

Perancangan Otomatisasi Manajemen Energi Restoran Berbasis IoT Dengan Algoritma Partikel Swarm Optimization

Hadista Imannur Muhammad^{1*}, Sofyan Mufti Prasetyo²

¹Program Studi Teknik Informatika, Universitas Pamulang, Jl. Raya Puspatek, Buaran, Kec.
Pamulang, Kota Tangerang Selatan, Banten 15310, Indonesia
e-mail: ¹hadistaplaymaker811@gmail.com, ²dosen01809@unpam.ac.id

*Corresponding Author

Submitted Date: November 06, 2025
Revised Date: May 05, 2026

Reviewed Date: April 23, 2026
Accepted Date: May 05, 2026

Abstract

This study aims to design an automated energy management system for restaurants by utilizing Internet of Things (IoT) technology and the Particle Swarm Optimization (PSO) algorithm. The developed system integrates IoT devices in the form of sensors and actuators to monitor and control energy consumption in real time. Data collected from various sensors are analyzed and optimized using the PSO algorithm to produce efficient energy management configurations, thereby reducing energy waste and lowering operational costs. The test results indicate that the designed automation system is capable of detecting energy usage patterns with high accuracy and providing optimal configuration recommendations in accordance with the restaurant's environmental conditions. Therefore, this research makes a significant contribution to improving energy efficiency and advancing IoT-based automation technology in the restaurant sector.

Keywords: Internet of Things (IoT), Particle Swarm Optimization (PSO), Energy Management Automation, Restaurant, Sensors, Actuators, Energy Optimization, Energy Efficiency

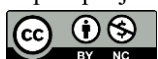
Abstrak

Penelitian ini bertujuan untuk merancang sistem otomasi manajemen energi pada restoran dengan memanfaatkan teknologi Internet of Things (IoT) dan algoritma Particle Swarm Optimization (PSO). Sistem yang dikembangkan mengintegrasikan perangkat IoT berupa sensor dan aktuator untuk memantau serta mengendalikan konsumsi energi secara real-time. Data yang diperoleh dari berbagai sensor dianalisis dan dioptimalkan menggunakan algoritma PSO guna menghasilkan konfigurasi pengaturan energi yang efisien, sehingga dapat mengurangi pemborosan energi dan menurunkan biaya operasional. Hasil pengujian menunjukkan bahwa sistem otomasi yang dirancang mampu mendeteksi pola penggunaan energi dengan akurasi tinggi serta memberikan rekomendasi pengaturan yang optimal sesuai dengan kondisi lingkungan restoran. Dengan demikian, penelitian ini memberikan kontribusi signifikan dalam peningkatan efisiensi energi dan pengembangan teknologi otomasi berbasis IoT di sektor restoran.

Kata Kunci: Internet of Things (IoT), Particle Swarm Optimization (PSO), Otomasi Manajemen Energi, Restoran, Sensor, Aktuator, Optimalisasi Energi, Efisiensi Energi

1. Pendahuluan

Teknologi Internet of Things (IoT) telah berkembang pesat dan menawarkan solusi inovatif dalam pengelolaan energi secara otomatis dan efisien (Alterazi dkk., 2022). Mikrokontroler seperti Arduino yang mudah diprogram dan terintegrasi dengan sensor PIR (Passive Infrared) memungkinkan pengumpulan data aktivitas pengguna secara real-time. (Romadhoni & Wahyudi, 2023) Sensor PIR mendeteksi kehadiran manusia yang dapat menjadi indikator kebutuhan penggunaan perangkat listrik di sebuah restoran.



Dengan dukungan platform IoT seperti Blynk, data aktivitas dan status perangkat dapat dipantau dan dikendalikan secara jarak jauh melalui aplikasi smartphone. Hal ini memberikan fleksibilitas dan kemudahan dalam manajemen energi secara real-time.

Selain itu, algoritma Particle Swarm Optimization (PSO) yang dikenal efektif untuk optimasi berbagai sistem dapat diimplementasikan secara sederhana di Arduino untuk mengatur jadwal nyala-mati perangkat listrik berdasarkan data aktivitas (Ahmad Putra dkk., 2023). Meskipun Arduino memiliki keterbatasan resource, versi PSO sederhana dapat dijalankan langsung tanpa perlu menggunakan bahasa pemrograman tambahan seperti Python.

Berbagai penelitian telah menunjukkan keberhasilan penggunaan PSO dalam optimasi (Habiby dkk., 2021). energi pada sistem kelistrikan dan pengelolaan beban listrik. Dengan mengintegrasikan Arduino, sensor PIR, Blynk, dan PSO sederhana, sistem otomatisasi manajemen energi restoran dapat diimplementasikan untuk meningkatkan efisiensi penggunaan energi dan menekan biaya operasional (Suri Tauladan & Novia Syamsir, 2022).

2. Metodologi Penelitian

Penelitian ini dilakukan melalui beberapa tahapan yang saling berkaitan, dimulai dari analisis kebutuhan, perancangan sistem, implementasi, hingga pengujian dan evaluasi. Tujuannya adalah untuk merancang sistem otomatisasi manajemen energi restoran berbasis **Internet of Things (IoT)** yang dapat mengatur penggunaan energi secara efisien (Nihayatus Saadah dkk., 2025). dengan bantuan **mikrokontroler Arduino, sensor PIR, dan algoritma Particle Swarm Optimization (PSO)** (Subekti dkk., 2024).

2.1. Analisis Kebutuhan Sistem

Tahap awal penelitian ini diawali dengan mengidentifikasi kebutuhan baik dari sisi perangkat keras maupun perangkat lunak. Komponen utama yang digunakan meliputi Arduino Uno sebagai pengendali utama, sensor PIR untuk mendeteksi aktivitas manusia, modul relay sebagai saklar otomatis untuk perangkat listrik, serta ESP32 yang berfungsi menghubungkan sistem ke jaringan internet. Dari sisi perangkat lunak, digunakan Blynk sebagai platform monitoring dan pengendalian sistem secara real-time melalui aplikasi smartphone.

2.2. Perancangan Sistem

Pada tahap ini, sistem dirancang agar dapat bekerja secara otomatis berdasarkan aktivitas pengguna di area restoran. Ketika sensor PIR mendeteksi adanya gerakan, data dikirimkan ke Arduino untuk diproses. Arduino kemudian menjalankan algoritma PSO untuk mencari waktu terbaik dalam menyalakan atau mematikan perangkat listrik. (Piu dkk., 2025) Semua data dan status sistem dikirimkan ke aplikasi Blynk, sehingga pengguna dapat memantau dan mengontrol sistem dengan mudah dari jarak jauh. (Gad, 2022)

2.3. Implementasi Algoritma PSO

Algoritma Particle Swarm Optimization digunakan karena mampu melakukan optimasi secara sederhana namun efektif. Dalam penelitian ini, PSO diimplementasikan langsung di Arduino dengan versi yang disesuaikan agar ringan dan efisien. Setiap partikel dalam algoritma mewakili nilai waktu tunda (delay) yang dioptimalkan untuk menentukan kapan perangkat listrik sebaiknya aktif atau nonaktif (Ahmad Putra dkk., 2023). Proses iterasi dilakukan hingga diperoleh hasil yang paling efisien berdasarkan aktivitas pengguna. (Faridha & Dewiani, 2024)

2.4. Monitoring dan Kontrol Menggunakan Blynk

Aplikasi Blynk berperan penting dalam sistem ini sebagai media penghubung antara pengguna dan perangkat. Melalui aplikasi tersebut, pengguna dapat melihat kondisi sensor, status perangkat, serta hasil perhitungan algoritma PSO secara langsung (Romadhoni & Wahyudi, 2023). Selain itu, pengguna juga dapat memberikan perintah manual, misalnya menyalakan atau mematikan lampu dari jarak jauh. Dengan demikian, sistem ini tidak hanya otomatis tetapi juga interaktif dan mudah dioperasikan.

2.5. Pengujian dan Evaluasi

Tahap terakhir adalah melakukan pengujian untuk memastikan sistem bekerja sesuai harapan. Pengujian dilakukan pada lingkungan simulasi restoran untuk melihat bagaimana sistem merespons aktivitas pengguna, seberapa cepat sensor mendeteksi gerakan, dan seberapa optimal PSO dalam mengatur waktu pengoperasian perangkat. Hasil

pengujian kemudian dibandingkan dengan kondisi tanpa sistem otomatis untuk melihat perbandingan efisiensi energi yang dihasilkan.

3. Hasil dan Diskusi

3.1. Analisis Kebutuhan Sistem

Sistem otomasi manajemen energi restoran berbasis IoT dengan algoritma Particle Swarm Optimization (PSO) membutuhkan perangkat keras dan perangkat lunak yang saling terintegrasi.

a. Kebutuhan Fungsional

1. Sistem mampu mendeteksi keberadaan manusia menggunakan sensor PIR.
2. Sistem dapat mengendalikan perangkat listrik melalui relay berdasarkan hasil optimasi PSO.
3. Sistem dapat dijalankan secara otomatis maupun manual melalui aplikasi Blynk.
3. Sistem menampilkan informasi status sensor, relay, dan hasil optimasi delayMati di aplikasi Blynk.
4. Sistem mampu mengirimkan data secara real-time ke pengguna.

b. Kebutuhan Non-Fungsional

1. Kinerja: sistem harus memiliki latensi rendah (<2 detik).
2. Reliabilitas: sistem harus tetap berjalan meskipun koneksi internet terputus (relay default ke kondisi aman).
3. Usability: antarmuka monitoring (Blynk) mudah dipahami.
4. Efisiensi: sistem dirancang hemat energi dan resource.

c. Kebutuhan Perangkat

Tabel 1. Kebutuhan Perangkat Sistem

Jenis	Spesifikasi	Fungsi
Mikrokontroler	ESP32 DevKit v1	Pusat pengendali sistem dan eksekusi PSO
Sensor	PIR HC-SR501	Deteksi gerakan manusia
Aktuator	Relay 1 channel	Mengendalikan perangkat listrik
Indikator	LED merah & hijau	Menunjukkan status sistem
Software	Arduino IDE	Pemrograman ESP32
Platform IoT	Blynk	Monitoring & kontrol berbasis cloud

3.2. Arsitektur Sistem

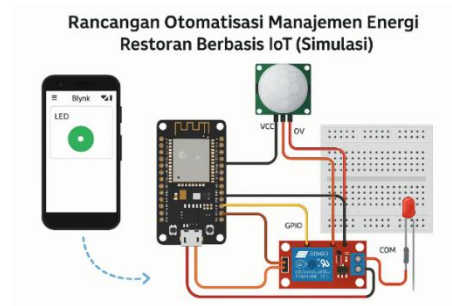
Alur kerja sistem:

Sensor PIR → ESP32 → Optimasi PSO → Relay/LED → Blynk Cloud → User (Smartphone).

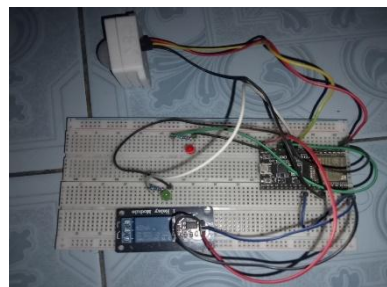
3.3. Perancangan Perangkat Keras

Rangkaian sederhana dirancang untuk mFitness Functionenhubungkan:(Wahyudi dkk., 2021)

1. Sensor PIR ke pin input ESP32.
 2. Relay ke pin output ESP32.
 3. LED indikator ke pin output ESP32.
- Fungsi tiap komponen:
1. PIR → input keberadaan manusia.
 2. Relay → saklar otomatis untuk perangkat listrik.
 3. LED → indikator status (aktif/tidak aktif).



Gambar 1. Sketsa rancangan PSO berbasis IoT (Simulasi)

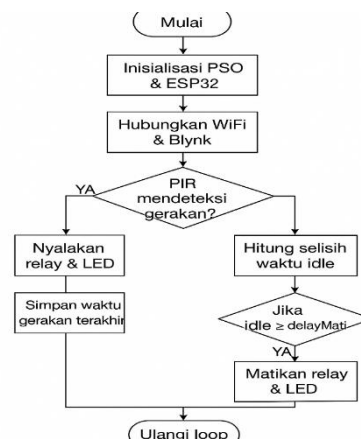


Gambar 2. Hasil Rancangan PSO berbasis IoT (Simulasi)

3.4. Perancangan Perangkat Lunak

a. Flowchart Sistem

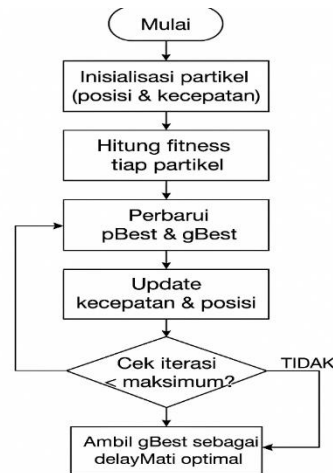
Mulai → Inisialisasi PSO & ESP32 → Hubungkan WiFi & Blynk → PIR mendeteksi gerakan?
YA → Nyalakan relay & LED → Simpan waktu gerakan terakhir TIDAK → Hitung selisih waktu idle
Jika idle $\geq$ delayMati → Matikan relay & LED → Jalankan iterasi PSO → Kirim data hasil ke Blynk → Ulangi loop



Gambar 3. Flowchart Sistem

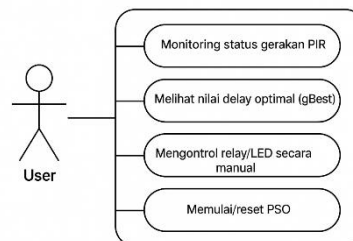
b. Flowchart PSO

Inisialisasi partikel (posisi & kecepatan) Hitung fitness tiap partikel Perbarui pBest & gBest Update kecepatan & posisi Cek iterasi < maksimum? YA → Ulangi TIDAK → Ambil gBest sebagai delayMati optimal



Gambar 4. Flowchart Sistem

3.5. Use Case Diagram



Gambar 5. Use Case Diagram

Aktor:

1. User → menggunakan aplikasi Blynk di smartphone.

Use Case:

1. Monitoring status gerakan PIR
User dapat melihat apakah sensor PIR mendeteksi gerakan.
2. Melihat nilai delay optimal (gBest)
User dapat melihat hasil optimasi dari PSO berupa delayMati terbaik.
3. Mengontrol relay/LED secara manual
User bisa menyalakan atau mematikan relay/LED tanpa menunggu otomatisasi.
4. Memulai/reset proses PSO
User dapat memicu sistem untuk memulai atau mereset proses optimasi PSO.

3.6. Implementasi Sistem

Tahap implementasi merupakan lanjutan dari tahap perancangan. Pada tahap implementasi, system yang telah dibuat akan di terapkan dan akan di lakukan pengujian menggunakan apilkasi Arduino dan Bylink.

a. Spesifikasi Perangkat Keras

1. ESP32 DevKit v1
2. Sensor PIR HC-SR501
3. Relay 1 channel
4. LED merah & hijau

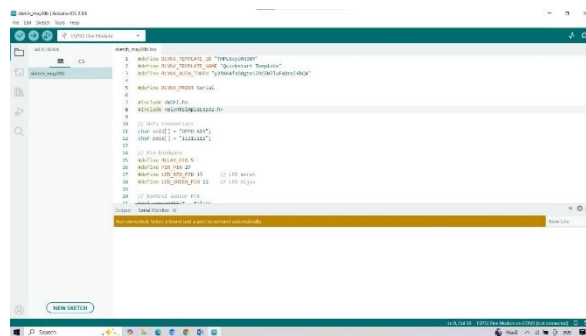
b. Spesifikasi Perangkat Lunak

1. Arduino IDE (C++).
2. Library: WiFi.h, BlynkSimpleEsp32.h.
3. Platform IoT: Blynk App (Android/iOS).

c. Implementasi aplikasi Arduino



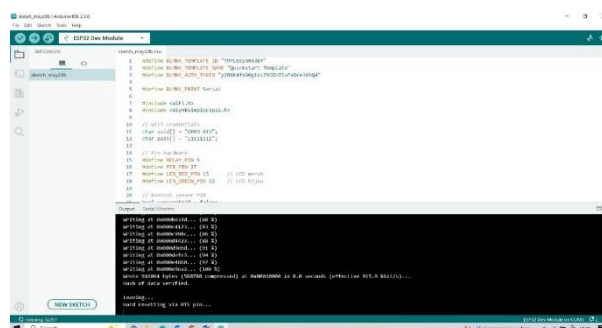
Gambar 6. Aplikasi Arduino



Gambar 7. Tampilan Aplikasi Arduino

Disini Implementasi Algoritma PSO dibuat untuk menjalankan programnya di upload ke aplikasi bylink dan hasil hitungan datanya akan muncul di serial monitor. Pada codingan ini ada beberapa bagian yang diubah untuk menentukan hasil hitungan Algoritma PSO yaitu:

1. Swarm size = 5.
2. Iterasi maksimum = 50.
3. DelayMati range = 2000 – 10000 ms.
4. Fitness function = delayMati + random noise.



Gambar 8. Hasil upload program Arduino ke Blynk

Jika Sudah berhasil source code tersebut, akan muncul notif di bagian output kotak hitam yang bertuliskan :

```
Wrote 916064 bytes (588788 compressed) at 0x00010000 in 8.0
915.9 kbit/s)...
Leaving...
Hard resetting via RTS pin...
```

Hasil yang sudah di upload sudah bisa di jalankan via Blynk yang muncul di serial monitor Arduino



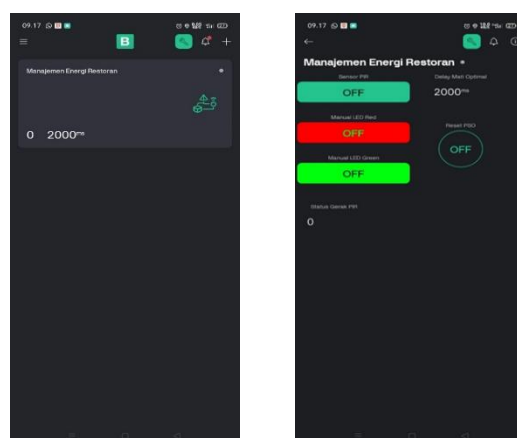
Gambar 9. Tampilan serial monitor Arduino berhasil ke Blynk

d. Implementasi aplikasi Blynk



Gambar 10. Aplikasi Blynk

Blynk adalah jembatan antara sistem IoT yang berbasis sensor dan algoritma optimasi dengan pengguna melalui aplikasi mobile, sehingga sistem otomasi energi bisa dikendalikan dan dipantau secara praktis (Romadhoni & Wahyudi, 2023)



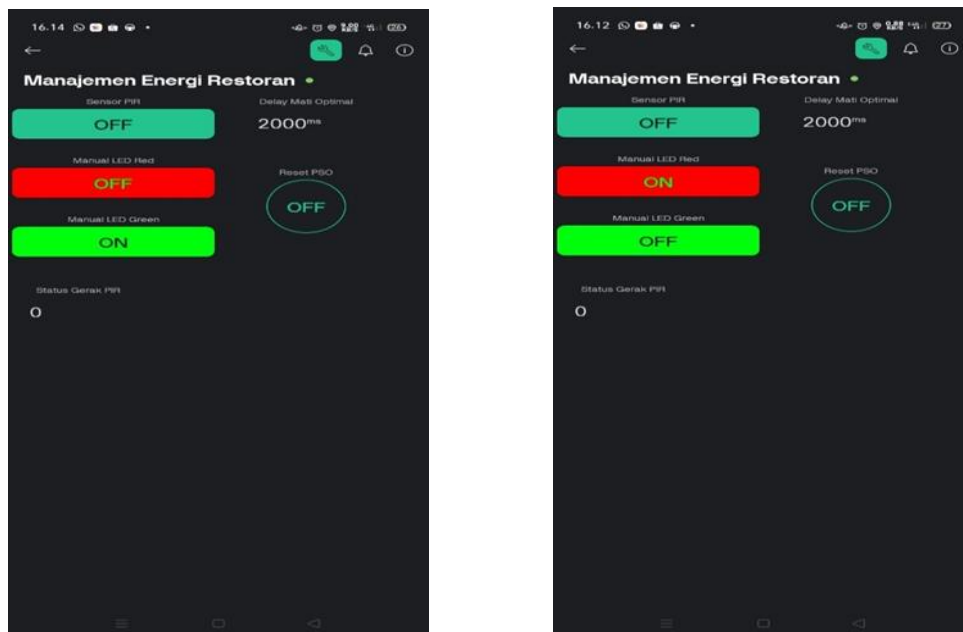
Gambar 10. Tampilan Awal Aplikasi Blynk

Berikut tampilan dashboard dan menu tombol device yang ada pada aplikasi Bylink. Dimana pada tampilan menu tersebut terdapat :

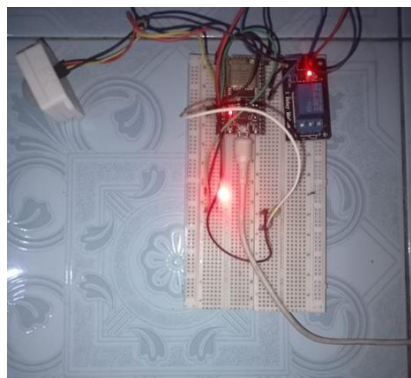
1. Tombol berwarna hijau tua untuk mengaktif atau nonaktifkan sensor PIR
2. Tombol berwarna merah untuk manual On atau Off lamp LED merah
3. Tombol berwarna hijau muda untuk manual On atau Off lampu LED hijau
4. Widget Status gerak PIR, ini merupakan tampilan hasil pergerakan yang dibaca melalui sensor PIR. Bila ada pergerakan akan muncul angka 1 (artinya gerakan terdeteksi) sebaliknya bila angka 0 (tidak terdapat gerakan).
5. Reset PSO, merupakan tombol untuk memulai metode perhitungan Particle Swarm Optimization untuk mendapatkan hasil berapa nilai Delay mati optimal yang di dapat hasil program tersebut.
6. Widget Delay Mati Optimal, hasil dari Reset PSO yang akan muncul di widget ini.

e. Mengoperasikan Sistem manual pada Bylink

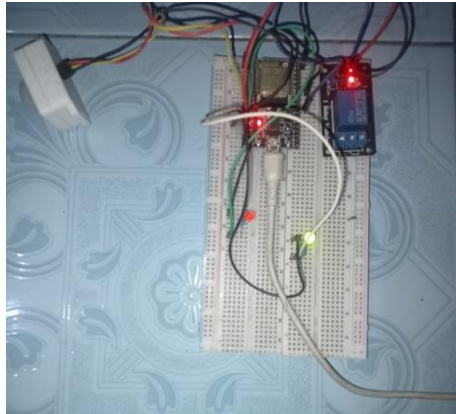
Pada langkah pertama mencoba simulasi program device Bylink dengan manual bila di klik salah satu tombol manual LED baik itu warna merah (RED) atau Hijau (GREEN) bila di klik ON maka salah satu lampu akan menyala bila di klik OFF maka salah satu lampu akan mati sesuai kebutuhan User.



Gambar 11. Tampilan manual tombol LED Red / Green



Gambar 12. Lampu LED Red menyala manual



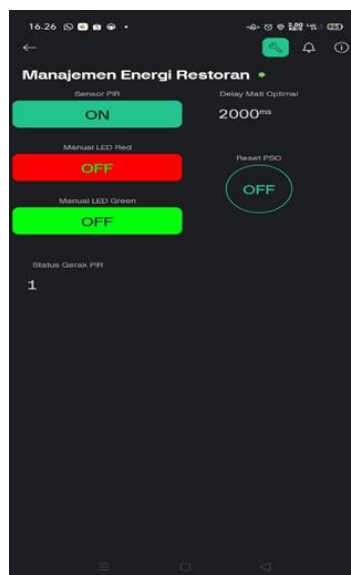
Gambar 13. Lampu LED Green menyala manual



Gambar 14. Tampilan Serial Monitor di Arduino

Pada tampilan serial monitor di Arduino terdapat hasil data tertulis LED Merah Manual ON dan LED Merah Manual OFF sama halnya dengan LED Hijau Manual.

f. Mengoperasikan aplikasi Bylink menggunakan sensor PIR



Gambar 15. Tampilan Bylink saat sensor PIR di aktifkan

Untuk tahap selanjutnya melakukan percobaan mengimplementasikan program menggunakan sensor PIR. Dimana apabila ada pergerakan manusia sensor PIR akan mengirimkan informasi ke dan ESP32 membaca informasi tersebut kemudian dilanjutkan memerintahkan modul relay untuk mengaktifkan atau menonaktifkan lampu LED ketika ada informasi gerakan atau tidak dari sensor PIR tersebut.

Di tampilan aplikasi Bylink ada widget status gerakan PIR menunjukkan angka 1 yang artinya ada pergerakan, apabila angka 0 berarti tidak ada pergerakan. Ketika ada pergerakan lampu LED tetap menyala meskipun dua tombol LED dalam keadaan OFF secara manual. Di aplikasi ini bisa menyesuaikan sesuai kebutuhan User bisa di otomatiskan semua bisa atau bisa di manualkan salah satu lampu tersebut.



Gambar 16. Kedua Lampu LED menyala secara otomatis menggunakan PIR



Gambar 17. Serial monitor tidak ada gerakan sensor PIR

Berikut tampilan serial monitor pada Arduino ketika tidak ada pergerakan pada sensor PIR, ketentuan dan hasil cara membacanya sebagai berikut:

1. Millis : waktu dalam mill detik sejak Esp dinyalakan, Contoh 152507 = 152 detik (2 menit, 53 detik)
2. Motion : status sensor PIR (1 = ada gerakan, 0 = tidak ada gerakan). Di log kamu semua 0 , artinya saat itu tidak ada deteksi gerakan.
3. Relay : status relay (1 = ON, 0 = OFF). Nilai 0 , relay dalam keadaan mati.
4. LED Red : status LED merah (1 = ON, 0 = OFF).
5. LED Green : status LED hijau (1 = ON, 0 = OFF).
6. gBest : belum hasil optimasi PSO (delayMati terbaik dalam milidetik). 8894 menemukan nilai optimal delay 6504 ms (6,5 detik).
7. Iterasi : jumlah iterasi PSO yang sudah berjalan. 0 artinya PSO belum di jalankan.

Cara membacanya :

1. Pada detik ke 152,
2. PIR tidak mendeteksi gerakan,
3. Relay OFF, LED Merah OFF, LED Hijau OFF,
4. Delay optimal (PSO belum di aktifkan) = 6504 ms,
5. Iterasi ke- 0 (PSO belum dijalankan).



1. Millis : waktu dalam millidetik sejak Esp dinyalakan, Contoh 3450198 = 345 detik (5 menit, 75 detik)
2. Motion : status sensor PIR (1 = ada gerakan, 0 = tidak ada gerakan). Di log semua 1 , artinya saat itu ada deteksi gerakan.
3. Relay : status relay (1 = ON, 0 = OFF). Nilai 1 , relay dalam keadaan hidup.
4. LED Red : status LED merah (1 = ON, 0 = OFF).
5. LED Green : status LED hijau (1 = ON, 0 = OFF).
6. gBest : belum hasil optimasi PSO (delayMati terbaik dalam milidetik). 8894 menemukan nilai optimal delay 8894 ms (8,8 detik).
7. Iterasi : jumlah iterasi PSO yang sudah berjalan. 0 artinya PSO belum di jalankan.

1. Pada detik ke 345,
2. PIR mendeteksi gerakan,
3. Relay ON, LED Merah ON, LED Hijau ON,
4. Delay optimal (PSO belum di aktifkan) = 8894 ms,
5. Iterasi ke- 0 (PSO belum dijalankan).

16.51

Manajemen Energi Restoran

Gerbang PIR

OFF

Manual LED Red

OFF

Manual LED Green

OFF

2000 Watt

Manual PIR

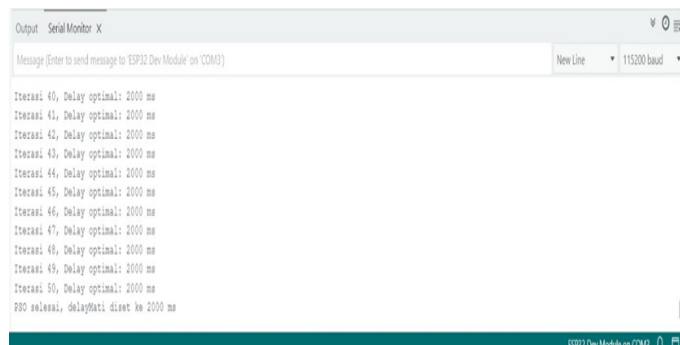
ON

Status Garak PIR

0

Gambar 19. PSO sudah di aktifkan

Ketika PSO sudah di aktifkan tampilan widget pada aplikasi Bylink yaitu Delay Mati Optimal akan menunjukan di angka 2000 ms setelah melakukan perhitungan iterasi sampai ke 50 dari sesuai perintah source code yang kita buat di aplikasi Arduino.



Gambar 20. Hasil perhitungan PSO

Artinya setelah program melukan pengujian menggunakan PSO yang di dapatkan hasil nilai Delay mati Optimalnya adalah 2000 ms. Sehingga ketika sensor PIR di aktifkan kembali dan mendapatkan informasi adanya gerakan atau gerakan terdeteksi maka lampu LED tersebut hanya menyala selama 2 detik saja. Yang dimana sebelumnya bisa di 5 detik atau 8 detik pada unsigned long delayMati di aplikasi Arduino sesuai kebutuhan User.

3.7. Pengujian Sistem

a. Pengujian Sensor PIR

Tabel 2. Hasil Pengujian Sensor PIR

Skenario	Input	Output yang Diharapkan	Output Aktual	Status
Gerakan manusia	Tangan melintas	PIR = HIGH	PIR = HIGH	OK
Tidak ada gerakan	Ruangan kosong	PIR = LOW	PIR = LOW	OK

b. Pengujian Relay & LED

Tabel 3. Hasil Pengujian Relay dan LED

Kondisi	Input	Output yang Diharapkan	Output Aktual	Status
Ada gerakan	PIR = HIGH	Relay ON, LED ON	Sesuai	OK
Tidak ada gerakan > delayMati	PIR = LOW	Relay OFF, LED OFF	Sesuai	OK

c. Pengujian Integrasi dengan Blynk

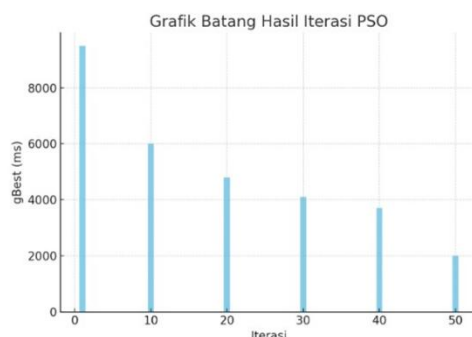
Tabel 4. Pengujian Integrasi dengan Aplikasi Blynk

Skenario	Input	Output Blynk	Status
PIR mendeteksi gerakan	PIR = HIGH	Widget V4 = 1 (motion)	OK
Delay optimal tercapai	gBest = 2000 ms	Widget V5 menampilkan 2000	OK
Kontrol manual LED	Tombol = ON	V1 LED menyala	OK

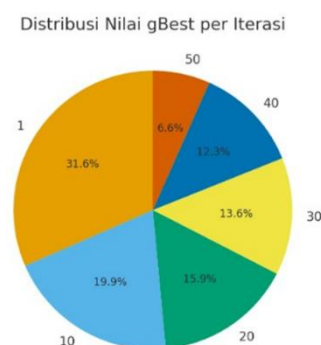
d. Pengujian Algoritma PSO

Tabel 5. Pengujian Algoritma PSO

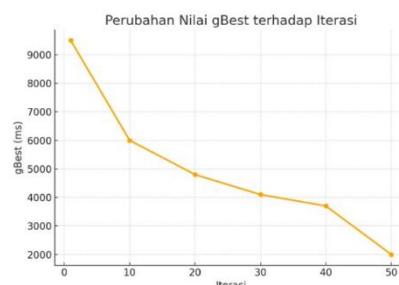
Iterasi	gBest (ms)
1	9500
10	6000
20	4800
30	4100
40	3700
50	2000



Gambar 18. Hasil Grafik Batang PSO



Gambar 19. Hasil Grafik Lingkaran PSO



Gambar 20. Hasil Grafik Chart PSO

Berikut hasil visualisasi terbaru untuk data gBest pada setiap iterasi PSO:

1. Grafik Batang – memperlihatkan penurunan nilai gBest dari 9500 ms menjadi 2000 ms.
2. Diagram Lingkaran (Pie Chart) – menunjukkan proporsi kontribusi setiap iterasi terhadap total gBest.
3. Grafik Garis (Chart) – menggambarkan tren konvergensi PSO menuju nilai optimal 2000 ms di iterasi ke-50.

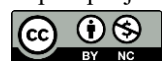
4. Kesimpulan

Berdasarkan hasil penelitian dan pengujian yang telah dilakukan, dapat disimpulkan beberapa hal sebagai berikut:

1. Sistem otomasi manajemen energi berbasis Internet of Things (IoT) berhasil dirancang dan diimplementasikan menggunakan ESP32, sensor PIR, relay, serta platform Blynk untuk pemantauan dan pengendalian secara real time. Sistem ini mampu mendeteksi keberadaan manusia dan mengatur nyala perangkat listrik secara otomatis.
2. Implementasi algoritma Particle Swarm Optimization (PSO) pada mikrokontroler ESP32 terbukti mampu melakukan optimasi terhadap parameter waktu delay mati (delayMati), dengan hasil konvergensi nilai gBest sebesar 2000 ms pada iterasi ke-50. Nilai ini menjadi waktu tunda Optimal untuk mematikan perangkat setelah tidak ada gerakan.
3. Dari hasil pengujian, sistem menunjukkan efisiensi energi hingga $\pm 80\%$ dibandingkan kondisi tanpa PSO. Sebelumnya perangkat tetap aktif rata-rata selama 10 detik setelah tidak ada gerakan, sedangkan setelah optimasi PSO hanya sekitar 2 detik.
4. Sistem memiliki reliabilitas tinggi dengan waktu respon sensor ke aplikasi Blynk kurang dari 1 detik, sehingga mampu bekerja secara real-time dan stabil pada lingkungan simulasi restoran.
5. Meskipun demikian, sistem masih memiliki keterbatasan pada ketergantungan koneksi internet untuk fungsi monitoring dan pengendalian, serta belum diuji menggunakan beban listrik nyata dalam skala besar.

Referensi

- Ahmad Putra, Z. M., Asri, P., Romadloni, F., & Arnestanta, R. R. (2023). Penerapan algoritma particle swarm optimization untuk meningkatkan efisiensi daya keluaran panel surya. *Jurnal Teknik Elektro dan Komputer TRIAC*, 10(2), 56–64. <https://doi.org/10.21107/triac.v10i2.20717>
- Alterazi, H. A., Kshirsagar, P. R., Manoharan, H., Selvarajan, S., Alhebaishi, N., Srivastava, G., & Lin, J. C. W. (2022). Prevention of cyber security with the Internet of Things using particle swarm optimization. *Sensors*, 22(16). <https://doi.org/10.3390/s22166117>
- Faridha, M., & Dewiani, D. (2024). Pemanfaatan aplikasi particle swarm optimization (PSO) untuk pengaturan pengurangan beban tenaga listrik. *Jurnal Teknik Elektro Uniba (JTE UNIBA)*, 8(2), 412–418. <https://doi.org/10.36277/jteuniba.v8i2.254>
- Gad, A. G. (2022). Particle swarm optimization algorithm and its applications: A systematic review. *Archives of Computational Methods in Engineering*, 29(5). <https://doi.org/10.1007/s11831-021-09694-4>
- Habiby, J. S., Triwiyatno, A., & Andromeda, T. (2021). Perancangan algoritma optimasi menggunakan particle swarm optimization (PSO) untuk model wind turbine farm. *AutoMech: Jurnal Teknik Mesin*, 1(1), 5–11. <https://doi.org/10.24269/jtm.v1i01.4123>
- Nihayatus Saadah, N., Nadziroh, F., Muna, N., Nisa', K., Pratiarso, A., Astawa, I. G. P., Santoso, T. B., Yulianto, S. S., & Putro, A. B. A. (2025). Optimizing LoRa gateway placement for marine buoy monitoring using particle swarm optimization (PSO). *Journal of Applied Informatics and Computing*, 9(5), 2264–2269. <https://doi.org/10.30871/jaic.v9i5.11026>
- Piu, S., Arifin, A., & Rizal, M. (2025). Optimization of electric energy usage for household customers based on machine learning and Internet of Things. *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, 5(1), 84–92.
- Romadhoni, R., & Wahyudi, D. (2023). Mengoptimalkan jaringan sensor nirkabel dalam aplikasi monitor lingkungan dengan teknologi IoT di Indonesia. *Jurnal Multidisiplin West Science*, 2(10), 858–867. <https://doi.org/10.58812/jmws.v2i10.704>
- Subekti, R., Ramadhan, N. J., & Hanifah, A. (2024). Implementasi algoritma particle swarm optimization pada AS/RS berbasis mikrokontroler Arduino. *The Indonesian Journal of Computer Science*, 13(4), 6642–6653. <http://ijcs.net/ijcs/index.php/ijcs/article/view/4261>
- Suri Tauladan, I., & Novia Syamsir, H. (2022). Optimasi setting directional overcurrent relay dengan menggunakan hybrid algoritma differential evolution-particle swarm optimization. *Jurnal Elektro dan Mesin Terapan*, 8(2), 194–204. <https://jurnal.pcr.ac.id/index.php/elementer>



- Wahyudi, M. F., Setiawidayat, S., & Hunaini, F. (2021). Metode particle swarm optimization untuk menentukan daya optimal turbin gas PLTGU Grati berdasarkan heat rate. *JASEE Journal of Application and Science on Electrical Engineering*, 2(1), 37–46. <https://doi.org/10.31328/jasee.v2i01.161>
- Bey, M., Kuila, P., Naik, B., & Ghosh, S. (2023). Quantum-inspired particle swarm optimization for efficient IoT service placement in edge computing systems. *Expert Systems with Applications*, 236, 121270. <https://doi.org/10.1016/j.eswa.2023.121270>
- Ravina, M., Galletta, S., Dagbetin, A., Kamaleldin, O. A. H., Mng'ombe, M., Mnyenyembe, L., Shanko, A., & Zanetti, M. (2021). Urban wastewater treatment in African countries: Evidence from the Hydroaid initiative. *Sustainability*, 13(22), 12828. <https://doi.org/10.3390/su132212828>
- Sandi, G. H., & Fatma, Y. (2023). Pemanfaatan teknologi Internet of Things (IoT) pada bidang pertanian. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 7(1), 1–5. <https://doi.org/10.36040/jati.v7i1.5892>
- Himeur, Y., Elnour, M., Fadli, F., Meskin, N., Petri, I., Rezgui, Y., Bensaali, F., & Amira, A. (2022). AI-big data analytics for building automation and management systems: A survey, actual challenges and future perspectives. *Artificial Intelligence Review*, 56(6), 4929–5021. <https://doi.org/10.1007/s10462-022-10286-2>
- Syahfitri, A. (2025). Internet of Things (IoT), sejarah, teknologi, dan penerapannya. *Uranus: Jurnal Ilmiah Teknik Elektro, Sains dan Informatika*, 3(1), 113–120.
- Setiadi Syaiful Iman, S., Khadafy, A. Z. M., Prastyo, M. A., Utomo, S. B., & Nugroho, D. (2025). Analisis desain turbin Archimedes screw pembangkit listrik tenaga mikrohidro. *TEKTONIK: Jurnal Ilmu Teknik*, 2(2), 216–225. <https://doi.org/10.62017/tektionik.v2i2.3831>
- Supriyono, H., Rochman, P. A. N., & Tokhi, M. O. (2021). IoT technology involving wheeled line follower robot for restaurant services automation. *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, 9(1), 100–113. <https://doi.org/10.26760/elkomika.v9i1.100>
- Sutanto, T. (2024). Desain sistem pemantauan dan pengendalian outlet restoran makanan cepat saji berbasis Internet of Things. *INFOTECH: Jurnal Informatika & Teknologi*, 5(2), 336–348. <https://doi.org/10.37373/infotech.v5i2.1464>
- Wijaya, I. M. A. C., & Winardi, S. (2022). Rancang bangun sistem nomer antrian berbasis Internet of Things (IoT). *Jurnal Saintekom: Sains, Teknologi, Komputer dan Manajemen*, 12(2), 176–188. <https://doi.org/10.33020/saintekom.v12i2.315>