

Kelompok 1

RANCANG BANGUN SISTEM MONITORING KELEMBABAN DAN SUHU BERBASIS IOT (HUMIDITY SENSOR PROJECT)



Disusun Oleh Kelompok 1:

1. **AAI** (NIM...)
2. **DAMAR** (NIM...)
3. **BAGUS** (NIM...)
4. **FATAN** (NIM...)
5. **NASHRUL** (20230140105)

[Teknologi Informasi] [Universitas Muhammadiyah Yogyakarta] 2026

DAFTAR ISI

1. BAB 1: PENDAHULUAN
2. BAB 2: PERANCANGAN SISTEM (UML)
3. BAB 3: IMPLEMENTASI APLIKASI
4. BAB 4: KESIMPULAN
5. DAFTAR PUSTAKA
6. LAMPIRAN

BAB 1: PENDAHULUAN

1.1 Latar Belakang

Pemantauan suhu dan kelembaban merupakan aspek penting dalam berbagai bidang, mulai dari pertanian, pergudangan, hingga kenyamanan ruangan. Teknologi *Internet of Things* (IoT) memungkinkan proses pemantauan ini dilakukan secara *real-time* dan terpusat. Dalam proyek ini, kami mengembangkan sistem monitoring menggunakan sensor DHT11 dan mikrokontroler ESP32 yang terintegrasi dengan backend server berbasis Node.js dan database MySQL.

1.2 Rumusan Masalah

1. Bagaimana merancang diagram alur sistem (Workflow) dari sensor hingga ke database?
2. Bagaimana struktur database (ERD) yang dibutuhkan untuk menyimpan log sensor?
3. Bagaimana interaksi pengguna dengan sistem (Use Case)?

1.3 Tujuan

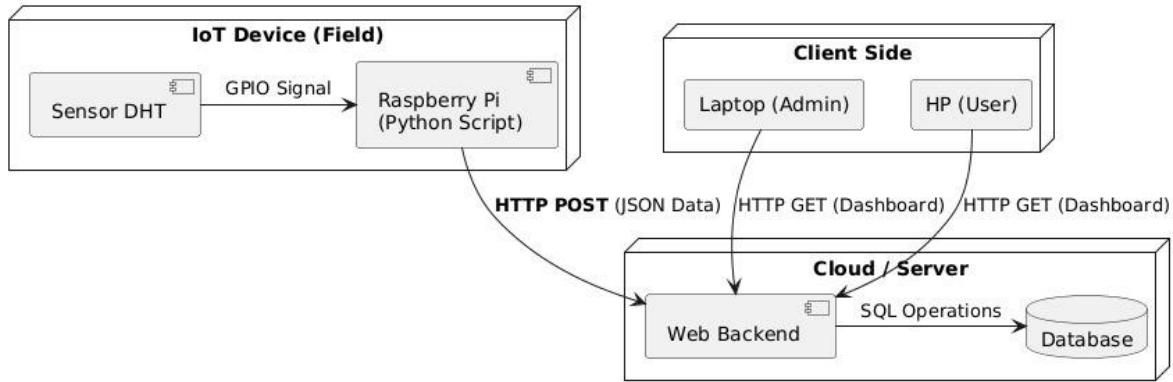
Tujuan dari laporan ini adalah mendokumentasikan perancangan sistem perangkat lunak (UML) dan implementasi aplikasi *backend* yang telah dibuat oleh Kelompok 1.

BAB 2: PERANCANGAN SISTEM (UML & DIAGRAM)

Pada bab ini dijelaskan desain sistem menggunakan 7 diagram utama yang telah dirancang.

2.1 Arsitektur Sistem

Gambar berikut menjelaskan topologi jaringan secara keseluruhan. Sistem terdiri dari *IoT Device* (Sensor DHT) yang mengirim sinyal GPIO ke mikrokontroler, kemudian data dikirim via HTTP POST ke *Cloud Server* (Node.js), dan disimpan ke Database. User mengakses data melalui Laptop atau HP.

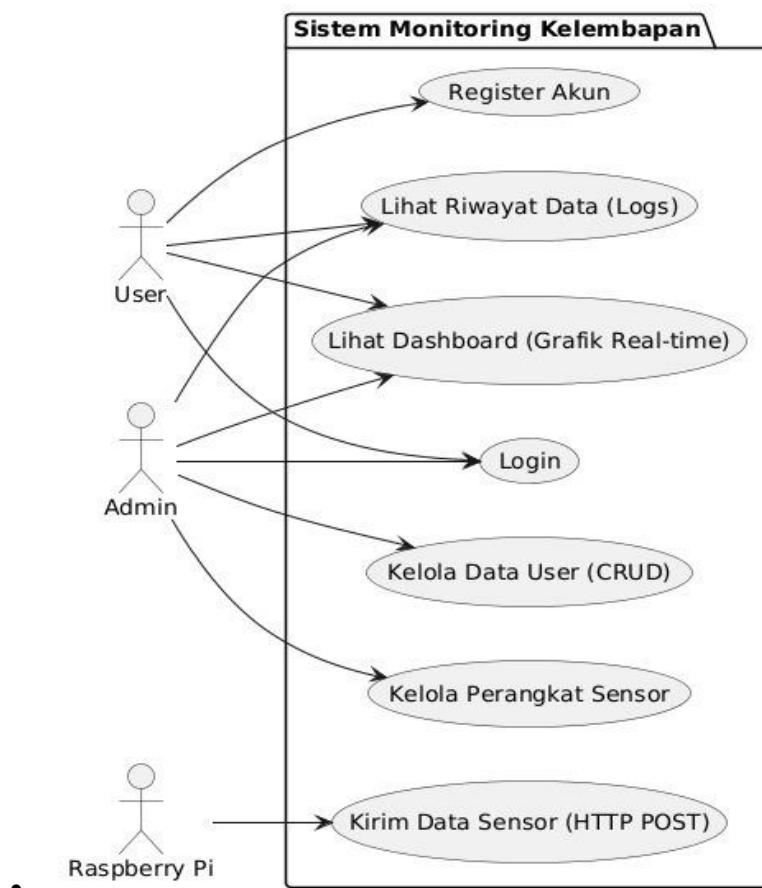


Gambar 2.1 Arsitektur Utama

2.2 Use Case Diagram

Diagram ini menggambarkan interaksi aktor dengan sistem.

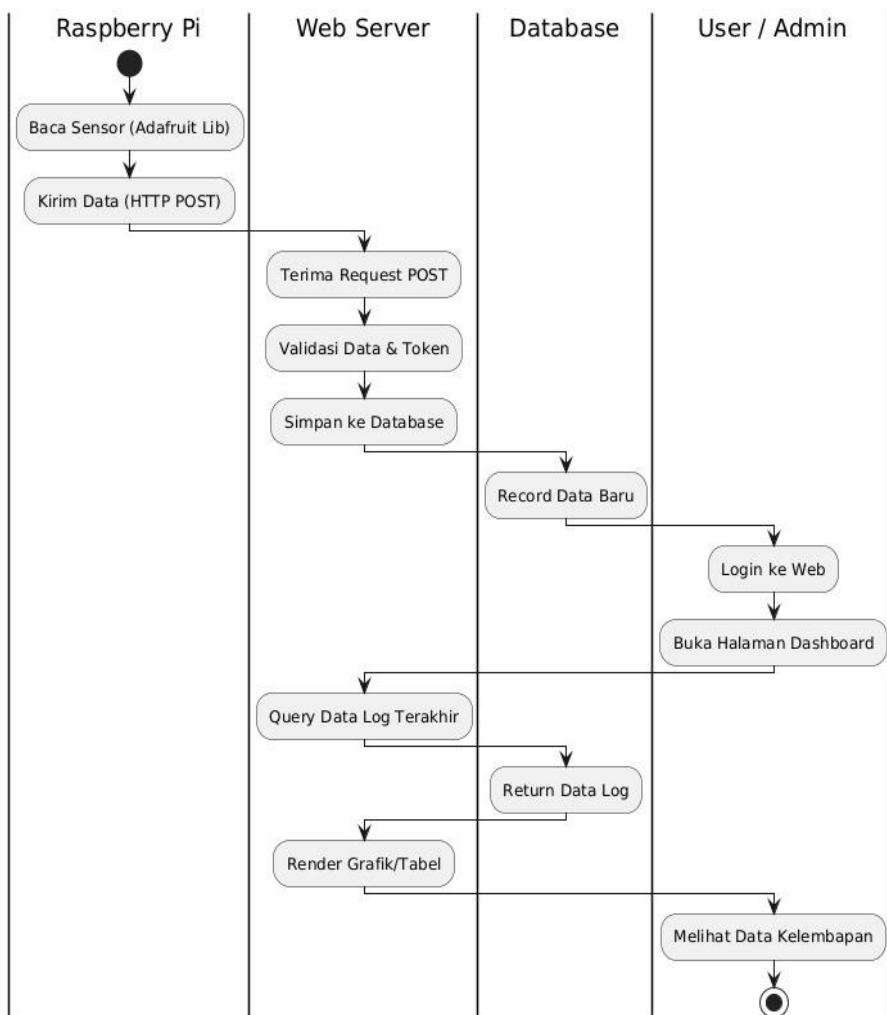
- **User:** Dapat melakukan Register, Login, melihat Dashboard, dan melihat Riwayat Data.
- **Admin:** Memiliki hak akses penuh termasuk mengelola data user dan perangkat sensor.
- **Raspberry Pi/Device:** Bertugas khusus untuk mengirim data sensor ke server.



Gambar 2.2 Use Case Diagram

2.3 Activity Diagram (Workflow Sistem)

Diagram ini menjelaskan alur kerja sistem secara *end-to-end* menggunakan *swimlane*. Dimulai dari Raspberry Pi membaca sensor, server menerima request POST, validasi token, hingga data tersimpan dan ditampilkan ke User dalam bentuk grafik.

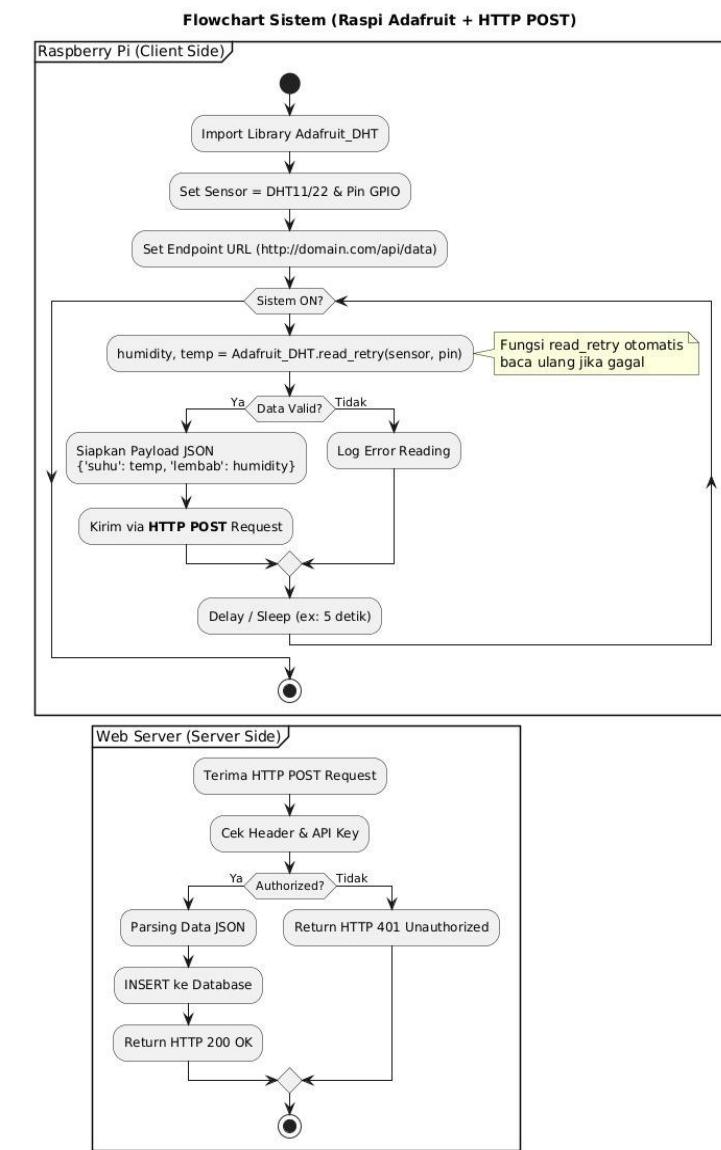


Gambar 2.3 Activity Diagram

2.4 Flowchart Logika Program

Diagram ini berfokus pada logika algoritma di sisi perangkat (Client) dan Server.

- **Sisi Client:** Terdapat logika *retry* (baca ulang) jika sensor gagal membaca, dan pengiriman JSON payload.
- **Sisi Server:** Terdapat pengecekan API Key. Jika *Authorized*, data disimpan (*INSERT*). Jika tidak, kembalikan error 401.

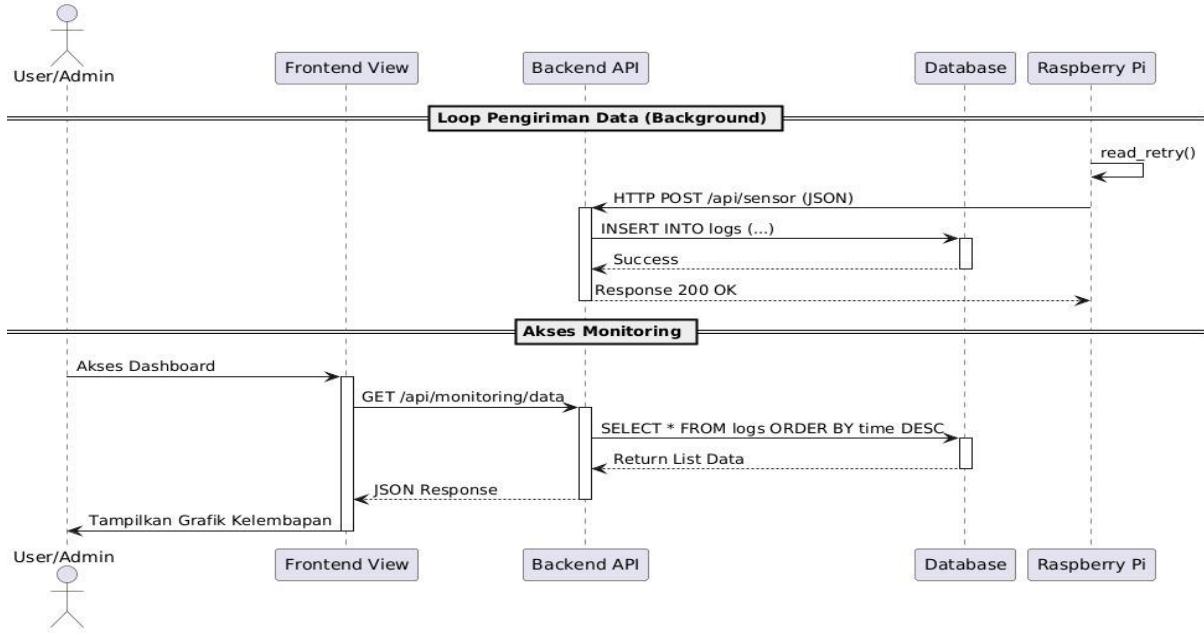


Gambar 2.4 Flowchart Sistem

2.5 Sequence Diagram

Diagram ini menjabarkan urutan waktu pengiriman pesan.

1. **Loop Pengiriman:** Device terus menerus mengirim HTTP POST ke API.
2. **Akses Monitoring:** Saat User membuka dashboard, Frontend meminta data (GET) ke Backend, Backend melakukan query SELECT ke Database, lalu data dikembalikan ke User.

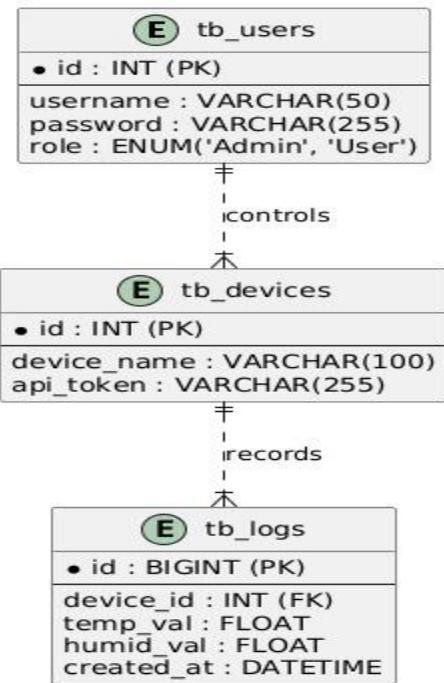


Gambar 2.5 Sequence Diagram

2.6 Class Diagram

Diagram ini merepresentasikan struktur kode program (OOP) di sisi Backend Node.js. Terdapat kelas utama seperti:

- **User:** Menangani login dan logout.
- **Device:** Menyimpan status perangkat dan lokasi.
- **SensorData:** Menyimpan atribut suhu, kelembaban, dan *timestamp*.

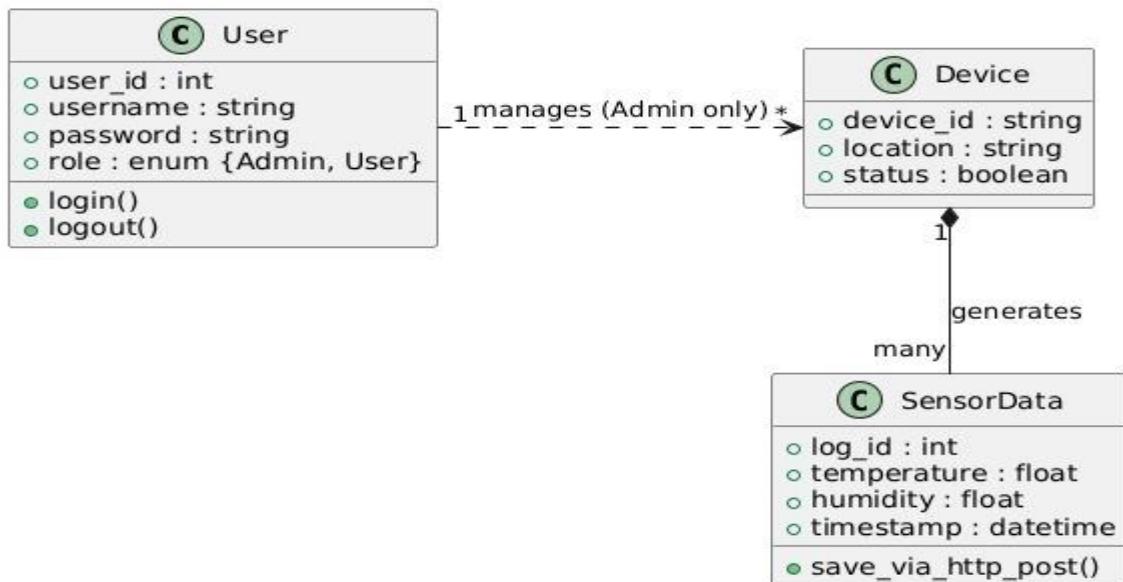


Gambar 2.6 Class Diagram

2.7 Entity Relationship Diagram (ERD)

Perancangan basis data menggunakan MySQL. Sistem ini menggunakan tabel relasional:

- tb_users: Menyimpan data pengguna.
- tb_devices: Menyimpan token API perangkat.
- tb_logs: Tabel transaksi utama yang menyimpan nilai suhu (temp_val) dan kelembaban (humid_val) yang berelasi dengan tabel device.



Gambar 2.7 Entity Relationship Diagram (ERD)