

# CSE 546 — Gym Search as a Service

**Group no:** 39

**Team members:**

*Radhika Nyati-1217133801- rnyati@asu.edu*

*Preeathi Raman-1217202480- praman5@asu.edu*

*Samyuktha Sridhar-1217186594 -ssridh55@asu.edu*

## 1. Introduction

The aim of this web application is to help the people in finding a gym in a particular city based on the gym's reviews. Nowadays, many gyms and other fitness centers are temporarily closed due to COVID-19. The gyms which are open are also for a limited time and limited occupancy only.

### 1.1 Problem Statement

It is harder than usual for people to search for gyms on various websites and then contact them to get their availability and check their COVID-19 guidelines. To understand in detail about the safety of the gym with respect to COVID-19, the people have to go through many reviews provided by others. This task becomes tedious when the number of fitness centers and its reviews increases. With the limited choices that the user has got, it is difficult to finalize a Gym to purchase membership without looking into fellow user reviews and other features. The unavailability of filtering gyms based on sessions such as Yoga, Zumba, Dance, etc under a single common platform makes it difficult for the users to go through all the details in all the websites. Also, there are not many platforms which recommend gyms in a city based on the sentiments of the reviews.

### 1.2 Proposed Solution

This application will let a user search for gyms in a city and get a list of gyms sorted by the sentiment score of their reviews. The users will also have an option to filter the gyms based on the above mentioned sessions and the users can also sort the list by distance from their current location. The users can also add reviews and subscribe to a gym and get notified whenever a change to the gym's information is made. The application also lets gym owners/managers register a gym and edit their details.

## 2. Background

With the current scenario of COVID-19, things are so uncertain and it is difficult for the people to use all the facilities like before without giving a second thought. Many gyms and other fitness centers are temporarily closed. Those that are open have some specific guidelines and restrictions due to the pandemic. It became harder than usual for the people to go through these details and reviews to come to a conclusion before purchasing a gym subscription. Without a common platform, it is hard for the

users to manage to view all details of all fitness centers and get updates about specific gyms of user preferences.

There is no common platform for viewing maximum occupancy for a group of gyms. The existing applications provide details about one particular fitness center only. People need to check their website regularly or need to contact them on a regular basis to get to know about current occupancy and COVID guidelines.

This application will serve as a single platform for the user to view details of all the fitness centers and help other users by providing reviews. All the reviews can also be analysed in a quicker manner by looking at the review summary and the users stay updated with the latest announcements and session related information from their subscribed list of gyms. In this manner, this web application is quite different from those that are existing.

## 2.1 Technologies

We have used a blend of services from the Google Cloud Platform such as the Google App Engine, Google Maps API and utilized the Platform as a service functionality in a whole.

- *Platform as a service(PaaS):*

With the Google App Engine, most of the resources are taken care of by Google. Autoscaling of resources based on the number of requests are managed by the App Engine by creating and terminating the instances.

- *Google Maps API:*

Google Maps API is used to retrieve the details of the gym with the name and city information. This API retrieves details such as the name, contact number, address, website, overall rating, reviews, opening hours, etc.

- *Google App Engine:*

Google App Engine is used as a serverless platform for developing and hosting web applications at scale. The front-end is hosted in the Google App Engine which provides automatic scaling to handle multiple concurrent user requests at a time.

- *Firestore:*

Firestore is used as the database to store details of the Gyms and Gym owner login details in two separate collections. Each Gym is stored as a document in the collection called 'Gyms'. The fields include name, address, contact number, reviews, summary, events, opening hours, etc. Each document in the 'Gym users' collection contains the login details such as the email id, password and contact number.

- *Scikit learn, Natural Language Toolkit:*

These suites and libraries are used to perform sentiment analysis on the user reviews for the gyms and then display the gyms based on the sentiment score computed.

- *HTML, CSS, Javascript:*

These technologies are used to provide a rich user interface to this application.

### 3. Design and Implementation

Application URL - <https://cc-project-2-310102.wl.r.appspot.com/>

#### 3.1 Architecture

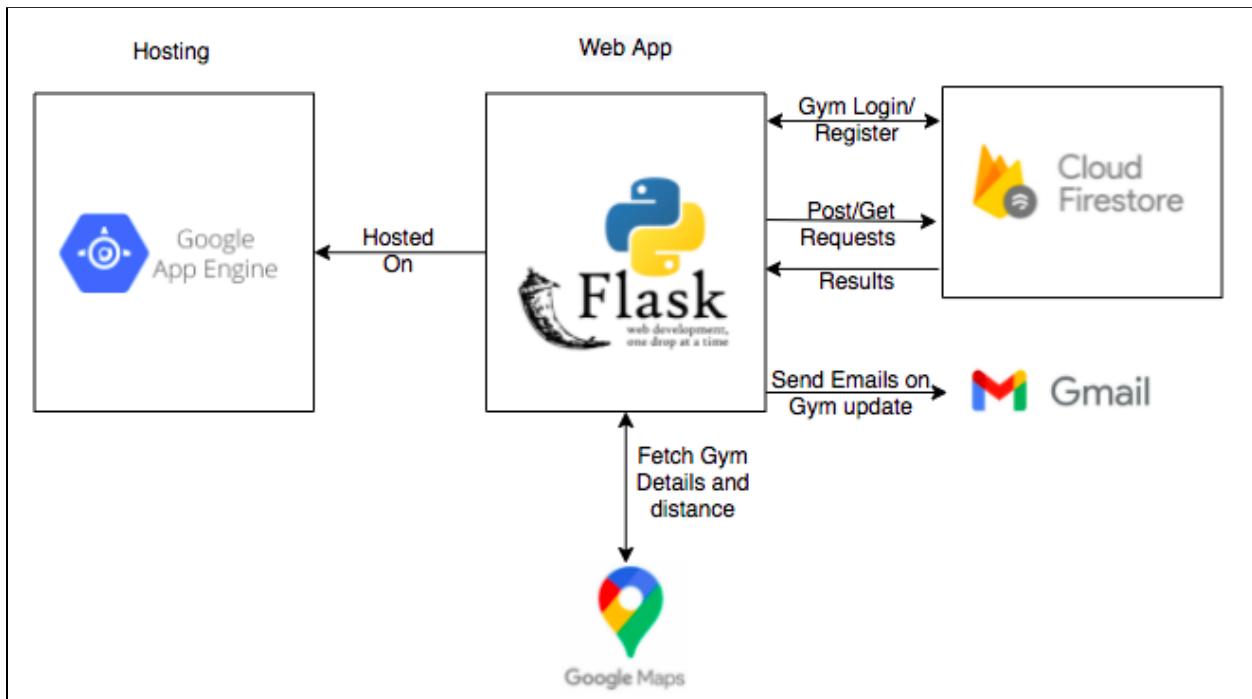


Fig. 1 Architecture Diagram

The above figure shows the architecture diagram of our design. GymSearch is a two-tiered architecture which consists of a web application tier and a back-end application tier. The web application is deployed in Google App Engine. Firestore has been used to store the gym and user details. We have used Google Maps API (placeAPI and distance matrix) to fetch the details about a gym and the distance between the gym and the user location respectively. We have also enabled mailing services using flask\_mail to notify subscribers regarding changes in the details of the gyms they are subscribed to.

The implementation of the design are as follows:

- **Frontend:** We have used Flask to build our web application. Our application can have two types of users: i) users searching for gyms and ii) gym managers/owners. The users searching for the gyms

are prompted to enter a city. Once a city name has been entered, the application displays a list of gyms in that city based on the sentiment score of their reviews. The users can filter the reviews based on three options:

- *Filter by sessions*: The list of gyms are filtered by the type of sessions conducted by them. The type of sessions include: Yoga, HIIT, Zumba, Pilates, Kickboxing, Full body fusion, Cycling, Personal training etc.
- *Filter by opening hours*: The list of gyms are filtered based on whether a gym is currently open or closed.
- *Filter by distance/recommendation*: On filtering based on distance, the users will be prompted to enter a location. The distance of the gyms from that location are computed and are sorted in ascending order of the distances. On filtering based on recommendation (default), the gyms list is sorted based on the descending order of sentiment score of their reviews.

The users can view a gym by clicking the name. The view\_gym page displays the information about the gym such as location, contact, occupancy, covid guidelines, sessions, reviews, review summaries, ratings, etc. The users can also add a review and give a type to their review. The users have an option to subscribe to a gym and get notified about the changes in the details provided by the gym.

The gym managers/owners can log in to their account by providing their account credentials. These details are validated before they are logged into their account. Once they are logged in, they can view their gym details, summary of their reviews and the subscriber list. The gym managers can edit the details provided. Once any edit is made, all the subscribers get notified via email regarding the change made. We have used flask\_mail to send emails.

The gym managers/owners can also register their gym by providing information like name, login credentials, occupancy, contact, address, covid guidelines, sessions. The application first checks if the gym is already registered or not. If yes, an error message is displayed informing the same. Once the gym is registered, we use PlaceAPI to fetch the details about the gym such as opening hours, ratings, reviews, phone number. The application also computes the sentiment score and summary of the reviews once the details are fetched.

- **Backend:** Our backend consists of the database and the API calls made. We have used firestore to store the gym information and google cloud map API to fetch details about a gym.
  - *Cloud Firestore*: Our application has used firestore as its database because of its scalability. Firestore lets us save the documents in the form key-value pairs and allows us to create multiple collections with multiple documents that can have many records. Each record is identified by its unique key. We have stored all the details about a gym in one collection. Each gym in that collection is a document which can be identified by its name. We have created another collection to store the gym managers' information.
  - *Google Map API*: We have used google map API to fetch details about a gym and to calculate distance between a gym and a given location. Google maps provide various APIs that can be used by the developers to build applications and fetch details about a place, append dynamic

maps or street views to the application, calculate distance and get directions for a place etc. We have used the PlaceAPI to fetch the up-to-date information about the gyms. This is done by making a HTTP request with the API key created. We first fetch the place id of the gym using geocoding. We then make a HTTP request with the API key, place id and the required fields to be fetched. We fetch details like url, formatted address, formatted phone number, overall ratings, reviews. We have also used the Distance Matrix API to get the distance between a gym and a location provided by the user.

- **Application Deployment using Google App Engine (HOST):**

We have used Google cloud platform, which is a Platform as a Service to host our application. We have deployed our application in Google App Engine(GAE). GAE is fully managed by GCP and we do not have to worry about the underlying infrastructure. GAE allows scalability of the applications and once deployed, the application can be used from anywhere. We have deployed the application with the instance class F2 and have configured it to have automatic scaling by provided some criteria such as minimum number of instances, maximum number of instances, maximum concurrent requests that can be handled by the user.

### **3.2 Autoscaling**

One of the bottlenecks in our application is handling a large number of requests. We have used auto scaling to help address this issue. Google App Engine provides various ways to scale the instances. We have made use of the automatic scaling provided by the google cloud platform. The app was deployed with a minimum instance of 0 and a maximum instance of 10. Each instance can handle a maximum of 50 concurrent requests. New instances are created based on the number of requests made and the instances are terminated based on the user requests. In this way, the application is made scalable.

### **3.3 Proposed Solution**

This application provides a single platform to search for gyms and check the occupancy based on the new COVID-19 guidelines and quickly glance through the review summaries. All the details about a particular gym such as the gym name, address, overall rating, reviews, COVID guidelines, maximum occupancy, session details such as Yoga, Zumba, HIIT, etc along with their occupancy are provided in the user view. Using this application, the users can quickly glance through the reviews by looking at the overall summary of reviews rather than reading each and every single review to come up to a conclusion. It will also provide recommendations to the gym's users based on their requirements. Users can search for the session of their choice in the nearby area. This application also provides a platform for users to help other users by providing reviews. Users can also subscribe to the gym and can get real time updates in the session's schedule, occupancy, COVID guidelines or operational hours. This feature eliminates the need to check websites again and again for any updates.

From the gym owners perspective, this application will help registered gym owners to expand their customer base and keep the existing users posted about any changes. This application provides an easy to use interface to gyms which allows them to update their schedule and other details. All the subscribed users will be notified via email whenever there is any change in the fitness sessions schedule, occupancy or operational hours. This feature will keep gym's users posted. Updating it on a website on a regular basis can be difficult for them as many gym owners or managers might not be that much tech savvy to own or maintain a website on a regular basis. Based on the summary of the reviews the gyms receive, this application can also help the gym owners by providing useful insights about peer competitors. In this manner, the gym owners can identify their drawbacks and work on bringing in more customers.

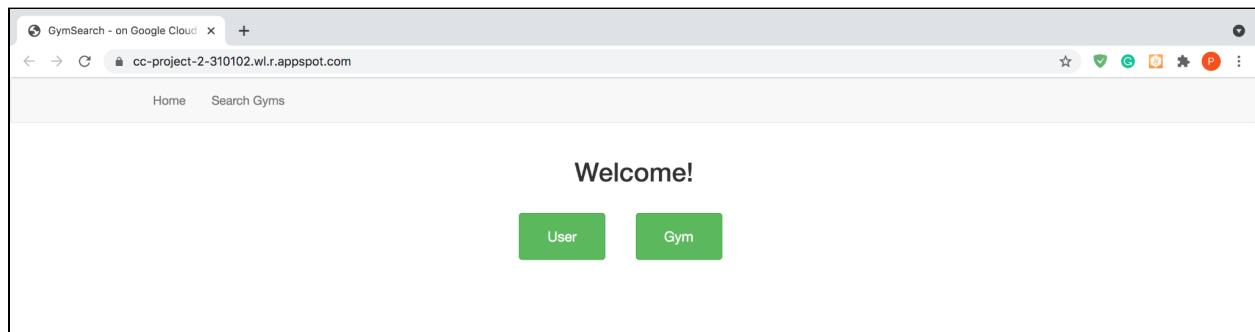
#### 4. Testing and evaluation

This application has two different users - Users searching for gyms and Gym owners.

##### 4.1 User Flow

The user can perform searching for the gym, adding reviews for the gyms and subscribe to the gyms of their preference. Upon launching the application, the user will be directed to the home page from where the user can click 'Search Gyms' or the 'User' button to lookup gyms based.

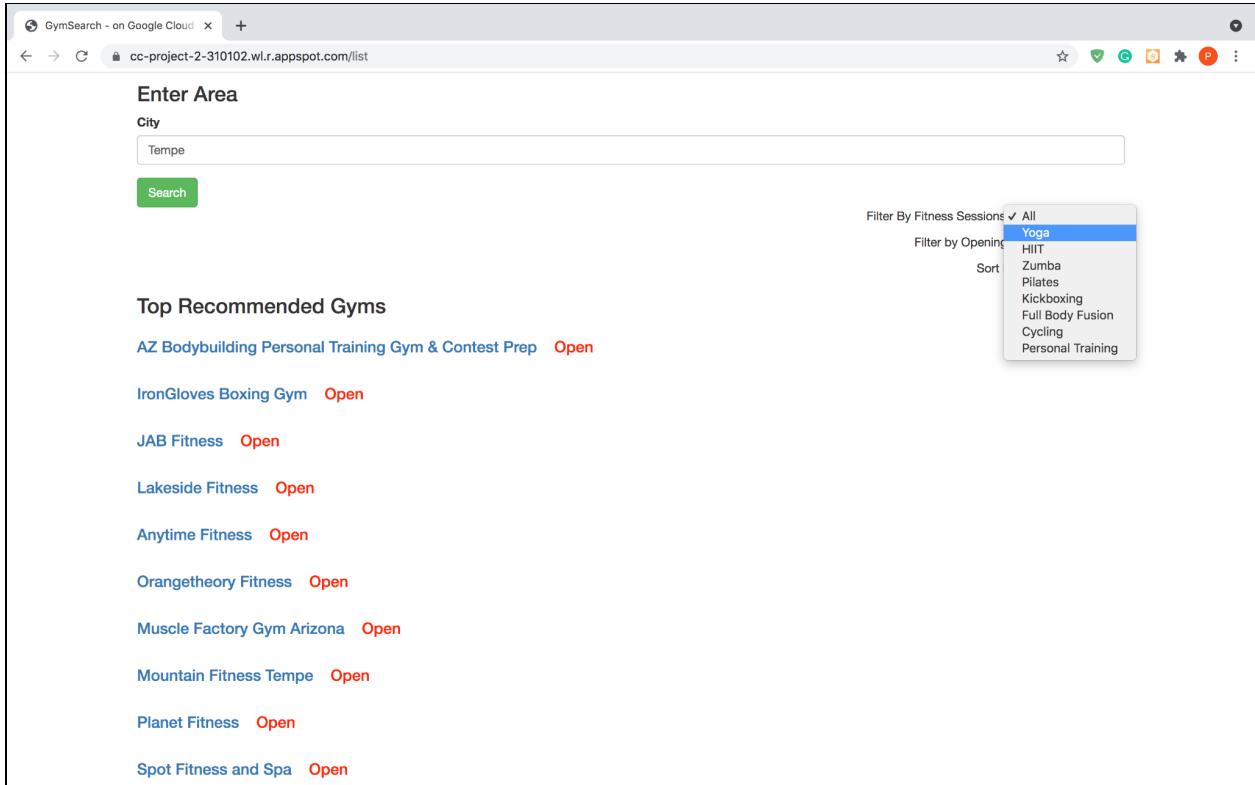
##### Home Page:



**Fig. 2 Home Page**

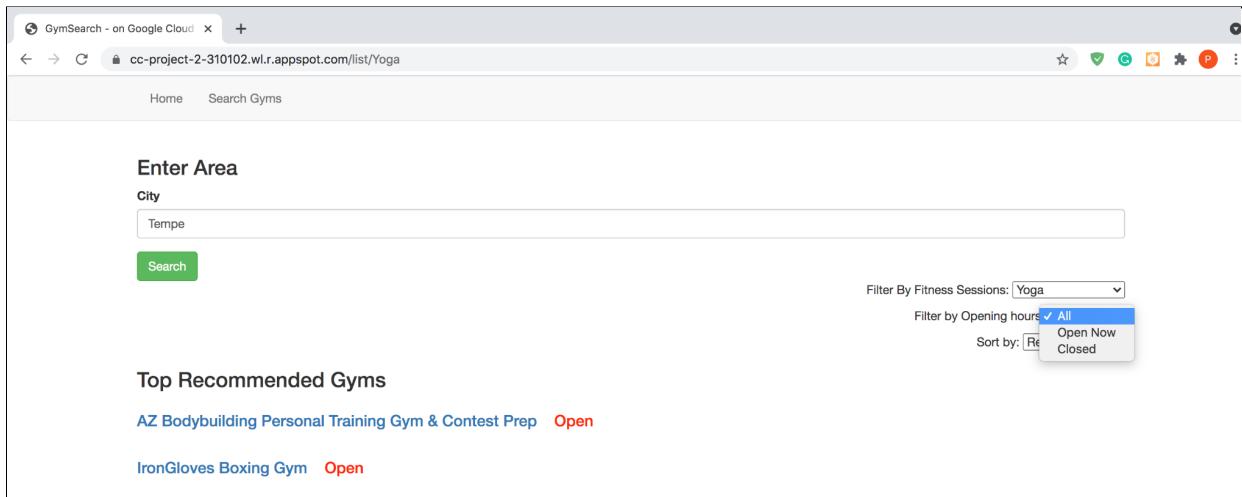
##### Gym Search Page:

Upon searching for 'Tempe', the gyms in Tempe are displayed and are sorted by default based on the decreasing order of sentiment score. The user will be able to filter the gyms based on sessions such as Yoga, HIIT, Zumba, Pilates, Kickboxing, Full body fusion, Cycling, Personal Training.



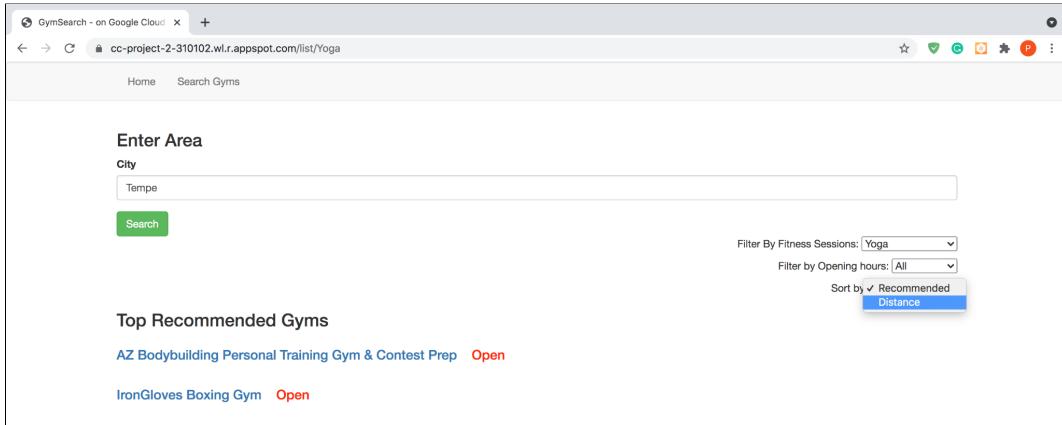
**Fig. 3 Gym search**

Other filters such as ‘Open now’ and ‘Closed’ can also be applied in the user search criteria. Based on the user selection, a set of Gyms will be displayed.



**Fig .4 Filter search**

One of the other ways to filter the gyms is to sort it based on the distance criteria. Upon clicking that option, the user will be prompted to enter the address details so that the new order of gyms will be displayed.



**Fig. 5 Sort the results**

A form interface for entering address details. It includes three dropdown menus: 'Filter By Fitness Sessions: Yoga', 'Filter by Opening hours: All', and 'Sort by: Distance'. Below these is an input field 'Enter your address:' containing '1019 E University Dr' with a blue border, and a green 'Submit' button below it.

**Fig. 6 Enter address details**

Upon clicking a specific gym, the user will be able to view the details of the gym, add reviews and subscribe to the gym.

The screenshot shows a web browser window with the title 'GymSearch - on Google Cloud'. The URL is 'cc-project-2-310102.wl.r.appspot.com/gyms/IronGloves%20Boxing%20Gym'. The page displays information about 'IronGloves Boxing Gym' located at 1425 E University Dr #B109, Tempe, AZ 85281. It has an overall rating of 4.7. The website is <https://ironglovesboxing.com>. Occupancy is 25. A section titled 'Covid Guidelines' states: 'Equipment must be wiped before and after use with sanitising wipes. Temperature will be checked for every person entering the Gym. Everyone should stand only on the marked areas of the floor while waiting for equipment. All the halls and group classes should be booked in advance. Walk in is not allowed. Consumption of food is not allowed inside the Gym. Everyone should wear a mask at all times. Showers are closed temporarily. Cafe and Spa are closed to minimise the spread of Covid-19. Use of scarves, ski masks and balaclavas are not allowed as substitute for masks. Cash registers and information desk are closed temporarily and unnecessary interaction is not encouraged. Resistance bands, yoga mats, yoga blocks, foam roller, other personal items must not be shared with others.' Below this is a table titled 'Weekly Workout Sessions' with the following data:

Type	Name	Time	Days	Occupancy
Cycling	Glow	6-7	Mon,Wed,Thurs	5
HIIT	HIIT	5-6	Mon	6
Yoga	Vinyasa	8-9	Tue,	5
Personal Training	Personal	9-10am	Mon	1

Under 'Opening Hours', it lists the days and times: Monday: 7:30AM–8:30PM, Tuesday: 7:30AM–8:30PM, Wednesday: 7:30AM–8:30PM, Thursday: 7:30AM–8:30PM, Friday: 7:30AM–6PM, Saturday: 7:30AM–1PM, Sunday: Closed.

Reviews section shows frequently occurred words: trainers(45), gym(126), training(54), place(54), boxing(86).

**Fig. 7 View Gym**

The user can subscribe to the gym using their email ID to receive regular updates from the gym. Subscribing here will add them to the subscribers list of the gym.

The first screenshot shows a form where an email address 'preeti3557@gmail.com' is entered into a field labeled 'Enter your email address:' and a 'Subscribe' button is visible.

The second screenshot shows a confirmation message 'Subscribed Successfully!' displayed on the screen.

The third screenshot is a box titled 'Subscribers:' containing three email addresses: 'samyu26897@gmail.com', 'praman5@asu.edu', and 'preeti3557@gmail.com'.

**Fig. 8 Subscribe to the Gym**

Users can add reviews by selecting filters and it can be seen at the gym detail page for both users and the gym owners.

The screenshot shows a web browser window titled 'GymSearch - on Google Cloud'. The URL is 'cc-project-2-310102.wl.r.appspot.com/gyms/add/IronGloves%20Boxing%20Gym'. The page has a header with 'Home' and 'Search Gyms' links. Below the header, the title 'Add Reviews for IronGloves Boxing Gym' is displayed. A dropdown menu 'Review Type' is set to 'Management'. There are three input fields: 'Name' (Radhika), 'Rating' (4), and 'Review' (Friendly staff). A green 'Save' button is at the bottom. The browser's address bar shows the full URL.

**Fig. 9 Add reviews**

**Name :** Radhika  
**Date :** 2021-04-30 17:27:23.512755+00:00  
**Rating :** 4  
**Review :** Friendly staff  
**Type :** Management

**Fig. 10 Reviews are displayed in the Gym page**

## 4.2 Gym Flow

### 4.2.1 Registration

Gym owners/managers can register themselves in this application using the registration form. They need to provide basic details like Name of the Gym, email address, password, contact number, covid guidelines, current occupancy of the gym, session details. They can add multiple sessions like HIIT, Zumba, Pilates, Cycling etc.

**Gym Registration**

**Name\***  
East Sac Fitness | Fitness in Sacramento

**Email\***  
eastsac@gmail.com

**Password\***  
....

**Contact Number\***  
(916) 599-3181

**Address\***  
Address Line 1: 4471 D St, City: Sacramento, State: CA, ZipCode: 95819

**Gym Occupancy**  
25

**Covid Guidelines**  
Wear a mask all the times  
The temperature will be checked upon entry

**Fitness Sessions**  
Type: Cycling Name: Glow Time: 10-11 AM Days: Mon, Wed Occupancy: 10

**Save** **Cancel**

**Fig. 11 Register a Gym**

At the time of registration, Google PlaceAPI is called to fetch the details like current opening hours, reviews etc. for each gym. The sentiment score and the review summarization is then computed for the fetched reviews and all the details are stored in the firestore. The application will not allow the user to register already registered gym or new gym with the already registered email address.

### Gym Registration

User Already Exists! Please Try to Login.

**Name\***  
Test

**Email\***  
proclub@gmail.com

### Gym Registration

Gym is already registered with other email address!

**Name\***  
PRO Club - Seattle

**Email\***  
proclub@gmail.com

**Fig. 12 Validate the gym registration**

The gyms and the gym owners details are stored in two different collections as shown in the figure.

The screenshot shows the Google Cloud Platform Firestore interface. The left sidebar has 'Firestore' selected. Under 'Data', it shows a tree structure: Root > Gyms > East Sac Fitness | Fitness in Sacramento. The 'Gyms' collection is expanded, showing documents for AZ Bodybuilding, Anytime Fitness, Better Bodies Fitness, Bodifi, Chuze Fitness, Club Fitness - Charlotte, CrossFit Utah Valley, Equinox Palo Alto, Gravity Fitness, Gymnastics Nevada Reno, and Unhoused Fitness. Each document has a detailed structure with fields like 'Area: "Sacramento"', 'Frequent\_Words', 'PlaceId', and 'Reviews'.

**Fig. 13 Gym collection in Firestore**

The screenshot shows the Google Cloud Platform Firestore interface. The left sidebar has 'Firestore' selected. Under 'Data', it shows a tree structure: Root > GymUsers > eastsac@gmail.com. The 'GymUsers' collection is expanded, showing documents for bodybuilding@gmail.com, chuze@gmail.com, clubapple@gmail.com, clubgym@gmail.com, crossgym@gmail.com, eastsac@gmail.com, and equinox@gmail.com. Each document has a detailed structure with fields like 'contact: "(916) 599-3181"', 'name: "East Sac Fitness | Fitness in Sacramento"', and 'password: "test"'.

**Fig. 14 Gym managers collection in Firestore**

After successfully registering the gym, gym owners/managers will be redirected to the Gym View Page. It is similar to the view page of the user but have extra features like ability to edit gym details and view subscribers list.

on Google Cloud | Data - Firestore - CC-Project- | Hello - preeti3557@gmail.com | GymSearch - on Google Cloud | gyms in sacramento - Google | +

Not Secure | cc-project-2-310102.wl.r.appspot.com/gyms/East%20Sac%20Fitness%20%7C%20Fitness%20in%20Sacramento/view

[Edit Gym](#) [Logout](#)

## East Sac Fitness | Fitness in Sacramento

**Address :** 4471 D St., Sacramento, CA 95819  
**Contact Number :** (916) 599-3181  
**Overall Rating :** 4.7  
**Website :** <http://eastssacfitness.com/>  
**Occupancy :** 25  
**Covid Guidelines :** Wear a mask all the times The temperature will be checked upon entry

**Weekly Workout Sessions:**

Type	Name	Time	Days	Occupancy
Cycling	Glow	10-11 AM	Mon, Wed	10

**Opening Hours**

- Monday : 5:00 AM – 9:00 PM
- Tuesday : 5:00 AM – 9:00 PM
- Wednesday : 5:00 AM – 9:00 PM
- Thursday : 5:00 AM – 9:00 PM
- Friday : 5:00 AM – 9:00 PM
- Saturday : 7:00 AM – 6:00 PM
- Sunday : 7:00 AM – 6:00 PM

**Subscribers:**  
No Subscribers found

**Reviews :**

Frequently occurred words in reviews

[gym\(8\)](#) [Sac\(3\)](#) [workout\(3\)](#) [Christmas\(4\)](#) [thanks\(3\)](#)

**Summary**

[christmas, eve](#) [east, christmas](#) [fitness, christmas](#) [great, christmas](#) [grumpy, christmas](#) [open, christmas](#) [personal, great](#)

**Filter By:** [All](#)

Fig. 15 Edit Gym Page

ch - on Google Cloud | Data - Firestore - CC-Project- | Hello - preeti3557@gmail.com | GymSearch - on Google Cloud | gyms in sacramento - Google | +

Not Secure | cc-project-2-310102.wl.r.appspot.com/gyms/East%20Sac%20Fitness%20%7C%20Fitness%20in%20Sacramento/view

## Reviews :

Frequently occurred words in reviews

[gym\(8\)](#) [Sac\(3\)](#) [workout\(3\)](#) [Christmas\(4\)](#) [thanks\(3\)](#)

**Summary**

[christmas, eve](#) [east, christmas](#) [fitness, christmas](#) [great, christmas](#) [grumpy, christmas](#) [open, christmas](#) [personal, great](#)

**Filter By:** [All](#) [General](#) [Covid-19](#)

**Name :** Morgan Nordahl  
**Date :** 5 months ago  
**Rating :** 5 stars  
**Review :** My brother and me enjoyed workout at this gym, actually all the equipments that we need were there, also we got free advices from a personal trainer, I do not remember his name lol but thanks for him, the problem with me was the distance from my apartment to the gym, it was a little far, but if I live close this gym i will not leave it forever, however thanks for the great people who work at the Reception, they always smile and do their job on the right way, thanks for all crew of the gym.  
**Type :** Covid-19

**Name :** Timothy Lee  
**Date :** a year ago  
**Rating :** 5 stars  
**Review :** This gym had everything I needed for a great workout. They were open till 6 PM on Christmas Eve when every other gym in Sac closed at 2 PM. Thanks for being there East Sac Fitness! You made my Christmas! If I hadn't been able to workout Christmas Eve I would of been grumpy all day on Christmas. Thanks again!  
**Type :** General

**Name :** user  
**Date :** 5 months ago  
**Rating :** 5 stars  
**Review :** Best yoga place in town. love the instructor and just rejuvenating

**Name :** user  
**Date :** 5 months ago  
**Rating :** 5 stars  
**Review :** My brother and me enjoyed workout at this gym, actually all the equipments that we need were there, also we got free advices from a personal trainer, I do not remember his name lol but thanks for him, the problem with me was the distance from my apartment to the gym, it was a little far, but if I live close this gym i will not leave it forever, however thanks for the great people who work at the Reception, they always smile and do their job on the right way, thanks for all crew of the gym.  
**Type :** Covid-19

**Name :** Morgan Nordahl  
**Date :** 5 months ago  
**Rating :** 5 stars  
**Review :** My brother and me enjoyed workout at this gym, actually all the equipments that we need were there, also we got free advices from a personal trainer, I do not remember his name lol but thanks for him, the problem with me was the distance from my apartment to the gym, it was a little far, but if I live close this gym i will not leave it forever, however thanks for the great people who work at the Reception, they always smile and do their job on the right way, thanks for all crew of the gym.  
**Type :** Covid-19

**Name :** Timothy Lee  
**Date :** a year ago  
**Rating :** 5 stars  
**Review :** This gym had everything I needed for a great workout. They were open till 6 PM on Christmas Eve when every other gym in Sac closed at 2 PM. Thanks for being there East Sac Fitness! You made my Christmas! If I hadn't been able to workout Christmas Eve I would of been grumpy all day on Christmas. Thanks again!  
**Type :** General

**Name :** user  
**Date :** 5 months ago  
**Rating :** 5 stars  
**Review :** Amazing staff. Check it out

**Name :** user  
**Date :** 5 months ago  
**Rating :** 5 stars  
**Review :** Amazing staff. Check it out

**Name :** Timothy Lee  
**Date :** a year ago  
**Rating :** 5 stars  
**Review :** Amazing staff. Check it out

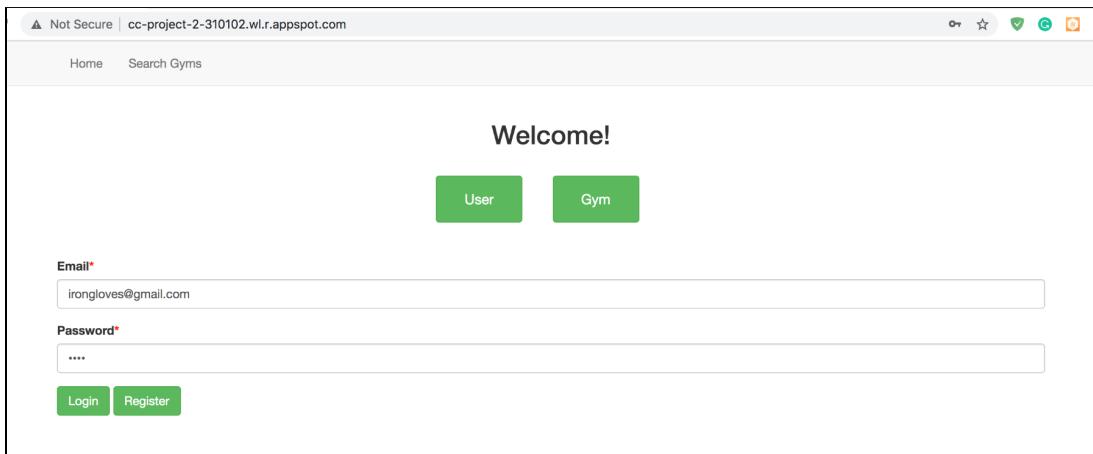
**Name :** user  
**Date :** 5 months ago  
**Rating :** 5 stars  
**Review :** Amazing staff. Check it out

**Name :** Timothy Lee  
**Date :** a year ago  
**Rating :** 5 stars  
**Review :** Amazing staff. Check it out

**Fig. 16 View Gym and filter reviews**

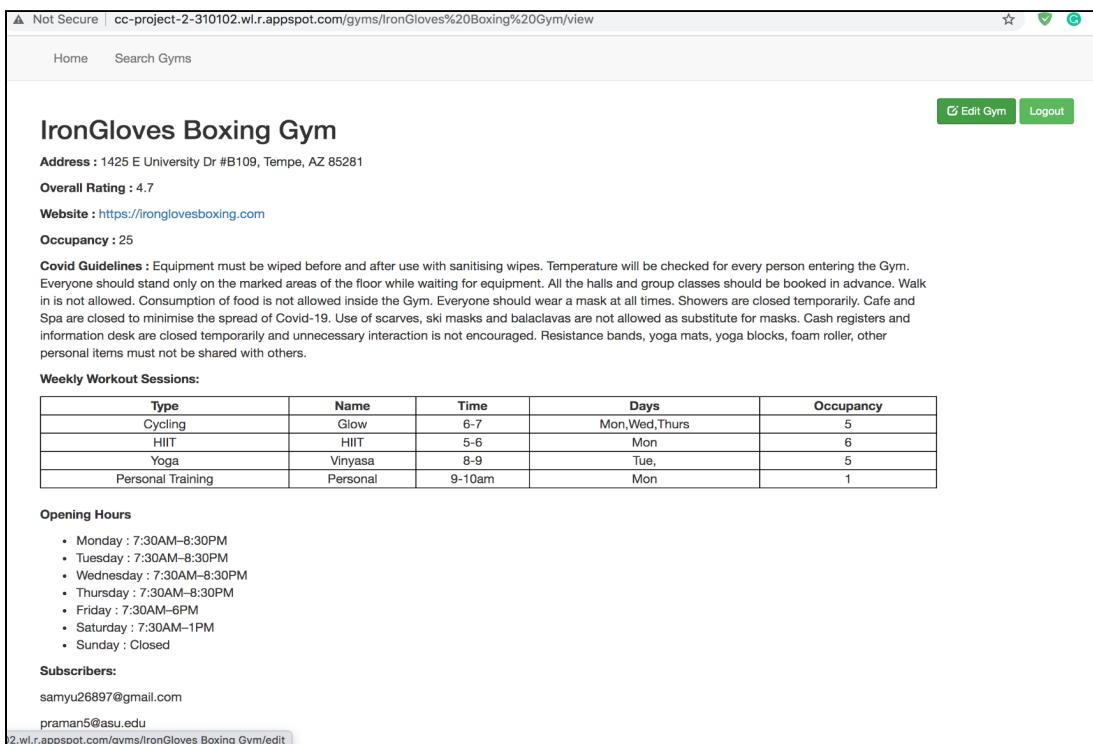
#### 4.2.2. Login Flow:

Gym owners can login to their account by using a registered email address and password. If the credentials are correct, they will be redirected to the View Gym details page.



The screenshot shows a web browser window with the URL "cc-project-2-310102.wl.r.appspot.com". The page title is "Welcome!". It features two green buttons: "User" and "Gym". Below these buttons are input fields for "Email\*" and "Password\*", both containing placeholder text. At the bottom are "Login" and "Register" buttons.

**Fig. 17 Gym Owner Login Page**



The screenshot shows a web browser window with the URL "cc-project-2-310102.wl.r.appspot.com/gyms/IronGloves%20Boxing%20Gym/view". The page title is "IronGloves Boxing Gym". It displays the gym's address (1425 E University Dr #B109, Tempe, AZ 85281), overall rating (4.7), website (<https://ironglovesboxing.com>), and occupancy (25). A "Covid Guidelines" section provides information about equipment sanitization, mask requirements, and social distancing. A "Weekly Workout Sessions" table lists classes like Cycling, HIIT, Yoga, and Personal Training with their respective times and days. An "Opening Hours" section lists daily operating times. A "Subscribers" section lists email addresses: samyu26897@gmail.com and praman5@asu.edu. There are "Edit Gym" and "Logout" buttons at the top right.

**Fig. 18 Gym Owner-Edit gym page**

Gym Owners can edit the details like current occupancy, current operation hours, scheduled sessions. As soon as any change is saved, an email will be sent to all the subscribed users informing them about the updates.

**Edit IronGloves Boxing Gym**

**Gym Occupancy**  
25

**Covid Guidelines**  
Equipment must be wiped before and after use with sanitising wipes. Temperature will be checked for every person entering the Gym. Everyone should stand only on the marked areas of the floor while waiting for equipment. All the halls and group classes should be booked in advance. Walk in is not allowed. Consumption of food is not allowed inside the Gym.

**Opening Hours**

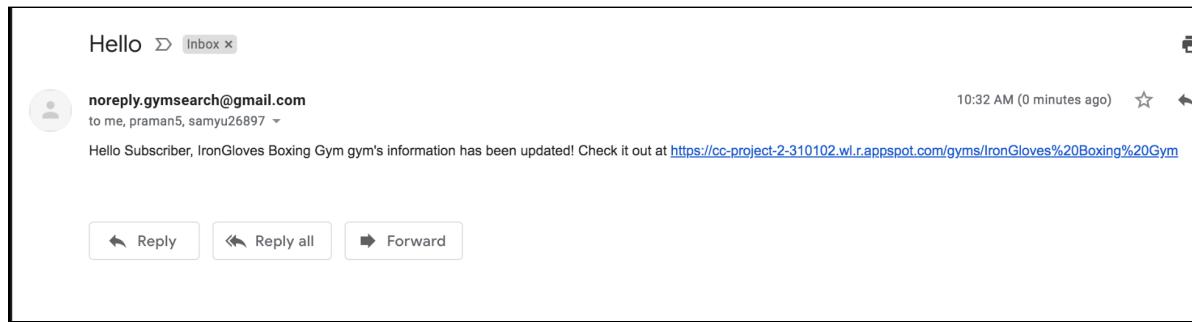
Monday :	7:30AM–8:30PM
Tuesday :	7:30AM–8:30PM
Wednesday :	7:30AM–8:30PM
Thursday :	7:30AM–8:30PM
Friday :	7:30AM–6PM
Saturday :	7:30AM–1PM
Sunday :	Closed

**Fitness Sessions**

Type: Cycling	Name: Glow	Time: 6-7	Days: Mon,Wed,Thurs	Occupancy: 5
Type: HIIT	Name: HIIT	Time: 5-6	Days: Mon	Occupancy: 6
Type: Yoga	Name: Vinyasa	Time: 8-9	Days: Tue,	Occupancy: 5
Type: Personal Training	Name: Personal	Time: 9-10am	Days: Mon	Occupancy: 1

**+Add Session**

**Fig. 19 Modify gym details page**



**Fig. 20 Subscription email update**

#### 4.3 Autescaling

We monitored the load using [Google App Engine Console](#).[1]. We used the automatic scaling feature provided by GAE to handle the load. We configured GAE automatic scaling so that it can have a minimum

of 0 instances and maximum of 10 instances running at a time. Each instance can work on 50 concurrent requests.

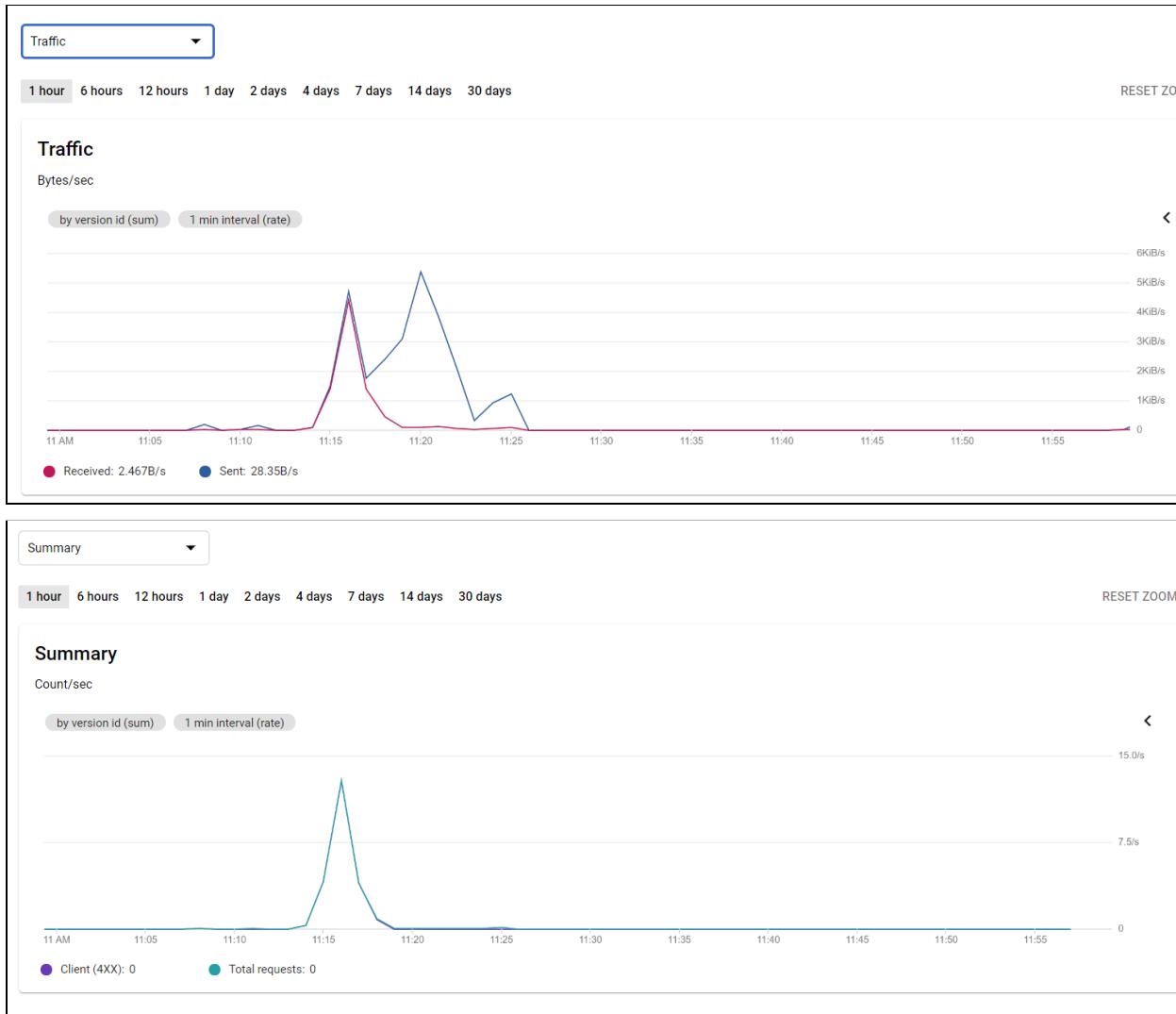
- Initially, there were no requests therefore 0 instances were running.

<input type="checkbox"/>	ID 	QPS 	Latency 	Requests	Errors	Memory	Start Time
This version has no instances deployed.							

**Fig. 21 Number of running instances**

- We used a python script which uses asyncio library to generate 1000 HTTP requests to this application. 7 instances were created automatically to handle the incoming traffic. Once all the requests were processed, instances were scaled down to 0.

<input type="checkbox"/>	ID 	QPS 	Latency 
<input checked="" type="checkbox"/>	00c61b117c1da4848019fe8f8e9a98983f68d6a955a43114a4cfda453c59708fe82cb561be647aeeb9bf7c084a9fd06afb7cbcd37c8ae888e8d2e3cd79490af6	0.2	603 n
<input checked="" type="checkbox"/>	00c61b117c201319e7fd172c2f0cf0248ea479907288c90d55633f156e088496181c6dfd9f751fbe98c046dd4a51fc3926761d72f824f1bc1747a3290b64f817	0.25	474 n
<input checked="" type="checkbox"/>	00c61b117c21182834755bc4f4eb7f6c54178c966a0adf3c0d630d10184a89e29329c67ce9e919dd943ce71646c646b0a28c896ef7241c123ba4a80d7835db3e	0.517	105 n
<input checked="" type="checkbox"/>	00c61b117c2b41fa788440b4b40b5a6b6674f01fb235cdb4448a05bd21f30ffdb6b0dcfc829f30a7d3988d535ce8b56c1f9f8f11f286439ef4fb373b0f9814548b	0	0 n
<input checked="" type="checkbox"/>	00c61b117c3c48e3d644eb36fc49ae815f30323c079ea9984f910207c12b4fa28385e19871f8ea9bdfcdc253084c49dd5d25a47228ef507214875fdcb33286f0e0	0	0 n
<input checked="" type="checkbox"/>	00c61b117cae03f9a78b178e980c6f2de1b31e911de6836a2f8d1b3dd48dd1a2d6df06c4fb12eea009f3fea6c189b0f80b2bcc1316e8345332dd232adc115b81be	0.35	87 n
<input checked="" type="checkbox"/>	00c61b117cf120e8e231e1fad3e140512cfe18febcef7191d6b81bb5fd3dcefd427e83d123b1041cc612dc1a2b2cce2fe4f772d79bc0afc5c9117c7c4e8bce1	0	0 n



**Fig. 22 Autoscaling- Traffic, summary of instances running**

## 5. Code

Some of the prerequisites to run this application are:

- Google cloud platform account
- Google maps API key

### 5.1 Program installation:

- Create a google cloud platform and create a project.
- In the cloud shell, upload the project and deploy it using the command : gcloud app deploy.
- Check the application version in the Google App Engine console.

### 5.2 Run web application:

- To run the application in the google shell editor, go to the folder GymSearch/.
- Use the command: `~/local/bin/gunicorn -b :8080 main:app` to run the application.
- Go to web preview and choose preview on port 8080 to access the application functions.

### 5.3 Code Description

This web application is built using Flask for the front-end. We have used Firestore as our database for query processing.

- **Templates:**
  - *home.html*: This is the landing page of our application. There can be two types of users in this application: users looking for gyms and gym owners/managers. To view and find gyms, the users can click the “Users” button. To login/register, the gym owners can click the “Gyms” button.
  - *gym\_list.html*: This page displays the list of recommended gyms. The users can enter the city name and the application displays top recommended gyms in that location. The gyms are sorted based on the sentiment score of their reviews. The users also have an option to apply filters and look for gyms with particular sessions such as yoga, zumba, cycling etc. The list can also be sorted based on the distance from the user's current location. The users are asked to enter the address. We have used Google map's distance matrix to compute the distance between each gym and display them after sorting. The users also have an option to view gyms based on their opening hours. To get the opening hours of each gym, we have used google map's place API. The users can click on a gym to view the details.
  - *view\_gym\_user.html*: This page displays the information about the selected gym such as the address, location, contact information, website link, sessions conducted, reviews, occupancy at a time, covid guidelines. The application also displays a summary for all the reviews in two ways: displaying frequently occurring words with their count, and displaying frequently used co-occurring words. The co-occurring words were found using nltk library and scikit-learn's countVectorize method. The users can also filter the reviews by their type: General, Covid-19, Management, Equipment. The users also have an option to add reviews for the gym by clicking the "Add review" button. The users can also subscribe to a gym to get notifications about any changes made in respect to the covid guidelines, occupancy, sessions or general information about the gym.
  - *add\_review.html*: This page displays a form which is used by the users to add review. The users will be required to add a review type, name, rating and review for the gym. Once a review is added, the review summary and the sentiment score for the gym gets updated.
  - *view\_gym\_user.html*: The gym managers/owners can log in to their account from the home page. Upon login, the details about their gym are displayed. This page displays all the details such as the location, contact information, website, subscribers, sessions, reviews. The

application also displays the review summary to the gym managers. The gym managers have an option to edit the details. Once an edit is made, all the subscribers of that gym get notified about the changes made.

- *add\_gym.html*: The gym owners/managers can register their gym to our application from this page. The gym managers will be required to provide details about the gym such as the gym name, login credentials, address and contact information. They will also have an option to provide the occupancy of the gym at a time, covid guidelines, sessions conducted. Once the gym registers itself, the application first checks if the gym already exists in the database or not. If yes, an error message is displayed stating that the gym already exists. Once the registration is complete, the application uses Google Maps PlaceAPI to fetch the rest of the details about the gym, such as ratings, reviews, opening hours. Once these details are fetched, the application computes the sentiment score and review summarization for the gym.
- **main.py**: This file handles all the requests from the front end and fetches data from the firestore. All these requests are handled by importing the libraries firestore, flask. Any errors in the application are also reported in the cloud console and we have also created a log entry in google cloud. We have created session variables here to handle the list of gyms and the city entered. This app is configured to run on port 8080 with host 127.0.0.1. [3]
- **firestore.py**: This file handles all the functions that make requests to the datastore. Main.py calls the functions in this file to retrieve data from the firestore. To view the list of gyms, the function “list\_on\_pref” is called. This function fetched data for the given city and sorted them based on their sentiment score before returning the list. Google PlaceAPI has been used here to fetch the current opening hours for each gym in the list. The functions *readGym* and *readGymUser* are used to fetch the details of the gym and the gym managers respectively. The function *updateGym* is used to update the details of a particular gym. The function *addReview* updates the “reviews” field whenever a new review is added by the users. Once the review is added, the sentiment score and review summarization is computed. The function *add\_subscriber* adds a new subscriber to a given gym. The function *add\_gym* creates a new document with the gym name as the ID and sets the field values with the details provided by the gym manager. Once the gym is added, our application uses google map’s PlaceAPI to fetch the details about the gym and updates the firestore with the new details. The sentiment score and the review summarization is then computed for the fetched reviews. *Add\_gym\_user* adds a new document to the ‘users’ collection when a gym registers itself. *getSpecificReviews* is used to filter the reviews based on specific fields such as ‘General’, ‘Covid-19’, ‘Management’, ‘Equipments’ etc. The *gymLogin* function is used to authorize the login information. *getSubscribers* function is used to fetch the details of the subscribers for a given gym. [4]
- **addReviewType.py**: This file is used to add a review type for the existing reviews fetched from the PlaceAPI. The review types can be general, Covid-19, management, equipment. We have made use

of nltk libraries to tokenize and get the part-of-speech tags for the words. We have made use of a list of words to identify the type of review.

- ***extractGymDetails.py***: This file consists of functions that are used to extract the details of the gym when it is registered. We have made use of google map's PlaceAPI to fetch the details such as place id, url, formatted\_address, formatted\_phone\_number, opening\_hours, website url, overall ratings, and reviews. These details are then updated in the firestore.
- ***reviewSummarize.py***: This file consists of functions that computes the summary of the reviews.[4] We have summarized the reviews in two ways: i) by displaying the frequently occurring words with their count. (ii) by displaying the frequently used co-occurring words. We have used the nltk library to get a set of stopwords and part-of-speech tags. We first tokenize the reviews and get their part-of-speech tags. We then create a new list after removing the stop words and words which are not tagged as nouns. We then get the frequency of the words in the list. To find the co-occurring words, we have used scikit-learn's countVectorize method, which converts the text documents to a 2D array of token counts. This helps identify the number of times the words in the document co-occur. We choose the top 10 co-occurring words and update the gym document details with it.
- ***sentimentScore.py***: This file is used to compute the sentiment score[5] of the reviews for a gym. We have used Valence Aware Dictionary and sEntiment Reasoner (VADER) to compute the scores. This is a lexicon and rule based sentiment analysis tool that can be used to label the reviews as positive, negative and neutral. The tool also gives a compound score of how positive or how negative the sentiments are.

## 6. Conclusions

### 6.1 Learning Outcomes

By developing this web application we learned to work on various services provided by the Google Cloud Platform such as the Google App Engine, Google Maps API, Firestore, etc. We got a chance to work on the front end technologies like HTML, CSS and Javascript to develop the User Interface. We learned to perform sentiment analysis using nltk and scikit libraries as a part of intelligence of this application. We accomplished in extracting the reviews from the Google Maps API and then summarizing it for user convenience. We were able to autoscale the resources based on the number of user requests and thus developed a scalable application.

### 6.2 Future Scope

This application can be extended by displaying real time updates of current occupancy in a particular gym such as the current occupancy of various hallways and equipment. The users can also register themselves for specific sessions offered by specific gyms such as the Zumba, Yoga, HIIT, etc. Cancellation feature can also be enabled so that it becomes hassle free for the user to cancel if they change their

mind. By registering in the particular gym's web page it makes it easier for other users to determine the remaining occupancy of every session without having to contact the gym.

## 7. References

[1]

<https://console.cloud.google.com/appengine/instances?serviceId=default&versionId=20210430t154953&project=cc-project-2-310102>

[2] <https://cloud.google.com/python/docs/getting-started?authuser=1>

[3] <https://firebase.google.com/docs/firestore/quickstart>

[4] S. R. Rahimi, A. T. Mozhdehi and M. Abdolahi, "An overview on extractive text summarization," 2017 IEEE 4th International Conference on Knowledge-Based Engineering and Innovation (KBEI), 2017, pp. 0054-0062, doi: 10.1109/KBEI.2017.8324874.

[5] P. Gupta, R. Tiwari and N. Robert, "Sentiment analysis and text summarization of online reviews: A survey," 2016 International Conference on Communication and Signal Processing (ICCSP), 2016, pp. 0241-0245, doi: 10.1109/ICCSP.2016.7754131.