

Step 1: Create virtual environment

```
python3.11 -m venv venv  
source venv/bin/activate
```

Step 2: install requirements.txt

```
pip install -r requirements.txt or pip3 install -r requirements.txt
```

To run the frontend

To run the project (secret configuration must be done before this, follow steps below):

Frontend commands

Go to the git repo:

```
streamlit run frontend/streamlit_app.py
```

Or

From the frontend folder

```
streamlit run frontend/streamlit_app.py
```

Backend commands

Go to backend folder:

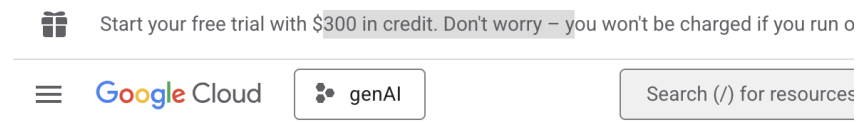
```
uvicorn app:app --reload
```

Setup for google client:

Google cloud:

<https://console.cloud.google.com/welcome?project=genai-456014>

The project should be yours (select from top left)



Click on API and services → Credentials

Start your free trial with \$300 in credit. Don't worry – you won't be charged if you run out of credit. [Learn more](#)

Google Cloud genAI Search (/) for resources, docs, products and more

Welcome

You're working in **genAI**

Project number: 185111628205 Project ID: genai-456014

[Dashboard](#) [Cloud Hub](#) [New](#)

[+ Create a VM](#) [+ Run a query in BigQuery](#) [+ Deploy an app](#)

[+ Create a storage bucket](#)

Quick access

[APIs and services](#) [IAM and admin](#)

Select oauth client ID

Start your free trial with \$300 in credit. Don't worry – you won't be charged if you run out of credit. [Learn more](#)

Google Cloud genAI Search (/) for resources, docs, products and more

APIs and services Credentials [+ Create credentials](#) [Delete](#) [Restore deleted credentials](#)

- Enabled APIs and services
- Library
- Credentials**
- OAuth consent screen
- Page usage agreements

Create credentials to a

Remember

API keys

☐ ☒ **Name**

No API keys to display

OAuth 2.0 Client IDs

API key
Identifies your project using a simple API key to check quota and access

OAuth client ID
Requests user consent so that your app can access the user's data.

Service account
Enables server-to-server, app-level authentication using robot accounts

Help me choose
Asks a few questions to help you decide which type of credential to use

After filling out the form, click on create oauth client below

Metrics

You haven't configured any OAuth clients for this project yet. [Create OAuth client](#)

Fill out these details below:

← Create OAuth client ID

Name *

Web client 1

The name of your OAuth 2.0 client. This name is only used to identify the client in the console and will not be shown to end users.



The domains of the URIs you add below will be automatically added to your [OAuth consent screen](#) as [authorised domains](#).

Authorised JavaScript origins ?

For use with requests from a browser

+ Add URI

Authorised redirect URIs ?

For use with requests from a web server

URIs 1 *

http://localhost:8501/oauth2callback

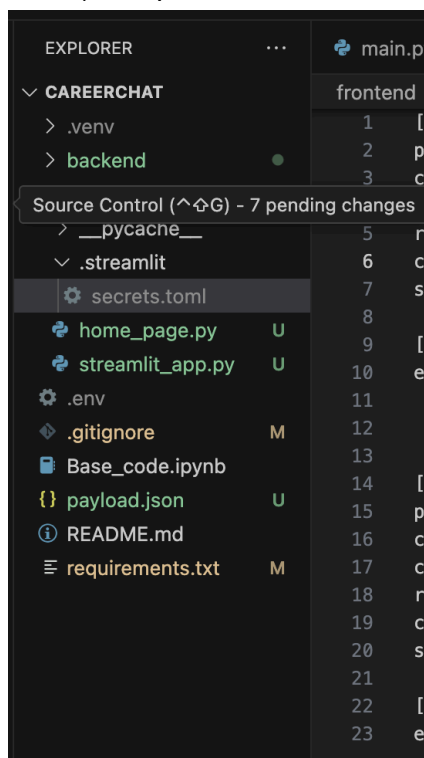
OAuth configuration created.



+ Add URI

After this click on create, you will get the secret and client ID, MAKE SURE TO COPY AND save it somewhere

Now that you have got the details above and saved your creds, Create a secrets.toml file (refer the location below) and paste it in .streamlit folder



Below is the content that you have to paste in the file

[auth]

provider = "google"

client_id = "" -- you have just created this (refer above)

client_secret = "" --you have just created this (refer above)

redirect_uri = "http://localhost:8501/oauth2callback"

cookie_secret = "sjpFEaPTxfqY8VTUGb7LYmmL8oh-MiglIg6CpHmTsPU" -- This one keep as is, might have to figure out later what to add in this

server_metadata_url = "https://accounts.google.com/.well-known/openid-configuration"

[general]

email = "your-email@example.com"