

PENGEMBANGAN APLIKASI MOBILE PENGISIAN DAYA KENDARAAN LISTRIK SESUAI STANDAR OCPP 1.6

Rachmawan Atmaji Perdana^{*1}, M. Nurkhoiri Hindratno², Tanzi M. Santoso³, Topan Try Harmanda⁴, Friyogi Resvy Mahda⁵, Yahya Andreas Lapian⁶, Monica Dwi W. S.⁷, Dionysius Aldion Renata⁸, Prasetyo Aji⁹, Meidy Layooari¹⁰, Kusnanda Supriatna¹¹, Rully Kusumajaya¹², Mohammad Hamdani¹³

^{1,2,3,4,5,6,7,8,9,10,11,12,13}Badan Riset dan Inovasi Nasional, Jakarta

Email: ¹rach020@brin.go.id, ²muha217@brin.go.id, ³tanz001@brin.go.id, ⁴topa003@brin.go.id, ⁵friy001@brin.go.id, ⁶yahy001@brin.go.id, ⁷moni001@brin.go.id, ⁸dion001@brin.go.id, ⁹pras010@brin.go.id, ¹⁰meid001@brin.go.id, ¹¹kusn001@brin.go.id, ¹²rull001@brin.go.id, ¹³moha049@brin.go.id

^{*}Penulis Korespondensi

(Naskah masuk: 26 Oktober 2022, diterima untuk diterbitkan: 03 November 2023)

Abstrak

Makalah ini membahas tentang pengembangan aplikasi mobile untuk pengisian daya kendaraan listrik, yang diberi nama SONIKApp. SONIKApp digunakan untuk melakukan reservasi dan pengendalian pengisian daya secara *remote*. Fitur reservasi diperlukan mengingat pengisian daya baterai kendaraan listrik membutuhkan waktu tunggu serta masih terbatasnya stasiun pengisian kendaraan listrik umum (SPKLU) di Indonesia. Fitur pengisian daya berbasis *remote* juga merupakan fitur penting yang memungkinkan pengguna untuk mengendalikan proses pengisian daya kendaraan listrik. SONIKApp juga terhubung dengan sistem manajemen SPKLU berbasis standar OCPP 1.6 secara *real-time* sehingga data yang ada di aplikasi selalu merupakan data terkini. Aplikasi ini dikembangkan dengan kerangka kerja front-end React Native, sehingga dapat berjalan di atas sistem operasi Android dan iOS. Hasil yang diperoleh adalah terciptanya aplikasi mobile SONIKApp yang memudahkan pengguna dalam pengisian daya kendaraan listrik. Telah dilakukan pengujian SONIKApp untuk fitur reservasi dan pengisian daya secara *remote* berdasarkan OCPP 1.6, dan hasilnya adalah aplikasi SONIKApp telah berhasil mengimplementasikan fitur-fitur tersebut.

Kata kunci: kendaraan listrik, reservasi, pengisian daya, OCPP, React Native, Spring Boot

DEVELOPMENT OF MOBILE APPLICATION FOR CHARGING ELECTRIC VEHICLES COMPLIED WITH OCPP STANDARD 1.6

Abstract

This paper discusses the development of a mobile application for charging electric vehicles, which is named SONIKApp. SONIKApp is used to make reservations and control charging remotely. The reservation feature is required considering that charging electric vehicle batteries requires waiting time and there are still inadequate number of public electric vehicle charging stations (SPKLU) available in Indonesia. The remote-based charging feature is also an important that allows users to control the process of charging electric vehicles from the applications. SONIKApp is also connected in real-time to the SPKLU management system based on the OCPP 1.6 standard so that the data presented in the application is always up-to-date. This application is developed with the React Native front-end framework, so it can run on both Android and iOS operating systems. We have successfully created the SONIKApp mobile application which makes it easier for users to charge electric vehicles. SONIKApp has been tested for reservation and remote charging features based on OCPP 1.6, and the result is that the SONIKApp application has successfully implemented these features.

Keywords: electric vehicle, reservation, charging, OCPP, React Native, Spring Boot

1. PENDAHULUAN

Pertumbuhan penggunaan kendaraan listrik di berbagai negara semakin meningkat dalam beberapa tahun terakhir seperti yang ditunjukkan dalam Global

EV Outlook 2022 yang diterbitkan oleh International Energy Agency (International Energy Agency, 2022), dimana jumlah penjualan kendaraan listrik di tahun 2021 (6,6 juta) dua kali lipat dari tahun sebelumnya dan peningkatan tiga kali lipat dalam 3 tahun terakhir.

Salah satu alasannya adalah kendaraan listrik dinilai jauh lebih ramah lingkungan dan bisa menjadi solusi masalah pemanasan global. Indonesia termasuk salah satu negara berkembang yang masih rendah dalam penjualan kendaraan listrik yaitu kurang dari 0,5% (International Energy Agency, 2022). Terkait dengan hal tersebut, Pemerintah Indonesia berusaha mendukung ekosistem kendaraan listrik melalui Peraturan Presiden Republik Indonesia Nomor 55 Tahun 2019 tentang Program Percepatan Kendaraan Listrik Berbaterai Untuk Transportasi Jalan (Kementerian Hukum dan Hak Asasi Manusia, 2019). Pada tahun 2020, Kementerian ESDM meluncurkan Peraturan Menteri No. 13/2020 tentang Penyediaan Infrastruktur Pengisian Listrik untuk Kendaraan Listrik (Kementerian Energi Sumber Daya Mineral, 2020). Setelah itu, Indonesia mulai gencar melakukan perubahan besar dalam penggunaan kendaraan listrik dengan mendukung pembangunan stasiun pengisian baterai kendaraan listrik untuk masyarakat (Stasiun Pengisian Kendaraan Listrik Umum disingkat SPKLU, atau *Charging Station/CS*) di beberapa titik strategis di beberapa kota (Dharmawan, S. Kumara, & Budiastira, 2021). Hal ini diharapkan dapat mendorong penggunaan kendaraan listrik di Indonesia.

Namun penggunaan kendaraan listrik masih memiliki kendala mengenai waktu pengisian yang relatif lama dan sangat bervariasi tergantung dari kapasitas baterai, kemampuan baterai untuk menerima daya pengisian, sisa daya yang ada pada baterai, dan daya pengisian dari *charger* (Junzhe, Min, Sangwoo, Tung-Yan, & Yifan, 2022). Selain itu, penggunaan berbagai jenis konektor pengisian yang digunakan oleh mobil listrik membuat operator SPKLU harus berinvestasi lebih banyak untuk membangun SPKLU-SPKLU yang dilengkapi dengan bermacam-macam tipe konektor. Untuk kasus di Indonesia, ketersediaan SPKLU yang masih terbatas dapat menjadi kendala dalam mendorong penggunaan kendaraan listrik. Karena pengguna akan menghadapi situasi di mana harus mengantre ketika ingin mengisi baterai kendaraan listriknya di SPKLU terdekat. PLN dalam salah satu paparannya (Asaad, 2021) memprediksi bahwa kebutuhan SPKLU di tahun 2022 sebanyak 2065, jumlah kendaraan listrik di Indonesia di tahun 2022 diperkirakan mencapai 20.639 unit. Berarti perbandingan antara jumlah SPKLU dengan kendaraan listrik adalah sekitar 1:10 yang artinya satu SPKLU akan melayani 10 kendaraan listrik. Kondisi ini dapat menimbulkan antrean pengguna kendaraan listrik untuk melakukan pengisian di SPKLU.

Kendala-kendala tersebut, salah satunya dapat diatasi dengan menyediakan sejenis aplikasi *mobile* yang dapat membantu pengguna ketika ingin melakukan pengisian daya baterai kendaraan listriknya. Aplikasi SONIKApp yang dikembangkan ini dapat membantu pengguna dalam memperoleh informasi terkait lokasi dan ketersediaan SPKLU

terdekat dari posisi pengguna beserta spesifikasinya. Aplikasi ini dilengkapi dengan fitur reservasi agar pengguna tidak perlu menunggu lama ataupun mengantre untuk mendapatkan giliran untuk mengisi daya, serta memiliki fitur untuk mengendalikan pengisian daya baterai kendaraan listrik secara *remote*. Fitur reservasi dan pengisian daya baterai ini kompatibel dengan standar OCPP 1.6 (Venkata, Niladri, Phaneendra, & Vasudeva, 2019) dan menggunakan layanan yang disediakan oleh *server* OCPP. *Server* OCPP adalah tulang punggung sistem manajemen stasiun pengisian (*Charging Station Management System/CSMS-SONIK Web*) yang dikembangkan oleh (Aji, Renata, Larasati, & Riza, 2020).

Adapun kebaruan dari aplikasi SONIKApp adalah sebagai berikut:

1. SONIKApp terintegrasi dengan CSMS sehingga dapat menjalankan fitur-fitur transaksi OCPP dengan baik dan telah teruji pada lingkungan produksi di SPKLU komersial
2. Berbeda dengan aplikasi-aplikasi mobile lain yang dikembangkan oleh (Flocea, Hîncu, & Robu, 2022), (Galbis, et al., 2022), dan (Muddalkar, Chaturkar, Ingole, & Wadaskar, 2022), SONIKApp tidak hanya memiliki fitur pencarian dan reservasi *charging station*, namun juga pengisian daya baterai yang dipicu dengan pemindaian kode QR.

2. KERANGKA TEORI

Pada bagian ini akan dijelaskan beberapa kerangka teori yang digunakan pada makalah, yakni *Open Charge Point Protocol*, *Charging Station Management System*, *Spring Boot*, dan *React Native*.

2.1. Open Charge Point Protocol (OCPP)

OCPP merupakan protokol yang dibuat untuk komunikasi antara titik pengisian daya pada CS dengan pusat sistem pengelolaan CS. Protokol ini dirancang sebagai protokol komunikasi terbuka yang memungkinkan komunikasi antara perangkat CS dan pusat sistem pengelolaan CS dari berbagai produsen yang berbeda.

OCPP diinisiasi oleh yayasan ELaadNL yang berkedudukan di Belanda dan selanjutnya dikembangkan dan dikelola oleh konsorsium global OCA (*Open Charge Alliance*) dari perusahaan terkemuka kendaraan listrik pribadi maupun umum. OCPP menentukan mekanisme komunikasi dengan titik pengisian daya (*Charge Point/CP*) untuk mengetahui kondisi dan status terkini CP seperti, apakah CP berfungsi dengan normal atau sedang digunakan. Selain itu, OCPP juga mengatur pengiriman instruksi kepada CP untuk melakukan tindakan tertentu seperti reservasi/pembatalan reservasi atau memulai/menghentikan proses pengisian baterai (de Leeuw, McMahon, & van Zuuren, 2017).

OCPD versi 1.6 diimplementasi pada aplikasi *mobile* SONIKApp yang dibahas dalam paper ini terutama yang terkait dengan fitur reservasi/pembatalan reservasi serta memulai/menghentikan proses pengisian daya baterai kendaraan listrik.

2.2. Charging Station Management System (CSMS) SONIK-Web

CSMS merupakan sistem pengelolaan CS, dimana berbagai CS dapat dipantau status, kondisi dan operasinya serta dilakukan pengaturan yang diperlukan secara daring. CSMS SONIK-Web yang dikembangkan oleh (Aji, Renata, Larasati, & Riza, 2020) digunakan untuk memantau dan mengendalikan CS dari berbagai produsen dan menggunakan OCPD versi 1.6 sebagai protokol untuk berkomunikasi dengan perangkat CS. CSMS SONIK-Web ini ditujukan untuk operator CS dan memiliki fitur-fitur untuk pengaturan CP, pengelolaan pengguna, memonitor aktivitas dan kondisi CP, riwayat kondisi dan pemakaian CP, sistem *alert*, transaksi dari masing-masing CP, serta fitur laporan penggunaan daya, energi dan biaya baik untuk masing-masing CP maupun seluruh CP yang dikelola pada CS.

Aplikasi *mobile* SONIKApp yang dijelaskan dalam paper ini, terhubung dengan CSMS SONIK-Web sedemikian rupa sehingga SONIKApp dapat memperoleh data terkini terkait status dan kondisi CP untuk dikirimkan kepada pengguna SONIKApp. Selain itu, melalui SONIKApp, dapat dilakukan pencarian lokasi CP yang dikelola CSMS SONIK-Web, melakukan reservasi/pembatalan reservasi serta memulai dan menghentikan pengisian daya baterai kendaraan listrik melalui SONIKApp.

2.3. Kerangka Kerja *Spring* dan *Spring Boot*

Kerangka Kerja *Spring* adalah sebuah kerangka kerja pengembangan aplikasi kode sumber terbuka berbasis Java/J2EE untuk pengembangan aplikasi yang lebih produktif. Fitur-fitur kerangka kerja *Spring* memungkinkan pengembangan yang efisien mulai dari *Web* sederhana hingga aplikasi *enterprise* yang kompleks (Jovanovic, Jagodic, Vujičić, & Randić, 2017). Konsep-konsep kerja utama yang diadopsi *Spring* adalah:

- *Inversion of Control* (IoC)
- *Dependency Injection* (DI)
- *Aspect Oriented Programming* (AOP)
- *Java Persistence API* (JPA)

Spring Boot adalah salah satu modul dari kerangka kerja *Spring* yang menyediakan fungsionalitas *Rapid Application Development* yang dibangun di atas Kerangka Kerja *Spring* untuk menyediakan akses yang cepat ke informasi dalam mengerjakan suatu *project*. Hal paling signifikan mengenai *Spring Boot* adalah kita hanya memerlukan sedikit konfigurasi *Spring*, sehingga membuat

pengembangan aplikasi *standalone* menjadi lebih mudah. *Server* Tertanam dan modul *Spring Framework* digunakan oleh *Spring Boot* untuk tujuan operasionalnya. (Mythily, Arun Raj, & Joseph, 2022).

2.4. React Native

Pengembangan aplikasi *mobile* mengharuskan pengembang agar memahami pengembangan di beberapa *platform* yang berbeda seperti dua *platform* utama yaitu *Android* dan *iOS* (Goadrich & Rogers, 2011). Pengembangan aplikasi *Android* membutuhkan pemahaman pada bahasa *Java* atau *Kotlin* sedangkan pengembangan aplikasi *iOS* membutuhkan pemahaman pada bahasa *Swift*, *C#*, dan *objective C* (Shubham, 2020). Selain memahami bahasa pemrograman tersebut, pengembang juga diharuskan memahami *Android IDE (Integrated Development Environment)* dan *XCode* sebagai *environment* untuk pengembangan aplikasi *Android* dan *iOS* (Vilček & Jakopec, 2017). Karena permasalahan tersebut, pengembang mulai menggunakan bahasa pemrograman *cross-platform* yang dapat mengembangkan aplikasi di *Android* dan *iOS* dengan hanya satu kode pemrograman, salah satu bahasa pemrograman tersebut adalah *React Native*.

3. PERANCANGAN SISTEM

Pemilihan kerangka kerja *Scrum* didasarkan atas kesesuaian dengan sistem kerja modular dan jangka waktu pengerjaan hanya 6 bulan dengan jumlah anggota tim yang terbatas, sementara anggota-anggota tim tersebut juga harus berbagi fokus untuk mengerjakan kegiatan lainnya (Rifky, Mursityo, & Prakoso, 2022). Selain itu, dalam kerangka kerja *Scrum* ini proses pengembangan dapat lebih mudah dipantau oleh *Product Owner*, sehingga dapat meminimalisir kesalahpahaman antara tim pengembang dan *Product Owner*.

3.1. Tahap Inisialisasi

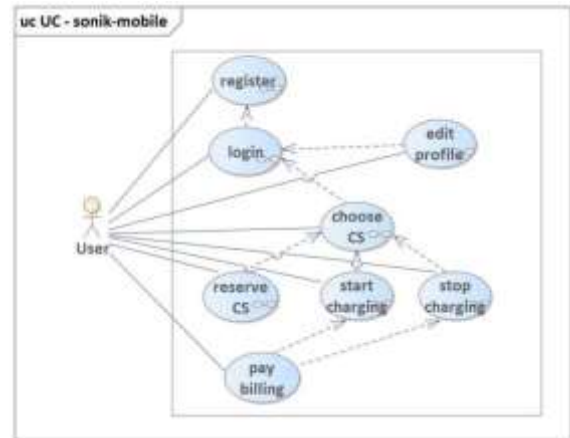
Tahap inisialisasi diawali dengan wawancara dengan *Product Owner*. Pada tahap ini, *Product Owner* menjelaskan mengenai kebutuhan pengembangan aplikasi *mobile* untuk mencari, melakukan reservasi, dan pengisian daya baterai di setiap CS yang akan dikembangkan. Pengisian daya baterai harus dapat dilakukan secara *remote*. Di sini kata *remote* bukan berarti kendali jarak jauh secara harfiah, namun dalam arti bahwa proses pengisian daya dapat diinisiasi dan dihentikan melalui aplikasi. Aplikasi ini diharapkan mampu berjalan di atas *platform Android* dan *iOS*, dengan tampilan yang menarik dan mudah digunakan. Aplikasi ini juga diharapkan mempunyai fitur pembayaran setelah pengguna melakukan pengisian daya.

3.2. Desain

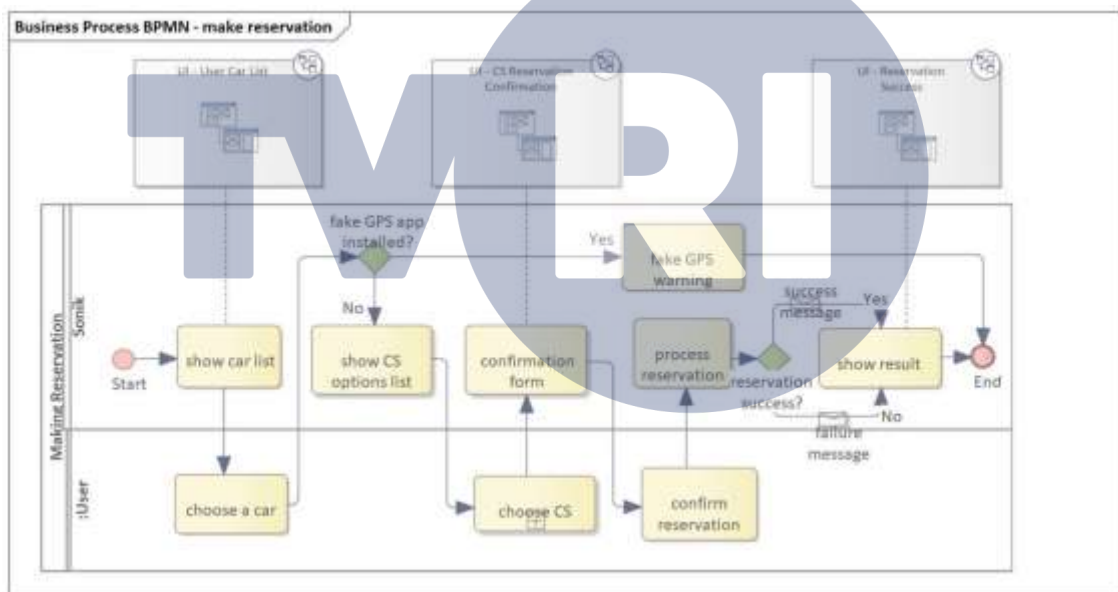
Use case aplikasi SONIKApp ditunjukkan pada Gambar 1. Proses bisnis reservasi secara lengkap ditunjukkan pada **Error! Reference source not found.**

- *Choose CS* (Memilih CS). Pengguna dapat melihat daftar CS dan memilih lokasi CS untuk melakukan reservasi dan pengisian daya melalui Aplikasi SONIKApp. CS yang tersedia adalah CS yang aktif, tidak sedang direservasi, dan berada dalam rentang jarak tertentu dari pengguna.
- *Reservasi CS*. Pengguna dapat melakukan reservasi CS melalui Aplikasi SONIKApp. Jika seorang pengguna melakukan reservasi, pengguna tersebut tidak akan diizinkan untuk melakukan reservasi di CS yang lain sebelum reservasi yang telah dilakukan, dituntaskan atau dibatalkan terlebih dahulu.
- *Start Charging*. Pengguna dapat memulai pengisian daya melalui Aplikasi SONIKApp.

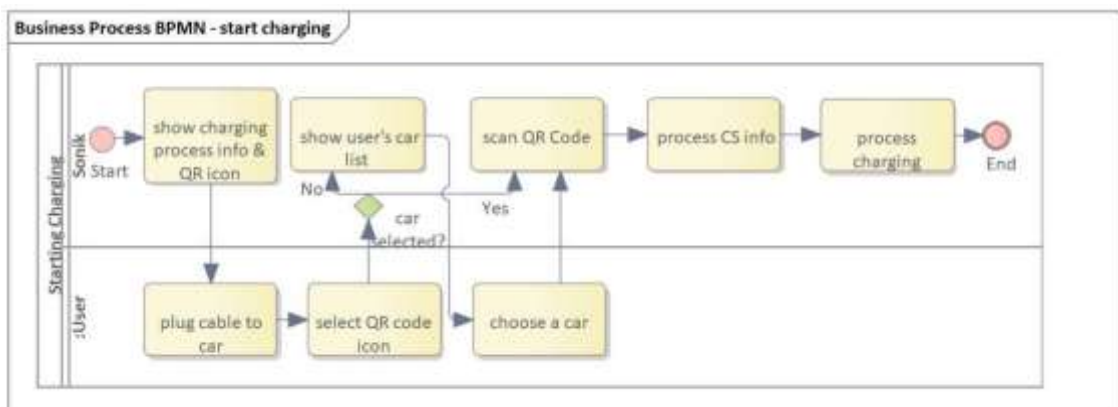
- *Stop Charging*. Pengguna dapat menghentikan pengisian daya melalui Aplikasi SONIKApp.
- *Pay Billing*. Pengguna dapat melakukan pembayaran transaksi pengisian daya melalui Aplikasi SONIKApp.



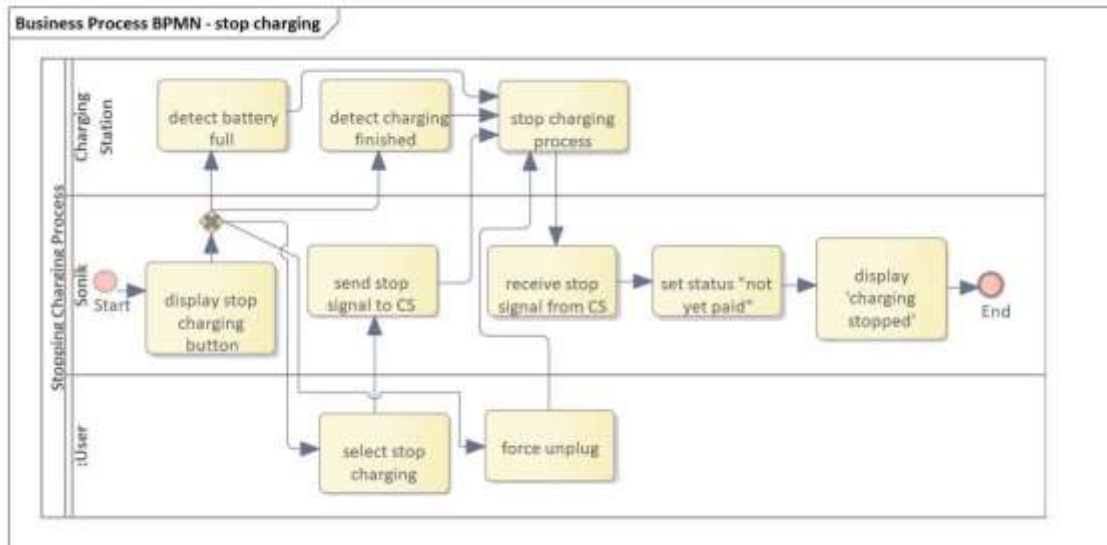
Gambar 1 Use Case Aplikasi SONIKApp



Gambar 2 Proses Bisnis Reservasi SPKLU



Gambar 3 Proses Bisnis Memulai Pengisian Daya Baterai



Gambar 4 Proses Bisnis Menghentikan Pengisian Daya Baterai

Pengisian daya baterai dilakukan dengan memindai kode QR pada perangkat *Charging Station*. Oleh karena itu, pengguna harus berada di sekitar lokasi untuk menghindari kemungkinan pengguna melakukan pengisian dari mana saja karena kode QR bersifat statis pada masing-masing *Charging Station*. Pembatasan seperti ini dipertimbangkan dalam mengimplementasikan API untuk memulai pengisian daya secara *remote*. Penghentian pengisian daya hanya dapat dilakukan oleh pengguna yang memulai pengisian daya lewat aplikasi *mobile* untuk menghindari pengguna lain menghentikan pengisian. Gambar 3 dan Gambar 4 masing-masing menunjukkan proses bisnis memulai dan menghentikan pengisian daya secara *remote*.



Gambar 5 Arsitektur Sistem SONIKApp

3.3. Implementasi Backend

3.3.1 SONIKApp OCPP Server

Pada Gambar 5, anatomi *backend* SONIKApp dibagi menjadi dua bagian, yaitu CSMS SONIK-Web

dan basis data OCPP (OCPP DB CSMS). CSMS SONIK-Web berisi logika sistem untuk mengimplementasikan fitur OCPP. CSMS SONIK-Web ini berkomunikasi langsung ke perangkat CS secara asinkron melalui protokol *websocket*. Sementara itu, OCPP DB CSMS menyimpan data terkait implementasi fitur OCPP, misalnya, status data atau kondisi stasiun pengisian yang terhubung.

3.3.2 SONIKApp OCPP API

SONIKApp OCPP API merupakan modifikasi yang ditambahkan ke CSMS SONIK-Web sehingga memiliki titik akhir-titik akhir (*endpoint*) REST yang dapat digunakan untuk menjalankan perintah OCPP atau fitur yang tersedia pada mesin. Pertimbangan untuk menambahkan OCPP API ini adalah untuk menyederhanakan pemanggilan perintah OCPP melalui protokol HTTP daripada melalui *websocket* yang merupakan protokol bawaan yang digunakan oleh CSMS SONIK-Web. REST API pada komponen ini akan dipanggil oleh SONIKApp mobile API, yang kemudian dipanggil melalui aplikasi mobile.

Empat *endpoint* yang kami buat di SONIKApp OCPP API adalah *RemoteStartTransaction*, *RemoteStopTransaction*, *ReserveNowTransaction*, dan *CancelReservationTransaction* seperti yang ditunjukkan pada Tabel 1. *Request Body* untuk empat *endpoint* tersebut disesuaikan dengan metode pada CSMS SONIK-Web. Metode pada CSMS SONIK-Web sendiri telah memenuhi spesifikasi protokol OCPP 1.6.

Proses yang dilakukan tidak dapat secara langsung memperoleh *Protocol Data Unit* (PDU) yang dikirim oleh CP yang menyediakan informasi terkait status sukses atau gagal dari sebuah tugas karena CSMS SONIK-Web bersifat asinkron. Untuk mengatasi masalah tersebut, sistem akan melihat baris data pada tabel di basis data OCPP yang mengindikasikan penyimpanan data dan perubahan status untuk tugas terkait. Fungsionalitas pekerjaan

tersebut diimplementasi pada SONIKApp Transaction API.

Tabel 1 Daftar *Endpoint* pada SONIKApp OCPP API Beserta *Request Body*-nya

No.	Endpoint	Parameter Request Body
1	[POST] Reserve Now	<ul style="list-style-type: none"> • Charge Box ID • ID Tag • Connector ID • Expiry Date
2	[POST] Cancel Reservation	<ul style="list-style-type: none"> • Charge Box ID • Reservation ID
3	[POST] Remote Start	<ul style="list-style-type: none"> • Charge Box ID • ID Tag • Connector ID • CS Coordinate • User Coordinate
4	[POST] Remote Stop	<ul style="list-style-type: none"> • Charge Box ID • Transaction ID

3.3.3 SONIKApp Mobile API

SONIKApp *Mobile* API ditulis dalam Java 11 JDK menggunakan kerangka kerja *Spring Boot* 2.4.3 untuk mengembangkan layanan terkait pengguna. API-API di sistem ini akan dipanggil oleh aplikasi *mobile* SONIKApp. Administrator menggunakan sistem ini untuk mengelola data *master* seperti data model (tipe) kendaraan dan pembayaran. Otentikasi dan otorisasi pengguna juga ditangani di sistem ini. Untuk menghubungkan pengguna aplikasi *mobile* ke CSMS SONIK-Web, sistem menambahkan *field* *idToken* dari *idToken* di tabel pengguna. *IdToken* berisi pengenalan yang digunakan untuk melakukan otorisasi pada CSMS dalam bentuk string unik (OCPP 1.6 bagian 7.28. *IdToken*)(de Leeuw, McMahon, & van Zuuren, 2017). *UserId* pada *backend* memiliki relasi *one-to-one* dengan *idToken*. Dalam implementasinya, *idToken* merupakan empat belas karakter alfanumerik unik yang dihasilkan pada saat proses pendaftaran pengguna ke API perangkat *mobile*.

Untuk alasan keamanan, kami menerapkan validasi input pengguna menggunakan Pustaka *javax validation* dengan anotasi. Validasi ini mencakup pengecekan parameter yang diperlukan, pengecekan data duplikat, pengecekan format *input* seperti format *email* dan format tanggal, pengecekan karakter yang diizinkan, pengecekan panjang karakter, pengecekan batasan ukuran *file*, dan pengecekan batasan ekstensi.

3.3.4 SONIKApp Transaction API

Backend API Transaksi SONIKApp, juga ditulis dalam kerangka kerja *Spring Boot* yang berperan sebagai jembatan/penghubung antara API *mobile* dan basis data OCPP. Dengan menggunakan *backend* ini, pengguna dapat mengetahui status CS yang tersedia, status reservasi, status transaksi, dan riwayat transaksi. *Endpoint* di *backend* ini diamankan dengan kunci API dan pengaturan *cross origin* sehingga hanya API *mobile* yang dapat mengaksesnya.

Beberapa *endpoint* yang tersedia di *backend* API Transaksi adalah:

1. [GET] List Charging Station : menampilkan daftar stasiun pengisian daya yang tersedia saat ini.
2. [GET] Transactions By Tag: menampilkan daftar riwayat transaksi pengguna.
3. [GET] Current Reservation: menampilkan daftar reservasi yang sedang dilakukan pengguna.
4. [GET] Current Remote Start: menampilkan daftar transaksi pengisian daya yang saat ini dilakukan pengguna.

3.4. Implementasi Frontend

Aplikasi *mobile* SONIKApp digunakan oleh pengguna untuk memanggil perintah-perintah OCPP. Aplikasi ini memanggil *endpoint* perintah OCPP di *backend* SONIKApp Mobile, yang kemudian akan diteruskan ke *backend* OCPP API. Berikutnya, untuk memastikan keberhasilan aksi ini, pengguna memanggil *endpoint* untuk memeriksa status keberhasilan perintah OCPP pada *mobile backend*, yang hasilnya akan diambil dari SONIKApp Transaction API.

4. HASIL DAN PEMBAHASAN

Aplikasi SONIKApp ini kemudian diuji baik secara fungsional, keamanan, dan kompatibilitas.

4.1. Pengujian Memulai dan Menghentikan Pengisian Daya Baterai

Proses pengujian memulai pengisian daya baterai diawali dengan menghubungkan plug CS ke baterai mobil dan memindai kode QR pada perangkat CS melalui fitur Pindai QR pada aplikasi. Jika proses pemindaian kode QR berhasil maka proses pengisian daya baterai dimulai, dan ini berarti pengisian berjalan dengan baik, yang juga berarti bahwa API memulai pengisian daya secara *remote* berjalan seperti yang diharapkan.

Keberhasilan pengujian pengisian daya baterai kendaraan listrik secara *remote* juga terlihat dari riwayat transaksi seperti terlihat pada Tabel 3, dimana kolom durasi mengartikan berapa lama proses pengisian berlangsung. Dengan dicatatnya durasi pengisian, berarti proses pengisian daya baterai kendaraan listrik dari awal hingga akhir telah berhasil dilakukan.



Gambar 6 Skema Aliran Data Memulai Pengisian Daya

Skema aliran data pada proses memulai pengisian daya baterai secara *remote* (Gambar 6)

dimulai dari aplikasi *mobile* SONIKApp mengirimkan data ID Tag pengguna dan nomor konektor *plug* pada CS yang ingin digunakan untuk pengisian daya ke CSMS melalui *endpoint Remote Start* di SONIKApp OCPP API. Kemudian parameter-parameter tersebut akan dikirimkan CSMS ke *Charge Point* (CP). Balasan dari CP yang mengindikasikan status keberhasilan/kegagalan proses memulai pengisian daya diterima oleh CSMS. Pesan ini kemudian diteruskan ke aplikasi SONIKApp sehingga dapat diketahui oleh pengguna.



Gambar 7 Skema Aliran Data Menghentikan Pengisian Daya

Skema aliran data pada proses menghentikan pengisian daya baterai secara *remote* (Gambar 7) dimulai dari aplikasi *mobile* SONIKApp mengirimkan data ID Transaksi pengisian daya yang sedang berlangsung (diperoleh dari *endpoint Current Remote Start* di SONIKApp *Transaction* API) ke CSMS melalui *endpoint Remote Stop* di SONIKApp OCPP API. Kemudian parameter-parameter tersebut akan dikirimkan CSMS ke *Charge Point* (CP). Balasan dari CP yang mengindikasikan status keberhasilan/kegagalan proses menghentikan pengisian daya diterima oleh CSMS. Pesan ini kemudian diteruskan ke aplikasi SONIKApp sehingga dapat diketahui oleh pengguna.

Langkah-langkah pengujian fungsionalitas ini ditampilkan pada Tabel 2.

Tabel 2 Skenario Pengujian Pengisian Daya Baterai

Skenario	Keterangan Hasil	Status Kesuksesan
Pengguna mengakses menu Pengisian pada halaman utama aplikasi Sonik Mobile	Sistem berhasil menghidupkan kamera serta menampilkan halaman pengisian dan notifikasi pindai QR untuk mulai melakukan pengisian	Berhasil
Pengguna melakukan pindai kode QR di <i>charging station</i>	Sistem berhasil mengambil data <i>charging</i> dan melakukan pengisian	Berhasil
Pengguna menekan tombol "Berhenti Mengisi"	Sistem berhasil menampilkan konfirmasi Anda yakin menghentikan pengisian. Jika klik tombol Ya, Berhenti maka pengisian akan berhenti dan melakukan pembayaran. Jika klik tombol Tidak maka akan tetap melanjutkan pengisian	Berhasil
Pengguna mengakses menu Monitor Pengisian pada halaman utama	Sistem berhasil menampilkan halaman monitoring transaksi pengisian dan akan menampilkan pesan Tidak Ada Transaksi	Berhasil

Skenario	Keterangan Hasil	Status Kesuksesan
aplikasi <i>mobile</i> SONIKApp	Pengisian jika belum pernah melakukan pengisian	
Klik tombol Kembali ke Menu Utama	Sistem menampilkan halaman utama aplikasi <i>mobile</i> SONIKApp	Berhasil



Gambar 8 Proses Memulai Pengisian Daya Baterai

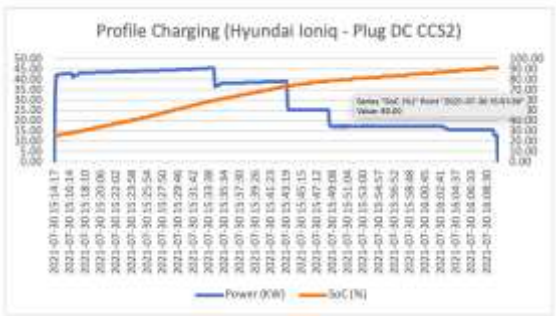


Gambar 9 Tampilan Untuk Menghentikan Pengisian Daya

Tabel 3 Sampel Riwayat Pengisian Daya

No	Nama SPKLU	Tanggal Transaksi	Durasi	Energi (WH)
1	AC Charging Station Serpong	28/03/2022	00:02:33	39
2	AC Charging Station Serpong	28/03/2022	00:19:17	510
3	EV Charger Site BPPT Puspptek	11/03/2022	00:00:25	41
4	EV Charger Site BPPT Puspptek	11/03/2022	00:01:05	96
5	CS Roda 2 EV Charger	24/01/2022	00:04:01	50
6	Site BPPT Thamrin	02/11/2021	00:00:51	75

Idealnya, pengisian baterai kendaraan listrik dengan kapasitas a KWh menggunakan perangkat pengisian daya dengan tenaga pengisian b KW memerlukan waktu a/b jam. Dalam kenyataannya proses pengisian pengisian baterai tidak berjalan linier tapi akan mulai melambat ketika sudah mencapai sekitar 80% seperti ditunjukkan Gambar 10. Grafik pengisian menunjukkan bahwa diperlukan waktu sekitar 37 menit untuk mengisi dari kapasitas awal 26% menuju 80% dan sekitar 54 menit untuk mencapai 91%.



Gambar 10 Profil Pengisian Daya Mobil Hyundai Ioniq berkapasitas 38KWh pada CS dengan kapasitas 50 KW

Ini berarti, pengguna kendaraan listrik memerlukan perlu waktu pengisian lebih lama dalam mengisi baterai kendaraan listrik dibandingkan mengisi BBM, sehingga informasi ketersediaan SPKLU diperlukan untuk mengurangi waktu tunggu pengguna kendaraan listrik.

4.2. Pengujian Reservasi SPKLU

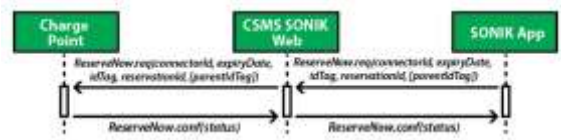
Pengujian reservasi dimulai dengan melakukan reservasi pada SPKLU yang tersedia dengan *plug* yang sesuai. Tampilan layar dari aplikasi *mobile* terlihat pada Gambar 6, mulai dari memilih SPKLU yang tersedia, melakukan reservasi, notifikasi bahwa reservasi berhasil, hingga detail reservasi dimana reservasi dapat dibatalkan.

Tabel 5 menunjukkan hasil pengujian. Berdasarkan hasil pengujian pada Tabel 5 dapat disimpulkan bahwa proses reservasi berfungsi dengan baik dimana CS memberikan respon. Artinya, API reservasi berjalan seperti yang diharapkan.

Respon “Gagal” dari CS EV *Charger Site* BPPT Puspipstek, dikarenakan fitur reservasi CS tidak diaktifkan (*disabled*). Ketika CS memiliki fitur reservasi seperti CS AC *Charging Station* Serpong, maka CS akan merespon “Berhasil” dan user akan menerima informasi bahwa CS telah direservasi atau dipesan.

Skema aliran data pada proses reservasi SPKLU (Gambar 11) dimulai dari aplikasi *mobile* SONIKApp mengirimkan data ID Tag pengguna, nomor konektor *plug* pada CS/SPKLU yang ingin digunakan untuk pengisian daya, dan tanggal kadaluarsa reservasi (pada aplikasi disetel secara *default* yakni 30 menit dari waktu saat ini) ke CSMS melalui *endpoint*

Reserve Now di SONIKApp OCPP API. Kemudian parameter-parameter tersebut akan dikirimkan CSMS ke *Charge Point* (CP). Balasan dari CP yang mengindikasikan status keberhasilan/kegagalan reservasi diterima oleh CSMS. Pesan ini kemudian diteruskan ke aplikasi SONIKApp sehingga dapat diketahui oleh pengguna.



Gambar 11 Skema Aliran Data Memulai Proses Reservasi



Gambar 12 Skema Aliran Data Membatalkan Proses Reservasi

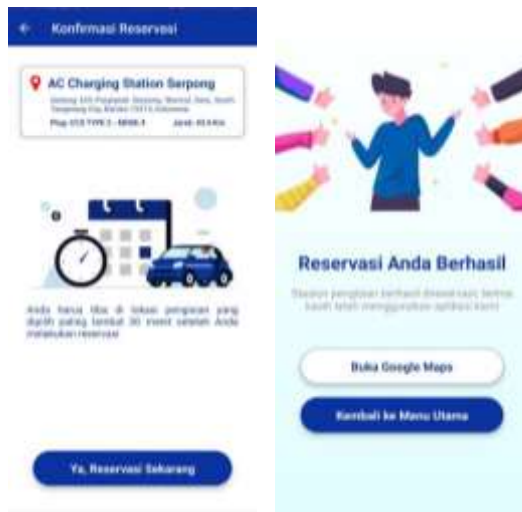
Skema aliran data pada proses membatalkan reservasi SPKLU (Gambar 12) dimulai dari aplikasi *mobile* SONIKApp mengirimkan data ID Reservasi yang sedang berlangsung (diperoleh dari *endpoint Current Reservation* di SONIKApp *Transaction API*) ke CSMS melalui *endpoint Cancel Reservation* di SONIKApp OCPP API. Kemudian parameter-parameter tersebut akan dikirimkan CSMS ke *Charge Point* (CP). Balasan dari CP yang mengindikasikan status pembatalan reservasi diterima oleh CSMS. Pesan ini kemudian diteruskan ke aplikasi SONIKApp sehingga dapat diketahui oleh pengguna.

Langkah-langkah pengujian fungsionalitas ini ditampilkan pada Tabel 4.

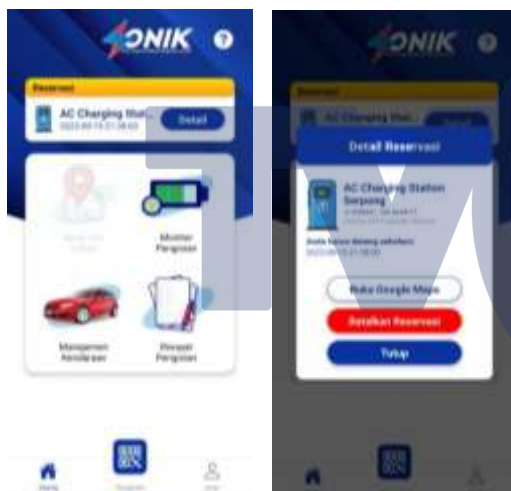
Tabel 4 Skenario Pengujian Reservasi

Skenario	Keterangan Hasil	Status Kesuksesan
Pengguna mengakses menu Reservasi Lokasi di halaman utama aplikasi Sonik Mobile dan klik tombol daftar	Sistem berhasil menampilkan halaman Pilih Kendaraan	Berhasil
Klik nama kendaraan	Sistem akan menampilkan daftar pilihan lokasi pengisian yang terdekat	Berhasil
Klik lokasi yang diinginkan	Sistem berhasil menampilkan detail Charging Station disertai peta petunjuk arah ke lokasi pengisian	Berhasil
Klik tombol Lanjutkan	Sistem berhasil menampilkan halaman konfirmasi reservasi pengisian	Berhasil
Klik tombol Ya, Reservasi Sekarang	Sistem berhasil melakukan reservasi dan menampilkan notifikasi Reservasi Anda Berhasil	Berhasil
Klik tombol detail reservasi di halaman utama	Sistem berhasil menampilkan halaman detail reservasi. Untuk membatalkan reservasi	Berhasil

Skenario	Keterangan Hasil	Status Kesuksesan
klik tombol Reservasi	Batalkan	



Gambar 13 Tampilan Memulai Proses Reservasi



Gambar 14 Tampilan Detail Reservasi

Tabel 5 Hasil Pengujian Reservasi

No	Nama SPKLU	Jenis Plug	Waktu Reservasi	Respon SPKLU
1	EV Charger Site BPPT Puspiptek	Type 2 (AC)	14/10/2021	Gagal
2	EV Charger Site BPPT Puspiptek	Chademo (DC)	11/08/2022	Gagal
3	EV Charger Site BPPT Thamrin	CCS (DC)	09/12/2021	Berhasil
4	EV Charger Site BPPT Bandung	CCS (DC)	03/10/2021	Berhasil
5	EV Charger Site BPPT Thamrin	ChaDeMo (DC)	06/10/2021	Berhasil
6	EV Charger Site BPPT Thamrin	Type 2 (AC)	06/12/2021	Berhasil

4.3. Pengujian Keamanan API

Pengujian keamanan API dibagi berdasarkan masalah keamanan, yaitu paparan data terkait kebocoran data sensitif, otentikasi bermasalah terkait aplikasi, *cross site scripting* terkait serangan dalam bentuk *code injection*. Pengujian dilakukan dengan menggunakan perangkat lunak yang memiliki fitur pengujian keamanan API. Hasil pengujian ditunjukkan pada Tabel 6. Dari empat aspek yang diuji, aspek kebocoran data menunjukkan hasil "Berisiko Rendah", yang berarti harus dipastikan bahwa URL *endpoint* API tidak menampilkan informasi sensitif seperti kunci API atau *token* sesi.

Tabel 6 Hasil Pengujian API

No	Masalah Keamanan	Hasil
1	Kebocoran Data	Resiko Rendah
2	Kerusakan Otentikasi	Memenuhi Standar Keamanan
3	Serangan Cross site scripting (XSS)	Memenuhi Standar Keamanan
4	Serangan Injeksi (SQLi, LDAP, CRLF)	Memenuhi Standar Keamanan

4.4. Pengujian Kompatibilitas Aplikasi

Pengujian kompatibilitas aplikasi SONIKApp juga dilakukan di atas beberapa jenis ponsel pintar bersistem operasi *Android*. Tabel 7 menunjukkan bahwa aplikasi SONIKApp berhasil di-*instal* dan dapat berjalan dengan baik sesuai dengan fungsi dan tampilan yang diharapkan di gawai-gawai tersebut.

Tabel 7 Hasil Pengujian Kompatibilitas SONIKApp

Jenis Perangkat	Resolusi Layar (piksel)	Sistem Operasi	Hasil
Samsung J6+	720 × 1480	Android 8.1 (Oreo)	Kompatibel
Samsung Note 10	2280 × 1080	Android 11	Kompatibel
OnePlus 5T	1080 × 2160	Android 10	Kompatibel
Sony Xperia XZ Premium	3840 × 2160	Android 9	Kompatibel
Xiaomi Redmi Note Pro 9	2400 × 1080	Android 11	Kompatibel
Samsung S10+	2280 × 1080	Android 11	Kompatibel

5. KESIMPULAN DAN SARAN

Sistem aplikasi mobile yang diberi nama SONIKApp telah dikembangkan. Aplikasi *mobile* ini dapat digunakan untuk mencari lokasi SPKLU terdekat. SONIKApp memiliki fitur-fitur utama yang terkait dengan pengisian baterai kendaraan listrik seperti pencarian lokasi SPKLU terdekat, reservasi dan membatalkan reservasi serta memulai dan menghentikan pengisian daya baterai kendaraan listrik melalui aplikasi. SONIKApp sudah teruji baik dari aspek fungsional, keamanan API, dan kompatibilitas pada sejumlah perangkat.

Aplikasi ini masih perlu ditingkatkan dengan penyempurnaan fitur pembayaran, penyesuaian dengan standar OCPP terbaru (versi 2.01), dan penggunaan arsitektur *microservices* untuk meningkatkan kinerja *backend* agar mampu melayani permintaan dalam jumlah besar.

DAFTAR PUSTAKA

- AJI, P., RENATA, D. A., LARASATI, A., & RIZA., 2020. Development of Electric Vehicle Charging Station Management System in Urban Areas. *International Conference on Technology and Policy in Energy and Electric Power (ICT-PEP)*.
- ASAAD, M. I., 2021, September 1. Retrieved from Gatrik ESDM: https://gatrik.esdm.go.id/assets/uploads/download_index/files/ab04d-road-map-pengembangan-infrastruktur-kendaraan-listrik-pln-.pdf
- DE LEEUW, R., MCMAHON, B., & VAN ZUUREN, K., 2017, September 28. Open Charge Point Protocol 1.6 Edition 2 Final.
- DHARMAWAN, I., S. KUMARA, I., & BUDIASTRA, I., 2021, September. Perkembangan Infrastruktur Pengisian Baterai Kendaraan Listrik di Indonesia. *Spektrum*, 8(2), 90-101.
- FLOCEA, R., HÎNCU, A., & ROBU, A., 2022. Electric Vehicle Smart Charging Reservation Algorithm. *Sensors*, 22. Retrieved from <https://doi.org/10.3390/s22082834>
- GALBIS, A., GARCÍA, M., GARCÍA, A., KARATZAS, S., CHASSIAKOS, A., LAZARI, V., & AGELI, O., 2022. Smart Tool Development for Customized Charging Services to EV Users. *World Electric Vehicle Journal*, 13(145). Retrieved from <https://doi.org/10.3390/wevj13080145>
- GOADRIC, M. H., & ROGERS, M. P., 2011. Smart smartphone development: iOS versus Android. *42nd ACM Technical Symposium on Computer Science Education*, (pp. 607-612).
- GÜLCÜOĞLU, E., USTUN, A. B., & SEYHAN, N., 2021. Comparison of Flutter and React Native Platforms. *Journal of Internet Applications and Management*, 12(2), 129-143.
- International Energy Agency., 2022. Global Electric Vehicle Outlook 2022.
- JOVANOVIĆ, Z., JAGODIĆ, D., VUJIČIĆ, D., & RANDIĆ, S., 2017. Java Spring Boot Rest WEB Service Integration with Java Artificial Intelligence Weka Framework. *UNITEH International Scientific Conference*, (p. 270). Gabrovo.
- JUNZHE, S., MIN, T., SANGWOO, H., TUNGYAN, W., & YIFAN, T., 2022. Electric vehicle battery remaining charging time estimation considering charging accuracy and charging profile prediction. *Journal of Energy Storage*, 49(104132).
- KAUSHIK, V., GUPTA, K., & GUPTA, D., 2019. React Native Application Development. *International Journal of Advanced Studies of Scientific Research*, 4(1).
- Kementrian Energi Sumber Daya Mineral. 2020. Peraturan Menteri No. 13/2020 tentang Penyediaan Infrastruktur Pengisian Listrik untuk Kendaraan Listrik. Jakarta, Indonesia.
- Kementrian Hukum dan Hak Asasi Manusia., 2019. Peraturan Presiden Republik Indonesia Nomor 55 Tahun 2019 tentang Program Percepatan Kendaraan Listrik Berbaterai Untuk Transportasi Jalan. Jakarta, Indonesia.
- MUDDALKAR, S. S., CHATURKAR, N. S., INGOLE, K. D., & WADASKAR, S. B., 2022. Electric Vehicle Charging Station Finding App. *International Journal of Advanced Research in Science, Communication and Technology*, 2(2).
- MYTHILY, M., ARUN RAJ, S., & JOSEPH, I. T., 2022. An Analysis of the Significance of Spring Boot in The Market. *Fifth International Conference on Inventive Computation Technologies*, (p. 1277).
- RIFKY, M., MURSITYO, Y., & PRAKOSO, B. S., 2022. Pengembangan Aplikasi Sistem Informasi Pendataan Bangunan (SIPBANG) Pada Dinas Pekerjaan Umum Dan Penataan Ruang (DPUPR) Kota Malang Menggunakan Framework Scrum. *Jurnal Teknologi Informasi dan Ilmu Komputer*, 9(1), 70.
- SHUBHAM, S. A., 2020. App Development using React Native, Expo and AWS. *International Journal of Trend in Scientific Research and Development*, 4(4), 1307-1309.
- VENKATA, T. P., NILADRI, D., PHANEENDRA, B., & VASUDEVA, B., 2019. Implementation of OCPP Protocol for Electric Vehicle Applications. *E3S Web of Conferences*, 87. doi:10.1051/e3sconf/20198701008
- VILČEK, T., & JAKOPEC, T., 2017. Comparative analysis of tools for development of native and hybrid mobile applications. *40th International Convention on Information and Communication Technology, Electronics and Microelectronics*, (pp. 1516-1521).