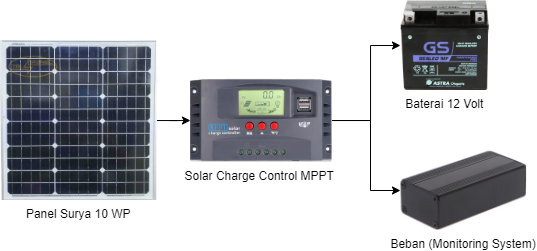
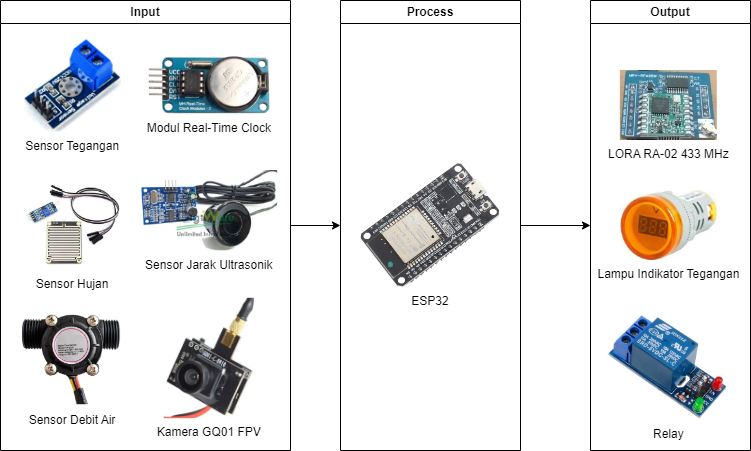
1. **Diagram Blok**
   1. **Power System**



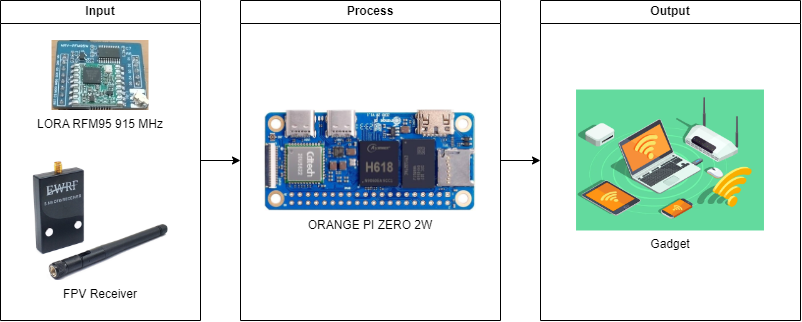
* 1. **Monitoring System (Node Sensor)**

Sistem ini berada Bersama Power System di panel box yang berada di objek penelitian.



* 1. **Gateway**

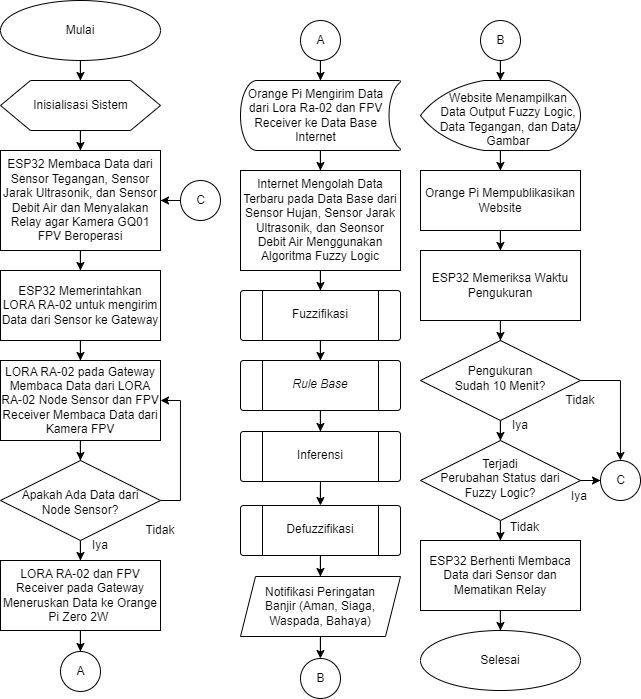
Sistem ini berada di tempat monitoring/kantor.



1. **Flowchart**

A diagram of a flowchart

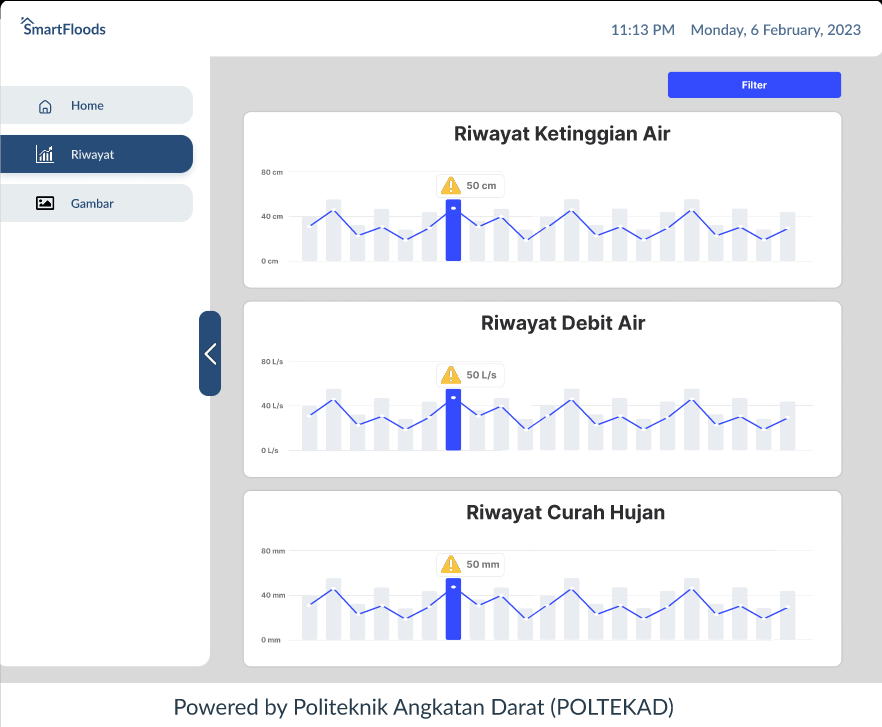
Description automatically generated



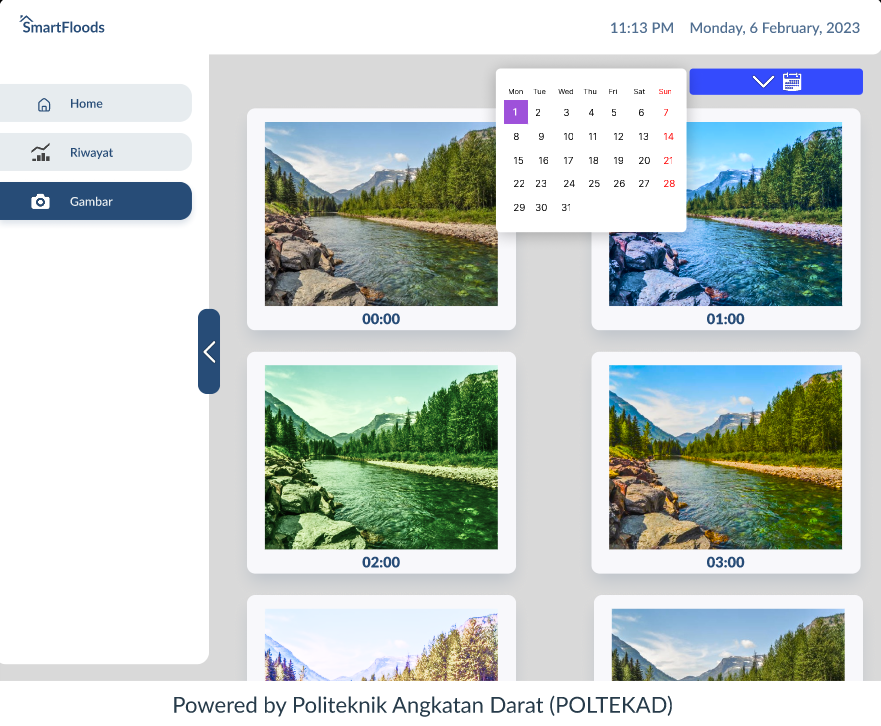
1. **Tampilan** **Website**
   1. **Tampilan Dashboard**



* 1. **Tampilan Riwayat Page**



* 1. **Tampilan Video History Page**



1. **Kode program**
   1. **Program Baterai dengan Trimmed Mean**

#define pinTegangan 35

float faktorTrim = 0.20; //5%-20%

int jumlahSampel = 50;

float maxV = 1260.00; // maksimum tegangan baterai

float minV = 900.00; // minimum tegangan baterai

bool notifBAT;

void setupBaterai() {

pinMode(pinTegangan, INPUT);

}

int persenBAT() {

int rawVoltage = analogRead(pinTegangan);

//ngambil data sesuai jumlah sample

float nilaiSensor[jumlahSampel];

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

nilaiSensor[i] = rawVoltage;

}

//proses pengurutan (sorting)

for (int i = 0; i < jumlahSampel - 1; i++) {

for (int j = i + 1; j < jumlahSampel; j++) {

if (nilaiSensor[i] > nilaiSensor[j]) {

float temp = nilaiSensor[i];

nilaiSensor[i] = nilaiSensor[j];

nilaiSensor[j] = temp;

}

}

}

//rata-rata (trimmed mean)

float sum = 0;

for (int i = jumlahSampel \* (faktorTrim / 2); i < jumlahSampel \* (1 - (faktorTrim / 2)); i++) {

sum += nilaiSensor[i];

}

int trimmedMean = sum / (jumlahSampel \* (1 - faktorTrim));

float regresiLinear = (0.0045 \* trimmedMean) + 0.6842;

int persen = map(regresiLinear \* 100, minV, maxV, 0, 100);

return persen;

}

* 1. **Program Kirim Data ke Thingsboard**

#include <WiFiManager.h> // https://github.com/tzapu/WiFiManager

#include <ArduinoJson.h>

#include <WiFi.h>

#include <HTTPClient.h>

String payloadAN;

void aturAssistNow();

void setupWiFi() {

WiFi.mode(WIFI\_STA);

WiFiManager wm;

bool res;

res = wm.autoConnect("Chandra's Skripsi");

if (!res) {

ESP.restart();

}

//koneksi fitur AssistNow!

aturAssistNow();

}

void kirimData(double lat, double lng, bool geofence, String Pengguna, bool statHB, String persenBat) {

WiFiClient client;

HTTPClient http;

String server = "http://demo.thingsboard.io/api/v1/yqlVrHDSfAHd864Bekkp/telemetry"; //HB2 http://demo.thingsboard.io/api/v1/t2kt1QuyO4zMMUbTbaRS/telemetry // HB1 http://demo.thingsboard.io/api/v1/yqlVrHDSfAHd864Bekkp/telemetry

http.addHeader("Content-Type", "application/json");

http.begin(client, server);

DynamicJsonDocument doc(1024);

char buffer[256];

doc["latitude"] = String(lat, 16);

doc["longitude"] = String(lng, 16);

doc["isInside"] = geofence == true ? "in" : "out";

doc["user"] = Pengguna;

doc["status"] = statHB == true ? "on" : "off";

doc["batt"] = persenBat;

size\_t n = serializeJson(doc, buffer);

int rc = http.POST(buffer);

http.end(); // Ensure http.end() is called

}

Note :

1. Fuzzy curah hujan
2. Fuzzy waterflow
3. Fuzzy Logic Sensor Ultrasonik

Ketinggian normal air dari permukaan jalan 60 + 90 = 150 cm

X = 90; Sensor baca = 150 cm

Aman = > 40 (130 cm dari sensor)

Siaga = 20 - 30 (110 – 120 cm dari sensor)

Waspada = 10 – 20 (100 – 110 cm dari sensor)

Bahaya = X + 5 (5-10) (95 – 100 cm dari sensor)

======. (delay >> variable global)

Aki 12,4 – 13,2

Diukur

=== regresi semua sensor

1. **Sensor ultrasonic**

a. berjarak 130 cm dari sensor maka indikator (aman)

b. berjarak 120 - 110 cm dari sensor maka indikator (siaga)

c. berjarak 110 - 100 cm dari sensor maka indikator (waspada)

d. berjarak 100 - 85 cm dari sensor maka indikator (bahaya)

Fuzzy Set:

Bahaya: μ\_bahaya(x) = trapesium(x, 85, 100, 100, 100)

Waspada: μ\_waspada(x) = segitiga(x, 100, 110, 120)

Siaga: μ\_siaga(x) = segitiga(x, 110, 120, 130)

Aman: μ\_aman(x) = trapesium(x, 120, 150, 150, 150)

1. **Sensor debit air**

a. flowRate < 10==>"Aman";

b. flowRate >= 10 && flowRate < 20 maka==> "Siaga";

c. flowRate >= 20 && flowRate < 30 maka==> "Waspada";

d. FlowRate >= 30 maka==> "Bahaya";

Fuzzy Set:

Bahaya: μ\_bahaya(x) = trapesium(x, 30, 35, 100, 100)

Waspada: μ\_waspada(x) = segitiga(x, 20, 30, 40)

Siaga: μ\_siaga(x) = segitiga(x, 10, 20, 30)

Aman: μ\_aman(x) = trapesium(x, 0, 0, 9, 10)

1. **Sensor hujan**

a. sensor\_hujan < 1023 ==>bahaya

b. sensor\_hujan >= 800 && sensor\_hujan < 2047 ==>waspada

c. sensor\_hujan >= 1800 && sensor\_hujan < 3071 ==>Siaga

d. sensor\_hujan >= 2800 ==>Aman

Fuzzy Set: Bahaya:

μ\_bahaya(x) = trapesium(x, 0, 0, 1022, 2046)

Waspada: μ\_waspada(x) = segitiga(x, 800, 2047, 3070)

Siaga: μ\_siaga(x) = segitiga(x, 1800, 3071, 4095)

Aman: μ\_aman(x) = trapesium(x, 2800, 4096, 4096, 4096)