240-203 CoE Software Lab II

2SA10 Firebase Cloud Function

Dr.Somchai LIMSIRORATANA

Department of Computer Engineering

Faculty of Engineering

Prince of Songkla University

Objective

1. Understand Event-Driven Serverless Computing
2. Understand Firebase Cloud Function
3. Can develop a simple cloud function application

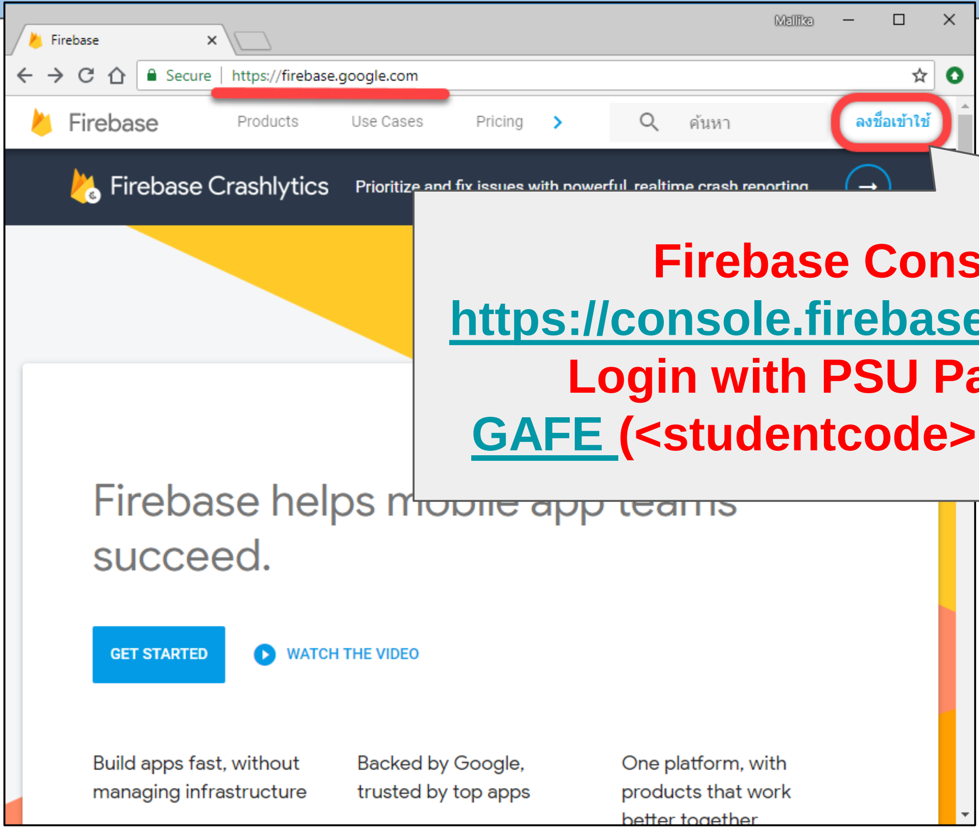
Scoring

- Attend = 20%
- Check Point #1 = 10%
- Check Point #2 = 20%
- Check Point #3 = 30%
- Check Point #4 = 20%, If possible

Outline

Setup Cloud Function Project

Sign-In Firebase Console



Firebase helps mobile app teams succeed.

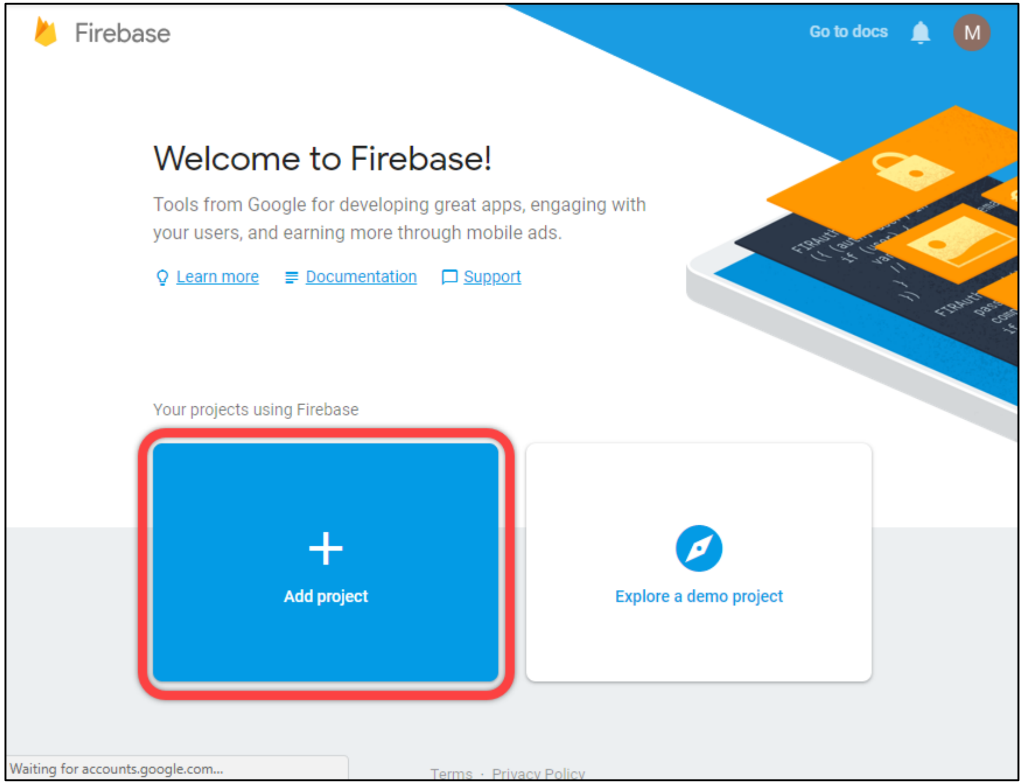
Build apps fast, without managing infrastructure

Backed by Google, trusted by top apps

One platform, with products that work better together

Firebase Console:
https://console.firebase.google.com
Login with PSU Passport:
GAFE (<studentcode>@psu.ac.th)

Create New Project



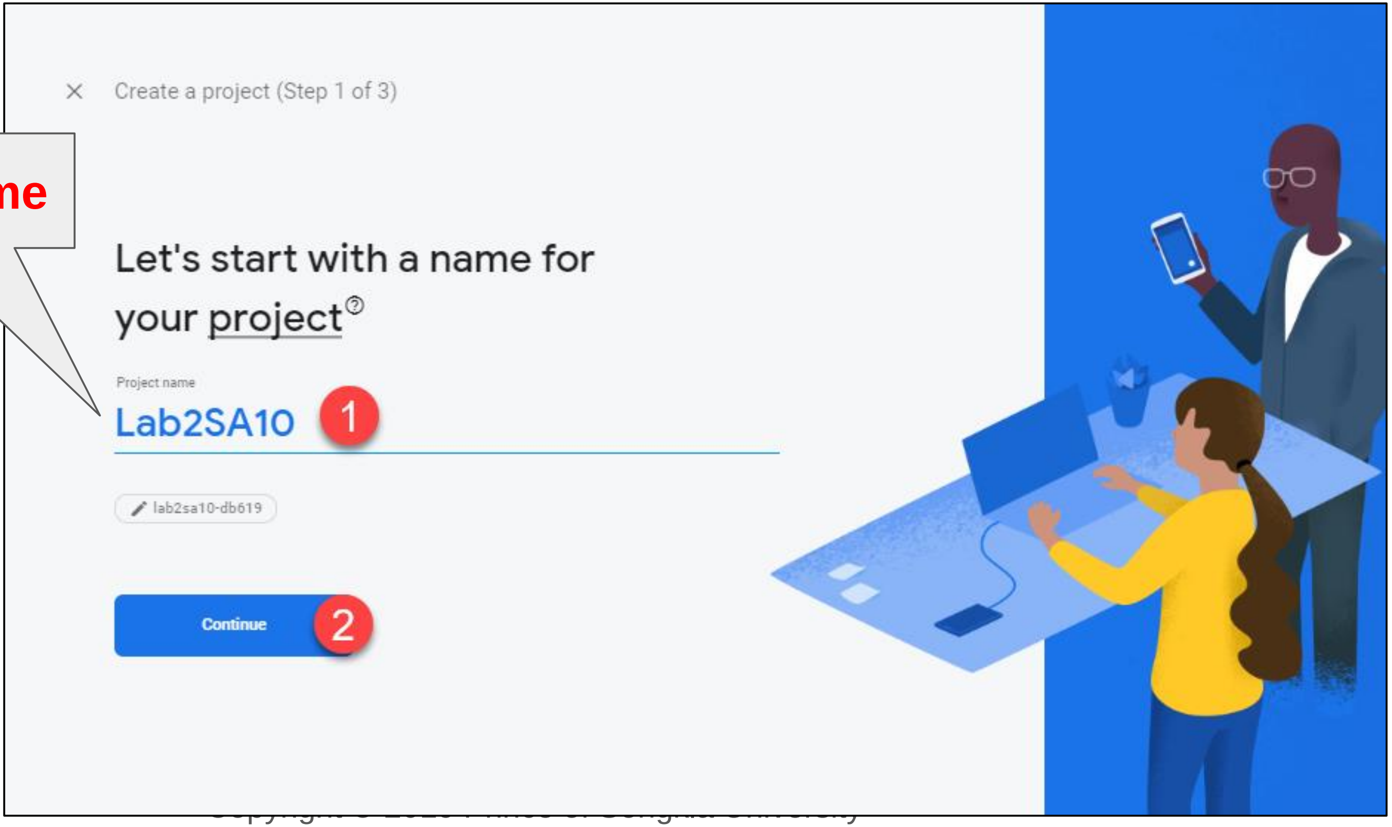
Welcome to Firebase!

Tools from Google for developing great apps, engaging with your users, and earning more through mobile ads.

Your projects using Firebase

Add Project

Project Name


Create a project (Step 1 of 3)

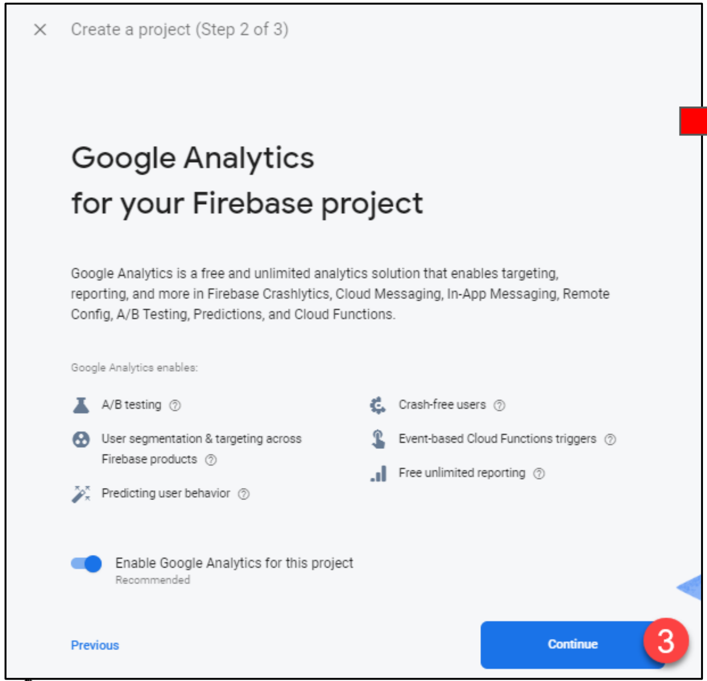
Let's start with a name for your project

Project name

Lab2SA10

lab2sa10-db619

Add Project



Create a project (Step 2 of 3)

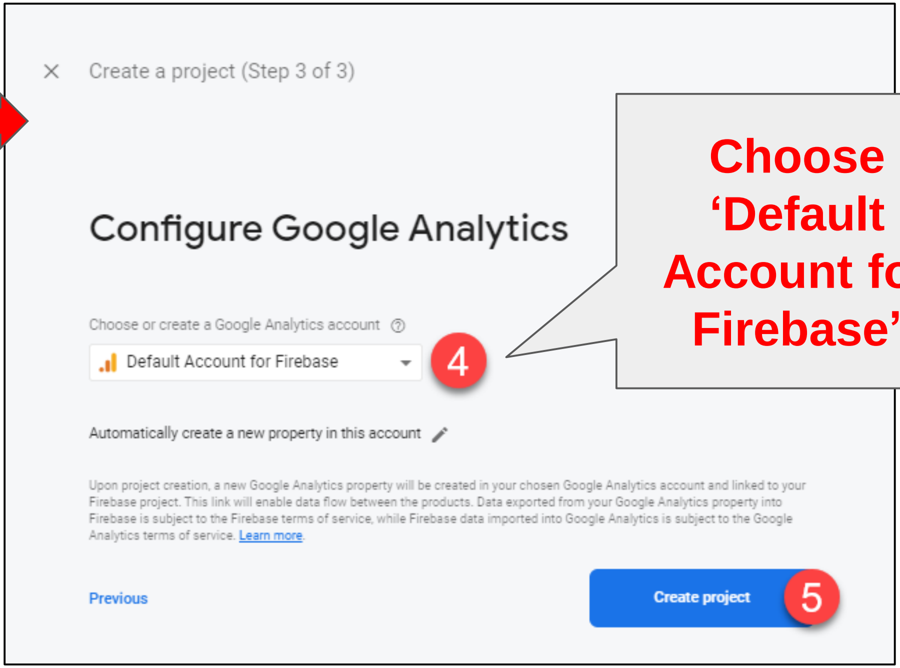
Google Analytics
for your Firebase project

Google Analytics is a free and unlimited analytics solution that enables targeting, reporting, and more in Firebase Crashlytics, Cloud Messaging, In-App Messaging, Remote Config, A/B Testing, Predictions, and Cloud Functions.

Google Analytics enables:

- A/B testing
- User segmentation & targeting across Firebase products
- Predicting user behavior
- Crash-free users
- Event-based Cloud Functions triggers
- Free unlimited reporting

Enable Google Analytics for this project
Recommended



Create a project (Step 3 of 3)

Configure Google Analytics

Choose or create a Google Analytics account

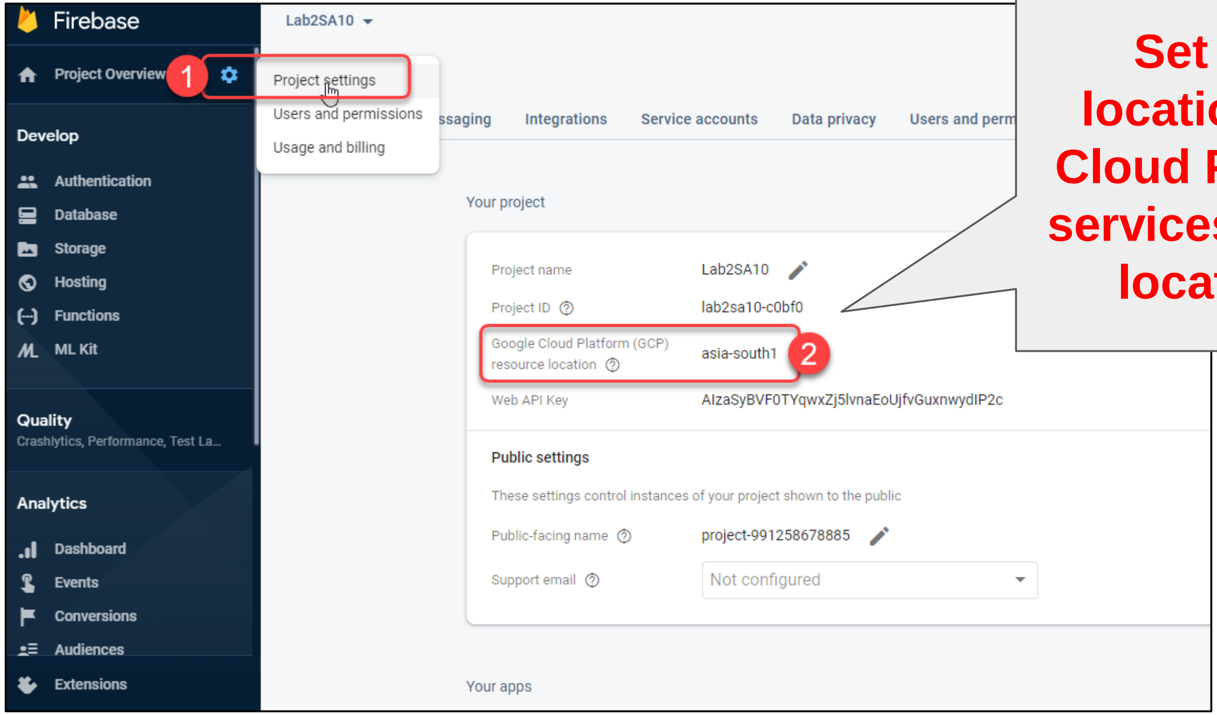
Default Account for Firebase

Automatically create a new property in this account

Upon project creation, a new Google Analytics property will be created in your chosen Google Analytics account and linked to your Firebase project. This link will enable data flow between the products. Data exported from your Google Analytics property into Firebase is subject to the Firebase terms of service, while Firebase data imported into Google Analytics is subject to the Google Analytics terms of service. Learn more.

Choose 'Default Account for Firebase'

Add Project



Your project

Project name	Lab2SA10
Project ID	lab2sa10-c0bf0
Google Cloud Platform (GCP) resource location	asia-south1
Web API Key	AIzaSyBVF0TYqwxZj5lvnaEoUjfvGuxnwydIP2c

Public settings

These settings control instances of your project shown to the public

Public-facing name	project-991258678885
Support email	Not configured

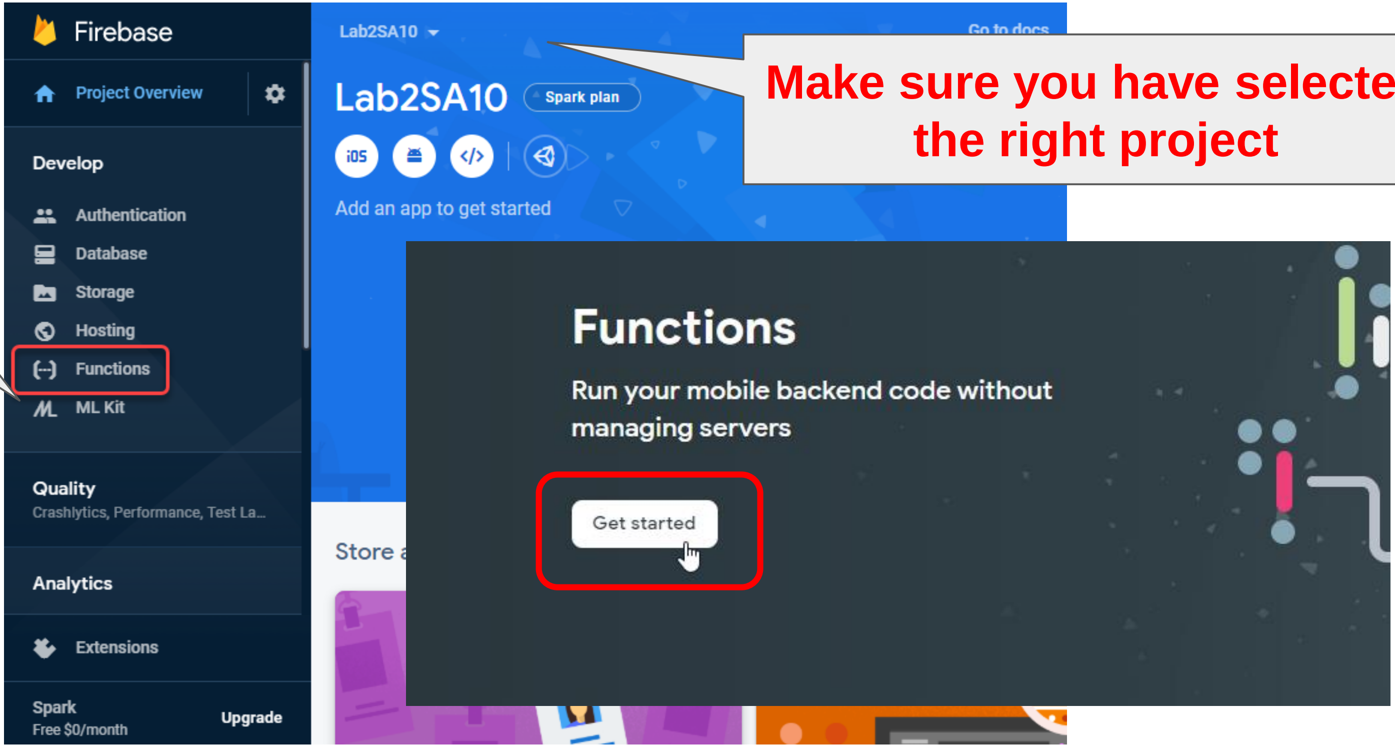
Your apps

Set the Default location for Google Cloud Platform (GCP) services that require a location setting.

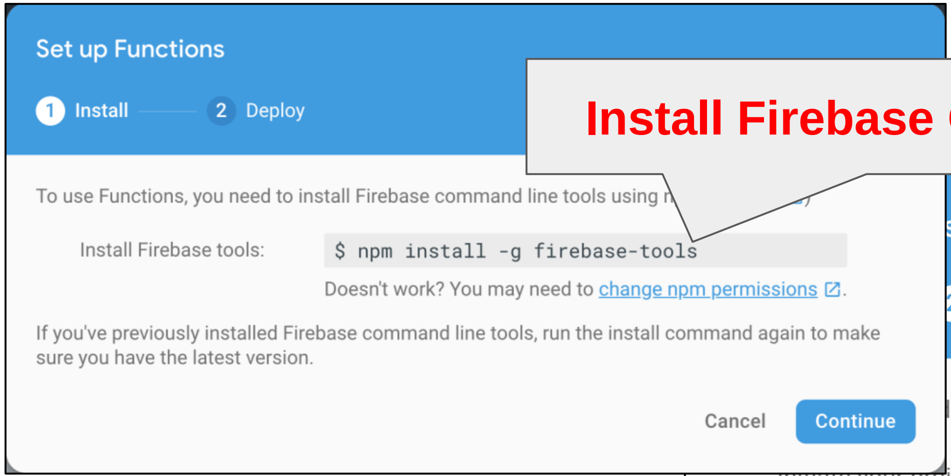
Start Function

Start Cloud Function

Make sure you have selected the right project



Setup Function



Set up Functions

1 Install 2 Deploy

To use Functions, you need to install Firebase command line tools using n

Install Firebase tools:

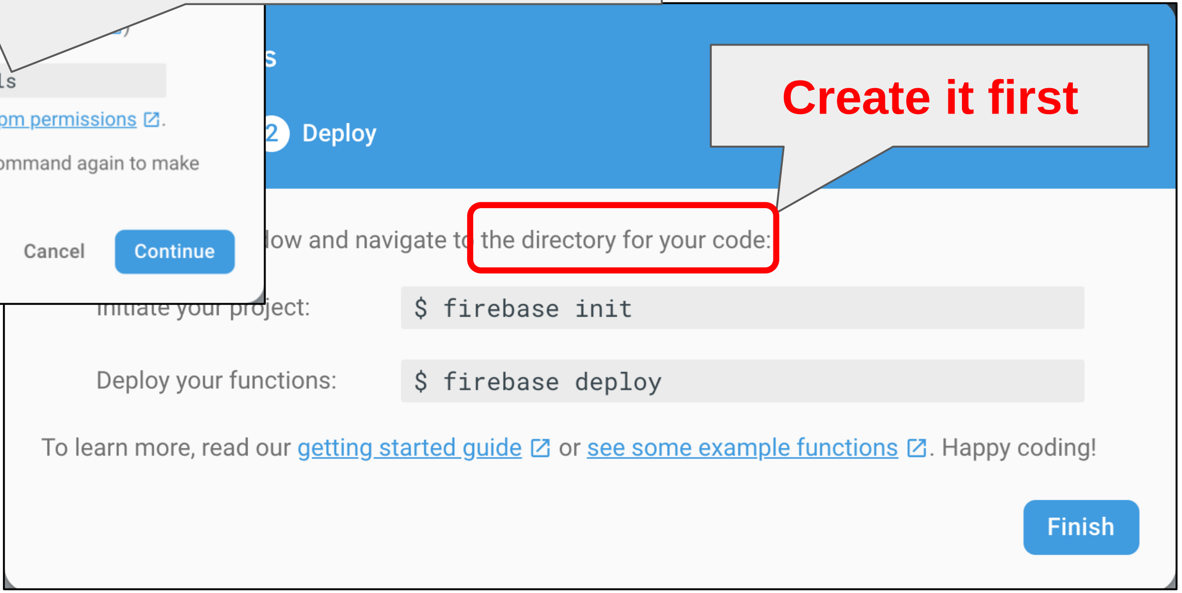
```
$ npm install -g firebase-tools
```

Doesn't work? You may need to change npm permissions.

If you've previously installed Firebase command line tools, run the install command again to make sure you have the latest version.

Cancel Continue

Install Firebase CLI on Node.js



2 Deploy

low and navigate to the directory for your code:

Initiate your project:

```
$ firebase init
```

Deploy your functions:

```
$ firebase deploy
```

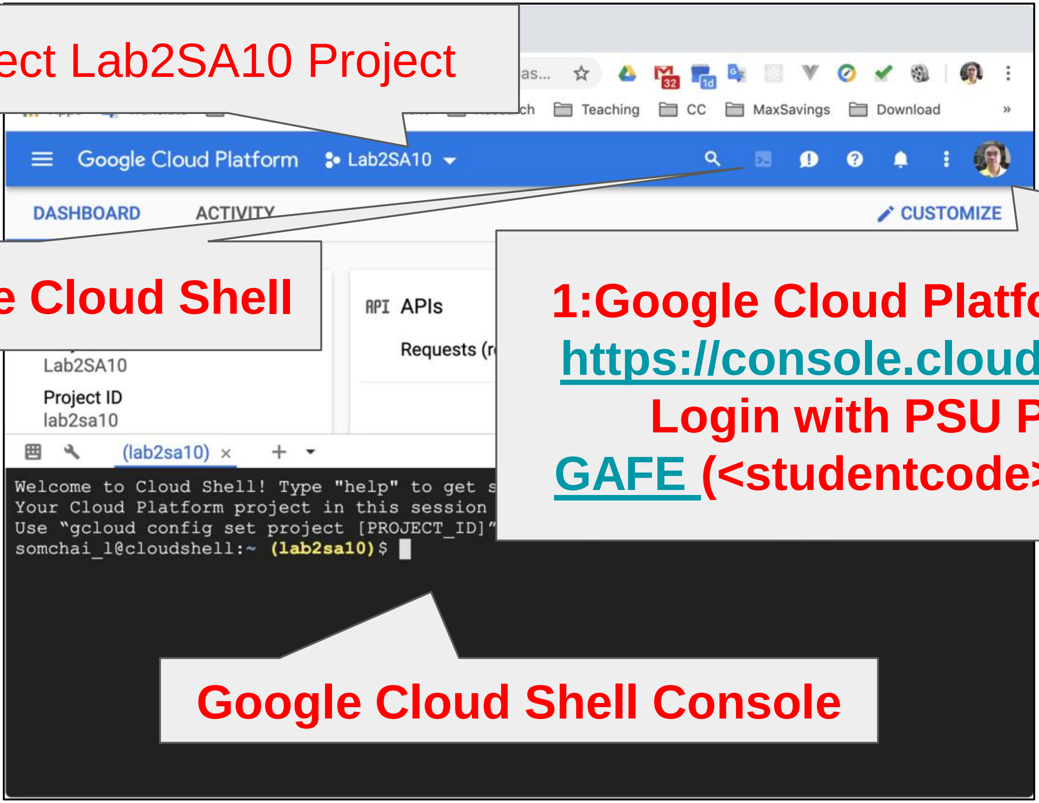
To learn more, read our getting started guide or see some example functions. Happy coding!

Finish

Create it first

Hello World Cloud Function

Open Google Cloud Shell



```
Welcome to Cloud Shell! Type "help" to get s
Your Cloud Platform project in this session
Use "gcloud config set project [PROJECT_ID]"
somchai_l@cloudshell:~ (lab2sa10)$
```

2:Select Lab2SA10 Project

3:Open Google Cloud Shell

1:Google Cloud Platform Console:
https://console.cloud.google.com
Login with PSU Passport:
GAFE (<studentcode>@psu.ac.th)

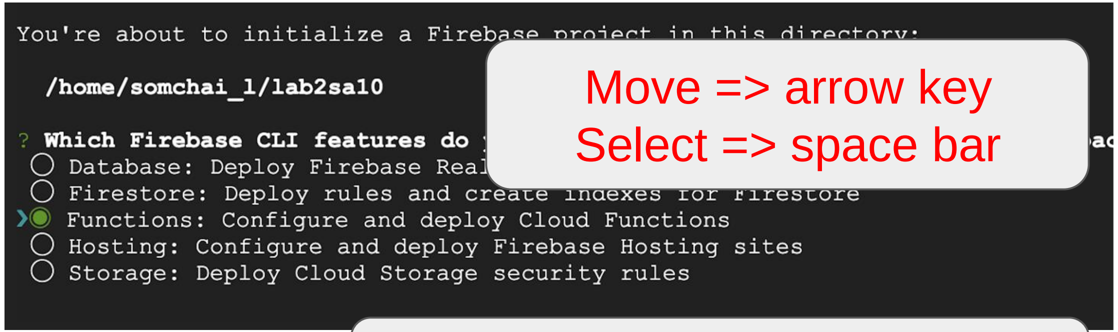
Google Cloud Shell Console

Check Node.js and Firebase Tools

- Node(>6): node -v
- Node(>6): npm -v
- Firebase Tool: firebase -V
    - Update: npm i -g firebase-tools

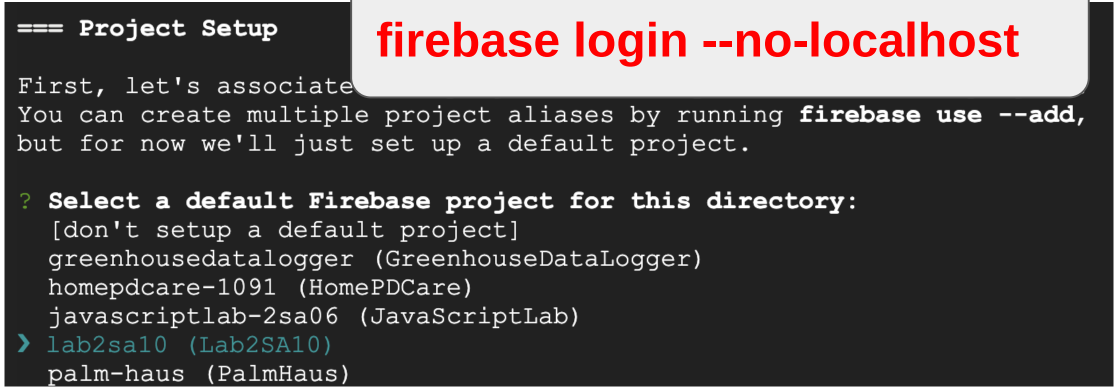
Initialize Cloud Functions

1. Create Directory:
   - mkdir lab2sa10
   - cd lab2sa10
2. Initialize:
   - firebase init
   - Select Function
   - Select Project
   - Select JavaScript
   - No,Yes



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

  /home/somchai_l/lab2sa10

? Which Firebase CLI features do 
  ◯ Database: Deploy Firebase Real
  ◯ Firestore: Deploy rules and create indexes for Firestore
❯◉ Functions: Configure and deploy Cloud Functions
  ◯ Hosting: Configure and deploy Firebase Hosting sites
  ◯ Storage: Deploy Cloud Storage security rules
```

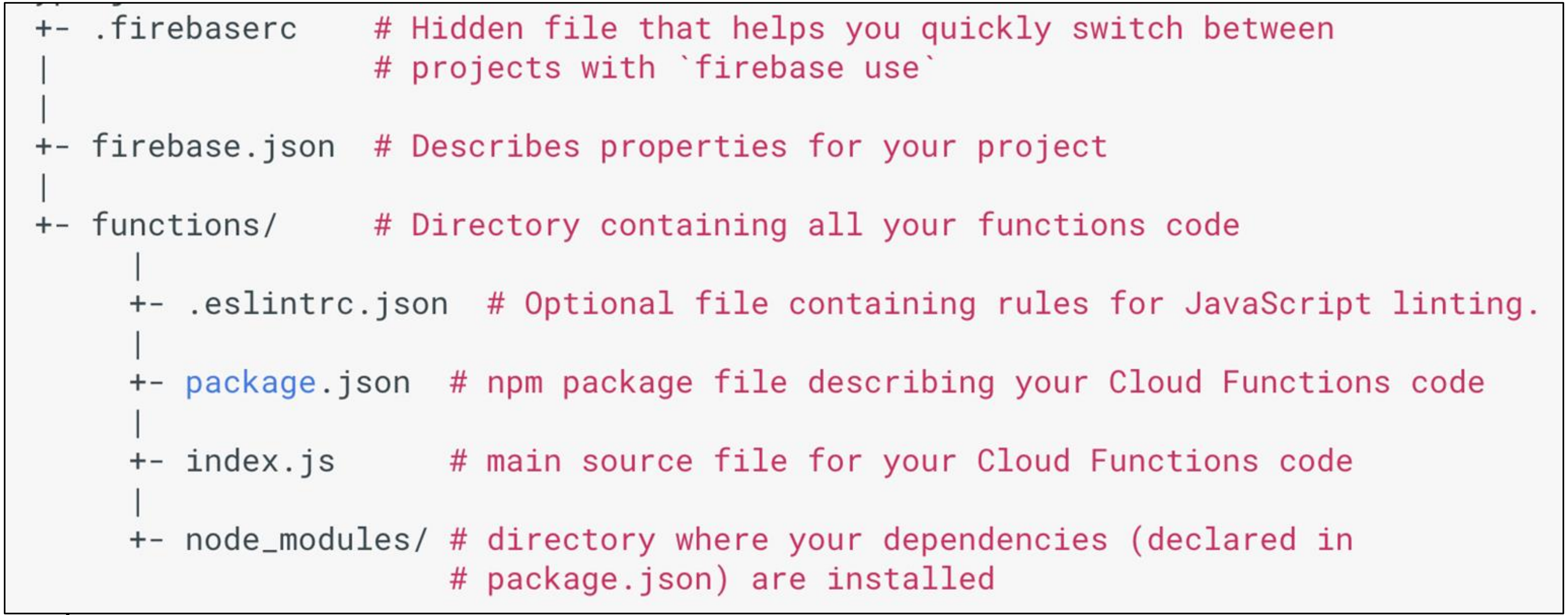
Move => arrow key
Select => space bar



```
=== Project Setup

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

? Select a default Firebase project for this directory:
  [don't setup a default project]
  greenhousedatalogger (GreenhouseDataLogger)
  homepdcare-1091 (HomePDCare)
  javascriptlab-2sa06 (JavaScriptLab)
❯ lab2sa10 (Lab2SA10)
  palm-haus (PalmHaus)
```

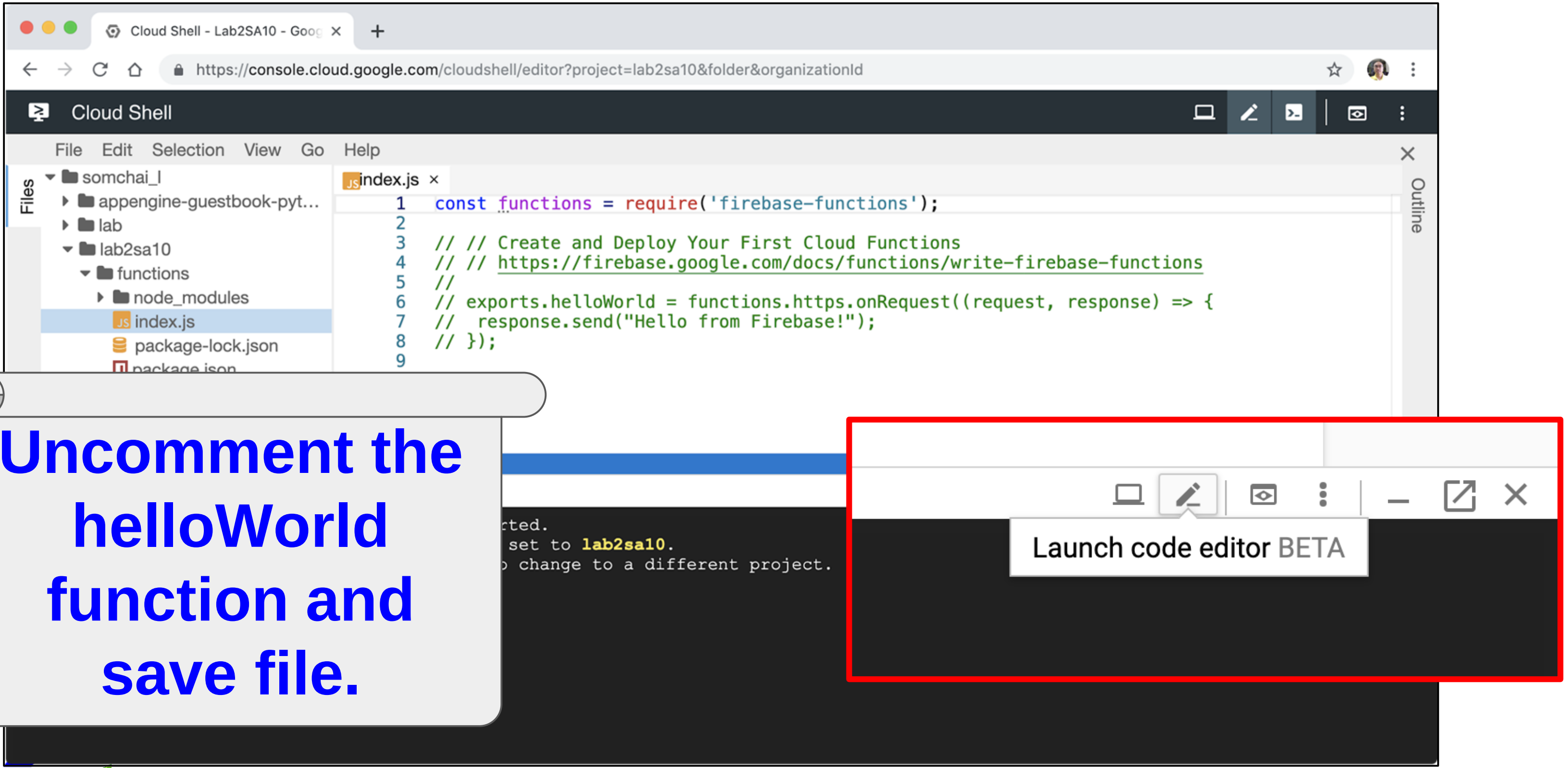
firebase login --no-localhost

Project Structure



```
 +- .firebaserc    # Hidden file that helps you quickly switch between
 |                 # projects with `firebase use`
 |
 +- firebase.json  # Describes properties for your project
 |
 +- functions/     # Directory containing all your functions code
      |
      +- .eslintrc.json  # Optional file containing rules for JavaScript linting.
      |
      +- package.json  # npm package file describing your Cloud Functions code
      |
      +- index.js      # main source file for your Cloud Functions code
      |
      +- node_modules/ # directory where your dependencies (declared in
                       # package.json) are installed
```

Edit index.js with Code Editor



Cloud Shell - Lab2SA10 - Goog
https://console.cloud.google.com/cloudshell/editor?project=lab2sa10&folder&organizationId
Cloud Shell
File Edit Selection View Go Help
Files
somchai_l
appengine-guestbook-pyt...
lab
lab2sa10
functions
node_modules
index.js
package-lock.json
package.json
Outline
Launch code editor BETA

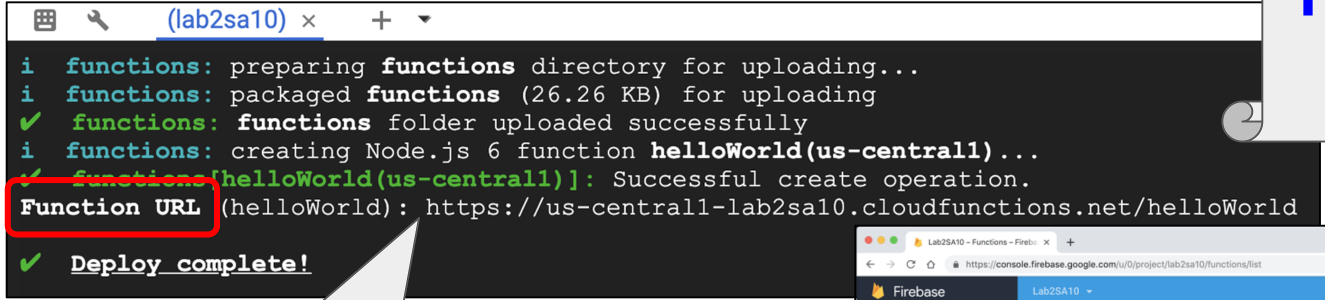
```
const functions = require('firebase-functions');

// // Create and Deploy Your First Cloud Functions
// // https://firebase.google.com/docs/functions/write-firebase-functions
//
// exports.helloWorld = functions.https.onRequest((request, response) => {
//  response.send("Hello from Firebase!");
// });

```

```
rted.
 set to lab2sa10.
 change to a different project.
```

Uncomment the helloWorld function and save file.

Deploy and Execute

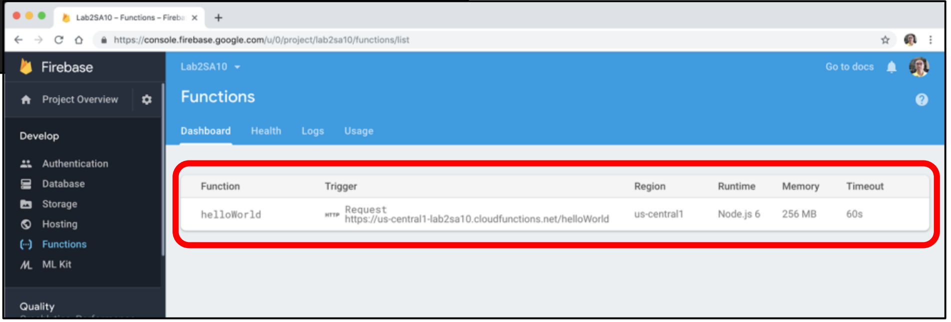
- Deploy: (in the project directory)
  - firebase deploy --only functions



```
i  functions: preparing functions directory for uploading...
i  functions: packaged functions (26.26 KB) for uploading
✔  functions: functions folder uploaded successfully
i  functions: creating Node.js 6 function helloWorld(us-central1)...
✔  functions[helloWorld(us-central1)]: Successful create operation.
Function URL (helloWorld): https://us-central1-lab2sa10.cloudfunctions.net/helloWorld

✔  Deploy complete!
```

Open the Function URL on browser

URL for HTTP function endpoints



Function	Trigger	Region	Runtime	Memory	Timeout
helloWorld	HTTP Request https://us-central1-lab2sa10.cloudfunctions.net/helloWorld	us-central1	Node.js 6	256 MB	60s

Understanding Code

```
// The Cloud Functions for Firebase SDK to create Cloud Functions and setup triggers.
const functions = require('firebase-functions');

exports.helloWorld = functions.https.onRequest((request, response) => {
        response.send("Hello from Firebase!");
});
```

onRequest -> HTTPS
onCall -> HTTPS + Authentication + FCM tokens + Deserializes

Cloud Function - Realtime Database

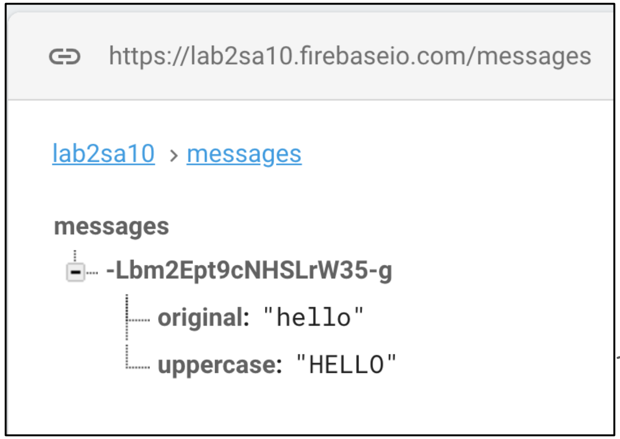
Check Point #1

```
const functions = require('firebase-functions');
// The Firebase Admin SDK to access the Firebase Realtime Database.
const admin = require('firebase-admin');
admin.initializeApp();
// Take the text parameter passed to this HTTP endpoint and insert it into the
// Realtime Database under the path /messages/:pushId/original
exports.addMessage = functions.https.onRequest((req, res) => {
  // Grab the text parameter.
  const original = req.query.text;
  // Push the new message into the Realtime Database using the Firebase Admin SDK.
  return admin.database().ref('/messages').push({original: original}).then((snapshot) => {
    // Redirect with 303 SEE OTHER to the URL of the pushed object in the Firebase console.
    return res.redirect(303, snapshot.ref.toString());
  });
});
// Listens for new messages added to /messages/:pushId/original and creates an
// uppercase version of the message to /messages/:pushId/uppercase
exports.makeUppercase = functions.database.ref('/messages/{pushId}/original')
    .onCreate((snapshot, context) => {
      // Grab the current value of what was written to the Realtime Database.
      const original = snapshot.val();
      console.log('Uppercasing', context.params.pushId, original);
      const uppercase = original.toUpperCase();
      // You must return a Promise when performing asynchronous tasks inside a Functions such as writing to the Realtime Database.
      // Setting an "uppercase" sibling in the Realtime Database returns a Promise.
      return snapshot.ref.parent.child('uppercase').set(uppercase);
    });
```

1. **Change the index.js file with this code.**
2. **Re-Deploy Project**
3. **Execute addMessage function**

Check Point #1: Hint

- Create realtime database in Firebase with nothing.
- Execute by passing message to text
  - https://<function host>/addMessage?text=uppercasemetoo



https://lab2sa10.firebaseio.com/messages

lab2sa10 › messages

```
messages
 -Lbm2Ept9cNHSLrW35-g
     original: "hello"
     uppercase: "HELLO"
```

Expected result will shown in Realtime Database in Firebase Console.

Understanding Code

```
// The Firebase Admin SDK to access the Firebase Realtime Database.
const admin = require('firebase-admin');
admin.initializeApp();

// Take the text parameter passed to this HTTP endpoint and insert it into the
exports.addMessage = functions.https.onRequest((req, res) => {
  const original = req.query.text; // Grab the text parameter.
  // Push the new message into the Realtime Database using the Firebase Admin SDK.
  return admin.database().ref('/messages').push({original: original}).then((snapshot) => {
    // Redirect (303) SEE OTHER to the URL of the pushed object in the Firebase console.
    return res.redirect(303, snapshot.ref.toString());
  });
});
```

Understanding Code

```
// Listens for new messages added to /messages/:pushId/original
exports.makeUppercase = functions.database.ref('/messages/{pushId}/original')
    .onCreate((snapshot, context) => {
      // Grab the current value of what was written to the Realtime Database.
      const original = snapshot.val();
      console.log('Uppercasing', context.params.pushId, original);
      const uppercase = original.toUpperCase();

      // writing to the Firebase Realtime Database.
      // Setting an "uppercase" sibling in the Realtime Database returns a Promise.
      return snapshot.ref.parent.child('uppercase').set(uppercase);
    });
```

Call Functions from App

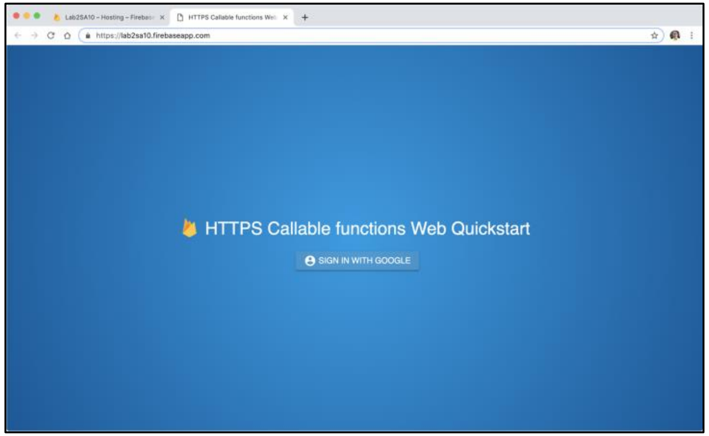
Callable Function

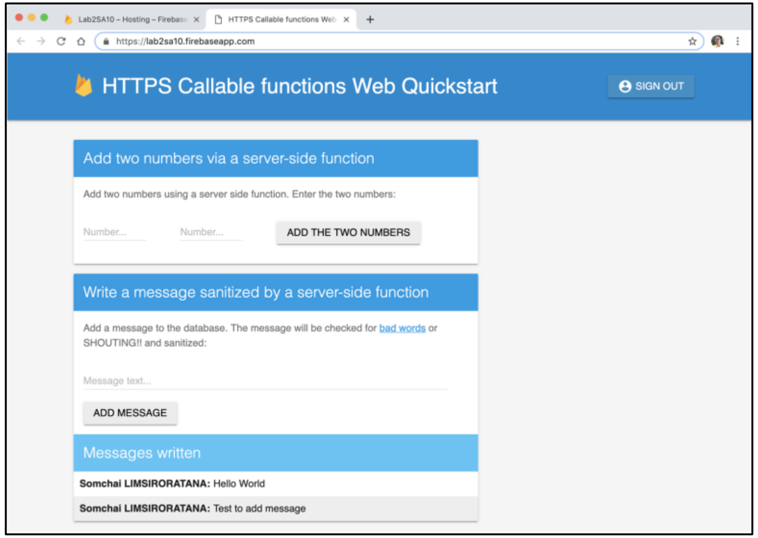
- The functions.https.onCall
  - Similar to other HTTP functions
  - Directly call from Firebase App
- Automatic included in requests:
  - Firebase Authentication
  - FCM tokens
- Automatic deserializes the request body
- Automatic validates auth tokens

Firebase (client) SDK on Android, iOS, Web, C++, Node.js, Java, Go, Python and Unity

Check Point #2

- Download example source code here.
- Upload/unzip to replace previous project folder.
- Deploy and Execute.


Lab2SA10 – Hosting – Firebase
HTTPS Callable functions Web
https://lab2sa10.firebaseapp.com
HTTPS Callable functions Web Quickstart
SIGN IN WITH GOOGLE


Lab2SA10 – Hosting – Firebase
HTTPS Callable functions Web
https://lab2sa10.firebaseapp.com
HTTPS Callable functions Web Quickstart
SIGN OUT
Add two numbers via a server-side function
Number...
Number...
ADD THE TWO NUMBERS
Write a message sanitized by a server-side function
Message text...
ADD MESSAGE
Messages written

Add two numbers using a server side function. Enter the two numbers:

Add a message to the database. The message will be checked for bad words or SHOUTING!! and sanitized:

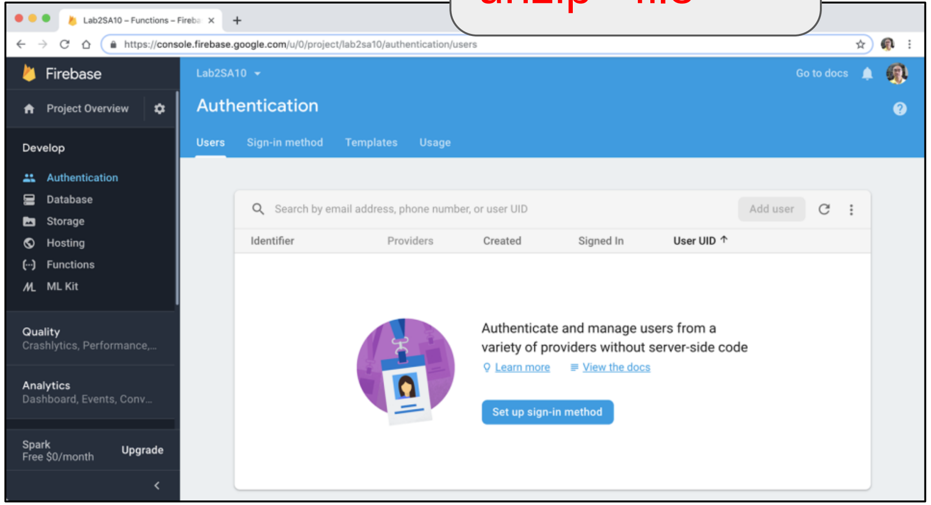
Somchai LIMSIRORATANA: Hello World

Somchai LIMSIRORATANA: Test to add message

Check Point #2: Hint

- Install dependency
  - cd function
  - npm install
- Setup Sign-In Method:
  - Firebase Authentication
  - Enable Google
- Deploy
  - Function & Hosting

cd <project2>
unzip <file>


Lab2SA10 – Functions – Fireba
https://console.firebase.google.com/u/0/project/lab2sa10/authentication/users
Firebase
Project Overview
Develop
Authentication
Database
Storage
Hosting
Functions
ML Kit
Quality
Crashlytics, Performance,...
Analytics
Dashboard, Events, Conv...
Spark
Free $0/month
Upgrade
Lab2SA10
Go to docs
Authentication
Users
Sign-in method
Templates
Usage
Search by email address, phone number, or user UID
Add user
Identifier
Providers
Created
Signed In
User UID
Authenticate and manage users from a variety of providers without server-side code
Learn more
View the docs
Set up sign-in method

Final Tasks

Final Tasks

- Check Point #3
  - Add more functions
    - Sub
    - Multiply
    - Divide
  - Modify Web App UI to use functions
    - index.html
    - scripts/main.js

- Check Point #4
  - Build Android Version From GitHub
  - Modify UI to use more 3 functions from Check Point #3

Check Point #4: Hint

- Download source code
  - From main project (quickstart-android), Clone or Download
- Using Android Studio open functions folder
- Run for testing first
- Modify for Sub,Multiply,Divide by copy from Add
  - UI File
    - app/src/main/res/layout/activity_main.xml
  - Method File
    - app/src/main/java/com/google/samples/quickstart/functions/java/MainActivity.java

Check Point

- Chk1 : https://www.youtube.com/watch?v=5wftZ6A7hwo&fbclid=IwAR0qBhgygeff6pqXMzzS1ZIM_WzvGrdOq6OIalAlPAzQjMlSoetDeYwrlpc
- Chk2 : https://www.youtube.com/watch?v=eF4jFhDSbx0&fbclid=IwAR3GspPcCV-HpFr5floIsJp6PND82antn9ENO6q72dCVQSzQCxcLUcvXpKw
- Chk3 : https://www.youtube.com/watch?v=vOM4AWd_NvQ&fbclid=IwAR3qRDgrlZV-LAUAy6xNceUx8EbuR9lP9WK0UeT6TOodNLk2Q34KnyddXYs
- Chk4 : https://www.youtube.com/watch?v=Oe6PXfMvHZA&fbclid=IwAR3zIqPtfaPGBFXwwAMpJw33p3dlU2MrMdZ1JwXIAE8OcSvifkZTq2BOmSw

Q & A