

# API Документация - BaiMuras Platform

## Обзор

BaiMuras Platform предоставляет RESTful API для интеграции с внешними системами и управления данными. API поддерживает JSON формат данных и использует JWT токены для аутентификации.

**Базовый URL:** `https://baimuras.space/api`

**Версия API:** v1

**Формат данных:** JSON

**Аутентификация:** JWT Bearer Token

## Аутентификация

### JWT Токены

API использует JWT (JSON Web Tokens) для аутентификации. Токен должен быть включен в заголовок `Authorization` каждого запроса:

```
Authorization: Bearer <your-jwt-token>
```

### Получение токена

**POST** `/api/v1/auth/login`

```
{
  "email": "user@example.com",
  "password": "password123"
}
```

**Ответ:**

```
{
  "access_token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9...",
  "refresh_token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9...",
  "user": {
    "id": 1,
    "email": "user@example.com",
    "name": "John Doe"
  }
}
```

### Обновление токена

**POST** `/api/v1/auth/refresh`

```
Authorization: Bearer <refresh-token>
```

**Ответ:**

```
{
  "access_token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9..."
}
```

## Выход из системы

**POST** /api/v1/auth/logout

```
Authorization: Bearer <access-token>
```

## Основные эндпоинты

### 1. Публичные маршруты

#### Главная страница

**GET** /

- Возвращает главную страницу сайта
- Не требует аутентификации

#### Портфолио

**GET** /portfolio

- Отображает портфолио проектов
- Не требует аутентификации

#### Контактная форма

**POST** /contact

```
{
  "name": "Иван Иванов",
  "email": "ivan@example.com",
  "phone": "+7 777 123 45 67",
  "message": "Интересует изготовление кухни"
}
```

#### Запрос консультации

**POST** /api/consultation

```
{
  "name": "Мария Петрова",
  "email": "maria@example.com",
  "phone": "+7 777 987 65 43",
  "service_type": "kitchen_design",
  "preferred_date": "2025-07-01",
  "message": "Хочу заказать дизайн кухни"
}
```

### 2. Управление пользователями

#### Получить профиль пользователя

**GET** /api/v1/auth/profile

```
Authorization: Bearer <access-token>
```

Ответ:

```
{
  "id": 1,
  "email": "user@example.com",
  "name": "John Doe",
  "role": "admin",
  "created_at": "2025-01-01T00:00:00Z",
  "last_login": "2025-06-21T10:30:00Z"
}
```

## Список пользователей

GET /api/v1/users

Параметры запроса:

- page (int): Номер страницы (по умолчанию: 1)
- per\_page (int): Количество записей на странице (по умолчанию: 20)
- role (string): Фильтр по роли

Ответ:

```
{
  "users": [
    {
      "id": 1,
      "email": "admin@baimuras.space",
      "name": "Admin User",
      "role": "admin",
      "created_at": "2025-01-01T00:00:00Z"
    }
  ],
  "pagination": {
    "page": 1,
    "pages": 5,
    "per_page": 20,
    "total": 100
  }
}
```

## Создать пользователя

POST /api/v1/users

```
{
  "email": "newuser@example.com",
  "name": "New User",
  "password": "securepassword123",
  "role": "user"
}
```

### Получить пользователя

GET /api/v1/users/{user\_id}

### Обновить пользователя

PUT /api/v1/users/{user\_id}

```
{
  "name": "Updated Name",
  "role": "manager"
}
```

### Удалить пользователя

DELETE /api/v1/users/{user\_id}

## 3. Управление лидами

### Список лидов

GET /api/v1/leads

#### Параметры запроса:

- page (int): Номер страницы
- per\_page (int): Количество записей на странице
- status (string): Фильтр по статусу ( new , contacted , qualified , converted )
- source (string): Фильтр по источнику

#### Ответ:

```
{
  "leads": [
    {
      "id": 1,
      "name": "Анна Смирнова",
      "email": "anna@example.com",
      "phone": "+7 777 111 22 33",
      "status": "new",
      "source": "website",
      "service_interest": "kitchen_design",
      "created_at": "2025-06-21T09:00:00Z",
      "notes": "Интересуется современным дизайном кухни"
    }
  ],
  "pagination": {
    "page": 1,
    "pages": 3,
    "per_page": 20,
    "total": 45
  }
}
```

### Создать лид

POST /api/v1/leads

```
{
  "name": "Петр Сидоров",
  "email": "petr@example.com",
  "phone": "+7 777 444 55 66",
  "source": "social_media",
  "service_interest": "custom_furniture",
  "notes": "Заинтересован в изготовлении шкафа-купе"
}
```

### Получить лид

**GET** /api/v1/leads/{lead\_id}

### Обновить лид

**PUT** /api/v1/leads/{lead\_id}

```
{
  "status": "contacted",
  "notes": "Связались по телефону, назначена встреча на завтра"
}
```

### Конвертировать лид в проект

**POST** /api/v1/leads/{lead\_id}/convert

```
{
  "project_name": "Кухня для семьи Сидоровых",
  "budget": 500000,
  "deadline": "2025-08-15"
}
```

## 4. Webhook эндпоинты

### n8n Webhook

**POST** /webhooks/n8n

Принимает данные от системы автоматизации n8n.

```
{
  "event_type": "lead_created",
  "data": {
    "lead_id": 123,
    "name": "Новый клиент",
    "email": "client@example.com"
  }
}
```

### Автоматизация лидов

**POST** /webhooks/lead-automation

Обрабатывает автоматические действия с лидами.

Напоминания о консультациях

POST /webhooks/consultation-reminder

Отправляет напоминания о предстоящих консультациях.

Обновление статуса проекта

POST /webhooks/project-status

Обновляет статус проектов из внешних систем.

5. Health Check

Проверка состояния API

GET /api/health

Ответ:

```
{
  "status": "healthy",
  "timestamp": "2025-06-21T12:00:00Z",
  "version": "2.0.0",
  "database": "connected",
  "redis": "connected"
}
```

GET /health

Базовая проверка состояния приложения.

Коды ответов

Код	Описание
200	OK - Запрос выполнен успешно
201	Created - Ресурс создан
400	Bad Request - Неверный запрос
401	Unauthorized - Требуется аутентификация
403	Forbidden - Недостаточно прав
404	Not Found - Ресурс не найден
422	Unprocessable Entity - Ошибка валидации
429	Too Many Requests - Превышен лимит запросов
500	Internal Server Error - Внутренняя ошибка сервера

## Ограничения запросов

API использует rate limiting для предотвращения злоупотреблений:

- **Общий лимит:** 1000 запросов в час на IP
- **Аутентификация:** 10 попыток входа в минуту
- **Создание ресурсов:** 100 запросов в час на пользователя

При превышении лимита возвращается код 429 с заголовками:

```
X-RateLimit-Limit: 1000
X-RateLimit-Remaining: 0
X-RateLimit-Reset: 1640995200
```

## Обработка ошибок

Все ошибки возвращаются в стандартном формате:

```
{
  "error": {
    "code": "VALIDATION_ERROR",
    "message": "Ошибка валидации данных",
    "details": {
      "email": ["Поле email обязательно для заполнения"],
      "phone": ["Неверный формат номера телефона"]
    }
  }
}
```

## Примеры использования

### Python (requests)

```
import requests

# Аутентификация
auth_response = requests.post('https://baimuras.space/api/v1/auth/login', json={
    'email': 'admin@baimuras.space',
    'password': 'your-password'
})
token = auth_response.json()['access_token']

# Получение списка лидов
headers = {'Authorization': f'Bearer {token}'}
leads_response = requests.get('https://baimuras.space/api/v1/leads', headers=headers)
leads = leads_response.json()['leads']

# Создание нового лида
new_lead = {
    'name': 'Тестовый клиент',
    'email': 'test@example.com',
    'phone': '+7 777 123 45 67',
    'source': 'api',
    'service_interest': 'kitchen_design'
}
create_response = requests.post('https://baimuras.space/api/v1/leads',
                                json=new_lead, headers=headers)
```

### JavaScript (fetch)

```
// Аутентификация
const authResponse = await fetch('https://baimuras.space/api/v1/auth/login', {
  method: 'POST',
  headers: { 'Content-Type': 'application/json' },
  body: JSON.stringify({
    email: 'admin@baimuras.space',
    password: 'your-password'
  })
});
const { access_token } = await authResponse.json();

// Получение профиля пользователя
const profileResponse = await fetch('https://baimuras.space/api/v1/auth/profile', {
  headers: { 'Authorization': `Bearer ${access_token}` }
});
const profile = await profileResponse.json();
```



## cURL

```
# Аутентификация
curl -X POST https://baimuras.space/api/v1/auth/login \
  -H "Content-Type: application/json" \
  -d '{"email":"admin@baimuras.space","password":"your-password"}'

# Получение списка пользователей
curl -X GET https://baimuras.space/api/v1/users \
  -H "Authorization: Bearer YOUR_TOKEN"

# Создание лида
curl -X POST https://baimuras.space/api/v1/leads \
  -H "Authorization: Bearer YOUR_TOKEN" \
  -H "Content-Type: application/json" \
  -d '{"name":"Новый клиент","email":"client@example.com","phone":"+7 777 123 45 67"}'
```

## Поддержка

---

Если у вас есть вопросы по API:

1. Проверьте документацию выше
2. Изучите примеры кода
3. Создайте Issue в репозитории: <https://github.com/ardakchapaev/baimuras.space/issues>
4. Свяжитесь с разработчиком: [admin@baimuras.space](mailto:admin@baimuras.space)

## Версионирование

---

API использует версионирование через URL путь ( /api/v1/ ). При внесении breaking changes будет создана новая версия API с сохранением обратной совместимости предыдущих версий.

---

**Последнее обновление:** 21 июня 2025 г.

**Версия документации:** 2.0.0