

# Using the Google Gemini Model as the Large Language Model in your RAG Application

**Step 1:** Login to Google. Create a Google Account if you don't have one.

**Step 2:** Get the **Gemini API Key**

Navigate to the url: <https://aistudio.google.com/app/apikey>

**API keys**

You can create a new project if you don't have one already or add API keys to an existing project. All projects are subject to the [Google Cloud Platform Terms of Service](#), which you agree to when creating a new project, while use of the Gemini API and Google AI Studio is subject to the [Gemini API Terms of Service](#).

Use your API keys securely. Do not share them or embed them in code the public can view.

If you use Gemini API from a project that has billing enabled, your use will be subject to [pay-as-you-go pricing](#).

**Create API key**

Your API keys are listed below. You can also view and manage your project and API keys in Google Cloud.

Google Cloud project name	API key	Action	Created	Plan
Generative Language Client	<a href="#">...eWHs</a>		Mar 11, 2024	Free of charge <a href="#">Set up Billing</a>

- Click on the Button - “Create API Key” - this will create a new API Key for you.
- Copy this API Key.

**Step 3: Update your .env File**

Add this line in your .env file:

GOOGLE\_API\_KEY=<your Google API Key>

#### Step 4: Update your RAG Application.

Navigate to this URL: <https://ai.google.dev/gemini-api/docs/models/gemini>

- You will find all the Gemini models here

#### Step 5: Update your RAG code. A Sample code is given below to take help from:

```
from langchain_google_genai import ChatGoogleGenerativeAI
# Import Google API Key
from dotenv import load_dotenv
import os
load_dotenv()

# Set your LLM to Google Gemini Model
llm = ChatGoogleGenerativeAI(model="gemini-pro",
google_api_key=os.getenv("GOOGLE_API_KEY"))

response = llm.invoke("Who is mahatma gandhi")
print(response.content)
```

For most scenarios, once you set the “llm” in your RAG to a Gemini model, the remaining code should be the same.