

Utilizando la API de Gemini en nuestros proyectos



Google AI Studio. Get API key

Google AI Studio

Get API key

Create new prompt

New tuned model

My library

Castillo de Ponferrada

View all

Getting started

Documentation

Prompt gallery

Gemini cookbook

Discourse forum

Build with Vertex AI on Google Cloud

Obtén una clave de API

Claves de APIs

Puedes crear un proyecto nuevo si aún no tienes uno o agregar claves de APIs a un proyecto existente. Todos los proyectos están sujetos a las [Condiciones del Servicio de Google Cloud Platform](#), que aceptas cuando creas un proyecto nuevo. El uso de la API de Gemini y de Google AI Studio está sujeto a las [Condiciones del Servicio de la API de Gemini](#).

Usa tus claves de APIs de forma segura. No las compartas ni las incorpores en código que el público pueda ver.

Si usas la API de Gemini desde un proyecto que tiene la facturación habilitada, tu uso estará sujeto a [precios de pago por uso](#).

CLAVE [Crear clave de API](#)

Tus claves de APIs se enumeran a continuación. También puedes ver y administrar tu proyecto y tus claves de APIs en Google Cloud.

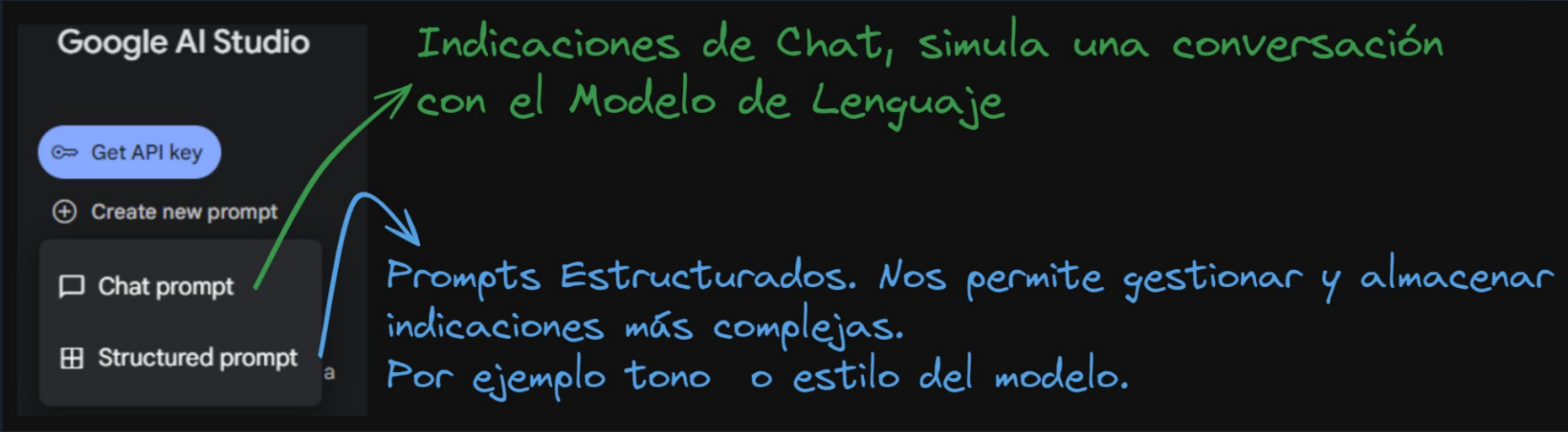
Número del proyecto	ID del proyecto	Clave de API	Fecha de creación	Plan
...5863	Generative Language Client	...e9Eo	28 may 2024	Pagada Ir a Facturación

Prueba rápidamente la API ejecutando un comando cURL

```
curl \
-H 'Content-Type: application/json' \
-d '{"contents":[{"parts":[{"text":"Explain how AI works"}]}]}' \
-X POST 'https://generativelanguage.googleapis.com/v1beta/models/gemini-1.5-flash-latest:generateContent?key=YOUR_API_KEY'
```

obtener
clave API

Google AI Studio. Indicaciones / Prompts



The image shows a screenshot of the Google AI Studio interface. On the left, there is a sidebar with the title "Google AI Studio". Below the title, there is a blue button labeled "Get API key". Below that, there is a link labeled "+ Create new prompt". At the bottom of the sidebar, there are two options: "Chat prompt" with a square icon and "Structured prompt" with a grid icon. A green arrow points from the "Chat prompt" option to the handwritten text "Indicaciones de Chat, simula una conversación con el Modelo de Lenguaje". A blue arrow points from the "Structured prompt" option to the handwritten text "Prompts Estructurados. Nos permite gestionar y almacenar indicaciones más complejas. Por ejemplo tono o estilo del modelo."

Google AI Studio

Get API key

+ Create new prompt

Chat prompt

Structured prompt

Indicaciones de Chat, simula una conversación con el Modelo de Lenguaje

Prompts Estructurados. Nos permite gestionar y almacenar indicaciones más complejas. Por ejemplo tono o estilo del modelo.

Google AI Studio. Estructura del Prompt.

Castillo de Ponferrada 

Insert:  Image

1 / 500 examples Actions ▾

☐ INPUT input:	☐ OUTPUT output:
☐ Datos del Castillo	<pre>## Datos técnicos de un castillo: **Estructura:** ***Ubicación:** ¿En qué región, país y coordenadas se encuentra? ***Tipo:** ¿Castillo real, fortaleza, castillo de montaña, etc.? ***Material de construcción:** ¿Piedra, ladrillo, madera, etc.? ***Planos:** ¿Planos de planta, elevación y sección? ***Número de torres:** ¿Número total y tipos de torres (redondas, cuadradas, etc.)? ***Muralla:** ¿Altura y grosor de las murallas? ***Puertas:** ¿Número y tipos de puertas (de entrada, de servicio, etc.)? ***Foso:** ¿Presencia de foso y tipo (seco, lleno de agua)? ***Otros elementos defensivos:** ¿Barbacana, torreón, puente levadizo, etc.? **Historia:** ***Fecha de construcción:** ¿Año de inicio y finalización de la construcción? ***Propósito:** ¿Para defensa militar, residencia real, prisión, etc.?</pre>

```
1 import os
2 import google.generativeai as genai
3
4 genai.configure(api_key=os.environ["GEMINI_API_KEY"])
5
6 # Crear el modelo
7 generation_config = {
8
9 }
10
11 model = genai.GenerativeModel(
12     model_name="gemini-1.5-flash",
13     generation_config=generation_config,
14     # safety_settings = Ajustar la configuración de seguridad
15 )
16
17 response = model.generate_content([
18     "input: Datos del Castillo",
19     "output: ## Datos técnicos de un castillo:\n\n**Estructura:**\n\n* **Ubicación:** ¿En" # Etc...
20     "input: ",
21     "Castillo de Ponferrada",
22     "output: ",
23 ])
24
25 print(response.text)
```

Google AI Studio. Configuración

```
1 import os
2 import google.generativeai as genai
3
4 genai.configure(api_key=os.environ["GEMINI_API_KEY"])
5
6 # Crear el modelo
7 generation_config = {
8
9 }
10
11 model = genai.GenerativeModel(
12     model_name="gemini-1.5-flash",
13     generation_config=generation_config,
14     # safety_settings = Ajustar la configuración de seguridad
15 )
16
17
18 response = model.generate_content([
19     "input: Datos del Castillo",
20     "output: ## Datos técnicos de un castillo:\n\n**Estructura:**\n\n* **Ubicación:** ¿En" # Etc...
21     "input: ",
22     "Castillo de Ponferrada",
23     "output: ",
24 ])
25
26 print(response.text)
```

Configurar los parámetros
Generales de Gemini

Seleccionar
el Modelo

Ajustar la configuración de
Seguridad

Google AI Studio. Configuración

Model

Gemini 1.5 Flash

Modelo a utilizar

Token Count

716 / 1,048,576

Temperature

Temperatura

La "temperatura" es un parámetro que controla la aleatoriedad de las respuestas generadas por el modelo en términos de creatividad y diversidad en sus respuesta.

- Temperatura Baja (cercana a 0): el modelo tiende a ser más conservador y predecible.

- Temperatura Alta (cercana a 1): el modelo introduce más aleatoriedad en la selección de palabras, permitiendo respuestas más creativas y variadas.

Add stop sequence

Add stop...

Stop Sequence

Una cadena de texto específica que indica al modelo que debe detener la generación de más texto. (Incluso Truncando el contenido)

Safety settings

Edit safety settings

Advanced settings

Output length

8192

Tokens

Número de Tokens máximo de salida esperados.

Tokens de Entrada / Máximo

Top P

0,95

Top P

Limita la selección de tokens a un subconjunto cuya probabilidad acumulada no excede p .
Top-p = 0.95, significa que el modelo considerará tokens cuya probabilidad acumulada alcance el 95% más probable

Google AI Studio. Configuración

Variantes del modelo

La API de Gemini ofrece diferentes modelos optimizados para casos de uso específicos. A continuación, se incluye una breve descripción general de las variantes disponibles de Gemini:

Variante del modelo	Entrada(s)	Resultado	Optimizado para
Gemini 1.5 Pro gemini-1.5-pro	Audio, imágenes, videos y texto	Texto	Tareas de razonamiento complejas, como la generación de código y texto, la edición de texto, la resolución de problemas y la extracción y generación de datos
Gemini 1.5 Flash gemini-1.5-flash	Audio, imágenes, videos y texto	Texto	Rendimiento rápido y versátil en una amplia variedad de tareas
Gemini 1.0 Pro gemini-1.0-pro	Texto	Texto	Tareas de lenguaje natural, chat de texto y código de varios turnos, y generación de código
Gemini 1.0 Pro Vision gemini-pro-vision	Imágenes, videos y texto	Texto	Tareas relacionadas con el aspecto visual, como generar descripciones de imágenes o identificar objetos en imágenes
Incorporación de texto text-embedding-004	Texto	Incorporaciones de texto	Medir la relación de las cadenas de texto

Google AI Studio. Configuración



```
1 import google.generativeai as genai
2
3
4 genai.configure(api_key=GOOGLE_API_KEY)
5
6 for model in genai.list_models():
7     if 'generateContent' in model.supported_generation_methods:
8         print(model.name)
```

Google AI Studio. Configuración

Umbral (Google AI Studio)	Umbral (API)	Descripción
No bloquear	BLOCK_NONE	Mostrar siempre, independientemente de la probabilidad de que haya contenido no seguro
Bloquear poco	BLOCK_ONLY_HIGH	Bloquear cuando haya alta probabilidad de incluir contenido no seguro
Bloquear algo	BLOCK_MEDIUM_AND_ABOVE	Bloquear cuando la probabilidad de tener contenido no seguro es media o alta
Bloquear la mayoría	BLOCK_LOW_AND_ABOVE	Bloquear cuando la probabilidad de que exista contenido no seguro sea baja, media o alta
	HARM_BLOCK_THRESHOLD_UNSPECIFIED	El umbral no está especificado; se bloquea con el umbral predeterminado

Google AI Studio. Configuración

 Run safety settings 

Adjust how likely you are to see responses that could be harmful. Content is blocked based on the probability that it is harmful.

Harassment

Block most



Hate Speech

Block some



Sexually Explicit

Block some



Dangerous Content

Block some



Reset defaults

```
1 from google.generativeai.types import HarmCategory, HarmBlockThreshold
2
3 model = genai.GenerativeModel(model_name='gemini-1.5-flash')
4 response = model.generate_content(
5     ['¿Como hacer una bomba??'],
6     safety_settings={
7         HarmCategory.HARM_CATEGORY_HATE_SPEECH: HarmBlockThreshold.BLOCK_LOW_AND_ABOVE,
8         HarmCategory.HARM_CATEGORY_HARASSMENT: HarmBlockThreshold.BLOCK_LOW_AND_ABOVE,
9     }
10 )
```