

Lampiran 1
Lampiran Foto Saat Observasi



Lampiran 2

Lampiran Script

Script Arduino

```
#include <Arduino.h>
#include <SPI.h>
#include <Wire.h>
#include <WiFi.h>
#include <HTTPClient.h>
#include "Ruine-AMG88xx.h"
#include "Ruine-MFRC522.h"
#include "esp_camera.h"

#include "soc/soc.h"
#include "soc/rtc_cntl_reg.h"

#define SDA_PIN 2
#define SCL_PIN 14
#define MOSI_PIN 13
#define MISO_PIN 12
#define SCK_PIN 14
#define CS_PIN 15

// CAMERA_MODEL_AI_THINKER
#define PWDN_GPIO_NUM 32
#define RESET_GPIO_NUM -1
#define XCLK_GPIO_NUM 0
#define SIOD_GPIO_NUM 26
#define SIOC_GPIO_NUM 27

#define Y9_GPIO_NUM 35
#define Y8_GPIO_NUM 34
#define Y7_GPIO_NUM 39
#define Y6_GPIO_NUM 36
#define Y5_GPIO_NUM 21
#define Y4_GPIO_NUM 19
#define Y3_GPIO_NUM 18
#define Y2_GPIO_NUM 5
#define VSYNC_GPIO_NUM 25
#define HREF_GPIO_NUM 23
#define PCLK_GPIO_NUM 22

float readTemperature();
void sendData(float temperature, long cardID);

TwoWire WIRE_I2C = Wire;
SPIClass SPI_HW = SPIClass(HSPI);

AMG88xx thermalCam;
MFRC522 rfidSensor;

const char *_SSID = "ASUS";
const char *_PASS = "12345678";
const char *_HOST = "192.168.43.159";
const char *_PATH = "/echo-server/upload.php";

WiFiClient client;

float pixels[AMG88xx_PIXEL_ARRAY_SIZE];
```

```

camera_config_t config;

void setup()
{
    // put your setup code here, to run once:
    uint32_t brown_reg_temp = READ_PERI_REG(RTC_CNTL_BROWN_OUT_REG);
    WRITE_PERI_REG(RTC_CNTL_BROWN_OUT_REG, 0);
    delay(100); Serial.begin(115200);
    Serial.println(F("\nECHO Project"));
    Serial.println();
    Serial.print("Connecting to ");
    Serial.println(_SSID);
    WiFi.mode(WIFI_STA);

    delay(200);

    WiFi.begin(_SSID, _PASS);
    while (WiFi.status() != WL_CONNECTED)
    {
        Serial.print(".");
        delay(500);
    }
    Serial.println();
    Serial.print("ESP32-CAM IP Address: ");
    Serial.println(WiFi.localIP());
    // WRITE_PERI_REG(RTC_CNTL_BROWN_OUT_REG, brown_reg_temp);

    delay(50);

    config.ledc_channel = LEDC_CHANNEL_0;
    config.ledc_timer = LEDC_TIMER_0;
    config.pin_d0 = Y2_GPIO_NUM;
    config.pin_d1 = Y3_GPIO_NUM;
    config.pin_d2 = Y4_GPIO_NUM;
    config.pin_d3 = Y5_GPIO_NUM;
    config.pin_d4 = Y6_GPIO_NUM;
    config.pin_d5 = Y7_GPIO_NUM;
    config.pin_d6 = Y8_GPIO_NUM;
    config.pin_d7 = Y9_GPIO_NUM;
    config.pin_xclk = XCLK_GPIO_NUM;
    config.pin_pclk = PCLK_GPIO_NUM;
    config.pin_vsync = VSYNC_GPIO_NUM;
    config.pin_href = HREF_GPIO_NUM;
    config.pin_sscb_sda = SIOD_GPIO_NUM;
    config.pin_sscb_scl = SIOC_GPIO_NUM;
    config.pin_pwdn = PWDN_GPIO_NUM;
    config.pin_reset = RESET_GPIO_NUM;
    config.xclk_freq_hz = 20000000;
    config.pixel_format = PIXFORMAT_JPEG;

    // init with high specs to pre-allocate larger buffers
    if (psramFound())
    {
        config.frame_size = FRAMESIZE_SVGA;
        config.jpeg_quality = 0; //0-63 lower number means higher
quality //10
        config.fb_count = 2;
    }
}

```

```

else
{
    config.frame_size = FRAMESIZE_CIF;
    config.jpeg_quality = 0; //0-63 lower number means higher
quality //12
    config.fb_count = 1;
}

// camera init
esp_err_t err = esp_camera_init(&config);
if (err != ESP_OK)
{
    Serial.printf("Camera init failed with error 0x%x", err);
    delay(1000);
    ESP.restart();
}

delay(50);

WIRE_I2C.begin(SDA_PIN, SCL_PIN);
thermalCam.begin(AMG88xx_ADDRESS, &WIRE_I2C);

delay(50);

SPI_HW.begin(SCK_PIN, MISO_PIN, MOSI_PIN, -1);
rfidSensor.PCD_Init(CS_PIN, &SPI_HW);
Serial.println("Waiting for a card...");
}

void loop()
{
    // put your main code here, to run
repeatedly:amg.readPixels(pixels);
    if (rfidSensor.PICC_IsNewCardPresent())
    {
        if (rfidSensor.PICC_ReadCardSerial())
        {
            long code = 0;
            for (byte i = 0; i < rfidSensor.uid.size; i++)
            {
                code = ((code + rfidSensor.uid.uidByte[i]) * 10);
            }

            SPI_HW.end();
            WIRE_I2C.beginTransaction(0x00);
            delay(5);
            WIRE_I2C.endTransmission();
            delay(5);

            float temperature = readTemperature();

            Serial.print("Card ID      : ");
            Serial.println(code);
            Serial.print("Temperature : ");
            Serial.print(temperature);
            Serial.println(" *C");
            Serial.print("Thermistor  : ");
            Serial.print(thermalCam.readThermistor());
            Serial.println(" *C\n");

            sendData(temperature, code);
            delay(5);
            SPI_HW.begin(SCK_PIN, MISO_PIN, MOSI_PIN, -1);

```

```

        delay(3000);
        Serial.println("Waiting for a card...");
    }
}

delay(500);
}

float readTemperature()
{
    thermalCam.readPixels(pixels);

    float maxTemperature = pixels[0];
    for (int i = 1; i < AMG88xx_PIXEL_ARRAY_SIZE; i++)
    {
        if (pixels[i] > maxTemperature)
        {
            maxTemperature = pixels[i];
        }
    }
    return maxTemperature + 5.0f;
}

void sendData(float temperature, long cardID)
{
    camera_fb_t *framebuffer = NULL;
    framebuffer = esp_camera_fb_get();
    if (!framebuffer)
    {
        Serial.println("Camera capture failed");
        delay(1000);
        ESP.restart();
    }

    Serial.printf("Connecting to server: %s\n", _HOST);
    if (client.connect(_HOST, 80))
    {
        Serial.println("Connection successful!");
        const char *HEADER = "POST %s HTTP/1.1\r\n"
            "Host: %s:%d\r\n"
            "Cache-Control: no-cache\r\n"
            "Accept-Encoding: gzip, deflate\r\n"
            "Connection: close\r\n"
            "Accept: */*\r\n"
            "Content-Length: %d\r\n"
            "Content-Type: multipart/form-data;
boundary=-----RUINEDELIMITER\r\n\r\n";
        const char *BODY1 = "-----RUINEDELIMITER\r\n"
            "Content-Disposition: form-data;
name=\"%s\";\r\n\r\n"
            "%ld\r\n";
        const char *BODY2 = "-----RUINEDELIMITER\r\n"
            "Content-Disposition: form-data;
name=\"%s\";\r\n\r\n"
            "%.2f\r\n";
        const char *BODY3 = "-----RUINEDELIMITER\r\n"
            "Content-Disposition: form-data;
name=\"%s\"; filename=\"%s\"\r\n"
            "Content-Type: %s\r\n\r\n";
        const char *TAIL = "\r\n-----RUINEDELIMITER--\r\n";
    }
}

```

```

size_t contentLength = framebuffer->len;
contentLength += strlen(BODY1) - 5;
contentLength += strlen(BODY2) - 6;
contentLength += strlen(BODY3) - 6;
contentLength += strlen(TAIL);

if (temperature == 0)
{
    contentLength += 4;
}
else if (temperature < 0)
{
    contentLength += floor(log10(abs(temperature))) + 5;
}
else
{
    contentLength += floor(log10(abs(temperature))) + 4;
}

if (cardID == 0)
{
    contentLength += 1;
}
else if (cardID < 0)
{
    contentLength += floor(log10(abs(cardID))) + 2;
}
else
{
    contentLength += floor(log10(abs(cardID))) + 1;
}

contentLength += 42;

client.printf(HEADER, PATH, HOST, 80, contentLength);
client.printf(BODY1, "card_id", cardID);
client.printf(BODY2, "temperature", temperature);
client.printf(BODY3, "photo", "image.jpg", "image/jpeg");

uint8_t *framebufferData = framebuffer->buf;
size_t framebufferLength = framebuffer->len;
for (size_t counter = 0; (counter + 1024) < framebufferLength;
counter += 1024)
{
    client.write(framebufferData, 1024);
    framebufferData += 1024;
}

size_t remainder = framebufferLength % 1024;
if (remainder > 0)
{
    client.write(framebufferData, remainder);
}

client.print(TAIL);

esp_camera_fb_return(framebuffer);

delay(20);

long timeout = millis() + 10000;
while (client.connected() && timeout > millis())
{

```

```

        if (client.available())
        {
            Serial.print((char) client.read());
        }
    }

    client.stop();
}
}

```

Lampiran Script Web

Index

```

<!doctype html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>Suhuku</title>

    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta1/dist/css/bootstrap.min.css" rel="stylesheet" crossorigin="anonymous">

    <style style rel="stylesheet">
img.person-img {
max-height: 500px;
width: auto;
}

table #photo-col {
width: 50%;
}

table #date-col {
width: 30%;
}
</style>

    <link rel="stylesheet" href="assets/pure-min.css">
    <link rel="stylesheet" href="assets/baby-blue.css">

    <script type='text/javascript' src='assets/ui.js'></script>
</head>
<body>
    <table width="100%" height="120px">
        <tr>
            <td>
                <center>
                    <h3>Universitas Muhammadiyah
                    Ponorogo</h3><h3>Fakultas Teknik</h3>
                    <h3>Teknik
                    Informatika</h3></center>
            </td>
        </tr>
    </table>
</body>

```

```

<body>
  <div id="layout">
    <a href="#menu" id="menuLink" class="menu-link">
<span></span> </a>

    <div id="menu">
      <div class="pure-menu pure-menu-open">
        <a href="/" class="pure-menu-heading">Menu</a>

        <ul>
          <li class=" ">
            <a href="index.php">Home</a>
          </li>
          <li class=" ">
            <a href="grafik.php">Grafik Pengunjung</a>
          </li>
          <li class=" ">
            <a href="about.php">About</a>
          </li>
        </ul>
      </div>
    </div><!-- div menu -->
  <!-- Content ----->
<body>
  <div class="container">
    <table class="table">
      <thead>
        <tr>
          <th id="date-col" scope="col">Date</th>
          <th scope="col">Card ID</th>
          <th scope="col">Temperature</th>
          <th scope="col">Status</th>
          <th id="photo-col" scope="col">Photo</th>
        </tr>
      </thead>
      <tbody id="data"></tbody>
    </table>
  </div>

  <script
src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.5.4/dist/umd/popper.min.js"
crossorigin="anonymous"></script>
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta1/dist/js/bootstrap.min.js"
crossorigin="anonymous"></script>
  <script src="https://cdn.jsdelivr.net/npm/axios/dist/axios.min.js"
crossorigin="anonymous"></script>
  <script
src="https://cdn.jsdelivr.net/npm/sweetalert2@10"></script>
  <script type="text/javascript">
(function() {
let tableData = document.querySelector('table tbody#data')

```



```
let previousRows = "  
function updateData() {
```



```

        axios.get('log-api.php')
        .then(function(response) {
        console.table(response.data)

        let rows = "";
        response.data.forEach(function(value, index) {
        rows += `<tr>`
        rows += `<th scope="row">${value['date']}</th>`
        rows += `<td>${value['card']}</td>`
        rows += `<td>${value['temperature']}°C</td>`
        rows += `<td>` + (value['temperature'] > 37.5 ? 'Waspada' : 'Normal') +
        `</td>`
        rows += `<td></td>`

        })

        if (previousRows !== rows) {
        Swal.fire({
        title: 'Sukses!', icon: 'info',
        showConfirmButton: false,
        timer: 3000
        })
        }

        previousRows = rows
        tableData.innerHTML = rows
        }
        })
        }
        updateData()
        setInterval(updateData, 2500)
        })()
        </script>
        </body>

        <!------->
        <div class="footer">
        <div class="legal pure-g">
        <div class="pure-u-1 pure-u-sm-1-
        2">
        <p class="legal-
        copyright">
        Ugik_Teknik Informatika16532584.
        &copy; 2021
        </p>
        </div>
        </div>
        </div>
        </div><!-- div main -->
        </div>
        <!--div layout -->
        </body>
    </html>

```

Grafik

```
<?php
require_once('graph-api.php');
?>

<!doctype html>
<html lang="en">

<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>Suhuku</title>

  <link rel="stylesheet" href="assets/pure-min.css">
  <link rel="stylesheet" href="assets/baby-blue.css">
</head>

<body>
  <div id="layout">
    <a href="#menu" id="menuLink" class="menu-link"> <span></span> </a>

    <div id="menu">
      <div class="pure-menu pure-menu-open">
        <a class="pure-menu-heading" href="">Menu</a>
        <ul>
          <li class="">
            <a href="index.php">Home</a>
          </li>
          <li class="">
            <a href="grafik.php">Grafik Pengunjung</a>
          </li>
          <li class="">
            <a href="about.php">About</a>
          </li>
        </ul>
      </div>
    </div>
    <!--div menu -->
    <!-- Content ----->
    <div id="main">

  </div>
</body>
</html>

<head>
  <script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
  <script type="text/javascript">
    google.charts.load('current', {
      'packages': ['corechart', 'bar']
    });
    google.charts.setOnLoadCallback(drawStuff);

    function drawStuff() {
```

```

var button = document.getElementById('change-chart');
var chartDiv = document.getElementById('chart_div');

var data = google.visualization.arrayToDataTable(JSON.parse('<?=$encoded ?>'));

var materialOptions = {
  width: 900,
  chart: {
    title: 'Grafik Pengunjung',
    subtitle: 'Universitas Muhammadiyah Ponorogo'
  },
  series: {
    0: {
      axis: 'distance'
    }, // Bind series 0 to an axis named 'distance'.
    1: {
      axis: 'brightness'
    } // Bind series 1 to an axis named 'brightness'.
  },
  axes: {
    y: {
      distance: {
        label: 'Pengunjung'
      }, // Left y-axis.
      brightness: {
        side: 'right',
        label: 'Jumlah Pengunjung'
      } // Right y-axis.
    }
  }
};

var classicOptions = {
  width: 900,
  series: {
    0: {
      targetAxisIndex: 0
    },
    1: {
      targetAxisIndex: 1
    }
  },
  title: 'Universitas Muhammadiyah Ponorogo',
  vAxes: {
    // Adds titles to each axis.
    0: {
      title: 'Jumlah Pengunjung'
    },
    1: {
      title: 'Jumlah Pengunjung'
    }
  }
};

function drawMaterialChart() {
  var materialChart = new google.charts.Bar(chartDiv);
  materialChart.draw(data, google.charts.Bar.convertOptions(materialOptions));
  button.innerText = 'Change to Classic';
  button.onclick = drawClassicChart;
}

function drawClassicChart() {

```

```

var classicChart = new google.visualization.ColumnChart(chartDiv);
classicChart.draw(data, classicOptions);
button.innerText = 'Change to Material';
button.onclick = drawMaterialChart;
}

drawMaterialChart();
};
</script>
</head>

<body>
<button id="change-chart">Change to Classic</button>
<br><br>
<div id="chart_div" style="width: 800px; height: 500px;"></div>
</body>

</html>

</div>

<!--FOOOOOOOOOOOOOOOOOOOO OTER-->
<div class="footer">
<div class="legal pure-g">

<div class="pure-u-1 pure-u-sm-1-2">

<p class="legal-copyright">
&copy; 2021 Ugik_Teknik Informatika16532584.
</p>
</div>
</div>

</div>
</div><!-- div main -->
</div>
<!--div layout -->

</body>
</html>

```

About

```

<!doctype html>
<html lang="en">
<head>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Suhuku</title>

<link rel="stylesheet" href="assets/pure-min.css">
<link rel="stylesheet" href="assets/baby-blue.css">

</head>
<body>
<div id="layout">
<a href="#menu" id="menuLink" class="menu-link"> <span></span> </a>

<div id="menu">
<div class="pure-menu pure-menu-open">

```

```

<a class="pure-menu-heading" href="/">Menu</a>

<ul>

    <li class=" ">
        <a href="index.php">Home</a>
    </li>
    <li class=" ">
        <a href="grafik.php">Grafik
Pengunjung</a>
    </li>
    <li class=" ">
        <a href="about.php">About</a>
    </li>
</ul>
</div>
<!-- Content -->
<div id="main">
<div class="header">
<h1>Suhuku</h1>
<h3>Algoritma Second Chance Dalam Mendeteksi Suhu
Tubuh Sebagai Antisipasi Penyebaran COVID-19</h3>
</div>
<div class="content">
<body>
<table width="200%" height="200px">
<tr>
<td width="650px" height="920px"></td>
<td>
</td>
</tr>
</table>
</body>
<body>
<table width="200%" height="200px">
<tr>
<td width="650px" height="920px"></td>
<td>
</td>
</tr>
</table>
</body>
<body>
<table width="200%" height="200px">
<tr>
<td width="650px" height="920px"></td>
<td>
</td>
</tr>
</table>
</body>
</div>
<!--FOOOOOOOOOOOOOOOOOOOOFTER-->
<div class="footer">
<div class="legal pure-g">
<div class="pure-u-1 pure-u-sm-1-2">

```

< p c l a s s = " l e g a l - c o p y r i g h t " >



Informatika16532584.

```

                                </p>
                                </div>
                            </div>
                        </div>
                    </div><!-- div main -->
                </div>
            <!--div layout -->
        </body>
</html>
```



Database.php

```
<?php
$hostname = 'localhost';
$username = 'root';
$password = '';
$databse = 'echo';

try {
    $connection = new PDO("mysql:host=${hostname};dbname=${databse}",
    $username, $password);
    $connection->setAttribute(PDO::ATTR_ERRMODE,
    PDO::ERRMODE_EXCEPTION);
} catch (PDOException $ex) {
    echo 'Error: ' . $ex->getMessage();
}
```

Graph-api

```
<?php

require 'database.php';

if ($connection instanceof PDO) {
    $result = $connection->query("SELECT DAYNAME(`created_at`) AS
`day`, COUNT(IF(`temperature` < 37.5, 1, NULL)) AS `normal`,
COUNT(IF(`temperature` >= 37.5, 1, NULL)) AS `waspada` FROM `logs`
GROUP BY DAYNAME(`created_at`)");
    $remappedResult = [['Hari', 'Suhu Normal', 'Suhu Waspada']];
    $merged = array_merge($remappedResult, $result-
>fetchAll(PDO::FETCH_NUM));
    $encoded = json_encode($merged, JSON_NUMERIC_CHECK);
}
```

Log-API

```
<?php

require 'database.php';

if ($connection instanceof PDO) {
    $result = $connection->query("SELECT *, DATE_FORMAT(created_at, '%d %M %Y, %l:%i %p') AS
date FROM `logs`");
    header('Content-Type: application/json');
    echo json_encode($result->fetchAll(PDO::FETCH_ASSOC));
}
```

Upload

```
<?php
$targetDir = "uploads/";
$logDate = mktime(date('H') + 0,
    date('i'),
    date('s'),
    date('m'),
    date('d'),
    date('y'));
$targetFile = $targetDir . date('s', $logDate) . basename($_FILES["photo"]["name"]);
$imageFileType = strtolower(pathinfo($targetFile, PATHINFO_EXTENSION));
```

```

if (isset($_POST["submit"])) {
    $check = getimagesize($_FILES["photo"]["tmp_name"]);
    if ($check !== false) {
        echo "File is an image - " . $check["mime"] . ".";
    } else {
        echo "File is not an image.";
        exit();
    }
}

if (file_exists($targetFile)) {
    echo "Sorry, file already exists.";
    exit();
}

if ($_FILES["photo"]["size"] > 500000) {
    echo "Sorry, your file is too large.";
    exit();
}

if ($imageFileType != "jpg" && $imageFileType != "jpeg") {
    echo $imageFileType . "<br />";
    echo "Sorry, JPG and JPEG files are allowed.";
    exit();
}

if (move_uploaded_file($_FILES["photo"]["tmp_name"], $targetFile)) {
    echo "The file " . basename($_FILES["photo"]["name"]) . " has been uploaded.";
} else {
    echo $targetFile . "<br />";
    echo "Sorry, there was an error uploading your file.";
    echo "Not uploaded because of error #".$_FILES["photo"]["error"];
    var_dump($_FILES['photo']);
}

require_once('database.php');

$cardID = $_POST['card_id'];
$temperature = $_POST['temperature'];

$stmt = $connection->prepare('INSERT INTO `logs` (`card`, `temperature`, `image`) VALUES (?, ?, ?)');
echo $stmt->execute(array($cardID, $temperature, $targetFile)) ? 'OK' : 'ERROR';

```