

# API Reference - ClientManager

---

## Índice

---

1. [Introdução](#)
2. [Autenticação](#)
3. [Endpoints de Autenticação](#)
4. [Endpoints de Clientes](#)
5. [Endpoints de Campanhas](#)
6. [Endpoints de Relatórios](#)
7. [Endpoints de Integrações](#)
8. [Endpoints de Agendamento](#)
9. [Webhooks](#)
10. [Códigos de Erro](#)

## Introdução

---

A API do ClientManager é uma API RESTful que permite integração completa com o sistema de gerenciamento de clientes de tráfego pago.

### Base URL

```
https://seu-dominio.com/api
```

### Formato de Resposta

Todas as respostas são em formato JSON:

```
{  
  "success": true,  
  "data": {},  
  "message": "Operação realizada com sucesso"  
}
```

## Headers Obrigatórios

```
Content-Type: application/json  
Authorization: Bearer {token}
```

## Autenticação

---

O sistema utiliza JWT (JSON Web Tokens) para autenticação. Após o login, você receberá um token que deve ser incluído no header `Authorization` de todas as requisições.

## Fluxo de Autenticação

1. Faça login com email/senha
2. Receba o JWT token
3. Inclua o token em todas as requisições
4. Renove o token quando necessário

## Roles Disponíveis

- `admin` : Acesso completo ao sistema
- `gestor` : Gerencia clientes e campanhas
- `cliente` : Acesso apenas aos próprios dados

## Endpoints de Autenticação

---

### POST /api/auth/login

Realiza login no sistema.

### Request Body:

```
{
  "email": "usuario@exemplo.com",
  "password": "senha123"
}
```

### Response:

```
{
  "success": true,
  "data": {
    "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",
    "user": {
      "id": "1",
      "name": "João Silva",
      "email": "joao@exemplo.com",
      "role": "gestor"
    }
  }
}
```

## POST /api/auth/register

Registra novo usuário (apenas admins).

### Request Body:

```
{
  "name": "Maria Santos",
  "email": "maria@exemplo.com",
  "password": "senha123",
  "role": "cliente"
}
```

### Response:

```
{
  "success": true,
  "data": {
    "id": "2",
    "name": "Maria Santos",
    "email": "maria@exemplo.com",
    "role": "cliente"
  }
}
```

## GET /api/auth/me

Retorna dados do usuário autenticado.

### Response:

```
{
  "success": true,
  "data": {
    "id": "1",
    "name": "João Silva",
    "email": "joao@exemplo.com",
    "role": "gestor",
    "createdAt": "2024-01-01T00:00:00Z"
  }
}
```



## Endpoints de Clientes

---

### GET /api/clients

Lista todos os clientes.

**Query Parameters:** - `page` : Número da página (padrão: 1) - `limit` : Itens por página (padrão: 10) - `search` : Busca por nome ou empresa - `segment` : Filtro por segmento

### Response:

```
{
  "success": true,
  "data": {
    "clients": [
      {
        "id": "1",
        "name": "Maria Silva",
        "email": "maria@empresaabc.com",
        "company": "Empresa ABC Ltda",
        "segment": "E-commerce",
        "monthlyBudget": 5000,
        "isActive": true,
        "createdAt": "2024-01-01T00:00:00Z"
      }
    ],
    "pagination": {
      "page": 1,
      "limit": 10,
      "total": 25,
      "pages": 3
    }
  }
}
```

## POST /api/clients

Cria novo cliente.

### Request Body:

```
{
  "name": "Pedro Santos",
  "email": "pedro@startupxyz.com",
  "phone": "+5511999999999",
  "whatsappNumber": "+5511999999999",
  "company": "Startup XYZ",
  "segment": "SaaS",
  "metaAdsAccountId": "act_123456789",
  "googleAdsAccountId": "123-456-7890",
  "monthlyBudget": 3000
}
```

## GET /api/clients/:id

Retorna dados de um cliente específico.

### Response:

```
{
  "success": true,
  "data": {
    "id": "1",
    "name": "Maria Silva",
    "email": "maria@empresaabc.com",
    "company": "Empresa ABC Ltda",
    "campaigns": [
      {
        "id": "1",
        "name": "Campanha Black Friday",
        "platform": "meta_ads",
        "status": "ativo"
      }
    ]
  }
}
```

## PUT /api/clients/:id

Atualiza dados do cliente.

## DELETE /api/clients/:id

Remove cliente do sistema.

## Endpoints de Campanhas

---

### GET /api/campaigns

Lista campanhas.

**Query Parameters:** - `clientId`: Filtro por cliente - `platform`: meta\_ads ou google\_ads - `status`: ativo, pausado, finalizado

**Response:**

```
{
  "success": true,
  "data": [
    {
      "id": "1",
      "name": "Campanha Black Friday",
      "platform": "meta_ads",
      "status": "ativo",
      "budget": 1000,
      "clientId": "1",
      "metrics": {
        "spend": 850,
        "impressions": 45000,
        "clicks": 920,
        "conversions": 28
      }
    }
  ]
}
```

## POST /api/campaigns

Cria nova campanha.

### Request Body:

```
{
  "name": "Nova Campanha",
  "platform": "meta_ads",
  "budget": 1500,
  "clientId": "1",
  "objective": "conversions",
  "targetAudience": "lookalike"
}
```

## PUT /api/campaigns/:id

Atualiza campanha.

## DELETE /api/campaigns/:id

Remove campanha.



## Endpoints de Relatórios

---

### POST /api/reports/generate

Gera novo relatório.

#### Request Body:

```
{
  "clientId": "1",
  "period": "weekly",
  "startDate": "2024-01-01",
  "endDate": "2024-01-07",
  "format": "whatsapp"
}
```

#### Response:

```
{
  "success": true,
  "data": {
    "reportData": {
      "clientName": "Empresa ABC Ltda",
      "totalSpend": 5000,
      "totalConversions": 150,
      "avgRoas": 3.2
    },
    "message": "📊 Relatório Semanal - Empresa ABC Ltda..."
  }
}
```

### GET /api/reports

Lista histórico de relatórios.

**Query Parameters:** - `clientId`: Filtro por cliente - `limit`: Número de relatórios (padrão: 10)

#### Response:



```
{
  "success": true,
  "data": [
    {
      "id": "1",
      "title": "Relatório Semanal - Empresa ABC",
      "period": "weekly",
      "status": "gerado",
      "createdAt": "2024-01-15T10:30:00Z",
      "clientId": "1"
    }
  ]
}
```

## Endpoints de Integrações

---

### POST /api/integrations/sync

Sincroniza dados das integrações.

#### Request Body:

```
{
  "platform": "meta_ads",
  "clientId": "1",
  "startDate": "2024-01-01",
  "endDate": "2024-01-07"
}
```

#### Response:

```
{
  "success": true,
  "data": {
    "platform": "meta_ads",
    "syncedCampaigns": 5,
    "syncedMetrics": 35,
    "lastSync": "2024-01-15T10:30:00Z"
  }
}
```

### GET /api/integrations/status

Verifica status das integrações.

#### Response:

```
{
  "success": true,
  "data": {
    "meta_ads": {
      "status": "connected",
      "lastSync": "2024-01-15T10:30:00Z"
    },
    "google_ads": {
      "status": "connected",
      "lastSync": "2024-01-15T10:25:00Z"
    },
    "evolution_api": {
      "status": "connected",
      "lastCheck": "2024-01-15T10:35:00Z"
    }
  }
}
```

## Endpoints de Agendamento

---

### GET /api/scheduler/tasks

Lista tarefas agendadas.

**Response:**

```
{
  "success": true,
  "data": [
    {
      "id": "1",
      "name": "Relatório Diário - Empresa ABC",
      "cronExpression": "0 9 * * *",
      "type": "report",
      "isActive": true,
      "lastRun": "2024-01-15T09:00:00Z",
      "nextRun": "2024-01-16T09:00:00Z"
    }
  ]
}
```

### POST /api/scheduler/tasks

Cria nova tarefa agendada.

**Request Body:**

```
{
  "name": "Relatório Semanal - Cliente XYZ",
  "cronExpression": "0 9 * * 1",
  "type": "report",
  "config": {
    "clientId": "2",
    "period": "weekly",
    "sendWhatsApp": true,
    "sendEmail": false
  },
  "isActive": true
}
```

## PUT /api/scheduler/tasks/:id

Atualiza tarefa agendada.

## DELETE /api/scheduler/tasks/:id

Remove tarefa agendada.



## Webhooks

---

## POST /api/webhooks/n8n

Recebe dados do n8n.

### Request Body:

```
{
  "event": "campaign_updated",
  "data": {
    "campaignId": "1",
    "status": "paused",
    "reason": "budget_exceeded"
  }
}
```

## POST /api/webhooks/evolution

Recebe status de mensagens da Evolution API.

### Request Body:

```
{
  "event": "message_delivered",
  "data": {
    "messageId": "msg_123",
    "recipient": "+5511999999999",
    "status": "delivered",
    "timestamp": "2024-01-15T10:30:00Z"
  }
}
```

## Códigos de Erro

---

### 400 - Bad Request

```
{
  "success": false,
  "error": "Dados inválidos",
  "details": {
    "field": "email",
    "message": "Email é obrigatório"
  }
}
```

### 401 - Unauthorized

```
{
  "success": false,
  "error": "Token inválido ou expirado"
}
```

### 403 - Forbidden

```
{
  "success": false,
  "error": "Sem permissão para acessar este recurso"
}
```

### 404 - Not Found

```
{
  "success": false,
  "error": "Recurso não encontrado"
}
```

## 500 - Internal Server Error

```
{
  "success": false,
  "error": "Erro interno do servidor"
}
```



## Exemplos de Uso

---

### Autenticação e Busca de Clientes

```
// Login
const loginResponse = await fetch('/api/auth/login', {
  method: 'POST',
  headers: {
    'Content-Type': 'application/json'
  },
  body: JSON.stringify({
    email: 'gestor@exemplo.com',
    password: 'senha123'
  })
});

const { data } = await loginResponse.json();
const token = data.token;

// Buscar clientes
const clientsResponse = await fetch('/api/clients', {
  headers: {
    'Authorization': `Bearer ${token}`,
    'Content-Type': 'application/json'
  }
});

const clients = await clientsResponse.json();
```

## Geração de Relatório

```
const reportResponse = await fetch('/api/reports/generate', {
  method: 'POST',
  headers: {
    'Authorization': `Bearer ${token}`,
    'Content-Type': 'application/json'
  },
  body: JSON.stringify({
    clientId: '1',
    period: 'weekly',
    format: 'whatsapp'
  })
});

const report = await reportResponse.json();
```



## Rate Limiting

---

- **Limite:** 100 requisições por minuto por IP
- **Headers de resposta:**
- `X-RateLimit-Limit` : Limite total
- `X-RateLimit-Remaining` : Requisições restantes
- `X-RateLimit-Reset` : Timestamp do reset



## SDKs e Bibliotecas

---

### JavaScript/Node.js

```
npm install clientmanager-sdk
```

### Python

```
pip install clientmanager-python
```

### PHP

```
composer require clientmanager/php-sdk
```

---

**Versão da API:** 1.0.0

**Última atualização:** Janeiro 2024