

DAFTAR PUSTAKA

- [1] Amir Hamzah. 2021. Prototipe Sistem Pembayaran Berbasis RFID Menggunakan Arduino Uno Pada *Vending Machine*. Yogyakarta: Institut Sains & Teknologi AKPRIND Yogyakarta.
- [2] Ryan. 2019. Rancang Bangun Dispenser Air Bersih Otomatis Berbasis Web Menggunakan Teknologi RFID. Manado: Universitas Sam Ratulangi Manado.
- [3] Ade Rachmawan. 2021. Rancang Bangun *Vending Machine* Dengan RFID Sebagai Pembayaran Elektronik Berbasis Arduino. Surabaya: Institut Teknologi Adhi Tama Surabaya.
- [4] Aldizar Hilmy. 2021. Perancangan Prototipe *Vending Machine* Berbasis RFID. Padang: Universitas Negeri Padang.
- [5] Wahid Hidayatullah. 2019. Prototipe Mesin Penjual Makanan Ringan Otomatis Menggunakan RFID. Cilacap: Politeknik Negeri Cilacap.
- [6] Junaidi dan Yuliyani Dwi Prabowo. 2018. *Project* Sistem Kendali Elektronik Berbasis Arduino. Bandar Lampung, BL: Aura.
- [7] Desmira. 2022. Aplikasi Sensor LDR Untuk Efisiensi Energi Pada Lampu Penerangan Jalan Umum. Serang: Universitas Sultan Ageng Tirtayasa Serang.
- [8] Fitriyono. 2017. Prototipe Kartu Berobat Pasien Puskesmas Menggunakan RFID, Jurnal *Coding* Sistem Komputer Untan, Universitas Tanjungpura Pontianak.
- [9] Bowo Eko Cahyono. 2019. Karakteristik Sensor LDR dan Aplikasinya Pada Alat Ukur Tingkat Kekeruhan Air Berbasis Arduino UNO. Jember: Universitas Jember.
- [10] Decy Nataliana. 2017. Implementasi *Prototype* Sistem *Smart Home* Dengan Pemanfaatan Kode Akses Berbasis Arduino Mega. Bandung: Institut Teknologi Nasional Bandung.
- [11] Mochamad Fajar Wicaksono. 2017. Implementasi Modul Wifi NodeMCU ESP8266 Untuk *Smart Home*. Jurnal Teknik Komputer – Komputika – Volume 6, No.1 – 2017.
- [12] CD Tulle. 2017. Bab II Dasar Teori NodeMCU. Jurnal STMIK Akakom. Vol. 07 No. 01.
- [13] Antoinete. 2014. Analisa Dan Simulasi Sistem Pengendalian Motor DC. Manado: Politeknik Negeri Manado.
- [14] H. S. Weku, E. V. C. Poekoel, R. F. Robot, and M. Eng. 2015. Rancang Bangun Alat Pemberi Pakan Ikan Otomatis Berbasis Mikrokontroler, J. Tek. Elektro dan Komput., vol. 4, no. 7, pp. 54–64.
- [15] Amin, Fatkhul Nur. 2016. Timbangan Berbasis Arduino dengan *Output* LCD dan Suara. Diss. Universitas Negeri Semarang.

- [16] Arief Pratama Zanofa. 2020. Pintu Gerbang Otomatis Berbasis Mikrokontroler Arduino Uno R3. Bandar Lampung: Universitas Taknokrat Indonesia.
- [17] Sitohang, Ely P., Dringhuzen J. Mamahit, and Novi S. Tulung. "Rancang Bangun Catu Daya DC Menggunakan Mikrokontroler ATmega 8535." *Jurnal Teknik Elektro dan Komputer* 7.2 (2018): 135-142.
- [18] Hidayat dan Ridlo Ferari Mauludi. 2014. Rancang Bangun Perangkat Elektronik Penampil Teks Dalam Kode *Braille* Berbasis Mikrokontroler. Jurnal Teknik Komputer. Unikom Bandung.
- [19] Purnomo. 2015. Perancangan dan Pembuatan Mesin Penjual Makanan Otomatis Menggunakan Relai Cerdas. Proyek Akhir. Universitas Muhammadiyah Surakarta.

LAMPIRAN

LAMPIRAN A

Program keseluruhan ESP8266

```
#include <ESP8266WiFi.h>
#include <ESP8266HTTPClient.h>
#include <WiFiClient.h>

WiFiClient wifiClient;

const char* ssid = "IOT";
const char* password = "sandigma88";

void setup() {
  Serial.begin(9600);
}

void setupWifi() {
  Serial.println("msg:Menghubungkan ke WiFi");
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
  }
  Serial.println("msg:Terhubung");
}

void loop() {

void httpRequest(String url) {
  Serial.print("msg:");
  Serial.println(url);
  Serial.print("msg:");
  Serial.println(url.length());
  if (WiFi.status() == WL_CONNECTED) {
    HTTPClient http;
    http.begin(wifiClient, url);
    int httpCode = http.GET();
```

```

    if (httpCode > 0) {
        String payload = http.getString();

        Serial.print("data:");
        Serial.println(payload);
    }
    http.end();
}
else {
    Serial.println("msg:wifi not connected");
    setupWifi();
    httpRequest(url);
}
}

void serialEvent() {
    String data = Serial.readStringUntil('\n');
    httpRequest(data.substring(data.indexOf('(')
+ 1, data.indexOf(')')));
}

```

Program Keseluruhan Sistem Mekanik

```

#include <ArduinoJson.h>

String adminUid = "6c8aeaae";
StaticJsonDocument<1024> doc;

void setup() {
    Serial.begin(9600);
    setupWifi();
    setupLcd();
    setupRfid();
    setupPosition();
}

void loop() {
    lcdClear();
}

```

```

lcdPrint(0, 0, "A.Pilih Minuman");
lcdPrint(0, 1, "B.Cek Saldo");
lcdPrint(0, 2, "C.Pengaturan");
while (1) {
    char c = keypadGetChar();
    if (c == 'A') {
        chooseDrink();
        break;
    }
    else if (c == 'B') {
        checkBalance();
        break;
    }
    else if (c == 'C') {
        setting();
        break;
    }
}
}
//-----
//                                PROGRAM VOID PEMBELIAN PRODUK
//-----//

void chooseDrink() {
    lcdClear();
    lcdPrint(0, 1, "Memproses");
    String productsString = wifiGetProducts();
    DeserializationError error =
deserializeJson(doc, productsString);
    if (error) {
        Serial.print(F("deserializeJson() failed:
"));
        Serial.println(error.c_str());
        return;
    }
    lcdClear();
    lcdPrint(0, 0, "A=" +
String(doc[0]["price"].as<int>()) + ", stok=" +
String(doc[0]["stock"].as<int>()));
    lcdPrint(0, 1, "B=" +

```

```

String(doc[1]["price"].as<int>()) + ", stok=" +
String(doc[1]["stock"].as<int>());
    lcdPrint(0, 2, "C=" +
String(doc[2]["price"].as<int>()) + ", stok=" +
String(doc[2]["stock"].as<int>());
    lcdPrint(0, 3, "D=" +
String(doc[3]["price"].as<int>()) + ", stok=" +
String(doc[3]["stock"].as<int>()));
    delay(3000);
    lcdClear();
    lcdPrint(0, 0, "Pilih Minuman");
    lcdPrint(0, 3, "*.Menu      #.Selesai");
    char c;
    String result = "";
    while (1) {
        lcdPrint(0, 1, "Jenis Minuman? (A-D)");
        lcdPrint(0, 2, result);
        int idSelected = 0;
        while (1) {
            c = keypadGetChar();
            if (c == 'A' || c == 'B' || c == 'C' || c
== 'D') {
                switch (c) {
                    case 'A':
                        idSelected = 0;
                        break;
                    case 'B':
                        idSelected = 1;
                        break;
                    case 'C':
                        idSelected = 2;
                        break;
                    case 'D':
                        idSelected = 3;
                        break;
                }
                if (doc[idSelected]["stock"].as<int>()
> 0) {

```



```

        result += String(c) + '=';
        break;
    }
}
else if (c == '*' || c == '#') {
    break;
}
}
if (c == '*' || c == '#') {
    break;
}
}
lcdPrint(0, 1, "Jumlah? (1-4)          ");
lcdPrint(0, 2, result);
while (1) {
    c = keypadGetChar();
    if (c == '1' || c == '2' || c == '3' || c
== '4') {
        if (String(c).toInt() <=
doc[idSelected]["stock"].as<int>()) {
            result += String(c) + ",";
            break;
        }
    }
    else if (c == '*') {
        break;
    }
}
if (c == '*' || c == '#') {
    break;
}
}
if (c == '#') {
    String uid = "";
    int balance = 0;
    lcdClear();
    lcdPrint(0, 0, "Cek Saldo");
    lcdPrint(0, 1, "(tempelkan kartu)");
    lcdPrint(0, 3, "*.Menu Utama");
    while (1) {

```

```

uid = rfidGetUid();
if (uid != "") {
    lcdClear();
    lcdPrint(0, 1, "Memproses");
    String balanceString =
wifiCheckBalance(uid);
    if (balanceString[0] == 'R') {
        lcdClear();
        lcdPrint(0, 1, "UID : " + uid);
        lcdPrint(0, 2, "Saldo:" +
balanceString.substring(0,
balanceString.length() - 1));
        delay(1000);
        balance =
balanceString.substring(2).toInt();
        break;
    }
    else {
        lcdClear();
        lcdPrint(0, 1, "Gagal");
        delay(3000);
        return -1;
    }
}
char c = keypadGetChar();
if (c == '*') {
    break;
}
}
int totalPrice = 0;
char lastChar = ',';
for (int i = 0; i < result.length(); i++) {
    char nowChar = result[i];
    if (lastChar == ',') {
        switch (nowChar) {
            case 'A' :
                totalPrice +=
doc[0]["price"].as<int>() * String(result[i +
2]).toInt();

```

```

        break;
        case 'B' :
            totalPrice +=
doc[1]["price"].as<int>() * String(result[i +
2]).toInt();
            break;
        case 'C' :
            totalPrice +=
doc[2]["price"].as<int>() * String(result[i +
2]).toInt();
            break;
        case 'D' :
            totalPrice +=
doc[3]["price"].as<int>() *
String(result[i + 2]).toInt();
            break;
    }
}
}
Serial.println(totalPrice);
Serial.println(balance);
Serial.println(totalPrice >= balance);
if (totalPrice >= balance) {
    lcdClear();
    lcdPrint(0, 1, "Saldo tidak cukup");
    delay(3000);
    return;
}
lcdClear();
lcdPrint(0, 0, result);
lcdPrint(0, 1, "total:" +
String(totalPrice));
lastChar = ',';
for (int i = 0; i < result.length(); i++) {
    char nowChar = result[i];
    if (lastChar == ',') {
        int rackIndex = 0;
        switch (nowChar) {
            case 'A' :

```



```

        lcdPrint(0, 1, "Memproses");
        String balance = wifiCheckBalance(uid);
        if (balance[0] == 'R') {
            lcdClear();
            lcdPrint(0, 1, "UID :" + uid);
            lcdPrint(0, 2, "Saldo:" +
balance.substring(0, balance.length() - 1));
            delay(3000);
            return balance.substring(2).toInt();
        }
        else {
            lcdClear();
            lcdPrint(0, 1, "Gagal");
            delay(3000);
            return -1;
        }
    }
    char c = keypadGetChar();
    if (c == '*') {
        break;
    }
}
}
//-----
PROGRAM VOID RISET POSISI
-----//

void setting() {
    lcdClear();
    lcdPrint(0, 0, "Tempelkan krtu admin");
    String uid = "";
    while (1) {
        uid = rfidGetUid();
        if (uid != "") {
            break;
        }
        char c = keypadGetChar();
        if (c == '*') {
            break;
        }
    }
}

```

```

}
if (uid == adminUid) {
    lcdClear();
    lcdPrint(0, 0, "Pengaturan");
    lcdPrint(0, 1, "A.Reset posisi");
    while (1) {
        char c = keypadGetChar();
        if (c == 'A') {
            settingResetPosition();
            break;
        }
        if (c == '*') {
            break;
        }
    }
    delay(1000);
}
}

void settingResetPosition() {
    // positionUpset(0);
    // positionUpset(1);
    // positionUpset(2);
    // positionUpset(3);
    positionReset(0);
    positionReset(1);
    positionReset(2);
    positionReset(3);
}

//-----
PROGRAM Pembacaan ID Kartu RFID
-----//

#include <SPI.h>
#include <MFRC522.h>
#include "Keyboard.h"

MFRC522 rfid(10, 9);

```

```

void setup() {
  Serial.begin(9600);
  Keyboard.begin();
  SPI.begin();
  rfid.PCD_Init();
}

void loop() {
  if (rfid.PICC_IsNewCardPresent()) {
    if (rfid.PICC_ReadCardSerial()) {
      String result = "";
      for (byte i = 0; i < rfid.uid.size; i++)
      {
        result += String (rfid.uid.uidByte[i] <
0x10 ? "0" : "");
        result += String (rfid.uid.uidByte[i],
HEX);
      }
      Serial.println(result);
      Keyboard.println(result);
      rfid.PICC_HaltA();
      rfid.PCD_StopCryptol();
    }
  }
}

```

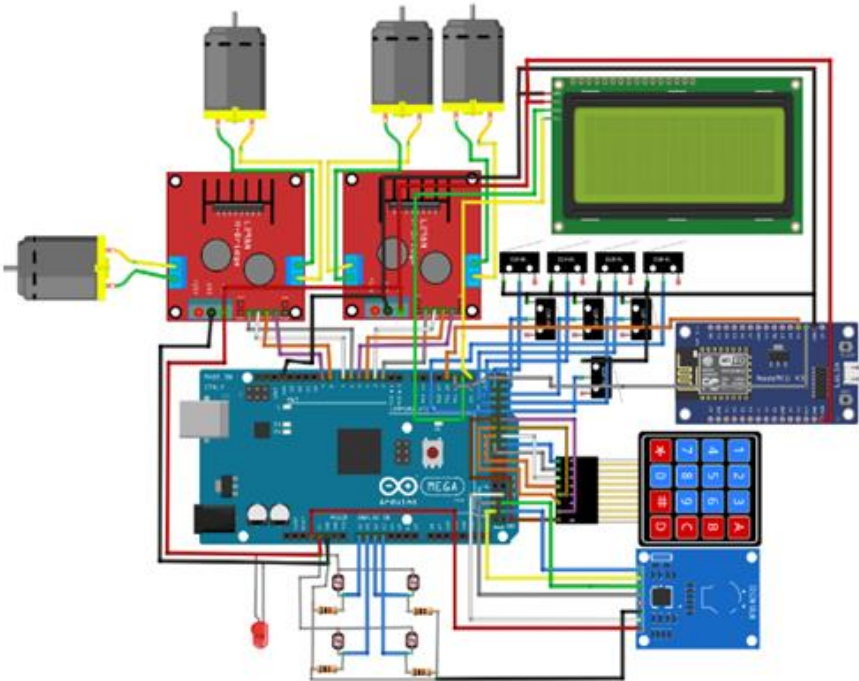
LAMPIRAN B

Gambar Alat



LAMPIRAN C

Rangkaian Wiring



BIODATA PENULIS



Nama : Adi Ilham Pradana
Tempat/Tanggal Lahir : Jepara, 26 Maret 2002
Alamat : Desa Tubanan Rt05
Rw03,Kec.Kembang
Kab.Jepara
Hp : 087824406416
Hobi : Bulutangkis

Riwayat Pendidikan :

| | |
|---------------------------|-------------------|
| SD N 3 Tubanan | Tahun 2007 – 2013 |
| SMP N 1 Bangsri | Tahun 2013 – 2016 |
| SMA N 1 Kembang | Tahun 2016 – 2019 |
| Politeknik Negeri Cilacap | Tahun 2019 – 2022 |