

API Endpoints

Autenticação

POST /api/signup

Cria novo usuário.

Request:

```
{  
  "email": "user@example.com",  
  "password": "senha123",  
  "name": "Nome do Usuário",  
  "tenantId": "tenant-id-opcional"  
}
```

Response (200):

```
{  
  "id": "user-id",  
  "email": "user@example.com",  
  "name": "Nome do Usuário"  
}
```

POST /api/auth/[...nextauth]

Endpoints NextAuth.js para login/logout.

Chat

POST /api/chat/message

Envia mensagem para o orquestrador.

Request:

```
{  
  "message": "Agendar reunião amanhã às 14h",  
  "conversationId": "conv-id-opcional",  
  "tenantId": "tenant-id",  
  "channel": "web"  
}
```

Response (200):

```
{
  "message": "Vou criar um compromisso:\n\n[17] Data: 30/01/2026...\n\nConfirmar\n2) Ajustar\n3) Cancelar",
  "agentUsed": "agent.secretary",
  "requiresConfirmation": true,
  "conversationId": "conv-id"
}
```

GET /api/chat/conversations/:id

Busca detalhes de uma conversa.

Response (200):

```
{
  "id": "conv-id",
  "channel": "web",
  "status": "active",
  "createdAt": "2026-01-29T10:00:00Z",
  "turns": [
    {
      "id": "turn-id",
      "role": "user",
      "content": "Agendar reunião",
      "createdAt": "2026-01-29T10:00:00Z"
    },
    {
      "id": "turn-id-2",
      "role": "assistant",
      "content": "Vou criar um compromisso...",
      "metadata": { "agentUsed": "agent.secretary" },
      "createdAt": "2026-01-29T10:00:01Z"
    }
  ],
  "context": {
    "summary": "",
    "activeAgent": "agent.secretary",
    "pendingAction": { ... }
  }
}
```

Admin - Tenants

GET /api/admin/tenants

Lista todos os tenants.

Response (200):

```
[
  {
    "id": "tenant-id",
    "slug": "empresa-pro",
    "name": "Empresa Pro",
    "status": "active",
    "plan": { "id": "plan-id", "name": "Pro", "tier": "pro" },
    "usersCount": 5,
    "conversationsCount": 120,
    "createdAt": "2026-01-01T00:00:00Z"
  }
]
```

POST /api/admin/tenants

Cria novo tenant.

Request:

```
{
  "slug": "nova-empresa",
  "name": "Nova Empresa",
  "planId": "plan-id"
}
```

GET /api/admin/tenants/:id

Busca detalhes de um tenant.

Response (200):

```
{
  "id": "tenant-id",
  "slug": "empresa-pro",
  "name": "Empresa Pro",
  "status": "active",
  "plan": { ... },
  "agentConfigs": [
    {
      "agentSlug": "agent.secretary",
      "enabled": true,
      "agent": { "name": "Secretaria Virtual" }
    }
  ],
  "integrationConfigs": [ ... ],
  "stats": {
    "users": 5,
    "conversations": 120,
    "calendarEvents": 45,
    "financialTxns": 230
  }
}
```

PATCH /api/admin/tenants/:id

Atualiza tenant.

Request:

```
{
  "name": "Novo Nome",
  "status": "inactive",
  "planId": "outro-plan-id"
}
```

Admin - Planos

GET /api/admin/plans

Lista todos os planos.

Response (200):

```
[
  {
    "id": "plan-id",
    "slug": "pro",
    "name": "Pro",
    "description": "Plano profissional",
    "tier": "pro",
    "limits": {
      "messages_per_month": null,
      "agent_actions_per_month": null
    },
    "features": {
      "auto_execution": true,
      "multi_agent_flows": true,
      "enabled_agents": ["agent.secretary", "agent.finance"]
    },
    "billing": {
      "price_monthly": 149.90,
      "currency": "BRL"
    },
    "tenantsCount": 15
  }
]
```

Admin - Agentes

GET /api/admin/agents

Lista todos os agentes.

Response (200):

```
[
  {
    "id": "agent-id",
    "slug": "agent.secretary",
    "name": "Secretaria Virtual",
    "description": "Agente de gerenciamento de agenda",
    "category": "productivity",
    "status": "active",
    "channelsSupported": ["web", "whatsapp"],
    "intentsSupported": ["agendar", "remarcar", "cancelar"],
    "planConstraints": { "min_plan_tier": "free" },
    "behavior": {
      "confirmation_required": true,
      "max_questions_per_turn": 2
    },
    "activeTenantsCount": 25
  }
]
```

Admin - Integrações

GET /api/admin/integrations

Lista todas as integrações.

Response (200):

```
[
  {
    "id": "integration-id",
    "slug": "integration.google.calendar",
    "name": "Google Calendar",
    "type": "calendar",
    "capabilities": { "read": true, "write": true },
    "supportedAgents": ["agent.secretary"],
    "syncStrategy": {
      "polling_interval_minutes": 15,
      "manual_sync_supported": true
    },
    "activeTenantsCount": 20
  }
]
```

Admin - Auditoria

GET /api/admin/audit

Lista logs de auditoria.

Query Parameters:

- tenantId (opcional)
- agentSlug (opcional)

- `action` (opcional)
- `limit` (opcional, default: 100)

Response (200):

```
[
  {
    "id": "log-id",
    "tenantId": "tenant-id",
    "tenantName": "Empresa Pro",
    "userId": "user-id",
    "userName": "João Silva",
    "agentSlug": "agent.secretary",
    "action": "create_event",
    "entityType": "calendar_event",
    "entityId": "event-id",
    "before": null,
    "after": {
      "title": "Reunião",
      "startTime": "2026-01-30T14:00:00Z"
    },
    "createdAt": "2026-01-29T10:00:00Z"
  }
]
```

Admin - Estatísticas

GET /api/admin/stats

Retorna estatísticas gerais do sistema.

Response (200):

```
{
  "overview": {
    "totalTenants": 50,
    "activeTenants": 45,
    "activeConversations": 120,
    "activeAgents": 2,
    "totalAuditLogs": 5430
  },
  "recentActivity": [
    {
      "id": "conv-id",
      "tenantName": "Empresa Pro",
      "channel": "web",
      "updatedAt": "2026-01-29T10:00:00Z"
    }
  ],
  "agentUsage": [
    { "agent": "agent.secretary", "count": 450 },
    { "agent": "agent.finance", "count": 230 }
  ]
}
```

Códigos de Erro

Código	Descrição
400	Bad Request - Parâmetros inválidos
401	Unauthorized - Não autenticado
403	Forbidden - Sem permissão
404	Not Found - Recurso não encontrado
500	Internal Server Error - Erro interno

Formato de Erro:

```
{  
  "error": "Descrição do erro"  
}
```