IMPLEMENTASI *LOCATION BASED SERVICE* PADA APLIKASI BENGKEL MOTOR BERBASIS ANDROID

(Skripsi)

Oleh

NUR HIDAYAT

NPM 1517051212

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2022

IMPLEMENTASI *LOCATION BASED SERVICE* PADA APLIKASI BENGKEL MOTOR BERBASIS ANDROID

Oleh

NUR HIDAYAT

Skripsi

Sebagai Salah Satu Syarat untuk Memperoleh Gelar
SARJANA KOMPUTER

Pada
Jurusan Ilmu Komputer
Fakultas Matematika dan Ilmu Pengetahuan Alam

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS LAMPUNG

BANDAR LAMPUNG

2022

ABSTRAK

IMPLEMENTASI *LOCATION BASED SERVICE* PADA APLIKASI BENGKEL MOTOR BERBASIS ANDROID

Oleh

NUR HIDAYAT

Bengkel dapat membantu pengguna kendaraan bermotor untuk merawat kendaraannya agar selalu dalam kondisi yang baik sehingga pengendara dapat mengurangi risiko kecelakaan yang dapat terjadi. Sering kali pengendara juga membutuhkan bengkel saat sedang di tengah perjalanan dan mengalami kesulitan mencari letak bengkel terdekat untuk memperbaiki kendaraannya. Dengan teknologi Android pada perangkat *mobile* saat ini penggunanya dapat menggunakan GPS untuk memberikan layanan berbasis lokasi atau *Location Based Service* (LBS). Teknologi GPS dapat dimanfaatkan sebagai pemandu dalam pencarian suatu tempat seperti bengkel.

Hasil penelitian menggunakan SDLC Model Waterfall yang dikerjakan secara bertahap dan berurutan dimulai dari komunikasi, perencanaan, pemodelan, kontruksi atau penulisan kode program dengan menggunakan bahasa pemrograman Kotlin serta memanfaatkan Mapbox dan Firebase sebagai penyedia layanan dan konten, pengujian program dilakukan menggunakan *black-box testing* untuk memastikan bahwa program telah sesuai dengan desainnya dan semua fungsi dapat dipergunakan dengan baik dan *user acceptance testing* untuk mengukur bagaimana sistem sudah sesuai dengan kebutuhan pengguna.

Penelitian ini telah berhasil melakukan Implementasi *Location Based Servic*e Pada Aplikasi Bengkel Motor Berbasis Android. Kesimpulan dari penelitian ini adalah Implementasi *Location Based Service* Pada Aplikasi Bengkel Motor Berbasis Android telah berhasil dibangun dan telah diuji agar sesuai dengan hasil yang diharapkan.

Kata Kunci: Bengkel, Kendaraan Bermotor, Android, *Mobile*, Waterfall, Kotlin, Mapbox, Firebase, *Location Based Service, Black-Box Testing, User Acceptance Testing*.

ABSTRACT

IMPLEMENTATION OF LOCATION BASED SERVICE IN ANDROID BASED MOTORCYCLE SHOP APPLICATIONS

By

NUR HIDAYAT

Motorcycle repair shops can help motorized vehicle users maintain their vehicles so that they are always in good condition so that motorists can reduce the risk of accidents. Often motorists also need a motorcycle repair shop when they are in the middle of the trip and have difficulty finding the nearest repair shop to repair their vehicle. With Android technology on mobile devices, users can now use GPS to provide location-based services (LBS). GPS technology can be used as a guide in the search for a place such as a workshop.

The results of the study using the Waterfall Model SDLC were carried out in stages and sequentially starting from communication, planning, modeling, construction, or writing program code using the Kotlin programming language and utilizing Mapbox and Firebase as service and content providers, program testing was carried out using black-box testing to ensure that the program is in accordance with its design and that all functions can be used properly and user acceptance testing to measure how the system is in accordance with user needs.

This research has succeeded in implementing Location Based Service on Android-Based Motorcycle Workshop Applications. This research concludes that the Implementation of Location Based Service on Android-Based Motorcycle Workshop Applications has been successfully built and tested to match the expected results.

Keywords: Motorcycle repair shop, Motorized Vehicle, Android, Mobile, Waterfall, Kotlin, Mapbox, Firebase, Location Based Service, Black-Box Testing, User Acceptance Testing.

Judul Skripsi	:	**IMPLEMENTASI *LOCATION BASED SERVICE* PADA APLIKASI BENGKEL MOTOR ERBASIS ANDROID**
Nama Mahasiswa	:	Nur Hidayat
Nomor Pokok Mahasiswa	:	1517051212
Jurusan	:	Ilmu Komputer
Fakultas	:	Matematika dan Ilmu Pengetahuan Alam

MENYETUJUI

1. Komisi Pembimbing

Dwi Sakethi, S.Si., M.Kom.
NIP 19680611 199802 1 001

Ardiansyah, S.Kom., M.Kom.
NIP 19870128 201803 1 001

2. Ketua Jurusan Ilmu Komputer

Didik Kurniawan, S.Si., M.T.
NIP 19800419 200501 1 004

MENGESAHKAN

1. Tim Penguji

   Ketua : **Dwi Sakethi, S.Si., M.Kom.** .....

   Sekretaris : **Ardiansyah, S.Kom., M.Kom.** .....

   Penguji
   Bukan Pembimbing : **Drs. Rd. Irwan Adi Pribadi, M.Kom.** .....

2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam

Dr. Eng. Suripto Dwi Yuwono, S.Si., M.T.
NIP 19740705 200003 1 001

Tanggal Lulus Ujian Skripsi: **20 Juni 2022**

PERNYATAAN

Saya yang bertanda tangan di bawah ini, menyatakan bahwa skripsi saya yang berjudul "Implementasi *Location Based Service* Pada Aplikasi Bengkel Motor Berbasis Android" merupakan karya saya sendiri dan bukan karya orang lain. Semua tulisan yang tertuang di skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila di kemudian hari terbukti skripsi saya merupakan hasil penjiplakan atau dibuat orang lain, maka saya bersedia menerima sanksi berupa pencabutan gelar yang telah saya terima.

Bandar Lampung, 20 Juni 2022

Nur Hidayat
NPM. 1517051212

RIWAYAT HIDUP

Penulis dilahirkan pada 20 Juni 1998 di Bandar Harapan, Terbanggi Besar, Lampung Tengah, sebagai anak kedua dari 3 saudara dengan Ayah bernama Saepudin dan Ibu bernama Nining Rohayati.

Penulis menyelesaikan pendidikan Sekolah Dasar (SD) di SD N 3 Poncowati pada tahun 2010, menyelesaikan Sekolah Menengah Pertama (SMP) di SMP N 1 Terbanggi Besar pada tahun 2013. Kemudian melanjutkan jenjang Sekolah Menengah Atas (SMA) di SMA Negeri 1 Terbanggi Besar dan lulus pada tahun 2015.

Pada tahun 2015, penulis terdaftar sebagai mahasiswa Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam. Pada bulan Januari-Maret 2018 Penulis melakukan Kerja Praktik di Pengadilan Agama Tanjung Karang. Pada bulan Juli-Agustus 2018 Penulis melaksanakan Kuliah Kerja Nyata di Tulung Pasik, Kecamatan Mataram Baru, Kabupaten Lampung Timur

MOTO

"Sesungguhnya Allah menyuruh (kamu) berlaku adil dan berbuat kebajikan, memberi bantuan kepada kerabat, dan Dia melarang (melakukan) perbuatan keji, kemungkaran, dan permusuhan. Dia memberi pengajaran kepadamu agar kamu dapat mengambil pelajaran."

(QS. An-Nahl 16: Ayat 90)

PERSEMBAHAN

Puji dan syukur saya ucapkan kepada Allah SWT atas segala nikmat dan karunia-Nya sehingga skripsi ini dapat diselesaikan.

Kupersembahkan skripsi ini untuk Ayah dan Ibu tercinta
yang memberikan kasih sayang, pengorbanan, usaha, dukungan moril,
maupun materi dan doa yang tiada henti untuk kesuksesanku
serta Adik dan keluarga besar yang selalu mendukung

Teruntuk sahabat dan teman-teman tersayang,
terima kasih untuk setiap canda tawa, tangis dan perjuangan yang telah kita lewati
bersama serta semua kenangan yang telah terukir selama ini

Keluarga Ilmu Komputer 2015
dan, Almamater yang kubanggakan
UNIVERSITAS LAMPUNG

SANWACANA

Assalamu'alaikum Warahmatullahi Wabarakatuh.

Alhamdulillahirabbil'alamin, segala puji bagi Allah Subhanahu wa ta'ala yang telah melimpahkan rahmat, hidayah, kesehatan, serta karunia-Nya sehingga Penulis dapat menyelesaikan penulisan skripsi yang berjudul "Implementasi Location Based Service Pada Aplikasi Bengkel Motor Berbasis Android" dengan baik. Shalawat serta salam semoga senantiasa tercurahkan kepada Nabi Muhammad Shalallahu 'alaihi wa sallam, keluarganya, sahabatnya, dan mudah-mudahan kita adalah umat yang diberikan syafa'atnya di hari akhir kelak, Aamiin ya rabbal 'alamiin.

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada semua pihak yang telah membantu dan berperan besar selama proses penyusunan skripsi ini, antara lain:

1. Kedua orang tua tercinta, Ayah dan Ibu, Kakak, Adik, berserta keluarga besar yang selalu memberi doa, motivasi, dan kasih sayang yang tak terhingga.
2. Bapak Dwi Sakethi, S.Si., M.Kom., sebagai Pembimbing Utama, yang telah membimbing serta memberikan kritik dan saran selama masa perkuliahan dan penyusunan skripsi sehingga Penulis bisa sampai ditahap ini.
3. Bapak Ardiansyah, S.Kom., M.Kom., sebagai Pembimbing II, yang telah membimbing serta memberikan kritik dan saran selama masa perkuliahan dan penyusunan skripsi sehingga Penulis bisa sampai ditahap ini.
4. Bapak Drs. Rd. Irwan Adi Pribadi, M.Kom., sebagai Pembahas Utama, yang telah memberikan saran dan masukan yang bermanfaat untuk perbaikan dalam penyusunan skripsi ini.

5. Bapak Aristoteles, S.Si., M.Si., sebagai Pembimbing Akademik yang telah membimbing, memotivasi, serta memberikan ide, kritik dan saran selama masa perkuliahan.
6. Bapak Dr. Eng. Suripto Dwi Yuwono, S.Si., M.T., selaku Dekan FMIPA Universitas Lampung.
7. Bapak Didik Kurniawan, S.Si., M.T., selaku Ketua Jurusan Ilmu Komputer FMIPA Universitas Lampung.
8. Bapak Dr. rer. nat. Akmal Junaidi, M.Sc., selaku Sekretaris Jurusan Ilmu Komputer FMIPA Universitas Lampung.
9. Bapak dan Ibu Dosen Jurusan Ilmu Komputer yang telah memberikan ilmu dan pengalaman hidup selama Penulis menjadi mahasiswa.
10. Ibu Ade Nora Maela yang telah membantu segala urusan administrasi Penulis di Jurusan Ilmu Komputer.
11. Teman-teman Ilmu Komputer kelas D angkatan 2015 yang menjadi teman pada setiap mata kuliah dan tempat berbagi canda tawa.
12. Keluarga besar Ilmu Komputer 2015 yang telah memberikan kenangan selama masa perkuliahan.
13. Almamater Tercinta, Universitas Lampung yang telah memberikan Penulis kesempatan untuk menempuh pendidikan perkuliahan S1 dengan baik.

Penulis telah berusaha semaksimal mungkin dalam penulisan skripsi ini untuk mencapai satu kelengkapan dan kesempurnaan. Penulis juga mengharapkan saran dan kritik yang bersifat membangun dari semua pihak. Akhirnya dengan segala kerendahan hati penulis berharap skripsi ini dapat memberikan manfaat baik kepada penulis khususnya maupun kepada pembaca pada umumnya.

Bandar Lampung, 20 Juni 2022
Penulis,

Nur Hidayat

DAFTAR ISI

Halaman

DAFTAR ISI ..... **iv**
DAFTAR GAMBAR ..... **vi**
DAFTAR TABEL ..... **viii**
I. PENDAHULUAN **1**
A. Latar Belakang 1
B. Rumusan Masalah 3
C. Batasan Masalah 3
D. Tujuan Penelitian 3
E. Manfaat Penelitian 3
II. TINJAUAN PUSTAKA **4**
A. Pengertian Bengkel Motor 4
B. *Location Based Services* (LBS) 4
C. Android 6
D. Mapbox 6
E. Firebase 7
F. Perangkat Lunak Pendukung 7
1. Android Studio 7
2. *Java Development Kit* (JDK) 8
G. UML *(Unified Modelling Language)* 8
1. *Use Case Diagram* 9
2. *Activity Diagram* 9
3. *Entity Relationship Diagram* 10
H. *User Acceptance Testing* 11
I. *Black-Box Testing* 11
J. *Skala Likert* 12

III. METODOLOGI PENELITIAN 13

A. Waktu dan Tempat Penelitian 13

B. Alat Pendukung Penelitian 13

1. Perangkat Keras *(Hardware)* 13
2. Perangkat Lunak *(Software)* 14

C. Tahapan Pengembangan Sistem 14

1. Tahap Komunikasi 15
2. Tahap Perencanaan 17
3. Tahap Pemodelan 17
4. Tahap Kontruksi 30
5. Tahap *Deployment* 34

IV. HASIL DAN PEMBAHASAN 35

A. Hasil 35

B. Implementasi Sistem 36

1. Tampilan Menu Peta 36
2. Tampilan Menu Ulasan 37
3. Tampilan Menu Login 38
4. Tampilan Menu Menambah Data Bengkel 39
5. Tampilan Menu Bengkel 40
6. Tampilan Menu Admin 40

C. Pengujian Sistem 41

1. Pengujian Fungsional 42
2. Pengujian *User Acceptance* 42

V. SIMPULAN DAN SARAN 45

A. Simpulan 45

B. Saran 46

DAFTAR PUSTAKA 47

LAMPIRAN 50

DAFTAR GAMBAR

Gambar Halaman

Gambar 1 Komponen Pendukung LBS (Anwar dkk., 2014). 5
Gambar 2 Metode *Waterfall* (Pressman, 2012)..... 15
Gambar 3 *Use Case Diagram* Aplikasi Bengkel. 16
Gambar 4 *Activity Diagram* Melihat Lokasi Dan Detail Bengkel. 18
Gambar 5 *Activity Diagram* Melihat Ulasan Dari Bengkel. 19
Gambar 6 *Activity Diagram* Menambah Ulasan Pada Bengkel. 20
Gambar 7 *Activity Diagram* Menambahkan Data Bengkel..... 21
Gambar 8 *Activity Diagram* Melihat Data Bengkel. 22
Gambar 9 *Activity Diagram* Mengedit Data Bengkel. 23
Gambar 10 *Activity Diagram* Memverifikasi Data Bengkel. 24
Gambar 11 *Entity Relationship Diagram.* 25
Gambar 12 Rancangan *Interface* Menu *Login.* 26
Gambar 13 Rancangan *Interface* Menu *Register*. 26
Gambar 14 Rancangan *Interface* Menu Lihat Lokasi Bengkel. 27
Gambar 15 Rancangan *Interface* Lihat Ulasan Bengkel. 27
Gambar 16 Rancangan *Interface* Menambahkan Ulasan. 28
Gambar 17 Rancangan *Interface* Menu Menambahkan Data Bengkel. 28
Gambar 18 Rancangan *Interface* Menu Mengelola Data Bengkel. 29
Gambar 19 Rancangan *Interface* Kelola Akun Profil. 29
Gambar 20 Menu Peta. 36
Gambar 21 (a) Menu Ulasan dan (b) Menu Menambah Ulasan 37
Gambar 22 Tampilan Menu *Login.* 38
Gambar 23 (a) Menu Pilih Lokasi dan (b) Menu Menambah Data Bengkel. 39
Gambar 24 Tampilan Menu Data Bengkel. 40

Gambar 25 Tampilan Menu Admin. 41

DAFTAR TABEL

Tabel Halaman

Tabel 1 Simbol dan Keterangan *Use Case Diagram.* 9
Tabel 2 Simbol dan Keterangan *Activity Diagram.* 10
Tabel 3 Identifikasi Pengujian. 30
Tabel 4 *Login.* 31
Tabel 5 Menambahkan Data. 31
Tabel 6 Mengubah Data. 32
Tabel 7 Peta. 32
Tabel 8 Informasi Akun. 33
Tabel 9 Ulasan. 33
Tabel 10. Kriteria Penilaian Responden 43
Tabel 11. Hasil Penilaian Responden 44

I. PENDAHULUAN

A. Latar Belakang

Jumlah kendaraan bermotor yang beredar di Indonesia pada periode 2016-2020 setiap tahunnya mengalami kenaikan sebesar 4,95 persen setiap tahunnya. Peningkatan jumlah tertinggi terjadi pada sepeda motor sebesar 5,03 persen setiap tahunnya (Yuwono, 2021). Kendaraan yang dipakai harus selalu dalam keadaan baik serta memenuhi persyaratan teknis layak jalan pada kendaraan bermotor. Sesuai dengan UU No. 22 Tahun 2009 Pasal 48 ayat 1 yang berbunyi: "Setiap kendaraan bermotor yang dioperasikan di jalan harus memenuhi persyaratan teknis dan layak jalan". Dengan memenuhi persyaratan teknis maka keselamatan dalam berkendara dan risiko kecelakaan yang terjadi di jalanan semakin berkurang.

Pengemudi kendaraan diharuskan mengecek kendaraannya sebelum bepergian agar kendaraan selalu dalam keadaan baik, maka diperlukan perawatan berkala dengan memanfaatkan jasa perbaikan pada kendaraan. Jasa yang dimaksud adalah bengkel motor. Bengkel motor merupakan salah satu usaha bisnis yang bergerak di bidang otomotif. Ketika pengendara melakukan perjalanan banyak hal yang tidak diinginkan terjadi di tengah perjalanan seperti masalah ban bocor, masalah mesin, masalah lampu, dan sebagainya sehingga pengendara harus mencari bengkel motor terdekat untuk memperbaikinya. Sering kali pengendara yang sedang dalam perjalanan mengalami kesulitan mencari letak bengkel motor terdekat untuk memperbaiki kendaraannya.

Dengan perkembangan teknologi saat ini khususnya pada telepon seluler membuat pengguna dapat memanfaatkannya untuk beragam fungsi tambahan. Penambahan GPS pada telepon seluler dapat dimanfaatkan untuk memberikan layanan berbasis lokasi atau *Location Based Service* (LBS) yaitu sebuah layanan informasi yang dapat diakses dengan *smartphone* melalui jaringan seluler dan mampu memanfaatkan posisi geografis pada perangkat *mobile* tersebut. LBS memungkinkan pengguna mendapatkan informasi sesuai dengan lokasi keberadaan pengguna (Putra dkk., 2016). Dengan adanya teknologi GPS pada Android ini dapat dimanfaatkan juga sebagai pemandu dalam pencarian suatu tempat seperti bengkel motor.

Telah banyak penelitian yang meneliti tentang implementasi *Location Based Service* (LBS) pada aplikasi Android di antaranya sistem untuk mempermudah dalam melakukan donasi barang bekas layak pakai dengan fitur menampilkan lokasi lembaga donasi serta daftar donasi yang dikelola lembaga tersebut dan fitur pesan untuk mempermudah komunikasi antar pengguna dan lembaga donasi (Pribadi *et al*., 2020). Sistem untuk menampilkan bengkel vespa terdekat, menampilkan alamat bengkel dan pencarian lokasi bengkel di Bandar Lampung (Apriawan dkk., 2021). Sistem untuk menampilkan lokasi ATM berdasarkan jenisnya, nama jalan atau di Bandar Lampung (Erika dkk., 2016). Namun pada ketiga penelitian di atas belum menampilkan informasi lokasi seperti jarak, waktu tempuh, dan rute.

Berdasarkan masalah yang telah dipaparkan, maka dikembangkan aplikasi bengkel motor dengan mengakses aplikasi menggunakan *smartphone* berbasis Android menggunakan metode *Location Based Service* (LBS). Diharapkan dengan dikembangkannya aplikasi ini dapat membantu pengguna kendaraan dalam melakukan pencarian bengkel motor terdekat.

B. Rumusan Masalah

Dalam penulisan penelitian ini, penulis menyimpulkan perumusan masalah yaitu bagaimana mengembangkan sebuah aplikasi bengkel motor berbasis Android dengan layanan *Location Based Service* (LBS) yang dapat memberikan informasi lokasi bengkel motor.

C. Batasan Masalah

Batasan masalah dari penelitian ini adalah sebagai berikut:

1. Aplikasi ini dikembangkan untuk *smartphone* berbasis Android.
2. Aplikasi ini membutuhkan perizinan internet dan GPS (*Global Positioning System)*.
3. Aplikasi ini menampilkan informasi lokasi bengkel, detail bengkel dan ulasan dari bengkel tersebut berdasarkan yang telah ditambahkan oleh pengguna.
4. Aplikasi ini menampilkan data bengkel untuk motor.

D. Tujuan Penelitian

Tujuan dari penelitian ini adalah mengembangkan suatu aplikasi berbasis Android dengan layanan *Location Based Service* (LBS).

E. Manfaat Penelitian

Manfaat dari penelitian yang dilakukan adalah untuk memudahkan pengguna kendaraan bermotor untuk menemukan bengkel motor terdekat.

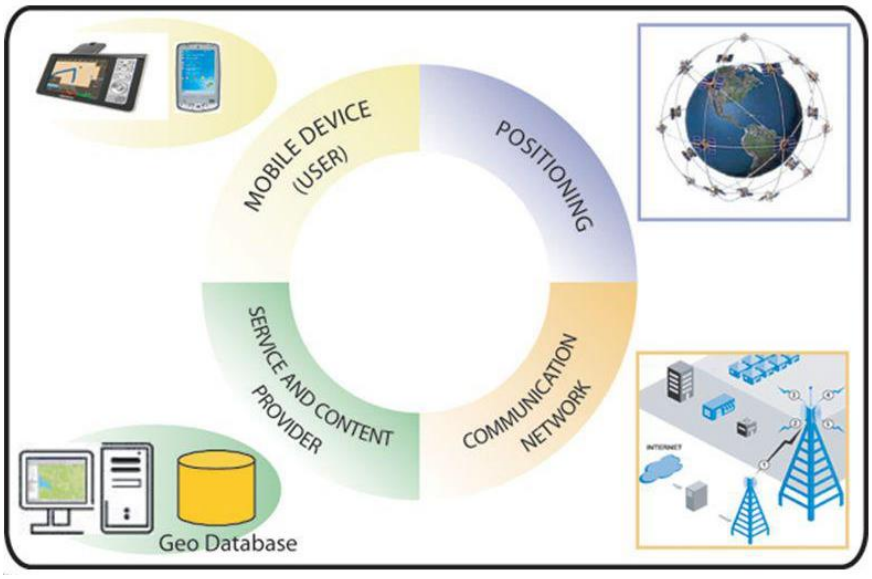
II. TINJAUAN PUSTAKA

A. Pengertian Bengkel Motor

Bengkel merupakan tempat yang menyediakan ruang dan peralatan untuk membuat, membentuk, merakit, mengubah bentuk, ataupun memperbaiki suatu benda menjadi bentuk yang baru atau kondisi yang lebih baik secara manfaat. Bengkel motor adalah tempat di mana kendaraan (motor) diperbaiki oleh teknisi atau tenaga mekanik. Bengkel ini pada umumnya berfungsi untuk melayani keperluan teknis pelanggan seperti memperbaiki kendaraan pelanggan yang mengalami kerusakan atau kesulitan pada kendaraannya (Rushamidiwinata dan Wartika, 2019).

B. *Location Based Services* (LBS)

Location Based 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 (Anwar dkk., 2014). Dalam Layanan Berbasis Lokasi terdapat lima komponen penting seperti terlihat pada Gambar 1.



Gambar 1 Komponen Pendukung LBS (Anwar dkk., 2014).

Terdapat empat komponen pendukung utama dalam teknologi Layanan Berbasis Lokasi, yaitu:

1. Peranti *Mobile* adalah salah satu komponen penting dalam LBS. Peranti ini berfungsi sebagai alat bantu (*tool*) bagi pengguna untuk meminta informasi. Hasil dari informasi yang diminta dapat berupa teks, suara, gambar dan lain sebagainya.
2. Jaringan Komunikasi Komponen ini berfungsi sebagai jalur penghubung yang dapat mengirimkan data-data yang dikirim oleh pengguna dari peranti *mobile* untuk kemudian dikirimkan ke penyedia layanan dan kemudian hasil permintaan tersebut dikirimkan kembali oleh penyedia layanan kepada pengguna.
3. Komponen *Positioning* (Penunjuk Posisi) Setiap layanan yang diberikan oleh penyedia layanan biasanya akan berdasarkan pada posisi pengguna yang meminta layanan tersebut. Oleh karena itu diperlukan komponen yang berfungsi sebagai pengolah/pemroses yang akan menentukan posisi pengguna layanan saat itu. Posisi pengguna tersebut bisa didapatkan melalui jaringan komunikasi *mobile* atau juga menggunakan *Global Positioning System* (GPS).

4. Penyedia layanan dan konten (*Service and Content Provider*) Penyedia layanan merupakan komponen LBS yang memberikan berbagai macam layanan yang bisa digunakan oleh pengguna.

C. Android

Android adalah sebuah sistem operasi untuk *smartphone* dan *tablet*. Sistem operasi dapat diilustrasikan sebagai jembatan antara peranti (*device*) dan penggunanya, sehingga pengguna dapat berinteraksi dan menjalankan aplikasi-aplikasi yang tersedia pada *device* (Zaki dan Putra, 2018).

Tidak hanya menjadi sistem operasi di *smartphone,* saat ini Android menjadi pesaing utama dari Apple pada saat operasi tablet PC. Pesatnya pertumbuhan Android selain faktor yang disebutkan di atas adalah karena Android itu sendiri adalah *platform* yang sangat lengkap baik itu sistem operasinya, aplikasi dan *tool* pengembangan, *market* aplikasi Android serta dukungan yang tinggi dari komunitas *open source* di dunia sehingga Android terus berkembang pesat dengan baik dari segi teknologi maupun dari segi jumlah *device* yang ada di dunia (Wibowo dan Setiyanto, 2012).

D. Mapbox

Mapbox adalah penyedia layanan yang memungkinkan untuk menyesuaikan peta sepenuhnya dalam setiap detail, termasuk tema warna, fitur yang ingin diberikan kepada pengguna, informasi yang ditampilkan, dan banyak lagi. Dengan menggunakan alat yang disediakan oleh Mapbox dapat mengunggah dan menerbitkan set data dan mengintegrasikannya dengan data Mapbox itu sendiri (Kastanakis, 2016).

E. Firebase

Firebase diluncurkan pada bulan April 2012 oleh James Tamplin dan Andrew Lee. Pada tahap awalnya, Firebase menyediakan API yang membantu mengintegrasikan modul *online chat* ke situs web. Sekarang Firebase adalah salah satu platform BaaS (*Backend as a Service*) yang dominan yang terus meningkatkan pengalaman *cloud* dengan memperkenalkan fitur dan fungsi baru. Firebase paling cocok jika waktu pengembangan singkat dan aplikasi meminta data secara real-time karena mudah untuk diskalakan (Srinivas, 2018).

F. Perangkat Lunak Pendukung

Berikut merupakan perangkat lunak pendukung dalam penunjang pembuatan aplikasi, yaitu:

1. Android Studio

Android Studio adalah IDE (*Integrated Development Environment*) resmi untuk pengembangan aplikasi Android dan bersifat *open source* atau gratis. Peluncuran Android Studio ini diumumkan oleh Google pada 16 Mei 2013 pada *event* Google I/O *Conference* untuk tahun 2013. Sejak saat itu, Android Studio menggantikan *Eclipse* sebagai IDE resmi untuk mengembangkan aplikasi Android. Android studio sendiri dikembangkan berdasarkan IntelliJ IDEA yang mirip dengan *Eclipse* disertai dengan *Android Development Tools* (ADT) *plugin* (Juansyah, 2015).

2. *Java Development Kit* (JDK)

Java Development Kit (JDK) adalah sekumpulan perangkat lunak yang dapat digunakan untuk mengembangkan perangkat lunak yang berbasis Java, sedangkan JRE adalah sebuah implementasi dari *Java Virtual Machine* yang benar-benar digunakan untuk menjalankan program Java. Biasanya, setiap JDK berisi satu atau lebih JRE dan berbagai alat pengembangan lain seperti sumber *compiler java, bundling, debuggers, development libraries* dan lain sebagainya (Juansyah, 2015).

G. UML *(Unified Modelling Language)*

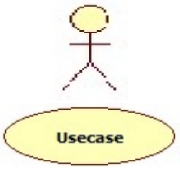
Unified Modeling Language (UML) adalah salah satu alat bantu yang sangat andal di dunia pengembangan sistem yang berorientasi obyek. Hal ini disebabkan karena UML menyediakan bahasa pemodelan visual yang memungkinkan bagi pengembang sistem untuk membuat visi mereka dalam bentuk yang baku, mudah dimengerti serta dilengkapi dengan mekanisme rancangan mereka yang lain. UML merupakan kesatuan dari bahasa pemodelan yang dikembangkan oleh Booch, *Object Modeling Technique* (OMT) dan *Object Oriented Software Engineering* (OOSE). Metode ini menjadikan proses analisis dan desain ke dalam empat tahapan iteratif, yaitu: identifikasi kelas-kelas dan obyek-obyek, identifikasi semantik dari hubungan obyek dan kelas tersebut, perincian *interface* dan implementasi (Munawar, 2005).

Unified Modelling Language (UML) adalah keluarga notasi grafis yang didukung oleh meta-model tunggal, yang membantu pendeskripsian dan desain sistem 20 perangkat lunak, khususnya sistem yang dikembangkan menggunakan pemrograman berorientasi objek. UML dideskripsikan oleh beberapa diagram seperti:

1. *Use Case Diagram*

Diagram *use case* atau *use case diagram* menyajikan interaksi antara *use case* dan aktor. Di mana aktor dapat berupa orang, peralatan, atau sistem lain yang berinteraksi dengan sistem yang sedang dikembangkan (Sholiq, 2006). Simbol dan keterangan *use case* pada Tabel 1.

Tabel 1 Simbol dan Keterangan *Use Case Diagram.*

SIMBOL	KETERANGAN
	Mewakili peran orang, sistem yang lain atau alat ketika berkomunikasi dengan *use case.* Abstraksi dari interaksi antara sistem dan *actor*.
Association ————	Abstraksi dari penghubung antara *actor* dan *use case.*
Generalisasi - - - - - - ->	Menunjukkan spesialisasi *actor* untuk dapat berpartisipasi dalam *use case*.
Extend <———	Menspesifikasikan bahwa *use case* target memperluas perilaku dari *use case* sumber pada suatu titik yang diberikan.
Include - - - - - - ->	Menunjukkan bahwa suatu *use case* seluruhnya merupakan fungsionalitas dari *use case* lainnya.

2. *Activity Diagram*

Diagram Aktivitas atau *Activity Diagram* menggambarkan aliran fungsional sistem. Dapat juga digunakan untuk menggambarkan

aliran kejadian (*flow of event*) dalam *use case* (Sholiq, 2006). Simbol dan keterangan *Activity Diagram* seperti pada Tabel 2.

Tabel 2 Simbol dan Keterangan *Activity Diagram.*

SIMBOL	KETERANGAN
Status Awal	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan/decision	Asosiasi percabangan di mana jika ada pilihan aktivitas lebih dari satu.
Status Akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

3. *Entity Relationship Diagram*

ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional. ERD memiliki beberapa aliran notasi seperti notasi Chen (dikembangkan oleh Peter Chen), Barker (dikembangkan oleh Richard Barker, Ian Palmer, Harry Ellis), notasi Crow's Foot, dan beberapa notasi lain. *Entity Relationship Diagram* (ERD) merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi (Meliani dkk., 2020).

H. *User Acceptance Testing*

User Acceptance Testing (UAT) merupakan salah satu metodologi yang sangat inovatif untuk mencegah kegagalan proyek IT. Dalam pengembangan perangkat lunak, terdapat tiga hal yang dilakukan dalam proses UAT yaitu:

1. UAT mengukur bagaimana sistem sudah sesuai dengan kebutuhan pengguna.
2. UAT mengekspos fungsionalitas/*logic* bisnis yang belum ditemukan, karena unit testing dan sistem testing tidak berfokus pada fungsionalitas/*logic* bisnis.
3. UAT membatasi bagaimana sistem telah selesai dibuat.

Efektifitas dalam pengujian sangat dibutuhkan dalam pengembangan sebuah aplikasi ataupun sistem informasi sehingga produk tersebut dapat sampai kepada pengguna dengan tepat waktu dan sesuai dengan kebutuhan pengguna (Munthe dkk., 2015).

I. *Black-Box Testing*

Pengujian *black-box* merupakan suatu teknik pengujian perangkat lunak dengan berfokus pada persyaratan fungsional. Pengujian *black-box* memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi *input* yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. Pengujian *black-box* berusaha menemukan kesalahan dalam kategori sebagai berikut:

1. Fungsi-fungsi yang tidak benar atau hilang.
2. Kesalahan *Interface*.
3. Kesalahan dalam struktur data atau akses *database* eksternal.
4. Kesalahan kinerja.
5. Inisialisasi dan kesalahan terminasi.

Pengujian *black-box* diaplikasikan selama tahap akhir pengujian, karena *black-box* memperhatikan struktur kontrol, maka perhatian berfokus pada domain informasi (Pressman, 2012).

J. *Skala Likert*

Menurut Azwar (2011), skala likert merupakan metode penskalaan pernyataan sikap yang menggunakan distribusi respons sebagai dasar penentuan nilai skalanya. Nilai skala setiap pernyataan tidak ditentukan oleh derajat *favourable* masing-masing, akan tetapi ditentukan oleh distribusi respons setuju dan tidak setuju dari sekelompok responden yang bertindak sebagai kelompok uji coba.

Skala Likert, yaitu skala yang berisi lima tingkat preferensi jawaban dengan pilihan sebagai berikut: 1=sangat setuju; 2=tidak setuju; 3=ragu-ragu atau netral; 4=setuju; 5=sangat setuju. Selanjutnya, penentuan kategori interval tinggi, sedang, atau rendah digunakan rumus sebagai berikut:

$$\mathrm{I} = \frac{\mathrm{NT} - \mathrm{NR}}{\mathrm{K}}$$

Keterangan:

I = Interval;

NT = Total nilai tertinggi;

NR = Total nilai terendah;

K = Kategori jawaban.

III. METODOLOGI PENELITIAN

A. Waktu dan Tempat Penelitian

Penelitian dilaksanakan di Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung yang berada di Jalan Soemantri Brojonegoro No.1 Gedung Meneng, Bandar Lampung. Penelitian ini dilaksanakan mulai bulan Juli 2019 sampai dengan Juni 2022.

B. Alat Pendukung Penelitian

Alat bantu yang digunakan berupa perangkat keras (*hardware)* dan perangkat lunak (*software).* Alat tersebut di antaranya adalah sebagai berikut:

1. Perangkat Keras *(Hardware)*

Perangkat keras yang digunakan dalam penelitian ini adalah laptop dengan spesifikasi berikut:

- *Processor*: Intel® Core™ i7 7700 HQ,
- *Memory*/RAM: DDR 4-2400 Kapasitas 16 Gigabyte,
- *Graphic*/VGA: Nvidia GeForce GTX 1050 Ti 2 Gigabyte,
- SSD: Kapasitas 512 Gigabyte,
- HDD: Kapasitas 1 Terabyte,

- *Baterai*: 6-*Cells*, Li-Ion 42 Whr,
- *Display*: Layar berukuran 15,6 *Inch*.

2. Perangkat Lunak *(Software)*

Perangkat lunak yang digunakan dalam penelitian ini adalah sebagai berikut:

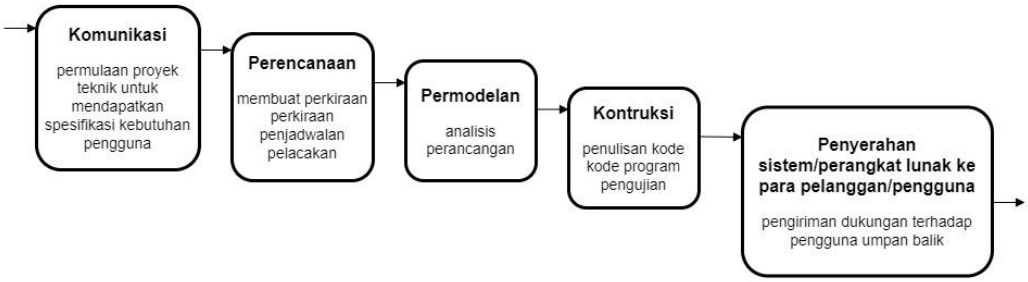
- Sistem Operasi Microsoft Windows 10 Pro 64 Bit,
- Android Studio dan *emulator* dengan Android versi 11 sebagai aplikasi untuk menulis dan menjalankan *code* program,
- Star UML sebagai aplikasi untuk membuat diagram model sistem informasi, dan
- ProtoDream.io sebagai aplikasi untuk membuat rancangan desain *interface* aplikasi.

C. Tahapan Pengembangan Sistem

Pada penelitian ini, metode pengembangan sistem yang digunakan adalah *Waterfall* yang merupakan salah satu metode dalam *System Development Live Cycle* (SDLC) yang mempunyai ciri khas pengerjaan setiap fase dalam *Waterfall* harus diselesaikan terlebih dahulu sebelum melanjutkan ke fase selanjutnya.

Metode waterfall kadang dinamakan siklus hidup klasik (*classic life cycle*), hal ini menyiratkan pendekatan yang sistematis dan berurutan (sekuensial) pada pengembangan perangkat lunak, yang dimulai dengan spesifikasi kebutuhan pengguna dan berlanjut melalui tahapan tahapan perencanaan (*planning*), pemodelan (*modeling*), kontruksi (*contruction*), serta penyerahan sistem/perangkat lunak ke para pelanggan/pengguna (*deployment*), yang

diakhiri dengan dukungan berkelanjutan pada perangkat lunak yang dihasilkan (Pressman, 2012). Tahapan Metode *Waterfall* dapat dilihat pada Gambar 2.



Gambar 2 Metode *Waterfall* (Pressman, 2012).

Tahapan-tahapan dari Metode *Waterfall* adalah sebagai berikut:

1. Tahap Komunikasi

Pada tahapan ini merupakan proses wawancara yang bertujuan untuk memahami kebutuhan dari pengembangan aplikasi yang akan dibuat serta batasan-batasannya. Selanjutnya Hasil dari wawancara tersebut direpresentasikan ke dalam *Use Case Diagram*.

1.1 Wawancara

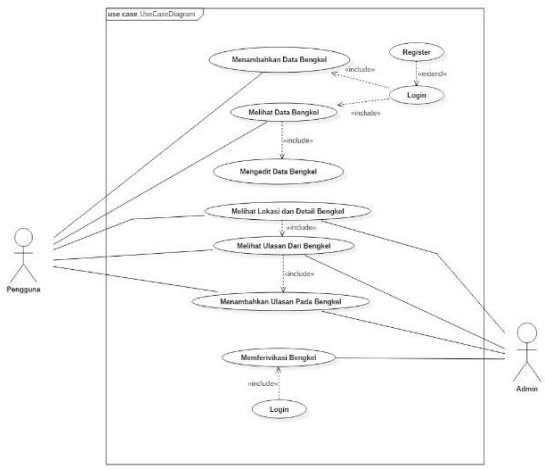
Wawancara ini melibatkan 5 responden. Daftar tabel hasil wawancara pada Aplikasi Bengkel Berbasis Android disajikan di Lampiran. Berikut adalah pertanyaan yang ditujukan untuk pengendara sepeda motor sebagai calon pengguna aplikasi:

a. Pertanyaan 1: Di mana biasanya anda memperbaiki sepeda motor anda?
b. Pertanyaan 2: Bagaimana jika kendaraan anda mengalami kerusakan saat dalam perjalanan?
c. Pertanyaan 3: Apakah anda pernah menggunakan aplikasi dengan layanan berbasis lokasi?
d. Pertanyaan 4: Fitur apakah yang bisa mempermudah anda dalam mencari bengkel terdekat pada aplikasi dengan layanan berbasis lokasi?

Dari wawancara yang telah dilakukan, dapat disimpulkan sistem yang akan dibangun yaitu sistem yang dapat menampilkan bengkel terdekat dengan informasi ulasan dan rute dari bengkel.

1.2 *Use Case Diagram*

Use Case Diagram adalah gambaran fungsionalitas dari suatu sistem, sehingga *customer* atau pengguna sistem paham dan mengerti mengenai kegunaan sistem yang akan dikembangkan.



Gambar 3 *Use Case Diagram* Aplikasi Bengkel.

Pada sistem ini pengguna dapat melihat lokasi, detail, ulasan dan menambah ulasan pada bengkel. Selain itu pengguna juga dapat menambahkan, melihat dan mengedit data bengkel jika pengguna sudah *login*. Sedangkan admin bertugas untuk memverifikasi bengkel yang telah ditambahkan oleh pengguna. Untuk memahami lebih lanjut *Use Case Diagram* sistem ini dapat dilihat pada Gambar 3.

2. Tahap Perencanaan

Tahap perencanaan menjelaskan tentang estimasi tugas teknis yang akan dilakukan dalam pengembangan sistem dan penjadwalan kerja yang akan dilaksanakan. Pada tahap perencanaan ini waktu pengerjaan sistem diperkirakan memakan waktu selama enam bulan dimulai dari bulan Maret sampai September tahun 2021 dengan tujuh tugas teknis yaitu membuat perencanaan, membuat *Activity Diagram*, membuat *Entity Relationship Diagram*, membuat desain antarmuka, membuat aplikasi, melakukan pengujian dan melakukan perbaikan aplikasi. Daftar tabel pada tahap perencanaan Aplikasi Bengkel Berbasis Android disajikan di Lampiran.

3. Tahap Pemodelan

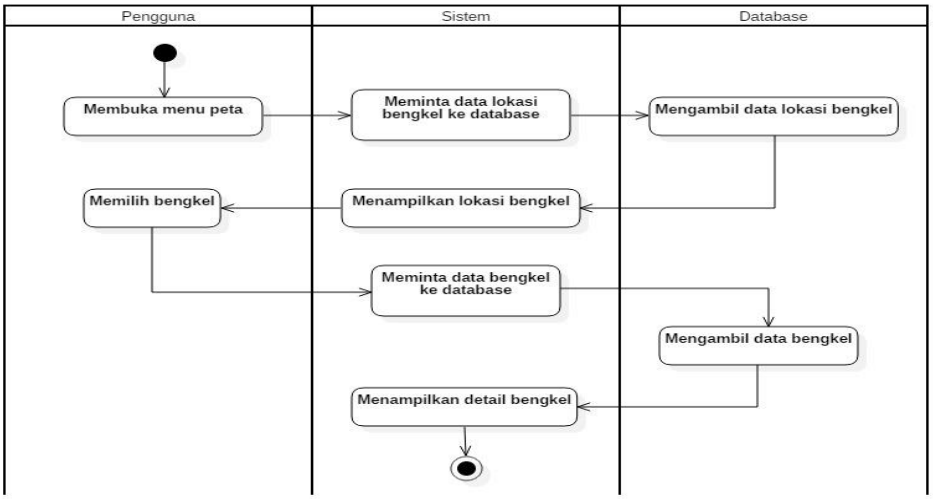
Tahap ini merupakan tahap pemodelan sistem yang akan dikembangkan yaitu pemodelan arsitektur sistem secara menyeluruh. Pada tahap pemodelan ini terdapat beberapa diagram yang digunakan untuk perancangan sistem yang akan dikembangkan yaitu *Activity Diagram*, *Entity Relationship Diagram*, dan desain *Interface* sistem. Berikut adalah desain dalam tahapan perancangan sistem yang akan dikembangkan.

3.1 *Activity Diagram*

Activity Diagram menggambarkan rangkaian aliran dan aktivitas, dan digunakan untuk mendeskripsikan aktivitas yang dibentuk dalam suatu operasi. Pembuatan *Activity Diagram* pada awal proses dapat membantu memahami keseluruhan proses.

3.1.1 *Activity Diagram* Melihat Lokasi Dan Detail Bengkel

Pada Gambar 4 menjelaskan proses yang dilakukan *user* untuk melihat lokasi dan detail bengkel.

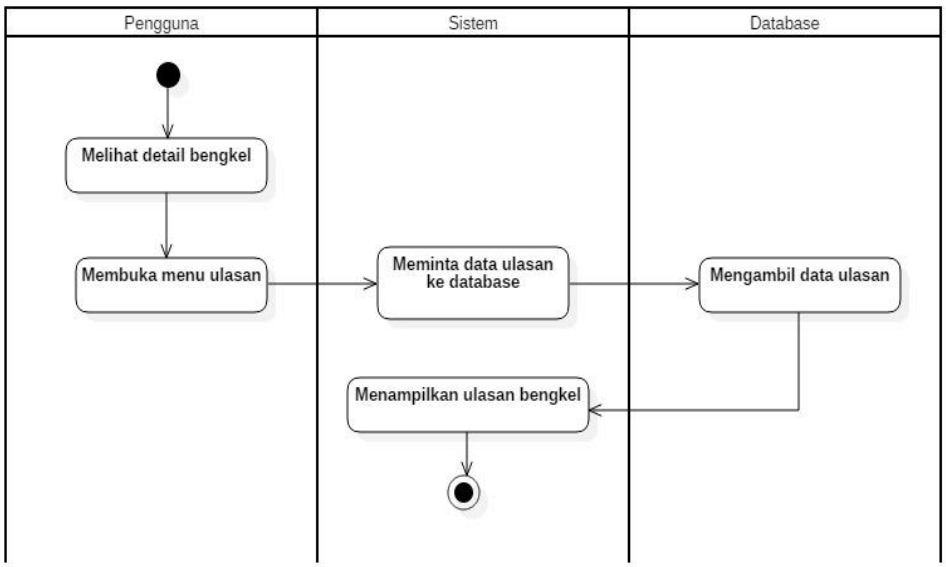


Gambar 4 *Activity Diagram* Melihat Lokasi Dan Detail Bengkel.

Dari halaman utama akan menampilkan menu peta dan sistem akan meminta data bengkel ke *database* untuk menampilkan lokasi bengkel pada peta. Pengguna dapat memilih bengkel yang tampil untuk melihat informasi lebih detail dari bengkel yang dipilih.

3.1.2 *Activity Diagram* Melihat Ulasan Dari Bengkel

Gambar 5 menjelaskan *Activity Diagram* untuk melihat ulasan dari bengkel.



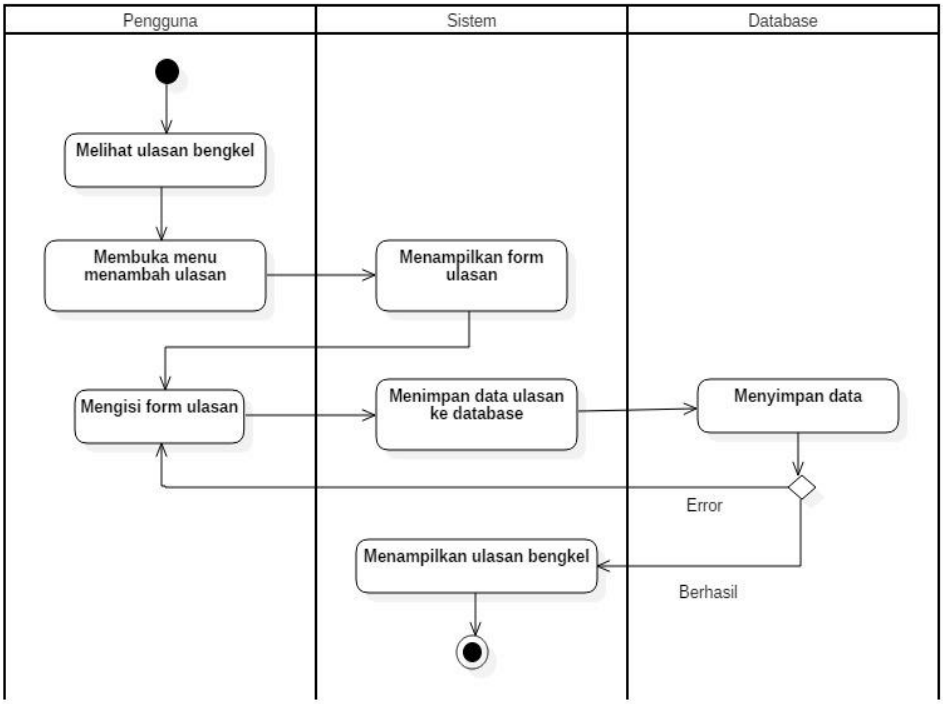
Gambar 5 *Activity Diagram* Melihat Ulasan Dari Bengkel.

Pengguna dapat membuka menu ulasan dari bengkel melalui menu detail bengkel yang dijelaskan pada Gambar 4. Setelah mengakses menu ulasan sistem akan meminta data ulasan ke *database* untuk ditampilkan pada pengguna.

3.1.3 *Activity Diagram* Menambah Ulasan Pada Bengkel

Pada Gambar 6 menjelaskan *Activity Diagram* untuk menambah ulasan pada bengkel. Pengguna dapat membuka menu tambah ulasan dari

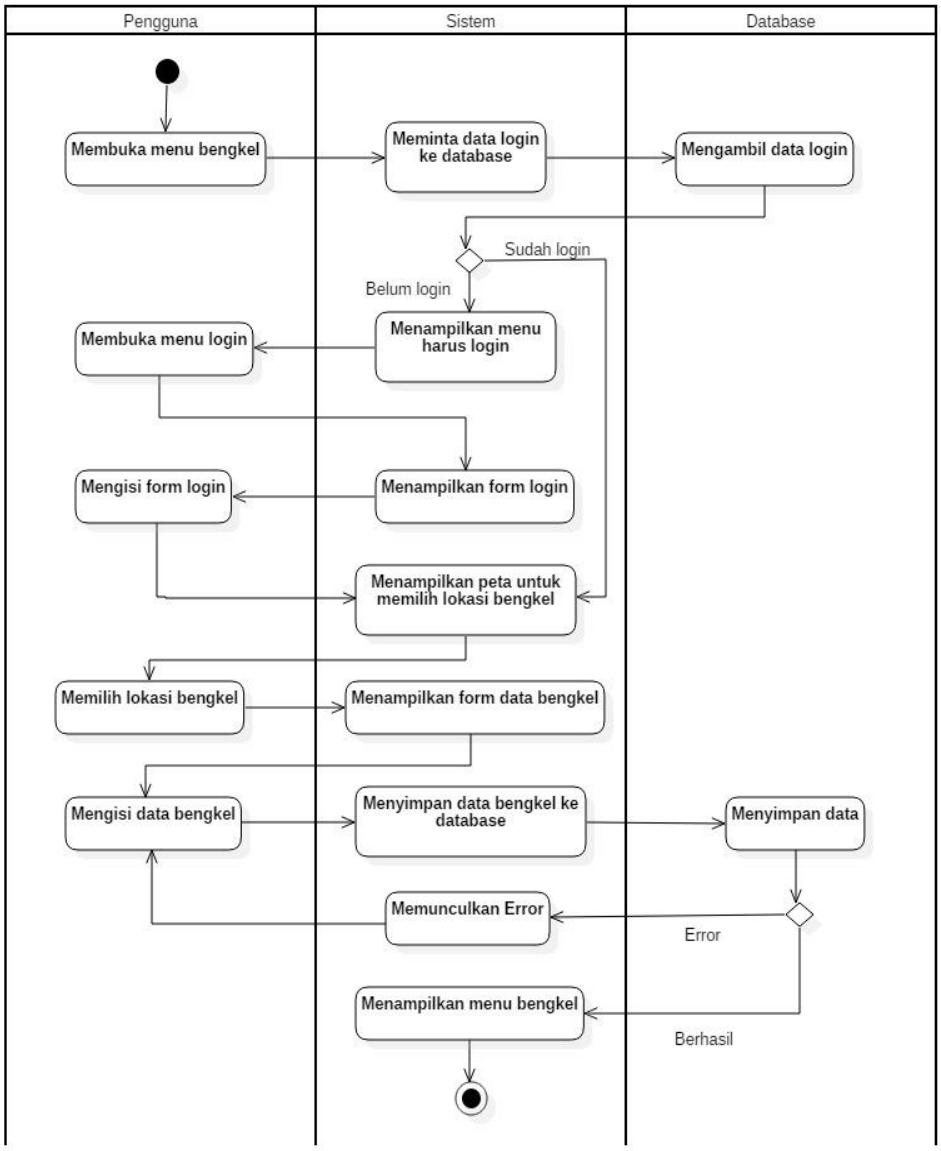
bengkel melalui menu ulasan yang dijelaskan pada Gambar 5. Setelah mengakses menu tambah ulasan sistem akan menampilkan *form* ulasan. Pengguna dapat mengisi ulasan dan sistem akan meminta *database* untuk menyimpan data yang telah diisi pengguna. Jika gagal pengguna akan diminta untuk memasukkan data ulasan kembali dan jika berhasil menyimpan data ke *database* maka sistem akan menampilkan menu ulasan bengkel.



Gambar 6 *Activity Diagram* Menambah Ulasan Pada Bengkel.

3.1.4 *Activity Diagram* Menambahkan Data Bengkel

Pada Gambar 7 menjelaskan *Activity Diagram* menambahkan data bengkel.



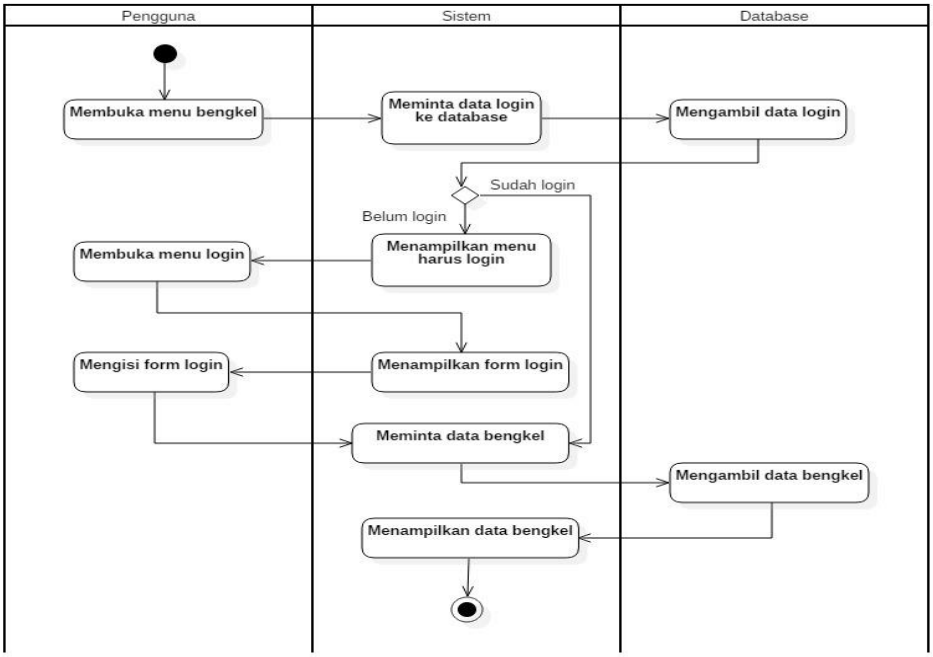
Gambar 7 *Activity Diagram* Menambahkan Data Bengkel.

Dari halaman utama pilih menu bengkel sistem akan meminta data *login* ke *database*. Jika pengguna sudah *login* maka sistem akan menampilkan peta. Dan jika pengguna belum *login* maka sistem akan menampilkan menu harus *login* sehingga pengguna dapat pindah ke menu *login*. Setelah mengakses menu *login* sistem akan menampilkan *form login*. Pengguna dapat mengisi data *login* dan sistem akan menampilkan peta. Pengguna dapat memilih lokasi bengkel kemudian mengisikan data bengkel dan

sistem akan meminta *database* untuk menyimpan data yang telah diisi pengguna. Jika gagal pengguna akan diminta untuk memasukkan data bengkel kembali dan jika berhasil menyimpan data ke *database* maka sistem akan menampilkan menu bengkel.

3.1.5 *Activity Diagram* Melihat Data Bengkel

Pada Gambar 8 menjelaskan *Activity Diagram* melihat data bengkel.



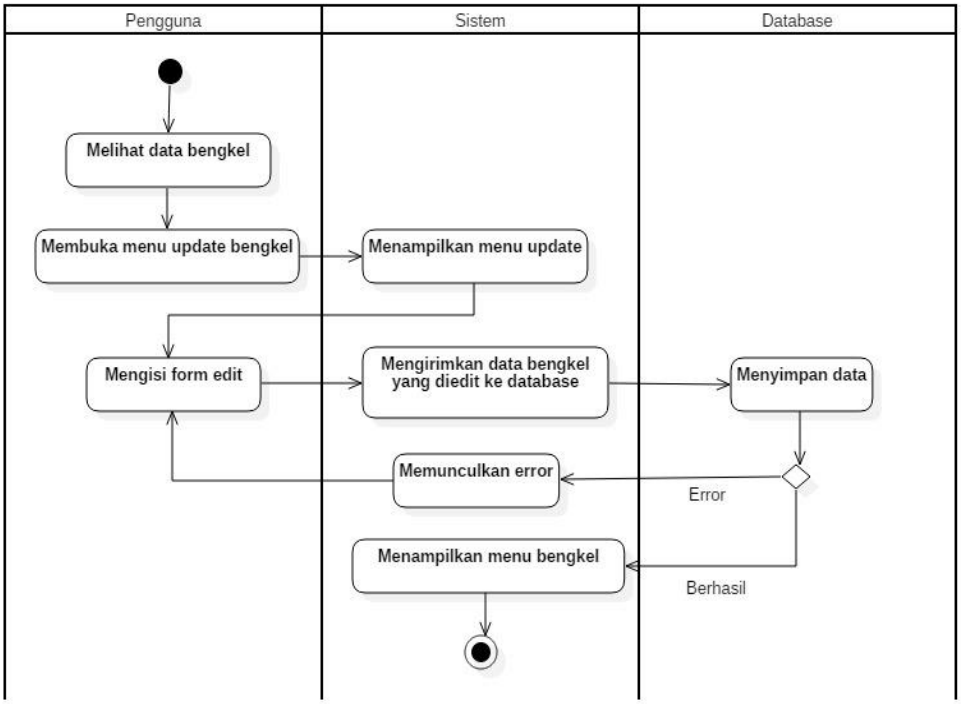
Gambar 8 *Activity Diagram* Melihat Data Bengkel.

Dari halaman utama pilih menu bengkel sistem akan meminta data *login* ke *database*. Jika pengguna sudah *login* maka sistem akan meminta data bengkel ke *database* untuk ditampilkan pada pengguna. Dan jika pengguna belum *login* maka sistem akan menampilkan menu harus *login*

sehingga pengguna dapat pindah ke menu *login*. Setelah mengakses menu *login* sistem akan menampilkan *form login*. Pengguna dapat mengisi data *login* dan sistem akan meminta data bengkel ke *database* untuk ditampilkan pada pengguna.

3.1.6 *Activity Diagram* Mengedit Data Bengkel

Pada Gambar 9 menjelaskan *Activity Diagram* untuk mengedit data bengkel.



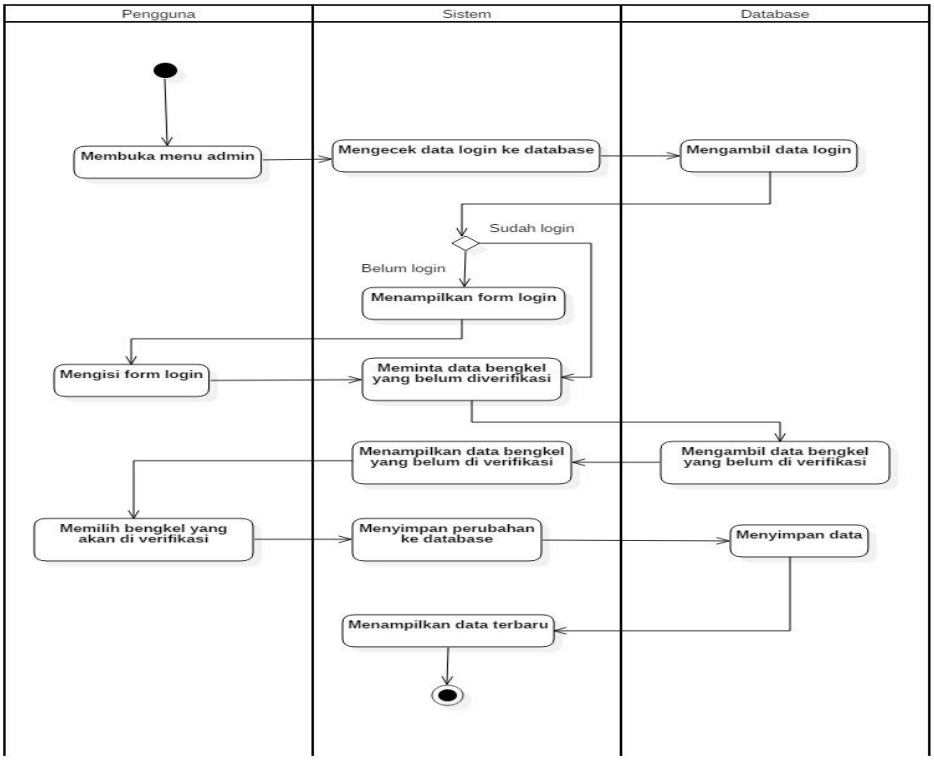
Gambar 9 *Activity Diagram* Mengedit Data Bengkel.

Pengguna dapat membuka menu edit data bengkel melalui menu data bengkel yang dijelaskan pada Gambar 8. Setelah mengakses menu edit

data bengkel sistem akan menampilkan *form* edit data bengkel. Pengguna dapat mengisi data bengkel dan sistem akan meminta *database* untuk menyimpan data yang telah diisi pengguna. Jika gagal pengguna akan diminta untuk memasukkan data ulasan kembali dan jika berhasil menyimpan data ke *database* maka sistem akan menampilkan menu bengkel.

3.1.7 *Activity Diagram* Memverifikasi Data Bengkel

Pada Gambar 10 menjelaskan *Activity Diagram* untuk memverifikasi data bengkel.

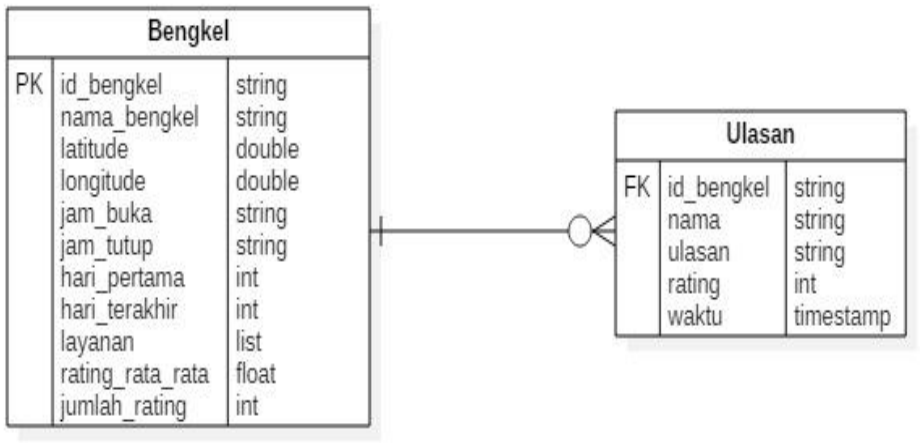


Gambar 10 *Activity Diagram* Memverifikasi Data Bengkel.

Pengguna dapat membuka menu admin melalui menu utama. Jika pengguna belum *login* aplikasi akan menampilkan form *login* tetapi jika pengguna sudah *login* maka aplikasi akan menampilkan data bengkel yang belum diverifikasi. Pengguna yang telah *login* akan berperan sebagai admin dan dapat melakukan verifikasi data bengkel.

3.2 *Entity Relationship Diagram*

Entity Relationship Diagram mendeskripsikan jenis-jenis objek dalam sistem dan berbagai macam hubungan statis yang terdapat di antara mereka. *Entity Relationship Diagram* Aplikasi Bengkel Berbasis Android ditampilkan pada Gambar 11.



Gambar 11 *Entity Relationship Diagram.*

3.3 Desain Antarmuka atau *Interface*

Rancangan *interface* pada Aplikasi Penjualan *Spare Part* dan Pencarian Bengkel di Wilayah Bandar Lampung adalah sebagai berikut:

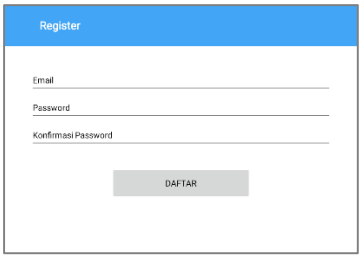
3.3.1 Rancangan *Interface* Menu *Login*

Perancangan *interface login* yang berfungsi sebagai menu untuk bengkel melakukan *login* akun dapat dilihat pada Gambar 12.

Gambar 12 Rancangan *Interface* Menu *Login*.

3.3.2 Rancangan *Interface* Menu *Register*

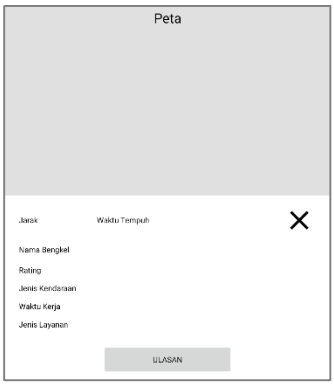
Perancangan *interface register* yang berfungsi sebagai menu untuk bengkel melakukan pendaftaran akun dapat dilihat pada Gambar 13.



Gambar 13 Rancangan *Interface* Menu *Register*.

3.3.3 Rancangan *Interface* Menu Lihat Lokasi Bengkel

Perancangan *interface* lihat bengkel yang berfungsi sebagai menu untuk pengguna melihat lokasi bengkel dapat dilihat pada Gambar 14.



Gambar 14 Rancangan *Interface* Menu Lihat Lokasi Bengkel.

3.3.4 Rancangan *Interface* Menu Melihat Ulasan Bengkel

Perancangan *interface* lihat ulasan bengkel yang berfungsi sebagai menu untuk pengguna melihat detail ulasan dari bengkel dapat dilihat pada Gambar 15.

Gambar 15 Rancangan *Interface* Lihat Ulasan Bengkel.

3.3.5 Rancangan *Interface* Menu Menambahkan Ulasan

Perancangan *interface* menambahkan ulasan yang berfungsi sebagai menu untuk pengguna menambahkan ulasan pada bengkel dapat dilihat pada Gambar 16.

Gambar 16 Rancangan *Interface* Menambahkan Ulasan.

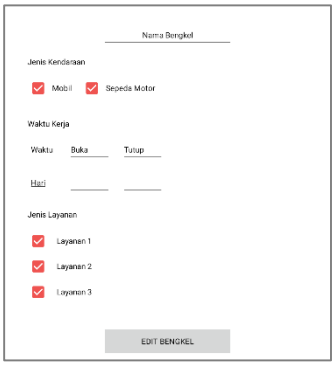
3.3.6 Rancangan *Interface* Menu Menambahkan Data Bengkel

Perancangan *interface* menu menambahkan data bengkel yang berfungsi sebagai menu untuk pengguna dapat menambahkan data bengkel dapat dilihat pada Gambar 17.

Gambar 17 Rancangan *Interface* Menu Menambahkan Data Bengkel.

3.3.7 Rancangan *Interface* Menu Mengelola Data Bengkel.

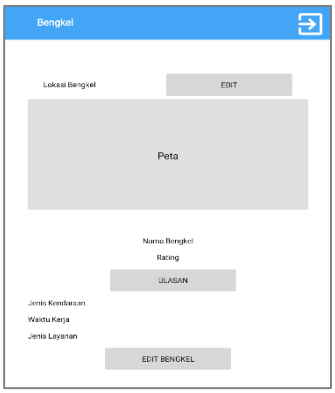
Perancangan *interface* menu mengelola data bengkel yang berfungsi sebagai menu pengguna melakukan pengelolaan data pada bengkel dapat dilihat pada Gambar 18.



Gambar 18 Rancangan *Interface* Menu Mengelola Data Bengkel.

3.3.8 Rancangan *Interface* Menu Bengkel

Perancangan *interface* menu bengkel yang berfungsi sebagai menu pengguna untuk melihat data bengkel yang sudah dibuat dapat dilihat pada Gambar 19.



Gambar 19 Rancangan *Interface* Kelola Akun Profil.

4. Tahap Kontruksi

Pada tahap kontruksi dilakukan implementasi dari segala yang telah direncanakan. Hasil dari pemodelan diimplementasikan dengan penulisan kode program yang menggunakan aplikasi Android Studio dengan bahasa pemrograman Kotlin. Selanjutnya dilakukan pengujian sistem yang telah dibuat untuk mengetahui apakah sistem berjalan sesuai dengan yang diharapkan atau tidak. Pada penelitian ini dilakukan pengujian fungsional dan pengujian *user acceptance.*

4.1 Pengujian Fungsional

Pada tahap ini dilakukan pengujian sistem untuk mengetahui apakah keseluruhan fungsi sistem berjalan dengan baik atau tidak. Adapun identifikasi pengujian yang akan dilakukan dapat dilihat pada Tabel 3 dan tahap pengujian pada Tabel 4 sampai Tabel 9.

Tabel 3 Identifikasi Pengujian.

Kelas Uji	**Butir Uji**	**Pengujian Aplikasi (PA)**	**Tingkat Pengujian**	**Jenis Pengujian**
Validasi	*Login*	**PA_01**	Pengujian Sistem	*Black-Box*
Pengelolaan Data	Menambahkan Data	**PA_02**	Pengujian Sistem	*Black-Box*
	Mengubah Data	**PA_03**	Pengujian Sistem	*Black-Box*

Tabel 3 (lanjutan)

Kelas Uji	**Butir Uji**	**Pengujian Aplikasi (PA)**	**Tingkat Pengujian**	**Jenis Pengujian**
Tampilan	Peta	**PA_04**	Pengujian Sistem	*Black-Box*
	Informasi Akun	**PA_05**	Pengujian Sistem	*Black-Box*
	Ulasan	**PA_06**	Pengujian Sistem	*Black-Box*

Tabel 4 *Login.*

Identi-fikasi	**Prosedur Pengujian**	**Masukkan**	**Hasil Yang Diharapkan**	**Kesimpulan**
PA_01	Aplikasi dapat menampilkan menu *login* untuk validasi bengkel	*Username* dan *password*	Berhasil masuk dan pindah ke menu selanjutnya	[ ] Berhasil [ ] Tidak Berhasil

Tabel 5 Menambahkan Data.

Identi-fikasi	**Prosedur Pengujian**	**Masukkan**	**Hasil Yang Diharapkan**	**Kesimpulan**
PA_02	Aplikasi dapat menambahkan data bengkel	Lokasi Bengkel, Nama Bengkel, Jenis Kendaraan, Waktu Kerja, Layanan Yang Disediakan	Data berhasil disimpan dapat mengakses menu bengkel	[ ] Berhasil [ ] Tidak Berhasil

Tabel 5 (lanjutan)

Identi-fikasi	Prosedur Pengujian	Masukkan	Hasil Yang Diharapkan	Kesimpulan
PA_02	Aplikasi dapat menambahkan ulasan pada bengkel	Penilaian, Nama, Ulasan	Data berhasil disimpan dan berhasil ditampilkan pada menu ulasan	[ ] Berhasil [ ] Tidak Berhasil

Tabel 6 Mengubah Data.

Identi-fikasi	Prosedur Pengujian	Masukkan	Hasil Yang Diharapkan	Kesimpulan
PA_03	Aplikasi dapat mengubah data bengkel	Lokasi Bengkel, Nama Bengkel, Jenis Kendaraan, Waktu Kerja, Layanan Yang Disediakan	Data berhasil diperbarui dan menampilkan data terbaru	[ ] Berhasil [ ] Tidak Berhasil

Tabel 7 Peta.

Identi-fikasi	Prosedur Pengujian	Masukkan	Hasil Yang Diharapkan	Kesimpulan
PA_04	Aplikasi dapat menampilkan data lokasi bengkel	-	Berhasil menampilkan data lokasi bengkel	[ ] Berhasil [ ] Tidak Berhasil

Tabel 7 (lanjutan)

Identi-fikasi	Prosedur Pengujian	Masukkan	Hasil Yang Diharapkan	Kesimpulan
PA_04	Aplikasi dapat menampilkan detail bengkel, rute, jarak dan waktu tempuh	-	Berhasil menampilkan detail bengkel, rute, jarak dan waktu tempuh	[ ] Berhasil [ ] Tidak Berhasil

Tabel 8 Informasi Akun.

Ident-fikasi	Prosedur Pengujian	Masukkan	Hasil Yang Diharapkan	Kesimpulan
PA_05	Aplikasi dapat menampilkan informasi bengkel	-	Dapat menampilkan informasi bengkel	[ ] Berhasil [ ] Tidak Berhasil

Tabel 9 Ulasan.

Ident-fikasi	Prosedur Pengujian	Masukkan	Hasil Yang Diharapkan	Kesimpulan
PA_06	Aplikasi dapat menampilkan ulasan bengkel	-	Dapat menampilkan ulasan bengkel	[ ] Berhasil [ ] Tidak Berhasil

4.2 Pengujian *User Acceptance*

Pada tahap ini dilakukan pengujian sistem untuk mengetahui apakah sistem sudah sesuai dengan kebutuhan pengguna. Pengujian *user*

acceptance ini dilakukan dengan menggunakan kuesioner yang memiliki 5 skala penilaian pada tiap pertanyaan yang ada.

5. Tahap *Deployment*

Tahap terakhir dari metode *waterfall* yaitu *deployment* atau tahap implementasi sistem yang telah dibuat kepada pengguna. Dalam tahap ini juga dilakukan evaluasi sistem dan pemeliharaan sistem yang telah dikembangkan.

V. SIMPULAN DAN SARAN

A. Simpulan

Berdasarkan hasil penelitian yang telah dilakukan, maka didapat kesimpulan sebagai berikut:

1. Aplikasi Bengkel telah berhasil dibuat untuk memudahkan pengguna kendaraan bermotor untuk menemukan bengkel terdekat.
2. Aplikasi Bengkel dapat menampilkan lokasi bengkel dan informasi detail dari bengkel yang dipilih seperti nama bengkel, jenis kendaraan, waktu kerja, layanan yang tersedia, *rating* dan ulasan. Selain itu aplikasi ini juga dapat menampilkan jarak, waktu tempuh dan rute dari pengguna ke lokasi bengkel.
3. Berdasarkan hasil pengujian *Black-Box Testing*, Aplikasi Bengkel berhasil mendapatkan hasil sesuai dengan yang diharapkan pada tiap kelas uji yang dilakukan.
4. Dari hasil data penilaian aplikasi menggunakan kuesioner, Aplikasi Bengkel mendapat persentase rata-rata *index* penilaian sebesar 88% dari total 10 responden, maka dapat disimpulkan hasil yang didapat masuk ke dalam kategori yang sangat baik.

B. Saran

Berdasarkan perancangan dan hasil implementasi sistem yang dilakukan, maka beberapa saran yang perlu diperhatikan dalam mengembangkan Aplikasi ini adalah sebagai berikut:

1. Menambahkan satu level *user* untuk pengguna yang ingin memberikan ulasan agar ulasan dapat dikelola dengan baik seperti diedit atau dihapus.
2. Memberikan pilihan rute lain jika tersedia pada sistem agar pengguna bisa memiliki pilihan rute.

DAFTAR PUSTAKA

DAFTAR PUSTAKA

Anwar, B., Jaya, H., dan Kusuma P. I. 2014. *Implementasi Location Based Service Berbasis Android Untuk Mengetahui Posisi User*. Medan: Jurnal SAINTIKOM. Vol. 13, No. 2.

Apriawan, M. R., Pribadi, I. A., Utami, Y. T. 2021. *Penerapan Aplikasi Pencarian Lokasi Bengkel di Kota Bandar Lampung (Studi Kasus Vespa)*. Jurnal Pepadun. Vol 2 No. 3.

Azwar, S. 2011. *Metode Penelitian*. Yogyakarta: Pustaka Pelajar.

Erika, O. R., Kurniawan, D., Febriansyah, F. E. 2016. *Aplikasi Pencarian Letak ATM Berbasis Android Dengan GIS Di Kota Bandar Lampung.* Jurnal Komputasi. Vol. 4, No. 1.

Juansyah, A. 2015. *Pembangunan Aplikasi Child Tracker Berbasis Assisted-Global Positioning System (A-GPS) Dengan Platform Android*. Jurnal Ilmiah Komputer dan Informatika (KOMPUTA) 1, Edisi 1 Vol 1 ISSN: 2089-9033.

Kastanakis, B. 2016. *Mapbox Cookbook.* Birmingham: Packt Publishing Ltd.

Meliani, B. P., Rosely, E., Hernawati, E. 2020. *Aplikasi Peminjaman Ruang Kelas Untuk Kuliah Pengganti Jadwal Reguler (Studi Kasus: Fakultas Ilmu Terapan)*. Vol.6, No.2 SSN: 2442-5826.

Munawar, 2005, *Pemodelan visual dengan UML, Graha ilmu*, Yogyakarta.

Munthe, R., Santosa, P. I., Ferdiana, R. 2015. *Usulan Metode Evaluasi User Acceptance Testing (UAT) dalam Pengembangan Perangkat Lunak*. ISSN 2087-2658.

Pressman, R. S. 2012. *Rekayasa Perangkat Lunak – Buku Satu, Pendekatan Praktisi (Edisi 7)*. Yogyakarta: Andi.

Pribadi, R. I. A., Pambudi, A., Ardiansyah. 2020. *eDonation Android Application for Used Goods Donation using Location-based Service*. Journal of Physics: Conference Series.

Putra, E. D. A., Ernawati., dan Coastera, F. F. 2016. *Penerapan Open Street Map Untuk Mencari Lokasi Atm Terdekat Dengan Algoritma Kruskal Berbasis*

Smartphone Android (Studi Kasus: Lokasi Atm Di Kota Bengkulu). Jurnal Rekursif, Vol. 4 No.2, ISSN 2303-0755.

Rushamidiwinata, M., Wartika. 2019. *Sistem Informasi Pelayanan Bengkel Pada Irian Motor*. Universitas Komputer Indonesia, Bandung

Sholiq, 2006. *Pemodelan Sistem Informasi Berorientasi Objek dangan UML*. Graha Ilmu, Yogyakarta.

Srinivas, A. K. 2018. *Mastering Firebase for Android Development*. Birmingham: Packt Publishing Ltd.

Wibowo, T. I., Setiyanto, N. A. 2012. *Aplikasi Mobile Informasi Pariwisata Kota Semarang Berbasis Android*. Techno.COM, Vol. 11, No. 3 124-133.

Yuwono, M. 2021. *Statistik Transportasi Darat 2020*. Jakarta: BPS RI/BPS-Statistics Indonesia

Zaki, B., Putra, S. D. 2018. *Aplikasi Bengkel Online Menggunakan Global Positioning System (GPS) Berbasis Android Pada CV. Rumah Otomotif*. LP3M STMIK Jayakarta, Vol.2 No.2 ISSN: 2597-3673