



LAB MANUAL 3

Deployment of Firebase Datalogging from IoT Devices

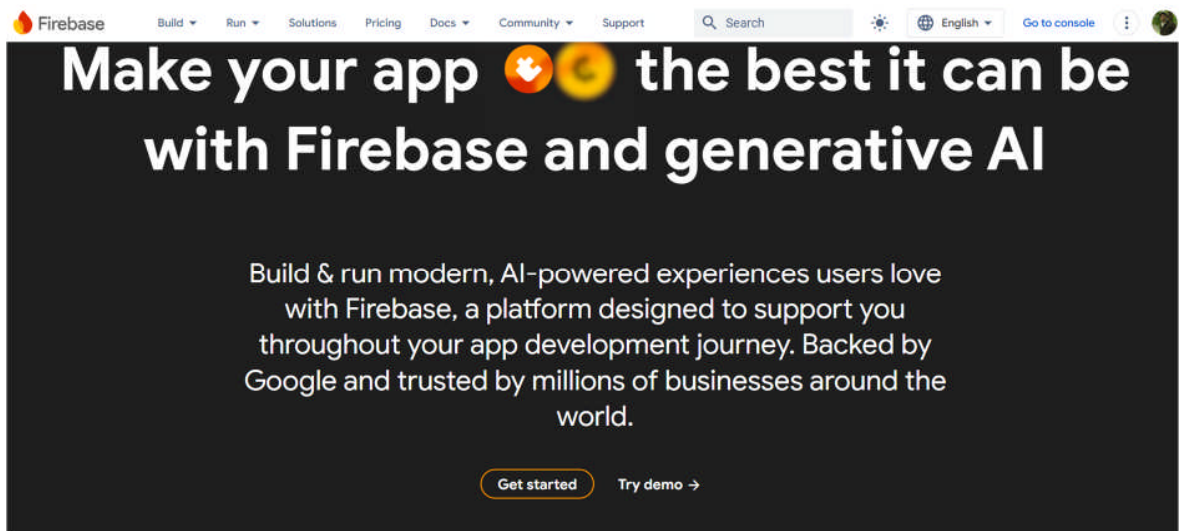
Deployment of Firebase Datalogging from IoT Devices

The Firebase Realtime Database is a cloud-hosted database. Data is stored as JSON and synchronized in realtime to every connected client. Lets implement a project to log the sensors data to Firebase realtime database

Steps-

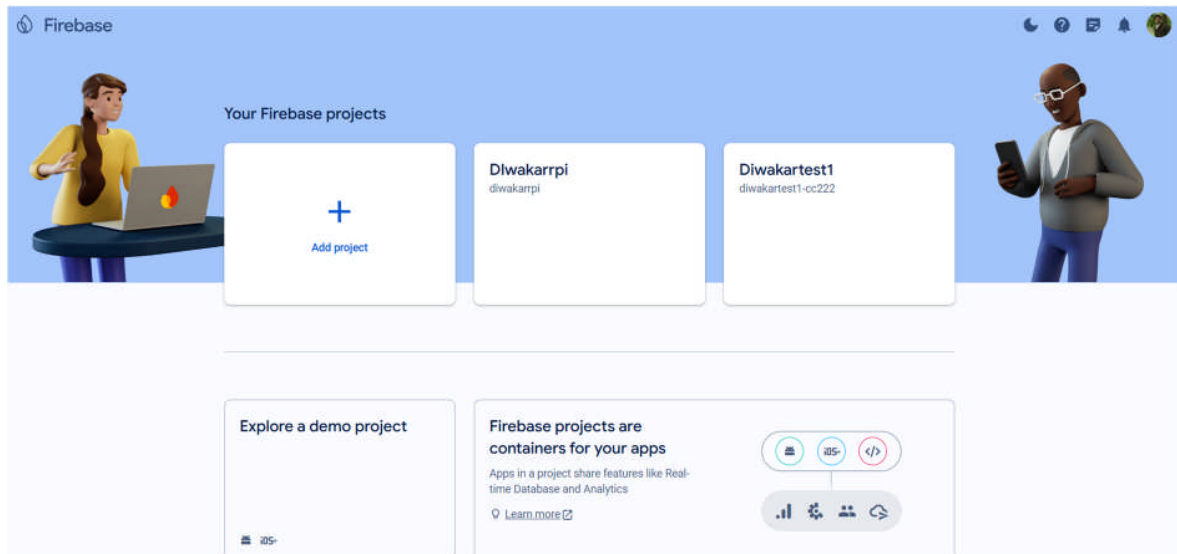
1. First things first, you need to create a new project in your console using below link

Link - <https://firebase.google.com/>

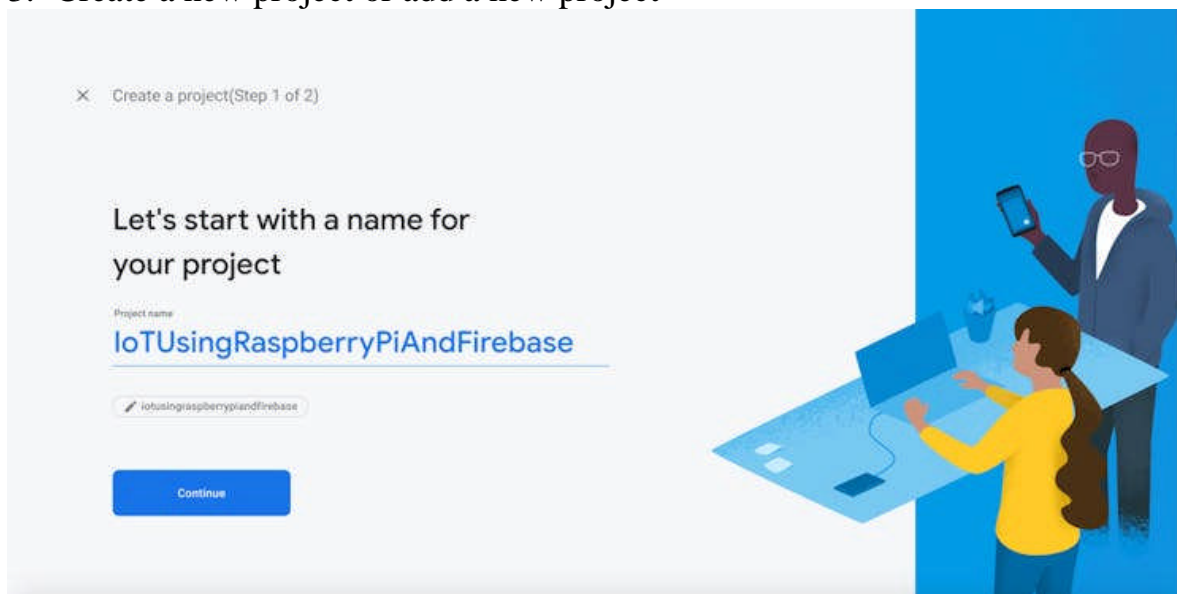


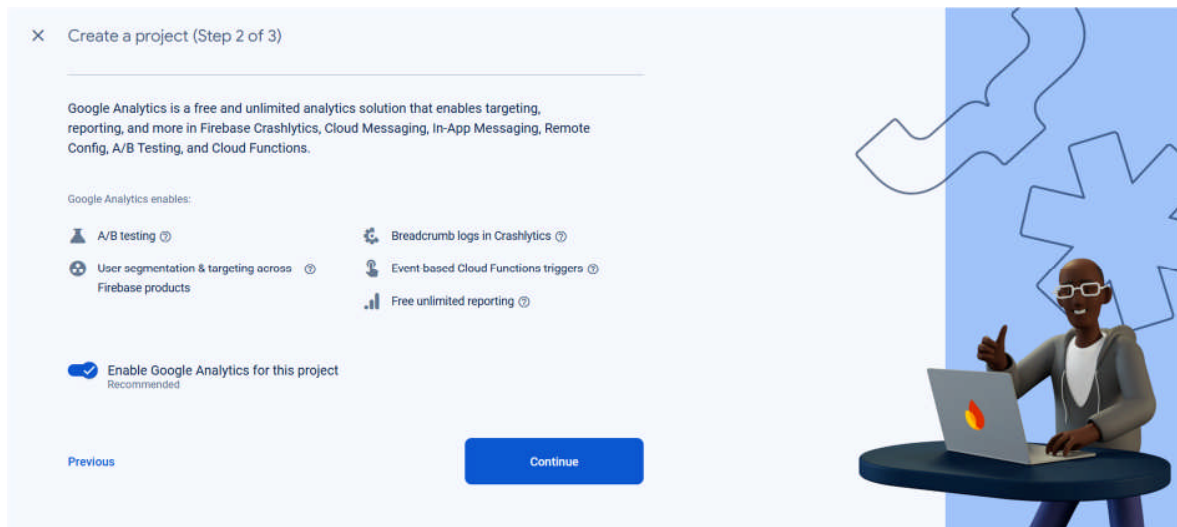
<https://firebase.google.com/>

2. Click on Get started and make login through Google account

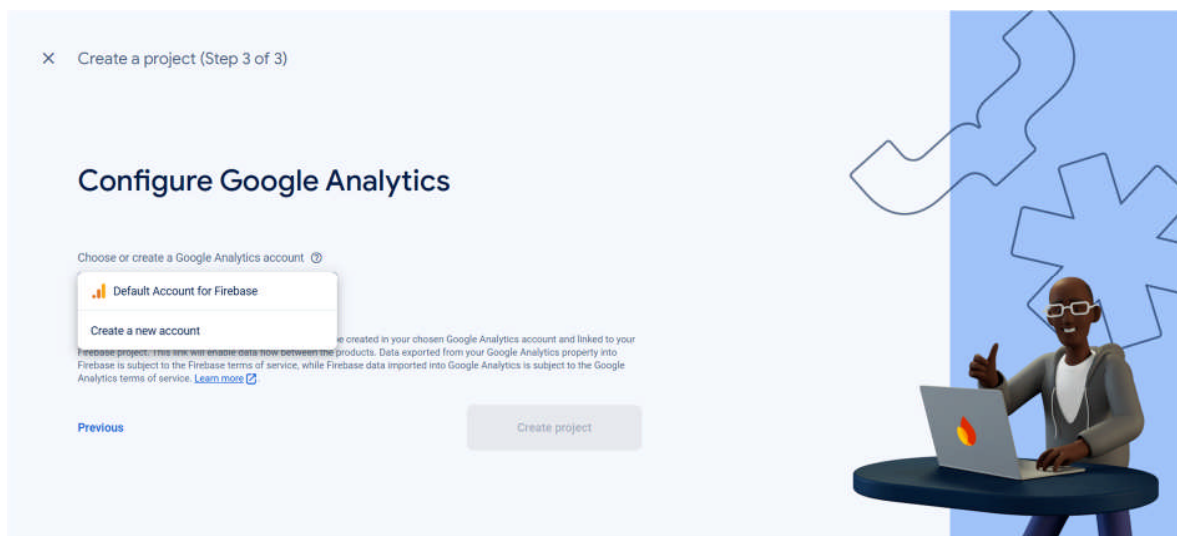


3. Create a new project or add a new project

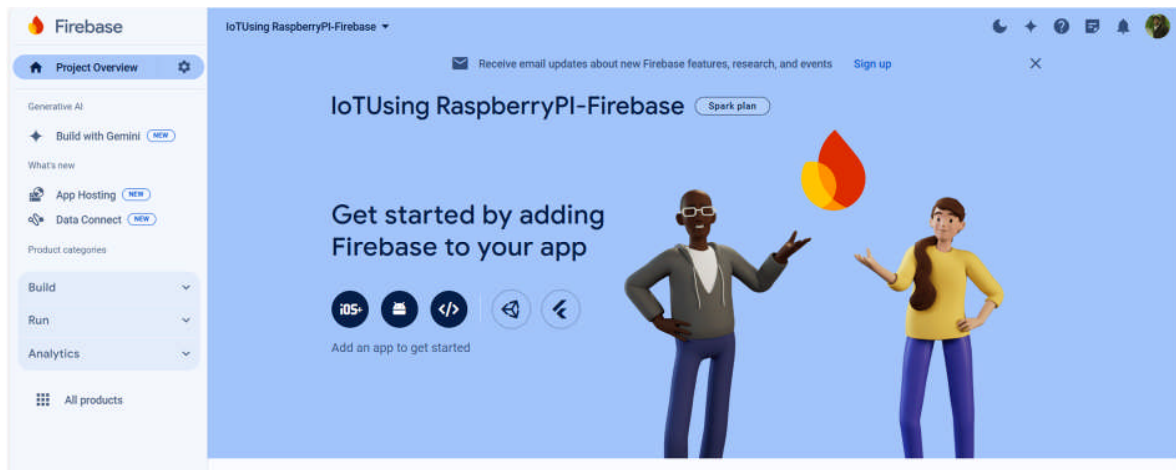




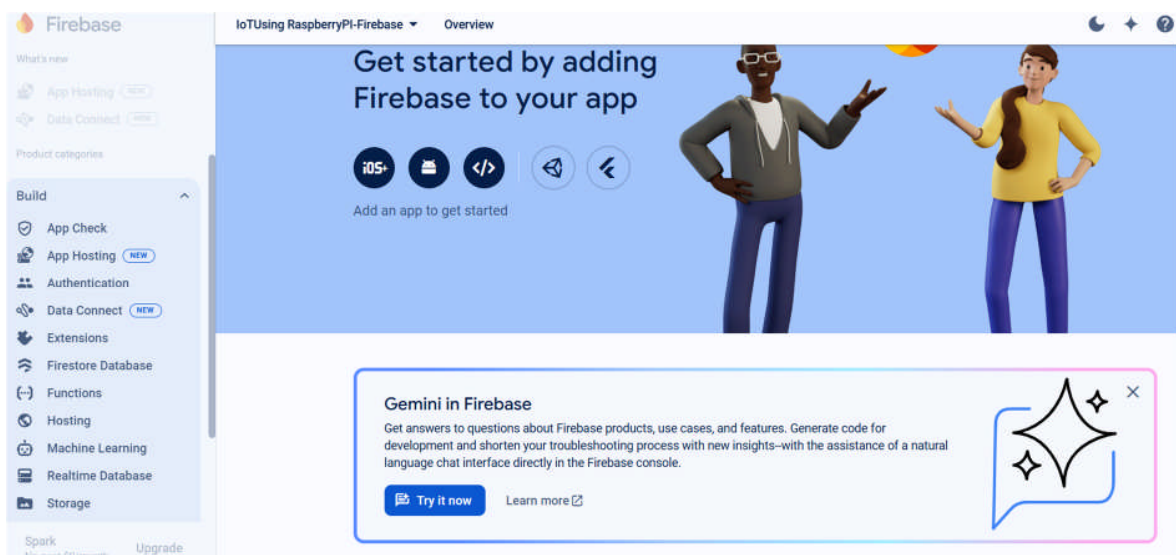
4. Click on continue and next need to select google analytic account to create project

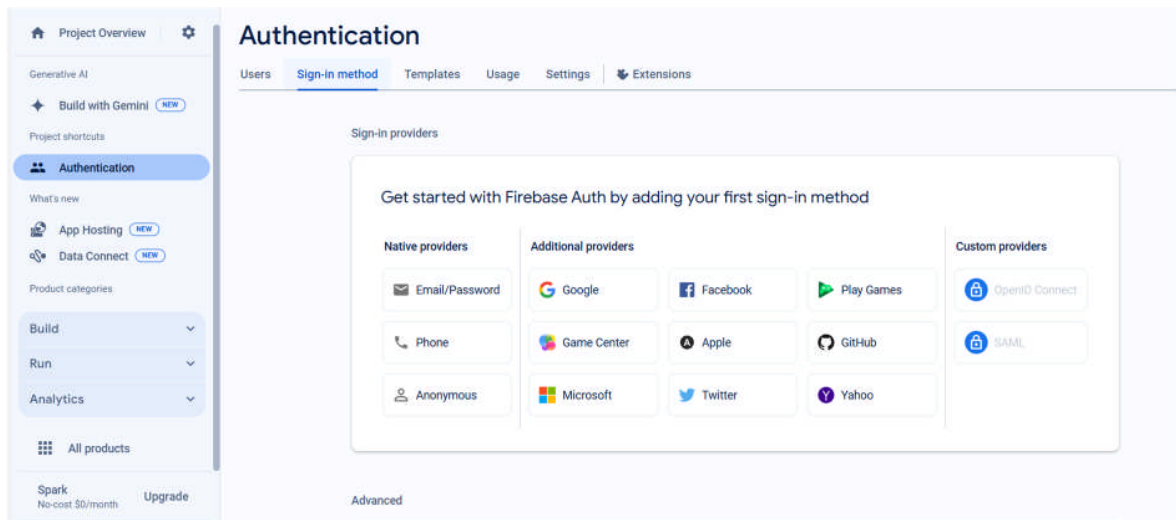


5. Finally will land on project page

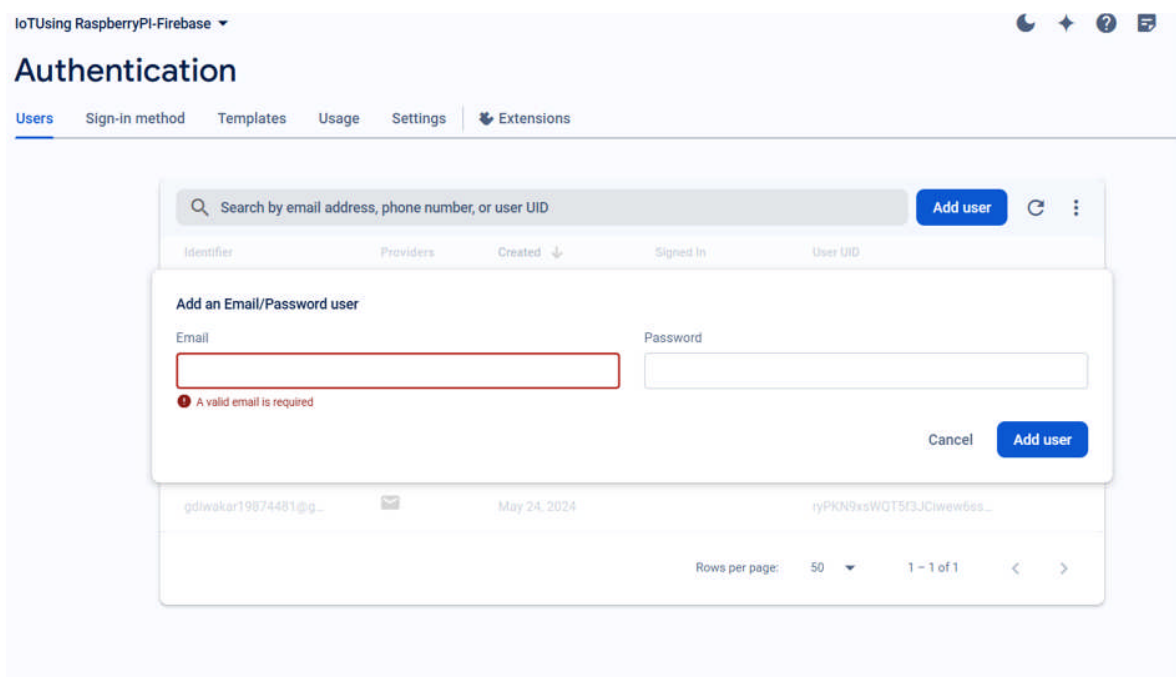


- Under Build head click on Authentication to proceed next. Set authentication mail & password

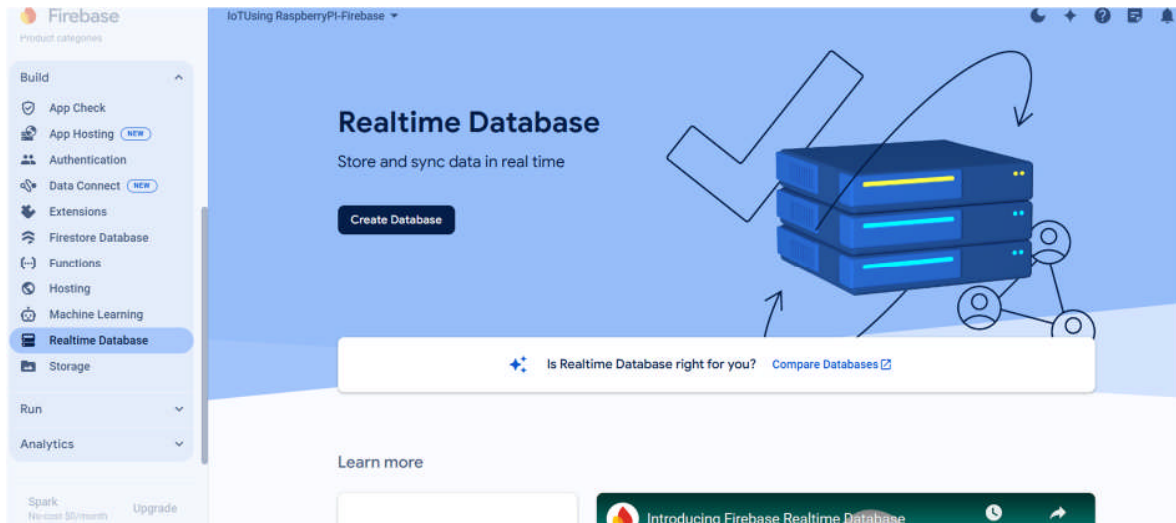




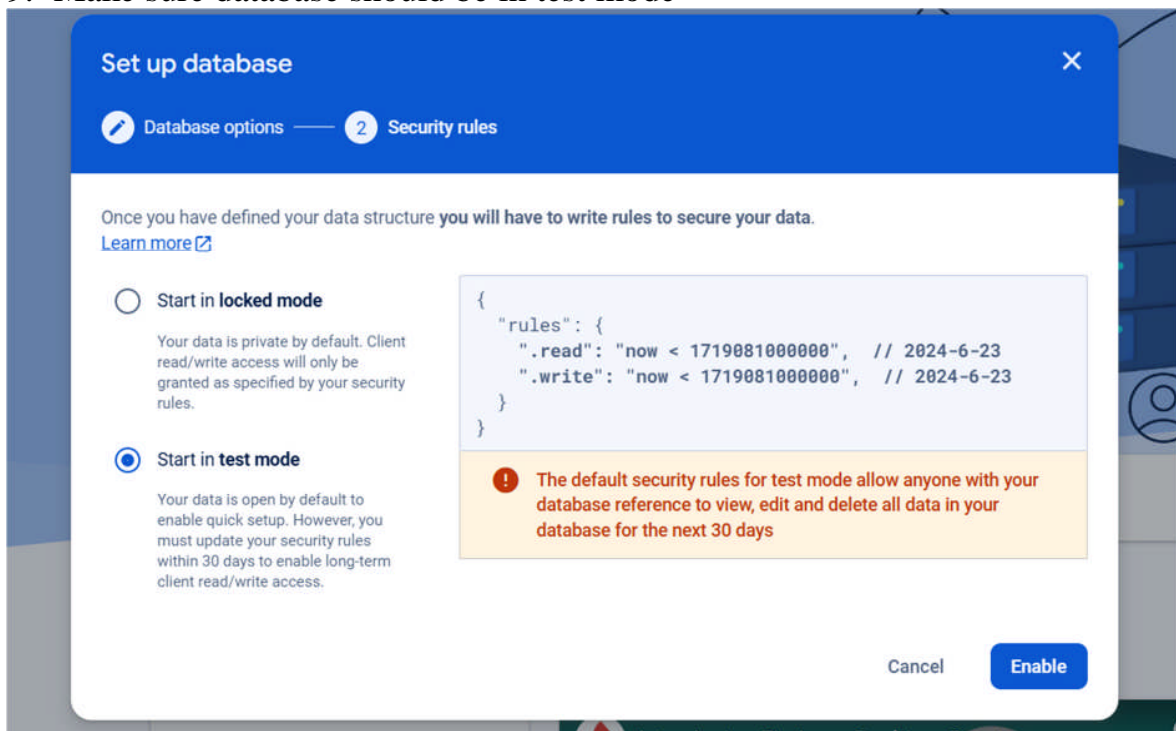
7. Add user as your mail-id and set password also for it



8. Again refer Build section, select Real Time Database and create one



9. Make sure database should be in test mode



10. Get the url of database

Realtime Database

Data Rules Backups Usage Extensions

Protect your Realtime Database resources from abuse, such as billing fraud or phishing [Configure App Check](#) X

<https://iotusing-raspberrypi-firebase-default-rtdb.firebaseio.com>

<https://iotusing-raspberrypi-firebase-default-rtdb.firebaseio.com/>: null

11. Fetch project details from Project settings

The screenshot shows the Firebase console interface. On the left, the 'Project settings' menu item is highlighted in the sidebar. The main content area displays the 'Realtime Database' page for the project 'IoTUsing RaspberryPi-Firebase'. The URL bar shows the database URL: <https://iotusing-raspberrypi-firebase-default-rtdb.firebaseio.com>. Below the URL bar, the value is shown as null.

Project settings

General Cloud Messaging Integrations Service accounts Data privacy Users and permissions

Your project

Project name	IoTUsing RaspberryPi-Firebase
Project ID	iotusing-raspberrypi-firebase
Project number	945187527358
Web API Key	AlzaSyC-Jb-xjQUI7h-VTBNbO_QjnEe55YexHis

Environment

This setting customizes your project for different stages of the app lifecycle

Environment type	Unspecified
------------------	-------------

12. Connect temperature & humidity sensor (A0 & A1) to DFRobot hat on RaspberryPI and run below code. Make sure to replace with your database credentials

Code-

```
# Install library pip3 install Pyrebase
```

```
import RPi.GPIO as GPIO
```

```
import time
```

```
import atexit
```

```
from dfadc import *
```

```
board_detect()
```

```
while board.begin() != board.STA_OK:
```

```
    print_board_status()
```

```
    print("board begin faild")
```

```
    time.sleep(2)
```

```
print("board begin success")
```

```
board.set_adc_enable()
```

```
atexit.register(GPIO.cleanup)
```

```
GPIO.setmode(GPIO.BCM)
```

```
from time import sleep
```

```
from math import isnan
```

```
from time import strftime
```

```
import math
```

```
import pyrebase
```

```
config = {
```

```
    "apiKey": "AIzaSyBLCJBhKJ6Gi1HN3QkqTqv1klMfdf8vBL4",
```

```
    "authDomain": "diwakarrpi.firebaseio.com",
```

```
    "databaseURL": "https://diwakarrpi-default-rtdb.firebaseio.com/",
```

```
    "storageBucket": "diwakarrpi.appspot.com"
```

```
}
```

```
firebase = pyrebase.initialize_app(config)
```

```
db = firebase.database()
```

```
while True:
```

```
    try:
```

```
# Print the values to the serial port
temp = board.get_adc_value(board.A0) # A0 channels read
humidity = board.get_adc_value(board.A1)
temperature = (temp/4096)* 100+20
humidity = (humidity/4096)* 100
print(temperature)
print(humidity)
data = {"Temperature" : temperature, "Humidity" : humidity}
db.child("Status").push(data)
db.update(data)
print("Sent to Firebase")

except RuntimeError as error:
    # Errors happen fairly often, DHT's are hard to read, just keep going
    print(error.args[0])
    time.sleep(2.0)
    continue
except Exception as error:
    dhtDevice.exit()
    raise error

time.sleep(2.0)
```

13. Observe the transferred data seems visible on firebase database

Realtime Database

Data Rules Backups Usage Extensions



Protect your Realtime Database resources from abuse, such as billing fraud or phishing

[Configure App Check](#)<https://diwakarrpi-default-rtdb.firebaseio.com><https://diwakarrpi-default-rtdb.firebaseio.com/>

Humidity: 0

Status

Temperature: 31.5966796875

Using GrovePI & RaspberryPI

Steps

1. Connect DHT sensor to port D7
2. Connect LCD to I2C port
3. Install library **pip3 install Pyrebase** & Run below code

Code

```
# pip3 install Pyrebase
```

```
from grovepi import *
```

```
from grove_rgb_lcd import *
```

```
import time
```

```
from math import isnan
```

```
from time import strftime
```

```
import math
```

```
import pyrebase
```

```
dht_sensor_port = 7
```

```
dht_sensor_type = 0 # 0 for DHT11 and 1 for DHT22
```

```
setRGB(0,255,0)
```

```
config = {
```

```

"apiKey": "AIzaSyBLCJBhKJ6Gi1HN3QkqTqvklMfdf8vBL4",
"authDomain": "diwakarrpi.firebaseio.com",
"databaseURL": "https://diwakarrpi-default-
rtddb.firebaseio.com/",
"storageBucket": "diwakarrpi.appspot.com"
}

```

```
firebase = pyrebase.initialize_app(config)
```

```
db = firebase.database()
```

```
while True:
```

```
    try:
```

```
        # Print the values to the serial port
```

```
        [t,h] = dht(dht_sensor_port,dht_sensor_type)
```

```
        print(f"Temp:{t} C Humidity:{h}%")
```

```
        setText_norefresh(f"Temp:{t} C\nHumidity:{h}%")
```

```
        print(t)
```

```
        print(h)
```

```
        data = {"Temperature" : t,"Humidity" : h}
```

```
        db.child("Status").push(data)
```

```
        db.update(data)
```

```
        print("Sent to Firebase")
```

```
    except RuntimeError as error:
```

```
        # Errors happen fairly often, DHT's are hard to read, just keep going
```

```
        print(error.args[0])
```

```
        time.sleep(2.0)
```

```
        continue
```

```
    except Exception as error:
```

```
        dhtDevice.exit()
```

```
        raise error
```

```
time.sleep(2.0)
```