

# Troubleshooting - Voz do Cliente

## Problemas Comuns e Soluções

### 1. Problemas de Instalação

**Erro: “Node.js version not supported”**

**Sintoma:**

```
error: The engine "node" is incompatible with this module
```

**Solução:**

```
# Verificar versão atual
node --version

# Instalar Node.js 18+ via nvm
curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.0/install.sh | bash
source ~/.bashrc
nvm install 18
nvm use 18

# Ou via NodeSource (Ubuntu/Debian)
curl -fsSL https://deb.nodesource.com/setup_18.x | sudo -E bash -
sudo apt-get install -y nodejs
```

**Erro: “Cannot find module ‘@prisma/client”**

**Sintoma:**

```
Error: Cannot find module '@prisma/client'
```

**Solução:**

```
# Gerar cliente Prisma
cd app
npx prisma generate

# Se persistir, reinstalar dependências
rm -rf node_modules package-lock.json
npm install
npx prisma generate
```

**Erro: “EACCES: permission denied”**

**Sintoma:**

```
npm ERR! Error: EACCES: permission denied, mkdir '/usr/local/lib/node_modules'
```

**Solução:**

```
# Configurar npm para usar diretório do usuário
mkdir ~/.npm-global
npm config set prefix '~/.npm-global'
echo 'export PATH=~/.npm-global/bin:$PATH' >> ~/.bashrc
source ~/.bashrc

# Ou usar nvm (recomendado)
curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.0/install.sh | bash
```

**2. Problemas de Banco de Dados****Erro: “Connection refused” PostgreSQL****Sintoma:**

```
Error: connect ECONNREFUSED 127.0.0.1:5432
```

**Diagnóstico:**

```
# Verificar se PostgreSQL está rodando
sudo systemctl status postgresql

# Verificar se está escutando na porta correta
sudo netstat -tlnp | grep 5432

# Verificar logs do PostgreSQL
sudo tail -f /var/log/postgresql/postgresql-*.log
```

**Solução:**

```
# Iniciar PostgreSQL
sudo systemctl start postgresql
sudo systemctl enable postgresql

# Se não iniciar, verificar configuração
sudo -u postgres psql -c "SELECT version();"

# Verificar configuração de conexão
sudo nano /etc/postgresql/*/main/postgresql.conf
# Verificar: listen_addresses = 'localhost'

sudo nano /etc/postgresql/*/main/pg_hba.conf
# Adicionar: local all all trust
```

**Erro: “Database does not exist”****Sintoma:**

```
Error: database "voz_cliente" does not exist
```

**Solução:**

```
# Conectar como superusuário
sudo -u postgres psql

# Criar banco de dados
CREATE DATABASE voz_cliente;
CREATE USER voc_user WITH ENCRYPTED PASSWORD 'sua_senha';
GRANT ALL PRIVILEGES ON DATABASE voz_cliente TO voc_user;
\q

# Atualizar .env com a string de conexão correta
DATABASE_URL="postgresql://voc_user:sua_senha@localhost:5432/voz_cliente"
```

### Erro: “Prisma schema not found”

#### Sintoma:

```
Error: Could not find a schema.prisma file
```

#### Solução:

```
# Verificar se está no diretório correto
cd app
ls -la prisma/

# Se não existir, verificar estrutura do projeto
find . -name "schema.prisma"

# Executar comandos Prisma do diretório correto
cd app
npx prisma generate
npx prisma db push
```

## 3. Problemas de APIs Externas

### Twitter API: “Unauthorized” (401)

#### Sintoma:

```
{"errors":[{"message":"Unauthorized","code":401}]}
```

#### Diagnóstico:

```
# Testar Bearer Token
curl -H "Authorization: Bearer $TWITTER_BEARER_TOKEN" \
  "https://api.twitter.com/2/tweets/search/recent?query=test&max_results=10"
```

#### Solução:

1. Verificar se o Bearer Token está correto no `.env`
2. Verificar se não há espaços extras no token
3. Regenerar token no Twitter Developer Portal
4. Verificar se a conta de desenvolvedor está ativa

## Reddit API: “Too Many Requests” (429)

### Sintoma:

```
{"error": 429, "message": "Too Many Requests"}
```

### Solução:

```
# A aplicação automaticamente usa dados simulados
# Verificar logs para confirmar
tail -f /var/log/voz-cliente/app.log | grep "Rate limit"

# Reduzir frequência de requests se necessário
# Aguardar reset do rate limit (1 minuto para Reddit)
```

## Timeout de Conexão com APIs

### Sintoma:

```
Error: timeout of 10000ms exceeded
```

### Solução:

```
# Verificar conectividade
ping api.twitter.com
ping oauth.reddit.com

# Verificar proxy/firewall
curl -I https://api.twitter.com/2/tweets/search/recent

# A aplicação automaticamente usa fallback para dados simulados
```

## 4. Problemas de Build e Deploy

### Erro: “Build failed” Next.js

### Sintoma:

```
Error: Build failed with errors
```

### Diagnóstico:

```
# Build com logs detalhados
npm run build -- --debug

# Verificar erros de TypeScript
npx tsc --noEmit

# Verificar erros de ESLint
npm run lint
```

### Solução:

```
# Limpar cache e rebuildar
rm -rf .next
npm run build

# Se erro de memória
export NODE_OPTIONS="--max-old-space-size=4096"
npm run build

# Verificar dependências
npm audit
npm audit fix
```

### Erro: “Port 3000 already in use”

#### Sintoma:

```
Error: listen EADDRINUSE: address already in use :::3000
```

#### Solução:

```
# Encontrar processo usando a porta
sudo lsof -i :3000
sudo netstat -tlnp | grep :3000

# Matar processo
sudo kill -9 <PID>

# Ou usar porta diferente
PORT=3001 npm run dev
```

### Erro: “Permission denied” em produção

#### Sintoma:

```
Error: EACCES: permission denied, open '/opt/voz-cliente/app/.env'
```

#### Solução:

```
# Verificar permissões
ls -la /opt/voz-cliente/app/

# Corrigir propriedade
sudo chown -R voc-app:voc-app /opt/voz-cliente/

# Verificar permissões de arquivos
chmod 644 /opt/voz-cliente/app/.env
chmod 755 /opt/voz-cliente/app/
```

## 5. Problemas de Performance

### Aplicação Lenta

#### Diagnóstico:

```
# Verificar uso de CPU e memória
htop
free -h
df -h

# Verificar logs de performance
tail -f /var/log/voz-cliente/app.log | grep "duration"

# Verificar conexões de banco
sudo -u postgres psql -c "SELECT * FROM pg_stat_activity;"
```

#### Solução:

```
# Otimizar PostgreSQL
sudo -u postgres psql -c "
ALTER SYSTEM SET shared_buffers = '256MB';
ALTER SYSTEM SET effective_cache_size = '1GB';
SELECT pg_reload_conf();
"

# Aumentar recursos do Node.js
export NODE_OPTIONS="--max-old-space-size=2048"

# Verificar índices do banco
sudo -u postgres psql voz_cliente -c "\di"
```

## Alto Uso de Memória

#### Sintoma:

```
# Processo Node.js usando muita RAM
```

#### Solução:

```
# Limitar memória do Node.js
export NODE_OPTIONS="--max-old-space-size=1024"

# Verificar vazamentos de memória
npm install -g clinic
clinic doctor -- npm start

# Reiniciar aplicação periodicamente (PM2)
pm2 start ecosystem.config.js
pm2 set pm2:autodump true
```

## 6. Problemas de Rede e Conectividade

### Erro: "Network timeout"

#### Sintoma:

```
Error: Network timeout
```

#### Diagnóstico:

```
# Testar conectividade
ping google.com
curl -I https://api.twitter.com

# Verificar DNS
nslookup api.twitter.com
dig api.twitter.com

# Verificar proxy/firewall
env | grep -i proxy
```

**Solução:**

```
# Configurar timeout maior (se necessário)
# A aplicação já tem timeout de 10s configurado

# Verificar configurações de rede
sudo systemctl status networking

# Verificar iptables/ufw
sudo ufw status
sudo iptables -L
```

**SSL/TLS Errors****Sintoma:**

```
Error: unable to verify the first certificate
```

**Solução:**

```
# Atualizar certificados
sudo apt update
sudo apt install ca-certificates

# Verificar certificados
openssl s_client -connect api.twitter.com:443

# Se necessário, ignorar SSL (apenas desenvolvimento)
export NODE_TLS_REJECT_UNAUTHORIZED=0
```

**7. Problemas de Logs e Monitoramento****Logs não aparecem****Diagnóstico:**

```
# Verificar se logs estão sendo gerados
ls -la /var/log/voz-cliente/
ls -la /opt/voz-cliente/logs/

# Verificar permissões
sudo ls -la /var/log/voz-cliente/
```

**Solução:**

```
# Criar diretório de logs
sudo mkdir -p /var/log/voz-cliente
sudo chown voc-app:voc-app /var/log/voz-cliente

# Verificar configuração do systemd
sudo systemctl cat voz-cliente

# Verificar se aplicação está escrevendo logs
sudo journalctl -u voz-cliente -f
```

**Logs muito grandes****Solução:**

```
# Configurar logrotate
sudo nano /etc/logrotate.d/voz-cliente

# Conteúdo:
/var/log/voz-cliente/*.log {
    daily
    missingok
    rotate 7
    compress
    delaycompress
    notifempty
    create 644 voc-app voc-app
}

# Testar configuração
sudo logrotate -d /etc/logrotate.d/voz-cliente
```

**8. Problemas de Segurança****Erro: “Blocked by CORS”****Sintoma:**

Access to fetch at '<http://localhost:3000/api/analyze>' from origin '<https://voz-cliente.empresa.com>' has been blocked by CORS policy

**Solução:**

```
# Verificar configuração de CORS no next.config.js
# Adicionar domínio permitido

# Para desenvolvimento
NEXTAUTH_URL=http://localhost:3000

# Para produção
NEXTAUTH_URL=https://voz-cliente.empresa.com
```

**Certificado SSL Expirado****Sintoma:**



SSL certificate problem: certificate has expired

### Solução:

```
# Verificar certificado
sudo certbot certificates

# Renovar certificado
sudo certbot renew

# Verificar renovação automática
sudo systemctl status certbot.timer

# Forçar renovação se necessário
sudo certbot renew --force-renewal
```

## Scripts de Diagnóstico

### Script de Verificação Geral

```
#!/bin/bash
# health-check.sh

echo "=== Verificação de Saúde - Voz do Cliente ==="

# Verificar serviços
echo "1. Verificando serviços..."
systemctl is-active postgresql
systemctl is-active nginx
systemctl is-active voz-cliente

# Verificar conectividade
echo "2. Verificando conectividade..."
curl -s -o /dev/null -w "%{http_code}" http://localhost:3000 || echo "Aplicação não responde"

# Verificar banco de dados
echo "3. Verificando banco de dados..."
sudo -u postgres psql -d voz_cliente -c "SELECT COUNT(*) FROM companies;" 2>/dev/null | echo "Erro no banco"

# Verificar logs recentes
echo "4. Verificando logs recentes..."
tail -n 5 /var/log/voz-cliente/app.log 2>/dev/null || echo "Logs não encontrados"

# Verificar uso de recursos
echo "5. Verificando recursos..."
free -h | grep Mem
df -h | grep -E "/$|/opt"

echo "=== Verificação concluída ==="
```

## Script de Limpeza

```
#!/bin/bash
# cleanup.sh

echo "=== Limpeza do Sistema ==="

# Limpar logs antigos
find /var/log/voz-cliente/ -name "*.log" -mtime +7 -delete 2>/dev/null

# Limpar cache do npm
npm cache clean --force

# Limpar builds antigos
rm -rf /opt/voz-cliente/app/app/.next

# Limpar backups antigos
find /opt/voz-cliente/backup/ -name "*.gz" -mtime +30 -delete 2>/dev/null

# Verificar espaço em disco
df -h

echo "=== Limpeza concluída ==="
```

## Contatos de Suporte

### Níveis de Suporte

#### Nível 1 - Problemas Básicos:

- Verificar documentação
- Consultar logs da aplicação
- Reiniciar serviços

#### Nível 2 - Problemas Técnicos:

- Problemas de configuração
- Erros de integração
- Performance

#### Nível 3 - Problemas Críticos:

- Falhas de segurança
- Corrupção de dados
- Problemas de arquitetura

### Informações para Suporte

Ao reportar problemas, inclua:

1. Descrição do problema
2. Passos para reproduzir
3. Logs relevantes
4. Configuração do ambiente
5. Versão da aplicação

```
# Coletar informações do sistema
echo "=== Informações do Sistema ===" > debug-info.txt
uname -a >> debug-info.txt
node --version >> debug-info.txt
npm --version >> debug-info.txt
psql --version >> debug-info.txt
echo "=== Logs Recentes ===" >> debug-info.txt
tail -n 50 /var/log/voz-cliente/app.log >> debug-info.txt 2>/dev/null
```

## Prevenção de Problemas

### Monitoramento Proativo

1. **Configurar alertas** para uso de recursos
2. **Monitorar logs** regularmente
3. **Verificar certificados SSL** mensalmente
4. **Testar backups** semanalmente
5. **Atualizar dependências** mensalmente

### Manutenção Regular

```
# Script de manutenção semanal
#!/bin/bash

# Atualizar sistema
sudo apt update && sudo apt upgrade -y

# Verificar logs de erro
grep -i error /var/log/voz-cliente/app.log | tail -10

# Verificar espaço em disco
df -h

# Backup automático
/usr/local/bin/backup-voz-cliente.sh

# Reiniciar aplicação se necessário
sudo systemctl restart voz-cliente
```

## Recursos Adicionais

- [Documentação do Next.js](https://nextjs.org/docs) (https://nextjs.org/docs)
- [Documentação do PostgreSQL](https://www.postgresql.org/docs/) (https://www.postgresql.org/docs/)
- [Documentação do Prisma](https://www.prisma.io/docs/) (https://www.prisma.io/docs/)
- [Status das APIs](https://api.twitterstat.us/) (https://api.twitterstat.us/)

Para problemas não cobertos neste guia, entre em contato com a equipe de desenvolvimento com as informações de debug coletadas.