

## DAFTAR PUSTAKA

- [1] A. Y. Putra dan Y. Away, “Rancang Bangun Sistem Data Logger Pergerakan Sepeda Motor Berbasis Mikrokontroler ATmega328P,” *Seminar Nasional dan Expo Teknik Elektro 2015*, pp. 72-77, 2015.
- [2] V. A. Suoth, H. I. Mosey dan R. C. Telleng, “Rancang bangun alat pendeteksi intensitas cahaya berbasis Sensor Light Dependent Resistance (LDR,” *JURNAL MIPA UNSRAT ONLINE*, vol. 7, no. 1, pp. 47-51, 2018.
- [3] M. Haryanti, “RANCANG BANGUN SISTEM KEAMANAN RUMAH MENGGUNAKAN RELAY,” *Jurnal Teknologi Elektro, Universitas Mercu Buana*, vol. 8, no. 3, pp. 181-186, 2017.
- [4] s. “Mengenal Arduino Software (IDE),” *Sinau Arduino*, 16 Maret 2016. [Online]. Available: <https://www.sinauarduino.com/artikel/mengenal-arduino-software-ide/>.
- [5] Y. “Internet of Things: Panduan Lengkap,” *Dewaweb*, 3 Agustus 2018. [Online]. Available: <https://www.dewaweb.com/blog/internet-of-things/>.
- [6] administrator, “Pushing Box,” 2019. [Online]. Available: <https://www.pushingbox.com/api.php>.

- [7] T. Dermanto, "Limit switch (saklar pembatas)," *Desain Sistem Kontrol*, 13 April 2014. [Online]. Available: <https://trikueni-desain-sistem.blogspot.com/2014/04/Limit-Switch.html>.
- [8] A. Faudin, "Apa itu Module NodeMCU ESP8266?," *Nyebarilmu.com*, 26 Juli 2017. [Online]. Available: <https://www.nyebarilmu.com/apa-itu-module-nodemcu-esp8266/>.
- [9] A. Faudin, "Cara mengakses sensor LDR menggunakan Arduino," *nyebarilmu.com*, 9 Agustus 2017. [Online]. Available: <https://www.nyebarilmu.com/cara-mengakses-sensor-ldr-menggunakan-arduino/>.
- [10] G. G. Febrianza, "Apa itu Google Sheet & Google App Script?," *medium.com*, 10 Juli 2017. [Online]. Available: <https://medium.com/@gungunfebrianza/apa-itu-google-sheet-google-app-script-be640875f105>.
- [11] S. Komputer, "Resistor : Pengertian, Fungsi, Kode Warna, Macam Macam dan Gambar," *jagad.id*, 10 Juli 2018. [Online]. Available: <https://jagad.id/resistor/>.
- [12] H. Noveka, "Cara Mengoperasikan Google Drive (Spreadsheet)," *ilmukomputer.org*, 6 Juni 2013. [Online]. Available:

<https://ilmukomputer.org/2013/06/06/cara-mengoperasikan-google-drive-spreadsheet/>.

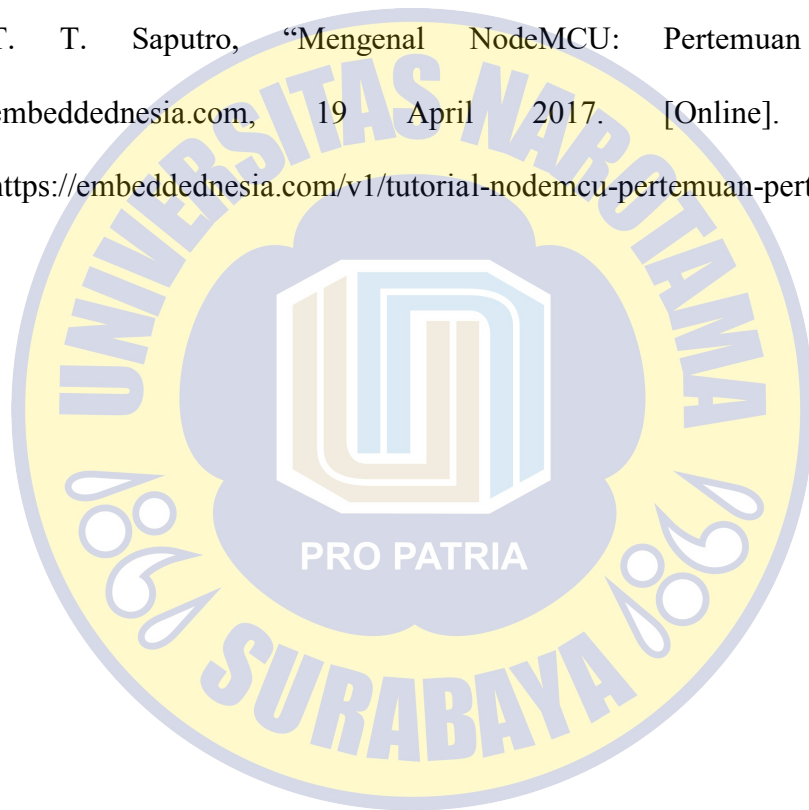
[13] R. Putra, “Sepeda motor | Rezkiputra60,” Rezkiputra60, 15 Juni 2014.

[Online]. Available: <https://rezkiputra60.wordpress.com/2014/06/15/sepeda-motor/>.

[14] T. T. Saputro, “Mengenal NodeMCU: Pertemuan Pertama,”

[embeddednesia.com](http://embeddednesia.com), 19 April 2017. [Online]. Available:

<https://embeddednesia.com/v1/tutorial-nodemcu-pertemuan-pertama/>.



## LAMPIRAN

### Lampiran 1. Kode Program NodeMCU ESP8266

```
#include <ESP8266WiFi.h>
```

```
int Led_OnBoard = D4;
```

```
int led1= D2
```

```
int led3 = D8;
```

```
int button1 = D0;
```

```
int button2 = D1;
```

```
int buttonState1 = 0;
```

```
int buttonState2 = 0;
```

```
int buttonState3 = 0;
```

```
int ldrValue = 0;
```

```
int ldrValue2 = 0;
```

```
int ldrValue3 = 0;
```

```
int ldrValue4 = 0;
```

```
const int ldrPin = D3;
```

```
const int ldrPin2 = D5;

const int ldrPin3 = D6;

const int ldrPin4 = D7;

const char* ssid = "Xperia X Performance";

const char* password = "rahasiaa";

const char *host = "api.pushingbox.com";

void setup() {

    // put your setup code here, to run once:

    delay(200);

    pinMode(Led_OnBoard, OUTPUT);

    pinMode(button1, INPUT);

    pinMode(button2, INPUT);

    pinMode(led3, OUTPUT);

    pinMode(ldrPin, INPUT);

    pinMode(ldrPin2, INPUT);

    pinMode(ldrPin3, INPUT);

    pinMode(ldrPin4, INPUT);
```

```
Serial.begin(115200);

WiFi.mode(WIFI_OFF);

delay(200);

WiFi.mode(WIFI_STA);

WiFi.begin(ssid, password);

Serial.println("");

Serial.print("Connecting");

// Wait for connection
while (WiFi.status() != WL_CONNECTED) {

    digitalWrite(Led_OnBoard, LOW);

    delay(200);

    Serial.print(".");

    digitalWrite(Led_OnBoard, HIGH);

    delay(200);

}

digitalWrite(Led_OnBoard, HIGH);

Serial.println("");
```

```
Serial.println("Sukses Tersambung");

Serial.println("Tersambung ke Jaringan/SSID");

Serial.print("IP address: ");

Serial.println(WiFi.localIP());

}

void loop()
{
digitalWrite(led1, HIGH);
digitalWrite(led3, HIGH);
digitalWrite(Led_OnBoard, LOW);

buttonState1 = digitalRead(button1);

buttonState2 = digitalRead(button2);

String Button_status1 = "";

if (buttonState1 == 0) {

Button_status1 = "NYALA";

//digitalWrite(led2, HIGH);
```

```
}

else{

    Button_status1 = "MATI";

    // digitalWrite(led2, LOW);

}

String Button_status2 = "";

if (buttonState2 == 0) {

    Button_status2 = "NYALA";

    // digitalWrite(led1, HIGH);

}

else{

    Button_status2 = "MATI";

    // digitalWrite(led1, LOW);

}

}

ldrValue=digitalRead(ldrPin);

String LdrStatus = "";

if (ldrValue == 0) {

    LdrStatus = "NYALA";
```



```
}

if (ldrValue == 1){

LdrStatus = "MATI";

}

ldrValue2=digitalRead(ldrPin2);

String LdrStatus2 = "";

if (ldrValue2 == 0) {

LdrStatus2 = "NYALA";

}

if (ldrValue2 == 1){

LdrStatus2 = "MATI";

}

ldrValue3=digitalRead(ldrPin3);

String LdrStatus3 = "";

if (ldrValue3 == 0) {

LdrStatus3 = "NYALA";

}

if (ldrValue3 == 1){
```



```
LdrStatus3 = "MATI";

}

ldrValue4=digitalRead(ldrPin4);

String LdrStatus4 = "";

if (ldrValue4 == 0) {

LdrStatus4 = "NYALA";

}

if (ldrValue4 == 1){

LdrStatus4 = "MATI";

} //Read Analog value of LDR

Serial.println("");

Serial.println("-----");

Serial.print("connecting to ");

Serial.println(host);

WiFiClient client;

const int httpPort = 80;

if (!client.connect(host, httpPort)) {
```

```
Serial.println("connection failed");

return;

}

String url = "/pushingbox?";

url += "devid=";

url += "v6A812AC231739DC";

url += "&REM1="+Button_status1;

url += "&REM2="+Button_status2;

url += "&LAMPU1="+LdrStatus;

url += "&LAMPU2="+LdrStatus2;

url += "&SEIN1="+LdrStatus3;

url += "&SEIN2="+LdrStatus4;

Serial.print("Requesting URL: ");

Serial.println(url);

client.print(String("GET ") + url + " HTTP/1.1\r\n" + "Host: " + host + "\r\n" +
"Connection: close\r\n\r\n");

unsigned long timeout = millis();
```

```

while (client.available() == 0) {

    if (millis() - timeout > 2000) {

        Serial.println(">>> Client Timeout !");

        client.stop();

        return;

    }

}

while(client.available()){

    String line = client.readStringUntil('\r');

    Serial.print(line);

    Serial.print("Data Sent!"); PRO PATRIA

}

Serial.println();

Serial.println("closing connection");

Serial.println("-----");

digitalWrite(Led_OnBoard, HIGH);

delay(200);

}

```

## Lampiran 2. Kode Google App Script

```
function doGet(e) {

    Logger.log( JSON.stringify(e) );

    var result = 'Ok';

    if (e.parameter == 'undefined') {

        result = 'No Parameters';

    }

    else {

        var sheet_id = '1fXMMuivYOOYwLJfo7Bd7L4FcHbIvT9bmUbG6HQd9eps';
        var sheet = SpreadsheetApp.openById(sheet_id).getActiveSheet();
        var newRow = sheet.getLastRow() + 1;

        var rowData = [];

        var Curr_Date = new Date();

        rowData[0] = Curr_Date;

        var Curr_Time = Utilities.formatDate(Curr_Date, "Asia/Jakarta",
'HH:mm:ss');

        rowData[1] = Curr_Time;

        for (var param in e.parameter) {

            Logger.log('In for loop, param=' + param);
```

```
var value = stripQuotes(e.parameter[param]);

Logger.log(param + '!' + e.parameter[param]);

switch (param) {

    case 'REM1':

        rowData[2] = value;

        break;

    case 'REM2':

        rowData[3] = value;

        break;

    case 'LAMPUI1':

        rowData[4] = value;

        break;

    case 'LAMPUI2':

        rowData[5] = value;

        break;

    case 'SEIN1':

        rowData[6] = value;

        break;
```



```
case 'SEIN2':

    rowData[7] = value;

    break;

default:

    result = "unsupported parameter";

}

}

Logger.log(JSON.stringify(rowData));

var newRange = sheet.getRange(newRow, 1, 1, rowData.length);

newRange.setValues([rowData]);

}

return ContentService.createTextOutput(result);

}

function stripQuotes(value) {

    return value.replace(/^[\"']|[\"']$/g, "");

}
```