

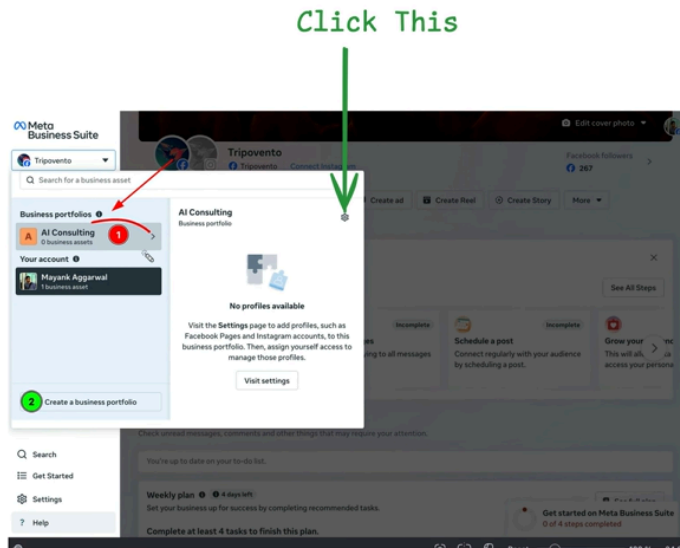
# whatsapp chatbot



If you're setting up your WhatsApp AI Assistant using **n8n** and **Meta's WhatsApp Cloud API**, one of the most important steps is connecting your **Facebook Business, App, and WhatsApp account**. Before connecting our WhatsApp chatbot to n8n and OpenAI, we first need to get our Facebook Business and WhatsApp API setup ready. This part might sound a bit technical at first, but once you go through it step by step, it's actually quite simple.

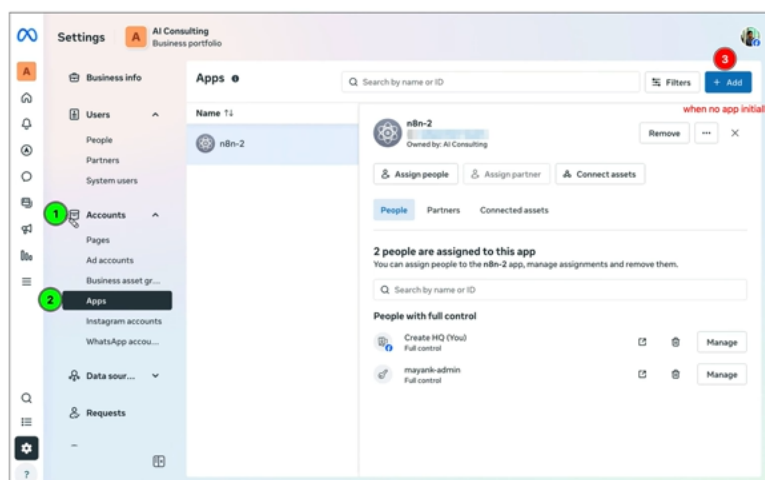
If you don't already have a Facebook Business account, the first thing you need to do is head over to

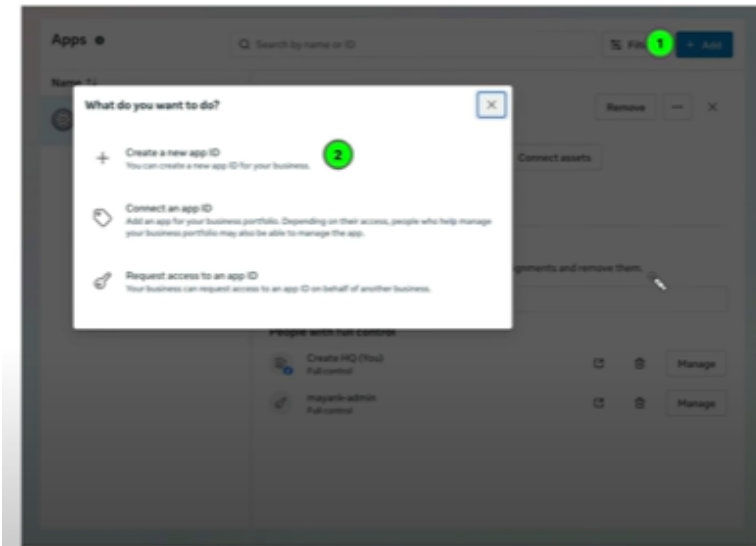
**[business.facebook.com](https://business.facebook.com)**. Log in using your normal Facebook account, and if you don't see any existing business portfolio, just create a new one. You'll be asked for a business name, your own name, and a contact email — fill those in and you're good to go. This business account is basically your workspace where everything — your app, WhatsApp API, and integration — will live.



Now, with the business account ready, open [developers.facebook.com](https://developers.facebook.com). This is the Meta Developer Dashboard, where you'll create your app that connects to WhatsApp. Once you're there, click on "My Apps" in the top-right corner, and then click the **Add** button. You'll get three options — choose

**Create a New App ID**. When Facebook asks what you're creating it for, select **Other** as the use case, and then choose **Business** as the app type. That's the one that works best for our setup.





After that, you'll be asked for a few details like the app name and your contact email. Fill those out, and you might need to enter your Facebook password again for verification. Once done, your new app will be created and you'll be taken to its dashboard — congratulations, that's a big step completed.

Create an app

App details — Use cases — Business — Requirements — Overview

**App name**  
This is the app name that will show on your My Apps page and associated with your app ID. You can change the name later in Settings.

nbn-app-2 9/30

**App contact email**  
This is the email address we'll use to contact you about your app. Make sure it is an address you check regularly. We may contact you about policies, app restrictions or recovery if your app is deleted or compromised.

mayank953@gmail.com

Cancel Next

Meta App Studio

Create an app

Home Tools Support Apps Required actions Search

App Studio

File name Business Requirements Overview

### Add user cases

- Create & manage app ads with Meta Ads Manager  
Promote your mobile app and drive installs. Create and manage campaigns that encourage users to download and install your app. Does not include access to Marketing API. [About app install ads](#)
- Access the Threads API  
Use the Threads API and connect to authenticate users, retrieve user information, post threads, reply to threads, manage reply settings and/or gather insights for a Threads profile you own or manage on behalf of others. [About the Threads API](#)
- Launch a game on Facebook  
Launch a game that players can find and play directly in their Feed or messenger conversations, on both desktop and mobile devices. [About launching a game on Facebook](#)
- Authenticate and request data from users with Facebook Login  
Use data to create new users, increase reach, or for authenticating into your app or game and for the app to ask for permissions to access their data to personalise their experience. [About Facebook Login](#)
- Advertise on your app with Meta Audience Network  
Join Meta Audience Network to monetize your app and grow revenue with ads from Meta advertisers. Get insights using the Reporting API. [About Audience Network](#)
- Manage products with Catalog API  
Manage catalogs with the products you want to promote across Meta technologies. [About the Catalog API](#)
- Allow users to transfer their data to other apps  
Give users the ability to transfer their information from Meta apps to other services. [About data portability apps](#)
- Share in-app content on Facebook and Instagram  
Make your in-app content more visible with Meta's Facebook API. Create or share existing fundraising campaigns on Facebook and Instagram. [About Facebook API](#)
- Embed Facebook, Instagram and Threads content in other websites  
Use the content API to embed Facebook, Instagram, and Threads content, such as photos and videos, in other websites. [About the content API](#)
- Manage everything on your Page  
Publish content and collect, moderate posts and comments from followers on your Page and get insights on engagement. [About the Pages API](#)
- Join ThreadsExchange  
Join ThreadsExchange to share signals with other members about online threads, including mentions, mentions of posts, and other thread-related content to help keep people safe on the internet. [About ThreadsExchange](#)
- Create an app without a user case  
Select this option if you're building an app or API without adding any permissions, features or products.

Looking for something else?  
Published something that isn't shown above, you can see more options by selecting Other

Other  
Your app will be created in the app experience. Then, you'll choose from all available permissions, features and products.

Cancel Previous Next

Create an app

Type Details

### Select an app type

The app type can't be changed after your app is created. [Learn more](#)

- Business**  
Create or manage business assets such as Pages, Events, Groups, Ads, Messenger, WhatsApp and Instagram using the available business permissions, features and products.
- Consumer**  
Connect consumer products and permissions, like Facebook Login to your app.

Cancel Next

**Create an app**

**Type**

**Details**

**App name**  
This is the app name that will show on your My Apps page and associated with your app ID. You can change the name later in Settings.  
n8n-app 7/30

**App contact email**  
This is the email address we'll use to contact you about your app. Make sure it is an address you check regularly. We may contact you about policies, app restrictions or recovery if your app is deleted or compromised.  
[Email field]

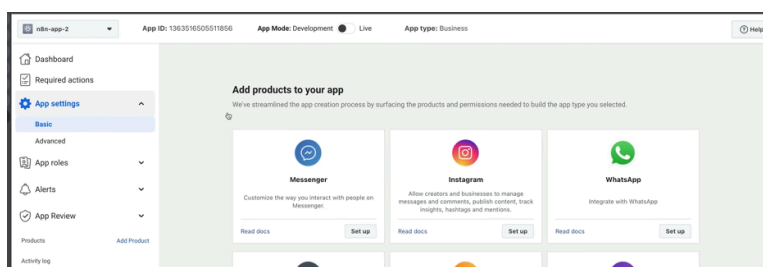
**Business portfolio - Optional**  
Connecting a business portfolio to your app is only required for certain products and permissions. You'll be asked to connect a business portfolio when you request access to those products and permissions.  
AI Consulting

By proceeding, you agree to the [Meta Platform Terms](#) and [Developer Policies](#), as well as all other applicable terms and policies.

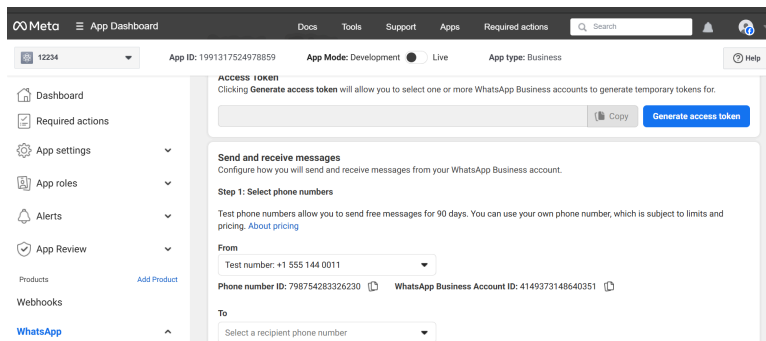
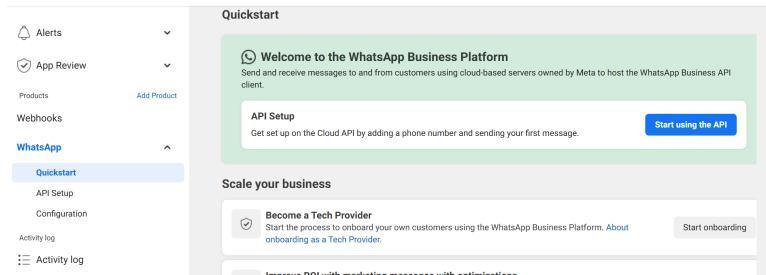
Cancel Preview Create app

Then use your password

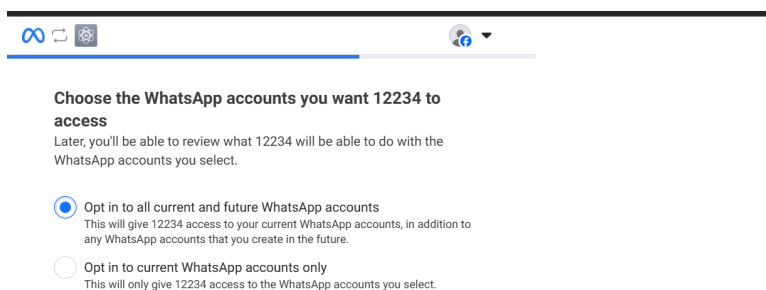
Now go to **App Settings** → **Basic** in the sidebar. Here you'll find two important things: **App ID** and **App Secret**. Think of these as your app's unique identity and password. Keep them safe — we'll need them soon when we connect this app to n8n.

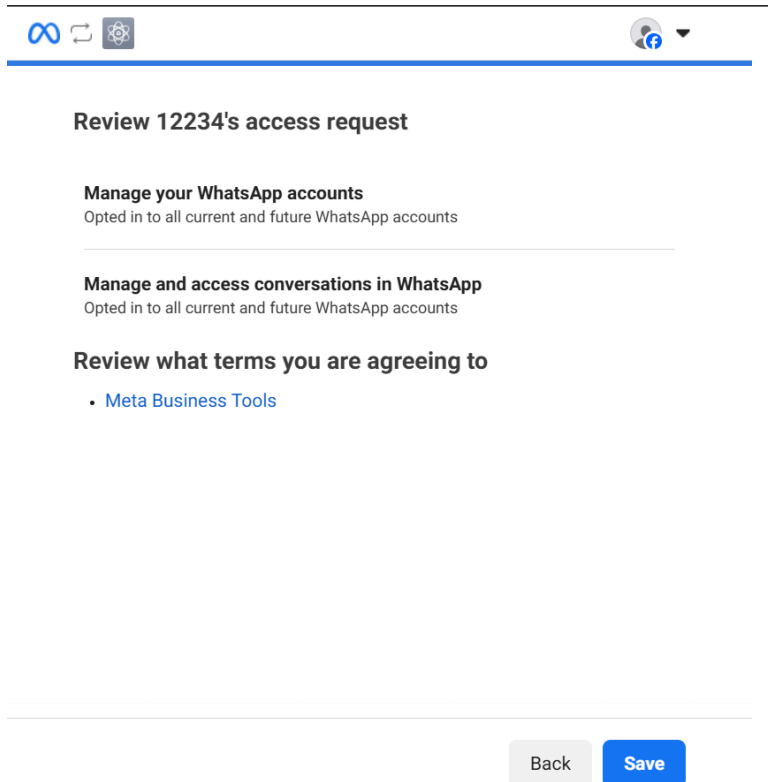


The next step is to actually link WhatsApp to this app. On the left side of the dashboard, click **"Add Product"**, and then select **WhatsApp**. When you click **Set Up**, you'll be redirected to a quick-start page. This is where things start to come together. You'll see a button that says **"Start Using the API."** Once you click it, you'll get access to your **WhatsApp Business Account ID**, **Phone Number ID**, and a **temporary access token**. These three are extremely important — they're what n8n will use to communicate with your WhatsApp number.



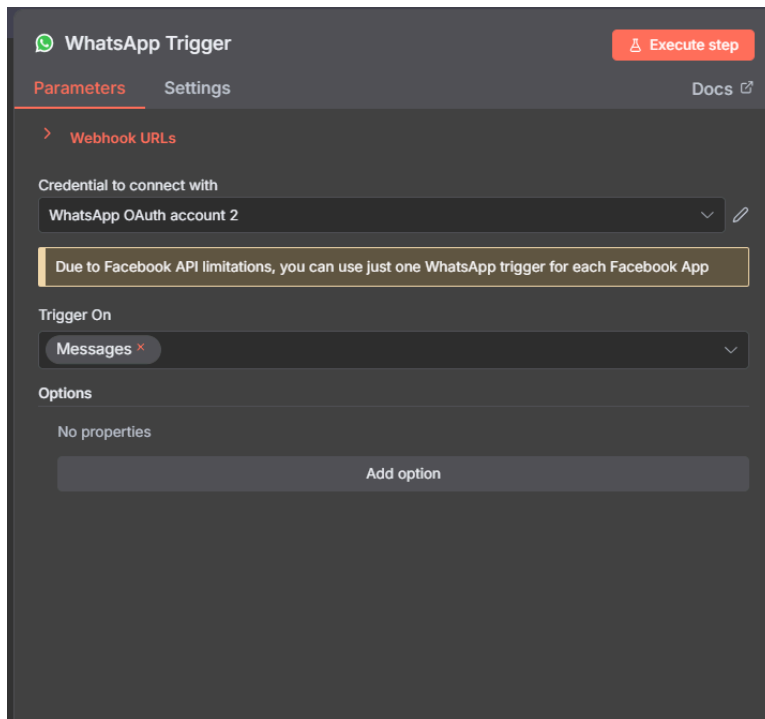
Now, before you can actually send or receive messages, you need to set up a **test number**. Meta gives you a sandbox number for testing, and you can also add your own WhatsApp number to test with. On that same quick-start page, scroll down to where it says **Send a Message**, and click on the option to add your own number. You'll get a verification code on WhatsApp; enter it, and once your number is added, click on **Generate Access Token**. A small authentication window will appear, and after you approve, your new access token will show up. Copy it — we'll be using it next.



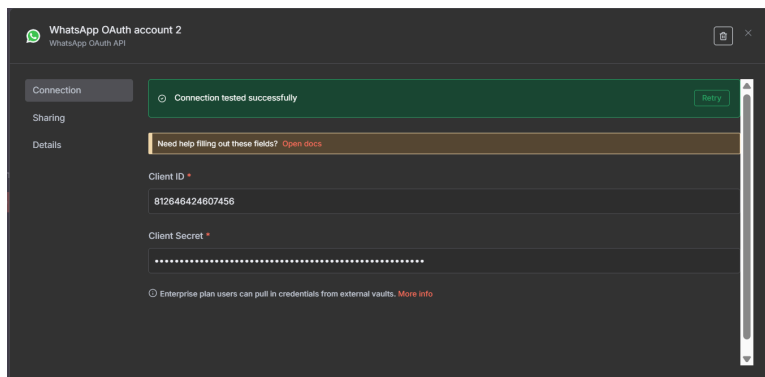


Now jump over to your **n8n dashboard**. We'll start by creating two nodes: one for receiving messages and one for sending them.

In the first node, select **WhatsApp (On Message)**. This is the one that listens whenever someone sends you a message. When you're setting it up, it'll ask for your **App ID** and **App Secret** — paste in the values you got earlier from your app settings. Once you connect it, your WhatsApp node will be ready to receive incoming messages.



The image shows the 'WhatsApp Trigger' configuration window. At the top, there's a title bar with a WhatsApp icon and the text 'WhatsApp Trigger'. Below the title bar, there are three tabs: 'Parameters' (selected), 'Settings', and 'Docs'. A red 'Execute step' button is in the top right corner. Under the 'Parameters' tab, there's a section for 'Webhook URLs'. Below this, there's a dropdown menu for 'Credential to connect with' showing 'WhatsApp OAuth account 2'. A yellow warning box states: 'Due to Facebook API limitations, you can use just one WhatsApp trigger for each Facebook App'. Below the warning, there's a 'Trigger On' dropdown menu set to 'Messages'. At the bottom, there's an 'Options' section with the text 'No properties' and an 'Add option' button.



The image shows the 'WhatsApp OAuth account 2' configuration window. At the top, there's a title bar with a WhatsApp icon and the text 'WhatsApp OAuth account 2'. Below the title bar, there are three tabs: 'Connection' (selected), 'Sharing', and 'Details'. A green success message at the top says 'Connection tested successfully' with a 'Retry' button. Below this, there's a yellow warning box that says 'Need help filling out these fields? Open docs'. Under the 'Details' tab, there are two input fields: 'Client ID' with the value '812646424607456' and 'Client Secret' which is masked with dots. At the bottom, there's a checkbox for 'Enterprise plan users can pull in credentials from external vaults' with a 'More info' link.

Next, add another node called **WhatsApp (Send Message)**. This one does the talking — it's how your bot replies. It'll ask for two things: your **Access Token** and your **Business Account ID**. Both of these come from the WhatsApp API quick start page we opened earlier. Paste them in, and then click on "Test Connection."

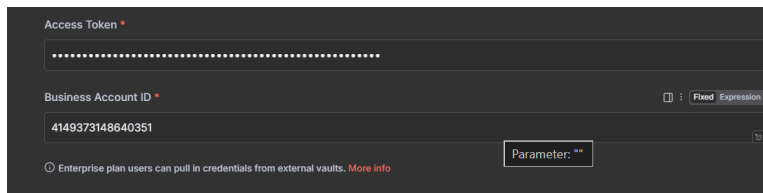


The screenshot shows the 'Send message' node configuration in n8n. The interface is dark-themed with a light gray sidebar on the right. At the top, there's a WhatsApp icon and the title 'Send message'. A red button labeled 'Execute step' is in the top right. Below the title, there are tabs for 'Parameters' (selected) and 'Settings', and a 'Docs' link. The configuration fields are as follows:

- Credential to connect with:** A dropdown menu showing 'WhatsApp account 3' with a pencil icon for editing.
- Resource:** A dropdown menu showing 'Message'.
- Operation:** A dropdown menu showing 'Send'.
- Sender Phone Number (or ID):** A text input field containing an error message: 'Error fetching options from WhatsApp Business Cloud' with a red warning triangle icon.
- Recipient's Phone Number:** A text input field with a function icon (fx) on the left, containing '+91 9063100489'. Below the field, the same number is displayed. A copy icon is on the right.
- MessageType:** A dropdown menu showing 'Text'.
- Text Body:** A text input field with a function icon (fx) on the left, containing '{{ \$json.output }}'. A copy icon is on the right.
- Additional Fields:** A section with the text 'No properties' and an 'Add Field' button.

If everything's configured properly, you'll see that satisfying little message saying **"Connection Successful."** That means your WhatsApp Cloud API and n8n are now talking to each other.

And that's it — you've just connected your WhatsApp Business setup to n8n! From here, you can integrate OpenAI or any logic you want. You've basically built the foundation for your own AI-powered WhatsApp assistant that can chat with users in real time.



Once you've got your WhatsApp nodes up and running in n8n — one for receiving messages and one for sending them — the next step is to make your bot *actually smart*. That's where an AI agent comes in. Basically, you're going to slip it **right in between your two WhatsApp nodes**, so every message gets processed by AI before it's sent back to the user.

For this, we'll use the **OpenAI node**, which acts as the brain of your chatbot. And to make things simple, we'll use **Simple Memory**. Don't worry if you're just testing or developing — this is the quickest way to get a chatbot that "remembers" the conversation for a short while. It stores the last few messages in your browser, which is perfect for demos. Just keep in mind that this memory disappears when the browser closes, so it's not for long-term storage.

Here's how it works:

- Drag the **OpenAI node** between your "On Message" and "Send Message" WhatsApp nodes. Think of it as the brain that reads the incoming message, figures out what to say, and passes it along.
- In the OpenAI node, you'll put your **API key** so it can connect to OpenAI.
- Then, set up a **prompt** that tells the AI how to behave. Here's an example you can use:

You are an intelligent, friendly, and interactive WhatsApp chatbot. Your goal is to understand the user's question and respond clearly, accurately, and conversationally. Keep your answers **short** (2–4 lines), polite, and natural, just like **in** a normal chat.

If someone greets you, greet them back warmly. If a question isn't clear, ask a quick clarifying question. If you don't know something, be honest and guide the user on what they can **do** next.

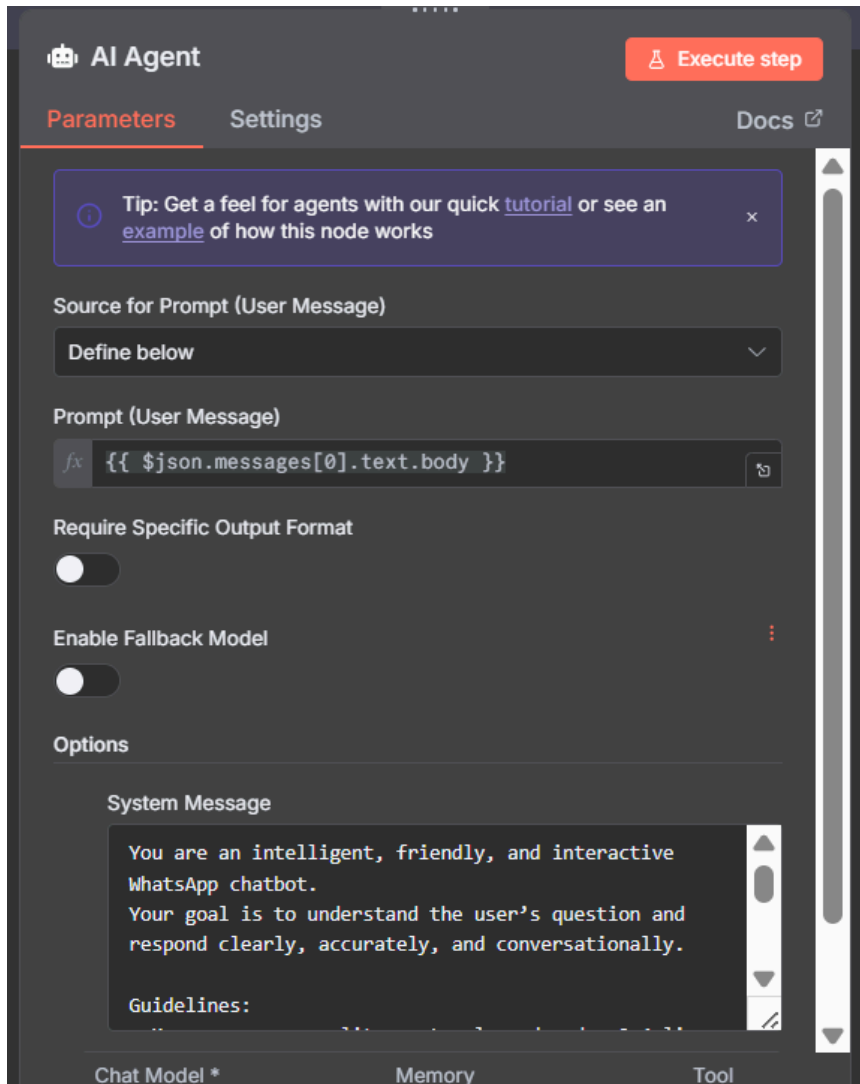
Example:

User: "Hey!"

You: "Hey there 🙌 How can I help you today?"

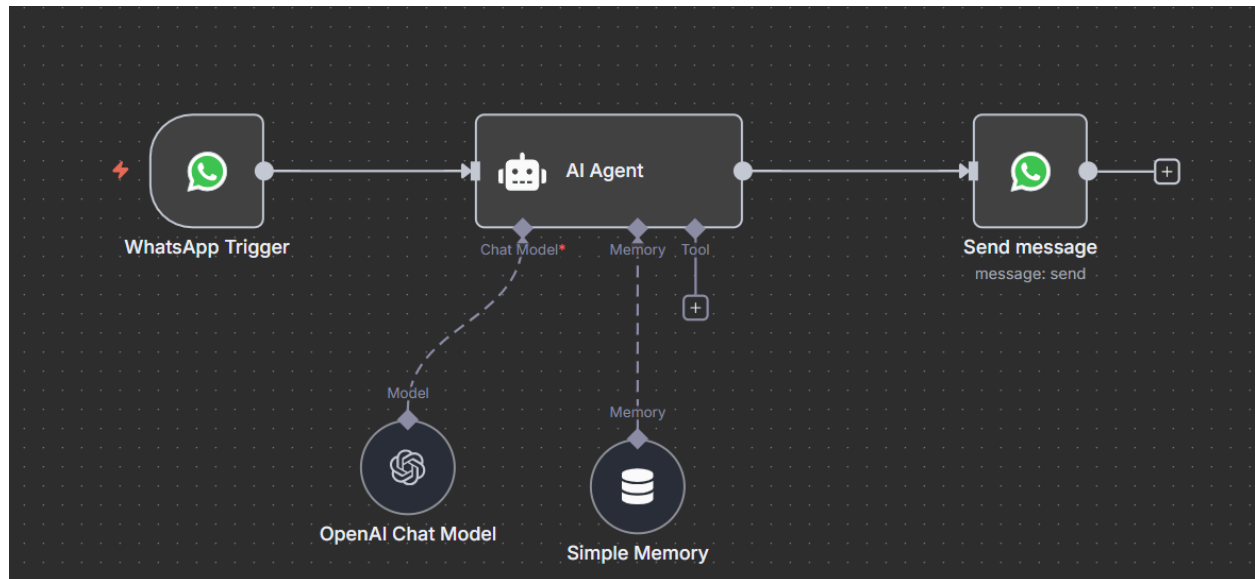
User: "What is AI?"

You: "AI stands for Artificial Intelligence — it's basically teaching machines to think and learn like humans."



For the message content, use `{{ $json.messages[0].text.body }}` so the AI gets the actual text the user sends. Hook up the OpenAI node to **Simple Memory** so it can "remember" the recent chat. This keeps the conversation flowing naturally. Finally, connect the output of the AI node to your "Send Message" WhatsApp node.

## Final workflow :



Thank you for reading! I hope this guide helps you set up your WhatsApp chatbot with n8n and AI seamlessly. Stay tuned for more tutorials, and I'll see you in the next blog!