**ComfyUI Custom node spec**

The objective is to create a custom ComfyUI node that interfaces with Google's Gemini 2.5 Flash Image model (a.k.a. Nano Banana) through API.

While a number of similar nodes like this exist, none of them provide the right combination of inputs, settings, and outputs that we need.

Specifically, one of our primary needs is to include an adjustable setting for content safety moderation, which the Google API enables, but very few nodes take advantage of.

This will be one of several nodes in the ETNodes node bundle.
Because of this, keep the node as a separate file, not just as a class within a common nodes.py file.

The requirements are as follows:

Node name/title: **ETNodes-Gemini-API-Image**

Inputs:

- 4 image input connection sockets that get batched and sent to the model (IMAGE format).
    - All inputs are optional: if no image is provided, the request should work as a text-to-image request.
        - Requests with no image provided may get internally routed to Imagen text-to-image models, but not sure about the exact mechanisms of that.
    - If at least one image is provided, it should work as an image editing request.
- A "prompt" text field (STRING format).

Settings:

- An "API_key" text single-line entry field (STRING format):
    - this expects the Gemini API key

    - if left blank, it tries to retrieve the API key from the GEMINI_API_KEY environment variable
- A “model” dropdown selector, defaulting to “gemini-2.5-flash-image”, additional options to be added later.
- A “safety_level” dropdown selector, with the following options:
    - none (corresponding to the BLOCK_NONE API option) - this is the default option
    - minimum (corresponding to the BLOCK_ONLY_HIGH API option)
    - medium (corresponding to the BLOCK_MEDIUM_AND_ABOVE API option)
    - maximum (corresponding to the BLOCK_LOW_AND_ABOVE API option)
- An “output_format” dropdown selector, with the options:
    - jpg
    - png - default option
- A “seed” entry selector, for the generation seed (INT format).
    - 42 as the default option
    - This may not be needed for image editing, although ComfyUI’s own node uses it. Investigate.
- A “control after generate” dropdown selector, with 4 options and associated behavior:
    - randomize - default option
    - increment
    - decrement
    - fixed
        - These may not be needed for image editing, although ComfyUI’s own node uses it. Investigate.
- A “num_images” entry selector, with options from 1 to 4 for the number of generated images in one request (INT format).
- An “aspect_ratio” dropdown selector, with the following options:
    - auto - default option
    - 1:1
    - 2:3
    - 3:2
    - 3:4
    - 4:3

- 4:5
- 5:4
- 9:16
- 16:9
- 21:9

Outputs:

- A single “image” output connection socket (IMAGE format).

The safety setting:

According to research the safety setting format in the API call looks something like this - I’m sure you can figure it out:

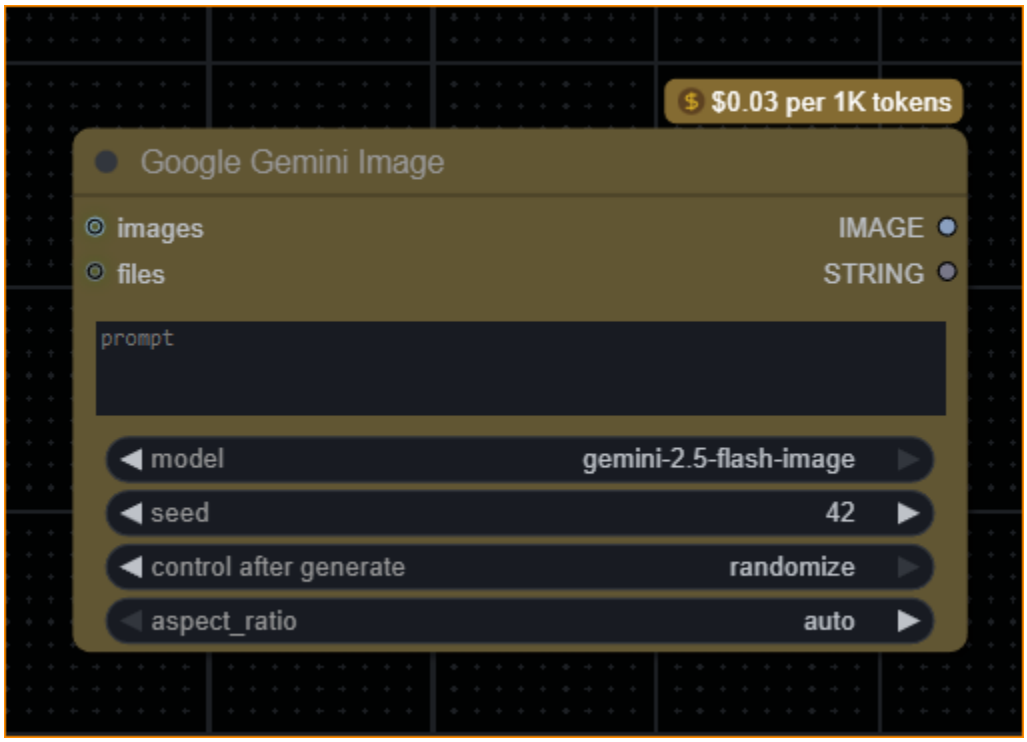
```
safety_settings = [

    {"category": HarmCategory.HARM_CATEGORY_HARASSMENT, "threshold": HarmBlockThreshold.BLOCK_NONE},
    {"category": HarmCategory.HARM_CATEGORY_HATE_SPEECH, "threshold": HarmBlockThreshold.BLOCK_NONE},
    {"category": HarmCategory.HARM_CATEGORY_SEXUALLY_EXPLICIT, "threshold": HarmBlockThreshold.BLOCK_NONE},
    {"category": HarmCategory.HARM_CATEGORY_DANGEROUS_CONTENT, "threshold": HarmBlockThreshold.BLOCK_NONE},

]
```

Additional requirements:

- Add a tooltip to all controls - for the time being the contents of the tooltips should just match the name of the control, to be filled out properly later.
- Make the header bar of the node the color #AE1016.

Reference

This is a screenshot of ComfyUI's own native Gemini API node - it's only for appearance reference, as it shares many commonalities with what we are trying to do here.




The source code of the same node - for reference for best practices - can also be found here:

D:\StabilityMatrix\Packages\ComfyUI\comfy_api_nodes\nodes_gemini.py