

Sistem Manajemen Parkir Mobil *On Street* dan *Off Street* Berbasis *Location Based Service* (LBS) dan *Google Maps API*

Cornelius Venti^{#1}, Yus Sholva^{#2}, Rudy Dwi Nyoto^{#3}

[#]Program Studi Informatika Fakultas Teknik Universitas Tanjungpura

Jl. Profesor Dokter H. Hadari Nawawi, Bansir Laut, Pontianak Tenggara, Kota Pontianak, Kalimantan Barat 78115

¹corneliusventi@gmail.com

²sholvariza@untan.ac.id

³rudydn@informatika.untan.ac.id

Abstrak— Bertambahnya jumlah kendaraan sering tidak diikuti dengan bertambahnya tempat parkir. Kondisi sangat menyulitkan pengendara untuk memarkir kendaraannya karena penuh. Sehingga informasi ketersediaan lahan parkir menjadi sangat penting khususnya parkir mobil karena memerlukan tempat parkir. Pengendara tidak mengetahui ketersediaan tempat parkir yang berada di sekitar destinasi dan membuat pengendara harus mencarinya setelah sampai destinasi. Dengan ini pengendara memerlukan suatu informasi tentang ketersediaan tempat parkir yang tersedia di sekitar destinasi sebelum sampai di destinasi. Maka sistem manajemen parkir sangat diperlukan untuk menyediakan informasi tentang tempat parkir yang tersedia sekitar destinasi. Salah satunya cara yang dapat dilakukan adalah dengan menggunakan prinsip *Location Based Service (LBS)* untuk menyediakan informasi tempat parkir sekitar lokasi destinasi dengan menampilkan posisi *latitude* and *longitude* dari posisi tempat parkir dalam bentuk peta digital menggunakan *Google Maps API*. Dan setiap ruang parkir pada tempat parkir diidentifikasi menggunakan *QR-Code* yang berisi kode unik setiap ruang parkir. Kemudian menggunakan *Google Maps API* untuk menyaring tempat parkir yang terdapat dalam radius terdekat dengan tempat destinasi dengan cara hanya memperlihatkan tempat parkir di dalam radius saja dan yang masih tersedia ruang parkirnya. Hasil pengujian memperlihatkan bahwa sistem manajemen parkir mobil *on street* dan *off street* berbasis *location based service* dan *google maps api* dapat bekerja dengan baik berdasarkan hasil akhir pengujian *User Acceptance Test* sebesar 93%.

Kata kunci— Parkir *On Street*, Parkir *Off Street*, *Latitude*, *Longitude*, *Location Based Service*, *Google Maps API*, *QR-Code*

I. PENDAHULUAN

Parkir adalah keadaan tidak bergerak suatu kendaraan yang bersifat sementara karena ditinggalkan oleh pengemudinya. Secara hukum dilarang untuk parkir di tengah jalan raya, namun parkir di sisi jalan umumnya

diperbolehkan. Fasilitas parkir dibangun bersama-sama dengan kebanyakan gedung, untuk memfasilitasi kendaraan pemakai gedung. Termasuk dalam pengertian parkir adalah setiap kendaraan yang berhenti pada tempat-tempat tertentu baik yang dinyatakan dengan rambu lalu lintas ataupun tidak, serta tidak semata-mata untuk kepentingan menaikkan dan/atau menurunkan orang dan/atau barang [1].

Parkir memerlukan suatu lahan / tempat parkir yang dapat menampung kendaraan yang ingin diparkirkan seperti tempat parkir di bahu jalan maupun di gedung parkir. Maka setiap memarkirkan kendaraan, pengendara harus mencari terlebih dahulu di suatu tempat parkir yang masih tersedia ruang untuk dapat memarkirkan kendaraan pengendara.

Parkir merupakan hal penting ketika berpergian kemana-mana, terutama tempat parkir mobil dengan semakin banyaknya pengendara mobil pribadi saat ini, seringkali menyebabkan padatnya tempat parkir mobil dan menyebabkan keterbatasan tempat parkir. Dan ini pengendara diharuskan mencari tempat parkir yang tersedia. Jika terjadi kepadatan tempat parkir di tepi jalan maka pengendara harus mencari secara perlahan, yang dapat menyebabkan kemacetan jalan. Sedangkan jika terjadi kepadatan tempat parkir di dalam bangunan maka pengendara harus menyusuri setiap lantai parkir, dan ini dapat memakan waktu yang lama.

Hal itu disebabkan pengendara tidak mengetahui ketersediaan tempat parkir yang berada di sekitar destinasi dan membuat pengendara harus mencarinya setelah sampai destinasi. Dengan ini pengendara memerlukan suatu informasi tentang ketersediaan tempat parkir yang tersedia di sekitar destinasi. Supaya pengendara dapat mengetahui tempat parkir yang dapat memarkirkan kendaraan pengendara tanpa harus mencarinya pada saat sudah di suatu tempat parkir.

Tetapi adapun kendala yang dari juru parkir dan pengelola parkir untuk dapat menyediakan informasi ketersediaan tempat parkir bagi setiap pengendara yaitu tidak ada media untuk menginformasikan ketersediaan tempat parkir dan tidak ada penanda akan ketersediaan ruang parkir yang bisa digunakan. Maka sistem manajemen parkir sangat diperlukan untuk menyediakan informasi tentang tempat parkir yang tersedia sekitar destinasi. Hal itu dapat dicapai salah satunya dengan menggunakan prinsip *Location Based Service* (LBS) untuk menyediakan informasi tempat parkir sekitar lokasi destinasi [2].

Dengan menerapkan prinsip LBS informasi tempat parkir di sekitar lokasi destinasi dengan menampilkannya dalam bentuk peta digital menggunakan *Google Maps API* [3]. Dengan menampilkan tempat parkir yang berada dalam radius tertentu sekitar lokasi destinasi. Pengemudi dapat memilih dan memesan salah satu tempat parkir yang dekat dengan lokasi destinasi.

Dengan *Google Maps API*, sistem dapat menampilkan posisi *latitude* and *longitude* dari posisi tempat parkir dalam bentuk peta digital [4]. Peta digital tersebut dapat membantu dalam visualisasi tempat-tempat parkir yang berada disekitar lokasi destinasi. *Google Maps API* juga dapat memfasilitasi sistem untuk menyaring tempat parkir yang terdapat dalam radius terdekat dengan tempat destinasi dengan cara hanya memperlihatkan tempat-tempat parkir di dalam radius saja dan yang masih tersedia tempat parkirnya.

Dengan *latitude* and *longitude* dari posisi tempat parkir yang tercatat dalam sistem maka pencarian tempat parkir dalam radius tertentu dari lokasi destinasi dapat tercapai. Akan tetapi dengan menggunakan *latitude* dan *longitude* mempunyai keterbatasan yaitu harus berada di lokasi yang terbuka saja. Sedangkan pada kondisi tempat parkir yang berada di dalam gedung penerapan LBS, posisi *latitude* dan *longitude* dari tempat parkir dapat mengalami masalah yaitu tempat parkir yang bertingkat tidak dapat mewakili setiap tempat parkir di gedung bertingkat. Salah satu solusi yang mungkin diterapkan adalah *QR Code*. Selama ini *QR-Code* dipakai untuk identifikasi, metode pembayaran dan lain-lain [5]. *QR-Code* juga dapat diterapkan pada tempat parkir sebagai penanda unik setiap tempat parkir di lokasinya maupun dalam sistem.

Dengan lokasi tempat parkir sudah dapat diidentifikasi menggunakan *latitude*, *longitude* dan *QR-code*, maka penunjukan arah (*direction*) menuju tempat parkir pun dapat dilakukan. Penunjukan arah tersebut dapat diterapkan menggunakan *Google Maps Direction API* dan *GPS* dari mobile device setiap pengendara. *Google Maps Direction API* menampilkan arah menuju tempat parkir pada peta digital dan *GPS* dari mobile device setiap pengendara dapat menampilkan lokasi keberadaan setiap pengendara.

Untuk menjamin ketersediaan tempat parkir maka perlu diterapkan sistem pemesanan (*Booking*). Sistem pemesanan ini memungkinkan pengendara untuk memesan tempat parkir sebelum sampai di lokasi tempat

parkir, ini dapat memudahkan pengendara mendapatkan tempat parkir di sekitar destinasi. Biaya pemesanan tempat parkir dapat menggunakan sistem saldo. Biaya pemesanan ini digunakan untuk memastikan pengendara akan benar-benar memakai tempat parkir tersebut.

Sistem saldo selain untuk pembayaran biaya pemesanan tempat parkir, sistem saldo ini juga dapat digunakan untuk pembayaran biaya parkir. Sistem saldo dibangun juga akan memberi kemudahan pengendara dalam pembayaran biaya parkir tanpa menggunakan uang tunai, ini dapat dicapai dengan cara melakukan pengisian (*top up*) saldo sebelum melakukan pemesanan tempat parkir. Pengisian saldo dapat dilakukan dengan cara transfer bank.

Dengan demikian, jika pengendara dapat mengetahui tentang ketersediaan tempat parkir mobil sekitar lokasi destinasi, memesan tempat parkir dan membayar biaya parkir secara online. Maka pengendara mobil dapat leluasa bepergian tanpa harus khawatir mencari tempat parkir lagi. Oleh karena itu, penulis bermaksud mengembangkan Sistem Manajemen Parkir Mobil *On Street* dan *Off Street* Berbasis *Location Based Service* (LBS) dan *Google Maps API*.

II. LANDASAN TEORI

A. Pengertian Parkir

Menurut PP No. 43 tahun 1993 parkir didefinisikan sebagai kendaraan yang berhenti pada tempat-tempat tertentu baik yang dinyatakan dengan rambu atau tidak, serta tidak semata-mata untuk kepentingan menaikkan atau menurunkan orang dan atau barang. Sedangkan definisi lain tentang parkir adalah keadaan dimana suatu kendaraan berhenti untuk sementara (menurunkan muatan) atau berhenti cukup lama. Sehingga tempat parkir ini harus ada pada saat akhir atau tujuan perjalanan sudah dicapai [6].

Parkir adalah tempat pemberhentian kendaraan dalam jangka waktu pendek atau lama, sesuai dengan kebutuhan pengendara. Parkir merupakan salah satu unsur prasarana transportasi yang tidak terpisahkan dari sistem jaringan transportasi, sehingga pengaturan parkir akan mempengaruhi kinerja suatu jaringan, terutama jaringan jalan raya [7].

Menurut penempatannya, jenis parkir dapat terbagi menjadi dua jenis antara lain [7]:

1. Parkir di tepi jalan (*on-street parking*). Yakni parkir dengan menggunakan badan jalan sebagai tempat parkir.
2. Parkir di luar badan jalan (*off-street parking*). Yakni parkir kendaraan di luar badan jalan bisa di halaman gedung perkantoran, supermarket, atau pada taman parkir.

B. Location Based Service

Location Based Service (LBS) merupakan suatu layanan yang bereaksi aktif terhadap perubahan entitas posisi sehingga mampu mendeteksi letak objek dan

memberikan layanan sesuai dengan letak objek yang telah diketahui tersebut. *Location Base Services* adalah aplikasi yang bergantung pada lokasi tertentu dan didefinisikan pula sebagai layanan informasi dengan memanfaatkan teknologi untuk mengetahui posisi sesuatu. Layanan berbasis lokasi menggunakan teknologi Positioning System, teknologi ini memungkinkan para pengguna dapat memperoleh informasi lokasi sesuai dengan kebutuhannya [2]. Dengan menerapkan LBS pada sistem ini, sistem dapat menyediakan informasi tempat parkir yang tersedia di sekitar lokasi destinasi yang dituju.

Untuk menggambarkan cara kerja LBS, anggaplah aplikasi LBS akan mencari informasi mengenai lokasi restoran yang berada di sekitar posisi pengguna [8].

1. Anggaplah sekarang fungsi pencarian telah diaktifkan, posisi pengguna sebenarnya dari perangkat mobile diperoleh dari *Positioning Service*. Hal ini dapat dilakukan baik oleh perangkat menggunakan GPS sendiri atau layanan posisi jaringan yang berasal dari *provider (Cell Tower)*. Setelah itu perangkat *mobile* pengguna mengirimkan permintaan informasi, yang berisi tujuan untuk mencari dan mengirimkan posisi melalui jaringan komunikasi ke *gateway* telekomunikasi.
2. *Gateway* memiliki tugas untuk bertukar pesan di antara jaringan komunikasi selular dan internet. Oleh karena itu dia mengetahui alamat *web* dari beberapa aplikasi *server* dan rute permintaan ke spesifik *server* tertentu. *Gateway* akan menyimpan juga informasi tentang perangkat *mobile* yang telah meminta informasi
3. Aplikasi *server* membaca permintaan dan mengaktifkan layanan yang terkait.
4. Kemudian, *service* menganalisis lagi pesan dan memutuskan mana informasi tambahan selain criteria pencarian (restoran + padang) dan posisi pengguna diperlukan untuk menjawab permintaan pengguna. Dalam kasus ini *service* akan menemukan bahwa pengguna membutuhkan informasi tentang restoran dari *database yellow pages* pada wilayah tertentu dan kemudian *service* tersebut akan meminta penyedia data untuk memberikan data tersebut
5. Selanjutnya *service* akan menemukan bahwa informasi tentang jalan, jarak dan cara yang diperlukan untuk memeriksa apakah restoran dapat dicapai
6. Setelah sekarang semua informasi *service* akan melakukan *buffer* spasial dan *query routing* untuk mendapatkan beberapa restoran terdekat. Setelah menghitung daftar restoran terdekat, hasil dikirim kembali ke pengguna melalui *internet*, *gateway* dan jaringan *mobile*.
7. Kemudian, informasi mengenai restoran akan disampaikan kepada pengguna baik dalam bentuk peta digital.

C. Google Maps API

Google Maps Application Programming Interface (API) merupakan suatu fitur aplikasi yang dikeluarkan oleh google untuk memfasilitasi pengguna yang ingin mengintegrasikan Google Maps ke dalam website masing-masing dengan menampilkan data point milik sendiri. Dengan menggunakan *Google Maps API*, *Google Maps* dapat di-embed pada website eksternal. Agar aplikasi *Google Maps* dapat muncul di website tertentu, diperlukan adanya API key. API key merupakan kode unik yang digenerasikan oleh google untuk suatu website tertentu, agar server *Google Maps* dapat mengenali, namun untuk Google Maps API versi 3 sudah tidak membutuhkan API key, sehingga mempermudah untuk menggunakan *Google Maps API*, dan pada versi ini juga terdapat beberapa perbedaan sintaks dari kode yang digunakan [3]. Dengan Google Maps API, sistem dapat menampilkan posisi latitude and longitude dari posisi tempat parkir dalam bentuk peta digital. Peta digital tersebut dapat membantu dalam visualisasi tempat-tempat parkir yang berada disekitar lokasi destinasi.

D. Latitude dan Longitude

Latitude adalah garis yang melintang di antara kutub utara dan kutub selatan, yang menghubungkan antara sisi timur dan barat bagian bumi. Garis ini memiliki posisi membentangi bumi, sama halnya seperti garis equator (khatulistiwa), tetapi dengan kondisi nilai tertentu. Garis lintang inilah yang dijadikan ukuran dalam mengukur sisi utara-selatan koordinat suatu titik di belahan bumi. Sedangkan *longitude* adalah garis membujur yang menghubungkan antara sisi utara dan sisi selatan bumi (kutub). Garis bujur ini digunakan untuk mengukur sisi barat-timur koordinat suatu titik di belahan bumi. Sama seperti equator pada latitude yang berada ditengah dan memiliki nilai 0 (nol) derajat, pada longitude, garis tengah yang bernilai 0 (nol) derajat disebut garis prime meridian (garis bujur). Sedangkan garis yang berada paling kiri memiliki nilai -90 derajat, dan yang paling kanan memiliki nilai 90 derajat [4]. Dengan *Latitude* dan *Longitude*, dapat diterapkan pada sistem ini sebagai format penyimpanan letak keberadaan tempat parkir dalam sistem.

E. Quick Response Code

Quick Response Code (QR-Code) adalah jenis barcode yang berbentuk dua dimensi yang dikembangkan oleh Denso Wave, sebuah divisi Denso Corporation, sebuah perusahaan di Jepang, yang dipublikasikan pada tahun 1994. QR merupakan singkatan dari *Quick Response* (respon / tanggapan cepat), sehingga fungsi atau tujuan utama dari teknologi ini adalah penyampaian informasi dengan cepat dan mendapat tanggapan atau respons yang cepat pula. Oleh karena itu *QR-code* dapat dengan mudah dibaca oleh pemindai. Berbeda dengan barcode biasa yang berbentuk satu dimensi dan menyimpan informasi secara horizontal, *QR-code* mampu menyimpan informasi secara horizontal dan vertikal. *QR-code* juga mampu menyimpan teks alfanumerik, kanji, kana, hiragana, simbol, biner, url,

dan control code. Adapun contoh dari *QR Code* yang dituangkan dalam Gambar 1. yang berisi <https://google.com> [5]. Dengan *QR-Code* dapat diterapkan pada sistem ini sebagai penanda unik setiap tempat parkir di lokasinya maupun dalam sistem.



Gambar. 1 Contoh QR code

F. Unified Modeling Language

Unified Model Language (UML) adalah sebuah ‘bahasa’ yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak [9]. UML (*Unified Modeling Language*) adalah standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berbasis objek [10]. Berdasarkan definisi tersebut dapat disimpulkan bahwa UML adalah standar bahasa komunikasi yang digunakan industri untuk membuat perancangan, analisis, desain, dan dokumentasi dalam pembuatan sistem piranti lunak. Dengan UML, perancangan sistem ini dapat lebih terstruktur.

G. Black Box Testing

Black box testing adalah tipe testing yang memerlukan perangkat lunak yang tidak diketahui kinerja internalnya. Sehingga para *tester* memandang perangkat lunak seperti layaknya sebuah “kotak hitam” yang tidak penting dilihat isinya, tapi dikenal proses testing di bagian luar. Jenis testing ini hanya memandang perangkat lunak dari sisi spesifikasi dan kebutuhan yang telah didefinisikan pada saat awal perancangan. Sebagai contoh, jika terdapat sebuah perangkat lunak yang merupakan sebuah sistem informasi inventory sebuah perusahaan. Maka pada jenis *white box testing*, perangkat lunak tersebut akan berusaha dibongkar listing programnya untuk kemudian di tes menggunakan teknik-teknik yang telah dijelaskan sebelumnya. Sedangkan pada jenis *black box testing*, perangkat lunak tersebut akan dieksekusi kemudian berusaha dites apakah telah memenuhi kebutuhan pengguna yang didefinisikan pada awal tanpa harus membongkar listing programnya [11]. Dengan *Black Box Testing*, sistem ini dapat diuji apakah fitur-fitur atau kebutuhan-kebutuhan fungsional telah terpenuhi dan berjalan dengan lancar.

H. User Acceptance Testing

User Acceptance Testing merupakan pengujian yang dilakukan oleh *end-user* dimana user tersebut adalah

staf/karyawan perusahaan yang langsung berinteraksi dengan sistem dan dilakukan verifikasi apakah fungsi yang ada telah berjalan sesuai dengan kebutuhan/fungsinya. *User Acceptance Testing* adalah metode *testing* dimana pengguna akhir melakukan percobaan untuk memvalidasi produk berdasarkan kebutuhan yang telah ditentukan. Pengujian ini bertujuan untuk membantu menemukan bug yang akan muncul pada saat penggunaan [12]. Dengan *User Acceptance Testing*, sistem dapat diuji oleh *end-user* (pengguna), agar mendapat penilaian terhadap sistem ini.

I. Hypertext Markup Language (HTML)

Hypertext Markup Language (HTML) merupakan bahasa struktur untuk membuat sebuah teks menjadi link yang dapat berpindah dari satu halaman ke halaman lainnya hanya dengan meng-klik teks tersebut dan menggunakan tanda (dalam bahasa inggris disebut ‘*mark*’) untuk menandai bagian - bagian dari teks agar teks tersebut memiliki tampilan dan fungsi tertentu. Sebagai contoh, teks yang berada di antara tanda (*mark*) tertentu akan menjadi tebal, miring, dan berwarna merah. Dan, jika berada diantara tanda lainnya, maka membuat teks tersebut menjadi sebuah link yang dapat di-klik. Dalam praktiknya nanti, tanda atau *mark* tersebut disebut dengan istilah *Tag* [13]. HTML dapat digunakan sebagai bahasa markup untuk mengstruktur tampilan dalam mengembangkan sistem.

J. Cascading Style Sheet

Cascading Style Sheet (CSS) merupakan kumpulan kode untuk mendesain atau mempercantik tampilan halaman. Dengan arti lain, dengan memanfaatkan CSS kita bisa mengubah desain standar yang dihasilkan oleh HTML menjadi variasi-variasi yang lebih kompleks. Dengan CSS-lah, sebuah desain website yang dibangun menggunakan HTML akan menjadi lebih menarik dan variatif [13]. CSS dapat digunakan untuk mempercantik tampilan dalam sistem.

K. JavaScript

JavaScript adalah script default di HTML. Meskipun namanya berbau Java, Javascript dan Java adalah dua bahasa pemrograman yang berbeda. Bersama HTML dan CSS, Javascript menjadi tiga teknologi paling utama dalam pembuatan konten website. Javascript sudah didukung oleh berbagai browser modern, sehingga tidak membutuhkan plug-ins apapun. Melalui Javascript Anda bisa mengubah konten halaman web, mengubah attribute tag HTML, mengubah aturan style di CSS, atau memvalidasi form yang telah diinput pengunjung sebelum mengirimkannya ke web server. Anda juga sering memasang kode-kode Javascript saat ingin menampilkan iklan atau widget pada halaman web Anda nantinya [14]. *JavaScript* dapat digunakan untuk membuat tampilan lebih interaktif dalam sistem.

L. PHP Hypertext Preprocessor

PHP merupakan bahasa pemrograman yang digunakan untuk membuat aplikasi berbasis website. Sebagai sebuah aplikasi, website tersebut hendaknya memiliki sifat dinamis dan interaktif. memiliki sifat dinamis artinya, website tersebut bisa berubah tampilan kontennya sesuai kondisi tertentu (misalnya, menampilkan produk yang berbeda-beda untuk setiap pengunjung). Interaktif artinya, website tersebut dapat memberi feedback bagi user (misalnya, menampilkan hasil pencarian produk) [15]. Dengan PHP dapat digunakan untuk membuat tampilan yang dinamis dalam sistem.

M. MySQL

MySQL adalah RDBMS yang cepat dan mudah digunakan, serta sudah banyak dipakai untuk berbagai kebutuhan. MySQL dikembangkan oleh MySQL AB Swedia. Hampir sebagian besar aplikasi website yang ada di internet dikembangkan menggunakan MySQL dan bahasa pemrograman lainnya, seperti PHP [16]. MySQL dapat digunakan untuk menyimpan data-data dengan cepat dan mudah dalam sistem.

N. Laravel Framework

Laravel adalah Framework PHP yang dirilis dibawah lisensi MIT dan dibangun dengan konsep MVC (*Model-View-Controller*). Laravel adalah pengembangan MVC yang ditulis dalam PHP yang dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan, serta untuk meningkatkan pengalaman bekerja dengan aplikasi dengan menyediakan sintaks yang ekspresif, jelas, dan menghemat waktu [17]. Laravel dapat digunakan untuk mempersingkat dan mempermudah pengembangan sistem.

III. ANALISIS DAN PERANCANGAN SISTEM

A. Analisis Proses

Adapun prinsip yang diimplementasikan di dalam sistem manajemen parkir yaitu *Location Based Service* (LBS). LBS merupakan suatu layanan yang bereaksi aktif terhadap perubahan entitas posisi sehingga mampu mendeteksi letak objek dan memberikan layanan sesuai dengan letak objek yang telah diketahui tersebut. Dengan penerapan LBS dalam sistem manajemen parkir mobil, sistem dapat menyajikan tempat-tempat parkir yang tersedia di sekitar destinasi tujuan. Penyajian tempat-tempat parkir tersebut di dalam peta digital dengan memanfaatkan *Google Maps API*. Pencarian tempat parkir di sekitar destinasi bisa dilakukan dengan adanya penyimpanan lokasi GPS setiap tempat parkir, yang dibandingkan dengan letak GPS *smartphone* setiap pengendara mobil. Pengenalan ketersediaan ruang parkir setiap tempat parkir dapat diterapkan dengan melakukan pencatatan setiap pemesanan dan pemakaian setiap ruang parkir serta dibantu dengan terdapatnya kode unik dan *qr-code* di setiap ruang parkir. Kode unik dan *QR-Code* ini

yang merepresentasikan setiap ruang parkir di dalam sistem dan juga penandaan setiap ruang parkir di lapangan.

B. Analisis Sistem Berjalan

Sebagian besar sistem-sistem manajemen parkir mobil yang sudah berjalan saat ini untuk *On Street* (bahu jalan) dan *Off Street* (gedung parkir) pada umum berupa manajemen parkir mobil di bahu jalan yang dikelola oleh juru parkir / tukang parkir dan manajemen parkir mobil di gedung parkir yang memakai karcis parkir dilengkapi dengan penampilan jumlah parkir yang masih tersedia pada layar monitor di luar gedung parkir.

C. Sistem yang Akan Dibangun

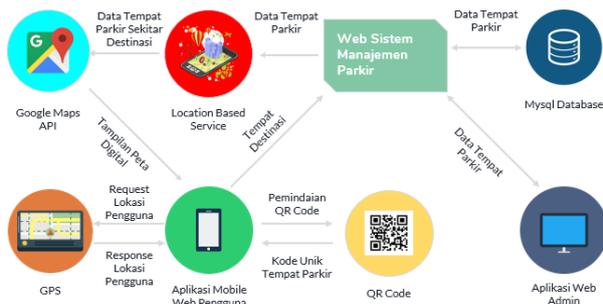
Berdasarkan analisis sistem yang sedang berjalan maka diperlukan sistem manajemen parkir yang meliputi pencarian, pemesanan dan penggunaan tempat parkir yang tersedia, serta pengolahan data tempat parkir, yang mana proses secara umum yang dirancang lebih kurangnya seperti berikut ini:

1. Pengguna
 - a. Melakukan registration pada sistem.
 - b. Melakukan login ke dalam sistem.
 - c. Mengisi ulang saldo melalui transfer bank.
 - d. Mendaftarkan kendaraan mobil pengguna.
 - e. Melihat tampilan peta dalam bentuk peta digital
 - f. Mencari tempat parkir di sekitar destinasi dengan mencari lokasi destinasinya atau memilihnya dalam peta digital.
 - g. Mendapat tempat-tempat parkir yang masih memiliki ruang parkir yang tersedia di sekitar destinasi pengguna dalam radius kurang lebih 200 meter.
 - h. Memilih dan memesan tempat parkir parkir dan ruang parkir yang sesuai dengan keinginan untuk durasi tertentu.
 - i. Mendapatkan petunjuk arah menuju tempat parkir yang telah dipilih.
 - j. Memindai QR-Code yang terdapat di ruang parkir yang telah pesan sesampainya di tempat parkir tersebut.
 - k. Memindai QR-Code yang terdapat di ruang parkir yang telah pesan sebelum pergi dari tempat parkir tersebut.
 - l. Mengubah profil dan kata sandi pengguna.
 - m. Melakukan logout.
2. Administrator
 - a. Melakukan *login* ke dalam sistem.
 - b. Melihat daftar pengguna.
 - c. Mendaftarkan operator lapangan / juru parkir.
 - d. Melihat daftar mobil.
 - e. Melihat daftar tempat parkir dan ruang parkir.
 - f. Mendaftarkan tempat parkir.
 - g. Mengubah profil dan kata sandi administrator.
 - h. Melakukan *logout*.
3. Admin Operator
 - a. Melakukan *login* ke dalam sistem.
 - b. Mendaftarkan daftar ruang parkir.

- c. Melihat daftar pemakaian parkir dan pemesanan parkir.
- d. Mengubah profil dan kata sandi operator.
- e. Melakukan *logout*.
- 4. Operator
 - a. Melakukan *login* ke dalam sistem.
 - b. Melihat daftar pemakaian parkir dan pemesanan parkir.
 - c. Mengubah profil dan kata sandi operator.
 - d. Melakukan *logout*.

D. Arsitektur Sistem

Adapun arsitektur sistem manajemen parkir yang dapat dilihat pada Gambar 2.



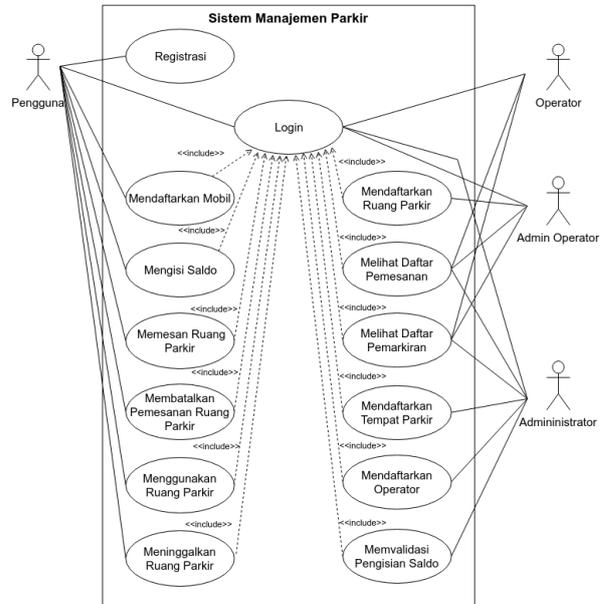
Gambar. 2 Arsitektur sistem

Arsitektur sistem seperti yang dipaparkan pada Gambar 2:

1. Pertama dalam sistem manajemen parkir ini memiliki tampilan antarmuka serba guna, dapat sebagai aplikasi web admin maupun aplikasi mobil web pengguna. Dari situ aplikasi web admin memiliki fitur menambah informasi tempat parkir. Penambahan dan penampilan informasi tempat parkir akan disimpan di database mysql.
2. Kedua, ketika pengguna mencari tempat parkir di sekitar destinasi. Kemudian sistem akan mencarinya dengan prinsip *location based service*. Sistem akan menampilkan layanan tempat parkir yang masih tersedia di lokasi destinasi dan memvisualisasikannya dengan peta digital menggunakan *google maps api*.
3. Seterusnya pengguna akan memilih tempat parkir sesuai dengan keinginannya. Kemudian sistem akan menunjukkan arahan menuju tempat parkir dengan menggunakan GPS. Untuk mengetahui lokasi pengemudi dan menampilkan jalurnya di peta digital.
4. Setelah pengemudi sampai di tempat parkir, pengemudi akan memindai *qr code* yang ada di tempat parkir yang dipesannya sebagai tanda pengemudi telah sampai ditempat parkir dan sebagai tanda tempat parkir sedang terpakai.

E. Rancangan Fungsional

Adapun rancangan fungsional menggunakan *Use Case Diagram* untuk menggambarkan hal-hal yang dapat dilakukan aktor-aktor di dalam sistem. Rancangan ini dapat dilihat pada Gambar 3.

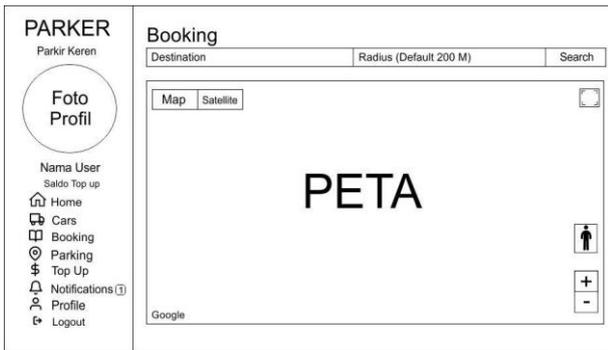


Gambar. 3 Use case diagram

Di dalam *use case diagram* diatas terdapat empat aktor yang berperan dalam sistem manajemen parkir, yaitu pengguna, admin operator, operator dan administrator. *Use case* yang terdapat di dalam sistem seperti berikut ini:

1. Pengguna
 - a. Registrasi
 - b. Login
 - c. Mendaftarkan saldo
 - d. Mengisi saldo
 - e. Memesan ruang parkir
 - f. Membatalkan pemesanan ruang parkir
 - g. Menggunakan ruang parkir
 - h. Meninggalkan ruang parkir
2. Administrator
 - a. Login
 - b. Mendaftarkan operator
 - c. Mendaftarkan tempat parkir
 - d. Memvalidasi pengisian saldo
 - e. Melihat daftar pemesanan
 - f. Melihat daftar pemarkiran
3. Admin Operator
 - a. Login
 - b. Mendaftarkan ruang parkir
 - c. Melihat daftar pemesanan
 - d. Melihat daftar pemarkiran
4. Operator
 - a. Login
 - b. Melihat daftar pemesanan
 - c. Melihat daftar pemarkiran

F. Rancangan Tampilan



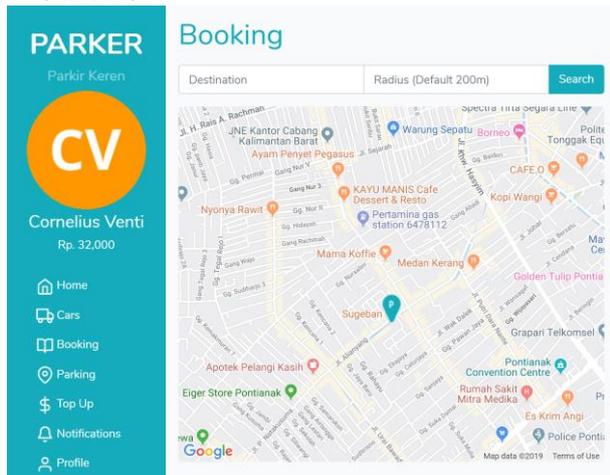
Gambar. 4 Perancangan halaman pemesanan ruang parkir

Rancangan tampilan yang dirancang untuk halaman pemesanan ruang parkir yang tersedia (Gambar 4). Rancangan ini terdiri dari menu *sidebar*, input teks untuk mengisi destinasi dan peta digital untuk menampilkan tempat-tempat parkir di sekitar destinasi yang masih tersedia ruang parkirnya.

IV. IMPLEMENTASI DAN HASIL PENGUJIAN

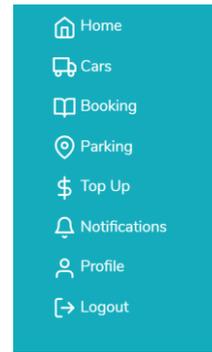
A. Implementasi

Hasil rancangan sistem memiliki kemampuan untuk melakukan pencarian dan pemesanan ruang parkir. Tampilan dari sistem yang telah dirancang dapat dilihat pada Gambar 5.



Gambar. 5 Halaman pemesanan ruang parkir

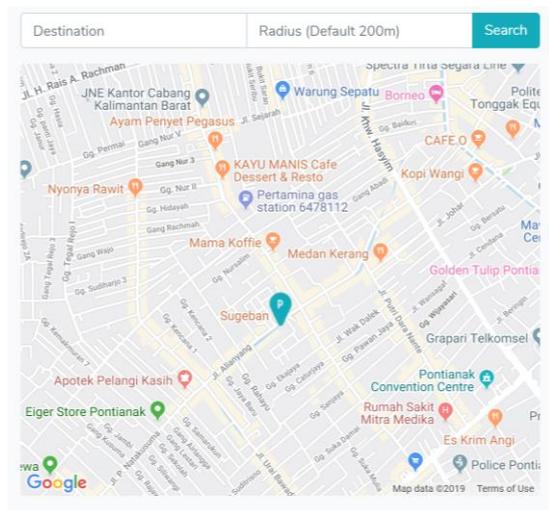
Tampilan halaman pemesanan untuk pengguna dapat mendapatkan informasi ketersediaan ruang parkir, mencari tempat parkir di sekitar destinasi dan memesan dan memesan ruang parkir pada suatu tempat parkir di sekitar destinasi.



Gambar. 6 Menu-menu pada sistem

Dimulai dari menu-menu pada sistem manajemen parkir (Gambar 6), terdiri dari:

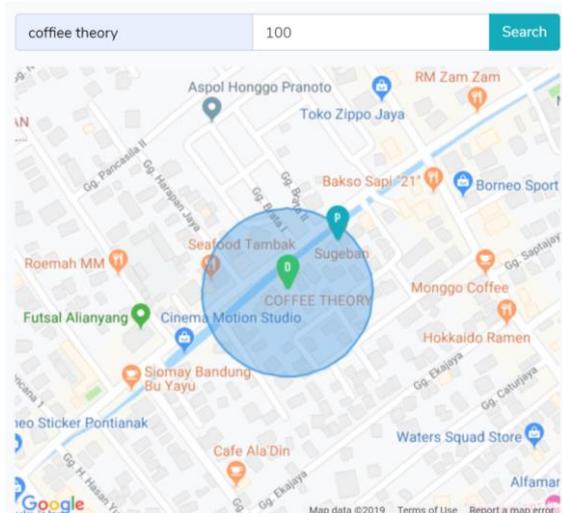
- a. Home, tampilan dashboard.
- b. Car, fitur yang digunakan untuk memasukkan no plat dan merek kendaraan pengguna sebelum melakukan pemesanan.
- c. Booking, fitur yang digunakan untuk melakukan pemesanan parkir.
- d. Parking, fitur yang digunakan untuk melakukan penggunaan parkir.
- e. Top Up, fitur yang digunakan untuk mengirimkan sebagian dana yang diperlukan dari rekening pengguna ke akun sistem manajemen parkir sebelum melakukan pemesanan.
- f. Notification, fitur yang digunakan untuk memberikan informasi kepada pengguna jika pemesanan yang telah dilakukan dalam keadaan masih berlaku atau tidak dan memberikan rincian pemesanan.
- g. Profile, fitur yang digunakan untuk mengatur nama, foto, email, dan untuk mengubah password sebelumnya.



Gambar. 7 Pencarian tempat parkir

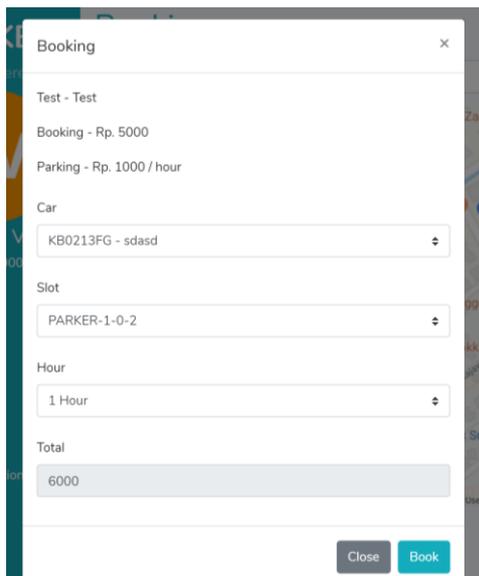
Pencarian tempat parkir (Gambar 7) merupakan bagian dari halaman pemesanan untuk pengguna dapat

mendapatkan informasi ketersediaan ruang parkir dan mencari tempat parkir di sekitar destinasi yang tersedia ruang parkir. Dengan cara memasukkan memasukkan destinasi di input text destinasi dan radius pencarian tempat parkir di sekitar destinasi (standar 200 meter).



Gambar. 8 Pemilihan tempat parkir

Pemilihan tempat parkir (Gambar 8) merupakan bagian dari halaman pemesanan untuk memilih tempat parkir yang telah di saring tempat parkir yang terdapat di sekitar destinasi dan radius yang pengguna masukan, serta tempat parkir yang masih tersedia ruang parkirnya. Kemudian pengguna dapat memilih salah satu tempat parkir yang ada.



Gambar. 9 Pemesanan Ruang Parkir

Pemilihan tempat parkir merupakan bagian dari halaman pemesanan untuk memilih tempat parker (Gambar 9) yang telah di saring tempat parkir yang terdapat di sekitar destinasi dan radius yang pengguna masukan, serta tempat parkir yang masih tersedia ruang

parkirnya. Kemudian pengguna dapat memilih salah satu tempat parkir yang ada.

B. Hasil Pengujian

Pengujian sistem manajemen parkir dilakukan untuk memastikan bahwa aplikasi yang dibangun telah sesuai dengan rancangan yang dibuat. Pengujian yang digunakan untuk menguji sistem ini adalah metode pengujian *Black Box* dan *User Acceptance Test*. Pengujian sistem manajemen parkir menggunakan metode *black box* akan ditampilkan dalam bentuk kasus uji - kasus uji sedangkan untuk *user acceptance test* akan ditampilkan dalam bentuk pertanyaan - pertanyaan yang sudah dibuat dan dijawab oleh pengguna sistem ini.

TABEL I
PENGUJIAN BLACK BOX TESTING

No	Kasus Uji	Hasil yang diharapkan	Hasil	Sesuai
1.	Mendaftarkan mobil	Pengguna dapat memasukan data mobil, sistem dapat menambah data, menginput mobil, edit data dan hapus data	Sistem dapat melakukan <i>create, input, edit</i> dan <i>delete</i> dengan baik	Sesuai
2.	Mengisi saldo	Ketika pengguna memasukan informasi pengisian saldo dan bukti transfer, sistem dapat menyimpan informasi tersebut	Sistem menyimpan informasi pengisian dan bukti transfer dengan benar	Sesuai
3.	Menyediakan informasi tempat parkir yang tersedia	Pengguna dapat mencari tempat parkir di sekitar destinasi, sistem dapat menampilkan tempat-tempat parkir yang tersedia ruang parkirnya	Sistem dapat mencari tempat parkir di sekitar destinasi dan menampilkan tempat parkir yang tersedia ruang parkir	Sesuai
4.	Memesan ruang parkir	Pengguna dapat menginput pemesanan ruang parkir dengan benar	Sistem menyimpan pemesanan tempat parkir	Sesuai
5.	Membatalkan pemesanan ruang parkir	Pengguna dapat membatalkan pemesanan ruang parkir, sistem dapat menghapus pemesanan tersebut	Sistem menghapus pemesanan tempat parkir	Sesuai
6.	Melihat arahan menuju ruang parkir	Sistem dapat menampilkan arahan menuju ruang parkir dalam bentuk peta digital	Sistem menampilkan arahan menuju ruang parkir	Sesuai
7.	Menggunakan ruang parkir	Ketika pengguna sampai di ruang parkir dan memindai qr-code nya, sistem dapat menyimpan waktu mulai penggunaan ruang parkir tersebut.	Sistem menyimpan waktu mulai penggunaan ruang parkir	Sesuai
8.	Meninggalkan ruang parkir	Ketika pengguna sebelum meninggalkan ruang parkir dan	Sistem menyimpan waktu berakhirnya penggunaan ruang	Sesuai

No	Kasus Uji	Hasil yang diharapkan	Hasil	Sesuai
		memindai qr-code nya, sistem dapat menyimpan waktu berakhirnya penggunaan ruang parkir tersebut.	parkir	

Hasil dari setiap kasus uji pada pengujian *Black Box* pada Tabel I yang diuji dengan skenario pengujian, terdapat hasil yang diharapkan pada setiap kasus uji dan hasil yang didapat serta kesesuaian antara hasil yang diharapkan dengan hasil yang didapat.

TABEL II
PENGUJIAN USER ACCEPTANCE TEST

No	Pertanyaan	Jumlah Nilai	Persentase Nilai
1.	Apakah tampilan pada sistem manajemen parkir ini menarik?	47	94%
2.	Apakah sistem ini mudah digunakan / <i>user friendly</i> ?	47	94%
3.	Apakah proses pengisian saldo mudah dilakukan?	47	94%
4.	Apakah informasi tempat parkir yang tersedia mudah dimengerti?	46	92%
5.	Apakah proses pencarian tempat parkir yang tersedia di sekitar destinasi tujuan mudah dilakukan?	46	92%
6.	Apakah proses pemesanan ruang parkir pada tempat parkir yang tersedia mudah dilakukan?	47	94%
7.	Apakah proses pembatalan pemesanan ruang parkir mudah dilakukan?	48	96%
8.	Apakah arahan menuju tempat parkir yang telah dipesan mudah dimengerti?	44	88%
9.	Apakah proses pemindaian qr-code ruang parkir mudah dilakukan?	46	92%
10.	Apakah sistem manajemen parkir ini sudah baik?	47	94%

Hasil akhir dari pengujian *User Acceptance Test* pada Tabel II dapat diperoleh dengan cara merata-ratakan hasil rekap setiap pertanyaan dan dibagi dengan jumlah pertanyaan, yaitu $(70\% + 80\% + 70\% + 70\% + 70\% + 80\% + 80\% + 50\% + 60\% + 80\%) / 10 = 93\%$.

V. KESIMPULAN DAN SARAN

A. Kesimpulan

Perancangan Sistem Manajemen Parkir Mobil *On Street* dan *Off Street* Berbasis *Location Based Service* dan *Google Maps API* bertujuan untuk menyediakan informasi ketersediaan tempat parkir di sekitar destinasi pengendara. Berdasarkan rancangan dan implementasi yang telah dilakukan dan diuraikan pada bab sebelumnya dalam penelitian ini, maka dapat disimpulkan sebagai berikut:

1. Sistem ini dapat menyediakan informasi ketersediaan ruang parkir pada tempat parkir di sekitar tempat destinasi.
2. Melalui sistem ini, pengguna / pengendara mobil dapat memesan ruang parkir pada tempat parkir yang tersedia di sekitar tempat destinasi.
3. Melalui sistem ini, pengendara mobil dapat mengikuti arahan menuju ruang parkir yang telah dipesan sebelumnya.
4. Dengan hasil akhir pengujian *User Acceptance Test*, maka dapat disimpulkan persentase penerimaan sistem manajemen parkir ini sebesar 93%.

B. Saran

Dalam pembuatan Sistem Manajemen Parkir Mobil *On Street* dan *Off Street* Berbasis *Location Based Service* dan *Google Maps API* ini dapat dikembangkan dan ditambahkan fitur-fitur yang akan membuat sistem manajemen parkir ini menjadi lebih flexibel, seperti:

1. Sistem manajemen parkir ini dapat dikembangkan untuk memiliki fitur memesan ruang parkir lebih dari satu. Hal ini dikarenakan agar pengendara dapat memesankan tempat parkir untuk keluarganya atau temannya disaat bersamaan.
2. Sistem manajemen parkir ini dapat dikembangkan untuk memiliki fitur mengisi saldo dengan pengecekan otomatis, supaya pengendara bisa langsung melakukan pemesanan tanpa perlu menunggu konfirmasi dari administrator.

REFERENSI

- [1] "Parkir." [Online]. Available: <https://id.wikipedia.org/wiki/Parkir>. [Accessed: 12-Aug-2019].
- [2] B. Anwar, H. Jaya, and P. I. Kusuma, "Implementasi Locations Based Service Berbasis Android Untuk Mengetahui Posisi User," *J. SAINTIKOM*, vol. 13, no. 2, pp. 121–133, 2014.
- [3] M. Sholeh, N. Widyastuti, and A. Mashuri, "Aplikasi Google Maps API Untuk Sistem Informasi Geografis," *Juita*, vol. 1, no. 3, 2011.
- [4] M. A. Novianta and E. Setyaningsih, "Sistem Informasi Monitoring Kereta Api Berbasis Web Server Menggunakan layanan GPRS," *Momentum*, vol. 17, no. 2, pp. 58–67, 2015.
- [5] R. Meimaharani, "E-Commerce Goody Bag Spunbond Menggunakan Qr Code Berbasis Web Responsif," *Simetris J. Tek. Mesin, Elektro dan Ilmu Komput.*, vol. 5, no. 2, pp. 127–135, 2014.
- [6] W. Suwardjoko, *Merencanakan Sistem Perangkutan*. Bandung: ITB, 1990.
- [7] D. Santosa, "Pengertian, Cara dan Jenis Parkir," [Http://www.galeripustaka.com](http://www.galeripustaka.com), 2013. [Online]. Available: <http://www.galeripustaka.com/2013/05/pengertian-cara-dan-jenis-parkir.html>. [Accessed: 11-Aug-2019].

- [8] S. Y. K. Rompas, "Location Based Service (LBS)," 2013. [Online]. Available: <https://supeerblog.blogspot.com/2013/05/location-based-services-lbs.html>. [Accessed: 28-Aug-2019].
- [9] V. Yasin, *Rekayasa Perangkat Lunak Berorientasi Objek*. Jakarta: Mitra Wacana Media, 2012.
- [10] R. Sukamto and S. M, *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung, 2018.
- [11] R. Soetam, *Konsep Dasar Rekayasa Perangkat Lunak (Software Engineering)*. Jakarta, 2011.
- [12] W. E. Perry., *Effective Methods for Software Testing(second edition)*, 3rd Editio. Indianapolis: Wiley Publishing, Inc, 2000.
- [13] J. Enterprise, *Pengenalan HTML dan CSS*. Jakarta: PT. Elex Media Komputindo, 2016.
- [14] Z. A. Rozi and S. Community, *Modern Web Design*. Jakarta: PT. Elex Media Komputindo, 2016.
- [15] J. Enterprise, *PHP Komplet*. Jakarta: PT. Elex Media Komputindo, 2017.
- [16] J. Enterprise, *Otodidak MySQL untuk Pemula*. Jakarta: PT. Elex Media Komputindo, 2017.
- [17] Y. Yudhanto and H. A. Prasetyo, *Panduan Mudah Belajar Framework Laravel*. Jakarta: PT. Elex Media Komputindo, 2018.