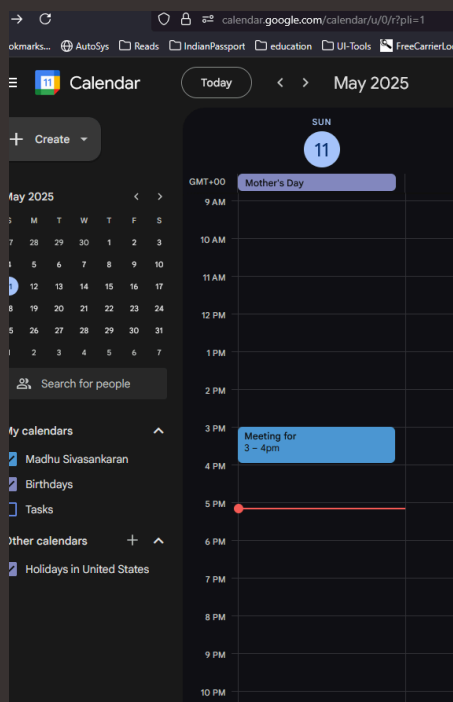




[nextwork.org](https://nextwork.org)

# Building an AI Workflow

1K



# Introducing Today's Project!

This project implements an intelligent workflow for sending SMS-based meeting invitations using n8n as the backend automation platform. Leveraging AI, the system dynamically interprets user inputs (e.g., messages, emails, or form submissions) to extract meeting details such as date, time, participants, and agenda. Once parsed, the workflow triggers SMS notifications to the specified recipients via an integrated SMS gateway (e.g., Twilio). The appeal for this workflow is Real-Time Notifications: Sends timely and personalized meeting invites directly to participants' phones via SMS.

## Tools and Techniques

n8n (Workflow Automation Platform) Used to build, automate, and orchestrate the entire workflow. Handles triggers, data processing, and service integration. OpenAI / Language Model API Powers the AI assistant to extract meeting details from unstructured text using natural language processing. Custom prompts guide the tone and structure of AI responses. SMS Gateway (e.g., Twilio or Vonage) Sends meeting invites via SMS to participants in real time. Integrated through n8n to automate the message delivery. Calendar Service (e.g., Google Calendar API) Optionally used to check or add events based on parsed meeting details. Ensures scheduling accuracy and availability. Webhook or Form Input Service Used to capture initial user input that contains the meeting request (e.g., via a web form, chatbot, or external app).

## Project reflection

About a day as I had family stuff to juggling family stuff on a mothers day

## placeholder

This project offers an opportunity to leverage automation and AI to streamline everyday tasks and improve efficiency.

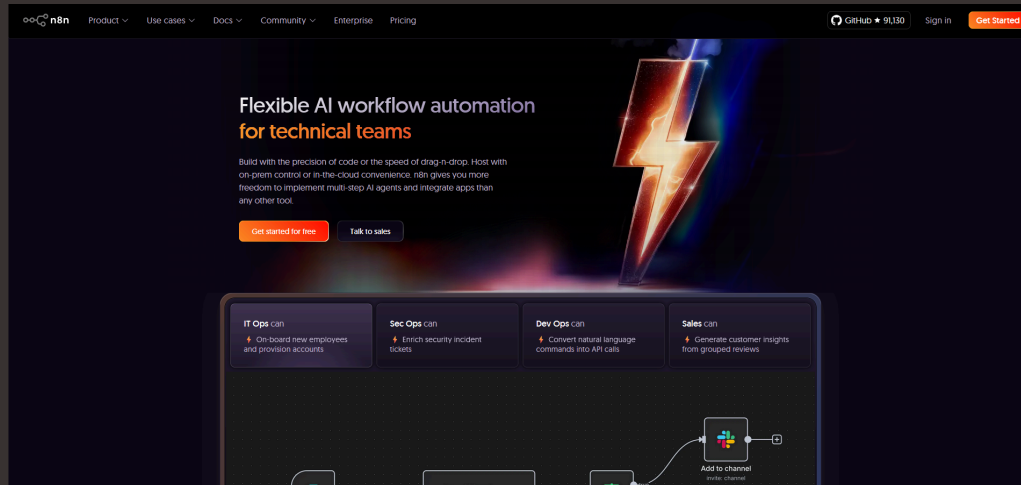
By integrating AI into the workflow for sending meeting invites via SMS, you're not only saving time but also enhancing user experience through seamless, personalized communication. The use of tools like n8n for workflow automation allows you to create flexible, scalable solutions that integrate with multiple services, while AI models help ensure responses are tailored and relevant. Ultimately, this project enhances productivity and communication by automating repetitive tasks and making processes smarter and more efficient.

# Exploring n8n

n8n is an attractive platform for automating workflows, especially because it provides a combination of powerful integrations, ease of use, customization, and privacy.

Whether you're automating simple tasks or creating complex AI-driven systems, n8n offers the tools and flexibility needed to build efficient, scalable automations.

N8N is a powerful, open-source workflow automation tool that allows users to connect apps and services through customizable, no-code or low-code workflows. It includes a rich library of over 350 built-in integrations (nodes) for services like Slack, Gmail, Twilio, and more, enabling automation of complex tasks without writing backend code. In the n8n free trial (cloud version), you typically get: Access to all core features and workflow capabilities A generous number of workflow executions (e.g., ~200 executions/day) Hosted infrastructure with no setup required Access to the workflow editor, templates, and community support Option to test premium features like versioning, execution logs, and role-based access controls (depending on plan)



# Starting an AI Workflow

To set up a workflow, I first configured the following triggers

- Webhook Trigger – activates when data is sent to a unique URL (e.g., from a web-form or another app)
- Schedule Trigger – runs the workflow at specific times or intervals (e.g., every hour, daily) in cases where the users have per-configured workflows that needs to sent out on specific dates and times.
- App-Specific Triggers – such as "High priority Trigger" (on new email to specific email ID) or "Stripe Trigger" (on new payment)
- Manual Trigger – allows you to test or run a workflow manually during development

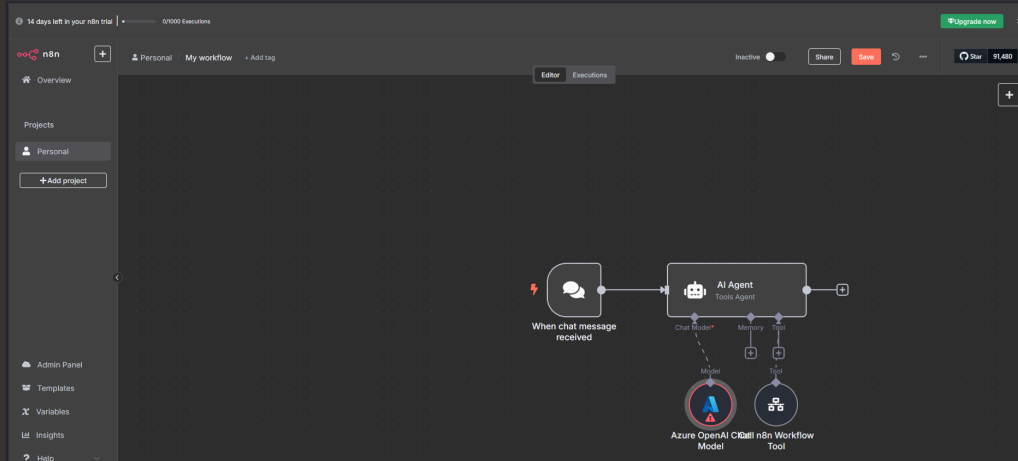
I connected my trigger with an AI agent node, in this project, I am building and agent that takes the incoming message from users and analyze the content to understand what the user is requesting to do and take the necessary steps to complete the request.

- A, The user could be requesting to setup a new meeting
- B, The user could be rescheduling a meeting
- C, The user could be sending a reminder as SMS to clients

1K

NextWork Student

[nextwork.org](https://nextwork.org)

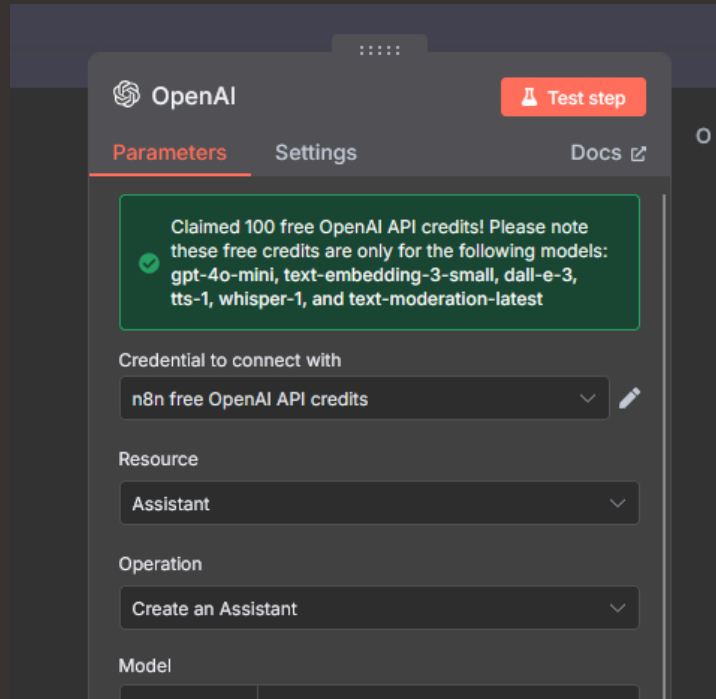


# Integrating ChatGPT

In an AI workflow, especially one built in n8n, following is the flow Trigger: User submits a meeting request through a web form. Input Data: The form data (date, time, participants) is captured. AI Model: The AI extracts relevant information from the form and structures the data for scheduling. Logic/Processing: The workflow checks for availability in the user's calendar. External Integration: The AI sends an SMS invite through Twilio with the meeting details. Output: User receives the SMS confirmation. Error Handling: If there's an error (e.g., failed SMS), an alert is sent to the administrator.

My workflows chat model uses Azure Open AI chat to securely handle user data and the client contact information, so as to comply with GDPR and CCPA. I could connect with OpenAI free model for testing with anonymized data, for development and test cases.

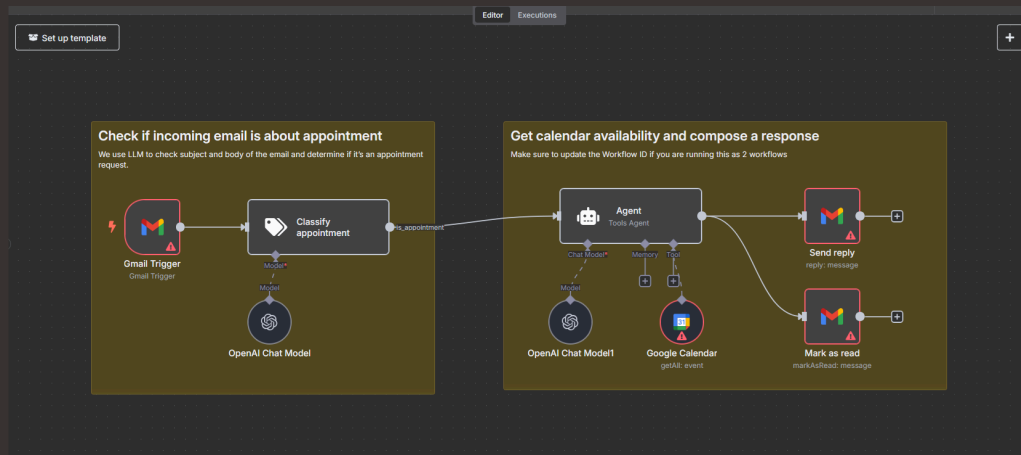




# Integrating Google Calendar

In an AI workflow, especially one built in n8n, a tool refers to any functional component or service that the AI can call or interact with to perform a specific task. Tools are often nodes or integrations that extend the AI's capabilities beyond text generation.

I am using the OAuth connector to give access to my google Calendar, I am giving the following access Read-Only Access Permissions: The system can view events and details in the calendar, but cannot add, modify, or delete events. Use Case: This is useful when the AI needs to check availability or read calendar events without making any changes. Read and Write Access Permissions: The system can view, create, update, and delete events on the calendar. Use Case: This is often used when the AI needs to add new meetings, update existing events, or cancel meetings directly from the workflow.



# Testing My Workflow

The workflow had issues with connecting to and creating a event because I went with least privilege method and only provided Read-Only Access, this did not work as expected and I had to provide Read and Write Access to create a event in my Calendar

I learned how to identify and fix issue with the following criteria in mind

- Key Insights from AI Agent Logs: Input and Output Data See exactly what prompt (including system message and user input) was sent to the AI. Review the AI's response to check if it matched expectations.
- Prompt Effectiveness Evaluate whether the system message or user prompt led to the correct tone, structure, and extracted details. Helps you fine-tune prompts for clarity, tone, or formatting.
- Model Behavior Over Time Spot inconsistencies in the AI's output across multiple runs (e.g., formatting drift, missing fields). Useful for prompt engineering and testing reliability.
- Error Messages or Failures If the AI node failed, logs may show: Rate limits or quota errors API authentication issues Malformed or missing input fields Latency or Performance Check how long the AI node took to respond—can help in identifying bottlenecks in execution.



Appointment request for May-11-2025 3 PM EST Inbox x

M


Madhu

Appointment request for May-11-2025 3 PM EST

12:39 PM (1 minute ago)

☆

M



12:39 PM (0 minutes ago)

☆

🔒

↩

⋮

It seems that I'm currently unable to access your calendar to check your availability for the requested appointment. However, since today is May 11, 2025, and the requested time is 3 PM, I can assume you may have other scheduled appointments around this time. To respond to Madhu, I suggest proposing a new time for the appointment. How about suggesting May 11, 2025, at 4 PM EST instead? Here's a draft response to Madhu: — Subject: Re: Appointment request for May-11-2025 3 PM EST Dear Madhu, Thank you for your email regarding an appointment on May 11, 2025, at 3 PM EST. Unfortunately, I have a prior commitment at that time. Could we possibly meet at 4 PM EST instead? Please let me know if that works for you. Looking forward to your reply. Best, [Your Name] — Feel free to modify it as needed!

↩ Reply

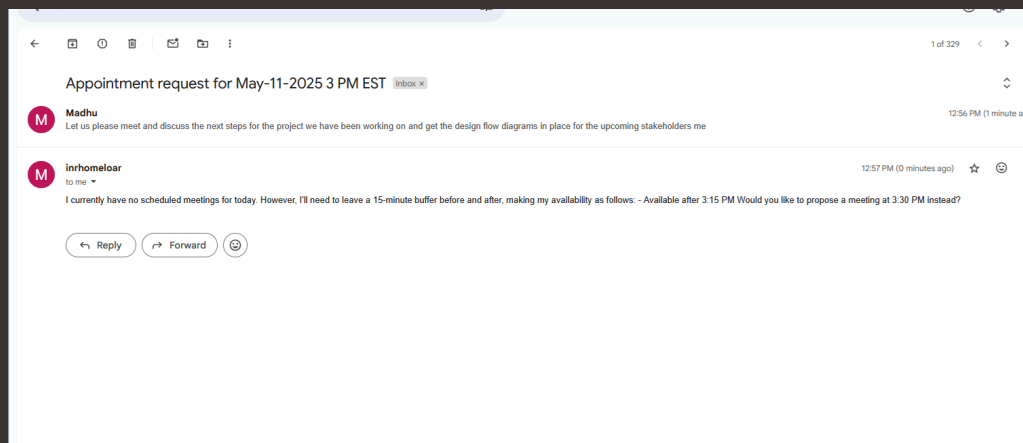
➡ Forward

🔒

# JSON Expressions

In the trouble shooting process we identified that even though the user had given access to their google account they had to explicitly provide access to their calendar, this access restriction caused the agent error out instead of setting up the meeting in the clients calendar. Our checklist mentioned in the previous step came in handy in the debugging process.

I had to change the authentication mechanism as google has updated their policy and simply having access to once email does not automatically also provide access to your google calendar



# System Messages

On my second test, my workflow successfully... but it still made an error as the screen shot show the clients name is showing up in the template as [Your Name] this will be reputationally damaging to the brand image as such I started looking into why the users Name was not getting injected into the template

In n8n, a system message refers to the initial instruction or guiding prompt given to an AI model (like OpenAI's ChatGPT) to define its behavior or tone throughout a conversation or task. It's commonly used in AI-related nodes to set the assistant's personality, style, and boundaries before user input is processed. For example, in an OpenAI node in n8n, you might include a system message like: "You are a helpful assistant who responds in a casual, friendly tone. Use simple, natural language." This helps ensure that the AI consistently responds in the desired way, regardless of the user's prompt. In short, the system message acts like the AI's role instructions, shaping how it interprets and responds to user inputs in your workflow.



NextWork Student

[nextwork.org](https://nextwork.org)

1 of 329

Appointment request for May-11-2025 3 PM EST

M

Madhu

Let us please meet and discuss the next steps for the project we have been working on and get the design flow diagrams in place for the upcoming stakeholders me

12:56 PM (3 minutes ago)

☆

M

inhomeloan

I currently have no scheduled meetings for today. However, I'll need to leave a 15-minute buffer before and after, making my availability as follows: - Available

12:57 PM (3 minutes ago)

☆

M

Madhu

Yes, 3:30 PM works for me please send me a meeting invite.

12:59 PM (0 minutes ago)

☆

M

inhomeloan

to me

It looks like I have no meetings scheduled for today, so I can confirm a meeting with Madhu at 3:30 PM. I'll send out the invite now. Here's the response I'll send — Subject: Meeting Confirmation for May 11, 2025 at 3:30 PM EST Hi Madhu, Thank you for confirming. I have scheduled our meeting for May 11, 2025, at 3:30 PM EST. I will send you the invite shortly. Looking forward to our conversation. Best regards, [Your Name] — Let me know if there's anything else you'd like to add!

12:59 PM (0 minutes ago)

☆

📎

🔍

⋮

↩ Reply

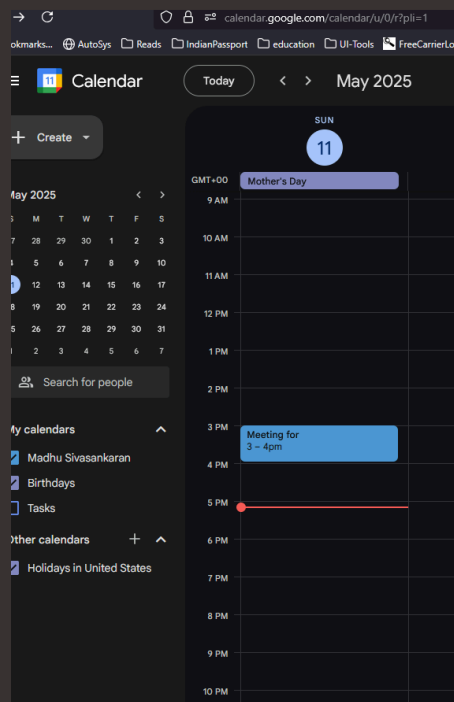
➡ Forward

📎



# Success!

Open Your Workflow in n8n Go to your workflow that uses an AI service (like OpenAI, OpenRouter, etc.). Locate the AI Node Click on the node where the AI model is being called (e.g., the OpenAI, HTTP Request, or custom GPT node). Edit the System Prompt / Message Inside the node, look for a field labeled something like: System Message System Prompt Or within the Messages array (if using Chat format) Update this message to better guide the AI. Example fix: Old (too vague): "Be helpful." Improved: "You are a clear, friendly assistant. Respond with simple, casual language and provide only the requested meeting details." Test the Workflow Use a Manual Trigger or test input to re-run the workflow and check if the AI's output now aligns with expectations.



# System Message Variation

To make the chat model more casual in n8n, you can adjust the prompt settings in the AI-related node (e.g., OpenAI, OpenRouter, or other language model nodes). Simply modify the system or user prompt to include guidance like: "Respond in a friendly, conversational tone" or "Use casual, informal language when replying." Additionally, you can fine-tune the temperature parameter (e.g., set it between 0.7–0.9) to encourage more creative and relaxed responses. These small changes help the AI adopt a more natural, chatty style while still staying on-topic. This is the Prompt I used "You are a helpful assistant who responds in a casual, friendly, and conversational tone. Keep responses clear, slightly informal, and easy to understand. Feel free to use contractions and natural phrasing."

Input (Meeting Info): "Project kickoff meeting on May 15th at 2:00 PM with the design and development teams. Agenda: Introductions, project goals, timeline overview, and Q&A." AI Response: Hey! ☺ Just a heads-up—we've got our Project Kickoff Meeting coming up on May 15th at 3:30 PM. It'll be with the design and dev teams. We'll go over intros, chat about project goals, walk through the timeline, and leave some time for Q&A at the end. Shouldn't take too long—looking forward to getting things rolling together!

When we added constraints with system message it intercepted the incoming request and managed to identify if the email was from a friend at work and changed the response tonality to use a more casual method of response.

