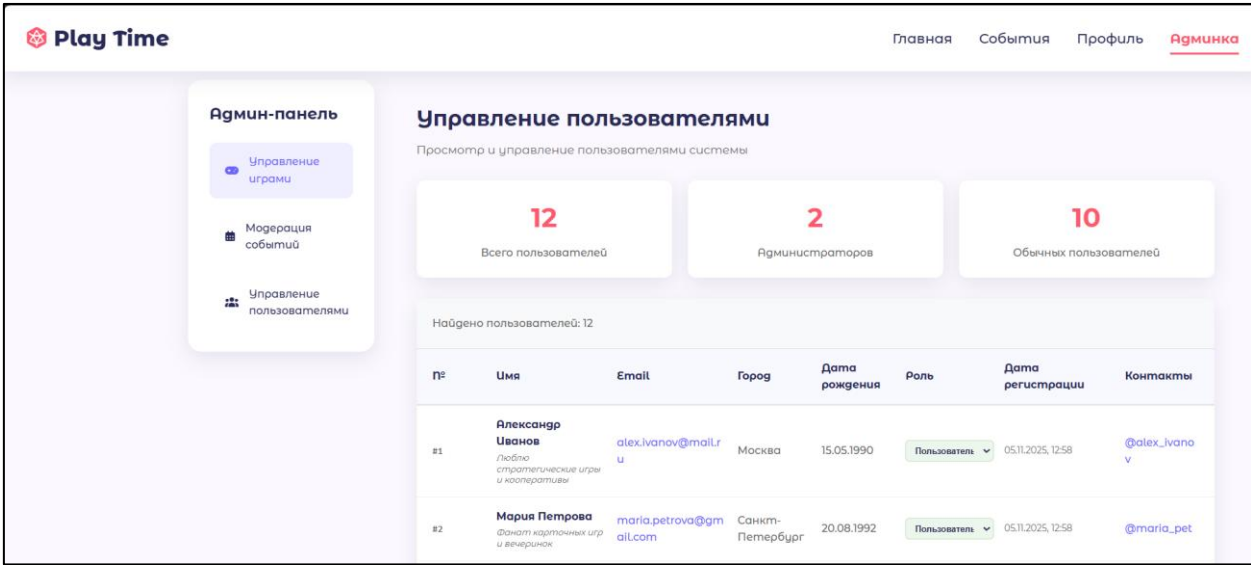


Рисунок 5.17 – Вкладка «Управление пользователями»

В данном разделе доступно изменение роли пользователей, выбрав подходящую в выпадающем списке конкретного пользователя. Пример представлен на рисунке 5.18.

#16	Руслан <i>Люблю играть в настольные игры и спать!</i>	ruslan@yandex.ru	Уфа	16.09.2000	<div>Пользователь ▾</div> <div>Пользователь</div> <div>Администратор</div>	15.11.2025, 02:12	Мой номер телефона: + 79864839211
-----	---	------------------	-----	------------	--	-------------------	---

Рисунок 5.18 – Смена роли у пользователя

ЗАКЛЮЧЕНИЕ

В ходе учебной практики была успешно выполнена задача по разработке полнофункционального веб-приложения «PlayTime», предназначенного для организации совместных игр в настольные игры. 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 MySQL;
- разработан полный жизненный цикл взаимодействия пользователя с платформой от регистрации и поиска событий с помощью гибкой системы фильтров до создания собственных мероприятий и управления участием в них;
- создан интуитивно понятный и адаптивный пользовательский интерфейс, соответствующий современным стандартам UI/UX-дизайна;
- реализована система ролей, обеспечивающая разделение функционала между обычными пользователями и администраторами системы. Администраторам предоставлены инструменты для модерации контента, управления каталогом игр и пользователями;

Практическая значимость проекта заключается в решении актуальной проблемы организации досуга, объединения разрозненных любителей настольных игр в единое сообщество и предоставления им удобного специализированного инструмента.

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 «PlayTime» считается завершенным и готовым к дальнейшему развитию. В качестве перспектив развития платформы можно выделить внедрение системы оповещений (push-уведомлений), разработку мобильного приложения и добавление системы рейтингов и отзывов об участниках.