

BAB III

METODE PENELITIAN

3.1 Metode Penelitian

Metode penelitian ini dipilih berdasarkan pedoman dalam Blokdyk (2020) karena R&D memungkinkan pengembangan sistem secara sistematis dari tahap analisis kebutuhan hingga pengujian, sehingga cocok untuk penelitian yang bertujuan menghasilkan *prototipe* fungsional. Penelitian ini menggunakan model *Research and Development* (R&D) yang terdiri dari empat tahapan utama, yaitu:



_ Gambar 3. 1 Model *Research and Development* (R&D)

(Sumber: Dokumen Pribadi, 2025)

Setiap tahap memiliki fokus dan aktivitas sebagai berikut:

3.1.1 Tahap Analisis Kebutuhan (*Requirement Analysis*)

Tahap *Requirement Analysis* adalah tahap awal penelitian yang bertujuan untuk mengidentifikasi secara rinci kebutuhan sistem agar sesuai dengan tujuan penelitian dan menentukan spesifikasi yang diperlukan. Analisis ini memastikan bahwa sistem yang dikembangkan dapat memenuhi tujuan penelitian dan kebutuhan pengguna. Pada tahap ini, dilakukan:

a. Objek Penelitian

Penelitian ini bertujuan untuk merancang dan mengembangkan sistem monitoring kualitas jaringan internet menggunakan mikrokontroler ESP32 yang dilengkapi dengan antarmuka pengguna melalui *Nextion LCD TFT Touchscreen* dan aplikasi *MIT App Inventor*. Sistem ini dirancang untuk memberikan solusi monitoring jaringan secara *real-time*, sederhana, dan efisien tanpa memerlukan perangkat tambahan yang kompleks.

Fokus utama dari sistem ini adalah mengukur parameter kualitas jaringan, yaitu kecepatan unduh (*download speed*), kecepatan unggah (*upload speed*), serta

latensi (ping). Pengukuran dilakukan dengan mengakses layanan *Cloudflare Speedtest* secara langsung melalui ESP32. Alat ini bersifat portabel dan mandiri karena mendapat daya dari powerbank, serta dirancang untuk menampilkan hasil pengukuran secara langsung pada layar *Nextion* dan aplikasi *MIT App Inventor* melalui komunikasi nirkabel. Selain itu, hasil pengukuran juga dicatat secara otomatis ke *Google Sheets* sebagai bentuk dokumentasi data.

Pengujian sistem dilakukan di dua lokasi berbeda dengan kondisi jaringan yang bervariasi, yaitu di lingkungan institusi (Universitas Pendidikan Indonesia Kampus Cibiru) dan di rumah peneliti. Pengukuran dilakukan secara berkala pada empat waktu berbeda dalam satu hari, yaitu pagi (07.00–09.00); siang (12.00–14.00); sore (15.00–17.00) dan di malam hari (19.00–21.00) yang pengukurannya hanya khusus dilakukan di rumah saja, untuk memperoleh data yang mewakili fluktuasi jaringan sepanjang hari di masing-masing lokasi.

b. Fokus Utama Komponen

- **ESP32 (ESPduino32)** : Berperan sebagai mikrokontroler utama yang menjalankan program untuk mengakses jaringan Wi-Fi, melakukan pengujian kualitas jaringan (*download*, *upload*, dan *ping*), mengirim hasil ke *Google Sheets*, serta menampilkan hasil pengukuran pada antarmuka *Nextion LCD TFT Touchscreen* dan aplikasi *MIT App Inventor*.

- **Nextion LCD TFT Touchscreen** : Berfungsi sebagai tampilan antarmuka pengguna yang menampilkan hasil pengukuran dalam bentuk numerik dan visual secara *real-time*. Layar ini dilengkapi dengan tombol sentuh untuk memulai proses pengukuran secara manual.

- **Lampu LED Indikator** : Digunakan sebagai penanda visual terhadap kualitas jaringan yang sedang dipantau.

- Hijau : menandakan kualitas jaringan baik.
- Kuning : menunjukkan kualitas jaringan cukup atau sedang.
- Merah : menunjukkan kualitas jaringan buruk.

Indikator ini memudahkan pengguna untuk mengetahui status jaringan secara cepat tanpa perlu membaca angka pada layar.

- *CP2102 USB to TTL* : Digunakan untuk mengunggah desain tampilan antarmuka ke *Nextion LCD TFT Touchscreen* melalui komunikasi serial saat tahap pengembangan.
- *Powerbank 10.000 mAh* : Bertindak sebagai sumber daya utama yang memungkinkan sistem bekerja secara mandiri tanpa harus tersambung langsung ke listrik PLN.
- *Kabel USB, Kabel Jumper* : Berperan sebagai penghubung utama antar komponen dalam sistem, seperti *ESP32, Nextion LCD TFT Touchscreen, LED RGB*, dan *Power saklar ON/OFF*.
- *Saklar On/Off* : Merupakan saklar tipe rocker berukuran kecil (15x10 mm) yang berfungsi sebagai pengendali utama aliran daya ke sistem. Keberadaan saklar ini memberikan kenyamanan operasional serta melindungi komponen dari kerusakan akibat koneksi daya langsung yang terus-menerus.

Penelitian ini bertujuan untuk mengembangkan sistem monitoring kualitas jaringan internet yang komprehensif dan efektif guna mendukung kebutuhan akan layanan internet yang berkualitas. Integrasi komponen-komponen utama ini diharapkan dapat memberikan informasi yang akurat dan relevan mengenai kondisi jaringan internet serta memungkinkan identifikasi perbaikan yang diperlukan selama periode penelitian. Melalui pengembangan sistem monitoring ini, diharapkan dapat tercapai peningkatan efisiensi operasional dalam pemantauan dan pemeliharaan jaringan internet. Dengan adanya informasi yang lebih terperinci tentang parameter kualitas jaringan, layanan internet yang disediakan dapat ditingkatkan secara signifikan, meningkatkan kepuasan pengguna, dan mendukung pertumbuhan teknologi informasi dan komunikasi secara menyeluruh.

c. Spesifikasi Komponen Sistem

Pada perancangan sistem monitoring kualitas jaringan internet berbasis *ESP32* dan *Nextion LCD TFT Touchscreen*, digunakan beberapa komponen utama yang masing-masing memiliki peran penting dalam sistem. Berikut ini adalah spesifikasi dari komponen-komponen yang digunakan:

Tabel 3. 1 Tabel Spesifikasi Komponen

No.	Komponen	Spesifikasi Utama
1.	ESP32 (ESPduino32)	Dual-core Xtensa LX6, Wi-Fi 802.11 b/g/n, Bluetooth 4.2, GPIO hingga 34 pin, mendukung UART/I2C/SPI, tegangan kerja 3.3V
2.	<i>Nextion LCD TFT 2.8</i> Inci	Resolusi 320x240 piksel, layar sentuh resistif, komunikasi serial TTL, kompatibel dengan <i>Nextion</i> Editor
3.	<i>CP2102 USB to TTL</i>	Konversi USB ke UART, mendukung logika 3.3V/5V, kecepatan <i>baudrate</i> hingga 1 Mbps
4.	Lampu LED Indikator <i>RGB</i> (Merah, Kuning, Hijau)	Jenis LED standar, tegangan kerja 2V–3V, digunakan sebagai penanda kualitas jaringan berdasarkan warna
5.	Powerbank 10000 mAh	Output 5V/2A, port USB-A, digunakan sebagai sumber daya sistem secara portabel
6.	Kabel <i>USB Micro</i>	Kabel data/pengisi daya tipe <i>micro USB</i> , panjang 30–50 cm, digunakan untuk suplai daya ke ESP32 dan proses pemrograman
7.	Kabel <i>Jumper</i>	Kabel konektor (<i>male-to-male</i> , <i>male-to-female</i>), panjang ± 20 cm, digunakan untuk sambungan antar pin komponen
8.	Saklar On/Off <i>Rocker</i> <i>Switch</i>	Ukuran 15x10 mm, warna merah, tipe <i>rocker switch</i> 2-pin, dipasang pada jalur daya utama dari powerbank ke ESP32.

Setiap komponen dipilih berdasarkan kesesuaian terhadap kebutuhan sistem, kemudahan integrasi, serta ketersediaan daya dan komunikasi. Spesifikasi tersebut juga mempertimbangkan efisiensi daya dan kemudahan dalam proses debugging maupun pengembangan sistem secara keseluruhan.

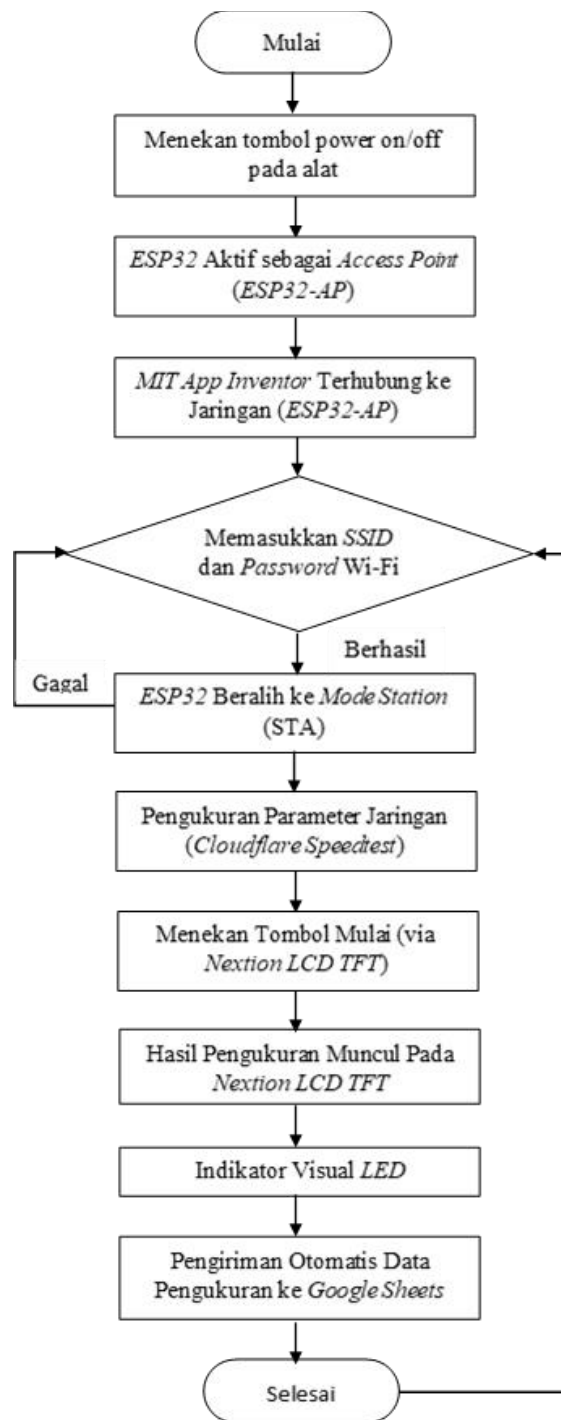
3.1.2 Tahap Perancangan (*Design Phase*)

Tahap perancangan memiliki tujuan utama untuk menyusun rancangan sistem sebelum diwujudkan ke dalam bentuk *prototipe*. Pada tahap ini, hasil analisis kebutuhan dijabarkan lebih rinci sehingga setiap modul, baik perangkat keras maupun perangkat lunak, dapat saling terhubung secara tepat. Proses perancangan mencakup beberapa langkah, mulai dari penyusunan *flowchart*, pembuatan blok diagram, perancangan arsitektur sistem, *wiring diagram*, hingga desain 3D alat. Selain itu, dirancang pula perangkat lunak beserta logika penetapan parameter kualitas jaringan agar sistem dapat bekerja sesuai tujuan.

3.1.2.1 *Flowchart* Sistem

Pengembangan prototipe sistem dilakukan melalui proses iteratif yang mencakup perancangan skematik rangkaian, pemrograman ESP32, perancangan tampilan antarmuka *Nextion*, dan integrasi dengan *MIT App Inventor*. Sistem dirancang agar portabel, mandiri, serta mudah digunakan untuk melakukan pengukuran kualitas jaringan Wi-Fi secara *real-time*.

Berikut adalah *flowchart* yang menggambarkan proses alur kerja sistem secara menyeluruh, mencakup penggunaan ESP32 dalam mode *Access Point* (ESP32-AP) dan integrasi dengan aplikasi *MIT App Inventor* :



Gambar 3. 2 Flowchart Sistem

(Sumber: Dokumen Pribadi, 2025)

Flowchart di atas mengilustrasikan bagaimana sistem monitoring jaringan bekerja. Berikut penjelasan untuk setiap langkahnya:

1. Mulai (*Start*): Sistem dinyalakan menggunakan saklar On/Off yang terhubung ke jalur daya dari powerbank ke ESP32. Setelah menyala, ESP32 memulai proses inisialisasi.
2. ESP32 Aktif sebagai *Access Point* (ESP32-AP): ESP32 membentuk jaringan Wi-Fi lokal (misalnya: *MoniFi_AP*) agar pengguna bisa terhubung melalui smartphone.
3. *MIT App Inventor* Terhubung ke ESP32-AP: Pengguna membuka aplikasi *MIT App Inventor* di ponsel dan terhubung ke jaringan Wi-Fi dari ESP32-AP.
4. Pengguna Memasukkan *SSID* dan *Password* Wi-Fi:
 - Aplikasi *MIT App Inventor* menyediakan form input untuk memasukkan nama *SSID* dan *password* jaringan Wi-Fi tujuan.
 - Data ini dikirim ke ESP32 melalui HTTP (misalnya lewat URL `http://192.168.4.1/setWifi?ssid=...&pass=...`).
5. ESP32 Beralih ke Mode Station (STA): ESP32 menyimpan *SSID* dan *password*, lalu mencoba menyambung ke jaringan Wi-Fi eksternal sesuai data yang diberikan.
 - Jika berhasil, ESP32 terhubung ke internet.
 - Jika gagal, ESP32 kembali ke AP mode dan meminta input ulang.
6. Pengukuran Parameter Jaringan: ESP32 mengakses server *Cloudflare* untuk mengukur:
 - Kecepatan unduh (*download speed*)
 - Kecepatan unggah (*upload speed*)
 - Latensi (*ping*)
7. Pengguna Menekan Tombol Mulai (via *Nextion LCD TFT*) :
 Setelah koneksi berhasil terhubung, pengguna dapat menekan tombol mulai pengukuran melalui layar *Nextion* agar dapat berpindah ke *slide* berikutnya dimana pada halaman tersebut berisi label tampilan hasil pengukuran dari parameter *upload speed*, *download speed* dan *ping*.

8. Menampilkan Hasil Pengukuran:

- *Nextion LCD TFT*: Menampilkan data numerik secara *real-time*.
- *MIT App Inventor*: Menampilkan hasil via koneksi HTTP dari ESP32

9. Analisis Hasil dan Indikator Visual:

- Hasil diolah dan ditampilkan ke pengguna.
- LED RGB menyala sesuai hasil:
 1. Hijau: Baik
 2. Kuning: Sedang
 3. Merah: Buruk

10. Pengiriman Otomatis ke *Google Sheets*:

Hasil pengukuran dikirim secara otomatis ke *Google Sheets* melalui *webhook* jika jaringan internet tersedia.

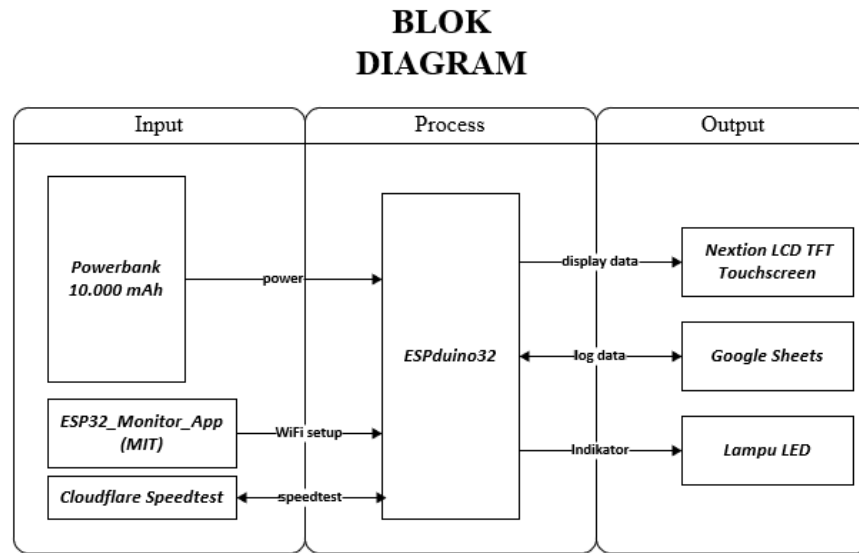
11. Selesai (Siap untuk Pengukuran Ulang):

Sistem kembali ke keadaan siap dan menunggu input selanjutnya dari pengguna.

Flowchart ini memperlihatkan bagaimana sistem dapat secara otomatis mengumpulkan, menganalisis, dan menampilkan data jaringan secara *real-time*, serta memberikan opsi bagi pengguna untuk mendapatkan informasi lebih lanjut jika diinginkan.

3.1.2.2 Blok Diagram

Blok diagram sistem dibuat dengan pendekatan IPO (*Input–Process–Output*), sehingga setiap komponen dapat ditempatkan sesuai fungsinya, mulai dari masukan yang diberikan ke sistem, unit pengolah, hingga keluaran yang dihasilkan. Rancangan blok diagram dapat dijelaskan sebagai berikut:



Gambar 3. 3 Blok Diagram Sistem

(Sumber: Dokumen Pribadi, 2025)

1. *Input*

Bagian input merupakan segala sesuatu yang masuk ke dalam sistem dan digunakan untuk mendukung proses pengolahan. Komponen yang termasuk ke dalam *input* adalah:

- Powerbank 10.000 mAh → ESP32 : berfungsi sebagai sumber energi utama untuk menyalakan mikrokontroler.
- Powerbank 10.000 mAh → *Nextion LCD TFT* : meskipun berada di sisi *output*, perangkat display tetap memerlukan daya langsung dari sumber *input*.
- *MIT App Inventor* → ESP32 : aplikasi berperan memberikan perintah konfigurasi (misalnya pengaturan SSID dan inisiasi pengukuran).
- *Cloudflare Speedtest* ⇔ ESP32 : layanan eksternal yang digunakan untuk menguji parameter jaringan (*download*, *upload*, dan *ping*) melalui komunikasi dua arah (*request-response*).

2. *Process*

Tahapan proses dipusatkan pada ESP32. Mikrokontroler ini menjadi inti pengendali yang menjalankan beberapa fungsi utama, yaitu:

- Menjalankan pengukuran kualitas jaringan melalui komunikasi dengan *Cloudflare Speedtest*.
- Mengolah data hasil pengukuran untuk dikategorikan ke dalam kondisi jaringan (baik, sedang, buruk).
- Mengirimkan data hasil ke berbagai saluran keluaran seperti tampilan visual, indikator LED, maupun penyimpanan data.

3. *Output*

Bagian output merepresentasikan hasil dari proses yang dilakukan oleh ESP32. Hasil tersebut dapat ditampilkan, diindikasikan, maupun disimpan untuk kebutuhan analisis. Komponen yang termasuk ke dalam *output* yaitu:

- *Nextion LCD TFT* $\leftarrow$ ESP32 : menampilkan hasil pengukuran berupa data numerik dan status jaringan.
- LED Indikator $\leftarrow$ ESP32 : memberikan isyarat visual menggunakan warna (hijau, kuning, merah) sesuai kategori kualitas jaringan.
- *Google Sheets* $\rightleftarrows$ ESP32 : menyimpan data hasil pengukuran secara otomatis melalui koneksi internet sekaligus memungkinkan verifikasi ulang.

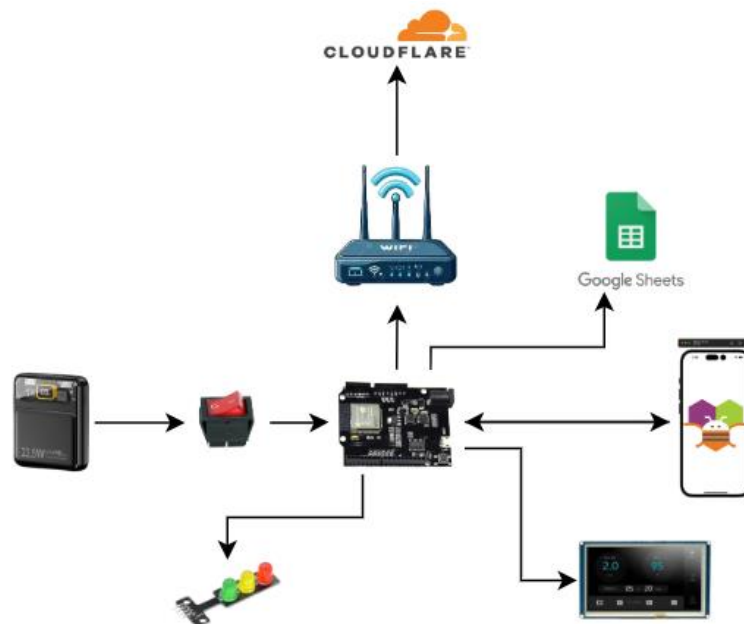
Keterangan Arah Panah:

1. Panah satu arah ($\rightarrow$) digunakan untuk aliran energi, perintah, maupun data hasil.
2. Panah dua arah ($\rightleftarrows$) menunjukkan adanya komunikasi timbal balik, misalnya proses *request-response* pada *Cloudflare* dan *Google Sheets*.

Dengan susunan ini, setiap komponen sistem dapat dipetakan dengan jelas sesuai peran dan alur kerjanya. ESP32 berfungsi sebagai pusat kendali, sementara

input memberikan daya dan perintah, dan output menyajikan hasil pengolahan baik dalam bentuk tampilan maupun penyimpanan data.

3.1.2.3 Arsitektur Diagram



Gambar 3. 4 Arsitektur Diagram

(Sumber: Dokumen Pribadi, 2025)

Arsitektur sistem monitoring kualitas jaringan seperti pada *Gambar 3.4 Arsitektur Diagram* diatas dirancang dengan pendekatan integrasi antara perangkat keras (*hardware*) dan perangkat lunak (*software*). ESP32 menjadi pusat pengendali utama yang memastikan setiap alur data, baik berupa masukan dari pengguna maupun hasil pengukuran kualitas jaringan, diproses terlebih dahulu sebelum diteruskan ke komponen lain. Dengan pola kerja ini, sistem dapat berjalan lebih terstruktur, terkontrol, dan konsisten dalam menyajikan informasi. Selain itu, rancangan arsitektur dibuat agar tidak hanya berfungsi secara lokal, tetapi juga mampu mendukung pemantauan jarak jauh melalui jaringan internet, sehingga sistem menjadi lebih fleksibel dan adaptif sesuai kebutuhan.

Secara umum, arsitektur sistem ini terbagi ke dalam lima komponen utama yang saling terhubung:

- Sumber daya dan kendali utama – Powerbank sebagai penyedia energi, sedangkan ESP32 berperan sebagai pusat logika dan pengendali alur komunikasi.
- Antarmuka lokal – *Nextion LCD TFT Touchscreen* menampilkan informasi hasil pengukuran sekaligus menjadi media interaksi dengan pengguna.
- Koneksi jaringan – Modul Wi-Fi internal ESP32 memungkinkan sistem terhubung ke internet untuk melakukan uji kualitas jaringan secara *real-time*.
- Pengolahan dan penyimpanan data – Data hasil pengukuran diolah oleh ESP32 kemudian dikirim ke *Google Sheets* sebagai media pencatatan log digital.
- Alur data antar komponen – Seluruh interaksi data berlangsung secara terintegrasi:
 - a. Data diterima:
 - ESP32 menerima input *SSID* dan *password* dari aplikasi *MIT App Inventor* untuk koneksi ke jaringan Wi-Fi.
 - ESP32 menerima input navigasi dari pengguna melalui layar *Nextion* (tombol *Start* dan tombol perpindahan halaman).
 - ESP32 menerima hasil uji jaringan dari *Cloudflare Speedtest*.
 - b. Data dikirimkan:
 - ESP32 mengirimkan hasil pengukuran (*download*, *upload*, *ping*) ke layar *Nextion* agar dapat ditampilkan secara *real-time*.
 - ESP32 mengirimkan data ke *Google Sheets* untuk pencatatan dan analisis historis.
 - ESP32 mengirimkan data ke aplikasi *MIT App Inventor* (Android) agar dapat diakses secara *mobile*.
 - ESP32 mengendalikan lampu LED indikator, yang menyala dengan warna berbeda sesuai kualitas jaringan (baik, sedang, buruk).

Dengan desain ini, sistem dapat memberikan keluaran yang beragam, baik berupa tampilan langsung pada layar lokal, indikator visual

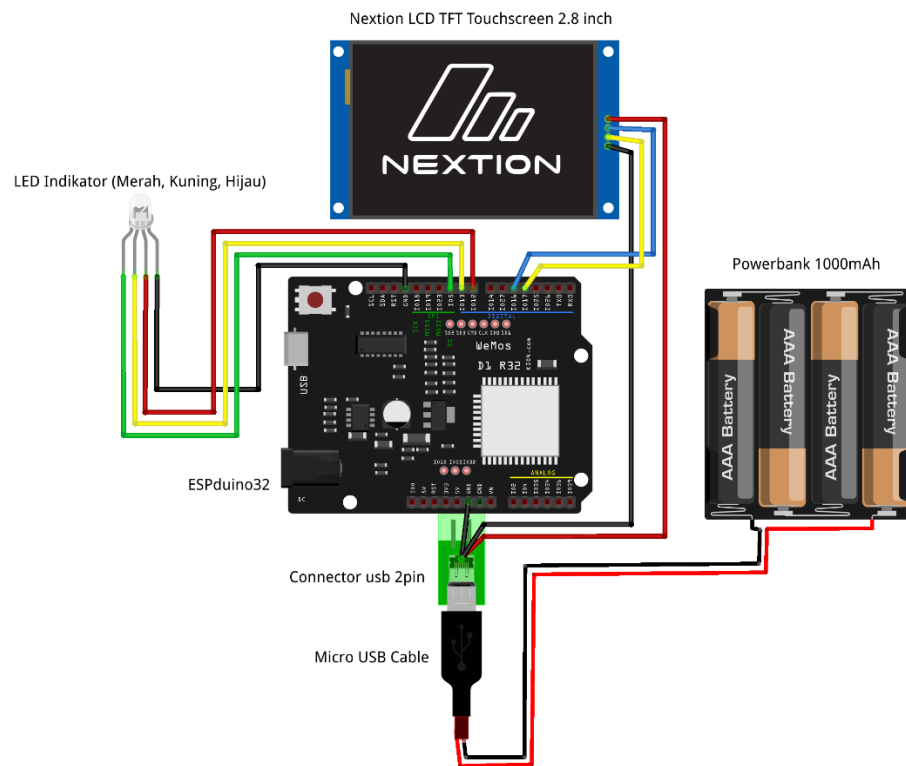
melalui LED, maupun dokumentasi berbasis *cloud*. Hal ini menjadikan sistem tidak hanya bermanfaat untuk pemantauan sesaat, tetapi juga untuk analisis jangka panjang yang mendukung peningkatan kualitas layanan jaringan.

3.1.2.4 Wiring Diagram

Pada perancangan perangkat, terdapat dua jenis *wiring* diagram dengan fungsi berbeda. *Wiring* diagram utama menunjukkan koneksi antar komponen agar sistem dapat berfungsi, sedangkan *wiring* diagram kedua dipakai khusus untuk proses *upload* desain antarmuka ke modul *Nextion LCD*. Pemisahan ini bertujuan memudahkan pemahaman dan mengurangi potensi kesalahan dalam perakitan maupun pemrograman.

- *Wiring* Diagram Alat

Wiring diagram alat memperlihatkan hubungan antara ESP32, *Nextion LCD* *TFT touchscreen*, LED indikator (merah, kuning, hijau), dan sumber daya dari powerbank. ESP32 berperan sebagai pusat kendali yang mengatur logika dan komunikasi antar perangkat. *Nextion LCD* menerima perintah serta menampilkan informasi status secara *real-time*. LED indikator berfungsi sebagai penanda kondisi sistem secara cepat, sedangkan powerbank berperan sebagai suplai daya agar alat dapat bekerja secara portabel. Dengan susunan ini, perangkat mampu beroperasi secara mandiri tanpa harus selalu terhubung ke komputer. Berikut merupakan gambar *wiring* diagram alat yang menunjukkan hubungan antar komponen utama dalam sistem.

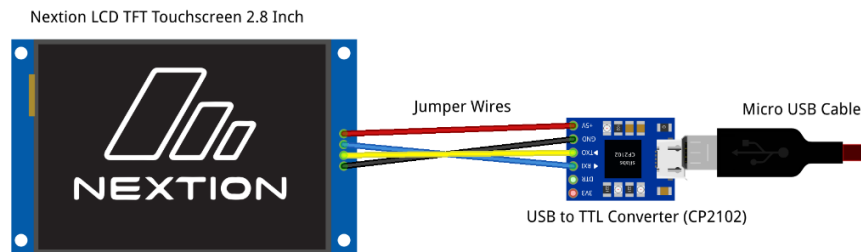


Gambar 3. 5 Wiring Diagram Alat

(Sumber: Dokumen Pribadi, 2025)

- *Wiring Diagram Upload Desain Interface*

Wiring diagram ini hanya digunakan saat proses pemrograman tampilan antarmuka pada *Nextion LCD TFT*. Modul *USB to TTL Converter* (CP2102) dipakai sebagai penghubung antara komputer dengan layar *Nextion* melalui kabel *micro USB*. Jalur komunikasi data menggunakan pin TX, RX, VCC, dan GND yang dihubungkan dengan tepat sesuai standar *TTL*. Dengan rancangan ini, desain antarmuka yang dibuat pada software *Nextion Editor* dapat diunggah langsung ke modul LCD sebelum dipasang ke sistem utama. Setelah proses *upload* selesai, *wiring* ini tidak lagi diperlukan, karena modul *Nextion* sudah menyimpan desain secara permanen di dalam memorinya. *Wiring diagram upload desain interface* pada gambar di bawah ini menunjukkan alur koneksi antar komponen.

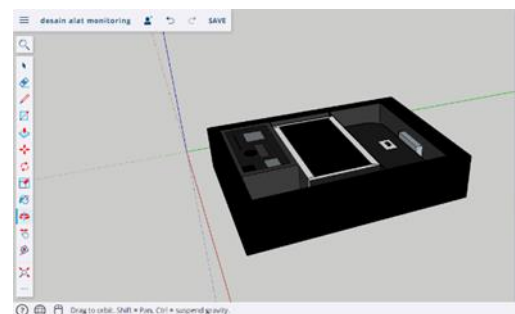
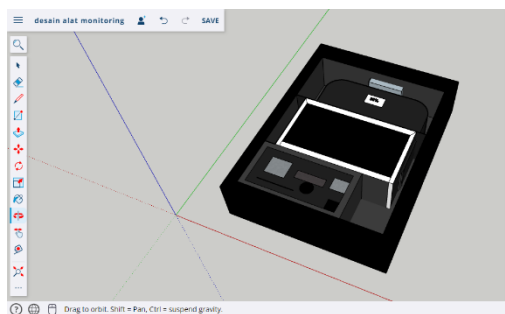


Gambar 3. 6 Wiring Diagram Upload Desain Interface

(Sumber: Dokumen Pribadi, 2025)

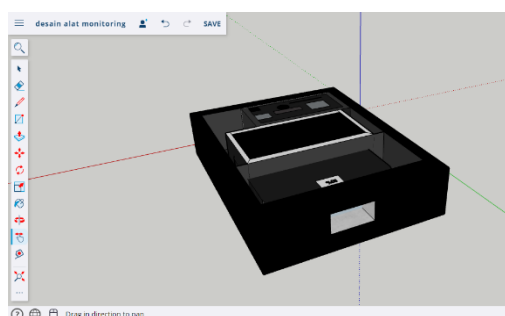
3.1.2.5 Desain 3D Alat

Desain alat monitoring kualitas jaringan internet yang digunakan dalam penelitian ini terdiri dari beberapa komponen utama yang diatur dengan sedemikian rupa sehingga dari segi fungsionalitas alat, efisiensi dan estetika tetap dapat terpenuhi. Di bawah ini merupakan tampilan gambar desain alat *monitoring* yang tampak dari berbagai sisi.



Gambar 3. 7 Desain 3D Alat Tampak Atas Gambar 3. 8 Desain 3D Alat Tampak Samping

(Sumber: Dokumen Pribadi, 2025)



Gambar 3. 9 Desain 3D Alat Tampak Belakang

(Sumber: Dokumen Pribadi, 2025)

Dari gambar desain alat di atas, terlihat bahwa:

- ESP32 ditempatkan di bagian bawah sebagai mikrokontroler
- *Nextion TFT 2.8 inch* diposisikan di bagian tengah *casing* untuk memudahkan akses dan pembacaan data oleh pengguna.
- Powerbank 10000 mAh terletak di bagian atas atau belakang *casing* sehingga memudahkan pengguna dalam memantau daya baterai dan proses pengisian dayanya.
- Kabel *USB* dan *jumper* disusun dengan rapi untuk menghubungkan semua komponen dengan stabil.

3.1.2.6 Perancangan Software

Dalam penelitian ini, pengembangan perangkat lunak memanfaatkan dua platform, yakni *Nextion Editor* untuk merancang antarmuka pada perangkat keras, serta *MIT App Inventor* yang digunakan sebagai aplikasi pendukung pada perangkat Android.

Berikut merupakan tahapan perancangan antarmuka dengan *Nextion Editor* dilakukan secara sistematis sebagai berikut:

1. Membuat proyek baru, yaitu dengan menentukan jenis modul dan resolusi layar sesuai perangkat yang digunakan (*Nextion 2.8 inci, 320×240 pixel*).
2. Menyusun komponen antarmuka, seperti tombol *Start*, kotak teks, serta bidang numerik untuk menampilkan hasil pengukuran *download speed*, *upload speed*, dan ping.
3. Merancang halaman informasi tambahan yang berisi deskripsi kualitas jaringan (baik, sedang, atau buruk) sehingga pengguna dapat langsung memahami hasil pengukuran.
4. Mengatur properti tiap komponen, meliputi ukuran, warna, teks, hingga *ID* objek agar mudah dipanggil oleh mikrokontroler.
5. Melakukan simulasi desain menggunakan fitur bawaan editor untuk memastikan tampilan bekerja sesuai rancangan sebelum diunggah ke perangkat fisik.

6. Mengunggah file hasil perancangan ke modul *Nextion LCD* melalui koneksi *USB to TTL* sehingga desain dapat dijalankan secara nyata.

Sementara itu, untuk tahapan perancangan aplikasi pendukung pada *MIT App Inventor* meliputi:

1. Membuat proyek baru dengan *template* aplikasi kosong sebagai dasar pengembangan.
2. Menambahkan elemen visual, misalnya Label untuk menampilkan parameter kecepatan jaringan dan Button sebagai kendali sinkronisasi data ke *Google Sheets*.
3. Menambahkan elemen non-visual, antara lain *Web Component* untuk menghubungkan aplikasi dengan *Google Sheets* melalui API, serta komponen komunikasi yang memungkinkan pertukaran data dengan ESP32 melalui Wi-Fi.
4. Menyusun blok logika program agar aplikasi mampu menerima data hasil pengukuran dari ESP32, menampilkannya secara *real-time* di layar ponsel, serta mengirimkan data tersebut ke *Google Sheets* untuk penyimpanan otomatis.
5. Melakukan pengujian aplikasi pada smartphone Android dengan memanfaatkan fitur *Live Testing* untuk memastikan seluruh fungsi berjalan sesuai kebutuhan.
6. Mengekspor aplikasi dalam format APK sehingga dapat dipasang dan digunakan pada perangkat Android lainnya.

3.1.2.7 Logika Penetapan Rentang Parameter Kualitas Jaringan

Penentuan rentang nilai dan klasifikasi kualitas jaringan internet dalam penelitian ini bertujuan untuk memberikan interpretasi yang praktis dan mudah dipahami terhadap hasil pengukuran dari sistem monitoring. Kategori kualitas jaringan dibagi menjadi tiga tingkat yakni baik, sedang dan buruk yang digunakan sebagai acuan dalam penilaian performa jaringan berdasarkan tiga parameter

utama: kecepatan unggah (*upload speed*), kecepatan unduh (*download speed*), dan latensi (ping).

Rentang nilai untuk masing-masing parameter ditetapkan berdasarkan:

- Hasil observasi empiris melalui serangkaian pengujian di dua lokasi berbeda (rumah dan kampus);
- Standar kualitas jaringan yang umum digunakan secara teknis dan praktis; dan
- Adaptasi terhadap keterbatasan perangkat seperti ESP32 dan metode pengukuran berbasis HTTP yang digunakan dalam sistem.

Pendekatan ini tidak hanya memberikan panduan evaluatif, tetapi juga memungkinkan penerjemahan hasil pengukuran ke dalam indikator visual menggunakan lampu LED RGB, yang sangat bermanfaat bagi pengguna non-teknis.

a. Upload Speed

- Baik : ≥ 1.0 Mbps
- Sedang : 0.70 – 0.99 Mbps
- Buruk : < 0.70 Mbps

Alasan Penetapan:

Rentang ini ditentukan berdasarkan kemampuan jaringan untuk menjalankan aktivitas unggah dasar seperti mengirim email dengan lampiran, mengunggah dokumen ke *cloud*, atau melakukan *video conference* standar. Nilai ≥ 1 Mbps dianggap memadai untuk kebutuhan tersebut. Sebaliknya, nilai di bawah 0.70 Mbps pada hasil pengujian menunjukkan ketidakstabilan yang berdampak pada keterlambatan proses unggah.

b. Download Speed

- Baik : ≥ 1.2 Mbps
- Sedang : 0.80 – 1.19 Mbps
- Buruk : < 0.80 Mbps

Alasan Penetapan:

Kecepatan unduh yang tinggi sangat penting untuk kegiatan seperti streaming video, *browsing* cepat, dan unduhan file. Dari hasil pengujian sistem, nilai ≥ 1.2 Mbps mampu memberikan pengalaman pengguna yang lancar. Nilai antara 0.8 hingga 1.19 Mbps dinilai cukup, sedangkan di bawah 0.8 Mbps sering kali menyebabkan *buffering* dan *delay*, terutama pada konten multimedia.

c. Ping (Latensi)

- Baik : < 50 ms
- Sedang : $50 - 99$ ms
- Buruk : ≥ 100 ms

Alasan Penetapan:

Ping adalah parameter penting untuk menilai responsivitas jaringan. Latensi di bawah 50 ms umumnya memberikan pengalaman *real-time* yang mulus, seperti untuk *video call* atau *online gaming*. Nilai antara 50–99 ms masih dapat diterima dalam banyak kasus, meskipun mungkin terdapat sedikit keterlambatan. Sementara itu, nilai ≥ 100 ms menunjukkan keterlambatan yang signifikan dan dapat mengganggu pengalaman komunikasi dua arah.

Rentang-rentang ini **tidak bersifat absolut** atau mengacu pada standar internasional tertentu, melainkan merupakan hasil adaptasi terhadap pengujian sistem secara langsung, dengan memperhatikan kondisi jaringan di lokasi uji dan keterbatasan alat. Pendekatan ini juga telah divalidasi secara fungsional melalui indikator LED RGB dan tampilan data pada layar *Nextion*, sebagaimana dijelaskan lebih lanjut dalam Bab IV.

Tahap perancangan bertujuan untuk menyusun rancangan sistem secara logis sebelum diwujudkan menjadi prototipe. Pada tahap ini, hasil analisis kebutuhan diterjemahkan menjadi spesifikasi rinci perangkat keras dan perangkat lunak, sehingga setiap modul dapat diimplementasikan dengan tepat. Menurut Adrian, Hidayat, dan Amrullah (2022), “*designing an IoT-based monitoring system*

requires translating user requirements into hardware and software specifications that can be effectively integrated into a functional prototype”.

Beberapa kegiatan utama pada tahap ini meliputi penyusunan arsitektur sistem secara konseptual, perencanaan alur kerja sistem (*workflow*) dari pengukuran kualitas jaringan hingga pencatatan data ke *Google Sheets*, serta penentuan koneksi antar komponen agar integrasi berjalan efisien. Meskipun detail teknis seperti *Flowchart*, Blok diagram, Arsitektur diagram, *Wiring* diagram komponen, dan Desain 3D alat akan dijelaskan lebih lanjut pada Bab 3.2 (Rancangan Sistem), tahap perancangan memastikan sistem siap diimplementasikan tanpa kesalahan mendasar.

3.1.3 Tahap Implementasi (*Implementation Phase*)

Tahap implementasi adalah proses mewujudkan rancangan sistem menjadi bentuk operasional yang nyata. Pada fase ini, desain logis yang telah disusun diterapkan ke perangkat keras dan perangkat lunak sehingga sistem dapat berjalan sesuai tujuan penelitian. Aktivitas implementasi dimulai dengan pemrograman mikrokontroler ESP32 untuk melakukan pengukuran kualitas jaringan internet, termasuk kecepatan unduh, kecepatan unggah, dan latensi. Data hasil pengukuran kemudian dikirim secara otomatis ke *Google Sheets* untuk dokumentasi dan analisis lebih lanjut.

Selain itu, tahap ini juga mencakup pengembangan antarmuka pengguna pada *Nextion LCD TFT Touchscreen* dan aplikasi *MIT App Inventor*. Antarmuka ini menampilkan hasil pengukuran secara *real-time* serta memungkinkan pengguna memulai proses pengukuran secara manual. Integrasi perangkat keras menjadi bagian penting, meliputi penyambungan LED indikator untuk menandai kualitas jaringan, pengaturan kabel *jumper* antar komponen, pemasangan saklar On/Off untuk mengatur aliran daya dari powerbank, serta penempatan modul agar sistem dapat berfungsi dengan efisien dan portabel.

Tahap implementasi juga melibatkan pengujian awal sistem untuk memastikan semua komponen bekerja dengan baik. Pengujian ini meliputi verifikasi koneksi antara ESP32 dan antarmuka pengguna, pengecekan pengiriman

data ke *Google Sheets*, serta pengamatan respon LED indikator terhadap perubahan kualitas jaringan. Dengan demikian, tahap implementasi memastikan sistem siap untuk diuji lebih lanjut dan dapat beroperasi secara nyata sesuai kebutuhan penelitian.

3.1.4 Tahap Pengujian (*Testing Phase*)

Penelitian ini menerapkan model *Research and Development* (R&D) sebagai kerangka utama dalam pengembangan sistem. Tahap pengujian difokuskan untuk mengevaluasi keandalan, akurasi, dan konsistensi sistem monitoring kualitas jaringan internet sebelum digunakan secara operasional. Pengujian dilakukan dengan pendekatan kuantitatif, sehingga memungkinkan pengumpulan data numerik yang objektif mengenai performa sistem di dua lokasi pengukuran, yaitu di lingkungan kampus dan rumah peneliti.

3.1.4.1 Objek dan Variabel Pengujian

Pengujian difokuskan pada dua variabel utama:

- Variabel Independen: Implementasi sistem monitoring berbasis mikrokontroler ESP32 yang terintegrasi dengan *Nextion LCD TFT Touchscreen* dan aplikasi *MIT App Inventor*. Sistem ini melakukan pengukuran kecepatan unduh, unggah, dan latensi secara *real-time* serta menyimpan data ke *Google Sheets*.
- Variabel Dependen: Akurasi pengukuran, responsivitas sistem, dan kenyamanan tampilan antarmuka. Indikator yang digunakan meliputi konsistensi hasil pengukuran antar waktu, kesesuaian data dengan kondisi jaringan aktual, waktu respons tombol, serta keterbacaan dan tata letak tampilan antarmuka.

3.1.4.2 Populasi dan Sampel

Populasi dalam penelitian ini mencakup seluruh kondisi jaringan Wi-Fi yang dapat dijangkau oleh sistem monitoring pada dua lokasi berbeda, yakni rumah dan kampus. Pemilihan sampel dilakukan secara *purposive*, dengan menitikberatkan pada perbedaan performa jaringan di kedua lokasi tersebut.

3.1.4.3 Skenario Pengujian

Pengujian sistem dilakukan untuk menilai sejauh mana perangkat ESPDuino32 yang terhubung dengan *Nextion LCD* mampu memantau kualitas jaringan Wi-Fi secara akurat dan konsisten. Proses pengujian dirancang sedekat mungkin dengan kondisi nyata, agar hasilnya dapat merepresentasikan pengalaman pengguna dalam kehidupan sehari-hari.

Dalam uji coba ini, terdapat beberapa parameter yang ditentukan sebagai dasar pelaksanaan, yaitu:

- **Lokasi Pengujian:** Pengujian dilaksanakan di dua tempat utama, yaitu rumah dan kampus. Keduanya menggunakan jaringan Wi-Fi yang berbeda sebagai sumber koneksi.
- **Waktu Pengujian:** Pengambilan data dilakukan pada empat sesi waktu berbeda dalam satu hari, yakni:
 Pagi: 07.00 – 09.00
 Siang: 12.00 – 14.00
 Sore: 15.00 – 17.00
 Malam: 19.00 – 21.00 (khusus di rumah)
- **Interval Pengambilan Data:** Untuk mendapatkan hasil yang representatif, pengukuran dilakukan setiap 5 menit sekali. Contohnya: 07.00, 07.05, 07.10, dan seterusnya.
- **Jenis Jaringan:** Seluruh pengujian menggunakan jaringan Wi-Fi, tanpa melibatkan jaringan seluler. Hal ini bertujuan untuk menjaga konsistensi sumber koneksi dan menghindari variabel luar yang sulit dikendalikan.

Fokus utama dari pengukuran ini mencakup:

- Kecepatan unggah (*upload speed*)
- Kecepatan unduh (*download speed*)
- Waktu respons jaringan (*ping*)

Setiap hasil pengamatan dicatat secara sistematis, lalu dianalisis menggunakan pendekatan statistik kuantitatif. Hal ini dimaksudkan agar pengambilan kesimpulan terhadap performa jaringan dapat dilakukan secara objektif dan berbasis data.

Sesudah skenario pengujian dirancang, setiap data yang diperoleh dicatat secara runtut berdasarkan lokasi serta waktu pelaksanaan. Agar informasi yang terkumpul lebih mudah dipahami, hasil pengamatan tersebut disajikan dalam bentuk tabel. Penyajian ini memudahkan pembaca dalam melihat perbedaan antar waktu maupun lokasi, sekaligus menjadi landasan untuk analisis statistik seperti rata-rata, simpangan baku, hingga uji t. Dengan cara ini, hubungan antara skenario pengujian dan tahap analisis dapat tergambarkan secara lebih jelas.

3.1.4.4 Teknik Pengumpulan Data

Data dikumpulkan melalui kombinasi:

- Pencatatan otomatis ke *Google Sheets* dari sistem monitoring.
- Observasi langsung oleh peneliti untuk memeriksa responsivitas sistem dan kestabilan perangkat.
- Dokumentasi visual, termasuk tampilan antarmuka pada *Nextion* dan *MIT App Inventor*, untuk menilai kenyamanan penggunaan dan kejelasan informasi.

3.1.4.5 Validitas dan Reliabilitas

- Validitas: Data diuji dengan membandingkan hasil pengukuran sistem terhadap kondisi jaringan aktual dan pola fluktuasi antar waktu serta lokasi. LED indikator RGB digunakan sebagai umpan balik visual tambahan.

- Reliabilitas: Sistem diuji menggunakan pendekatan pengulangan pengukuran (*test-retest*) untuk memastikan konsistensi hasil dalam interval waktu tertentu dan kestabilan pengukuran di lokasi yang sama. Analisis sederhana seperti standar deviasi digunakan untuk mengevaluasi stabilitas data.

3.1.4.6 Analisis Data

Hasil pengukuran dianalisis secara kuantitatif, mencakup perhitungan rata-rata, nilai tengah, standar deviasi serta uji-t. Data divisualisasikan dalam bentuk tabel dan grafik untuk memperlihatkan fluktuasi performa jaringan berdasarkan lokasi dan waktu pengujian. Pengujian hipotesis dilakukan secara terbatas, menggunakan uji-t dua sampel untuk melihat perbedaan rata-rata antara dua lokasi, serta mempertimbangkan pengaruh waktu terhadap hasil pengukuran.

Menurut Cinta, Wibawa, Hikmawan, dan Sari (2024), “*testing a network monitoring system is essential to ensure reliability and accuracy in measuring QoS parameters under various conditions*”. Tahap pengujian ini memastikan sistem berfungsi sesuai tujuan penelitian dan siap digunakan secara operasional, sekaligus memberikan dasar evaluasi yang kuat untuk kesimpulan penelitian.

- Rumus-rumus statistik yang dipakai

Rumus-rumus dasar yang digunakan dalam analisis ini adalah sebagai berikut:

1. Rata-rata (*Mean*)

Digunakan untuk mengetahui nilai tengah dari kumpulan data yang diamati.

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad (3.1)$$

$\bar{x}$ = nilai rata-rata

x_i = data ke-i

n = jumlah data

2. Standar Deviasi (*Standard Deviation*)

Menggambarkan seberapa besar sebaran atau variasi data terhadap nilai rata-ratanya.

$$s = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2} \quad (3.2)$$

s = standar deviasi sampel

$\bar{x}$ = nilai rata-rata

x_i = data ke-i

n = jumlah data

Selanjutnya merupakan rumus langkah demi langkah (*Welch's t-test*, dua arah, $\alpha=0,05$; $n_1=n_2=10$):

3. Selisih *mean*:

$$\Delta = \bar{x}_{\text{rumah}} - \bar{x}_{\text{kampus}} \quad (3.3)$$

Nilai Δ merepresentasikan perbedaan rata-rata hasil pengukuran antara dua lokasi, yaitu rumah dan kampus. Besaran ini menjadi indikator awal untuk mengetahui apakah terdapat jarak yang cukup berarti antara performa jaringan di kedua tempat. Jika Δ bernilai mendekati nol, maka kondisi jaringan relatif setara, sedangkan semakin besar nilai Δ menunjukkan perbedaan yang lebih mencolok.

4. *Standard Error (SE) Welch*:

$$SE = \sqrt{(s_{\text{rumah}}^2 / n_{\text{rumah}}) + (s_{\text{kampus}}^2 / n_{\text{kampus}})} \quad (3.4)$$

SE menggambarkan tingkat ketidakpastian dalam memperkirakan perbedaan rata-rata antar kelompok. Nilai SE yang kecil menandakan bahwa estimasi perbedaan lebih presisi, sedangkan SE besar menunjukkan adanya variasi yang tinggi pada data sampel.

5. Statistik t:

$$t = \Delta / SE \quad (3.5)$$

Statistik uji t menunjukkan seberapa besar perbedaan rata-rata relatif terhadap variasi data. Semakin tinggi nilai t, semakin kecil kemungkinan perbedaan tersebut muncul hanya karena faktor kebetulan. Nilai ini kemudian dibandingkan dengan distribusi t untuk menentukan signifikansi.

6. Derajat bebas (df) Welch:

$$df = \left(\frac{s_r^2}{n_r} + \frac{s_k^2}{n_k} \right)^2 \quad (3.6) \\ / \left[\left(\frac{s_r^2}{n_r} \right)^2 / (n_r - 1) + \left(\frac{s_k^2}{n_k} \right)^2 / (n_k - 1) \right]$$

Derajat bebas digunakan untuk menentukan distribusi probabilitas t yang sesuai. Pendekatan *Welch* dipilih karena lebih fleksibel dibandingkan asumsi homogenitas varians pada *Student's t-test*, sehingga lebih tepat bila varians antar kelompok tidak sama.

7. Interval kepercayaan (CI 95%):

$$\Delta \pm t(1-\alpha/2, df) \times SE \quad (3.7)$$

Interval kepercayaan memberikan rentang nilai yang diyakini memuat perbedaan rata-rata sebenarnya dengan tingkat keyakinan 95%. Jika rentang ini tidak meliputi angka nol, maka perbedaan yang ditemukan dapat dianggap signifikan secara statistik.

8. Ukuran efek (*Hedges g*):

Selain signifikansi statistik, penelitian yang menggunakan uji-t juga perlu memperhatikan ukuran efek. Ukuran efek memberikan informasi seberapa besar perbedaan yang terjadi antar kelompok, bukan hanya apakah perbedaan tersebut signifikan. Salah satu ukuran efek yang umum digunakan adalah *Hedges g*, yang merupakan pengembangan dari *Cohen's d* dengan penyesuaian terhadap bias pada sampel kecil. Perhitungannya dapat dijabarkan sebagai berikut:

9. *Pooled SD:*

$$sp = \sqrt{((n_r - 1) \cdot s_r^2 + (n_k - 1) \cdot s_k^2) / (n_r + n_k - 2)} \quad (3.8)$$

Rumus ini digunakan untuk mendapatkan simpangan baku gabungan dari dua kelompok.

10. *Cohen's d* = Δ / sp (3.9)

Δ merupakan selisih rata-rata antara kelompok eksperimen dan kelompok kontrol.

11. Koreksi bias: $J = 1 - (3 / (4(n_r + n_k) - 9))$ (3.10)

Faktor koreksi ini penting untuk mengurangi bias terutama pada ukuran sampel yang relatif kecil.

12. *Hedges g* = $J \times d$ (3.11)

Inilah nilai ukuran efek yang sudah dikoreksi.

3.1.4.7 Tabel Pengujian Awal

Sebagai bagian dari proses verifikasi performa sistem, dilakukan serangkaian pengujian di dua lokasi berbeda, yaitu rumah dan kampus, dengan memanfaatkan koneksi Wi-Fi sebagai sumber jaringan. Untuk masing-masing lokasi, ditampilkan 10 data sampel hasil pengukuran yang dipilih secara representatif berdasarkan variasi kondisi jaringan yang terjadi (baik, sedang, hingga buruk).

Data yang ditampilkan di bawah ini hanya sebagian kecil dari keseluruhan pengujian yang telah dilakukan. Tujuannya adalah untuk memberikan gambaran awal mengenai performa sistem dalam kondisi nyata. Adapun data secara lengkap dapat dilihat pada Lampiran A (rumah) dan (kampus). Untuk mendokumentasikan hasil yang diperoleh selama pengujian, disusunlah tabel sebagai berikut:

Tabel 3. 2 Pengujian Awal Jaringan Wi-Fi Rumah

No	Tempat Pengujian	Waktu Pengujian	Tanggal Pengujian	Jam	Upload Speed (Mbps)	Download Speed (Mbps)	Ping (ms)	Catatan
1.	Rumah	Pagi	27-07-2025	7:10	1.2	1.46	22	Kondisi stabil di pagi hari
2.	Rumah	Pagi	27-07-2025	7:15	0.68	1.45	20	Upload rendah, download stabil
3.	Rumah	Siang	27-07-2025	12:20	0.87	0.72	80	Kecepatan rendah dan ping tinggi
4.	Rumah	Siang	27-07-2025	12:30	0.89	0.73	80	Mirip kondisi sebelumnya (siang)
5.	Rumah	Sore	27-07-2025	15:00	1.33	1.03	111	Ping tinggi, kecepatan sedang
6.	Rumah	Sore	27-07-2025	15:35	1.07	1.03	112	Jaringan tidak stabil (ping tinggi)

No	Tempat Pengujian	Waktu Pengujian	Tanggal Pengujian	Jam	Upload Speed (Mbps)	Download Speed (Mbps)	Ping (ms)	Catatan
7.	Rumah	Sore	27-07-2025	15:45	1.26	1.56	21	Performa bagus di sore hari
8.	Rumah	Malam	26-07-2025	20:30	0.72	1.45	21	Ping bagus, kecepatan cukup
9.	Rumah	Malam	26-07-2025	20:35	0.45	1.34	21	Upload lambat, download lumayan
10.	Rumah	Malam	26-07-2025	20:50	0.68	1.45	21	Download stabil, ping bagus

Berdasarkan Tabel 3.2, kondisi jaringan Wi-Fi di rumah tampak cukup berfluktuasi sepanjang hari. Pada pagi hari, kualitas jaringan cenderung stabil. Hal ini terlihat dari kecepatan unduh yang konsisten di kisaran 1,4 Mbps dengan nilai ping rendah sekitar 20 ms, meskipun kecepatan unggah masih relatif kecil. Memasuki siang hari, kualitas jaringan menurun cukup drastis. Kecepatan unduh dan unggah turun, sementara ping melonjak hingga mencapai 80 ms, yang menunjukkan respons jaringan menjadi lambat.

Pada sore hari, jaringan terlihat tidak stabil; beberapa kali pengukuran bahkan mencatat ping lebih dari 100 ms, meskipun sempat ada hasil baik sekitar pukul 15:45. Berbeda dengan kondisi sebelumnya, pada malam hari jaringan kembali menunjukkan perbaikan. Ping menurun ke angka 21 ms dan kecepatan unduh kembali stabil, walaupun kecepatan unggah tetap rendah.

Tabel 3. 3 Pengujian Awal Jaringan Wi-Fi Kampus

No.	Tempat Pengujian	Waktu Pengujian	Tanggal Pengujian	Jam	Upload Speed (Mbps)	Download Speed (Mbps)	Ping (ms)	Catatan
1.	Kampus	Pagi	27-07-2025	7:20	1.29	1.51	20	Posisi dekat dengan router, koneksi lancar
2.	Kampus	Pagi	27-07-2025	7:30	1.27	1.52	20	Stabilitas tinggi di pagi hari
3.	Kampus	Pagi	27-07-2025	7:35	1.32	1.53	21	Performa tinggi, latensi rendah
4.	Kampus	Pagi	27-07-2025	7:55	1.36	1.52	20	Jaringan optimal di pagi hari
5.	Kampus	Pagi	27-07-2025	8:00	1.34	1.49	26	Latensi sedikit

No.	Tempat Pengujian	Waktu Pengujian	Tanggal Pengujian	Jam	Upload Speed (Mbps)	Download Speed (Mbps)	Ping (ms)	Catatan
								meningkat
6.	Kampus	Pagi	27-07-2025	8:20	1.33	1.51	22	Koneksi konsisten
7.	Kampus	Pagi	27-07-2025	8:50	1.33	1.5	21	Performa terjaga
8.	Kampus	Siang	27-07-2025	12:05	1.31	1.47	21	Stabil di tengah hari
9.	Kampus	Siang	27-07-2025	13:00	1.28	1.52	20	Performa konsisten, ping baik
10.	Kampus	Sore	27-07-2025	15:50	1.3	1.56	21	Koneksi tetap stabil di sore hari

Berdasarkan Tabel 3.3, jaringan Wi-Fi di kampus cenderung stabil dan menunjukkan performa yang lebih baik dibandingkan jaringan di rumah. Kecepatan unduh tercatat berada di kisaran 1,47 hingga 1,56 Mbps, dengan rata-rata sekitar 1,51 Mbps, dan selisih antar pengukuran relatif kecil. Kecepatan unggah pun cukup konsisten, berkisar antara 1,27 hingga 1,36 Mbps, dengan rata-rata 1,31 Mbps. Untuk nilai ping, hasil pengujian menunjukkan kestabilan pada rentang 20–26 ms, dengan rata-rata sekitar 21 ms. Hal ini mengindikasikan bahwa koneksi kampus

memiliki latensi rendah dan lebih responsif. Performa terbaik terlihat pada pagi hari, ketika kecepatan unduh mencapai 1,53 Mbps dengan ping hanya 20 ms.

Sementara itu, pada siang dan sore hari kondisi jaringan tetap terjaga, meskipun sempat terjadi sedikit lonjakan ping hingga 26 ms sekitar pukul 08:00.

Secara keseluruhan, hasil uji menunjukkan bahwa jaringan Wi-Fi kampus memiliki stabilitas yang baik serta latensi yang rendah, sehingga lebih mendukung kelancaran aktivitas *online* dibandingkan jaringan rumah.

Keterangan:

- **Tempat Pengujian:** Lokasi tempat sistem diuji, seperti rumah atau kampus.
- **Waktu Pengujian:** Merujuk pada sesi waktu tertentu (pagi, siang, sore, malam).
- **Tanggal dan Jam:** Menandakan kapan pengujian dilakukan secara rinci.
- **Upload Speed dan Download Speed:** Diukur dalam megabit per detik (Mbps) menggunakan protokol pengukuran berbasis *Cloudflare Speedtest*.
- **Ping:** Waktu tunda komunikasi data antar perangkat dan server, diukur dalam milidetik (ms).
- **Catatan:** Kolom ini berisi informasi tambahan selama proses pengujian berlangsung, misalnya gangguan sinyal atau cuaca.