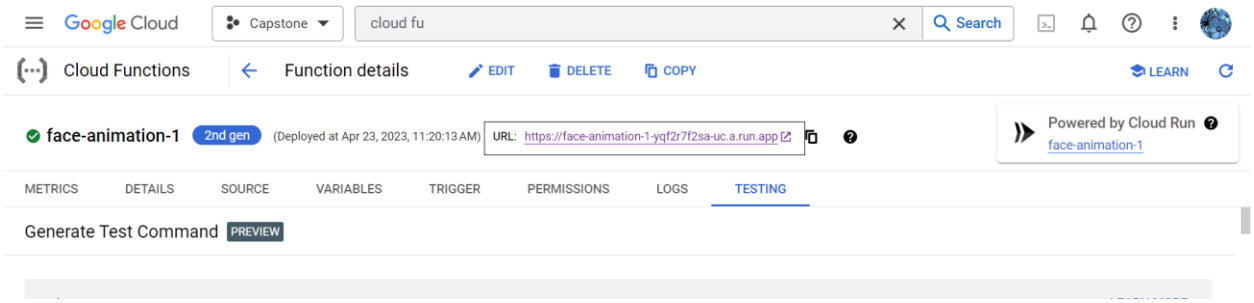
This document details the setup for the automation of the image animation in the Google Cloud Function.

1. Git branch

The branch for image animation is the “imageAnimationFinal” branch. In the ios-app/api/app.py the following lines need to be changed:

```
from joblib import Parallel, delayed
import requests
from google.cloud import storage
def sumall(audio):
    json_data = {
        'text': 'Hello, World!',
        'audio': audio,
        'image': 'photo.png',
        # replace name with the name of the pateint
1       'name': "BananaMonster",
        # replace patientId with Id of patient
2       'patientId': "1",
        # replace fpID with id of FP
3       'fpId': "11"
    }
    headers = {
        'Content-Type': 'application/json',
        # Replace bearer with gcp cloud token from shell
4       'Authorization': 'bearer
    }
5   # replace url below with url of the cloud function url
    response = requests.post( https://face-animation-1-yqf2r7f2sa-uc.a.run.app',headers=headers, json=json_data,timeout=6000 )
    return response
# 6eplace audios1 with the audio files stored in gcp cloud
audios1 = ["How are you doing today", "Today is","Where do you live","Do you remember the time when you lost your first tooth","It is","It is the year twenty twenty-three"]
Parallel(n_jobs=30, verbose=10)(delayed(sumall)(i) for i in audios1)
```

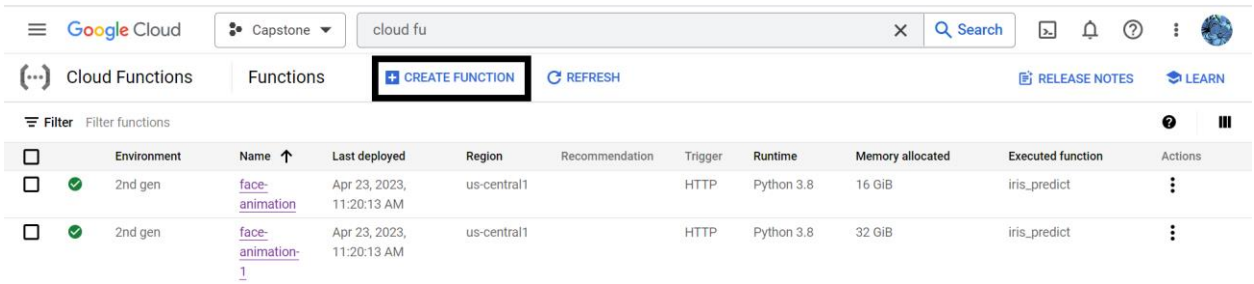
1. The “name” field needs to be replaced with the name of the patient.
2. The “patientId” field needs to be replaced with the ID of the corresponding patient.
3. The “fpId” field needs to be replaced with the ID of the familiar person.
4. The “Authorization” needs to be filled in with the bearer code which is obtained by running “gcloud auth print-identity-token” in the cloud shell.
5. Replace the URL field with the URL from the cloud function as seen below:



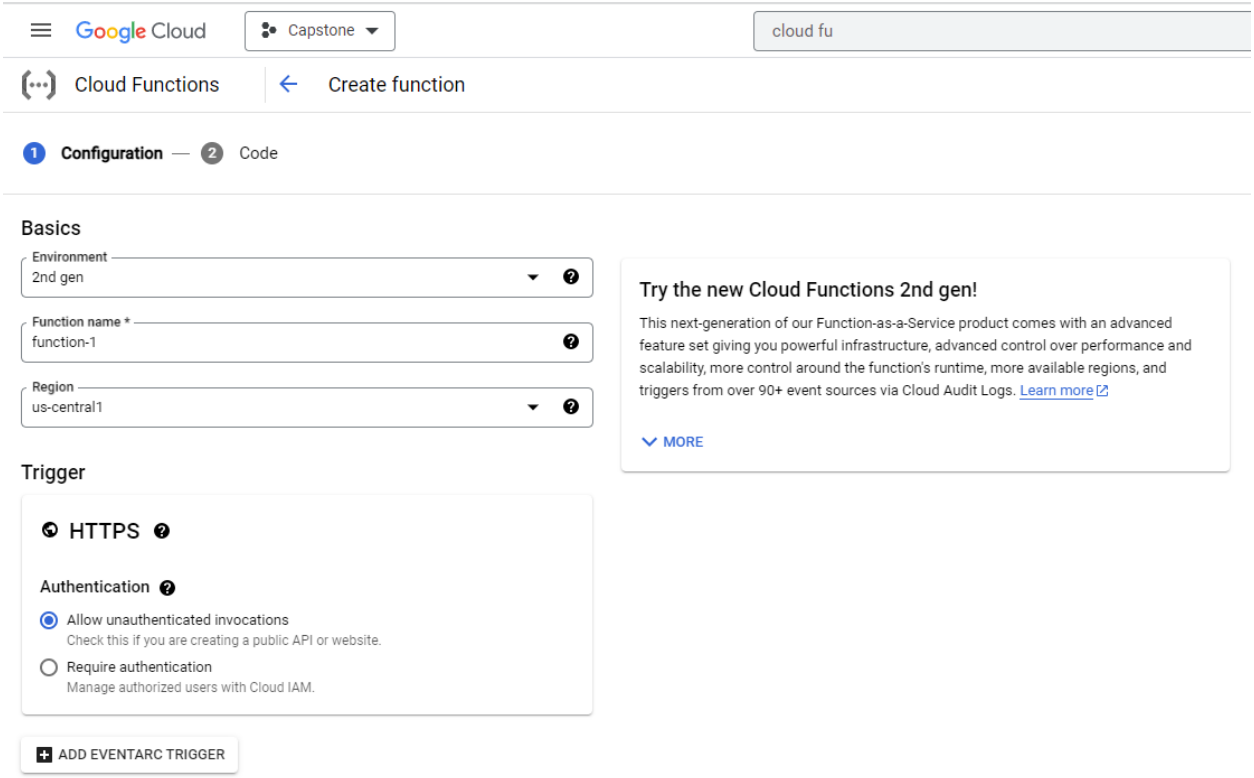
6. In the audio array, store the name of the audio files that you want to animate the familiar person to say.

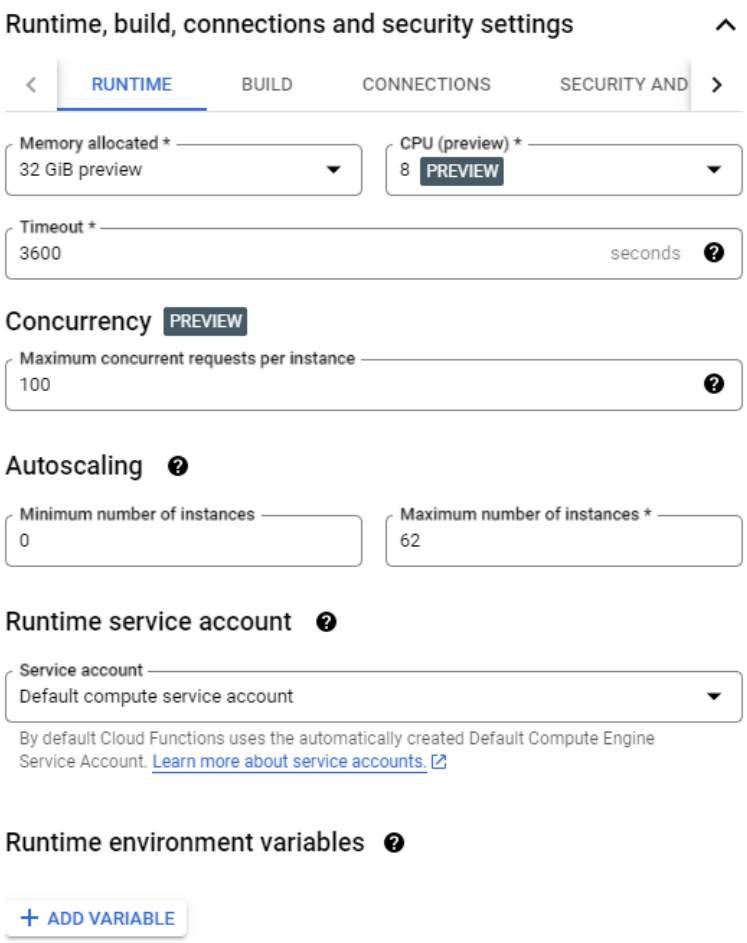
2. Setting up the Google Cloud Function

1. Create a cloud function:

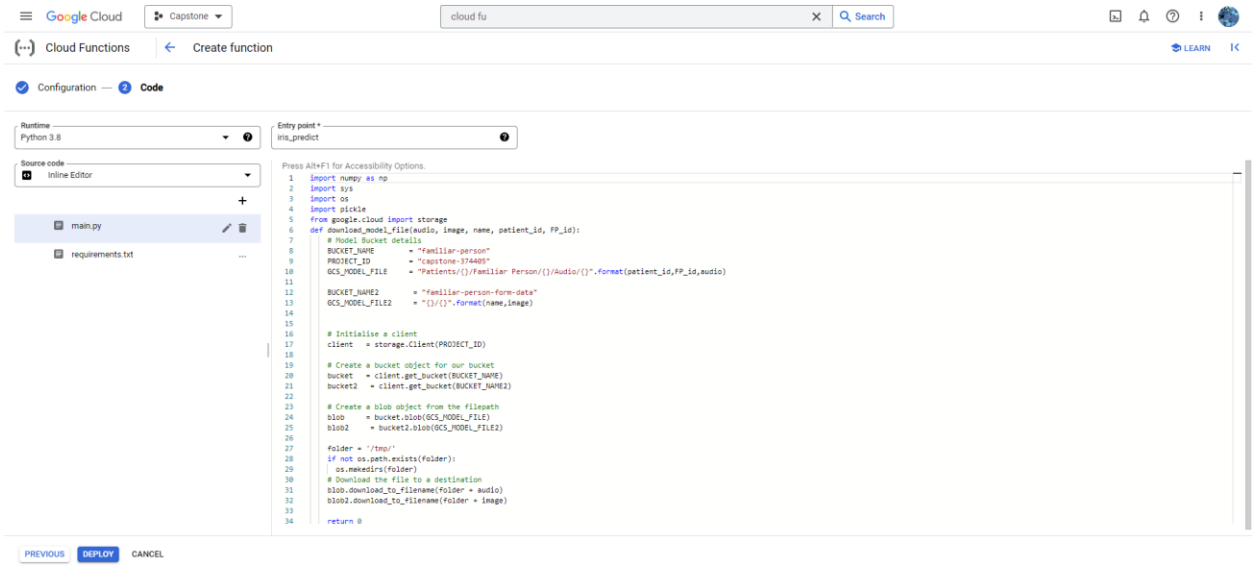


2. Use the following configuration values:





3. Copy in the code from main.py and requirements.txt from ImageAnimation/ directory in this repo into the main.py and requirements.txt file in the cloud function:



4. Deploy the cloud function and test the image animation by creating a new profile through the frontend.