# Build a QnA ChatBot using Gemini Pro

Last Updated : 05 Jan, 2024

interactions. They use natural language processing (NLP) and machine learning algorithms to understand and respond to user queries, providing a personalized experience.

**Gemini** is an AI model made by the Google Deepmind team. It is a very capable model which is a multimodal type. That means we can generate text, analyze images, audio, video, etc. We can make text generators, chatbots, image analyzers, etc.

In this tutorial, we will build a QnA ChatBot using Gemini Pro API and Flask in Python. The API is available for free with which we will create a web application.

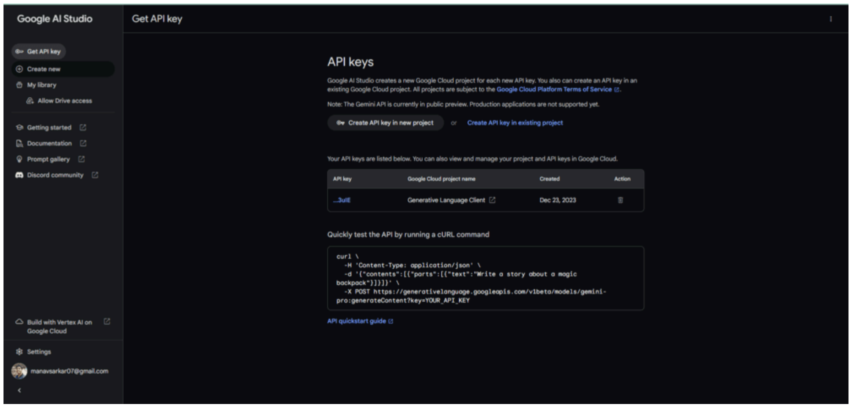
## Create An API for Gemini Pro

Below are the steps to create an API for Gemini Pro:

### Get API Key

Navigate to **Google AI Studio**(https://makersuite.google.com/app/apikey) and click on **Get API key**.

```
https://makersuite.google.com/app/apikey
```



API keys

Google AI Studio creates a new Google Cloud project for each new API key. You also can create an API key in an existing Google Cloud project. All projects are subject to the Google Cloud Platform Terms of Service.

Note: The Gemini API is currently in public preview. Production applications are not supported yet.

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

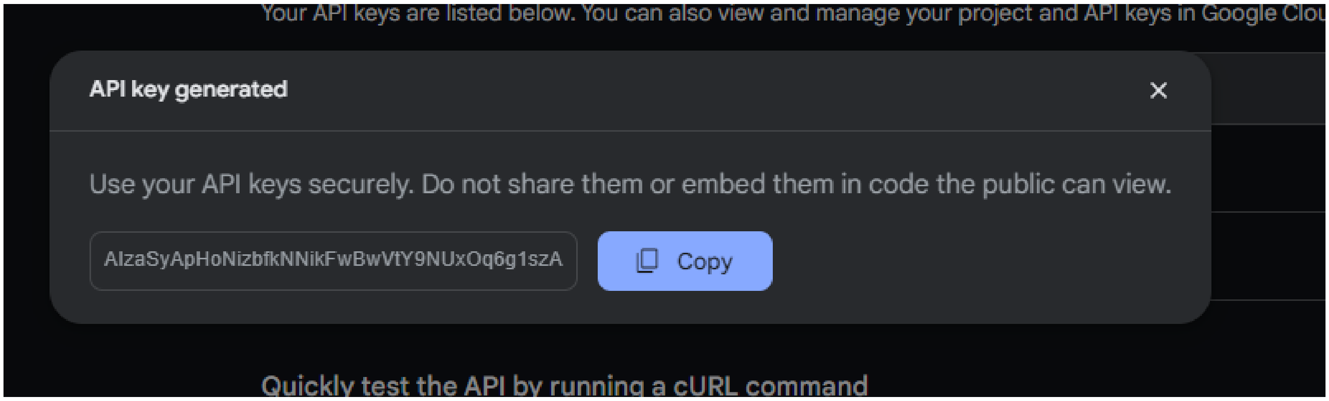
| API key | Google Cloud project name | Created | Action |
|---|---|---|---|
| ...3uIE | Generative Language Client | Dec 23, 2023 | |

Quickly test the API by running a cURL command

```
curl \
  -H 'Content-Type: application/json' \
  -d '{"contents":[{"parts":[{"text":"Write a story about a magic
backpack"}]}]}' \
  -X POST https://generativelanguage.googleapis.com/v1beta/models/gemini-
pro:generateContent?key=YOUR_API_KEY
```

### Create a API Key

Create API Key in new project button, and copy the generated API key. Copy the API Key and use in generating the chatbot.



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

API key generated

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

AIzaSyApHoNizbfkNNikFwBwVfY9NUxOq6g1szA

Quickly test the API by running a cURL command

## Creating a QnA Chatbot Using Flask

Below are the step by step procedure to build a QnA Chatbot using Gemini Pro and Flask in Python:

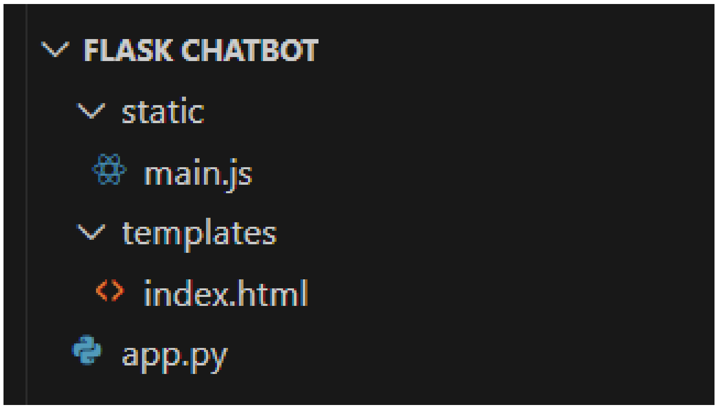
### Installation

Before starting, we will have to install the following things to progress further:

- Install Flask
- Install Python

### File Structure

Project Structure should be as follows:



FLASK CHATBOT
- static
  - main.js
- templates
  - index.html
- app.py

### Create a Python File

In this Python code, a Flask web application is initialized with an instance named `app`. A route decorator directs the root URL to render the **`index.html`** template. When executed, the Flask application runs in debug mode, facilitating development with real-time error feedback.

**app.py**

Python3

```python
from flask import Flask, render_template

app = Flask(__name__)

@app.route('/')
def index():
    return render_template('index.html')

if __name__ == '__main__':
    app.run(debug=True)
```

### Create a UI in HTML

Inside the **templates folder**, create "**index.html**" file and start writing this code. In this HTML code, a chatbot interface is structured with a title, message display area, and input field. External libraries, Tailwind CSS for styling and Showdown for markdown conversion, are imported. Additionally, Google's Generative AI is integrated, and a JavaScript module (`main.js`) is included to manage chatbot interactions.

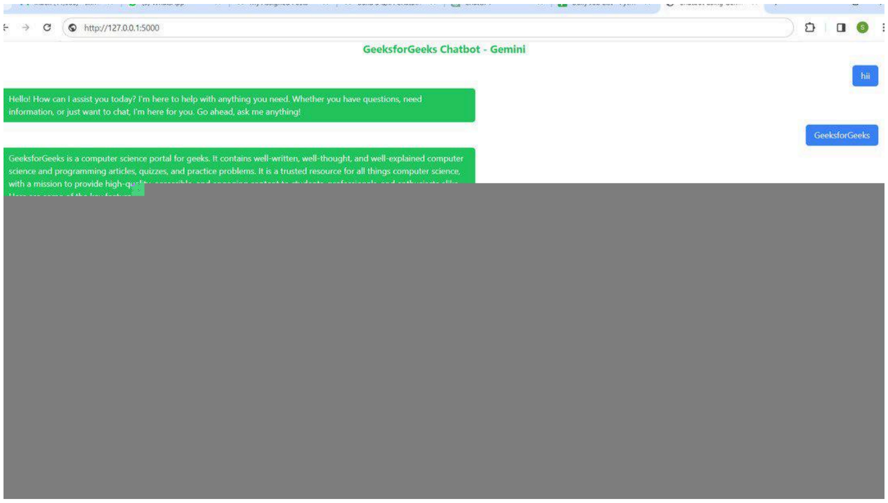
**index.html**

HTML

```html
<!DOCTYPE html>
<html lang="en">
  <head>
```

```
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width,
                                   initial-scale=1.0" />
    <title>Chatbot using Gemini | GeeksforGeeks</title>
    <script src="https://cdn.tailwindcss.com"></script>
    <script
       src="https://cdnjs.cloudflare.com/ajax/libs/showdown/2.1.0/showdown.min.js"></script>
    <style type="text/tailwindcss">
      @layer base {
        ul {
          list-style-type: circle;
        }
        ol {
          list-style-type: decimal;
        }
      }
    </style>
  </head>
  <body class="flex flex-col justify-between h-screen">
    <div>
      <p class="text-xl font-bold text-center text-green-500">
        GeeksforGeeks Chatbot - Gemini
      </p>
      <div class="overflow-y-auto" style="max-height: 80vh">
        <div id="messageHolder" class="flex flex-col m-4"></div>
      </div>
    </div>
    <div class="flex flex-row m-4">
      <input
        type="text"
        class="flex-1 border rounded-md m-2 border-green-600
               p-2 outline-none ring-2 ring-green-600
               border-transparent"
        placeholder="Chat..."
        name="chat"
        id="chat"
      />
      <button id="btn" class="m-2 bg-blue-500 p-2
                              rounded-md text-white">
        Send
      </button>
    </div>
    <script type="importmap">
      {
        "imports": {
          "@google/generative-ai": "https://esm.run/@google/generative-ai"
        }
      }
    </script>
    <script type="module" src="{{ url_for('static',
                               filename='main.js') }}"></script>
  </body>
</html>
```

Output



http://127.0.0.1:5000

GeeksforGeeks Chatbot - Gemini

hii

Hello! How can I assist you today? I'm here to help with anything you need. Whether you have questions, need information, or just want to chat, I'm here for you. Go ahead, ask me anything!

GeeksforGeeks

GeeksforGeeks is a computer science portal for geeks. It contains well-written, well-thought, and well-explained computer science and programming articles, quizzes, and practice problems. It is a trusted resource for all things computer science, with a mission to provide high-q

## Write JavaScript File

Inside the **static folder**, create a "**main.js**" file and start writing this code. In this [JavaScript](#) code, the `GoogleGenerativeAI` module is initialized with an API key to establish a chat instance with the "gemini-pro" model. The `chatGemini` function manages user interactions, sending messages for processing and displaying the chatbot's responses. Utility functions render messages on the webpage, and event listeners capture user inputs for chat interactions.

**chatGemini():** Here we interact with the API and send the message and store the result in variable HTML after converting the markdown text to HTML using showdownjs.

**addMessage():** Here we add a div container with the message content each time a message is added from user as well as gemini pro bot.

JavaScript

```javascript
import { GoogleGenerativeAI } from "@google/generative-ai";
const conv = new showdown.Converter();

const genAI = new GoogleGenerativeAI("AIzaSyCg3KDDSpMZGQnzlFWHAXhr4fBjkc_3uIE");
const gen_model = genAI.getGenerativeModel({ model: "gemini-pro" });
const chat = gen_model.startChat({
    generationConfig: {
        maxOutputTokens: 1000,
    },
});

const chatGemini = async (message) => {
    addMessage(message, "end");
    let res = await chat.sendMessage(message);
    res = await res.response;
    console.log(res);
    let html = conv.makeHtml(res.text());
    addMessage(html, "start");
}
const addMessage = (msg, direction) => {
    const messageHolder = document.getElementById("messageHolder");
    const message = document.createElement("div");
    const colour = direction !== "start" ? "blue" : "green";
    message.innerHTML = `
     <div class="flex flex-col items-${direction}">
            <div class="bg-${colour}-500 px-4 py-2 rounded-md text-white w-fit
            max-w-4xl mb-1">${msg}</div>
          </div>
    `
    messageHolder.appendChild(message);
}

const messageInput = document.getElementById("chat");
const sendBtn = document.getElementById("btn");

sendBtn.addEventListener("click", function () {
    const message = messageInput.value;
    chatGemini(message);
    messageInput.value = "";
});
```

**In Depth Explaination**

To start a chat session, we first need to import the **Google GenerativeAI module**

```
import { GoogleGenerativeAI } from "@google/generative-ai";
```

Then, we need to create a model for chat sessions called **gemini-pro**.

```
const genAI = new GoogleGenerativeAI("API_KEY_HERE");
const gen_model = genAI.getGenerativeModel({ model: "gemini-pro" });
```

After creating the model, we need to start a chat session using the **startChat()** method and pass parameters accordingly.

```
const chat = gen_model.startChat({
    generationConfig: {
        maxOutputTokens: 1000,
    },
});
```

To send a message and receive response, we use sendMessage and text methods.

```
let res = await chat.sendMessage(message);
res = await res.response;
let text= res.text();
```

The text returned is in markdown format and we will convert it into HTML for our purpose.

## Run the Program

To run this Flask application, just write the following command in your terminal:

```
python app.py
```

Access the application in your web browser at `http://127.0.0.1:5000`

### Video Demonstration

M mana... 1

Previous Article

Next Article

10 AI Chatbots for Creative Content Generation

## Similar Reads

### Gemini Pro vs Gemini Ultra: Review (2024)

Gemini is the latest artificial intelligence model developed by Google. It is a cutting-edge technology that goes beyond text comprehension. It possesses the remarkable ability to understand and analyze images, videos, and audio. As a multimodal...

8 min read

### Difference Between Gemini Nano vs Gemini Pro

Gemini is an advanced artificial intelligence model developed by Google. It possesses the remarkable ability to comprehend not only text but also images, videos, and audio. This multimodal model is renowned for its proficiency in tackling intricate tasks in...

8 min read

### Gemini Advanced vs ChatGPT Plus vs Copilot Pro: Which AI Chatbot to Subscribe to?

In the fast-paced digital world, chatbots backed by AI are being employed by businesses and individuals to make operations more efficient, improve customer service, and enhance communication channels. Various platforms have been developed with t...

6 min read

### Gemini Pro vs GPT-3.5: Which Tool Is Best?

In this world today, the advancement of AI has been quite fast in the past few years. The AI today can do a lot of work that was previously a dream. AI can today write essays, give you recipes for a delicious salad, generate a highly detailed image for you of...

9 min read

### What Is Google Gemini AI? How to Use the New Chatbot Model

Big news came in the world of artificial intelligence as Google unveiled Gemini, a powerful AI model family set to rival OpenAI's GPT-4. According to Google, Gemini's standout feature is its ability to outperform nearly all other models, showcasing...

6 min read

### Creating a Telegram Chatbot Powered by Gemini AI

Imagine having a smart, always-available assistant right in your favorite messaging app! That's what happens when we bring together two exciting technologies: artificial intelligence (AI) and messaging platforms. In this article, we'll walk you through...

6 min read

### Dataset for Chatbot : Key Features and Benefits of Chatbot Training Datasets

Chatbots rely on high-quality training datasets for effective conversation. These datasets provide the foundation for natural language understanding (NLU) and dialogue generation. Furthermore, transformer-based models like BERT or GPT are powerf...

7 min read

### Build an AI Chatbot in Python using Cohere API

A chatbot is a technology that is made to mimic human-user communication. It makes use of machine learning, natural language processing (NLP), and artificial intelligence (AI) techniques to comprehend and react in a conversational way to user inquiries or...

4 min read

### Build Chatbot Webapp with LangChain

LangChain is a Python module that allows you to develop applications powered by language models. It provides a framework for connecting language models to other data sources and interacting with various APIs. LangChain is designed to be easy to use,...

13 min read

### AI Revolution in Video Editing: Adobe’s Premiere Pro and OpenAI Partnership

The world of video editing is on the cusp of a revolution. Adobe, the industry leader with its widely-used Premiere Pro software, is exploring a partnership with OpenAI, a cutting-edge research lab specializing in artificial intelligence (AI). This collaboration...

6 min read

**Article Tags :**

Machine Learning

Geeks Premier League

AI-ML-DS

Gemini

+5 More

**Practice Tags :**

Machine Learning

python

Corporate & Communications Address:-
A-143, 9th Floor, Sovereign Corporate
Tower, Sector- 136, Noida, Uttar Pradesh
(201305) | Registered Address:- K 061,
Tower K, Gulshan Vivante Apartment,
Sector 137, Noida, Gautam Buddh
Nagar, Uttar Pradesh, 201305

**Company**

About Us

Legal

In Media

Contact Us

Advertise with us

GFG Corporate Solution

Placement Training Program

GeeksforGeeks Community

**DSA**

Data Structures

Algorithms

DSA for Beginners

Basic DSA Problems

DSA Roadmap

Top 100 DSA Interview Problems

DSA Roadmap by Sandeep Jain

All Cheat Sheets

**Web Technologies**

HTML

CSS

JavaScript

TypeScript

ReactJS

NextJS

Bootstrap

Web Design

**Computer Science**

Operating Systems

Computer Network

Database Management System

Software Engineering

Digital Logic Design

**Languages**

Python

Java

C++

PHP

GoLang

SQL

R Language

Android Tutorial

Tutorials Archive

**Data Science & ML**

Data Science With Python

Data Science For Beginner

Machine Learning

ML Maths

Data Visualisation

Pandas

NumPy

NLP

Deep Learning

**Python Tutorial**

Python Programming Examples

Python Projects

Python Tkinter

Web Scraping

OpenCV Tutorial

Python Interview Question

Django

**DevOps**

Git

Linux

AWS

Docker

Kubernetes

Engineering Maths
Software Development
Software Testing

Azure
GCP
DevOps Roadmap

**System Design**
High Level Design
Low Level Design
UML Diagrams
Interview Guide
Design Patterns
OOAD
System Design Bootcamp
Interview Questions

**Inteview Preparation**
Competitive Programming
Top DS or Algo for CP
Company-Wise Recruitment Process
Company-Wise Preparation
Aptitude Preparation
Puzzles

**School Subjects**
Mathematics
Physics
Chemistry
Biology
Social Science
English Grammar
Commerce
World GK

**GeeksforGeeks Videos**
DSA
Python
Java
C++
Web Development
Data Science
CS Subjects