

Aplikasi Mobile dalam Sistem Parkir Pintar pada Basement TULT berbasis IoT

1st Hieronimus Mao Putra
Koban Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

hieronimuskoban@student.telkomuniversity.ac.id

2nd Uke Kurniawan Usman
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

ukeusman@telkomuniversity.ac.id

3rd Sri Astuti
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

sriastuti@telkomuniversity.ac.id

Abstrak—Parkiran basement *Telkom University Landmark Tower* adalah fasilitas penting bagi dosen dan pegawai. Namun, saat ini, pengelolaan parkir menghadapi tantangan dalam hal efisiensi dan ketersediaan tempat parkir yang sulit diakses dengan cepat. Untuk mengatasi masalah ini, kami mengusulkan solusi berbasis IoT (*Internet of Things*) yang inovatif. Dalam sistem ini, setiap tempat parkir akan dilengkapi dengan sensor parkir yang terhubung secara *real-time*. Sensor ini akan mendeteksi keberadaan kendaraan dan mengirimkan informasi tentang ketersediaan tempat ke perangkat ESP32. Server akan mengelola dan menyimpan data status tempat parkir. Data ini kemudian dapat diakses oleh pengguna melalui aplikasi seluler yang menyediakan informasi waktu nyata tentang tempat parkir yang tersedia di area pilihan pengguna. Dengan pengelolaan tempat parkir yang lebih efektif dan *real-time*, dosen dan pegawai dapat menghemat waktu yang sebelumnya terbuang untuk mencari tempat parkir. Kinerja sistem akan dianalisis berdasarkan parameter seperti keterlambatan dalam deteksi dan kecepatan respons aplikasi seluler. Dengan implementasi yang sukses, sistem ini berpotensi menjadi model bagi institusi lain dalam pengelolaan parkir yang modern dan efektif.

Kata kunci— Parkiran basement, Sistem parkir pintar, *Internet of Things*, Aplikasi Seluler

I. PENDAHULUAN

Ada banyak perangkat bergerak yang bisa di manfaatkan saat ini, salah satunya adalah teknologi IoT (*Internet of Things*) [2]. IoT (*Internet of Things*) adalah konsep objek sehari-hari yang terhubung ke internet serta mampu mengidentifikasi ke perangkat lain [3]. Nantinya, perangkat *Internet of Things* yang terdiri dari beberapa perangkat didalamnya ini akan terhubung ke sebuah aplikasi mobile yang sudah dibentuk. Pada aplikasi mobile tersebut, pengguna dapat melakukan pemantauan ketersediaan lahan parkir di area parkir basement TULT.

Dengan banyaknya pilihan komponen yang tersedia di pasar, memilih komponen yang tepat juga dapat menjadi tantangan [4]. Proses perakitan sistem parkir pintar yang rumit dapat meningkatkan biaya produksi dan memperlambat proses implementasinya. Sistem parkir pintar harus diuji secara menyeluruh untuk memastikan bahwa mereka bekerja sebaik mungkin. Nantinya, kehadiran sistem ini akan dapat memudahkan pengguna, karena pengguna tidak perlu mencari lahan parkir yang kosong dengan waktu yang lama yang akan menyebabkan penumpukan kendaraan di lokasi parkir

Banyak universitas di seluruh dunia saat ini mendukung kampanye kampus pintar. Sistem parkir, salah satu komponen yang dapat diubah, dapat diotomatisasi dan dikelola secara efisien dengan teknologi *Internet of Things (IoT)*. Masalah yang timbul dari penggunaan sistem parkir pintar yang diterapkan saat ini adalah dimana para pengendara yang akan memarkirkan kendaraannya tidak mengetahui terkait keadaan lahan parkir apakah terdapat lahan parkir yang tersedia atau tidak. Dengan penggunaan sistem parkir yang ada pada saat ini secara berkelanjutan dan meningkatnya penggunaan kendaraan bermotor, sistem parkir yang diterapkan saat ini berpotensi menyebabkan penumpukan kendaraan di lokasi parkir, terutama pada jam-jam sibuk, dikarenakan antrian kendaraan yang menunggu untuk masuk. Sistem parkir pintar dikembangkan untuk membantu pengguna mencari tempat parkir dengan menggunakan aplikasi di telepon genggam mereka [1].

II. KAJIAN TEORI

A. Internet of Things (IoT)

Internet of Things (IoT) mencakup berbagai jenis perangkat, mulai dari perangkat konsumen hingga sensor industri, yang dilengkapi dengan teknologi komunikasi seperti Wi-Fi, Bluetooth, RFID, atau jaringan seluler, dan memiliki kemampuan untuk berkomunikasi dan berinteraksi dengan perangkat lain, sistem, dan aplikasi. Jika diterapkan, *Internet of Things (IoT)* juga memiliki kemampuan untuk secara otomatis dan *real-time* mengidentifikasi, menemukan, melacak, dan memantau objek, serta memicu kejadian yang terkait. Manajemen ekonomi, operasi produksi, manajemen sosial, dan bahkan kehidupan pribadi telah sangat dipengaruhi oleh perkembangan dan penerapan komputer, internet, dan teknologi informasi dan komunikasi lainnya.

B. Mikrokontroler

Mikrokontroler adalah sirkuit terpadu yang ditempatkan dalam setiap komponen yang diperlukan untuk melakukan operasi yang diperlukan dan dapat melakukan tugas tertentu secara rutin tanpa memerlukan boks tambahan. Mikrokontroler yang dibutuhkan dalam proyek ini adalah mikrokontroler yang telah dilengkapi dengan modul *wireless*

terintegrasi yang memiliki kemampuan untuk mengirimkan data dengan kecepatan rata-rata 4000-15000 data per menit tergantung pada versi *wireless* yang didukung di frekuensi 160/240Mhz [5]. Mikrokontroler yang digunakan memiliki tegangan 5 volt sampai 12 volt dengan arus kerja rata-rata 80mA. Selain itu, mikrokontroler tersebut juga mampu terhubung dengan berbagai sensor atau perangkat lain melalui antarmuka salah satu nya GPIO (*General Purpose Input/Output*). Mikrokontroler tersebut juga biasanya sudah didukung protokol keamanan seperti WPA2, WPA3, atau keamanan lainnya untuk melindungi koneksi dan data.

C. Sensor Ultrasonik

Sensor Internet of Things (IoT) adalah perangkat yang terhubung ke internet yang dapat mengumpulkan data dari objek tertentu atau lingkungan fisik sekitarnya dan kemudian mengirimkannya ke platform atau sistem yang juga terhubung ke internet. Sensor IoT dapat mengukur dan memantau berbagai parameter, seperti suhu, kelembaban, tekanan, cahaya, gerakan, gas, dan kebisingan. Dalam proyek ini, sensor yang digunakan adalah sensor ultrasonik. Sensor jarak ultrasonik memanfaatkan gelombang bunyi ultrasonik untuk mendeteksi objek di depannya, mirip dengan cara hewan kelalawar melakukannya. Cara kerjanya adalah, pemancar gelombang ultrasonik (transmitter) mengirimkan gelombang suara ultrasonik, yang dipantulkan oleh objek, dan diterima oleh penerima (penerima) pada sensor ultrasonik. Namun, pada kenyataannya, bentuk objek atau target berbeda [6].

D. OLED Display

OLED Display untuk Internet of Things (IoT) adalah layar yang menggunakan teknologi OLED dan dipasang pada berbagai perangkat IoT. OLED memiliki beberapa kelebihan yang membuatnya cocok untuk digunakan dalam berbagai perangkat IoT, karena layar IoT biasanya memerlukan layar kecil dengan informasi yang sedikit. Selain itu, OLED dapat mengubah jumlah daya yang dikonsumsi berdasarkan jumlah piksel yang aktif, yang membuatnya lebih efisien dalam jangka panjang. Selain itu, kecepatan respons OLED Display sangat penting untuk aplikasi IoT yang membutuhkan pembaruan data dan interaksi pengguna yang responsif secara real-time.

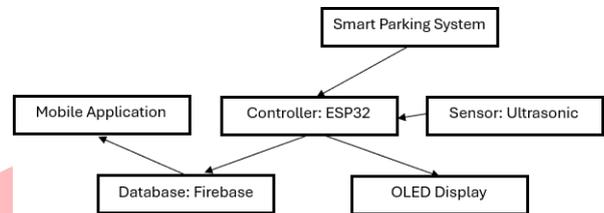
E. Android Studio

Android Studio adalah IDE resmi yang disediakan oleh Google untuk pengembangan aplikasi Android. IDE ini dirancang khusus untuk memfasilitasi pengembangan aplikasi yang berjalan di perangkat Android, seperti smartphone, tablet, TV, dan perangkat IoT berbasis Android. Android Studio menyediakan editor kode yang kuat dengan fitur seperti pelengkapan kode otomatis, refactoring, dan analisis kode untuk membantu pengembang menulis kode yang lebih efisien dan bebas dari kesalahan. Android Studio dilengkapi dengan emulator Android yang memungkinkan pengembang menguji aplikasi di berbagai konfigurasi perangkat virtual tanpa perlu perangkat fisik. Emulator ini mendukung berbagai versi Android dan ukuran layar. Android Studio menggunakan sistem build Gradle yang fleksibel dan kuat, memungkinkan pengelolaan dependensi, konfigurasi build yang kompleks, dan integrasi dengan alat otomatisasi.

III. METODE

A. Desain Solusi Sistem

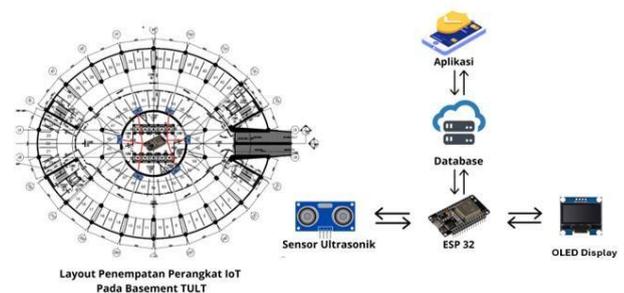
Aplikasi parkir pintar yang dibangun dalam penelitian ini terhubung ke perangkat IoT (*Internet of Things*) dan menampilkan informasi tentang ketersediaan lahan parkir yang kosong. Aplikasi ini dibuat menggunakan platform Android Studio, dan database *Firebase*. Aplikasi ini



GAMBAR 1.
Diagram Blok Sistem Parkir Pintar

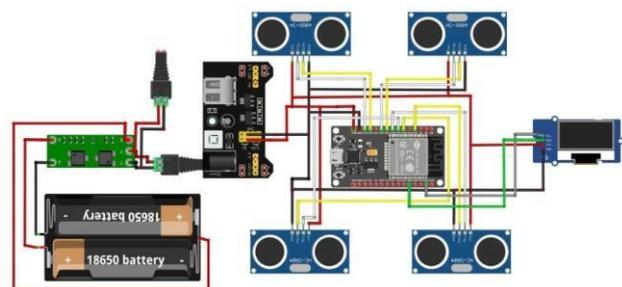
memungkinkan pengguna untuk melihat tempat parkir untuk memarkirkan kendaraannya. Sistem yang digunakan pada penelitian ini digambarkan pada blok diagram sistem Gambar 1.

Gambar 1 merupakan gambar dari diagram blok sistem parkir pintar dimana sistem parkir pintar tersebut akan terhubung ke sensor ultrasonik, mikrokontroler ESP 32, dan OLED display. Lalu mikrokontroler akan mengirimkan informasi data ke database. Sedangkan pada aplikasi mobile akan terhubung ke database.



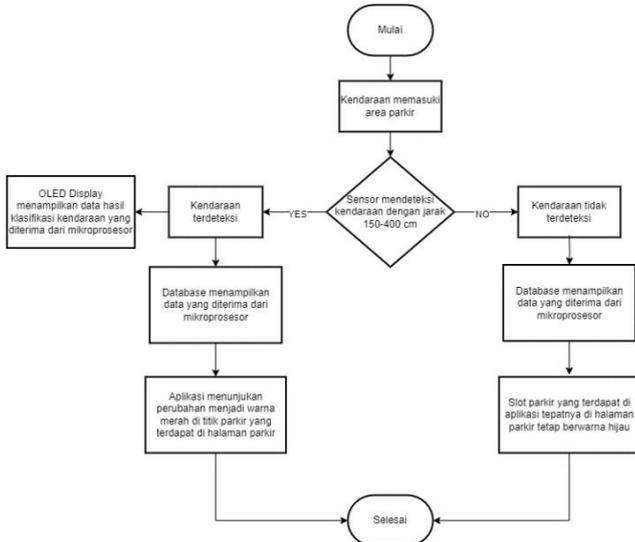
GAMBAR 2.
Desain Layout dan Diagram Blok Sistem

Gambar 2 merupakan desain layout dan diagram blok sistem. Alur kerja sistem dimulai dari sensor-sensor yang sudah diletakkan di beberapa titik di layout parkir *basement TULT*, akan terhubung ke mikrokontroler ESP 32, selanjutnya data yang ada di ESP 32 akan dikirim ke database, dan akan disimpan lalu diproses di database, sehingga hasil data yang sudah diproses akan terhubung ke aplikasi Android yang dibuat, oleh karena itu aplikasi dapat menampilkan informasi data yang ada di database tadi.



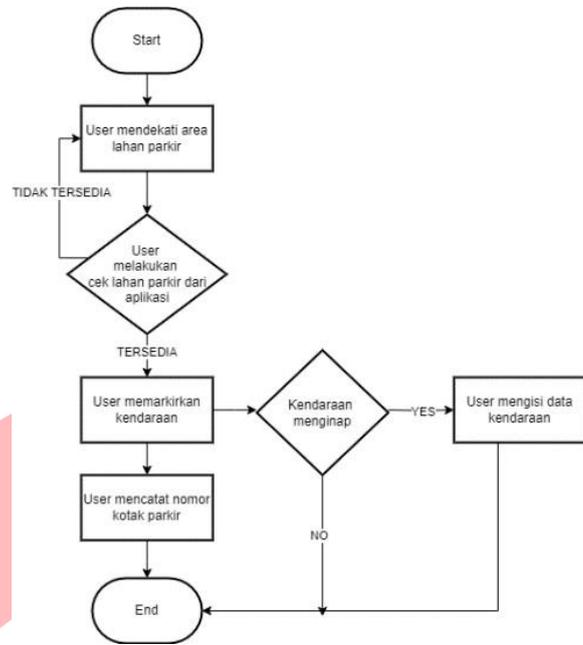
GAMBAR 1.
Wiring Sistem Parkir Pintar

Rangkaian hardware yang digunakan pada *prototype* sistem parkir pintar dapat dilihat pada gambar 3. Dimana terdapat 4 sensor ultrasonik yang digunakan yang terhubung ke mikrokontroler esp32, dimana sensor ultrasonik bertugas untuk menangkap informasi jika terdapat benda di bawah sensor dan mengirimkan data informasi tersebut ke mikrokontroler. Lalu terdapat sebuah OLED Display yang digunakan untuk menampilkan data yang didapat oleh mikrokontroler, dan sebuah rangkaian catu daya sebagai salah satu sumber daya listrik yang dapat digunakan untuk menjalankan perangkat sistem parkir pintar.



GAMBAR 2.
Flowchart Perangkat IoT

Gambar 4 merupakan proses dari keseluruhan perangkat IoT yang terhubung dengan sistem parkir pintar *basement* TULT, alur kerja sistem dimulai dari *user* memasuki area parkir, selanjutnya sensor mendeteksi kendaraan *user* dengan rentang jarak terdeteksi yaitu 150-400 cm. Apabila sensor mendeteksi, maka data yang didapat oleh mikrokontroler akan diteruskan ke OLED Display yang kemudian akan ditampilkan berupa data keadaan tiap slot parkir beserta klasifikasinya apakah kendaraan tersebut termasuk dalam kendaraan berjenis sedan, mpv, ataupun suv. Saat kendaraan terdeteksi, *database* juga akan menampilkan data yang diterima dari mikroprosesor. Apabila sensor tidak mendeteksi kendaraan, maka sistem akan lanjut melakukan pemantauan kondisi parkiran.



GAMBAR 3.
Flowchart Pengoperasian Aplikasi Sistem ParkirPintar

Gambar 5 merupakan proses interaksi yang dilakukan oleh pengguna pada *user interface* aplikasi. Proses dimulai dari melihat lahan parkir, lalu cek lahan parkir kosong, apabila tersedia maka kendaraan dapat diparkirkan, selanjutnya tercatat nomor kotak parkir lalu selesai. Apabila ingin menginap kendaraan yang lebih dari jam 10 malam, maka setelah kendaraan terparkir, pengguna mengisi data kendaraan dan selesai. Lalu, apabila lahan parkir kosong tidak tersedia maka menu akan kembali ke awal.

B. Desain Aplikasi Android

Desain aplikasi Android ini dirancang untuk memberikan pengalaman pengguna yang intuitif dan efisien dalam mengelola informasi terkait parkir. Sedangkan aplikasi Android dirancang untuk menyediakan data informasi *real-time* terkait ketersediaan kotak parkir. Informasi terkait lahan parkir (Gambar 7 (a)) yang dikirim dari server *database* nantinya akan ditampilkan dalam bentuk *graphical user interface* (GUI). Saat pengguna ingin menginap kendaraan di area parkir, maka pengguna perlu mengisi data yang ada (Gambar 7 (c)). Setelah pengguna mengisi data, maka data kendaraan akan ditampilkan pada halaman “Parkir Menginap” (Gambar 7 (b)). Fitur lain yang dapat dimanfaatkan oleh pengguna adalah fitur catatan parkir (Gambar 8 (a)), dimana pengguna dapat mencatat dimana letak kendaraan diparkirkan.



GAMBAR 4. (a) Halaman Login, (b) Halaman Register, (c)Halaman Utama



GAMBAR 5. (a)

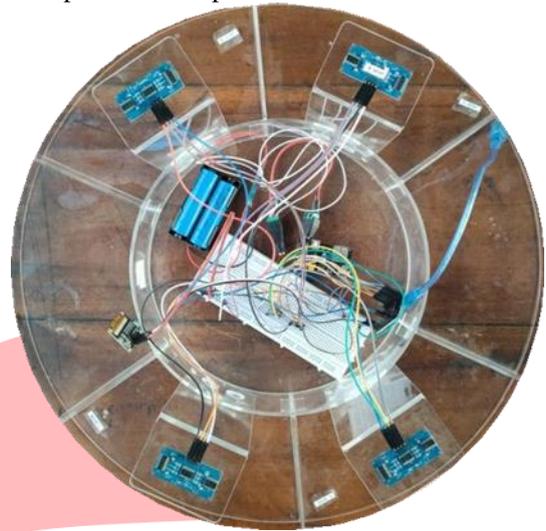
Lihat Parkir, (b) Parkir Menginap, (c) Isi Data Menginap



GAMBAR 6.

(a) Lokasi Anda, (b) Pengaturan

A. Deskripsi Umum Implementasi



GAMBAR 10.

Prototype Sistem Parkir Pintar Area Basement TULT

C. FIREBASE



GAMBAR 9.
Firebase

Pada penelitian ini, Firebase database, suatu website yang memiliki bentuk logo seperti pada gambar 9, digunakan untuk menyimpan data dan informasi pengguna. Data ini disimpan dalam database secara real-time, yang dapat digunakan oleh pengguna untuk melihat lokasi parkir tersedia. Firebase juga digunakan untuk menyimpan akun pengguna dan menyimpan data catatan jika terdapat pengguna yang akan menginapkan kendaraan.

IV. HASIL DAN PEMBAHASAN

Bagian ini berisi paparan objektif peneliti terhadap hasil-hasil penelitian berupa penjelasan dan analisis terhadap penemuan-penemuan penelitian, penjelasan serta penafsiran dari data dan hubungan yang diperoleh, serta pembuatan generalisasi dari penemuan. Apabila terdapat hipotesis, maka pada bagian ini juga menjelaskan proses pengujian hipotesis beserta hasilnya.

Hasil penelitian harus disajikan secara jelas dan sistematis supaya mudah dibaca dan dipahami. Penyajian hasil penelitian dapat dilakukan dengan cara deskriptif (naratif), menggunakan tabulasi, tabel atau grafik, atau dengan menggunakan gabungan dua atau ketiganya secara sekaligus. Penggunaan ketiga cara tersebut disesuaikan dengan jenis data dan sejauh mana deskripsi data akan dijelaskan. Misalnya, pada awal peneliti memaparkan narasi temuannya, kemudian didukung dengan sajian data dalam bentuk tabulasi, tabel atau grafik. Peneliti juga menyajikan data-data hasil penelitian, kemudian didukung grafik dilanjutkan deskripsi naratif [10 pts]. Berikan kemungkinan pengembangan atau penelitian ke depan terkait penelitian ini

Prototype yang dibuat memiliki desain seperti yang ditunjukkan pada Gambar 10. Desain ini mencakup beberapa sekat parkir yang masing-masing dilengkapi dengan sensor ultrasonik, mikrokontroler ESP32, dan layar OLED. Sensor ultrasonik berfungsi untuk mendeteksi keberadaan dan jarak kendaraan di setiap sekat parkir. Berdasarkan jarak yang terukur, sistem dapat mengklasifikasikan jenis kendaraan yang terparkir.

Setiap sensor ultrasonik terhubung ke mikrokontroler ESP32 yang mengolah data dari sensor dan mengirimkannya secara *real-time* ke *Firestore database*. Data ini kemudian dapat diakses melalui aplikasi *mobile* yang memungkinkan pengguna untuk melihat ketersediaan dan status parkir di *basement TULT*. Selain itu, informasi status parkir juga ditampilkan secara *real-time* di layar OLED yang terpasang pada *prototype*.

Selain deteksi keberadaan dan jarak kendaraan, sistem juga mampu mengidentifikasi jenis kendaraan berdasarkan data yang diterima dari sensor ultrasonik. Informasi ini berguna untuk memberikan gambaran lebih detail tentang penggunaan ruang parkir, sehingga pengelolaan parkir dapat dioptimalkan berdasarkan jenis kendaraan yang paling sering digunakan.

B. Implementasi Aplikasi Mobile

Dalam mengimplementasikan aplikasi mobile, peneliti menggunakan software android studio sebagai software utama dan java sebagai bahasa pemrogramannya. Pembuatan aplikasi mobile ini tersusun dari beberapa project atau activity seperti *MainActivity*, *LihatParkirActivity*, *TambahMenginapActivity*, dan *TambahLocationActivity*.

Project *MainActivity* berisi mengenai bagian *package* yang digunakan untuk mendefinisikan paket kelas-kelas dalam file ini. Lalu terdapat baris-baris impor yang digunakan untuk mengimpor kelas-kelas yang diperlukan dari berbagai paket Android dan pustaka *Firestore*. Selanjutnya, terdapat method *onCreate()*, yang merupakan titik awal eksekusi saat aktivitas dimulai. Pada bagian ini, aktivitas diinisialisasi dan binding dengan layout XML dilakukan menggunakan *Data Binding Library*. Terdapat juga fungsi *checkLogin()* untuk memeriksa apakah pengguna sudah login ke dalam aplikasi. Jika tidak, maka aktivitas login akan dimulai. Terakhir, terdapat bagian untuk menetapkan listener untuk *Bottom Navigation View*, sehingga ketika item

navigasi dipilih, fragment yang sesuai akan ditampilkan.

Pada project ini, terdapat beberapa metode yang digunakan yang mana memiliki fungsi yang berbeda. Metode pertama adalah metode `getData()` yang digunakan untuk mendapatkan data dari Firebase untuk setiap area parkir. Metode ini digunakan untuk mendapatkan data dari Firebase untuk setiap area parkir berdasarkan kunci yang diberikan. Lalu terdapat metode `addValueEventListener()` yang digunakan untuk menambahkan pendengar data yang akan mengeksekusi logika setiap kali data berubah. Listener ini digunakan untuk menangani perubahan data dari Firebase. Ketika data berubah, metode `onDataChange()` akan dipanggil. Jika ada kesalahan dalam membaca data, metode `onCancelled()` akan dipanggil.

Fungsi utama dalam project ini adalah untuk membuat halaman dimana user dapat mengisikan data kendaraan seandainya ingin menginapkan kendaraan. Pada project ini, terdapat metode yang digunakan untuk menyimpan data ke *database Firebase*. Data diambil dari input pengguna yang dimasukkan melalui antarmuka pengguna. Metode ini juga melakukan validasi input pengguna. Jika input tidak valid, akan ditampilkan pesan kesalahan. Setelah itu, metode membuat referensi ke *database Firebase*, membuat kunci unik untuk entri data baru, membuat objek data baru dengan nilai-nilai yang diberikan, dan menyimpan data ke *database Firebase*.

Metode juga menangani respons berhasil dan gagal dari operasi penyimpanan dengan menampilkan pesan toast sesuai.

Project `TambahLocationActivity` adalah project yang dibentuk untuk menampilkan fitur menambahkan lokasi parkir dimana kendaraan diparkirkan. Pada project ini, terdapat metode `saveVariable()` yang digunakan untuk menyimpan variabel (dalam hal ini, lokasi yang dipilih) ke dalam `SharedPreferences`. `SharedPreferences` digunakan untuk menyimpan data secara persisten dalam bentuk pasangan kunci-nilai. Di sini, variabel `key` digunakan sebagai kunci untuk mengidentifikasi data, sedangkan `value` adalah nilai yang disimpan. Setelah nilai disimpan, metode mengatur hasil ke `RESULT_OK` dan membuat intent baru tanpa data tambahan sebelum menutup aktivitas. Ini dilakukan untuk memberi tahu aktivitas pemanggil bahwa operasi telah berhasil dan aktivitas saat ini akan ditutup.

C. Pengujian & Analisis Kualitatif terhadap Aplikasi Mobile

Pengujian kualitatif dilakukan untuk mengetahui pengalaman user terhadap aplikasi yang sudah dibuat. Pengujian dilakukan dengan cara menyebarkan kuesioner kepada user yang telah diberikan akses untuk melakukan download terhadap aplikasi maupun pengujian prototype dan melihat semua fitur atau menu yang terdapat pada aplikasi. Kuesioner dibuat dalam 2 bagian dengan total 23 pertanyaan setuju/tidak setuju berskala penilaian 1-5, dimana poin 1 untuk sangat tidak setuju dan poin 5 untuk sangat setuju. Bagian pertama berisi pertanyaan terkait *user interface* seperti desain tampilan dari aplikasi secara keseluruhan, perpaduan warna dan tema, serta ukuran teks dan ikon. Sedangkan, pada bagian 2 berisi pertanyaan terkait *user*

experience seperti apakah pengguna mengalami kebingungan dalam menjalankan aplikasi dan apakah informasi yang dibutuhkan oleh pengguna dapat ditampilkan dengan baik. Hasil pengujian kualitatif dapat dilihat pada Tabel 1 berikut:

TABEL 1.
Hasil Survei terhadap Aplikasi Mobile

Question Number	Scoring Scale				
	1	2	3	4	5
Question about User Interface					
1	0	1	13	29	9
2	0	0	11	24	17
3	0	1	9	34	8
4	0	0	14	26	12
5	0	1	21	23	7
6	0	1	17	24	10
7	0	1	16	29	6
8	0	1	13	22	16
9	0	1	11	29	11
10	0	0	8	35	9
Question about User Experience and Interaksi with the Application					
1	0	0	14	31	7
2	0	1	11	29	11
3	0	0	5	34	13
4	0	0	10	25	17
5	0	0	5	29	18
6	0	0	5	35	12
7	0	0	20	27	5
8	0	0	11	35	6
9	0	1	11	29	11
10	0	0	4	30	18
11	0	1	11	34	6
12	0	1	7	31	13
13	0	0	3	32	17

Berdasarkan hasil survei yang telah dilakukan, total didapatkan sebanyak 52 responden, yang mana hasilnya dapat dilihat pada Tabel 1. Dapat terlihat bahwa rata-rata pada jawaban kuesioner, user memilih bobot penilaian bernilai 4 yang berarti user rata-rata memberikan penilaian positif atau setuju terhadap tampilan, fitur, dan interaksi terhadap aplikasi. Sebagian besar responden memiliki persepsi positif tentang aplikasi yang mereka gunakan, dapat dilihat dimana banyaknya pilihan bobot penilaian 4 menunjukkan bahwa sebagian besar merasa puas dengan berbagai aspek aplikasi tersebut.

V. KESIMPULAN

solusi prototype yang telah diciptakan dapat dianggap sebagai sukses dalam menjawab permasalahan sistem parkir yang saat ini akan diterapkan di *basement TULT*. Dalam pengujian kualitatif terhadap aplikasi yang telah dilakukan yaitu dengan cara menyebarkan kuesioner kepada *user* yang telah diberikan akses untuk melakukan download terhadap aplikasi maupun pengujian *prototype* dan melihat semua fitur atau menu yang terdapat pada aplikasi. Hasil pengujian aplikasi *mobile* sistem parkir pintar menunjukkan tingkat kepuasan yang tinggi di antara pengguna. Berdasarkan

survei, rata-rata responden memberikan point 4 pada setiap pertanyaan, yang mana poin 4 menunjukkan kategori baik. Nantinya, jika solusi ini akan diimplementasikan pada basement TULT, terdapat beberapa penyesuaian yang perlu dilakukan seperti, memasang tiang setinggi 4 meter sesuai jarak jangkauan sensor yang menjorok ke bagian tengah slot parkir, sehingga sensor dapat dengan tepat mendeteksi bagian tengah kendaraan. Selain itu, sumber daya untuk perangkat IoT yang nantinya diimplementasikan juga harus berasal dari arus listrik utama atau *AC power* karena sistem parkir pintar ini akan beroperasi terus-menerus sepanjang hari untuk memantau dan mengelola ketersediaan tempat parkir. Tentu saja mengandalkan baterai atau sumber daya terbarukan saja tidak cukup untuk memastikan operasi tanpa henti. Arus listrik utama memastikan sistem selalu aktif dan dapat diandalkan. Dalam rangkuman rencana pengembangan berkelanjutan, beberapa rencana yang telah disusun termasuk meningkatkan keandalan sensor, meningkatkan kualitas jaringan, dan meningkatkan fitur-fitur yang tersedia dalam aplikasi. Dengan demikian, solusi yang telah dibuat dekat terus dikembangkan dan ditingkatkan untuk memenuhi kebutuhan *user* yang lebih baik.

REFERENSI

- [1] M. M. S. Ismail *et al.*, "IoT Based Smart Parking System," in *Journal of Physics: Conference Series*, Institute of Physics Publishing, Dec. 2019. doi: 10.1088/1742-6596/1424/1/012021.
- [2] D. Wijayanti, N. Lestari, N. Khairani Daulay, and P. Studi Rekayasa Sistem Komputer, "Jusikom : Jurnal Sistem Komputer Musirawas PROTOTYPE SISTEM MONITORING PARKIR PINTAR BERBASIS IOT (INTERNET OF THINGS)."
- [3] T. Nursyahbani, R. Munadi, and N. B. Karna, "Pengembangan Sistem Parkir Pintar Berbasis IoT IoT-Based Smart Parking System Development."
- [4] "RANCANG BANGUN PROTOTYPE SISTEM PARKIR PINTAR BERBASIS IOT Design of IoT-Based Smart Parking System Prototype."
- [5] H. Yuliansyah Teknik Elektro, I. Teknologi Sumatera Jalan Terusan Ryacudu, D. Way Hui, K. Jati Agung, and L. Selatan, "Uji Kinerja Pengiriman Data Secara Wireless Menggunakan Modul ESP8266 Berbasis Rest Architecture," 2016.
- [6] T. Nursyahbani, R. Munadi, and N. B. Karna, "Pengembangan Sistem Parkir Pintar Berbasis IoT IoT-Based Smart Parking System Development."