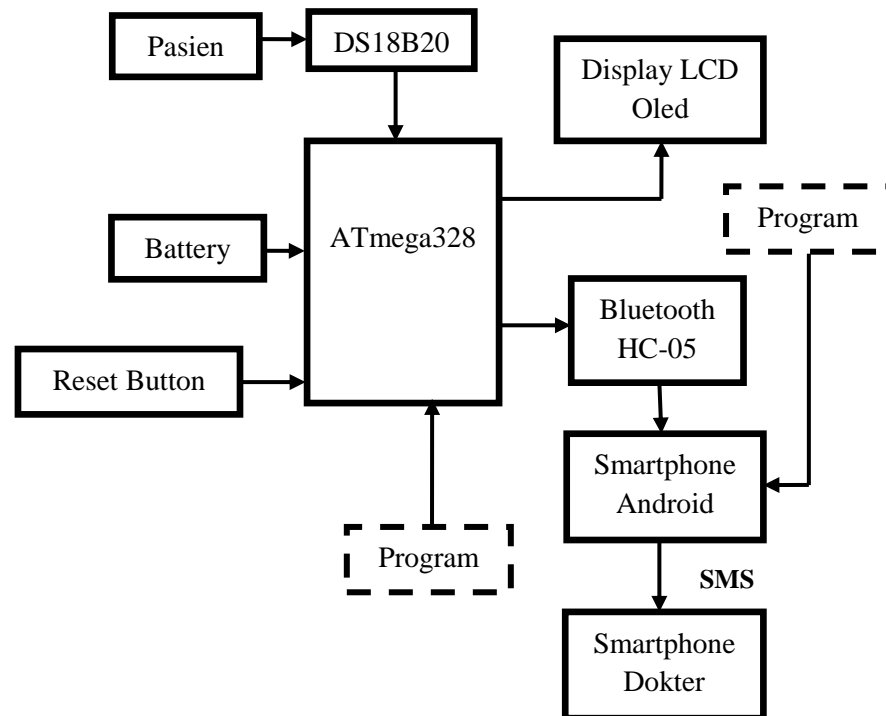


BAB III METODOLOGI PENELITIAN

3.1 Diagram Blok Sistem



Gambar 3. 1 Blok diagram alat keseluruhan

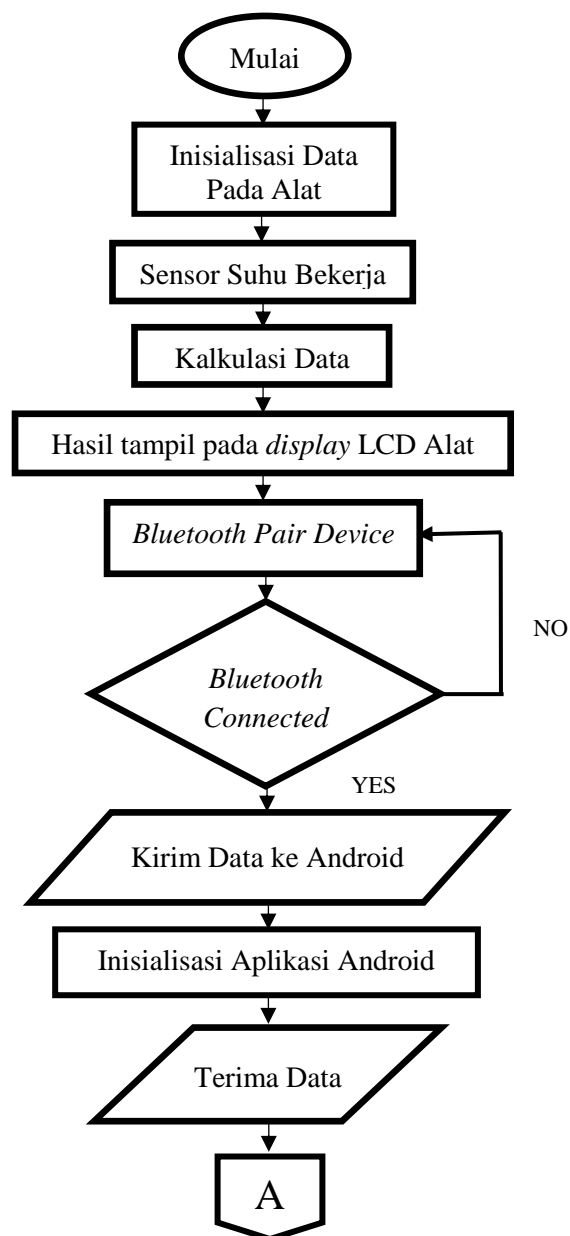
Penjelasan blok diagram parameter Suhu Tubuh :

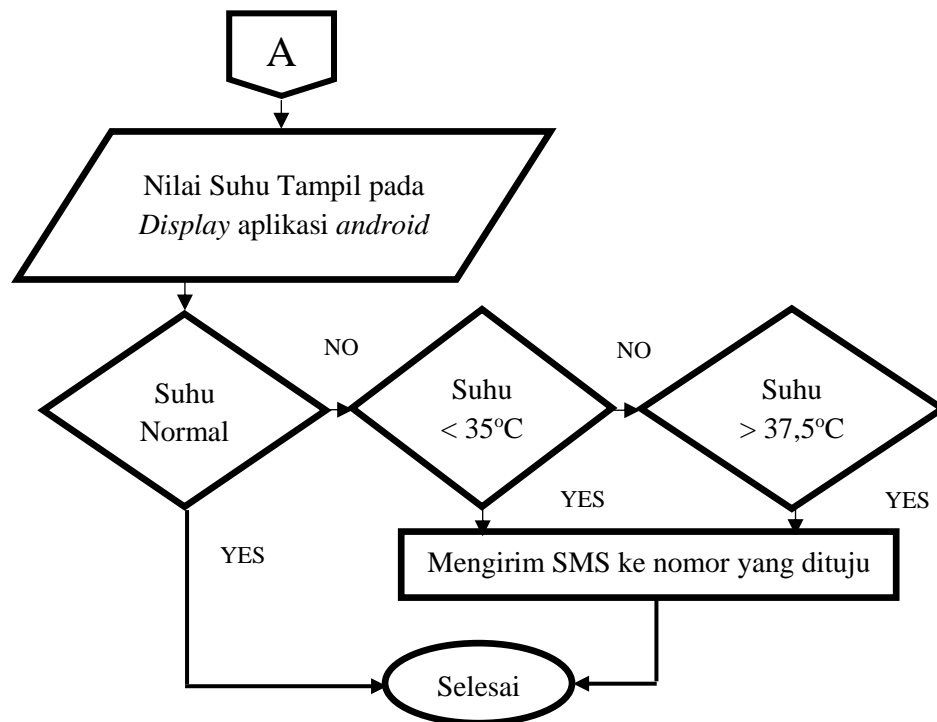
Pasien akan dipasangkan sensor suhu DS18B20 yang diletakkan pada ketiak pasien. Kemudian sensor tersebut akan mengukur tingkat suhu tubuh pasien dan hasilnya akan dikirim langsung ke *microcontroller*. Setelah masuk ke *microcontroller* hasil akan diolah dan kemudian akan ditampilkan pada LCD.. Hasil yang telah masuk ke *microntrroller* tadi juga akan dikirim ke aplikasi *Android* melalui *Bluetooth* dan akan tampil pada *interface* aplikasi *Android*. Jika hasil nilai suhu tubuh yang terhitung yaitu dibawah $35,5^{\circ}\text{C}$ atau diatas $37,5^{\circ}\text{C}$, maka aplikasi

Android akan mengirim SMS ke *user* (dokter, perawat, atau keluarga pasien) untuk mengetahui keadaan pasien dan dapat melakukan tindakan medis dengan cepat.

3.2 Diagram Alir Proses/Program

Berikut adalah diagram alir proses kerja alat dari pengambilan data suhu tubuh oleh sensor dan ditampilkan pada LCD *Oled*, kemudian mengirim data suhu ke aplikasi *android* untuk ditampilkan dan pengiriman SMS.





Gambar 3. 2 Diagram alir Kerja Alat dan Aplikasi *Android*.

Penjelasan diagram :

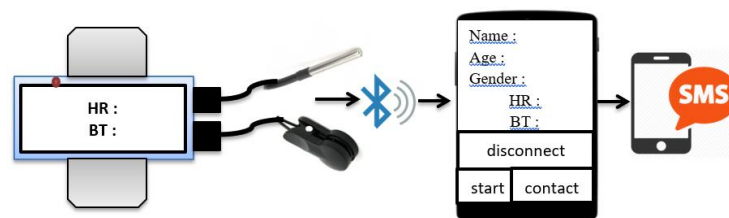
Saat pertama kali alat dinyalakan, maka alat akan melakukan inialisasi data. Setelah itu sensor suhu akan bekerja untuk membaca suhu tubuh pada pasien. Kemudian *microcontroller* akan melakukan kalkulasi data untuk menghitung data yang diambil dari pasien dan akan ditampilkan pada *display LCD Oled* pada alat. Data juga akan masuk kerangkaian *Bluetooth* untuk dikirim ke aplikasi *Android* dan akan ditampilkan pada aplikasi *Android* tersebut.

Saat aplikasi *Android* diaktifkan, maka aplikasi akan melakukan proses inialisasi data. Saat aplikasi telah melakukan inialisasi dan sudah terhubung ke alat oleh *Bluetooth*, maka aplikasi otomatis menerima data dan menampilkan data tersebut. Untuk proses penghubungan antara pengirim dan penerima data, yaitu

dengan menggunakan dan menyesuaikan ID *Bluetooth* yang mengirim. Jika ID *Bluetooth* dari pengirim sudah ditemukan pada *Android*, maka penerima dapat memilih ID *Bluetooth* tersebut dan jika sudah terhubung maka data sudah dapat ditampilkan. Jika suhu bernilai dibawah $35,5^{\circ}\text{C}$ atau diatas $37,5^{\circ}\text{C}$, aplikasi akan langsung mengirim SMS ke nomor yang dituju. Untuk proses pengiriman data tidak normal dengan menggunakan SMS harus menuliskan nomor kontak yang akan dituju dahulu dibagian kolom pada aplikasi *android*.

3.3 Diagram Mekanis Sistem

Gambar mekanisme alat modul yang telah penulis buat dan bentuk aplikasi *android* dapat dilihat pada Gambar 3.3 berikut.



Gambar 3. 3 Mekanisme alat.

Penjelasan gambar:

Pada modul alat terdapat LCD *Oled*, tombol *power* dan tombol *reset*.

Untuk sensor suhu tubuh menggunakan sensor DS18B20. Data hasil pengukuran akan ditampilkan pada LCD dan dikirim ke *android* melalui *bluetooth* HC-05.

Kemudian pada aplikasi *android* akan menampilkan data suhu tubuh. Jika data suhu tubuh pasien berada di luar batas normal ($35^{\circ}\text{C} - 37,5^{\circ}\text{C}$), maka akan mengirim SMS mengenai gejala yang diderita pasien secara otomatis ke ponsel tujuan.

3.4 Alat dan Bahan

3.4.1 Bahan yang Digunakan

Adapun komponen-komponen alat yang digunakan oleh penulis, antara lain :

1. Atmega328
2. LCD
3. *Bluetooth* HC-05
4. Sensor Suhu DS18B20
5. Sensor *Ear Clip*
6. LED
7. Kapasitor
8. *Resistor*
9. *Push Button*
10. *Multiturn*

3.4.2 Alat yang Digunakan

Alat-alat yang digunakan oleh penulis untuk pembuatan alat, antara lain :

1. PCB
2. Tang Potong
3. Osiloskop
4. Solder Listrik
5. *Multimeter*

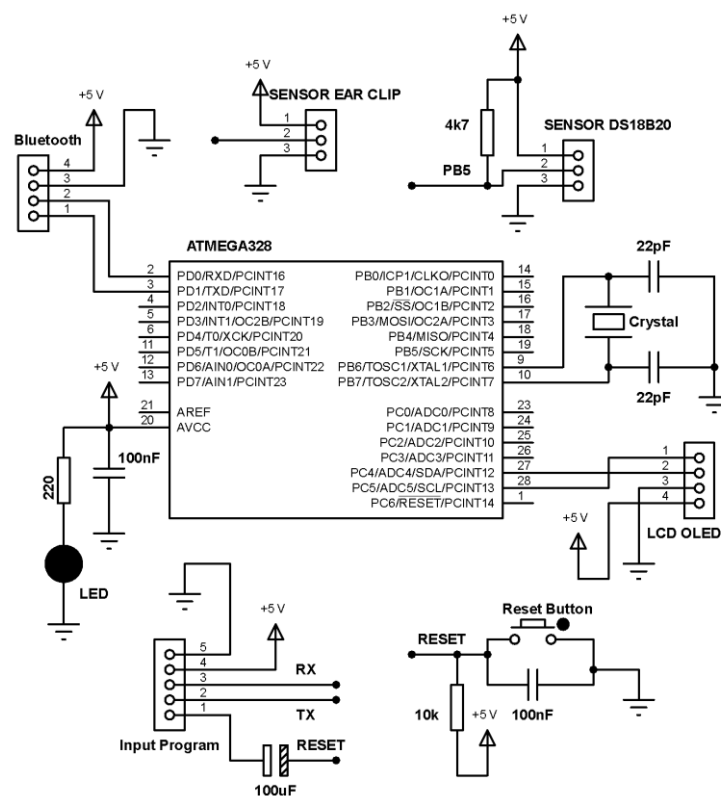
3.5 Perancangan Perangkat Keras

Perangkat keras dirancang untuk mengendalikan cara kerja dari alat yang penulis buat. Adapun perangkat keras yang dibutuhkan untuk pembuatan alat

monitoring suhu tubuh ini adalah rangkaian *microcontroller* dan rangkaian sensor suhu DS18B20.

3.5.1 Blok Rangkaian *Minimum System*

Rangkaian *minimum system* berfungsi untuk mengendalikan kerja dari perhitungan sensor suhu DS18B20 dan menampilkan hasil data suhu ke LCD *Oled* dengan bantuan program. Rangkaian ini menggunakan IC *ATmega 328 Arduino*. Adapun gambar rangkaian *minimum system Atmega 328* dapat dilihat pada Gambar 3.4 berikut.



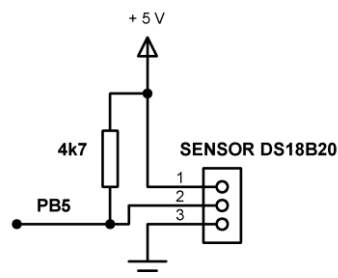
Gambar 3. 4 Rangkaian *Microcontroller*.

Prinsip kerja dari rangkaian *minimum system* ini yaitu menggunakan IC *ATmega 328* untuk melakukan perintah terhadap rangkaian lain seperti sensor suhu

dan LCD *Oled*. Rangkaian ini memerlukan tegangan *input* +5 V dan *ground* untuk dapat bekerja. Konektor *Input Program* berguna untuk men-*download* program *Arduino* untuk dimasukkan ke *microcontroller*. Konektor LCD *Oled* berguna untuk menghubungkan LCD dengan *microcontroller* untuk menampilkan *output* data suhu tubuh. Konektor *bluetooth* berguna untuk menghubungkan modul *bluetooth* HC-05 dengan pin TX dan RX pada *microcontroller* untuk proses pengiriman data suhu tubuh ke *android*. Konektor sensor DS18B20 berguna untuk menghubungkan sensor suhu tubuh dengan *microcontroller* yang keluaran dari sensor akan masuk ke pin PB5. Rangkaian *reset* yang mana terdapat tombol *reset* berguna untuk mengulang kembali kerja program. LED berguna sebagai indikator bahwa rangkaian *minimum system* telah dimasuki tegangan dan bekerja.

3.5.2 Blok Rangkaian Sensor DS18B20

Rangkaian ini berfungsi untuk menghubungkan sensor suhu dengan *microcontroller*. Adapun gambar rangkaian sensor suhu dapat dilihat pada Gambar 3.5 sebagai berikut.



Gambar 3. 5 Rangkaian Sensor Suhu DS18B20.

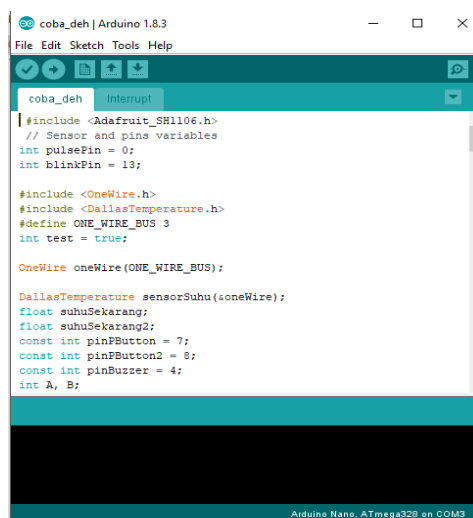
Prinsip kerja dari rangkaian ini yaitu pada sensor DS18B20 terdapat 3 pin seperti pin *Vcc*, *ground*, dan *output* data. Tegangan yang dibutuhkan sensor ini untuk

bekerja adalah +5 V. *Output data* dari sensor akan masuk ke pin digital 13 *microcontroller* yaitu pada pin PB5.

3.6 Perancangan Perangkat Lunak

Perangkat lunak dirancang untuk mengendalikan cara kerja dari alat *monitoring* detak jantung dan suhu tubuh berbasis interaksi *android* dilengkapi *telemedicine* (Parameter Suhu Tubuh) ini. Adapun perangkat lunak yang dibutuhkan untuk pembuatan alat yaitu *software arduino* dan *MIT App Inventor*.

3.6.1 Software Arduino



```

coba_deh | Arduino 1.8.3
File Edit Sketch Tools Help
coba_deh Interrupt
#include <Adafruit_SH1106.h>
// Sensor and pins variables
int pulsePin = 0;
int blinkPin = 13;

#include <OneWire.h>
#include <DallasTemperature.h>
#define ONE_WIRE_BUS 3
int test = true;

OneWire oneWire(ONE_WIRE_BUS);

DallasTemperature sensorSuhu(oneWire);
float suhuSekarang;
float suhuSekarang2;
const int pinPButton = 7;
const int pinPButton2 = 8;
const int pinBusser = 4;
int A, B;
  
```

Gambar 3. 6 *Software Arduino.*

Software arduino ini digunakan sebagai pengaksesan dari *hardware* yang telah di buat. Dimana semua perintah yang diinginkan dapat ditanamkan pada minimum sistem nanti, seperti perintah data yang akan di olah atau ditampilkan, kemudian perintah data yang akan dikirim ke *android*. Adapun listing program yang dibuat menggunakan aplikasi *Arduino* antara lain:

a. Program Sensor Suhu DS18B20

```
#include <OneWire.h>

int DS18S20_Pin = 13;

OneWire ds(DS18S20_Pin);

void setup(void) {
  Serial.begin(115200); }

void loop(void) {

  float temperature = getTemp();

  Serial.println(temperature);

  delay(2000); }

float getTemp(){

byte data[12];

  byte addr[8];

  if ( !ds.search(addr)) {

ds.reset_search();

  return 0; }

  if ( OneWire::crc8( addr, 7) != addr[7]) {

  Serial.println("CRC is not valid!");

  return 0; }

  if ( addr[0] != 0x10 && addr[0] != 0x28) {

  Serial.print("Deviceot recognized");

  return 0; }

ds.reset();
```

```

ds.select(addr);

ds.write(0x44,1);

byte present = ds.reset();

ds.select(addr);

ds.write(0xBE);

for (int i = 0; i < 9; i++) {

  data[i] = ds.read();  }

ds.reset_search();

byte MSB = data[1];

byte LSB = data[0];

float tempRead = ((MSB << 8) | LSB);

  float TemperatureSum = tempRead / 16;

return TemperatureSum; }

```

Penjelasan program:

Program di atas menggunakan library One Wire yang diperlukan untuk pembacaan sensor suhu DS18B20 dengan memanggil “`#include<onewire.h>`”.

Kemudian *output* sensor suhu diatur masuk ke pin 13 pada *microcontroller* dengan menuliskan “`int DS18S20_Pin = 13; OneWire ds(DS18S20_Pin);` “. Untuk mengatur *baudrate* dengan menulis “`Serial.begin(115200);`” pada *void setup* dan *baudrate* yang digunakan menjadi 115200.

Pada *coding* “*float getTemp()*” memuat *coding* pengolahan data dari *output* sensor di *microcontroller* . kemudian *coding* “*float temperature = getTemp();*” untuk menyamakan definisi dari data yang berada di dalam variabel “*getTemp()*” dengan variabel “*temperature*”.

Untuk menampilkan *output* data yang telah di olah oleh *microcontroller* dengan menulis *coding* “*Serial.println(temperature);*” dan hasilnya dapat dilihat pada *serial monitor* dan mengatur *baudrate* 115200.

b. Program LCD *Oled*

```
#include <Adafruit_SH1106.h>

#define OLED_Address 0x3C

Adafruit_SH1106 oled(1);

void setup() {

    oled.begin(SH1106_SWITCHCAPVCC, OLED_Address);

    oled.setTextSize(1);

    oled.setTextColor(WHITE);

    oled.setCursor(33,20);

    oled.print("Welcome To");

    oled.setCursor(30,33);

    oled.print("Our Project");

    oled.display();
```

```

delay(1000);

oled.clearDisplay();

void loop(){

oled.setCursor(0,50);

oled.print(suhuSekarang);

oled.print(" C");

```

Penjelasan program:

“*#include <Adafruit_SH1106.h>*” berfungsi untuk memanggil *library* dari LCD *Oled* dengan tipe SH1106. Untuk memanggil tipe data dan pin dengan memanggil “*#define OLED_Address 0x3C*”.

Untuk menyatakan LCD bekerja pada program dengan menuliskan *coding* “*oled.begin(SH1106_SWITCHCAPVCC, OLED_Address);*” pada *void setup*. Kemudian “*oled.setTextSize(1);*” berfungsi untuk mengatur besar tulisan yang ingin ditampilkan. *Coding* “*oled.setTextColor(WHITE);*” berfungsi untuk mengatur warna tulisan pada *display* dan yang diatur pada program yaitu warna putih.

Untuk mengatur posisi letak tulisan pada *display* dengan *coding* “*oled.setCursor(33,20);*” yang berarti letak vertikal berada pada titik 33 dan letak horizontal tulisan berada di titik 20. *Coding* “*oled.print("Welcome To");*” berfungsi untuk menampilkan tulisan dan tulisan “*Welcome To*” akan tampil pada *display*.

Coding “*oled.display();*” berfungsi untuk menyatakan bahwa letak tulisan dan tulisan yang ingin ditampilkan ke LCD sudah benar. *Coding* “*delay(2000);*” berfungsi untuk menyatur jeda waktu yang tampil selama 2000 ms atau 2 detik. Dan *coding* “*oled.clearDisplay();*” berfungsi untuk menghapus tampilan selama waktu jeda yang diatur dan menampilkan tampilan selanjutnya.

c. Program Pengiriman Data Serial

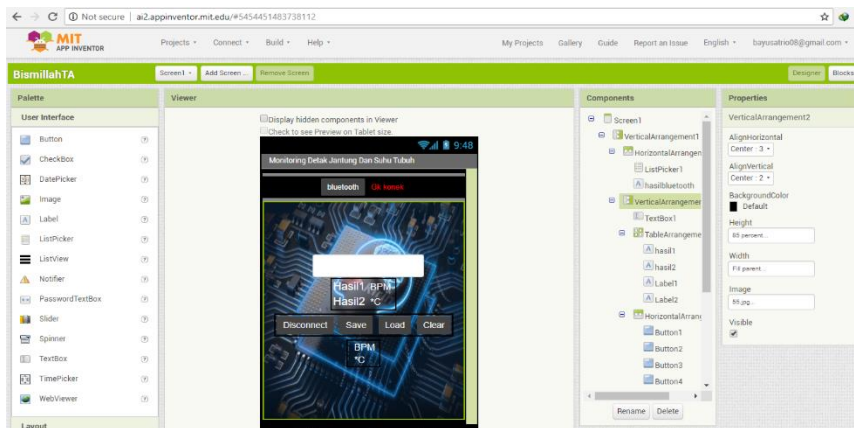
```
void setup() {
  Serial.begin(115200); }
void loop() {
  Serial.println(temperature); }
```

Penjelasan program:

Coding “*Serial.begin(115200);*” berfungsi untuk menginisialisasikan besar *baudrate* yang telah diatur pada *bluetooth* untuk kerja pengiriman data.

Untuk proses mengirimkan data dengan menulis *coding* “*Serial.println(temperature);*” . variabel “*temperature*” digunakan sebagai variabel yang didalamnya terdapat data suhu tubuh yang akan dikirim melalui *data serial*.

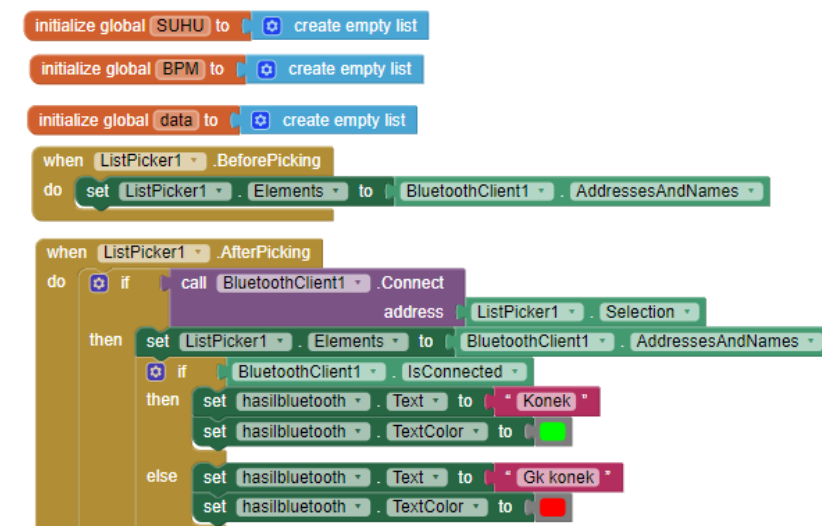
3.6.2 Software MIT App Inventor



Gambar 3. 7 MIT App Inventor.

MIT *App Inventor* adalah sebuah *website* yang menyediakan fasilitas untuk membuat sebuah aplikasi *android*, disini aplikasi yang telah penulis buat dan penulis desain untuk dapat menerima data, menyimpan data dan mengirim SMS ke nomor yang akan dituju. Adapun listing program yang dibuat menggunakan *website* ini antara lain:

a. Program Penerimaan Data Serial



```

when ListPicker1 .AfterPicking
do
  if call BluetoothClient1 .Connect
      address ListPicker1 . Selection
  then
    set ListPicker1 . Elements to BluetoothClient1 . AddressesAndNames
  if BluetoothClient1 . IsConnected
  then
    set hasilbluetooth . Text to "Konek"
    set hasilbluetooth . TextColor to green
  else
    set hasilbluetooth . Text to "Gk konek"
    set hasilbluetooth . TextColor to red

```

Penjelasan program:

```

initialize global SUHU to create empty list
initialize global BPM to create empty list
initialize global data to create empty list

```

Pada *coding* berfungsi untuk menginisialisasi suatu variabel yang akan dipakai. "SUHU" merupakan salah satu variabel.

```

when ListPicker1 .BeforePicking
do
  set ListPicker1 . Elements to BluetoothClient1 . AddressesAndNames

```

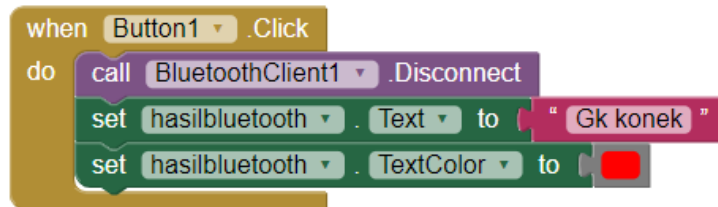
Berfungsi untuk membuat suatu *list bluetooth* tentang nama dan alamat dari *bluetooth* pengirim yang akan disambungkan.

```

when ListPicker1 .AfterPicking
do
  if call BluetoothClient1 .Connect
      address ListPicker1 . Selection
  then
    set ListPicker1 . Elements to BluetoothClient1 . AddressesAndNames
  if BluetoothClient1 . IsConnected
  then
    set hasilbluetooth . Text to "Konek"
    set hasilbluetooth . TextColor to green
  else
    set hasilbluetooth . Text to "Gk konek"
    set hasilbluetooth . TextColor to red

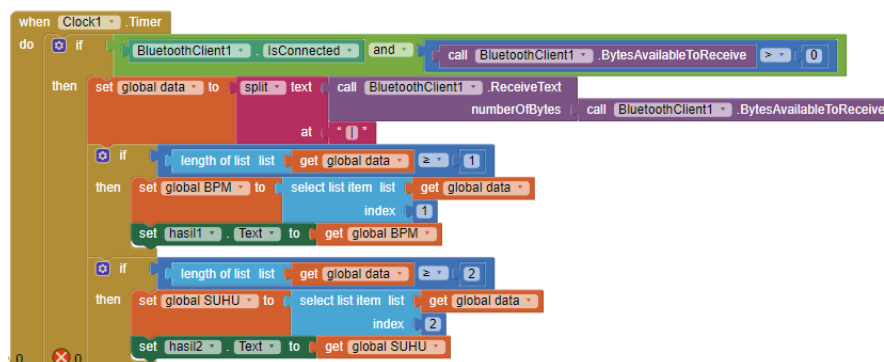
```

Berfungsi untuk melakukan pemilihan nama *bluetooth* di *list* yang akan dihubungkan. Jika *bluetooth* telah dihubungkan, maka tombol "*bluetooth*" akan berganti tulisan dari "Gk konek" menjadi "Konek"



Berfungsi untuk menginisialisasikan tombol 1 atau tombol "Disconnect" untuk memutus hubungan *bluetooth* dan menghentikan pengiriman data. Tombol *bluetooth* akan berganti nama dari "Konek" menjadi "Gk konek".

b. Program Menampilkan data di *Android*



Penjelasan Program :

Berfungsi untuk menampilkan data yang telah dikirim oleh *bluetooth* berupa data suhu tubuh pasien dari modul alat. "*global data*" merupakan variabel yang menyimpan semua data yang dikirim oleh *bluetooth*.

Kemudian salah satu data tersebut akan dibagi ke bagian variabel “*global SUHU*” agar data yang ditampilkan data suhu tubuh saja dan memberi tanda *split* (|) sebagai pembeda data agar tidak bersambung saat ditampilkan, dan juga mengatur *index* data yaitu 2 mengikuti pengaturan yang dibuat di program *Arduino* sehingga data yang kedua yang akan dibaca pada program ini.

c. Program Penyimpanan Data Terakhir

```

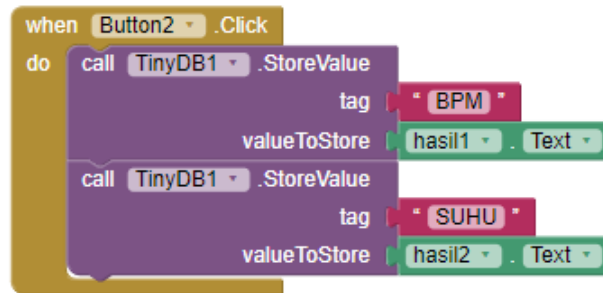
when Button2 .Click
do
  call TinyDB1 .StoreValue
  tag "BPM"
  valueToStore hasil1 .Text
  call TinyDB1 .StoreValue
  tag "SUHU"
  valueToStore hasil2 .Text

when Button4 .Click
do
  call TinyDB1 .ClearAll
  set DataTersimpan1 .Text to " "
  set DataTersimpan2 .Text to " "

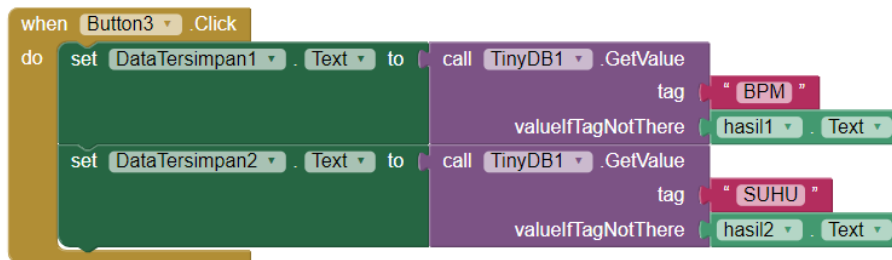
when Button3 .Click
do
  set DataTersimpan1 .Text to call TinyDB1 .GetValue
  tag "BPM"
  valueIfTagNotThere hasil1 .Text
  set DataTersimpan2 .Text to call TinyDB1 .GetValue
  tag "SUHU"
  valueIfTagNotThere hasil2 .Text

```

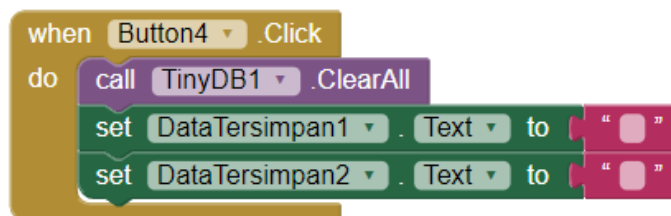
Penjelasan program:



Berfungsi untuk menginisialisasikan tombol 2 atau tombol "Save" untuk menyimpan data terakhir suhu tubuh pada aplikasi.



Berfungsi untuk menginisialisasikan tombol 3 atau tombol "Load" untuk menampilkan data terakhir suhu tubuh yang telah disimpan pada aplikasi.

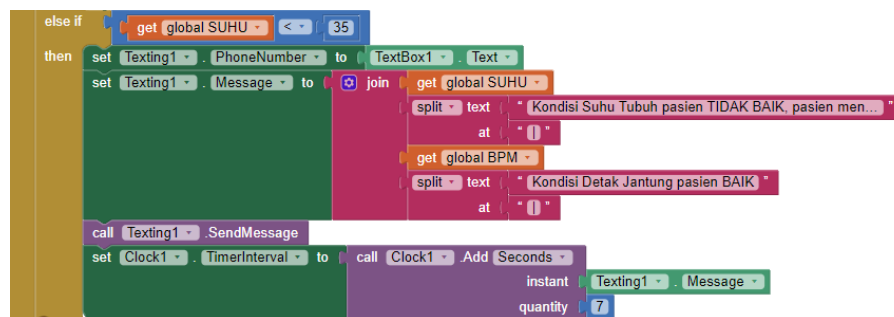


Berfungsi untuk menginisialisasikan tombol 4 atau tombol "Clear" untuk menghapus data terakhir suhu tubuh yang telah ditampilkan pada aplikasi.

d. Program Pengiriman SMS



Penjelasan program:



Berfungsi untuk pengiriman SMS otomatis saat data pada variabel “*global SUHU*” berada dibawah nilai 35 maka akan mengirimkan nilai tersebut dan kondisi pasien ke nomor yang telah ditulis pada *TextBox* di aplikasi *Android*. “*TimerInterval*” berfungsi untuk mengatur waktu pengiriman SMS saat program berlaku dan maksud “7” adalah akan mengirim SMS dihitung selama 7 detik dari saat data tersebut tercapai.

3.7 Variabel Penelitian

3.7.1 Variabel Bebas

Sebagai variabel bebas yaitu suhu tubuh.

3.7.2 Variabel Tergantung

Sebagai variabel tergantung yaitu alat pengukur suhu tubuh pada ketiak.

3.7.3 Variabel Terkendali

Sebagai variabel terkendali yaitu *microcontroller*, *Bluetooth*, dan aplikasi *Android*.

3.8 Definisi Operasional

Berikut adalah definisi operasional dari tugas akhir penulis.

Tabel 3.1 Definisi operasional

Variabel	Definisi	Alat Ukur	Hasil Ukur	Skala-ukur
Suhu Tubuh Manusia	Besaran panas yang keluar dari tubuh	Termometer	Suhu normal: 36- 37,5 °C Hipotermia: < 36 °C Hipertermia: >37,5 °C	Rasio (°C)
Sensor Suhu Tubuh	Pendeteksi Suhu Badan	Osiloskop, <i>Avometer</i>	0-100 mV	Rasio (mV)
<i>Bluetooth</i>	Untuk mengirim data BPM ke aplikasi <i>Android</i>	<i>Multimeter</i>	0. Bekerja 1.Tidak bekerja	Nominal
<i>Microcontroller</i>	Rangkaian pengendali sistem yang diatur oleh program	<i>Multimeter</i>	0. Bekerja 1.Tidak bekerja	Nominal

<i>Android</i>	Untuk menampilkan nilai data suhu tubuh	-	0. Bekerja 1. Tidak bekerja	Nominal
----------------	---	---	--------------------------------	---------

3.9 Teknik Analisis Data

Setelah dilakukan pengukuran maka akan dilakukan perhitungan data yang diperoleh sehingga dapat dianalisa menggunakan rumus, antara lain :

3.9.1 Rata – rata

Rata – rata adalah bilangan yang di dapat dari hasil pembagian jumlah nilai data oleh banyaknya data dalam kumpulan tersebut.

$$\text{Rumus rata – rata adalah : } \bar{X} = \frac{\sum X_n}{n} \quad (3.1)$$

Dimana : \bar{X} = rata – rata

$X_1, \dots, X_n (\sum X_n)$ = jumlah nilai data

N = banyak data (1,2,3...,n)

3.9.2 Error (Kesalahan)

Error (kesalahan) adalah selisih antara *mean* terhadap masing – masing data.

$$\text{Rumus } Error \text{ adalah : } Error \% = \frac{\bar{Y} - \bar{X}}{\bar{Y}} \times 100\% \quad (3.2)$$

Dimana : \bar{Y} = Rata – rata data pembanding atau kalibrator

\bar{X} = Rata – rata data modul

3.9.3 Standart Deviasi

Standart Deviasi adalah suatu nilai yang menunjukkan tingkat (derajat) variasi kelompok data atau ukuran standart penyimpangan dari *mean*.

Rumus Standart Deviasi adalah :

$$\text{STD} = \sqrt{\frac{(X_1 - \bar{X})^2 + (X_2 - \bar{X})^2 + \dots + (X_n - \bar{X})^2}{(n - 1)}} \quad (3.3)$$

Dimana : STD = Standart deviasi

X = rata – rata

X₁,...,X_n = nilai data

n = banyak data (1,2,3...n)

3.10 Langkah Penggunaan Alat

1. Menghubungkan *Ear Clip sensor* dan sensor DS18B20 ke alat.
2. Memasangkan *Ear Clip Sensor* dan sensor DS18B20 ke pasien.
3. Menyalakan alat dengan menekan tombol *power*. *Bluetooth* juga akan otomatis menyala.
4. Memastikan *Ear Clip Sensor* bekerja untuk mengambil data dengan melihat nyala LED warna merah (indikator LED sensor) berkedip dengan teratur, maka pengambilan data dari sensor sudah benar.
5. Menghubungkan alat dengan *smartphone android*, dengan menyalakan *bluetooth* pada *android* dan mencari nama *bluetooth* dari alat.
6. Jika berhasil tersambung, data dari alat akan muncul pada *display android*.

7. Data terakhir alat dapat disimpan sementara pada aplikasi *android* dengan menekan tombol “*Save*”. Kemudian untuk melihat data yang telah disimpan dengan menekan tombol “*Load*” dan untuk menghapus data simpanan yang telah muncul pada *display* dengan menekan tombol “*Clear*”.
8. Mengisi nomor telpon ke bagian kolom yang terdapat pada aplikasi agar saat data objek di luar keadaan normal dapat otomatis mengirim pesan kepada nomor yang dituju.

3.11 Pengujian dan Hasil Pengujian

Setelah membuat modul, maka langkah berikutnya melakukan pengujian dan pengukuran. Untuk itu penulis, melakukan pendataan melalui beberapa tahap proses pengukuran dan pengujian. Tujuan pengukuran dan pengujian adalah untuk mengetahui kepekatan dari pembuatan modul dan memastikan masing- masing bagian (komponen) dari seluruh rangkaian modul telah berfungsi sesuai apa yang telah direncanakan.

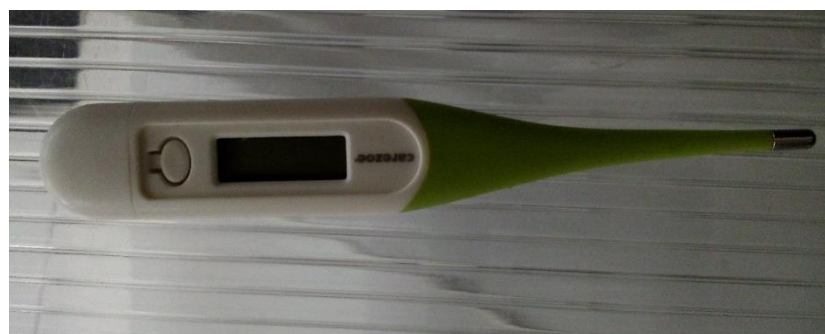
Langkah-langkah pengukuran dan pengujian modul ini dapat diuraikan dalam beberapa tahap, sebagai berikut:

1. Menyiapkan alat yang dibutuhkan, terutama modul alat dan alat pembanding.
2. Menyiapkan tabel untuk hasil pengukuran.
3. Menguji modul dengan mengadakan pengukuran suhu tubuh pada *thermometer digital*.

4. Mencatat hasil pengukuran dan perhitungan dalam tabel yang telah disediakan.
5. Melakukan perhitungan terhadap hasil pengukuran untuk mengetahui tingkat rata-rata, standar deviasi dan *error*.

Dalam pengukuran suhu tubuh ini penulis akan membandingkannya dengan alat *thermometer digital* yang sudah layak digunakan dengan alat modul yang dibuat oleh penulis. Berikut alat pembanding beserta spesifikasi alat:

1. Nama : *Thermometer digital*
2. Merk : *Carezoe*
3. Jenis : *Digital Thermometer*
4. Tipe : KD3414
5. Waktu Pengukuran : 60 detik
6. Jarak Pengukuran : 32.0 °C – 42.9 °C



Gambar 3. 8 *Thermometer Digital.*

Alat pembanding ini sebagai acuan dalam pengukuran dan perhitungan suhu tubuh. Pengujian dilakukan dengan membandingkan nilai suhu tubuh antara alat pembanding dengan modul alat secara bersamaan.

