



Perancangan Aplikasi Sistem Parkir Otomatis menggunakan ERP Odoo Berbasis *Internet of Things*

Yusup Firmansyah^a, Rahmat Hidayat^b, Ulinnuha Latifa^c

^aProdi Teknik Elektro, Fakultas Teknik, Universitas Singaperbangsa Karawang, yhusup6@gmail.com

^bProdi Teknik Elektro, Fakultas Teknik, Universitas Singaperbangsa Karawang, rahmathidayatt377@gmail.com

^cProdi Teknik Elektro, Fakultas Teknik, Universitas Singaperbangsa Karawang, ulinnuha.latifa@ft.unsika.ac.id

Submitted: 22-04-2021, Reviewed: 28-04-2021, Accepted 29-04-2021

<http://doi.org/10.22216/jsi.v7i1.233>

Abstrak

Universitas Singaperbangsa Karawang (UNSIKA) is a favorite Public universities in Karawang, West Java. UNSIKA parking system has not met the expectations of a good parking system, especially in the field of security. This research aims to solve the problem by designing an IoT-based automated parking system application with a high level of security, informative to provide comfort for users and managers. The method in the security system is to use barcodes and fingerprints, as for the application that will be made using ERP Odoo. The results of this study are a tool called BIOPARKS (Biometric Parking System) that can reduce the risk of theft in parking lots. Also, data management applications such as vehicle data, vehicle owner data, reports in and out of vehicles, as well as a monitor of parking conditions. The use of barcode and fingerprint technology in the parking system makes the security on the parking lot more secure so that the owner of the vehicle becomes calm. Applications designed to make data management more structured, as well as facilitate monitoring of parking conditions.

Keywords: Parking, Odoo, ERP, IoT, Application

Abstrak

Universitas Singaperbangsa Karawang (UNSIKA) merupakan PTN favorit di Karawang, Jawa Barat. Sistem perparkiran UNSIKA belum memenuhi harapan terhadap sistem perparkiran yang baik terutama pada bidang keamanannya. Penelitian ini bertujuan untuk mengatasi masalah tersebut dengan cara merancang aplikasi sistem parkir otomatis berbasis IoT dengan tingkat keamanan yang tinggi, informatif sehingga memberi kenyamanan bagi pengguna maupun pengelola. Metode dalam sistem pengamanannya yaitu menggunakan *barcode* dan *fingerprint*. Adapun aplikasi yang akan dibuat ini menggunakan ERP Odoo. Hasil penelitian ini merupakan sebuah alat dengan nama BIOPARKS (*Biometric Parking System*) yang dapat menekan resiko pencurian pada parkiran. Selain itu dihasilkan juga aplikasi pengelola data seperti data kendaraan, data pemilik kendaraan, laporan keluar-masuk kendaraan, serta berfungsi juga sebagai monitor kondisi parkiran. Pemanfaatan teknologi *barcode* dan *fingerprint* pada sistem parkir membuat keamanan pada parkiran lebih terjamin, sehingga pemilik kendaraan menjadi tenang. Aplikasi yang dirancang membuat pengelolaan data menjadi lebih terstruktur, serta mempermudah monitoring kondisi parkiran.

Kata kunci: Parkir, Odoo, ERP, IoT, Aplikasi

© 2021 Jurnal Sains dan Informatika

1. Pendahuluan

Tempat parkir merupakan bagian yang tak dapat dipisahkan dari sistem transportasi. Oleh karena itu diperlukan manajemen tata kelola parkir agar pengguna parkir seperti mahasiswa, dosen, maupun karyawan merasa nyaman dan aman.

Pengembangan sistem parkir biasanya bertujuan untuk memberikan keamanan dan kenyamanan, sehingga kondisi kendaraan dapat tertata dengan baik dalam hal penempatan kendaraan dan jaminan keselamatan, serta dapat digunakan selama 24 jam [1].

Pada penelitian ini, penulis menggabungkan dua teknologi sekaligus untuk meningkatkan tingkat keamanan parkirnya, yaitu *fingerprint* dan *barcode*. Kemudian untuk manajemen datanya menggunakan aplikasi berbasis *website* yang dibuat menggunakan *framework ERP Odoo*. Sehingga dapat diakses dimanapun, kapanpun, baik di PC, laptop maupun *mobile*. Menurut Efmi Maiyana dkk, jangkauan distribusi informasi sistem berbasis web sangat luas, sehingga informasi dapat tersampaikan dengan mudah kepada masyarakat [2].

Tujuan penelitian ini adalah merancang sistem parkir otomatis yang aman dan nyaman berbasis *IoT* dan terintegrasi dengan sistem *ERP (Enterprise Resource Planning)*.

2. Tinjauan Pustaka/Penelitian Sebelumnya

Penelitian-penelitian sebelumnya yang relevan diantaranya, Rani Oktaviani dkk, membuat aplikasi sistem parkir berbasis desktop menggunakan *Java Netbeans* dan metode pengamanannya menggunakan *RFID* [3], Adlan Bagus Pradana dkk, membuat aplikasi sistem parkir berbasis *mobile* dan memanfaatkan teknologi *QRcode*[4], Muhammad Rizki Utomo dkk, menggunakan *fingerprint* untuk menambah tingkat keamanan parkir[5].

Dari beberapa penelitian yang disebutkan diatas, metode dalam meningkatkan keamanannya menggunakan satu teknologi saja seperti *RFID* saja, *barcode* saja, ataupun *fingerprint* saja. Selain itu Aplikasi berbasis desktop mempunyai kelemahan yaitu tidak portabel [6] sehingga kurang efektif karena hanya bisa diakses di komputer atau laptop yang terinstall aplikasi tersebut.

Pada penelitian-penelitian sebelumnya juga belum ada yang mengintegrasikan sistem parkir dengan sistem *ERP*, terutama dengan *ERP Odoo*

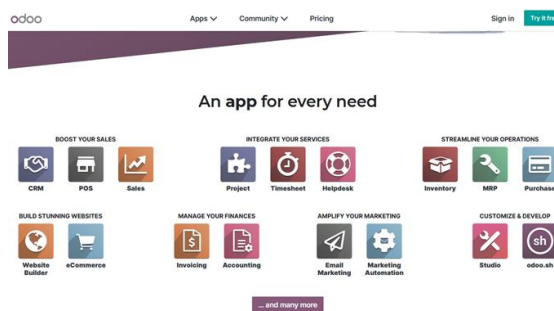
2.1 ERP (Enterprise Resource Planning)

ERP (Entreprise Resource Planning) atau yang dalam Bahasa Indonesia adalah perencanaan sumber daya perusahaan, merupakan sistem informasi untuk perusahaan yang berguna untuk mengintegrasikan dan mengotomasikan proses bisnis yang berhubungan dengan aspek operasi, produksi ataupun distribusi dari perusahaan yang bersangkutan. *ERP* terbagi ke *ERP* berlisensi dan *ERP* yang *open source*. Ada beberapa *ERP Open Source* yang dapat dipilih untuk diimplementasikan seperti; *Odoo*, *Compiere*, *Open Bravo*, dsb [7].

2.2 Odoo

Odoo adalah aplikasi berbasis web yang dibangun menggunakan Bahasa *python*, *javascript*, dan *XML*

sebagai aplikasi sistem informasi sehingga memudahkan *user* dalam mengelola data perusahaan[8]–[10].



Gambar 1. Modul-modul yang tersedia pada Odoo

Odoo dibangun menggunakan bahasa pemrograman *python*, *XML* dan *javascript* dan *postgresql* sebagai *database*-nya. Dalam sejarah pertama kali *Odoo* bernama *TinyERP*, kemudian pada tahun 2009 berganti nama menjadi *OpenERP* setelah itu dikembangkan kembali menjadi versi 8 berganti nama menjadi *Odoo* [11].

Odoo merupakan satu-satunya sistem manajemen yang tidak hanya digunakan oleh perusahaan-perusahaan besar saja, tetapi juga digunakan oleh perusahaan kecil dan independen. *Odoo* pun dapat diaplikasikan pada berbagai macam sektor, seperti sektor perdagangan, tekstil, agrikultural, dan lainnya. Perbedaan tersebut menggambarkan tingkat fleksibilitas *Odoo* yang sangat tinggi sehingga dapat menjangkau seluruh jenis perusahaan yang ada [12].

2.3 Barcode Scanner

Teknologi *barcode* memberikan keuntungan seperti membantu pembacaan dan penginputan data menjadi lebih cepat, karena *barcode scanner* merekam dan membaca lebih cepat daripada penginputan manual, Penginputan data yang dilakukan lebih tepat, mengurangi biaya dan waktu dalam proses penginputan maupun pencarian data, meningkatkan kinerja yang lebih efektif, kemampuan bersaing dengan *competitor* [13].

2.3 Fingerprint Sensor

Fingerprint sensor merupakan sebuah perangkat elektronik yang berfungsi untuk membaca sidik jari pengguna untuk keperluan verifikasi identitas [14].

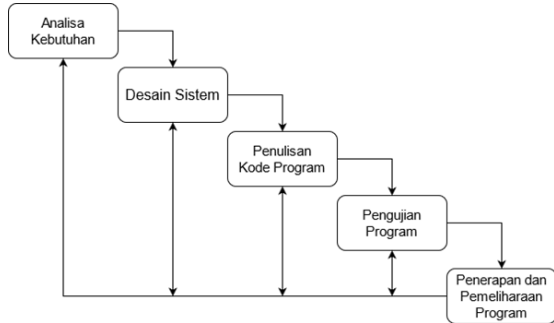
Pendeteksian sidik jari dilakukan dengan cara mengambil gambar sidik jari yang dipindai pada sensor untuk mendapatkan pola sidik jarinya[15].

Penggunaan sensor *fingerprint* dapat memberikan banyak keuntungan, seperti akurasi yang tinggi, tingkat keamanan yang tinggi, dan penggunaan yang mudah [16].

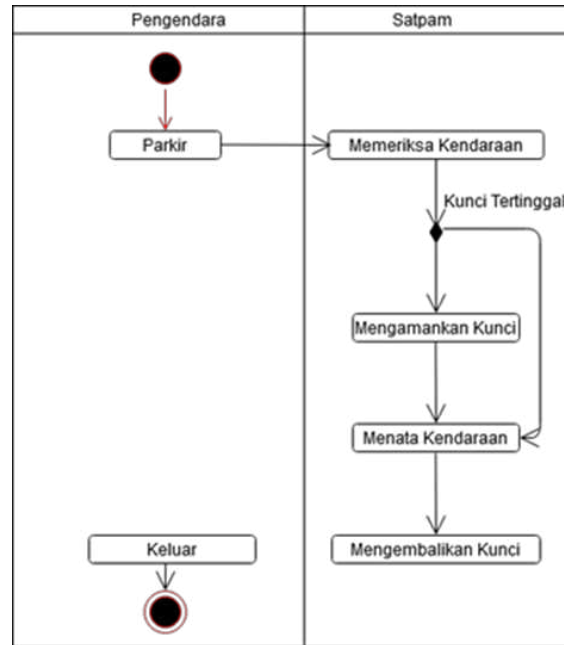
3. Metodologi Penelitian

Pada penelitian ini metode atau model yang dipakai dalam perancangan *software* adalah metode air terjun atau *waterfall development model*.

Prinsip kerja dari model ini yaitu semua aktivitas harus direncanakan dan diproses terlebih dahulu secara berurutan sebelum mulai mengerjakannya [17].



Gambar 2. Waterfall Development Model

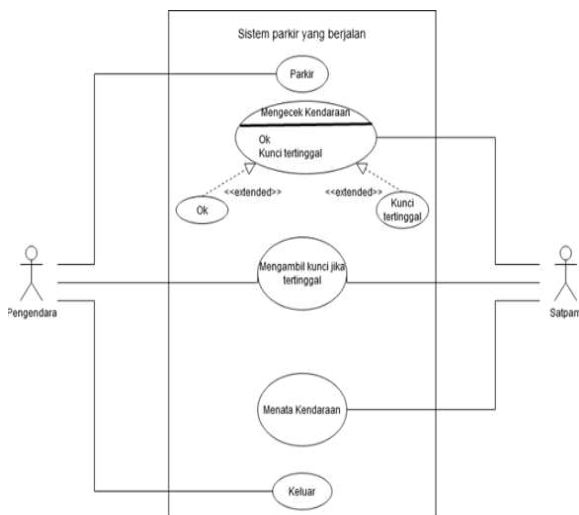


Gambar 4. Activity Diagram Sistem parkir yang berjalan di UNSIKA

Berdasarkan Metode *Waterfall Development Model* perancangan *software* dimulai dari Analisa kebutuhan, kemudian desain sistem, penulisan kode program (*coding*), pengujian, dan penerapan serta pemeliharaan program.

3.1 Analisa Kebutuhan

Sistem parkir yang berjalan di Universitas Singaperbangsa Karawang sekarang ini seperti pada Gambar 3 dan Gambar 4 berikut ini



Gambar 3. Use Case Diagram Sistem parkir yang berjalan di UNSIKA

Gambar 3 dan Gambar 4 menjelaskan bahwa sistem parkir masih bersifat manual sehingga proses mulai kendaraan masuk dan keluar maupun laporan masih dilakukan oleh satpam.

Berdasarkan data diatas, masalah-masalah yang ada pada sistem parkir Universitas Singaperbangsa diantaranya lain:

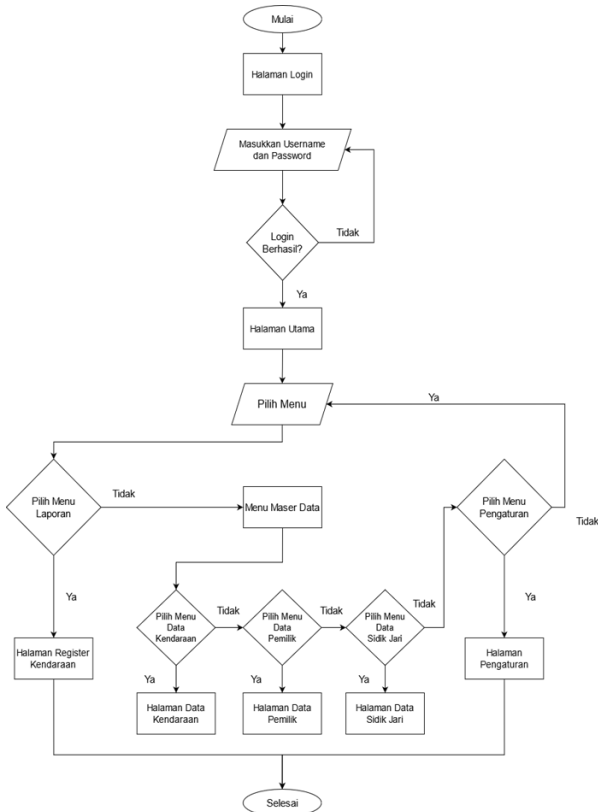
- Sistem parkir yang masih manual
- Tingkat keamanannya sangat rendah
- Tidak adanya data kendaraan
- Tidak adanya data pemilik kendaraan
- Tidak adanya data riwayat keluar-masuk kendaraan
- Tidak adanya informasi kondisi parkir

Adapun yang dibutuhkan pada sistem parkir untuk mengatasi masalah-masalah tersebut adalah sebagai berikut:

- Sistem parkir otomatis
- Tingkat keamanan yang tinggi
- Data pemilik kendaraan
- Data kendaraan
- Data riwayat keluar-masuk kendaraan
- Informasi mengenai kondisi parkir

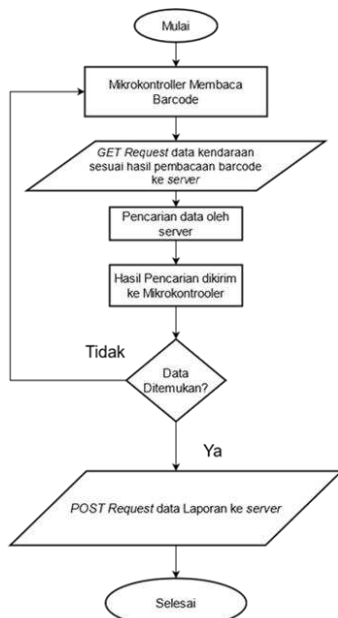
3.2 Desain Sistem

Aplikasi yang dirancang dibagi menjadi dua, yaitu aplikasi yang berbasis web dan berbasis *mobile*. Aplikasi web dibuat menggunakan *software Odoo Community Versi 13.0* dan berfungsi untuk mengolah data yang dikirim oleh *Arduino*.



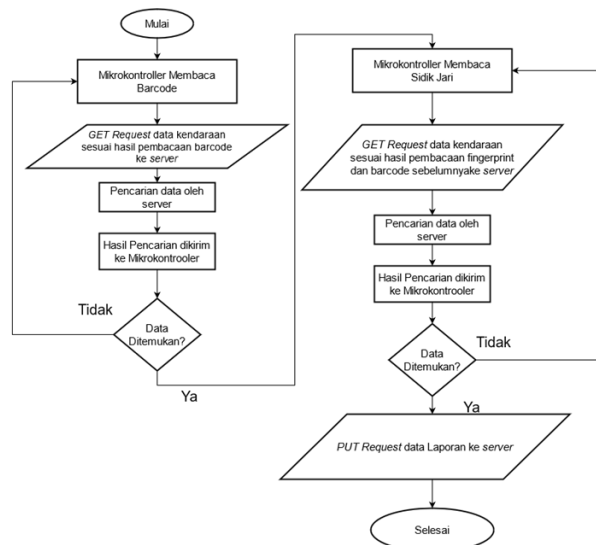
Gambar 5. Diagram Alir software

Gambar 5 menjelaskan rancangan alur dari aplikasi web yang dibuat, mulai dari halaman login dan halaman utama, dimana halaman utama berisi beberapa menu seperti menu laporan, master data, dan pengaturan. Semua data yang terbuat disimpan pada database, dan database yang digunakan adalah PostgreSQL. PostgreSQL merupakan database management system yang dapat berjalan pada sistem operasi yang paling modern termasuk Windows, MAC, dan Linux [18].



Gambar 6. Komunikasi dengan mikrokontroler pada saat masuk

Gambar 6 menjelaskan diagram alur integrasi aplikasi server dengan mikrokontroler pada saat memasuki parkir untuk validasi data. Komunikasi yang digunakan adalah HTTP, yaitu sebuah protokol komunikasi yang akan digunakan dalam WWW (World Wide Web) antar komputer yang saling terhubung dalam jaringan internet [19], [20]. Dimulai dari mikrokontroler yang melakukan pemindaian barcode lalu Mikrokontroler akan mengirimkan hasil pembacaan barcode ke server, dengan metode HTTP Request GET pada endpoint API yang telah disediakan. Setelah itu server akan mencari data kendaraan sesuai dengan yang direquest. Kemudian server akan mengirimkan kembali ke mikrokontroler hasil dari pencarian data tersebut. Jika data tidak ditemukan maka akan kembali ke pembacaan barcode. Jika data ditemukan maka mikrokontroler akan membuat data laporan kendaraan masuk parkir pada server.



Gambar 7. Komunikasi dengan mikrokontroler pada saat keluar. Gambar 7 menjelaskan diagram alur integrasi aplikasi server dengan mikrokontroler pada saat keluar parkir untuk validasi data hasil pembacaan barcode scanner dan sensor fingerprint. Mula-mula Mikrokontroler akan mengirimkan hasil pembacaan barcode ke server, setelah itu server akan mencari data kendaraan yang nomor mesinnya sama dengan hasil pembacaan barcode. Kemudian server akan mengirimkan kembali ke mikrokontroler hasil dari pencarian data tersebut. Jika data tidak ditemukan maka akan kembali ke pembacaan barcode.

Jika data ditemukan maka akan masuk validasi selanjutnya yaitu pembacaan sidik jari. Hasil pembacaan sidik jari dikirimkan ke server untuk dicek apakah cocok atau tidak dengan barcode sebelumnya di scan. Jika tidak, maka mikrokontroler akan kembali melakukan pembacaan sidik jari. Jika hasilnya cocok, maka mikrokontroler akan membuat data laporan kendaraan keluar parkir pada server.

Untuk perancangan aplikasi *mobile* dirancang menggunakan aplikasi *Blynk*. *Blynk* adalah *IOT (Internet Layanan Things)* yang dirancang untuk membuat *remote control* dan data sensor membaca dari perangkat *ESP8266* ataupun *Arduino* dengan sangat cepat dan mudah [21]–[23].



Gambar 8. Tampilan aplikasi *mobile*

Aplikasi pada *mobile* ini digunakan untuk memonitoring kondisi parkir, berisi informasi mengenai sisa kuota parkir dan jumlah kendaraan yang berada didalam area parkir, serta jika sisa kuota sudah habis, maka akan menampilkan *pop-up* notifikasi.

3.3 Pengkodean

Setelah desain dibuat, Hasil desain tersebut selanjutnya diterjemahkan menjadi baris-baris kode program yang dapat dimengerti oleh mesin. Bahasa pemrograman yang dipakai untuk memprogram mikrokontroler adalah Bahasa pemrograman *C++* dan ditulis menggunakan aplikasi *Arduino IDE*. Selain sebagai *text-editor* *Arduino IDE* juga berfungsi sebagai *compiler* dan memasangnya pada *board* mikorokontroler. Sedangkan untuk aplikasi web, menggunakan *framework* *Odo* dengan Bahasa pemrograman *Python*, *XML*, dan *Javascript* serta ditulis menggunakan aplikasi *Visual Studio Code* Sebagai teks editor.

3.4 Pengujian Program

Kode-kode program yang telah ditulis kemudian diuji terlebih dahulu sebelum diterapkan pada *hardware*. Pengujian program dilakukan untuk memastikan bahwa tidak ada *error* setelah program diterapkan pada *hardware*. Pengujian program yang dilakukan yaitu, seperti pengecekan hasil yang dikeluarkan sesuai atau tidak, terdapat *error* atau tidak, dan sejenisnya.

3.5 Penerapan dan Pemeliharaan Program

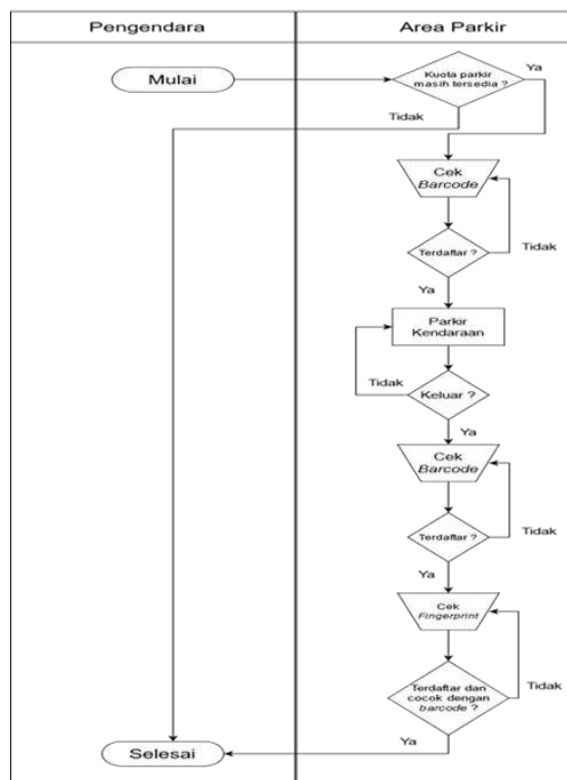
Program yang telah diuji selanjutnya diterapkan pada *hardware* yang telah dirancang. Kemudian dilakukan pemeriksaan ulang program pada saat diterapkan pada *hardware*. Pemeliharaan program juga diperlukan karena bisa saja ditemukan *error* pada kondisi-kondisi tertentu setelah program diterapkan pada *hardware*.

4. Hasil dan Pembahasan

Hasil perancangan dari sistem parkir yang dibuat adalah sebagai berikut.

3.1 Hasil Komparasi Sistem

Setelah sistem parkir yang dirancang diterapkan, proses parkir kendaraan di Universitas Singaperbangsa akan berubah. Seperti yang diperlihatkan oleh Gambar 9 berikut ini.



Gambar 9. *Flow of Diagram* Sistem parkir setelah diperbarui

Proses parkir dimulai dari pengendara melihat informasi kuota parkir, jika masih tersedia maka pengendara bisa parkir kendaraannya. Namun jika tidak tersedia maka harus menunggu sampai kuota parkir tersedia kembali atau tidak jadi parkir. Kemudian saat akan meninggalkan area parkir, pengendara harus memvalidasi data kendaraan (*barcode*) dan data pemilik kendaraan (*fingerprint*).

Proses pemindaian dilakukan sebanyak dua kali, yaitu pada saat masuk, pengendara harus memindai *barcode*

untuk membuka palang pintu. Kemudian pada saat pengendara akan keluar, pengendara harus memindai *barcode* dan *fingerprint*. Pemindaian *barcode* berfungsi untuk memverifikasi kendaraannya.

Hasil perbandingan proses parkir sebelum dan sesudah diperbarui dapat dilihat pada table dibawah.

Tabel 1. Perbandingan Proses Parkir Sebelum dan Seseudah Diperbarui

No	Keluar/ Masuk	Sebelum	Sesudah
1	Masuk	-	Memindai <i>barcode</i>
2	Keluar	-	Memindai <i>barcode</i> dan <i>fingerprint</i>

Berdasarkan Tabel 1 diatas, terlihat terdapat perbedaan proses keluar kendaraan pada area parkir yang signifikan. Sistem parkir baru yang dirancang memerlukan 2 tahapan sehingga membutuhkan waktu lebih lama. Namun kelemahan tersebut dapat diterima karena meningkatnya tingkat keamanan. Tingkat keamanan meningkat karena terjadi pemeriksaan kecocokan pengendara-kendaraan. Dalam hal ini ketika adanya upaya pencurian, pencuri harus memalsukan 2 identitas sekaligus (*barcode* dan *fingerprint*), serta harus ada kesesuaian antara 2 identitas tersebut.

Selain itu terdapat pula perbedaan dari sisi informasi pada sistem parkir sebelum dan sesudah diperbarui, dapat dilihat pada Tabel 2.

Tabel 2 Perbandingan Sistem Informasi pada Parkiran Sebelum dan Seseudah Diperbarui

No	Informasi	Sebelum	Sesudah
1	Jumlah Kendaraan di dalam parkiran	Tidak ada	Ada
2	Sisa Kuota parkiran	Tidak ada	Ada
3	Riwayat Keluar kendaraan	Tidak ada	Ada

Berdasarkan Tabel 2 sistem parkir yang baru dapat memberikan informasi kepada admin maupun satpam. Ketika Sisa kuota parkiran, satpam akan mengetahui hal tersebut dan dapat menutup pintu parkir sampai dengan sisa kuota parkir tersedia kembali. Sehingga informasi tersebut juga akan sampai kepada pengendara.

3.2 Hasil Validasi Keamanan

Validasi kemanan merupakan hal yang penting dari penelitian ini, karena tujuan dari perancangan sistem parkir ini adalah meningkatkan keamanan. Data hasil uji

coba validasi kemanan bisa dilihat pada tabel dibawah ini.

Tabel 3 Hasil Uji Coba Validasi berdasarkan Pemindaian Identitas

Percobaan	<i>Barcode</i>	<i>Fingerprint</i>		Pintu
		Terdaftar	Cocok	
1		Ya	Ya	Terbuka
2	Terdaftar	Ya	Tidak	Tidak Terbuka
3		Tidak	-	Tidak Terbuka
4	Tidak Terdaftar	-	-	Tidak Terbuka

Berdasarkan Tabel 3, Terlihat bahwa untuk membuka pintu parkir, langkah pertama yaitu harus memindai *barcode* yang terdaftar, karena pemindaian *fingerprint* hanya bisa dilakukan jika *barcode* yang dipindai terdaftar. setelah itu syarat kecocokan antara *barcode* dengan *fingerprint* harus terpenuhi. Walaupun *barcode* dan *fingerprint* yang dipindai terdaftar semua, jika pasangan *barcode-fingerprint* tersebut tidak cocok tetap tidak akan bisa membuka pintu parkir.

Tabel 4 Hasil Uji Coba Validasi berdasarkan Kondisi

No	Kondisi Awal	Percobaan	
		Masuk	Keluar
1	Tidak Parkir	Bisa	Tidak Bisa
2	Parkir	Tidak Bisa	Bisa

Berdasarkan Tabel 4, Menjelaskan validasi masuk atau keluar kendaraan berdasarkan kondisinya. Untuk mengeluarkan kendaraan dari parkiran, kendaraan harus dalam kondisi parkir, artinya identitas yang dipindai pada saat keluar harus identitas yang sama dengan identitas yang dipindai pada saat masuk parkiran sebelumnya. Begitu juga pada saat akan memasuki parkiran, kendaraan harus dalam kondisi tidak parkir, agar data laporan yang dihasilkan valid. Hal ini turut meningkatkan kemanan, karena semua data kendaraan yang masuk dan keluar tercatat dalam database beserta waktu masuk dan keluarnya.

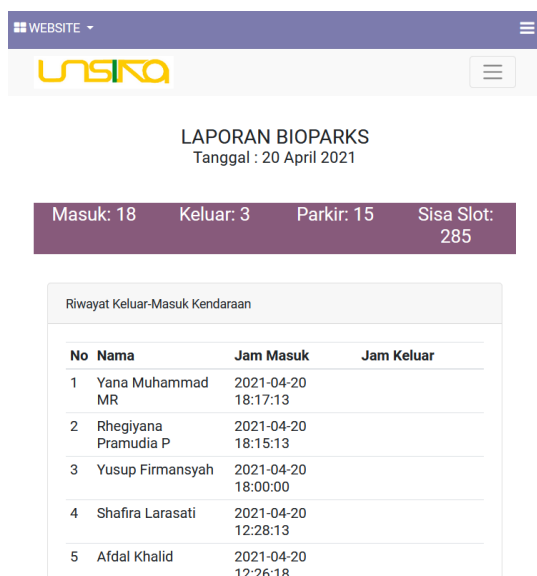
3.3 Hasil Design Aplikasi Web

Hasil dari Aplikasi yang dirancang adalah sebagai berikut



Gambar 10. Halaman Laporan (role Admin)

Gambar 10 merupakan halaman yang menampilkan semua data laporan keluar-masuk kendaraan, Menu-menu utamanya berada di *navbar*, diantaranya yaitu Master Data yang berisikan Data Kendaraan, Data Pemilik Kendaraan dan Data Sidik Jadi. Pada menu pengaturan merupakan halaman untuk mengatur kapasitas maksimum area parkir. Halaman ini beserta menu-menu nya hanya bisa diakses oleh admin (harus log in).



Gambar 11. Halaman Monitoring (Public)

Gambar 11 merupakan halaman monitoring data keluar-masuk kendaraan per hari beserta jumlah rekapitulasinya. Halaman ini bisa diakses publik (tanpa harus login) dengan cara mengakses url '192.168.1.8:8069/tim-bioparks/unsika' langsung.

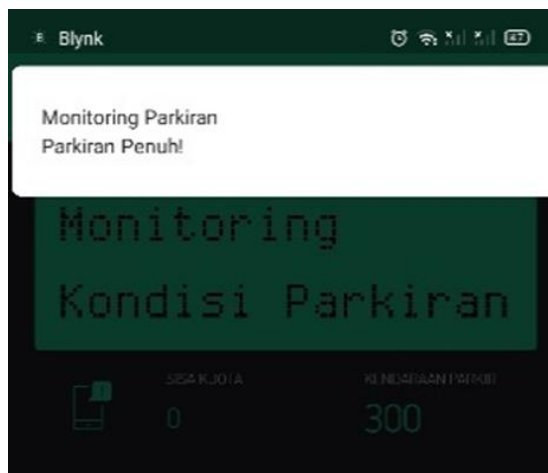
3.4 Hasil Design Aplikasi Mobile

Aplikasi *mobile* dari sistem parkir ini dibuat menggunakan aplikasi Blynk. Hasilnya adalah sebagai berikut.



Gambar 12. Monitoring via mobile

Pada aplikasi *mobile* menampilkan Informasi jumlah kendaraan parkir dan sisa slot parkir yang sama dengan data rekapitulasi pada halaman monitoring di *website*. Fitur lainnya dari aplikasi *mobile* ini adalah akan mengirimkan notifikasi jika kuota parkir sudah penuh, tampilannya dapat dilihat pada gambar 13.



Gambar 13. Notifikasi pada aplikasi mobile

Pada gambar 13, diperlihatkan bahwa terdapat notifikasi parkir sudah penuh. Oleh karena itu, pada saat kondisi parkir sudah penuh, satpam atau petugas bisa menutup pintu masuk parkir agar tidak terjadi *overload*.

5. Kesimpulan

Kesimpulan dari penelitian ini yaitu:

1. Pemanfaatan teknologi *barcode* dan *sensor fingerprint* menambah tingkat keamanan, sehingga resiko adanya tindak pencurian dapat

- ditekan dan membuat pengguna parkir merasa tenang menyimpan kendaraannya. [7]
2. Aplikasi yang dibangun menggunakan ERP Odoo memudahkan pengelola dalam mengatur data-data parkir.
 3. Aplikasi berbasis *web* sangat efisien untuk diterapkan pada sistem parkir karena sifatnya yang fleksibel dapat diakses dimanapun selama memiliki *web browser*. [8]
 4. Penerapan konsep IoT pada sistem ini memudahkan dalam memonitoring kondisi parkir

Ucapan Terima Kasih

Penulis mengucapkan terimakasih kepada bapak Rahmat Hidayat dan Ibu Ulinnuha Latifa yang telah membimbing saya dalam penulisan artikel ini, kepada Rhegiyana Pramudia Pahrudin dan Siti Robiah Adawiah selaku rekan seperjuangan dalam menyelesaikan alat uji ini. Tidak lupa juga penulis mengucapkan terima kasih kepada *reviewer* dan *proofreader* yang sudah meluangkan waktunya untuk menelaah artikel ini [9]

6. Daftar Rujukan

- [1] A. P. Wahyu and Suhendri, "Peningkatan Sistem Keamanan Parkir dengan Teknologi Artificial Intelligence Imaging," *JOINT*, vol. 01, no. 02, pp. 39–42, 2019.
- [2] E. Maiyana, S. Zakir, Isnardi, D. Kurnia, and N. Fardiantia, "Akuntabilitas Pengelolaan Dana Donatur Panti Asuhan Dengan Sistem Informasi Berbasis Web," *J. Sains dan Inform.*, vol. 6, no. 2, p. 99, 2020.
- [3] R. Oktaviani and V. Septiana Windyasari, "Aplikasi Sistem Parkir Kendaraan Bermotor Menggunakan Teknologi Radio Frequency Identification (RFID) Di Universitas Islam Syekh Yusuf Tangerang," *JIMTEK*, vol. 1, no. 2, p. 96, 2020.
- [4] A. B. Pradana, C. Ma'rifadiyah, D. Jatinugroho, and F. Z. Abidin, "Perancangan Sistem Perparkiran Rendah Biaya Berbasis Ponsel Cerdas Android," *JTE*, vol. 11, no. 1, pp. 31–35, 2019.
- [5] M. R. Utomo, "Perancangan Dan Implementasi Sistem Keamanan Parkir Dengan Fingerprint Pada Parkiran Telkom University," vol. 6, no. 1, pp. 3–8, 2019.
- [6] A. R. Yudiantika, E. S. Pasinggi, I. P. Sari, and B. S. Hantono, "Implementasi Augmented Reality Di Museum: Studi Awal Perancangan Aplikasi Edukasi Untuk Pengunjung Museum," *Paradig. Komput. dan Inform.*, 2017.

C. Hardjono, "Perancangan Dan Implementasi Erp (Enterprise Resource Planning) Modul Sales And Warehouse Management Pada Cv. Brada," *eProceedings Eng.*, vol. 4, no. 3, pp. 4983–4993, 2017.

A. W. Pradhana, S. F. S. Gumilang, and R. W. Witjaksono, "Pengembangan Sistem Informasi Berbasis Enterprise Resource Planning Modul Warehouse Management pada Odoo dengan Metode Rapid Application Development di PT. Brodo Ganesha Indonesia," *eProceedings Eng.*, vol. 3, no. 2, pp. 3337–3345, 2016.

S.- Suminten, S. Amelia, and I. D. Sintawati, "Penerapan Enterprise Resource Planning Penjualan Aksesoris Berbasis Odoo," *JSil (Jurnal Sist. Informasi)*, vol. 6, no. 1, p. 69, 2019.

I. D. A. Putri, R. Saedudin, and W. Witjaksono, "Pengembangan Sistem Erp Purchase Menggunakan Odoo Pada Pt.Pdu Dengan Metoda Asap," *e-Proceeding Eng.*, vol. 3, no. 2, pp. 3236–3247, 2016.

G. D. Girang, R. Andreswari, and U. Y. Kurnia Septo Hedyanto, "Perancangan Sistem Enterprise Resource Planning Modul Payroll Berbasis Odoo 8 Dengan Metode Rapid Application Development Pada PT. XYZ," *J. Rekayasa Sist. Ind.*, vol. 4, no. 02, pp. 130–137, 2017.

C. A. Lestari and Suhendri, "Implementasi Odoo Dengan Modul Accounting and Finance Di SD Islam Tunas Mandiri," *J. Inform. Terpadu*, vol. 3 No.1, pp. 1–6, 2017.

N. A. Fa', R. I. Salmawati, and R. I. Prasasti, "Aplikasi Layanan Pengamanan Parkir Berbasis Identify Card Di Kampus Uns," 2019.

O. R. Arsyad, Kurnia, and P. Kartika, "Rancang Bangun Alat Pengaman Brankas Menggunakan Sensor Sidik Jari Berbasis Arduino," *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 5, no. 1, pp. 1–6, 2021.

A. Iskandar, Muhajirin, and Lisah, "Sistem Keamanan Pintu Berbasis Arduino Mega," *J. Inform. Upgris*, vol. 3, no. 2, 2017.

B. F. Billyan, A. Bhawiyuga, and R. Primananda, "Implementasi Metode Klasifikasi Fuzzy K-Nearest Neighbor (FK-NN) Untuk Fingerprint Access Point Pada Indoor Positioning," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 1, no. 11, 2017.

- [17] M. Destiningrum and Q. J. Adrian, "Sistem Informasi Penjadwalan Dokter Berbasis Web Dengan Menggunakan Framework Codeigniter (Studi Kasus: Rumah Sakit Yukum Medical Centre)," *J. Teknoinfo*, vol. 11, no. 2, p. 33, 2017.
- [18] M. A. Rosid, "Implementasi JSON untuk Minimasi Penggunaan Jumlah Kolom Suatu Tabel Pada Database PostgreSQL," *JOINCS (Journal Informatics, Network, Comput. Sci.*, vol. 1, no. 1, p. 33, 2017.
- [19] A. Tedyyana and R. Kurniati, "Membuat Web Server Menggunakan Dinamic Domain," *J. Teknol. Inf. Komun. Digit. Zo.*, vol. 7, no. 1, pp. 1–10, 2016.
- [20] R. A. Purnama and A. T. Laksono Putra, "Aplikasi Web Server Berbasis Bahasa C Sharp," *J. Tek. Komput.*, 2018.
- [21] S. P. Tamba, A. H. M. Nasution, S. Indriani, N. Fadhilah, and C. Arifin, "Pengontrolan Lampu Jarak Jauh Dengan Nodemcu Menggunakan Blynk," *J. Tek. Inf. dan Komput.*, vol. 2, no. 1, pp. 93–98, 2019.
- [22] Handi, H. Fitriyah, and G. E. Setyawan, "Sistem Pemantauan Menggunakan Blynk dan Pengendalian Penyiraman Tanaman Jamur Dengan Metode Logika Fuzzy," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 4, pp. 3258–3265, 2019.
- [23] Arafat, "SISTEM PENGAMANAN PINTU RUMAH BERBASIS Internet Of Things (Iot) Dengan ESP8266," *Technologia*, vol. 7, 2016.