

Implementasi Sistem Monitoring *CPU Usage* dan *Temperature* pada *Processor* berbasis *Internet of Things* menggunakan *Windows Management Instrumentation*

Arfin Teguh Fernanto¹, Vera Suryani², Endro Ariyanto³

^{1,2,3}Fakultas Informatika, Universitas Telkom, Bandung

¹arfintf@students.telkomuniversity.ac.id, ²verasuryani@telkomuniversity.ac.id,

³endroa@telkomuniversity.ac.id

Abstrak

Penggunaan komputer yang terlalu berlebihan merupakan hal yang tidak asing lagi untuk pengguna komputer, akibat penggunaan yang berlebihan dapat menyebabkan kenaikan *temperature* dan juga *CPU Usage* pada *processor*. Kebiasaan menggunakan komputer pada *temperature* yang tinggi dapat menyebabkan permasalahan seperti respon komputer semakin lambat, hingga permasalahan *hardware*. Untuk dapat mengantisipasi kemungkinan tersebut, diperlukan sistem monitoring yang dapat memberikan notifikasi, dan dapat mengontrol komputer dari jarak jauh. Pembuatan sistem monitoring dibuat dengan memanfaatkan sistem dari *Windows Management Instrumentation* untuk mendapatkan *temperature* dan *CPU Usage* dari komputer, dimana data nantinya akan diberikan ke aplikasi *Blynk* melalui *NodeMCU ESP8266*. Pada aplikasi *Blynk* terdapat sistem yang dapat memberikan notifikasi kepada pengguna bila *temperature* atau *CPU Usage* melebihi batas normal dan secara otomatis dapat menyalakan kipas tambahan, dan juga dapat mengontrol komputer untuk melakukan *sleep*, *shutdown*, atau *restart* melalui *smartphone*. Berdasarkan hasil pengujian diperoleh nilai *error rate* terkecil 1,56 % pada *temperature* dan 4,35 % pada *CPU Usage* saat dibandingkan dengan *software* monitoring lainnya, untuk *delay* mendapatkan nilai 0,50 ms pada kecepatan internet 9,8Mbps dan 0,28 ms pada kecepatan internet 20Mbps.

Kata Kunci : *blynk*, *windows management instrumentation*, *cpu temperature*, *cpu usage*

Abstract

Excessive computer use is a usual thing for computer users, due to excessive use it can cause an increase temperature and CPU Usage on the processor. The habit of using computers at high temperatures can cause problems such as slower computer response, to hardware problems. To be able to anticipate this possibility, a monitoring system that can provide notifications and can remotely control the computer is needed. The monitoring system is made by utilizing a system from *Windows Management Instrumentation* to get temperature and CPU usage from the computer, where the data will be given to the *Blynk* application via *NodeMCU ESP8266*. In the *Blynk* application there is a system that can notify users when the temperature or CPU Usage exceeds normal limits and can automatically turn on additional fans, and also control the computer to sleep, shutdown, or restart via a smartphone. Based on the test results, the smallest error rate is 1,56% at temperature and 4,35% on CPU Usage when compared to other monitoring software, for the delay get a value of 0,50 ms at 9,8Mbps internet speed and 0,28 ms at 20Mbps internet speed.

Keywords : *blynk*, *windows management instrumentation*, *cpu temperature*, *cpu usage*

1. Pendahuluan Latar Belakang

Menurut sebuah artikel ilmiah, salah satu alat atau komponen pada komputer yang memiliki proses distribusi panas adalah *processor* [1]. *CPU (Control Processing Unit)* atau biasa dikenal dengan *processor*, merupakan sirkuit elektronik di dalam komputer yang dimana berfungsi sebagai otak dan pusat pengendalian komputer untuk melakukan instruksi dari program komputer dengan melakukan aritmatika dasar, logika, pengontrolan, dan *input / output (I / O)* operasi yang ditentukan

oleh instruksi [2]. *Processor* yang terlalu panas (*overheat*) dapat menyebabkan kurang optimalnya kinerja pada *hardware* sehingga membuat komputer menjadi lambat.

Banyak pengguna komputer yang menggunakan komputer tanpa melihat kondisi komputer, baik itu *CPU Usage* maupun *temperature* pada *processor*. seorang *hardcore gamers* dapat memainkan *game* sehari hingga 10 – 12 jam [12], mereka lebih memilih untuk meninggalkan komputernya daripada mematikan komputernya pada saat tidak dimainkan. Hal ini, dapat berpeluang terjadi *overheating* dan merusak komponen-komponen komputernya.

Dikarenakan hal tersebut, maka penulis tertarik merancang sistem monitoring komputer yang memanfaatkan *Windows Management Instrumentation* yang dimana *Windows Management Instrumentation* merupakan layanan inti *Windows* dan *Windows Server* yang menyediakan tempat penyimpanan untuk menanyakan perangkat keras komputer, sistem operasi, aplikasi, dan sebagainya. Saat perlu memantau ratusan atau ribuan sistem komputer [4] dan memanfaatkan *Internet of Things* sehingga dapat memantau kondisi komputer dari manapun dan kapanpun sehingga akan mempermudah pengguna dalam melakukan pemantauan tanpa harus memantau kondisi komputer tanpa harus dekat dengan komputer .

Topik dan Batasannya

Penelitian yang dibangun adalah sistem monitoring yang dapat memantau komputer dari jarak jauh secara *real-time* yang memiliki sistem *alert* yang dapat memberitahu pengguna jika *temperature*, dan *CPU Usage* komputernya sudah tinggi dan dapat mengontrol komputer.

Batasan masalah dari penelitian ini adalah data yang diambil menggunakan *Windows Management Instrumentation* dengan menggunakan *library* pada *OpenHardwareMonitor* serta menggunakan aplikasi *Blynk* untuk melakukan pemantauan, memberikan notifikasi, mengontrol komputer, dan menyalakan *relay*.

Tujuan

Tujuan dari penelitian ini adalah membuat sistem berbasis *IoT* yang dapat memonitoring *CPU Usage* dan *temperature processor* yang dapat memberikan sistem *alert* ke aplikasi dan mengontrol komputer untuk melakukan *shutdown*, *restart*, atau *sleep* melalui aplikasi *Blynk* pada *smartphone* dan mengetahui hasil performansi sistem yang telah dibangun.

Organisasi Tulisan

Struktur penulisan penelitian ini berupa pendahuluan yang berisi latar belakang permasalahan, studi terkait yang berisi teori dari penelitian sebelumnya yang mendukung penelitian ini, gambaran umum sistem yang berisi alur proses pembuatan sistem, pengujian dan analisis yang berisi hasil dari pengujian sistem yang sudah dibuat beserta analisisnya, dan kesimpulan yang berisi hasil akhir dari pengujian yang dilakukan beserta saran untuk pengujian berikutnya.

2. Penelitian Sebelumnya

Penelitian pertama yang dijadikan referensi dalam penulisan adalah [5] pada artikel adalah membuat sebuah alat sensor *temperature* dan *humidity* dengan menggunakan *DHT11* dan *Arduino* untuk mendeteksi suhu yang dimiliki oleh laptop, tujuan dari penelitian ini adalah untuk mendeteksi suhu secara terus menerus, memberikan informasi suhu laptop.

Adapun penelitian kedua yang menjadi referensi adalah [1] yang dimana penelitian membuat alat monitoring suhu komputer dengan menggunakan sensor *LM35* dengan menggunakan *port parallel* dan mengolah data *ADC* dari *IC-LM35* untuk menunjukkan suhu yang dimiliki oleh laptop.

Penelitian ketiga yang dijadikan referensi adalah [6] pada penelitian menggunakan pemantauan suhu jarak jauh menggunakan *LM35*, *Arduino Uno*, dan modul *Wi-Fi ESP8266*, serta menggunakan koneksi ke server *MQTT*, pada penelitian *Arduino Uno* berfungsi untuk membaca suhu dari sensor *LM35*, dan *ESP8266* berfungsi untuk mengimplementasikan *client MQTT* sehingga dapat melakukan pemantauan jarak jauh.

Penelitian keempat yang dijadikan referensi adalah [7] pada penelitian merancang sebuah sistem monitoring suhu dengan memanfaatkan mikrokontroler *W5100* pada modul *WIZ110SR* sebagai regulator komunikasi data dengan jalur *Ethernet TCP/IP*, dan jalur data *RS232* beserta

mikrokontroler AT8535 sebagai sistem yang bertanggung jawab untuk pengontrolan suhu. Sensor suhu yang dipakai adalah LM35. Cara kerja sistemnya adalah mikroprosesor pada WIZ10SR dimanupulasi untuk dapat mengetahui keadaan suhu, setelah data didapatkan akan diproses oleh mikroprosesor, kemudian data suhu yang dihasilkan oleh sensor diteruskan ke mikrokontroler W5100 pada modul WIZ10SR melalui port RS232 yang kemudian data suhu dikirimkan kepada user melalui jaringan komputer.

Semua penelitian tersebut menunjukkan cara melakukan dan memonitoring *temperature* komputer, namun pada setiap penelitian hanya memonitoring atau menangkap suhu dari *area* komputer. Pada dasarnya, komputer memiliki berbagai macam perangkat yang dapat menghasilkan panas, dan menyebabkan kerusakan-kerusakan lain. Jika hanya melakukan pemantauan terhadap *area* komputer, maka pengguna tidak akan mengetahui apakah salah satu perangkatnya mengalami *overheat* atau tidak. Adapun dari permasalahan tersebut penulis mengangkat judul "Implementasi Sistem Monitoring *Temperature* dan *CPU Usage* pada Komputer berbasis *Internet of Things (IOT)* menggunakan *Windows Management Instrumentation*" dimana menurut Clint Huffman [4] *Windows Management Instrumentation (WMI)* merupakan layanan inti dari *Windows* dan *Windows Server* yang menyediakan *repository* untuk dapat mendapatkan data *hardware*, *system operation*, aplikasi, dsb.

Dengan adanya *WMI* pengguna dapat meminta data secara langsung kepada sistem, tentang berapa *temperature* yang dimiliki oleh salah satu perangkat. Dalam hal ini penulis mengambil perangkat *CPU (processor)* sebagai tema yang diangkat, sehingga tingkat keakurasiannya dapat dikatakan cukup akurat. *ESP8266* serta *Blynk* digunakan untuk mengkomunikasikan komputer dengan *smartphone* sehingga sistem dapat dikatakan termasuk *Internet of Things (IoT)*.

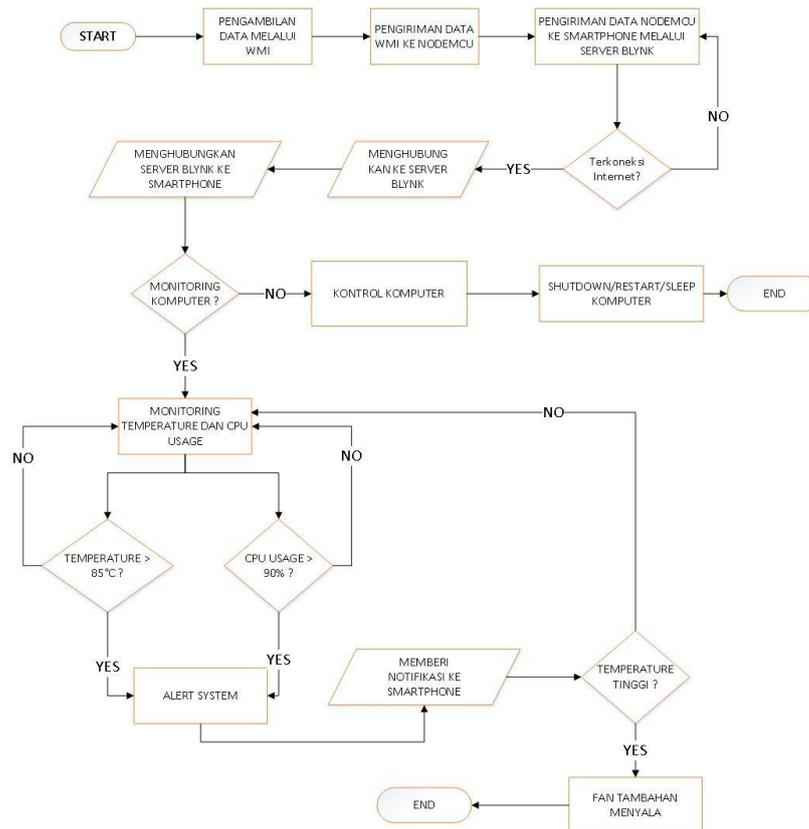
3. Sistem yang Dibangun

3.1 Gambaran Umum Sistem

Pada penelitian ini dilakukan pembangunan sistem yang akan dibuat yaitu data *CPU usage* dan *temperature processor* diambil dari komputer melalui *WMI*, kemudian data *WMI* akan diterima oleh *NodeMCU*, selain berfungsi sebagai perangkat yang menerima data dari *WMI*. *NodeMCU* juga berfungsi sebagai perangkat yang membuat *relay* menjadi *switch* pada *fan* yang nantinya *fan* akan secara otomatis menyala dan mati tergantung dari suhu pada komputer, data yang ada pada *NodeMCU* akan dibawa melalui *Wi-Fi* menuju *smartphone* melalui aplikasi *Blynk*. sehingga terjadi fungsi *Internet of Things (IoT)*.

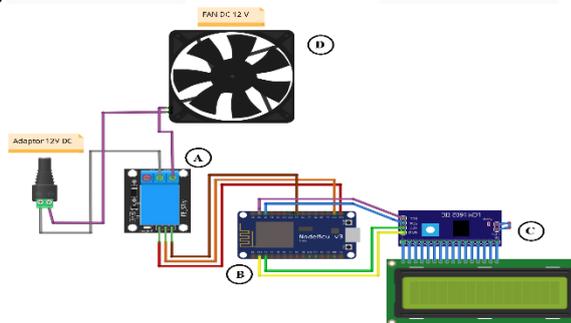
Setelah data yang telah didapat terkirim pada *Blynk*, maka *smartphone* dapat dipakai untuk memonitoring kondisi *CPU Usage* dan *temperature processor*. Jika *temperature* pada komputer > batas yang telah ditentukan, maka akan keluar *alert* yang memberitahu pengguna bahwa *temperature* melewati batas wajar, kemudian secara otomatis, *fan* tambahan akan dinyalakan, jika *temperature* < batas yang telah ditentukan, *fan* tambahan akan mati. Jika *CPU Usage* dari komputer > batas penggunaan *CPU Usage* yang telah ditentukan maka akan keluar *alert* yang memberitahu pengguna bahwa *CPU Usage* melebihi batas normal.

LCD 16x2 digunakan untuk melihat *output* data yang didapatkan secara langsung tanpa menggunakan *smartphone*.



Gambar 1. Flowchart Sistem Monitoring CPU Usage dan Temperature

Alat-alat yang Dibangun



Gambar 2. Desain Perakitan Hardware untuk Sistem Monitoring

Gambar 2 merupakan desain dari hardware yang akan digunakan pada perancangan Implementasi Sistem Monitoring CPU Usage dan Temperature pada Processor berbasis Internet of Things (IOT) menggunakan Windows Management Instrumentation. Terdapat beberapa komponen hardware diantaranya.

- a. Relay Board
Relay Board atau Relay digunakan sebagai switch untuk mengendalikan Fan tambahan.
- b. NodeMCU ESP8266
ESP8266 adalah sebuah embedded chip yang di desain untuk komunikasi berbasis Wi-Fi. Chip ini memiliki output serial TTL dan GPIO. ESP8266 dapat digunakan secara sendiri (standalone) maupun digabungkan dengan pengendali lainnya seperti mikrokontroler [8].

- c. *I2C LCD 16x2*
Media yang digunakan untuk menampilkan data *CPU Usage* dan *temperature processor* yang didapatkan dari komputer.
- d. *Fan Komputer*
Perangkat tambahan yang digunakan ketika *temperature* pada komputer > 70°C yang secara otomatis akan menyala.

Perangkat saat di Implementasikan



Gambar 3. Implementasi alat yang dirancang

Software yang Digunakan

Software yang digunakan adalah *Blynk*, yang merupakan papan atau *dashboard digital* dimana pengguna dapat menyesuaikan kebutuhan dengan mengatur tampilan antar muka. Papan ini sangat mudah untuk digunakan semua pengguna [9]. Aplikasi *Blynk* memiliki antarmuka grafis dalam pembuatannya, untuk pembuatan antarmuka dapat menggunakan *Drag and Drop*. *Blynk* juga terbagi menjadi 3 komponen utama yaitu :

- *Blynk Apps*
- *Blynk Server*
- *Blynk Library*

4. Penelitian dan Analisis

Pada penelitian ini yang bertujuan untuk melakukan pemantauan *temperature* dan *CPU Usage* dan melakukan pengontrolan komputer secara *remote*. Beberapa skenario pengujian dilakukan untuk mendapatkan hasil yang terbaik pada tabel 1:

Tabel 1. Skenario Pengujian

No	Skenario	Hasil yang diharapkan
1	Membandingkan data <i>CPU Usage</i> dan <i>temperature</i> yang dimunculkan dengan <i>software monitoring</i> komputer lain .	Data <i>CPU Usage</i> dan <i>temperature</i> yang didapatkan tidak terlalu besar perbedaannya.
2	Mengirimkan data <i>CPU Usage</i> dan <i>temperature</i> dengan <i>bandwith</i> yang berbeda.	Perbedaan <i>bandwith</i> tidak mempengaruhi waktu pengiriman data secara besar.
3	Melakukan <i>stress test</i> atau memperkecil batasan penggunaan <i>temperature</i> dan <i>CPU Usage</i> sehingga dapat melihat apakah fungsi <i>alert</i> berfungsi dengan baik.	<i>Alert</i> berfungsi dengan baik dan dapat melakukan pemantauan secara <i>real-time</i> .
4	Melakukan pengujian seluruh fungsionalitas alat yang dibuat untuk mengetahui alat yang telah dibuat berfungsi dengan baik	Seluruh alat yang dibuat berfungsi dengan baik

Pengujian Perbandingan data *CPU Usage* dan *Temperature*

Pengujian dilakukan dengan mendapatkan *temperature* dari semua *core processor* untuk menghitung akurasi alat dan *error rate* yang dibandingkan dengan program pemantauan *HWiNFO64*, *HWMonitor*, dan *Aida64 Business* untuk *temperature*, dan *HWiNFO64*, *HWMonitoring* dan *NZXT Cam* untuk *CPU Usage*. Pengujian dilakukan pada keadaan komputer pada saat *idle* dan hanya membuka program dan aplikasi pemrograman, dengan *baudrate* dari alat sebesar 9600 dengan pengujian perbandingan sebanyak 10x.

Hasil perhitungan akan ditentukan oleh rumus berikut:

$$\%Error = \frac{|\text{Nilai Perkiraan} - \text{Nilai Eksak}|}{\text{Nilai Eksak}} \times 100 \quad (1)$$

$$Accuracy = \frac{\text{Jumlah Data yang sama}}{\text{Banyaknya Data}} \times 100 \quad (2)$$

Tabel 2. Hasil Pengujian Akurasi Perbandingan data *Temperature*

AlatRakitan dengan	Core #0	Core #1	Core #2	Core #3	Core #4	Core #5
HWiNFO64	80 %	80 %	100 %	100 %	100 %	100 %
HWMonitoring	100 %	90 %	70 %	90 %	100 %	70 %
Aida64 Business	90 %	80 %	70 %	100 %	100 %	90 %

Pengujian dilakukan 10x dan mendapatkan hasil akurasi melalui hasil perbandingan antara alat rakitan dengan *software* monitoring akurasi diambil dengan *range* 2, jika perbandingan antara alat dengan *software* monitoring > 2 maka dianggap tidak sama. Hasil *temperature* yang dikeluarkan dari setiap *core* mendapatkan hasil akurasi terkecil dengan nilai akurasi 70% dan nilai akurasi terbesar dengan nilai 100%.

Tabel 3. Hasil Pengujian Akurasi Perbandingan data *CPU Usage*

	NZXT CAM	HWiNFO64	HWMonitor
Alat	70%	90 %	100 %

Pengujian dilakukan 10x dan mendapatkan hasil akurasi melalui hasil perbandingan antara alat rakitan dengan *software* monitoring akurasi diambil dengan *range* 2, jika perbandingan antara alat dengan *software* monitoring > 2 maka dianggap tidak sama, dengan hasil akurasi terkecil pada *NZXT CAM* dengan nilai akurasi 70% dan akurasi terbesar pada *HWMonitor* dengan akurasi 100%.

Tabel 4. Hasil Pengujian *Error Rate* Perbandingan data *Temperature*

Alat Rakitan dengan	Core #0	Core #1	Core #2	Core #3	Core #4	Core #5
HWiNFO64	5,20 %	3,77 %	2,32 %	2,16 %	2,60 %	3,48 %
HWMonitoring	2,91 %	3,17 %	5,29 %	1,56 %	2,42 %	2,87 %
Aida64 Business	4,23 %	2,89 %	5,82 %	1,85 %	1,59 %	3,47 %

Pengujian dilakukan 10x dan mendapatkan hasil *error rate* melalui hasil perbandingan selisih *output* antara alat rakitan dengan *software* monitoring berdasarkan *temperature* pada setiap *core*, dan mendapatkan hasil *error rate* terkecil pada *HWMonitoring* Core #3 dengan *error rate* 1,56% dan *error rate* terbesar pada *Aida64 Business* pada Core #2 dengan *error rate* 5,82%.

Tabel 5. Hasil Pengujian *Error Rate* Perbandingan data *CPU Usage*

	NZXT CAM	HWiNFO64	HWMonitor
Alat	14,60%	4,35 %	6,98 %

Pengujian dilakukan 10x dan mendapatkan hasil akurasi melalui hasil perbandingan antara alat rakitan dengan *software* monitoring berdasarkan *CPU Usage* yang digunakan dan mendapatkan hasil *error rate* 14,60% pada *NZXT CAM*, 4,35% pada *HWiNFO64*, dan 6,98 % pada *HWMonitor*.

Analisis Hasil Pengujian Perbandingan data *CPU Usage* dan *Temperature*

Berdasarkan hasil pengujian yang telah dilakukan akurasi yang didapatkan pada perbandingan data *temperatur* didapatkan akurasi terkecil 70% dan akurasi terbesar 100%, dan pada perbandingan *CPU Usage* akurasi yang didapatkan adalah 70% untuk yang terkecil dan 100% untuk yang terbesar. Akurasi didapatkan dengan cara mengambil *range 2* pada setiap perbandingan.

Error rate yang didapatkan berdasarkan dari selisih *output* perbandingan alat dengan *software* monitoring tanpa menggunakan *range*, dimana *error rate* yang didapatkan adalah 1,56% untuk yang terkecil dan 14,60% untuk *error rate* terbesar.

Pengujian *Delay* Pengiriman Data

Pengujian data dilakukan secara manual menggunakan *Postman* yang di mana data dimasukkan sebanyak 60 kali. Pengujian bertujuan untuk mengukur rata-rata *delay* transmisi yang terjadi pada saat pengiriman data yang diberikan oleh komputer menuju *Blynk*. Hasil perhitungan akan ditentukan dengan rumus berikut:

$$\text{Delay rata - rata} = \frac{\text{Total Delay}}{\text{Total Packet yang dikirim}}$$

Percobaan pertama menggunakan *hotspot* dari *smartphone* dengan menggunakan provider Telkomsel.

Tabel 6. Pengujian *Delay* menggunakan aplikasi *Postman* dan menggunakan provider Telkomsel

Total Percobaan	Rata-Rata <i>Delay</i>	Satuan waktu
60	0,50	ms

Pengujian dilakukan dengan memasukkan data sebanyak 60x dan mendapatkan hasil rata-rata *delay* sebesar 0,50 ms.

Tabel 7. Pengujian *Delay* menggunakan aplikasi *Postman* dan provider Indihome 20 Mbps

Total Percobaan	Rata-Rata <i>Delay</i>	Satuan waktu
60	0,28	ms

Pengujian dilakukan dengan memasukkan data sebanyak 60x dan mendapatkan hasil rata-rata *delay* sebesar 0,28 ms.

Analisis *Delay* Pengiriman Data

Berdasarkan pengujian yang telah dilakukan, hasil yang diperoleh dengan menggunakan provider Telkomsel adalah sebesar 0,50 ms sedangkan rata-rata *delay* yang diperoleh saat menggunakan Indihome dengan kecepatan 20 Mbps adalah sebesar 0,28 ms. Berdasarkan perusahaan riset jaringan mobile *Open Signal* merilis hasil "Laporan Pengalaman Jaringan Seluler Juli 2019" menyatakan bahwa kecepatan unduh rata-rata di jaringan Telkomsel mencapai 9,8 Mbps [11]. Menurut *TIPHON* [12], besarnya *delay* dapat diklasifikasikan sebagai berikut:

Tabel 8. Klasifikasi *Delay*

Kategori <i>Delay</i>	Besar <i>Delay</i> (ms)	Indeks
Sangat Bagus	<150	4
Bagus	150 s/d 300	3
Sedang	300 s/d 450	2
Buruk	>450	1

Pengujian Sistem *Alert* alat yang dirancang

Pengujian bertujuan untuk mengetahui apakah fungsi-fungsi yang dibuat pada aplikasi *Blynk* dapat berfungsi dengan baik

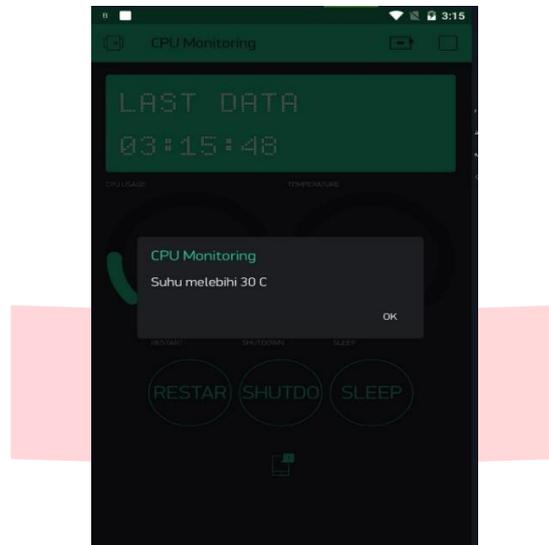
Untuk pengujian sistem *alert* dibagi menjadi beberapa tahap yaitu:

- *Temperature* melebihi batas *temperature* yang telah ditentukan.
- *CPU Usage* melebihi batas *CPU Usage* yang telah ditentukan.

- *CPU Usage* dan *temperature* melebihi batas yang telah ditentukan.

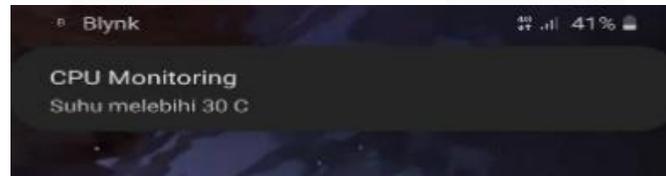
Pengujian *Temperature* melebihi batas yang telah ditentukan

Dalam pengujian ini batas suhu *temperature* yang ditentukan adalah 30°C untuk memudahkan melakukan pengujian karena rata-rata komputer yang saya gunakan memiliki suhu 25 - 40°C pada saat *idle*. Jika suhu mencapai 30°C aplikasi *Blynk* dibuka akan mengeluarkan *alert* berupa gambar 4.



Gambar 4. Hasil pengujian *Sistem Alert* batas *Temperature* 30°C pada *Blynk*

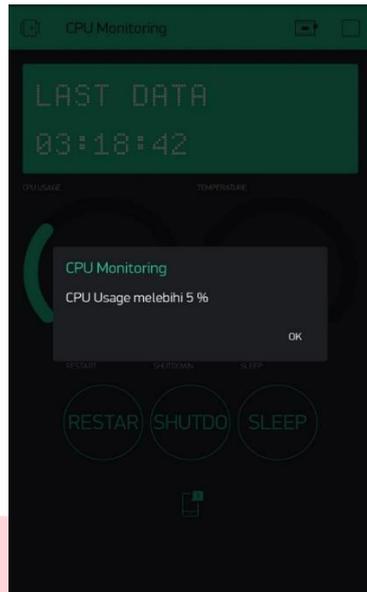
Namun apabila aplikasi *Blynk* dalam keadaan tertutup maka akan menghasilkan notifikasi seperti pada gambar 5.



Gambar 5. Hasil pengujian *Sistem Alert* batas *Temperature* 30°C pada *Smartphone*

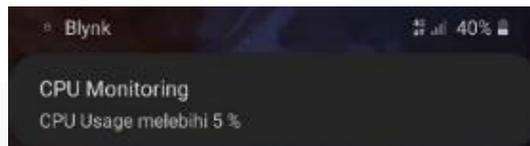
Pengujian *CPU Usage* melebihi batas yang telah ditentukan

Dalam pengujian ini batas *CPU Usage* yang ditentukan adalah 5% untuk memudahkan melakukan pengujian karena rata-rata menggunakan *CPU Usage* komputer yang dipakai sekitar 3 – 14% pada saat *idle*. Jika *CPU Usage* mencapai 5%. Maka jika aplikasi *Blynk* dibuka akan mengeluarkan *alert* berupa gambar 6.



Gambar 6. Hasil pengujian Sistem Alert batas Usage 5% pada Blynk

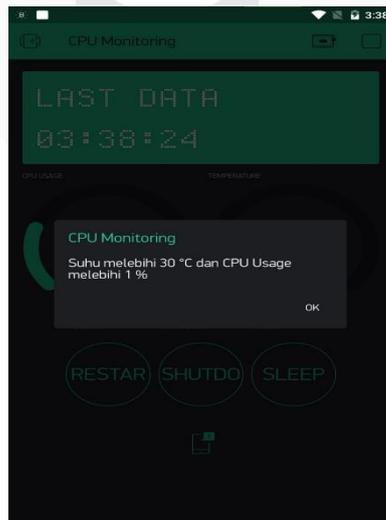
Namun apabila aplikasi *Blynk* dalam keadaan tertutup maka akan menghasilkan notifikasi seperti pada gambar 7.



Gambar 7. Hasil pengujian Sistem Alert batas Usage 5% pada Smartphone

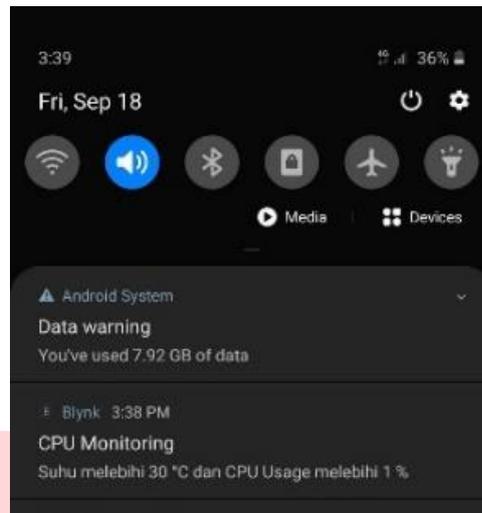
Pengujian CPU Usage dan Temperature melebihi batas

Dalam pengujian ini batas *CPU Usage* yang ditentukan adalah 1% dan *temperature* yang ditentukan adalah 30°C, Setelah dilakukan pengujian, jika aplikasi *Blynk* dibuka akan mengeluarkan alert berupa gambar 8.



Gambar 8. Hasil pengujian Sistem Alert batas Temperature 30°C dan batas Usage 1% pada Blynk

Namun apabila aplikasi *Blynk* dalam keadaan tertutup maka akan menghasilkan notifikasi seperti pada gambar 9.



Gambar 9. Hasil pengujian *Sistem Alert* batas *Temperature 30°C* dan batas *Usage 1%* pada *Blynk* pada *Smartphone*

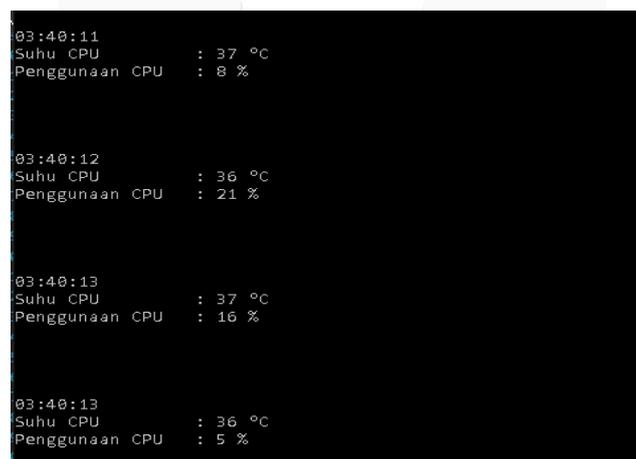
Analisis Pengujian *Sistem Alert*

Berdasarkan hasil pengujian yang telah dilakukan, sistem alert akan mengeluarkan notifikasi jika kondisinya telah terpenuhi, yang dimana kondisi di sini adalah melebihi batas dari *CPU temperature* dan *CPU Usage* pada komputer, sedangkan notifikasi tidak akan keluar jika kondisi yang dibutuhkan belum terpenuhi.

Pengujian Alat untuk Pengambilan Data *CPU Usage* dan *Temperature*

Pengambilan data dilakukan dengan menggunakan *library* dari *OpenHardwareMonitor*, yang dimana merupakan sebuah aplikasi monitoring *Open Source* yang memanfaatkan informasi yang diberikan oleh *Windows Management Instrumentation*.

Pengambilan data dilakukan menggunakan baudrate 9600. Suhu dalam satuan °C (derajat *celcius*) sedangkan penggunaan *CPU Usage* dalam % (*persen*).



Gambar 10. Hasil Pengambilan Data

Pengujian *Relay* dan *Fan* Tambahan

Pengujian *relay* dan *fan* tambahan ini bertujuan untuk mengetahui apakah *relay* dapat menghidupkan dan mematikan komputer sesuai dengan *CPU temperature* yang telah ditentukan sehingga dengan begitu dapat terlihat berfungsi dengan baik atau tidak. Pengujian dilakukan sebanyak 7 kali untuk setiap kondisi.

Tabel 9. Pengujian *Relay dan Fan*

Suhu (°C)	Usage/Load (%)	Kondisi Fan	Keterangan
<i>Low</i>	<i>Low</i>	Mati	Baik
<i>High</i>	<i>High</i>	Hidup	Baik
<i>Low</i>	<i>High</i>	Mati	Baik
<i>High</i>	<i>Low</i>	Hidup	Baik

Pengujian *Shutdown, Restart dan Sleep*

Pengujian dilakukan sebanyak 5 kali pada aplikasi *Blynk* untuk mengetahui apakah alat yang dibuat berfungsi dengan baik dan melakukan tugas yang telah diberikan dengan benar.

Tabel 10. Pengujian *Relay dan Fan*

Total Pengujian	Tombol	Keterangan	Kondisi
5x	<i>Shutdown</i>	Berhasil mematikan Komputer	Baik
5x	<i>Restart</i>	Berhasil merestart komputer	Baik
5x	<i>Sleep</i>	Berhasil <i>sleep</i> komputer	Baik

Analisis Pengujian Alat

- Analisis Pengujian Alat Pengambilan Data *CPU Usage* dan *Temperature*
Berdasarkan hasil pengujian yang telah dilakukan untuk fungsi pengambilan data *CPU Usage* dan *temperature*, kecepatan data yang dikeluarkan dan ditransfer adalah berdasarkan dari *baudrate* yang ditentukan, semakin tinggi *baudrate* yang ditentukan maka akan semakin cepat pembacaannya dan jika semakin rendah *baudrate* yang ditentukan maka akan semakin lambat pembacaannya.
- Analisis Pengujian *Relay dan Fan* Tambahan
Berdasarkan hasil pengujian yang telah dilakukan dapat disimpulkan
 - Relay* tidak akan menyalakan *fan* tambahan jika suhu dan *CPU Usage* masih dalam batas yang telah ditentukan.
 - Jika suhu dan *CPU Usage* sudah melebihi batas yang telah ditentukan *relay* akan menyalakan *fan* tambahan.
 - Jika suhu masih dalam batas yang telah ditentukan namun *CPU Usage* sudah melebihi batas yang telah ditentukan, *relay* tidak akan menyalakan *fan* tambahan.
 - Jika suhu sudah melebihi batas yang telah ditentukan, dan *CPU Usage* masih dalam batas yang telah ditentukan, maka *relay* akan menyalakan *fan* tambahan.
- Analisis Pengujian *Shutdown, Restart dan Sleep*
Berdasarkan hasil pengujian yang telah dilakukan dapat ditarik kesimpulan berupa :
 - Setelah menekan tombol *shutdown* pada aplikasi *Blynk*, maka komputer akan melakukan *shutdown* secara otomatis.
 - Setelah menekan tombol *restart* pada aplikasi *Blynk*, maka komputer akan mereboot dengan sendirinya.
 - Setelah menekan tombol *sleep* pada aplikasi *Blynk*, maka komputer akan memasuki status *hibernate* yang nanti dapat dinyalakan kembali dan melanjutkan aktivitas yang sebelumnya dilakukan pada komputer tanpa harus *restart* komputer kembali.

5. Kesimpulan

Berdasarkan pengujian yang telah dilakukan dalam penelitian ini maka dapat ditarik beberapa kesimpulan sebagai berikut:

- Alat yang dibangun berhasil mengambil data *CPU Usage* dan *temperature* pada komputer dan dapat melakukan pemantauan secara *real-time* melalui *smartphone*, ataupun dilihat pada *LCD* yang terpasang.

2. Sistem *alert* akan menyalakan notifikasi jika suhu komputer ataupun *CPU Usage* naik melebihi batas normal, sehingga dapat mempermudah pengguna dalam melakukan pemantauan tanpa harus melihat langsung pada komputer, dan dapat melakukan kontrol pada komputer untuk melakukan *shutdown*, *restart* ataupun *sleep* dengan menggunakan *smartphone* jika pengguna jauh dari komputer.
3. Alat yang dibangun dapat menghasilkan akurasi yang cukup besar yaitu akurasi terkecil 70% dan akurasi terbesar 100%, serta *error rate* yang dihasilkan kecil dimana *error rate* terkecil adalah 1,56% dan terbesar adalah 14,60% sehingga tidak memiliki rentang perbedaan terlalu besar antara alat yang dirancang dengan *software* monitoring *hardware* lainnya.
4. *Delay* transmisi yang dihasilkan kecil dan bagus yaitu 0,50 ms dengan menggunakan Telkomsel dan 0,28 ms dengan menggunakan Indihome 20Mbps. *Delay* transmisi yang dihasilkan antara pengiriman dapat dikatakan cepat sehingga dapat digunakan secara *real-time* saat proses pemantauan.

5.1 Saran

Berdasarkan penelitian yang telah dilakukan dalam penelitian ini maka terdapat beberapa hal yang dapat menjadi saran demi menyempurnakan penelitian ini

1. Dari hasil penelitian ini diketahui bahwa untuk membaca *delay* transmisi digunakan secara manual, sehingga belum sepenuhnya mengetahui keakuratan *delay* transmisi yang terjadi pada saat pengiriman data, disarankan untuk membaca *delay* transmisi secara otomatis sehingga dapat mengetahui keakuratan *delay* transmisi yang terjadi.
2. Saat pengujian *processor* yang dipakai adalah sebanyak 6 *core* secara bersamaan tetapi aplikasi *Blynk* hanya mengeluarkan data *temperature* secara bergantian. Untuk itu disarankan mencoba cara lain yang dapat mengeluarkan semua hasil pada setiap *core* yang berbeda secara bersamaan.
3. Pada saat pengujian waktu pengambilan data pada alat tidak sama dengan *software* lain. Untuk itu diperlukan penyamaan waktu dengan salah satu *software* monitoring untuk mendapatkan perbandingan akurasi yang lebih baik.
4. Dari hasil penelitian diketahui alat yang dibangun hanya dapat digunakan pada 1 komputer atau 1 server saja, disarankan mencoba bagaimana cara untuk dapat melakukan pemantauan untuk banyak server.

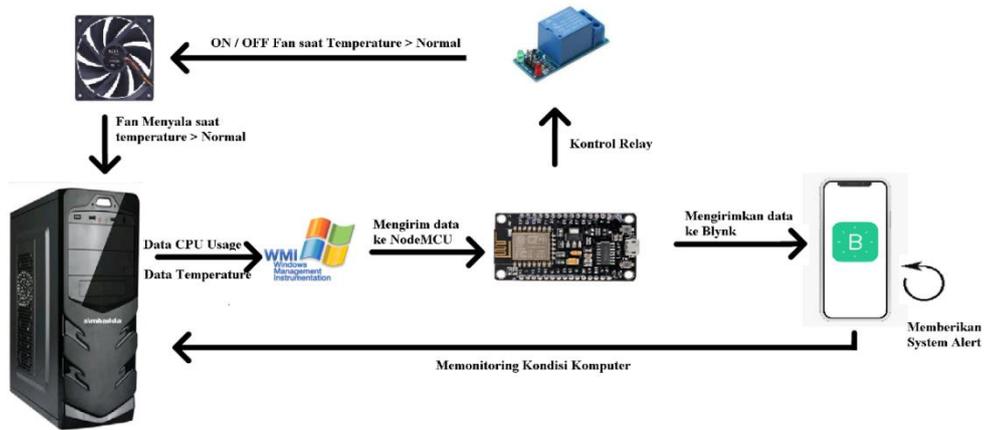
REFERENCE

- [1] H. Santoso, Ilmawan S. (2018). Sensor Monitoring Suhu Komputer Berbasis Port Paralel Dengan Menggunakan Rangkaian IC LM35 dan ADC 0804. *Jurnal Reaktom*, 3(2), 36-38.
- [2] E. Oktafiana. (2017). Tugas Jurnal Ilmiah Tentang Processor. Jurnal. Bogor: Fakultas Informasi Teknologi El-Rahma Education Centre.
- [3] M. Zainab Zsevril. (2018). Pendeteksi Suhu pada Perangkat Server Komputer Menggunakan Sensor DS18B20 dengan Spider Web View Berbasis Android. Skripsi. Medan: Fakultas Ilmu Komputer Dda Teknologi Informasi Universitas Sumatera Utara.
- [4] C. Huffman. (2014) . Windows Performance Analysis Field Guide. Elsevier Inc.
- [5] H. Dinda., Mutia R., dan Fajri Y. (2019). Penggunaan DHT11 dan Arduino Uno Sebagai Pendeteksi Suhu Pada Laptop. *Journal of Chemical Information and Modeling*. 53(9), 1689–1699.
- [6] T. Budioko, (2016). Sistem Monitoring Suhu Jarak Jauh Berbasis Internet Of Things Menggunakan Protokol MQTT. *Seminar Riset Teknologi Informasi (SRITI)*, 353–358.
- [7] F. Nurahmadi, (2011) “Sistem Kontrol dan Monitoring Suhu Jarak Jauh Memanfaatkan Embedded system Mikroprosesor W5100 dan ATmega8535,” *Indonesian J. Electron. Instrum (IJEIS)*, 1(2), 55-56.
- [8] M. I. Mahali, (2017). Smart Door Locks Based on Internet of Things Concept with mobile Backend as a Service. *Electronics, Informatics, and Vocational Education (Elinvo)*, 1(3), 171–181.

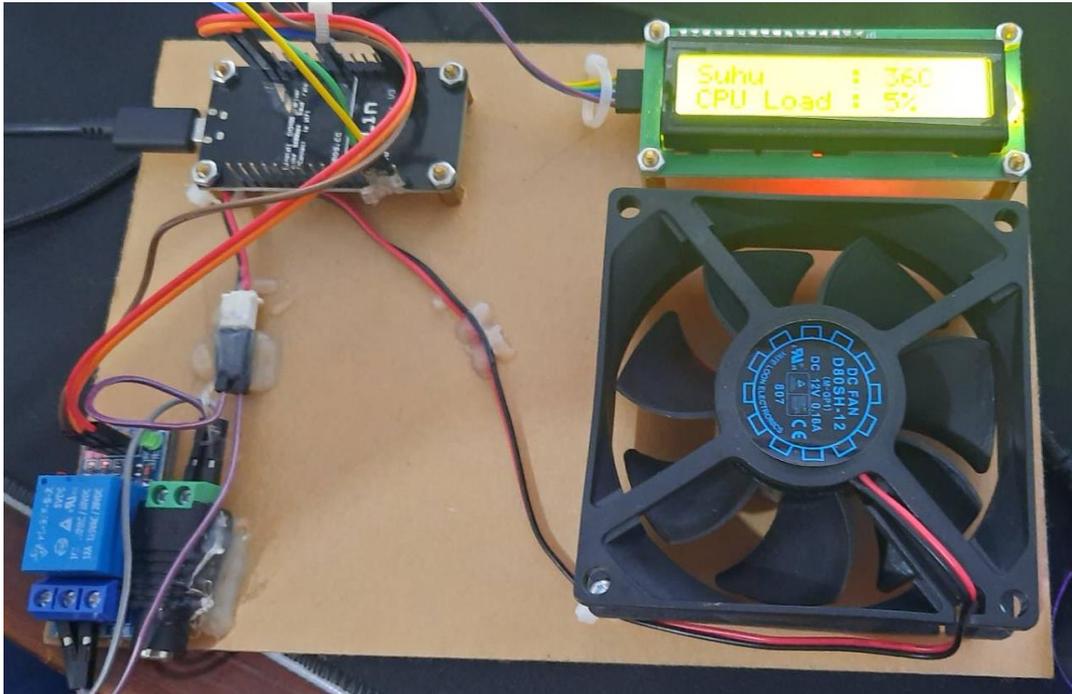
- [9] D. S. Hiral, M. S. Shah, Dan Umair. S A. S. (2017). Internet of Things (IoT): Integration of Blynk for Domestic Usability. *Vjer-Vishwakarma Journal of Engineering Research*, 1(4).
- [10] H. Khatri. (2020). Laporan Pengalaman Jaringan Seluler July 2020. [Online] Available at: <https://www.opensignal.com/in/reports/2020/07/indonesia/mobile-network-experience> [Accessed 1 December 2020].
- [11] Y. A. Pranata, Fibriani I., dan Satryo B. U. (2016). Analisis Optimasi Kinerja Quality of Service Pada Layanan Komunikasi Data Menggunakan Ns-2 Di Pt. PIn (Persero) Jember. *Sinergi*, 20(2), 149.
- [12] J. S. Schendel. 2013. *TRACE : Tennessee Research and Creative Exchange The Post-human Gamer : Reflections on Fieldwork in World of Warcraft*.



LAMPIRAN 1. Gambaran Umum Alat.



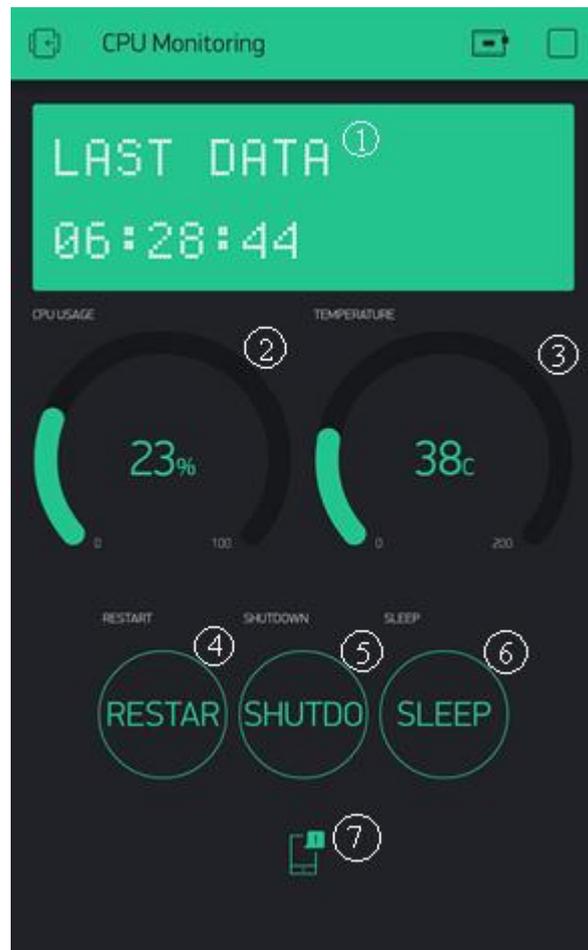
Lampiran 2. Tampilan Alat yang telah Dirakit.



Lampiran 3. Tampilan Pengujian Output Alat yang Dirakit.



Lampiran 4. Tampilan Blynk Apps yang akan Dipakai.



Lampiran 5. Tabel Pengujian Temperature dengan Alat yang diuji.

No	Core #0	Core #1	Core #2	Core #3	Core #4	Core #5	Satuan
1	38	37	37	36	37	36	°C
2	38	38	38	37	37	37	°C
3	38	38	37	37	37	36	°C
4	38	37	38	36	38	36	°C
5	38	37	37	36	37	40	°C
6	39	39	41	37	37	36	°C
7	37	37	37	36	37	36	°C
8	38	37	39	37	37	36	°C
9	39	38	39	37	37	37	°C
10	39	39	38	37	38	36	°C

Lampiran 6. Tabel Pengujian Temperature dengan program HWiNFO64.

No	Core #0	Core #1	Core #2	Core #3	Core #4	Core #5	Satuan
1	40	38	38	37	38	38	°C
2	39	38	39	38	39	38	°C
3	36	37	37	36	36	36	°C
4	39	41	39	37	39	38	°C
5	37	38	38	36	37	36	°C
6	47	37	39	37	37	36	°C
7	38	37	37	36	37	37	°C
8	38	39	40	37	39	38	°C
9	44	42	40	39	39	38	°C
10	38	39	37	35	37	36	°C

Lampiran 7. Tabel Pengujian Temperature dengan program HWMonitoring.

No	Core #0	Core #1	Core #2	Core #3	Core #4	Core #5	Satuan
1	37	38	37	36	36	36	°C
2	37	37	39	36	36	37	°C
3	38	37	36	37	36	38	°C
4	37	37	37	36	36	36	°C
5	37	37	37	37	39	37	°C
6	37	38	37	37	38	36	°C
7	39	40	45	37	38	39	°C
8	39	38	36	37	37	36	°C
9	38	36	37	37	37	37	°C
10	38	37	39	40	38	39	°C

Lampiran 8. Table Pengujian Temperature dengan program Aida64 Business.

No	Core #0	Core #1	Core #2	Core #3	Core #4	Core #5	Satuan
1	37	37	37	36	36	35	°C
2	38	37	40	37	38	36	°C
3	37	41	38	37	37	37	°C
4	38	38	37	37	38	39	°C
5	38	37	44	38	38	39	°C
6	37	36	36	39	38	38	°C
7	38	38	40	37	38	37	°C
8	48	38	40	37	38	37	°C
9	37	37	37	36	37	36	°C
10	38	39	37	37	38	37	°C

Lampiran 9. Tabel Pengujian CPU Usage.

No	Alat yang diuji	NZXT CAM	HWiNFO64	HWMonitor
1	13 %	15 %	15 %	13 %
2	15 %	16 %	17 %	17 %
3	16 %	13 %	16 %	16 %
4	14 %	15 %	15 %	14 %
5	12 %	15 %	12 %	13 %
6	14 %	16 %	14 %	16 %
7	15 %	12 %	15 %	13 %
8	15 %	14 %	15 %	14 %
9	14 %	12 %	14 %	14 %
10	15 %	13 %	17 %	13 %

Lampiran 10. Pengujian Delay menggunakan aplikasi Postman dan menggunakan provider telkomsel.

Data Ke-	Hasil Delay	Size Data yang dikirim	Satuan
1	140	131	ms
2	84	131	ms
3	58	131	ms
4	108	131	ms
5	50	131	ms
6	66	131	ms
7	75	131	ms
8	76	131	ms
9	51	131	ms
10	62	131	ms
11	73	131	ms
12	56	131	ms
13	138	131	ms
14	46	131	ms
15	84	131	ms
16	63	131	ms
17	59	131	ms
18	63	131	ms
19	50	131	ms
20	72	131	ms
21	55	131	ms
22	45	131	ms
23	48	131	ms
24	49	131	ms
25	54	131	ms
26	50	131	ms
27	71	131	ms
28	78	131	ms
29	74	131	ms
30	80	131	ms
31	47	131	ms
32	57	131	ms
33	65	131	ms
34	73	131	ms

35	78	131	ms
36	72	131	ms
37	81	131	ms
38	72	131	ms
39	65	131	ms
40	71	131	ms
41	43	131	ms
42	70	131	ms
43	44	131	ms
44	52	131	ms
45	73	131	ms
46	55	131	ms
47	68	131	ms
48	69	131	ms
49	47	131	ms
50	65	131	ms
51	74	131	ms
52	57	131	ms
53	50	131	ms
54	67	131	ms
55	53	131	ms
56	56	131	ms
57	52	131	ms
58	72	131	ms
59	43	131	ms
60	63	131	ms

Lampiran 11. Pengujian Delay menggunakan aplikasi Postman dan provider Indihome 20 Mbps.

Data Ke-	Hasil Delay	Satuan	Size Data yang dikirim
1	37	ms	131
2	44	ms	131
3	51	ms	131
4	44	ms	131
5	43	ms	131
6	37	ms	131
7	23	ms	131
8	37	ms	131
9	36	ms	131
10	37	ms	131
11	26	ms	131
12	24	ms	131
13	38	ms	131
14	43	ms	131
15	43	ms	131
16	40	ms	131
17	36	ms	131
18	21	ms	131
19	42	ms	131
20	47	ms	131
21	38	ms	131
22	38	ms	131
23	34	ms	131
24	37	ms	131
25	46	ms	131
26	52	ms	131
27	42	ms	131
28	37	ms	131
29	43	ms	131
30	34	ms	131
31	22	ms	131
32	22	ms	131
33	47	ms	131
34	42	ms	131

35	37	ms	131
36	37	ms	131
37	42	ms	131
38	23	ms	131
39	43	ms	131
40	35	ms	131
41	38	ms	131
42	45	ms	131
43	37	ms	131
44	40	ms	131
45	37	ms	131
46	38	ms	131
47	40	ms	131
48	37	ms	131
49	21	ms	131
50	36	ms	131
51	37	ms	131
52	24	ms	131
53	38	ms	131
54	38	ms	131
55	41	ms	131
56	38	ms	131
57	37	ms	131
58	49	ms	131
59	39	ms	131
60	39	ms	131