


LAMPIRAN

1. Surat Penelitian

**FAKULTAS SAINS DAN TEKNOLOGI**
UNIVERSITAS LABUHANBATU
PROGRAM STUDI :
AGROTEKNOLOGI - TEKNOLOGI INFORMASI - SISTEM INFORMASI - MANAJEMEN INFORMATIKA
Jl. SM. Raja No. 126-A KM. 3,5 Aek Tapa - Rantauprapat - Sumatera Utara - Pos 21415
Telp./Fax. (0624) 21901

Nomor : 04/TI/FST-ULB/I/2026
Hal : Permohonan Izin Penelitian

Kepada Yth.
Bapak Lurah Urung Kompas
di -
Tempat


Sehubungan dengan rencana Penelitian untuk Skripsi/Tugas Akhir Mahasiswa Program Studi S-1 Teknologi Informasi Fakultas Sains dan Teknologi tersebut dibawah ini :

Nama : FADHILAH YASMIN MUNTHE
NPM : 2208100022
Program Studi : T-1 Teknologi Informasi
Judul Tugas Akhir : RANCANG BANGUN PADA SISTEM PENDETEKSI SINAR UV UNTUK HIMBAUAN KESEHATAN MANUSIA BERBASIS IOT
Lokasi Penelitian : Perumahan Graha Kompas Idaman Kel. Urung kompas, kec. Rantau selatan, Rantau prapat, Kab. Labuhanbatu, Sumatera Utara


Untuk keperluan tersebut diatas, agar kiranya dapat memberi izin pelaksanaan penelitian di wilayah Bapak/Ibu. Dalam proses pelaksanaanya segala sesuatu yang berkaitan dengan penelitian tersebut akan diselesaikan oleh mahasiswa yang bersangkutan.

Demikian hal ini kami sampaikan atas perhatian dan bantuannya diucapkan terima kasih.

Rantauprapat, 20 Januari 2026
Fakultas Sains dan Teknologi
Ka. Prodi Teknologi Informasi


Rahmadani Pane, S.Kom, M.Kom
NIBN. 0110058601

2. Surat Balasan Penelitian



PEMERINTAH KABUPATEN LABUHANBATU
KECAMATAN RANTAU SELATAN
KELURAHAN URUNG KOMPAS
 Jl. Suka Dame Rantauprapat Kode Pos 21415

REKOMENDASI PENELITIAN
 Nomor : 072/ 113 /Kesos-Pemb/2026

Sehubungan dengan Surat Ka. Prodi Teknologi Informasi Fakultas Sains dan Teknologi Universitas Labuhanbatu Nomor : 04/TI/FST-ULB/I/2026, tanggal 20 Januari 2026 perihal Permohonan Izin Penelitian.

Dengan ini memberikan Izin Penelitian Kepada :

NO.	NAMA	NIM	JUDUL
1.	FADHILAH YASMIN MUNTHE	2208100022	RANCANG BANGUN PADA SISTEM PENDETEKSI SINAR UV UNTUK HIMBAUAN KESEHATAN MANUSIA BERBASIS IOT

Dengan Ketentuan sebagai berikut :

1. Melaksanakan pengambilan data dimaksud pada tanggal 21 Januari 2026 s/d selesai yang dilaksanakan di Perumahan Graha Kompas Idaman Kelurahan Urung Kompas Kecamatan Rantau Selatan Kabupaten Labuhanbatu.
2. Mematuhi segala peraturan yang berlaku di lokasi penelitian dan tetap berkoordinasi dengan masyarakat setempat.
3. Melaporkan hasil penelitian Kepada Kepala Kelurahan Urung Kompas Kecamatan Rantau Selatan Kabupaten Labuhanbatu.

Demikian Izin Penelitian ini diperbuat untuk dapat dipergunakan seperlunya.

Ditetapkan di : Urung Kompas
 Pada Tanggal : 21 Januari 2026
KEPALA KELURAHAN URUNG KOMPAS

 RM. AZMIL ALAMSYAH RITONGA, SE
 PENATA
 NIP. 19771209 200801 1 001

3. Dokumentasi Penelitian



4. Script

```

#include <WiFi.h>
#include <PubSubClient.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <ArduinoJson.h>
#include <HTTPClient.h>

// ===== WIFI ===== const char* ssid =
"DILAAA_"; const char* password = "dilaaa2703";

// ===== MQTT =====
const char* mqttServer = "broker.hivemq.com"; const int  mqttPort = 1883; const char*
topicUV = "dila/uv"; const char* topicNotif = "dila/notifikasi";

// ===== THINGSPEAK ===== String apiKeyTS =
"K0N7JYTLNRGN14KA"; const char* serverTS = "https://api.thingspeak.com/update";

// ===== PIN =====
#define UV_PIN 34
#define LED_HIJAU 26
#define LED_MERAH 27

// ===== LCD =====
LiquidCrystal_I2C lcd(0x27, 16, 2);

// ===== VARIABEL =====
#define SAMPLE_COUNT 20
int lastLevel = -1;

WiFiClient espClient;
PubSubClient mqttClient(espClient);

// ===== SETUP ===== void setup() {
Serial.begin(115200);
pinMode(LED_HIJAU, OUTPUT); pinMode(LED_MERAH, OUTPUT);
analogSetPinAttenuation(UV_PIN, ADC_11db);

lcd.init();
lcd.backlight();
lcd.print("Connecting WiFi");

WiFi.begin(ssid, password); while (WiFi.status() != WL_CONNECTED) { delay(500);
Serial.print(".");
}
Serial.println("\nWiFi Connected!");

lcd.clear(); lcd.print("WiFi Connected"); delay(1000);

```

```

mqttClient.setServer(mqttServer, mqttPort); connectMQTT();
}

// ===== MQTT RECONNECT ===== void
connectMQTT() { while (!mqttClient.connected()) {
  Serial.print("Connecting MQTT...");
  String clientId = "ESP32-UV-" + String(random(0xffff), HEX); if
(mqttClient.connect(clientId.c_str())) {
  Serial.println("MQTT Connected!");
  } else {
  Serial.print("Gagal rc=");
  Serial.print(mqttClient.state()); Serial.println(" retry 3s..."); delay(3000);
  }
}
}

// ===== LOOP ===== void loop() { if
(!mqttClient.connected()) connectMQTT(); mqttClient.loop();

float uvIndex = readUV(); int level = getUVLevel(uvIndex);

Serial.print("UV Index: "); Serial.print(uvIndex, 1);
Serial.print(" | Level: "); Serial.println(level);

lcd.clear();
lcd.setCursor(0, 0); lcd.print("UV Index:"); lcd.setCursor(10, 0); lcd.print(uvIndex, 1);
lcd.setCursor(0, 1); if (level <= 1) { lcd.print("AMAN");
digitalWrite(LED_HIJAU, HIGH); digitalWrite(LED_MERAH, LOW);
} else { lcd.print("BAHAYA"); digitalWrite(LED_HIJAU, LOW);
digitalWrite(LED_MERAH, HIGH);
}

publishUV(uvIndex, level);

if (level != lastLevel) { publishNotifikasi(uvIndex, level); lastLevel = level;
}

sendToThingSpeak(uvIndex); delay(20000);
}

// ===== BACA UV ===== float readUV() { long
total = 0; for (int i = 0; i < SAMPLE_COUNT; i++) { total += analogRead(UV_PIN);
delay(5); } float uvIndex = (total / (float)SAMPLE_COUNT) / 190.0; if (uvIndex < 0)
uvIndex = 0; if (uvIndex > 11) uvIndex = 11; return uvIndex;
}

// ===== LEVEL UV =====

```

```

int getUVLevel(float uv) { if (uv <= 2) return 0; if (uv <= 5) return 1; if (uv <= 7)
return 2; if (uv <= 10) return 3; return 4;
}

// ===== PUBLISH DATA UV =====
void publishUV(float uvIndex, int level) {
StaticJsonDocument<128> doc; doc["uvIndex"] = uvIndex;
doc["level"] = level; char payload[128]; serializeJson(doc, payload);
mqttClient.publish(topicUV, payload);
Serial.println("Publish UV: " + String(payload));
}

// ===== PUBLISH NOTIFIKASI =====
void publishNotifikasi(float uvIndex, int level) { StaticJsonDocument<128> doc;
doc["uvIndex"] = uvIndex; doc["level"] = level; char payload[128]; serializeJson(doc,
payload);
mqttClient.publish(topicNotif, payload);
Serial.println("Publish Notif: " + String(payload));
}

// ===== THINGSPEAK =====
void sendToThingSpeak(float uv) {
if (WiFi.status() != WL_CONNECTED) return;
HTTPClient http;
String url = String(serverTS) + "?api_key=" + apiKeyTS + "&field1=" + uv;
http.begin(url); int code = http.GET();
Serial.print("ThingSpeak: "); Serial.println(code); http.end();
}

```