

LAMPIRAN

Lampiran 1. Panduan Penggunaan

PANDUAN PENGGUNAAN SISTEM GERBANG OTOMATIS BERBASIS RFID, WEB, DAN ESP32

A. Tujuan Penggunaan Sistem

Sistem Gerbang otomatis ini dirancang untuk mengontrol akses keluar dan masuk kendaraan menggunakan kartu RFID maupun control melalui website. Sistem akan membuka dan menutup gerbang secara otomatis setelah menerima perintah yang valid. Selain itu, sistem juga mampu menampilkan informasi pada LCD dan mengirimkan aktivitas ke Telegram.

B. Menyalakan Sistem

Sebelum menggunakan sistem, pastikan seluruh perangkat telah terhubung dengan baik dan sumber daya tersedia.

LANGKAH-LANGKAH:

1. Pastikan adaptor atau baterai telah terhubung ke rangkaian.
2. Geser saklar ON/OFF ke posisi ON.
3. Tunggu beberapa detik hingga ESP32 melakukan proses inisialisasi.
4. LCD akan menampilkan informasi bahwa sistem sedang mempersiapkan perangkat.
5. Setelah proses selesai, LCD menampilkan pesan bahwa sistem siap digunakan.

C. Membuka Gerbang Menggunakan RFID



RFID merupakan metode utama untuk membuka gerbang. Setiap pengguna harus memiliki kartu RFID yang telah didaftarkan pada sistem.

LANGKAH-LANGKAH

1. Dekatkan kartu RFID ke RFID Reader.
2. Tahan kartu selama 1–2 detik hingga terbaca oleh sistem.
3. Sistem akan memeriksa data kartu yang diterima.
4. Jika kartu terdaftar, sistem akan memberikan izin akses.
5. Motor DC akan bergerak membuka gerbang.
6. Ketika gerbang mencapai posisi terbuka penuh, limit switch akan menghentikan motor secara otomatis.

D. RFID Tidak Terdaftar

Sistem akan menolak akses apabila kartu RFID yang digunakan tidak terdapat dalam database.

LANGKAH-LANGKAH

1. Tempelkan kartu RFID yang tidak terdaftar.
2. Sistem membaca UID kartu.
3. Sistem melakukan proses verifikasi.
4. Karena kartu tidak ditemukan dalam database, akses ditolak.

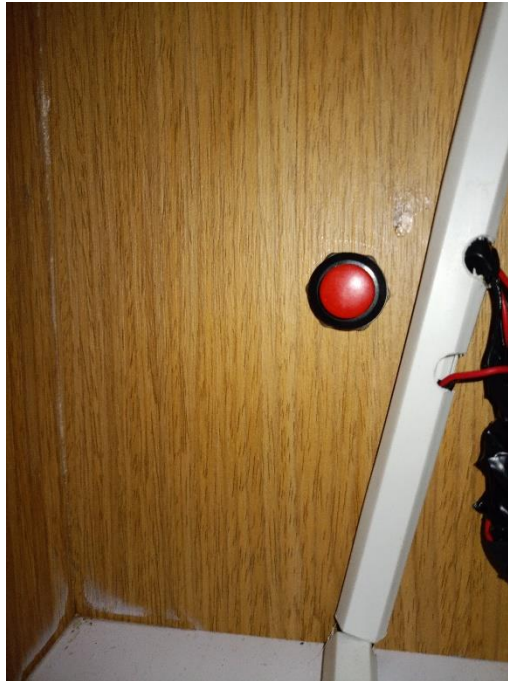
E. Membuka Gerbang Melalui Website

Selain menggunakan RFID, pengguna juga dapat mengendalikan gerbang melalui website yang telah terhubung dengan ESP32.

LANGKAH-LANGKAH

1. Buka website sistem gerbang menggunakan browser.
2. Login menggunakan akun yang telah diberikan.
3. Masuk ke halaman kontrol gerbang.
4. Tekan tombol "Buka Gerbang".
5. Website mengirimkan perintah ke ESP32 melalui jaringan *internet*.
6. ESP32 menjalankan motor DC untuk membuka gerbang.

F. Menutup/Membuka Dari Dalam Gerbang Menggunakan Push Button



Push Button digunakan untuk menutup gerbang secara manual setelah kendaraan berhasil melewati area gerbang.

LANGKAH-LANGKAH:

1. Pastikan kamu berada di area dalam rumah untuk membuka dari dalam.
2. Tekan Push Button satu kali.
3. Sistem menerima perintah penutupan.
4. Motor DC bergerak menutup/membuka gerbang.
5. Ketika limit switch posisi tutup aktif dan limit switch posisi buka aktif, motor akan berhenti secara otomatis.

G. Menutup Gerbang Melalui Website

Apabila pengguna berada jauh dari lokasi gerbang, penutupan dapat dilakukan melalui website.

LANGKAH-LANGKAH:

1. Buka halaman kontrol gerbang.

2. Tekan tombol "Tutup Gerbang".
3. Website mengirimkan perintah ke ESP32.
4. ESP32 mengaktifkan motor DC untuk menutup gerbang.
5. Setelah limit switch tutup aktif, motor berhenti secara otomatis.

H. Monitoring Notifikasi Telegram

Setiap aktivitas sistem akan dikirimkan ke Telegram sebagai media monitoring jarak jauh.

Informasi yang Dikirim

“Peringatan Gerbang Buka Paksa!!!”

I. Mematikan Sistem

Setelah penggunaan selesai, sistem dapat dimatikan dengan aman.

LANGKAH-LANGKAH

1. Pastikan gerbang dalam keadaan tertutup.
2. Geser saklar ke posisi OFF.
3. Pastikan LCD mati dan seluruh perangkat berhenti bekerja.
4. Lepaskan sumber daya apabila sistem tidak digunakan dalam waktu lama.

Lampiran 2. Source Code Program

KODINGAN ARDUINO IDE

```
#include <SPI.h>

#include <MFRC522.h>

#include <Wire.h>

#include <LiquidCrystal_I2C.h>

#include <WiFi.h>

#include <HTTPClient.h>

#include <ArduinoJson.h>

#include <UniversalTelegramBot.h>

#include <WiFiClientSecure.h>

#include <Preferences.h>


// ===== CONFIG WIFI =====

const char* ssid = "AC mati";

const char* password = "00008888";


// ===== CONFIG WEBSITE (API) =====

const char* getStatusUrl = "http://10.195.165.93:8000/api/gate/status";

const char* callbackUrl =

"http://10.195.165.93:8000/api/gate/callback";

const char* verifyRfidUrl = "http://10.195.165.93:8000/api/gate/verify-

rfid";


// ===== TELEGRAM =====

#define BOT_TOKEN "8786248548:AAE3TtGrTkveZjTtONDDzQ6mH0hAk1pvwVE"

#define CHAT_ID "-1003952386220"
```

```

WiFiClientSecure clientSecure;

UniversalTelegramBot bot(BOT_TOKEN, clientSecure);

Preferences preferences;

// ===== PIN =====

#define SS_PIN 5

#define RST_PIN 27

#define SCK_PIN 18

#define MISO_PIN 19

#define MOSI_PIN 23

#define BUZZER 2

#define BUZZER_ALARM 4

#define BUTTON 13

#define IN1 26

#define IN2 25

#define ENA 14

#define LIMIT_BUKA 33

#define LIMIT_TUTUP 32

MFRC522 mfrc522(SS_PIN, RST_PIN);

LiquidCrystal_I2C lcd(0x27, 16, 2);

bool gerbangTerbuka = false;

bool motorAktif = false;

bool alarmAktif = false;

bool lastLimitTutup = HIGH;

```

```

bool modeDaftar = false;

// --- VARIABEL ANTREAN OFFLINE (ARRAY) ---
#define MAX_OFFLINE 10
String pendingActions[MAX_OFFLINE];
String pendingMethods[MAX_OFFLINE];
int pendingCount = 0;

unsigned long lastWebCheck = 0;
unsigned long lastTelegramCheck = 0;
unsigned long lastTelegramAlarm = 0;

// LCD
unsigned long lastScrollTime = 0;
int scrollPos = 0;
String pesanJalan = "          SILAHKAN TEMPELKAN KARTU ANDA          ";

bool tampilPesanSementara = false;
unsigned long pesanSampai = 0;

void setup() {
    Serial.begin(115200);
    WiFi.begin(ssid, password);
    clientSecure.setInsecure();
    preferences.begin("rfid-data", false);
    SPI.begin(SCK_PIN, MISO_PIN, MOSI_PIN, SS_PIN);
    mfrc522.PCD_Init();
}

```

```

Wire.begin(21, 22);

lcd.init();

lcd.backlight();

pinMode(BUZZER, OUTPUT);

pinMode(BUZZER_ALARM, OUTPUT);

pinMode(BUTTON, INPUT_PULLUP);

pinMode(IN1, OUTPUT);

pinMode(IN2, OUTPUT);

pinMode(LIMIT_BUKA, INPUT_PULLUP);

pinMode(LIMIT_TUTUP, INPUT_PULLUP);

ledcAttach(ENA, 1000, 8);
}

String getUID() {
    String uidStr = "";
    for (byte i = 0; i < mfrc522.uid.size; i++) {
        uidStr += (mfrc522.uid.uidByte[i] < 0x10 ? "0" : "");
        uidStr += String(mfrc522.uid.uidByte[i], HEX);
    }
    uidStr.toUpperCase();
    return uidStr;
}

// --- FUNGSI BARU UNTUK MENAMBAH ANTREAN OFFLINE ---
void tambahDataOffline(String action, String method) {
    if (pendingCount < MAX_OFFLINE) {
        pendingActions[pendingCount] = action;
    }
}

```

```

    pendingMethods[pendingCount] = method;

    pendingCount++;

    Serial.println("Data offline disimpan: " + action + " via " + method);
} else {
    Serial.println("Memori offline penuh!");
}
}

```

```

void scrollText() {
    if (motorAktif || alarmAktif) return;
    if (tampilPesanSementara) {
        if (millis() < pesanSampai) return;
        else {
            tampilPesanSementara = false;
            scrollPos = 0;
            lcd.clear();
        }
    }
}

```

```

if (modeDaftar) {
    lcd.setCursor(0, 0);
    lcd.print(" MODE TAMBAH ID ");
    lcd.setCursor(0, 1);
    lcd.print(" SILAKAN TAP... ");
    return;
}

```

```

    if (millis() - lastScrollTime > 250) {
        lcd.setCursor(0, 0);
        lcd.print(pesanJalan.substring(scrollPos, scrollPos + 16));
        scrollPos++;
        if (scrollPos > pesanJalan.length() - 16) scrollPos = 0;
        lastScrollTime = millis();
    }
    lcd.setCursor(0, 1);
    lcd.print(gerbangTerbuka ? "STATUS: TERBUKA " : "STATUS: TERTUTUP");
}

void sendCallback(String action, String method) {
    if (WiFi.status() == WL_CONNECTED) {
        HTTPClient http;
        http.begin(callbackUrl);
        http.addHeader("Content-Type", "application/json");
        String json = "{\"action\":\"" + action + "\", \"method\":\"" + method
+ "\"}";

        Serial.println("Mengirim Callback: " + json);
        int httpCode = http.POST(json);

        if (httpCode > 0) {
            Serial.printf("[HTTP] Respon: %d\n", httpCode);
            if (httpCode == 200) Serial.println("Database Berhasil
Diperbarui.");
        } else {

```

```

        Serial.printf("[HTTP]      Gagal,      error:      %s\n",
http.errorToString(httpCode).c_str());
    }
    http.end();
}
}

```

```

void moveGate(bool open) {
    if (alarmAktif) return;
    motorAktif = true;
    lcd.clear();
    lcd.print(open ? "MEMBUKA..." : "MENUTUP...");
    digitalWrite(BUZZER, HIGH);
    delay(200);
    digitalWrite(BUZZER, LOW);
    digitalWrite(IN1, open);
    digitalWrite(IN2, !open);
    ledcWrite(ENA, 100);
    int limitPin = open ? LIMIT_BUKA : LIMIT_TUTUP;
    unsigned long startTime = millis();
    while (digitalRead(limitPin) == HIGH && millis() - startTime < 1500000)
    {
        delay(10);
    }
    ledcWrite(ENA, 0);
    digitalWrite(IN1, LOW);
    digitalWrite(IN2, LOW);
}

```

```

    gerbangTerbuka = open;

    motorAktif = false;

    lcd.clear();
}

void checkWebCommand() {

    if (WiFi.status() == WL_CONNECTED && !motorAktif) {

        HTTPClient http;

        http.begin(getStatusUrl);

        int httpCode = http.GET();

        if (httpCode == 200) {

            String payload = http.getString();

            StaticJsonDocument<200> doc;

            deserializeJson(doc, payload);

            String command = doc["command"];

            if (command == "buka" && !gerbangTerbuka) moveGate(true);

            else if (command == "tutup" && gerbangTerbuka) moveGate(false);

            else if (command == "scan") { modeDaftar = true; }

            else if (command == "standby") { modeDaftar = false; }

            else if (command == "reset" && alarmAktif) {

                alarmAktif = false;

                digitalWrite(BUZZER_ALARM, LOW);

                lcd.clear();

                lcd.print("ALARM RESET");
            }
        }
    }
}

```

```

        bot.sendMessage(CHAT_ID, "✔ Alarm berhasil di-reset melalui
Website Dashboard!", "");
        delay(2000);
        sendCallback("standby", "dashboard-web");
    }
    else if (command == "sukses") {
        lcd.clear();
        lcd.print("  DAFTAR KARTU  ");
        lcd.setCursor(0, 1);
        lcd.print("    BERHASIL    ");
        digitalWrite(BUZZER, HIGH); delay(100); digitalWrite(BUZZER,
LOW); delay(100);
        digitalWrite(BUZZER, HIGH); delay(100); digitalWrite(BUZZER,
LOW);
        tampilPesanSementara = true;
        pesanSampai = millis() + 4000;
        sendCallback("standby", "system");
    }
    else if (command == "gagal_daftar") {
        lcd.clear();
        lcd.print(" DAFTAR GAGAL! ");
        lcd.setCursor(0, 1);
        lcd.print("KARTU SUDAH ADA ");

        digitalWrite(BUZZER, HIGH);
        delay(1000);
        digitalWrite(BUZZER, LOW);

```

```

        tampilPesanSementara = true;

        pesanSampai = millis() + 4000;

        sendCallback("standby", "system");
    }
}

http.end();
}
}

void checkTelegramReset() {
    if (millis() - lastTelegramCheck > 2000) {
        int numNewMessages = bot.getUpdates(bot.last_message_received + 1);
        while (numNewMessages) {
            for (int i = 0; i < numNewMessages; i++) {
                String text = bot.messages[i].text;
                String chat_id = bot.messages[i].chat_id;
                if (chat_id != CHAT_ID) continue;
                if (text == "/reset") {
                    if (alarmAktif) {
                        alarmAktif = false;
                        digitalWrite(BUZZER_ALARM, LOW);
                        lcd.clear();
                        lcd.print("ALARM RESET");
                        bot.sendMessage(CHAT_ID, "✔ Alarm berhasil di-reset", "");
                        sendCallback("standby", "telegram-bot");
                        delay(2000);
                    }
                }
            }
        }
    }
}

```

```

        } else {

            bot.sendMessage(CHAT_ID, "⚠ Alarm tidak aktif", "");

        }

    }

    numNewMessages = bot.getUpdates(bot.last_message_received + 1);

}

lastTelegramCheck = millis();

}

}

bool checkLocalRFID(String uid) {

    return preferences.isKey(uid.c_str());

}

void loop() {

    scrollText();

    // --- LOGIKA SINKRONISASI SAAT ONLINE ---

    if (WiFi.status() == WL_CONNECTED) {

        if (pendingCount > 0) {

            Serial.println("Koneksi Pulih. Memulai Sinkronisasi Data Offline...");

            delay(5000); // Jeda krusial agar koneksi internet benar-benar siap

            for (int i = 0; i < pendingCount; i++) {

                sendCallback(pendingActions[i], pendingMethods[i]);

            }

        }

    }

}

```

```

        String notif = "i [SINKRONISASI] Kejadian " + pendingActions[i] +
" via " + pendingMethods[i] + " saat offline telah dicatat.";

        bot.sendMessage(CHAT_ID, notif, "");

        delay(1000);

    }

    pendingCount = 0;

    Serial.println("Sinkronisasi Selesai.");

}

checkTelegramReset();

if (millis() - lastWebCheck > 2000) {

    checkWebCommand();

    lastWebCheck = millis();

}

}

// --- BUTTON ---

if (digitalRead(BUTTON) == LOW) {

    delay(200);

    if (alarmAktif) {

        alarmAktif = false;

        digitalWrite(BUZZER_ALARM, LOW);

        lcd.clear();

        lcd.print("ALARM RESET");

        if (WiFi.status() == WL_CONNECTED) {

```

```

        bot.sendMessage(CHAT_ID, "✓ Alarm berhasil di-reset melalui
tombol Button!", "");

        sendCallback("standby", "button-fisik");
    } else {
        tambahDataOffline("standby", "button-fisik");
    }
    delay(2000);
} else if (!motorAktif) {
    bool target = !gerbangTerbuka;
    moveGate(target);
    if (WiFi.status() == WL_CONNECTED) {
        sendCallback(target ? "buka" : "tutup", "button");
    } else {
        tambahDataOffline(target ? "buka" : "tutup", "button");
    }
}
while (digitalRead(BUTTON) == LOW);
}

// --- RFID (HYBRID) ---

if (!alarmAktif && !motorAktif && mfrc522.PICC_IsNewCardPresent() &&
mfrc522.PICC_ReadCardSerial()) {
    String scannedUID = getUID();

    if (modeDaftar) {
        lcd.clear();
        lcd.print("MEMBACA KARTU...");
    }
}

```

```

lcd.setCursor(0, 1);
lcd.print("TUNGGU SEBENTAR.");

if (WiFi.status() == WL_CONNECTED) {
    HTTPClient http;
    http.begin(verifyRfidUrl);
    http.addHeader("Content-Type", "application/json");
    String json = "{\"uid\":\"" + scannedUID + "\"}";
    http.POST(json);
    http.end();
}

digitalWrite(BUZZER, HIGH); delay(200); digitalWrite(BUZZER, LOW);
delay(1500);

mfrc522.PICC_HaltA();
mfrc522.PCD_StopCrypto1();
}
else {
    lcd.clear();
    lcd.print("MEMERIKSA KARTU.");
    lcd.setCursor(0, 1);
    lcd.print("TUNGGU SEBENTAR.");

    bool valid = false;
    String ownerName = "";

```

```

if (WiFi.status() == WL_CONNECTED) {
    HTTPClient http;
    http.begin(verifyRfidUrl);
    http.addHeader("Content-Type", "application/json");
    String json = "{\"uid\":\"" + scannedUID + "\"}";
    int httpCode = http.POST(json);

    if (httpCode == 200) {
        String payload = http.getString();
        StaticJsonDocument<200> doc;
        deserializeJson(doc, payload);
        valid = doc["valid"];
        if(valid){
            ownerName = doc["owner"].as<String>();
            preferences.putBool(scannedUID.c_str(), true);
        } else {
            if (preferences.isKey(scannedUID.c_str())) {
                preferences.remove(scannedUID.c_str());
            }
        }
    }
    http.end();
}

if (!valid && checkLocalRFID(scannedUID)) {
    valid = true;
    ownerName = "USER LOKAL";
}

```

```

    }

    if (valid) {
        lcd.clear();
        lcd.print("SELAMAT DATANG");
        lcd.setCursor(0, 1);
        lcd.print("" + ownerName);
        digitalWrite(BUZZER, HIGH); delay(100); digitalWrite(BUZZER,
LOW); delay(50);
        digitalWrite(BUZZER, HIGH); delay(100); digitalWrite(BUZZER,
LOW);
        delay(2000);
        bool target = !gerbangTerbuka;
        moveGate(target);
        if (WiFi.status() == WL_CONNECTED) {
            sendCallback(target ? "buka" : "tutup", "rfid-" + scannedUID);
        } else {
            tambahDataOffline(target ? "buka" : "tutup", "rfid-" +
scannedUID);
        }
    } else {
        lcd.clear();
        lcd.print("AKSES DITOLAK ");
        lcd.setCursor(0, 1);
        lcd.print("KARTU TDK VALID ");
        digitalWrite(BUZZER, HIGH); delay(500); digitalWrite(BUZZER,
LOW);
    }
}

```

```

        tampilPesanSementara = true;

        pesanSampai = millis() + 3000;
    }

    mfrc522.PICC_HaltA();

    mfrc522.PCD_StopCrypto1();
}

}

// --- DETEKSI GERBANG DIBUKA PAKSA ---

bool currentLimitTutup = digitalRead(LIMIT_TUTUP);

if (lastLimitTutup == LOW && currentLimitTutup == HIGH && !motorAktif
&& !gerbangTerbuka) {
    alarmAktif = true;

    if (WiFi.status() == WL_CONNECTED) {
        bot.sendMessage(CHAT_ID, "🚪 GERBANG DIBUKA PAKSA!", "");

        sendCallback("alarm", "forced");
    } else {
        tambahDataOffline("alarm", "forced");
    }
}

lastLimitTutup = currentLimitTutup;

// --- LOOP ALARM ---

if (alarmAktif) {
    digitalWrite(BUZZER_ALARM, (millis() / 500) % 2);

    lcd.setCursor(0, 0);

    lcd.print("!!! ALARM !!!  ");
}

```

```

        if (WiFi.status() == WL_CONNECTED && millis() - lastTelegramAlarm >
10000) {

            bot.sendMessage(CHAT_ID, "🚪 GERBANG MASIH DIBUKA PAKSA!", "");

            lastTelegramAlarm = millis();

        }
    } else {

        digitalWrite(BUZZER_ALARM, LOW);

    }
}

```

Lampiran 3. Surat Permohonan Ijin

PERUMAHAN PURI RARASTITI

Alamat: 4424+75J, Tanglog, Dukuhturi, Kec. Dukuhturi, Kabupaten Tegal, Jawa Tengah

Nomor : 572.03/KOM-HN/III/2026

Lampiran : -

Perihal : Surat Keterangan Observasi

Dengan Hormat,

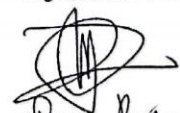
Menindak lanjuti surat permohonan ijin observasi Tugas Akhir (TA) dengan Nomor: 572.03/KOM-HN/III/2026. Yang bertanda tangan dibawah ini pemilik/penghuni rumah di Perumahan Puri Rarastiti menerangkan dengan sesungguhnya bahwa:

No	Nama	Nim	Program Studi
1.	Dhanu Anumerta Hudiarto	23040082	DIII TEKNIK KOOMPUTER
2.	Akhmad Sabbit Aqdami	23040097	DIII TEKNIK KOOMPUTER

Yang bersangkutan telah melaksanakan observasi di Perumahan Puri Rarastiti sejak 5 Desember 2025.

Demikian surat keterangan ini dibuat dengan sebenarnya untuk dapat dipergunakan sebagaimana mestinya.

Tegal, Maret 2026


(..Purwa...Purwa..)

Lampiran 4. Surat Kesediaan Membimbing Dosen Pembimbing I

SURAT KESEDIAAN MEMBIMBING TA

Yang bertanda tangan di bawah ini:

NAMA : Nurohim, S.ST, M.Kom.
NIDN : 0625067701
NIPY : 09.017.342
Jabatan Struktural : -
Jabatan Fungsional : Lektor

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

Nama : Dhanu Anumerta Hudiarto
NIM : 23040082
Program Studi : D-3 Teknik Komputer
Judul TA : RANCANG BANGUN SISTEM OTOMATISASI
PINTU GERBANG RUMAH BERBASIS ESP32

Demikian pernyataan ini dibuat agar dapat dilaksanakan sebagaimana mestinya.

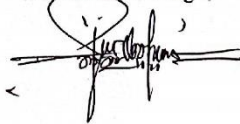
Tegal, 10 Maret 2026

Menyetujui

Ka Prodi DIII Teknik Komputer,

Dosen Pembimbing I,


Arif Rakhman, SE, S.Pd, M.Kom.
NIPY. 05.016.291


Nurohim, S.ST, M.Kom.
NIPY. 09.017.342

Lampiran 5. Surat Kesiediaan Membimbing Dosen Pembimbing II

SURAT KESEDIAAN MEMBIMBING TA

Yang bertanda tangan di bawah ini:

NAMA : Arif Rakhman, SE, S.Pd, M.Kom.
NIDN : 0623118301
NIPY : 05.016.291
Jabatan Struktural : Ka. Prodi DIII Teknik Komputer
Jabatan Fungsional : Lektor

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

Nama : Dhanu Anumerta Hudiarto
NIM : 23040082
Program Studi : D-3 Teknik Komputer
Judul TA : RANCANG BANGUN SISTEM OTOMATISASI
PINTU GERBANG RUMAH BERBASIS ESP32

Demikian pernyataan ini dibuat agar dapat dilaksanakan sebagaimana mestinya.

Tegal, 10 Maret 2026

Menyetujui

Ka Prodi DIII Teknik Komputer, Dosen Pembimbing II,


Arif Rakhman, SE, S.Pd, M.Kom.
NIPY. 05.016.291


Arif Rakhman, SE, S.Pd, M.Kom.
NIPY. 05.016.291

Lampiran 6. Surat Observasi

	Universitas Harkat Negeri	Sekolah Vokasi Program Studi D-3 Teknik Komputer												
Nomor	: 572.03/KOM-HN/III/2026													
Lampiran	: -													
Perihal	: Permohonan Izin Observasi Tugas Akhir (TA)													
Kepada Yth.														
Kepala Perumahan Puri Rarastiti														
Di Tanglong, Dukuhturi, Kec. Dukuhturi, Kab. Tegal, Jawa Tengah														
Dengan Hormat,														
Sehubungan dengan tugas mata kuliah Tugas Akhir (TA) yang akan diselenggarakan di semester V Program Studi Diploma III Teknik Komputer Universitas Harkat Negeri, Maka dengan ini kami mengajukan izin observasi pengambilan data di Perumahan Puri Rarastiti yang Bapak / Ibu Pimpin, untuk kepentingan dalam pembuatan produk Tugas Akhir, dengan Mahasiswa sebagai berikut:														
<table border="1"><thead><tr><th>No</th><th>NIM</th><th>NAMA</th><th>NO HP</th></tr></thead><tbody><tr><td>1</td><td>23040082</td><td>DHANU ANUMERTA HUDIARTO</td><td>085830800358</td></tr><tr><td>2</td><td>23040097</td><td>AKHMAD SABBIT AQDAMI</td><td>081911669856</td></tr></tbody></table>			No	NIM	NAMA	NO HP	1	23040082	DHANU ANUMERTA HUDIARTO	085830800358	2	23040097	AKHMAD SABBIT AQDAMI	081911669856
No	NIM	NAMA	NO HP											
1	23040082	DHANU ANUMERTA HUDIARTO	085830800358											
2	23040097	AKHMAD SABBIT AQDAMI	081911669856											
Demikian surat permohonan ini kami sampaikan atas izin dan kerjasamanya kami sampaikan terima kasih.														
Tegal, 10 Maret 2026														
Ka. Prodi DIII Teknik Komputer Universitas Harkat Negeri														
														
Arif Rakhman SE, S.Pd, M.Kom NIPY. 05.016.291														
Program Studi D-3 Teknik Komputer, Sekolah Vokasi Gedung Utama Ruang B.03 Kampus Mataram - Jalan Mataram No.11 Kota Tegal WhatsApp: +62-85291833673 Surel: komputer@harkatnegeri.ac.id														

Lampiran 7. Dokumentasi Observasi

