

Jurnal Mahasiswa Aplikasi Teknologi Komputer dan Informasi

Vol. 8 No. 2 Thn 2026 Hal 276-284
E ISSN : 2685-6565

PROTOTYPE SISTEM PARKIR BERBASIS ESP32 UNTUK PENDETEKSIAN DAN INFORMASI STATUS SPOT PARKIR

Wikky Santoso[1], Wahyu Joni Kurniawan[2]

[a]*Institut Bisnis dan Teknologi Pelita Indonesia, Jl. Ahmad Yani No. 78-80, Pekanbaru,wikky.santoso@student.pelitaindonesia.ac.id*
[b]*Institut Bisnis dan Teknologi Pelita Indonesia, Jl. Ahmad Yani No. 78-80, wahyu.jonikurniawan@lecturer.pelitaindonesia.ac.id*

INFORMASI ARTIKEL

Sejarah Artikel:
Diterima Redaksi: 11 Juli 2026
Revisi Akhir: 30 Juli 2026
Diterbitkan *Online*: 30 Agustus 2026

KATA KUNCI

ESP32*, Internet of Things*, Reservasi Parkir*,* Sensor Ultrasonik, Sistem Parkir, *Smart Parking*.

KORESPONDENSI

E-mail: Wikky.santoso@student.pelitaindonesia.ac.id

A B S T R A C T

Permasalahan ketersediaan informasi slot parkir masih menjadi salah satu kendala pada lingkungan pendidikan dengan tingkat aktivitas kendaraan yang tinggi. Sistem parkir konvensional yang masih mengandalkan pengamatan secara manual menyebabkan pengguna kesulitan mengetahui ketersediaan slot parkir sebelum memasuki area parkir. Penelitian ini bertujuan untuk merancang dan membangun prototipe sistem parkir berbasis *Internet of Things* (IoT) yang mampu mendeteksi kondisi slot parkir serta menyediakan informasi parkir melalui aplikasi berbasis web. Sistem dikembangkan menggunakan mikrokontroler ESP32 yang terintegrasi dengan sensor ultrasonik HC-SR04 untuk mendeteksi keberadaan kendaraan pada slot parkir. Data hasil pembacaan sensor dikirimkan melalui jaringan *Wi-Fi* menuju *backend* berbasis PHP untuk diproses dan disimpan pada basis data MySQL/MariaDB. Informasi tersebut kemudian ditampilkan melalui aplikasi web yang menyediakan fitur monitoring slot parkir, reservasi pengguna, pengelolaan sistem oleh administrator, dan laporan okupansi parkir. Pengujian dilakukan pada lingkungan prototipe terkontrol dengan metode pengujian sensor, komunikasi perangkat IoT, pengujian basis data, dan pengujian fungsional menggunakan metode *Black Box Testing*. Hasil pengujian menunjukkan bahwa sistem mampu mendeteksi kondisi slot parkir, melakukan komunikasi data antara ESP32 dan backend, menyimpan informasi parkir, serta menjalankan fungsi utama aplikasi sesuai dengan rancangan.

1. PENDAHULUAN

Aktivitas mahasiswa, dosen, dan civitas akademika di Kampus Institut Bisnis dan Teknologi Pelita Indonesia (IBTPI) Pekanbaru tidak hanya berlangsung dalam kegiatan akademik sehari-hari, tetapi juga melibatkan berbagai kegiatan tambahan seperti seminar, kegiatan organisasi mahasiswa, acara institusi, maupun kegiatan lainnya yang melibatkan banyak peserta. Peningkatan aktivitas tersebut pada kondisi tertentu dapat menyebabkan meningkatnya jumlah kendaraan yang masuk ke area kampus dalam waktu yang bersamaan. Kondisi ini dapat memengaruhi penggunaan fasilitas parkir, terutama ketika jumlah kendaraan yang datang meningkat dan pengguna membutuhkan informasi mengenai ketersediaan area parkir yang tersedia.

Institut Bisnis dan Teknologi Pelita Indonesia (IBTPI) sebagai salah satu institusi pendidikan tinggi memiliki aktivitas kampus yang cukup beragam sehingga pada kondisi tertentu dapat terjadi peningkatan jumlah kendaraan, khususnya ketika berlangsung kegiatan atau acara tertentu. Berdasarkan hasil observasi dan wawancara langsung dengan pengelola parkir terhadap kondisi area parkir IBTPI, diketahui bahwa proses pencarian lokasi parkir masih dilakukan secara manual. Pengguna kendaraan perlu melakukan pengecekan secara langsung dengan berkeliling area parkir untuk menemukan slot yang masih tersedia. Selain itu, belum terdapat sistem informasi yang memberikan informasi mengenai kondisi ketersediaan slot parkir secara langsung kepada pengguna sebelum memasuki area parkir.

Kondisi tersebut menyebabkan pengguna kendaraan belum memiliki informasi awal mengenai lokasi parkir yang tersedia, sehingga proses pencarian slot parkir masih bergantung pada pengamatan secara langsung. Pergerakan kendaraan yang dilakukan secara berulang untuk mencari lokasi kosong dapat meningkatkan waktu pencarian parkir dan berpotensi menyebabkan kepadatan pada jalur kendaraan, terutama ketika terjadi peningkatan jumlah kendaraan dalam waktu yang bersamaan. Hal ini menunjukkan bahwa diperlukan suatu sistem yang mampu memberikan informasi kondisi slot parkir secara lebih efektif agar proses monitoring dan penggunaan area parkir dapat dilakukan dengan lebih terstruktur.

Berdasarkan permasalahan tersebut, diperlukan suatu solusi berbasis teknologi yang mampu memberikan informasi mengenai ketersediaan slot parkir secara cepat dan akurat. Salah satu pendekatan yang dapat diterapkan adalah pengembangan sistem parkir berbasis Internet of Things (IoT).

Teknologi IoT memungkinkan perangkat fisik seperti sensor dan mikrokontroler melakukan proses pengumpulan data, pengolahan informasi, serta komunikasi data melalui jaringan. Dalam penerapannya pada sistem parkir, sensor dapat digunakan untuk mendeteksi keberadaan kendaraan pada suatu slot, kemudian data tersebut diproses oleh mikrokontroler dan dikirimkan menuju server untuk ditampilkan melalui aplikasi monitoring.

Penelitian sebelumnya menunjukkan bahwa penerapan teknologi IoT dapat mendukung pengembangan sistem parkir cerdas (smart parking system) [3]. menunjukkan bahwa penggunaan sensor ultrasonik dan mikrokontroler dapat digunakan untuk mendeteksi keberadaan kendaraan serta memberikan informasi status slot parkir secara real-time. Penelitian [15]. mengembangkan sistem parkir berbasis IoT yang terhubung dengan jaringan WiFi dan aplikasi web, sehingga pengguna dapat memperoleh informasi ketersediaan parkir dengan lebih mudah dibandingkan sistem manual.

Oleh karena itu, penelitian ini dilakukan untuk merancang sebuah sistem parkir berbasis IoT menggunakan mikrokontroler ESP32 dan sensor ultrasonik HC-SR04. Sistem yang dikembangkan berupa prototype sebagai proof-of-concept yang mensimulasikan salah satu area parkir di lingkungan IBTPI. Implementasi sistem dibatasi pada empat titik parkir kendaraan roda empat sebagai representasi area parkir kampus. Setiap slot parkir dilengkapi sensor ultrasonik HC-SR04 untuk mendeteksi keberadaan kendaraan berdasarkan hasil pembacaan jarak terhadap objek. Status slot ditentukan berdasarkan nilai ambang batas (*threshold*) yang telah ditentukan, yaitu apabila jarak objek berada di bawah nilai batas maka slot dikategorikan sebagai terisi, sedangkan apabila jarak berada di atas atau sama dengan nilai batas maka slot dikategorikan sebagai kosong. Data hasil pendeteksian kemudian diproses oleh ESP32 dan dikirimkan melalui jaringan WiFi menuju server untuk disimpan pada database MySQL serta ditampilkan melalui aplikasi web.

2. TINJAUAN PUSTAKA

2.1. *Parkir*

Berdasarkan Undang-Undang No. 22 Tahun 2009 tentang Lalu Lintas dan Angkutan Jalan, parkir didefinisikan sebagai keadaan kendaraan berhenti atau tidak bergerak untuk beberapa saat dan ditinggalkan pengemudinya [20]. Parkir merupakan bagian terintegrasi dari sistem transportasi yang berfungsi sebagai tempat pemberhentian sementara kendaraan guna memfasilitasi aktivitas pengguna ruang jalan maupun area khusus di luar badan jalan.

Menurut [19], parkir merupakan aktivitas penyimpanan sementara kendaraan pada suatu ruang yang telah disediakan, baik di badan jalan (*on-street parking*) maupun di luar badan jalan (*off-street parking*). Manajemen parkir yang baik sangat berpengaruh terhadap efisiensi ruang kota, kelancaran sirkulasi lalu lintas, serta kenyamanan bagi para pengendara.

Sistem parkir merupakan suatu mekanisme terstruktur yang digunakan untuk mengatur, memantau, dan mengelola kendaraan yang masuk, keluar, serta menempati area parkir agar pemanfaatan ruang menjadi lebih tertib dan terkontrol [19]. Sistem ini diimplementasikan untuk mengoptimalkan kapasitas lahan, menjaga kelancaran sirkulasi, dan meminimalkan waktu hambatan sirkulasi kendaraan di dalam area parkir.

2.2. *Internet of Things*

Internet of Things (IoT) adalah sebuah konsep yang menghubungkan berbagai perangkat fisik atau objek ke jaringan internet, sehingga perangkat tersebut dapat saling berkomunikasi, bertukar data, dan mengeksekusi perintah secara otomatis tanpa memerlukan interaksi manusia secara langsung [11]. Perangkat dalam ekosistem IoT umumnya terdiri atas sensor sebagai akuisisi data, mikrokontroler atau embedded system sebagai unit pemroses, modul komunikasi untuk transmisi data, serta platform berbasis cloud untuk penyimpanan dan visualisasi informasi [18].

2.3. Perangkat Keras (*Hardware)*

Perangkat keras (hardware) merupakan seluruh komponen fisik yang membentuk suatu sistem komputer atau elektronik yang dapat diidentifikasi secara visual dan fisik [1]. Komponen ini meliputi unit pemrosesan utama, memori, media penyimpanan, perangkat masukan (input device) seperti sensor, perangkat keluaran (output device), serta modul transmisi data yang menghubungkan antarperangkat [17]. Dalam sistem embedded, perangkat keras berfungsi sebagai infrastruktur fisik untuk mengeksekusi instruksi program logis.

2.4. Perangkat Lunak (*Software)*

Perangkat lunak (software) adalah sekumpulan data elektronik, instruksi, atau program yang disimpan dan dijalankan oleh sistem komputer [1]. Perangkat lunak mencakup kode program, prosedur, dan dokumentasi terkait yang berfungsi untuk menjembatani interaksi antara pengguna dengan perangkat keras agar sistem dapat melakukan pemrosesan data secara terarah sesuai fungsi yang diinginkan [6].

2.5. Pengembangan Aplikasi Web

Pengembangan aplikasi web merupakan proses perancangan, pembuatan, pengujian, dan pemeliharaan aplikasi yang dijalankan melalui browser dengan memanfaatkan jaringan internet maupun jaringan lokal. Aplikasi web memungkinkan pengguna untuk mengakses sistem tanpa perlu melakukan instalasi perangkat lunak secara langsung pada komputer, karena seluruh proses dijalankan melalui server dan ditampilkan melalui antarmuka web.

Secara umum, pengembangan aplikasi web terdiri dari dua bagian utama, yaitu frontend dan backend. Frontend merupakan bagian yang berhubungan langsung dengan pengguna, seperti tampilan halaman, form input, dashboard, dan navigasi sistem. Backend merupakan bagian yang bertugas mengelola logika sistem, pemrosesan data, koneksi database, serta komunikasi antara server dan antarmuka pengguna.

2.6. Database

Database merupakan kumpulan data yang tersusun secara sistematis dan terorganisir sehingga dapat disimpan, dikelola, serta diakses dengan mudah oleh pengguna maupun sistem. Database digunakan untuk menyimpan informasi penting agar data dapat diproses secara efisien, akurat, dan terstruktur.

Dalam sistem informasi, database berfungsi sebagai pusat penyimpanan data seperti data pengguna, transaksi, laporan, maupun status suatu sistem. Pengelolaan database dilakukan menggunakan *Database Management System* (DBMS), yaitu perangkat lunak yang memungkinkan proses penyimpanan, pencarian, pembaruan, dan penghapusan data berjalan dengan baik.

2.7. Metode *Evolutionary Prototype*

Metode prototype merupakan pendekatan pengembangan

sistem yang membangun model awal atau rancangan sederhana sebagai representasi calon sistem sebelum pengembangan penuh dilakukan. Pendekatan ini membantu pengguna memahami bentuk dan fungsi sistem sehingga kebutuhan dapat diidentifikasi secara lebih jelas setelah pengguna berinteraksi langsung dengan prototype yang disediakan [6]. Dalam pengembangan perangkat lunak komputer, metode prototype sering dipilih ketika kebutuhan pengguna belum sepenuhnya terdefinisi atau masih bersifat ambigu pada tahap awal proyek, sehingga dibutuhkan media konkret untuk menyelaraskan pemahaman antara pengguna dan pengembang [10].

2.8. *Unified Modeling Language (UML)*

Unified Modeling Language (UML) merupakan bahasa pemodelan standar yang digunakan untuk memvisualisasikan, merancang, dan mendokumentasikan sistem perangkat lunak. UML memberikan notasi grafis yang mampu menjelaskan struktur dan perilaku sistem, sehingga sangat membantu dalam proses analisis maupun perancangan sistem. Dengan UML, pengembang dapat menggambarkan interaksi antara aktor (pengguna) dengan sistem, aliran proses, hubungan antar kelas, hingga komunikasi data antar komponen perangkat lunak.

2.8.1 *Use Case Diagram*

Use Case Diagram merupakan diagram UML yang menggambarkan interaksi antara *actor* (pengguna atau sistem eksternal) dan sistem. Fungsi utama *Use Case* Diagram adalah untuk mendokumentasikan apa yang dapat dilakukan oleh setiap *actor* terhadap sistem dari perspektif user. Diagram ini membantu mengidentifikasi kebutuhan fungsional sistem dengan menunjukkan *use cases* (fungsionalitas sistem) dan aktor yang terlibat. *Use Case* Diagram sangat berguna dalam fase mengumpulkan kebutuhan untuk memastikan bahwa semua fungsi yang dibutuhkan telah teridentifikasi dengan jelas sebelum perilisan dimulai.

2.8.2 *Activity Diagram*

Activity Diagram merupakan diagram UML yang menggambarkan aliran proses atau alur kerja dalam sistem. Fungsi utama *Activity Diagram* adalah untuk menunjukkan urutan aktivitas, *decision points* (percabangan logika), *parallel activities*, dan alur kontrol dalam sebuah proses. Diagram ini membantu memahami aliran bisnis secara menyeluruh dan mengidentifikasi area yang mungkin menjadi *bottleneck* atau *inefficiency*. *Activity Diagram* sangat berguna untuk mendokumentasikan proses bisnis, alur sistem, dan algoritma kompleks dengan cara yang visual dan mudah dipahami.

2.8.3 *Sequnce Diagram*

Sequence Diagram merupakan diagram UML yang menggambarkan interaksi antar objek atau komponen dalam sistem melalui pertukaran pesan dalam urutan waktu tertentu. Fungsi utama *Sequence Diagram* adalah untuk menunjukkan bagaimana objek berkomunikasi satu sama lain, pesan apa yang dikirim, dan dalam urutan apa pesan tersebut dikirim. Diagram ini berguna untuk memvisualisasikan komunikasi antar komponen dalam sistem, memvalidasi logika interaksi, dan mengidentifikasi interaksi yang hilang atau kondisi yang kurang.

2.8.4 *Class Diagram*

Class Diagram merupakan diagram UML yang menggambarkan struktur kelas, atribut, metode, dan hubungan antar classes dalam sistem objek orientasi. Fungsi utama *Class Diagram* adalah untuk menunjukkan desain struktur dari sistem dari perspektif program objek orientasi. Diagram ini mendokumentasikan definisi kelas kelas, properti yang dimiliki setiap class, operasi/metode yang dapat dilakukan, serta relasi antar kelas seperti pewarisan, komposisi, dan agregasi.

3. METODOLOGI

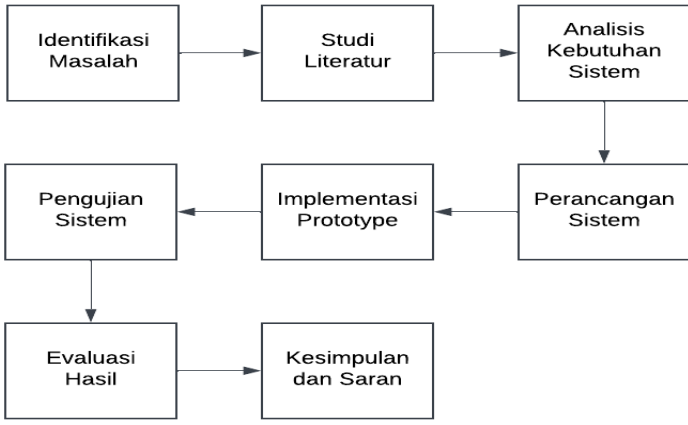
3.1 Tempat Dan Waktu Penelitian

Area parkir IBTPI digunakan sebagai objek studi kasus untuk mengetahui kebutuhan sistem, sedangkan proses implementasi dan pengujian dilakukan menggunakan prototype miniatur parkir. Pemilihan lokasi tersebut dilakukan sebagai acuan dalam proses analisis kebutuhan sistem dan perancangan alur monitoring status parkir.

Penelitian dilaksanakan pada periode Januari 2025 sampai Mei 2025 yang mencakup tahap analisis kebutuhan, perancangan sistem, implementasi prototype, serta pengujian sistem. Pengujian dilakukan pada prototype yang telah dirancang tanpa implementasi langsung pada area parkiran sebenarnya.

3.2 Kerangka Pemikiran

Adapun tahap-tahap dalam kerangka penelitian dapat dijelaskan sebagai berikut:



***Gambar 1**. Kerangka Penelitian*

3.3 Metode *Prototype (Evolutionary Prototyping)*

Metode prototype digunakan dalam penelitian ini karena sesuai dengan karakteristik pengembangan sistem parkir yang melibatkan integrasi perangkat keras dan perangkat lunak. Metode ini memungkinkan proses pengembangan dilakukan secara bertahap sehingga sistem dapat diuji dan disesuaikan sejak tahap awal. Selain itu, metode prototype memberikan fleksibilitas dalam melakukan perbaikan berdasarkan hasil evaluasi, sehingga dapat meminimalkan kesalahan desain dan memastikan sistem yang dikembangkan sesuai dengan kebutuhan penelitian.

3.4 Pengumpulan Data

Pengumpulan data dilakukan untuk memperoleh informasi yang dibutuhkan dalam rangka mencapai tujuan penelitian. Teknik yang digunakan dalam mengumpulkan data penelitian ini meliputi pengamatan langsung, eksperimen, serta studi literatur dan dokumentasi, yang diuraikan sebagai berikut: 1). Observasi, Observasi dilakukan dengan cara mengamati langsung kondisi area parkir di lingkungan Kampus Institut Bisnis dan Teknologi Pelita Indonesia (IBTPI) Pekanbaru. Observasi dilakukan untuk memperoleh data kondisi area parkir sebagai referensi dalam merancang prototype. Observasi ini tidak digunakan sebagai tahap implementasi sistem, melainkan sebagai dasar penyusunan simulasi prototype. Hasil dari pengamatan ini kemudian dijadikan acuan untuk menentukan fitur apa saja yang sangat dibutuhkan dan bagaimana sistem seharusnya bekerja nantinya. 2). Eksperimen, eksperimen menjadi teknik pengumpulan data utama karena penelitian ini berfokus pada pembuatan prototype. Proses ini dilakukan melalui tahapan merakit perangkat keras yang terdiri dari ESP32 dan sensor ultrasonik HC-SR04, kemudian memprogramnya menggunakan Arduino IDE. Selama proses

eksperimen berlangsung, data seperti nilai jarak yang terbaca oleh sensor, status logika slot parkir (kosong atau terisi), serta keberhasilan pengiriman data ke server melalui jaringan WiFi dicatat secara langsung. Data-data hasil uji coba inilah yang nantinya akan dibuktikan dan dianalisis pada bab pengujian sistem. 3). Studi Literatur, Studi literatur dilakukan dengan cara mengumpulkan dan mempelajari berbagai referensi seperti jurnal ilmiah, buku, artikel penelitian terdahulu, serta dokumentasi resmi dari komponen yang digunakan. Fokus utama dari pencarian referensi ini adalah mencari teori mengenai konsep Internet of Things (IoT), spesifikasi mikrokontroler ESP32, prinsip kerja sensor HC-SR04, serta metode perancangan sistem berbasis prototype. Selain untuk memperkuat landasan teori, data dari literatur ini juga sangat membantu dalam mencari solusi saat proses perakitan dan pemrograman menghadapi kendala teknis.

4. HASIL DAN PEMBAHASAN

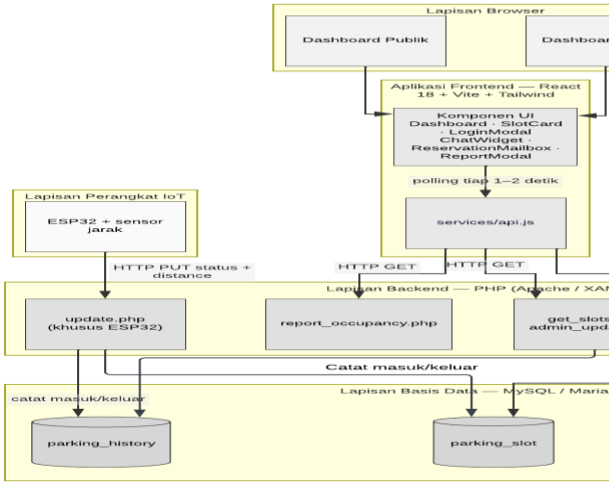
4.1. Perancangan Sistem

Perancangan sistem pada penelitian ini dilakukan untuk memetakan secara detail bagaimana alur kerja, arsitektur, logika pemrograman, serta struktur basis data dari prototype sistem parkir berbasis ESP32 yang dibangun. Tahapan perancangan ini menjadi acuan utama sebelum masuk ke proses implementasi pada perangkat keras dan perangkat lunak.

Pada tahap ini, seluruh komponen sistem, baik perangkat keras (*hardware*) maupun perangkat lunak *(software*), diintegrasikan sehingga mampu bekerja secara terpadu dalam mendukung fungsi utama sistem parkir *berbasis Internet of Things* (IoT). Secara keseluruhan, sistem yang dibangun mampu menyajikan informasi status parkir secara akurat dan real-time, serta mendukung proses monitoring dan pengelolaan parkir secara lebih efektif dan efisien.

4.1.1 Perancangan Arsitektur Sistem

Arsitektur sistem parkir yang dirancang menggambarkan hubungan antara perangkat keras, perangkat lunak, dan pengguna dalam proses *monitoring* kondisi slot parkir.

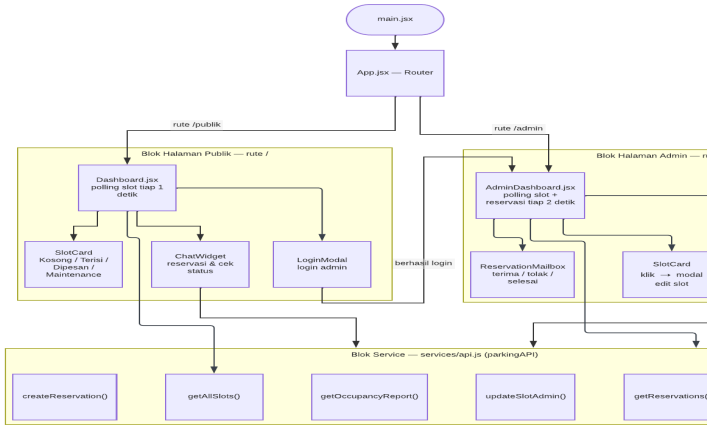


Gambar 2. *Alur Arsitektur Sistem*

Berdasarkan Gambar 2. aliran data utama sistem dimulai ketika sensor jarak mendeteksi kondisi pada slot parkir. Data hasil pembacaan sensor diproses oleh ESP32 untuk menentukan status slot, kemudian informasi status dan jarak dikirim menuju update.php melalui protokol HTTP. Backend selanjutnya memperbarui kondisi terkini pada tabel parking_slot. Apabila terjadi perubahan kondisi slot, seperti perubahan dari kosong menjadi terisi atau sebaliknya, sistem juga mencatat kejadian tersebut ke dalam tabel parking_history.

4.1.2 Perancangan Blok Diagram Sistem

Blok diagram digunakan untuk menggambarkan hubungan fungsional antarbagian utama pada sistem parkir berbasis *Internet of Things* (IoT) yang dikembangkan.

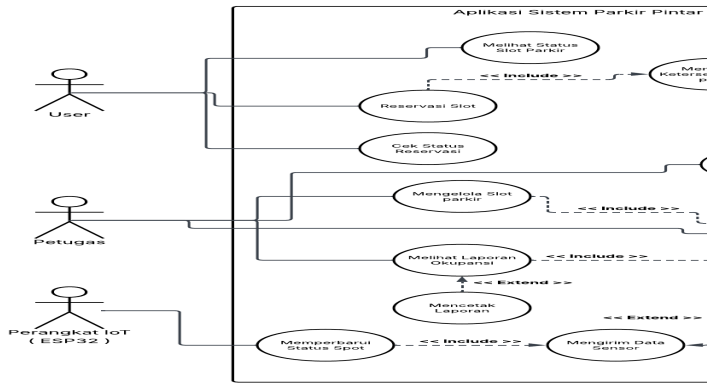


Gambar 3. *Blok Diagram Sistem*

Berdasarkan Gambar 3. Pada bagian input sistem menggunakan sensor ultrasonik HC-SR04 yang dipasang pada setiap slot parkir untuk membaca jarak antara sensor dengan objek kendaraan. Hasil pembacaan jarak tersebut menjadi parameter utama dalam menentukan kondisi slot parkir. Data dari sensor kemudian diteruskan menuju ESP32 sebagai pusat kendali sistem.

4.1.3 *Use Case Diagram*

Use Case Diagram pada sistem parkir berbasis IoT menggambarkan interaksi antara aktor dengan fitur yang tersedia pada sistem.



Gambar 4. *Use Case Diagram Sistem Parkir*

Berdasarkan Gambar 4, terdapat tiga aktor yang berinteraksi dengan aplikasi, yaitu User, Admin, dan Perangkat IoT (ESP32). Ketiga aktor memiliki peran yang berbeda sesuai dengan fungsi masing-masing dalam sistem.

User merupakan pengguna umum yang dapat mengakses *Dashboard* Publik untuk melihat informasi kondisi slot parkir, melakukan reservasi, serta melakukan pengecekan status reservasi.

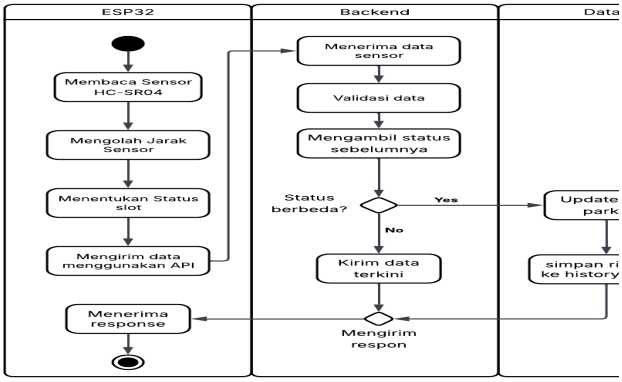
Admin merupakan pengguna dengan hak akses pengelolaan sistem. Admin dapat melakukan *login*, memantau kondisi slot, mengubah konfigurasi slot parkir, mengelola data reservasi, serta melihat laporan okupansi parkir.

ESP32 berperan sebagai perangkat IoT yang mengirimkan data hasil pembacaan sensor ultrasonik menuju sistem. Data tersebut digunakan untuk memperbarui kondisi slot parkir pada *database*.

4.1.4 *Activity Diagram*

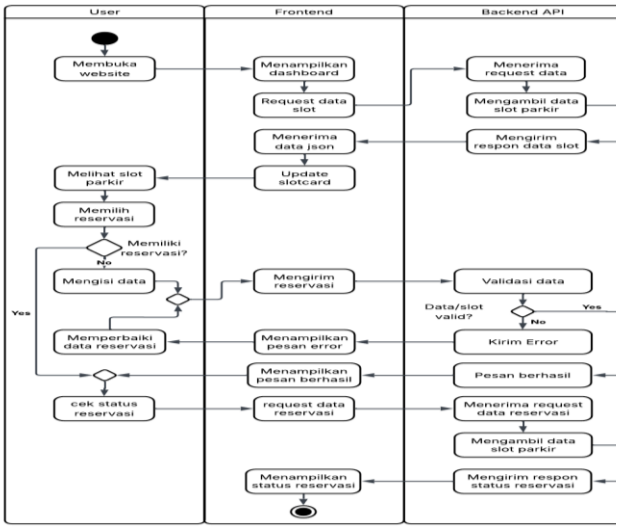
Activity Diagram digunakan untuk menggambarkan alur proses utama dalam sistem parkir berbasis IoT. Pada penelitian ini, *Activity Diagram* dibagi menjadi tiga bagian utama, yaitu aktivitas perangkat IoT dalam melakukan pendeteksian kondisi

slot parkir, aktivitas pengguna dalam menggunakan *Dashboard* Publik, dan aktivitas administrator dalam melakukan pengelolaan sistem melalui *Dashboard* Admin.



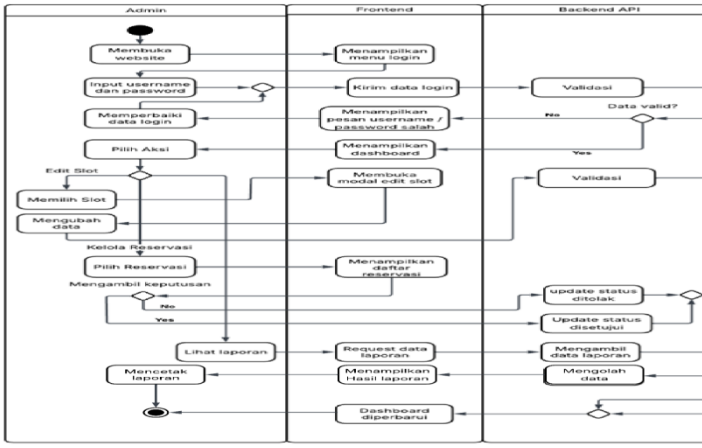
Gambar 5. *Activity Diagram Proses Monitoring IoT Berbasis ESP32*

Activity Diagram proses *monitoring* IoT Berbasis ESP32 menggambarkan proses pembacaan kondisi slot parkir oleh sensor HC-SR04, pengolahan data oleh ESP32, pengiriman data ke *backend*, hingga penyimpanan informasi pada database. Sistem menentukan status slot berdasarkan nilai *threshold*, kemudian memperbarui data kondisi parkir pada sistem.



Gambar 6. *Activity Diagram User Dashboard*

Activity Diagram User Dashboard menggambarkan aktivitas pengguna dalam mengakses *Dashboard* Publik. Pengguna dapat melihat kondisi slot parkir, melakukan pengajuan reservasi, serta melakukan pengecekan status reservasi melalui aplikasi web.



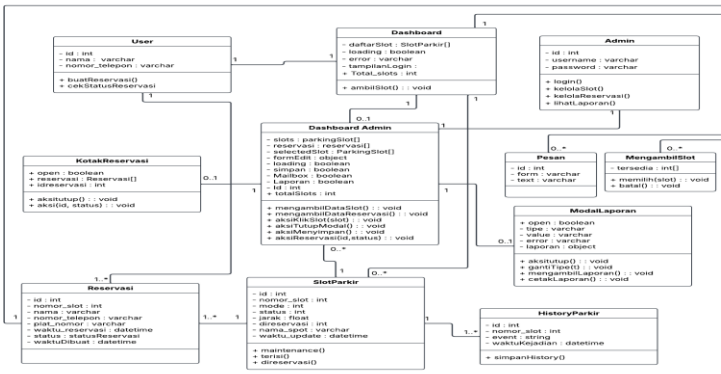
Gambar 7. *Activity Diagram Admin Dashboard*

Activity Diagram Admin Dashboard menggambarkan aktivitas administrator dalam mengelola sistem parkir. Admin dapat melakukan proses login, memantau kondisi slot, mengubah konfigurasi slot, mengelola reservasi, serta melihat laporan okupansi parkir.

4.1.5 *Class Diagram*

Class Diagram digunakan untuk menggambarkan struktur data dan hubungan antar kelas yang digunakan pada sistem parkir berbasis IoT. Diagram ini menunjukkan komponen utama yang mendukung proses *monitoring* slot parkir, pengelolaan reservasi, serta penyimpanan histori penggunaan parkir.

Pada sistem yang dikembangkan, terdapat beberapa kelas utama yaitu SlotParkir, Reservasi, dan HistoryParkir. Kelas-kelas tersebut merepresentasikan data utama yang digunakan dalam proses pengelolaan sistem.



Gambar 8. *Class Diagram Sistem Parkir Berbasis IoT*

Kelas SlotParkir digunakan untuk menyimpan informasi kondisi setiap slot parkir. Atribut pada kelas ini meliputi nomor slot, status slot, mode slot, nilai jarak sensor, nama slot, dan waktu pembaruan data. Data pada kelas ini diperoleh dari hasil pembacaan sensor HC-SR04 yang diproses oleh ESP32.

Kelas Reservasi digunakan untuk menyimpan data pengajuan reservasi yang dilakukan oleh pengguna. Data yang dikelola meliputi identitas pengguna, nomor kendaraan, slot yang dipilih, waktu reservasi, dan status pengajuan reservasi.

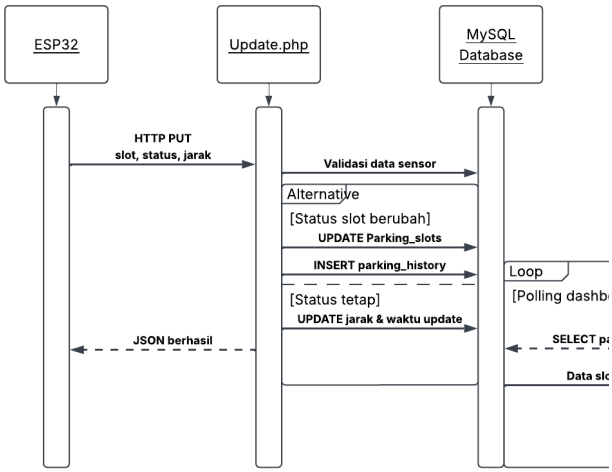
Kelas HistoryParkir digunakan untuk menyimpan riwayat perubahan kondisi slot parkir. Data histori ini digunakan sebagai sumber informasi dalam proses penyusunan laporan okupansi parkir.

4.1.6 *Sequence Diagram*

Sequence Diagram digunakan untuk menggambarkan urutan komunikasi antar komponen dalam sistem parkir

berbasis IoT. Diagram ini menunjukkan proses pertukaran data antara perangkat ESP32, *backend* API, *database*, dan aplikasi web dalam menjalankan fungsi monitoring serta pengelolaan parkir.

Pada penelitian ini terdapat dua Sequence Diagram utama, yaitu Sequence Diagram Sistem Parkir dan Sequence Diagram Proses Reservasi Slot Parkir.

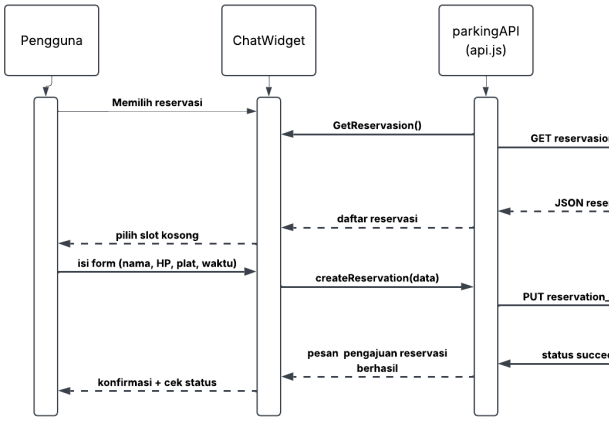


Gambar 9. *Sequence Diagram Sistem Parkir*

Sequence Diagram Sistem Parkir menggambarkan proses komunikasi antara ESP32, *backend* API, *database*, dan aplikasi web dalam melakukan monitoring kondisi slot parkir.

Proses dimulai ketika sensor HC-SR04 membaca kondisi slot parkir. Data hasil pembacaan berupa nomor slot, status parkir, dan nilai jarak sensor diproses oleh ESP32 untuk menentukan kondisi slot berdasarkan nilai *threshold*. Setelah itu, ESP32 mengirimkan data menuju *backend* API melalui jaringan *WiFi*.

Backend menerima data dari ESP32, kemudian melakukan proses validasi dan memperbarui informasi kondisi slot pada database. Apabila terjadi perubahan status slot, sistem mencatat perubahan tersebut pada tabel parking_history. Selanjutnya, aplikasi web mengambil data terbaru melalui *backend* API untuk ditampilkan pada *Dashboard* Publik dan *Dashboard* Admin.



Gambar 10. *Sequence Diagram Proses Reservasi Slot Parkir*

Sequence Diagram Proses Reservasi Slot Parkir menggambarkan alur komunikasi antara pengguna, aplikasi web, *backend* API, dan *database* dalam proses pengajuan reservasi.

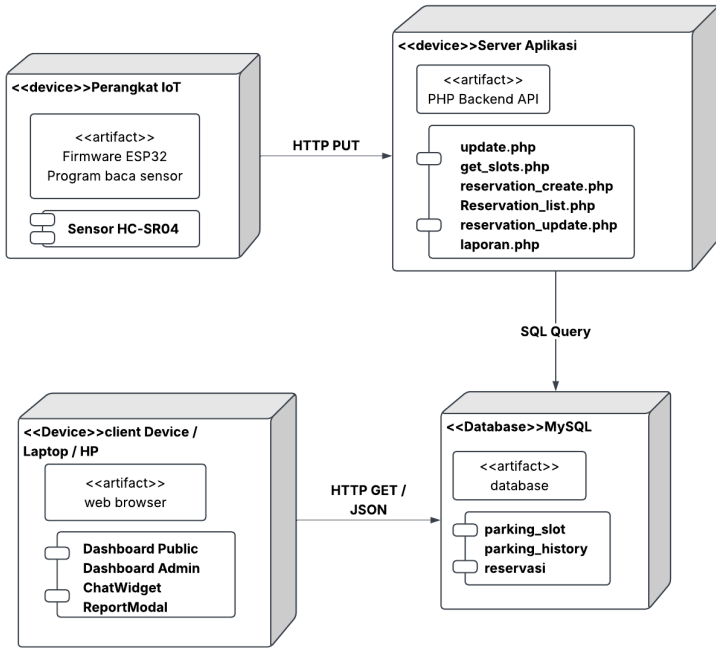
Proses dimulai ketika pengguna memilih fitur reservasi pada aplikasi web dan mengirimkan data reservasi. Data tersebut diterima oleh *backend* API untuk dilakukan validasi sebelum disimpan ke *database*.

Setelah data berhasil divalidasi, informasi reservasi disimpan pada tabel reservation. Administrator kemudian dapat melihat dan mengelola pengajuan reservasi melalui *Dashboard* Admin, seperti menerima, menolak, atau memperbarui status reservasi.

4.1.6 *Deployment Diagram*

Deployment Diagram digunakan untuk menggambarkan konfigurasi fisik komponen sistem parkir berbasis IoT serta hubungan komunikasi antar perangkat yang digunakan. Diagram ini menunjukkan bagaimana perangkat keras, perangkat lunak, dan media komunikasi saling terhubung dalam mendukung proses pendeteksian, pengolahan, penyimpanan, dan penyajian informasi parkir.

Pada sistem yang dikembangkan terdapat empat komponen utama, yaitu perangkat IoT, server aplikasi, *database*, dan perangkat pengguna.



Gambar 11. *Deployment Diagram Sistem Parkir Berbasis IoT*

Perangkat IoT terdiri dari mikrokontroler ESP32 dan empat sensor ultrasonik HC-SR04 yang dipasang pada masing-masing slot parkir. Sensor berfungsi membaca jarak objek kendaraan, kemudian data hasil pembacaan diproses oleh ESP32 untuk menentukan kondisi slot parkir. Setelah proses pengolahan selesai, ESP32 mengirimkan data berupa nomor slot, status parkir, dan nilai jarak sensor menuju server aplikasi melalui jaringan *WiFi*.

Server aplikasi berfungsi sebagai penghubung antara perangkat IoT, *database*, dan aplikasi web. *Backend* menerima data yang dikirimkan oleh ESP32, melakukan proses pengolahan data, serta menyediakan layanan API yang digunakan oleh aplikasi *frontend*. Pada sistem ini, *backend* menangani beberapa proses utama, yaitu penerimaan data sensor, pengelolaan kondisi slot parkir, proses reservasi, pengelolaan konfigurasi slot oleh admin, serta penyediaan data laporan okupansi.

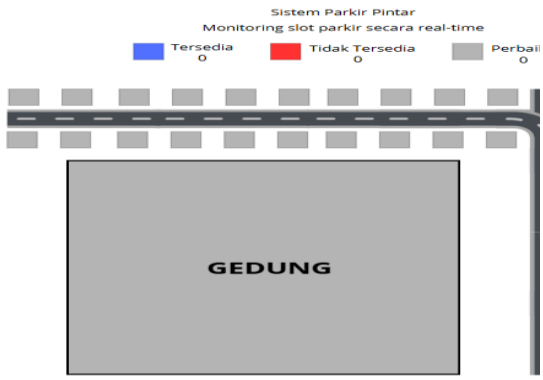
4.2 Perancangan Antarmuka Pengguna

Perancangan antarmuka pengguna dilakukan untuk menentukan tampilan dan struktur interaksi pengguna dengan sistem parkir berbasis IoT. Antarmuka dirancang agar pengguna dapat memperoleh informasi kondisi slot parkir dengan mudah, sedangkan administrator dapat melakukan

pengelolaan sistem melalui halaman khusus. Pada sistem yang dikembangkan, antarmuka dibagi menjadi dua bagian utama, yaitu: 1). Dashboard Publik, yang digunakan pengguna umum untuk melihat kondisi slot parkir dan melakukan reservasi. 2). Dashboard Admin, yang digunakan administrator untuk melakukan monitoring dan pengelolaan sistem.

4.2.1 Perancangan Halaman Dashboard User

Halaman Dashboard User dirancang sebagai halaman utama bagi pengguna umum untuk memperoleh informasi mengenai ketersediaan slot parkir.



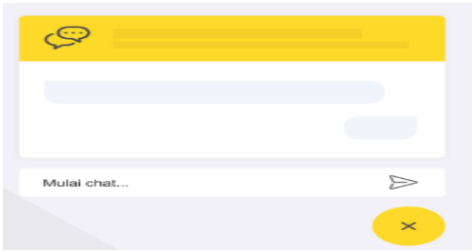
Gambar 12. Desain Halaman Dashboard User

Setiap slot parkir menampilkan beberapa informasi penting, antara lain: Nama slot parkir (misalnya A1, A2, B1),Status parkir (kosong atau terisi), Indikator visual (warna hijau untuk kosong, merah untuk terisi)

Selain itu, pada bagian atas dashboard ditampilkan ringkasan jumlah slot, seperti jumlah slot tersedia, slot terisi, dan slot dalam kondisi maintenance. Informasi ini bertujuan untuk memberikan gambaran cepat kepada pengguna mengenai kondisi parkir secara keseluruhan.

4.2.2 Perancangan Fitur Reservasi Pengguna

Fitur reservasi dirancang untuk memungkinkan pengguna melakukan pemesanan slot parkir sebelum menggunakan area parkir.

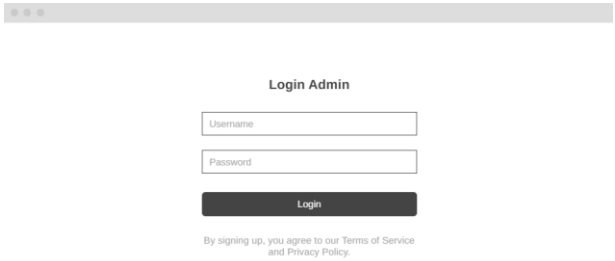


Gambar 13. Perancangan Fitur Reservasi

Proses reservasi dilakukan melalui komponen ChatWidget yang menyediakan beberapa tahapan, yaitu: 1). Memilih fitur reservasi. 2). Memilih slot parkir yang tersedia. 3). Mengisi data pengguna dan kendaraan. 4). Mengirimkan pengajuan reservasi. 5). Melihat status pengajuan reservasi. Data yang dikirim pengguna akan diproses oleh backend dan disimpan pada tabel reservation.

4.2.3 Perancangan Halaman Login Admin

Halaman login admin digunakan sebagai gerbang autentikasi sebelum admin dapat mengakses sistem.



Gambar 14. Desain Halaman Login Admin

Pada halaman ini terdapat dua input utama, yaitu username dan password, serta tombol login untuk melakukan proses autentikasi.

4.2.4 Perancangan Halaman Dashboard Admin

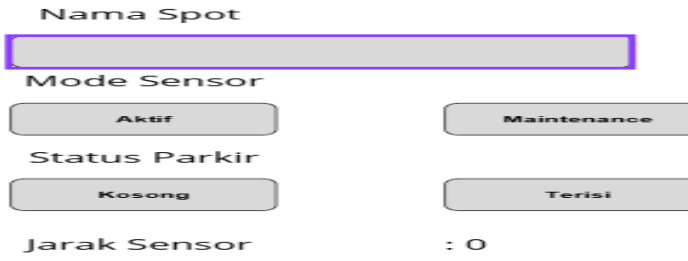
Halaman dashboard admin dirancang untuk memberikan kontrol penuh terhadap sistem parkir. Pada halaman ini, admin dapat melihat dan mengelola seluruh slot parkir yang tersedia.



Gambar 15. Desain Halaman Dashboard Admin

Pada halaman ini administrator dapat melakukan beberapa aktivitas, yaitu: monitoring kondisi seluruh slot parkir. mengubah konfigurasi slot. mengelola reservasi pengguna. melihat laporan okupansi parkir.

Informasi setiap slot ditampilkan dalam bentuk kartu yang berisi: nama slot. status slot. mode slot. nilai jarak sensor. Administrator juga dapat membuka menu pengaturan slot untuk melakukan perubahan data sesuai kebutuhan sistem.



Gambar 16. Desain Jendela Modal Pengaturan Slot Parkir

Jendela modal tersebut menyediakan fitur kontrol spesifik bagi admin, yang meliputi: perubahan nama slot. pengaturan mode slot (aktif/maintenance). perubahan status slot secara manual. melihat nilai pembacaan sensor terakhir.

Perancangan ini digunakan untuk membantu administrator melakukan pengelolaan ketika terdapat kondisi tertentu, seperti gangguan sensor atau kebutuhan pemeliharaan sistem.

4.2. Implementasi Sistem

4.2.1 Implementasi Perangkat Keras

Implementasi perangkat keras dilakukan dengan merakit komponen utama sistem sesuai rancangan yang telah dibuat sebelumnya. Komponen utama yang digunakan terdiri dari ESP32 sebagai mikrokontroler, empat sensor ultrasonik HC-SR04 sebagai pendeteksi kendaraan, serta LED indikator sebagai penanda kondisi slot parkir

4.2.2 Implementasi Firmware ESP32

Implementasi firmware dilakukan menggunakan Arduino IDE dengan bahasa pemrograman C/C++. Program pada ESP32 berfungsi untuk mengatur proses pembacaan sensor, menentukan status slot parkir, mengendalikan LED indikator, mengambil konfigurasi mode slot, serta mengirimkan data menuju backend.

4.3 Hasil Pengujian Sistem

Berdasarkan hasil pengujian yang telah dilakukan, sistem parkir berbasis Internet of Things (IoT) berhasil menjalankan fungsi utama sesuai dengan rancangan. Sensor HC-SR04 mampu mendeteksi kondisi slot parkir pada prototype, ESP32 berhasil melakukan komunikasi dengan backend, database mampu menyimpan data sistem, serta aplikasi web mampu menjalankan fungsi monitoring, reservasi, pengelolaan admin, dan laporan okupansi.

Hasil pengujian menunjukkan bahwa sistem telah mampu merepresentasikan konsep smart parking berbasis IoT pada lingkungan prototype yang terkontrol. Namun, hasil tersebut belum menggambarkan seluruh kondisi implementasi lapangan karena pengujian belum dilakukan pada area parkir sebenarnya dengan variasi lingkungan yang lebih kompleks.

4.4 Evaluasi Dan Pembahasan Sistem

Evaluasi sistem dilakukan berdasarkan hasil implementasi dan pengujian prototype sistem parkir berbasis Internet of Things (IoT). Evaluasi bertujuan untuk mengetahui kesesuaian antara sistem yang dikembangkan dengan tujuan penelitian, yaitu kemampuan sistem dalam melakukan pendeteksian kondisi slot parkir, mengirimkan dan mengolah data, serta menyediakan informasi parkir melalui aplikasi berbasis web.

Berdasarkan hasil pengujian sensor ultrasonik HC-SR04, sistem mampu mendeteksi kondisi slot parkir berdasarkan hasil pembacaan jarak terhadap objek. Penentuan kondisi slot dilakukan menggunakan nilai ambang batas (threshold) yang telah ditentukan, sehingga sistem dapat mengklasifikasikan kondisi slot menjadi kosong atau terisi. Hasil pengujian menunjukkan bahwa sensor mampu memberikan hasil deteksi yang sesuai dengan kondisi aktual pada lingkungan prototype.

Pada aspek komunikasi data, ESP32 mampu melakukan pengiriman data hasil pembacaan sensor menuju backend melalui jaringan WiFi. Data berupa nomor slot, status slot, dan nilai jarak sensor dapat diterima oleh sistem, kemudian diproses dan disimpan pada database. Hal ini menunjukkan bahwa integrasi antara perangkat IoT, backend, dan database telah berjalan sesuai dengan rancangan.

Pada aspek aplikasi web, sistem mampu menyediakan media monitoring dan pengelolaan parkir melalui Dashboard Publik dan Dashboard Admin. Dashboard Publik dapat menampilkan informasi kondisi slot parkir kepada pengguna, sedangkan Dashboard Admin dapat digunakan untuk melakukan pengelolaan slot, pengelolaan reservasi, serta penyajian laporan okupansi berdasarkan data histori penggunaan slot.

Berdasarkan hasil pengujian fungsional menggunakan metode Black Box Testing, fitur utama sistem dapat berjalan sesuai dengan rancangan. Sistem mampu menjalankan proses monitoring slot parkir, pengelolaan reservasi, pengaturan slot, serta penyajian laporan penggunaan parkir pada lingkungan prototype.

Meskipun sistem telah mampu memenuhi fungsi utama penelitian, terdapat beberapa keterbatasan yang perlu diperhatikan. Sistem masih dikembangkan dalam bentuk prototype dengan jumlah empat slot parkir sehingga belum menggambarkan kondisi penerapan pada area parkir sebenarnya dengan kapasitas yang lebih besar. Selain itu, pengujian masih dilakukan pada lingkungan terkontrol sehingga belum mencakup pengaruh faktor eksternal seperti variasi kendaraan, kondisi cuaca, gangguan lingkungan, dan kestabilan jaringan.

Secara keseluruhan, prototype sistem parkir berbasis ESP32 yang dikembangkan telah mampu mengintegrasikan sensor ultrasonik, mikrokontroler, backend API, database, dan aplikasi web dalam satu sistem terpadu. Sistem telah berhasil melakukan pendeteksian kondisi slot, penyimpanan data, serta penyajian informasi parkir kepada pengguna. Namun, diperlukan pengembangan dan pengujian lebih lanjut sebelum sistem dapat diterapkan pada area parkir sebenarnya.

5. KESIMPULAN DAN SARAN

Berdasarkan hasil perancangan, implementasi, pengujian, dan evaluasi yang telah dilakukan pada sistem parkir berbasis Internet of Things (IoT) menggunakan ESP32, sensor ultrasonik HC-SR04, backend API, database, dan aplikasi berbasis web, maka dapat diperoleh beberapa kesimpulan sebagai berikut: 1). Sistem parkir berbasis IoT berhasil dirancang dan diimplementasikan dalam bentuk prototype menggunakan ESP32 dan sensor ultrasonik HC-SR04 untuk melakukan pendeteksian kondisi slot parkir secara otomatis. Sensor mampu membaca perubahan jarak objek dan menentukan kondisi slot parkir berdasarkan nilai ambang batas (threshold) yang telah ditentukan, yaitu kondisi kosong atau terisi. Hasil pengujian menunjukkan bahwa sensor mampu melakukan klasifikasi kondisi slot sesuai dengan kondisi aktual pada lingkungan pengujian prototype. 2). Sistem pengolahan dan penyajian informasi parkir berhasil dikembangkan melalui integrasi perangkat ESP32, backend API, database MySQL/MariaDB, dan aplikasi berbasis web. Data hasil pembacaan sensor dapat dikirimkan melalui jaringan WiFi, disimpan pada database, serta ditampilkan melalui Dashboard Publik dan Dashboard Admin. Sistem juga mampu mendukung fungsi tambahan berupa pengelolaan slot, reservasi pengguna, serta penyajian laporan okupansi berdasarkan data histori parkir. 3). Berdasarkan hasil pengujian dan evaluasi sistem, seluruh komponen utama berhasil berjalan sesuai rancangan. Pengujian komunikasi ESP32 dengan server menunjukkan data dapat diterima dan diproses oleh sistem, pengujian database menunjukkan data dapat disimpan dan diperbarui dengan baik, serta pengujian fungsional menggunakan metode Black Box Testing menunjukkan fitur utama sistem dapat berjalan sesuai kebutuhan. Namun, sistem masih berada pada tahap prototype sehingga pengujian lanjutan pada area parkir sebenarnya masih diperlukan untuk mengetahui pengaruh faktor eksternal seperti kondisi lingkungan, variasi kendaraan, dan kestabilan jaringan.

DAFTAR PUSTAKA

Daftar Pustaka ditulis mengikuti format ***IEEE style*** berikut:

Referensi Cetak:

Jurnal

[1] Abdurahman, D., Nugroho, A., Sulthan Al Aziz, H., Yosua, A., Solihin Hasibuan, M., Fauziah Adha, F., Firmansyah, R., & Irsyad Noer, F. (2022). Pengenalan Software Dan Hardware Komputer Kepada Siswa Madrasah Tsanawiyah Raudhatussa'Adah. *Abdi Jurnal Publikasi*, *1*(2), 81–84.

[2] Abidin, M. Z., & Pulungan, R. (2020). *A Systematic Review of Machine-vision-based Smart Parking Systems*. *7*(2),

213–227.
[3] Allbadi, Y., Shehab, J. N., & Jasim, M. M. (2021). The Smart Parking System Using Ultrasonic Control Sensors. *IOP Conference Series: Materials Science and Engineering*, *1076*(1), 012064. https://doi.org/10.1088/1757-899x/1076/1/012064
[4] Aprilisa, S., & Aulia, R. (2024). Penerapan Metode *Prototype* dalam Pengembangan Sistem Informasi Inventory Barang Berbasis Web. *Jurnal Teknik Industri Terintegrasi*, *7*(1), 333–340. https://doi.org/10.31004/jutin.v7i1.24749
[5] C M, L., & K K, A. (2022). Spot: Smart Parking System Based on Iot. *SSRN Electronic Journal*, *9*(05), 73–77. https://doi.org/10.2139/ssrn.4298984
[6] Ii, B. A. B. (2023). *Profile Politeknik Negeri Sriwijaya*. 6–29. https://www.polsri.ac.id/profile-polsri/lambang-politeknik/
[7] Irnawati, O., Darwati, I., Irnawati, O., & Darwati, I. (2021). Evolutionary *Prototype* Dalam Perancangan Sistem. *JUSIM (Jurnal Sistem Informasi Musirawas)*, *6*(1), 1–8.
[8] Irnawati, O., Maulana, A., Kuswanto, A. D., Elektro, T., Bina, U., Informatika, S., Informasi, S., Bina, U., Informatika, S., Mini, T., Indah, I., Tiket, P., & *Prototype*, M. E. (2021). *Evolutionary Prototype Dalam Penerapan Sistem Informasi E-Ticket Planetarium*
[9] Khanna, A., & Anand, R. (2016). IoT based smart parking system. *2016 International Conference on Internet of Things and Applications, IOTA 2016*, 266–270. https://doi.org/10.1109/IOTA.2016.7562735
[10] Kustanto, P., Bram Khalil, R., & Noe'man, A. (2024). Penerapan Metode *Prototype* dalam Perancangan Media Pembelajaran Interaktif. *Journal of Students' Research in Computer Science*, *5*(1), 83–94. https://doi.org/10.31599/6x0dfz47
[11] Mphale, O., Gorejena, K. N., & Nojila, O. (2024). The Future of Things: A Comprehensive Overview of *Internet of Things* History, Definitions, Technologies, Architectures, Communication and beyond. *Journal of Information Systems and Informatics*, *6*(2), 1263–1286. https://doi.org/10.51519/journalisi.v6i2.738
[12] Nithya, R., Priya, V., Sathiya Kumar, C., Dheeba, J., & Chandraprabha, K. (2022). A Smart Parking System: An IoT Based Computer Vision Approach for Free Parking Spot Detection Using Faster R-CNN with YOLOv3 Method. *Wireless Personal Communications*, *125*(4), 3205–3225. https://doi.org/10.1007/s11277-022-09705-y
[13] Prammudya, M. A. (2020). *Sistem Informasi Inventory Berbasis Web Pada Cv . Estika ADVERTISING*. 20–26.
[14] Primus, M., Saputra, A., & Pramono, B. (2022). Penerapan Sistem Parkir Berbasis Wireless Sensor Network (WSN) Pada Bangunan Multi Lantai. *Seminar Nasional Fakultas Teknik (SNFT) UNIMAL*, *1*(1), 488–497.
[15] Rahmatillah, A., Informatika, J., Vitra, I., Informatika, P. J., Dwi, K., & Informatika, I. J. (2022). *MoParking: Sistem Monitoring Parkiran Mobil Berbasis IoT*. 1–50.
[16] Rangga, H., Putra, P., Hutavea, A. S., Permana, Y., & Utama, Y. (2025). *Pengembangan Sistem Deteksi Ruang Parkir Otomatis Menggunakan Sensor Inframerah Dan Teknologi Arduino (The development of an automatic parking space detection system using infrared sensors and arduino technology) Arduino Uno berperan sebagai mikrokontro*. *01*, 1–13.
[17] Razaqi, R. S., Yuliana, D., Suparto, A. A., & Munawwir, Z. (2024). Pelatihan Hardware dan Software untuk Perawatan dan Perbaikan Komputer. *Dst*, *4*(1), 1–6. https://doi.org/10.47709/dst.v4i1.3592
[18] Saini, S. S., Malhi, S. S., Saro, B. K., Khan, M. S., & Kaur, A. (2022). Virtual Reality: A Survey of Enabling Technologies and its Applications in IoT. *Proceedings of the International Conference on Electronics and Renewable Systems, ICEARS 2022*, 597–603. https://doi.org/10.1109/ICEARS53579.2022.9752424
[19] Susanthi, Y., Prijono, A., & Army, P. F. (2024). *Sistem Pemberi Informasi dan Visualisasi Ketersediaan Parkir pada Lahan Parkir Bertingkat Berbasis IoT Menggunakan IoT-Based System for Providing Information and Visualizing Parking Availability in Multi-Storey Parking Lots Using ESP32*. *10*(2).
[20] Teturan, Y. E., & Oja, H. (2023). Pengelolaan Lahan Parkir Untuk Peningkatan Pendapatan Asli Daerah. *Societas : Jurnal Ilmu Administrasi Dan Sosial*, *12*(1), 78–94. https://doi.org/10.35724/sjias.v12i1.5074