

## Relatório 5 - Prática: Construindo um fluxo n8n(II)

Guilherme Mileib

### Descrição da atividade

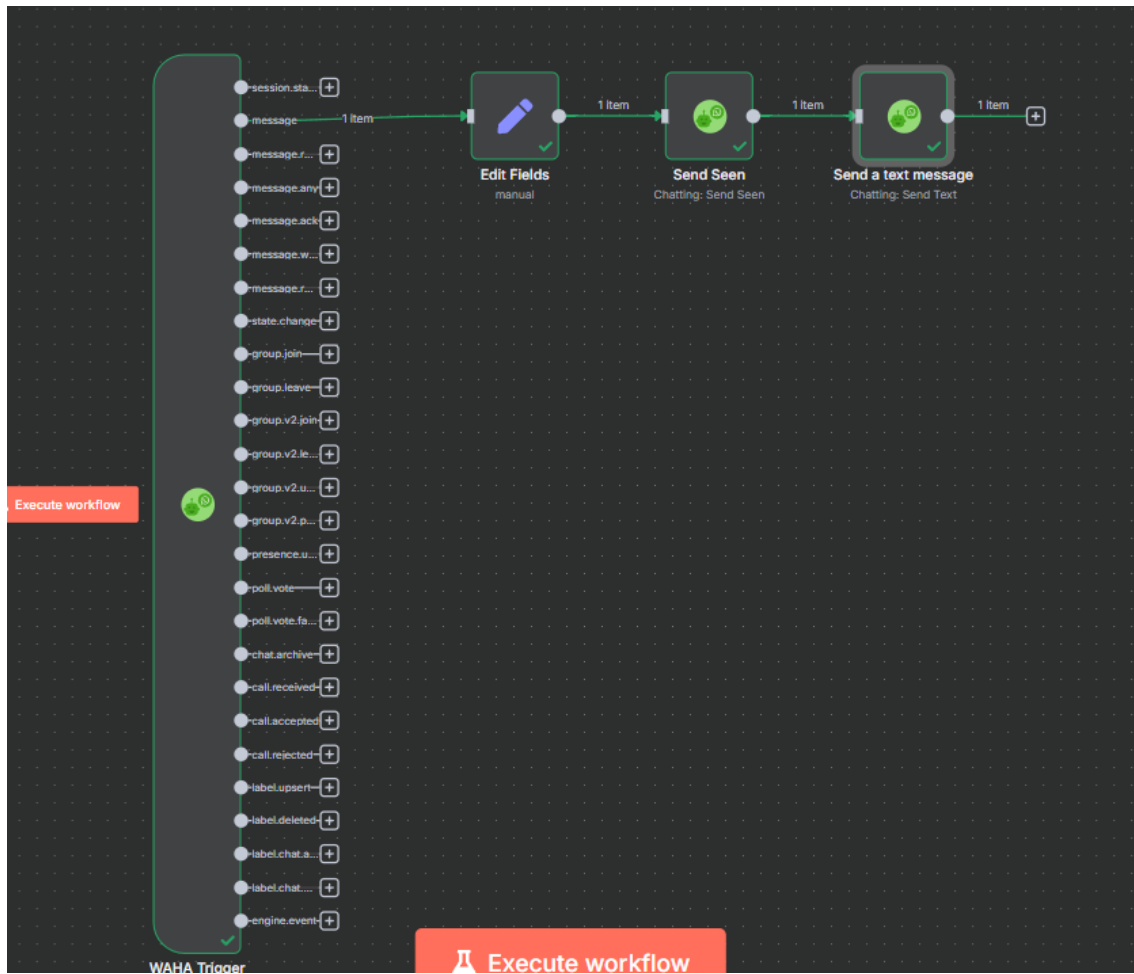
Nesse card pude realizar a implementação de um agente com Low Code, dessa vez com o n8n. Que permite baixar extensões, como usamos da API de whatsapp do Waha. Que pudemos acessar tanto pelo n8n, baixando seus nodes e pelo Docker baixando suas dependências, uma API bastante completa e fácil de utilizar. Muito útil para criar FAQs simples para sistemas de empresa.

### Sobre a parte prática 0.1:

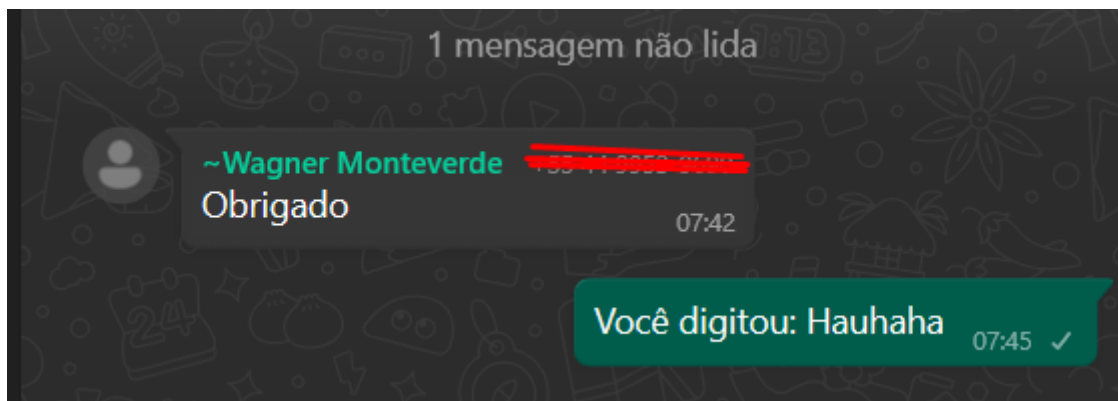
Primeiro configuramos o trigger de request do Waha no work flow:

The screenshot displays the n8n workflow editor. On the left, the 'Waha Trigger' node is configured with a 'Webhook URL' of 'http://localhost:5678/webhook/c9a0b112-1883-499e-829b-32cd00444856/waha'. Below this, a list of events is shown, including 'session.status', 'message', 'message.reaction', 'message.any', 'message.ack', 'message.waiting', 'message.revoked', 'state.change', 'group.join', 'group.leave', 'group.v2.join', 'group.v2.leave', 'group.v2.update', 'group.v2.participants', 'presence.update', 'poll.vote', 'poll.vote.failed', and 'chat.archive'. On the right, the 'OUTPUT' section shows the JSON data received from the trigger, which includes fields like 'id', 'timestamp', 'event', 'session', 'metadata', 'me', and 'payload'.

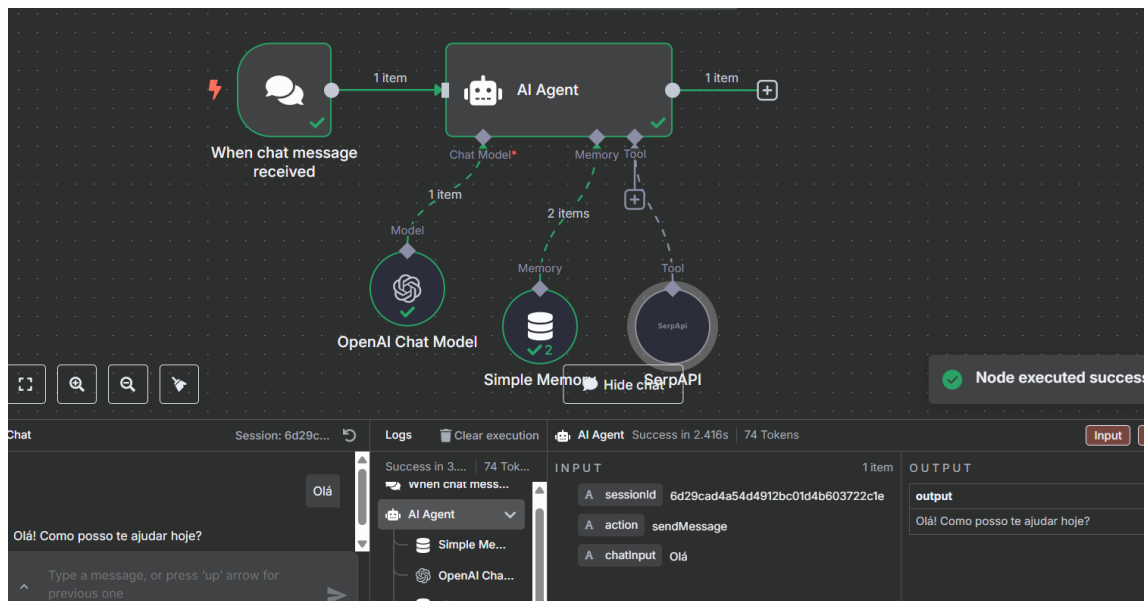
```
{
  "id": "evt_01x045xp24ybt906y8natty01",
  "timestamp": 1752487090244,
  "event": "session.status",
  "session": "default",
  "metadata": {
  },
  "me": null,
  "payload": {
    "name": "default",
    "status": "STARTING"
  },
  "engine": "NOMEB",
  "environment": {
    "version": "2025.7.5",
    "engine": "NOMEB",
    "tier": "CORE",
    "browser": null
  }
}
```



Após o fluxo funcionar:



Sobre a aplicação 0.2:



Nesse projeto utilizei um nó para escutar as mensagens, um nó de agente de AI que apenas toma as decisões com base na entrada do usuário os outros nós que ele tem ligado. Aproveitando a mesma API que usei no Langchain, utilizei a key da openAI, mas poderia ter usado uma da groq também. Utilizei uma memória simples de buffer, das últimas 5 mensagens mandadas. Por fim utilizei a key da SerpAPI, gratuita para caso fizesse alguma busca na web.

## Dificuldades

Nesse projeto eu já estava familiarizado com os softwares utilizados na aplicação como n8n que já havia usado e o Docker que já utilizei em outras aplicações e já vinha usando-o com n8n no localhost. Acredito que por isso não tive nenhuma dificuldade no projeto.

## Conclusões

Diferente do último card, que aplicamos um fluxo com langchain linkando o chatbot, o n8n parece ser muito mais fácil de configurar. Além do mais a totalidade de aplicação de chatbots evoluiu bastante, com antes a maioria das pessoas usando faq's com processamentos de cliques em cards no whatsapp hoje temos o processamento do pensamento por si só realizado pelas máquinas.

Eu não juntei os dois workflows nessa aplicação devido ao tempo, mas com certeza tentarei utilizar a API da Waha e os workflows de chatbot com API da OpenAI para chats de repostas. Já pesquisei também e vi que possui outras APIs para whatsapp muito interessantes como a da Evolution, que também nós de extensão no n8n.

## Referencias

- <https://blog.n8n.io/how-to-make-ai-chatbot/> (Site do projeto de passo a passo do chatbot)
- [https://www.youtube.com/watch?v=FyQivMjb3\\_8](https://www.youtube.com/watch?v=FyQivMjb3_8) (fluxo n8n com api do whatsapp)
- <https://waha.devlike.pro/docs/overview/introduction/> (documentação Waha)