Deployment Manager Guestbook 1

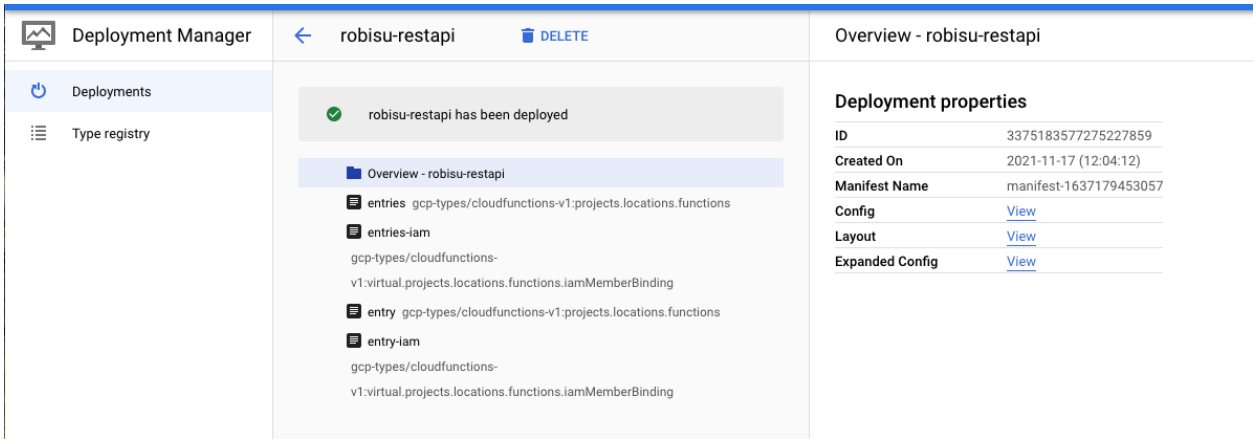
ML APIs 2

Firebase 8

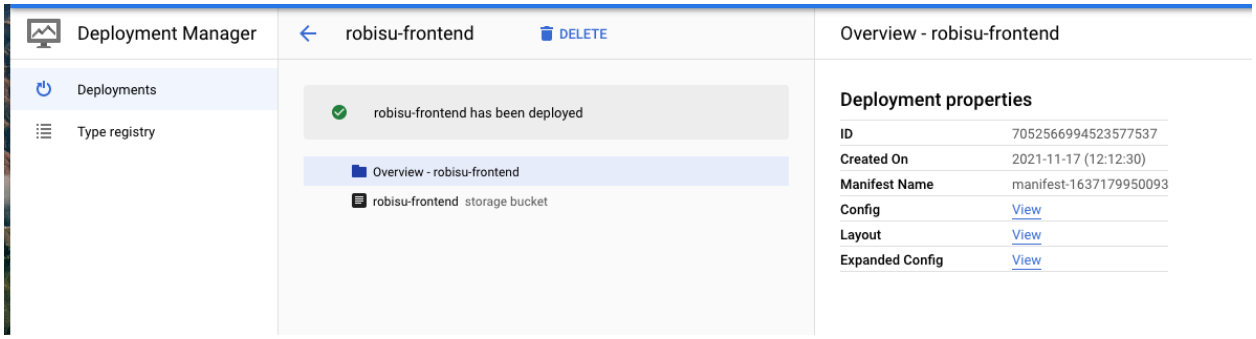
OAuth2 Guestbook 12

Deployment Manager Guestbook

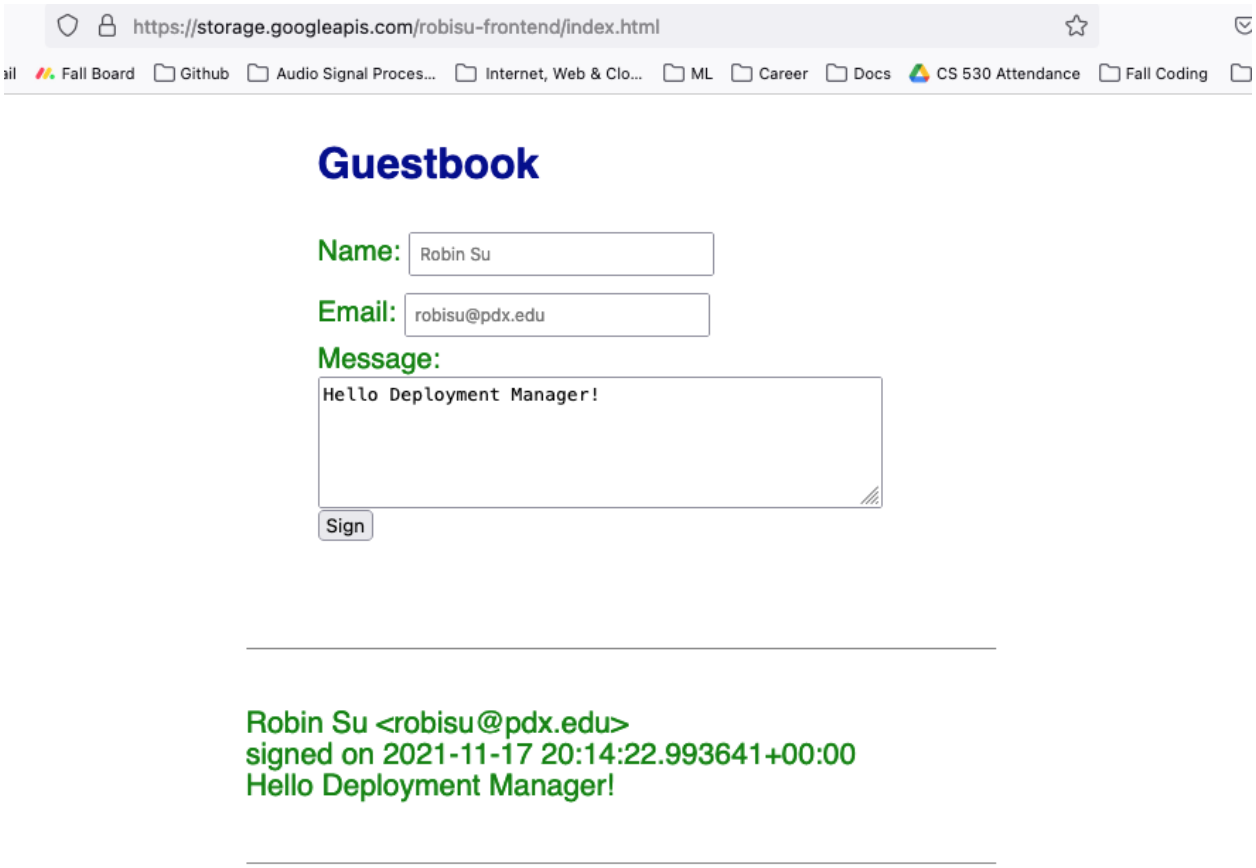
REST API Deployed with Deployment Manager (Resources):



Guestbook Front-end deployed with Deployment Manager



Deployed static site:



ML APIs

detect.py labels output

```
(env) robisu@cloudshell:~/python-vision/samples/snippets/detect (cloud-f21-robin-su-robisu)$ python detec
.py labels-uri gs://ml-api-codelab/birds.jpg
Labels:
Bird
Ratite
Cloud
Sky
Plant
Beak
Green
Neck
Casuariiformes
Ostrich
(env) robisu@cloudshell:~/python-vision/samples/snippets/detect (cloud-f21-robin-su-robisu)$
```

- *What is the name of the function?*
  - **detect_labels_uri**
- *What type of Vision client is instantiated in it?*
  - **ImageAnnotatorClient**
- *What method is invoked in the Vision client to perform the detection?*
  - **label_detection(image=image)**
- *What is the name of the attribute in the response object that contains the results we seek?*
  - **response.label_annotations**

detect.py logo detection output

```
(env) robisu@cloudshell:~/python-vision/samples/snippets/detect (cloud-f21-robin-su-robisu)$ python detec
.py logos logofile.jpg
Logos:
Oberlin College
```

- *What method is invoked in the Vision client to perform the detection?*
  - **logo_detection(image)**

speech-to-text

```
(env) robisu@cloudshell:~/python-speech/samples/snippets (cloud-f21-robin-su-robisu)$ python transcribe.p
 resources/audio.raw
Transcript: how old is the Brooklyn Bridge
```

- *What is the name of the function?*
  - **transcribe_file()**
- *What method is invoked in the Speech client to perform the detection?*
  - **client.recognize(config=config, audio=audio)**
- *What is the name of the attribute in the response object that contains the results we seek?*

- **response.results**

Translate

```
(env) robisu@cloudshell:~/python-translate/samples/snippets (cloud-f21-robin-su-robisu)$ python snippets.
y translate-text en '你有沒有帶外套'
Text: 你有沒有帶外套
Translation: Did you bring a jacket
Detected source language: zh-TW
```

- *What is the name of the function?*
  - **translate_text(target, text)**
- *What method is invoked in the Translate client to perform the detection?*
  - **translate_client.translate(text, target_language=target)**
- *What is the name of the attribute in the response object that contains the results we seek?*
  - **result[“translatedText”]**

Natural language

```
(env) robisu@cloudshell:~ (cloud-f21-robin-su-robisu)$ python language.py 'homework is awful!'
'homework is awful!' has sentiment score of -0.800000011920929
Entity Names Are:
homework
(env) robisu@cloudshell:~ (cloud-f21-robin-su-robisu)$ python language.py 'homework is ok'
'homework is ok' has sentiment score of 0.30000001192092896
Entity Names Are:
homework
(env) robisu@cloudshell:~ (cloud-f21-robin-su-robisu)$ python language.py 'homework is awesome?'
'homework is awesome?' has sentiment score of 0.4000000059604645
Entity Names Are:
homework
(env) robisu@cloudshell:~ (cloud-f21-robin-su-robisu)$ python language.py 'homework is awesome!'
'homework is awesome!' has sentiment score of 0.8999999761581421
Entity Names Are:
homework
```

```
(env) robisu@cloudshell:~ (cloud-f21-robin-su-robisu)$ python language.py 'The protestors in Oregon put o
 gas masks and wore yellow t-shirts'
'The protestors in Oregon put on gas masks and wore yellow t-shirts' has sentiment score of -0.6000000238
18579
Entity Names Are:
protestors
gas masks
Oregon
t-shirts
(env) robisu@cloudshell:~ (cloud-f21-robin-su-robisu)$
```

Compare audio to image

- *What is the name of the function that performs the transcription?*
  - **transcribe_gcs(language, gcs_uri)**
- *What is the name of the function that performs the translation?*
  - **translate_text(target, text)**
- *What is the name of the function that performs the entity analysis on the translation?*
  - **entities_text(text)**
- *What is the name of the function that performs the entity analysis on the image?*
  - **detect_labels_uri(uri)**

python solution.py de-DE gs://ml-api-codelab/de-ball.wav gs://ml-api-codelab/football.jpg

```
(env) robisu@cloudshell:~ (cloud-f21-robin-su-robisu)$ python solution.py de-DE gs://ml-api-cod
lab/de-ball.wav gs://ml-api-codelab/football.jpg
Transcription: willst du mit uns Fußball spielen
Translation: do you want to play soccer with us?
Entities: ['soccer']
Image labels: ['Soccer', 'Sports equipment', 'Plant', 'Football', 'Playing sports', 'Ball', 'Pl
yer', 'Competition event', 'Soccer ball', 'Ball game']
The audio and image do not appear to be related.
(env) robisu@cloudshell:~ (cloud-f21-robin-su-robisu)$
```

The program is checking for exact equivalence in the strings, so the capitalization of “soccer” in the image labels doesn’t match the entity in the text - getting rid of capitalization of the words should fix this.

python solution.py tr-TR gs://ml-api-codelab/tr-bike.wav gs://ml-api-codelab/bicycle.jpg

```
(env) robisu@cloudshell:~ (cloud-f21-robin-su-robisu)$ python solution.py tr-TR gs://ml-api-cod
lab/tr-bike.wav gs://ml-api-codelab/bicycle.jpg
Transcription: bisikletimi sokağa bırak
Translation: leave my bike on the street
Entities: ['bike', 'street']
Image labels: ['Bicycle', 'Clothing', 'Footwear', 'Tire', 'Wheel', 'Bicycles--Equipment and sup
lies', 'Land vehicle', 'Shoe', 'Bicycle frame', 'Bicycle wheel']
The audio and image do not appear to be related.
```

The program is unable to relate ‘bike’ to ‘bicycle’, so it needs to have a more intelligent check for how the two words are related, maybe by taking into account alternative words for ‘bicycle’. Again, the check is too literal in looking for matching strings.

python solution.py tr-TR gs://ml-api-codelab/tr-ostrich.wav gs://ml-api-codelab/birds.jpg

```
(env) robisu@cloudshell:~ (cloud-f21-robin-su-robisu)$ python solution.py tr-TR gs://ml-api-cod
lab/tr-ostrich.wav gs://ml-api-codelab/birds.jpg
Transcription: çok fazla deve kuşu var
Translation: there are too many ostriches
Entities: ['ostriches']
Image labels: ['Bird', 'Ratite', 'Cloud', 'Sky', 'Plant', 'Beak', 'Green', 'Neck', 'Casuariifor
es', 'Ostrich']
The audio and image do not appear to be related.
```

The program misses the singular and plural forms of ‘ostrich’ (as well as capitalization). The check for matches should take the singular/plural forms into account.

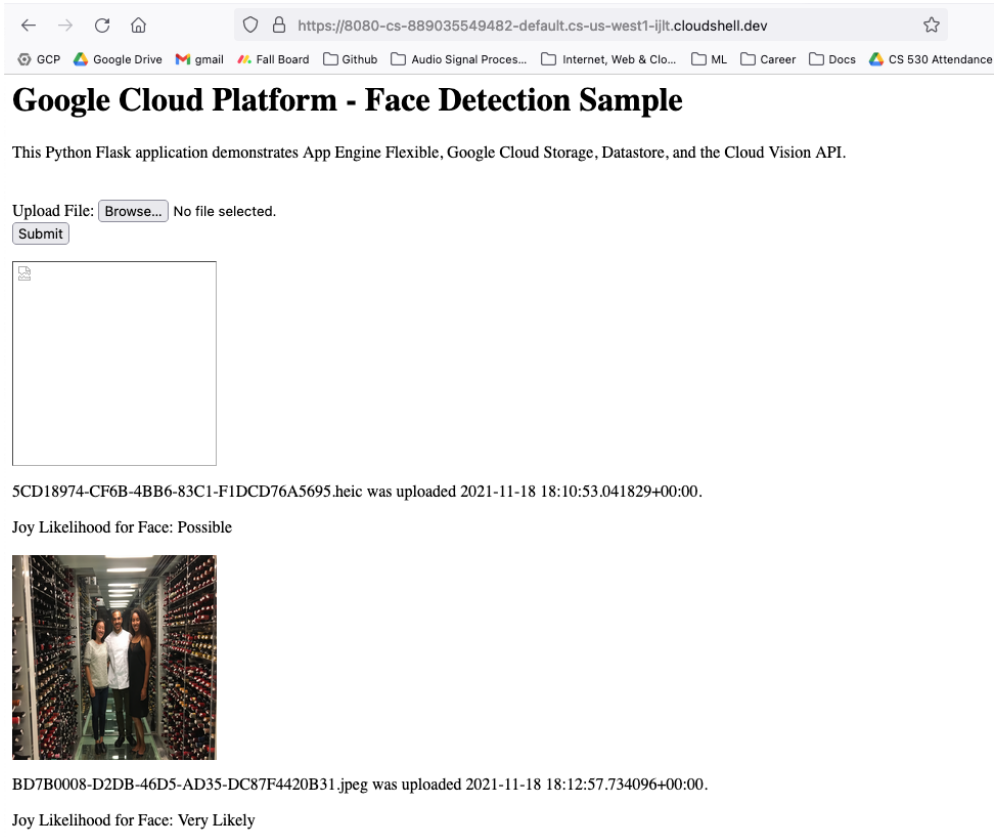
Video Intelligence

- *What are the top 3 labels that the Video Intelligence API associates with the video and what is its confidence in them?*
    - **sports - 0.9218811392784119, basketball - 0.9137870669364929, player - 0.8446521162986755**

Open up labels.py. Answer the following questions:

- *What is the name of the client class in the package that is used?*
    - **VideoIntelligenceServiceClient**
- *What method is used in that class to perform the annotation?*
    - **annotate_video(input_uri=path, features=features)**

Likelihood of Joy:



Open main.py and view the code for the default route. Answer the following questions:

- *What line of code creates the query for previous detections?*

    - **#39**
- *What line of code sends the query to Cloud Datastore?*
    - **#40**

Then, view the upload_photo route.

- *Show the line that retrieves the name of the storage bucket to use.*
    - **#54:** `bucket = storage_client.get_bucket(CLOUD_STORAGE_BUCKET)`
- *What form field is used to specify the uploaded photo?*
    - **photo.filename**
- *Show the line that copies the photo's contents to the storage bucket.*
    - **#58:** `blob.upload_from_string(photo.read(), content_type=photo.content_type)`
- *What method in Vision's annotation client is used to perform the analysis?*
    - `vision_client.face_detection(image=image)`
- *What fields are stored in Cloud Datastore for each image?*
    - **blob_name, image_public_url, timestamp, joy**
- *What happens at the end of the upload_photo route?*
    - **the user is redirected to the default route**

Firebase

- *What other domains are given access to this Firebase project by default?*
    - **localhost, fir-robisu.firebaseapp.com, fir-robisu.web.app**

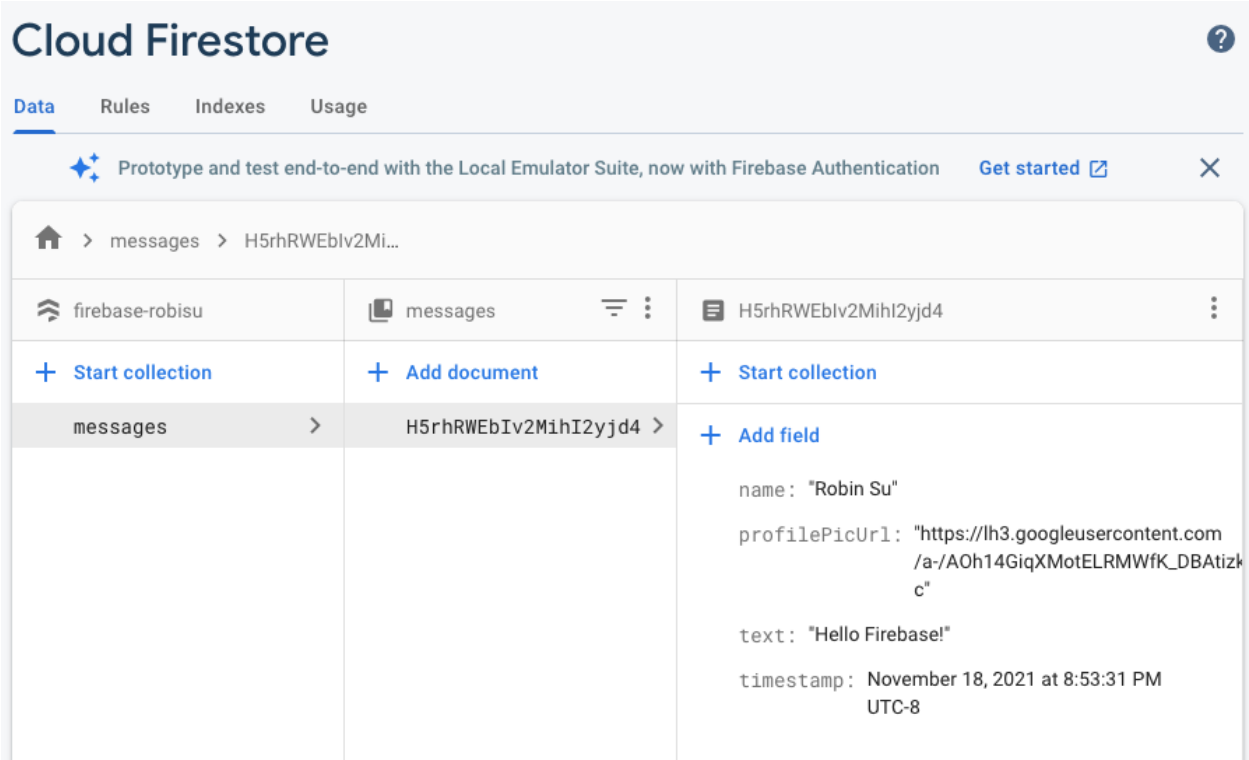
Add Authentication

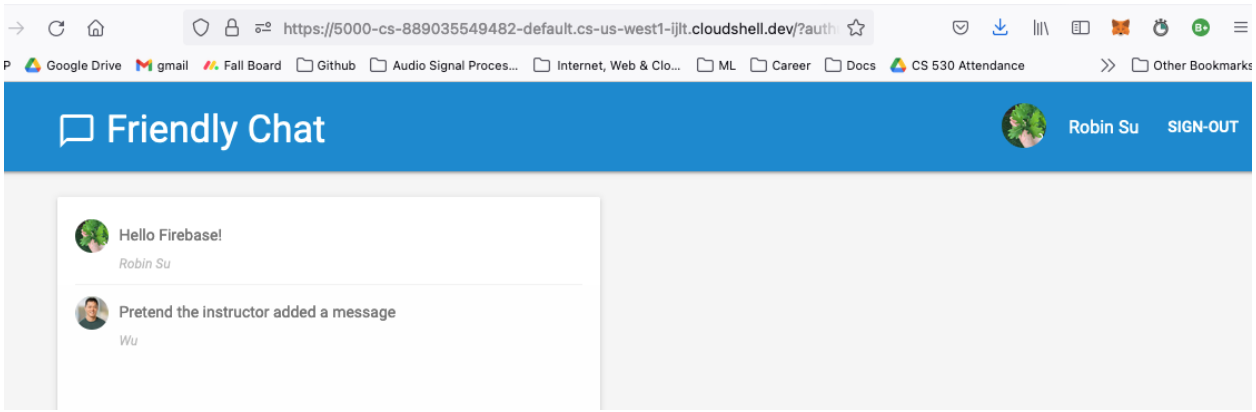
- *What missing functions deal with user authentication?*
    - **signIn(), signOut(), initFirebaseAuth(), isUserSignedIn(()**
- *What missing functions deal with sending and receiving messages?*
    - **saveMessage(msgTxt), saveImageMessage, saveMessagingDeviceToken()**

authStateObserver()

- *What are the names of the elements that are hidden when the user is signed out?*
    - **userNameElement, userPicElement, signOutButtonElement**
- *What is the name of the element that is not hidden when the user is signed out?*

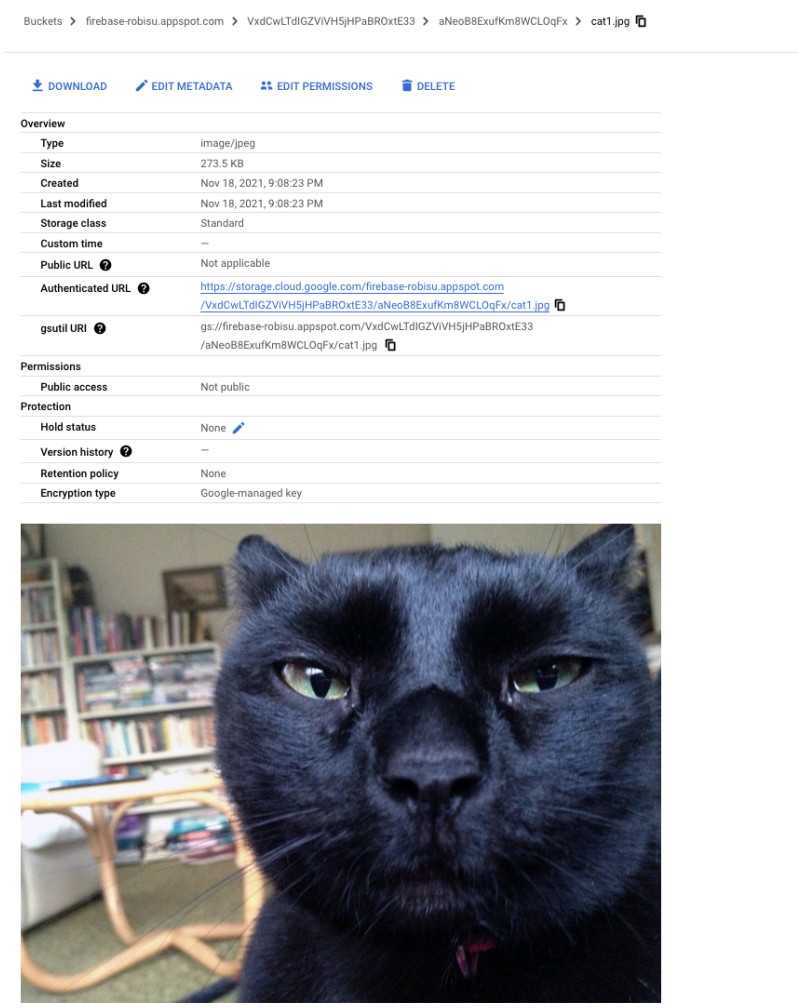
- **signInButtonElement**





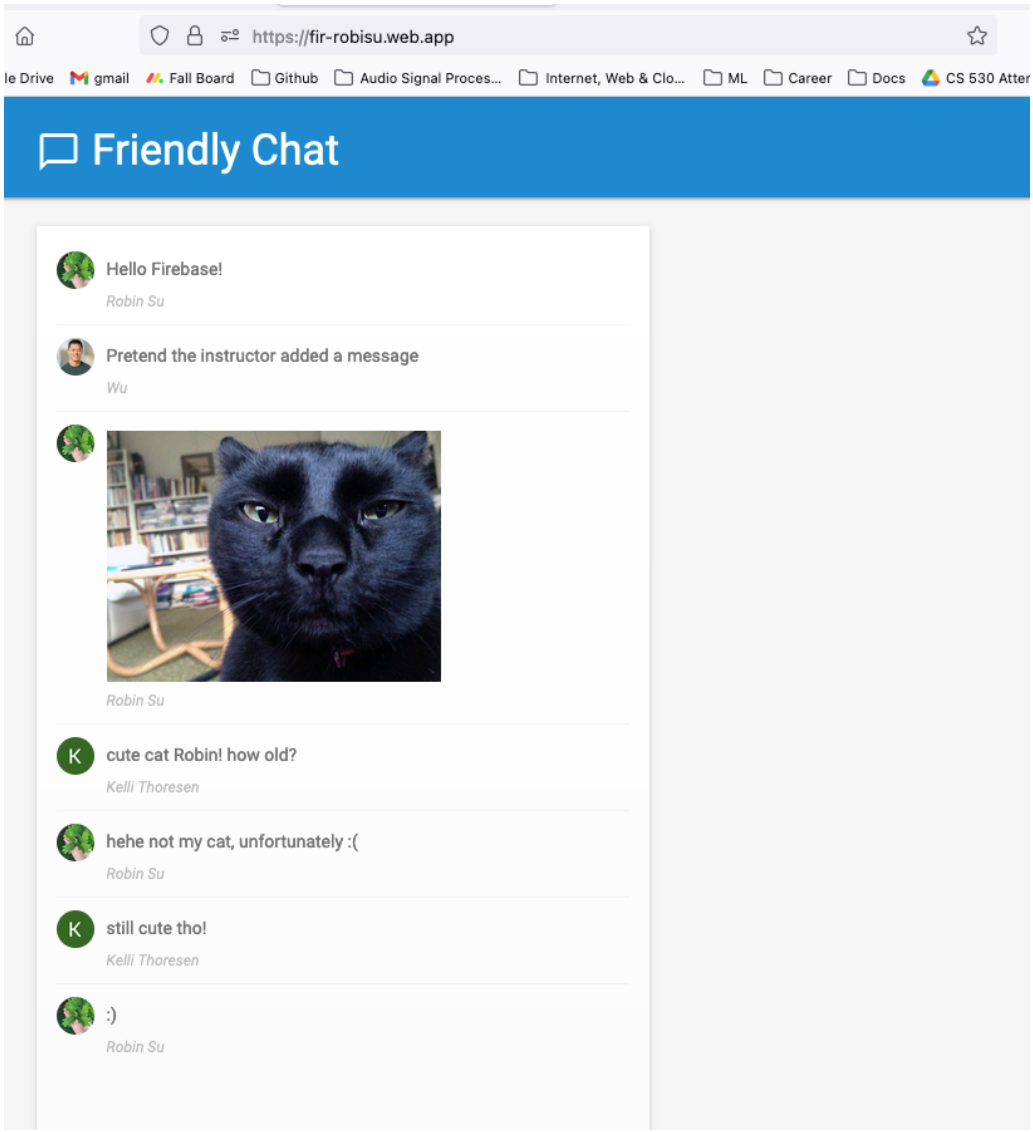
- *What is the URL of the image that is first shown in the UI as the message is loading?*
    - https://www.google.com/images/spin-32.gif?a

- *How do the fields in an image document differ from that of the text document?*
    - **Instead of a 'text' field, there is a 'imageUrl' field**
- *What URL and storage location can the image be found at?*
    - **imageUrl: https://firebasestorage.googleapis.com/v0/b/firebase-robisu.appspot.com/o/VxdCwLTdlGZViVH5jHPaBROxtE33%2FaNeoB8ExufKm8WCLOqFx%2Fcat1.jpg?alt=media&token=3cb17ab5-e2c4-4eb8-8f00-7d4571c954f5**
    - **storageUri: VxdCwLTdlGZViVH5jHPaBROxtE33/aNeoB8ExufKm8WCLOqFx/cat1.jpg**

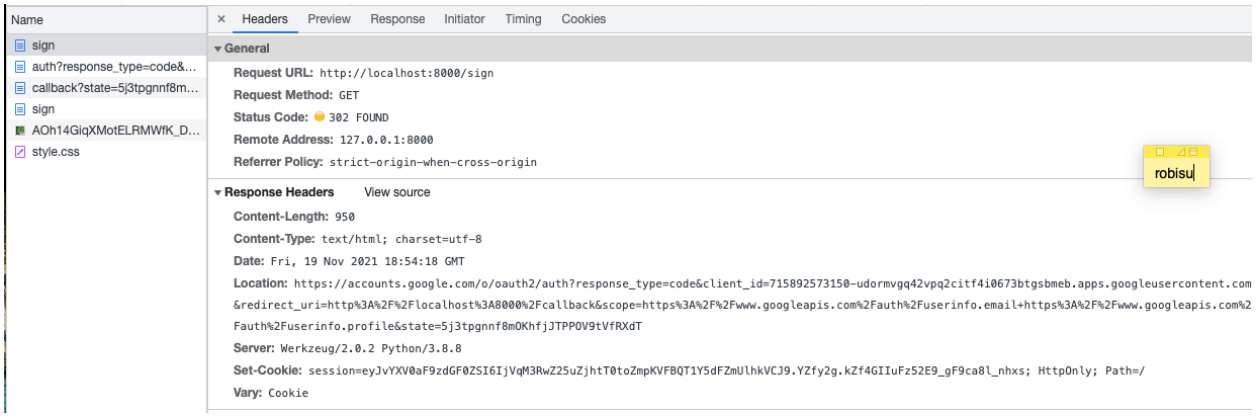


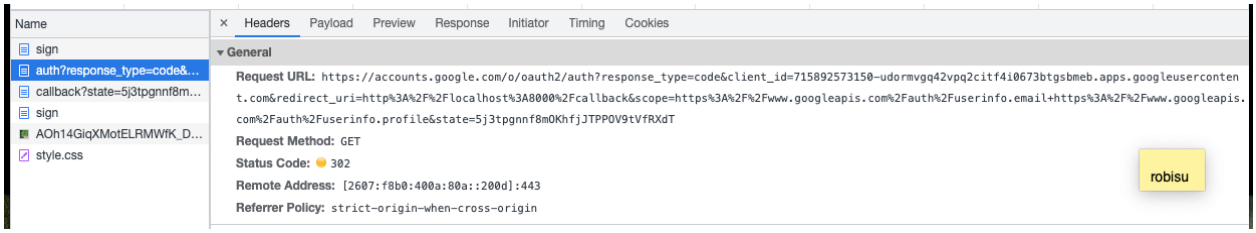
firebase.json

- *What directory is the application going to be served from?*
  - **./public**
- *What does the Cache-Control setting configured for the HTML and Javascript files do?*
  - **because 'max-age' is set to 0, instead of serving a cached version of the application, the page will make sure to fetch the most current version of the page possible (which is desirable for a real-time application that will always fetch the latest messages).**

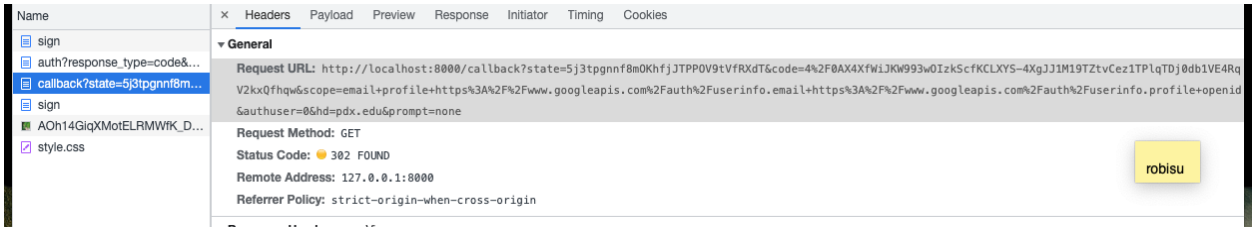


OAuth2 Guestbook



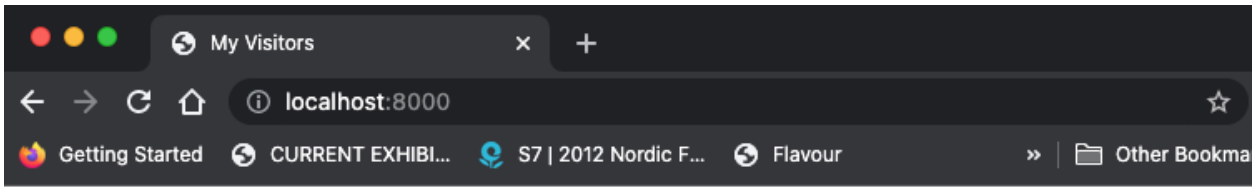


- *Based on the description of the source code, what lines of code in our application are responsible for the second request?*
    - **Lines #23-28 form the authorization url that directs the application to the authorization endpoint with Google**



The user is redirected back to localhost:8000/sign once Google has verified that the user is authenticated. The Guestbook app gets the session token, which means it is authorized to pull user profile and email info.

URL for embedded image:



Guestbook

Sign here | Logout

Entries

Robin Su <robisu@pdx.edu>
on 2021-11-19
Hello OAuth2 from pdx.edu!

Robin Su <robinjsu@gmail.com>
on 2021-11-19
Hello OAuth2 from gmail.com!