UmmaBot Documentation and Setup Guide

This document gives a guide on setting up UmmaBot end-to-end: environment prerequisites, API keys, Google Sheets logging, installing dependencies, running the backend, wiring the frontend, retraining the bot with a new PDF, hardening CORS, and common troubleshooting.

1) Overviews

Architecture

- **Frontend**: Static HTML/CSS/JS (chat widget + upload page).
- **Backend**: FastAPI (UmmaChatbot.py) serving:
  - POST /chat – main Q&A
  - POST /feedback – lightweight feedback logger
  - POST /upload – upload a training PDF which builds a new Chroma DB
  - POST /admin/reload_db – reload the currently active Chroma DB pointed to by active_db.txt
  - GET /ping – health check
- **Vector DB**: Chroma, with unique directories under db_groq/<uuid>/. The "active" DB path is stored in active_db.txt.
- **LLM**: Groq (Llama 3) via langchain_groq.
- **Embeddings**: HuggingFaceEmbeddings (all-MiniLM-L6-v2).
- **Logging**: Google Sheets log ("UmmaBot Logs" spreadsheet).

2) Prerequisites

- **Python** 3.10+
- **pip** (latest)
- A **Groq** account & API key

- A **Google Cloud Service Account** with Google Sheets and Google Drive APIs enabled, plus a service key JSON file

3) Environment Variables (.env)

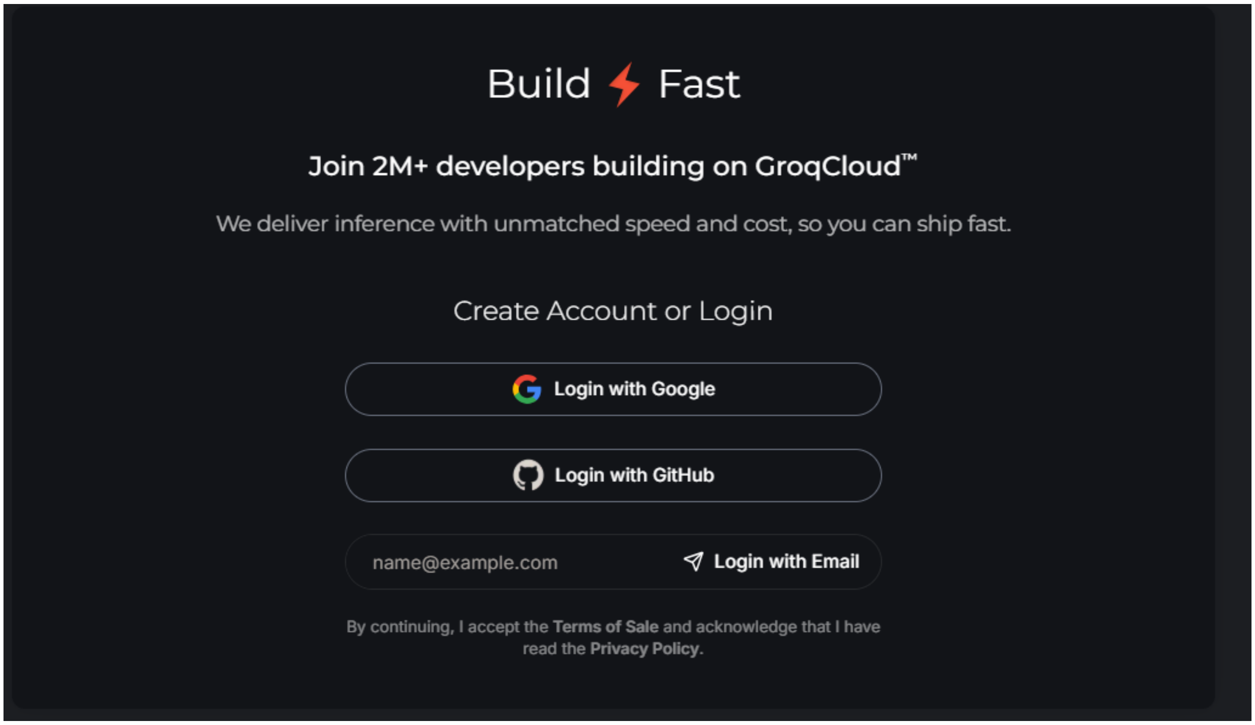
Update the .env file in the project root with new Groq key:

GROQ_API_KEY=_groq_api_key_here

The backend reads this at startup via python-dotenv.

Follow the steps below to get the api key

1. Open the Groq Console
   Go to: https://console.groq.com/
2. Create an account or log in
   On the console home you'll see "Create Account or Login". You can sign in with Google, GitHub, or email.



Build ⚡ Fast

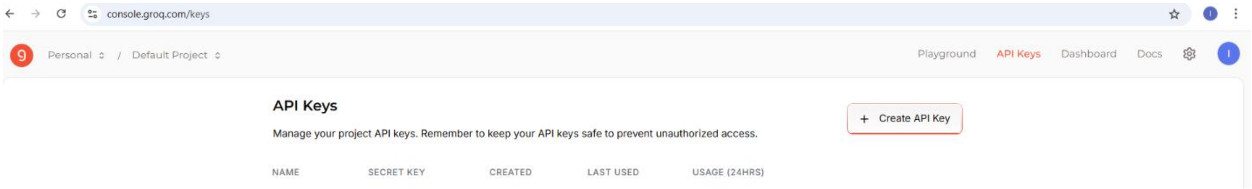
Join 2M+ developers building on GroqCloud™

We deliver inference with unmatched speed and cost, so you can ship fast.

Create Account or Login

By continuing, I accept the Terms of Sale and acknowledge that I have read the Privacy Policy.

3. Go to the API Keys page
   link: https://console.groq.com/keys

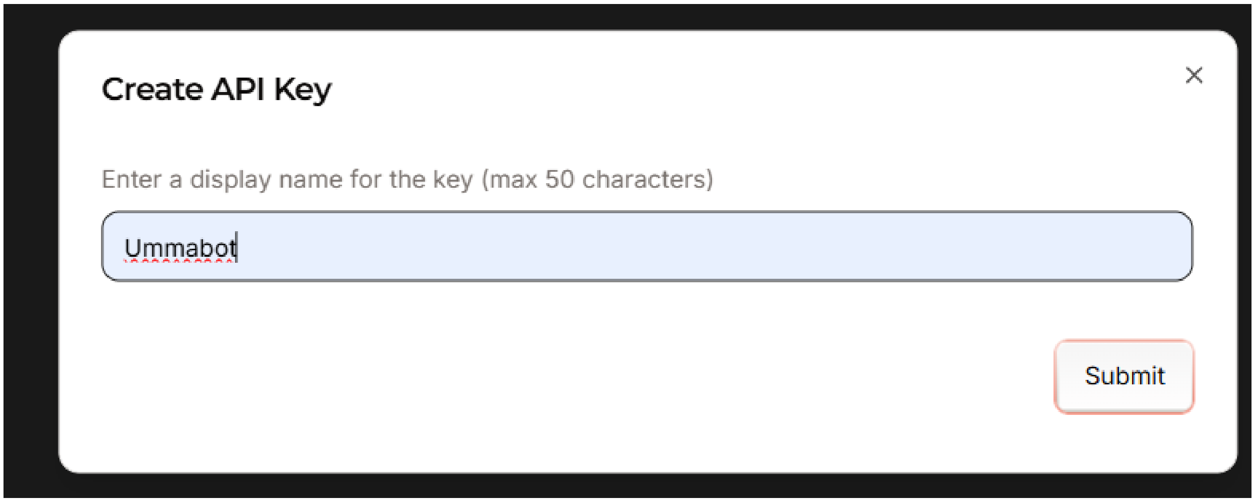


API Keys

Manage your project API keys. Remember to keep your API keys safe to prevent unauthorized access.

NAME SECRET KEY CREATED LAST USED USAGE (24HRS)

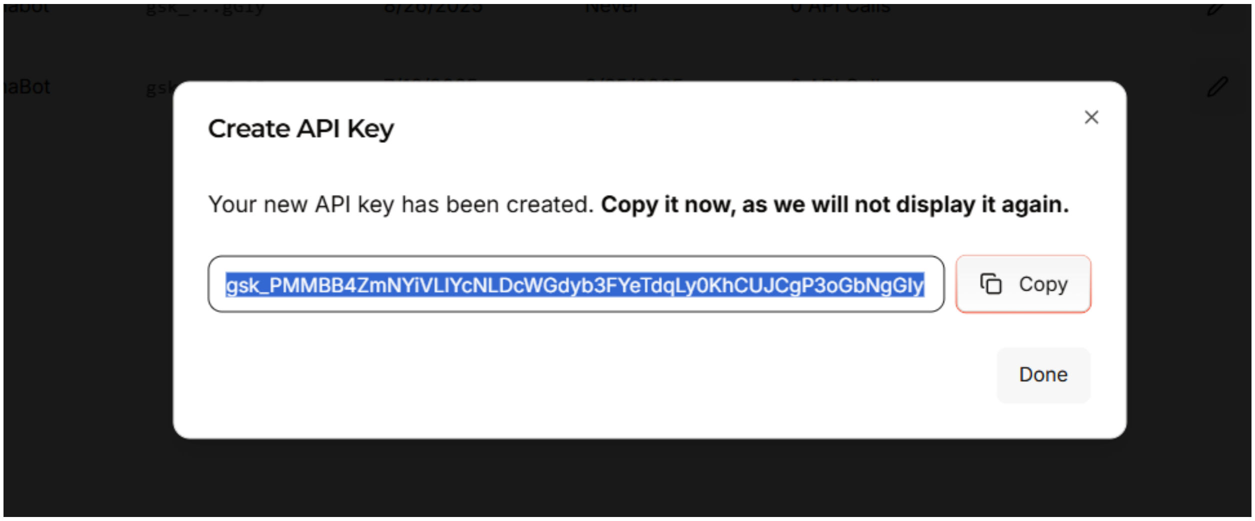
4. Create a new key
   Click "Create API Key", give it a name like "UmmaBot" confirm and copy it.



Create API Key

Enter a display name for the key (max 50 characters)

Ummabot

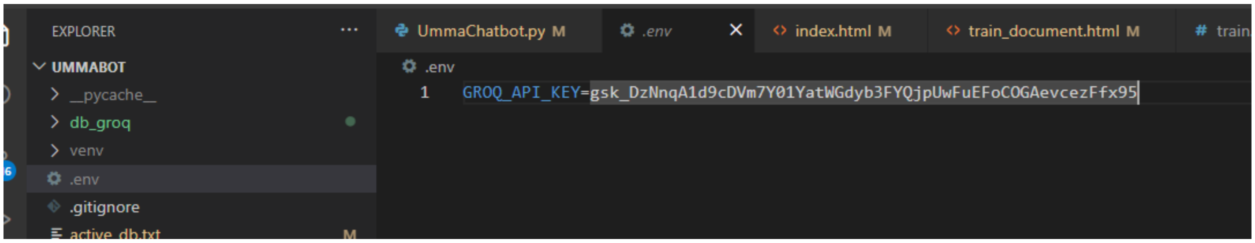


Create API Key

Your new API key has been created. **Copy it now, as we will not display it again.**

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

5. Paste it to the env file as the api key



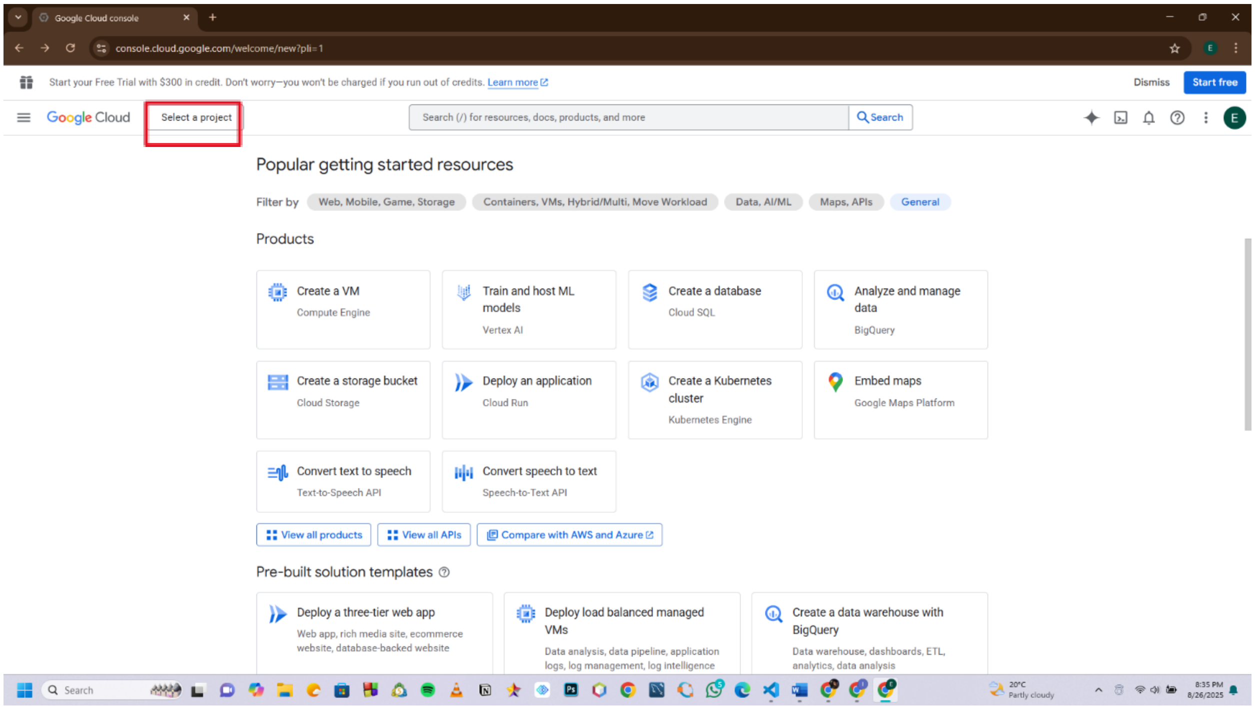
```
GROQ_API_KEY=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
```

4) Google Sheets Setup (Logging)

1. Create a Google Cloud Project

I. Go to Google Cloud Console: https://console.cloud.google.com/

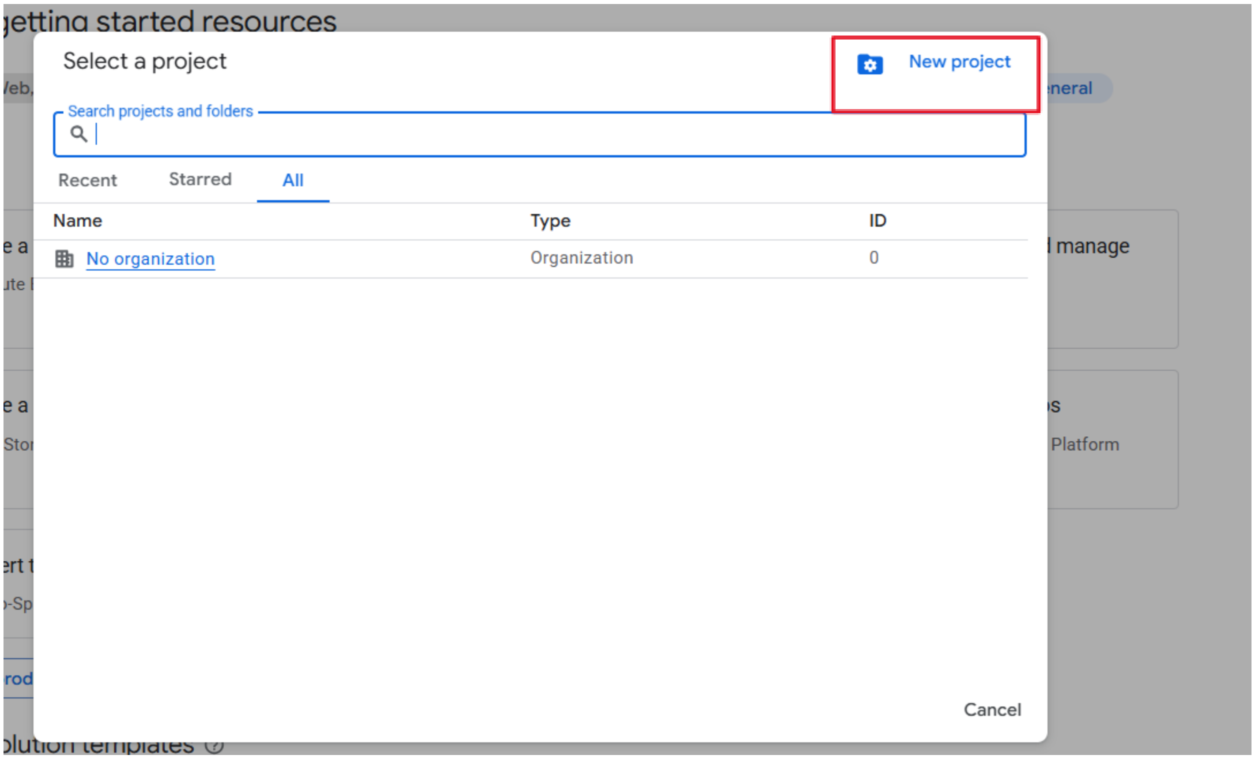
II. At the top, click the project dropdown (next to the Google Cloud logo).



Start your Free Trial with $300 in credit. Don't worry—you won't be charged if you run out of credits. Learn more

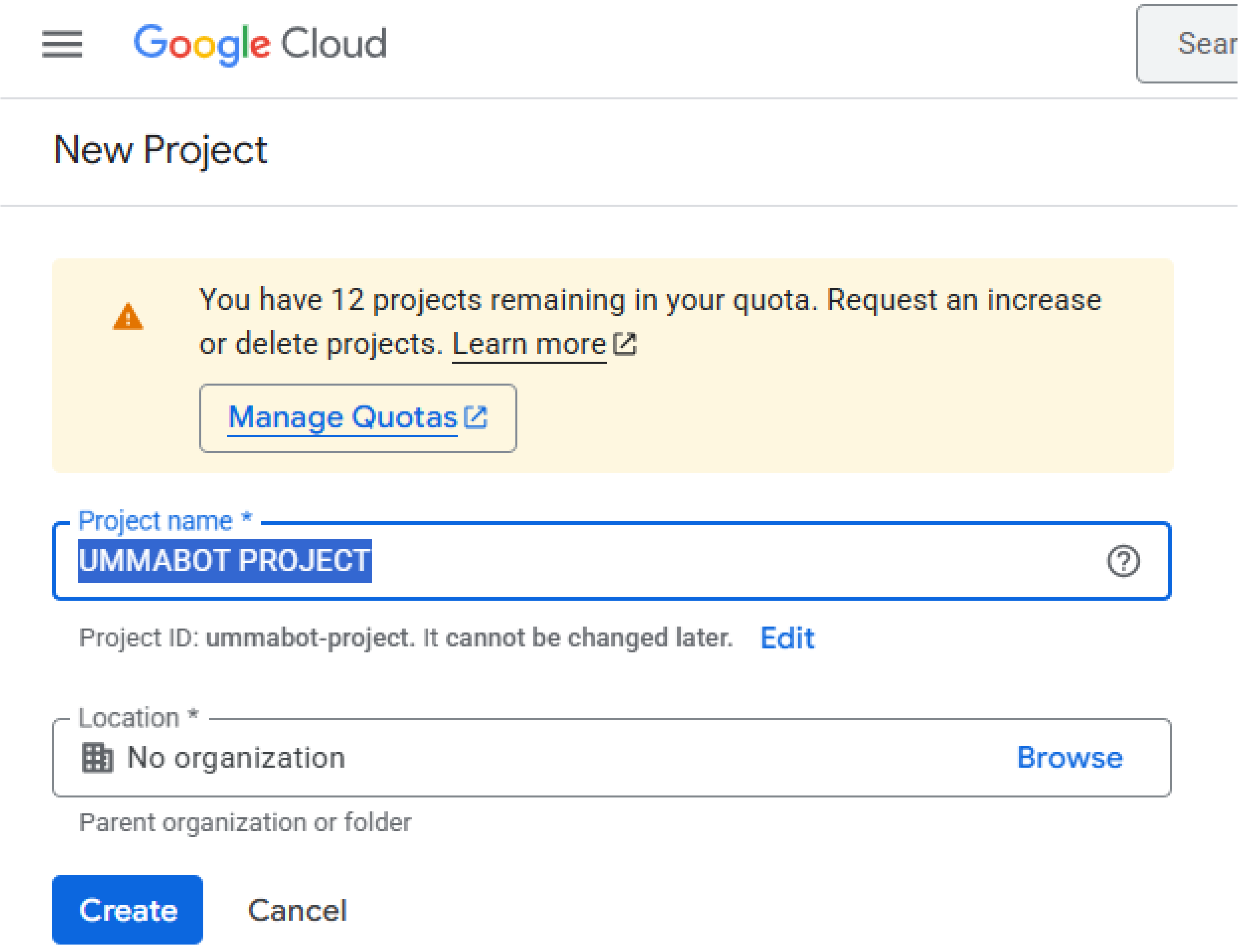
Google Cloud console
console.cloud.google.com/welcome/new?pli=1
Dismiss
Start free
Google Cloud
Select a project
Search (/) for resources, docs, products, and more
Search
Popular getting started resources
Filter by
Web, Mobile, Game, Storage
Containers, VMs, Hybrid/Multi, Move Workload
Data, AI/ML
Maps, APIs
General
Products
Create a VM
Compute Engine
Train and host ML models
Vertex AI
Create a database
Cloud SQL
Analyze and manage data
BigQuery
Create a storage bucket
Cloud Storage
Deploy an application
Cloud Run
Create a Kubernetes cluster
Kubernetes Engine
Embed maps
Google Maps Platform
Convert text to speech
Text-to-Speech API
Convert speech to text
Speech-to-Text API
View all products
View all APIs
Compare with AWS and Azure
Pre-built solution templates
Deploy a three-tier web app
Web app, rich media site, ecommerce website, database-backed website
Deploy load balanced managed VMs
Data analysis, data pipeline, application logs, log management, log intelligence
Create a data warehouse with BigQuery
Data warehouse, dashboards, ETL, analytics, data analysis
Search
20°C
Partly cloudy
8:35 PM
8/26/2025

III. Click **New Project**.



Name	Type	ID
No organization	Organization	0

IV. Give your project a name (e.g., `UmmaBot Project`).



New Project

You have 12 projects remaining in your quota. Request an increase or delete projects. Learn more

Project name *: UMMABOT PROJECT

Project ID: ummabot-project. It cannot be changed later. Edit

Location *: No organization Browse

Parent organization or folder

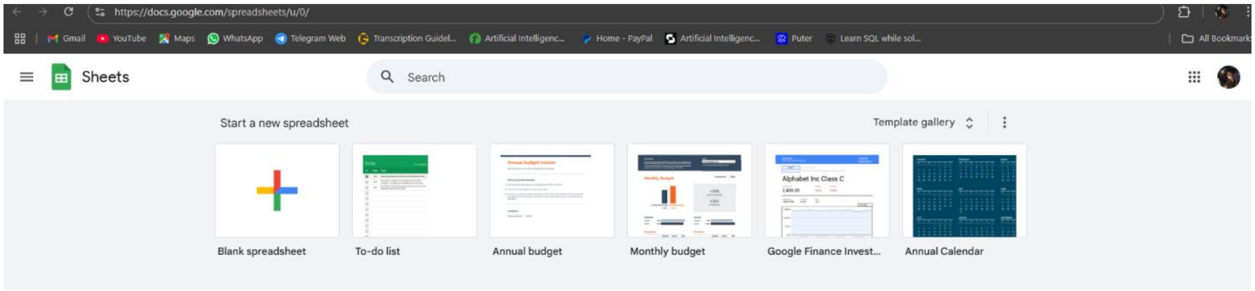
V. (Optional) Select your organization or leave it as "No organization" if personal.

VI. Click **Create**.

VII. Once created, make sure it is **selected** in the project dropdown before continuing.

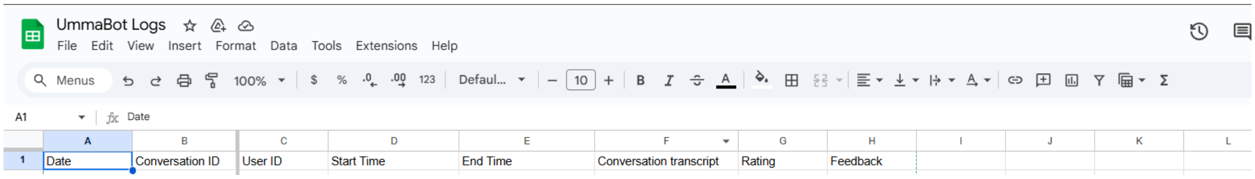
2. Visit https://docs.google.com/spreadsheets/u/0/

   Create a spreadsheet named exactly: **UmmaBot Logs**.



Add the following columns :

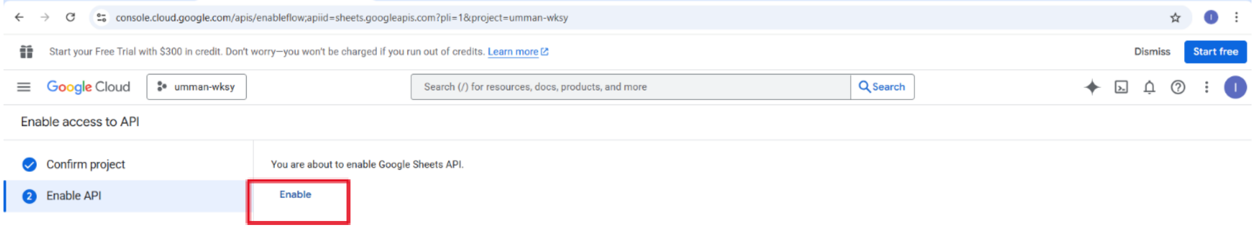
Date, ConversationID, UserID, Start time, End time, Conversation Transcript, Rating, Feedback



	A	B	C	D	E	F	G	H	I	J	K	L
1	Date	Conversation ID	User ID	Start Time	End Time	Conversation transcript	Rating	Feedback				

3. **Enable APIs** for your Google Cloud project:
   a. Google Sheets API

https://console.cloud.google.com/flows/enableapi?apiid=sheets.googleapis.com


console.cloud.google.com/apis/enableflow;apiid=sheets.googleapis.com?pli=1&project=umman-wksy

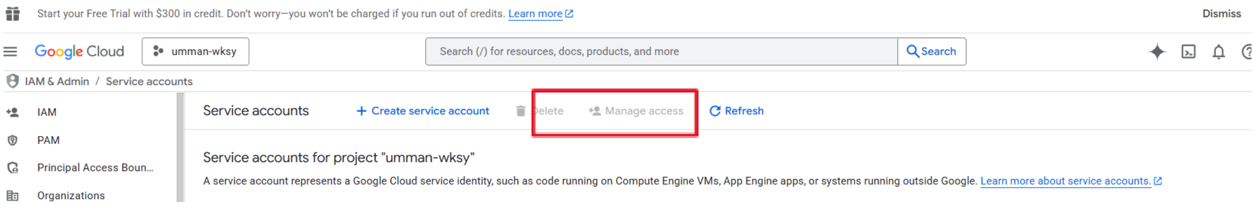
Start your Free Trial with $300 in credit. Don't worry—you won't be charged if you run out of credits. Learn more

Enable access to API

You are about to enable Google Sheets API.

4. **Create a Service Account** and download its **JSON key** file.

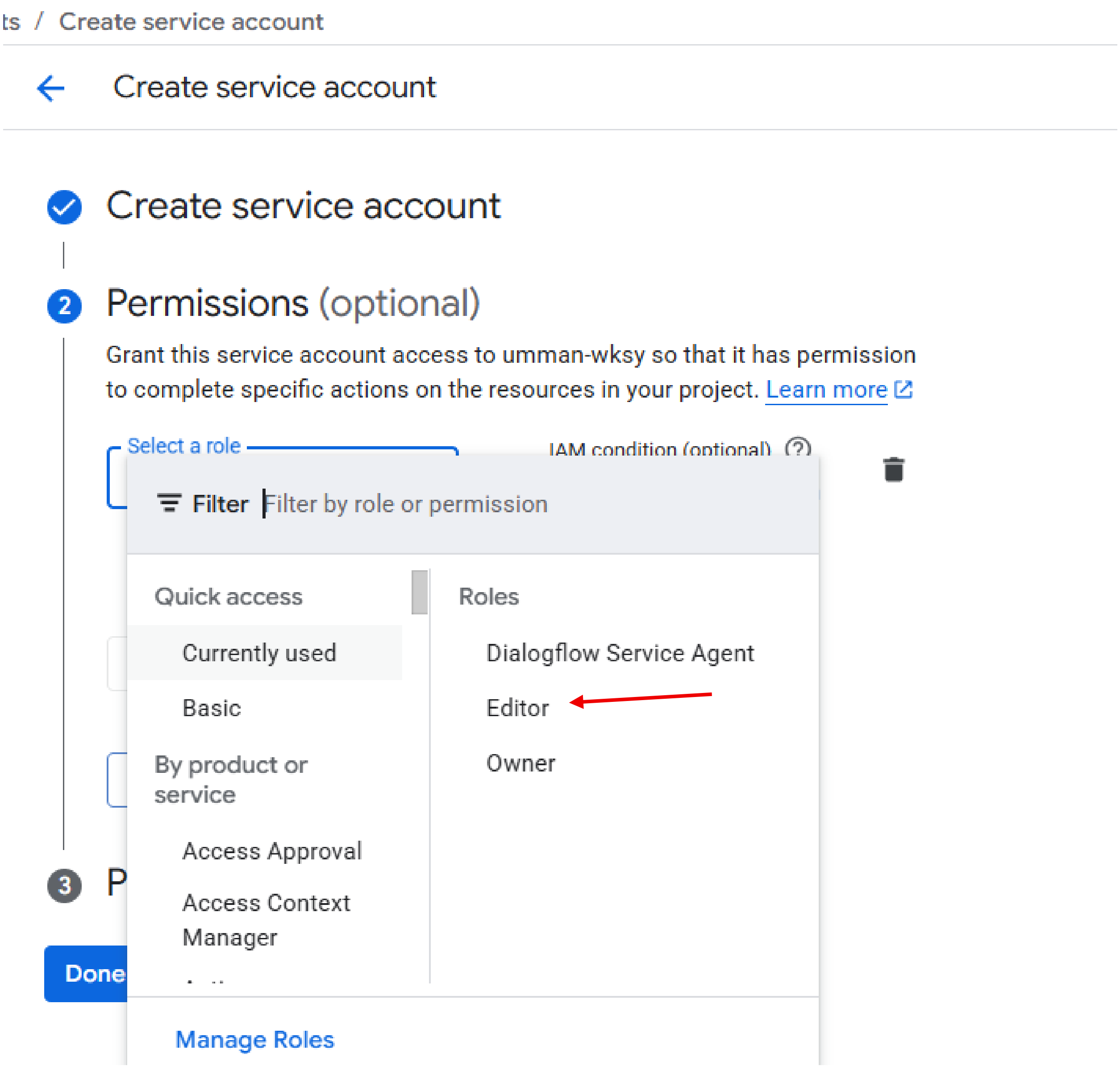
   I. Go to Service Accounts page: https://console.cloud.google.com/iam-admin/serviceaccounts
   II. Click **Create Service Account**.
   III. Assign a name (e.g., UmmaBot Service).


Start your Free Trial with $300 in credit. Don't worry—you won't be charged if you run out of credits. Learn more

Service accounts for project "umman-wksy"

A service account represents a Google Cloud service identity, such as code running on Compute Engine VMs, App Engine apps, or systems running outside Google. Learn more about service accounts.

   IV. Grant role: **Editor** (or at least Sheets/Drive access).



ts / Create service account

Create service account

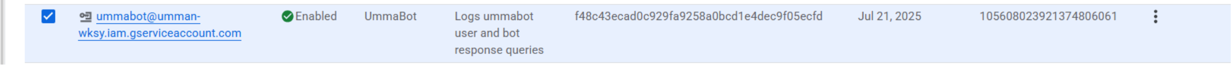
✓ Create service account

2 Permissions (optional)

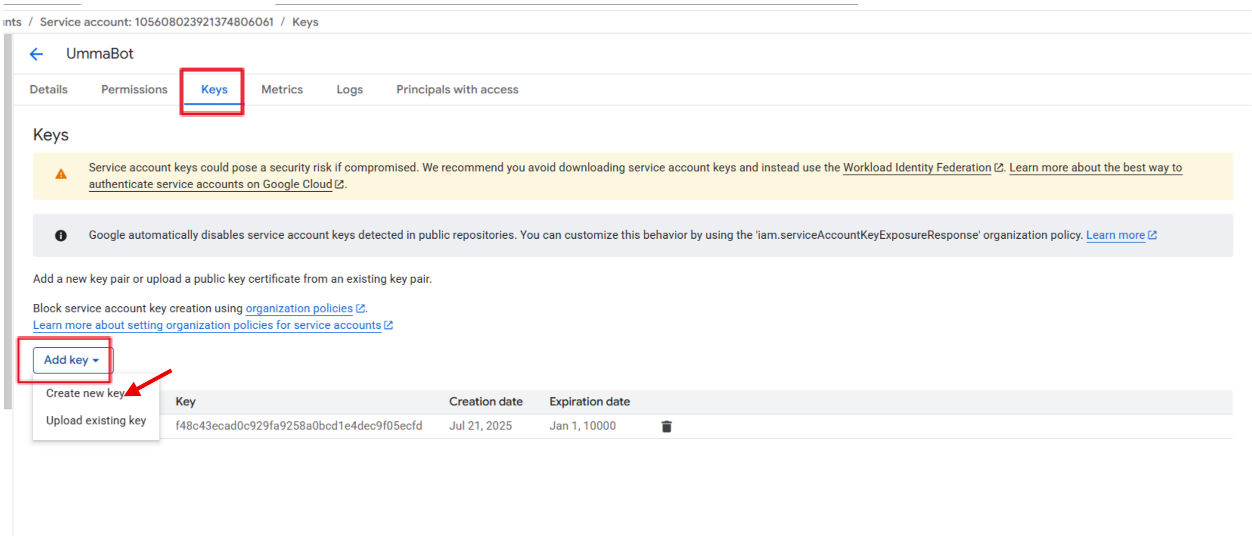
Grant this service account access to umman-wksy so that it has permission to complete specific actions on the resources in your project. Learn more

V. After creation, click the service account → **Keys** tab → **Add Key** → **Create new key** → **JSON**. Select key type as JSON

VI. This downloads your JSON key file.



☑	ummabot@umman-wksy.iam.gserviceaccount.com	Enabled	UmmaBot	Logs ummabot user and bot response queries	f48c43ecad0c929fa9258a0bcd1e4dec9f05ecfd	Jul 21, 2025	105608023921374806061	⋮



Keys

Service account keys could pose a security risk if compromised. We recommend you avoid downloading service account keys and instead use the Workload Identity Federation. Learn more about the best way to authenticate service accounts on Google Cloud.

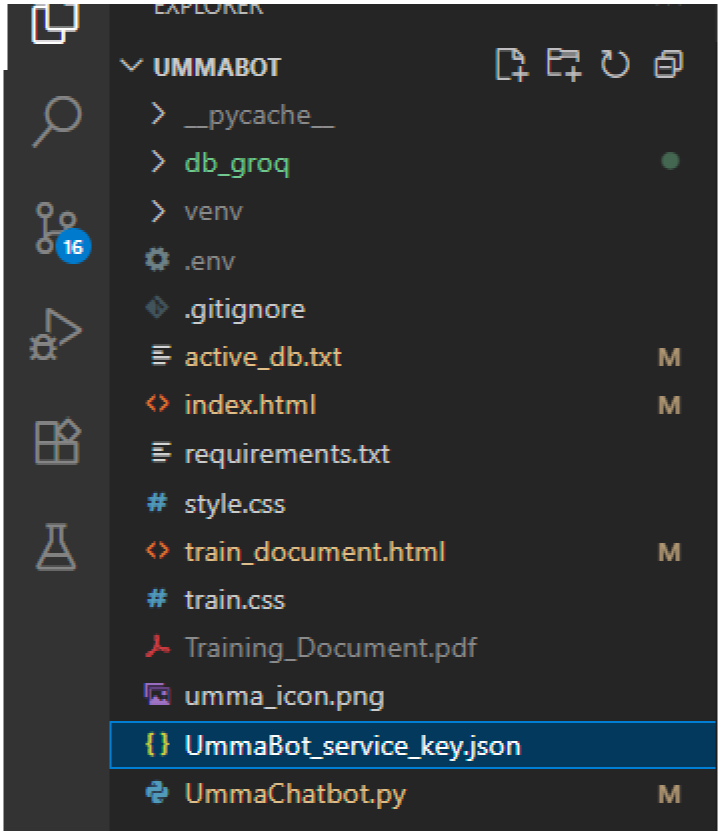
Google automatically disables service account keys detected in public repositories. You can customize this behavior by using the 'iam.serviceAccountKeyExposureResponse' organization policy. Learn more

Add a new key pair or upload a public key certificate from an existing key pair.

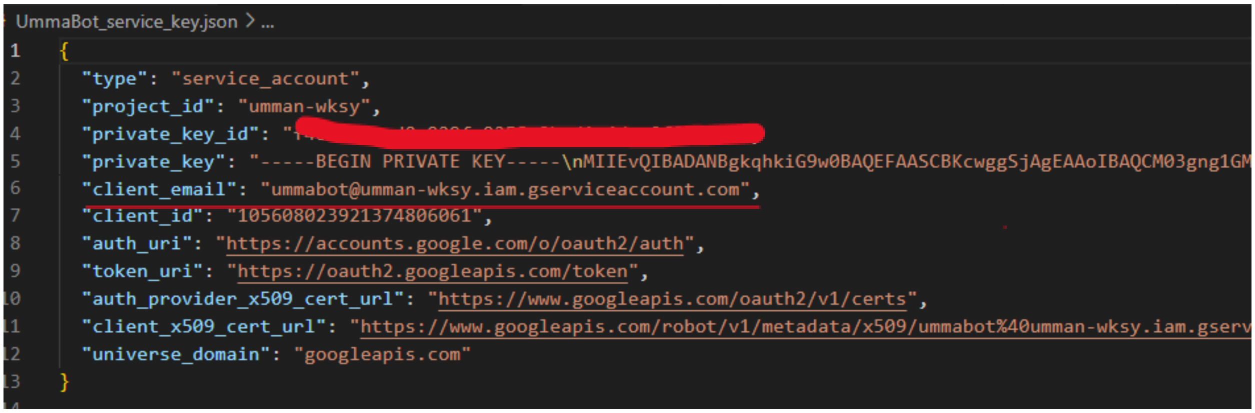
Block service account key creation using organization policies.
Learn more about setting organization policies for service accounts

Key	Creation date	Expiration date
f48c43ecad0c929fa9258a0bcd1e4dec9f05ecfd	Jul 21, 2025	Jan 1, 10000

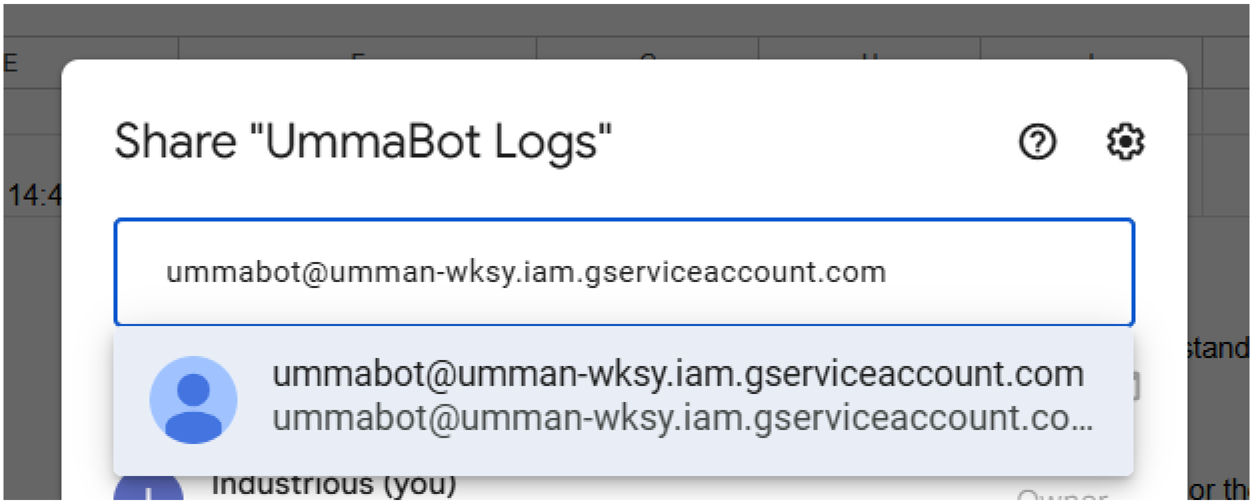
5. Place and rename the JSON file to the project root as:UmmaBot_service_key.json



6. **Share the spreadsheet** with the **service account email** (found in the JSON) as **Editor**.



```
{
  "type": "service_account",
  "project_id": "umman-wksy",
  "private_key_id": "[redacted]
  "private_key": "-----BEGIN PRIVATE KEY-----\nMIIEvQIBADANBgkqhkiG9w0BAQEFAASCBKcwggSjAgEAAoIBAQCM03gng1GM
  "client_email": "ummabot@umman-wksy.iam.gserviceaccount.com",
  "client_id": "105608023921374806061",
  "auth_uri": "https://accounts.google.com/o/oauth2/auth",
  "token_uri": "https://oauth2.googleapis.com/token",
  "auth_provider_x509_cert_url": "https://www.googleapis.com/oauth2/v1/certs",
  "client_x509_cert_url": "https://www.googleapis.com/robot/v1/metadata/x509/ummabot%40umman-wksy.iam.gserv
  "universe_domain": "googleapis.com"
}
```


Share "UmmaBot Logs"

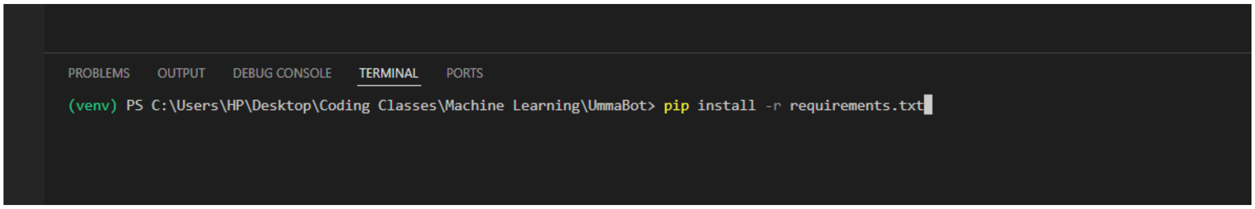
ummabot@umman-wksy.iam.gserviceaccount.com

ummabot@umman-wksy.iam.gserviceaccount.com
ummabot@umman-wksy.iam.gserviceaccount.co...

5) Install Dependencies

Install all the dependencies required by the project stored in the requirements.txt file with the command

------- pip install -r requirements.txt -------



```
(venv) PS C:\Users\HP\Desktop\Coding Classes\Machine Learning\UmmaBot> pip install -r requirements.txt
```

6) Backend Configuration

File locations the code expects

- **Service key**: UmmaBot_service_key.json in the project root
- **Active DB pointer**: active_db.txt (project root). Created automatically on first upload.
- **Vector DBs**: db_groq/<uuid>/ (auto-created).

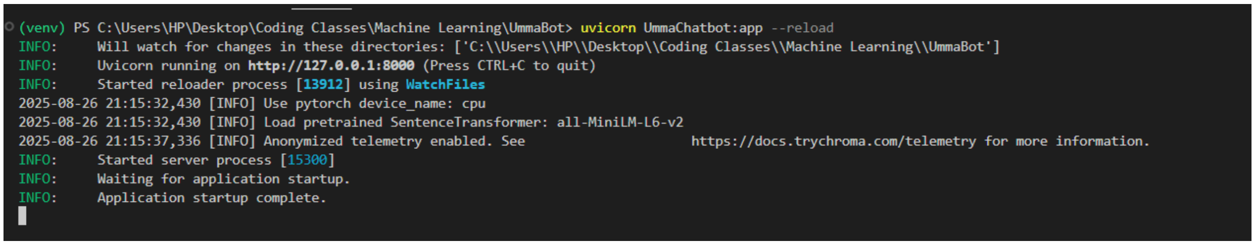
Rate limiting & sessions (already built-in)

- 30 requests per 5 minutes per user_id
- 30-minute session timeout

7) Start the Backend

Start the backend with the command (inside the project folder)

_____ uvicorn UmmaChatbot:app --reload --host 0.0.0.0 --port 8000 ______



```
(venv) PS C:\Users\HP\Desktop\Coding Classes\Machine Learning\UmmaBot> uvicorn UmmaChatbot:app --reload
INFO:     Will watch for changes in these directories: ['C:\\Users\\HP\\Desktop\\Coding Classes\\Machine Learning\\UmmaBot']
INFO:     Uvicorn running on http://127.0.0.1:8000 (Press CTRL+C to quit)
INFO:     Started reloader process [13912] using WatchFiles
2025-08-26 21:15:32,430 [INFO] Use pytorch device_name: cpu
2025-08-26 21:15:32,430 [INFO] Load pretrained SentenceTransformer: all-MiniLM-L6-v2
2025-08-26 21:15:37,336 [INFO] Anonymized telemetry enabled. See                 https://docs.trychroma.com/telemetry for more information.
INFO:     Started server process [15300]
INFO:     Waiting for application startup.
INFO:     Application startup complete.
```

Health check:

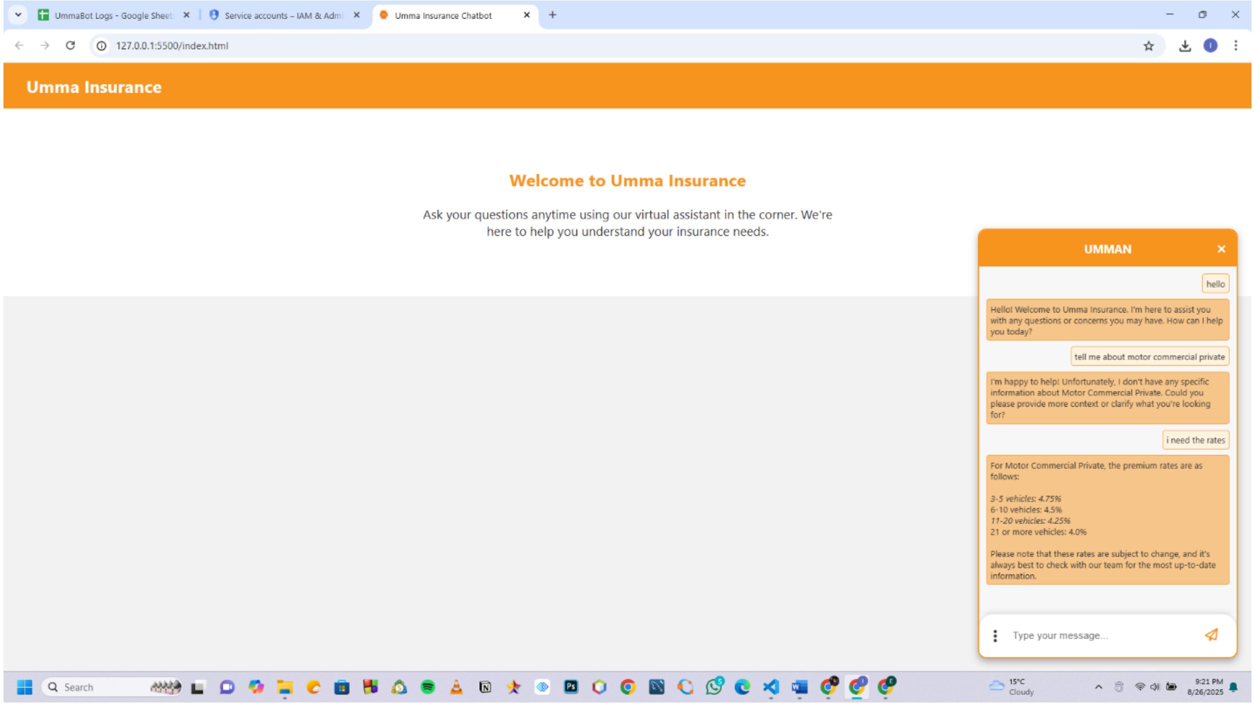
GET http://localhost:8000/ping

→ {"status":"Up and running"}

8) Frontend Configuration

There are two frontend pages:

- **index.html**- which has a demo website for hosting the chat widget


UmmaBot Logs - Google Sheet
Service accounts – IAM & Adm
Umma Insurance Chatbot
127.0.0.1:5500/index.html
15°C
Cloudy
9:21 PM
8/26/2025

Umma Insurance

Welcome to Umma Insurance

Ask your questions anytime using our virtual assistant in the corner. We're here to help you understand your insurance needs.

UMMAN

hello

Hello! Welcome to Umma Insurance. I'm here to assist you with any questions or concerns you may have. How can I help you today?

tell me about motor commercial private

I'm happy to help! Unfortunately, I don't have any specific information about Motor Commercial Private. Could you please provide more context or clarify what you're looking for?

i need the rates

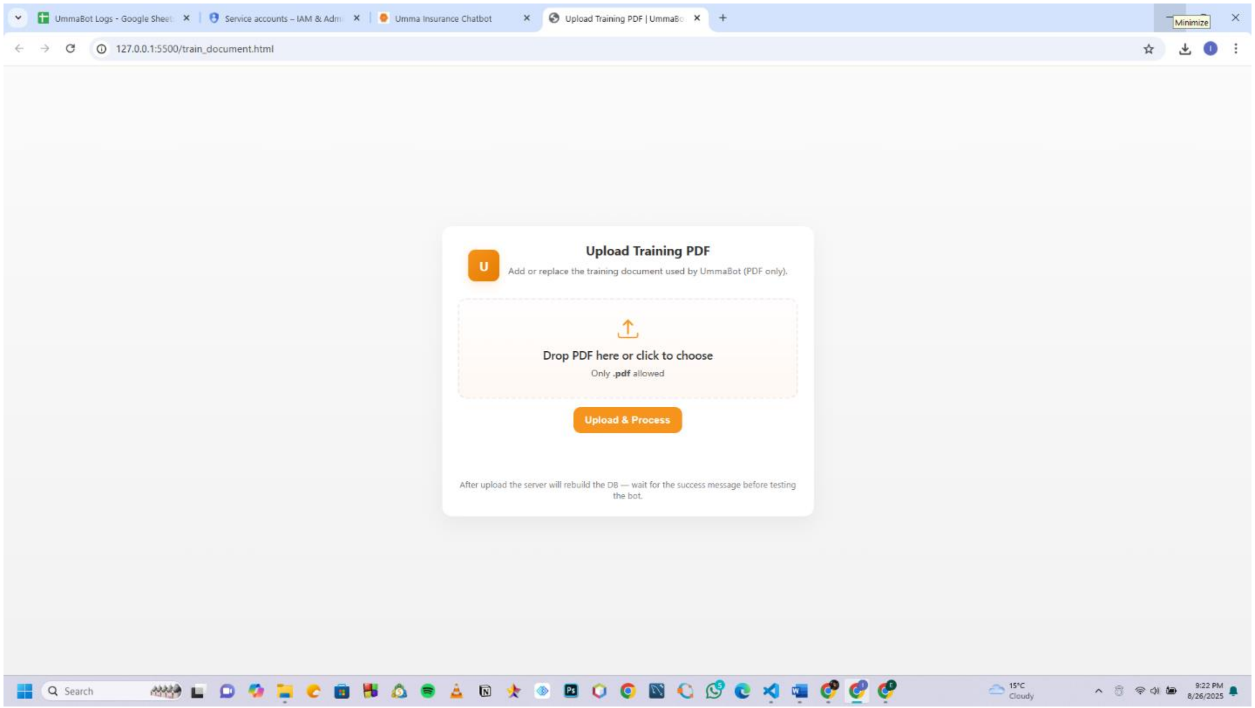
For Motor Commercial Private, the premium rates are as follows:

3-5 vehicles: 4.75%
6-10 vehicles: 4.5%
11-20 vehicles: 4.25%
21 or more vehicles: 4.0%

Please note that these rates are subject to change, and it's always best to check with our team for the most up-to-date information.

Type your message...

- **train_document.html-** resides on the admin side for new training material for the bot.


UmmaBot Logs - Google Sheet
Service accounts – IAM & Adm
Umma Insurance Chatbot
Upload Training PDF | UmmaBo
Minimize
127.0.0.1:5500/train_document.html
U
15°C
Cloudy
9:22 PM
8/26/2025

Upload Training PDF

Add or replace the training document used by UmmaBot (PDF only).

Drop PDF here or click to choose

Only **.pdf** allowed

Upload & Process

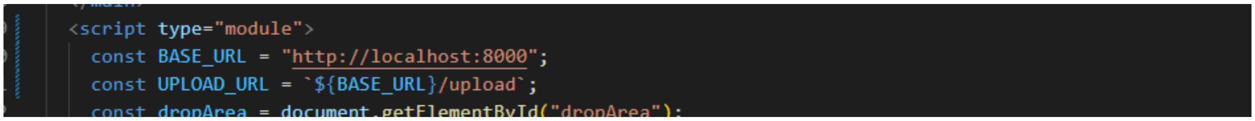
After upload the server will rebuild the DB — wait for the success message before testing the bot.

10)Base URL

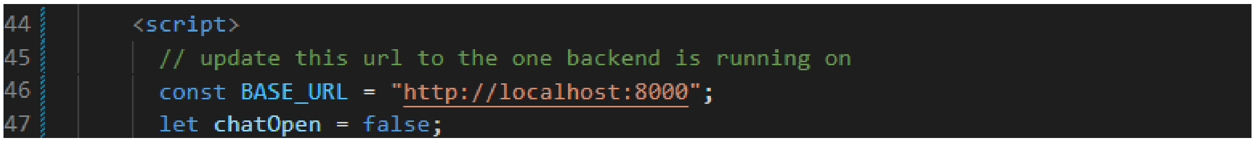
Set the backend URL once:

Iside the html files update the base url to the port the backend is running on

const BASE_URL = "http://localhost:8000";



```
  <script type="module">
    const BASE_URL = "http://localhost:8000";
    const UPLOAD_URL = `${BASE_URL}/upload`;
    const dropArea = document.getElementById("dropArea");
```



```
      <script>
        // update this url to the one backend is running on
        const BASE_URL = "http://localhost:8000";
        let chatOpen = false;
```

10) CORS (Very Important)

The backend currently allows all origins:

```
app.add_middleware(
    CORSMiddleware,
    allow_origins=["*"],  # permissive
    allow_credentials=True,
    allow_methods=["*"],
    allow_headers=["*"],
)
```

Restrict this for production to your exact frontend origin(the umma base url)

```
app.add_middleware(
    CORSMiddleware,
    allow_origins=[“https://www.ummainsurance.com/”],
    allow_credentials=True,
    allow_methods=["POST", "GET", "OPTIONS"],
    allow_headers=["Content-Type", "Authorization"],
)
```

11) Using the App

A) Chatting

The chat widget posts to:

POST {BASE_URL}/chat

Body: { "question": string, "user_id": string }

- The backend manages per-user session history and returns answers.
- If the backend returns "__SHOW_FEEDBACK__", the frontend pops the feedback UI.

B) Feedback

POST {BASE_URL}/feedback

Body: { "user_id": string, "rating": string, "feedback": string, "final": boolean }

- First send a **rating only** with final: false (optional).
- Then send the **final submission** with final: true (with/without a comment).
- Successful submissions append to the Google Sheet.

12) Training / Retraining the Bot

1. Open the train.html.
2. Set BASE_URL at the top of the script to the backend.
3. Drag/drop or select a PDF and click Upload & Process.
4. The backend:
   - Parses and chunks the PDF
   - Builds a new Chroma DB in a unique folder db_groq/<uuid>/
   - Updates active_db.txt to point to the new DB
   - Re-initializes QA chains so the new content is immediately live
5. You'll see a success message:
   "Training document processed successfully."

13) Endpoint Reference

- POST /chat
  Req: {question, user_id}
  Resp: {response: string | "__SHOW_FEEDBACK__"}
- POST /feedback
  Req: {user_id, rating, feedback, final}
  Resp: status JSON
- POST /upload
  Form: file (PDF only)
  Resp: status JSON (Active DB: db_groq/<uuid>)
- POST /admin/reload_db
  Reloads DB from active_db.txt.
- GET /ping
  Health check.

14) Troubleshooting

A) CORS errors in browser console

- Symptom: “CORS policy” blocked.
- Fix: In FastAPI CORS, set allow_origins to the frontend’s exact URL (including scheme). Allow OPTIONS.

B) Google Sheets isn’t logging

- Ensure spreadsheet name **exactly** UmmaBot Logs.
- Share it with the **service account email** as **Editor**.
- Confirm UmmaBot_service_key.json is in the project root.
- Check server logs for “Google Sheets Initialization failed”.

C) 404/Network errors from frontend

- Verify BASE_URL points to the backend.
- Check the backend is running and accessible from the browser.