

API Документация BaiMuras Platform

Обзор API

BaiMuras Platform предоставляет RESTful API для интеграции с внешними системами, управления контентом и автоматизации бизнес-процессов.

Базовая информация

- **Base URL:** `https://baimuras.space/api/v1`
- **Протокол:** HTTPS только
- **Формат данных:** JSON
- **Аутентификация:** Bearer Token / API Key
- **Версионирование:** URL path (`/api/v1/`)
- **Rate Limiting:** 1000 запросов/час для аутентифицированных пользователей

Аутентификация

Bearer Token

```
Authorization: Bearer YOUR_API_TOKEN
```

API Key

```
X-API-Key: YOUR_API_KEY
```

Получение токена

```
POST /api/v1/auth/token
Content-Type: application/json
```

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

Ответ:

```
{
  "access_token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9...",
  "refresh_token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9...",
  "expires_in": 3600,
  "token_type": "Bearer"
}
```

Стандартные ответы

Успешный ответ

```
{
  "success": true,
  "data": {
    // Данные ответа
  },
  "message": "Операция выполнена успешно",
  "timestamp": "2025-06-20T08:23:20Z"
}
```

Ошибка

```
{
  "success": false,
  "error": {
    "code": "VALIDATION_ERROR",
    "message": "Ошибка валидации данных",
    "details": {
      "field": "email",
      "reason": "Неверный формат email"
    }
  },
  "timestamp": "2025-06-20T08:23:20Z"
}
```

Коды состояния HTTP

- 200 - OK
- 201 - Created
- 400 - Bad Request
- 401 - Unauthorized
- 403 - Forbidden
- 404 - Not Found
- 422 - Unprocessable Entity
- 429 - Too Many Requests
- 500 - Internal Server Error

Портфолио API

Получить все проекты

```
GET /api/v1/portfolio/projects
```

Параметры запроса:

- page (int, optional) - Номер страницы (по умолчанию: 1)
- per_page (int, optional) - Количество элементов на странице (по умолчанию: 10, максимум: 100)
- category (string, optional) - Фильтр по категории
- status (string, optional) - Фильтр по статусу (active , draft , archived)

- `sort` (string, optional) - Сортировка (`created_at` , `updated_at` , `title`)
- `order` (string, optional) - Порядок сортировки (`asc` , `desc`)

Пример запроса:

```
GET /api/v1/portfolio/projects?  
page=1&per_page=5&category=web&sort=created_at&order=desc
```

Ответ:

```
{  
  "success": true,  
  "data": {  
    "projects": [  
      {  
        "id": 1,  
        "title": "Корпоративный сайт",  
        "description": "Современный корпоративный сайт с адаптивным дизайном",  
        "category": "web",  
        "status": "active",  
        "images": [  
          {  
            "id": 1,  
            "url": "https://i.ytimg.com/vi/AVBzyHPbSqI/maxresdefault.jpg",  
            "alt": "Главная страница сайта",  
            "is_primary": true  
          }  
        ],  
        "technologies": ["HTML5", "CSS3", "JavaScript", "Python", "Flask"],  
        "client": "ТОО Компания",  
        "completion_date": "2024-12-15",  
        "project_url": "https://example.com",  
        "created_at": "2024-11-01T10:00:00Z",  
        "updated_at": "2024-12-15T15:30:00Z"  
      }  
    ],  
    "pagination": {  
      "page": 1,  
      "per_page": 5,  
      "total": 25,  
      "pages": 5,  
      "has_next": true,  
      "has_prev": false  
    }  
  }  
}
```

Получить проект по ID

```
GET /api/v1/portfolio/projects/{id}
```

Ответ:

```
{
  "success": true,
  "data": {
    "id": 1,
    "title": "Корпоративный сайт",
    "description": "Подробное описание проекта...",
    "category": "web",
    "status": "active",
    "images": [...],
    "technologies": [...],
    "client": "ООО Компания",
    "completion_date": "2024-12-15",
    "project_url": "https://example.com",
    "created_at": "2024-11-01T10:00:00Z",
    "updated_at": "2024-12-15T15:30:00Z"
  }
}
```

Создать новый проект

```
POST /api/v1/portfolio/projects
Authorization: Bearer YOUR_TOKEN
Content-Type: application/json
```

```
{
  "title": "Новый проект",
  "description": "Описание нового проекта",
  "category": "web",
  "status": "draft",
  "technologies": ["React", "Node.js", "MongoDB"],
  "client": "Новый клиент",
  "project_url": "https://newproject.com"
}
```

Ответ:

```
{
  "success": true,
  "data": {
    "id": 26,
    "title": "Новый проект",
    // ... остальные поля
  },
  "message": "Проект успешно создан"
}
```

Обновить проект

```
PUT /api/v1/portfolio/projects/{id}
Authorization: Bearer YOUR_TOKEN
Content-Type: application/json

{
  "title": "Обновленное название",
  "status": "active"
}
```

Удалить проект

```
DELETE /api/v1/portfolio/projects/{id}
Authorization: Bearer YOUR_TOKEN
```

Контакты API

Отправить сообщение

```
POST /api/v1/contact/messages
Content-Type: application/json

{
  "name": "Иван Иванов",
  "email": "ivan@example.com",
  "phone": "+7 777 123 45 67",
  "subject": "Вопрос по сотрудничеству",
  "message": "Здравствуйте! Хотел бы обсудить возможность сотрудничества...",
  "project_type": "website",
  "budget": "100000-500000"
}
```

Ответ:

```
{
  "success": true,
  "data": {
    "id": 123,
    "status": "received",
    "created_at": "2025-06-20T08:23:20Z"
  },
  "message": "Сообщение успешно отправлено"
}
```

Получить сообщения (только для админов)

```
GET /api/v1/contact/messages
Authorization: Bearer ADMIN_TOKEN
```

Параметры:

- `page` (int) - Номер страницы
- `per_page` (int) - Количество на странице
- `status` (string) - Фильтр по статусу (`new` , `read` , `replied` , `archived`)
- `date_from` (string) - Дата от (ISO 8601)
- `date_to` (string) - Дата до (ISO 8601)

Пользователи API

Получить профиль пользователя

```
GET /api/v1/users/profile
Authorization: Bearer YOUR_TOKEN
```

Ответ:

```
{
  "success": true,
  "data": {
    "id": 1,
    "email": "user@example.com",
    "name": "Иван Иванов",
    "role": "user",
    "avatar": "https://lh3.googleusercontent.com/X8LuYsGddUvyGns8yNt3lsqXU-etopUi9saFCQ-VMIIImDW0plr-ZvBRjhnKh4V2r6UEMaBMXUB-kJSD_RrHbWdmIp2RUnVJgcbiJ_S3l_k0AseWWI6JiLccLcL0cRFpnba-n4bjl0W3FvHbHdMs_ToZE",
    "created_at": "2024-01-15T10:00:00Z",
    "last_login": "2025-06-20T07:30:00Z",
    "preferences": {
      "language": "ru",
      "notifications": true
    }
  }
}
```

Обновить профиль

```
PUT /api/v1/users/profile
Authorization: Bearer YOUR_TOKEN
Content-Type: application/json

{
  "name": "Новое имя",
  "preferences": {
    "language": "kz",
    "notifications": false
  }
}
```

Изменить пароль

```
POST /api/v1/users/change-password
```

```
Authorization: Bearer YOUR_TOKEN
```

```
Content-Type: application/json
```

```
{
```

```
  "current_password": "old_password",
```

```
  "new_password": "new_secure_password",
```

```
  "confirm_password": "new_secure_password"
```

```
}
```

Аналитика API

Получить статистику сайта

```
GET /api/v1/analytics/stats
```

```
Authorization: Bearer ADMIN_TOKEN
```

Параметры:

- `period` (string) - Период (`day` , `week` , `month` , `year`)
- `date_from` (string) - Дата начала
- `date_to` (string) - Дата окончания

Ответ:

```
{
  "success": true,
  "data": {
    "visitors": {
      "total": 1250,
      "unique": 980,
      "returning": 270
    },
    "pageviews": {
      "total": 3450,
      "per_visitor": 2.76
    },
    "popular_pages": [
      {
        "path": "/portfolio",
        "views": 850,
        "percentage": 24.6
      },
      {
        "path": "/",
        "views": 720,
        "percentage": 20.9
      }
    ],
    "traffic_sources": {
      "direct": 45.2,
      "search": 32.1,
      "social": 15.7,
      "referral": 7.0
    },
    "devices": {
      "desktop": 58.3,
      "mobile": 35.2,
      "tablet": 6.5
    }
  }
}
```

Получить данные о конверсиях

```
GET /api/v1/analytics/conversions
Authorization: Bearer ADMIN_TOKEN
```

Ответ:


```
{
  "success": true,
  "data": {
    "contact_form": {
      "submissions": 45,
      "conversion_rate": 3.6
    },
    "project_inquiries": {
      "total": 28,
      "qualified": 18,
      "conversion_rate": 64.3
    },
    "goals": [
      {
        "name": "Contact Form Submission",
        "completions": 45,
        "value": 150000
      }
    ]
  }
}
```

Файлы API

Загрузить файл

```
POST /api/v1/files/upload
Authorization: Bearer YOUR_TOKEN
Content-Type: multipart/form-data

file: [binary data]
category: "portfolio" | "avatar" | "document"
```

Ответ:

```
{
  "success": true,
  "data": {
    "id": 123,
    "filename": "project_image.jpg",
    "original_filename": "my_project.jpg",
    "url": "https://i.pinimg.com/originals/e6/e8/52/e6e85246639b7a64923c5a7b-
cea1009e.jpg",
    "size": 245760,
    "mime_type": "image/jpeg",
    "category": "portfolio",
    "uploaded_at": "2025-06-20T08:23:20Z"
  }
}
```

Получить информацию о файле

```
GET /api/v1/files/{id}
Authorization: Bearer YOUR_TOKEN
```

Удалить файл

```
DELETE /api/v1/files/{id}
Authorization: Bearer YOUR_TOKEN
```

Системные API

Проверка состояния системы

```
GET /api/v1/system/health
```

Ответ:

```
{
  "success": true,
  "data": {
    "status": "healthy",
    "version": "2.0.0",
    "uptime": 86400,
    "services": {
      "database": "healthy",
      "redis": "healthy",
      "email": "healthy",
      "storage": "healthy"
    },
    "metrics": {
      "memory_usage": 65.2,
      "cpu_usage": 23.1,
      "disk_usage": 45.8
    }
  }
}
```

Получить информацию о версии

```
GET /api/v1/system/version
```

Ответ:

```
{
  "success": true,
  "data": {
    "version": "2.0.0",
    "build": "20250620-082320",
    "environment": "production",
    "python_version": "3.11.0",
    "flask_version": "3.0.0"
  }
}
```

Webhook API

Настройка webhook

```
POST /api/v1/webhooks
Authorization: Bearer ADMIN_TOKEN
Content-Type: application/json

{
  "url": "https://your-app.com/webhook",
  "events": ["contact.created", "project.updated"],
  "secret": "your_webhook_secret",
  "active": true
}
```

События webhook

- `contact.created` - Новое сообщение через форму контактов
- `contact.updated` - Обновление статуса сообщения
- `project.created` - Создание нового проекта
- `project.updated` - Обновление проекта
- `user.registered` - Регистрация нового пользователя

Формат webhook payload

```
{
  "event": "contact.created",
  "timestamp": "2025-06-20T08:23:20Z",
  "data": {
    "id": 123,
    "name": "Иван Иванов",
    "email": "ivan@example.com",
    "message": "Текст сообщения"
  },
  "signature": "sha256=..."
}
```

Примеры использования

JavaScript (Fetch API)

```
// Получение списка проектов
async function getProjects() {
  try {
    const response = await fetch('https://baimuras.space/api/v1/portfolio/projects', {
      headers: {
        'Authorization': 'Bearer YOUR_TOKEN',
        'Content-Type': 'application/json'
      }
    });

    const data = await response.json();

    if (data.success) {
      console.log('Проекты:', data.data.projects);
    } else {
      console.error('Ошибка:', data.error.message);
    }
  } catch (error) {
    console.error('Ошибка запроса:', error);
  }
}

// Отправка сообщения
async function sendMessage(messageData) {
  try {
    const response = await fetch('https://baimuras.space/api/v1/contact/messages', {
      method: 'POST',
      headers: {
        'Content-Type': 'application/json'
      },
      body: JSON.stringify(messageData)
    });

    const data = await response.json();
    return data;
  } catch (error) {
    console.error('Ошибка отправки:', error);
  }
}
```

Python (requests)

```
import requests

class BaiMurasAPI:
    def __init__(self, base_url, token=None):
        self.base_url = base_url
        self.token = token
        self.session = requests.Session()

        if token:
            self.session.headers.update({
                'Authorization': f'Bearer {token}',
                'Content-Type': 'application/json'
            })

    def get_projects(self, **params):
        """Получить список проектов"""
        response = self.session.get(
            f'{self.base_url}/portfolio/projects',
            params=params
        )
        return response.json()

    def create_project(self, project_data):
        """Создать новый проект"""
        response = self.session.post(
            f'{self.base_url}/portfolio/projects',
            json=project_data
        )
        return response.json()

    def send_message(self, message_data):
        """Отправить сообщение"""
        response = self.session.post(
            f'{self.base_url}/contact/messages',
            json=message_data
        )
        return response.json()

# Использование
api = BaiMurasAPI('https://baimuras.space/api/v1', 'YOUR_TOKEN')

# Получить проекты
projects = api.get_projects(category='web', per_page=5)
print(projects)

# Отправить сообщение
message = {
    'name': 'Тест',
    'email': 'test@example.com',
    'message': 'Тестовое сообщение'
}
result = api.send_message(message)
print(result)
```

PHP (cURL)

```
<?php
class BaiMurasAPI {
    private $baseUrl;
    private $token;

    public function __construct($baseUrl, $token = null) {
        $this->baseUrl = $baseUrl;
        $this->token = $token;
    }

    private function makeRequest($method, $endpoint, $data = null) {
        $curl = curl_init();

        $headers = ['Content-Type: application/json'];
        if ($this->token) {
            $headers[] = 'Authorization: Bearer ' . $this->token;
        }

        curl_setopt_array($curl, [
            CURLOPT_URL => $this->baseUrl . $endpoint,
            CURLOPT_RETURNTRANSFER => true,
            CURLOPT_CUSTOMREQUEST => $method,
            CURLOPT_HTTPHEADER => $headers,
            CURLOPT_POSTFIELDS => $data ? json_encode($data) : null,
        ]);

        $response = curl_exec($curl);
        curl_close($curl);

        return json_decode($response, true);
    }

    public function getProjects($params = []) {
        $query = http_build_query($params);
        return $this->makeRequest('GET', '/portfolio/projects?' . $query);
    }

    public function sendMessage($messageData) {
        return $this->makeRequest('POST', '/contact/messages', $messageData);
    }
}

// Использование
$sapi = new BaiMurasAPI('https://baimuras.space/api/v1', 'YOUR_TOKEN');

$projects = $sapi->getProjects(['category' => 'web']);
print_r($projects);
?>
```

Безопасность API

Rate Limiting

- Аутентифицированные пользователи: 1000 запросов/час
- Неаутентифицированные: 100 запросов/час

- Админы: 5000 запросов/час

Заголовки Rate Limiting

```
X-RateLimit-Limit: 1000  
X-RateLimit-Remaining: 999  
X-RateLimit-Reset: 1640995200
```

Валидация данных

Все входящие данные проходят строгую валидацию:

- Проверка типов данных
- Ограничения длины строк
- Валидация email адресов
- Санитизация HTML
- Проверка на SQL инъекции

CORS настройки

```
Access-Control-Allow-Origin: https://yourdomain.com  
Access-Control-Allow-Methods: GET, POST, PUT, DELETE, OPTIONS  
Access-Control-Allow-Headers: Authorization, Content-Type, X-API-Key  
Access-Control-Max-Age: 86400
```

Мониторинг API

Метрики

- Количество запросов в секунду
- Время ответа API
- Коды ошибок
- Использование rate limit

Логирование

Все API запросы логируются с информацией:

- Timestamp
- IP адрес
- User Agent
- Endpoint
- Метод
- Статус ответа
- Время выполнения

Поддержка и помощь

Документация

- **Swagger UI:** <https://baimuras.space/api/docs>
- **Postman Collection:** [Скачать](https://baimuras.space/api/postman-collection.json) (<https://baimuras.space/api/postman-collection.json>)

Поддержка

- **Email:** api-support@baimuras.space
- **GitHub Issues:** <https://github.com/ardakchapaev/baimuras.space/issues>
- **Telegram:** [@baimuras_support](https://t.me/baimuras_support)

Обновления API

Подпишитесь на уведомления об обновлениях API:

- **Changelog:** <https://baimuras.space/api/changelog>
- **RSS:** <https://baimuras.space/api/changelog.rss>

Версия документации: 2.0.0

Последнее обновление: 20 июня 2025 г.