

## DAFTAR PUSTAKA

1. Permatasari Putri, Widjiyanto, Nadifta Ulta Amalia, Nurhidayati Indah, Saptaningtyas Haryani, "Pembuatan Cookies Berbahan tepung Menir Organik KWT BUMP PT.Pengayom Tani Sejagad", Vol. 6, No. 4, 2021.
2. BPS, "Luas Panen dan Produksi Padi di Indonesia 2020", 2021..
3. Robert Molenaar, "PANEN DAN PASCAPANEN PADI, JAGUNG DAN KEDELAI", Volume 26 No. 1, Februari 2020.
4. Musyarofah, Pudjo suharso, Titin Kartini, "PREFERENSI PETANI DALAM PENJUALAN GABAH PASCA PANEN DI ESA SUMUR MATI KECAMATAN SUMBERASIH KABUPATEN PROBOLINGGO", Volume 11 Nomor 1, 2020.
5. Anang Sulisty, Fendik Ardiansyah, "PENGARUH KARAKTERISTIK BERAS TERHADAP HARGA BERAS DIKOTA TARAKAN", Jurnal BORNEO SIANTEK, Volume 4, Nomor 1, April 2021.
6. BPS, "Rata-Rata Harga Beras Bulanan di Tingkat Penggilingan Menurut Kualitas (Rupiah/Kg)", 2022.
7. Vita Fitriani, Woro Setiaboma, dan Lasuardi Permana, "KARAKTERISK SIFAT FISIKOKIMIA SERPIHAN SEREAL BERAS MENIR DENGAN PENAMBAH TEPUNG PISANG", Jurnal Riset Teknologi Industri, Vol. 15 No. 2 Des 2021.
8. Fitriani, Vita, Woro Setiaboma, and Lasuardi Permana. "KARAKTERISASI SIFAT FISIKOKIMIA SERPIHAN SEREAL BERAS MENIR DENGAN PENAMBAHAN TEPUNG PISANG THE CHARACTERIZATION OF PHYSICO-CHEMICAL PROPERTIES OF BROKEN RICE CEREAL FLAKES WITH THE ADDITION OF BANANA FLOUR." (2021).
9. Zuhri, Dian Al-Munawar. "MEMPELAJARI KINERJA DAN PERAWATAN MESIN RICE MILLING UNIT DI UPJA SARIDADI KECAMATAN JATEN KABUPATEN KARANGANYAR." (2022).
10. Satrio Adhi Nugroho, I Ketut Dedy Suryawan, I Nyoman Kusuma Wardana, "PENERAPAN MIKROKONTROLER SEBAGAI SISTEM KENDALI PERANGKAT LISTRIK BERBASIS ANDROID", Eksplora Informatika, 2019.
11. Zanoفا, Arief Pratama, et al. "Pintu Gerbang Otomatis Berbasis

- Mikrokontroler Arduino UNO R3." *Jurnal Teknik Dan Sistem Komputer* 1.1 (2020): 22-27.
12. Laghari, Asif Ali, et al. "A review and state of art of Internet of Things (IoT)." *Archives of Computational Methods in Engineering* (2021): 1-19.
  13. Adani, Farhan, and Salma Salsabil. "Internet of Things: Sejarah teknologi dan penerapannya." *Jurnal Online Sekolah Tinggi Teknologi Mandala* 14.2 (2019): 92-99.
  14. Setiawan, Bagus Dimas, and Syafri Aprudi. "PENGEMBANGAN SIGNBOARD GUNA MENINGKATKAN DAYA TARIK PENJUALAN PADA UMKM DI DESA U1 PAGAR SARI KABUPATEN MUSI RAWAS." *Jurnal Masda* 1.2 (2022): 61-69.
  15. Peraturan Menteri Pertanian Republik Indonesia Nomor 07/Permentan/OT.140/2/2020, "Tentang Pedoman Umum Supervisi Dan Pendampingan Pelaksanaan Program Dan Kegiatan Utama Kementerian Pertanian Tahun Anggaran 2020", 2020.
  16. Moh Agus Trilaksono, Agus Tri, ALI AKBAR, and HARIS MAHMUDI. RANCANG BANGUN MESIN PENGAYAK PADI SEMI OTOMATIS KAPASITAS 200 KG/JAM MENGGUNAKAN MOTOR LISTRIK. Diss. Universitas Nusantara PGRI Kediri, 2022.
  17. Kurnia, Muhammad Ridho, and Amani Dharma Tarigan. "Rancang Bangun Sistem Filterisasi Pada Mesin Perontok Padi Otomatis Berbasis Mikrokontroler." *Kumpulan Karya Ilmiah Mahasiswa Fakultas sains dan Tekhnologi* 1.1 (2019)
  18. Muslimin, Abdul Muis, and Titin Lestari. "Perancangan alat timbangan digital berbasis arduino Leonardo menggunakan sensor load cell." *Jurnal Natural* 17.1 (2021): 50-63.
  19. Sulaksono, Bambang, and A. Mastiko. "Perancangan Mesin Pengayak Getar Kapasitas 2 m<sup>3</sup>/jam." *Prosiding Seminar Rekayasa Teknologi (SemResTek)*. 2020.
  20. Wardiningrum, Aprilia Dila, et al. "Rancang Bangun dan Uji Kinerja Mesin Pembersihan dan Pengayakan tipe-Grizzly untuk Beras." *Rona Teknik Pertanian* 14.2 (2021).
  21. Ahdan, Syaiful, Erliyan Redy Susanto, and Nana Rachmana Syambas. "Proposed Design and Modeling of Smart Energy Dashboard System by Implementing IoT (Internet of Things)

- Based on Mobile Devices." *2019 IEEE 13th International Conference on Telecommunication Systems, Services, and Applications (TSSA)*. IEEE, 2019.
22. Ardana, Hartica Putri, et al. "IDENTIFIKASI ETNOSAINS DALAM KEARIFAN LOKAL MALOMANG SEBAGAI SUMBER BELAJAR IPA." *Journal of Chemistry Education and Integration* 2.1: 10-20. 2023.
  23. RAHMI, ELIZA. FAKTOR-FAKTOR YANG MEMPENGARUHI PRODUKSI BERAS DI KILANG PADI RIMO TANI KECAMATAN KAWAY XVI KABUPATEN ACEH BARAT. Diss. UPT PERPUSTAKAAN, 2022.
  24. A. Lestari and O. Candra, "Prototype Sistem Pensortir Barang di Industri Menggunakan Loadcell berbasis Arduino Uno," *JTEV (Jurnal Tek. Elektro dan Vokasional)*, vol. 7, no. 1, p. 27, 2021, doi: 10.24036/jtev.v7i1.111504.
  25. Rosyidi, Muhammad Sa'ad. *Rancang Bangun Alat Pembersih Dan Penyortir Ukuran Telur Asin Berbasis Arduino Mega 2560*. Diss. Institut Teknologi Nasional Malang, 2019.
  26. N. N. Sam, M. Rifaldi, N. R. Wibowo, M. Nur, and P. Bosowa, "Rancang Bangun Modul Praktik Load Cell dengan Kapasitas 20 Kg Berbasis Arduino Nano," *Mechatronics J. Prof. Entrep.*, vol. 2, no. 1, pp. 21–26, 2020.
  27. Samsugi, Selamat, et al. "Penerapan Penjadwalan Pakan Ikan Hias Molly Menggunakan Mikrokontroler Arduino UNO dan Sensor RTC DS3231." *Jurnal Teknologi dan Sistem Tertanam* 3.2 (2022).
  28. NURUL HIDAYATI LUSITA DEWI, NURUL HIDAYATI LUSITA DEWI. Prototype smart home dengan modul nodemcu esp8266 berbasis internet of things (iot). Diss. UNIVERSITAS ISLAM MAJAPAHIT MOJOKERTO, 2019.
  29. Pratama, Muhammad Adam, and Mahendra Widyartono. "Rancang Bangun Prototipe Pemantauan Biaya Tagihan Listrik Berbasis Arduino Mega." *JURNAL TEKNIK ELEKTRO* 9.2 (2020).
  30. Raj, V. Narasimha, et al. "Design and Fabrication of Automated Soap Cutting Machine." *Journal of Pharmaceutical Negative Results* (2022): 27-31.

31. Ramadhan, Moch. TA: Rancang Bangun Pintu Gerbang Rumah Otomatis Berbasis Android Melalui Global Position System (GPS) dan Bluetooth. Diss. Universitas Dinamika, 2019.
32. Kurniawan, Putra. Prototipe Sistem Keamanan Kotak Infak Berbasis Mikrokontroler. Diss. Universitas Islam Riau, 2020.
33. Sejati, Hanafi. PENDETEKSI KERUSAKAN LAMPU LALU LINTAS MENGGUNAKAN SMS BERBASIS ARDUINO UNO R3. Diss. STMIK AKAKOM Yogyakarta, 2019.
34. Rifa'i, Achmad. Komparasi Kinerja Sensor Ultrasonik HC-SR04 dan Ultrasonik HY-SRF05 untuk Sistem Pengukuran Jarak pada Automatic Hand Washing Machine Station. Diss. Institut Teknologi Kalimantan, 2021.
35. Hutapea, Alvin Laili Tamami, Muhammad Hafizd, and Budi Supriyadi. "ALAT PENYIRAMAN TANAMAN OTOMATIS MENGGUNAKAN SENSOR KELEMBABAN TANAH DAN SENSOR ULTRASONIK BERBASIS ARDUINO." *Akrab Juara: Jurnal Ilmu-ilmu Sosial* 8.1 (2023): 219-227.
36. Ferdinansyah, Sefhia Wulandari. "Penyiraman Bibit Akasia Menggunakan Limit Switch." (2021).
37. MANIK, TETY GAMRISTI. "DESAIN SISTEM BUKA TUTUP PINTU HOTEL OTOMATIS MENGGUNAKAN E-KTP BERBASIS ARDUINO UNO." (2022).
38. Saputra, Bustommy, et al. "Rancang bangun jemuran otomatis menggunakan arduino uno dan mikrokontroler." *J. Satya Inform* 6.1 (2021): 1-9.
39. Nindayanti, Riska, and Martin Bernard. "Analisis Minat Belajar Siswa SMP Kelas VII Berbantuan MIT App Inventor pada Materi Aritmatika Sosial." *JPMI (Jurnal Pembelajaran Matematika Inovatif)* 5.3 (2022): 751-758.

## LAMPIRAN A

### Listing Program Arduino

```
#include "HX711.h"
#include <movingAvg.h>
#include <Servo.h>
#include <SoftwareSerial.h>
#define pulSaring 4
#define pulGeser 5
#define dirGeser 6
#define DOUT 8
#define CLK 9
#define DOUT1 10
#define CLK1 11
#define v_sensor A0
#define curent A1
#define pinLimD A2
#define btn A3
#define limDepan A4

String dataSerial;
int starting;
unsigned long back;
int tombol;
float calibration_factor = 21.25;
float gramMenir, gramBeras;
int dataVolt;
int volt;
int dataCurent;
int dataCurentAvg;
int limD;
int limB;
int terima = 0;
int count, countSistem;
float arus;
int datalimitDepan;
int countLimitDepan;
HX711 scale;
HX711 timbang;
```

```

movingAvg avgAcs(10);
Servo bendung;
SoftwareSerial ss(2, 3);
void setup() {
  Serial.begin(9600);
  ss.begin(9600);
  pinMode(pulSaring , OUTPUT);
  pinMode(pulGeser, OUTPUT);
  pinMode(dirGeser, OUTPUT);
  pinMode(v_sensor , INPUT);
  pinMode(curent, INPUT);
  pinMode(pinLimD, INPUT_PULLUP);
  pinMode(limDepan, INPUT_PULLUP);
  pinMode(btn, INPUT_PULLUP);
  bendung.attach(7);
  bendung.write(0);
  avgAcs.begin();
  scale.begin(DOUT, CLK);
  scale.set_scale(110.60);//faktor kalibrasi
  timbang.begin(DOUT1, CLK1);
  timbang.set_scale(105.20);
  scale.tare();
  timbang.tare();
  delay(100);
}

void loop() {
  limD = digitalRead(pinLimD);
  dataLimitDepan = digitalRead(limDepan);
  // limB = digitalRead(pinLimB);
  gramBeras = scale.get_units();// 1000, 4;//
  gramMenir = timbang.get_units() ;// 1000, 4;//
  dataVolt = analogRead(v_sensor);
  volt = map(dataVolt, 0, 904, 0, 24);//
  dataCurent = analogRead(curent);
  // dataCurentAvg = avgAcs.reading(dataCurent);
  arus = (2.5 - (dataCurent * (5.0 / 1023.0))) / 0.185;//
  tombol = digitalRead(btn);
  printing();
}

```

```

if (datalimitDepan == LOW) {
    countLimitDepan++;
    delay(250);
}
if (countLimitDepan > 1) {
    countLimitDepan = 1;
}
if (gramBeras <= 0.0 or gramMenir <= 0.0)
{
    gramBeras = 0;
    gramMenir = 0;
}
if (arus <= 0.0) {
    arus = arus * -1.0;
}
if (tombol == LOW) {
    countSistem++;
    scale.tare();
    timbang.tare();
    delay(250);
}
if (countSistem >> 1) {
    countSistem = 0;
}
if (starting == 1) {
    countSistem = 1;
}
if (countSistem == 1 ) {
    if (gramBeras > 2000) {
        //tutup
        // stepperSaring(false);
        if (limD == LOW) {
            countSistem = 0;
            starting = 0;
        }
        else if (countLimitDepan == 1) {
            stepperGes("mundur");
            bendung.write(0);
        }
    }
}

```

```

else {
    steperGes("mundur");
    bendung.write(0);
}
}
else if (countLimitDepan == 1) {
    steperGes("mundur");
    if (limD == LOW) {
        countLimitDepan = 0;
    }
}
else {
    bendung.write(90);//BUKA
    steperGes("maju");
    //    steperSaring(true);
}
}
else {
    bendung.write(90);
    //r    penggeser(true);
    //    steperSaring(false);
}
}

void steperGes(String x) {
    if (x == "maju") {
        digitalWrite(dirGeser, HIGH);
        if (limD == LOW) {
            Serial.println(" x=" + String(x));
            for (long i = 0; i < 3200; i++) {
                digitalWrite(pulSaring, HIGH);
                digitalWrite(pulGeser, HIGH);
                delayMicroseconds(50);
                digitalWrite(pulSaring, LOW);
                digitalWrite(pulGeser, LOW);
                delayMicroseconds(50);
            }
        }
    }
    else if (limD == HIGH) {

```



```

    for (long i = 0; i < 3200; i++) {
        digitalWrite(pulSaring, HIGH);
        digitalWrite(pulGeser, HIGH);
        delayMicroseconds(50);
        digitalWrite(pulSaring, LOW);
        digitalWrite(pulGeser, LOW);
        delayMicroseconds(50);
    }
}
}
else if (x == "mundur") {
    if (limD == HIGH) {
        digitalWrite(dirGeser, LOW);
        for (long i = 0; i < 3200; i++) {
            digitalWrite(pulSaring, HIGH);
            digitalWrite(pulGeser, HIGH);
            delayMicroseconds(50);
            digitalWrite(pulSaring, LOW);
            digitalWrite(pulGeser, LOW);
            delayMicroseconds(50);
        }
    }
}
// else if (limD == LOW) {
//     digitalWrite(pulGeser, LOW);
// }
if (limD == LOW) {
    digitalWrite(pulGeser, LOW);
}
}
void stepperSaring(bool x) {
    if (x == true) {
        for (long i = 0; i < 3200 * 4; i++) {
            digitalWrite(pulSaring, HIGH);
            delayMicroseconds(50);
            digitalWrite(pulSaring, LOW);
            delayMicroseconds(50);
        }
    }
}
}

```

```

else {
    digitalWrite(pulSaring, LOW);
}
}

void printing() {
    unsigned long timerMillis = millis();
    if ((timerMillis - back) > 500) {
        Serial.println(" ");
        Serial.println("gramBeras=" + String(gramBeras));
        Serial.println(" gramMenir=" + String(gramMenir));
        Serial.println(" adcVolt=" + String(dataVolt));
        Serial.println(" volt=" + String(volt));
        Serial.println(" arus=" + String(arus));
        Serial.println(" tombol=" + String(tombol));
        Serial.println(" count=" + String(countSistem));
        Serial.println(" limD=" + String(limD));
        Serial.println(" countLim Depan=" + String(countLimitDepan));
        Serial.println(" dataLimitDepan=" + String(datalimitDepan));
        Serial.println(" kontrol=" + String(dataSerial));
        Serial.println("_____");
        back = millis();
    }
    if (ss.available() > 0) {
        dataSerial = ss.readStringUntil('\n');
    }
    starting = dataSerial.toInt();
    ss.print("{");
    ss.print(gramBeras);
    ss.print(",");
    ss.print(gramMenir);
    ss.print(",");
    ss.print(volt);
    ss.print(",");
    ss.print(arus);
    ss.print(",");
    ss.print("0");
    ss.println("}");
}

```

## LAMPIRAN B

### Listing Program ESP8266

```
#include <SoftwareSerial.h>
#include <ESP8266WiFi.h>
#include <FirebaseESP8266.h>

#include <addons/TokenHelper.h>
#include <addons/RTDBHelper.h>
#define WIFI_SSID "Ta dhamar"
#define WIFI_PASSWORD "00000000"
#define API_KEY "AIzaSyCBPQ8BaAm4IKyKJQPKDNxhZgnPPTos
1k8"
#define DATABASE_URL "https://momon1702-default
rtdb.firebaseio.com"
#define USER_EMAIL "tadhamar1@gmail.com"
#define USER_PASSWORD "tadhamar1802"
//Define Firebase Data object

FirebaseData fbdo;
FirebaseAuth auth;
FirebaseConfig config;

#define trig 15//d8
#define echo 13//d7
String sendToArd;
String forReset;
int resetState;
int sending;
unsigned long pref;
unsigned long sendDataPrevMillis = 0;
unsigned long count = 0;
String dataSerial;
String berasStr, menirStr, voltStr, arusStr;
String kontrol;
float beras, menir, volt, arus;
int duration, distance, jarak;
SoftwareSerial ss(14, 12);
```

```

void setup()
{
  Serial.begin(115200);
  ss.begin(9600);
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  Serial.print("Connecting to Wi-Fi");
  while (WiFi.status() != WL_CONNECTED)
  {
    Serial.print(">>>>");
    delay(100);
  }
  Serial.println();
  Serial.print("Connected with IP: ");
  Serial.println(WiFi.localIP());
  Serial.println();
  Serial.printf("Firebase          Client          v%s\n\n",
    FIREBASE_CLIENT_VERSION);
  config.api_key = API_KEY;
  auth.user.email = USER_EMAIL;
  auth.user.password = USER_PASSWORD;
  config.database_url = DATABASE_URL;
  config.token_status_callback = tokenStatusCallback; //see
    addons/TokenHelper.h
  Firebase.begin(&config, &auth);
  Firebase.reconnectWiFi(true);
  Firebase.setDoubleDigits(5);

  pinMode(trig, OUTPUT);
  pinMode(echo, INPUT);
  pinMode(2, OUTPUT);
  digitalWrite(2, HIGH);
  delay(500);
  digitalWrite(2, LOW);
  delay(500);
}

void loop()

```

```

{
  bacaSerial();
  digitalWrite(trig, LOW);
  delayMicroseconds(8);
  digitalWrite(trig, HIGH);
  delayMicroseconds(8);
  digitalWrite(trig, LOW);
  delayMicroseconds(8);
  duration = pulseIn(echo, HIGH);
  distance = (duration / 2) / 29.1;
  jarak = map(distance, 27, 0, 0, 100);
  if (jarak <= 20) {
    digitalWrite(2, HIGH);
  }
  else {
    digitalWrite(2, LOW);
  }
}

if (Firebase.ready() && (millis() - sendDataPrevMillis > 10000 ||
  sendDataPrevMillis == 0))
{
  kontrol = Firebase.getString(fbdo, "/seting") ? fbdo.to<const char
    *>() : fbdo.errorReason().c_str();
  Firebase.setFloat(fbdo,    "/beras",    beras)    ?    "ok"    :
    fbdo.errorReason().c_str();
  Firebase.setFloat(fbdo,    "/menir",    menir)    ?    "ok"    :
    fbdo.errorReason().c_str();
  Firebase.setFloat(fbdo,    "/volt",    volt)    ?    "ok"    :
    fbdo.errorReason().c_str();
  Firebase.setInt(fbdo,    "/arus",    arus)    ?    "ok"    :
    fbdo.errorReason().c_str();

  // if (kontrol == "1") {
  //   sending = 1;
  //   delay(500);
  //   Firebase.setString(fbdo, "/seting", "0") ? "ok" :
  //     fbdo.errorReason().c_str();
  // }
}

```

```

        ss.println(kontrol);
        sendDataPrevMillis = millis();
    }
    // if (resetState == 0) {
    //     sending = 0;
    // }

}

void bacaSerial() {
    if (ss.available() > 0) {
        dataSerial = ss.readStringUntil('\n');
        int data1 = dataSerial.indexOf('{');
        int data2 = dataSerial.indexOf(',', data1 + 1);
        int data3 = dataSerial.indexOf(',', data2 + 1);
        int data4 = dataSerial.indexOf(',', data3 + 1);
        int data5 = dataSerial.indexOf(" ", data4 + 1);
        int data6 = dataSerial.indexOf("}", data5 + 1);

        berasStr = dataSerial.substring(data1 + 1, data2);
        menirStr = dataSerial.substring(data2 + 1, data3);
        voltStr = dataSerial.substring(data3 + 1, data4);
        arusStr = dataSerial.substring(data4 + 1, data5);
        forReset = dataSerial.substring(data5 + 1, data6);

    }
    beras = berasStr.toFloat();
    menir = menirStr.toFloat();
    volt = voltStr.toFloat();
    arus = arusStr.toFloat();
    resetState = forReset.toInt();

    if ((millis() - pref) > 500) {
        //     Serial.println("temp=" + String(temp));
        Serial.println("beras=" + String(beras) + "G");
        Serial.println("menir=" + String(menir) + "G");
        Serial.println("volt=" + String(volt) + "V");
        Serial.println("arus=" + String(arus) + "A");
    }
}

```

```
Serial.println("ultra=" + String(distance) + "Cm");
Serial.println("control=" + String(kontrol));
Serial.println("sending=" + String(sending));
Serial.println(forReset);

    Serial.println("++++++++++++++++++++++++++++++++++++++++
++++++++++++++++++++++++++++++++++++++++");
    pref = millis();
}

}
```

## BIODATA PENULIS



Nama : Adhilla Dhamar Kuncoro  
Tempat/Tanggal Lahir : Cilacap, 18 Februari 2003  
Alamat : JL. Lengkong RT 01/15, Mertasinga,  
Cilacap Utara, Cilacap, Jawa Tengah.  
Telepon/Hp : +62 82223059878  
Hobi : Melihat  
Motto : *“Aspire, Yearn, Unveil Newness and The  
Moon Is Beautiful Isn’t It”*

### Riwayat Pendidikan :

- SDN Gunung Simpang 03 Cilacap Tahun 2008 – 2014
- SMP Negeri 4 Cilacap Tahun 2014 – 2017
- SMK Negeri 2 Cilacap Tahun 2017 – 2020
- Politeknik Negeri Cilacap Tahun 2020 – 2023

Penulis telah mengikuti seminar Tugas Akhir pada tanggal 21 Agustus 2023 sebagai salah satu persyaratan untuk memperoleh gelar Ahli Madya (A.Md).