



Instruções de Configuração Z-API

O que é Z-API?

Z-API é uma solução brasileira que permite integrar WhatsApp via WhatsApp Web. Diferente do Twilio (que usa WhatsApp Business API oficial), a Z-API funciona conectando seu próprio número de WhatsApp pessoal ou comercial.

Diferenças entre Twilio e Z-API

Aspecto	Twilio WhatsApp	Z-API
Tipo	WhatsApp Business API oficial	WhatsApp Web
Aprovação	Requer aprovação do WhatsApp/Meta	Imediato
Número	Número dedicado do Twilio	Seu próprio número
Custo	Por mensagem (~\$0.005-0.05)	Plano mensal (~R\$50-200)
Templates	Obrigatório para iniciar conversa	Não necessário
Limitações	Janela de 24h para resposta	Sem limitação
Risco	Baixo (oficial)	Médio (pode ser bloqueado)



Criando uma Instância Z-API

Passo 1: Criar conta na Z-API

1. Acesse <https://z-api.io> (<https://z-api.io>)
2. Clique em **“Criar conta”** ou **“Começar grátis”**
3. Preencha seus dados e confirme o email

Passo 2: Criar uma instância

1. No painel, clique em **“Nova Instância”** ou **“Criar Instância”**
2. Dê um nome para sua instância (ex: “FlowChat Bot”)
3. Escolha o plano desejado (há plano gratuito para testes)

Passo 3: Conectar seu WhatsApp

1. Após criar a instância, clique nela para abrir
2. Aparecerá um **QR Code**

3. No seu celular:
 - Abra o WhatsApp
 - Vá em **Configurações > Aparelhos conectados**
 - Clique em **“Conectar um aparelho”**
 - Escaneie o QR Code
4. Aguarde a conexão ser estabelecida

Obtendo as Credenciais

Instance ID e Token

Após criar a instância, você verá algo como:

```
https://api.z-api.io/instances/XXXXXXXXXXXXX/token/YYYYYYYYYYYYY/  
                                ↑                ↑  
                        INSTANCE_ID        TOKEN
```

Copie esses valores para suas variáveis de ambiente:

- `ZAPI_INSTANCE_ID` = XXXXXXXXXXXXX
- `ZAPI_TOKEN` = YYYYYYYYYYYY

Client-Token (Security Token)

O Client-Token é um token de segurança adicional (recomendado):

1. No painel Z-API, clique na sua instância
2. Vá na aba **“Segurança”** ou **“Security”**
3. Ative o **“Security Token”**
4. Copie o token gerado para `ZAPI_CLIENT_TOKEN`

Configurando o Webhook

Para receber mensagens do WhatsApp, configure o webhook na Z-API:

Passo 1: Acessar configurações

1. No painel Z-API, clique na sua instância
2. Vá na aba **“Webhooks”**

Passo 2: Configurar URL do webhook

Configure a URL do seu servidor:

```
https://seu-dominio.vercel.app/webhook/whatsapp
```

Ou se estiver testando localmente com ngrok:

```
https://xxx-xxx-xxx.ngrok.io/webhook/whatsapp
```

Passo 3: Selecionar eventos

Ative pelo menos estes eventos:

- ☒ **Mensagens recebidas** (ReceivedCallback)
- ☒ **Mensagens enviadas** (SendCallback) - opcional
- ☒ **Status de entrega** (DeliveryCallback) - opcional

Passo 4: Testar webhook

1. Clique em **“Testar Webhook”** no painel Z-API
2. Ou envie uma mensagem de teste pelo WhatsApp
3. Verifique os logs do seu servidor



Exemplo de Requisição e Resposta

Mensagem recebida (webhook)

O Z-API envia para seu webhook:

```
{
  "phone": "5511999999999",
  "fromMe": false,
  "messageId": "3EB0B430D6B33FA1C4FF",
  "moment": 1640995200000,
  "type": "ReceivedCallback",
  "text": {
    "message": "Olá, quero fazer um orçamento"
  }
}
```

Enviando mensagem

Seu servidor envia para Z-API:

```
curl -X POST "https://api.z-api.io/instances/{INSTANCE_ID}/token/{TOKEN}/send-text" \
-H "Content-Type: application/json" \
-H "Client-Token: {CLIENT_TOKEN}" \
-d '{
  "phone": "5511999999999",
  "message": "Olá! Como posso ajudar?"
}'
```

Resposta:

```
{
  "zapiMessageId": "3EB0B430D6B33FA1C4FF",
  "messageId": "3EB0B430D6B33FA1C4FF",
  "id": "3EB0B430D6B33FA1C4FF"
}
```

Boas Práticas de Segurança

1. **Sempre use Client-Token** - Protege seu webhook contra chamadas não autorizadas
2. **Valide o fromMe** - Ignore mensagens onde `fromMe: true` para evitar loops
3. **Use HTTPS** - Nunca exponha seu webhook em HTTP
4. **Monitore a conexão** - O WhatsApp pode desconectar se ficar muito tempo inativo

Limitações e Cuidados

Limitações da Z-API

1. **Conexão via WhatsApp Web**
 - Precisa manter o celular conectado à internet
 - Pode desconectar ocasionalmente
2. **Risco de bloqueio**
 - WhatsApp pode bloquear números que enviam muitas mensagens
 - Evite spam e mensagens em massa
3. **Sem suporte a templates**
 - Não tem templates pré-aprovados como a API oficial
 - Mas também não tem restrição de janela de 24h

Recomendações

- **Para produção pequena/média:** Z-API é excelente
- **Para alto volume:** Considere WhatsApp Business API oficial
- **Mantenha backup:** Tenha um número reserva em caso de bloqueio

Testando Localmente

Com ngrok

1. Instale ngrok: `npm install -g ngrok`
2. Inicie seu servidor local: `uvicorn main:app --reload`
3. Exponha com ngrok: `ngrok http 8000`
4. Configure o webhook na Z-API com a URL do ngrok

Teste manual

```
# Testar processamento de mensagem (sem enviar via Z-API)
curl -X POST http://localhost:8000/api/test/message \
  -H "Content-Type: application/json" \
  -d '{"phone": "5511999999999", "message": "oi"}'

# Enviar mensagem real via Z-API
curl -X POST "http://localhost:8000/api/send?phone=5511999999999&message=teste"

# Verificar status da conexão Z-API
curl http://localhost:8000/zapi/status
```

Migração do Twilio para Z-API

Se você estava usando o Twilio anteriormente, siga estes passos para migrar:

Passo 1: Remover variáveis do Twilio

Localmente (arquivo `.env`):

Remova ou comente as seguintes variáveis do seu arquivo `.env`:

```
# Remova ou comente estas linhas:  
# TWILIO_ACCOUNT_SID=ACxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  
# TWILIO_AUTH_TOKEN=xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  
# TWILIO_WHATSAPP_FROM=whatsapp:+14155238886
```

Na Vercel:

1. Acesse o painel da Vercel
2. Vá em **Settings > Environment Variables**
3. Remova as variáveis:
 - `TWILIO_ACCOUNT_SID`
 - `TWILIO_AUTH_TOKEN`
 - `TWILIO_WHATSAPP_FROM`

Passo 2: Adicionar variáveis Z-API

Localmente (arquivo `.env`):

```
ZAPI_INSTANCE_ID=seu_instance_id  
ZAPI_TOKEN=seu_token  
ZAPI_CLIENT_TOKEN=seu_client_token
```

Na Vercel:

Adicione as mesmas variáveis em **Settings > Environment Variables**.

Passo 3: Fazer novo deploy

Na Vercel, faça um novo deploy para aplicar as mudanças:

```
vercel --prod
```

Nota sobre compatibilidade

A aplicação está configurada para ignorar variáveis extras no arquivo `.env` (como as antigas variáveis do Twilio). Isso facilita a migração gradual, mas é recomendado remover as variáveis não utilizadas para manter o ambiente limpo.



Recursos Úteis

- [Documentação Z-API](https://developer.z-api.io/) (https://developer.z-api.io/)

- [Painel Z-API](https://z-api.io) (https://z-api.io)
- [Status da API](https://status.z-api.io) (https://status.z-api.io)
- [Suporte Z-API](https://z-api.io/suporte) (https://z-api.io/suporte)