Activity 5: Sensors and IoT

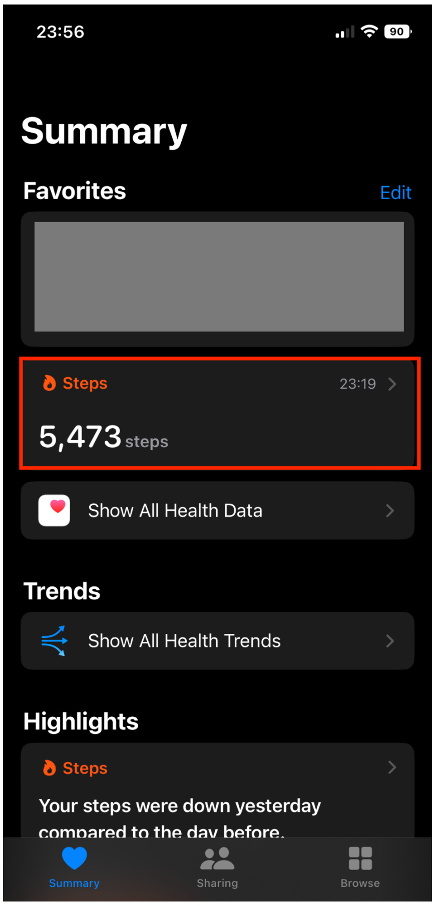
Group No : 35

Group Member :

1. Chotpisit Adunsehawat 6531313221
2. Sarin Yongsuwan 6532182521
3. Punyaphat Surakiatkamjorn 6432106821
4. Athiwat Kongkaeo 6531347621

Part 3: Pedometer Web Application

In this part, we will develop a “conceptual” IoT system in which each group creates a web application that serves as a “Pedometer ” (A portable device used for counting steps a person takes.)



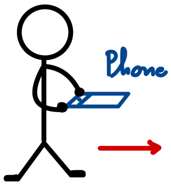
Your steps were down yesterday compared to the day before.

An example of application using mobile sensors to count steps in real life

Activity 5: Sensors and IoT (Part 3)

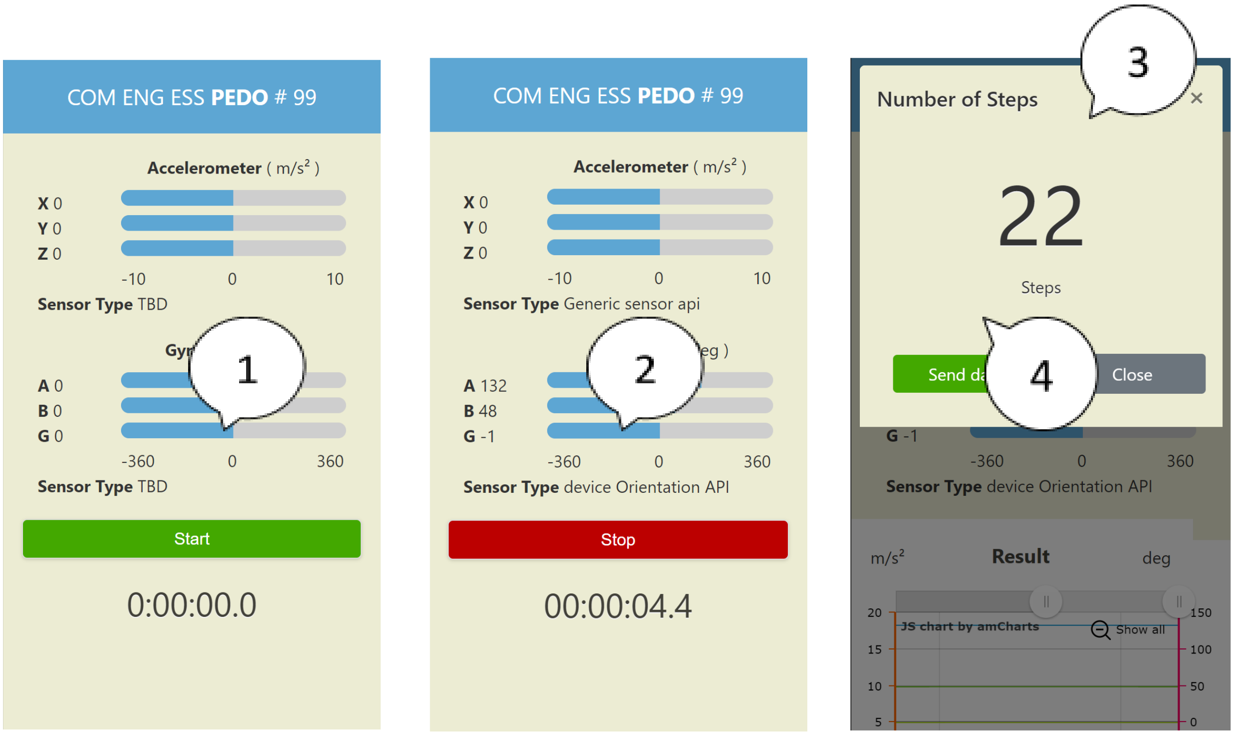
To achieve such an application, you are provided with a prototype web application; however, the provided application does not have any accuracy in calculating the correct number of steps right now.

For this part, you will have to deploy the pedometer application via Firebase with a developed algorithm for step counting. The application should be able to send the step data to a database server you have prepared in the Group Preparation Assignment and the data will also get displayed in a web UI.



Below is a scenario demonstrating how the app would be used.

1. The user clicks the **"Start"** button and starts walking.
2. While the user walks, the app reads values from the phone's accelerometer and gyroscope. The user clicks the **"Stop"** button when finishing the current walking session.
3. The app interprets sensors' data, calculates the number of steps the user walked during the session, and shows the number of steps on screen.
4. The user clicks the **"Send data"** button to send "number of steps" (integer) and "group number" (integer) to the server.



COM ENG ESS **PEDO** # 99

Accelerometer (m/s²)

X 0
Y 0
Z 0

-10 0 10

Sensor Type TBD

Gyr

A 0
B 0
G 0

-360 0 360

Sensor Type TBD

Start

0:00:00.0

COM ENG ESS **PEDO** # 99

Accelerometer (m/s²)

X 0
Y 0
Z 0

-10 0 10

Sensor Type Generic sensor api

eg)

A 132
B 48
G -1

-360 0 360

Sensor Type device Orientation API

Stop

00:00:04.4

Number of Steps ×

22

Steps

Send da Close

G -1

-360 0 360

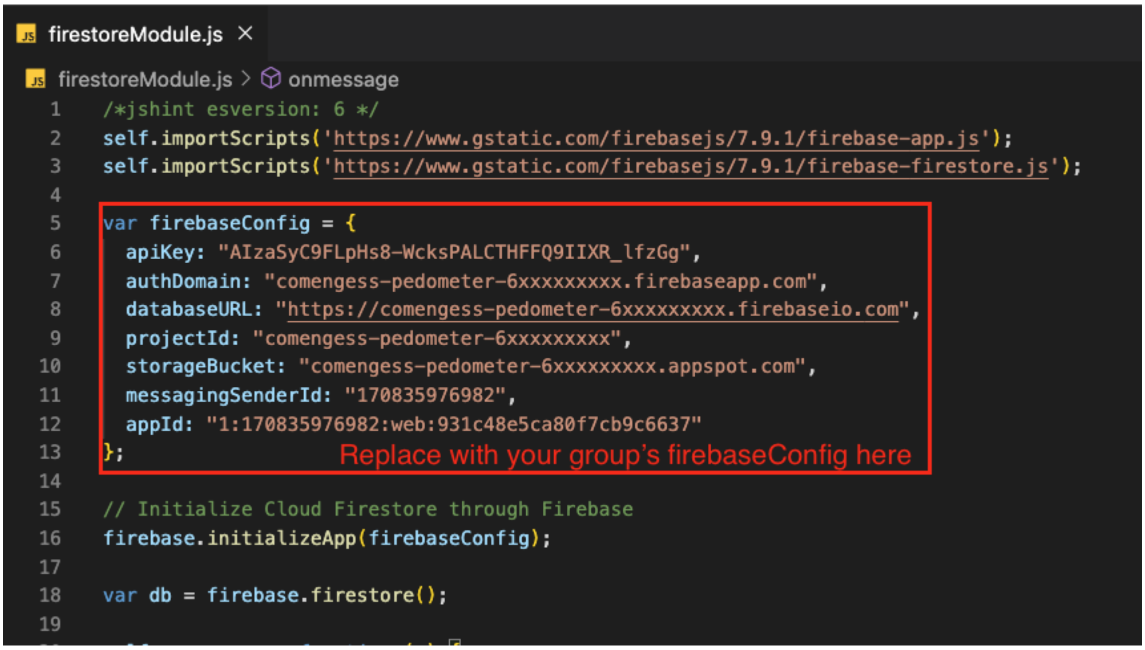
Sensor Type device Orientation API

m/s² **Result** deg

Activity 5: Sensors and IoT (Part 3)

Step 1: Download the prototype web application from the course material section on myCourseVille.

Step 2: *****IMPORTANT***** Copy the **firebaseConfig** variable from the submitted file (groupXX_config.txt) and paste it in the **firestoreModule.js** file (replace the old variable in that file). This will allow your application to know your Firebase Database Server's location, so it would be able to send data to your server.



```
/*jshint esversion: 6 */
self.importScripts('https://www.gstatic.com/firebasejs/7.9.1/firebase-app.js');
self.importScripts('https://www.gstatic.com/firebasejs/7.9.1/firebase-firestore.js');

var firebaseConfig = {
  apiKey: "AIzaSyC9FLpHs8-WcksPALCTHFFQ9IIXR_lfzGg",
  authDomain: "comengess-pedometer-6xxxxxxxxx.firebaseapp.com",
  databaseURL: "https://comengess-pedometer-6xxxxxxxxx.firebaseio.com",
  projectId: "comengess-pedometer-6xxxxxxxxx",
  storageBucket: "comengess-pedometer-6xxxxxxxxx.appspot.com",
  messagingSenderId: "170835976982",
  appId: "1:170835976982:web:931c48e5ca80f7cb9c6637"
};

// Initialize Cloud Firestore through Firebase
firebase.initializeApp(firebaseConfig);

var db = firebase.firestore();

```

Replace with your group's firebaseConfig here

Step 3: Install Firebase CLI on your local PC, set up Firebase Hosting and deploy the app.

1. **Install firebase cli**
   - Download the LTS version from https://nodejs.org/en/ (You can skip this if your computer already has Node.js installed.)
   - Type command provided below in your local PC's terminal. This command will install the Firebase Command Line Interface (firebase cli) globally on your local PC, which enables you with the globally accessible firebase commands.

     ```
     npm install -g firebase-tools
     ```
   - After you have installed firebase cli, you may recheck by typing command in your terminal:

     ```
     firebase -V
     ```

to ensure that the installation was completed. If the version of the firebase is displayed on your terminal (should be 11.23.1), then you are ready to proceed to the next step.

2. Login to your account via Firebase CLI

Type command provided below in your terminal.

```
firebase login --no-localhost
```

You should see something like this: Type Y and then press Enter.

```
i  Firebase optionally collects CLI and Emulator Suite usage and error
reporting information to help improve our products. Data is collected in
accordance with Google's privacy policy
(https://policies.google.com/privacy) and is not used to identify you.

? Allow Firebase to collect CLI and Emulator Suite usage and error reporting
information? (Y/n) Y
```

After pressing Enter, you will see a response like this:

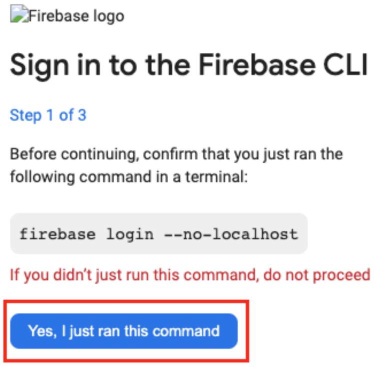
```
i  To change your data collection preference at any time, run `firebase
logout` and log in again.

To sign in to the Firebase CLI:

1. Take note of your session ID:

   [YOUR_SESSION_ID]

2. Visit the URL below on any device and follow the instructions to get your
code:

   [URL]

3. Paste or enter the authorization code below once you have it:

? Enter authorization code: [PASTE_AUTHORIZATION_CODE_HERE]
```

- Please observe your session ID *[YOUR_SESSION_ID]* at 1.
- Copy the *[URL]* from 2. Open your web browser and go to the link you have copied. You will be asked to sign in with your Google Account.

Activity 5: Sensors and IoT (Part 3)

IMPORTANT Please choose the account that has access to your Firebase project that your group has set up in the Group Preparation Assignment. After you choose your account, click **"Allow"** at the bottom to allow Firebase CLI to access your account.

- (Step 1 of 3), click **"Yes, I just ran this command"** to confirm that you are the one who just ran the login command on your terminal.



Firebase logo

Sign in to the Firebase CLI

Step 1 of 3

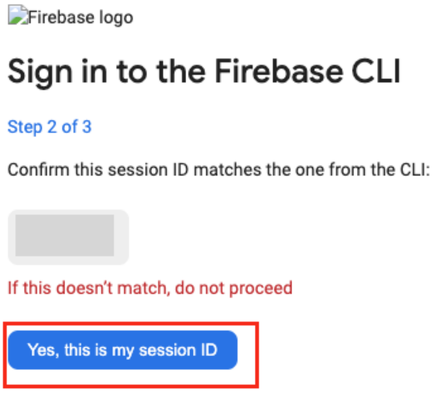
Before continuing, confirm that you just ran the following command in a terminal:

```
firebase login --no-localhost
```

If you didn't just run this command, do not proceed

Yes, I just ran this command

- (Step 2 of 3) Next, confirm that the session ID appeared on the webpage matches with your *[YOUR_SESSION_ID]* that you noted before. If yes, click **"Yes, this is my session ID"**.



Firebase logo

Sign in to the Firebase CLI

Step 2 of 3

Confirm this session ID matches the one from the CLI:

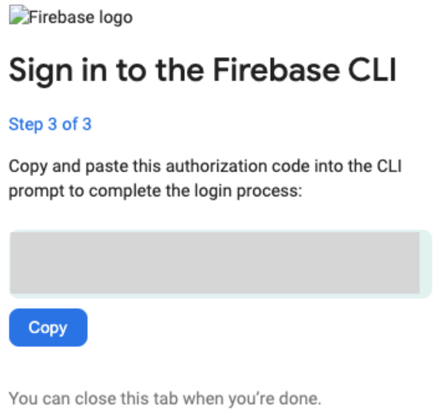
[redacted]

If this doesn't match, do not proceed

Yes, this is my session ID

Activity 5: Sensors and IoT (Part 3)

- (Step 3 of 3) Copy the *[AUTHORIZATION_CODE]* appeared on the webpage and then paste to your terminal, then hit Enter to complete the login process.



Firebase logo

Sign in to the Firebase CLI

Step 3 of 3

Copy and paste this authorization code into the CLI prompt to complete the login process:

Copy

You can close this tab when you're done.

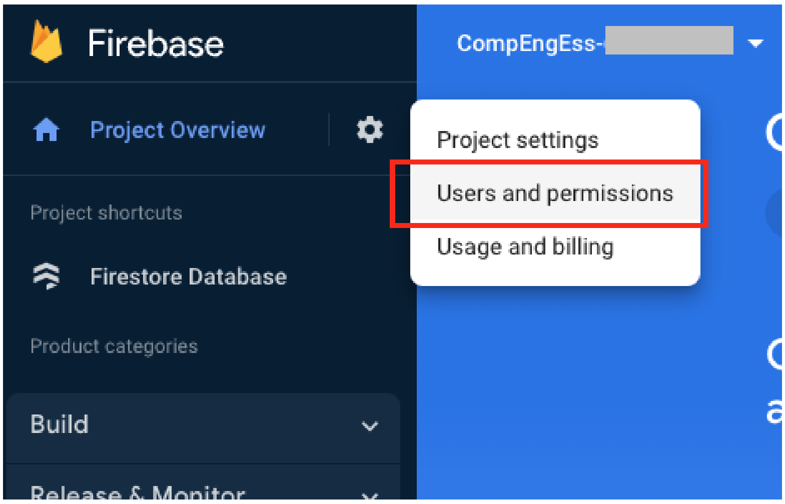
After enter the authorization code on your terminal, if you see something like this:

```
✔ Success! Logged in as [YOUR_GOOGLE_ACCOUNT_EMAIL]
```

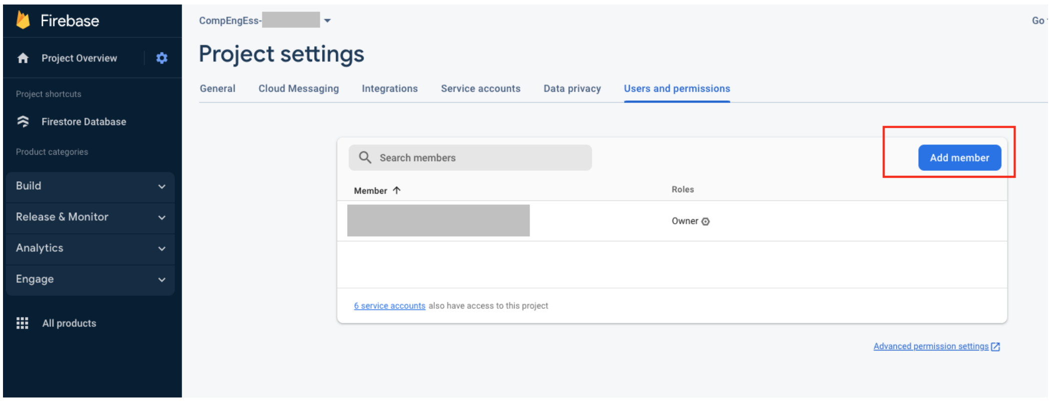
Then, congratulations you have successfully logged into your account via firebase cli on your local PC. You can proceed to the next step.

Note

- You may add other users (your group members) to access your firebase project by going to your firebase project's overview webpage. Then click **Users and permissions** as shown in the image below. Then click **"Add Member"**



Activity 5: Sensors and IoT (Part 3)



- If you have logged into the wrong account, please use this command to logout, and then try redoing this step again with the another account you prefer.

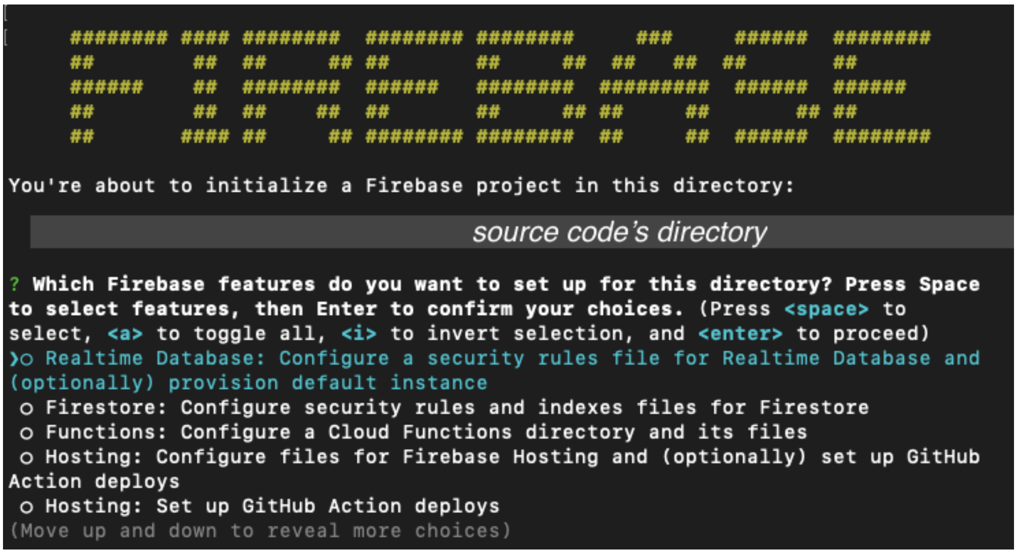
```
firebase logout
```

3. Init your app with firebase hosting

- ○ In your local PC's terminal, use **cd** command to navigate into the application's source code folder which you have downloaded from myCourseVille earlier.
- ○ Type command provided below in your terminal.

```
firebase init
```

You should see something like this,



```
     ######## #### ########  ######## ########     ###     ######  ########
     ##        ##  ##     ## ##       ##     ##  ##   ##  ##       ##
     ######    ##  ########  ######   ########  #########  ######  ######
     ##        ##  ##    ##  ##       ##     ## ##     ##       ## ##
     ##       #### ##     ## ######## ########  ##     ##  ######  ########

You're about to initialize a Firebase project in this directory:

                          source code's directory

? Which Firebase features do you want to set up for this directory? Press Space
to select features, then Enter to confirm your choices. (Press <space> to
select, <a> to toggle all, <i> to invert selection, and <enter> to proceed)
❯○ Realtime Database: Configure a security rules file for Realtime Database and
(optionally) provision default instance
 ○ Firestore: Configure security rules and indexes files for Firestore
 ○ Functions: Configure a Cloud Functions directory and its files
 ○ Hosting: Configure files for Firebase Hosting and (optionally) set up GitHub
Action deploys
 ○ Hosting: Set up GitHub Action deploys
(Move up and down to reveal more choices)
```

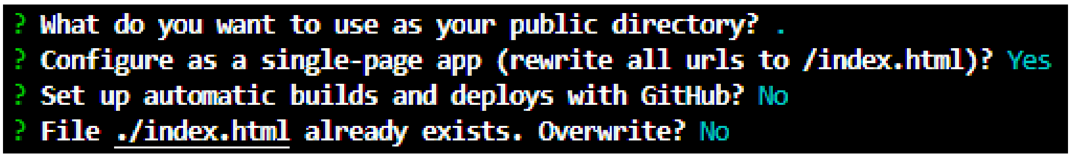
- ○ Then, select "**Hosting: Configure files for Firebase Hosting and (optionally) set up GitHub Action deploys**" by using arrow keys and pressing the **Spacebar** at that row, then press **Enter**.

Activity 5: Sensors and IoT (Part 3)

- Select **Use an existing project**, then select the name of your project **“comengess-6xxxxxxxxx”**, which you’ve already created in the Group Preparation assignment.

```
? Which Firebase features do you want to set up for this directory?
Press Space to select features, then Enter to
confirm your choices. Hosting: Configure files for Firebase Hosting
and (optionally) set up GitHub Action deploys

=== Project Setup

First, let's associate this project directory with a Firebase project.
You can create multiple project aliases by running firebase use --add,
but for now we'll just set up a default project.

? Please select an option: (Use arrow keys)
❯ Use an existing project
  Create a new project
  Add Firebase to an existing Google Cloud Platform project
  Don't set up a default project
```

- After you’ve selected the firebase project, Firebase CLI will ask you some questions about the hosting setup, ***IMPORTANT*** please type the answers as follows. (press **Enter** after finish typing answer in each questions):



```
? What do you want to use as your public directory? .
? Configure as a single-page app (rewrite all urls to /index.html)? Yes
? Set up automatic builds and deploys with GitHub? No
? File ./index.html already exists. Overwrite? No
```

- If you see something like this, then you can proceed to the next step.

```
i  Skipping write of ./index.html

i  Writing configuration info to firebase.json...
i  Writing project information to .firebaserc...
i  Writing gitignore file to .gitignore...

✔  Firebase initialization complete!
```

Activity 5: Sensors and IoT (Part 3)

4. Deploy your app with firebase hosting

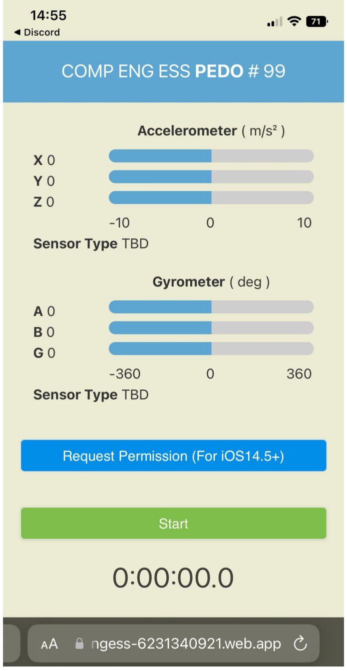
- Type command provided below in your terminal.

```
firebase deploy
```

You should see something like this,

```
=== Deploying to 'compengess-6xxxxxxxxx'...

i  deploying hosting
i  hosting[compengess-6xxxxxxxxx]: beginning deploy...
i  hosting[compengess-6xxxxxxxxx]: found 8 files in .
✔  hosting[compengess-6xxxxxxxxx]: file upload complete
i  hosting[compengess-6xxxxxxxxx]: finalizing version...
✔  hosting[compengess-6xxxxxxxxx]: version finalized
i  hosting[compengess-6xxxxxxxxx]: releasing new version...
✔  hosting[compengess-6xxxxxxxxx]: release complete

✔  Deploy complete!

Project Console: [PROJECT_OVERVIEW_URL]
Hosting URL: [HOSTING_URL]
```

- You can try this web application on any device using *[HOSTING_URL]*. If you can access the website on your own device, then you are ready to go to the next step.



14:55
◂ Discord

COMP ENG ESS **PEDO** # 99

Accelerometer (m/s²)

X 0
Y 0
Z 0

-10 0 10

Sensor Type TBD

Gyrometer (deg)

A 0
B 0
G 0

-360 0 360

Sensor Type TBD

Request Permission (For iOS14.5+)

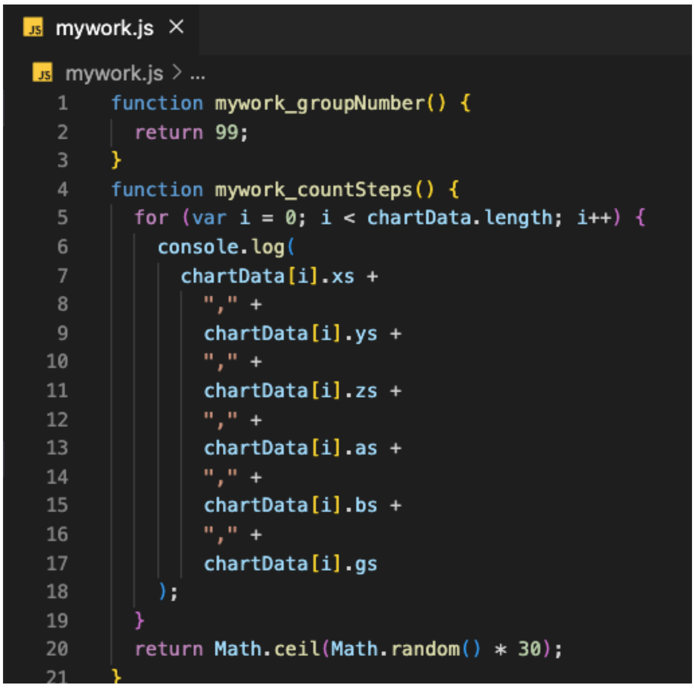
Start

0:00:00.0

AA ngess-6231340921.web.app

Activity 5: Sensors and IoT (Part 3)

Step 4: Your main focus for this part is to improve the step counting algorithm. Go to file **mywork.js** located in the source code's folder.



```
function mywork_groupNumber() {
  return 99;
}
function mywork_countSteps() {
  for (var i = 0; i < chartData.length; i++) {
    console.log(
      chartData[i].xs +
        "," +
        chartData[i].ys +
        "," +
        chartData[i].zs +
        "," +
        chartData[i].as +
        "," +
        chartData[i].bs +
        "," +
        chartData[i].gs
    );
  }
  return Math.ceil(Math.random() * 30);
}
```

- ** TODO ** There are currently 2 javascript functions in this file which are ***mywork_groupNumber()*** and ***mywork_Countsteps().*** Your job is to modify the code of both functions so that:
  - ***mywork_groupNumber()*** : returns your group number (integer).
  - ***mywork_countSteps()*** : returns the number of steps (integer) calculated based on the sensors' data. You can access the recorded sensors' data in the variable named **chartData** (already provided), which is an array of objects. In this array, each element is an object representing the data recorded from the sensors in a point of time in ascending order (each timestep takes 1/5 second). Here is an example of what data might look like in the chartData variable.

```
chartData = [
  {xs: 0, ys: -2, zs: -9, as: 0, bs: 0, gs: 0},
  {xs: -0.01, ys: -0.02, zs: 0, as: 0, bs: 2, gs: -1},
  {xs: 0, ys: -0.01, zs: 0.01, as: 0, bs: 2, gs: -1},
  {xs: -0.01, ys: -0.02, zs: 0, as: 0, bs: 2, gs: -1},
  {xs: 0.01, ys: 0.32, zs: 0, as: 355, bs: 2, gs: -1},
  …
]
```

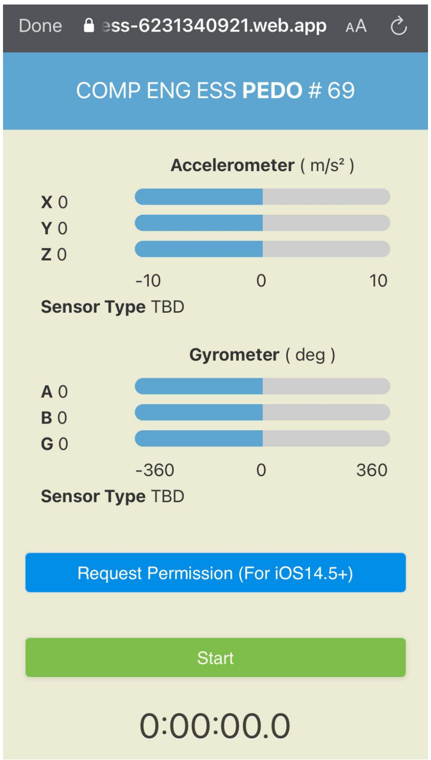
(xs = x, ys = y, zs = z, as = alpha, bs = beta, gs = gamma)

Activity 5: Sensors and IoT (Part 3)

Note You may edit or write additional functions in "mywork.js" only.
Modifying codes in other files is forbidden.

- After you've made some changes in "mywork.js", you need to run **'firebase deploy'** command again in order to deliver your changes into deployment. If you cannot see your changes, try removing the cache in your device's web browser first and then try accessing your group's URL again. We recommend that you may access your website in incognito/private mode as it doesn't store your web data in cache, so you can see your changes immediately after you've deployed.

- ****Requirements**** Your developed application should be able to:
  - display your group number on the navigation bar located on the top



(Here is an example if your group number is 69)

  - calculate the number of steps a user took when a user presses the "Stop" button. (it doesn't have to be 100% accurate, but at least you should deliver some potential ideas about how can you detect when a user take a step from the sensors' data)
  - send the data (number of steps) to your Firebase Database when a user presses the "Send Data" button

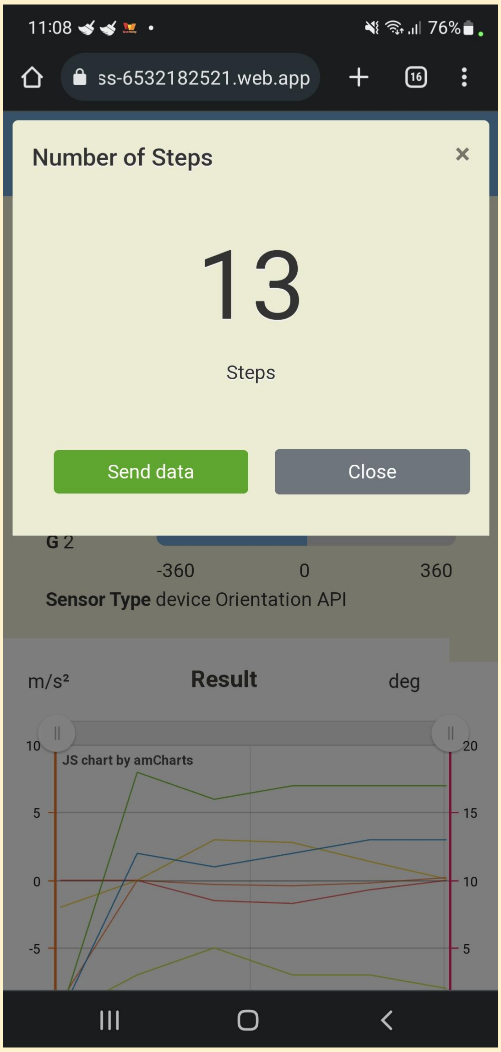
Activity 5: Sensors and IoT (Part 3)

Once you finish, students must inform an instructor or a TA for inspection. You will have to express your idea of your step counting algorithm and also have to demonstrate the walking and show that data is completely sent to your Firebase server.

- Your group pedometer web application's hosting URL

https://compengess-6532182521.web.app/

- Capture the screen of your web application on your mobile device to prove that your application is successfully deployed via firebase.



Number of Steps

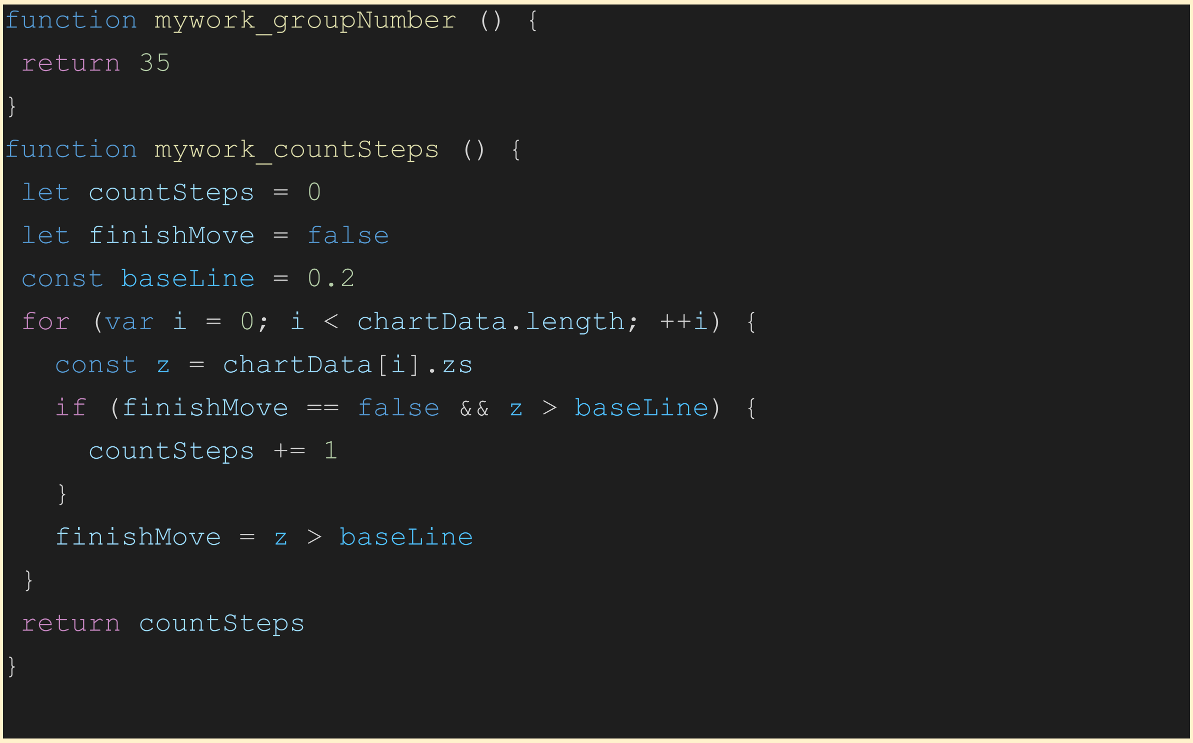
13

Steps

Sensor Type device Orientation API

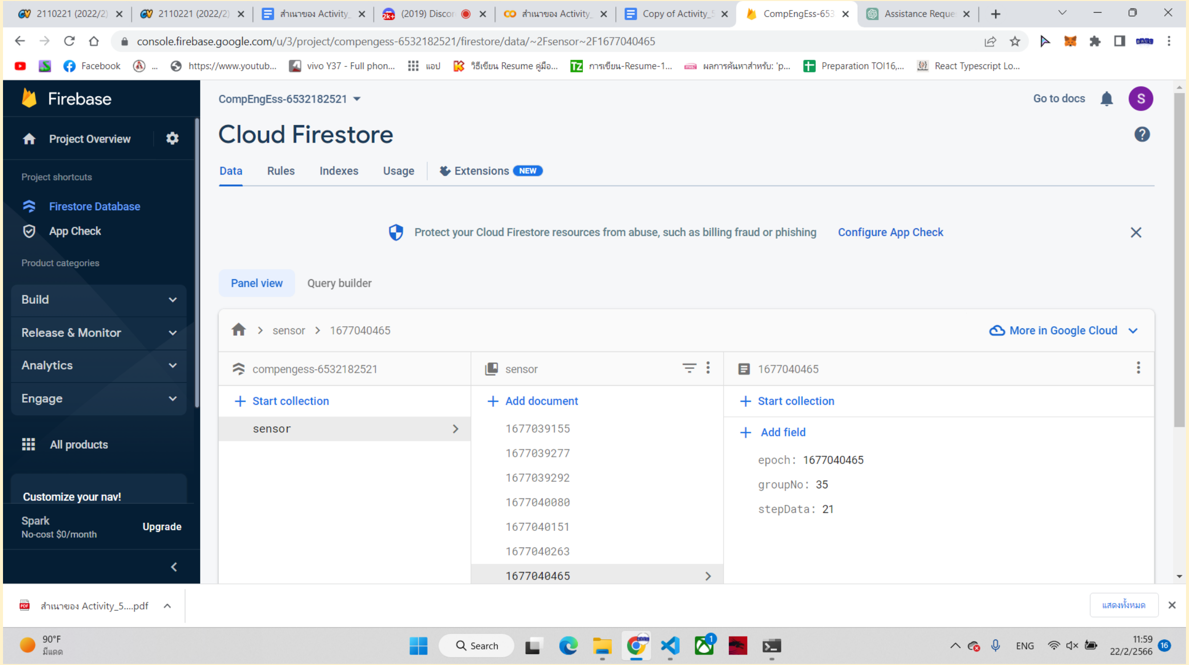
- Copy or capture the code that you modify and paste below. **(mywork.js)**

Activity 5: Sensors and IoT (Part 3)



```
function mywork_groupNumber () {
 return 35
}
function mywork_countSteps () {
 let countSteps = 0
 let finishMove = false
 const baseLine = 0.2
 for (var i = 0; i < chartData.length; ++i) {
   const z = chartData[i].zs
   if (finishMove == false && z > baseLine) {
     countSteps += 1
   }
   finishMove = z > baseLine
 }
 return countSteps
}
```

- Capture the screen that shows that the data is completely sent to the Firebase server.



console.firebase.google.com/u/3/project/compengess-6532182521/firestore/data/~2Fsensor~2F1677040465

CompEngEss-6532182521

Cloud Firestore

Protect your Cloud Firestore resources from abuse, such as billing fraud or phishing Configure App Check

sensor > 1677040465

compengess-6532182521	sensor	1677040465
+ Start collection	+ Add document	+ Start collection
sensor	1677039155	+ Add field
	1677039277	epoch: 1677040465
	1677039292	groupNo: 35
	1677040080	stepData: 21
	1677040151	
	1677040263	
	1677040465	

Activity 5: Sensors and IoT (Part 3)

— THIS IS THE END OF PART 3 —

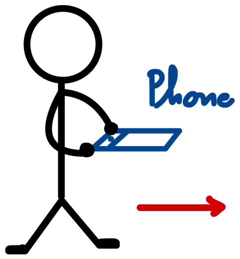
A Chance for the “Outstanding Factor”

Your main focus for the outstanding part is improving the algorithm for calculating the number of steps based on the sensors’ data more accurately.

Note:

- Some of the applications that have the most accurate number of steps count will be awarded this week “Outstanding Factor”.
- **You can call TA to test your application ONLY ONCE for the outstanding score**, so please check that your code and application work properly before calling TA.

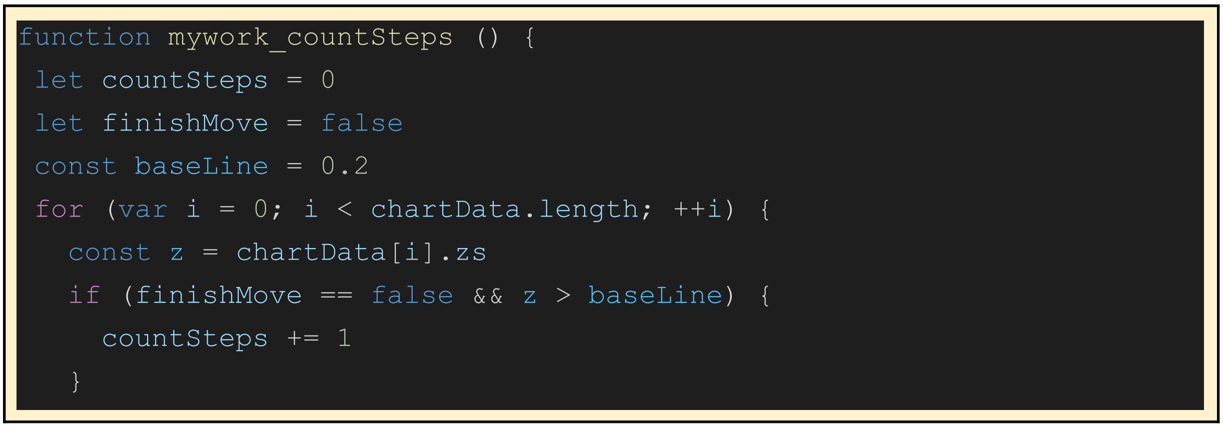
When you call TA to check your work, TA will hold **TA's** phone like the figure below, open your web application URL, and then start testing your application.



There will be **3 test cases** you should consider that your application should count the number of steps accurately.

1. Straight walking.
2. Straight walking halfway, turn around, and walk back.
3. Zigzag walking.

- Copy or capture the code that you improve and paste it below. **(mywork.js)**



```
function mywork_countSteps () {
 let countSteps = 0
 let finishMove = false
 const baseLine = 0.2
 for (var i = 0; i < chartData.length; ++i) {
   const z = chartData[i].zs
   if (finishMove == false && z > baseLine) {
     countSteps += 1
   }
```

```
    finishMove = z > baseLine
}
return countSteps
```

— THIS IS THE END OF THE OUTSTANDING PART —