DAFTAR PUSTAKA

Achmad, Y. F., & Yulfitri, A. (2020). Pengujian Sistem Pendukung Keputusan Menggunakan Black Box Testing Studi Kasus E-Wisudawan di Institut Sains dan Teknologi Al-Kamal. *Jurnal Ilmu Komputer*.

Andani, S. R. (2013). Fuzzy Mamdani Dalam Menentukan Tingkat Keberhasilan Dosen Mengajar. *Seminar Nasional Informatika 2013; AMIK Tunas Bangsa Pematangsiantar*.

Arafat, S. M. (2016). Sistem Pengaman Pintu Rumah Berbasis Internet of Things Dengan ESP8266. *Jurnal Ilmiah Fakultas Teknik "Technologia" Vol 7, No.4*, 262-268.

Arief, M. R. (2011). Pemrograman Web Dinamis Menggunakan PHP dan Mysql. *Yogyakarta: Andi*.

Artono, B., & Putra, R. G. (2017). Pelatihan dan Pembuatan Internet of Things untuk Lampu Hias Ruang Tamu dengan Bahan Paralon Bekas di Kelurahan Kedungbunder, Kecamatan Sutojayan Blitar, Propinsi Jawa Timur. *J-ADIMAS (Jurnal Pengabdian kepada Masyarakat)*, 64-66.

Bayu, W. (2016, November 17). *Tabel PPM dan pH Nutrisi Hidroponik*. Diambil kembali dari Hidroponikpedia: www.hidroponikpedia.com

Dalle, J., Mutalib, A. A., Shaari, N., & Salam, S. N. (2019). *Pengangtar Intraksi Manusia dan Komputer.* Depok: Rajagrafindo Persada.

DFRobot. (2021, Maret 2). *Gravity: Analog TDS Sensor/Meter for Arduino.* Retrieved from DFRobot Drive The Future: https://www.dfrobot.com/product-1662.html?search=Water

DFRobot. (2021, Maret 3). *SKU: Sen0161-V2.* Retrieved from DFRobot Drive The Future: https://wiki.dfrobot.com/Gravity__Analog_pH_Sensor_Meter_Kit_V2_SKU_SEN0161-V2

Dr.Susilawati, M. (2019). *Dasar-Dasar Bertani Hidroponik.* Palembang: Unsri Press.

Electron, C. (2021, Maret 4). *Electron Components.* Retrieved from ESP-12E ESP8266 Wifi Wireless IoT Board Module: https://www.electroncomponents.com/esp-12e-8266-iot-internet-wifi-module

Fatori, M. F., Faizah, F., & Arisandi, D. (2017). Rancang Kontrol dan Monitoring Sistem Proteksi Pompa Air Submersible Berbasis Arduino Nano. *Prosding SNITP (Seminar Nasional Inovasi Teknologi Penerbangan)*, 177-181.

Haryanto, E., Suhartini, T., & R, E. (2001). *Sawi dan Selada.* Jakarta: Penebar Swadaya.

Hendriono, D. (2021, April 25). *Henduino Library.* Retrieved from Mengenal Arduino 2560: https://henduino.github.io/library/board/mengenal-arduino-mega2560/

Heribowo, K., & Budiana, N. (2003). *Hidroponik Sayuran : Untuk hobi dan bisnis.* Jakarta: Penebar Swadaya.

Iman, K. (2022, Juli 27). *LCD Dengan I2C Module untuk Arduino.* Retrieved from Khoirul Iman: https://khoiruliman.wordpress.com/2016/06/07/lcd-dengan-i2c-module-untuk-arduino/

Interstacks. (2021, Maret 5). *Relay board 4 channel.* Retrieved from Interstacks: https://www.interstacks.com/blocks/relay-board-4-channel/

Kusumadewi, S. (2002). *Analisis dan Desain SIstem Fuzzy menggunakan Tool Box Matlab, Edisi Pertama.* Jakarta: Graha Ilmu.

Kusumadewi, S. (2004). Fuzzy Quantification Theory I untuk Analisis Hubungan Antara Penilaian Kinerja Dosen Oleh Mahasiswa, Kehadiran Dosen dan Nilai Kelulusan Mahasiswa. *Media Informatika, Vol. 2. No.1,* 57-65.

Kusumadewi, S., & Hari, P. (2010). *Aplikasi logika fuzzy untuk pendukung keputusan.* Yogyakarta: Graha Ilmu.

Marsudi. (2011). Analisis Pendapatan Beberapa Usahatani Sayuran Daun Di Kabupaten Pidie. *Jurnal Agribusiness and Agricurtural Socio Economic "AGRISEP" Vol. 11 No.2*.

Maslim, M. S., & Dwiandiyanta, B. Y. (2017). Implementasi Metode Logika Fuzzy dalam Pembangunan Sistem Optimalisasi Lampu Lalu Lintas. *UNIVERSITAS ATMA JAYA YOGYAKARTA "UAJY"*.

Masykur, F., Gernowo, R., & Suhartono. (2012). Implementasi Sistem Pakar Diagnosis Penyakit Diabetes Mellitus Menggunakan Metode Fuzzy Logic Berbasis Web. *Jurnal Sistem Informasi Bisnis*, 11-17.

Matsuoka, T., & Suhardiyanto, H. (1992). Thermal and Flowing Aspects of Growing Petty Tomato in Cooled NFT Solution During Summer. *Environment Control in Biology, Vol.30*, 119-125.

Mike. (2022, Juli 27). *Using A 16x2 I2C LCD Display With ESP32.* Retrieved from Open Source Hardware Projects : Electronics-Lab.com: https://www.electronics-lab.com/using-16x2-i2c-lcd-display-esp32/

Mulyatiningsih, E., Santoso, D., & Usman, T. (2013). *Buku Ajar Mata Kuliah Metodologi Penelitian Pendidikan.* Yogyakarta: Universitas Negeri Yogyakarta.

Priyaungga, B. A., Dwi, B. A., Syahroni, M., Nurul, T. S., & Saifudin. (2020). Pengujian Black Box pada Aplikasi Perpustakaan Menggunakan Teknik Equivalance Partitions. *Jurnal Teknologi Sistem Informasi dan Aplikasi*, 3.

Putra, D. W., & Andriani, R. (2019). Unified Modeling Language (UML) dalam Perancangan SIstem Informasi Permohonan Pembayaran Restitusi SPPD. *TEKNOIF*, 32-39.

Riduwan. (2013). *Skala Pengukuran Variabel-Variabel Penelitian.* Bandung: Alfabeta.

Robostore. (2022, April 8). *Display LCD 20X4 Retroilluminato I2C Per ARDUINO E RASPBERRY PI.* Retrieved from Robot Store: https://www.robotstore.it/Display-LCD-20x4-retroilluminato-I2C-per-Arduino-e-Raspberry-Pi

Rohman, A. (2014). *Mengenal Framework "Laravel"*. Tangerang: Ilmuti.

Rukmana, R. (1994). *Bertanam Petsai dan Sawi.* Yogyakarta: Kanisius.

Rusli, M. (2017). *Dasar Perancangan Kendali Logika Fuzzy.* Malang: UB Press.

S, R., & S, S. (1967). *Industrial Radiology.* Moscow: Mir Publishers.

Salman, A. G. (2012). *Pemodelan Dasar Sistem Fuzzy*. Retrieved from Binus University School of Computer Science: https://socs.binus.ac.id/2012/03/02/pemodelan-dasar-sistem-fuzzy/

Satzinger, J. W., Jackson, R. B., & Burd, S. D. (2012). *System Analysis and Design in a Changing World.* Buston: Course Technology.

Sofwan, A. (2011). *Belajar Mysql dengan Phpmyadmin Pendahuluan.* Ilmukomputer.org. Retrieved from Ilmukomputer.

Sotyohadi, Dewa, W. S., & Somawirata, I. K. (2020, Mei). Perancangan Pengatur Kandungan TDS dan PH pada Larutan Nutrisi Hidroponik Menggunakan Metode Fuzzy Logic. *Artificial Inteligence & Applications*, 33.

Sugiyono. (2015). *Metode Penelitian Pendidikan Pendekatan Kuantitatif, Kualitatif, dan R&D.* Bandung: Alfabeta.

Suhardiyanto, H. (2006). *Teknologi Hidroponik untuk Budidaya Tanaman.* Bogor: Institute Pertanian Bogor Press.

Suhartini, Suryadarma, P., & Budiwati. (2017). Utilization of Pesticides Vegetable On Pest Control Plutella Xylostella On Plants Mustard Greens (Brassica juncea L.) Towards Sustainable Agriculture. *Jurusan Pendidikan Biologi; FMIPA; Universitas Negeri Yogyakarta*, 37.

Suliyanti. (2016). Rancang Bangun Alat Pengukur Tingkat Kesuburan Tanah Paska Panen. *Politeknik Negeri Sriwijaya*, 3.

Sunaryono, H., & Rismunandar. (1984). *Kunci Bercocok Tanam Sayur-Sayuran Penting di Indonesia.* Bandung: CV. Sinar Baru.

Supandi, A., & Azwardi. (2015). Penerapan Logika Fuzzy dan Pulse Width Modulation untuk Sistem Kendali Kecepatan Robot Line Follower. *Jurnal Informatika, Sistem Kendali, dan Komputer*, 1-10.

Sutedjo, M. M. (2010). *Pupuk dan Cara Pemupukan.* Jakarta: Rineka Cipta.

Sutiyoso, Y. (2003). *Meramu Pupuk Hidroponik: Tanaman Sayuran, Tanaman Buah, Tanaman Bunga.* Jakarta: Penebar Swadaya.

Sutojo, T., Mulyanto, E., & Vincent, S. (2001). *Kecerdasan Buatan.* Yogyakarta: Andi.

Tripama, B., & Yahya, M. R. (2018). Respon Konsentrasi Nutrisi Hidroponik Terhadap Tiga Jenis Tanaman Sawi. *Agritrop*.

Turang, D. A. (2015). Pengembangan Sistem Relay Pengendalian dan Penghematan Pemakaian Lampu Berbasis Mobile. *Seminar Nasional Infromatika 2015 (SemnasIF 2015)*, 75-85.

Wibowo, S., & Asriyanti, A. (2013). Application of NFT Hydroponic on Cultivation of Pakcoy (Brassica Rapa Chinensis). *Jurnal Penelitian Pertanian Terapan*, 159-167.

Widiyaman, T. (2016). *Pengertian Modul Wifi ESP8266.* Warriornux. Dipetik Juni 29, 2020, dari https://www.warriornux.com/pengertian-modul-wifi-esp8266/

Wilianto, & Kurniawan, A. (2018). Sejarah, cara kerja dan manfat Internet of Things. *J. Matrix, vol. 8, no. 2*, 36-41.

Windu, G., & Grace, G. (2013). *Sukses Membangun Aplikasi Penjualan dengan Java.* Jakarta: Elex Media Komputindo.

Yanto, R. (2018). *Menajemen Basis Data Menggunakan MYSQL* (Edisi 1, Cetakan Pertama ed.). Yogyakarta: Deepublish, Jl Rajawali, G. Elang 6, No.3, Drono, Sardonoharjo, Ngaglik, Sleman.

Yazid, E. (2009). Penerapan Kendali Cerdas pada Sistem Tangki Air Menggunakan Logika Fuzzy. *Jurnal Fisika Himpunan Fisika Indonesia*, 11-23.

Yusarah, A. (2018). Saluran dan Margin Pemasaran Cabai Merah (Kasus Pemasaran Cabai Merah Besar Dari Kecamatan Taraju Kabupaten Tasikmalaya Sampai ke Pasar Induk Kramat Jati dan Pasar Jatinegara). *Sarjana thesis, Universitas Siliwangi*.

LAMPIRAN

Lampiran 1: 25 Kombinasi Aturan Fuzzy

Tabel 6.1 Kombinasi aturan fuzzy

Nomor Aturan	Kondisi		Tindakan			
	pH	TDS	PAB	PPU	PPD	PNA
1	Asam kuat	Sangat Kurang	M	L	M	L
2	Asam kuat	Kurang	M	L	M	S
3	Asam kuat	Normal	M	L	M	M
4	Asam kuat	Banyak	S	L	M	M
5	Asam kuat	Sangat banyak	L	L	M	M
6	Asam	Sangat kurang	M	S	M	L
7	Asam	Kurang	M	S	M	S
8	Asam	Normal	M	S	M	M
9	Asam	Banyak	S	S	M	M
10	Asam	Sangat banyak	L	S	M	M
11	Normal	Sangat kurang	M	M	M	L
12	Normal	kurang	M	M	M	S
13	Normal	Normal	M	M	M	M
14	Normal	Banyak	S	M	M	M
15	Normal	Sangat banyak	L	M	M	M
16	Basa	Sangat kurang	M	M	S	M
17	Basa	Kurang	M	M	S	S
18	Basa	Normal	M	M	S	M
19	Basa	Banyak	S	M	S	M
20	Basa	Sangat banyak	L	M	S	M
21	Basa Kuat	Sangat kurang	M	M	L	L
22	Basa Kuat	Kurang	M	M	L	S

23	Basa kuat	Normal	M	M	L	M
24	Basa kuat	Banyak	S	M	L	M
25	Basa kuat	Sangat banyak	L	M	L	M

Lampiran 2: *Source Code Arduino*

```
#include <EEPROM.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 20, 4);

//Inisialisasi variabel pin sensor PH meter
const int PH_PIN = A0;
float phValue;

//inisialisasi pin Sensor TDS Meter
#include "GravityTDS.h"
#define TDS_PIN A1
GravityTDS gravityTds;
float tdsValue;

// inisialisasi pin sensor Suhu meter
#include <OneWire.h>
#include <DallasTemperature.h>
OneWire pin_DS18B20(5);
DallasTemperature DS18B20(&pin_DS18B20);
// int temperature = 25; // gunakan ini untuk menggunakan library bawaan
sensor
int temperature;

// inisialisasi Modul WiFi
#include <KRwifi.h>
char *ssid = "Dattulu";
char *pass = "bismillah";
char *server = "ta.mubarakabaz.my.id";

// inisisalisasi pin Relay Pompa
#define pompaPHNaik 8
#define pompaPHTurun 9
#define pompaNutrisi 10
#define pompaAirBaku 11

float pab, pna, ppu, ppd;

byte statusPompaPU;
byte statusPompaPD;
byte statusPompaNA;
byte statusPompaAB;

// inisialisasi variabel untuk menampung nilai setpoint yang diterima dari
server.

float phSetA, phSetB, tdsSetA, tdsSetB;
```

```
// library ArduinoJSON
#include <ArduinoJson.h>

// tombol untuk reset
void (*resetFunc)(void) = 0;

//timer
unsigned long waktuMulai = 0;
int timer;

/*
   * Library fuzzy menggunakan proses
   * (MAX-MIN) dan (Minimum Mamdani) untuk inferensi,
   * Komposisi, dan (Centers of Area) untuk defuzzifikasi
   */

#include <Fuzzy.h>
#include <FuzzyComposition.h>
#include <FuzzyIO.h>
#include <FuzzyInput.h>
#include <FuzzyOutput.h>
#include <FuzzyRule.h>
#include <FuzzyRuleAntecedent.h>
#include <FuzzyRuleConsequent.h>
#include <FuzzySet.h>

// inisialisasi nama kelas fwuzzy
Fuzzy *fuzzy_main_obj = new Fuzzy();

// inisialisasi nilai member  fuzzy input pH
FuzzySet *asamkuat = new FuzzySet(0, 0, 0, 3);
FuzzySet *asamlemah = new FuzzySet(0, 3, 3, 5.5);
FuzzySet *phnetral = new FuzzySet(4, 5.5, 6.5, 8.5);
FuzzySet *basalemah = new FuzzySet(6.5, 10, 10, 14);
FuzzySet *basakuat = new FuzzySet(10, 14, 14, 14);

// inisialisasi nilai member  fuzzy input TDS
FuzzySet *sangatkurang = new FuzzySet(0, 0, 500, 750);
FuzzySet *kurang = new FuzzySet(500, 750, 750, 1050);
FuzzySet *tdsnormal = new FuzzySet(750, 1050, 1400, 1600);
FuzzySet *banyak = new FuzzySet(1400, 1600, 1600, 2000);
FuzzySet *sangatbanyak = new FuzzySet(1600, 1800, 2000, 2000);

// inisialisasi nilai member fuzzy output (estimasi waktu) aktif pompa pH naik
(basa)
FuzzySet *ppumati = new FuzzySet(0, 0, 0, 6);
FuzzySet *ppusedang = new FuzzySet(5, 10, 10, 15);
FuzzySet *ppulama = new FuzzySet(14, 20, 20, 20);
```

```
// inisialisasi nilai member fuzzy output (estimasi waktu) Pompa PH Turun
(Asam)
FuzzySet *ppdmati = new FuzzySet(0, 0, 0, 6);
FuzzySet *ppdsedang = new FuzzySet(5, 10, 10, 15);
FuzzySet *ppdlama = new FuzzySet(14, 20, 20, 20);

// inisialisasi nilai member fuzzy output (estimasi waktu) Pompa Air Baku
FuzzySet *pabmati = new FuzzySet(0, 0, 0, 6);
FuzzySet *pabsedang = new FuzzySet(5, 10, 10, 15);
FuzzySet *pablama = new FuzzySet(14, 20, 20, 20);

// inisialisasi nilai member fuzzy output Pompa Nutrisi (estimasi waktu)
FuzzySet *ppnmati = new FuzzySet(0, 0, 0, 6);
FuzzySet *ppnsedang = new FuzzySet(5, 10, 10, 15);
FuzzySet *ppnlama = new FuzzySet(14, 20, 20, 20);

void setup() {
  //...
  // Komunikasi serial untuk debugging
  Serial.begin(115200);

  // inisialisasi pin sensor PH meter
  pinMode(PH_PIN, INPUT);

  // Inisialisasi sensor Suhu
  DS18B20.begin();

  // inisialisasi pin Sensor TDS Meter
  gravityTds.setPin(TDS_PIN);
  gravityTds.setAref(5.0);  // nilai referensi tegangan, default arduino 5V
  gravityTds.begin();       // inisialisasi sensor TDS

  // ...
  pinMode(pompaPHNaik, OUTPUT);      // inisialisasi pin sebagai OUTPUT
  digitalWrite(pompaPHNaik, HIGH);   // set relay off
  pinMode(pompaPHTurun, OUTPUT);     // inisialisasi pin sebagai OUTPUT
  digitalWrite(pompaPHTurun, HIGH);  // set relay off
  pinMode(pompaNutrisi, OUTPUT);     // inisialisasi pin sebagai OUTPUT
  digitalWrite(pompaNutrisi, HIGH);  // set relay off
  pinMode(pompaAirBaku, OUTPUT);     // inisialisasi pin sebagai OUTPUT
  digitalWrite(pompaAirBaku, HIGH);  // set relay off

  // Mengatur tampilan LCD: saat pertama kali hidup
  lcd.begin();
  lcd.setCursor(0, 0);
  lcd.print("Selamat Datang");
  lcd.setCursor(0, 1);
```

```
lcd.print("Di Sistem Cerdas");
lcd.setCursor(0, 2);
lcd.print("Connecting...");

// inisialisasi WiFi (Set SSID & Password WiFi)
setWifi(ssid, pass);

// Setting lcd jika berhasil terhubung ke WiFi
lcd.clear();
lcd.setCursor(0, 1);
lcd.print("Berhasil");
lcd.setCursor(0, 2);
lcd.print("Terhubung!");

delay(1000); // delay 1 detik

lcd.clear();

// Membaca data dari EEPROM
phSetA = EEPROM.read(201);
phSetB = EEPROM.read(202);
tdsSetA = EEPROM.read(203);
tdsSetB = EEPROM.read(204);

// Memulai proses Fuzzifikasi
/*
  * inisialisasi semua inputan menjadi crisp input
  */

// inisialisasi Fuzzy Set variabel pH
FuzzyInput *set_PH = new FuzzyInput(1);
set_PH->addFuzzySet(asamkuat);
set_PH->addFuzzySet(asamlemah);
set_PH->addFuzzySet(phnetral);
set_PH->addFuzzySet(basalemah);
set_PH->addFuzzySet(basakuat);
fuzzy_main_obj->addFuzzyInput(set_PH);

// inisialisasai fuzzy set variabel tds
FuzzyInput *set_TDS = new FuzzyInput(2);
set_TDS->addFuzzySet(sangatkurang);
set_TDS->addFuzzySet(kurang);
set_TDS->addFuzzySet(tdsnormal);
set_TDS->addFuzzySet(banyak);
set_TDS->addFuzzySet(sangatbanyak);
fuzzy_main_obj->addFuzzyInput(set_TDS);

// inisialisasi fuzzy set variabel untuk nilai output 1
```

```
// pompa pH naik
FuzzyOutput *set_PPU = new FuzzyOutput(1);
set_PPU->addFuzzySet(ppulama);
set_PPU->addFuzzySet(ppusedang);
set_PPU->addFuzzySet(ppulama);
fuzzy_main_obj->addFuzzyOutput(set_PPU);

// inisialisasi fuzzy set variabel untuk nilai output 2
// pompa pH turun
FuzzyOutput *set_PPD = new FuzzyOutput(2);
set_PPD->addFuzzySet(ppdlama);
set_PPD->addFuzzySet(ppdsedang);
set_PPD->addFuzzySet(ppdlama);
fuzzy_main_obj->addFuzzyOutput(set_PPD);

// inisialisasi fuzzy set variabel untuk nilai output 3
// pompa Nutrisi
FuzzyOutput *set_PPN = new FuzzyOutput(3);
set_PPN->addFuzzySet(ppnlama);
set_PPN->addFuzzySet(ppnsedang);
set_PPN->addFuzzySet(ppnlama);
fuzzy_main_obj->addFuzzyOutput(set_PPN);

// inisialisasi fuzzy set variabel untuk nilai output 4
// pompa air baku
FuzzyOutput *set_PAB = new FuzzyOutput(4);
set_PAB->addFuzzySet(pablama);
set_PAB->addFuzzySet(pabsedang);
set_PAB->addFuzzySet(pablama);
fuzzy_main_obj->addFuzzyOutput(set_PAB);

// Membuat FuzzyRule ////////////////////// 01
FuzzyRuleAntecedent *AK_SK = new FuzzyRuleAntecedent();
AK_SK->joinWithAND(asamkuat, sangatkurang);

FuzzyRuleConsequent *LMLM = new FuzzyRuleConsequent();
LMLM->addOutput(ppulama);
LMLM->addOutput(ppdmati);
LMLM->addOutput(ppnlama);
LMLM->addOutput(pabmati);

FuzzyRule *fzRule1 = new FuzzyRule(1, AK_SK, LMLM);
fuzzy_main_obj->addFuzzyRule(fzRule1);

// Membuat FuzzyRule ////////////////////// 02
FuzzyRuleAntecedent *AK_K = new FuzzyRuleAntecedent();
AK_K->joinWithAND(asamkuat, kurang);
```

```
FuzzyRuleConsequent *LMCM = new FuzzyRuleConsequent();
LMCM->addOutput(ppulama);
LMCM->addOutput(ppdmati);
LMCM->addOutput(ppnsedang);
LMCM->addOutput(pabmati);

FuzzyRule *fzRule2 = new FuzzyRule(2, AK_K, LMCM);
fuzzy_main_obj->addFuzzyRule(fzRule2);

// Membuat FuzzyRule ////////////////////// 03
FuzzyRuleAntecedent *AK_TN = new FuzzyRuleAntecedent();
AK_TN->joinWithAND(asamkuat, tdsnormal);

FuzzyRuleConsequent *LMMM = new FuzzyRuleConsequent();
LMMM->addOutput(ppulama);
LMMM->addOutput(ppdmati);
LMMM->addOutput(ppnmati);
LMMM->addOutput(pabmati);

FuzzyRule *fzRule3 = new FuzzyRule(3, AK_TN, LMMM);
fuzzy_main_obj->addFuzzyRule(fzRule3);

// Membuat FuzzyRule ////////////////////// 04
FuzzyRuleAntecedent *AK_B = new FuzzyRuleAntecedent();
AK_B->joinWithAND(asamkuat, banyak);

FuzzyRuleConsequent *LMMC = new FuzzyRuleConsequent();
LMMC->addOutput(ppulama);
LMMC->addOutput(ppdmati);
LMMC->addOutput(ppnmati);
LMMC->addOutput(pabsedang);

FuzzyRule *fzRule4 = new FuzzyRule(4, AK_B, LMMC);
fuzzy_main_obj->addFuzzyRule(fzRule4);

// Membuat FuzzyRule ////////////////////// 05
FuzzyRuleAntecedent *AK_SB = new FuzzyRuleAntecedent();
AK_SB->joinWithAND(asamkuat, sangatbanyak);

FuzzyRuleConsequent *LMML = new FuzzyRuleConsequent();
LMML->addOutput(ppulama);
LMML->addOutput(ppdmati);
LMML->addOutput(ppnmati);
LMML->addOutput(pablama);

FuzzyRule *fzRule5 = new FuzzyRule(5, AK_SB, LMML);
fuzzy_main_obj->addFuzzyRule(fzRule5);
```

```
// Membuat FuzzyRule ////////////////////// 06
FuzzyRuleAntecedent *AL_SK = new FuzzyRuleAntecedent();
AL_SK->joinWithAND(asamlemah, sangatkurang);

FuzzyRuleConsequent *CMLM = new FuzzyRuleConsequent();
CMLM->addOutput(ppusedang);
CMLM->addOutput(ppdmati);
CMLM->addOutput(ppnlama);
CMLM->addOutput(pabmati);

FuzzyRule *fzRule6 = new FuzzyRule(6, AL_SK, CMLM);
fuzzy_main_obj->addFuzzyRule(fzRule6);

// Membuat FuzzyRule ////////////////////// 07
FuzzyRuleAntecedent *AL_K = new FuzzyRuleAntecedent();
AL_K->joinWithAND(asamlemah, kurang);

FuzzyRuleConsequent *CMCM = new FuzzyRuleConsequent();
CMCM->addOutput(ppusedang);
CMCM->addOutput(ppdmati);
CMCM->addOutput(ppnsedang);
CMCM->addOutput(pabmati);

FuzzyRule *fzRule7 = new FuzzyRule(7, AL_K, CMCM);
fuzzy_main_obj->addFuzzyRule(fzRule7);

// Membuat FuzzyRule ////////////////////// 08
FuzzyRuleAntecedent *AL_N = new FuzzyRuleAntecedent();
AL_N->joinWithAND(asamlemah, tdsnormal);

FuzzyRuleConsequent *CMMM = new FuzzyRuleConsequent();
CMMM->addOutput(ppusedang);
CMMM->addOutput(ppdmati);
CMMM->addOutput(ppnmati);
CMMM->addOutput(pabmati);

FuzzyRule *fzRule8 = new FuzzyRule(8, AL_N, CMMM);
fuzzy_main_obj->addFuzzyRule(fzRule8);

// Membuat FuzzyRule ////////////////////// 09
FuzzyRuleAntecedent *AL_B = new FuzzyRuleAntecedent();
AL_B->joinWithAND(asamlemah, banyak);

FuzzyRuleConsequent *CMMC = new FuzzyRuleConsequent();
CMMC->addOutput(ppusedang);
CMMC->addOutput(ppdmati);
CMMC->addOutput(ppnmati);
CMMC->addOutput(pabsedang);
```

```
FuzzyRule *fzRule9 = new FuzzyRule(9, AL_B, CMMC);
fuzzy_main_obj->addFuzzyRule(fzRule9);

// Membuat FuzzyRule ////////////////////// 10
FuzzyRuleAntecedent *AL_SB = new FuzzyRuleAntecedent();
AL_SB->joinWithAND(asamlemah, sangatbanyak);

FuzzyRuleConsequent *CMML = new FuzzyRuleConsequent();
CMML->addOutput(ppusedang);
CMML->addOutput(ppdmati);
CMML->addOutput(ppnmati);
CMML->addOutput(pablama);

FuzzyRule *fzRule10 = new FuzzyRule(10, AL_SB, CMML);
fuzzy_main_obj->addFuzzyRule(fzRule10);

// Membuat FuzzyRule ////////////////////// 11
FuzzyRuleAntecedent *N_SK = new FuzzyRuleAntecedent();
N_SK->joinWithAND(phnetral, sangatkurang);

FuzzyRuleConsequent *MMLM = new FuzzyRuleConsequent();
MMLM->addOutput(ppumati);
MMLM->addOutput(ppdmati);
MMLM->addOutput(ppnlama);
MMLM->addOutput(pabmati);

FuzzyRule *fzRule11 = new FuzzyRule(11, N_SK, MMLM);
fuzzy_main_obj->addFuzzyRule(fzRule11);

// Membuat FuzzyRule ////////////////////// 12
FuzzyRuleAntecedent *N_K = new FuzzyRuleAntecedent();
N_K->joinWithAND(phnetral, kurang);

FuzzyRuleConsequent *MMCM = new FuzzyRuleConsequent();
MMCM->addOutput(ppumati);
MMCM->addOutput(ppdmati);
MMCM->addOutput(ppnsedang);
MMCM->addOutput(pabmati);

FuzzyRule *fzRule12 = new FuzzyRule(12, N_K, MMCM);
fuzzy_main_obj->addFuzzyRule(fzRule12);

// Membuat FuzzyRule ////////////////////// 13
FuzzyRuleAntecedent *N_N = new FuzzyRuleAntecedent();
N_N->joinWithAND(phnetral, tdsnormal);

FuzzyRuleConsequent *MMMM = new FuzzyRuleConsequent();
```

```
MMMM->addOutput(ppumati);
MMMM->addOutput(ppdmati);
MMMM->addOutput(ppnmati);
MMMM->addOutput(pabmati);

FuzzyRule *fzRule13 = new FuzzyRule(13, N_N, MMMM);
fuzzy_main_obj->addFuzzyRule(fzRule13);

// Membuat FuzzyRule ////////////////////// 14
FuzzyRuleAntecedent *N_B = new FuzzyRuleAntecedent();
N_B->joinWithAND(phnetral, banyak);

FuzzyRuleConsequent *MMMC = new FuzzyRuleConsequent();
MMMC->addOutput(ppumati);
MMMC->addOutput(ppdmati);
MMMC->addOutput(ppnmati);
MMMC->addOutput(pabsedang);

FuzzyRule *fzRule14 = new FuzzyRule(14, N_B, MMMC);
fuzzy_main_obj->addFuzzyRule(fzRule14);

// Membuat FuzzyRule ////////////////////// 15
FuzzyRuleAntecedent *N_SB = new FuzzyRuleAntecedent();
N_SB->joinWithAND(phnetral, sangatbanyak);

FuzzyRuleConsequent *MMML = new FuzzyRuleConsequent();
MMML->addOutput(ppumati);
MMML->addOutput(ppdmati);
MMML->addOutput(ppnmati);
MMML->addOutput(pablama);

FuzzyRule *fzRule15 = new FuzzyRule(15, N_SB, MMML);
fuzzy_main_obj->addFuzzyRule(fzRule15);

// Membuat FuzzyRule ////////////////////// 16
FuzzyRuleAntecedent *BL_SK = new FuzzyRuleAntecedent();
BL_SK->joinWithAND(basalemah, sangatkurang);

FuzzyRuleConsequent *MCLM = new FuzzyRuleConsequent();
MCLM->addOutput(ppumati);
MCLM->addOutput(ppdsedang);
MCLM->addOutput(ppnlama);
MCLM->addOutput(pabmati);

FuzzyRule *fzRule16 = new FuzzyRule(16, BL_SK, MCLM);
fuzzy_main_obj->addFuzzyRule(fzRule16);

// Membuat FuzzyRule ////////////////////// 17
```

```
FuzzyRuleAntecedent *BL_K = new FuzzyRuleAntecedent();
BL_K->joinWithAND(basalemah, kurang);

FuzzyRuleConsequent *MCCM = new FuzzyRuleConsequent();
MCCM->addOutput(ppumati);
MCCM->addOutput(ppdsedang);
MCCM->addOutput(ppnsedang);
MCCM->addOutput(pabmati);

FuzzyRule *fzRule17 = new FuzzyRule(17, BL_K, MCCM);
fuzzy_main_obj->addFuzzyRule(fzRule17);

// Membuat FuzzyRule ////////////////////// 18
FuzzyRuleAntecedent *BL_N = new FuzzyRuleAntecedent();
BL_N->joinWithAND(basalemah, tdsnormal);

FuzzyRuleConsequent *MCMM = new FuzzyRuleConsequent();
MCMM->addOutput(ppumati);
MCMM->addOutput(ppdsedang);
MCMM->addOutput(ppnmati);
MCMM->addOutput(pabmati);

FuzzyRule *fzRule18 = new FuzzyRule(18, BL_N, MCMM);
fuzzy_main_obj->addFuzzyRule(fzRule18);

// Membuat FuzzyRule ////////////////////// 19
FuzzyRuleAntecedent *BL_B = new FuzzyRuleAntecedent();
BL_B->joinWithAND(basalemah, banyak);

FuzzyRuleConsequent *MCMC = new FuzzyRuleConsequent();
MCMC->addOutput(ppumati);
MCMC->addOutput(ppdsedang);
MCMC->addOutput(ppnmati);
MCMC->addOutput(pabsedang);

FuzzyRule *fzRule19 = new FuzzyRule(19, BL_B, MCMC);
fuzzy_main_obj->addFuzzyRule(fzRule19);

// Membuat FuzzyRule ////////////////////// 20
FuzzyRuleAntecedent *BL_SB = new FuzzyRuleAntecedent();
BL_SB->joinWithAND(basalemah, sangatbanyak);

FuzzyRuleConsequent *MCML = new FuzzyRuleConsequent();
MCML->addOutput(ppumati);
MCML->addOutput(ppdsedang);
MCML->addOutput(ppnmati);
MCML->addOutput(pablama);
```

```
FuzzyRule *fzRule20 = new FuzzyRule(20, BL_SB, MCML);
fuzzy_main_obj->addFuzzyRule(fzRule20);

// Membuat FuzzyRule ////////////////////// 21
FuzzyRuleAntecedent *BK_SK = new FuzzyRuleAntecedent();
BK_SK->joinWithAND(basakuat, sangatkurang);

FuzzyRuleConsequent *MLLM = new FuzzyRuleConsequent();
MLLM->addOutput(ppumati);
MLLM->addOutput(ppdlama);
MLLM->addOutput(ppnlama);
MLLM->addOutput(pabmati);

FuzzyRule *fzRule21 = new FuzzyRule(21, BK_SK, MLLM);
fuzzy_main_obj->addFuzzyRule(fzRule21);

// Membuat FuzzyRule ////////////////////// 22
FuzzyRuleAntecedent *BK_K = new FuzzyRuleAntecedent();
BK_K->joinWithAND(basakuat, kurang);

FuzzyRuleConsequent *MLCM = new FuzzyRuleConsequent();
MLCM->addOutput(ppumati);
MLCM->addOutput(ppdlama);
MLCM->addOutput(ppnsedang);
MLCM->addOutput(pabmati);

FuzzyRule *fzRule22 = new FuzzyRule(22, BK_K, MLCM);
fuzzy_main_obj->addFuzzyRule(fzRule22);

// Membuat FuzzyRule ////////////////////// 23
FuzzyRuleAntecedent *BK_N = new FuzzyRuleAntecedent();
BK_N->joinWithAND(basakuat, tdsnormal);

FuzzyRuleConsequent *MLMM = new FuzzyRuleConsequent();
MLMM->addOutput(ppumati);
MLMM->addOutput(ppdlama);
MLMM->addOutput(ppnmati);
MLMM->addOutput(pabmati);

FuzzyRule *fzRule23 = new FuzzyRule(23, BK_N, MLMM);
fuzzy_main_obj->addFuzzyRule(fzRule23);

// Membuat FuzzyRule ////////////////////// 24
FuzzyRuleAntecedent *BK_B = new FuzzyRuleAntecedent();
BK_B->joinWithAND(basakuat, banyak);

FuzzyRuleConsequent *MLMC = new FuzzyRuleConsequent();
MLMC->addOutput(ppumati);
```

```
  MLMC->addOutput(ppdlama);
  MLMC->addOutput(ppnmati);
  MLMC->addOutput(pabsedang);

  FuzzyRule *fzRule24 = new FuzzyRule(24, BK_B, MLMC);
  fuzzy_main_obj->addFuzzyRule(fzRule24);

  // Membuat FuzzyRule ////////////////////// 25
  FuzzyRuleAntecedent *BK_SB = new FuzzyRuleAntecedent();
  BK_SB->joinWithAND(basakuat, sangatbanyak);

  FuzzyRuleConsequent *MLML = new FuzzyRuleConsequent();
  MLML->addOutput(ppumati);
  MLML->addOutput(ppdlama);
  MLML->addOutput(ppnmati);
  MLML->addOutput(pablama);

  FuzzyRule *fzRule25 = new FuzzyRule(25, BK_SB, MLML);
  fuzzy_main_obj->addFuzzyRule(fzRule25);
}

void loop() {
  // ...
  // Print Debugging untuk mengecek nilai input
  // setiap 5 detik
  if (millis() - waktuMulai > 5000U) { // U = Unsigned Integer
    waktuMulai = millis();

    // Pembacaan sensor Suhu
    DS18B20.requestTemperatures();
    temperature = DS18B20.getTempCByIndex(0);

    // Pembacaan sensor pH
    int ph = analogRead(PH_PIN);
    float voltage = 5 / 1024.0 * ph;
    // kalibrasi sensor ph (menggunakan regresi linear)
    // -5.4 -> nilai minimum pembacaan sensor
    phValue = (-5.400 * voltage) + 18.14;

    // Pembacaan sensor TDS
    gravityTds.setAdcRange(1024);
    gravityTds.setTemperature(temperature);
    gravityTds.update();
    tdsValue = gravityTds.getTdsValue();

    /*
    * ================================
    * Proses Fuzzifikasi
```

```
* Memetakan nilai crisp yang akan di fuzzifikasi
*/
fuzzy_main_obj->setInput(1, phValue);
fuzzy_main_obj->setInput(2, tdsValue);

// memulai proses fuzzifikasi, komposisi, dan inferensi
fuzzy_main_obj->fuzzify();

/*
* ================================
* Cek Nilai Relevansi/Akurasi hasil Fuzzifikasi
* Pada Nilai Sensor PH METER
*/
float relevansi_asamkuat = asamkuat->getPertinence();
float relevansi_asamlemah = asamlemah->getPertinence();
float relevansi_phnetral = phnetral->getPertinence();
float relevansi_basalemah = basalemah->getPertinence();
float relevansi_basakuat = basakuat->getPertinence();

Serial.println(" ");

Serial.print("AK: ");
Serial.print(relevansi_asamkuat);
Serial.print("\t");
Serial.print("AL: ");
Serial.print(relevansi_asamlemah);
Serial.print("\t");
Serial.print("PN: ");
Serial.print(relevansi_phnetral);
Serial.print("\t");
Serial.print("BL: ");
Serial.print(relevansi_basalemah);
Serial.print("\t");
Serial.print("BK: ");
Serial.print(relevansi_basakuat);

Serial.println(" ");

/*
* ================================
* Cek Nilai Relevansi/Akurasi hasil Fuzzifikasi
* Pada Nilai Sensor PH METER
*/

float relevansi_sangatkurang = sangatkurang->getPertinence();
float relevansi_kurang = kurang->getPertinence();
float relevansi_tdsnormal = tdsnormal->getPertinence();
float relevansi_banyak = banyak->getPertinence();
```

```
float relevansi_sangatbanyak = sangatbanyak->getPertinence();

Serial.print("SK: ");
Serial.print(relevansi_sangatkurang);
Serial.print("\t");
Serial.print("K: ");
Serial.print(relevansi_kurang);
Serial.print("\t");
Serial.print("TN: ");
Serial.print(relevansi_tdsnormal);
Serial.print("\t");
Serial.print("B: ");
Serial.print(relevansi_banyak);
Serial.print("\t");
Serial.print("SB: ");
Serial.print(relevansi_sangatbanyak);

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

/*
* ================================
* Proses Defuzzifikasi
*/
// proses defuzzifikasi pada nilai output
float ppu = fuzzy_main_obj->defuzzify(1);
float ppd = fuzzy_main_obj->defuzzify(2);
float pna = fuzzy_main_obj->defuzzify(3);
float pab = fuzzy_main_obj->defuzzify(4);

// bulatkan nilai output yang dihasilkan
int pb_ppu = round(ppu);
int pb_ppd = round(ppd);
int pb_pna = round(pna);
int pb_pab = round(pab);

unsigned long delay_ppu = pb_ppu * 1000;
unsigned long delay_ppd = pb_ppd * 1000;
unsigned long delay_pna = pb_pna * 1000;
unsigned long delay_pab = pb_pab * 1000;

// Estimasi Waktu Pompa Aktif
statusPompaPU = ppu;
statusPompaPD = ppd;
statusPompaNA = pna;
statusPompaAB = pab;

Serial.println("");
```

```
// Logika Pompa
if(phValue < phSetA){
  digitalWrite(pompaPHNaik, LOW);
  digitalWrite(pompaPHNaik, HIGH);
  delay(delay_ppu);
} else if(phValue > phSetB){
  digitalWrite(pompaPHTurun, LOW);
  digitalWrite(pompaPHTurun, HIGH);
  delay(delay_ppd);
} else if(tdsValue < tdsSetA){
  digitalWrite(pompaNutrisi, LOW);
  digitalWrite(pompaNutrisi, HIGH);
  delay(delay_pna);
} else if(tdsValue > tdsSetB){
  digitalWrite(pompaAirBaku, LOW);
  digitalWrite(pompaAirBaku, HIGH);
  delay(delay_pab);
}

Serial.println("");
Serial.print("Suhu: "); Serial.print(temperature);
Serial.println("");

Serial.println("=====+ copy & paste nilai ke matlab (ph | tds) +=====");
Serial.print(phValue);
Serial.print(" ");
Serial.print(tdsValue);
Serial.println(" ");

Serial.println(" ");
Serial.println(" Status Pompa ");
Serial.println(" ");
Serial.print("PH Naik: ");
Serial.print(ppu);
Serial.print("\t");
Serial.print("PH Turun: ");
Serial.print(ppd);
Serial.print("\t");
Serial.print("Nutrisi: ");
Serial.print(pna);
Serial.print("\t");
Serial.print("AirBaku: ");
Serial.print(pab);
Serial.print("\t");
Serial.println(" ");
Serial.println(" ");
```

```
    // tampilkan di LCD
    lcd.setCursor(0, 0);
    lcd.print(" Suhu  :");
    lcd.print(temperature, 0);
    lcd.setCursor(0, 1);
    lcd.print(" PH  :");
    lcd.print(phValue, 1);
    lcd.setCursor(0, 2);
    lcd.print(" TDS :");
    lcd.print(tdsValue, 0);
    lcd.setCursor(0, 3);
    lcd.print("U:");
    lcd.print(statusPompaPU);
    lcd.print(" ");
    lcd.setCursor(6, 3);
    lcd.print("D:");
    lcd.print(statusPompaPD);
    lcd.print(" ");
    lcd.setCursor(11, 3);
    lcd.print("N:");
    lcd.print(statusPompaNA);
    lcd.print(" ");
    lcd.setCursor(16, 3);
    lcd.print("A:");
    lcd.print(statusPompaAB);

    timer++;
    waktuMulai = millis();
  }

  // Kirim Data ke Server Setiap 120 Detik
  if (timer >= 5) {
    String path = String() + "/input.php?suhu=" + temperature + "&ph=" +
phValue + "&tds=" + tdsValue + "&pu=" + statusPompaPU + "&pd=" +
statusPompaPD + "&na=" + statusPompaNA + "&ab=" + statusPompaAB;
    httpGet(server, path, 80);
    Serial.println(server + path);
    Serial.print("Respon: ");
    Serial.println(getData);

    StaticJsonDocument<128> doc;

    DeserializationError error = deserializeJson(doc, getData);

    if (error) {
      Serial.print(F("deserializeJson() failed: "));
      Serial.println(error.f_str());
      return;
```

```
    }

    phSetA   = doc["ph_set_a"];
    phSetB   = doc["ph_set_b"];
    tdsSetA  = doc["tds_set_a"];
    tdsSetB  = doc["tds_set_b"];

    // DeserializationError error = deserializeJson(doc, getData);
    // if (error) {
    //   Serial.print(F("deserializeJson() failed: "));
    //   Serial.println(error.f_str());
    //   return;
    // }

    Serial.println("phSetA : "); Serial.print(phSetA);
    Serial.println("phSetB : "); Serial.print(phSetB);
    Serial.println("tdsSetA : "); Serial.print(tdsSetA);
    Serial.println("tdsSetB : "); Serial.print(tdsSetB);

    //Simpan ke EEPROM
    EEPROM.write(201, phSetA);
    EEPROM.write(202, phSetB);
    EEPROM.write(203, tdsSetA);
    EEPROM.write(204, tdsSetB);

    if (getHttpStatus == "Koneksi Gagal") {
      resetFunc();
    }

    timer = 0;
    statusPompaPU = 0;
    statusPompaPD = 0;
    statusPompaNA = 0;
    statusPompaAB = 0;
  }
}
```

Lampiran 3 *Source Code Fuzzy Matlab*

```
[System]
Name='Fuzzy'
Type='mamdani'
Version=2.0
NumInputs=2
NumOutputs=4
NumRules=25
AndMethod='min'
OrMethod='max'
ImpMethod='min'
AggMethod='max'
DefuzzMethod='centroid'

[Input1]
Name='PH'
Range=[0 14]
NumMFs=5
MF1='asamKuat':'trapmf',[0 0 0 3]
MF2='basaKuat':'trapmf',[9.5 14 14 14]
MF3='normal':'trapmf',[4 5.5 6.5 8.5]
MF4='asam':'trapmf',[1 3 3 5.5]
MF5='basa':'trapmf',[6.5 9.5 9.5 14]

[Input2]
Name='TDS'
Range=[0 2000]
NumMFs=5
MF1='sangatBanyak':'trapmf',[1600 1800 2000 2000]
MF2='sangatKurang':'trapmf',[0 0 500 750]
MF3='normal':'trapmf',[750 1050 1400 1600]
MF4='kurang':'trapmf',[500 750 750 1050]
```

```
MF5='banyak':'trapmf',[1400 1600 1600 2000]

[Output1]
Name='PAB'
Range=[0 20]
NumMFs=3
MF1='MATI':'trapmf',[0 0 0 6]
MF2='LAMA':'trapmf',[14 20 20 20]
MF3='CEPAT':'trapmf',[5 10 10 15]

[Output2]
Name='PPU'
Range=[0 20]
NumMFs=3
MF1='MATI':'trapmf',[0 0 0 6]
MF2='LAMA':'trapmf',[14 20 20 20]
MF3='CEPAT':'trapmf',[5 10 10 15]

[Output3]
Name='PPD'
Range=[0 20]
NumMFs=3
MF1='MATI':'trapmf',[0 0 0 6]
MF2='LAMA':'trapmf',[14 20 20 20]
MF3='CEPAT':'trapmf',[5 10 10 15]

[Output4]
Name='PNA'
Range=[0 20]
NumMFs=3
MF1='MATI':'trapmf',[0 0 0 6]
MF2='LAMA':'trapmf',[14 20 20 20]
```

MF3='CEPAT':'trapmf',[5 10 10 15]

[Rules]

1 2, 1 2 1 2 (1) : 1

1 4, 1 2 1 3 (1) : 1

1 3, 1 2 1 1 (1) : 1

1 5, 3 2 1 1 (1) : 1

1 1, 2 2 1 1 (1) : 1

4 2, 1 3 1 2 (1) : 1

4 4, 1 3 1 3 (1) : 1

4 3, 1 3 1 1 (1) : 1

4 5, 3 3 1 1 (1) : 1

4 1, 2 3 1 1 (1) : 1

3 2, 1 1 1 2 (1) : 1

3 4, 1 1 1 3 (1) : 1

3 3, 1 1 1 1 (1) : 1

3 5, 3 1 1 1 (1) : 1

3 1, 2 1 1 1 (1) : 1

5 2, 1 1 3 2 (1) : 1

5 4, 1 1 3 3 (1) : 1

5 3, 1 1 3 1 (1) : 1

5 5, 3 1 3 1 (1) : 1

5 1, 2 1 3 1 (1) : 1

2 2, 1 1 2 2 (1) : 1

2 4, 1 1 2 3 (1) : 1

2 3, 1 1 2 1 (1) : 1

2 5, 3 1 2 1 (1) : 1

2 1, 2 1 2 1 (1) : 1