

## DAFTAR PUSTAKA

- [1] Ummul Khair, Alat Pemberi Makan Kucing Otomatis Berbasis Arduino Uno Pada Pet Shop, Teknik Informatika, Fakultas Teknik Dan Komputer, Universitas Harapan Medan, (2021),
- [2] Trisman Zega, Rancang Bangun Timbangan Digital Pada Mesin Perontok Padi Otomatis Berbasis Mikrokontroler, Jurnal, (2019),
- [3] Darussalam, Pemanfaatan *Rfid*, *Loadcell*, Dan Sensor *Infrared* Untuk Miniatur Penukaran Botol Plastik Bekas, Laboratorium Instrumentasi Dan Otomasi, Fakultas Teknik Dan Sains, Universitas Ibn Khaldun Bogor, (2017),
- [4] Ricky Gidion, Purwarupa Mesin Penjual Beras Otomatis Berbasis *Radio Frequency Identification* Dengan Antarmuka *Website*, Jurnal Komputer Dan Aplikasi Volume 07, No. 03 (2019), Hal 132-143,
- [5] Al Habsyi Yesa, Otomatisasi Pelayanan Binatu Berbasis *Raspberry Pi* Untuk Meningkatkan Efektivitas Dan Efisiensi Kegiatan Operasional Dan Pelayanan Binatu, Jurnal Teknik Its Vol. 5, No. 2, (2016),
- [6] Hidayani TU, Miharani T, Rahman A, Hermanto D. Rancang Bangun Timbangan Buah Digital Dengan Keluaran Berat Dan Harga. [Online]. Palembang; 2013
- [7] Amin, Fatkhul Nur. *Timbangan Berbasis Arduino dengan Output LCD dan Suara*. Diss. Universitas Negeri Semarang, 2016, 27 Juli 2022
- [8] Saghoa, Yohanes C., Sherwin RUA Sompie, Maryono and Novi M. Tulung. "Kotak Penyimpanan Uang Berbasis Mikrokontroler Arduino Uno." *Jurnal Teknik Elektro Dan Komputer 7.2* (2018): 167-174, 27 Juli 2022
- [9] 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, 27 Juli 2022
- [10] Nusyirwan, Deny, Michael Dudikof Aritonang, and Prasetya Perwira Putra Perdana. "Penyaringan Air Keruh Menggunakan Sensor LDR Dan Bluetooth HC-05 Sebagai Media Pengontrolan Guna Meningkatkan MUTU Kebersihan Air Di Sekolah." *LOGISTA- Jurnal Ilmiah Pengabdian kepada Masyarakat 3.1* (2019): 37-46, 29 Juli 2022

- [11] Jaelani, Iskandar, Sherwin RUA Sompie, and Dringhuzen J. Mamahit. "Rancang Bangun Rumah Pintar Otomatis Berbasis Sensor Suhu, Sensor Cahaya, Dan Sensor Hujan." *Jurnal Teknik Elektro dan Komputer* 5.1 (2015): 1-10, 29 Juli 2022
- [12] Efrianto, Efrianto, Ridwan Ridwan, and Iman Fahruzi. "Sistem pengaman motor menggunakan smartcard politeknik negeri batam." *Jurnal Integrasi* 8.1 (2016): 1-5, 29 Juli 2022
- [13] H. Hidayat and A. Nugraha, "Perancangan Perangkat Elektronik Media Pembelajaran Iqra dalam Kode Braille," *J. Sist. Komput.*, vol. 5, no. 2, pp. 65-71, 2015.
- [14] Waluyanti, Sri, dkk. 2008. E-Book Teknik Audio Video. Direktorat Pembinaan SMK.
- [15] Yanto, Mingki., & Joewono.A. (2007). Alat Pengetesan Kurva Polarisasi Speaker. Widya Teknik, 163-172.
- [16] Efrianto, Ridwan, Iman Fahruzi, Sistem Pengaman Motor Menggunakan Smartcard Politeknik Negeri Batam. Batam Polytechnics Electrical Engineering Study Program
- [17] MR microbot. Datasheet modul Relay [12]Figa Undala, Dedi Triyanto, Yulrio Brianorman. Prototype Sistem Keamanan Pintu Menggunakan Radio Frequency Identification (RFID) Dengan Kata Sandi Berbasis Mikrokontroler. Jurusan Sistem Komputer, Fakultas MIPA Universitas Tanjungpura
- [18] Faizal Zulmi, Rancang Bangun Alat Pendeteksi Jarak Aman Pada Kendaraan Berbasis Arduino. Program Studi Teknik Elektro - Fakultas Teknik Universitas Mercu Buana, Jakarta,2

## LAMPIRAN A

### PROGRAM TIMBANGAN OTOMATIS

```
#include "SPIFFS.h"
#include "FS.h"
#include "HX711.h"
#include <SPI.h>
#include <MFRC522.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <Keypad_I2C.h>
#include <Keypad.h>
#include <Wire.h>
#include "EEPROM.h"
#include <WiFi.h>

//addreas untuk keypad i2c
#define I2CADDR 0x26
//pin data rfid
#define RST_PIN    25
#define SS_PIN     5

//initial rfid
MFRC522 mfrc522(SS_PIN, RST_PIN);
MFRC522::MIFARE_Key key;
HX711 scale;

//variabel yang digunakan keseluruhan
long dataPrint;
unsigned long sebelum = 0, inter = 3000;
char customKey;
uint8_t dataPin = 15;
uint8_t clockPin = 4;
uint32_t start, stop;
long f, cell;
const byte ROWS = 4; //four rows
const byte COLS = 4; //four columns
//define the cymbols on the buttons of the keypads
char hexaKeys[ROWS][COLS] = {
  {'1', '2', '3', 'a'},
```

```

    {'4', '5', '6', 'b'},
    {'7', '8', '9', 'c'},
    {'*', '0', '#', 'd'}
};

//raw colom keypad
byte rowPins[ROWS] = {0, 1, 2, 3}; //connect to the row pinouts of the keypad
byte colPins[COLS] = {4, 5, 6, 7}; //connect to the column pinouts of the keypad
Keypad_I2C customKeypad( makeKeymap(hexaKeys), rowPins, colPins,
ROWS, COLS, I2CADDR);
LiquidCrystal_I2C lcd(0x27, 20, 4); //lcd 20x4
LiquidCrystal_I2C lcdc(0x20, 16, 2); //lcd16x2
//variabel
unsigned long ERROR;
long tagihan = 10000; // isikan tagihan merchant kelipatan 1000, max
255rb
unsigned long apel = 0, jambu = 0, jeruk = 0, melon = 0;
bool notif = true;
int up, datain;
long saldo;
byte digit;
unsigned long awal = 0, akhir = 5000;
String datastr;
long OLDsaldo;
int OLDdigit;
float settingApel, settingJambu, settingJeruk, settingMelon;
String settApel, settJambu, settJeruk, settMelon;
int forSet, addr = 0;
int pinState;

void setup() {
  Serial.begin(9600);
  Serial2.begin(9600);
  SPIFFS.begin();

  customKeypad.begin( ); //langkah awal komunikasi keypad i2c
  SPI.begin(); //langkah awal komunikasi rfid (SPI communication)
  mfrc522.PCD_Init(); //deklarasi rfid
  pinMode(13, INPUT_PULLUP); //pin tombol tare
  EEPROM.begin(64);

  // proses mencari alamat yang valid pada i2c keypad
  for (byte i = 0; i < 6; i++) {

```

```

    key.keyByte[i] = 0xFF;
}
Serial.println("EMONEY Tanpa Server - Merchant");
Serial.println();
Serial.println("Peringatan : Data akan di simpan pada RFID Card pada sector
#1 (blocks #4)");
Serial.println();

lcd.begin();//komunikasi lcd 20x4
lcdc.begin();//komunikasi lcd 16x2
lcd.backlight();//lapu lcd diaktifkan
lcdc.backlight();

// lcd.print("EMONEY - Merchant");
if (tagihan > 255000) {
    while (1) {
        Serial.println("ERROR - Tagihan merchant lebih dari 255rb");
    }
}
Serial.print("Tagihan Merchant : ");
Serial.println(tagihan);

scale.begin(dataPin, clockPin);// pin hx711
scale.set_scale(-116.00); // TODO you need to calibrate this yourself.
scale.tare();//proses kalibrasi
scale.read_average();//mecari nilai rata rata loadcell

//prtisi untuk penyimpanan data harga
apel = readFile(SPIFFS, "/settingapel.txt").toInt();
jambu = readFile(SPIFFS, "/settingjambu.txt").toInt();
jeruk = readFile(SPIFFS, "/settingjeruk.txt").toInt();
melon = readFile(SPIFFS, "/settingmelon.txt").toInt();

delay(1000);
lcd.clear();
}

void loop() {
    apel = readFile(SPIFFS, "/settingapel.txt").toInt();
    jambu = readFile(SPIFFS, "/settingjambu.txt").toInt();
    jeruk = readFile(SPIFFS, "/settingjeruk.txt").toInt();
}

```

```

melon = readFile(SPIFFS, "/settingmelon.txt").toInt();

customKey = customKeypad.getKey();// read data keypad
pinState = digitalRead(13);//read data tombol
Serial.println("data= " + String(pinState));
S // -----
erial.println("digit= " + String(digit));

load();//fungsi pembacaan loadcell

Serial.print(customKey);

//-----logika untuk pilihan jenis buah-----
if (customKey == 'a')up = 0;
else if (customKey == 'b')up = 1;
else if (customKey == 'c')up = 2;
else if (customKey == 'd')up = 3;
harga_set();//fungsi untuk seting harga buah

//==== output logika dan perhitungan massa menjadi harga dan tampil
lcd=====
if (up == 0) {
  lcd.clear();

  digit = ((f * apel) / 1000) / 1000;
  dataPrint = (f * apel) / 1000;
  Serial.println("ERROR=" + String(ERROR));
  if (dataPrint <= 0) {
    dataPrint = 0;
  }

  if (pinState == LOW) {
    scale.tare();
    lcdc.clear();
    delay(100);
    lcdc.setCursor(0, 0);
    lcdc.print("ON_calibration..");
    delay(3000);
    lcdc.clear();
  }
  lcd.setCursor(0, 0);
  lcd.print(">APEL /Kg=Rp" + String(apel));

```

```

lcd.setCursor(0, 1);
lcd.print(" JAMBU /Kg=Rp" + String(jambu));
lcd.setCursor(0, 2);
lcd.print(" JERUK /Kg=Rp" + String(jeruk));
lcd.setCursor(0, 3);
lcd.print(" MELON /Kg=Rp" + String(melon));
////////////////////////////////////
lcdc.setCursor(0, 0);
lcdc.print("Harga=Rp      ");
lcdc.setCursor(9, 0);
lcdc.print(dataPrint , 1);
lcdc.setCursor(0, 1);
lcdc.print("Gr=      ");
lcdc.setCursor(3, 1);
lcdc.print(f);
}
else if (up == 1) {
  lcd.clear();
  digit = ((f * jambu) / 1000) / 1000;
  dataPrint = (f * jambu) / 1000;
  if (dataPrint <= 0) {
    dataPrint = 0;
  }
  if (pinState == LOW) {
    scale.tare();
    lcdc.clear();
    delay(100);
    lcdc.setCursor(0, 0);
    lcdc.print("ON_calibration..");
    delay(3000);
    lcdc.clear();
  }
  lcd.setCursor(0, 0);
  lcd.print(" APEL /Kg=Rp" + String(apel));
  lcd.setCursor(0, 1);
  lcd.print(">JAMBU /Kg=Rp" + String(jambu));
  lcd.setCursor(0, 2);
  lcd.print(" JERUK /Kg=Rp" + String(jeruk));
  lcd.setCursor(0, 3);
  lcd.print(" MELON /Kg=Rp" + String(melon));

  lcdc.setCursor(0, 0);
  lcdc.print("Harga=Rp      ");
  lcdc.setCursor(9, 0);

```

```

    lcdc.print(dataPrint, 1);
    lcdc.setCursor(0, 1);
    lcdc.print("Gr=      ");
    lcdc.setCursor(3, 1);
    lcdc.print(f);
}
else if (up == 2) {
    lcd.clear();
    digit = ((f * jeruk) / 1000) / 1000;
    dataPrint = (f * jeruk) / 1000;
    if (dataPrint <= 0) {
        dataPrint = 0;
    }
    if (pinState == LOW) {
        scale.tare();
        lcdc.clear();
        delay(100);
        lcdc.setCursor(0, 0);
        lcdc.print("ON_calibration..");
        delay(3000);
        lcdc.clear();
    }
    lcd.setCursor(0, 0);
    lcd.setCursor(0, 0);
    lcd.print(" APEL /Kg=Rp" + String(apel));
    lcd.setCursor(0, 1);
    lcd.print(" JAMBU /Kg=Rp" + String(jambu));
    lcd.setCursor(0, 2);
    lcd.print(">JERUK /Kg=Rp" + String(jeruk));
    lcd.setCursor(0, 3);
    lcd.print(" MELON /Kg=Rp" + String(melon));
    ///////////////////////////////////////////////////////////////////
    lcdc.setCursor(0, 0);
    lcdc.print("Harga=Rp      ");
    lcdc.setCursor(9, 0);
    lcdc.print(dataPrint, 1);
    lcdc.setCursor(0, 1);
    lcdc.print("Gr=      ");
    lcdc.setCursor(3, 1);
    lcdc.print(f);
}
else if (up == 3) {
    lcd.clear();
    digit = ((f * melon) / 1000) / 1000;

```



```

dataPrint = (f * melon) / 1000;
if (dataPrint <= 0) {
    dataPrint = 0;
}
if (pinState == LOW) {
    scale.tare();
    lcdc.clear();
    delay(100);
    lcdc.setCursor(0, 0);
    lcdc.print("ON_calibration..");
    delay(3000);
    lcdc.clear();
}
lcd.setCursor(0, 0);
lcd.print(" APEL /Kg=Rp" + String(apel));
lcd.setCursor(0, 1);
lcd.print(" JAMBU /Kg=Rp" + String(jambu));
lcd.setCursor(0, 2);
lcd.print(" JERUK /Kg=Rp" + String(jeruk));
lcd.setCursor(0, 3);
lcd.print(">MELON /Kg=Rp" + String(melon));

////////////////////////////////////
lcdc.setCursor(0, 0);
lcdc.print("Harga=Rp      ");
lcdc.setCursor(9, 0);
lcdc.print(dataPrint, 1);
lcdc.setCursor(0, 1);
lcdc.print("Gr=          ");
lcdc.setCursor(3, 1);
lcdc.print(f);
}
//=====batas logika
nya=====

if (notif) {
    notif = false;
    Serial.println();
    Serial.println("_____");
    Serial.println("Silahkan tap kartu");
    Serial.print("Setiap TAP, saldo berkurang sebesar ");
    Serial.println(digit * 1000);
}

```

```

//=====perubahan kartu yang dibaca =====
if ( ! mfrc522.PICC_IsNewCardPresent() ) {
    return;
}
if ( ! mfrc522.PICC_ReadCardSerial() ) {
    return;
}

//=====
=====
Serial.println();
Serial.print("Card UID:");

    dump_byte_array(mfrc522.uid.uidByte, mfrc522.uid.size);// menyimpan data
kartu ke esp32

Serial.println();
Serial.print("Tipe Kartu : ");
MFRC522::PICC_Type piccType =
mfrc522.PICC_GetType(mfrc522.uid.sak);
Serial.println(mfrc522.PICC_GetTypeName(piccType));
// Cek kesesuaian kartu

if ( piccType != MFRC522::PICC_TYPE_MIFARE_MINI
    && piccType != MFRC522::PICC_TYPE_MIFARE_1K
    && piccType != MFRC522::PICC_TYPE_MIFARE_4K ) {
    Serial.println(F("NGGO MIFARE Classic cards 1KB - 13.56MHz."));//
    notif = true;
    delay(2000);
    resetReader();//fungsi reset kartu menjadi nilai saldo
    return;
}
// that is: sector #1, covering block #4 up to and including block #7
byte sector      = 1;
byte blockAddr   = 4;

MFRC522::StatusCode status;//status kartu ne apakah dia 1k/4k/13.56
byte buffer[18];
byte size = sizeof(buffer);

//Serial.println("Current data in sector:");

//membaca data kartu yg ada di sektor 1 blok4 kalkulasi nilai saldo

```

```

mfr522.PICC_DumpMifareClassicSectorToSerial(&(mfr522.uid), &key,
sector);
Serial.println();

// Baca Saldo yang ada dari RFID Card
status = (MFRC522::StatusCode) mfr522.MIFARE_Read(blockAddr,
buffer, &size);
if (status != MFRC522::STATUS_OK) {
    Serial.println("Gagal Baca Kartu RFID");
    //Serial.println(mfr522.GetStatusCodeName(status));
    resetReader();
    return;
}
OLDdigit = buffer[0]; //data saldo disimpan sementara pada bufer 0
OLDSaldo = OLDDigit; //saldo sing ws diwaca-saldo harga bayar
OLDSaldo *= 1000; // nilai data rfid *1000
//tampilkan nilai saldo kartu di lcd
lcd.clear();
Serial.print("Saldo Kartu Sebelumnya : ");
Serial.println(OLDSaldo);
Serial.println();
lcd.setCursor(0, 2);
lcd.print("Saldo Kartu");
lcd.setCursor(0, 3);
lcd.print(OLDSaldo);
delay(1000);
lcd.setCursor(0, 2);
lcd.print(" ");
lcd.setCursor(0, 3);
lcd.print(" ");
//-----
// Kurangi Saldo sebesar tagihan bayar
if (OLDDigit < digit) {
    lcd.clear();
    Serial.println("Saldo belum di kurangi, saldo tidak cukup, silahkan isi saldo
terlebih dahulu");
    Serial.println("GAGAL bayar merchant");
    lcd.setCursor(0, 2);
    lcd.print("GAGAL Bayar");
    lcd.setCursor(0, 3);
    lcd.print("Saldo Kurang");
    delay(1000);
    lcd.setCursor(0, 2);
    lcd.print(" ");
}

```

```

lcd.setCursor(0, 3);
lcd.print("          ");

resetReader();//reset rfid ben ra diwaca terus
return;
}

OLDdigit -= digit ; //rumus pengurangan saldo kartu - bayar e

byte dataBlock[] = {
  //0, 1, 2, 3, 4, 5, 6, 7, 8, 9,10,11,12,13,14,15
  OLDdigit, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
};

//proses pengiriman data sisa saldo dari esp ke kartu
status = (MFRC522::StatusCode) mfr522.MIFARE_Write(blockAddr,
dataBlock, 16);
if (status != MFRC522::STATUS_OK) {
  Serial.println("GAGAL Write Saldo pada Kartu RFID");
  //Serial.println(mfr522.GetStatusCodeName(status));
}
Serial.println();

status = (MFRC522::StatusCode) mfr522.MIFARE_Read(blockAddr,
buffer, &size);
if (status != MFRC522::STATUS_OK) {
  Serial.println("Gagal Baca Kartu RFID");
  //Serial.println(mfr522.GetStatusCodeName(status));
}

Serial.println();

Serial.println("Mengurangi Saldo...");
if (buffer[0] == dataBlock[0]) {
  lcd.clear();
  up = 0;

  saldo = buffer[0];
  saldo *= 1000;
  //Serial.print("data digit ke 0 : ");
  //Serial.println(buffer[0]);
  Serial.print("=====>>>>>> Saldo kartu sekarang :
");

```

```

Serial.println(saldo);
Serial.println("_ Berhasil bayar Merchant _");
lcd.setCursor(0, 2);
lcd.print("BERHASIL Bayar");
lcd.setCursor(0, 3);
lcd.print("Sisa Saldo ");
lcd.print(saldo);
delay(1000);
lcd.setCursor(0, 2);
lcd.print("      ");
lcd.setCursor(0, 3);
lcd.print("      ");
}
////////////////////////////////////
else {
  Serial.println("----- GAGAL bayar Merchant -----");
}

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

resetReader();
}

//limit data kartu max 255 k
void dump_byte_array(byte *buffer, byte bufferSize) {
  for (byte i = 0; i < bufferSize; i++) {
    Serial.print(buffer[i] < 0x10 ? " 0" : " ");
    Serial.print(buffer[i], HEX);
  }
}

void resetReader() { //fungsi reset kartu
  // Halt PICC
  mfrc522.PICC_HaltA();
  // Stop encryption on PCD
  mfrc522.PCD_StopCrypto1();

  notif = true;
}

void load() { //pembacaan loadcell
  f = scale.get_units();
  if (f <= 0) {

```

```

    f = 0;
}
cell = f * 1000;

Serial.print ("kg=");
Serial.println(f);
}
void harga_set() { //jika * push variabel forset bertambah 1
    if (customKey == '*') {
        up = 4;
        lcd.clear();
        delay(300);
        forSet++;
    }

    switch (forSet) {
        case 1:
            if (customKey == '0' || customKey == '1' || customKey == '2' || customKey
            == '3' || customKey == '4' || customKey == '5' || customKey == '6' || customKey
            == '7' || customKey == '8' || customKey == '9') {
                settApel = settApel + customKey;
            }
            else if (pinState == LOW) {
                settApel = " ";
            }
            //ketik data harga di keypad

            //jika # ditekan maka data yang sudah di ketik akan di save pada partisi
spifs
            if (customKey == '#') {
                settingApel = settApel.toInt();
                forSet = 0;
                apel = settingApel;
                writeFile(SPIFFS, "/settingapel.txt", settApel.c_str()); //format
penyimpanan data pada spifs
                lcd.clear();
                up = 1;
            }
            lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("  SET HARGA APEL  ");
            lcd.setCursor(0, 1);
            lcd.print("Rp " + String(settApel));

```

```

lcd.print("          ");
lcd.setCursor(0, 2);
lcd.print("          ");
lcd.setCursor(0, 3);
lcd.print("          ");
break;

```

case 2:

```

if (customKey == '0' || customKey == '1' || customKey == '2' || customKey
== '3' || customKey == '4' || customKey == '5' || customKey == '6' || customKey
== '7' || customKey == '8' || customKey == '9') {
    settJambu = settJambu + customKey;
}
else if (pinState == LOW) {
    settJambu = " ";
}

```

```

if (customKey == '#') {
    settingJambu = settJambu.toInt();
    forSet = 0;
    jambu = settingJambu;
    writeFile(SPIFFS, "/settingjambu.txt", settJambu.c_str());
    lcd.clear();
    up = 2;
}
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("  SET HARGA JAMBU  ");
lcd.setCursor(0, 1);
lcd.print("Rp " + String(settJambu));
lcd.print("          ");
lcd.setCursor(0, 2);
lcd.print("          ");
lcd.setCursor(0, 3);
lcd.print("          ");
break;

```

case 3:

```

if (customKey == '0' || customKey == '1' || customKey == '2' || customKey
== '3' || customKey == '4' || customKey == '5' || customKey == '6' || customKey
== '7' || customKey == '8' || customKey == '9') {
    settJeruk = settJeruk + customKey;
}

```

```

else if (pinState == LOW) {
    settJeruk = " ";
}
if (customKey == '#') {
    settingJeruk = settJeruk.toInt();
    forSet = 0;
    jeruk = settingJeruk;
    writeFile(SPIFFS, "/settingjeruk.txt", settJeruk.c_str());
    lcd.clear();
    up = 3;
}
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("  SET HARGA JERUK  ");
lcd.setCursor(0, 1);
lcd.print("Rp " + String(settJeruk));
lcd.print("          ");
lcd.setCursor(0, 2);
lcd.print("          ");
lcd.setCursor(0, 3);
lcd.print("          ");
break;
case 4:

    if (customKey == '0' || customKey == '1' || customKey == '2' || customKey
== '3' || customKey == '4' || customKey == '5' || customKey == '6' || customKey
== '7' || customKey == '8' || customKey == '9') {
        settMelon = settMelon + customKey;
    }
    else if (pinState == LOW) {
        settMelon = " ";
    }
    if (customKey == '#') {
        settingMelon = settMelon.toInt();
        forSet = 0;
        melon = settingMelon;
        writeFile(SPIFFS, "/settingmelon.txt", settMelon.c_str());
        lcd.clear();
        up = 4;
    }
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("  SET HARGA MELON  ");
    lcd.setCursor(0, 1);

```



```

    lcd.print("Rp " + String(settMelon));
    lcd.print("      ");
    lcd.setCursor(0, 2);
    lcd.print("      ");
    lcd.setCursor(0, 3);
    lcd.print("      ");
    break;
case 5:
    forSet = 0;

    break;
}
}

String readFile(fs::FS &fs, const char * path) {
    File file = fs.open(path, "r");
    if (!file || file.isDirectory())
    {
        return String();
    }
    String fileContent;
    while (file.available())
    {
        fileContent += String((char)file.read());
    }
    return fileContent;
}

void writeFile(fs::FS &fs, const char * path, const char * message)
{
    File file = fs.open(path, "w");
    if (!file)
    {
        return;
    }
    file.print(message);
}

```

## LAMPIRAN B

### PROGRAM ALAT TOP UP

```
#include <MFRC522.h>
#include <WiFi.h>
#include <FirebaseESP32.h>

#include <addons/TokenHelper.h>
#include <addons/RTDBHelper.h>
#define WIFI_SSID "Lidar"
#define WIFI_PASSWORD "langkapaket"
#define API_KEY "AlZaSyBLGbdN3ks6NHzqCLXS8iENhXgJg6_fHcY"
#define DATABASE_URL "ngising-geni-default-rtdb.firebaseio.com"
#define USER_EMAIL "stopkontaksmart@gmail.com"
#define USER_PASSWORD "smartstopkontak2022"
#define RST_PIN    4    // Configurable, see typical pin layout above
#define SS_PIN     5    // Configurable, see typical pin layout above

FirebaseData fbdo;
FirebaseAuth auth;
FirebaseConfig config;

MFRC522 mfrc522(SS_PIN, RST_PIN); // Create MFRC522 instance.
MFRC522::MIFARE_Key key;

bool notif = true;
bool isiSaldo = false;
String input;
long saldo, proveus;
int digit;
String tambahsaldo;
long OLDSaldo;
int OLDDigit;

void setup() {
  Serial.begin(115200); // Initialize serial communications with the PC
  firebaseSetup();
  SPI.begin(); // Init SPI bus
  mfrc522.PCD_Init(); // Init MFRC522 card
```

```

for (byte i = 0; i < 6; i++) {
    key.keyByte[i] = 0xFF;
}

Serial.println("EMONEY Tanpa Server Cek dan Tambah Saldo");
Serial.println();
Serial.println("Peringatan : Data akan di simpan pada RFID Card pada sector
#1 (blocks #4)");
}

void loop() {
    if ((millis() - proveus) > 500) {
        topup();
        proveus = millis();
    }
    else {
        tambahsaldo = Firebase.getString(fbdo, F("/isi saldo")) ? fbdo.to<const char
*>() : fbdo.errorReason().c_str();
        Serial.println("tambahsaldo: " + tambahsaldo);
        saldo = tambahsaldo.toInt();
        if (saldo > 255) {
            saldo = 0;
            Serial.println("Saldo tidak boleh lebih dari 255");
        }
        if (saldo < 0) {
            saldo = 0;
            Serial.println("Saldo tidak boleh kurang dari 0");
        }
        if (saldo > 0) {
            isiSaldo = true;
        }
        else {
            isiSaldo = false;
        }
        digit = saldo;
        saldo *= 1000;
        Serial.print("saldo yang di input : ");
        Serial.println(saldo);
        Serial.println("Silahkan tap kartu untuk menambah saldo kartu");
    }
}

void firebasesetup() {
    WiFi.begin(WIFI_SSID, WIFI_PASSWORD);

```

```

Serial.print("Connecting to Wi-Fi");
while (WiFi.status() != WL_CONNECTED)
{
  Serial.print(".");
  delay(300);
}
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);
}

void topup() {
  if (notif) {
    notif = false;
    Serial.println();

Serial.println("_____");
Serial.println("_____");
    Serial.println("Silahkan input jumlah saldo dan tap kartu");
    Serial.println("tiap decimal dikali 1.000");
    Serial.println("contoh input 9 => saldo = 9ribu || max saldo 255rb");
    Serial.println();
    Serial.println("CEK SALDO LANGSUNG TAP");
  }
  if ( ! mfrc522.PICC_IsNewCardPresent()) {
    return;
  }

  if ( ! mfrc522.PICC_ReadCardSerial()) {
    return;
  }

  Serial.println();
  Serial.print("Card UID:");

```

```

dump_byte_array(mfrc522.uid.uidByte, mfrc522.uid.size);

Serial.println();
Serial.print("Tipe Kartu : ");
MFRC522::PICC_Type piccType =
mfrc522.PICC_GetType(mfrc522.uid.sak);
Serial.println(mfrc522.PICC_GetTypeName(piccType));

// Cek kesesuaian kartu
if ( piccType != MFRC522::PICC_TYPE_MIFARE_MINI
    && piccType != MFRC522::PICC_TYPE_MIFARE_1K
    && piccType != MFRC522::PICC_TYPE_MIFARE_4K) {
    Serial.println(F("Kode ini hanya dapat digunakan pada MIFARE Classic
cards 1KB - 13.56MHz."));
    notif = true;
    delay(2000);
    resetReader();
    return;
}

// that is: sector #1, covering block #4 up to and including block #7
byte sector      = 1;
byte blockAddr   = 4;

MFRC522::StatusCode status;
byte buffer[18];
byte size = sizeof(buffer);

// Show the whole sector as it currently is
//Serial.println("Current data in sector:");
mfrc522.PICC_DumpMifareClassicSectorToSerial(&(mfrc522.uid), &key,
sector);
Serial.println();

if (isiSaldo) {
    // Baca Saldo yang ada dari RFID Card
    status = (MFRC522::StatusCode) mfrc522.MIFARE_Read(blockAddr,
buffer, &size);
    if (status != MFRC522::STATUS_OK) {
        Serial.println("Gagal Baca Kartu RFID");
        //Serial.println(mfrc522.GetStatusCodeName(status));
        resetReader();
        return;
    }
}

```

```

OLDdigit = buffer[0];
OLDsaldo = OLDdigit;
OLDsaldo *= 1000;
Serial.print("Saldo Kartu Sebelumnya : ");
Serial.println(OLDsaldo);
Serial.println();

// Tambah saldo dan Write Saldo pada RFID Card
saldo += OLDsaldo;
digit += OLDdigit;

if (digit > 255) {
    saldo = 0;
    digit = 0;
    Serial.println("Saldo sebelum di tambah saldo baru melebihi 255 ribu");
    Serial.println("GAGAL tambah saldo");
    resetReader();
    return;
}

byte dataBlock[] = {
    //0, 1, 2, 3, 4, 5, 6, 7, 8, 9,10,11,12,13,14,15
    digit, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
};
status = (MFRC522::StatusCode) mfrfc522.MIFARE_Write(blockAddr,
dataBlock, 16);
if (status != MFRC522::STATUS_OK) {
    Serial.println("GAGAL Write Saldo pada Kartu RFID");
    //Serial.println(mfrfc522.GetStatusCodeName(status));
}
Serial.println();

status = (MFRC522::StatusCode) mfrfc522.MIFARE_Read(blockAddr,
buffer, &size);
if (status != MFRC522::STATUS_OK) {
    Serial.println("Gagal Baca Kartu RFID");
    //Serial.println(mfrfc522.GetStatusCodeName(status));
}

Serial.println();

Serial.println("Menambahkan Saldo...");
if (buffer[0] == dataBlock[0]) {
    Firebase.setString(fbdo, F("/isi saldo"), String(0));
}

```

```

//Serial.print("data digit ke 0 : ");
//Serial.println(buffer[0]);
Serial.print("Saldo kartu sekarang : ");
Serial.println(saldo);
Serial.println("_____ Berhasil isi saldo pada kartu _____");
} else {
  Serial.println("----- GAGAL ISI SALDO -----");
}
} else {
  status = (MFRC522::StatusCode) mfr522.MIFARE_Read(blockAddr,
buffer, &size);
if (status != MFRC522::STATUS_OK) {
  Serial.println("Gagal Baca Kartu RFID");
  //Serial.println(mfr522.GetStatusCodeName(status));
  saldo = 0;
  digit = 0;
  resetReader();
  return;
}

Serial.println();

Serial.println("Cek Saldo Kartu");
//Serial.print("data digit ke 0 : ");
//Serial.println(buffer[0]);
saldo = buffer[0];
saldo *= 1000;
Serial.print("=====>>>> Saldo kartu sekarang : ");
Serial.println(saldo);
}

saldo = 0;
digit = 0;

Serial.println();

// Dump the sector data
//Serial.println("Current data in sector:");
//mfr522.PICC_DumpMifareClassicSectorToSerial(&(mfr522.uid), &key,
sector);
Serial.println();

resetReader();
}

```

```
void dump_byte_array(byte *buffer, byte bufferSize) {  
  for (byte i = 0; i < bufferSize; i++) {  
    Serial.print(buffer[i] < 0x10 ? " 0" : " ");  
    Serial.print(buffer[i], HEX);  
  }  
}
```

```
void resetReader() {  
  // Halt PICC  
  mfrc522.PICC_HaltA();  
  // Stop encryption on PCD  
  mfrc522.PCD_StopCrypto1();  
  
  notif = true;  
  isiSaldo = false;
```

```
}
```



## LAMPIRAN C

**Gambar Alat**  
**Prototype Timbangan Otomatis Menggunakan RFID dan ESP 32**



**Gambar Alat Top Up Saldo**



## BIODATA PENULIS



Nama : Yanuar Rizky Setiaji  
Tempat/Tanggal Lahir : Cilacap / 22 Januari 2002  
Alamat : Jl. Macan 28 RT.03 RW.02,  
Kec. Cilacap Utara,  
Kab. Cilacap, Jawa  
Tengah  
Hp : (+62) 85641540853  
Hobi : Memancing  
Motto : Jangan Takut Salah Ketika Itu  
Benar

### **Riwayat Pendidikan :**

SD N 07 Mertasinga	Tahun 2007 – 2013
SMP N 07 Cilacap	Tahun 2013 – 2016
SMK N 02 Cilacap	Tahun 2016 – 2019
Politeknik Negeri Cilacap	Tahun 2019 – 2022

Penulis telah mengikuti seminar Tugas Akhir pada tanggal 11 Agustus 2022 , sebagai salah satu persyaratan untuk memperoleh gelar Ahli Madya (A. Md).