

Sistem Monitoring Kondisi AC untuk Menentukan Waktu Servis Menggunakan Algoritma K-Nearest Neighbor

I Made Mika Parwita

Institut Teknologi dan Bisnis STIKOM Bali

e-mail: mika_parwita@stikom-bali.ac.id

Diajukan: 24 Januari 2024; Direvisi: 13 Februari 2024; Diterima: 16 Februari 2024

Abstrak

Air Conditioner (AC) merupakan perangkat penting di wilayah beriklim panas untuk menjaga kenyamanan termal, mengurangi suhu, dan mengontrol kelembapan. Pengoperasiannya melibatkan proses pemindahan panas dari dalam ke luar ruangan melalui beberapa komponen utama, seperti evaporator, kompresor, kondensor, dan katup ekspansi. Agar AC bekerja optimal, perawatan rutin pada komponen seperti evaporator dan filter udara diperlukan. Perawatan berkala ini membantu menjaga efisiensi pendinginan serta kualitas udara dalam ruangan, meskipun terkadang perawatan dilakukan saat kondisi AC masih optimal, yang meningkatkan biaya. Penelitian ini mengembangkan alat monitoring berbasis sensor suhu dan kelembapan yang dipasang di dalam dan di luar ruangan, untuk memantau kinerja AC secara lebih akurat. Alat ini juga mengumpulkan data pengaturan AC seperti mode, suhu, dan fan speed. Data tersebut dikirim dan diolah di web server yang akan mengirimkan notifikasi jika terjadi anomali. Sistem ini juga menyalakan LED sebagai indikasi perawatan AC diperlukan, sehingga perawatan dilakukan secara tepat waktu. Sistem monitoring kondisi AC berhasil diimplementasikan dan diuji dengan baik. Sistem ini mampu mendeteksi anomali, memberikan notifikasi perawatan, serta mengirimkan data secara real-time. Implementasi teknologi Internet of Things (IoT) menggunakan algoritma K-Nearest Neighbor (KNN) memungkinkan pemantauan kondisi AC yang lebih efisien dan tepat waktu.

Kata kunci: Monitoring AC, Waktu servis, Rekomendasi, KNN.

Abstract

The Air Conditioner (AC) is a crucial device in hot climates to maintain thermal comfort, reduce temperature, and control humidity. Its operation involves transferring heat from inside to outside through several key components, such as the evaporator, compressor, condenser, and expansion valve. For optimal performance, regular maintenance on components like the evaporator and air filter is necessary. Routine maintenance helps maintain cooling efficiency and indoor air quality, although it is sometimes performed when the AC is still operating optimally, which increases costs. This study developed a monitoring tool using temperature and humidity sensors installed both indoors and outdoors to more accurately monitor AC performance. The tool also collects AC settings data such as mode, temperature, and fan speed. This data is sent and processed on a web server, which will send notifications in case of anomalies. The system also activates an LED as an indication that AC maintenance is needed, ensuring that maintenance is performed at the right time. The AC condition monitoring system was successfully implemented and tested. This system can detect anomalies, provide maintenance notifications, and transmit real-time data. The implementation of Internet of Things (IoT) technology using the K-Nearest Neighbor (KNN) algorithm enables more efficient and timely AC condition monitoring.

Keywords: AC monitoring, Maintenance timing, Recommendation, KNN.

1. Pendahuluan

Air Conditioner (AC) merupakan perangkat yang umum digunakan di wilayah beriklim panas untuk menjaga kenyamanan termal di dalam ruangan. Penggunaan AC menjadi solusi efektif untuk mengurangi suhu udara yang tinggi serta mengontrol kelembapan, sehingga menciptakan lingkungan yang lebih sejuk dan nyaman bagi penghuni. Penggunaan AC semakin meningkat seiring dengan berkembangnya urbanisasi dan meningkatnya kebutuhan akan kenyamanan termal di berbagai lingkungan, baik di rumah, perkantoran, maupun fasilitas umum [1]. Selain fungsinya dalam menurunkan suhu udara, AC juga berperan dalam mengurangi kelembapan yang sering kali tinggi di daerah beriklim tropis atau subtropis.

AC juga berperan dalam menjaga kualitas udara di dalam ruangan dengan menyaring debu dan partikel lainnya.

AC bekerja dengan prinsip dasar memindahkan panas dari dalam ruangan ke luar, sehingga menghasilkan udara sejuk di dalam ruangan. Proses ini melibatkan beberapa komponen utama, yaitu evaporator, kompresor, kondensor, dan katup ekspansi [2]. Udara panas dari dalam ruangan diserap oleh evaporator, di mana refrigeran (zat pendingin) menguap dan menyerap panas tersebut. Selanjutnya, refrigeran yang sudah berubah menjadi gas panas dikompresi oleh kompresor, lalu diarahkan ke kondensor di bagian luar ruangan, di mana panas dilepaskan ke lingkungan sekitar. Refrigeran yang sudah mendingin kembali ke bentuk cair dan siap untuk memulai siklus pendinginan lagi.

Agar AC dapat bekerja dengan optimal, penting untuk menjaga kebersihan komponen-komponennya, terutama evaporator dan filter udara [3]. Jika evaporator dan filter udara kotor, aliran udara akan terhambat, sehingga proses penyerapan panas tidak efisien. Hal ini bisa menyebabkan AC bekerja lebih keras untuk mencapai suhu yang diinginkan, yang pada akhirnya meningkatkan konsumsi energi dan mengurangi umur perangkat. Membersihkan AC secara berkala, termasuk mengganti atau membersihkan filter udara, membantu menjaga efisiensi pendinginan dan memperpanjang masa pakai perangkat. Perawatan rutin ini juga penting untuk memastikan kualitas udara di dalam ruangan tetap baik, mengurangi risiko penumpukan debu dan alergen yang dapat menyebabkan masalah kesehatan. Perawatan untuk membersihkan atau mencuci AC pada umumnya dilakukan menggunakan interval waktu, misalnya sebulan sekali. Permasalahan yang muncul dari hal tersebut adalah perawatan terkadang dilakukan padahal kondisi AC masih dapat beroperasi dengan optimal. Penggunaan interval waktu tidak menjamin perawatan dilakukan di saat yang tepat, sehingga hal ini berdampak pada tingginya biaya perawatan.

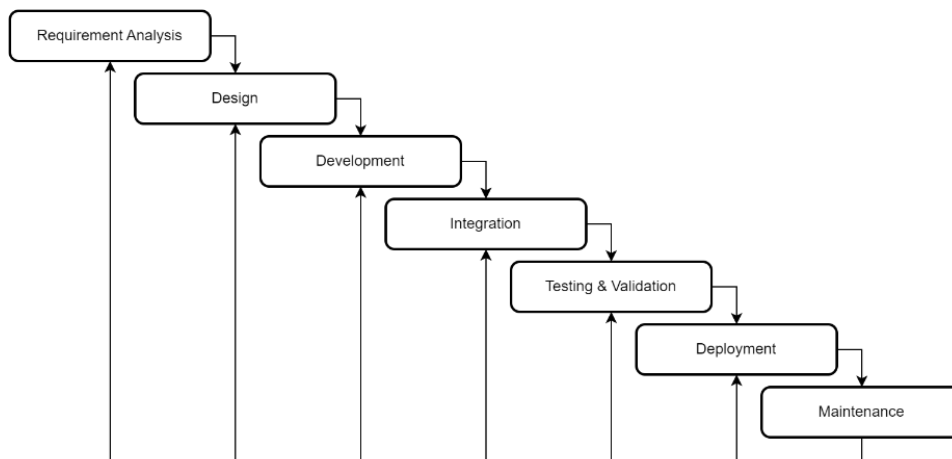
Beberapa penelitian sudah pernah dilakukan untuk mengatasi permasalahan tersebut. Aries Setiawan, dkk membuat sistem informasi pengingat untuk perawatan AC [4]. Penelitian tersebut bertujuan untuk mengingatkan pengguna untuk merawat AC secara rutin. Namun, penjadwalan masih dilakukan secara manual tanpa mempertimbangkan intensitas pemakaian AC. Ilham Fadli Surbakti, dkk menciptakan alat deteksi kesehatan AC menggunakan konsep Internet of Things (IoT) dengan menerapkan algoritma K-Nearest Neighbor (KNN) [5]. Penelitian tersebut dimulai dengan pengumpulan data menggunakan embedded system, kemudian menerapkan algoritma KNN untuk menilai apakah terdapat anomali dalam data suhu ruangan. Jika anomali terdeteksi, sistem akan membunyikan buzzer sebagai tanda bahwa AC sudah dalam keadaan tidak sehat. Alat yang diciptakan hanya menggunakan satu sensor suhu dan kelembapan (DHT11) yang dipasang di dalam ruangan tempat AC berada.

Penelitian ini merupakan pengembangan dari penelitian sebelumnya. Alat yang diciptakan pada penelitian ini menggunakan sensor suhu dan kelembapan yang dipasang di dalam dan di luar ruangan. Hal ini dilakukan karena suhu di dalam ruangan juga dipengaruhi oleh suhu di luar ruangan. Selain itu, alat juga mengumpulkan data pengaturan AC seperti mode AC, suhu AC, dan fan speed. Data tersebut dikirimkan, disimpan, dan diolah pada web server. Anomali dideteksi dengan menggunakan algoritma KNN [6]. Jika anomali ditemukan, web server akan mengirimkan notifikasi berupa email ke perangkat pengguna. Selain itu, web server juga memberikan respons ke embedded system jika anomali ditemukan. Embedded system akan menyalakan LED sebagai tanda bahwa anomali terjadi dan sudah waktunya untuk melakukan perawatan terhadap AC.

2. Metode Penelitian

2.1. Model Pengembangan Sistem

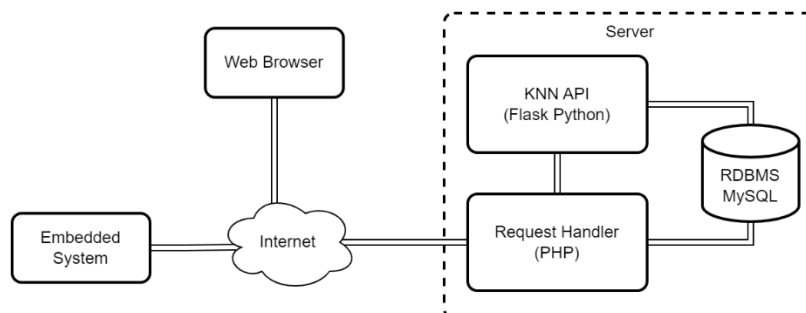
Penelitian ini mengadopsi model *Software Development Life Cycle* (SDLC) dengan beberapa penyesuaian untuk mengembangkan alat dan sistem [7], [8].



Gambar 1. Model pengembangan sistem.

Gambar 1 menunjukkan model pengembangan sistem yang dilakukan berdasarkan SDLC model Waterfall. Requirement analysis atau analisis kebutuhan dilakukan dengan mengidentifikasi kebutuhan fungsional dan non fungsional pada sistem. Tahap ini juga meliputi analisis perangkat yang dibutuhkan oleh sistem. Tahap design dilakukan dengan membuat rancangan sistem. Rancangan yang dibuat berupa rancangan *hardware* dan *software* yang nantinya diintegrasikan untuk membentuk satu kesatuan sistem secara utuh. *Hardware* dirancang menggunakan circuit diagram atau diagram komponen yang memperlihatkan rangkaian dari komponen *hardware* yang digunakan. Algoritma yang diterapkan pada *hardware* (embedded system) dinotasikan menggunakan flowchart. Sistem monitoring dan rekomendasi perawatan AC dibangun berbasis web yang digambarkan menggunakan Data Flow Diagram (DFD) untuk memperlihatkan modul-modul yang dibuat. Data model dinotasikan dalam bentuk schema diagram karena hanya terdapat satu entitas data.

Pada tahap development, sistem dibangun berdasarkan desain yang dibuat. Embedded system dirangkai berdasarkan circuit diagram. Program yang ditanamkan pada alat dibangun menggunakan bahasa pemrograman C++ berdasarkan notasi flowchart atau algoritma. Modul pada sistem monitoring dibangun berdasarkan DFD dengan basis data dibuat berdasarkan data model. Backend untuk penerapan KNN dibuat menggunakan bahasa pemrograman Python dengan library micro web framework Flask sebagai RESTful API. Pada web server, PHP digunakan untuk memproses request dari client. PHP berperan sebagai middleman yang menghubungkan antara client request dengan API yang memproses KNN. Data yang dikirimkan oleh embedded system disimpan ke dalam Relational Database Management System (RDBMS) MySQL. Data tersebut digunakan untuk proses monitoring dan untuk menentukan waktu perawatan yang tepat menggunakan KNN. Gambar 2 menunjukkan gambaran umum arsitektur sistem yang dibangun.



Gambar 2. Arsitektur Sistem.

Tahap integration dilakukan untuk mengintegrasikan *hardware* dan *software* yang dibangun agar dapat bekerja dengan baik. Tahap selanjutnya, yaitu testing and validation, dilakukan dengan menguji sistem secara fungsional. Hal ini dilakukan untuk memastikan sistem dapat bekerja dengan baik sesuai fungsinya. Pada tahap deployment, sistem dipasang dan dijalankan di lingkungan yang sebenarnya. Jika masih terdapat hal yang perlu diperbaiki dan dikembangkan, hal tersebut dilakukan pada tahap maintenance.

2.2. K-Nearest Neighbor

K-Nearest Neighbors (KNN) adalah salah satu algoritma yang sering digunakan dalam bidang pembelajaran mesin (machine learning) untuk tugas klasifikasi dan regresi. KNN termasuk dalam kategori algoritma pembelajaran yang berbasis instance (instance-based learning) atau dikenal juga sebagai lazy learning. Ini berarti bahwa KNN tidak melakukan proses pembelajaran eksplisit pada tahap training, melainkan menunggu hingga data baru (instance) datang dan kemudian melakukan klasifikasi atau prediksi berdasarkan data yang sudah ada [9].

Pada dasarnya, KNN bekerja dengan mencari sejumlah tetangga terdekat (nearest neighbors) dari data baru yang ingin diprediksi. Berikut adalah langkah-langkah utama dalam penerapan KNN [10]:

- **Pemilihan Nilai K :**
Parameter utama dalam KNN adalah jumlah tetangga terdekat yang akan digunakan, yang disebut dengan K . Misalnya, jika $K=3$, maka KNN akan mencari 3 tetangga terdekat dari data yang baru.
- **Menghitung Jarak:**
Setelah nilai K dipilih, langkah selanjutnya adalah menghitung jarak antara data baru dengan setiap instance dalam dataset pelatihan. Jarak yang digunakan adalah jarak Euclidean. Jarak Euclidean antara dua titik data $x_1 = (x_{11}, x_{12}, \dots, x_{1n})$ dan $x_2 = (x_{21}, x_{22}, \dots, x_{2n})$ dalam ruang berdimensi n dihitung dengan persamaan:

$$d(x_1, x_2) = \sqrt{\sum_{i=1}^n (x_{1i} - x_{2i})^2} \quad (1)$$

- **Menentukan Tetangga Terdekat:**
Setelah menghitung jarak, K instance dengan jarak terdekat dipilih sebagai tetangga terdekat.
- **Klasifikasi/Penentuan Output:**
Output ditentukan sebagai rata-rata dari nilai-nilai yang dihasilkan oleh K tetangga terdekat.

3. Hasil dan Pembahasan

3.1. Requirement Analysis

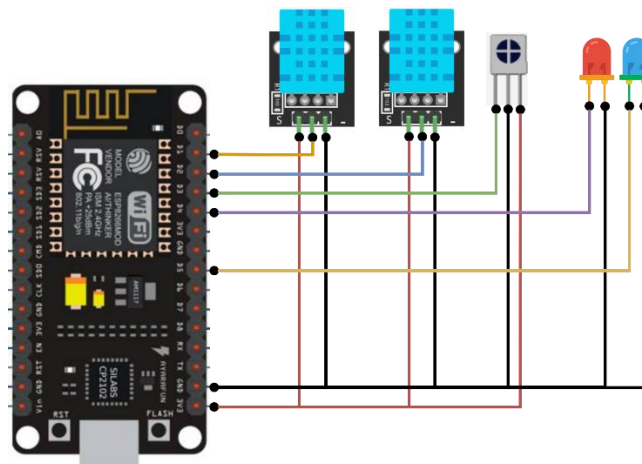
Berdasarkan metode pengembangan sistem yang telah dibahas, langkah pertama yang dilakukan pada penelitian ini adalah analisis kebutuhan. Berikut adalah hasil dari analisis kebutuhan fungsional, non-fungsional, spesifikasi *hardware*, dan spesifikasi *software* dari sistem yang dibangun:

1. **Kebutuhan Fungsional**
 - a. Alat dapat membaca suhu dan kelembapan di dalam dan di luar ruangan.
 - b. Alat dapat membaca pengaturan AC yang dikirimkan remote control ke AC dan memberikan umpan balik sebagai pertanda bahwa data pengaturan remote control berhasil dibaca.
 - c. Alat dapat mengirimkan data yang dibaca ke web server.
 - d. Alat dapat menerima respons jika terdapat rekomendasi perawatan AC dari web server.
 - e. Alat dapat memberikan tanda jika terdapat rekomendasi perawatan AC dari web server.
 - f. *Website* dapat memproses data dari alat dan menyimpannya ke RDBMS.
 - g. *Website* dapat melakukan request rekomendasi ke REST API dengan menyertakan data untuk diproses.
 - h. *Website* dapat mengirimkan notifikasi berupa email ke pengguna jika terdapat rekomendasi untuk merawat AC.
 - i. *Website* dapat menampilkan 10 data terakhir yang dikirimkan oleh alat dalam bentuk grafik secara realtime.
 - j. *Website* dapat menampilkan riwayat data yang berhasil disimpan.
 - k. REST API dapat memproses data yang dikirimkan melalui web server dan data yang terkumpul pada database, menggunakan metode KNN yang menghasilkan rekomendasi apakah AC sudah saatnya untuk dirawat atau belum.
2. **Kebutuhan Non-fungsional**
 - a. Alat dan *website* untuk proses monitoring harus dapat berkomunikasi dengan cepat sehingga data yang ditampilkan bersifat realtime.
 - b. REST API harus dapat memproses data relatif cepat.
 - c. Alat dirancang untuk tahan lama, dan tahan terhadap percikan air, paparan debu, dan jatuh.

3. Spesifikasi *Hardware*
 - a. Pemrosesan utama alat menggunakan NodeMCU/ESP8266.
 - b. Sensor suhu dan kelembapan menggunakan DHT11.
 - c. Indikator alat menggunakan LED 5mm.
 - d. IR receiver yang digunakan untuk membaca perintah dari remote control AC adalah HX1838.
4. Spesifikasi *Software*
 - a. Server menggunakan sistem operasi Linux Ubuntu.
 - b. Embedded system ditulis menggunakan Arduino IDE dengan bahasa C++.
 - c. *Website* dibuat menggunakan bahasa PHP pada back-end serta HTML, CSS, JavaScript pada front-end.
 - d. REST API dibangun menggunakan bahasa pemrograman Python dengan framework Flask.
 - e. Menggunakan Apache web server.
 - f. Menggunakan RDBMS MySQL.

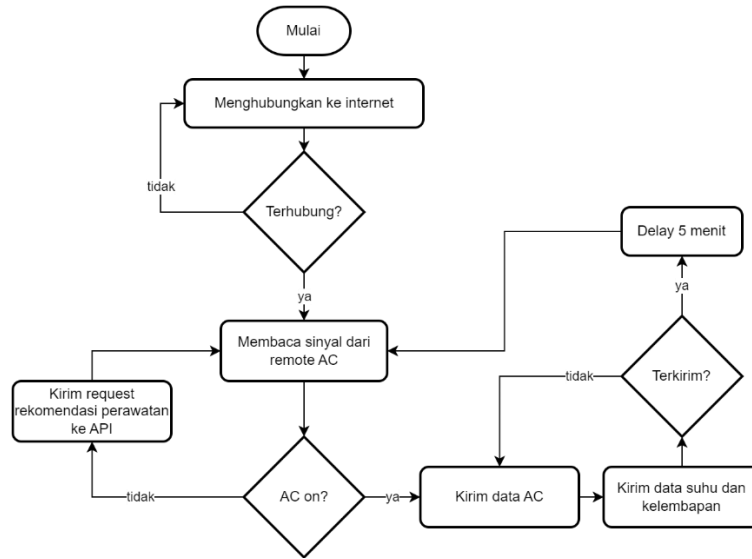
3.2. Design

Gambar 3 menunjukkan rancangan rangkaian dari alat yang dibuat. Pemrosesan utama menggunakan NodeMCU ESP8266. Sensor yang digunakan yaitu dua sensor DHT11 untuk mendapatkan suhu dan kelembapan di dalam dan di luar ruangan. Sensor DHT11 untuk pembacaan data di dalam ruangan dihubungkan ke pin D1, sedangkan sensor DHT11 untuk pembacaan data di luar ruangan dihubungkan ke pin D2 pada NodeMCU. Sensor IR receiver HX1838 dihubungkan ke pin D3 yang fungsinya adalah menangkap perintah pengaturan yang dikirimkan ke AC oleh remote control. Alat ini menggunakan dua LED berwarna merah dan biru. LED warna merah, yang dihubungkan ke pin D4, berfungsi sebagai indikator jika perawatan AC sudah direkomendasikan untuk dilakukan. LED berwarna biru, yang dihubungkan ke pin D5, berfungsi sebagai indikator bahwa IR receiver telah berhasil menangkap sinyal dari remote control.



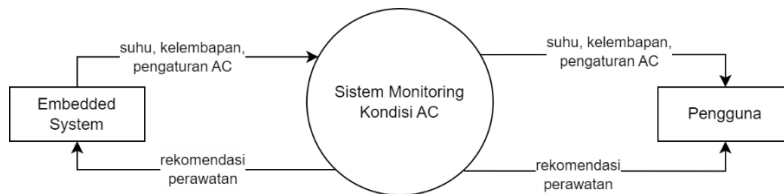
Gambar 3. Rangkaian komponen alat.

Data suhu dan kelembapan di dalam dan di luar ruangan hanya dikirimkan ketika AC dalam keadaan menyala. Ketika AC menyala, data suhu dan kelembapan dikirim ke server menggunakan interval waktu 5 menit. Ketika AC dimatikan, maka alat mengirimkan request rekomendasi perawatan untuk data yang sudah terkumpul. Gambar 4 menunjukkan rancangan algoritma secara umum pada alat yang dibuat.



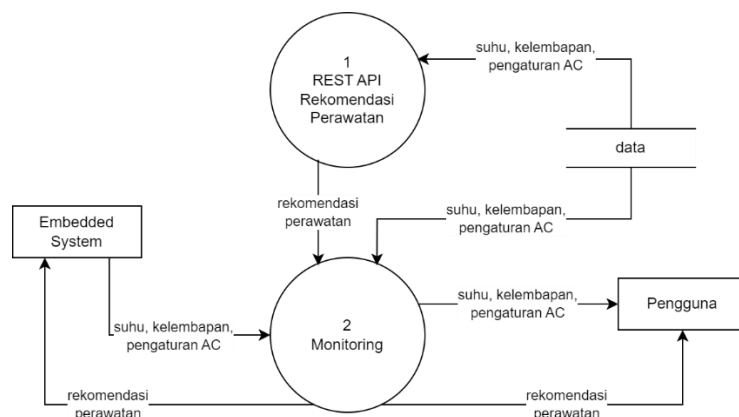
Gambar 4. Flowchart algoritma alat secara umum.

Gambar 5 menunjukkan DFD Level 0, atau diagram konteks dari sistem yang dibangun. Data suhu, kelembapan, dan pengaturan AC dikirim dari embedded system kemudian disimpan ke dalam database. Rekomendasi perawatan, jika ada, langsung dikirimkan ke embedded system dan pengguna melalui SMTP server (email) ketika rekomendasi perawatan dihasilkan oleh sistem.



Gambar 5. DFD Level 0.

Gambar 6 menunjukkan rancangan alur data dan modul yang ada pada sistem secara lebih detail. Sistem dibagi menjadi dua modul besar yaitu REST API yang berfungsi untuk memproses data menggunakan metode KNN sehingga menghasilkan rekomendasi perawatan. Modul monitoring digunakan sebagai interface oleh entitas untuk dapat terhubung dengan REST API dan juga berfungsi untuk menampilkan data dalam proses monitoring.



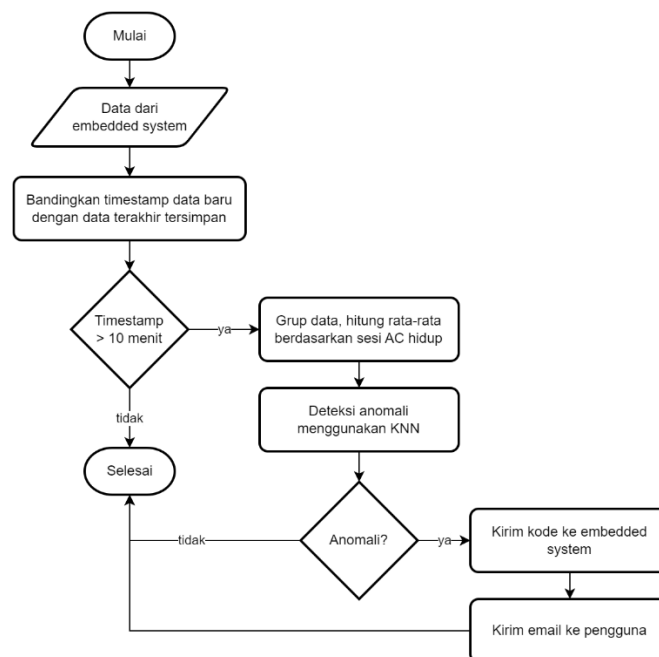
Gambar 6. DFD Level 1.

Berdasarkan Gambar 6, terdapat satu tabel yang digunakan untuk menyimpan data yang dikirim oleh embedded system. Gambar 7 menunjukkan atribut yang ada pada tabel data.

data	
PK	Id
	indoor_temperature
	indoor_humidity
	outdoor_temperature
	outdoor_humidity
	ac_power
	ac_mode
	ac_temperature
	ac_fan
	datetime_captured

Gambar 7. Struktur tabel data.

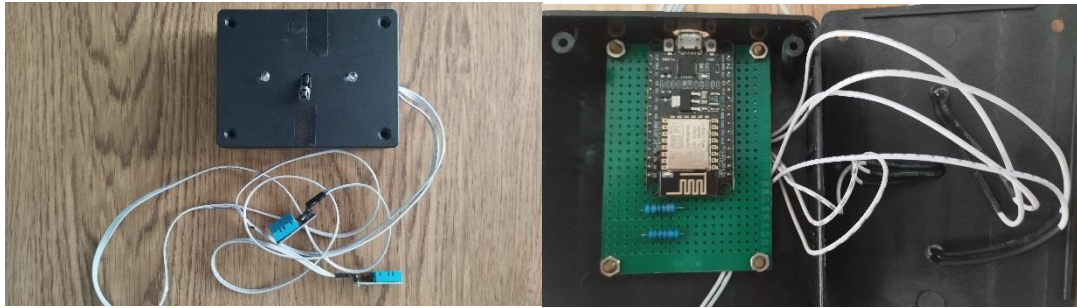
Gambar 8 menunjukkan alur proses yang terjadi pada REST API. Terdapat satu end-point yaitu permintaan rekomendasi perawatan. Namun, tidak setiap request yang masuk menghasilkan rekomendasi. Data dikelompokkan terlebih dahulu berdasarkan sesi AC hidup. Satu sesi data atau satu grup data yaitu mulai dari AC menyala hingga AC mati. Pada satu grup data ini, dihitung rata-rata untuk masing-masing kolom. Hal ini dilakukan karena pada saat AC baru hidup, belum ada perubahan yang signifikan terhadap suhu dan kelembapan dan dapat menyebabkan hasil rekomendasi bernilai false positive.



Gambar 8. Alur proses deteksi anomali pada REST API.

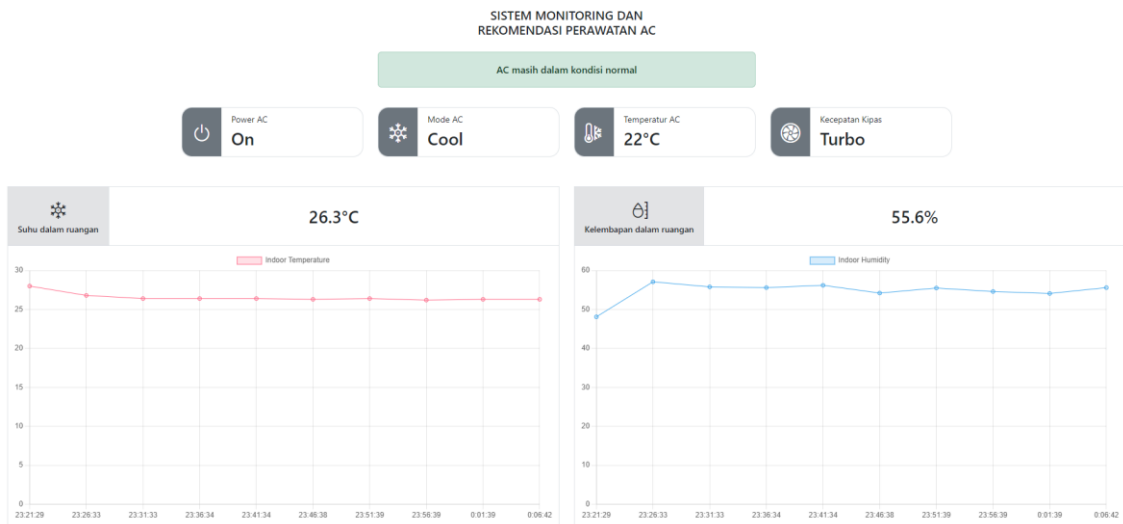
3.3. Development

Proses development pada penelitian ini dimulai dengan mengembangkan embedded system sebagai sumber data yang diolah. Pengembangan sistem dilakukan menggunakan Arduino IDE dengan bahasa pemrograman C++. Library utama yang digunakan adalah DHT dan IRremoteESP8266. Pada kasus penelitian ini, tipe AC yang digunakan belum didukung oleh library IRremoteESP8266, namun kode HEX dari IR berhasil dibaca. Tipe AC yang digunakan pada penelitian ini adalah AC dengan merk Samsung tipe AR05JRFSVURXSE. Sinyal IR dari remote control yang diterjemahkan oleh library tersebut terdiri dari 44 HEX karakter. Data power AC dapat ditemukan pada karakter ke-14, data mode AC pada karakter ke-40, data temperatur AC pada karakter ke-38, dan data kecepatan kipas pada karakter ke-41. Gambar 9 menunjukkan hasil implementasi embedded system.



Gambar 9. Hasil implementasi alat.

Tahap selanjutnya yaitu mengembangkan aplikasi berbasis *website* untuk menerima dan mengolah data yang dikirimkan oleh embedded system. Web dikembangkan menggunakan bahasa PHP pada sisi back-end. Sisi front-end dikembangkan menggunakan bahasa HTML, CSS, dan JavaScript. Gambar 10 menunjukkan tampilan dari sistem monitoring yang dibuat.



Gambar 10. Monitoring data.

Gambar 11 menunjukkan tampilan dari riwayat pencatatan data yang dilakukan. Terdapat total 2.664 data yang terkumpul pada saat penelitian ini dibuat.

Show 10 entries Search:

ID	Suhu In	Kelembapan In	Suhu Out	Kelembapan Out	Mode AC	Suhu AC	Kipas AC	Waktu
1	28	25.6	48.1	56.7	Cool	23	Turbo	2023-08-13 23:21:29
2	26.8	26.4	57.1	79.9	Cool	23	Turbo	2023-08-13 23:26:33
3	26.4	26.1	55.8	81.1	Cool	23	Turbo	2023-08-13 23:31:33
4	26.4	26.5	55.6	79.8	Cool	23	Turbo	2023-08-13 23:36:34
5	26.4	25.9	56.2	81.1	Cool	23	Turbo	2023-08-13 23:41:34
6	26.3	26.1	54.2	80.5	Cool	23	Turbo	2023-08-13 23:46:38
7	26.4	26.1	55.5	80.5	Cool	23	Turbo	2023-08-13 23:51:39
8	26.2	25.8	54.6	81.2	Cool	23	Turbo	2023-08-13 23:56:39
9	26.3	26.3	54.1	80	Cool	23	Turbo	2023-08-14 00:01:39
10	26.3	26	55.6	81.1	Cool	23	Turbo	2023-08-14 00:06:42

Showing 1 to 10 of 2,664 entries Previous 1 2 3 4 5 ... 267 Next

Gambar 11. Riwayat pencatatan data.

REST API yang berfungsi untuk memberikan rekomendasi perawatan AC dibangun menggunakan bahasa Python dengan framework Flask. Ketika permintaan rekomendasi dikirimkan ke web server oleh embedded system, PHP langsung mengirimkan request ke API melalui CURL. Berdasarkan data yang dikumpulkan, pengaturan mode AC dan kipas AC sama untuk semua data. Oleh karena itu, pada kasus ini, data yang digunakan adalah data suhu (dalam dan luar ruangan), kelembapan (dalam dan luar ruangan), dan suhu AC. Data yang tersimpan dikelompokkan ke dalam grup. Satu grup data adalah pada saat AC dihidupkan hingga AC dimatikan. Data pada grup tersebut dihitung rata-ratanya per kolom untuk dihitung jaraknya menggunakan Euclidean Distance. Berdasarkan hasil eksperimen, ditentukan threshold untuk mendeteksi anomali yaitu $rata-rata + 2 \times standar\ deviasi$ per baris data. Jika terdapat rekomendasi perawatan terhadap data terakhir, maka response dikirimkan ke web server untuk diteruskan ke embedded system dan notifikasi berupa email dikirimkan ke pengguna. Gambar 12 menunjukkan contoh hasil perhitungan dari data yang telah terkumpul.

	group	tempIn	tempOut	humIn	humOut	acTemp	anomaly
0	1	25.681373	25.035294	55.957843	79.097059	23.000000	False
1	3	26.997368	32.707895	60.555263	59.057895	23.000000	False
2	5	26.497857	26.782857	57.416429	83.817857	23.000000	False
3	7	27.014286	31.600000	70.850000	66.921429	23.000000	False
4	9	26.218571	26.471429	56.470714	82.781429	21.785714	False
5	11	26.014754	25.724590	55.043443	83.014754	22.000000	False
6	13	25.821951	26.675610	61.621951	80.007317	22.000000	False
7	15	25.212879	25.040909	59.698485	87.335606	22.000000	False
8	17	26.613333	31.113333	65.940000	65.000000	22.000000	False
9	19	25.925342	25.750685	61.302055	80.222603	22.082192	False
10	21	26.160465	26.022481	57.866667	80.134884	21.007752	False
11	23	26.691364	28.199545	59.640909	75.347273	22.000000	False
12	25	27.538710	32.183871	60.232258	60.212903	22.000000	False
13	27	26.295200	26.104000	55.285600	76.721600	20.000000	False
14	29	27.244118	31.452941	57.538235	56.261765	22.000000	False
15	31	26.342282	27.091946	58.031544	75.877181	23.013423	False

Gambar 12. Contoh hasil perhitungan KNN.

3.4. Integration

Dalam fase integrasi embedded system, beberapa aktivitas penting dilakukan untuk memastikan bahwa komponen perangkat keras dan perangkat lunak berfungsi dengan baik bersama-sama. Pertama, dilakukan integrasi komponen, di mana perangkat lunak dihubungkan dengan perangkat keras untuk memastikan komunikasi dan fungsionalitas yang tepat. Selanjutnya, dilakukan pengujian integrasi sistem secara menyeluruh untuk memeriksa bagaimana semua komponen bekerja secara bersamaan dan memastikan bahwa sistem memenuhi spesifikasi yang diharapkan. Pada tahap ini, masalah yang ditemukan selama integrasi, seperti kesalahan komunikasi atau masalah waktu, diidentifikasi dan diperbaiki.

3.5. Testing and Validation

Tabel 1 menunjukkan hasil pengujian fungsional yang dilakukan pada sistem monitoring dan rekomendasi perawatan AC yang dibangun.

Tabel 1. Black-box testing.

Fitur	Input	Ekspektasi Output	Keterangan
Membaca suhu dan kelembapan di dalam dan di luar ruangan	Sensor suhu dan kelembapan	Data suhu dan kelembapan yang akurat untuk dalam dan luar ruangan	Sesuai
Membaca pengaturan AC dari remote control dan memberikan umpan balik	Pengaturan AC dari remote control	Umpan balik bahwa data pengaturan remote berhasil dibaca	Sesuai
Mengirimkan data ke web server	Data suhu, kelembapan, dan pengaturan AC	Data dikirimkan dengan benar ke web server	Sesuai
Menerima respons rekomendasi perawatan AC dari web server	Permintaan rekomendasi ke web server	Respons yang mengandung rekomendasi perawatan AC jika ada	Sesuai
Memberikan tanda jika terdapat rekomendasi perawatan AC dari web server	Respons rekomendasi dari web server	Tanda atau notifikasi bahwa ada rekomendasi perawatan AC	Sesuai
Memproses dan menyimpan data dari alat ke RDBMS	Data yang dikirimkan dari alat	Data disimpan dengan benar ke dalam RDBMS	Sesuai
Request rekomendasi ke REST API dengan data	Data dikirimkan dari website ke REST API	Respons rekomendasi dari REST API	Sesuai
Mengirimkan notifikasi email kepada pengguna jika ada rekomendasi perawatan AC	Rekomendasi perawatan AC dari REST API	Email notifikasi dikirim ke pengguna	Sesuai

Fitur	Input	Ekspektasi Output	Keterangan
Menampilkan 10 data terakhir dalam bentuk grafik secara real-time	Data terakhir dari alat	Grafik yang menampilkan 10 data terakhir secara real-time	Sesuai
Menampilkan riwayat data yang berhasil disimpan	Data yang tersimpan dalam RDBMS	Daftar riwayat data yang berhasil disimpan ditampilkan	Sesuai
Memproses data dengan metode KNN untuk rekomendasi perawatan AC	Data dari web server dan database	Rekomendasi apakah AC perlu dirawat atau tidak berdasarkan metode KNN	Sesuai

REST API diuji menggunakan lingkungan yang dimodifikasi yaitu AC dihidupkan dengan pintu terbuka. Pada saat hal ini dilakukan berulang-ulang, sistem dapat mendeteksi bahwa anomali terjadi. LED merah pada embedded system berhasil menyala dan notifikasi email dikirimkan ke email pengguna.

3.6. Deployment

Pada fase ini, sistem diimplementasikan di lingkungan sebenarnya. Embedded system diletakkan di dekat AC sehingga alat juga mampu menerima sinyal dari remote AC. *Website* dan API dihosting pada Virtual Private Server (VPS) yang sama. Server yang digunakan menggunakan sistem operasi Linux Ubuntu.

3.7. Maintenance

Pada fase ini, dilakukan monitoring terhadap sistem yang telah berjalan di lingkungan sebenarnya. Pada saat laporan dibuat, belum ditemukan kesalahan lanjut yang terjadi pada sistem.

4. Kesimpulan

Sistem monitoring kondisi AC berbasis IoT yang diimplementasikan menggunakan algoritma K-Nearest Neighbor (KNN) terbukti mampu mendeteksi anomali pada kinerja AC secara *real-time*. Sistem ini secara efisien mengumpulkan dan mengirimkan data yang berkaitan dengan kondisi AC, seperti suhu, kelembaban, dan durasi penggunaan, sehingga memungkinkan pemantauan yang lebih cermat dan prediktif. Dengan kemampuan ini, pengguna dapat mengantisipasi potensi kerusakan dan melakukan perawatan yang lebih tepat waktu.

Selain itu, sistem ini memberikan notifikasi kepada pengguna jika terdeteksi adanya kebutuhan perawatan atau kerusakan pada AC. Notifikasi ini mempermudah pengguna dalam mengelola perangkat AC dan meningkatkan umur pakai serta efisiensi kinerja perangkat. Dengan teknologi IoT yang canggih, sistem monitoring ini mendukung penghematan energi dan biaya melalui pemeliharaan yang lebih efisien dan terukur.

Daftar Pustaka

- [1] Y. Arnas, K. G. S. M. Ismail, Z. Kurniawati, B. Kurnianto, I. H. Wibowo, and N. Kalbuana, "Pelatihan perawatan/service AC untuk masyarakat sekitar Politeknik Penerbangan Indonesia Curug," *Penamas: Journal of Community Service*, vol. 1, no. 2, pp. 90–99, 2021.
- [2] A. Rachman, L. Nesti, and A. Yanto, "Study of the VRF/VRV Air Conditioning System in the Hospital: A Review of Economic and Environmental Aspects," *Jurnal Teknik Mesin*, vol. 12, no. 1, pp. 17–25, 2022.
- [3] K. Y. Utomo, A. Sugiharto, S. Wismanto, H. P. Cahyono, A. R. Madaskala, and others, "Analisis Sebelum dan Sesudah Dilakukan Proses Pembersihan Terhadap Performa AC Tipe Split Wall Kapasitas 1 1/2 PK," *Jurnal Teknologi Kedirgantaraan*, vol. 6, no. 1, 2021.
- [4] A. Setiawan, S. Wibowo, and I. Farida, "Optimalisasi Umur Pemakaian AC Melalui Sistem Informasi Reminder Perawatan," *Edumatic: Jurnal Pendidikan Informatika*, vol. 5, no. 1, pp. 118–127, 2021.
- [5] I. F. Surbakti, A. G. Putrada, and C. W. Wijiutomo, "Alat Deteksi Kesehatan AC Berbasis Internet of Things Menggunakan Algoritma KNearest Neighbor," *eProceedings of Engineering*, vol. 5, no. 3, 2018.
- [6] L. Wang, J. Li, U. A. Bhatti, and Y. Liu, "Anomaly detection in wireless sensor networks based on KNN," in *Artificial Intelligence and Security: 5th International Conference, ICAIS 2019, New York, NY, USA, July 26–28, 2019, Proceedings, Part III 5*, 2019, pp. 632–643.
- [7] N. R. Maulana and R. Djutalov, "Implementasi Metode SDLC Untuk Pembuatan Sistem Penguncian Pintu Menggunakan Fingerprint Dengan Modul NodeMCU Esp8266 Berbasis Internet Of Things," *LOGIC: Jurnal Ilmu Komputer dan Pendidikan*, vol. 1, no. 1, pp. 40–49, 2022.

-
- [8] M. Alfawair, “Internet-of-Things: A System Development Life Cycle (SDLC),” *J. Theoretical Appl. Inform. Technol.*, vol. 100, no. 5, pp. 1643–1653, 2022.
- [9] Z. Panjaitan, E. Elfitriani, W. R. Maya, and C. D. Siahaan, “Aplikasi Sistem Pakar Untuk Mendiagnosa Lebih Dini Penyakit Kolera Pada Anak Menggunakan Metode K-Nearest Neighbor (KNN),” *Journal Of Science And Social Research*, vol. 5, no. 2, pp. 220–229, 2022.
- [10] F. T. Admojo and others, “Klasifikasi Aroma Alkohol Menggunakan Metode KNN,” *Indonesian Journal of Data and Science*, vol. 1, no. 2, pp. 34–38, 2020.