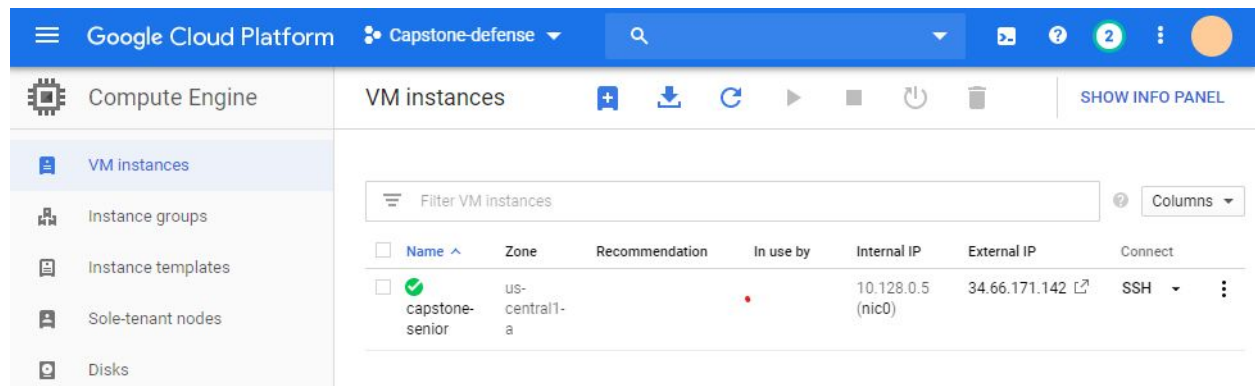Jagjot Singh
Dr. Nilesh Patel
CSI4999
1/23/2020

LogBook Entry For Sprint 1

I wasn't sure how you wanted the proof for the work but I did two all nighters and drank a lot of coffee. Heres what I did, and can you provide feedback so I know what I can do for next logbook thanks in advance!

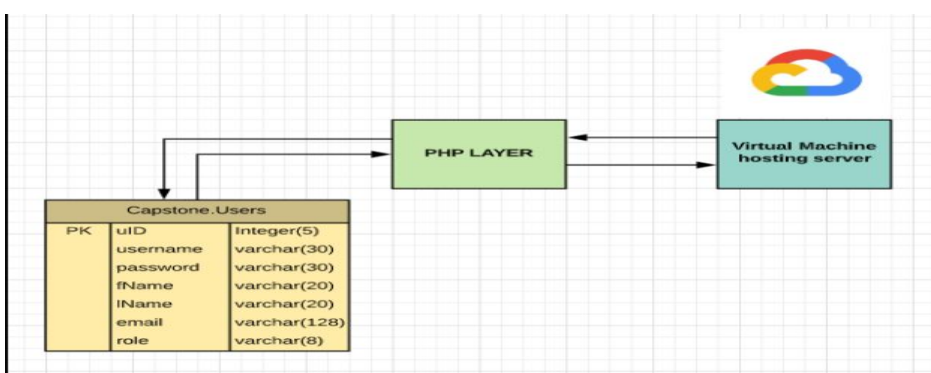1) Developed backend server on google cloud

Setup the server in Ubuntu LTS 16, and created a LAMP server or (linux, apache2, mysql, and php). This virtual machine instance is being used to host the website at the moment. **Effort hours: ~3-4 hours** → mostly because issues arise with setting permissions for mysql reading.

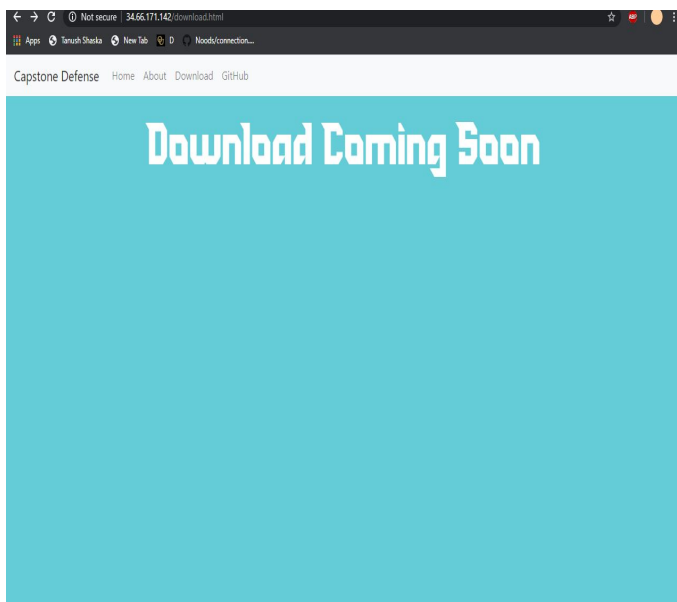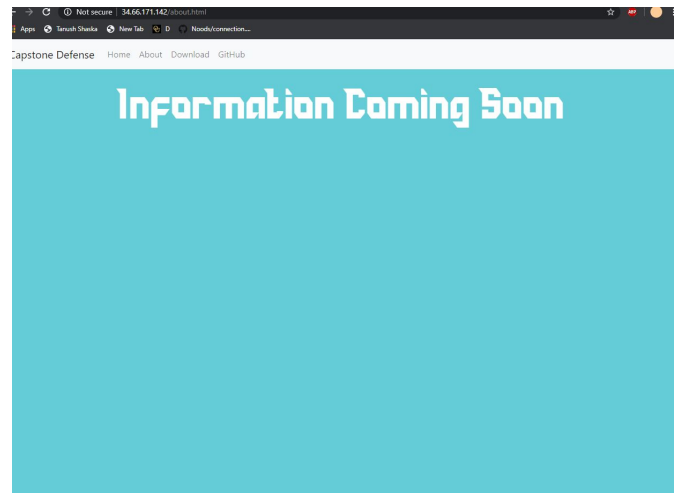2) Created tables in mySQL and ERD diagrams *UPDATE* status was added to table

3) Created front end design for website: **All front end code can be viewed in website folder for master. Effort Hours: ~4-5 hours *number varies refer uptop**

```
mysql> select * from users;
+-----+----------+------------+-------------+---------+---------------------------+---------+
| uID | username | password   | fName       | lName   | email                     | role    |
+-----+----------+------------+-------------+---------+---------------------------+---------+
|   8 | Pogger   | Pass       | test        | test    | test@gmail.com            | default |
|   9 | Admin    | HelloWorld | Adminstrator| Jackson | johnnyappleseed@aol.com   | admin   |
+-----+----------+------------+-------------+---------+---------------------------+---------+
2 rows in set (0.00 sec)
```

4) Backend connections to sql server and php made so users can register and input **~2-3 hours**

5) Setting up email server to send out emails: **Effort Hours ~3 hours**

Register

John

TODOhidethis

John

Doe

mlohair@oaklaı

Submit

Account Verification For Capstone Defense Inbox ×

capstonedef@gmail.com
to me ▾

11:21 AM (3 minutes ago)
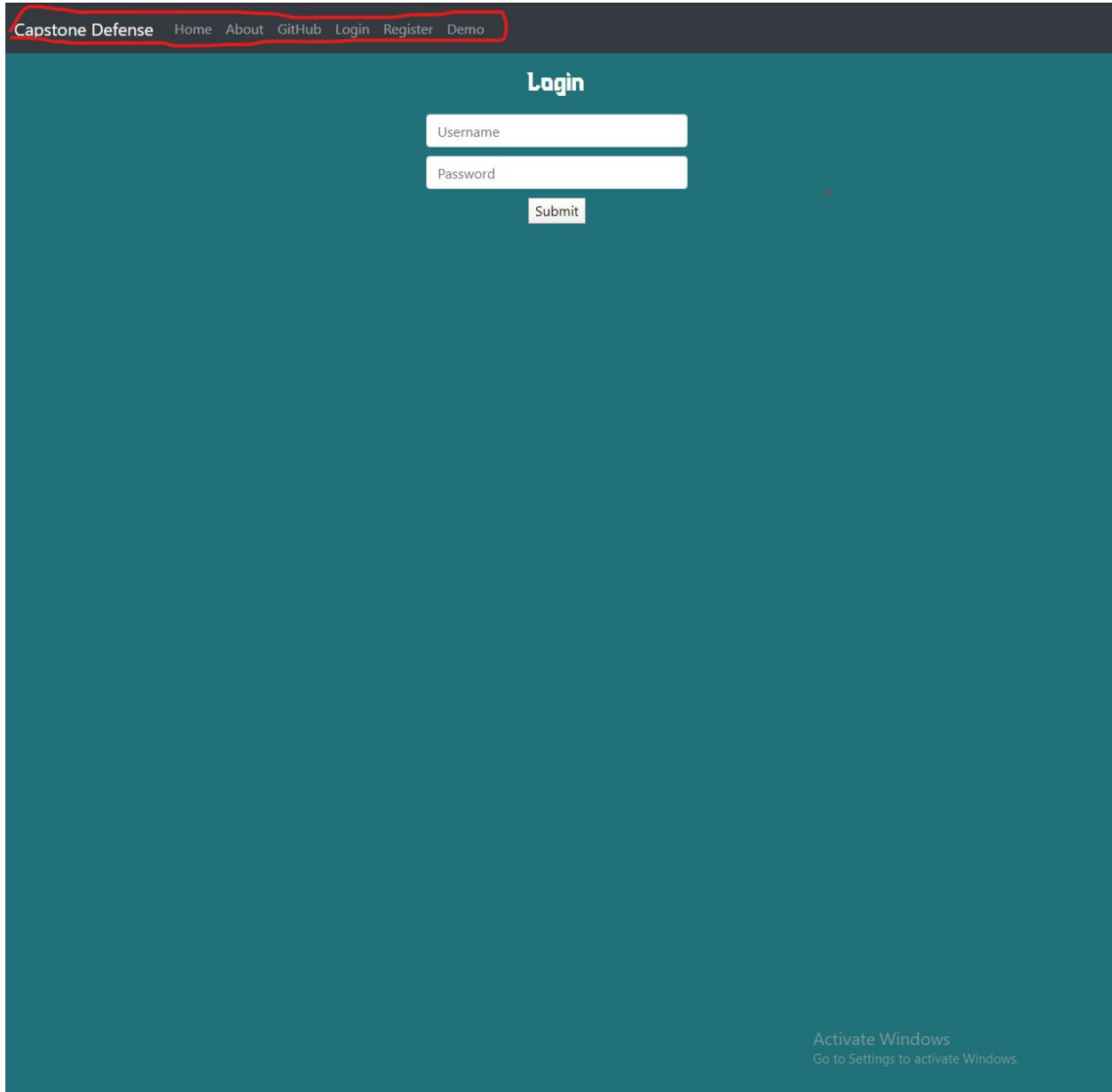
Visit the following link to activate your account:

6) Creating code to handle security, behavior of inputs, and clean up of front end/backend **~5-6 hours**
   These include:
   Carosoul view
   Different php files
   Email server clean up
   Registration fix for incorrect inputs
   ….
   Alot more but I didn't log everything

7) Worked on presentation **~1 hour**

Sprint 2 Documentation

Details: This sprint was used as a way to tidy up backend, and add features to the website. This also included fixing up the backend logic and incorporating demo developed by team onto website. Refer below to all th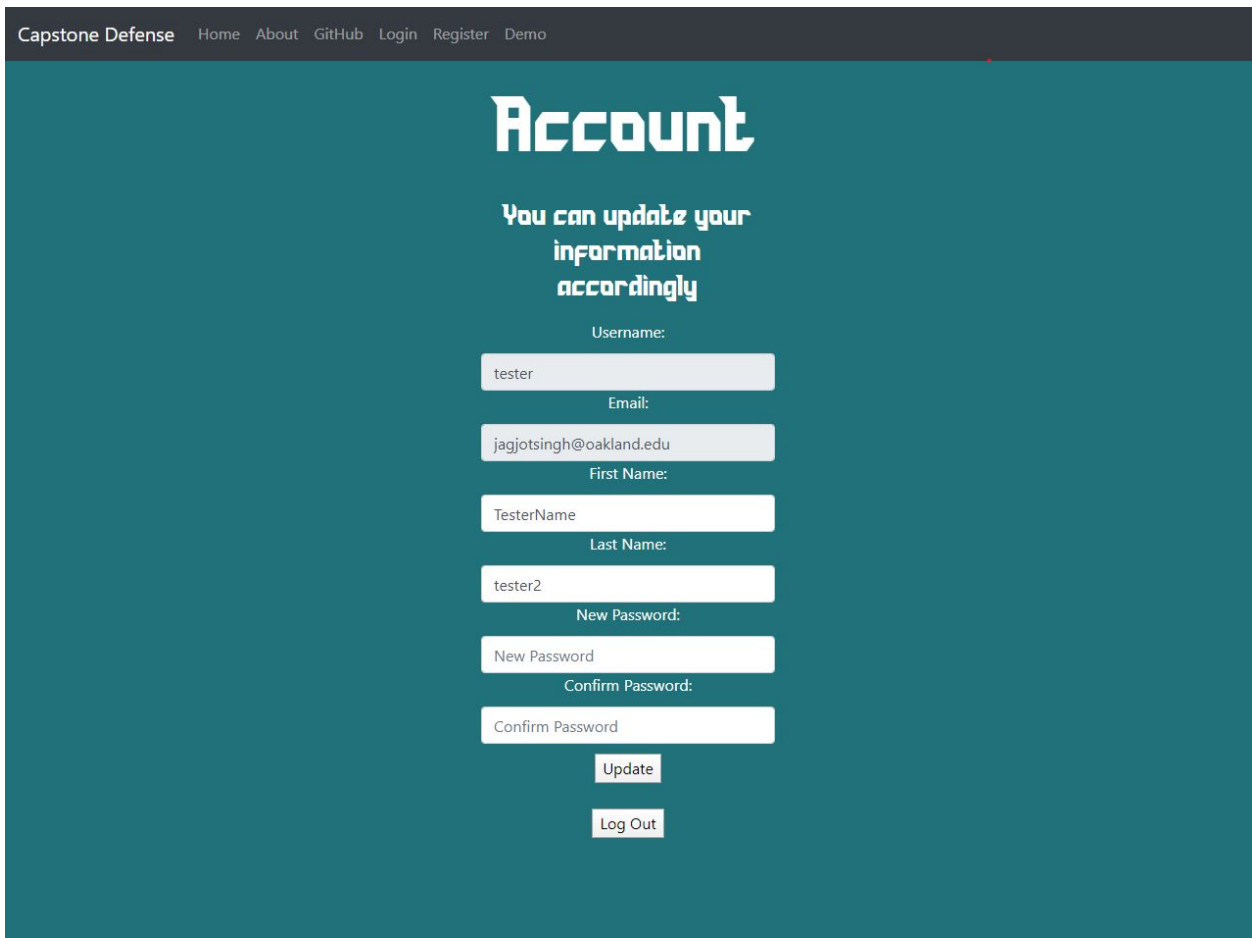e work done…ALL COMMITS AND WORK ARE SEEN ON GITHUB UNDER https://github.com/JTBridges/Capstone-Defence and specific commits https://github.com/JTBridges/Capstone-Defence/commits?author=jagjotsingh1

(1) Clean up pages and fix navbar issues **~2 hours**

This included separating login page, register, demo, and everything else to tidy up backend for server

(2) Work on update profile page, link backend, and working on sessions php ~**5 hours**

Included setting up all the formatting, and the majority of the time came into incorporating sessions to hold global variables to search with. The front end looks a bit bad, but that can be fixed with an overhaul on UI design for the website sprint 3.

(3) Updated table backend and got email verification with link working. **~4 hours**
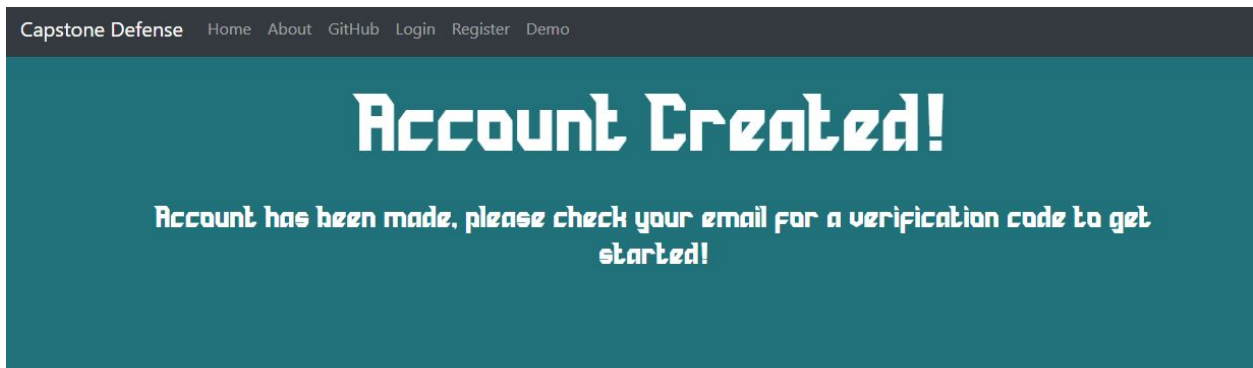
The hash is generated and sent via the mail exchange server to target email. The email also includes a verification page to unlock accounts to post scores.

(4) Created landing pages and also incorporated logic on inputs fe)required forms, valid email, etc. **~6 hours**

Landing pages didn't take too long but the logic and what security we need took a long time. As its hard to demonstrate that looks, the backend is processing inputs on what the user submits in forms and checks if they are valid responses. This took a very long time.
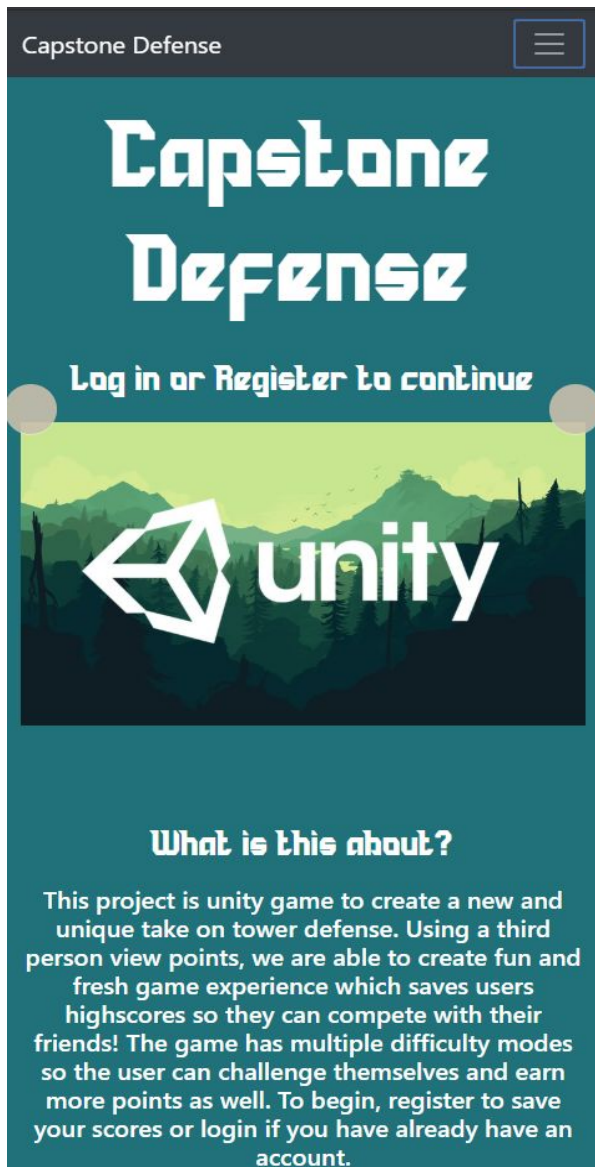
(5) Presentation, documentation, and extra information **~1 hour**
This part includes adding to slides, and also documentation/commenting the code so other people can view. I also created a guide on how to connect to the server using filezilla; however, have not made this public because of security concerns for our server. We essentially are using a ppk file to connect in using FILEZILLA SFTP.

Sprint 3 Documentation

Details: This sprint was an overhaul on the backend logic and integrating ORM, firebase, and leaderboards. The backend is being prepped for integrating the unity mobile game to communicate with website end using python, php, and unity backend. Bugs were also fixed. Refer below to all the work done…ALL COMMITS AND WORK ARE SEEN ON GITHUB UNDER https://github.com/JTBridges/Capstone-Defence and specific commits https://github.com/JTBridges/Capstone-Defence/commits?author=jagjotsingh1

    (1) Bug fixes on navbar for mobile were fixed **~1 hour**
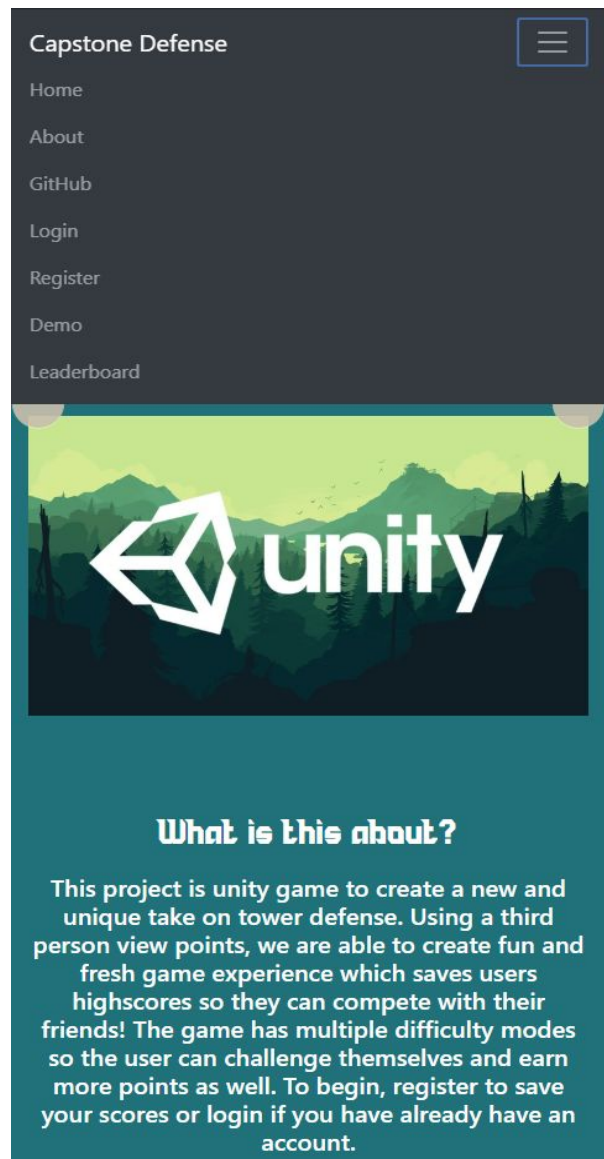
(2) Adding leaderboards table and some tweaks on website **~4 hours**
This part included connecting databases to the table to generate scores table for leaderboards. At the moment, they are dummy data generated in SQL but this part also included making some cosmetic changes to the website. The functions are ready to be implemented for unity data. Added leaderboards to own page, home page showing top 10, and profile page showing recent scores.

(3) Porting PHP to Python logic/queries **~7 hours**
This was a huge accomplishment this sprint because now we were able to write easier API and handle logic a lot better. As well, this adds another layer of security separating the backend from the front end. All PHP except a few logic here and there were ported to Python. Not sure how to show this, but please refer to the code in github commits.

(4) Firebase Research/Implementation + google authentication **~6 hours +**

I did a lot of research on firebase and whether it would work for our project. I tried to implement the noSQL into our website but continuously got errors to the point realized that the communication between noSQL and a relational database is pretty hard to implement. Also there really is no need for a noSQL database because that would mean scraping all of our current config in SQL to port over to noSQL config. Firebase handles setting up google auth backend but at this moment, I was able to implement google sign in to a degree but also realized its not really a necessity at this point. If we decide to go the google auth route that is fine, but firebase seems to be off the table in terms of noSQL configurations. I had the code included showing work I did on it but scrapped it out because it was making the code look messy.
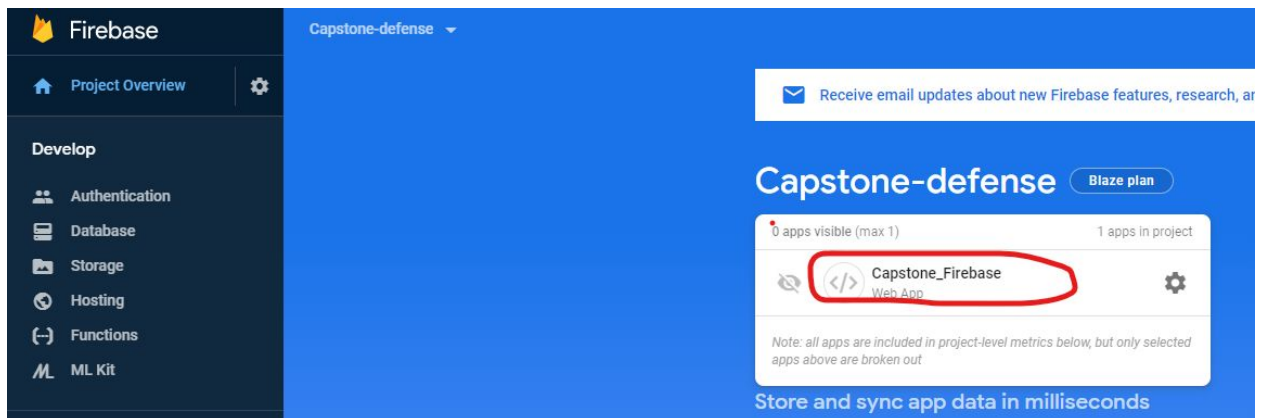
Sprint 4 Documentation

Details: This sprint was an overhaul on the backend logic and integrating ORM, firebase, and leaderboards. The backend is connected; however, the unity game is not ready to send scores data yet. An API was built to so the server side duties are done and just need to be implemented on client side. Bugs were also fixed, UI changes, navbar cleanup,etc . Refer below to all the work done…ALL COMMITS AND WORK ARE SEEN ON GITHUB UNDER https://github.com/JTBridges/Capstone-Defence and specific commits https://github.com/JTBridges/Capstone-Defence/commits?author=jagjotsingh1

(1) Bug fixes on navbar to show login status and logic handling **~2 hours**
   This part includes adding logic so once a user logins in, a session is started and if a session is active, clicking that button takes the user to account page. If a user logouts, the button returns back to the standard login button.

(2) Leaderboards connection to mySQL ported over to Python **~3 hours**

This part includes updating the dummy values held in the tables to pull data from DB. Every leaderboard has unique logic where the DB in index pulls top 10 scores while myAccount page pulls data for logged in user (top 5 scores). Leaderboards displays all data within the scores and ranks them.

(3) UI changes to website including button changes, page design redirects **~5 hours**
Simply made the website more user friendly and more appealing. After user logins, it takes the user to demo page, a logged in user can see their account info and change information as well. Added a confirm password in registration, have redirect links for 'need account?' 'Have account?'

(4) RESTAPI, and flask setup. Backend ready to go. **~12 hours**
This is what took the longest time this sprint. I essentially created our webAPI to handle POST requests sent to the server from our client machine. Kinks are being worked out in terms of security; however, the API is working and I am able to push data via url. This means our connection between server and game are almost complete.
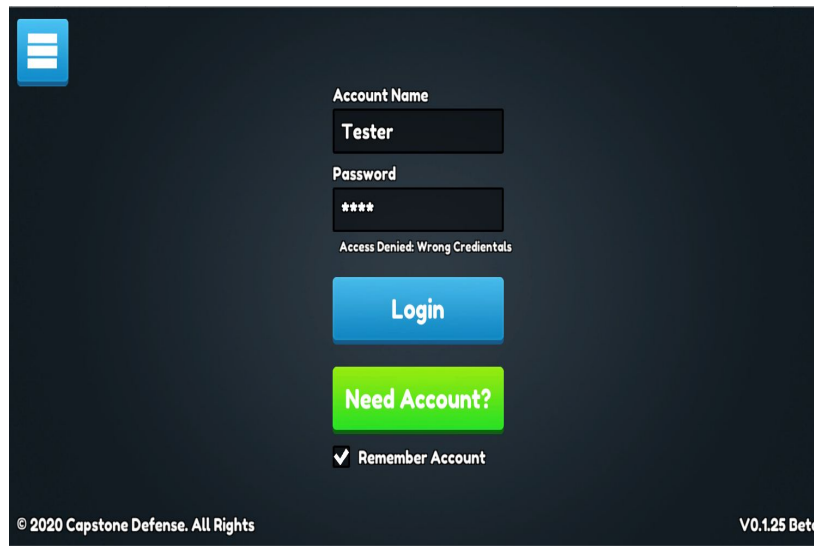
Sprint 5 Documentation

Details: This sprint was a bit slow in terms of progress but the major updates include creating the login screen and making the api fully functional for login and inserting scores https://github.com/JTBridges/Capstone-Defence and specific commits https://github.com/JTBridges/Capstone-Defence/commits?author=jagjotsingh1

    (1) Creating login screen for unity game ~6 hours

    (2) Creating login api and leaderboards functionality ~5 hours

    (3) Worked on presentation ~1 hour

Sprint 6 Documentation

Details: This sprint was again slow but I was able to get what I needed to get done interms of server side code, and unity 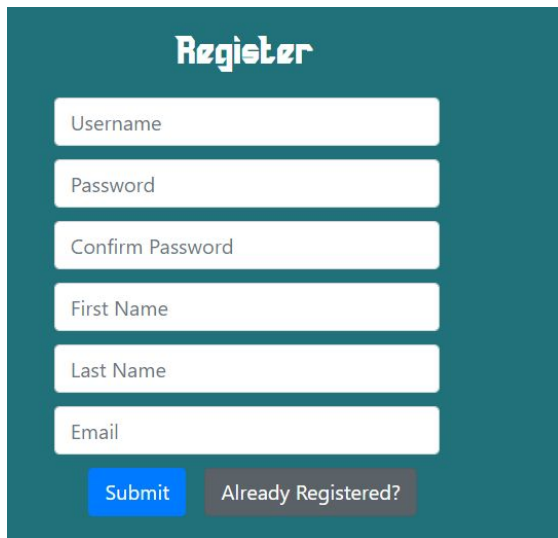leaderboards code. I also fixed a couple of bugs on website. https://github.com/JTBridges/Capstone-Defence and specific commits https://github.com/JTBridges/Capstone-Defence/commits?author=jagjotsingh1
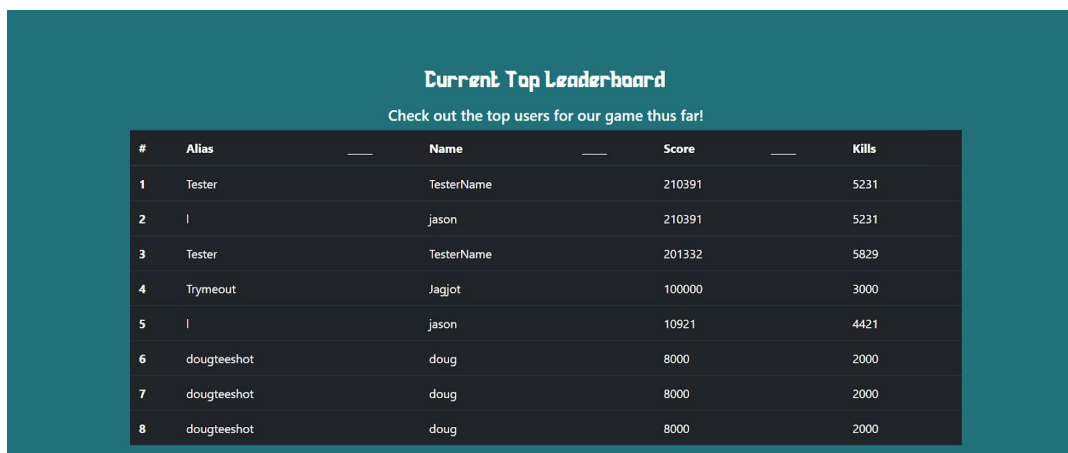
(1) Scores are added to end game via post request ~5-6 hours

(2) Bug fixes on website ~4 hours

Fixed already registered bug

Leaderboards security bug so locked users could post scores

(3) My part is essentially done, just doing some wrap up stuff on unity game and website **~1 hour**

I have been a bit sick through the last week or so. Its been tough to be ontop of my stuff such as submitting the logbook. I hope you can understand and not give me late penalty. Its bit of a medical emergency at the moment..