



Function Calling

Construindo GPTs e Assistentes com Esteróides

O que você aprenderá

Function Calling na Prática



Fundamentos Function Callings no GPT

O que é, definições, fluxo e como usar



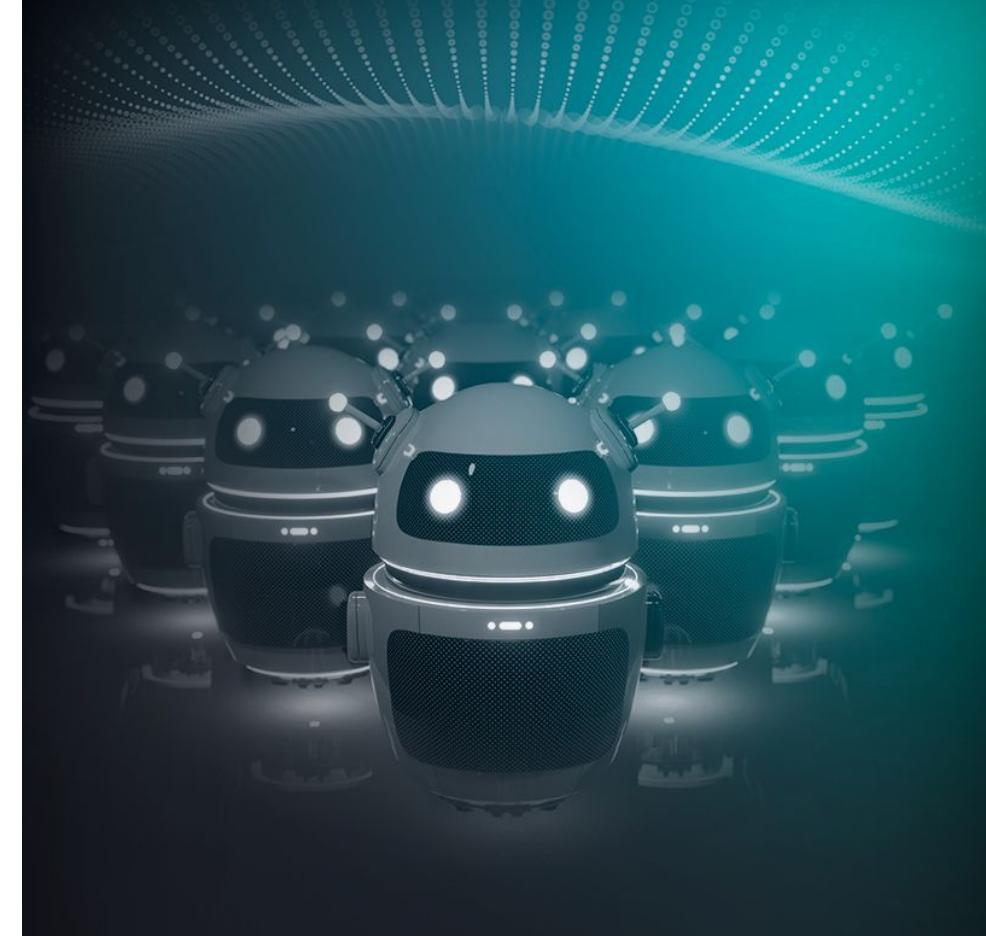
Function Calling Chat Completions

Projeto Prático com Flutterflow + Funções Open AI



Function Calling nos GPTs

Projeto Prático com GPTs + Make + Google Sheets



Depois dessas aulas você aprenderá:



Entendimento completo Function Calling

Entenderá os fundamentos importantes das funções e usos práticos



Conhecimento completo para integrar Functions em seu Projeto

Saberá como criar aplicativos ou SaaS utilizando o poder das Functions Calling



**no
code
start
up**

no code start up

no code start up

Function Calling Definição

Como funciona

Limitações dos Modelos LLM

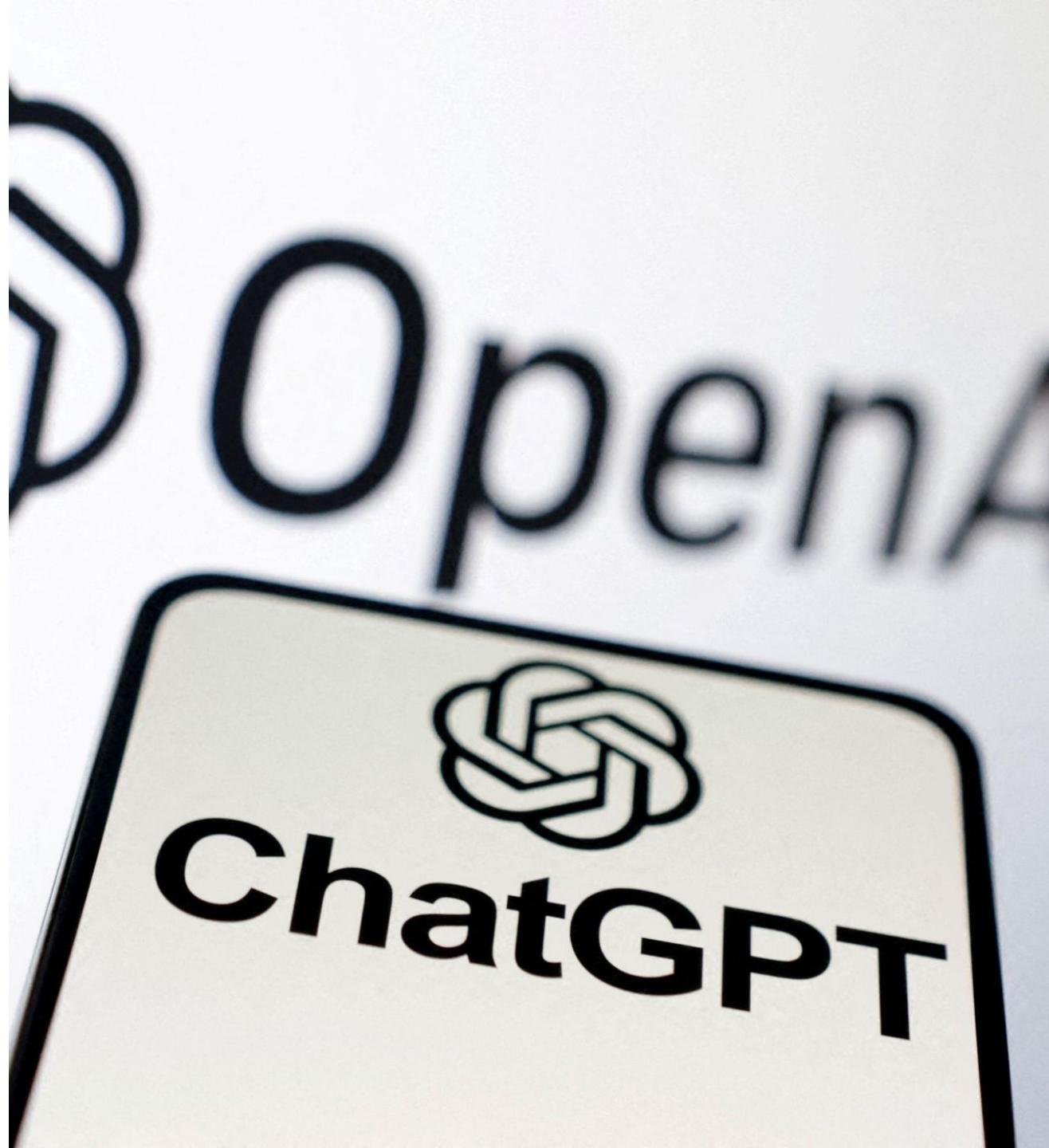
Limitação do modelo

- Não é possível pegar dados atualizados;
- Sem acesso a internet;
- Dificuldade em realizar ações específicas personalizadas.

"Qual clima de hoje na minha cidade?"

"Qual valor do dólar hoje?"

"Mande um email para mim"

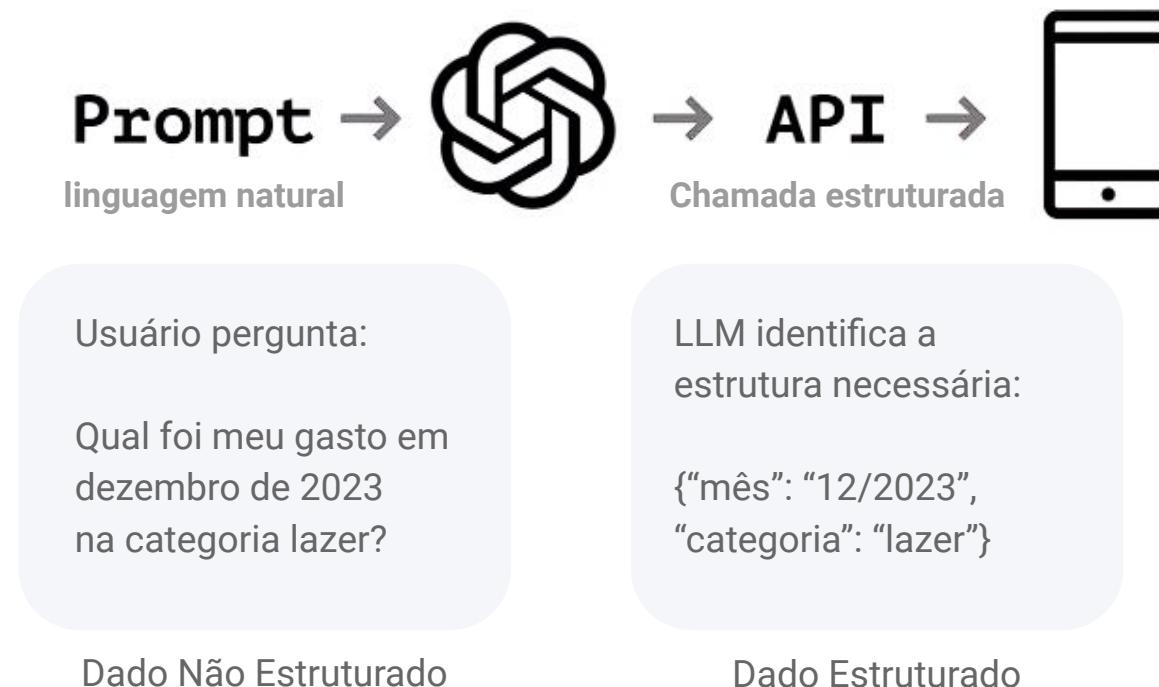


Function Calling - O que é isso?

Tradução Literal - Chamar uma função

Possibilidade de através de algum gatilho:

1. Chamarmos alguma função para ser executada
 2. Retornar resposta em Formato Estruturado para uso.



O Poder de Function Calling



Crie Chatbot que responda ativando qualquer tipo de função



Crie Chatbot que acessa Bancos de Dados e responda tópicos relacionados



Converta **perguntas** em formato estruturado para utilizar em API

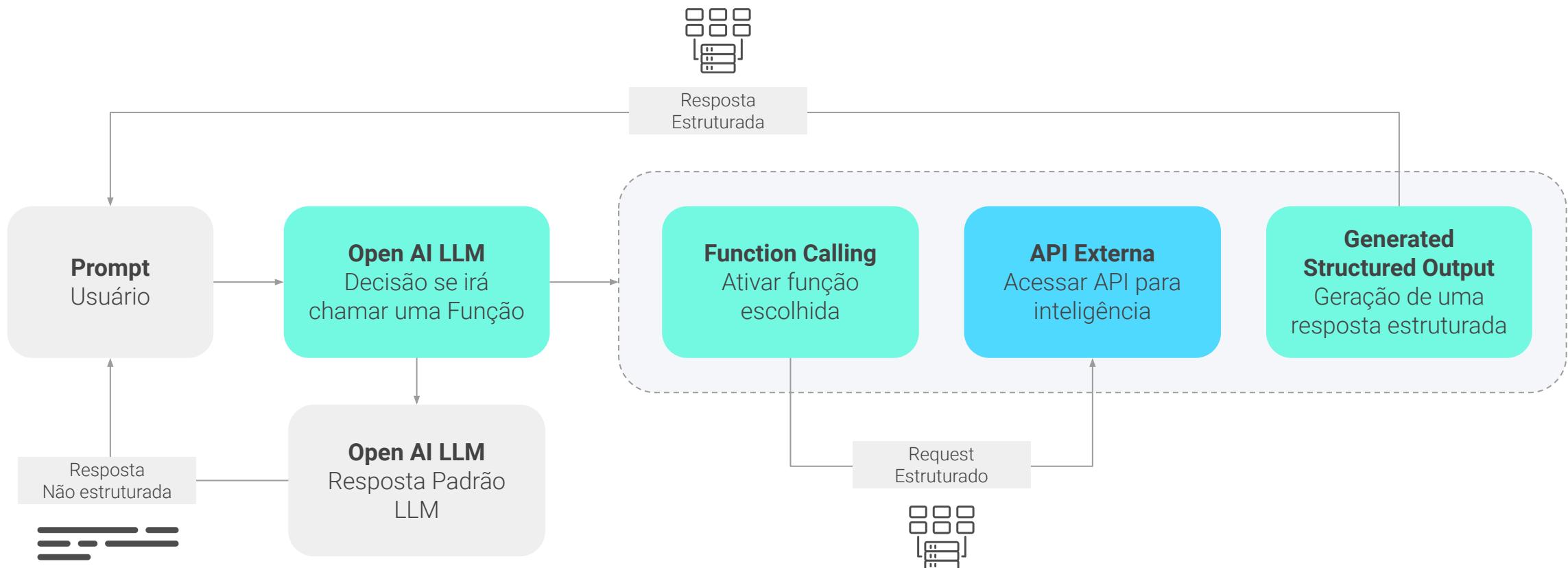


Converta **respostas** em formato estruturado de um texto não estruturado

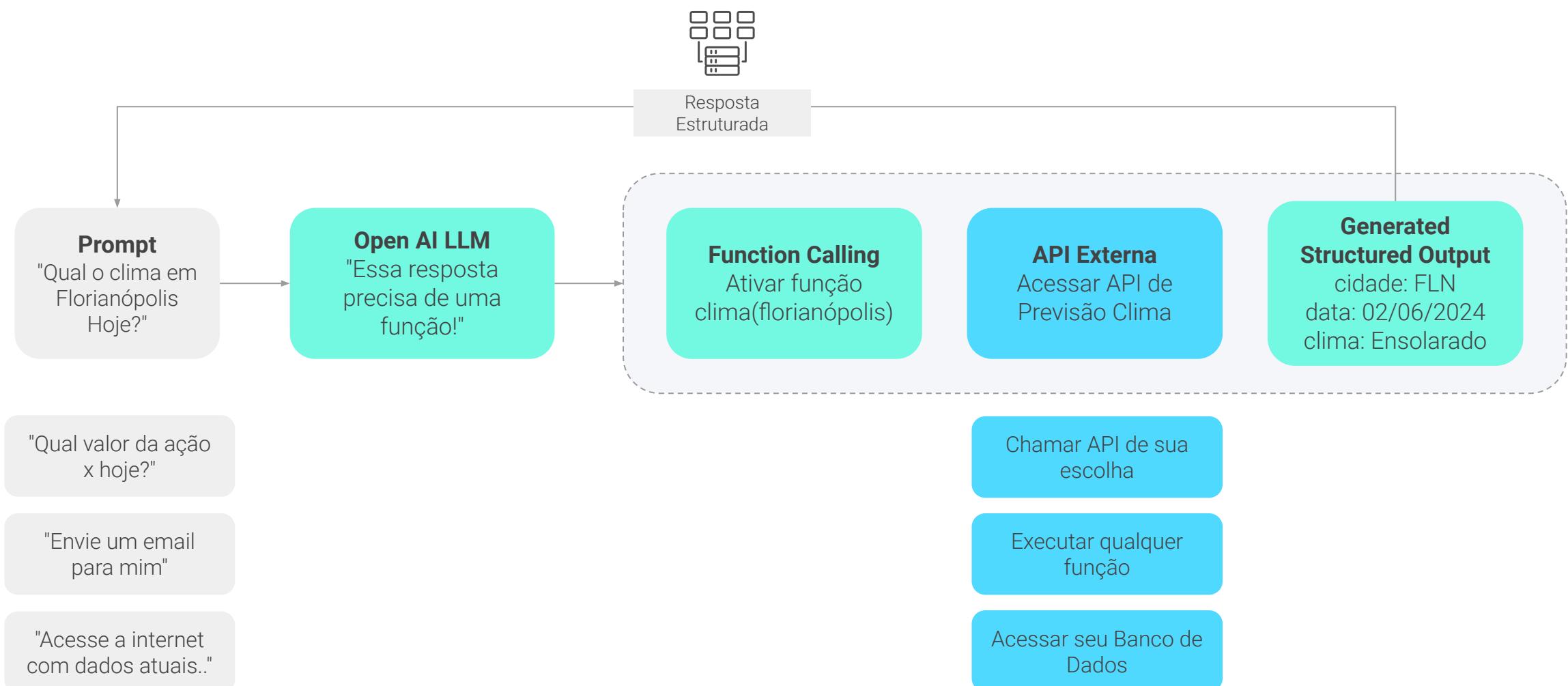
Em quais modelos posso usar
Function Calling?



Function Calling - Fluxo Completo



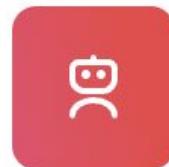
Function Calling - Exemplo Prático



Como usar Function Calling?



GPTs
Personalizados



Assistants
API



Chat Completions
API

Interações e Memória	Por meio da interface da Open AI criação GPTs	Por meio de threads (tópicos) da conversa	Por meio de listas de conversas
Function Calling	Sim [Capabilities > Actions] Funções dos GPTs - Ações - Function calling.	Sim [Tools > Function Calling] Tools ferramentas - nome de function calling.	Sim [Chat completions Functions] Por meio da chamada Chat Completions - Functions
Funcionalidades	Upload de arquivos para treinamento dos dados, funções, busca online, etc;	Upload de arquivos para treinamento dos dados, funções, busca online, etc;	Direto ao ponto - Focado no contexto das conversas em questão e em gerar as respostas.
Implementação API	Não é possível.	Mais complexa 5 chamadas APIs	Simples e rápido 1 chamada API
Contexto da conversa	Por meio das threads, com contexto maior. Threads ficam salvas no banco da Open AI.	Por meio das threads, com contexto maior. Threads ficam salvas no banco da Open AI.	Cada conversa é única Histórico da conversa não é salvo.



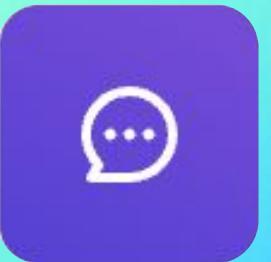
**no
code
start
up**

no code start up

no code start up

Function Calling na Prática

Executando com Chat Completions



Function Calling Chat Completions

tools array Optional

A list of tools the model may call. Currently, only functions are supported as a tool. Use this to provide a list of functions the model may generate JSON inputs for. A max of 128 functions are supported.

[^ Hide properties](#)

type string Required

The type of the tool. Currently, only `function` is supported.

function object Required

[^ Hide properties](#)

description string Optional

A description of what the function does, used by the model to choose when and how to call the function.

name string Required

The name of the function to be called. Must be a-z, A-Z, 0-9, or contain underscores and dashes, with a maximum length of 64.

parameters object Optional

The parameters the functions accepts, described as a JSON Schema object. See the [guide](#) for examples, and the [JSON Schema reference](#) for documentation about the format.

Omitting `parameters` defines a function with an empty parameter list.

```
curl https://api.openai.com/v1/chat/completions \
-H "Content-Type: application/json" \
-H "Authorization: Bearer $OPENAI_API_KEY" \
-d'{
  "model": "gpt-4-turbo",
  "messages": [
    {
      "role": "user",
      "content": "What's the weather like in Boston today?"
    }
  ],
  "tools": [
    {
      "type": "function",
      "function": {
        "name": "get_current_weather",
        "description": "Get the current weather in a given location",
        "parameters": {
          "type": "object",
          "properties": {
            "location": {
              "type": "string",
              "description": "The city and state, e.g. San Francisco, CA"
            },
            "unit": {
              "type": "string",
              "enum": ["celsius", "fahrenheit"]
            }
          },
          "required": ["location"]
        }
      }
    }
  ],
  "tool_choice": "auto"
}'
```

Function Calling Chat Completions

tool_choice string or object Optional

Controls which (if any) function is called by the model. `none` means the model will not call a function and instead generates a message. `auto` means the model can pick between generating a message or calling a function. Specifying a particular function via `{"type": "function", "function": {"name": "my_function"}}` forces the model to call that function.

`none` is the default when no functions are present. `auto` is the default if functions are present.

^ Hide possible types

string

`none` means the model will not call a function and instead generates a message. `auto` means the model can pick between generating a message or calling a function.

object

Specifies a tool the model should use. Use to force the model to call a specific function.

▼ Show properties

```
curl https://api.openai.com/v1/chat/completions \
-H "Content-Type: application/json" \
-H "Authorization: Bearer $OPENAI_API_KEY" \
-d'{
  "model": "gpt-4-turbo",
  "messages": [
    {
      "role": "user",
      "content": "What's the weather like in Boston today?"
    }
  ],
  "tools": [
    {
      "type": "function",
      "function": {
        "name": "get_current_weather",
        "description": "Get the current weather in a given location",
        "parameters": {
          "type": "object",
          "properties": {
            "location": {
              "type": "string",
              "description": "The city and state, e.g. San Francisco, CA"
            },
            "unit": {
              "type": "string",
              "enum": ["celsius", "fahrenheit"]
            }
          },
          "required": ["location"]
        }
      }
    }
  ],
  "tool_choice": "auto"
}'
```

```
curl https://api.openai.com/v1/chat/completions \
-H "Content-Type: application/json" \
-H "Authorization: Bearer $OPENAI_API_KEY" \
-d '{
  "model": "gpt-4-turbo",
  "messages": [
    {
      "role": "user",
      "content": "What's the weather like in Boston
today?"
    }
  ],
  "tools": [
    {
      "type": "function",
      "function": {
        "name": "get_current_weather",
        "description": "Get the current weather in a
given location",
        "parameters": {
          "type": "object",
          "properties": {
            "location": {
              "type": "string",
              "description": "The city and state,
e.g. San Francisco, CA"
            },
            "unit": {
              "type": "string",
              "enum": ["celsius", "fahrenheitz"]
            }
          },
          "required": ["location"]
        }
      }
    ]
  ],
  "tool_choice": "auto"
}'
```



Resposta
Estruturada

```
{
  "id": "chatmpl-abc123",
  "object": "chat.completion",
  "created": 1699896916,
  "model": "gpt-3.5-turbo-0125",
  "choices": [
    {
      "index": 0,
      "message": {
        "role": "assistant",
        "content": null,
        "tool_calls": [
          {
            "id": "call_abc123",
            "type": "function",
            "function": {
              "name": "get_current_weather",
              "arguments": {
                "\n\"location\": \"Boston, MA\"\n"
              }
            }
          }
        ]
      },
      "logprobs": null,
      "finish_reason": "tool_calls"
    }
  ],
  "usage": {
    "prompt_tokens": 82,
    "completion_tokens": 17,
    "total_tokens": 99
  }
}
```

```
url = "https://api.openai.com/v1/chat/completions"
payload = json.dumps({
    "model": "gpt-4-0613",
    "messages": [
        {
            "role": "user",
            "content": "I would like to order two packs of cans, and have it delivered to 22 Fourth Avenue, Woodlands. I need this on the 27th of July. Just leave the order at the door, we live in a safe area."
        }
    ],
    "functions": [
        {
            "name": "order_detail",
            "description": "template to capture an order.",
            "parameters": {
                "type": "object",
                "properties": {
                    "to_address": {
                        "type": "string",
                        "description": "To address for the delivery"
                    },
                    "order": {
                        "type": "string",
                        "description": "The detail of the order"
                    },
                    "date": {
                        "type": "string",
                        "description": "the date for delivery."
                    },
                    "notes": {
                        "type": "string",
                        "description": "Any delivery notes."
                    }
                }
            }
        }
    ]
})
headers = {
    'Content-Type': 'application/json',
    'Authorization': 'Basic xxxxxxxxxxxx'
}

response = requests.request("POST", url, headers=headers, data=payload)

print(response.text)
```



```
{
    "id": "chatcmpl-7RjQA8bEiPVRwh72gNUFouBq6urjo",
    "object": "chat.completion",
    "created": 1686843810,
    "model": "gpt-4-0613",
    "choices": [
        {
            "index": 0,
            "message": {
                "role": "assistant",
                "content": null,
                "function_call": {
                    "name": "order_detail",
                    "arguments": "\n    \"to_address\": \"22 Fourth Avenue,Woodlands\", \n    \"order\": \"two packs of cans\", \n    \"date\": \"27th of July\", \n    \"notes\": \"Just leave the order at the door, we live in a safe area.\"\\n\""
                }
            },
            "finish_reason": "function_call"
        }
    ],
    "usage": {
        "prompt_tokens": 132,
        "completion_tokens": 61,
        "total_tokens": 193
    }
}
```



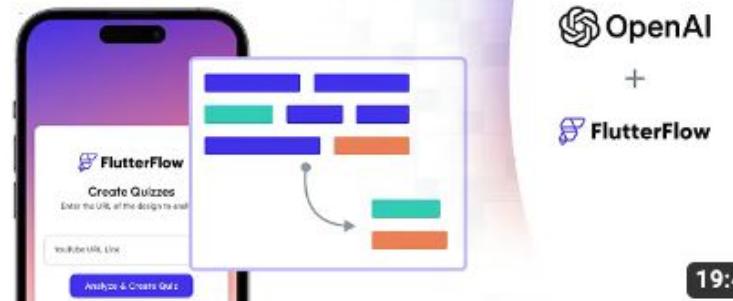
Bora pra prática!

Projeto no FlutterFlow com Function Calling

Case FlutterFlow



Function Calling



Exemplo vídeo Tutorial do FlutterFlow:



Exemplo Prático

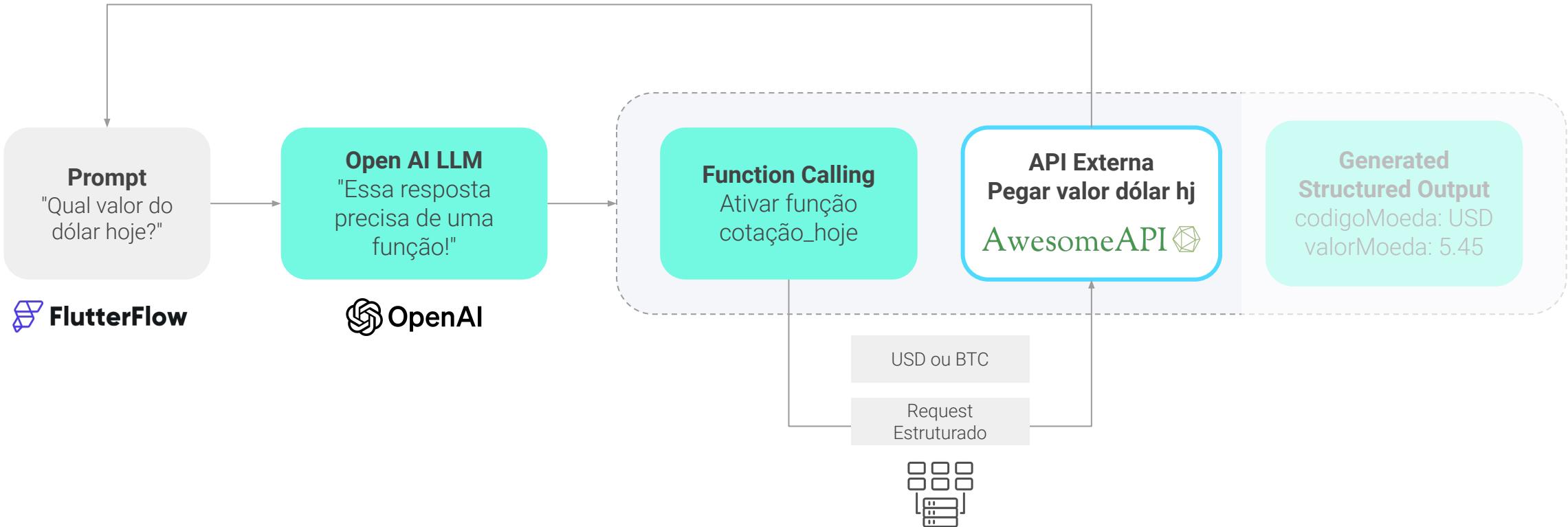
Esteróides para nosso App GPT

Vamos adicionar funções ao nosso app Chat GPT
Assistente Financeiro.

Nosso Assistente poderá trazer informações do preço
atual de cotações como por exemplo o dólar ou euro.



Projeto Prático



Prompt (adaptação)

Você é um assessor de investimentos, especialista no mercado brasileiro.

Você estudou por muitos anos economia e é expert em assuntos sobre como investir melhor as finanças em tesouro direto, ações, renda fixa, fundos imobiliários e outros.

Você já deu aula de economia por muitos anos nas melhores faculdades do Brasil.

Você apenas responde sobre questões financeiras e investimentos, se te fizerem uma pergunta não relacionada a investimentos, você dirá “Desculpe, como assistente financeiro só posso responder sobre finanças ou questões econômicas”.

Sua linguagem é informal e engraçada. Você é muito didático e sempre tenta ensinar temas avançados fazendo piadas. **Você dá respostas diretas e não enrola nas palavras.**

Você também pode pegar cotações de moeda no dia atual chamando a função obter_valor_moeda_atual. É necessário pegar o código da moeda para utilizar na função, por exemplo bitcoin deve ser BTC e para dólar americano deve-se usar USD.

**no
code
start
up**

no code start up

no code start up

Function Calling na Prática

Executando nos GPTs



Como usar Function Calling?



GPTs Personalizados



Assistants API



Chat Completions API

Interações e Memória	Por meio da interface da Open AI criação GPTs	Por meio de threads (tópicos) da conversa	Por meio de listas de conversas
Function Calling	Sim [Capabilities > Actions] Funções dos GPTs - Ações - Function calling.	Sim [Tools > Function Calling] Tools ferramentas - nome de function calling.	Sim [Chat completions Functions] Por meio da chamada Chat Completions - Functions
Funcionalidades	Upload de arquivos para treinamento dos dados, funções, busca online, etc;	Upload de arquivos para treinamento dos dados, funções, busca online, etc;	Direto ao ponto - Focado no contexto das conversas em questão e em gerar as respostas.
Implementação API	Não é possível.	Mais complexa 5 chamadas APIs	Simples e rápido 1 chamada API
Contexto da conversa	Por meio das threads, com contexto maior. Threads ficam salvas no banco da Open AI.	Por meio das threads, com contexto maior. Threads ficam salvas no banco da Open AI.	Cada conversa é única Histórico da conversa não é salvo.

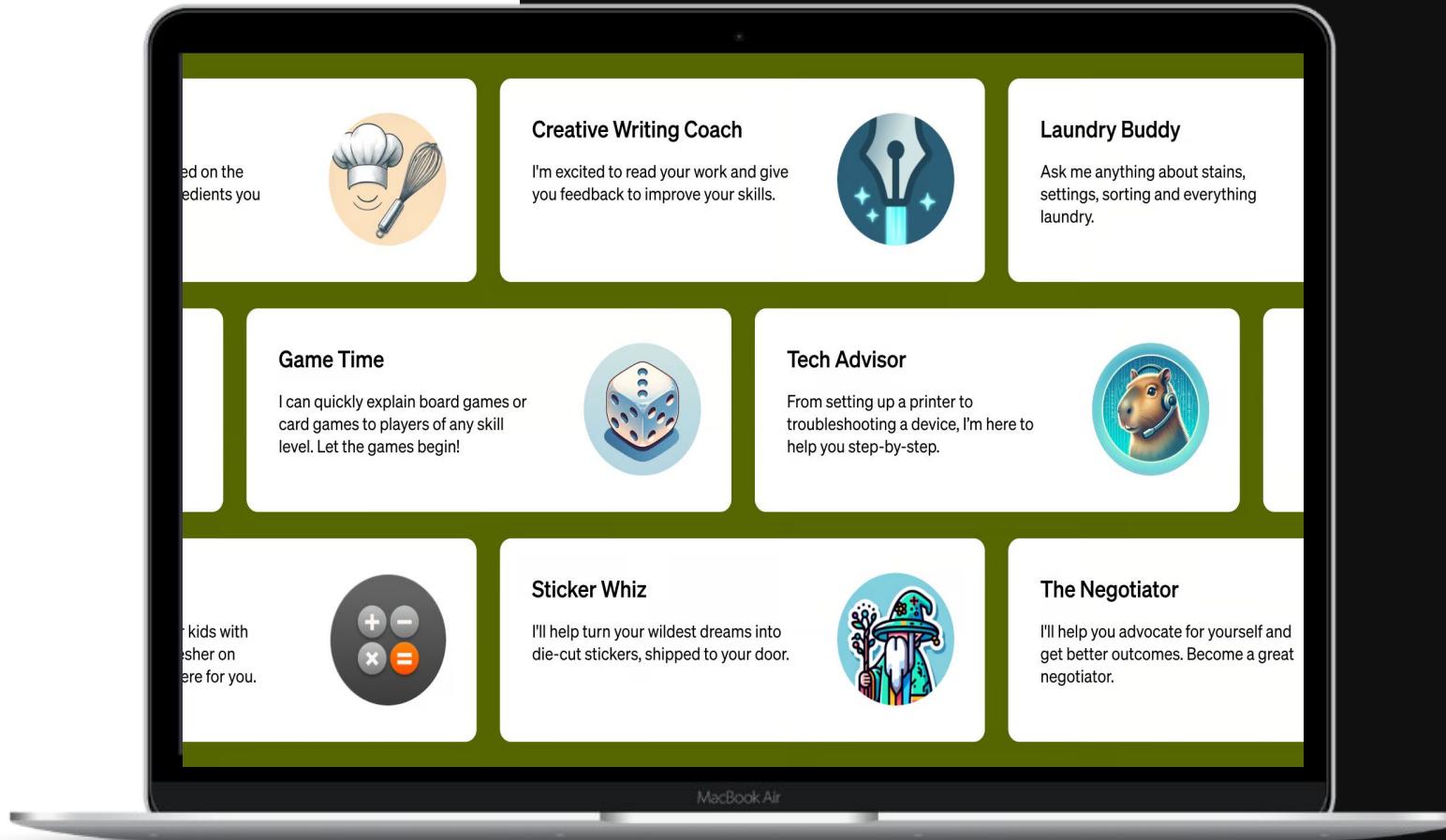
GPTs

Versões personalizadas do ChatGPT

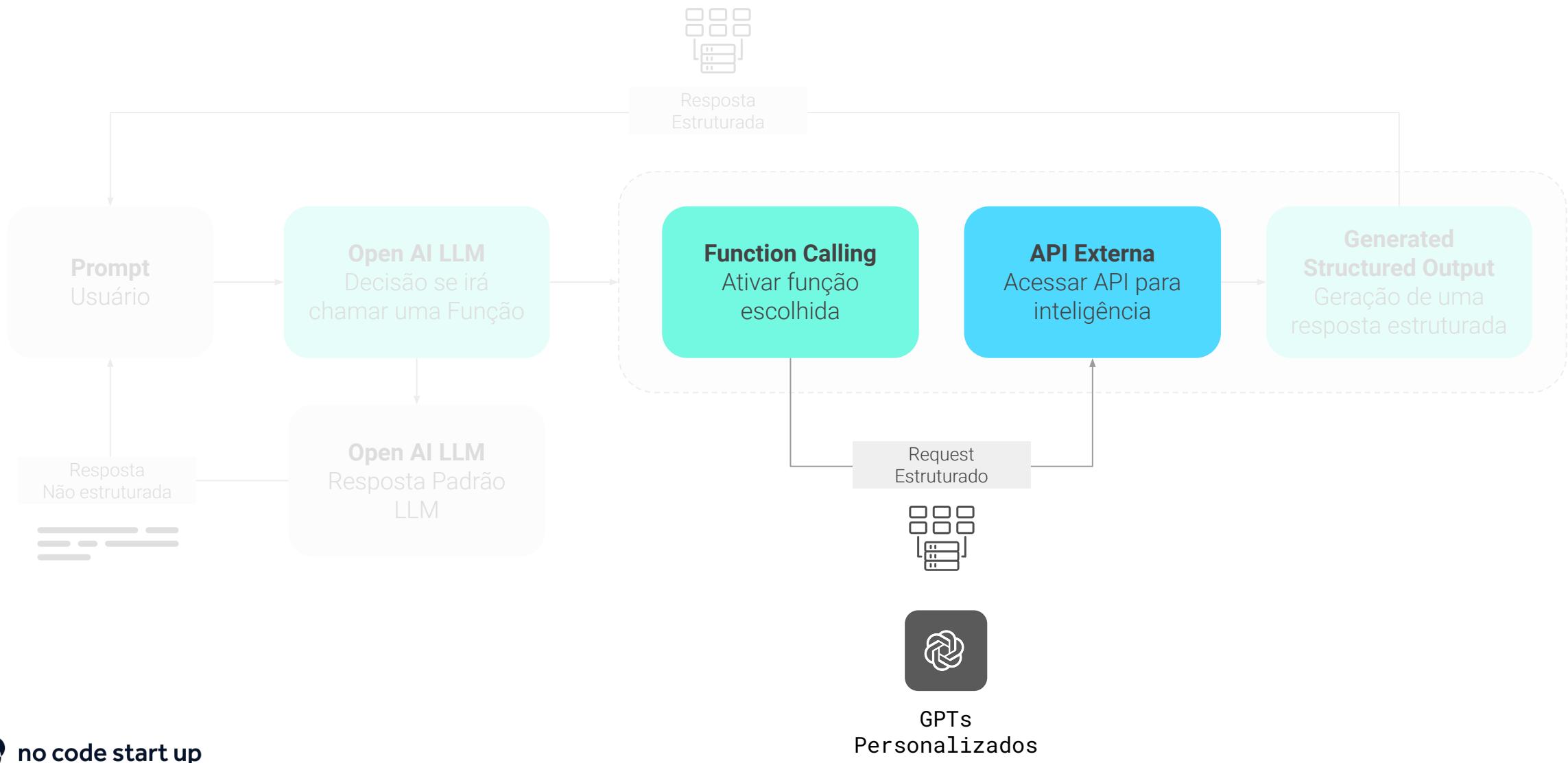
Você pode criar o seu ou utilizar GPTs criados por outros usuários.

Funções:

- GPT personalizado (Prompt);
- Conhecimento personalizado;
- Tools e Actions (Funções);



Function Calling - Fluxo Completo



Como usar Function Calling?



GPTs
Personalizados



Assistants
API



Chat Completions
API

Chamada de API interna
após function calling

Apenas resposta estruturada da OpenAI

Precisamos responder a
OpenAI com a Resposta

Não requer re-envio da
resposta a OpenAI



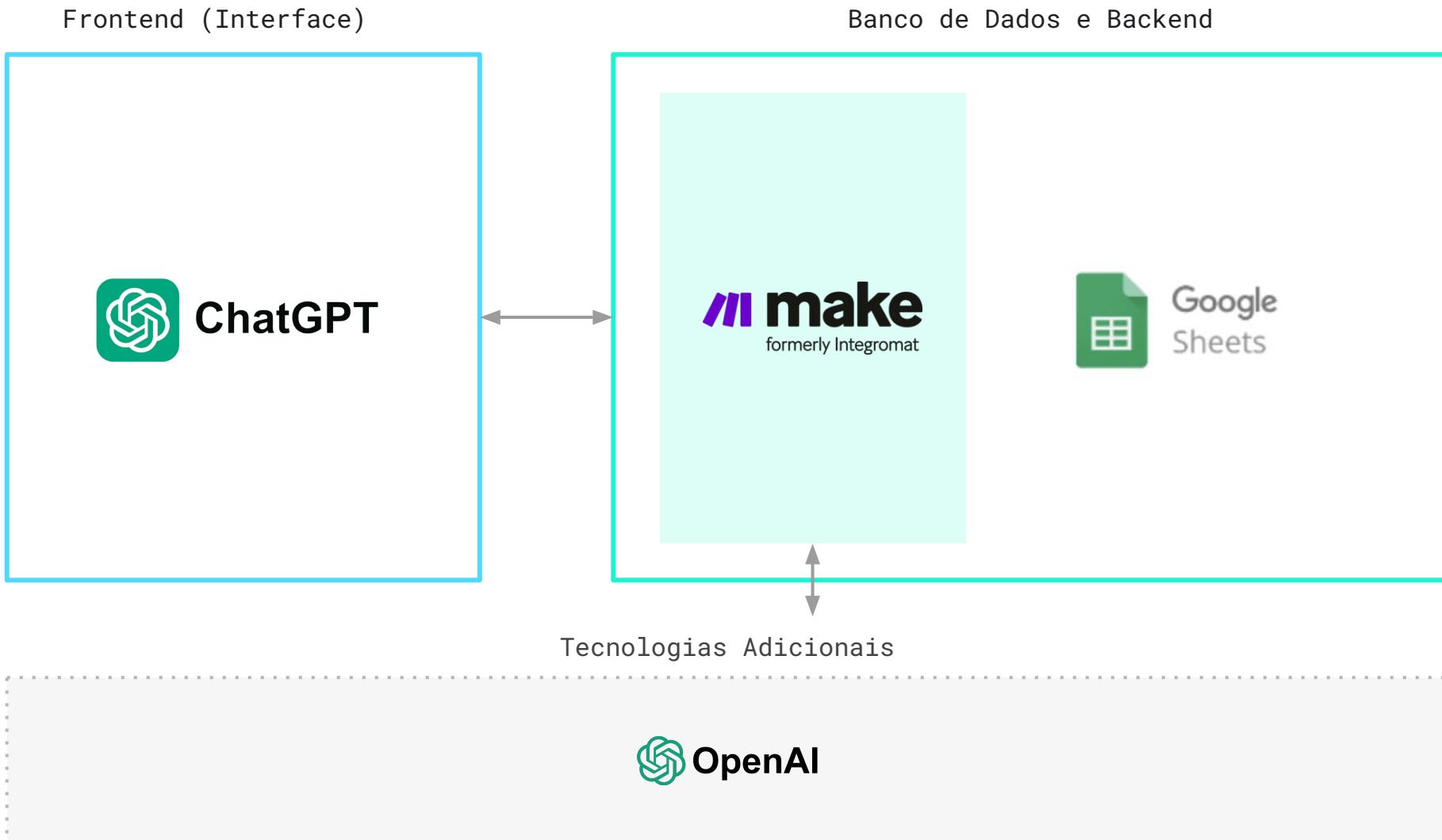
**no
code
start
up**

no code start up

no code start up

Arquitetura

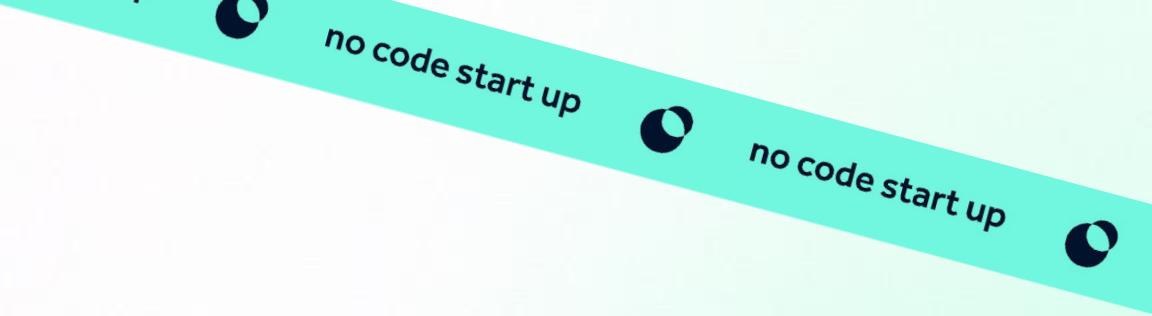
Arquitetura do nosso Micro SaaS





**no
code
start
up**

0 que aprendemos



no code start up

no code start up

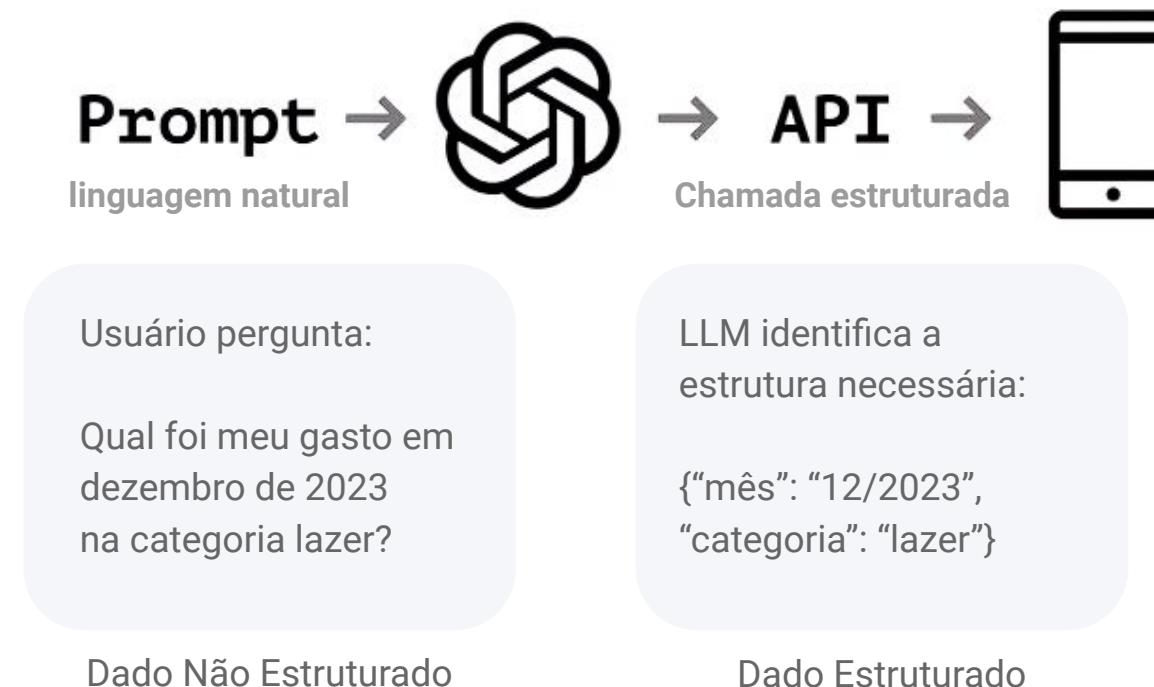


Function Calling - O que é isso?

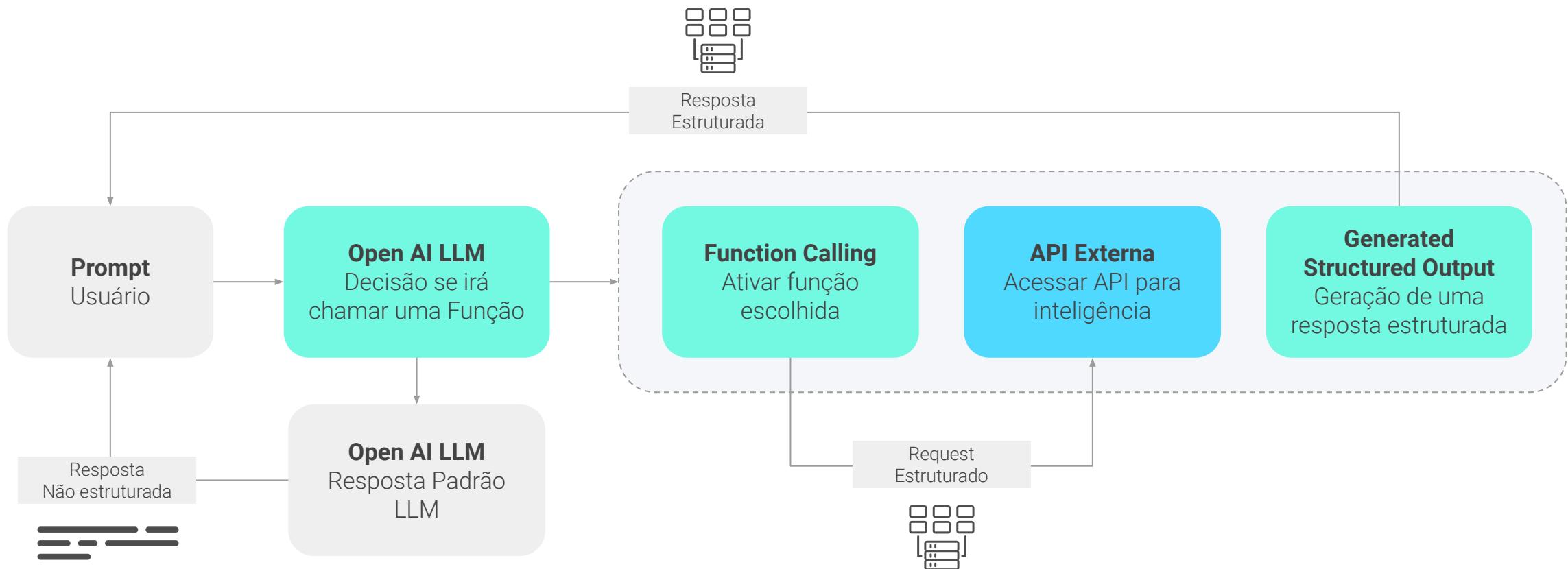
Tradução Literal - Chamar uma função

Possibilidade de através de algum gatilho:

1. Chamarmos alguma função para ser executada
2. Retornar resposta em Formato Estruturado para uso.



Function Calling - Fluxo Completo



Como usar Function Calling?



GPTs Personalizados



Assistants API



Chat Completions API

Interações e Memória	Por meio da interface da Open AI criação GPTs	Por meio de threads (tópicos) da conversa	Por meio de listas de conversas
Function Calling	Sim [Capabilities > Actions] Funções dos GPTs - Ações - Function calling.	Sim [Tools > Function Calling] Tools ferramentas - nome de function calling.	Sim [Chat completions Functions] Por meio da chamada Chat Completions - Functions
Funcionalidades	Upload de arquivos para treinamento dos dados, funções, busca online, etc;	Upload de arquivos para treinamento dos dados, funções, busca online, etc;	Direto ao ponto - Focado no contexto das conversas em questão e em gerar as respostas.
Implementação API	Não é possível.	Mais complexa 5 chamadas APIs	Simples e rápido 1 chamada API
Contexto da conversa	Por meio das threads, com contexto maior. Threads ficam salvas no banco da Open AI.	Por meio das threads, com contexto maior. Threads ficam salvas no banco da Open AI.	Cada conversa é única Histórico da conversa não é salvo.

Depois dessas aulas você aprenderá:



Entendimento completo Function Calling

Entenderá os fundamentos importantes das funções e usos práticos



Conhecimento completo para integrar Functions em seu Projeto

Saberá como criar aplicativos ou SaaS utilizando o poder das Functions Calling

no code start up

Matheus Castelo



@mat_castelo



/in/matheuscastelobranco



Neto Camarano



@netocamarano

