

APLIKASI PENGENDALIAN DAN MANAJEMEN BATERAI CERDAS

APPLICATION SMART BATTERIES CONTROLLING AND MANAGEMENT

Frans Wijaya Kurnia Hartanto¹, Randy Erfa S², Casi Setianingsih³

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

franswall@student.telkomuniversity.ac.id¹, resaputra@telkomuniversity.ac.id²,
setiacasie@telkomuniversity.ac.id³

Abstrak

Pada masa sekarang ini energi yang dimanfaatkan oleh manusia sangat bergantung kepada listrik konvensional yang menggunakan bahan bakar minyak dan batu bara sebagai sumber olahan energinya. Maka dari itu dibutuhkan energi pembaharuan yang tidak membutuhkan tempat khusus yang bisa di simpan di baterai agar lebih mempermudah penggunaan dan pengisian daya pada tempat penyimpanan daya tersebut. Pada kasus ini pemilihan energi yang digunakan dan alat penampung daya/energi akan sangat berpengaruh, oleh karena itu ada beberapa faktor yang berpengaruh seperti alat, dan aplikasi yang di hubungkan melalui internet. Pada penelitian ini dilakukan perancangan aplikasi web yang didasari oleh alat yang menyimpan energi ke baterai melalui panel surya dan dapat lakukannya management dan *controlling* pada alat yang telah dibuat. Sehingga pada aplikasi web yang dibuat nanti akan ditampilkan data, lama pengisian dan pengontrol untuk melakukan *cut off* pada alat. Hasil akhir dari aplikasi web yang akan dibuat dapat menampilkan data yang diterima, status pengisian baterai, dan menampilkan berapa lama baterai akan terisi, serta mengontrol pengisian battery yang berfungsi untuk on/off saat pengisian.

Kata kunci: *Aplikasi Web, Python, Smart Battery, Monitoring, Controlling*

Abstract

At this time the energy used by humans is very dependent on conventional electricity which uses fuel oil and coal as a source of processed energy. Therefore, renewable energy is needed that does not require a special place that can be stored in the battery to make it easier to use and charge the power storage area. In this case, the selection of energy used and the power/energy storage device will be very influential, therefore there are several influencing factors such as tools and applications that are connected via the internet. In this research, a web application design is based on a device that saves energy to the battery through solar panels and can do management and controlling on the tool that has been made. So that the web application created later will display data, charging time and controller to cut off the device. The final result of the web application that will be created can display the received data, battery charging status, and display how long the battery will be charged, as well as control the battery charging function to on/off while charging.

Keywords: *Aplikasi Web, Python, Smart Battery, Monitoring, Controlling*

1. Pendahuluan

Pada zaman modern ini teknologi selalu menjadi bagian dari kehidupan manusia, banyak negara-negara yang berlomba untuk menjadikan *Smart City* dengan teknologi yang mereka kembangkan, salah satunya penggunaan baterai. Baterai ini menjadi salah satu alternatif untuk digunakan sebagai penyimpanan daya atau energi, baterai juga banyak jenisnya. Di era modern seperti ini banyak negara yang mengembangkan baterai untuk dijadikan *Smart Battery* yang difungsikan untuk berbagai macam penggunaan. Termasuk dalam penggunaan teknologi sering kali beda di perangkat pintar yaitu laptop dan perangkat seluler. Penggunaan baterai yang berlebihan

dapat mengurangi jangka pakai atau umur baterai pun menjadi lebih pendek karena terlalu lama mengisi daya.

Semakin banyaknya kemajuan teknologi, semakin membutuhkan energi yang *extra* untuk beroperasi. Kita tidak pernah tau bahwa sejauh mana baterai ini akan bertahan lebih lama dan bagaimana cara pengguna bisa mengontrolnya dengan sendiri. Dalam pemanfaatan baterai perlu adanya perkembangan dalam pengontrolan dan pemanage yang mampu menghemat dan menghindari terjadinya sesuatu yang tidak diinginkan. Seperti adanya kasus meledaknya sebuah teknologi yang berdampak pada baterainya karena pengaruhnya *overheat* (pengisian yang berlebih). Hal ini biasanya ada sesuatu yang tidak beres di sirkuit pengisian, sehingga pengisian daya terus berlanjut dan mendorong reaksi kimia. Sebagai salah satu komponen utama sebuah baterai tentu menjadi sesuatu yang penting untuk dijaga dan dirawat agar tetap baik dan maksimal.

Dalam pemanfaatan baterai perlu dikembangkan dalam sistem control dan management yang dapat membantu memudahkan pengguna. Oleh karena itu, kita membutuhkan aplikasi yang dapat menampilkan status pengisian battery dan waktu pengisian battery sehingga memudahkan kita pengguna memanaganya, serta adanya controlling yang disediakan sehingga membantu pengguna mengontrol on/off smart battery

2. Dasar Teori

2.1 Smart Battery

Smart Battery adalah untuk mewujudkan kecerdasan baterai atas dasar baterai kimia umum dan sistem manajemen baterai pintar (atau disebut control komunikasi sistem perlindungan).

Smart Battery dapat memantau keadaan baterai dan berkomunikasi langsung dengan pengisi daya pintar dan memberitahukan kebutuhan pengisian daya ke pengisi daya pintar kapan saja. *Smart Battery* juga dapat berkomunikasi langsung dengan peralatan pintar (seperti *mainframe* computer laptop) melalui Smbus, menggunakan *Smart Battery* dapat menunjukkan dan memantau keadaan baterai secara akurat dan memperpanjang masa pakai baterai. Teknologi kunci *Smart Battery* adalah sistem manajemen baterai cerdas. [1]

2.2 Smart Battery Management System

Beberapa pengertian dari *Smart Battery Management System* sebagai berikut:

2.2.1 Smart Battery System (SBS)

Pada tahun 1995, *Intel* dan *Duracell* awalnya mengusulkan definisi Sistem Baterai Cerdas, Mereka berdua dan banyak yang terakhir bergabung dengan perusahaan seperti *IBM*, *Sony* dan *Sharp* bersama-sama mendirikan *Smart Battery* forum pelaksana sistem (SBS-IF) dan merupakan komposisi, fungsi masing-masing bagian, format definisi parameter, cara komunikasi dan berbagai protokol tentang *Smart Battery* sistem dan kemudian aplikasi Sistem Baterai Cerdas yang sebenarnya dimulai. Sistemnya dulu diterapkan computer laptop. [2][3]

2.2.2 Battery Management System (BMS)

Battery Management System (BMS) adalah perangkat yang digunakan untuk menyeimbangkan, pemantauan dan perlindungan baterai yang disusun secara seri atau kemasan baterai. BMS dilengkapi dengan *passive cell balancing*, sensor tegangan masing-masing baterai, sensor arus, sensor suhu, sirkuit proteksi untuk memutuskan arus dan modul *bluetooth* terintegrasi pada perangkat android sebagai monitoring. [3]

2.3 Aplikasi Web

Suatu aplikasi yang diakses menggunakan penjelajah web melalui suatu jaringan seperti internet atau intranet. aplikasi web juga merupakan suatu aplikasi perangkat lunak computer yang di kodekan dalam Bahasa yang mendukung penjelasan web (contoh seperti: *ASP*, *Perl*, *Java*, *Java Script*, *PHP*, *Python*, *Ruby*, dll)

Menggunakan analitik, laporan akan dihasilkan dan dapat dipantau oleh admin melalui *dashboard admin*. Berdasarkan data yang dikumpulkan. [4][11]

2.4 IoT

Internet of Things atau dikenal juga dengan singkatan IoT, Merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari sebuah konektivitas internet yang memungkinkan kita untuk menghubungkan mesin, peralatan dan benda fisik lainnya dengan sensor jaringan dan actuator yang diperuntukan memperoleh data dan mengelola kinerjanya sendiri, sehingga memungkinkan mesin untuk berkolaborasi dan bahkan bertindak berdasarkan informasi baru yang diperoleh secara independen.

IoT adalah sebuah gagasan dimana semua benda didunia nyata dapat berkomunikasi satu dengan yang lain sebagai bagian dari satu kesatuan system terpadu menggunakan jaringan internet. [13]

2.5 Antares

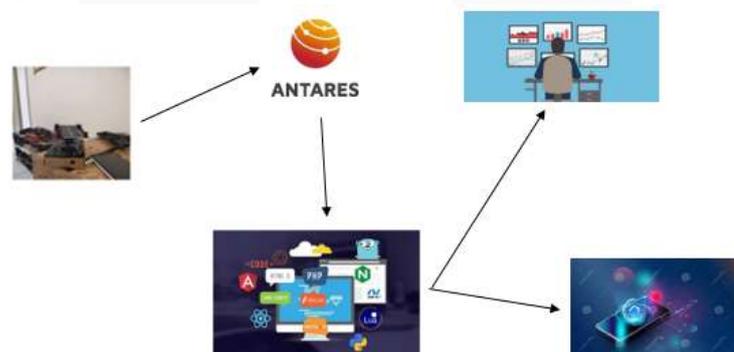
Antares merupakan sebuah horizontal IoT Platform, Banyak kasus-kasus IoT yang dapat dipecahkan dengan menggunakan layanan kami, contohnya adalah *smart home*, *smart metering*, *asset tracking*, *smart building*, dan lain-lain. [16]



Gambar 2.2 Antares [16]

3. Perancangan Sistem

3.1. Desain Sistem



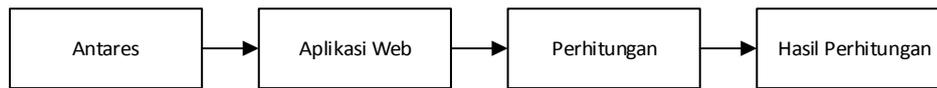
Gambar 3.1 Design Keseluruhan Sistem

Kerja system aplikasi web ini membutuhkan support dari hardware yang telah dibuat oleh Muhammad Eriansyah dikarenakan pada aplikasi web ini membutuhkan data yang akan digunakan untuk fitur management yang menghitung berapa lama baterai akan terisi melalui rumus yang telah didapat.

Data *raw* akan dikirim oleh hardware menuju Antares yang berfungsi sebagai wadah data yang digunakan. Data *raw* yang telah masuk Antares akan langsung diolah seketika saat data *raw* diterima oleh aplikasi web.

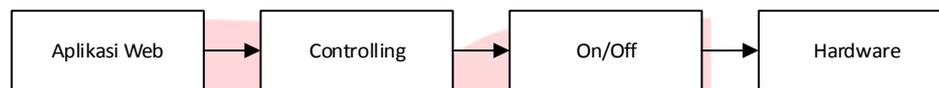
Aplikasi web yang dibuat menggunakan Bahasa Python dengan framework Flask aplikasi ini memiliki dua halaman yaitu *Home* dan *App*, Pada halaman *Home* hanya ditampilkan deskripsi dari aplikasi yang dibuat oleh penulis, lalu untuk halaman *App* akan digunakan sebagai aplikasi utama yang menjadi sistem management, dan sistem *controlling* melalui aplikasi web.

Pada sistem management akan ditampilkan data Antares yang telah hardware kirim, lalu data akan digunakan untuk perhitungan dengan tujuan mengetahui berapa lama waktu yang diperlukan untuk mengisi baterai yang terdapat pada hardware, lalu pada sistem controlling aplikasi membantu hardware untuk melakukan cut off pada pengisian baterai. Pada gambar dibawah akan dijelaskan lebih lanjut bagaimana kerja dari masing-masing sistem yang digunakan.



Gambar 3.2 Diagram Block Management

Sistem management yang dirancang akan dimulai saat data raw sudah masuk Antares dengan tahap berikutnya data akan diterima oleh aplikasi web lalu di tampilkan pada aplikasi web, kemudian data yang diterima akan ditampilkan dan dihitung dengan rumus untuk menghasilkan waktu yang diperlukan untuk mengisi baterai pada hardware



Gambar 3.3 Diagram Block Controlling

Sistem *controlling* yang di rancang akan melakukan *controlling* pada hardware yang dilakukan pada aplikasi web, tugas dari sistem *controlling* ini adalah memutus/*cut off* arus yang masuk ke baterai sehingga pengisian akan terhenti.

4. Hasil dan Analisis

4.1 Hasil Percobaan

Pada aplikasi ini dilakukan tiga jenis pengujian yaitu *Alpha*, *Controlling*, dan *Beta*. Pengujian *Alpha* dari program akan dicek apakah logika logika dari program tersebut sudah sesuai dengan apa yang diharapkan. Pengujian *Controlling* dari program akan memastikan bahwa aplikasi web dapat melakukan *cut off* dengan dua kali pengecekan saat kondisi *on* dan *off*. Serta Pengujian *Beta* yang dilakukan secara objektif dengan mengukur *usability system* dimana sistem akan dilihat melalui video demo aplikasi web.

4.2 Pengujian Alpha

Adapun scenario pengujian alpha yang akan dilakukan dapat dilihat pada table dibawah ini.

Tabel 4.1 Pengujian *Alpha*

No	Fitur yang Diuji	Detail Pengujian	Jenis Pengujian
1	Membuka Aplikasi	Menampilkan Halaman Awal	Black Box
2	Berpindah Tab	Berpindah Tab	Black Box
3	Management Data	Menerima Data dan refresh halaman aplikasi setaip 10 detik	Black Box
4	Controlling	<i>Cut off Battery</i>	Black Box
5	Management Perhitungan	Menampilkan Hasil Perhitungan	Black Box

Pada skenario pengujian terdapat 5 fitur yang bisa pada **table 4.1** pengujian dilakukan hanya dengan mengamati hasil eksekusi melalui data uji dan memeriksa fungsionalitas dari aplikasi.

Pengujian ini dilakukan secara beberapa kali demi mendapatkan hasil yang maksimal dan sesuai dengan apa yang penulis inginkan. Dan pengujian ini juga dapat dikatakan berhasil ketika hasil yang didapat sama setelah beberapa percobaan.

4.3 Pengujian Controlling

Pengujian *controlling* adalah pengujian yang dilakukan untuk memastikan bahwa aplikasi web dapat melakukan cut off, pengujian dilakukan dua kali dengan pengecekan saat tombol on dan saat tombol off seperti pada table dibawah.

Tabel 4.7 Skenario Pengujian *Controlling*

Keadaan Tombol	Kode	Kondisi	Hasil
Off (<i>default</i>)	0	Low	Arus tetap masuk ke Baterai
On	1	High	Arus Terhenti
Off	0	Low	Arus kembali masuk ke baterai

Untuk kode diatas menandakan bahwa sistem sedang berada pada posisi on / off, pada kode 0 diartikan bahwa sistem sedang dalam keadaan off atau tidak sedang dilakukan *cut off* sebaliknya pada kode 1 sistem sedang berada di kondisi on atau sedang dilakukan *cut off*.

Untuk kondisi diatas diartikan bahwa arus yang didapat oleh hardware masuk atau tidak. Pada kondisi low mengartikan bahwa sistem sedang bekerja dan arus akan tetap masuk ke baterai, dan pada kondisi high arus akan terhenti.

4.4 Pengujian Beta

Skenario pengujian untuk memenuhi task pengujian usability testing sebagai berikut:

Tabel 4.8 Task Pengujian *Usability Testing*

No	Task
1	Membuka Google Form dari link yang sudah disebarakan
2	Menonton demo yang telah diupload di youtube
3	Memberikan Penilaian ketika sudah menonton demo diplatform youtube dari pernyataan yang sudah disediakan

5. Kesimpulan Saran

5.1 Kesimpulan

Kesimpulan yang dapat diambil dari penelitian tugas akhir ini yaitu:

1. Aplikasi web untuk management smart battery sudah berjalan dengan baik sesuai fungsinya dengan hasil uji alpha/fungsionalitas sebesar 100%
2. Aplikasi web untuk controlling smart battery sudah diujikan ke perangkat Solar Panel Sistem dan sudah berjalan sesuai fungsinya dengan akurasi rules sebesar 100%
3. Pada pengujian alpha sudah berjalan dengan hasil yang sesuai dengan yang diharapkan, pada uji validasi dan realibilitas juga sudah mendapatkan hasil yang valid dan realible.

5.2 Saran

Saran untuk pengembangan system ke depannya sebagai berikut:

1. Pada penelitian yang akan datang diharapkan aplikasi web dapat menggunakan server sehingga dapat digunakan pada mobile phone.
2. Pada penelitian yang akan datang diharapkan tampilan pada aplikasi web akan dibuat lebih menarik sehingga dapat membuat user mengerti dengan apa yang mau disampaikan oleh aplikasi web.

Referensi:

- [1] Piede liu, and Xiujian Zhang. "The Design of Smart Battery Management Systems". Vol 6. 2011
- [2] Xiujian Zhang, Peide Liu, and Darui Wang. "The Design and Implementation of Smart Battery Management Systems Balance Technology". Vol 6. 2011
- [3] Muhammad Fatahilla, Ponco Siwindarto, Ekas Maulana. "Perancangan Sistem Monitoring dan Proteksi Battery Management System (BMS) pada E-Bike". Vol 5. 2017
- [4] Akhila Joseph, Anjali, Suhaila BM, and Mahesh BL "Smart Garbage Management System". Vol 7. 2020
- [5] Petchiammal S, Priscilla JD, Priyadharshini A, Raviga RM, and Sundararajan TVP. "Smart Monitoring of Water Management System". Vol 6. 2020.

- [6] Ninadlanke, and SheetalKoul. “*Smart Traffic Management System*”. Vol 75. 2013.
- [7] Dolly Handarly, dan Jefri Lianda. “*Sistem Monitoring Daya Listrik Berbasis IoT*”. Vol 3. 2018
- [8] Tara Alshahadeh, and Akin Marsap. “*Smart Cities, Smarter Management: Developing A Smart Framework for Smart City Projects Management in Europe*”. Vol 6. 2018
- [9] B. Zhou et al. “*Smart home energy management systems: Concept, configurations, and scheduling strategies*”. vol. 61. 2016.
- [10] Athul Krishnan, Aparna P, Aswathy Thulasy, Soorya Mohan, Keerthy and As Pillai “*Smart Energy Management System*”. Vol 5. 2018
- [11] Jaja Miharja dan Elzet. “*Perancangan Sistem Informasi Berbasis Web Pada Hotel Boutique Pesona Cikarang*”
- [13] Omar Pahlevi, Astriana Mulyani, dan Miftahul Khoir. “*Sistem Informasi Inventori Barang Menggunakan Metode Object Oriented di PT.Livaza Teknologi Indonesia Jakarta*”. Vol 5. 2018
- [14] Yoyon Efendi. “*Internet of Things(IoT) system pengendalian lampu menggunakan Raspberry PI berbasis Mobile*”. Vol 4. 2018
- [15] Imran Rahman, Balbir Singh Mahinder Singh, Pandian Vasant, and M. Abdullah-Al-Wadud. “*Swarm Intelligence-Based Smart Energy Allocation Strategy for Charging Stations of Plug-In Hybrid Electric Vehicle*”. 2015
- [16] Electrical Technology. “How to calculate the battery charging time & Battery charging Current – example”. Diakses pada 15 mei 2021
- [17] Antares. 2020. <https://antares.id/id/docs.html>, diakses 3 Maret 2021
- [18] Sekolah Koding. “Belajar Flask”, <https://sekolahkoding.com/belajar/flask>, diakses pada 26 juli pukul 02.44