

Implementasi Web SCADA Pada Sistem PLTS

Rose Sunky^{1*}, Riki Mukhaiyar²

^{1,2} Prodi Teknik Elektro Industri, Fakultas Teknik, Universitas Negeri Padang, Indonesia

^{*}Corresponding author, rosesunky18@gmail.com

Abstrak

Salah satu isu yang semakin mendesak untuk dicari solusinya adalah krisis energi yang semakin meningkat dan dampak negatifnya terhadap lingkungan. Dalam menghadapi masalah ini, salah satu alternatif sumber energi yang sangat menjanjikan adalah energi surya yang dapat diubah menjadi energi listrik melalui penggunaan sistem Pembangkit Listrik Tenaga Surya (PLTS). Namun, masih banyak kendala yang dihadapi dalam penggunaan PLTS, terutama pengontrolan dan *monitoring* yang efektif pada penggunaan energi listrik di rumah, sehingga dapat mengakibatkan penggunaan energi listrik yang tidak efisien. Oleh karena itu, dibuat sebuah sistem SCADA yang dapat mengontrol dan memonitoring sistem PLTS. Dalam konteks ini, penelitian memfokuskan pada pemantauan arus, tegangan, dan daya yang dihasilkan oleh modul surya, serta pengendalian nyala lampu sebagai output daya PLTS. Data dari sensor yang terpasang pada PLTS dikirim ke sebuah web server melalui jaringan, memungkinkan pengguna untuk memonitor dan mengontrol sistem melalui alamat IP. Web server ini dikembangkan menggunakan bahasa pemrograman Python dan dijalankan pada mikrokontroler ESP32 dengan *firmware* MicroPython. Hasil eksperimen menunjukkan bahwa rata-rata daya puncak yang dihasilkan oleh PLTS adalah sekitar 18 Volt pada interval waktu tertentu, kemudian diubah oleh *Solar Charge Controller* (SCC) menjadi 14 Volt untuk mengisi baterai aki 12 Volt 3.5 Ah. Hasil analisis data menunjukkan selisih tegangan rata-rata sebesar 0.097%, selisih arus rata-rata sebesar 2.128%, dan selisih daya rata-rata sebesar 2.121%. Hasil ini menegaskan bahwa sistem SCADA berbasis web ini mampu memberikan informasi yang akurat dan dapat diandalkan dalam pemantauan dan pengendalian sistem PLTS.

INFO.

Info. Artikel:

No. 510

Received. September, 14, 2023

Revised. October, 11, 2023

Accepted. October, 20, 2023

Page. 792 – 798

Kata kunci:

- ✓ SCADA
- ✓ PLTS
- ✓ Micropython
- ✓ Interface
- ✓ SCC

Abstract

One of the pressing issues that requires a solution is the escalating energy crisis and its adverse environmental impacts. In addressing this problem, one highly promising alternative energy source is solar energy, which can be converted into electrical energy through the use of Solar Power Generation Systems (SPGS). However, there are still numerous challenges associated with the utilization of SPGS, particularly effective control and monitoring of electricity usage in homes, leading to inefficient energy consumption. Therefore, a SCADA (Supervisory Control and Data Acquisition) system has been developed to control and monitor SPGS. In this context, the research focuses on monitoring the current, voltage, and power generated by solar modules, as well as controlling the illumination of lights as the output of the SPGS. Data from sensors installed in the SPGS are transmitted to a web server via a network, enabling users to monitor and control the system through an IP address. This web server is developed using the Python programming language and runs on an ESP32 microcontroller with MicroPython firmware. The experimental results indicate that the average peak power generated by the SPGS is approximately 18 Volts at specific intervals, which is then converted to 14 Volts by the Solar Charge Controller (SCC) for charging a 12 Volt 3.5 Ah battery. Data from the sensors are compared with measurements obtained using a multimeter and displayed through the web server interface. The data analysis results show a voltage difference of 0.097%, a current difference of 2.128%, and a power difference of 2.121% on average. These findings affirm that this web-based SCADA system is capable of providing accurate and reliable information for monitoring and controlling the SPGS.

PENDAHULUAN

Sumber energi alternatif yang ramah lingkungan sangat diperlukan guna mengurangi ketergantungan pada energi konvensional yang semakin terbatas, serta memanfaatkan sumber energi yang tersedia namun tidak merusak lingkungan [1][2]. Indonesia, sebagai negara beriklim tropis, memiliki potensi yang sangat besar dalam menggunakan energi matahari sebagai sumber daya yang dapat berperan penting dalam pengembangan Pembangkit Listrik Tenaga Surya (PLTS) dalam skala yang besar[3][4][5]. Dengan memanfaatkan energi surya, penggunaan energi konvensional yang tidak ramah lingkungan dan terbatas dapat digantikan. Penggunaan sel surya untuk menghasilkan energi dari sinar matahari merupakan solusi yang tepat. Sebelum mengadopsi sumber energi terbarukan ini, perlu dilakukan pengukuran yang akurat terhadap potensi sumber daya yang tersedia, termasuk intensitas cahaya matahari[6][7].

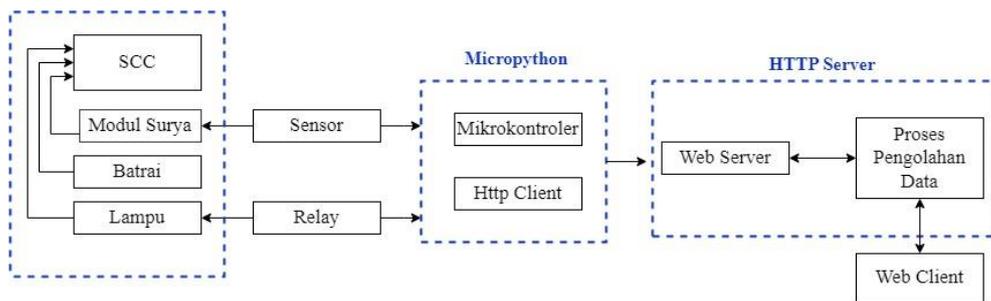
Daya listrik yang dihasilkan dari sel surya sangat dipengaruhi oleh berbagai kondisi lingkungan tempat sel surya ditempatkan. Faktor-faktor seperti intensitas cahaya matahari dan arah sinar matahari berperan penting dalam menentukan seberapa efisien sel surya mengonversi energi matahari menjadi listrik. Intensitas cahaya matahari yang tinggi meningkatkan produksi listrik, sementara cuaca buruk seperti awan tebal atau hujan dapat menguranginya. Selain itu, perubahan posisi matahari seiring waktu memerlukan penyesuaian sudut sel surya untuk memaksimalkan penangkapan sinar matahari. Lingkungan sekitar yang dinamis juga berdampak pada fluktuasi daya keluaran sel surya [8][9]. Untuk mencegah kerusakan dan penurunan kinerja sel surya, diperlukan suatu sistem yang dapat memantau kinerja sel surya. Hal ini bertujuan untuk mengambil langkah-langkah antisipatif guna mencegah kerusakan dan penurunan kualitas modul surya tersebut[10][11][12].

MicroPython adalah implementasi bahasa pemrograman Python yang dioptimalkan untuk digunakan pada perangkat keras terbatas seperti mikrokontroler dan sistem semikonduktor lainnya. Dikembangkan oleh Damien George pada tahun 2013, MicroPython menyediakan lingkungan pemrograman yang ringan dan efisien untuk mengembangkan perangkat keras[13]. Dalam MicroPython, pengembang dapat menulis kode Python yang dieksekusi langsung pada perangkat keras. Ini memungkinkan pengendalian langsung terhadap berbagai periferal seperti GPIO (*General Purpose Input/Output*), ADC (*Analog-to-Digital Converter*), SPI (*Serial Peripheral Interface*), I2C (*Inter-Integrated Circuit*), dan lainnya. Dengan kata lain, MicroPython memungkinkan pengembang untuk mengontrol dan memanfaatkan kemampuan perangkat keras secara langsung menggunakan bahasa pemrograman yang akrab dan mudah dipahami, yaitu Python[14][15]. Salah satu keunggulan utama MicroPython adalah kemampuannya untuk berjalan pada perangkat keras dengan sumber daya terbatas, seperti mikrokontroler dengan kecepatan prosesor rendah dan memori yang terbatas. Interpreter MicroPython yang ringan dan ukuran kode yang lebih kecil memungkinkan pengembang untuk menjalankan kode Python bahkan pada perangkat dengan keterbatasan sumber daya tersebut.

Berdasarkan pemaparan di atas maka dibuatlah tugas akhir yang menerapkan web SCADA pada sistem PLTS dengan tujuan memantau arus, tegangan, dan daya yang dihasilkan oleh modul surya, serta mengontrol nyala lampu sebagai *output* dari daya yang dihasilkan oleh modul surya. Data yang diperoleh dari sensor akan dikirim ke web server yang akan memungkinkan pengguna untuk memantau dan mengontrol melalui alamat IP. Sistem web server ini akan diprogram menggunakan bahasa pemrograman Python pada ESP32 dengan memanfaatkan *firmware* dari MicroPython.

METODE PENELITIAN

Penelitian ini adalah sebuah eksperimen yang melibatkan serangkaian tahapan, dimulai dari tahap perancangan, perakitan alat, pengujian perangkat, pengumpulan data, dan analisis data. Tahap perancangan dianggap sebagai langkah awal yang memiliki signifikansi besar dalam proses produksi dan pengembangan alat ini. Tujuan utamanya adalah untuk mempermudah dan mengkoordinasikan proses produksi, sekaligus mempertimbangkan berbagai aspek seperti material, komponen mekanik, perangkat keras, dan perangkat lunak yang diperlukan. Sistem web SCADA yang diterapkan dalam penelitian ini dapat dijelaskan dalam bentuk blok diagram pada Gambar 1.



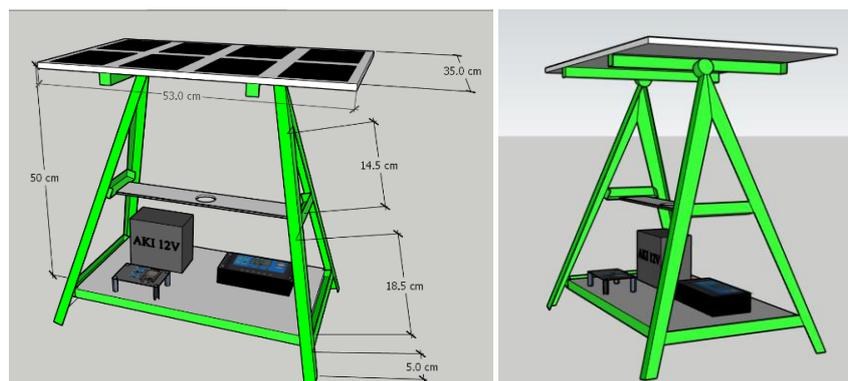
Gambar 1. Blok Diagram Sistem

Ketika sinar matahari mencapai modul surya yang terdiri dari sel-sel surya yang terhubung secara seri dan paralel, foton-foton dalam sinar matahari merangsang elektron-elektron dalam bahan semikonduktor, menghasilkan tegangan listrik. Tegangan yang dihasilkan dialirkan ke SCC (*Solar Charge Controller*) untuk memastikan pengisian yang efisien dan aman dari panel surya ke baterai, melindungi baterai dari *overcharging* (pengisian berlebih) dan *deep discharging* (pengosongan berlebih), serta mengontrol aliran daya ke beban yang terhubung.

Sensor arus ACS712 dan sensor tegangan akan dipasang di *output* modul surya yang menuju SCC sehingga daya yang dihasilkan oleh modul surya dapat dimonitor melalui web server. Modul relai akan dipasang pada *output* SCC yang menuju beban lampu, sehingga pengguna dapat mengendalikan nyala lampu melalui web server. Mikrokontroler ESP32 akan berfungsi sebagai pengendali yang mengatur sensor dan relai sesuai kebutuhan. Mikrokontroler ESP32 akan diprogram menggunakan bahasa pemrograman Python dengan menghapus *firmware* yang ada dan memperbarui dengan *firmware* yang disediakan oleh MicroPython.

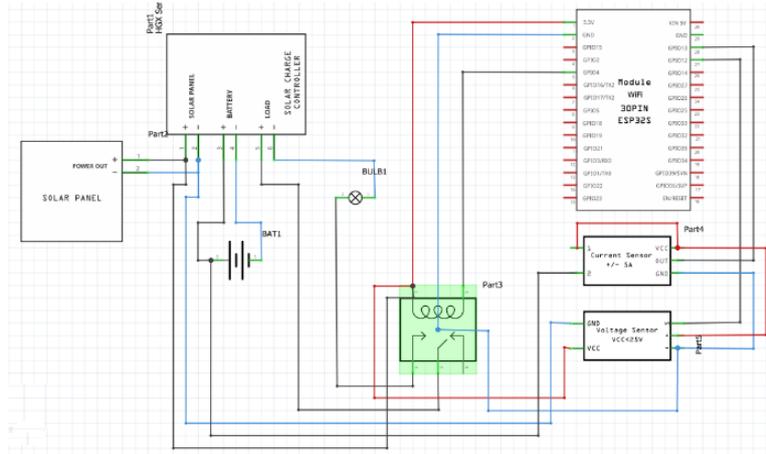
Rancangan Hardware

Penelitian ini dirancang menggunakan satu batang besi siku yang di potong dan hubungkan sedemikian rupa dan diberi akrilik/papan sebagai landasan untuk dudukan lampu dan komponen-komponen lainnya. Alat ini juga dapat dibongkar pasang karena setiap bagian dihubungkan menggunakan mur-baut. Kemiringan dari modul surya juga dapat di *adjust* secara manual dengan menekan salah satu sisi depan atau belakang modul surya. Seperti yang dapat dilihat pada Gambar 2.



Gambar 2. Desain Hardware

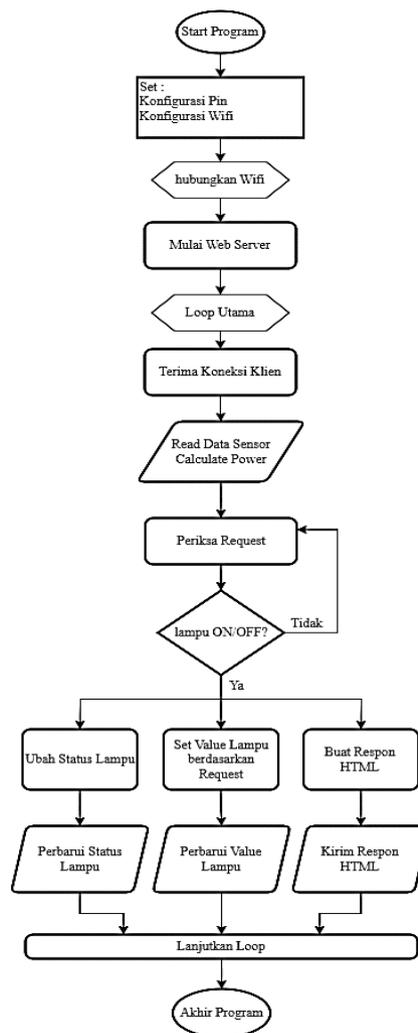
Perancangan *Wiring* Alat berfungsi untuk merencanakan dan mengatur penghubungan kabel atau komponen listrik dalam sebuah alat atau sistem. Berikut *wiring* diagram dapat dilihat pada Gambar 3.



Gambar 3. Skematis Wiring Diagram

Rancangan Program

Flowchart pada Gambar 4. bertujuan untuk memberikan gambaran langkah-langkah dan percabangan pada program secara keseluruhan agar program yang akan dibuat sesuai dengan rancangan yang dibuat.



Gambar 4. Flowchart Program

Dari *Flowchart* di atas dapat dijelaskan sebagai berikut:

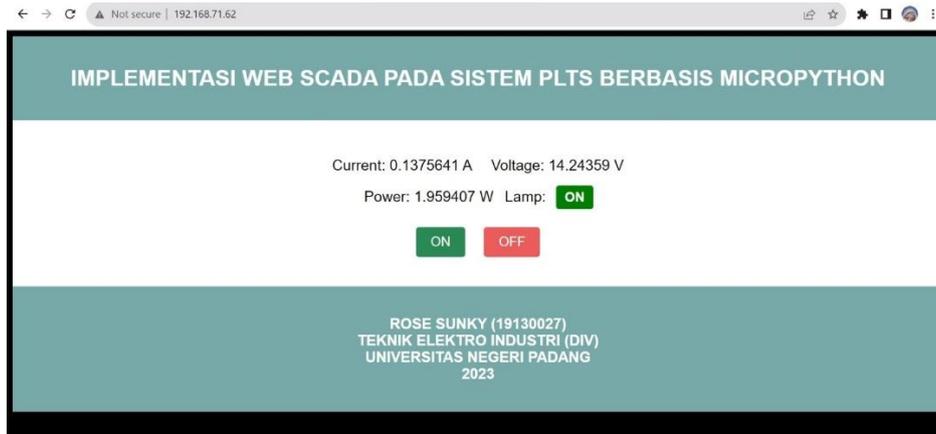
1. **Start Program:** Program dimulai dari titik ini.
2. **Set Konfigurasi Pin:** Konfigurasi pin yang digunakan untuk menghubungkan sensor arus, sensor tegangan, dan lampu dinyatakan di sini. Sementara, **Set Konfigurasi WiFi:** Konfigurasi SSID dan *password* WiFi yang akan digunakan oleh ESP32 dinyatakan di sini.
3. **Hubungkan WiFi:** ESP32 terhubung ke jaringan WiFi dengan menggunakan SSID dan *password* yang telah dikonfigurasi sebelumnya.
4. **Memulai Web Server:** Web server di ESP32 diinisialisasi dan siap menerima koneksi dari klien.
5. **Loop Utama:** Program masuk ke dalam *loop* utama, di mana ia akan terus berjalan dan menunggu koneksi dari klien.
6. **Terima Koneksi Client:** Ketika ada klien yang terhubung, koneksi diterima, dan program melanjutkan eksekusi.
7. **Baca Arus dan Tegangan:** Sensor arus dan tegangan dibaca untuk mendapatkan nilai arus dan tegangan saat ini.
8. **Periksa Jenis Permintaan:** Permintaan yang dikirim oleh klien diperiksa untuk mengetahui apakah ada permintaan kontrol lampu.
9. **Jika Permintaan Kontrol Lampu:** Jika permintaan yang diterima adalah permintaan kontrol lampu, maka status lampu akan diubah berdasarkan permintaan tersebut.
10. **Ubah Status Lampu:** Status lampu (hidup atau mati) akan diubah berdasarkan permintaan kontrol lampu yang diterima.
11. **Set Value Lampu:** Pin yang mengendalikan lampu akan diatur sesuai dengan status lampu yang baru.
12. **Buat Respons HTML:** Respons HTML dibuat dengan menggunakan data arus, tegangan, daya, dan status lampu yang terkini.
13. **Perbarui Status Lampu:** Status lampu yang baru diperbarui dan disimpan.
14. **Perbarui Value Lampu:** *Value* yang mengendalikan lampu diperbarui sesuai dengan status lampu yang baru.
15. **Kirim Respons HTML:** Respons HTML dikirim ke klien yang terhubung.
16. **Lanjutkan Loop:** Program melanjutkan *loop* utama untuk menerima koneksi dan memproses permintaan selanjutnya.
17. **Akhiri Program:** Program berakhir di titik ini.

HASIL DAN PEMBAHASAN

Pada penelitian ini digunakan sensor ACS712 dan *Voltage Divider* sebagai alat ukur pada sistem PLTS yang kemudian hasil pengukuran dikonversikan ke web server melalui modul esp32 dapat dilihat pada Gambar 5 dan Gambar 6. Pengujian *Hardware* dan *Software* dilakukan pada siang hari dengan interval waktu pengambilan data setiap 30 menit.



Gambar 5. Tampilan *Hardware*



Gambar 6. Display Interface web server

Tabel 1. Perbandingan Data Terukur Web SCADA dengan Aktual

Waktu	Vin	Hasil Pengukuran			Hasil Data Monitoring		
		V	I	W	V	I	W
11.00	18.20	14.28	0.13	1.8564	14.28748	0.1327843	1.8971530
11.30	18.29	14.31	0.13	1.8603	14.31867	0.1338725	1.9168761
12.00	18.34	14.56	0.18	2.6208	14.57632	0.185	2.6966192
12.30	18.08	14.25	0.14	1.9950	14.26901	0.1395067	1.9906225
13.00	18.10	14.20	0.14	1.9880	14.21601	0.1395519	1.9838712
13.30	18.00	14.19	0.13	1.8447	14.19905	0.1346451	1.9118325
14.00	17.98	14.17	0.13	1.8421	14.17312	0.1326513	1.8800828
14.30	17.87	14.02	0.12	1.6824	14.05260	0.1246754	1.7520137
15.00	17.45	13.91	0.12	1.6692	13.91221	0.1215542	1.6910876
15.30	17.30	13.53	0.12	1.6236	13.53178	0.1234253	1.6701640
Rata-rata		14.14	0.13	1.8983	14.15363	0.1367667	1.9390323

Tabel 2. Persentase Error

Waktu	% Error = (HP-HM)/HP×100%		
	V	I	W
11.00	0.052	2.142	2.195
11.30	0.061	2.979	3.041
12.00	0.112	2.778	2.893
12.30	0.133	0.352	0.219
13.00	0.133	0.320	0.208
13.30	0.113	3.573	3.639
14.00	0.064	2.039	2.062
14.30	0.022	3.896	4.138
15.00	0.233	1.295	1.311
15.30	0.016	2.854	2.868
Rata-rata	0.094	2.223	2.257

Berdasarkan Tabel 1 rata-rata daya puncak yang diperoleh sebesar 18 Volt pada interval waktu pukul 11.00-13.30 Kemudian, tegangan ini diubah oleh Solar Charge Controller (SCC) menjadi 14 Volt

untuk mengisi baterai aki 12 Volt 3.5 Ah. Data yang diperoleh merupakan perbandingan antara hasil pengukuran menggunakan multimeter dengan hasil pengukuran dari modul sensor yang ditampilkan melalui *interface* web server. Dari data yang diperoleh dapat dihitung bahwa rata-rata selisih tegangan sebesar 0.097 %, rata-rata selisih arus sebesar 2.128 %, dan rata-rata selisih daya sebesar 2.121 %.

KESIMPULAN

Berdasarkan hasil penelitian dan pengujian yang telah dilakukan, dapat disimpulkan bahwa intensitas matahari memiliki pengaruh signifikan terhadap daya yang dihasilkan oleh modul surya. Selain itu, penerapan sistem web SCADA pada sistem PLTS menunjukkan tingkat akurasi yang lebih tinggi dibandingkan dengan penggunaan alat ukur multimeter. Hal ini disebabkan oleh keterbatasan alat ukur multimeter dalam mengukur dengan presisi yang sama, terutama karena hanya dapat mengukur hingga dua angka desimal setelah tanda koma. Hasil perhitungan menunjukkan bahwa rata-rata kesalahan (*error*) hanya sebesar 0.097% untuk tegangan, menegaskan bahwa sistem SCADA web mampu memberikan hasil pengukuran yang lebih akurat dan dapat diandalkan dalam pemantauan dan pengendalian sistem PLTS.

DAFTAR PUSTAKA

- [1] Tim Sekretaris Jenderal Dewan Energi Nasional, "Indonesia Energy Out Look 2019," *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 1689–1699, 2019.
- [2] F. Afif and A. Martin, "Tinjauan Potensi dan Kebijakan Energi Surya di Indonesia," *J. Engine Energi, Manufaktur, dan Mater.*, vol. 6, no. 1, p. 43, 2022, doi: 10.30588/jeemm.v6i1.997.
- [3] N. Hidayanto, Sansuadi, R. C. Nugroho, F. R. F. D. P. S. B. K. Dewi, S. C. B. S. A. Nugroho, and V. A. N. A. F. F. A. Firdaus, "Statistik Ketenagalistrikan 2021," *Kementerian. ESDM*, vol. 119, no. 4, pp. 361–416, 2021.
- [4] E. P. Laksana *et al.*, "Potential Usage of Solar Energy as a Renewable Energy Source in Petukangan Utara, South Jakarta," *J. Rekayasa Elektr.*, vol. 17, no. 4, pp. 212–216, 2021, doi: 10.17529/jre.v17i4.22538.
- [5] Y. Tiandho, I. Dinata, W. Sunanda, R. F. Gusa, and D. Novitasari, "Solar energy potential in Bangka Belitung islands, Indonesia," *IOP Conf. Ser. Earth Environ. Sci.*, vol. 257, no. 1, 2019, doi: 10.1088/1755-1315/257/1/012022.
- [6] A. Mellit and S. Kalogirou, "Solar radiation and photovoltaic systems: Modeling and simulation," *Handb. Artif. Intell. Tech. Photovolt. Syst.*, pp. 1–41, Jan. 2022, doi: 10.1016/B978-0-12-820641-6.00001-6.
- [7] J. Ratela, R. G. T. H. Simanjuntak, N. Wijaya, and Y. Nantan, "Panel Surya Sebagai Sumber Penerangan pada Dermaga Berbasis Internet of Things (IoT)," vol. 4, no. 1, pp. 53–61, 2022.
- [8] F. Anwar and T. Rijanto, "Analisis Perencanaan Plts On Grid Menggunakan Helioscope (Studi Kasus PLTS On Grid 40 KwP Di Gedung Asrama Putri Universitas Airlangga)," vol. 4, no. 2, pp. 724–737, 2023.
- [9] R. Mayangsari and M. Yuhendri, "Sistem Kontrol dan Monitoring Pembangkit Listrik Tenaga Surya Berbasis Human Machine Interface dan Internet of Thing," vol. 4, no. 2, pp. 738–749, 2023.
- [10] F. Ciasaka *et al.*, "Perancangan Sistem Kendali Supervisi Dan Akuisisi Data (Scada) Pada Panel Surya Berbasis Internet of Things," 2023.
- [11] M. A. R. Effendy, "*Pengawasan Kapasitas Panel Surya Berbasis Iot Menggunakan Arduino Uno Pada Plts Pematang Johar," *J. Ilm. Mhs. Tek.*, vol. 2, no. 3, pp. 1–13, 2022.
- [12] M. S. W. Pratama, E. Tridianto, and P. D. Permatasari, "Monitoring Renewable Energy Dengan Konsep Mini SCADA Menggunakan IoT," *Sentikuin*, vol. 2, pp. 1–6, 2019, [Online]. Available: <https://pro.unitri.ac.id/index.php/sentikuin/article/view/136>
- [13] U. Ter Horst, H. Hasberg, and S. Schulz, "MicroPython-based sensor node with asymmetric encryption for ubiquitous sensor networks," *2021 IEEE Int. IOT, Electron. Mechatronics Conf. IEMTRONICS 2021 - Proc.*, 2021, doi: 10.1109/IEMTRONICS52119.2021.9422596.
- [14] V. M. Ionescu and F. M. Enescu, "Investigating the performance of MicroPython and C on ESP32 and STM32 microcontrollers," *2020 IEEE 26th Int. Symp. Des. Technol. Electron. Packag. SIITME 2020 - Conf. Proc.*, pp. 234–237, 2020, doi: 10.1109/SIITME50350.2020.9292199.
- [15] G. Gaspar, P. Fabo, M. Kuba, J. Flochova, J. Dudak, and Z. Florkova, "Development of IoT applications based on the MicroPython platform for Industry 4.0 implementation," *Proc. 2020 19th Int. Conf. Mechatronics - Mechatronika, ME 2020*, 2020, doi: 10.1109/ME49197.2020.9286455.