



Nama: **Arsyadana Estu Aziz (121140068)**  
Mata Kuliah: **Pervasive Computing (IF4025)**

Tugas Ke: **1**  
Tanggal: **2 Oktober 2024**

## 1 Introduction

Perkenalkan nama Saya Arsyadana Estu Aziz, dan saya akan menjelaskan tentang proses penggunaan *Realtime Database* di Firebase dan melakukan proses deployment di Vercel. Pengerjaan laporan dilakukan dalam LaTeX [1]

## 2 Realtime Database

Apa itu *Realtime Database*? Sebuah layanan basis data dari Firebase untuk dapat melakukan CRUD dengan skema No-SQL database [2], Pada kesempatan kali ini akan dilakukan pembuatan sebuah *Realtime Database* dengan mengirimkan data proses dan ditampilkan ke halaman web sederhana dengan Flask.

Hal pertama yang perlu dilakukan adalah dengan membuat project di console firebase dengan cara Berikut ([Firebase Console](#))

Selanjutnya, yang perlu dilakukan adalah aktivasi fitur *Realtime Database* dan mendapatkan service account dengan cara generate *private key* dan simpan file **.json** dalam struktur direktori file anda.

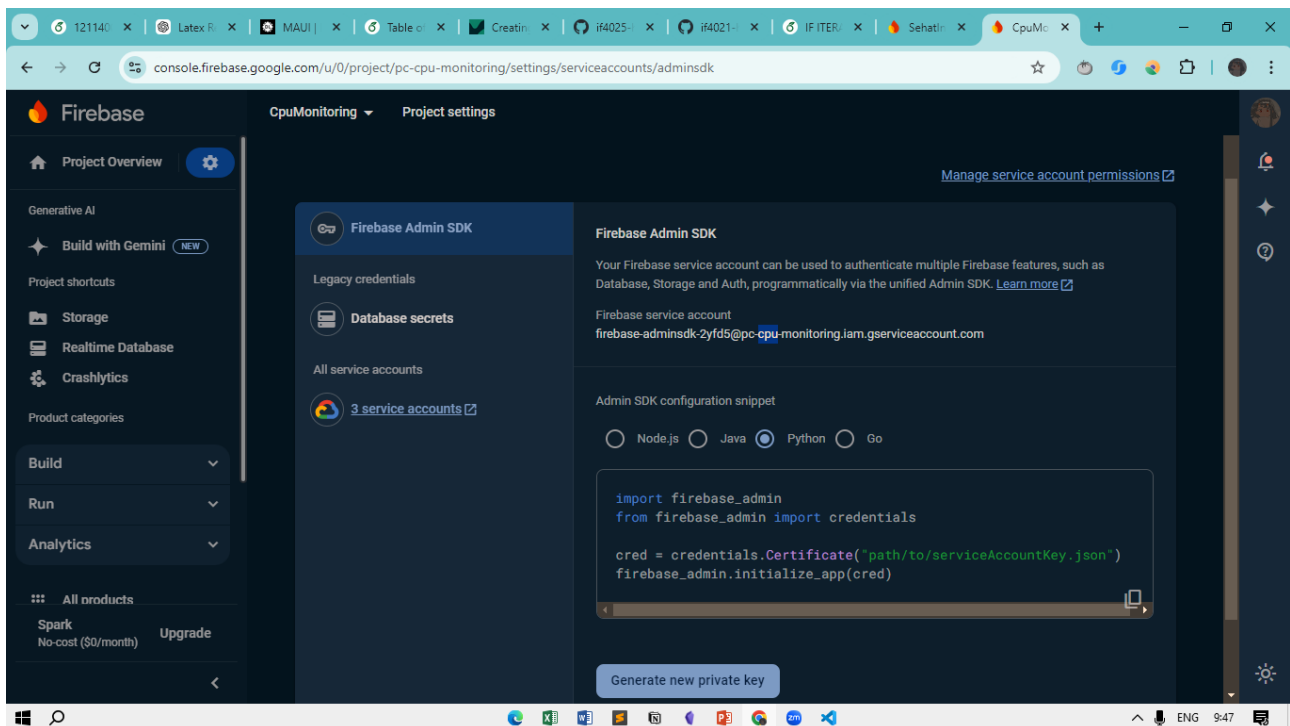


Figure 1: Firebase Service Account Creation

Satu hal sudah selesai, selanjutnya kita akan membuat sebuah folder project baru **real\_time\_database**, jangan lupa untuk membuat enviroment conda dan melakukan instalasi dalam file **requirements.txt**

```
1 Flask==2.0.1
2 Werkzeug==2.0.1
3 firebase-admin==5.0.0
4 psutil
```

### 3 Codebase

#### 3.1 Sending some data into the database

Pada bagian ini, kita akan memasuki bagian terseru, yakni membuat program sederhana untuk menyimpan dan menampilkan data tersebut dalam halaman web sederhana, kita akan menggunakan *Flask* untuk membuat itu. Untuk source code bisa dilihat pada *code snippets* berikut [1](#)

```
1
2 import firebase_admin
3 from firebase_admin import credentials, db
4 import psutil
5 import time
6
7 # Inisialisasi Firebase Admin SDK
8 cred = credentials.Certificate("pc-cpu-monitoring-firebase-adminsdk-2yfd5-cf147c47f5.json")
9 firebase_admin.initialize_app(cred, {
10     'databaseURL': 'https://pc-cpu-monitoring-default-rtdb.asia-southeast1.firebaseio.com/'
11 })
12
13 # Referensi ke node 'cpu_usage' di Firebase
14 ref = db.reference('cpu_usage')
15
16 # Fungsi untuk mengirim data CPU usage ke Firebase
17 def push_cpu_data():
18     while True:
19         cpu_percent = psutil.cpu_percent(interval=1) # Mengambil CPU usage
20
21         # Disk usage relative to current Drive (D)
22         disk_usage = psutil.disk_usage('/')
23         disk_metrics = {
24             'total_disk_gb': disk_usage.total / (1024**3), # in GB
25             'used_disk_gb': disk_usage.used / (1024**3), # in GB
26             'free_disk_gb': disk_usage.free / (1024**3), # in GB
27             'disk_percent': disk_usage.percent # percentage used
28         }
29
30         # Network statistics
31         net_io = psutil.net_io_counters()
32         network_metrics = {
33             'bytes_sent_mb': net_io.bytes_sent / (1024**2), # in MB
34             'bytes_recv_mb': net_io.bytes_recv / (1024**2), # in MB
35             'packets_sent': net_io.packets_sent, # packet count
36             'packets_recv': net_io.packets_recv # packet count
37         }
38
39         data = {
40             'cpu': cpu_percent,
41             'disk': disk_metrics,
42             'net': network_metrics,
43             'timestamp': time.time() # Menyimpan timestamp
```

```

45     }
46
47     ref.push(data) # Mengirim data ke Firebase
48
49     print(f"Data sent to Firebase: {data}")
50     time.sleep(5) # Mengirim data setiap 5 detik
51
52 if __name__ == "__main__":
53     push_cpu_data()

```

Kode 1: Python Send Data.py

dapat dilihat bahwa disini terdapat beberapa point penting, yang pertama yakni kita menggunakan file service accounts yang telah didapat sebelumnya dan ditaruh ke dalam struktur file agar bisa mendapatkan akses kedalam firebase.

```

1
2 # Inisialisasi Firebase Admin SDK
3 cred = credentials.Certificate("pc-cpu-monitoring-firebase-adminsdk-2yfd5-cf147c47f5.json")
4 firebase_admin.initialize_app(cred, {
5     'databaseURL': 'https://pc-cpu-monitoring-default-rtdb.asia-southeast1.firebaseio.com/'
6 })
7
8 # Referensi ke node 'cpu_usage' di Firebase
9 ref = db.reference('cpu_usage')

```

Kode 2: Firebase Credentials

selanjutnya cukup sederhana, kita menggunakan library **psutil** untuk mendapatkan data proses dari komputer (CPU, Disk dan Network) dan kita kirimkan dengan methode **ref.push()** untuk setiap 5 detik

### 3.2 Getting the data from Firebase

Pada bagian ini, kita menggunakan Flask sebagai framework untuk mensimulasikan grafik penggunaan data dan cara untuk mendapatkan data tersebut

```

1
2 from flask import Flask, render_template, jsonify
3 import firebase_admin
4 from firebase_admin import credentials, db
5 import os
6
7 app = Flask(__name__)
8
9 # Inisialisasi Firebase Admin SDK
10 cred = credentials.Certificate("pc-cpu-monitoring-firebase-adminsdk-2yfd5-cf147c47f5.json")
11 firebase_admin.initialize_app(cred, {
12     'databaseURL': 'https://pc-cpu-monitoring-default-rtdb.asia-southeast1.firebaseio.com/'
13 })
14
15 # Referensi ke node 'cpu_usage' di Firebase
16 ref = db.reference('cpu_usage')
17
18 @app.route('/')
19 def index():
20     return render_template('index.html')
21
22 ## Getting the last currently updated snapshot data
23 ## And put it in the List Comprehension before sending the Json data to App.py
24 @app.route('/data')
25 def get_data():
26     snapshot = ref.order_by_key().limit_to_last(10).get()

```

```

27 data = [{ 'cpu': value['cpu'], "disk": value["disk"], "net": value["net"], 'timestamp': value['
    timestamp']} for _, value in snapshot.items()]
28 return jsonify(data)

```

pada bagian ini, kita melakukan routing untuk menampilkan grafik pada route "/" dan untuk mendapatkan data dengan route **"/data"**. Kenapa perlu ada routing? karena kita akan melakukan fetching data dan menyimpan data tersebut pada halaman **templates/index.html**.

```

1 function updateChart() {
2   fetch("/data")
3     .then((response) => response.json())
4     .then((data) => {
5       console.log(data);
6       const timestamps = data.map((item) =>
7         new Date(item.timestamp * 1000).toLocaleTimeString()
8       );
9       // Getting the data and convert to corresponding array of cpu, disk and network
10      const cpuUsages = data.map((item) => item.cpu);
11      const diskUsage = data.map((item) => item.disk.disk_percent);
12      const networkUsage = data.map((item) => item.net.bytes_sent_mb);
13
14      cpuChart.data.labels = timestamps;
15      cpuChart.data.datasets[0].data = cpuUsages;
16      cpuChart.data.datasets[1].data = diskUsage;
17      cpuChart.data.datasets[2].data = networkUsage;
18
19      cpuChart.update();
20    });
21 }
22
23 setInterval(updateChart, 5000); // Update setiap 5 detik

```

Kita mendapatkan data ini, kemudian untuk dapat ditampilkan dalam bentuk grafik.

```

1 <body>
2 <h1>Real-Time CPU Usage</h1>
3 <canvas id="cpuChart" width="400" height="200"></canvas>
4
5 <script>
6   var ctx = document.getElementById("cpuChart").getContext("2d");
7   var cpuChart = new Chart(ctx, {
8     type: "line",
9     data: {
10       labels: [],
11       datasets: [
12         {
13           label: "CPU Usage (%)",
14           data: [],
15           borderColor: "rgba(75, 192, 192, 1)",
16           borderWidth: 2,
17           fill: false,
18         },
19         {
20           label: "Disk Used (D: Drive)",
21           data: [],
22           borderColor: "rgba(6, 89, 43, 0.3)",
23           borderWidth: 2,
24           fill: false,
25         },
26         {
27           label: "Packets Sends",
28           data: [],
29           borderColor: "rgba(31, 95, 186, 0.8)",

```

```

30         borderWidth: 2,
31         fill: false,
32     },
33 },
34 },
35 options: {
36     scales: {
37         y: {
38             beginAtZero: true,
39             max: 100,
40         },
41     },
42 },
43 });

```

Lalu terakhir, anda bisa menggunakan perintah **flask run** untuk menjalankan kode ini.

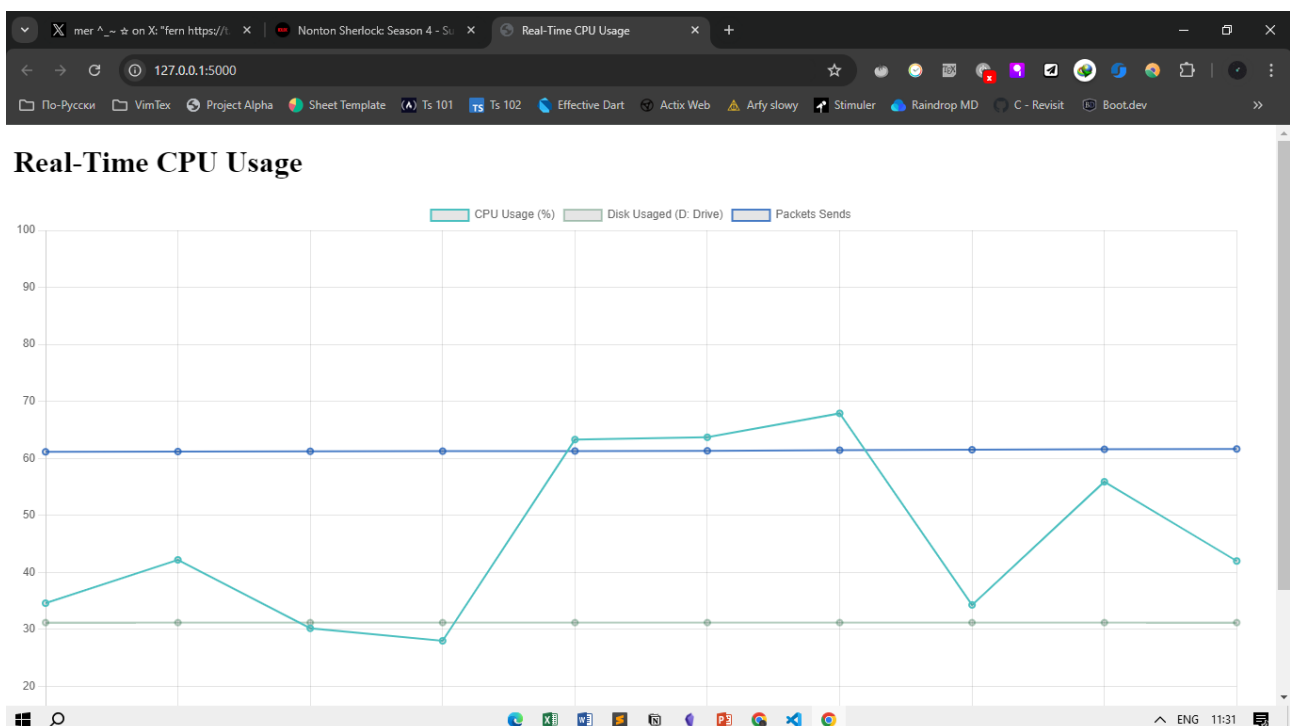


Figure 2: Firebase Service Account Creation

## 4 Deploying to Vercel

Bagian terseru dari tugas ini, yakni mendeploy program yang sudah kita buat ke internet. Dengan menggunakan **vercel**. Hal pertama yang perlu dilakukan adalah membuat sebuah akun vercel pada halaman berikut ([Vercel](#)) Setelah itu, kita perlu menginstall vercel sebelum melakukan proses deploy, kita akan melakukan instalasi **vercel CLI** (Pastikan anda sudah menginstal NPM Package)[3]

```
1 npm install -g vercel
```

Voila, kalian bisa melihat hasil akhirnya di halaman situs ini [Vercel Page](#)

## References

- [1] (-) Latex report project structure. [Online]. Available: <https://chatgpt.com/share/66fcebe9-0098-800c-b13b-d511a3b10be5>
- [2] (-) Read and write data on the web. [Online]. Available: <https://firebase.google.com/docs/database/web/read-and-write>
- [3] Vivek. (2024) A step-by-step guide to deploying your website on vercel platform. [Online]. Available: <https://medium.com/@vivek76/a-step-by-step-guide-to-deploying-your-website-on-vercel-59c08437049b#:~:text=Before%20you%20can%20deploy%20your,directed%20to%20the%20Vercel%20dashboard.&text=After%20logging%20into%20your%20Vercel,the%20%E2%80%9CImport%20Project%E2%80%9D%20button.>