

BAB III

METODOLOGI PENELITIAN

3.1 Pendekatan Penelitian

Penelitian ini menggunakan pendekatan pengembangan prototipe sistem berbasis *Internet of Things (IoT)*. Pengembangan dilaksanakan dengan model Waterfall, sedangkan evaluasi dilakukan melalui pengujian fungsional, pengujian pembacaan sensor, pengujian waktu respons, pengujian konektivitas, dan pengujian data logging. Pemisahan jenis pengujian diperlukan karena Black-box Testing digunakan untuk memeriksa kesesuaian fungsi, sedangkan ketelitian sensor dan reliabilitas komunikasi membutuhkan pengukuran kuantitatif tersendiri.

1. Mengontrol perangkat elektronik seperti lampu dan kipas angin secara otomatis, baik melalui perintah suara (*voice command*) menggunakan *Google Assistant* maupun berdasarkan kondisi lingkungan dan jadwal yang telah ditetapkan.
2. Memantau kondisi lingkungan secara *real-time*, termasuk suhu, kelembapan, kualitas udara, dan tekanan udara, menggunakan sensor *DHT22*, *MQ-135*, dan *BME280* yang terhubung ke *platform IoT cloud Blynk*.
3. Memberikan notifikasi dan peringatan kesehatan secara aktual kepada pengguna jika kondisi udara mencapai ambang batas tertentu, sebagai langkah preventif untuk menjaga kesehatan penghuni ruangan.

Untuk memastikan pengembangan sistem berjalan secara terstruktur dan terkontrol, penelitian ini menggunakan metode *Waterfall* sebagai kerangka kerja pengembangan. Model *Waterfall* dipilih karena sifatnya yang linear dan berurutan, sehingga setiap tahapan mulai dari analisis kebutuhan, perancangan sistem, implementasi, pengujian, hingga pemeliharaan dapat dilaksanakan secara sistematis dan mudah dievaluasi. Pendekatan linear ini sangat sesuai dengan pengembangan sistem *IoT* pada skala ruangan, di mana output dari satu tahap menjadi input bagi tahap berikutnya.

Selain itu, untuk memastikan bahwa semua fungsi sistem bekerja sesuai spesifikasi, penelitian ini menerapkan Black-box Testing sebagai metode pengujian utama. Metode ini fokus pada verifikasi input dan output sistem, tanpa memeriksa struktur internal kode program, sehingga pengujian dapat difokuskan pada performa sensor, aktuator, dan integrasi dengan platform *cloud*.

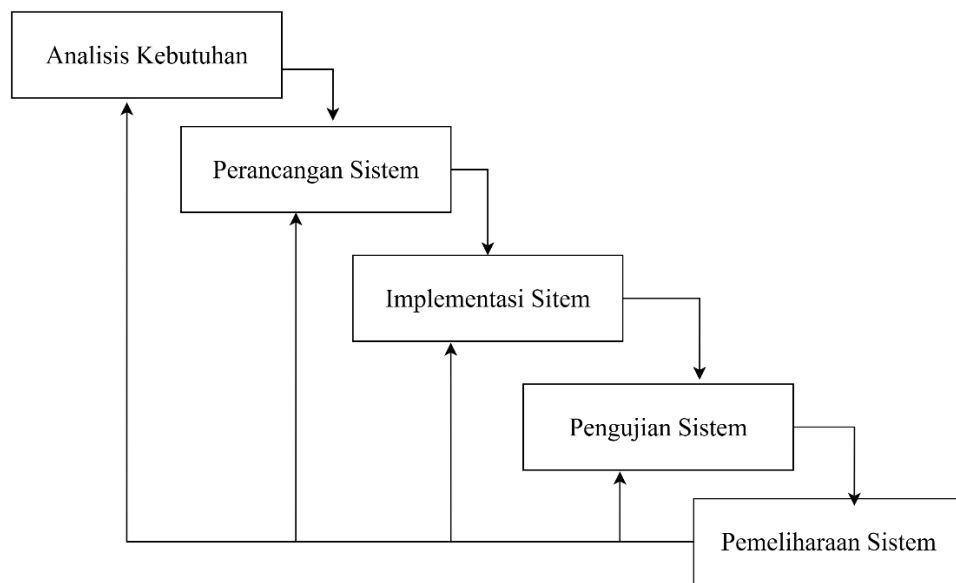
Dengan metode ini, pengujian sistem mencakup:

1. Respons perangkat elektronik terhadap perintah suara dan jadwal otomatis.
2. Keakuratan pembacaan *sensor* lingkungan.
3. Notifikasi dan peringatan yang dikirim ke pengguna melalui aplikasi *Blynk*.

Penggunaan kombinasi *Waterfall* dan Black-box pada penelitian ini memungkinkan pengembangan sistem *smart room* yang terstruktur, terdokumentasi dengan baik, dan dapat diuji secara fungsional sebelum

diterapkan pada ruangan nyata. Dengan pendekatan ini, diharapkan sistem yang dibangun tidak hanya memenuhi kebutuhan operasional, tetapi juga memiliki reliabilitas tinggi dalam menjaga kualitas udara dan kenyamanan pengguna.

3.2 Tahapan Penelitian (*Waterfall*)



Gambar 3. 1 Tahapan Penelitian

Penelitian ini menggunakan model *Waterfall* sebagai kerangka kerja pengembangan sistem *smart room* berbasis *IoT*. Model *Waterfall* dipilih karena sifatnya yang linear dan berurutan, sehingga setiap tahapan pengembangan dilakukan secara sistematis, dengan *output* dari satu tahap menjadi *input* bagi tahap berikutnya .

Tahapan penelitian yang dilakukan meliputi:

3.2.1 Analisis Kebutuhan

Tahap pertama dalam model *Waterfall* adalah analisis kebutuhan sistem, Pada tahap ini dilakukan:

1. Identifikasi masalah yang ada di lingkungan ruangan, seperti kurangnya integrasi kontrol perangkat elektronik dengan pemantauan kualitas udara.
2. Penentuan kebutuhan pengguna, yaitu kemampuan mengontrol lampu dan kipas secara otomatis melalui perintah suara dan jadwal, serta memantau suhu, kelembapan, kualitas udara, dan tekanan udara secara *real-time*.
3. Penentuan perangkat keras dan perangkat lunak yang akan digunakan, termasuk *ESP32*, *sensor DHT22*, *MQ-135*, *BME280*, *relay*, modul *LCD*, *platform Blynk*, dan *Google Assistant*.

Hasil dari tahap analisis kebutuhan ini menjadi dasar dalam merancang sistem, memastikan semua kebutuhan fungsional dan non-fungsional tercakup sebelum masuk ke tahap desain.

3.2.2 Perancangan Sistem

Tahap kedua adalah perancangan sistem, yang fokus pada menyusun arsitektur dan alur kerja sistem:

1. Penyusunan diagram blok sistem dan flowchart untuk menggambarkan hubungan antar komponen, alur pembacaan sensor, proses kendali perangkat, pengiriman data ke *cloud*, serta keluaran sistem.

2. Perancangan perangkat keras yang meliputi pemilihan komponen, penyusunan koneksi sensor *DHT22*, *MQ-135*, *BME280*, *relay*, *LCD 20×4*, catu daya, serta pemetaan pin pada *ESP32*.
3. Perancangan perangkat lunak yang meliputi alur pembacaan sensor, koneksi Wi-Fi dan *Blynk*, aturan otomatisasi, notifikasi, kontrol lampu dan kipas, serta pengiriman data ke *Google Sheets*.
4. Perancangan media penyimpanan data sensor menggunakan *spreadsheet Google Sheets* yang dihubungkan melalui *Google Apps Script*. Data yang disimpan dibatasi pada *Timestamp*, Suhu, Kelembapan, Tekanan, dan *PPM*.

Tahap ini memastikan rancangan sistem siap diimplementasikan sesuai kebutuhan yang telah dianalisis sebelumnya.

3.2.3 Implementasi sistem

Tahap ketiga adalah implementasi sistem:

1. Memprogram *ESP32* agar mampu membaca data sensor *DHT22*, *MQ-135*, dan *BME280*, lalu mengirim data ke *platform cloud Blynk*.
2. Mengonfigurasi *Google Assistant* atau *Google Home*, notifikasi *Android*, dan *MacroDroid*. *MacroDroid* mencocokkan kata kunci perintah dan mengirimkan *HTTP*

GET menuju *Blynk External API* untuk memperbarui virtual pin kendali lampu dan kipas.

3. Mengintegrasikan modul *relay* dan *LCD 20×4* agar perangkat dapat dikontrol dan data sensor ditampilkan secara lokal maupun di aplikasi.

Tahap ini menjadikan rancangan abstrak menjadi sistem yang dapat diuji secara nyata .

3.2.4 Pengujian Sistem (*Blackbox Testing*)

Tahap keempat adalah pengujian sistem menggunakan metode Black-box Testing. Metode ini menekankan pengujian fungsi input-output tanpa meninjau struktur internal kode program. Pengujian dilakukan pada semua komponen sistem:

1. Lampu menyala dan mati sesuai perintah suara dari *Google Assistant*.
2. Lampu dapat dikendalikan secara manual melalui tombol pada aplikasi *Blynk*.
3. Kipas dapat dikendalikan secara manual melalui tombol pada aplikasi *Blynk*.
4. Kipas menyala otomatis ketika suhu ruangan melewati ambang batas.
5. Sensor *MQ-135* menampilkan status kualitas udara dan mengirimkan notifikasi ketika nilai *PPM* melewati ambang batas.

6. Sensor *DHT22* dan *BME280* memberikan pembacaan suhu, kelembapan, dan tekanan udara secara *real-time* ke aplikasi *Blynk* dan modul *LCD*.
7. *Datastream Blynk* menampilkan data sensor sesuai *virtual pin* yang telah dirancang.
8. Data sensor berhasil dikirim dan disimpan ke *Google Sheets* melalui *Google Apps Script*.
9. Jadwal otomatis lampu berjalan sesuai waktu yang ditentukan, yaitu lampu *ON* pada malam hari dan *OFF* pada pagi hari.
10. Sistem mampu melakukan koneksi ulang ketika jaringan Wi-Fi terputus sementara.

Hasil pengujian digunakan untuk menilai reliabilitas sistem dan memastikan seluruh fungsi berjalan sesuai spesifikasi.

3.2.5 Pemeliharaan Sistem

Tahap terakhir adalah pemeliharaan dan evaluasi sistem:

1. Memperbaiki *bug* dan menyempurnakan fungsi sistem berdasarkan hasil pengujian *Blackbox*.
2. Memastikan integrasi perangkat keras dan perangkat lunak berjalan stabil.
3. Menyusun dokumentasi sistem dan panduan pengguna agar memudahkan operasional dan pemeliharaan di masa depan.

3.3 Metode Pengumpulan Data

Dalam penelitian ini, data yang dikumpulkan terdiri dari data primer dan data sekunder, yang digunakan untuk mendukung pengembangan dan pengujian sistem *smart room* berbasis *IoT*.

3.3.1 Data Primer

Data primer diperoleh secara langsung dari sistem yang dikembangkan, yaitu melalui *sensor* dan perangkat *IoT* yang terpasang di ruangan tertutup. Komponen dan jenis data yang dikumpulkan meliputi:

1. *Sensor DHT22*

mengukur suhu dan kelembapan ruangan secara *real-time*.

Data ini digunakan untuk memantau kondisi lingkungan fisik dan sebagai dasar otomatisasi perangkat seperti kipas angin.

2. *Sensor MQ-135*

menghasilkan nilai estimasi PPM sebagai indikator relatif perubahan kondisi udara. MQ-135 memiliki sensitivitas terhadap beberapa jenis gas, sehingga hasil pembacaannya tidak digunakan untuk menentukan konsentrasi satu jenis gas secara selektif .

3. *Sensor BME280*

mengukur tekanan udara, yang membantu sistem menampilkan informasi kondisi cuaca sederhana dan memantau perubahan lingkungan secara *real-time* .

4. Status Perangkat

kondisi *ON/OFF* lampu dan kipas, termasuk mode operasional (otomatis, manual, atau perintah suara) yang dikontrol melalui *Google Assistant* dan *Blynk* .

Data primer ini dikumpulkan melalui *ESP32* sebagai mikrokontroler utama. Data *sensor* yang terdiri atas *Timestamp*, Suhu, Kelembapan, Tekanan, dan *PPM* dikirim serta disimpan ke *Google Sheets* melalui *Google Apps Script*.

3.3.2 Data Sekunder

Data sekunder diperoleh dari berbagai sumber literatur dan penelitian sebelumnya, yang berfungsi sebagai acuan dalam:

1. Menentukan kebutuhan dan spesifikasi sistem .
2. Mendukung desain dan pemilihan *sensor*, *aktuator*, serta metode pengembangan.
3. Memberikan dasar teori terkait *smart room*, *smart home*, kualitas udara, dan implementasi *IoT*.

3.3.3 Prosedur Pengumpulan Data

Prosedur pengumpulan data primer dilakukan sebagai berikut:

1. *Menginstal* semua *sensor* (*DHT22*, *MQ-135*, *BME280*) dan menghubungkan perangkat ke *ESP32*.
2. Menghubungkan *ESP32* ke *platform Blynk* dan mengonfigurasi perintah suara melalui *Google Assistant/Google Home* dan *MacroDroid*.

3. Menjalankan sistem secara kontinu dan merekam semua pembacaan *sensor*, status perangkat, dan notifikasi yang dihasilkan.
4. Sebelum pengujian, MQ-135 menjalani proses preheating selama 12 jam. Nilai R_0 dihitung secara berkala selama proses tersebut, kemudian nilai rata-rata dikunci dalam program ESP32 sebagai parameter perhitungan estimasi PPM.
5. Melakukan pengujian fungsional dengan metode *Blackbox Testing* untuk memastikan sistem bekerja sesuai spesifikasi tanpa meninjau kode internal .
6. Menganalisis data sensor yang tersimpan pada *Google Sheets* untuk menilai kinerja sistem, validitas pengukuran sensor, dan kestabilan proses pengiriman data dari *ESP32*.

3.4 Alat dan Bahan Penelitian

3.4.1 Perangkat Keras

Tabel 3. 1 Perangkat Keras Sistem

No.	Perangkat	Fungsi
1	<i>ESP32</i>	Membaca sensor, menjalankan logika kendali, dan berkomunikasi dengan <i>Blynk</i> .
2	<i>DHT22</i>	Mengukur suhu dan kelembapan ruangan.
3	<i>MQ-135</i>	Menghasilkan indikator relatif perubahan kualitas udara.
4	<i>BME280</i>	Mengukur tekanan udara.

No.	Perangkat	Fungsi
5	LCD 20×4 I2C	Menampilkan data sensor dan status aktuator secara lokal.
6	Modul <i>relay</i>	Menghubungkan <i>ESP32</i> dengan lampu dan kipas.
7	Lampu dan kipas	Beban/aktuator yang dikendalikan.
8	Alat referensi	Termohigrometer digital HTC-1 sebagai pembanding kelembapan, altimeter/barometer digital sebagai pembanding suhu dan tekanan udara, serta alat pengukur kualitas udara digital sebagai pembanding estimasi PPM MQ-135.

3.4.2 Perangkat Lunak dan Platform

Tabel 3. 2 Perangkat Lunak Sistem

No.	Perangkat lunak/platform	Fungsi
1	Arduino IDE	Menulis, mengompilasi, dan mengunggah program <i>ESP32</i> .
2	<i>Blynk</i>	<i>Dashboard</i> , <i>datastream</i> , notifikasi, dan kendali jarak jauh.
3	<i>Google Assistant/Google Home</i>	Antarmuka perintah suara.
4	<i>MacroDroid</i>	Mendeteksi notifikasi dan mengirimkan <i>HTTP GET</i> .
5	<i>Blynk External API</i>	Memperbarui virtual pin dari <i>MacroDroid</i> .
6	<i>Google Apps Script</i>	Endpoint penerima data dari <i>ESP32</i> .
7	<i>Google Sheets</i>	Menyimpan riwayat data sensor.