

Kling 2.1 + n8n

Image Prompt AI Agent

Prompt:

Create a prompt for a biblical scene

System Message:

** Purpose **

You transform a short-form video idea (inspired by a biblical character or scene, TikTok-vlog style) into a rich, highly descriptive image prompt for an AI image generation tool. Your goal is to keep the viral, relatable, and emotional vibe, but produce a single still image that captures the mood, story, and setting.

** Think Tool Prompt (used before generation) **

How can I turn this biblical TikTok-style idea and environment into a cinematic AI image prompt that feels relatable, humorous or emotional, and visually stunning as a single image

** Output Requirements **

Return only the complete, detailed image prompt, no extra text.
Each image prompt must include:

- Scene description: clear setting with terrain, buildings, light, and key details
- Character focus: who is in the scene, pose, expression, outfit (biblical era)
- Atmosphere and texture: light conditions, dust, wind, fabric movement, shadows, imperfections
- Visual style: cinematic tone, slight vintage softness or analog warmth, realistic and natural composition

** Example Output **

A biblical man stands in front of a half-built wooden ark, pointing toward the sky as if explaining something important. He wears worn ancient robes, his beard slightly unkempt, his expression determined but tired. Around him, muddy ground with scattered tools and straw. The sky is cloudy with soft golden light breaking through. A light breeze lifts the edges of his robe and stirs the straw. The image has a warm, cinematic feel with gentle film grain and natural shadows.

** Formatting Rules **

To make sure the output works inside JSON, follow these simple rules:

- don't use emojis in the text
- only provide text that is in proper JSON format
- don't provide "" marks inside description

Structured Output Parser:

```
{  
  "image_prompt": ""  
}
```

Create Video AI Agent

Prompt:

Create a short cinematic video prompt based on the following biblical image prompt: {{ \$json.output.image_prompt }}

System Message:

** Purpose **

You transform a detailed biblical image prompt into a cinematic video prompt and simple title designed for short-form AI video tools.

Your goal is to expand the static image into a living scene by describing motion, atmosphere, pacing, camera work, and subtle story dynamics while keeping the same mood and style.

** Think Tool Prompt (used before generation) **

How can I take this vivid biblical image description and turn it into an immersive, emotionally engaging video scene with clear camera movement, live action elements, and a short-form cinematic feel

** Input **

You receive one field:

image_prompt

** Output Requirements **

Return only the complete, detailed video prompt and title, no extra text.

Each video prompt must include:

- Scene expansion: build on the static image to add motion, action, and unfolding details
- Character focus: describe what the character does, subtle expressions, gestures, or interactions
- Camera movement: specify one cinematic motion (e.g. slow push-in, handheld tracking, dolly backward, orbit)
- Light and atmosphere: match the original lighting, but enrich with dynamic elements like wind, moving shadows, or changing clouds
- Visual style: keep a cinematic, slightly vintage or analog warmth, realistic texture, short-form pacing

** Example Output **

"prompt" A biblical man stands in front of a half-built wooden ark, gesturing passionately toward the sky as dark clouds slowly drift overhead. He shifts his stance, pointing back to the unfinished structure, his robe edges fluttering in the breeze. The camera follows him with a slow handheld tracking shot as he moves side to side, his voice unheard but his urgency clear. Mud and straw move underfoot while distant villagers pause to watch. Soft golden rays break through the clouds, creating shifting highlights on the ark's wooden beams. The scene has a warm, cinematic tone with natural textures and gentle vintage softness.

"title" This Man Stood Alone With His Unfinished Ark and a Message Nobody Wanted to Hear

** Formatting Rules **

To make sure the output works inside JSON, follow these simple rules:

- don't use emojis in the text
- only provide text that is in proper JSON format
- don't provide "" marks inside description
- no line breaks, don't use \n, just provide plain text

Structured Output Parser:

```
{  
  "video_prompt": "",  
  "title": ""  
}
```

[Google Sheets ← COPY HERE](#)

Create Image (HTTP Request)

Method: POST

URL: <https://queue.fal.run/fal-ai/flux-pro/v1.1>

Authentication: Generic Credential Type

Credential Type: Header Auth

Send Body: Enabled

Body Content Type: JSON

Specify Body: Using Fields Below

Name: prompt

Value:

```
{{ $('Image Prompt AI Agent').item.json.output.image_prompt }}
```

Get Image (HTTP Request)

Method: GET

URL: {{ \$json.response_url }}

Authentication: Generic Credential Type

Credential Type: Header Auth

Create Video with Kling 2.1 (HTTP Request)

Method: POST

URL: <https://queue.fal.run/fal-ai/kling-video/v2.1/standard/image-to-video>

Authentication: Generic Credential Type

Credential Type: Header Auth

Send Body: Enabled

Body Content Type: JSON

Specify Body: Using Fields Below

Parameter 1

Name: prompt

Value:

```
{{ $('Video Prompt AI Agent').item.json.output.video_prompt }}
```

Parameter 2

Name: image_url

Value:

```
{{ $json.images[0].url }}
```

Get Video (HTTP Request)

Method: POST

URL: [https://queue.fal.run/fal-ai/kling-video/requests/{{ \\$json.request_id }}/status](https://queue.fal.run/fal-ai/kling-video/requests/{{ $json.request_id }}/status)

Authentication: Generic Credential Type

Credential Type: Header Auth

Get Video (HTTP Request)

Method: POST

URL: {{ \$json.response_url }}

Authentication: Generic Credential Type

Credential Type: Header Auth

Upload to Blotato (HTTP Request)

Method: POST

URL: <https://backend.blotato.com/v2/media>

Authentication: Generic Credential Type

Generic Auth Type: Header Auth

Send Body: Enabled

Body Content Type: JSON

Specify Body: Using Fields Below

Name: url

Value: {{ \$('Get Video URL').item.json.video.url }}

1. Publish on TikTok (HTTP Request)

Method: POST

URL: <https://backend.blotato.com/v2/posts>

Authentication: Generic Credential Type

Generic Auth Type: Header Auth

Send Body: Enabled

Body Content Type: JSON

Specify Body: Using JSON

```
{
  "post": {
    "target": {
      "targetType": "tiktok",
      "isYourBrand": false,
      "disabledDuet": false,
      "privacyLevel": "PUBLIC_TO_EVERYONE",
      "isAiGenerated": true,
      "disabledStitch": false,
      "disabledComments": false,
      "isBrandedContent": false,
      "autoAddMusic": true
    },
  },
  "content": {
    "text": "{{ $('Create an Idea AI Agent').item.json.output.description }}",
    "platform": "tiktok",
    "mediaUrls": ["{{ $('Upload to Blotato').item.json.url }}"],
  },
  "accountId": "5757"
}
```

2. Publish on Youtube (HTTP Request)

Method: POST

URL: <https://backend.blotato.com/v2/posts>

Authentication: Generic Credential Type

Generic Auth Type: Header Auth

Send Body: Enabled

Body Content Type: JSON

Specify Body: Using JSON

```
{
  "post": {
    "target": {
      "targetType": "youtube",
      "title": "{{ $('Video Prompt AI Agent').item.json.output.title }}",
      "privacyStatus": "unlisted",
      "shouldNotifySubscribers": false
    },
    "content": {
      "text": "{{ $('Video Prompt AI Agent').item.json.output.title }}",
      "platform": "youtube",
      "mediaUrls": [{{ $('Upload to Blotato').item.json.url }}]
    },
    "accountId": "4416"
  }
}
```

3. Publish on Facebook (HTTP Request)

Method: POST

URL: <https://backend.blotato.com/v2/posts>

Authentication: Generic Credential Type

Generic Auth Type: Header Auth

Send Body: Enabled

Body Content Type: JSON

Specify Body: Using JSON

```
{
  "post": {
    "accountId": "YOUR_ACCOUNT_ID",
    "target": {
      "targetType": "facebook",
      "pageId": "YOUR_PAGE_ID"
    },
    "content": {
      "text": "YOUR_TEXT_VARIABLE",
      "mediaUrls": ["YOUR_VIDEO_URL"],
      "platform": "facebook"
    }
  }
}
```

4. Publish on LinkedIn (HTTP Request)

Method: POST

URL: <https://backend.blotato.com/v2/posts>

Authentication: Generic Credential Type

Generic Auth Type: Header Auth

Send Body: Enabled

Body Content Type: JSON

Specify Body: Using JSON

```
{
  "post": {
    "target": {
      "targetType": "linkedin"
    },
    "content": {
      "text": "YOUR_TEXT",
      "platform": "linkedin",
      "mediaUrls": ["YOUR_VIDEO_URL"]
    },
    "accountId": "YOUR_ACCOUNT_ID"
  }
}
```