

LAMPIRAN

Lampiran 1 Surat Kesediaan Membimbing TA pembimbing 1

SURAT KESEDIAAN MEMBIMBING TA

Yang bertanda tangan dibawah ini

Nama : Ida Afriliana, ST, M.Kom.
NIDN : 0624047703
NIPY : 12.013.168
Jabatan Struktural : Ka. Prodi
Jabatan Fungsional : Lektor

Dengan ini menyatakan bersedia untuk menjadi pembimbing I pada Tugas Akhir mahasiswa berikut:

Nama : Khaerunisa Anabila
NIM : 22040115
Program Studi : Diploma III Teknik Komputer

Judul TA : PENGEMBANGAN RANCANG BANGUN SMART BOX
PENERIMA PAKET BERBASIS *IOT* DAN *BARCODE*

Demikian pernyataan ini dibuat agar dapat dilaksanakan sebagaimana mestinya.

Mengetahui

Ka. Prodi DIII Teknik Komputer,



Ida Afriliana, ST, M.Kom.
NIPY. 12.013.168

Tegal, Mei 2025

Dosen Pembimbing I,



Ida Afriliana, ST, M.Kom.
NIPY. 12.013.168

Lampiran 2 Surat Kesiediaan Membimbing TA Pembimbing 2

SURAT KESEDIAAN MEMBIMBING TA

Yang bertanda tangan dibawah ini:

Nama	: Abdul Basit, S.Kom.,M.T.
NIDN	: 0608129106
NIPY	: 01.015.098
Jabatan Struktural	: Sekretaris Prodi
Jabatan Fungsional	: Lektor

Dengan ini menyatakan bersedia untuk menjadi pembimbing II pada Tugas Akhir mahasiswa berikut:

Nama	: Khaerunisa Anabila
NIM	: 22040115
Program Studi	: Diploma III Teknik Komputer

Judul TA : PENGEMBANGAN RANCANG BANGUN SMART BOX
PENERIMA PAKET BERBASIS *IOT* DAN *BARCODE*.

Demikian pernyataan ini dibuat agar dapat dilaksanakan sebagaimana mestinya.

Tegal, Mei 2025

Mengetahui	Dosen Pembimbing II,
Ka. Prodi DIII Teknik Komputer,	


Ida Afriziana, ST, M.Kom.
NIPY. 12.013.168


Abdul Basit, S.Kom.,M.T.
NIPY. 01.015.098

Lampiran 3 Source Code

```
// Wifi
#include <ESP8266WiFi.h>
#include <ESP8266WiFiMulti.h>
#include <ESP8266HTTPClient.h>

ESP8266WiFiMulti WiFiMulti;

HTTPClient http;
WiFiClient client;
#define USE_SERIAL Serial

String host = "http://dheimuts.web.id/data";
String urlGetQueue = host + "/getQueue";
String urlUpdateQueue = host + "/updateQueue?idQueue=";

String response = "";

unsigned long previousMillis = 0;
const long interval = 2000; // 2 detik

#define relay_off HIGH
#define relay_on LOW

#define pin_doorlock D5

void setup() {
    Serial.begin(115200);

    USE_SERIAL.begin(115200);
    USE_SERIAL.setDebugOutput(false);

    for(uint8_t t = 4; t > 0; t--) {
        USE_SERIAL.printf("[SETUP] Tunggu %d...\n", t);
        USE_SERIAL.flush();
        delay(1000);
    }

    WiFi.mode(WIFI_STA);
    WiFiMulti.addAP("Galay A05 3671", "09092003");

    for (int u = 1; u <= 5; u++)
    {
        if ((WiFiMulti.run() == WL_CONNECTED))
```

```

    {
        USE_SERIAL.println("Wifi Connected");
        USE_SERIAL.flush();
        delay(1000);
    }
    else
    {
        Serial.println("Wifi not Connected");
        delay(1000);
    }
}

pinMode(pin_doorlock, OUTPUT);
digitalWrite(pin_doorlock, relay_off);

Serial.println();
Serial.println("SMART BOX");
Serial.println();

delay(300);
}

void loop() {
    unsigned long currentMillis = millis();

    if (currentMillis - previousMillis >= interval) {
        previousMillis = currentMillis;

        getQueue();
    }
}

void getQueue() {
    if ((WiFiMulti.run() == WL_CONNECTED)) {
        String fullUrl = urlGetQueue;

        http.begin(client, fullUrl);
        int httpCode = http.GET();

        if (httpCode > 0) {
            if (httpCode == 200) {
                response = http.getString();
                Serial.println("Response server: " + response);
                Serial.println();
            }
        }
    }
}

```

```

String statusQueue = getValue(response, '#', 0);
int idQueue = getValue(response, '#', 1).toInt();

Serial.println("Status Queue : " + statusQueue);

if (statusQueue == "available") {
    Serial.println("ID Queue : " + (String) idQueue);
    Serial.println("=====");
    Serial.println();

    http.end();

    Serial.println("Smartbox terbuka");
    Serial.println();
    digitalWrite(pin_doorlock, relay_on);
    delay(5000);
    digitalWrite(pin_doorlock, relay_off);

    updateQueue(idQueue);
} else {
    Serial.println("=====");
    Serial.println();
}
} else {
    Serial.printf("HTTP Error [%d]\n", httpCode);
}
} else {
    Serial.println("Gagal ambil antrian dari server");
}

http.end();
} else {
    Serial.println("WiFi tidak terhubung saat ambil data");
}
}

void updateQueue(int idQueue) {
    if ((WiFiMulti.run() == WL_CONNECTED)) {
        String fullUrl = urlUpdateQueue + (String) idQueue;

        http.begin(client, fullUrl);
        int httpCode = http.GET();

        if (httpCode > 0) {
            if (httpCode == 200) {

```

```

        response = http.getString();
        Serial.println("Response server: " + response);
        Serial.println();
    } else {
        Serial.printf("HTTP Error [%d]\n", httpCode);
    }
} else {
    Serial.println("Gagal kirim history ke server");
}

http.end();
} else {
    Serial.println("WiFi tidak terhubung saat kirim data");
}
}

String getValue(String data, char separator, int index) {
    int found = 0;
    int strIndex[] = { 0, -1 };
    int maxIndex = data.length() - 1;

    for (int i = 0; i <= maxIndex && found <= index; i++) {
        if (data.charAt(i) == separator || i == maxIndex) {
            found++;
            strIndex[0] = strIndex[1] + 1;
            strIndex[1] = (i == maxIndex) ? i + 1 : i;
        }
    }

    return (found > index) ? data.substring(strIndex[0],
strIndex[1]) : "";
}

```