



CHECKLIST DE CONFIGURAÇÃO SDR

MANUAL COMPLETO DE IMPLEMENTAÇÃO – TEMPLATE “AGENTEVENDE” NO N8N

OBJETIVO DESTE MANUAL

Este documento orienta passo a passo a implementação do fluxo “AGENTEVENDE” no n8n. Ideal para:

- Aulas práticas de automação
- Times técnicos aplicando IA com LangChain
- Projetos de atendimento automatizado com WhatsApp + Supabase + OpenAI

1. IMPORTAÇÃO DO TEMPLATE

Passo a Passo

1. Acesse o seu **n8n Editor** (ex: <https://editor.n8n.seudominio.com>)
2. Clique no menu de **Importação** (ícone de pasta com seta) no canto superior direito
3. Selecione “**Importar arquivo**” ou **cole o conteúdo JSON** do template
4. O workflow será criado com o nome padrão: [My workflow 10](#)

Dica Didática:

Renomeie o fluxo com um nome significativo, como:

“Fluxo Atendimento WhatsApp – AGENTEVENDE”

2. CONFIGURAÇÃO DE CREDENCIAIS

O fluxo utiliza 3 integrações principais. Crie e vincule as seguintes credenciais:

Tipo	Onde Criar	Observações
OpenAI API	https://platform.openai.com/account/api-keys	Crie uma API Key e configure como “OpenAI account”
Supabase API	Dashboard Supabase (Project > API Keys)	Pegue URL do projeto e a service_role_key
Google Sheets OAuth2	Configurado via conta Google autorizada	Dê acesso ao documento usado no FAQ

Como vincular:

1. Clique no nó que usa a API (ex: “Supabase1”, “OpenAI1” ou “FAQ - GOOGLE SHEETS”)
2. Na aba “**Credenciais**”, selecione a credencial criada

3. TESTE DO WEBHOOK INICIAL

Nó: [Evolution API](#)

- Tipo: [Webhook](#)
- Método: [POST](#)
- URL: </68db7bec-7f2f-4948-be67-54fd27fb0770>

Teste Rápido:

1. Clique em “**Listen for test event**”

2. Envie uma requisição `POST` via Postman com o payload simulado (tem um exemplo salvo no próprio workflow)
 3. Alternativa: Clique em “Set mock data” para testar manualmente
-

4. ENTENDENDO A ESTRUTURA DO FLUXO

O fluxo está dividido logicamente com Sticky Notes que facilitam o entendimento:

FLUXO 1 – Entrada e Tratamento de Mensagem

- Recebe dados da mensagem do WhatsApp
- Verifica tipo: texto, imagem ou áudio
- Prepara os dados para processamento

FLUXO 2 – Cadastro e Thread

- Consulta Supabase para verificar se lead existe
- Cria uma nova thread se necessário
- Salva ou atualiza informações na tabela `LEADS`

AGENTE DE I.A – Processamento OpenAI

- Usa LangChain Tools (SDR) com assistente configurado
- Executa lógica contextual com prompt personalizado
- Chama tools específicas (áudio, FAQ, agendamento, etc)

RETORNO – Envio da Resposta

- Divide mensagens longas (`Code4`)
 - Calcula delay (`Contabilizar Caracteres`)
 - Envia resposta com `HTTP Request` via Evolution API
-

5. CONDIÇÕES LÓGICAS & FLUXO DE TIPO DE MENSAGEM

Nó: `Tipo de Mensagem` (Switch Node)

Define o caminho dependendo do tipo de mensagem:

Tipo Recebido	Ramo Ativado
<code>conversation</code>	Mensagem de texto simples
<code>audioMessage</code>	Aciona transcrição e processamento de áudio
<code>imageMessage</code>	Aciona análise de imagem via OpenAI

6. ESTRUTURA DA TABELA SUPABASE

Tabela `LEADS` deve conter os campos:

Campo	Tipo	Finalidade
<code>number</code>	String	Número do lead (WhatsApp ID)
<code>thread_id</code>	String	ID da thread do assistente
<code>created_at</code>	DateTime	Quando o lead foi criado
<code>timeout</code>	DateTime	Tempo de expiração do contexto
<code>lead_name</code>	String	Nome do cliente

 Certifique-se de que os nomes batem exatamente com os usados no workflow.

7. CONFIGURAÇÃO DAS TOOLS (LangChain Tools)

As ferramentas da IA são chamadas via o nó `SDR`, usando `ai_tool`.

Ferramentas Disponíveis:

Tool	Finalidade
Áudio	Envia áudio com resposta do agente
FAQ	Consulta dúvidas frequentes via Google Sheets
Agendamento	Agenda reuniões
Reagendamento	Atualiza horário agendado
Cancelamento	Cancela reuniões
Disponibilidade	Verifica horários disponíveis
dia_semana	Determina o dia da semana por data
calculator	Realiza cálculos automáticos

Detalhes Técnicos:

- Cada Tool usa `workflowId` de outro fluxo
- Passa dados como `nome`, `email`, `hora`, `histórico` etc.
- O `Historico_conversas` vem do campo `Mensagem Principal`

8. PROCESSAMENTO DE RESPOSTA E ENVIO FINAL

Divisão de Texto

- Código JS em `Code4` separa a mensagem em partes menores (200+ caracteres)

Delay

- Nó `Contabilizar Caracteres` calcula delay proporcional ao tamanho da mensagem

Envio

- `Enviar Texto1`: envia mensagem formatada para Evolution API
- O delay simula "digitando...", deixando o bot mais humano

9. VALIDAÇÃO FINAL

Simule os seguintes cenários para testar tudo:

1.  Texto simples
2.  Áudio do lead
3.  Imagem enviada
4.  Pergunta de FAQ
5.  Agendamento completo

Verifique:

- Dados inseridos no Supabase?
- Thread criada com sucesso?
- IA respondeu corretamente?
- Tool acionada automaticamente?

10. ATIVAÇÃO DO WORKFLOW

1. Certifique-se de que **todos os nós estão sem erros** (ícones vermelhos)
2. Clique em "**Activate**"
3. Faça o webhook funcionar em produção sem "Listen for Test Event"

ERROS COMUNS & SOLUÇÕES

Erro	Causa	Solução
✗ Webhook não recebe	Está em modo Test	Ative e teste em modo Produção
✗ API Key inválida	Key incorreta ou não atribuída	Crie nova e vincule no nó
✗ Dados não aparecem na Supabase	Campos incorretos	Verifique nomes e tipos de campo
✗ Tool LangChain não responde	Falta workflowId ou variável	Revise campo <code>workflowId</code> e variáveis
✗ Mensagem final não enviada	Delay ou texto quebrado mal feito	Teste cada trecho isolado com <code>Execute Node</code>