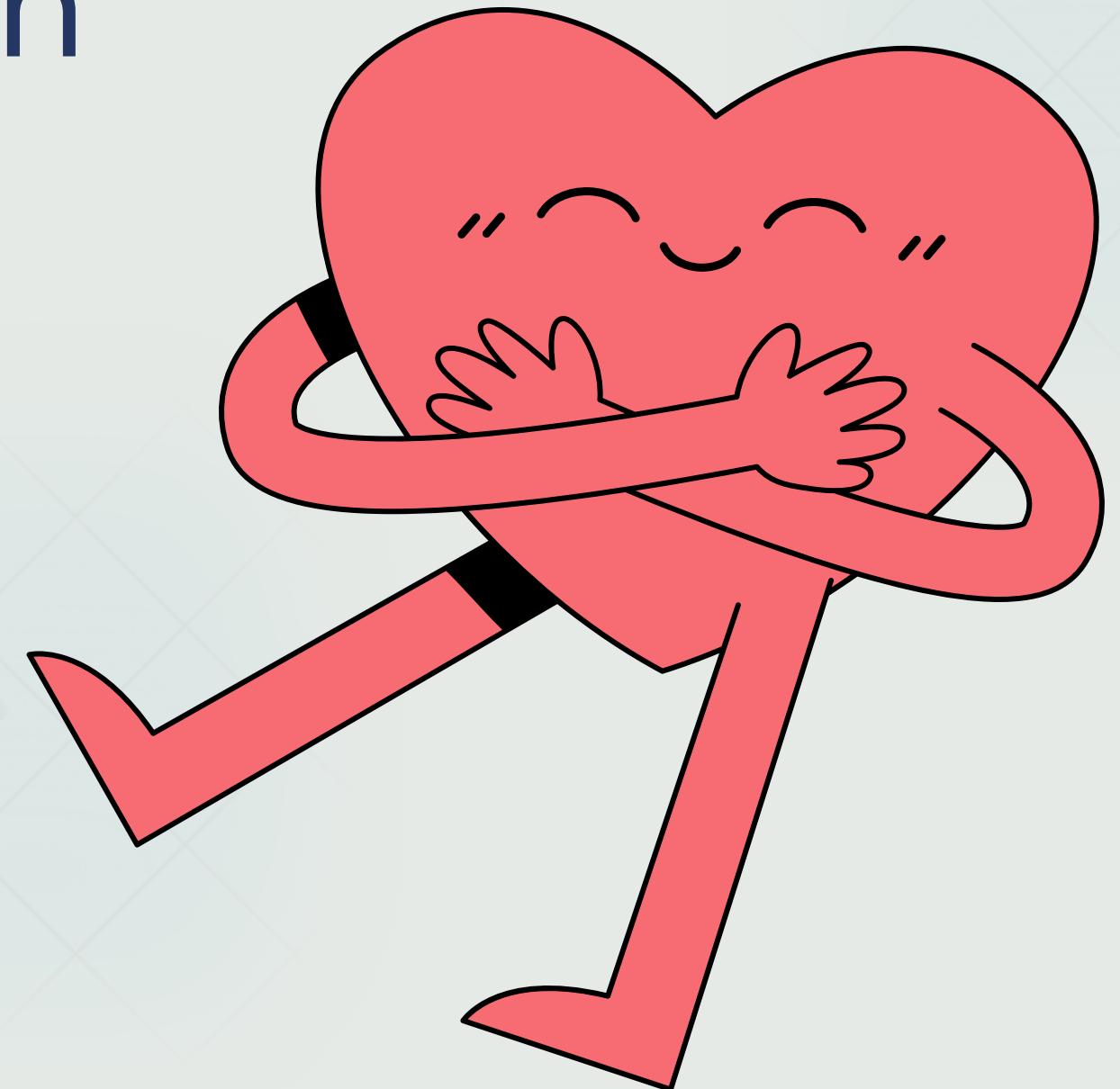




# Sistem monitoring Kesehatan tubuh berdasarkan detak jantung dan suhu tubuh berbasis Internet Of Things

Money Proyek dari Startup Kami



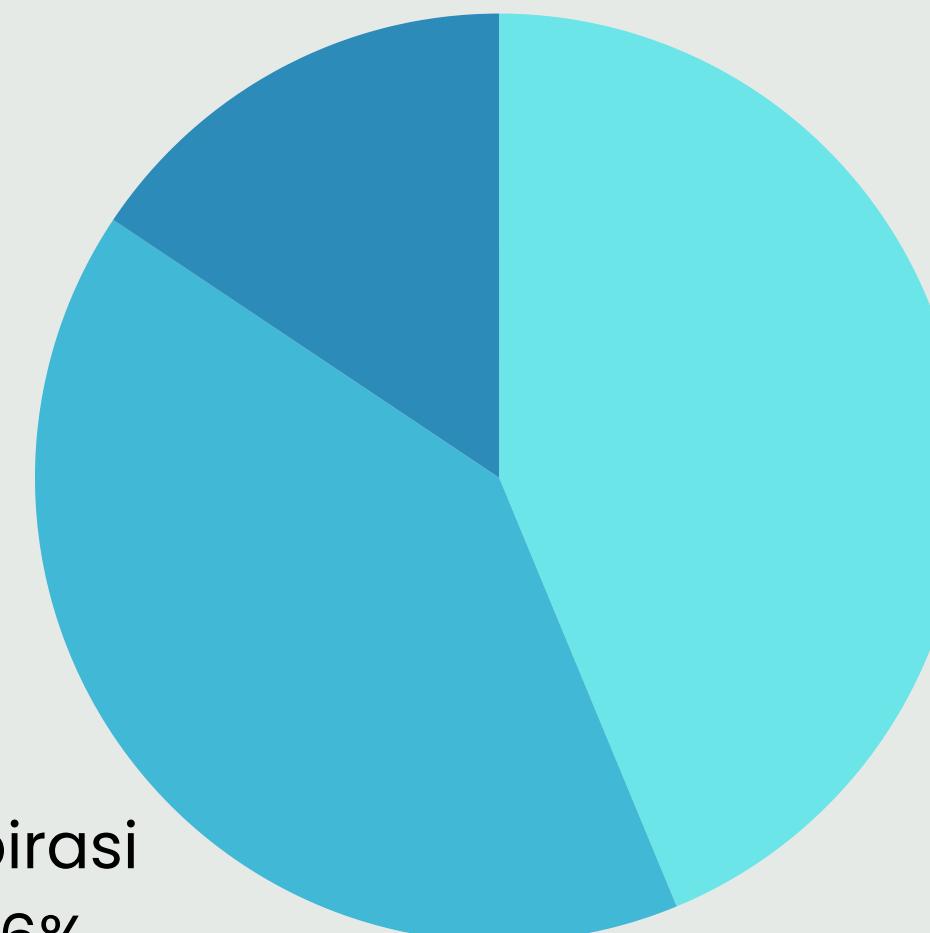
Surabaya, 11 November 2024

## **Prevalensi Kejadian Mati Mendadak Tahun 2020 – 2021**

Sistem Saraf Pusat  
15.6%

Respirasi  
40.6%

Kardiovaskuler  
43.8%

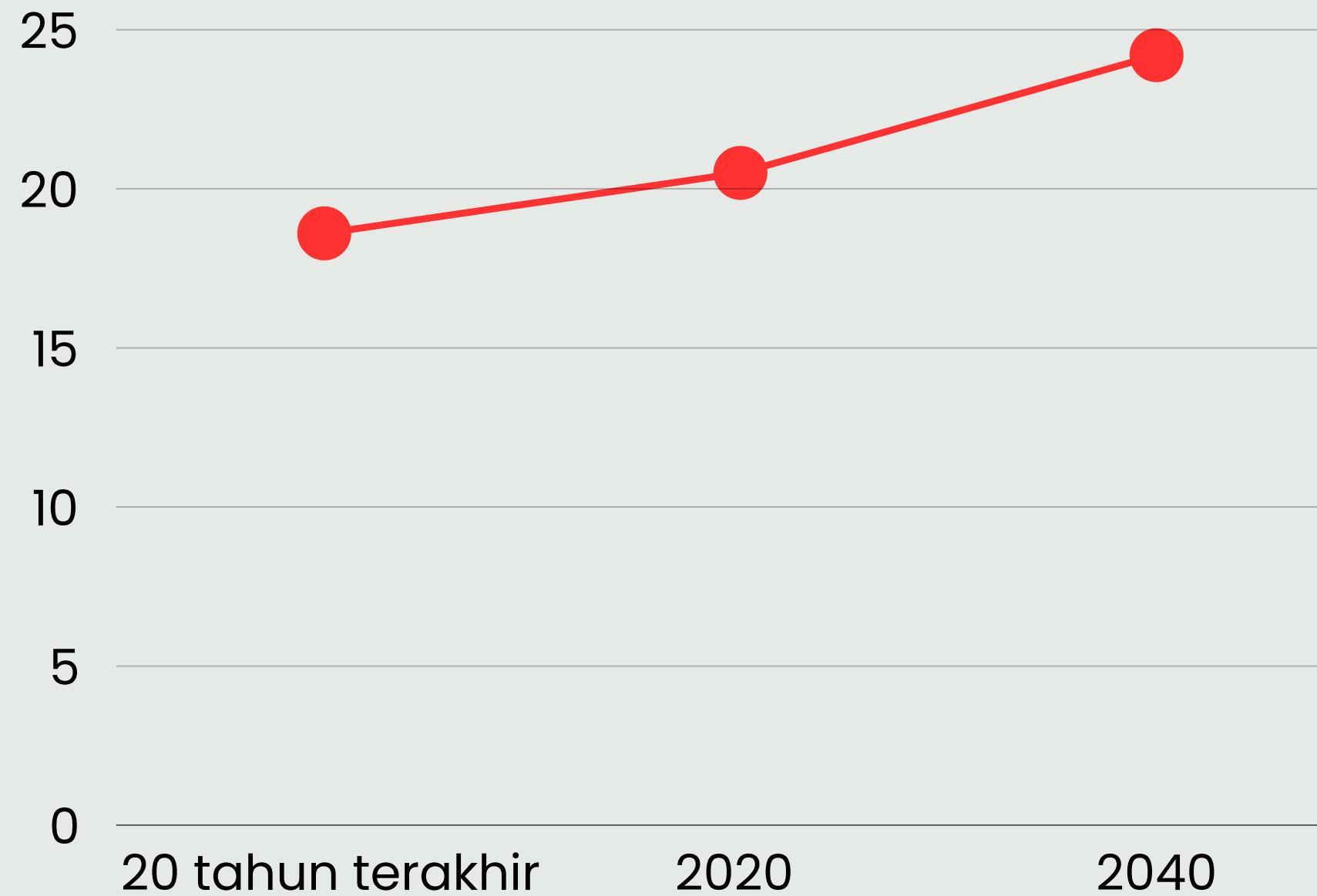


**Kardiovaskuler**  
**43.8%**

*Sumber : Fakumi Medical Journal  
Andi Nurul Hikmah R Agussalim ,dkk-2024*



## *Prediksi Penyakit Jantung*



Meningkat  
1.5% setiap  
tahunnya

Sumber : Global Status Report on NCD 2019  
(IHME)



# Periksa ke klinik



# Menggunakan CT Scan



# Inovasi Monitoring Kesehatan Jantung berbasis Internet of Things



Fani Tafazzul



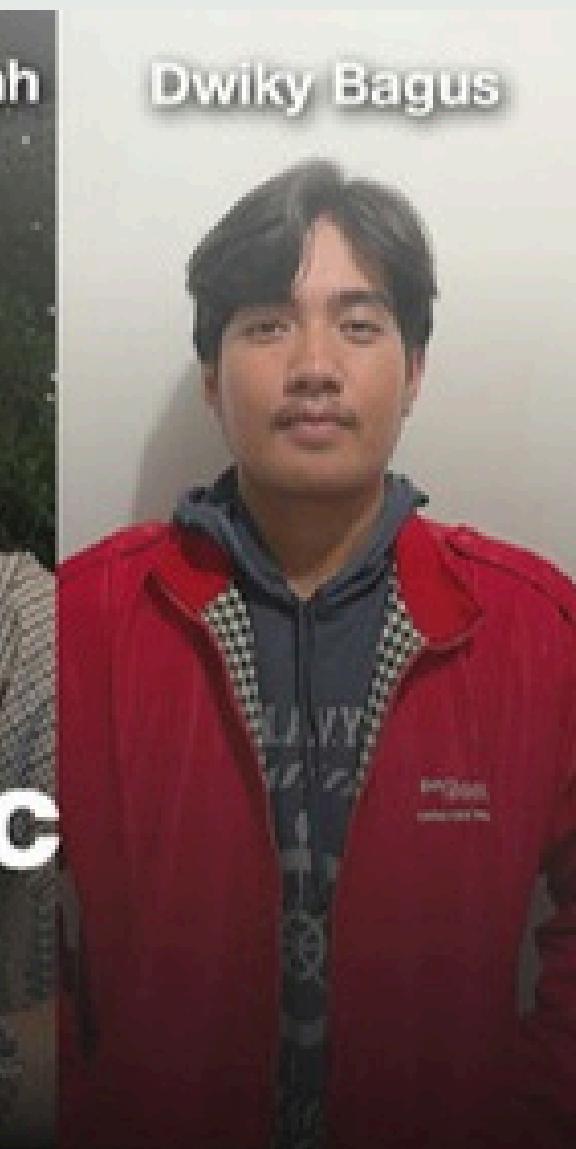
Suryo Nadi



Rendy Firmansyah



Dwicky Bagus



Arfin Nurur

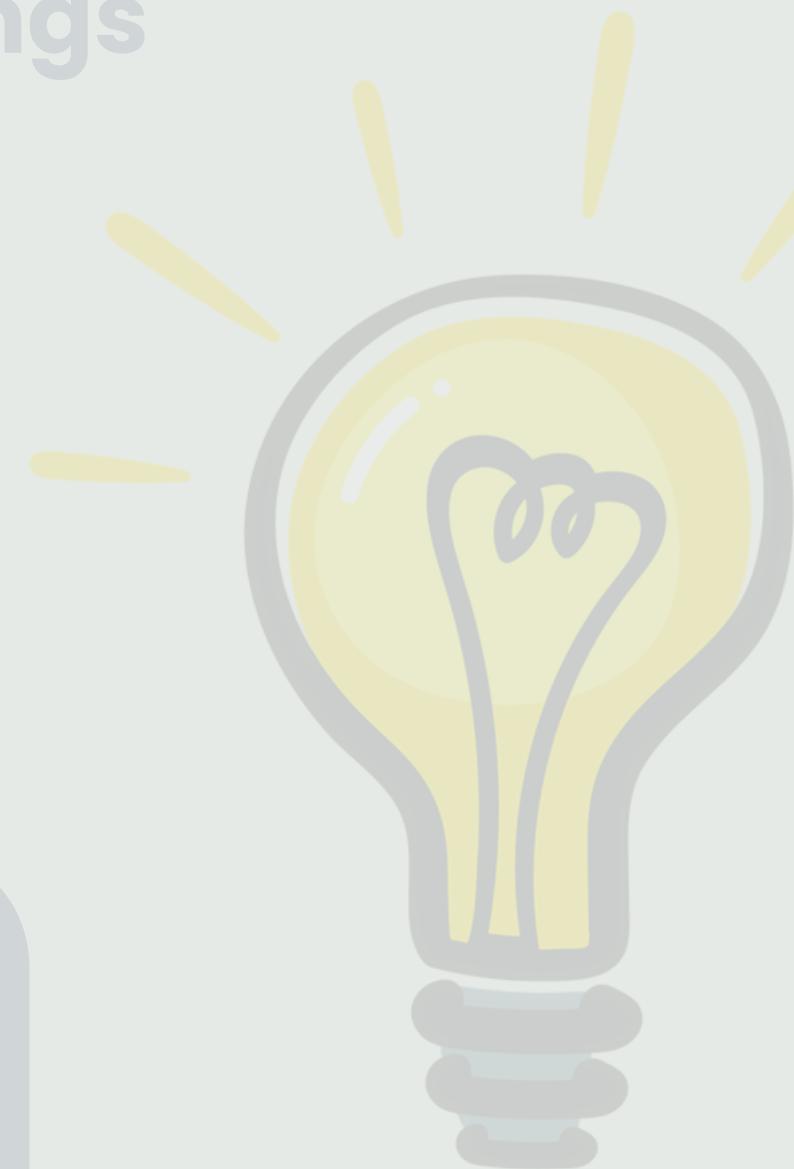


Supported By

Kopi Good Day 



# Inovasi Monitoring Kesehatan Jantung berbasis Internet of Things



Arfin Nurur Robi  
*Divisi Hardware*

Surya  
*Divisi Software Backend*

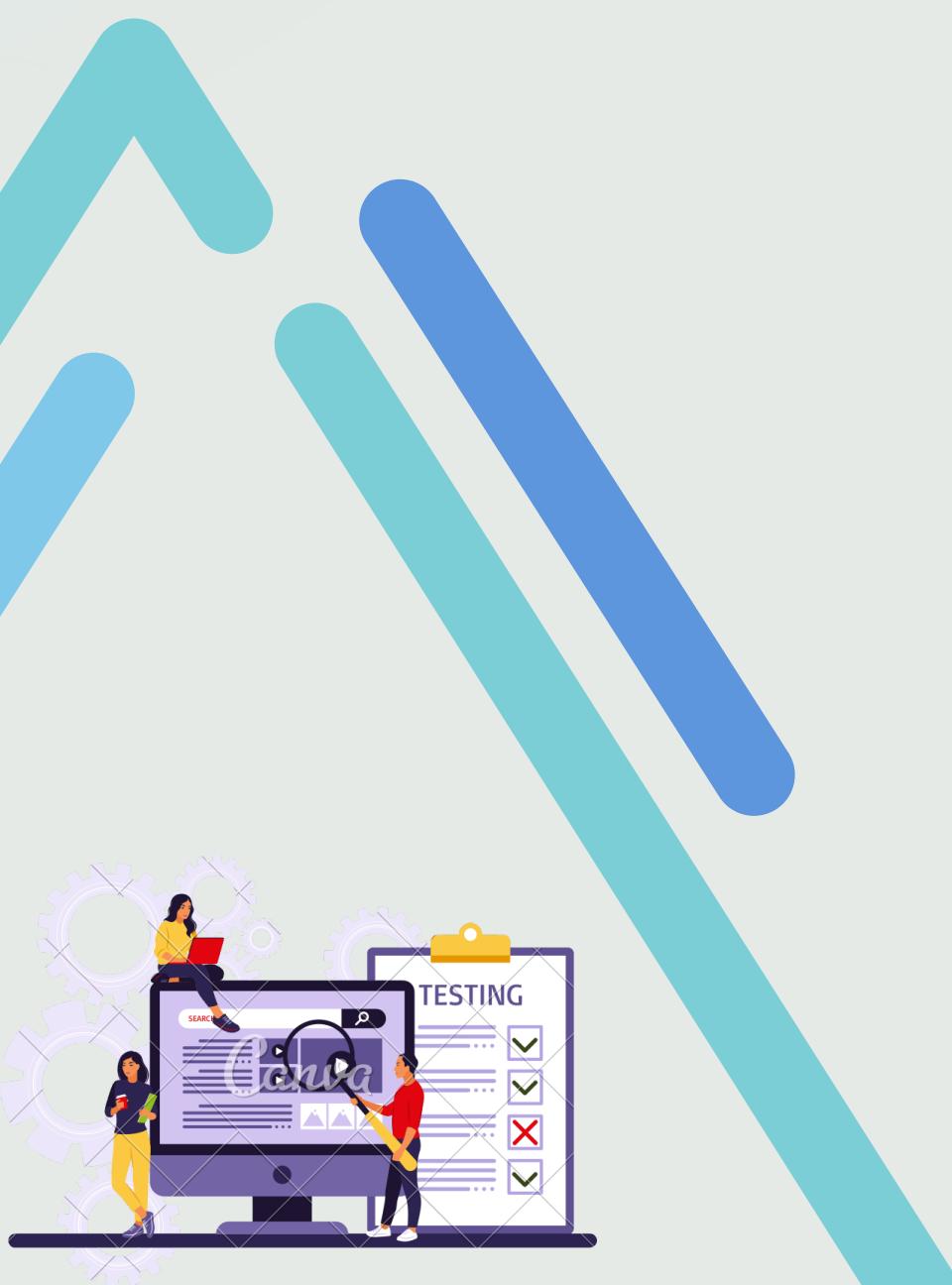
Dwicky  
*Divisi Software Front End*

M. Fani Tafazzul Hilmi  
*Divisi Design Product*

Ira Adi Nata  
*Divisi Firmware*



# Hasil yang dicapai





# Tampak Produk



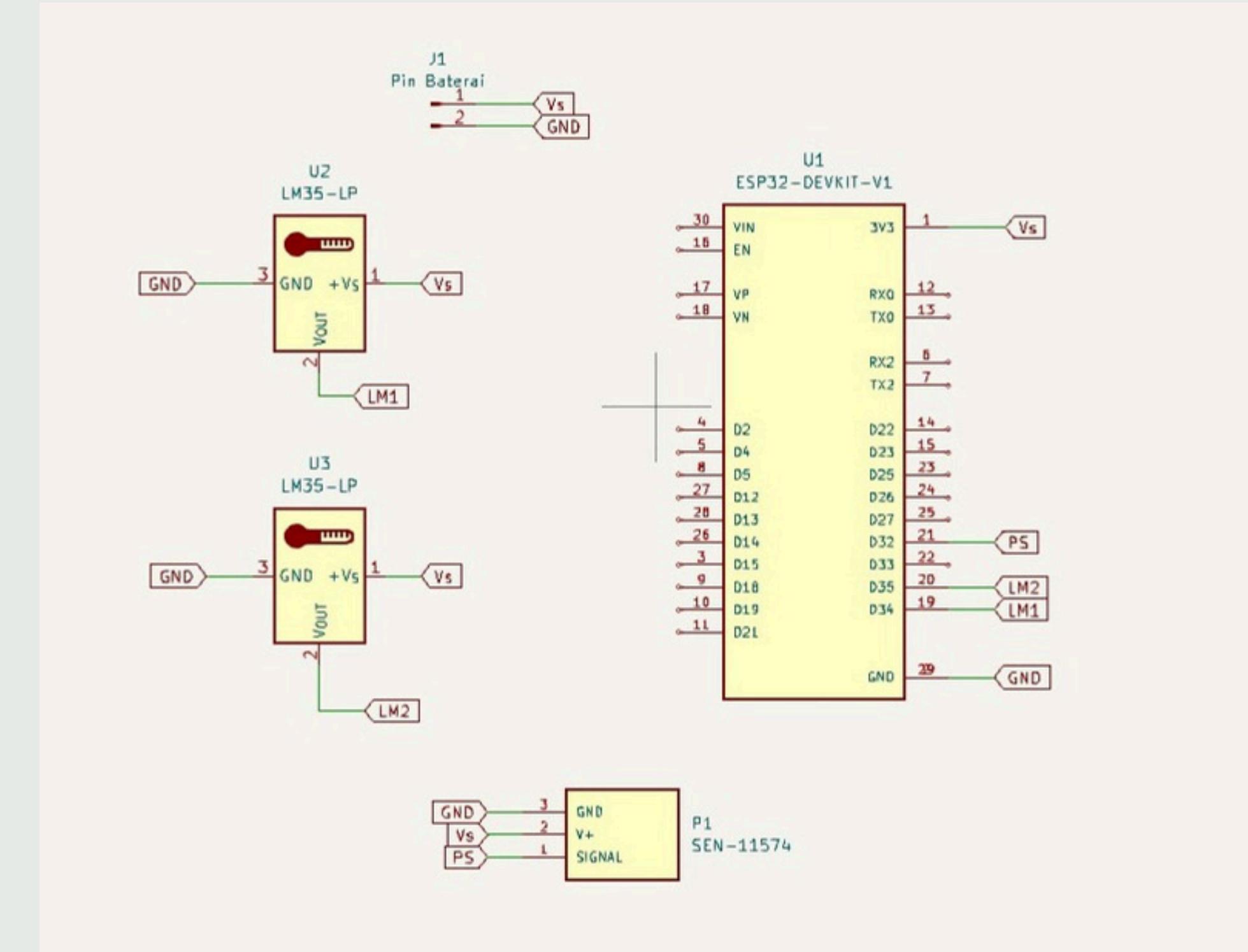


# *Adapun Bagian bagian untuk menyempurnakan Produk*



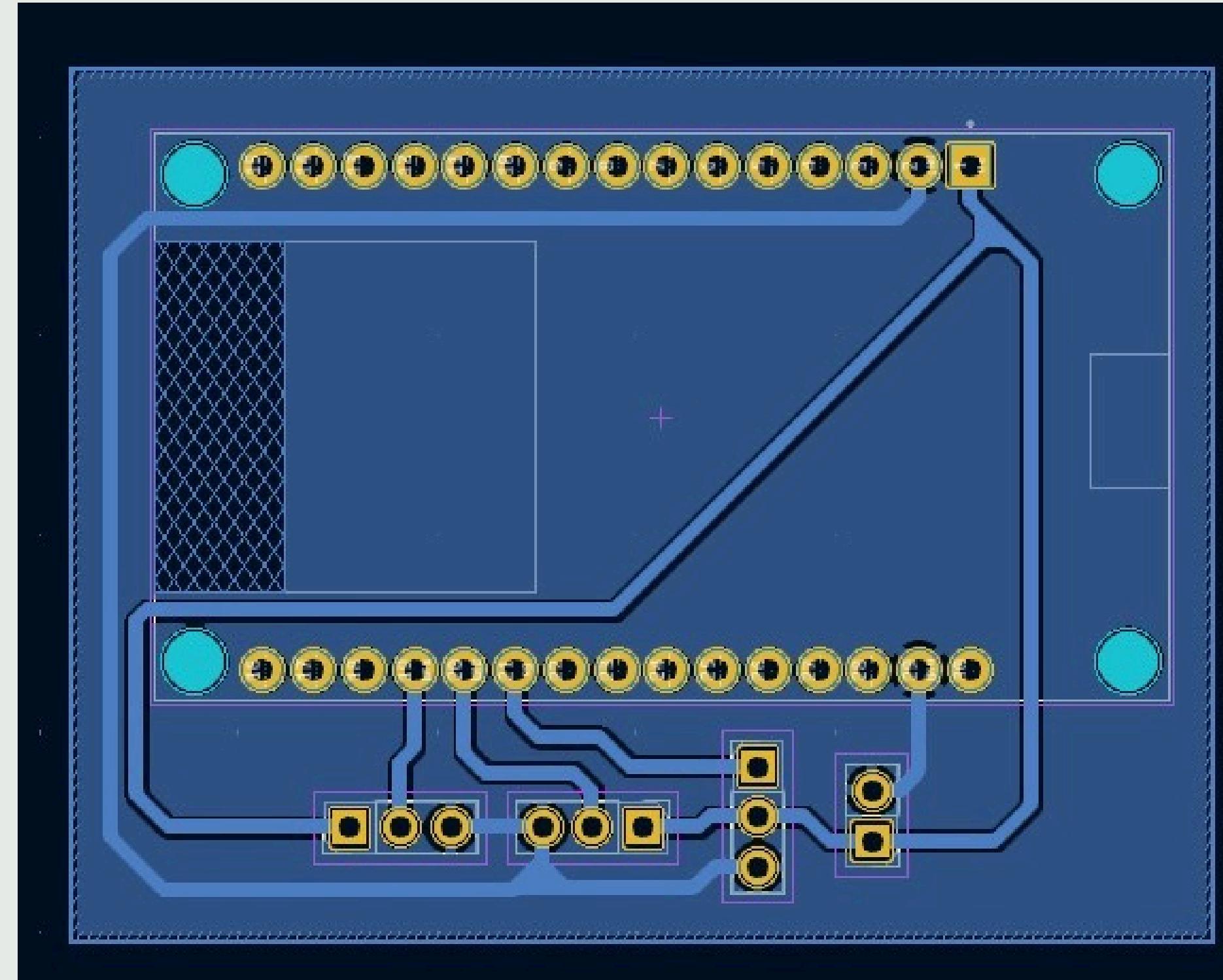


# Hardware





# Hardware





# Firmware

The screenshot shows a Firefox browser window running a Wokwi project. The project interface includes:

- IDE:** Shows the `sketch.ino` file with the following code:

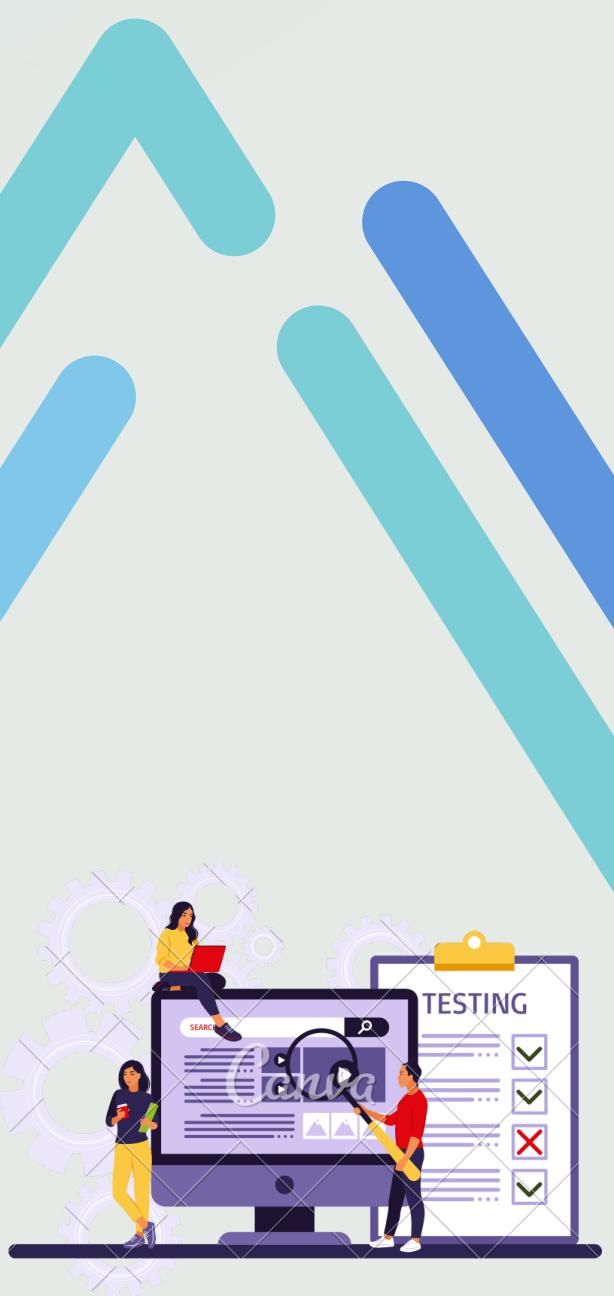
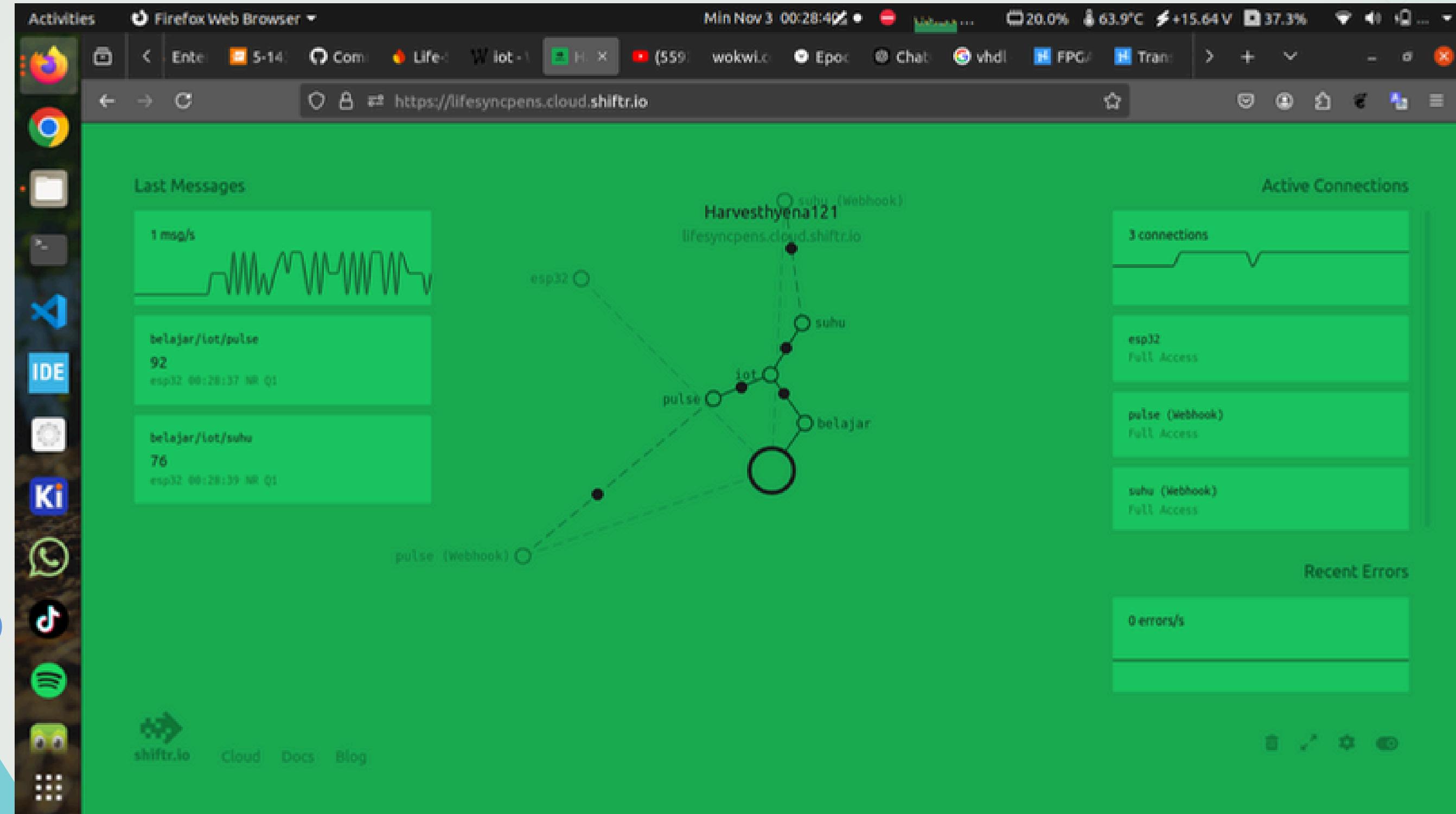
```
1 #include <WiFi.h>
2 #include <MQTT.h>
3 #include <NusabotSimpleTimer.h>
4
5 const char *ssid = "Wokwi-GUEST"; // Replace with your network SSID
6 const char *password = ""; // Replace with your network password
7
8 const int potPin1 = 34; // LM35 temperature sensor
9 const int potPin2 = 35; // Another sensor
10 const int potPin3 = 32; // Pulse sensor
11 int potValue1 = 0;
12 int potValue2 = 0;
13 int potValue3 = 0;
14
15 WiFiClient net;
16 MQTTClient client;
17 NusabotSimpleTimer timer;
18
19 void connectWiFi() {
20     Serial.print("Connecting to WiFi...");
21     WiFi.begin(ssid, password);
22
23     // Wait until connected
24     while (WiFi.status() != WL_CONNECTED) {
25         delay(500);
26         Serial.print(".");
27     }
28
29     // Connected to Wi-Fi
30     Serial.println("\nConnected to WiFi!");
31 }
```

- Simulation:** Displays a circuit diagram with an ESP32 board and three potentiometers connected to pins 34, 35, and 32.
- Output:** Shows the serial port output with the following text:

```
Potentiometer Value3: 92
Potentiometer Value1: 76
Potentiometer Value2: 16
Potentiometer Value3: 92
Potentiometer Value1: 76
Potenti
```



# Communication





# Communication

The screenshot shows a web browser window with the URL [Mesyncpens.cloud.shiftr.io/settings/webhooks](https://Mesyncpens.cloud.shiftr.io/settings/webhooks). The page is titled "Webhooks" and displays a list of configured webhooks. On the left, there's a sidebar with "Settings" and "Webhooks" selected. The main area shows six entries:

Name	URL	Path
pulse	<a href="https://lifesync-ectb4-default-rtdb.firebaseio.com/southeast1.firebaseio.com/belajar/iot/pulse">https://lifesync-ectb4-default-rtdb.firebaseio.com/southeast1.firebaseio.com/belajar/iot/pulse</a>	belajar/iot/pulse
suhu1	<a href="https://lifesync-ectb4-default-rtdb.firebaseio.com/southeast1.firebaseio.com/belajar/suhu1">https://lifesync-ectb4-default-rtdb.firebaseio.com/southeast1.firebaseio.com/belajar/suhu1</a>	belajar/iot/suhu1
suhu2	<a href="https://lifesync-ectb4-default-rtdb.firebaseio.com/southeast1.firebaseio.com/belajar/suhu2">https://lifesync-ectb4-default-rtdb.firebaseio.com/southeast1.firebaseio.com/belajar/suhu2</a>	belajar/iot/suhu2
pulse-device3	<a href="https://lifesync-ectb4-default-rtdb.firebaseio.com/southeast1.firebaseio.com/2045-03/pulse.json">https://lifesync-ectb4-default-rtdb.firebaseio.com/southeast1.firebaseio.com/2045-03/pulse.json</a>	2045-03/iot/pulse
suhu1-device3	<a href="https://lifesync-ectb4-default-rtdb.firebaseio.com/southeast1.firebaseio.com/2045-03/suhu1.json">https://lifesync-ectb4-default-rtdb.firebaseio.com/southeast1.firebaseio.com/2045-03/suhu1.json</a>	2045-03/iot/suhu1
suhu2-device3	<a href="https://lifesync-ectb4-default-rtdb.firebaseio.com/southeast1.firebaseio.com/2045-03/suhu2.json">https://lifesync-ectb4-default-rtdb.firebaseio.com/southeast1.firebaseio.com/2045-03/suhu2.json</a>	2045-03/iot/suhu2

Below the table, a note states: "Webhooks will appear as connections in the real-time graph." The browser interface includes a toolbar with various icons and a status bar at the bottom.



# Communication

The screenshot shows the Firebase Realtime Database console for the project 'lifesync'. The database structure is as follows:

```
belajar
  pulse
    204503
    userId123
      data: "89"
  suhu1
    204503
    userId123
    suhu2
  users
```

The database is located in Singapore (asia-southeast1). The 'pulse' node contains two entries: '204503' and 'userId123'. The 'userId123' entry has a child 'data' with the value '89'. The 'suhu1' node also contains three entries: '204503', 'userId123', and 'suhu2'. The 'users' node is currently empty.



# Pengujian



# Pengujian LM35

Sensor 1 LM35	Sensor 2 LM35	Selisih
29	28	1
47	46	1
77	79	2
132	130	2
146	144	2
128	130	2



## Kesimpulan



**Proyek ini menawarkan solusi inovatif untuk memantau kesehatan secara real-time. Menggunakan sensor detak jantung dan suhu tubuh yang terhubung ke platform IoT melalui ESP32 dan MQTT, data kesehatan dikirim cepat ke cloud dan diolah dengan fuzzy logic untuk analisis yang lebih akurat. Pengguna dapat mengakses hasilnya melalui aplikasi, dan sistem ini mendukung pemantauan mandiri serta telemedicine, dengan notifikasi otomatis saat ada anomali untuk situasi kritis.**



KAMPUS  
INOVASI



# Terimakasih

*Partner kami :*

