

---

## Perancangan Sistem Pendingin Laptop Otomatis Berbasis Arduino Menggunakan Sensor Suhu DHT11

Muhammad Syafirul Mukminin <sup>1)</sup>, Ansar Rizal <sup>2)</sup>, Dwi Titi Maesaroh <sup>3)</sup>

<sup>1</sup> D3 Teknik Komputer, Teknologi Informasi, Politeknik negeri Samarinda,  
Jl. Cipto Mangun Kusumo, Gunung Panjang, Kec. Samarinda Seberang, Kota Samarinda,  
Kalimantan Timur 75131, Indonesia  
e-mail: [Syafirulaja@gmail.com](mailto:Syafirulaja@gmail.com)<sup>1</sup>, [AnsarRizal@gmail.com](mailto:AnsarRizal@gmail.com)<sup>2</sup>,  
[dwititi.maesaroh@polnes.ac.id](mailto:dwititi.maesaroh@polnes.ac.id)<sup>3</sup>

---

### ABSTRAK

Penurunan performa laptop akibat kenaikan suhu ekstrem saat pengoperasian durasi lama sering menghambat produktivitas pengguna. Penelitian ini menghadirkan solusi teknis melalui perancangan dan pengujian sistem pendingin otomatis berbasis mikrokontroler Arduino Uno untuk mengatasi kendala suhu berlebih tersebut. Perangkat menggunakan sensor DHT11 sebagai instrumen utama pendeteksi fluktuasi suhu secara real-time. Mekanisme kerja sistem bermula ketika sensor mengirimkan data kenaikan suhu menuju unit Arduino untuk pengolahan perintah selanjutnya. Integrasi komponen relay berfungsi sebagai sakelar otomatis penggerak kipas DC 5V, sedangkan modul LCD I2C 16x2 menampilkan data suhu serta indikator operasional kipas bagi pengguna. Metodologi pengujian mencakup perakitan perangkat keras, pengembangan kode program, hingga verifikasi fungsionalitas pada perangkat keras laptop secara langsung. Hasil observasi laboratorium menunjukkan bahwa sensor DHT11 memiliki tingkat sensitivitas yang akurat dalam merespons perubahan suhu. Prototipe ini beroperasi secara presisi karena relay segera mengaktifkan unit kipas saat suhu mencapai ambang batas yang telah ditentukan. Sistem memutuskan aliran listrik ke kipas secara otomatis guna efisiensi energi sesaat setelah suhu kembali stabil pada level normal. Layar LCD menyajikan informasi numerik suhu dan status operasional perangkat secara kontinu tanpa kendala latensi data. Secara komprehensif, inovasi pendingin eksternal ini menunjukkan reliabilitas tinggi sebagai alternatif terjangkau dalam memelihara stabilitas kinerja laptop dari risiko panas berlebih.

Kata kunci: Kata kunci: Arduino Uno, DHT11, sistem pendingin laptop, relay, kipas otomatis.

### ABSTRACT

*It is always frustrating when a laptop suddenly slows down or lags just because it gets too hot from hours of work or gaming. This exact issue of overheating is what drove this research to find a practical solution by designing and testing an Arduino Uno-based automatic laptop cooling system. To monitor temperature shifts, the system relies on a DHT11 sensor. The workflow is straightforward yet effective: the moment the sensor detects a temperature rise, the data is instantly sent to the Arduino for processing. A relay is deliberately integrated as an automatic switch to turn on a 5V DC fan, while a 16x2 I2C LCD is placed upfront so users can monitor real-time temperature changes and fan status directly. The experiment was conducted hands-on, starting from assembling the hardware components and programming the software to real-world testing on a laptop. From a*

*series of laboratory tests, the DHT11 sensor proved to be sensitive enough to capture heat fluctuations. The system operated exactly as intended; when the laptop started heating up and hit a specific threshold, the relay immediately triggered the fan to spin. Interestingly, once the cool air from the fan managed to drop the laptop's temperature back to a safe limit, the relay automatically cut off the power, shutting down the fan to save energy. This dynamic process, including the temperature figures and the "ON/OFF" status of the fan, was clearly visible on the LCD screen without any noticeable delay. Overall, this additional cooling system prototype works effectively and could serve as an affordable alternative to maintain laptop performance stability against the threat of overheating.*

*Keywords: Laptop temperature, Arduino Uno, DHT11 sensor, Automatic fan, Overheat*

## PENDAHULUAN

Rasanya sulit membayangkan keseharian kita saat ini tanpa kehadiran laptop. Sadar atau tidak, perangkat ini sudah bergeser peran; dari yang dulunya dianggap barang mewah, kini menjadi kebutuhan pokok yang mutlak. Kita membutuhkannya untuk menyelesaikan tumpukan tugas kuliah, menuntaskan pekerjaan, atau sekadar mencari hiburan saat penat. Sayangnya, ada konsekuensi yang sering kali luput dari perhatian. Ketika laptop dipaksa bekerja berjam-jam terutama saat menjalankan perangkat lunak yang haus daya suhu di dalam komponen internalnya pasti akan melonjak drastis. Efeknya pun bisa langsung kita rasakan. Sistem mulai melambat, *lag* terjadi di mana-mana, dan jika kondisi ini terus dibiarkan, komponen perangkat keras di dalamnya terancam rusak sebelum waktunya [1]. Fenomena suhu berlebih atau *overheating* ini sebenarnya adalah masalah klasik. Akar persoalannya sering kali bersumber dari keterbatasan sistem pendingin bawaan pabrik yang kapasitasnya memang dirancang minimalis, sehingga kewalahan saat menghadapi beban kerja ekstra dalam durasi yang panjang. Bagi pengguna, ketidakstabilan performa ini jelas sangat mengganggu. Berangkat dari realitas tersebut, saya melihat bahwa laptop membutuhkan bantuan eksternal berupa sistem pendingin tambahan. Namun, intervensi ini tidak boleh merepotkan pengguna. Pendingin

tersebut harus mampu bekerja secara mandiri dan otomatis, bergerak dinamis mengikuti fluktuasi suhu perangkat agar proses pembuangan panas menjadi lebih presisi sekaligus efisien [2].

Kabar baiknya, perkembangan teknologi mikrokontroler saat ini sudah sangat pesat, menawarkan solusi yang tidak hanya murah tetapi juga mudah untuk dimodifikasi. Di kalangan mahasiswa, papan Arduino Uno menjadi salah satu opsi paling populer karena harganya yang ramah di kantong dan ekosistemnya yang terbuka. Melalui modul mikro ini, kita dapat membangun sebuah sistem otomasi sederhana. Langkah pertamanya adalah mengintegrasikan sensor DHT11, yang bertugas sebagai komponen pemantau untuk membaca fluktuasi suhu bodi laptop secara langsung dan *real-time* [3]. Logika program yang tertanam di dalam Arduino kemudian akan memerintahkan modul relay. Komponen ini bertindak selaku saklar elektronik yang memutuskan atau mengalirkan arus listrik menuju kipas secara otomatis. Agar sistem ini tidak terkesan bekerja secara misterius atau berada dalam kotak hitam (*black box*), sebuah layar LCD I2C 16x2 sengaja ditambahkan. Kehadiran layar ini penting sebagai media visual interaktif, sehingga pengguna bisa mengetahui secara pasti berapa angka suhu laptop mereka pada detik tersebut [4].

Jika kita membedah literatur dan beberapa penelitian terdahulu, kolaborasi

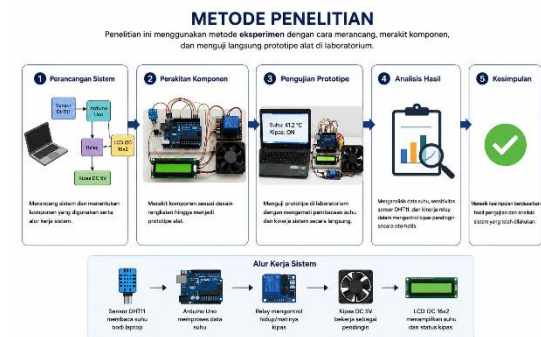
antara sensor suhu lingkungan dan mikrokontroler memang sudah teruji keandalannya untuk urusan monitoring maupun kendali otomatis. Rekam jejak keberhasilan inilah yang memantapkan langkah saya untuk merancang sebuah prototipe pendingin laptop otomatis berbasis Arduino Uno. Struktur alat yang dibangun dalam penelitian ini mengintegrasikan lima komponen esensial: Arduino Uno sebagai pusat pengendali logika, sensor DHT11 sebagai pemindai hawa panas, relay untuk kendali daya listrik, LCD I2C untuk indikator status, serta kipas DC 5V sebagai eksekutor utama yang mengembuskan udara dingin[5].

Secara teknis, mekanisme alat ini dirancang untuk mendeteksi suhu tepat di bagian bawah bodi laptop, area di mana panas biasanya paling banyak terjebak. Data mentah berupa besaran fisis dari sensor akan dikirim ke pin input Arduino, diproses melalui kode pemrograman, lalu dievaluasi untuk menentukan kapan kipas harus mulai berputar. Mengacu pada metodologi eksperimen kendali suhu yang pernah dikembangkan oleh Baldwin dan Solihin, pengujian langsung di meja praktik menghasilkan catatan data numerik yang cukup menarik[6]. Suhu awal laptop berada di angka 30.4°C, namun perlahan merangkak naik hingga menyentuh 33.7°C akibat beban kerja komputasi sistem. Di sinilah sistem otomasi menunjukkan responsnya. Begitu sensor mendeteksi suhu merayap naik ke angka 32.5°C, Arduino langsung memicu relay untuk menyalakan kipas eksternal. Tiupan angin yang dihasilkan terbukti efektif. Tidak berselang lama setelah kipas aktif, suhu bodi laptop berhasil diredam turun kembali ke angka 31.7°C. Merujuk pada analisis komparasi kestabilan termal mikrokontroler oleh Setyabudi dan Suwahyadi, penurunan suhu ke titik aman ini secara otomatis memicu perintah baru bagi relay untuk memutus arus, sehingga kipas mati

dengan sendirinya tanpa perlu campur tangan manusia[7].

Melalui seluruh tahapan perancangan dan pengujian di atas, tujuan inti dari penelitian ini adalah untuk membuktikan sekaligus mengukur tingkat keandalan dari sistem pendingin otomatis yang dibuat. Harapan yang ingin dicapai sebenarnya sederhana namun berdampak nyata. Lewat kehadiran alat ini, risiko kerusakan akibat *overheating* dapat ditekan sekecil mungkin, performa laptop tetap konsisten di titik optimal, dan kita sebagai pengguna dapat bekerja dengan tenang tanpa perlu khawatir perangkat tiba-tiba melambat di saat-saat krusial.

## METODE PENELITIAN



Gambar1. Metode Penelitian dan rangka

Berdasarkan gambar Penelitian di atas, penelitian ini menggunakan metode eksperimen, yang dilakukan dengan cara merancang, merakit komponen, dan menguji langsung prototipe alat di laboratorium. Fokus utamanya adalah melihat sensitivitas sensor DHT11 dalam membaca perubahan suhu bodi laptop serta memastikan kemampuan relay dalam mengontrol kipas pendingin secara otomatis berdasarkan parameter yang telah ditentukan.

Sistem yang dibangun ini mengandalkan beberapa komponen utama yang saling terintegrasi. Arduino Uno dipilih sebagai pusat kendali untuk memproses data suhu dari sensor DHT11. Output dari Arduino kemudian mengontrol relay selaku saklar elektronik

untuk menggerakkan kipas DC 5V sebagai eksekutor pendingin, sementara modul LCD I2C 16x2 digunakan untuk menampilkan data suhu dan status kipas secara *realtime*.



Gambar2. Kerangka Kerja Penelitian

Berdasarkan gambar diatas secara praktis, prosedur penelitian ini dibagi menjadi tiga tahapan terstruktur:

- 1. Perancangan Perangkat Keras (Hardware):** Menghubungkan seluruh komponen fisik (Arduino, sensor DHT11, relay, LCD, dan kipas) di atas papan rangkaian menggunakan kabel *jumper*.
- 2. Perancangan Perangkat Lunak (Software):** Menyusun baris kode pemrograman menggunakan Arduino IDE dengan bahasa C/C++. Logika dasarnya adalah memerintahkan program untuk membaca input sensor, memperbarui tampilan LCD, dan memicu aktivitas relay saat suhu menyentuh batas tertentu.
- 3. Pengujian Sistem** Tahap akhir dari kerangka kerja ini adalah menguji keandalan alat melalui skenario nyata. Sensor DHT11 ditempelkan secara presisi tepat pada area bawah bodi laptop titik kritis tempat menumpuknya hawa panas komputasi. Prosedur pengujian fungsional ini dilakukan untuk memvalidasi performa input dan output sistem secara simultan: Metode Pengujian Input: Suhu laptop diubah secara bertahap (kondisi dingin → normal →

panas) akibat aktivitas kerja perangkat. Evaluasi dilakukan untuk memastikan keakuratan sensor dalam menangkap fluktuasi panas bodi laptop tanpa penundaan (*lag*).

Metode Pengujian Output: Mengamati respon mekanis dari sirkuit kontrol. Saat suhu berada di bawah parameter batas (Kondisi Dingin), relay dipastikan tetap *OFF* dan kipas mati. Namun, begitu suhu melonjak naik melewati parameter kritis (Kondisi Panas), modul relay harus langsung beralih ke posisi *ON*, memutar kipas secara instan, dan memperbarui status pada tampilan teks LCD.

Prototipe ini dinyatakan sukses dan bekerja dengan baik apabila seluruh fungsi input (pembacaan akurat), proses (logika Arduino), dan output (kendali relay, putaran kipas, serta visualisasi LCD) dapat beroperasi secara berulang, otomatis, dan stabil di bawah parameter operasional yang sudah direncanakan.

Fase awal pengerjaan fisik dimulai dengan merakit perangkat keras (*hardware*). Langkah ini mengintegrasikan papan mikrokontroler Arduino Uno sebagai unit kendali utama dengan komponen-komponen penunjang lainnya di atas papan rangkaian (*breadboard*) menggunakan bantuan kabel *jumper*. Konfigurasi ini sangat membantu dalam menyusun rangkaian secara fleksibel tanpa perlu melakukan penyolderan permanen. Adapun daftar komponen sirkuit beserta fungsi spesifiknya dijabarkan secara rinci pada tabel dibawah ini:

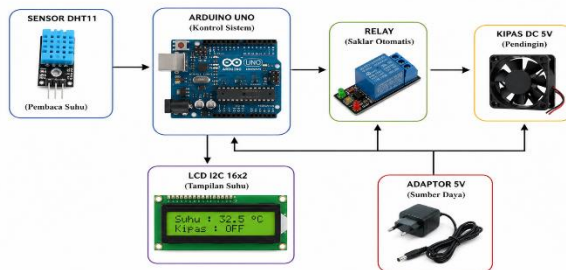
Tabel 1 : Daftar Komponen

NO	Nama Komponen	FUNGSI
1	Arduino Uno	Mikrokontroler Utama

2	Sensor DHT11	Membaca suhu
3	LCD	Menampilkan Suhu
4	Relay	Mengontrol Kipas
5	Kipas DC 5V	Pendingin Laptop
6	Adaptor	Sumber daya
7	BreadBoard	Tempat Rangkaian
8	Kabel Jumper	Penghubung Komponen

Mengingat beban kerja komponen elektrik ini cukup konstan selama proses monitoring berlangsung, sistem disuplai oleh daya terpusat dari luar berupa adaptor 5V DC untuk menjamin stabilitas kinerja mikrokontroler dan mencegah terjadinya penurunan tegangan (*drop voltage*). Interkoneksi data dan distribusi daya listrik ini bekerja mengadopsi siklus kendali tertutup yang terbagi menjadi tiga fase utama, yaitu input, proses, dan output, sebagaimana dicermati pada skema diagram blok sirkuit

DIAGRAM BLOK SISTEM

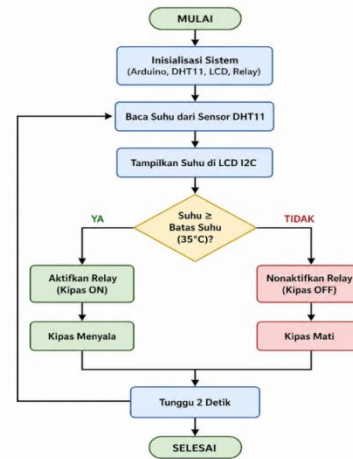


Gambar 1 Diagram Blok

Alur kerja dimulai dari sisi input, di mana sensor DHT11 bertugas mengendus temperatur aktual secara *real-time* tepat di area bawah bodi laptop titik kritis tempat terjebaknya akumulasi hawa panas komputasi. Data fisis hasil pembacaan suhu tersebut kemudian dikirim ke papan Arduino Uno untuk diterjemahkan menjadi informasi digital pada fase proses. Begitu datanya siap dievaluasi, Arduino melempar data temperatur tersebut ke layar LCD I2C 16x2 agar pengguna dapat memantau pergerakan suhu secara langsung.

Inti dari otomatisasi sistem ini berjalan ketika data digital masuk ke dalam logika program perangkat lunak (*software*) menggunakan bahasa

pemrograman C/C++ pada Arduino IDE. Alur skematik pengambilan keputusan logis oleh program dirancang mengikuti bagan *flowchart* pada Gambar 2:



Gambar 2 flowchart Sistem

Berdasarkan bagan alur di atas, program diawali dengan inisialisasi komponen, diikuti dengan pembacaan suhu oleh sensor DHT11 untuk ditampilkan pada LCD I2C. Selanjutnya, Arduino secara konstan membandingkan data suhu tersebut dengan ambang batas aman yang sudah ditentukan di dalam program. Jika hawa panas bodi laptop melonjak dan menyentuh atau melewati parameter batas kritis tersebut ( $Suhu > \text{Batas}$ ), Arduino mengirimkan sinyal digital *HIGH* untuk menutup sirkuit saklar pada modul relay (*Relay ON*). Arus listrik akan mengalir secara instan dan kipas DC 5V berputar untuk mengembuskan udara dingin ke area bodi laptop. Sebaliknya, saat sirkulasi udara dari kipas berhasil meredam hawa panas hingga temperatur laptop turun kembali ke zona aman di bawah ambang batas, Arduino mengeksekusi kondisi *Relay OFF* untuk memutus kontak saklar. Kipas pun otomatis berhenti berputar demi efisiensi konsumsi energi listrik, dan seluruh siklus pengujian input-output ini berjalan secara berulang untuk memantau kestabilan performa alat.

Tabel 2: Koneksi Pin

KOMPONEN	PIN KOMPONEN	PIN ARDUINO
	VCC (+)	5V

DHT11	GND (-)	GND
	DATA (OUT)	D2
LCD 16x2 I2C	VCC	5V
	GND	GND
	SDA	A4
	SCL	A5
RELAY	VCC	5V
	GND	GND
	IN	D7

**HASIL DAN PEMBAHASAN**

**1. Hasil Perancangan Sistem**

Dari perancangan yang sudah dilakukan, bisa disimpulkan bahwa sistem pendingin otomatis ini berfungsi dengan baik dan sesuai ekspektasi. Indikator keberhasilan ini terlihat dari bagaimana sensor DHT11 bekerja; ia tidak hanya membaca suhu lingkungan laptop secara aktual, tapi juga berhasil menjembatani data tersebut ke LCD 16x2 untuk ditampilkan secara *real-time*. Selain itu, integrasi pada bagian *output* juga berjalan lancar. Arduino Uno terbukti mampu mengomandoi relay dan kipas DC 5V secara otomatis, menyesuaikan putaran kipas dengan naik-turunnya suhu yang terdeteksi.



Berdasarkan dokumentasi pada layar LCD, sistem terbukti mampu menampilkan data perubahan suhu secara *real-time*. Saat pengujian dilakukan, sensor membaca suhu lingkungan berada di angka 30.4°C. Bisa dilihat bahwa dalam kondisi tersebut, status kipas yang tertera masih dalam keadaan "MATI". Hal ini terjadi karena suhu 30.4°C

rupanya belum menyentuh atau melewati ambang batas minimal yang kita atur di dalam program untuk mengaktifkan sistem pendingin.



Ketika pengujian dilanjutkan dan suhu menyentuh angka 32.00°C, sistem langsung merespons dengan mengaktifkan kipas secara otomatis. Perubahan status ini juga terekam pada baris kedua layar LCD yang kini menampilkan keterangan "NYALA". Respons cepat ini membuktikan bahwa instruksi kondisional di dalam program Arduino sudah berjalan dengan tepat, di mana kipas akan langsung bekerja mengeksekusi pendinginan begitu suhu mencapai suhu yang telah ditentukan.

**2. Hasil Pengujian Pembacaan Sensor Dht11**

Pengujian terhadap sensor DHT11 ini sengaja dilakukan untuk menguji sensitivitas pembacaannya. Kita perlu tahu bagaimana performa sensor saat mendeteksi naik-turunnya suhu laptop dalam berbagai skenario penggunaan. Sebagai acuan analisis, variasi data suhu yang berhasil ditangkap oleh sensor selama pengujian telah dikelompokkan secara detail ke dalam tabel berikut.

WAKTU (Detik)	SUHU	STATUS RELAY	KIPAS	TAMPILAN LCD
0	30.4°C	OFF	MATI	SUHU: 30.4°C KIPAS: MATI
30	30.8°C	OFF	MATI	SUHU: 30.8°C KIPAS: MATI

60	31.1°C	OFF	MATI	SUHU: 31.1°C KIPAS: MATI
90	32.5°C	ON	NYALA	SUHU: 32.5°C KIPAS: NYALA
120	31.7°C	OFF	MATI	SUHU: 31.7°C KIPAS: MATI
150	32.2°C	ON	NYALA	SUHU: 32.2°C KIPAS: NYALA
180	32.7°C	ON	NYALA	SUHU: 32.7°C KIPAS: MATI
210	32.4°C	ON	NYALA	SUHU: 32.4°C KIPAS: NYALA
240	32.9°C	ON	NYALA	SUHU: 32.9°C KIPAS: NYALA
270	33.7°C	ON	NYALA	SUHU: 33.7°C KIPAS: NYALA
300	32.6°C	ON	NYALA	SUHU: 32.6°C KIPAS: NYALA

Jika mencermati data pada tabel di atas, terlihat jelas bagaimana dinamika sistem pendingin ini bekerja dalam menjaga suhu laptop. Pada detik awal hingga detik ke-60, sensor DHT11 membaca kenaikan suhu secara bertahap dari 30.4°C ke 31.1°C, di mana posisi kipas dan relay masih mati karena belum menyentuh angka *threshold*. Titik balik terjadi pada detik ke-90 saat suhu melonjak ke 32.5°C; di momen inilah relay langsung berpindah ke kondisi *ON* dan memicu kipas untuk menyala.

Menariknya, efek pendinginan langsung terlihat di detik ke-120 saat suhu berhasil ditekan turun ke angka 31.7°C, yang otomatis membuat sistem kembali mematikan kipas. Namun, karena laptop terus dipacu untuk bekerja dan menghasilkan panas konstan, suhu kembali merangkak naik pada detik-detik berikutnya hingga menyentuh puncaknya di 33.7°C pada detik ke-270. Siklus naik-

turun ini membuktikan bahwa sistem tidak hanya sekedar membaca suhu secara *real-time*, tetapi juga sukses mengeksekusi logika saklar otomatis pada kipas sesuai dengan batas parameter yang telah direncanakan.

## KESIMPULAN DAN SARAN

### Kesimpulan

Tahapan perancangan dan pengujian menunjukkan bahwa sistem pendingin laptop otomatis berbasis Arduino Uno dengan sensor DHT11 berfungsi optimal sesuai target riset. Sensor DHT11 mendeteksi fluktuasi suhu pada area bawah laptop dan menyajikan data tersebut melalui layar LCD secara waktu nyata. Perangkat ini pun mengendalikan operasional relay serta kipas pendingin secara mandiri berdasarkan parameter ambang suhu yang telah ditetapkan.

Relay menggerakkan kipas pendingin secara otomatis begitu suhu laptop mencapai ambang batas aktivasi. Mekanisme ini membantu proses pelepasan panas pada perangkat secara langsung. Relay segera memutus aliran dan mematikan kipas otomatis saat suhu kembali menurun. Dinamika fluktuasi suhu selama pengujian membuktikan kemampuan sistem dalam merespons kondisi termal laptop secara presisi. Hasil tersebut menunjukkan bahwa seluruh rangkaian bekerja secara optimal sesuai dengan prosedur pendinginan yang sedang berlangsung.

Hasil pengujian menunjukkan bahwa sistem pendingin otomatis ini dapat membantu menjaga kestabilan suhu laptop dan mengurangi panas berlebih selama penggunaan. Selain itu, penggunaan LCD memudahkan pengguna dalam memantau kondisi suhu dan status kipas secara langsung. Dengan demikian, sistem yang dirancang dapat bekerja secara efektif sebagai pendingin laptop otomatis berbasis Arduino Uno dan sensor DHT11.

## Saran

Melihat dari hasil evaluasi *prototype* yang sudah dibuat, saya rasa ada beberapa catatan penting yang bisa dijadikan rekomendasi untuk penelitian selanjutnya. Poin pertama yang cukup krusial adalah soal pemilihan sensor. Sensor DHT11 yang dipakai sekarang rentang pengukuran derajat suhunya masih terbatas. Selain itu, kalau kondisi lingkungan di sekitar laptop berubah terlalu drastis, sensor ini agak rentan mengalami *error* pembacaan. Jadi, untuk ke depannya, akan jauh lebih bagus kalau spesifikasi sensor ditingkatkan ke jenis DHT22 atau LM35 agar data temperatur yang didapat bisa lebih akurat dan stabil.

## DAFTAR PUSTAKA

- [1]. B. R. Challapureddy, M. Paleti, and R. Jilte, "Advancement of Cooling Methods in Laptops," pp. 345–352, Jan. 2024, doi: 10.1007/978-981-99-8343-8\_32.
- [2]. B. R. Challapureddy, M. Paleti, and R. Jilte, "Advancement of Cooling Methods in Laptops," pp. 345–352, Jan. 2024, doi: 10.1007/978-981-99-8343-8\_32.
- [3]. M. S. Batubara and P. H. Hutabarat, "Design of Automatic Fan Base On Arduino Uno Microcontroller And DHT11 Sensor," *Jurnal Teknologi dan Open Source*, Dec. 2025, doi: 10.36378/jtos.v8i2.5012.
- [4]. S. Mahalakshmi, B. Mohan, N. A. Latha, J. R. F. Raj, K. Poonkodi, and A. Madhumathi, "Real-Time Smart Fan Speed Adjustment with Buzzer Alerts for Environmental Temperature," pp. 1589–1594, Aug. 2025, doi: 10.1109/iccpct65132.2025.11176679. *Jurnal Tekno*, vol. 15, no. 2, hal. 89-94, 2018.
- [5]. M. S. Batubara and P. H. Hutabarat, "Design of Automatic Fan Base Kendala lain yang saya rasakan selama pengujian adalah penggunaan modul relay untuk mengontrol kipas. Mekanisme *on-off* konvensional seperti ini terasa kurang efisien. Tiap kali kipas menyala, relay akan memicu putaran maksimal secara instan, yang akhirnya malah menimbulkan suara bising konstan. Sebagai solusi teknis, sistem pendingin ini ke depannya bisa dikembangkan menggunakan metode PWM (*Pulse Width Modulation*) dengan bantuan transistor atau MOSFET. Lewat metode ini, kecepatan putaran kipas bias dia diatur agar bergerak dinamis bisa melambat atau melaju kencang mengikuti fluktuasi suhu laptop. On Arduino Uno Microcontroller And DHT11 Sensor," *Jurnal Teknologi dan Open Source*, Dec. 2025, doi: 10.36378/jtos.v8i2.5012.
- [6]. Y. Sun, "Design of locally cooled step-less fan for industrial use using Arduino programming," *Applied and Computational Engineering*, Aug. 2024, doi: 10.54254/2755-2721/81/2024bj0052.
- [7]. M. S. Batubara and P. H. Hutabarat, "Design of Automatic Fan Base On Arduino Uno Microcontroller And DHT11 Sensor," *Jurnal Teknologi dan Open Source*, Dec. 2025, doi: 10.36378/jtos.v8i2.5012.
- [8]. K. Bangur, "Cooling of laptop CPUs: study on heat transfers and blade designs.," vol. 5, no. 2, pp. 10–20, Sept. 2025, doi: 10.36676/jmk.v5.i2.55.
- [9]. H. K. Siregar dan M. Azmi, "Pemanfaatan Transistor MOSFET Sebagai Driver Kecepatan Motor DC Menggunakan Sinyal PWM Arduino," *Jurnal Sains dan Teknologi*, vol. 13, no. 1, hal. 28-35, 2023.

- [10]. N. K. T. Nastiti, "Rancang Bangun Pengendali Suhu Perangkat Keras Komputer Berbasis Arduino Uno," *Jurnal Ilmiah Pendidikan Teknik Elektro (Vektor)*, vol. 4, no. 2, hal. 99-106, 2020.
- [11]. J. J. S. P. Putra dan I. G. A. P. R. Agung, "Sistem Monitoring Suhu Real-Time Menggunakan Modul I2C LCD 16x2 Berbasis Mikrokontroler," *Jurnal Spektrum*, vol. 7, no. 4, hal. 52-59, 2020.
- [12]. A. S. R. M. Sinaga, "Penggunaan Komponen Relay Sebagai Pengendali Beban AC dan DC Berbasis Arduino," *Jurnal Informatika dan Komputer*, vol. 5, no. 2, hal. 142-148, 2020.
- [13]. M. G. S. Wibowo dan A. Ashari, "Penerapan Sistem Otomasi Kontrol Kipas pada Perangkat Elektronik Guna Mencegah Overheating," *Jurnal Teknologi Informasi*, vol. 15, no. 1, hal. 12-20, 2021.
- [14]. R. F. S. Budi dan T. Sutabri, "Rancang Bangun Prototipe Sistem Pendingin Otomatis pada Ruang Server Menggunakan Arduino Uno," *Jurnal Sistem Informasi dan Ilmu Komputer*, vol. 2, no. 3, hal. 204-211, 2024.
- [15]. "Device Components' Ambient Temperature Monitoring Using Arduino Uno for Efficient Feedback Cooling System," Jan. 2025, doi: 10.13140/rg.2.2.21768.74245.