Create a Content Specialist AI Agent in **Make.com**

Follow the steps below to build an AI agent that finds the latest AI news, turns it into an Instagram reel script + LinkedIn post, generates an image, and emails everything to you each day.

Overview

This guide walks you through creating a Make.com scenario (workflow) and an AI Agent that:

- Searches the web for the most important AI news (recent launch/announcement)
- Parses the result (title, summary, URL) as structured JSON
- Uses a large language model (e.g., Gemini) to generate:
    - A 150-word Instagram reel script
    - A LinkedIn post (caption) in a clean format
- Generates a single representative word for an image and uses an image model to create the image
- Emails the HTML output (script + post) with the image attached
- Runs automatically on a schedule (e.g., daily at 11:00 AM)

Prerequisites

1. **Make.com account** (free plan is fine to get started)
2. **An LLM-based news/search provider** account (Perplexity or Open AI Assistant a similar web-aware LLM) and its API key
3. **Generative LLM account** (for text + images) — the video uses Gemini (AI Studio) or any other model with text/image APIs
4. **Gmail account** (or other email service connected to Make) to send emails

Quick architecture (high level)

Trigger (Webhook) → News search module (Perplexity) → JSON parser → Gemini text completion (script + LinkedIn) → Gemini single-word extractor → Gemini Image generation → Gmail (send email with HTML + image attachment)

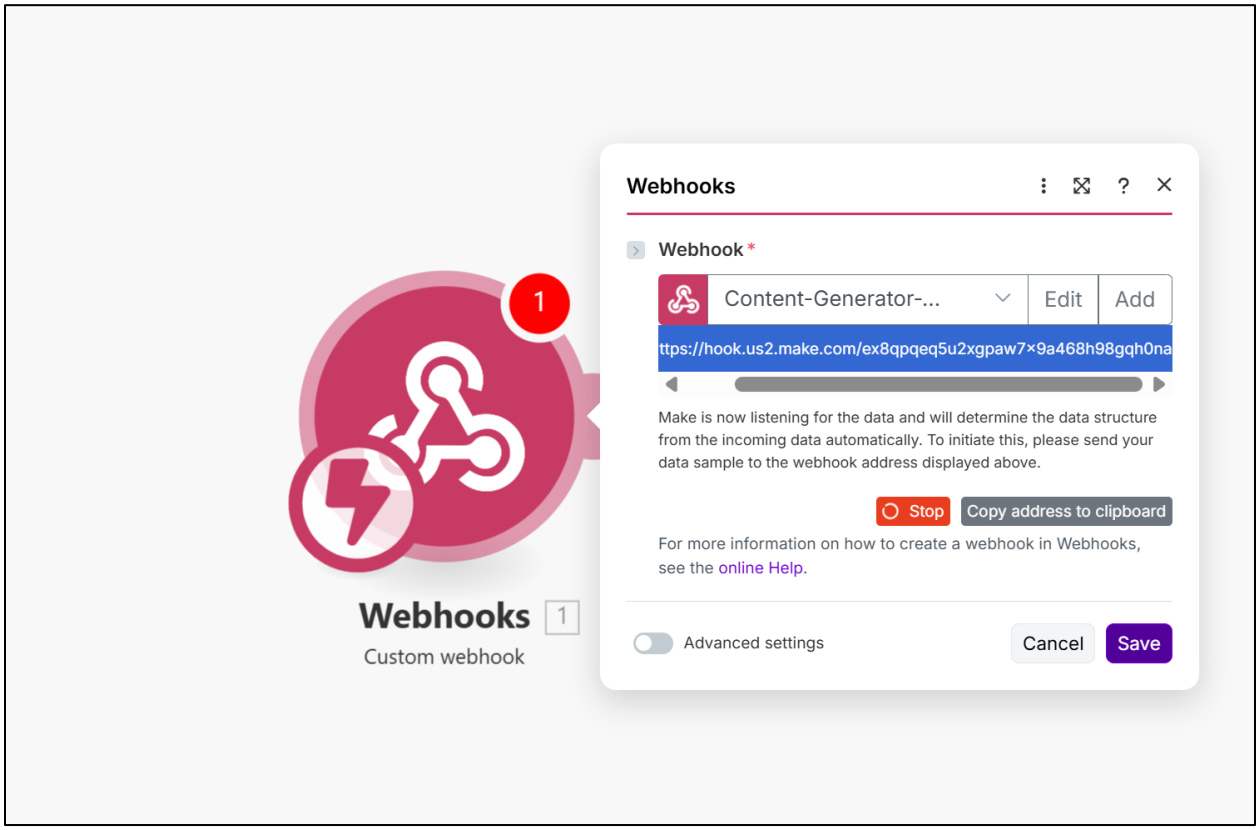
Detailed Steps

Each numbered step corresponds to a module or action in Make.com. Read the whole step first, then follow it in Make.com.

1. Create the scenario & trigger

1. Log into **Make.com** → go to **Scenarios** → **Create a new scenario**.
2. For the trigger, add the **Webhooks** module → choose **Custom webhook** → **Add a webhook**.
3. Name it (example: content-generator-webhook) and **Save**. Copy the webhook URL — you'll use this to run/test the scenario.

Tip: During testing, open the webhook URL in your browser to trigger a run (or use Make's test/run button).



2. Add a news-search module (Perplexity or similar)

1. Click the + to add a module after the webhook.
2. Search for your news/search LLM (the video used Perplexity). Choose a module like **Create a Chat Completion** (or equivalent) for that service.
3. **Connect** the module using the API key (obtain from the provider's dashboard).
4. Choose the news-search model (video used sonar / sonar-pro as an example). Use whatever the provider recommends for web search.
5. **Prompt** (example) — set role = user and paste this prompt (modify timeframe as needed):

Find and analyze the most important AI tool or product that was launched in the last 4-5 days.

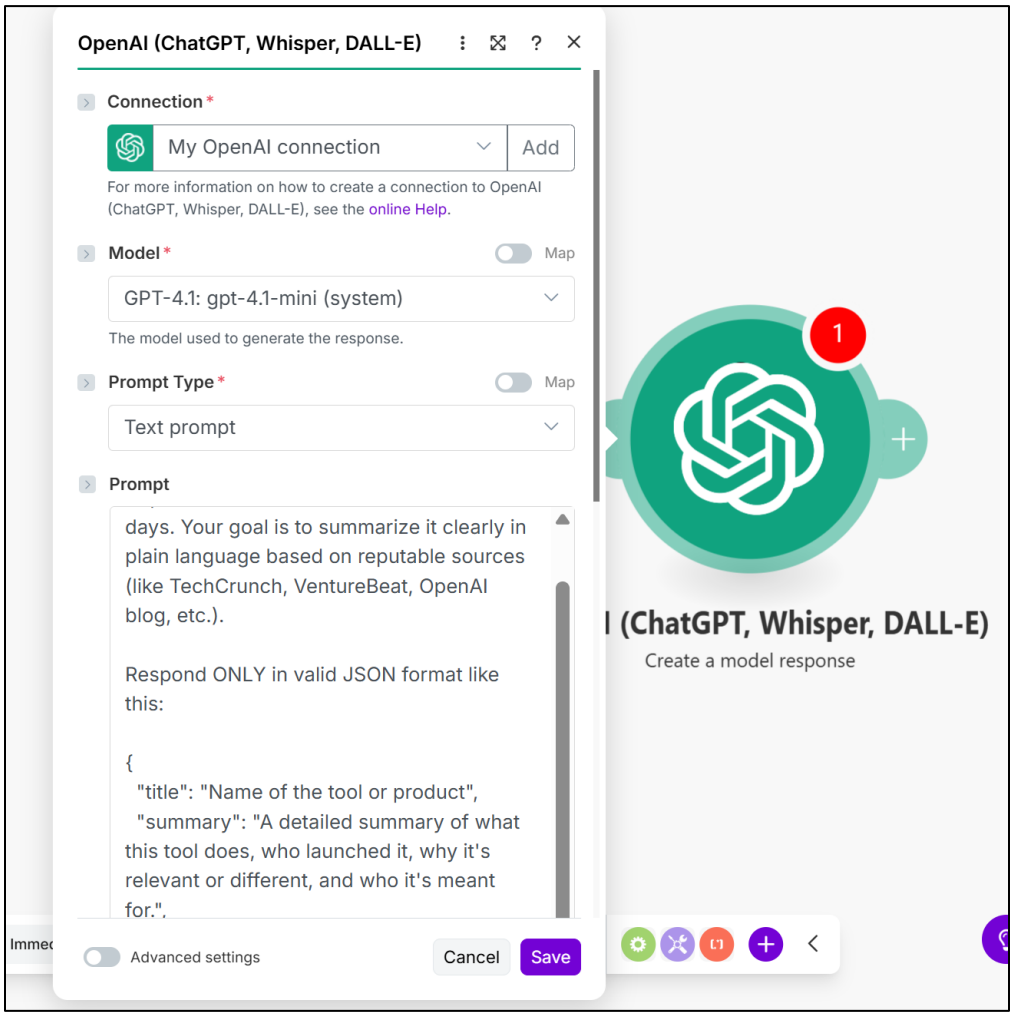
Your goal is to summarize it in plain language based on reputable sources.

Respond only in valid JSON with these fields: title, summary, url.

Example:

```
{
  "title": "...",
  "summary": "...",
  "url": "https://..."
}
```

6. Save the module and run a test to ensure it returns a JSON-like string or a search payload containing the best source.



3. Wrap results as JSON and parse

1. If the news module returns a free-text output, add **Tools → Compose a string** (or similar) to wrap the output inside braces or build a JSON string.
2. Add a **JSON** (Parse) module to convert the JSON string into separate fields: title, summary, url.
3. Configure the JSON parser to create three outputs (all text) so downstream modules can reference them as variables.

Sample parsed JSON structure

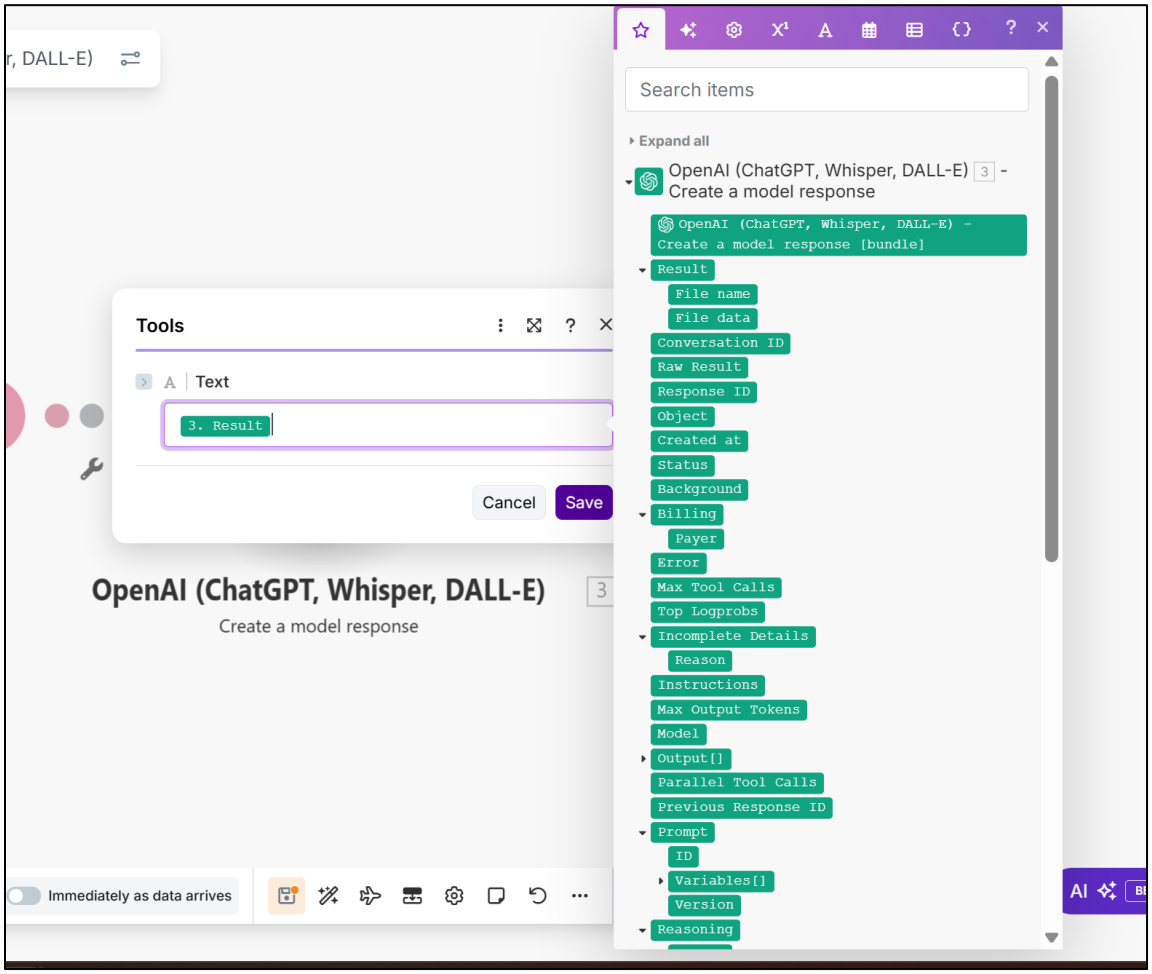
{

"title": "New AI Summarizer 3.0",

"summary": "A plain-language summary of what the tool does and why it matters.",

"url": "https://example.com/article"

}



4. Add the LLM module to create your reel script + LinkedIn post (Gemini or other)

1. Add a **Create a Chat Completion** module (Gemini / your text model).
2. Connect with the model API key and pick a fast/high-quality model (the video used Gemini 2.5 flash).
3. In the module, set up a message with the **role: user** and provide a long prompt that:
    - Tells the model to act as a top-tier content strategist
    - Supplies the {{title}}, {{summary}}, and {{url}} variables (use Make's variable picker)
    - Requests output in HTML for email
    - Requests the Instagram reel script (150 words) and the LinkedIn post (format rules)

Example prompt (paste into the module)

Act like a top-tier content strategist. Below is today's top AI news.

Return the result in clean HTML using this structure:

<h2>{{title}}</h2>

<h3>Instagram reel</h3>

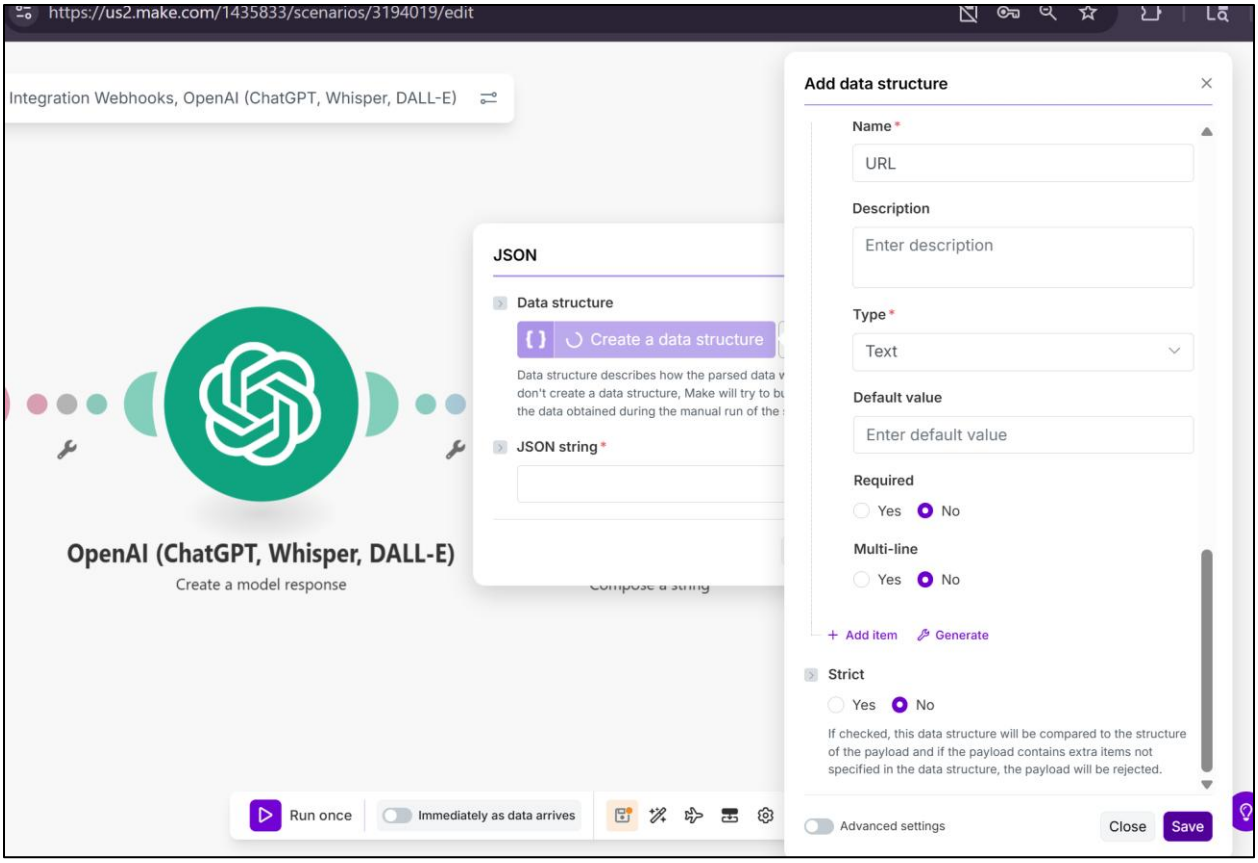
<!-- reel script here -->

<h3>LinkedIn post</h3>

<!-- linkedIn caption here -->

Respond only with the HTML.

4. Save and run the module to verify that the model returns the requested HTML output.

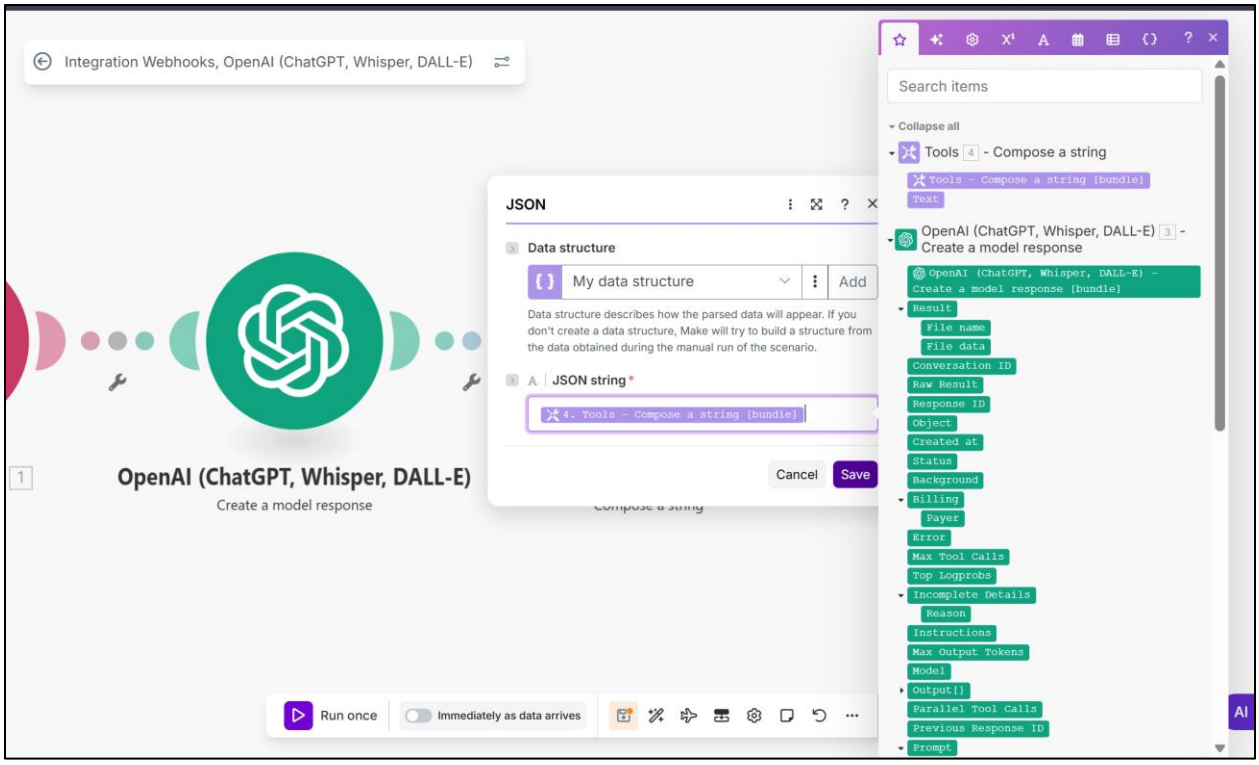


5. Extract a single keyword for the image

1. Add another text completion module that **returns just one word** (the central topic/keyword to put on the image).
2. Prompt example:

Based on this news summary: {{summary}} return the single most important word that best represents the topic. Return only one word, lowercase.

3. Save and test.

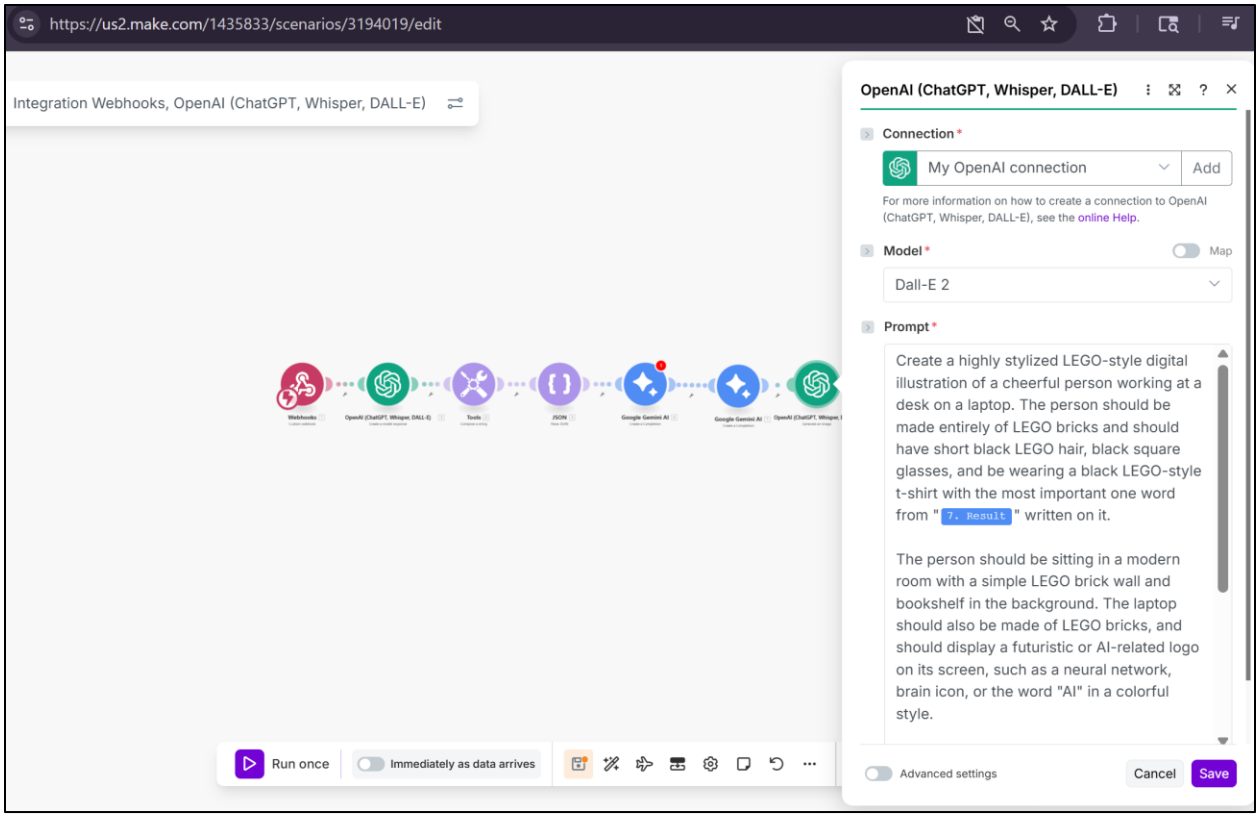


6. Create the image (Image generation module)

1. Add the provider's **Create Image** module (the video used Gemini Image Gen 3.0).
2. Configure the size/aspect ratio (the video used 1:1) and any content allowance flags.
3. Build an image prompt that uses the single word and a style. Example prompt for a LEGO style image:

A high-quality 1:1 LEGO-style scene: a person working on a laptop with the word "{{keyword}}" bolded on the t-shirt and on the laptop screen; bright, modern, professional composition; shallow depth of field; pleasant studio lighting.

4. Save and test the image generation. Set the module to output the image file for use as an attachment.



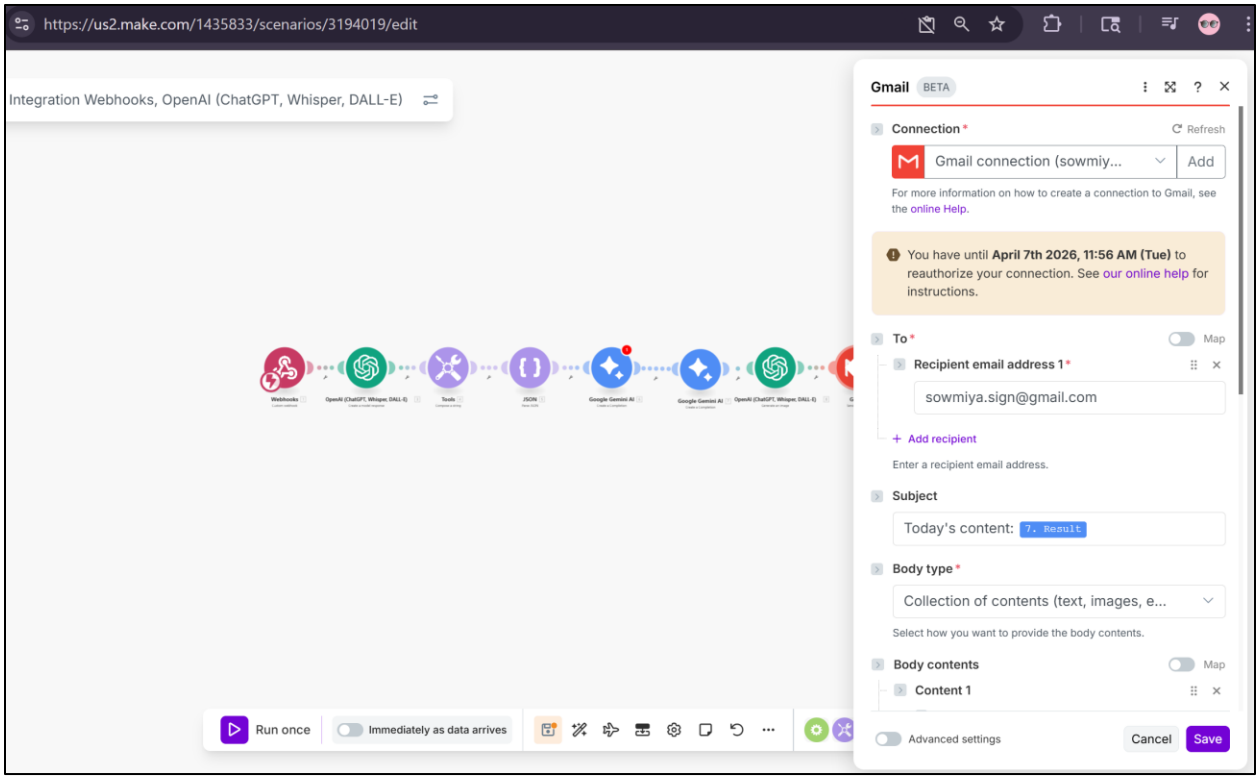
7. Send the results via email (Gmail module)

1. Add **Gmail → Send an email** module (or use another email provider module).
2. Connect your email account (authorise Make to send on your behalf).
3. Configure fields:
    - **To:** your email address
    - **Subject:** use the Gemini title/output variable (e.g., Today's top AI news: {{title}})
    - **Body:** insert the HTML returned by the Gemini text module (use the variable picker to place the HTML) — make sure the Gmail module supports HTML body.
    - **Attachments:** choose the image output from the image module. Set a content ID like AI_IMAGE if you want the image inline.

If you want the image inline:

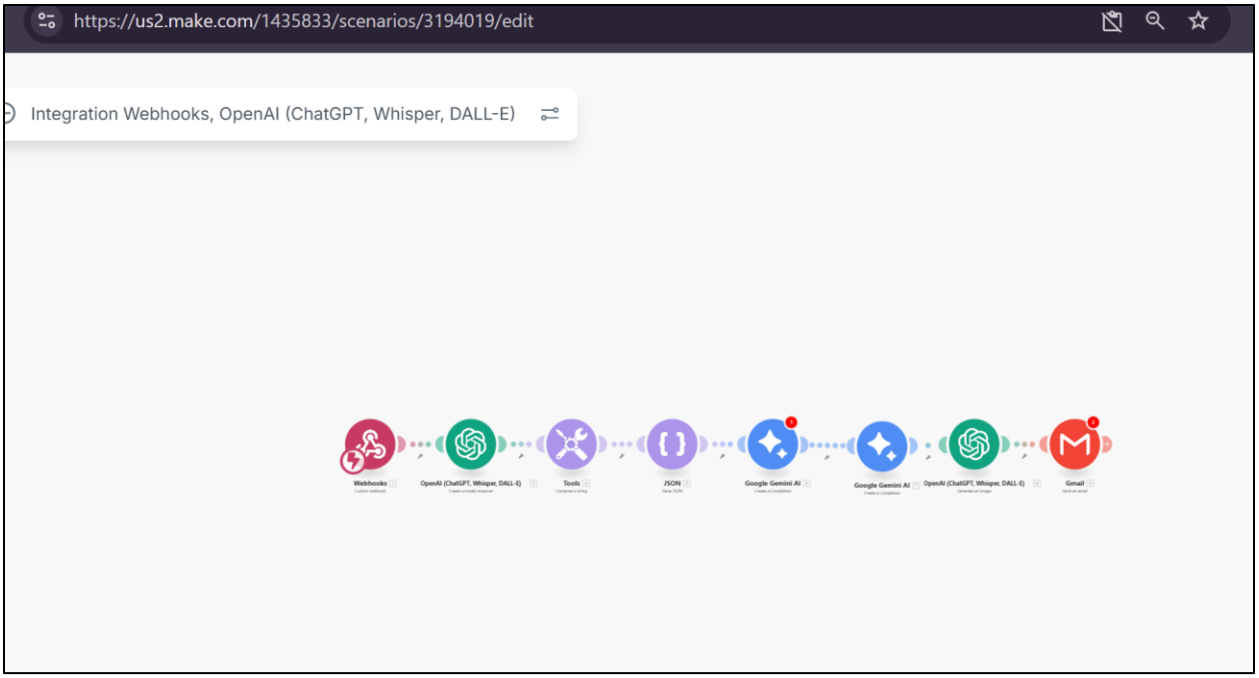
- In the email body reference the image with an inline CID: <img src="cid:AI_IMAGE" alt="AI image"> and set the attachment content ID to AI_IMAGE in the attachment mapping.

4. Save and run the whole scenario to verify the mail arrives with the HTML and the image.



8. Test the entire flow

1. Click **Save** and then **Run once** in Make.
2. Trigger the webhook (open the webhook URL in a browser or trigger via an HTTP request).
3. Inspect the execution results for each module:
    - Confirm Perplexity output contains title, summary, url.
    - Confirm the Gemini text completion returned HTML that matches your expected format.
    - Confirm image module returned a file.
    - Confirm Gmail module shows successful send.
4. If a module fails, click into it in Make and examine the error message and raw output for debugging.


https://us2.make.com/1435833/scenarios/3194019/edit
Integration Webhooks, OpenAI (ChatGPT, Whisper, DALL-E)
Webhooks
OpenAI (ChatGPT, Whisper, DALL-E)
Tools
JSON
Google Gemini AI
Google Gemini AI
OpenAI (ChatGPT, Whisper, DALL-E)
Gmail