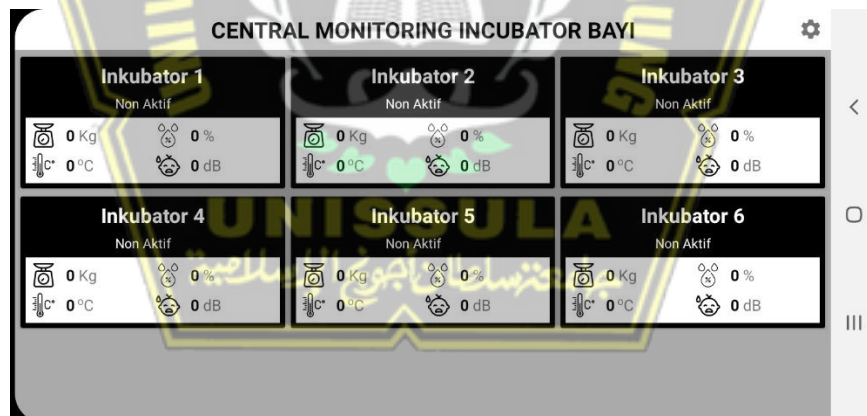


LAMPIRAN I

1. Gambar hasil pembuatan *hardware* alat prototipe inkubator bayi berbasis IoT



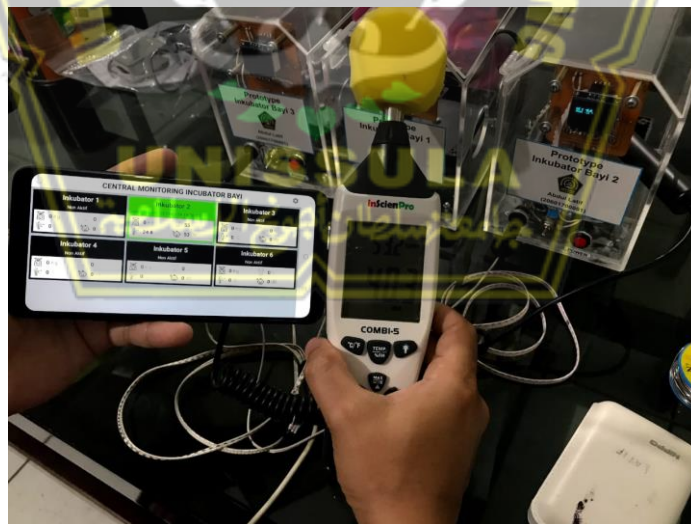
2. Gambar hasil pembuatan software aplikasi android sentral monitoring inkubator bayi berbasis IoT pada HP Android



3. Gambar hasil pembuatan software aplikasi android sentral monitoring inkubator bayi berbasis IoT pada HP Android



4. Gambar saat proses pengambilan data kelembapan prototipe inkubator bayi dengan alat pembandi merk envirometer



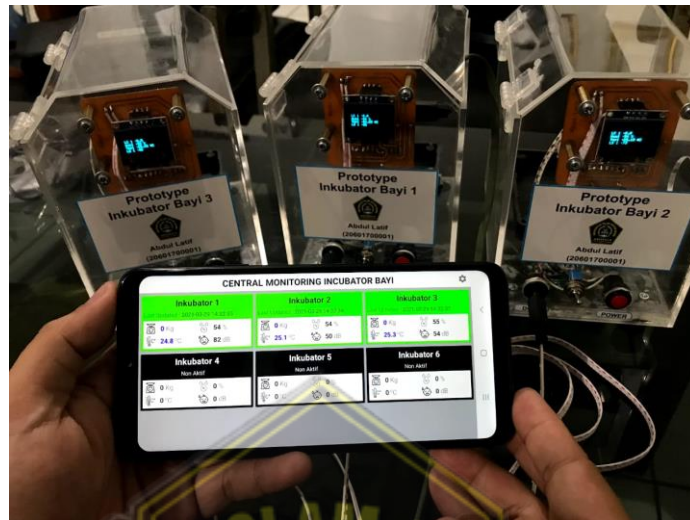
5. Gambar saat proses pengambilan data suhu antara prototipe inkubator bayi dengan alat pembanding merk envirometer



6. Gambar saat proses pengambilan data sensor suara antara prototipe inkubator bayi dengan alat pembanding merk envirometer



7. Gambar saat pengujian terintegrasi antara alat dengan HP Android



LAMPIRAN II

1. Program Arduino

Program Inkubator Bayi

```

#include <SoftwareSerial.h>
#include <ESP8266WiFi.h>
#include <ESP8266HTTPClient.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#include <HX711.h>
// #include <DHT.h>
#include <dhtnew.h>
// OLED
#define SCREEN_WIDTH 128 // OLED display width, in pixels
#define SCREEN_HEIGHT 64 // OLED display height, in pixels
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, -1);
// Sensor Suhu dan Kelembapan DHT22
#define DHTPIN 2 // what digital pin we're connected to
DHTNEW dht(DHTPIN);
// Sensor Berat HX711
const int LOADCELL_DOUT_PIN = 14;
const int LOADCELL_SCK_PIN = 0;
HX711 scale;
// ON/OFF pengiriman data inkubator
#define DTSEND 16
// data awal
const char* ssid = "TUNAS INTI";
const char* password = "tersenyumlah";
const char* fingerprint =
"7B:38:8B:A9:64:0D:A4:BE:40:BD:4E:C2:3D:96:5F:70:2B:F7:A4:68";

```

```
String idAlat = "1";
double berat = 1;
double suhu = 1;
int kelembapan = 1;
int tangisan = 1;
const int sampleWindow = 50;
bool bstatus_alat = false;
void setup() {
  Wire.begin();
  Serial.begin(9600);
  pinMode(DTSEND,INPUT);
  setting_wifi();
  settingawal_oled();
  display.clearDisplay();
  display.setTextSize(1);
  display.setTextColor(WHITE);
  display.setCursor(55, 10);
  display.println("LATIF");
  display.display();
  display.setCursor(40, 20);
  display.println("TESISKU");
  display.display();
  delay(500);
  display.clearDisplay();
  display.setTextColor(WHITE);
  display.setCursor(10, 10);
  display.println("SETTING");
  display.display();
  display.setCursor(10, 20);
  display.println("LOAD CELL");
  display.display();
```

```

delay(500);
scale.begin(LOADCELL_DOUT_PIN, LOADCELL_SCK_PIN);
scale.set_scale(2280.f);           // this value is obtained by calibrating the
scale with known weights; see the README for details
scale.tare();           // reset the scale to 0
display.clearDisplay();
display.setTextColor(WHITE);
display.setCursor(10, 10);
display.println("SETTING");
display.display();
display.setCursor(10, 20);
display.println("DHT22");
display.display();
delay(500);
//dht.begin();
dht.setDisableIRQ(true);
//ESP.wdtDisable();
}
void loop() {
  delay(500);
  //digitalWrite(LED_BUILTIN, HIGH);
  delay(500);
  if (WiFi.status() == WL_CONNECTED) { //Check WiFi connection status
    if(digitalRead(DTSEND) == HIGH){
      if(!bstatus_alat){
        bstatus_alat = data_alat_status(!bstatus_alat);
      }
      //ambil data berat
      float BB = scale.get_units();
      berat = ((6.06*BB) - 16.35)/1000;
      if (berat < 0) {

```

```

    berat = 0;
}
//ambil data suhu dalam celcius
dht.read();
//float t = dht.readTemperature();
float t = dht.getTemperature();
    suhu = t;
//float f = dht.readTemperature(true);
//float hif = dht.computeHeatIndex(f, h);
//float hic = dht.computeHeatIndex(t, h, false);
//ambil data kelembapan
//float h = dht.readHumidity();
float h = dht.getHumidity();
kelembapan = h-23;
    //ambil data suara tangisan
tangisan = sensor_suara();
    //munculkan di OLED
display.clearDisplay();
display.setTextColor(WHITE);
display.setCursor(10, 20);
display.print("LC 1 :");
display.print(berat,2);
display.display();
display.setTextColor(WHITE);
display.setCursor(10, 30);
display.print("Hum1 :");
display.print(kelembapan);
display.print(" %");
display.display();
display.setTextColor(WHITE);
display.setCursor(10, 40);

```



```

display.print("Tmp1 :");
display.print(suhu);
display.print(" *C");
//display.print(f);
//display.print(" *F\t");
display.display();
//display.setTextSize(1);
display.setTextColor(WHITE);
display.setCursor(10, 50);
display.print("LS 1 :");
display.print(tangisan);
display.display();
delay(200);
bstatus_alat=data_kondisi_update();
}else{
  Serial.println("Inkubator kondisi OFF, tidak mengirimkan data.");
  bstatus_alat=false;
  data_alat_status(bstatus_alat);
}
}else{
  Serial.println("Tidak mengirimkan data. Cek WIFI");
}
Serial.println("----- ");
}

```

Program Status Alat

```

bool data_alat_status(bool bstatus_alat){
  bool rstatus_alat = false;
  char* cstatus_alat = "nonaktif";
  if(bstatus_alat){
    cstatus_alat = "aktif";
  }
}

```

```

}
//setting cert
std::unique_ptr<BearSSL::WiFiClientSecure>client(new
BearSSL::WiFiClientSecure);
client->setFingerprint(fingerprint);
//kirm data ke server
String url =
"https://cminkubator.xyz/api/alatubahstatus.php?id_alat="+idAlat+"&status_alat=
"+cstatus_alat;
Serial.println("Parameter Data :");
Serial.println("id_lat      : "+(String)idAlat);
Serial.println("status_alat : "+(String)cstatus_alat);
Serial.println("Link API   : ");
Serial.println(url);
  HTTPClient http; //Declare an object of class HTTPClient
  http.begin(*client,url);
  //http.begin(url); //Specify request destination
  int httpCode = http.GET(); //Send the
request
  if (httpCode > 0) { //Check the returning code
    String payload = http.getString(); //Get the request response payload
    Serial.print("Response Data : ");
    Serial.println(payload);
    rstatus_alat = true;
  }else {
    Serial.println("Response Data Error :");
    Serial.println(http.errorToString(httpCode).c_str());
  }
  http.end(); //Close connection
  return rstatus_alat;
}

```

Program Update Data ke Server

```

bool data_kondisi_update(){
    bool rstatus_alat = false;

    //setting cert
    std::unique_ptr<BearSSL::WiFiClientSecure>client(new
BearSSL::WiFiClientSecure);
    client->setFingerprint(fingerprint);

    //kirm data ke server

    String url =
"https://cminkubator.xyz/api/kondisiupdate.php?id_alat="+idAlat+"&berat="+ber
at+"&suhu="+suhu+"&kelembapan="+kelembapan+"&tangisan="+tangisan;
    //String url = "https://cminkubator.xyz/api/alatcek.php?id_alat="+idAlat;
    Serial.println("Parameter Data :");
    Serial.println("berat   : "+(String)berat);
    Serial.println("suhu    : "+(String)suhu);
    Serial.println("kelembapan: "+(String)kelembapan);
    Serial.println("tangisan : "+(String)tangisan);
    Serial.println("Link API  :");
    Serial.println(url);
    HTTPClient http; //Declare an object of class HTTPClient
    http.begin(*client,url);
    //http.begin(url); //Specify request destination

    int httpCode = http.GET(); //Send the
request

    if (httpCode > 0) { //Check the returning code
        String payload = http.getString(); //Get the request response payload
        Serial.print("Response Data : ");
        Serial.println(payload);
        if(payload.indexOf("aktif") >= 0){
            rstatus_alat=true;
        }
    }
}

```

```

}else {
  Serial.println("Response Data Error :");
  Serial.println(http.errorToString(httpCode).c_str());
}
http.end(); //Close connection
return rstatus_alat;
}
Program sensor suara
int sensor_suara()
{
  unsigned long startMillis = millis();
  unsigned int peakToPeak = 20; // peak-to-peak level
  unsigned int signalMin = 1024;
  unsigned int signalMax = 0;
  while(millis() - startMillis < sampleWindow)
  {
    unsigned int sample = analogRead(A0);
    if(sample < 1024)
    {
      if(sample > signalMax)
      {
        signalMax = sample;
      }else if(sample < signalMin)
      {
        signalMin = sample;
      }
    }
  }
  peakToPeak = signalMax - signalMin;
  float db_float = map(peakToPeak, 20, 500, 29.5, 90);
  int db_int = (int) db_float;

```

```

return db_int;
}

```

Program Setting Wifi

```

void setting_wifi()
{
  Serial.print("CENTRAL MONITORING INCUBATOR1");
  Serial.print("OK");
  WiFi.begin(ssid, password);
  Serial.println("");
  while (WiFi.status() != WL_CONNECTED) {
    //digitalWrite(LED_BUILTIN,HIGH);
    delay(200);
    Serial.print(".");
    //digitalWrite(LED_BUILTIN,LOW);
    delay(200);
  }
  Serial.println("");
  Serial.print("Successfully connected to : ");
  Serial.println(ssid);
  Serial.print("IP address: ");
  Serial.println(WiFi.localIP());
  Serial.println("Device Ready!");
  Serial.println("");
}

```

Program Oled

```
void settingawal_oled()
{
  if(!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) { // Address 0x3D for
128x64
    Serial.println(F("SSD1306 allocation failed"));
    for(;;);
  }
  delay(500);
  display.clearDisplay();
  display.setTextSize(1);
  display.setTextColor(WHITE);
  display.setCursor(0, 0);
  display.println("Initialization OLED");
  display.display();
  delay(1500);
  display.clearDisplay();
  display.setTextSize(1);
  display.setTextColor(WHITE);
  display.setCursor(0, 0);
  display.println("OLED OKE");
  display.display();
  delay(1500);
  display.clearDisplay();
}
```

2. Nilai Rings parameter berat badan, suhu, kelembapan dan tangis bayi

```
if($berat<2.5){
    $dataAll['status_berat']="0";
    $countData++;
}elseif($berat>4){
    $dataAll['status_berat']="2";
    $countData++;
}else{
    $dataAll['status_berat']="1";
}

if($suhu<33){
    $dataAll['status_suhu']="0";
    $countData++;
}elseif($suhu>35.5){
    $dataAll['status_suhu']="2";
    $countData++;
}else{
    $dataAll['status_suhu']="1";
}

if($kelembapan<40){
    $dataAll['status_kelembapan']="0";
    $countData++;
}elseif($kelembapan>60){
    $dataAll['status_kelembapan']="2";
    $countData++;
}else{
    $dataAll['status_kelembapan']="1";
}

if($tangisan<0){
    $dataAll['status_tangisan']="0";
    $countData++;
}elseif($tangisan>70){
```