

**APLIKASI PEMESANAN SERVIS KENDARAAN PADA  
BENGKEL BERBASIS ANDROID MENGGUNAKAN  
METODE HAVERSINE FORMULA**

**TUGAS AKHIR**

**Karya tulis sebagai salah satu syarat  
untuk memperoleh gelar Sarjana dari  
Universitas Fajar**

**Oleh**

**MARHANDAM PALINDUNG**

**1720221086**



**TEKNIK ELEKTRO  
FAKULTAS TEKNIK  
UNIVERSITAS FAJAR**

**2022**

HALAMAN PENGESAHAN

APLIKASI PEMESANAN SERVICE KENDARAAN PADA BENGKEL BERBASIS  
ANDROID MENGGUNAKAN METODE HAVERSINE FORMULA

Oleh  
**MARHANDAM PALINDUNG**  
1720221086

Menyetujui  
Tim Pembimbing  
Tanggal 05 Maret 2022

Pembimbing I

Febriansyah, S.T., M.T.  
NIDN. 0921029003

Pembimbing II

Muh. Sakir, S.Kom., M.T.  
NIDN. 1010078304

Mengetahui

Dekan  
UNIFA  
UNIVERSITAS INDAH  
DEKAN FAKULTAS  
TEKNIK  
S.T., M.T.  
NIDN. 0906107701

Ketua Program Studi

Asmawaty Azis, S.T., M.T.  
NIDN. 0905058504

## PERNYATAAN ORISINALITAS

Penulis dengan ini menyatakan bahwa Tugas Akhir:

“Aplikasi Pemesanan Servis Kendaraan Pada Bengkel Berbasis Android Menggunakan Metode Haversine” adalah karya original saya dan setiap serta seluruh sumber acuan telah ditulis sesuai dengan Panduan Penulisan Ilmiah yang berlaku di Fakultas Teknik Universitas Fajar.

Makassar, 05 Maret 2022

Yang Menyatakan



MARHANDAM PALINDUNG

## ABSTRAK

**Aplikasi Pemesanan Servis Kendaraan Pada Bengkel Berbasis Android Menggunakan Metode Haversine, Marhandam Palindung.** Permasalahan pada kendaraan merupakan hal yang biasa terjadi pada saat berkendara, baik roda dua ataupun roda empat. Sebagai pengendara tentunya akan segera mencari bengkel terdekat. Maka dari itu dibuatlah sebuah aplikasi pemesanan servis kendaraan pada bengkel. Tujuan utama penelitian ini adalah mempermudah pengendara dalam mengakses bengkel serta dapat memilih atau memesan servis pada bengkel disekitar lokasi. Adapun salah satu metode untuk menyelesaikan permasalahan tersebut yaitu dengan menggunakan Metode Haversine. Berdasarkan hasil pengujian Metode Haversine Formula dapat menghitung jarak antara lokasi pengguna (pemesan servis) dan lokasi bengkel dan berdasarkan hasil jarak tersebut maka sistem pada aplikasi dapat menampilkan daftar bengkel disekitar lokasi pengguna.

Kata Kunci: Haversine Formula, Android, Bengkel, Kotlin, Servis

## ABSTRACT

**Vehicle Service Booking Application at Workshops Android Based Using Haversine Formula, Marhandam Palindung.** Problems with vehicles are common when driving, both two-wheeled and four-wheeled. As a driver, of course, you will immediately look for the nearest repair shop. Therefore, an application for ordering vehicle service was made at the workshop. This research aims to make it easier for motorists to access the workshop and can choose or order services at workshops around the location. One method to solve this problem is to use the Haversine method. Based on the test results, the Haversine Formula method can calculate the distance between the user's location (service order) and the workshop's location. Based on the results of the distance, the system in the application can display a list of workshops around the user's location.

**Keywords:** Haversine Formula, Android, Workshop, Kotlin, Service

## KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas berkat rahmat-Nyalah sehingga penulis dapat menyelesaikan proposal penelitian yang berjudul **“APLIKASI PEMESANAN SERVIS KENDARAAN PADA BENGKEL BERBASIS ANDROID MENGGUNAKAN METODE HAVERSINE FORMULA”** tepat pada waktunya.

Adapun tujuan dari penulisan proposal penelitian ini adalah untuk memenuhi sebagian persyaratan guna memperoleh gelar Sarjana Teknik Jurusan Teknik Elektro.

Pada kesempatan ini penulis hendak menyampaikan terima kasih kepada semua pihak yang telah memberikan dukungan moril maupun materil sehingga proposal penelitian ini dapat selesai.

Meskipun telah berusaha menyelesaikan proposal penelitian ini sebaik mungkin, penulis menyadari bahwa proposal penelitian ini masih ada kekurangan. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun dari para pembaca guna menyempurnakan segala kekurangan dalam penyusunan proposal penelitian ini.

Akhir kata, penulis berharap semoga proposal penelitian ini berguna bagi para pembaca dan pihak-pihak lain yang berkepentingan.

Makassar, 05 Maret 2022

Penulis

## DAFTAR ISI

<b>HALAMAN PENGESAHAN</b> .....	Error! Bookmark not defined.
<b>PERNYATAAN ORISINALITAS</b> .....	Error! Bookmark not defined.
<b>ABSTRAK</b> .....	iv
<b>ABSTRACT</b> .....	v
<b>KATA PENGANTAR</b> .....	vi
<b>DAFTAR ISI</b> .....	vii
<b>DAFTAR GAMBAR</b> .....	ix
<b>DAFTAR TABEL</b> .....	xii
<b>BAB I PENDAHULUAN</b> .....	1
<b>I.1 Latar Belakang Penelitian</b> .....	1
<b>I.2 Rumusan Masalah</b> .....	3
<b>I.3 Tujuan Penelitian</b> .....	3
<b>I.4 Batasan Masalah</b> .....	3
<b>I.5 Manfaat Penelitian</b> .....	3
<b>BAB II TINJAUAN PUSTAKA</b> .....	5
<b>II.1 Tinjauan Teori</b> .....	5
II.1.1 Pengembangan atau <i>Development</i> .....	5
II.1.2 Sistem.....	5
II.1.3 Aplikasi .....	6
II.1.4 Bengkel .....	7
II.1.5 Android .....	8
II.1.6 Android Studio.....	9
II.1.7 Kotlin .....	11
II.1.8 Google Maps .....	12
II.1.9 Firebase .....	13
II.1.10 Metode Haversine Formula.....	14
II.1.11 Laravel.....	17
II.1.12 Perancangan Sistem .....	18
II.1.13 Pengujian Sistem.....	19

<b>II.2 Penelitian Terdahulu (State Of The Art)</b> .....	20
<b>II.3 Kerangka Pemikiran</b> .....	23
<b>BAB III METODE PENELITIAN</b> .....	<b>25</b>
<b>III.1 Tahapan Penelitian</b> .....	25
<b>III.2 Rancangan Penelitian Sistem</b> .....	26
III.2.1 Sistem yang Sedang Berjalan .....	26
III.2.2 Sistem yang Direncanakan .....	27
III.2.2.1 Use Case Diagram .....	27
III.2.2.2 Activity Diagram.....	28
III.2.2.3 Sequence Diagram.....	34
III.2.2.4 Class Diagram .....	39
III.2.2.5 Desain Interface.....	40
<b>III.3 Waktu dan Tempat Penelitian</b> .....	41
<b>III.4 Alat dan Bahan Penelitian</b> .....	41
III.4.1 Perangkat Keras .....	41
III.4.2 Perangkat Lunak .....	41
<b>III.5 Metode Pengumpulan Data</b> .....	41
<b>III.6 Metode Analisis Data/Pengujian Sistem</b> .....	42
<b>III.7 Jadwal Penelitian</b> .....	43
<b>BAB IV PEMBAHASAN DAN PEMBAHASAN</b> .....	<b>44</b>
<b>IV.1 Hasil</b> .....	44
IV.1.1 <i>Interface</i> Website Database Aplikasi.....	44
IV.1.2 <i>Interface</i> Pada Aplikasi Android Dari Sisi Pengguna .....	47
IV.1.3 <i>Interface</i> Pada Aplikasi Android Dari Sisi Bengkel.....	56
<b>IV.2 Pembahasan</b> .....	61
<b>BAB V KESIMPULAN DAN SARAN</b> .....	<b>114</b>
<b>V.1 Kesimpulan</b> .....	114
<b>V.2 Saran</b> .....	114
<b>DAFTAR PUSTAKA</b> .....	<b>114</b>



## DAFTAR GAMBAR

Gambar II.1 Proses Pengerjaan Service .....	7
Gambar II.2 Logo Android .....	8
Gambar II.3 Logo Android Studio .....	10
Gambar II.4 Logo Koltin.....	11
Gambar II.5 Logo Google Maps .....	12
Gambar II.6 Logo Firebase .....	13
Gambar II.7 Segitiga Bola Diselesaikan Dengan Hukum Haversine Formula.....	17
Gambar II.8 Framework Laravel.....	17
Gambar II.9 Kerangka Pemikiran .....	24
Gambar III.1 Tahapan Penelitian.....	25
Gambar III.2 Sistem yang Sedang Berjalan.....	26
Gambar III.3 Use Case Diagram Admin.....	27
Gambar III.4 Use Case Diagram .....	27
Gambar III.5 Activity Diagram Diagram Login Admin .....	28
Gambar III.6 Activity Diagram Kelola Pengguna, Bengkel, dan Transaksi.....	29
Gambar III.7 Activity Diagram Registrasi User dan Bengkel .....	30
Gambar III.8 Activity Diagram Login User dan Bengkel .....	30
Gambar III.9 Activity Diagram Pemesanan Servis.....	31
Gambar III.10 Activity Diagram Chat .....	32
Gambar III.11 Activity Diagram User Mencari Bengkel Terdekat .....	32
Gambar III.12 Activity Diagram Bengkel Nota Layanan Servis.....	33
Gambar III.13 Activity Diagram Konfirmasi Pesanan .....	34
Gambar III.14 Sequence Diagram Login Admin .....	35
Gambar III.15 Sequence Diagram Admin Kelola User .....	35
Gambar III.16 Sequence Diagram Daftar Akun User dan Bengkel.....	36
Gambar III.17 Sequence Diagram Login Akun User dan Bengkel .....	36
Gambar III.18 Sequence Diagram Pemesanan Servis.....	37
Gambar III.19 Sequence Diagram User Cari Bengkel Terdekat.....	37
Gambar III.20 Sequence Diagram Konfirmasi Pesanan .....	38

Gambar III.21 Sequence Diagram Chat .....	38
Gambar III.22 Sequence Diagram Konfirmasi Tempat Service .....	39
Gambar III.23 Class Diagram .....	39
Gambar III.24 Desain Interface.....	40
Gambar IV.1 Tampilan Halaman Login Admin.....	44
Gambar IV.2 Tampilan Halaman Dashboard.....	44
Gambar IV.3 Tampilan Halaman Daftar User .....	45
Gambar IV.4 Tampilan Halaman Daftar Bengkel .....	45
Gambar IV.5 Tampilan Halaman Daftar Transaksi .....	46
Gambar IV.6 Tampilan Spalsh Screen Aplikasi Pengguna .....	47
Gambar IV.7 Tampilan Halaman Utama Aplikasi Pengguna .....	47
Gambar IV.8 Tampilan Halaman Profile Pengguna .....	48
Gambar IV.9 Tampilan Halaman Fitur Lokasi .....	49
Gambar IV.10 Tampilan Halaman Fitur Service .....	49
Gambar IV.11 Tampilan Halaman Confirm Location .....	50
Gambar IV.12 Halaman Daftar Bengkel.....	51
Gambar IV.13 Halaman Daftar Bengkel.....	51
Gambar IV.14 Halaman Daftar Service .....	52
Gambar IV.15 Halaman Daftar Pesanan (Keranjang).....	53
Gambar IV.16 Halaman Riwayat .....	53
Gambar IV.17 Halaman Detail Proses Pengguna (Tidak Mengetahui Permasalahan) .....	54
Gambar IV.18 Halaman Detail Proses Pengguna (Mengetahui Permasalahan) ...	55
Gambar IV.19 Tampilan Halaman Fitur Chat.....	55
Gambar IV.20 Tampilan Spalsh Screen Aplikasi Bengkel .....	56
Gambar IV.21 Tampilan Halaman Utama Aplikasi Bengkel .....	57
Gambar IV.22 Tampilan Halaman Profile Bengkel.....	57
Gambar IV.23 Tampilan Halaman Service .....	58
Gambar IV.24 Tampilan Halaman Service Order.....	58
Gambar IV.25 Tampilan Halaman Detail Confirm Bengkel .....	59
Gambar IV.26 Tampilan Halaman Detail Process Bengkel.....	60

Gambar IV.27 Tampilan Halaman Fitur Chat Bengkel .....	60
Gambar IV.28 Perintah Konfigurasi Firebase Admin SDK For PHP.....	83
Gambar IV.29 Script Integrasi Antara Website dan Firebase Realtime Database	84
Gambar IV.30 Deklarasi Variabel resultLocation .....	84
Gambar IV.31 Script Untuk Mendapatkan Lokasi Terakhir.....	84
Gambar IV.32 Script Mengambil Lokasi Pengguna .....	85
Gambar IV.33 Halaman Place Picker .....	85
Gambar IV.34 Script Implementasi Metode Haversine Formula .....	86
Gambar IV.35 Script Untuk Mengambil Semua Data Lokasi Bengkel .....	87
Gambar IV.36 Flowchart Login Pengguna, Bengkel, dan Admin.....	94
Gambar IV.37 Flowgraph Login Pengguna, Bengkel, dan Admin.....	95
Gambar IV.38 Daftar Akun Bengkel .....	97
Gambar IV.39 Flowgraph Daftar Akun Bengkel.....	97
Gambar IV.40 Flowchart Daftar Pengguna .....	99
Gambar IV.41 Flowgraph Daftar Pengguna .....	99
Gambar IV.42 Flowchart Titik Lokasi Bengkel Terdekat .....	101
Gambar IV.43 Flowgraph Titik Lokasi Bengkel Terdekat .....	101
Gambar IV.44 Flowchart Pemesanan Service di Fitur Service dan Darurat (Tidak Mengetahui Permasalahan) .....	103
Gambar IV.45 Flowgraph Pemesanan Service di Fitur Service dan Darurat (Tidak Mengetahui Permasalahan) .....	104
Gambar IV.46 Flowchart Pemesanan Service di Fitur Service dan Darurat (Mengetahui Permasalahan).....	107
Gambar IV.47 Flowgraph Pemesanan Service di Fitur Service dan Darurat (Mengetahui Permasalahan).....	108
Gambar IV.48 Flowchart Konfirmasi Pesanan Service .....	110
Gambar IV.49 Flowgraph Konfirmasi Pesanan Service Pada Sisi Bengkel .....	110

## DAFTAR TABEL

Tabel II.1 Penelitian Terdahulu.....	21
Tabel III.1 Jadwal Penelitian.....	43
Tabel IV.1 Pengujian Halaman Login Admin.....	61
Tabel IV.2 Pengujian Menu Navigation Sidebar Admin.....	62
Tabel IV.3 Pengujian Halaman Kelola Pengguna.....	63
Tabel IV.4 Pengujian Halaman Kelola Bengkel.....	63
Tabel IV.5 Pengujian Halaman Kelola Transaksi.....	64
Tabel IV.6 Pengujian Halaman Home Pengguna.....	64
Tabel IV.7 Pengujian Halaman Profile Pengguna.....	65
Tabel IV.8 Pengujian Halaman pada Fitur Lokasi Pengguna.....	66
Tabel IV.9 Pengujian Halaman Service dan Darurat Pada Android.....	67
Tabel IV.10 Pengujian Halaman Daftar Bengkel.....	68
Tabel IV.11 Pengujian Halaman Daftar Service.....	69
Tabel IV.12 Pengujian Halaman Keranjang Saya.....	69
Tabel IV.13 Pengujian Halaman Riwayat Pengguna.....	69
Tabel IV.14 Pengujian Halaman Detail Proses Pengguna.....	70
Tabel IV.15 Pengujian Halaman Chat Pengguna.....	71
Tabel IV.16 Pengujian Halaman Home Bengkel.....	72
Tabel IV.17 Pengujian Halaman Profile Bengkel.....	73
Tabel IV.18 Pengujian Halaman Service Bengkel.....	74
Tabel IV.19 Pengujian Halaman Tambah Service Dan Detail Service.....	75
Tabel IV.20 Pengujian Halaman Service Order Bengkel.....	75
Tabel IV.21 Pengujian Halaman Detail Confirm Bengkel.....	76
Tabel IV.22 Pengujian Halaman Detail Process Bengkel.....	77
Tabel IV.23 Pengujian Halaman Chat Bengkel.....	78
Tabel IV.24 Rekapitulasi Hasil Pengujian Black Box.....	78
Tabel IV.25 Hasil Pengujian Metode Haversine Secara Manual.....	92
Tabel IV.26 Hasil Pengujian Metode Haversine Menggunakan Website www.movable-type.co.uk.....	93

Tabel IV.27 Skenario Test Case Login Pengguna, Bengkel, dan Admin.....	96
Tabel IV.28 Skenario Test Case Daftar Akun Bengkel.....	98
Tabel IV.29 Skenario Test Case Daftar Pengguna.....	100
Tabel IV.30 Skenario Test Case Titik Lokasi Bengkel Terdekat.....	102
Tabel IV.31 Skenario Test Case Pemesanan Service di Fitur Service dan Darurat (Tidak Mengetahui Permasalahan).....	104
Tabel IV.32 Skenario Test Case Pemesanan Service di Fitur Service dan Darurat (Mengetahui Permasalahan).....	108
Tabel IV.33 Skenario Test Case Konfirmasi Pesanan Service Pada Sisi Bengkel.....	110
Tabel IV.34 Hasil Pengujian White Box.....	111



# **BAB I**

## **PENDAHULUAN**

### **I.1 Latar Belakang Penelitian**

Saat ini, perkembangan teknologi sangat pesat, dimana teknologi menjadi basis dalam segala aktivitas manusia, segala hal yang dulunya dilakukan secara manual sekarang dilakukan secara digital. Hal ini bisa dilihat dari jumlah pengguna smartphone aktif di dunia. Laporan menunjukkan bahwa pengguna ponsel pintar (*smartphone*) terbesar berada di Tiongkok pada 2020. Jumlahnya mencapai 911,9 juta pengguna. Posisi Tiongkok disusul oleh India dengan 439,4 juta pengguna smartphone. Setelahnya ada Amerika Serikat dengan 270 juta pengguna smartphone. Kemudian, Indonesia menempati posisi keempat dengan 160,23 juta pengguna smartphone. Penetrasi smartphone di dalam negeri telah mencapai 58,6% dari total populasi (Newzoo, 2021). Dari jumlah itu sebanyak 90% perangkat sudah dibekali dengan sistem GPS. Dengan adanya GPS, dapat mempermudah pengguna *smartphone* untuk melihat informasi lokasi terkini.

Jumlah kendaraan bermotor di Indonesia mencapai lebih dari 133 juta unit pada tahun 2019. Data itu terangkum dalam catatan Badan Pusat Statistik (BPS). Jumlah kendaraan naik sekitar lima persen sejak dua tahun lalu. Pada tahun 2019, jumlah kendaraan naik bertambah 7.108.236 unit atau meningkat 5,3 persen menjadi 133.617.012 unit. Mobil jenis penumpang (*passanger car*) menyumbang 11,6 persen dari total kendaraan di Indonesia. Jumlah mobil penumpang mencapai 15.592.419 unit pada tahun 2019. Jumlah ini naik dari jumlah di tahun 2018 sebanyak 14.830. Dengan mengacu pada data tersebut yaitu jumlah kendaraan yang terus mengalami kenaikan di Indonesia membuat peluang usaha bengkel semakin terbuka. Namun pada era digitalisasi ini dibutuhkan sebuah inovasi, apalagi dilihat dari sisi pengendara yang enggan membawa kendaraan mereka ke tempat bengkel karena faktor kesibukan, lamanya proses pengerjaan/service kendaraan, mengantri, dan mungkin faktor

lainnya adalah pandemi Covid-19 yang mengharuskan setiap orang untuk *stay at home* atau tetap tinggal di rumah untuk menghindari kerumunan agar terhindar dari Covid-19.

Dengan melihat beberapa penelitian yang serupa dengan pembuatan aplikasi yang berkaitan dengan bengkel yang dilakukan sebelumnya oleh A. Ferico Octaviansyah Pasaribu, Dedi Darwis, Agus Irawan, dan Ade Surahman yang berjudul “Sistem Informasi Geografis untuk Pencarian Lokasi Bengkel Mobil di Wilayah Kota Bandar Lampung” membahas suatu sistem informasi geografis untuk pencarian bengkel mobil terdekat dari titik lokasi pengguna di wilayah Bandar Lampung. Referensi selanjutnya yang berkaitan dengan metode Haversine Formula oleh Farid dan Yulanda Yunus yaitu “Analisa Algoritma Haversine Formula untuk Pencarian Lokasi Terdekat Rumah Sakit dan Puskesmas Provinsi Gorontalo” yang membahas suatu analisa algoritma Haversine Formula untuk lokasi pemetaan Rumah Sakit dan Puskesmas untuk pencarian lokasi terdekat ke rumah sakit satu ke rumah sakit lainnya dan puskesmas ke puskesmas lainnya.

Berdasarkan masalah diatas, maka dirancanglah sebuah aplikasi yaitu “Aplikasi Pemesanan Service Kendaraan Pada Bengkel Berbasis Android Menggunakan Metode Haversine Formula”. Aplikasi ini merupakan aplikasi dengan sistem order layanan otomotif dimana pihak bengkel/monitor yang akan datang ke titik lokasi kendaraan/rumah pengendara sehingga pengendara tidak perlu datang ke tempat bengkel untuk menyewa jasa montir dan menghindari kerumunan agar tidak terpapar Covid-19. Dengan bantuan GPS dari smartphone, pengendara dapat mengirimkan lokasi saat ini ke pihak bengkel/montir sehingga pihak bengkel/montir dapat datang ke lokasi dengan tepat waktu. Dengan adanya penelitian ini diharapkan membantu banyak orang khususnya bagi pengendara mobil maupun motor serta diharapkan menjadi sumber referensi untuk peneliti selanjutnya untuk mengembangkan aplikasi yang dibuat oleh peneliti sebelumnya.



## **I.2 Rumusan Masalah**

Berdasarkan latar belakang masalah yang telah diuraikan di atas, maka yang menjadi pokok permasalahannya adalah:

1. Bagaimana merancang dan mengembangkan aplikasi pemesanan *service* kendaraan pada bengkel berbasis Android?
2. Bagaimana merancang sistem pemetaan lokasi bengkel pada aplikasi pemesanan *service* kendaraan pada bengkel berbasis Android?
3. Bagaimana mengimplementasikan metode *Haversine Formula* pada aplikasi pemesanan *service* kendaraan berbasis Android?

## **I.3 Tujuan Penelitian**

Adapun tujuan dari penelitian ini yaitu:

1. Untuk merancang dan mengembangkan aplikasi pemesanan *service* kendaraan berbasis Android
2. Untuk merancang sistem pemetaan lokasi bengkel pada aplikasi pemesanan *service* kendaraan pada bengkel berbasis Android
3. Untuk mengimplementasikan metode *Haversine Formula* pada aplikasi pemesanan *service* kendaraan berbasis Android

## **I.4 Batasan Masalah**

Agar terfokus pada masalah yang dibahas maka batasan masalah yang diambil adalah sebagai berikut:

1. Aplikasi hanya berbentuk *platform* Android
2. Integrated Development Environment atau IDE yang digunakan adalah Android Studio dengan bahasa pemrograman menggunakan *Kotlin*
3. Aplikasi menggunakan Firebase sebagai *database*

## **I.5 Manfaat Penelitian**

Setiap penyusunan penelitian ini diharapkan dapat bermanfaat bagi perkembangan media informasi, Adapun manfaat yang diharapkan yaitu:

1. Manfaat bagi pihak kampus, diharapkan hasil peneliti ini memberikan referensi untuk melakukan penelitian berikutnya ataupun untuk bahan peajaran bagi institusi yang terkait
2. Bagi peneliti dapat menambah wawasan, pengetahuan dan kemampuan dalam mengembangkan aplikasi berbasis android
3. Meberikan kemudahan bagi pengendara dalam perawatan kendaraan

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **II.1 Tinjauan Teori**

##### **II.1.1 Pengembangan atau *Development***

Menurut Hasibuan (2011, p. 68) Pengembangan (*Development*) adalah fungsi operasional kedua dari manajemen Personalia, pengembangan pegawai perlu dilakukan secara terencana dan berkesinambungan agar pengembangan dapat dilaksanakan dengan baik, harus lebih dahulu ditetapkan suatu program pengembangan pegawai. (Hasibuan, 2011, p. 69) Dalam bukunya Manajemen Sumber Daya Manusia mengatakan bahwa Pengembangan adalah suatu usaha untuk meningkatkan kemampuan teknis, teoritis, Konseptual, dan Moral karyawan sesuai dengan kebutuhan pekerjaan/jabatan melalui pendidikan dan latihan. Sedangkan Menurut Siagian & Sondang (2012), menyatakan pengembangan (*development*) meliputi kesempatan belajar yang bertujuan untuk lebih meningkatkan pengetahuan (*knowledge*) dan keahlian (*skill*) yang diperlukan dalam pekerjaan yang sedang dijalani.

##### **II.1.2 Sistem**

Menurut Fatansyah (2015, p. 11) bahwa “Sistem adalah sebuah tatanan (keterpaduan) yang terdiri atas sejumlah komponen fungsional (dengan satuan fungsi dan tugas khusus) yang saling berhubungan dan secara bersama-sama bertujuan untuk memenuhi suatu proses tertentu”.

Secara garis besar jenis sistem dapat dibedakan menjadi dua kategori, yaitu:

1. Berdasarkan Keterbukaan
  - a. Sistem terbuka, yaitu suatu sistem yang dapat dipengaruhi oleh pihak luar karena adanya akses terbuka.

- b. Sistem tertutup, yaitu suatu sistem yang tidak dipengaruhi oleh pihak luar karena aksesnya tertutup.
2. Berdasarkan Komponen
    - a. Sistem fisik, yaitu suatu sistem yang memiliki komponen energi dan materi.
    - b. Sistem non-fisik, yaitu suatu sistem yang bentuknya abstrak, misalnya berupa ide, konsep, dan hal-hal lainnya.

### II.1.3 Aplikasi

Menurut Kadir (2008) program aplikasi adalah program siap pakai atau program yang direka untuk melaksanakan suatu fungsi bagi pengguna atau aplikasi yang lain. Aplikasi juga diartikan sebagai penggunaan atau penerapan suatu konsep yang menjadi pokok pembahasan atau sebagai program komputer yang dibuat untuk menolong manusia dalam melaksanakan tugas tertentu. Aplikasi *software* yang dirancang untuk penggunaan praktisi khusus, klasifikasi luas ini dapat dibagi menjadi 2 (dua) yaitu:

1. Aplikasi *software* spesialis, program dengan dokumentasi tergabung yang dirancang untuk menjalankan tugas tertentu.
2. Aplikasi paket, suatu program dengan dokumentasi tergabung yang dirancang untuk jenis masalah tertentu.

Dari kedua pengertian di atas dapat disimpulkan bahwa aplikasi adalah sekumpulan perintah atau kode yang disusun secara sistematis untuk menjalankan suatu perintah yang diberikan oleh manusia melalui komponen atau *hardware* komputer yang digunakan oleh manusia dalam menjalankan program aplikasi, dengan demikian bisa membantu manusia untuk memberikan solusi dari apa yang diinginkan.

#### II.1.4 Bengkel

Bengkel Adalah salah satu tempat dimana kendaraan seperti motor dan mobil ditempatkan dengan beberapa macam masalah, seperti kebutuhan , kerusakan serta perbaikan. Dimana bengkel adalah salah satu sarana paling tepat untuk menjadikan kendaraan bermotor seperti mobil dan motor ditempatkan.



Gambar II.1 Proses Pengerjaan Service

(Sumber: <https://www.catatanpagi.com/tips-sukses-menjalankan-usaha-bengkel-motor.html>)

Menurut Damanik (2019) “Bengkel merupakan suatu bangunan atau ruang untuk perawatan, pemeliharaan, perbaikan, penggantian peralatan dan mesin, tempat pembuatan bagian-bagian mesin, perkakas bengkel yang cukup sering diakses di setiap kehidupan”. Hal tersebut disebabkan oleh sifat alami barang-barang perlengkapan kehidupan yang selalu membutuhkan perawatan serta mengalami kerusakan dari waktu ke waktu. Termasuk Mesin pada kendaraan yang juga mengalami kerusakan seiring pemakaiannya, sehingga diperlukan perbaikan.

### II.1.5 Android

Android adalah sebuah *mobile platform* pertama yang lengkap dan gratis yang dikembangkan dengan menggunakan *Software Development Kit* (SDK) dengan *tools* yang cukup untuk mengembangkan aplikasi yang *powerful* dan kaya akan fitur (Jony, 2015). Android adalah sistem operasi yang dikeluarkan oleh Google. Android dibuat khusus untuk *smartphone* dan *tablet*. Berbagai macam produsen telah menggunakan Android sebagai sistem operasi untuk peranti (*device*) yang mereka produksi. Android juga mempunyai *store* dengan lebih dari 3 miliar pengguna aktif.



Gambar II.2 Logo Android

(Sumber: <https://mediaformasi.com/2019/08/logo-baru-android/>)

Kelebihan Android:

1. Merupakan sistem operasi *open source*, sehingga mudah dikembangkan oleh *developer*.
2. Mudah dikustomisasi dan dimodifikasi.
3. Android dapat dijalankan pada banyak pilihan spesifikasi *hardware*.
4. Dukungan aplikasi yang sangat banyak dan beragam.
5. Mudah dipahami, sehingga dapat diperbaiki apabila mengalami kerusakan sistem.
6. Dapat diaplikasikan di banyak peralatan elektronik.
7. Dikembangkan oleh salah satu raksasa telekomunikasi dunia, yaitu Google.
8. Merupakan salah satu sistem operasi yang cepat dan *responsive*.

Kekurangan Android:

1. Memiliki proses kerja sistem yang cukup berat, sehingga memakan RAM cukup banyak.
2. Terkadang apabila disandingkan dengan spesifikasi *hardware* yang buruk, menjadi kurang *responsive*.
3. Sistem operasi yang di custom sering tidak stabil dan kurang optimal.
4. Sistem operasi Android tampaknya menuntut pengguna untuk harus memiliki koneksi internet dalam keadaan aktif. Seperti minimalnya perlu koneksi internet GPRS.
5. Memang terdapat banyak Aplikasi Android yang dapat digunakan secara gratis, akan tetapi seringkali pada aplikasi yang digunakan akan memunculkan iklan.
6. Baterai pada smartphone dengan sistem Android akan sangat boros dibandingkan OS lainnya.

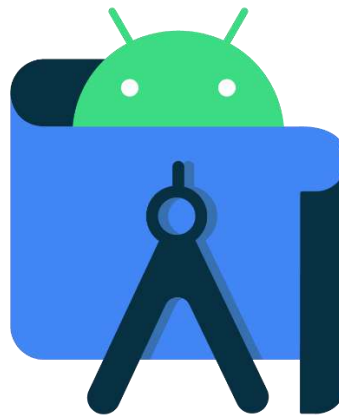
Pada tahun 2013, Android menjadi *operation system* (OS) terlaris pada *tablet* dan *smartphone*. Kini *market share* Android sedikitnya 70% dari total penjualan *smartphone* di tingkat global (statista.com). Tercatat pada tahun 2016 *Android store* memiliki lebih dari 2.8 juta aplikasi. Android juga menarik bagi perusahaan teknologi yang membutuhkan barang siap jadi, biaya rendah dan kustomisasi OS untuk perangkat teknologi tinggi mereka. Hal ini menjadi daya tarik bagi banyak perusahaan, sehingga mereka memilih Android.

*Source code* dari Android bersifat *open source*. Ini adalah hal menarik bagi komunitas *developer*, karena lisensi *open source* sangat mendukung untuk mengembangkan produknya dengan aman.

### **II.1.6 Android Studio**

Berbicara tentang pemrograman tentunya tak lepas dari *Integrated Development Environment* (IDE). Pada 2014 Google mengeluarkan IDE yang bernama Android Studio yang berbasiskan

IntelliJ IDEA. Menurut (Arifin, Marisa, & Wijaya, 2016) Android studio adalah sebuah IDE untuk Android Development yang diperkenalkan Google pada acara Google I/O 2013. Android Studio merupakan pengembangan dari Eclipse serta IDE resmi untuk pengembangan aplikasi Android.



Gambar II.3 Logo Android Studio

(Sumber: <https://android-developers.googleblog.com/2020/10/android-studio-41.html>)

Android Studio adalah *Integrated Development Environment* (IDE) resmi untuk pengembangan aplikasi Android, yang didasarkan pada IntelliJ IDEA. Selain sebagai *editor code* dan fitur *developer* IntelliJ yang andal, Android Studio menawarkan banyak fitur yang meningkatkan produktivitas Anda dalam membuat aplikasi Android, dengan menggunakan Android Studio, para *developer* dapat membuat aplikasi dari nol hingga dipublikasikan ke dalam *store*. Android Studio juga mempunyai beberapa fitur *built-in* yang sangat membantu para *developer* untuk memaksimalkan proses pembuatan aplikasi. Fitur-fitur ini misalnya Gradle, Code Completion, dan beragam integrasi dengan layanan dari Google, seperti Firebase.



### II.1.7 Kotlin

Pada tahun 2010 lalu JetBrains memulai sebuah proyek *open-sources* baru. Proyek ini merupakan sebuah bahasa pemrograman *statically typed* yang menargetkan JVM, Android, JavaScript dan Native. Kotlin adalah nama sebuah pulau di Rusia menjadi inspirasi oleh tim pengembang untuk ditetapkan sebagai nama bahasa pemrograman tersebut. Mungkin karena pulau tersebut terletak tidak jauh dari markas tim programmer JetBrains, yaitu Saint Petersburg, Rusia. Bahasa Kotlin pertama kali dirilis pada bulan Februari 2016 dengan versi 1.0. Saat ini sudah mencapai versi 1.5.21 per rilis 14 Juli 2021.



Gambar II.4 Logo Kotlin

(Sumber: [https://id.wikipedia.org/wiki/Kotlin\\_\(bahasa\\_pemrograman\)](https://id.wikipedia.org/wiki/Kotlin_(bahasa_pemrograman)))

Di balik kemajuan pesatnya Kotlin, terdapat pengembang-pengembang hebat dan kreatif dari JetBrains. Kotlin dikembangkan oleh lebih dari 50 developer pimpinan Andrey Breslav. Mereka semua terinspirasi dari bahasa pemrograman yang sudah ada seperti Java, Scala, JavaScript, C# dan juga Groovy. Yang membedakannya salah satunya, JetBrains memastikan bahwa Kotlin sangat mudah dipelajari. Kotlin dikembangkan di bawah lisensi Apache 2.0 dan kode sumbernya bisa Anda akses di laman GitHub-nya. Kita pun bisa berkontribusi dengan mengirimkan *pull request* ke repository-nya.

Kotlin dapat digunakan untuk berbagai macam pengembangan aplikasi, baik itu *server* atau *backend*, *website*, maupun mobile Android. Bahkan saat ini tengah dikembangkan Kotlin/Native. Kotlin/Native memungkinkan developer untuk menggunakannya

sebagai bahasa pemrograman dalam pengembangan aplikasi di platform lain seperti embedded system, desktop, macOS, dan iOS. Bahkan tak menutup kemungkinan Kotlin juga bisa digunakan untuk data science dan machine learning. Dukungan tools untuk Kotlin, sangat kuat. Kita bisa dengan mudah menggunakan Kotlin pada IDE seperti IntelliJ IDEA, Android Studio, Eclipse, dan NetBeans.

### II.1.8 Google Maps

Google Maps adalah layanan pemetaan web yang dikembangkan Google. Dalam Google Maps, Anda dapat melihat citra satelit, foto udara, peta jalan, pemandangan jalan panorama interaktif 360° (*Street View*), kondisi lalu lintas dalam waktu nyata, dan perencanaan rute untuk bepergian dengan berjalan kaki, mobil, sepeda, udara (dalam versi beta) dan transportasi umum. Pada tahun 2020, Google Maps telah digunakan oleh lebih dari 1 miliar orang setiap bulannya. Layanan Google Maps dapat diakses secara gratis dari Google. Peta ini juga dapat diakses melalui *browser web* atau sebagai aplikasi untuk perangkat seluler. Google Maps juga digunakan untuk mendapatkan petunjuk arah langkah demi langkah, menemukan informasi tentang bisnis lokal, dan masih banyak lagi.



## Google Maps

Gambar II.5 Logo Google Maps

(Sumber: [https://id.wikipedia.org/wiki/Google\\_Maps](https://id.wikipedia.org/wiki/Google_Maps))

### II.1.9 Firebase

Firestore merupakan *platform* untuk aplikasi *realtime*. Ketika data berubah, maka aplikasi yang terhubung dengan firestore akan meng-update secara langsung melalui setiap *device* (perangkat) baik website ataupun mobile (Sanadi, Achmad, & Dewiani, 2018). Firestore adalah suatu layanan dari Google untuk memberikan kemudahan bahkan mempermudah para *developer* aplikasi dalam mengembangkan aplikasinya. Firestore alias BaaS (*Backend as a Service*) merupakan solusi yang ditawarkan oleh Google untuk mempercepat pekerjaan *developer*.



Gambar II.6 Logo Firebase

(Sumber: <https://www.dicoding.com/blog/apa-itu-firebase-pengertian-jenis-jenis-dan-fungsi-kegunaannya/>)

Firestore didirikan pertama kali pada tahun 2011 oleh Andrew Lee dan James Tamplin. Produk Firestore yang pertama kali adalah Realtime Database. Realtime Database digunakan developer untuk menyimpan data dan synchronize ke banyak *user*. Kemudian ia berkembang sebagai layanan pengembang aplikasi. Pada bulan Oktober 2014, perusahaan tersebut diakuisisi oleh Google. Di bawah naungan Google fitur-fitur pada Firestore diperbanyak, berikut fitur-fitur pada Firestore:

#### 1. Firebase Analytics

Analytics merupakan inti dari Firebase sendiri yang diberikan secara gratis dan penuh. Kita dapat mengamati tingkah laku pengguna dalam penggunaan aplikasi oleh user dan menilai atribusi dalam satu Dashboard

#### 2. Firebase Cloud Messaging and Notifications

Firebase Cloud Messaging (FCM) dapat mengirim dan menerima pesan lintas platform seperti Android, iOS dan web.

#### 3. Firebase Authentication

Firebase Auth merupakan layanan sistem otentikasi yang menerapkan client-side code, sehingga user dapat mendaftar/login ke aplikasi. Firebase Auth mendukung email & password, Facebook, Twitter, GitHub, dan Google Sign-In.

#### 4. Firebase Cloud Firestore

Cloud Firestore adalah database yang bersifat fleksibel yang membuat datamu tetap terkoneksi di aplikasi user melalui listener realtime dan menawarkan layanan secara offline untuk aplikasi seluler dan web.

#### 5. Firebase Realtime Database

Firebase Realtime Database adalah database yang di-host melalui cloud. Data disimpan dan dieksekusi dalam bentuk JSON dan disinkronkan secara realtime ke setiap user yang terkoneksi.

#### 6. Firebase Hosting

Firebase Hosting merupakan suatu layanan hosting konten web. Hanya dengan satu instruksi, dapat mengimplementasikan aplikasi web serta menyajikan konten statis maupun dinamis ke CDN (jaringan penayangan konten) global dengan cepat.

### **II.1.10 Metode Haversine Formula**

Menurut Farid & Yunus (2017) Algoritma Haversine Formula adalah metode untuk menghitung jarak antara dua titik di bumi

berdasarkan panjang garis lurus antar dua titik tanpa mengabaikan kelengkungan yang dimiliki bumi.

Formula ini pertama kali ditemukan oleh James Andrew di tahun 1805, dan digunakan pertama kali oleh Josef de Mendoza y Ríos di tahun 1801. Istilah haversine ini sendiri diciptakan pada tahun 1835 oleh Prof. James Inman. Josef de Mendoza y Rios menggunakan haversine pertama kali dalam penelitiannya tentang “Masalah Utama Astronomi Nautical”, Proc. Royal Soc, Dec 22. 1796. Haversine digunakan untuk menemukan jarak antar bintang.

Formula Haversine adalah persamaan yang digunakan dalam navigasi, yang memberikan jarak lingkaran besar antara dua titik pada permukaan bola (bumi) berdasarkan bujur dan lintang. Formula Haversine merupakan suatu metode untuk mengetahui jarak antar dua titik dengan memperhitungkan bahwa bumi bukanlah sebuah bidang datar namun adalah sebuah bidang yang memiliki derajat kelengkungan

Haversine Formula menerapkan konsep trigonometri yang merupakan bagian dari geometri. Rumus Haversine ialah persamaan yang penting dalam navigasi, yang menghasilkan jarak lingkaran besar antara dua titik (latitude dan longitude) pada permukaan bola (bumi) berdasarkan bujur dan lintang. Penggunaan rumus ini cukup akurat untuk sebagian besar perhitungan, juga mengabaikan ketinggian bukit dan kedalaman lembah di permukaan bumi. Berikut bentuk Rumus Haversine Formula:

$$\Delta lat = lat2 - lat1$$

$$\Delta lng = long2 - long1$$

$$a = \sin^2 \frac{\Delta lat}{2} + \cos(lat1) \times \cos(lat2) \times \sin^2 \frac{\Delta lng}{2}$$

$$c = 2 \times \text{atan2}(\sqrt{a}, \sqrt{1-a})$$

$$d = R \times c$$

Dimana:

R = jari-jari bumi sebesar 6371(km)

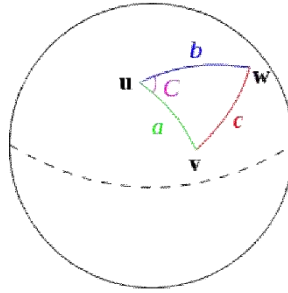
$\Delta lat$  = besaran perubahan latitude

$\Delta long$  = besaran perubahan longitude

C = kalkulasi perpotongan sumbu

d = jarak (km)

Hukum Haversine adalah semua persamaan yang digunakan berdasarkan bentuk bumi yang bulat (spherical earth) dengan menghilangkan faktor bahwa bumi itu sedikit elips (elipsoidal factor). Ini merupakan kasus khusus dari formula umum dalam trigonometri bola, hukum haversines, yang berkaitan dengan sisi dan sudut segitiga bola. Dalam unit bola, sebuah "segitiga" pada permukaan bola didefinisikan sebagai lingkaran-lingkaran besar yang menghubungkan tiga poin u, v, dan w pada bola. Jika panjang dari ketiga sisi adalah (dari u ke v), b (dari u ke w), dan c (dari v ke w), dan sudut sudut yang berlawanan c adalah C. maka hukum haversines menjadi:  $\text{Haversine}(c) = \text{haversine}(a-b) + \sin(a) \sin(b) \text{haversine}(C)$ .



Gambar II.7 Segitiga Bola Diselesaikan Dengan Hukum Haversine Formula

### II.1.11 Laravel

Laravel adalah web *framework* PHP yang bersifat *open source* dan gratis yang dibuat oleh Taylor Otweel yang dapat digunakan dalam mengembangkan web *applications* dengan menggunakan arsitektur MVC (*Model View Controller*). *Framework* Laravel mudah dipahami dan memudahkan dalam hal *authentication*, *routing*, *session manager*, *caching*, dan beberapa kegunaan lain dari komponen-komponen di Laravel. Laravel juga menyediakan fitur seperti *database migration* dan integrasi unit testing support yang memudahkan developer untuk membangun aplikasi yang kompleks (Chen et al, 2017).



Gambar II.8 Framework Laravel

(Sumber: <https://laravel-news.com/uploading-transforming-and-managing-media-in-laravel>)

## II.1.12 Perancangan Sistem

Perancangan sistem yang digunakan untuk penelitian ini ialah uml, ada beberapa perancangan sistem yaitu:

### 1. *Unified Modeling Language (UML)*

Menurut Rahman Abdillah, dkk (2019) *Unified Modeling Language* atau lebih sering dikenal dengan sebutan UML, adalah salah satu metode dalam teknik rekayasa perangkat lunak yang digunakan untuk menggambarkan alur dan cara kerja sistem, fungsi, tujuan dan mekanisme kontrol sistem tersebut. Dalam teknik rekayasa perangkat lunak bidang analisis dan perancangan sistem informasi, saat ini lebih banyak menggunakan gabungan dari konsep pemrograman berorientasi objek dengan teknik pembuatan perangkat lunak, dimana suatu sistem dilihat sebagai objek tersendiri yang sudah mencakup data dan proses atau dapat bekerja secara mandiri dalam satu set sistem (package). Dalam teknik perancangan sistem informasi, terdapat 4 model UML yang paling efektif penggunaannya untuk menggambarkan desain sistem (Dennis et al. 2012), yaitu:

#### a. *Use Case Diagram*

*Use Case diagram* digunakan untuk mengkomunikasikan interaksi manusia (actor) dengan apa yang bisa dilakukan oleh sistem. Sebuah Use Case dapat mewakili beberapa jalur interaksi manusia dengan sistem dan setiap jalur disebut sebagai skenario.

#### b. *Class Diagram*

*Class Diagram* adalah model statis yang mendukung tampilan data dan informasi dari keseluruhan sistem. Penggunaan Class Diagram dikaitkan dengan struktur basis data sistem atau dapat menggantikan ERD pada proses penggambaran diagram rekayasa perangkat lunak yang konvensional.

#### c. *Sequence Diagram*



*Sequence* diagram mengilustrasikan objek-objek yang terdapat pada Use Case dan menggambarkan arus pesan antara satu sama lain pada Objek Use Case. Sequence Diagram bersifat dinamis dan lebih banyak menampilkan aktifitas objek berdasarkan urutan waktu.

d. *Behavioral State Machine Diagram*

Diagram ini digunakan untuk menunjukkan sisi dinamis dari sebuah sistem. Dalam keilmuan pemrograman berbasis Objek, diagram ini tidak digunakan sebagai pelengkap Class, namun lebih banyak digunakan untuk membantu menjelaskan algoritma sistem dari sisi Method.

2. *Data Flow Diagram (DFD)*

*Data Flow Diagram (DFD)* merupakan alat bantu yang digunakan untuk menggambarkan sistem secara lengkap dan jelas, baik sistem yang sudah ada maupun sistem yang masih dalam rancangan. *Data Flow Diagram (DFD)* ini menjelaskan mengenai aliran data, informasi proses, basis data dan sumber tujuan data yang dilakukan oleh sistem. Tingkatan atau level *Data Flow Diagram (DFD)* dimulai dari diagram Konteks, yaitu menjelaskan dan menggambarkan mengenai sistem secara umum yang terdiri dari beberapa eksternal entity (elemen-elemen di luar sistem) yang memberikan input ke dalam sistem. Diagram Konteks tersebut akan diuraikan lagi ke dalam beberapa level diagram yang ada dalam sistem sehingga menghasilkan uraian sistem yang lebih terinci.

### **II.1.13 Pengujian Sistem**

Pengujian sistem yang sering digunakan untuk menguji dari kerja suatu sistem terdapat beberapa yaitu :

1. *White Box*

Menurut Pressman (2010), *white-box testing*, atau yang disebut *glass-box testing*, adalah metode perancangan *test case* yang menggunakan penjelasan struktur kontrol sebagai bagian dari *component-level design* untuk membuat *test cases*. Dengan menggunakan metode *white-box testing*, *software engineer* dapat membuat *test case* yang menjamin semua jalur independen di dalam modul telah dieksekusi sekurangnya sekali, menguji semua keputusan logikal pada nilai *true* dan *false*, menjalankan semua perulangan pada batasannya dan dalam batas operasionalnya, dan menguji struktur data internal untuk memastikan kebenarannya.

## 2. *Black Box*

Menurut Myers, dkk (2011), *black box testing* merupakan salah satu strategi pengujian yang dikenal juga sebagai *data driven testing* atau input/output testing. Untuk menggunakan metode ini, program dipandang sebagai kotak hitam. Tujuannya adalah untuk mengabaikan internal *behavior* dan struktur program. Sehingga penguji dapat berkonsentrasi menemukan ketiadaan saat program tidak berjalan sesuai spesifikasi nya. Pada pendekatan ini, kasus data hanya dibuat berdasarkan spesifikasi nya tanpa perlu mengetahui struktur internal program.

## II.2 Penelitian Terdahulu (State Of The Art)

Penelitian terdahulu bertujuan untuk mendapatkan bahan perbandingan dan acuan. selain itu, untuk menghindari anggapan kesamaan dengan penelitian ini. Maka dalam kajian pustaka ini peneliti mencantumkan hasil-hasil penelitian terdahulu sebagai berikut:

Tabel II.1 Penelitian Terdahulu

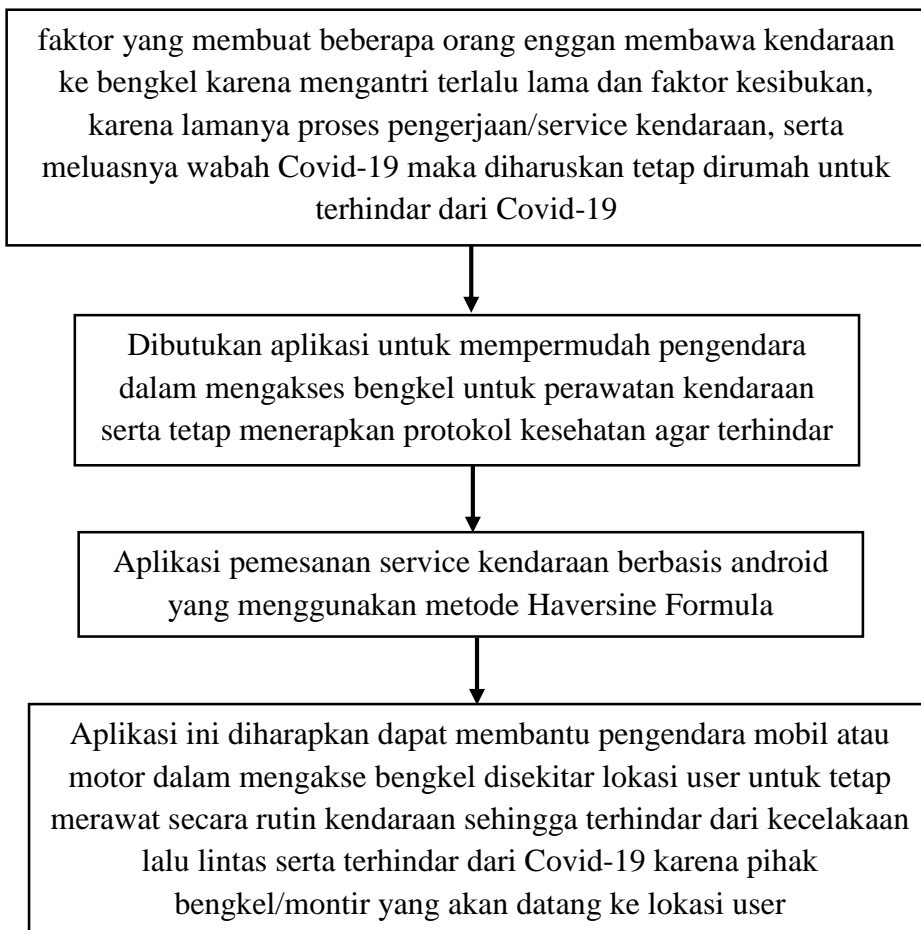
Nama Peneliti	Judul Penelitian	Tahun Penelitian	Metode yang digunakan	Hasil Penelitian
Helmi Kurniawan	Perancangan Sistem Informasi Bengkel Mobil Berbasis Web	2015	Metode Waterfall	rancangan sistem informasi bengkel mobil berbasis web
Moch. Fatchur Rozy, A. Prasita Nugroho, Moch. Nurcholis	Aplikasi Pelayanan dan Pengolahan Data Bengkel Secara Elektronik Berbasis Web	2017	-	Hasil yang diperoleh dari pengembangan sistem ini adalah dengan melakukan tahapan analisa sistem, analisa perancangan sistem dan perancangan basis data sehingga mempermudah kegiatan perusahaan, mengolah data, membuat laporan menjadi lebih mudah dan efisien, penyajian informasi yang akurat, relevan dan tepat waktu membuat pihak manajemen dalam mengambil keputusan.

A. Ferico Octaviansyah Pasaribu, Dedi Darwis, Agus Irawan, Ade Surahman	Sistem Informasi Geografis Untuk Pencarian Lokasi Bengkel Mobil Di Wilayah Kota Bandar Lampung	2019	-	Aplikasi Sistem Informasi Geografis untuk Pencarian Bengkel Mobil terdekat dari titik lokasi pengguna di Wilayah Bandar Lampung
Farid dan Yulanda Yunus	Analisa Algoritma Haversine Formula Untuk Pencarian Lokasi Terdekat Rumah Sakit Dan Puskesmas Probinsi Gorontalo	2017	Haversine Formula	suatu analisa algoritma Haversine Formula lokasi pemetaan Rumah Sakit dan Puskesmas untuk pencarian lokasi terdekat ke rumah sakit satu ke rumah sakit lainnya dan puskesmas ke puskesmas lainnya.
Sarif Ifan Purnawan, Fitri Marisa, Indra Dharma Wijaya	Aplikasi Pencarian Pariwisata Dan Tempat Oleh -Oleh Terdekat	2018	Haversine Formula	sebuah sistem pencarian lokasi hotel di kota Semarang berbasis android yang menggunakan metode Haversine Formula

	Menggunakan Metode Haversine Berbasis Android			
--	---	--	--	--

### II.3 Kerangka Pemikiran

Perkembangan teknologi merupakan keadaan yang dibutuhkan di masa globalisasi yang menuntut adanya proses instan. Aplikasi pemesanan service kendaraan berbasis android yang akan memungkinkan pengendara (*user*) untuk dengan mudah mengakses bengkel tanpa pergi ke lokasi bengkel dimana aplikasi terhubung ke jaringan internet menggunakan smartphome android mereka yang tentunya saat ini hampir semua pihak menggunakannya. Berdasarkan uraian di atas, maka kerangka pikir penelitian ini dapat digambarkan dalam bentuk diagram sebagai berikut:

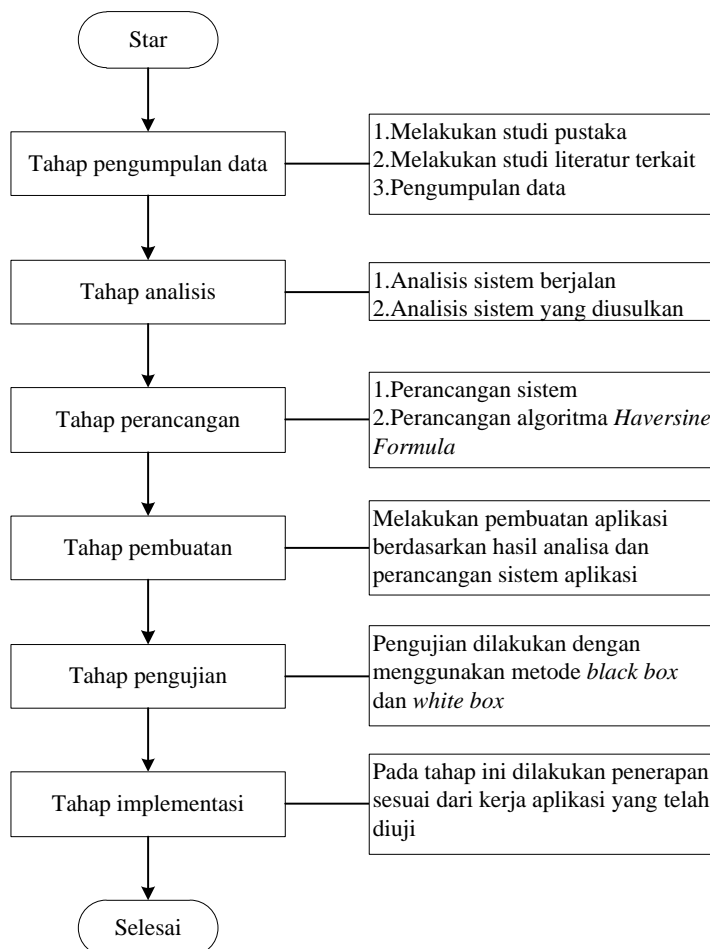


Gambar II.9 Kerangka Pemikiran

## BAB III METODE PENELITIAN

### III.1 Tahapan Penelitian

Tahapan penelitian mencakup langkah - langkah pelaksanaan dari awal sampai akhir, adapun langkahnya sebagai berikut:

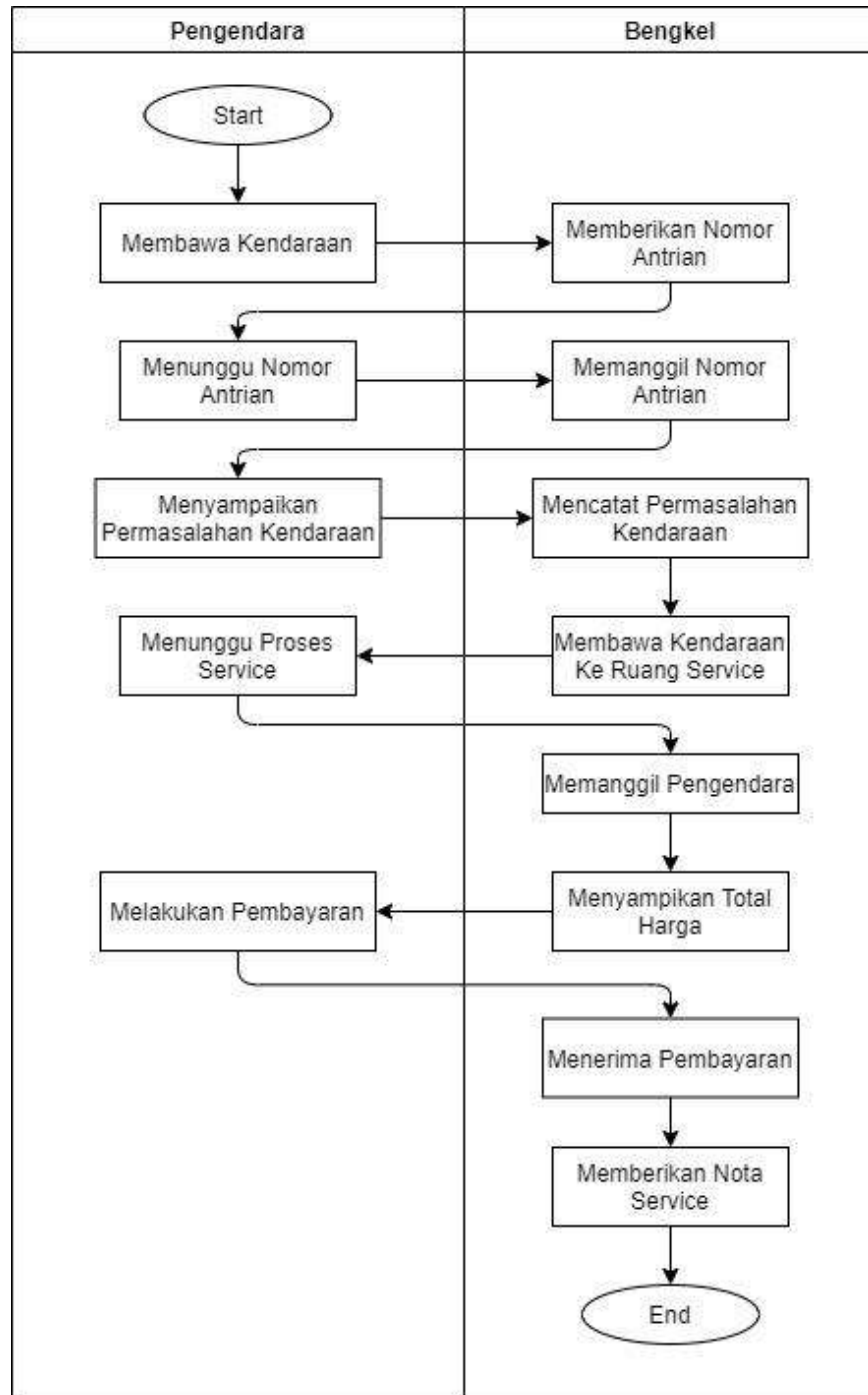


Gambar III.1 Tahapan Penelitian

Pada gambar diatas dapat dilihat bahwa tahap penelitian yang akan dilakukan adalah mulai dari pengumpulan data dari sumber-sumber mengenai penelitian terkait, tahap analisis sistem berjalan dan yang akan direncanakan, tahap perancangan mulai dari design tahap pembuatan serta tahap pengujian kerja dari aplikasi yang dibuat.

## III.2 Rancangan Penelitian Sistem

### III.2.1 Sistem yang Sedang Berjalan



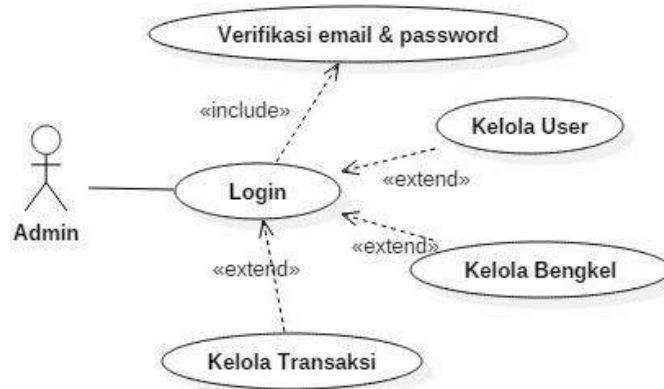
Gambar III.2 Sistem yang Sedang Berjalan



### III.2.2 Sistem yang Direncanakan

#### III.2.2.1 Use Case Diagram

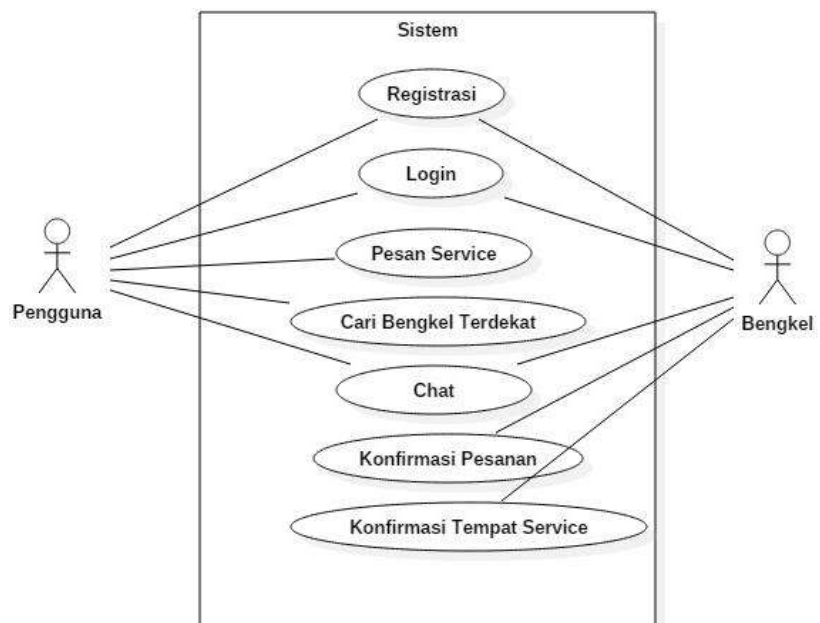
##### 1. Use Case Diagram Admin



Gambar III.3 Use Case Diagram Admin

Pada Gambar diatas fungsi admin ialah dapat mengelola baik pengguna (user) ataupun pihan bengkel serta dapat mengelola transaksi antara user dan pihak bengkel.

##### 2. Use Case Diagram Pengguna dan Pihak Bengkel



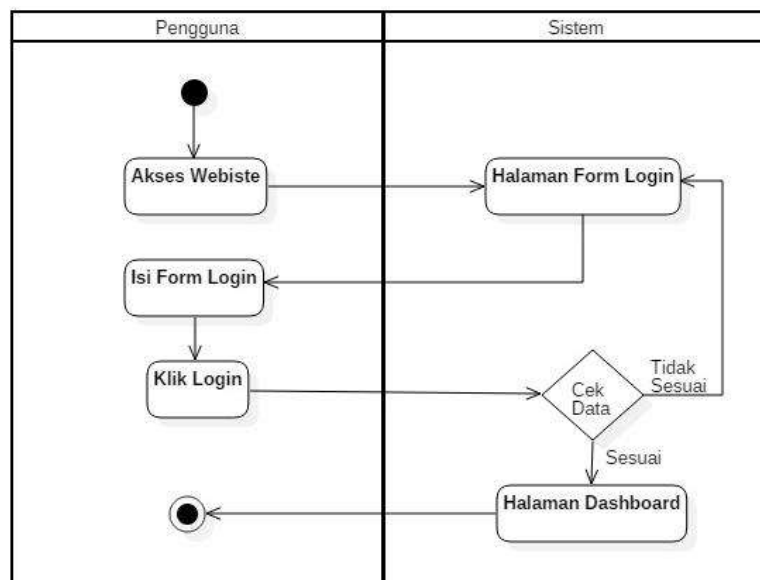
Gambar III.4 Use Case Diagram

Pada Gambar III.4 menampilkan apa saja yang dapat dilakukan oleh user atau pelanggan dan pihak bengkel.

### III.2.2.2 Activity Diagram

Aktivitas diagram digunakan untuk menunjukkan aliran pesan dari suatu aktivitas ke aktivitas lainnya. Di dalam sistem ini terdapat berbagai macam aktivitas yang dapat dilakukan oleh user atau pelanggan dan pihak bengkel atau montir. Dari aktivitas-aktivitas tersebut, dapat dibuat activity diagram yang dapat dilihat pada gambar dibawah.

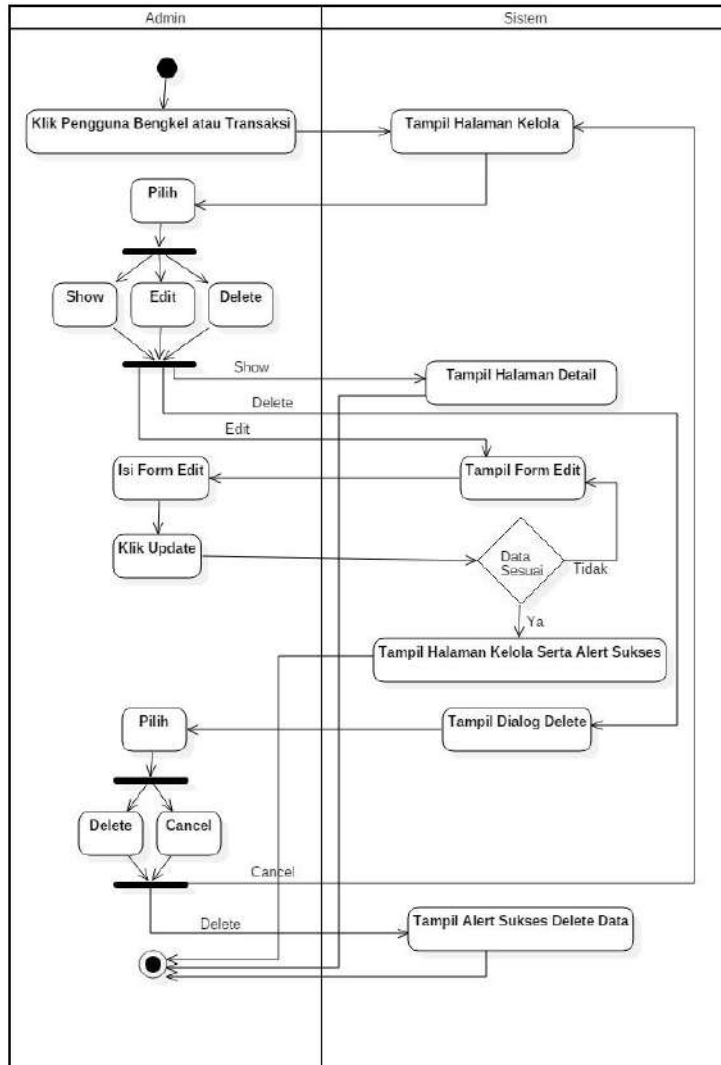
#### 1. Activity Diagram Diagram Login Admin



Gambar III.5 Activity Diagram Diagram Login Admin

Pada activity diagram diatas menjelaskan bagaimana admin dapat mengakses web admin dengan cara login terlebih dahulu.

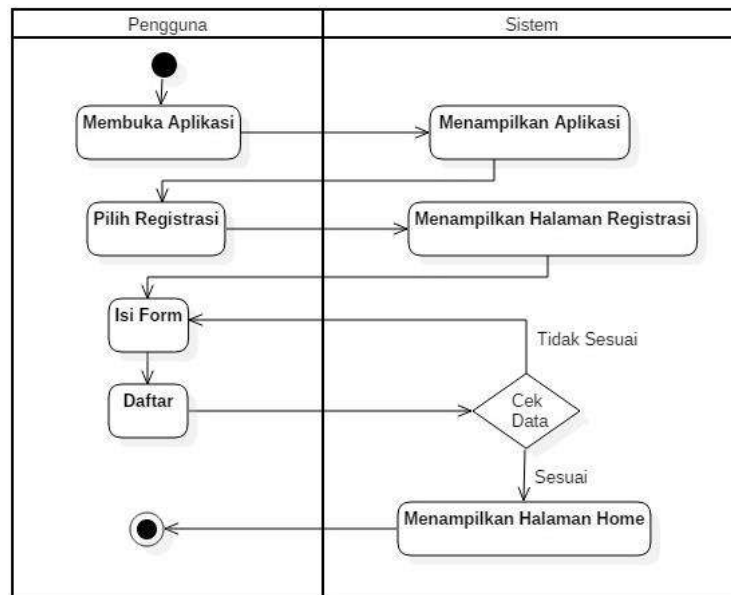
## 2. Activity Diagram Admin Kelola Pengguna, Bengkel, dan Transaksi



Gambar III.6 Activity Diagram Kelola Pengguna, Bengkel, dan Transaksi

Salah satu fungsi admin adalah dapat mengelolah pengguna, bengkel, dan transaksi pada gambar III.6 admin dapat melihat, mengedit serta menghapus data pengguna, data bengkel maupun data transaksi.

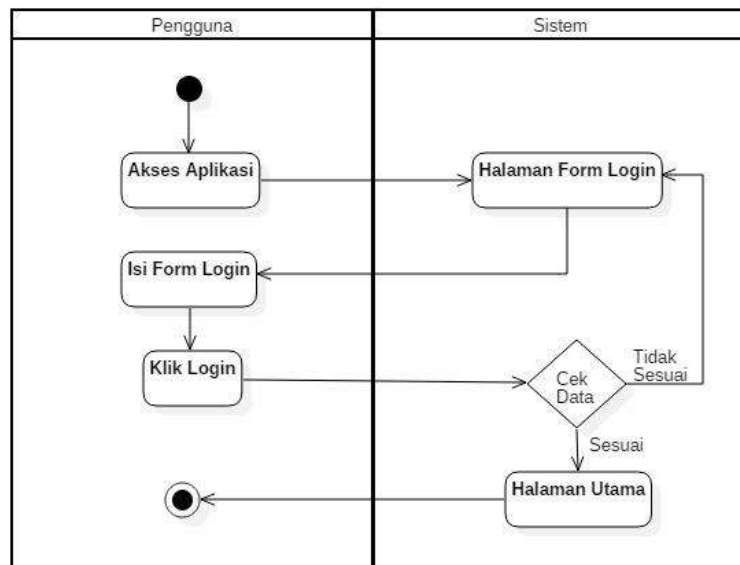
### 3. Activity Diagram Registrasi User dan Bengkel



Gambar III.7 Activity Diagram Registrasi User dan Bengkel

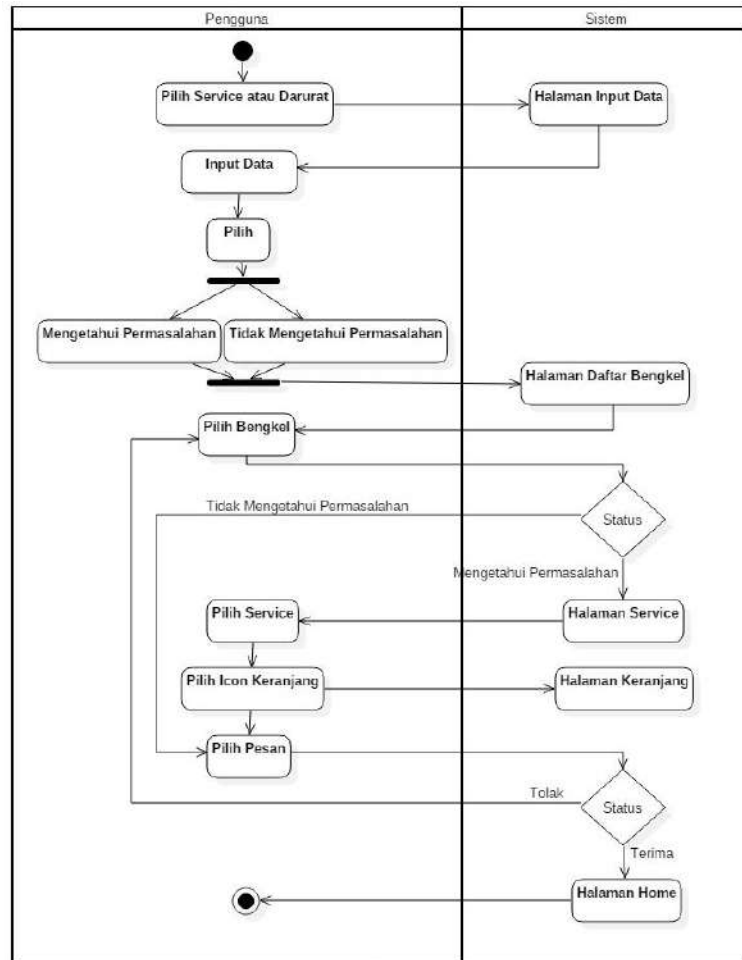
Pada Activity Diagram diatas menjelaskan bahwa bagaimana pengguna dapat memesan layanan service hingga memilih bengkel terdekat yang otomatis akan ditampilkan berdasarkan titik lokasi pengguna.

### 4. Activity Diagram Login User dan Bengkel



Gambar III.8 Activity Diagram Login User dan Bengkel

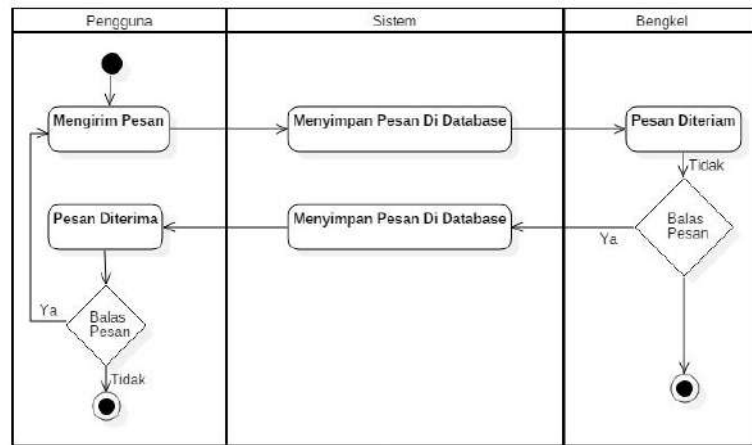
## 5. Activity Diagram Pemesanan Service



Gambar III.9 Activity Diagram Pemesanan Service

Pada gambar diatas menjelaskan bahwa alur pemesanan service, dimana sebelum memesan service pengguna dapat mengisi pesan-pesan keluhan terhadap kendaraan, lalu pengguna akan menginput titik lokasi melalui halaman peta, setelah menekan tombol cari bengkel maka system akan merequest data yang sesuai diinput pada database dan akan menampilkan data yang diperlukan pada halaman service.

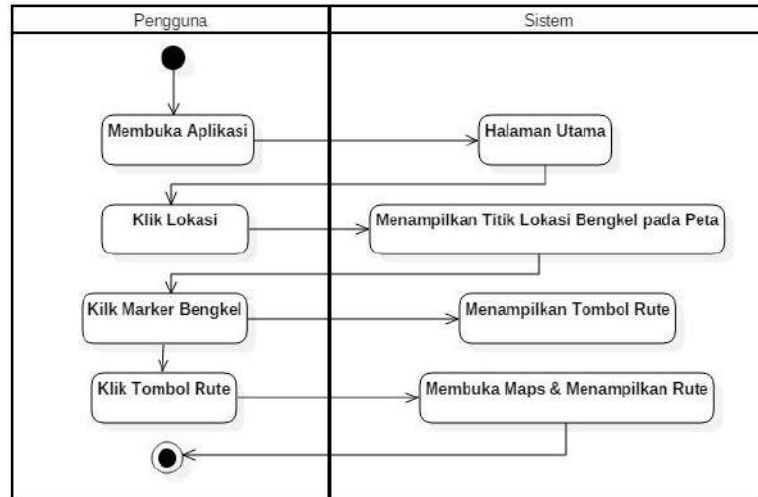
## 6. Activity Diagram Chat



Gambar III.10 Activity Diagram Chat

Activity diagram ini menjelaskan bahwa pihak bengkel dan pengguna dapat saling berkomunikasi malalui teks.

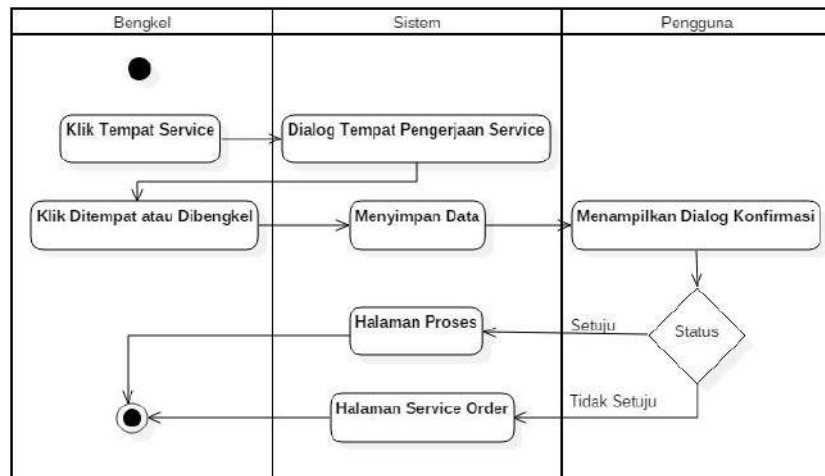
## 7. Activity Diagram User Mencari Bengkel Terdekat



Gambar III.11 Activity Diagram User Mencari Bengkel Terdekat

Pada gambar diatas menjelaskan bahwa bagaimana user dapat secara langsung melihat semua titik-titik (*marker*) lokasi bengkel disekitar pengguna.

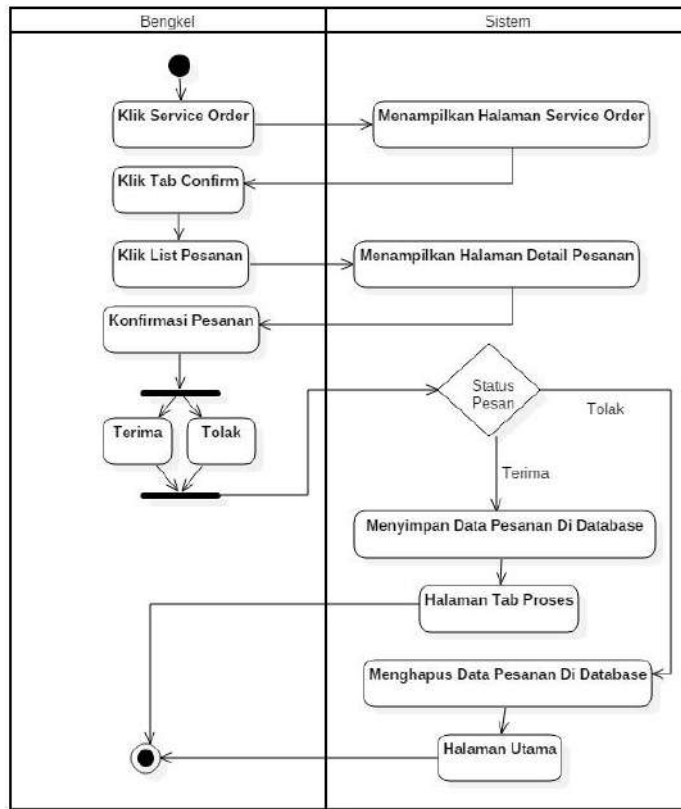
### 8. Activity Diagram Konfirmasi Tempat Service



Gambar III.12 Activity Diagram Bengkel Nota Layanan Service

Pada gambar diatas menjelaskan bahwa bagaimana pihak bengkel menentukan apakah pengerjaan service dapat dilakukan ditempat atau dibawa ke bengkel dengan persetujuan pengguna terlebih dahulu.

## 9. Activity Diagram Konfirmasi Pesanan



Gambar III.13 Activity Diagram Konfirmasi Pesanan

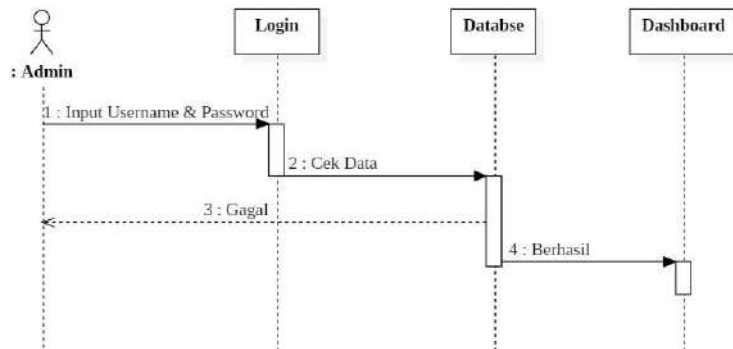
Pada gambar diatas menjelaskan bahwa bagaimana pihak bengkel mengkonfirmasi pesanan service, apakah diterima atau ditolak.

### III.2.2.3 Sequence Diagram

Sequenced diagram adalah salah satu dari diagram-diagram yang ada pada UML, sequence diagram ini adalah diagram yang menggambarkan kolaborasi dinamis antara sejumlah object juga interaksi antar object.

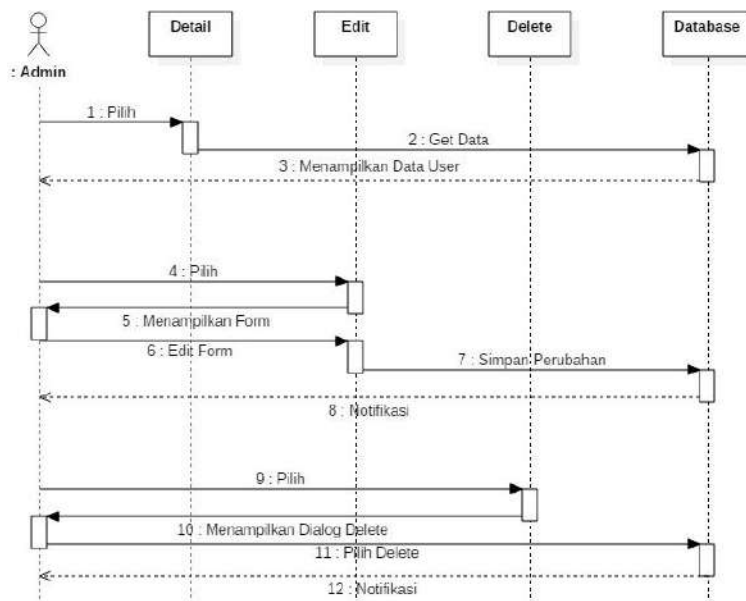


### 1. Sequence Diagram Login Admin



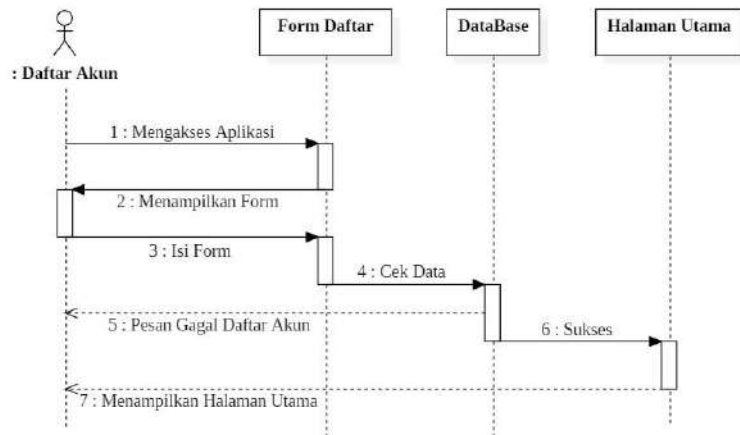
Gambar III.14 Sequence Diagram Login Admin

### 2. Sequence Diagram Admin Kelola Pengguna, Bengkel, dan Transaksi



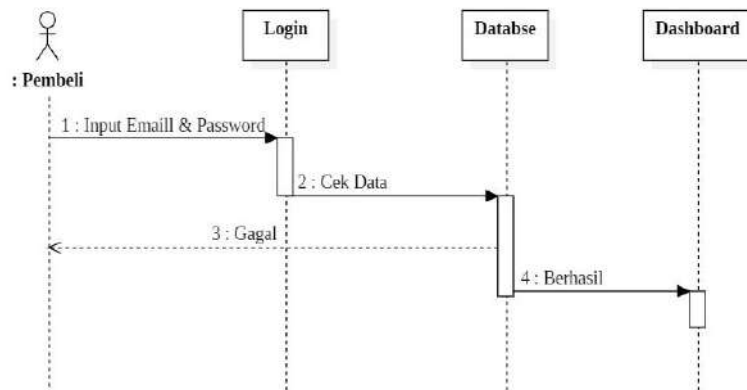
Gambar III.15 Sequence Diagram Admin Kelola User

### 3. Sequence Diagram Daftar Akun User dan Bengkel



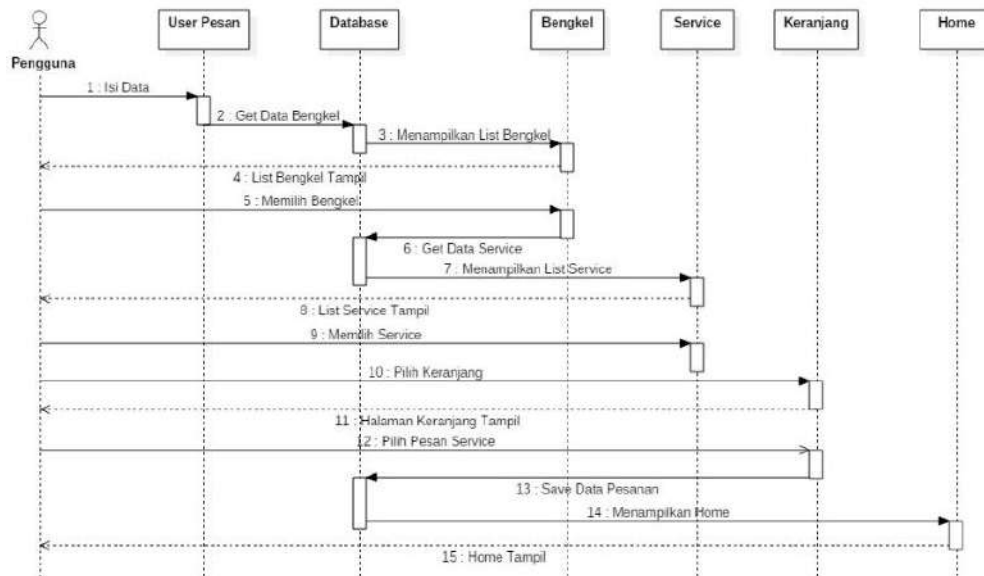
Gambar III.16 Sequence Diagram Daftar Akun User dan Bengkel

### 4. Sequence Diagram Login Akun User dan Bengkel



Gambar III.17 Sequence Diagram Login Akun User dan Bengkel

### 5. Sequence Diagram Pemesanan Service



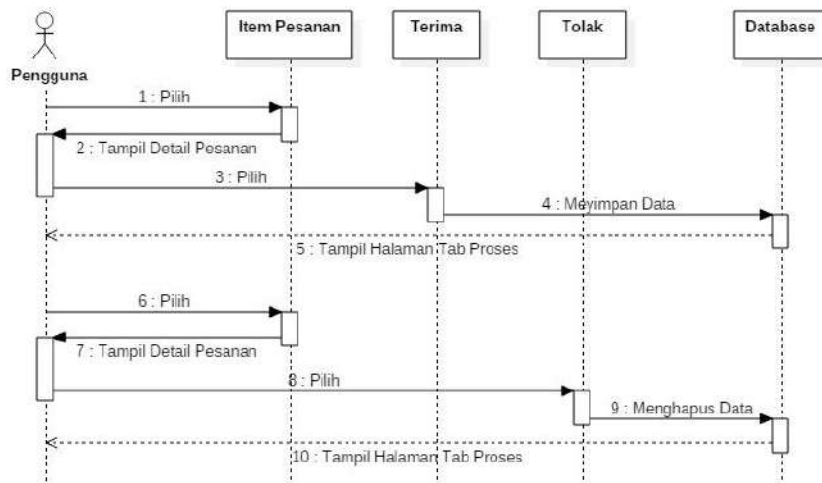
Gambar III.18 Sequence Diagram Pemesanan Service

### 6. Sequence Diagram User Cari Bengkel Terdekat



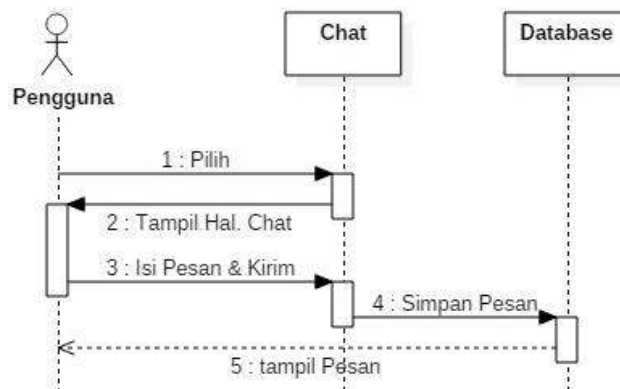
Gambar III.19 Sequence Diagram User Cari Bengkel Terdekat

### 7. Sequence Diagram Konfirmasi Pesanan



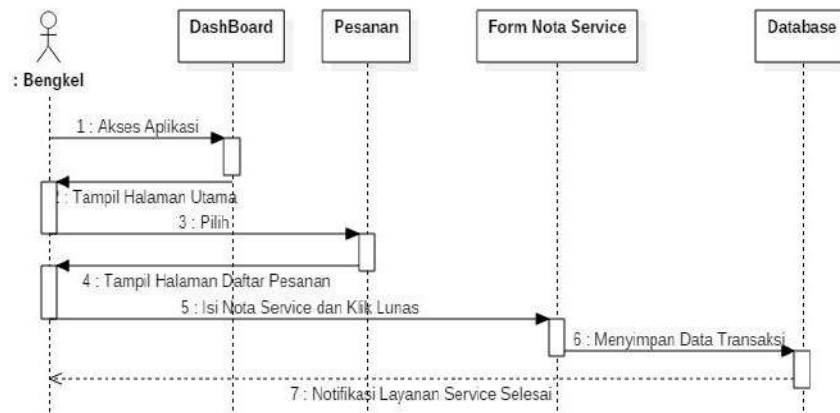
Gambar III.20 Sequence Diagram Konfirmasi Pesanan

### 8. Sequence Diagram Chat



Gambar III.21 Sequence Diagram Chat

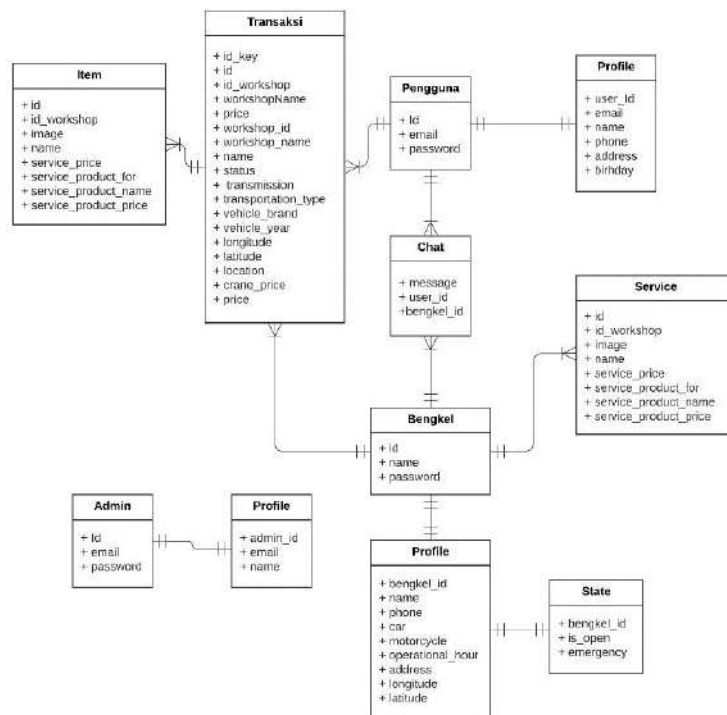
## 9. Sequence Diagram Konfirmasi Tempat Service



Gambar III.22 Sequence Diagram Konfirmasi Tempat Service

### III.2.2.4 Class Diagram

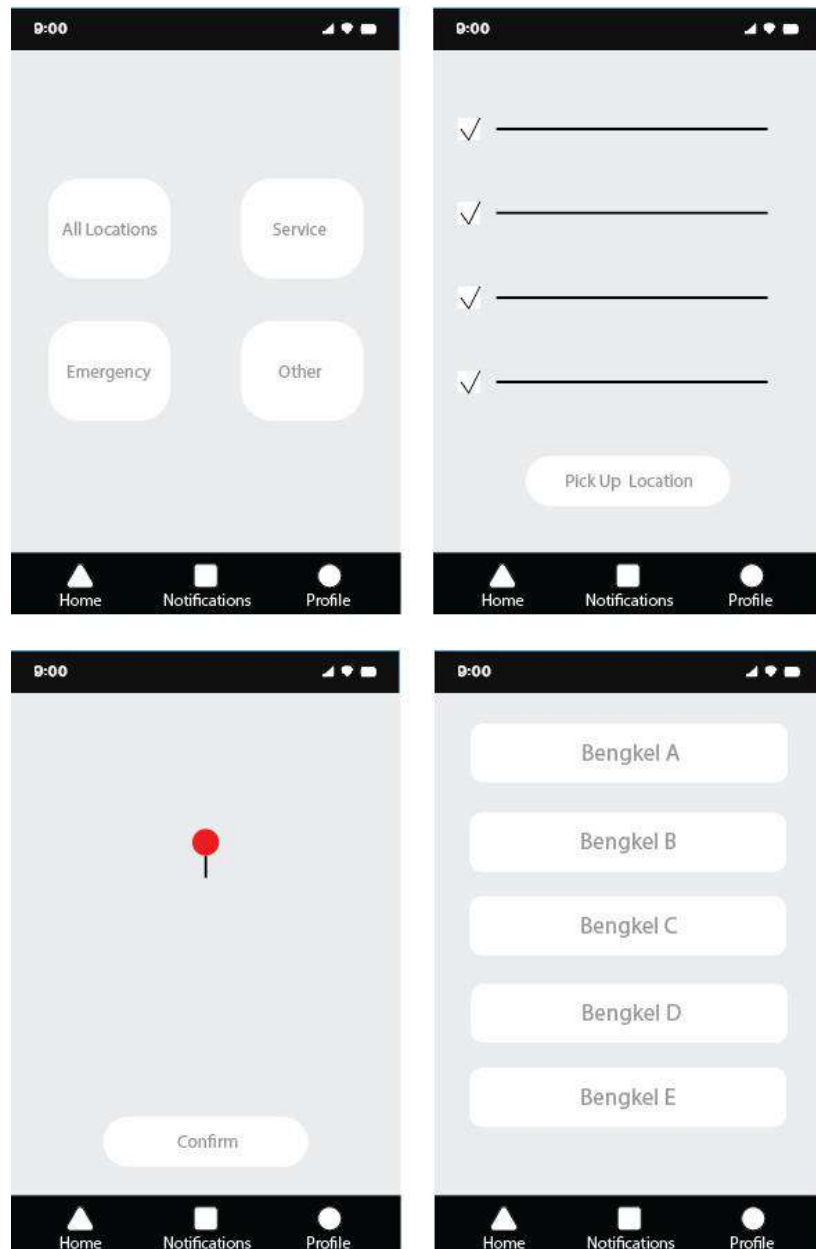
Class Diagram merupakan diagram yang digunakan untuk menampilkan kelas-kelas yang ada dalam sistem berupa objek yang sedang dikembangkan dari kelas yang satu ke kelas yang lain yang mempunyai relasi. Setiap kelas dalam class diagram terdiri dari nama kelas, atribut, dan operasi dari kelas tersebut.



Gambar III.23 Class Diagram

### III.2.2.5 Desain Interface

Dalam pembuatan aplikasi ini diperlukan desain untuk gambaran bentuk fisik terhadap sistem yang akan dibangun sehingga mempermudah user mengerti maksud dan isi halaman antar-muka dalam waktu yang singkat.



Gambar III.24 Desain Interface

### **III.3 Waktu dan Tempat Penelitian**

Adapun Waktu dan Tempat penelitian dilakukan ketika peneliti telah menyelesaikan seminar proposal, dari semester genap juni 2021 - oktober 2021. Penelitian ini dilaksanakan di bengkel Dua Putra, jalan Usman Salangke 110E, Sungguminasa, Kabupaten Gowa.

### **III.4 Alat dan Bahan Penelitian**

Dalam pelaksanaan dan penyusunan Laporan Tugas Akhir (TA) ini digunakan alat dan bahan sebagai berikut:

#### **III.4.1 Perangkat Keras**

Perangkat Keras (Hardware) yang digunakan dalam pembuatan sistem ini dengan spesifikasi sebagai berikut:

1. Perangkat komputer atau laptop
2. Ram DDR4 8 GB
3. Smartphone Android
4. Mouse

#### **III.4.2 Perangkat Lunak**

Adapun perangkat lunak yang digunakan adalah:

1. OS Windows 10
2. Android Studio
3. Emulator
4. Web Browser
5. Firebase

### **III.5 Metode Pengumpulan Data**

Metode pengumpulan data berupa suatu pernyataan tentang sifat, keadaan, kegiatan tertentu dan sejenisnya. Pengumpulan data dilakukan untuk mendapatkan suatu informasi yang dibutuhkan dalam mencapai tujuan penelitian. Dalam penyusunan tugas akhir ini penulis mengambil objek penelitian pada bengkel Dua Putra, jalan Usman Salangke 110E,

Sungguminasa. Pengumpulan data dalam penelitian menggunakan 3 cara, berikut merupakan uraian yang digunakan:

1. Observasi

Suatu metode pengumpulan data yang dilakukan dengan mengamati Langsung, melihat dan mengambil suatu data yang dibutuhkan di tempat Penelitian itu dilakukan. Observasi juga bisa diartikan sebagai proses yang kompleks. Pengumpulan data yang dilakukan di bengkel Dua Putra, jalan Usman Salangke 110E, Sungguminasa.

2. Wawancara

Wawancara merupakan salah satu teknik pengumpulan data yang dilakukan Melalui tatap muka langsung dengan narasumber dengan cara tanya jawab Langsung. Wawancara dilakukan dengan Pemilik bengkel Dua Putra, yang berhubungan dengan data yang terkait.

3. Studi Literatur

Studi Literatur adalah cara untuk menyelesaikan persoalan dengan menelusuri sumber-sumber tulisan yang pernah dibuat sebelumnya. Dengan kata lain, istilah Studi Literatur ini juga sangat familiar dengan sebutan studi pustaka. Dalam sebuah penelitian yang akan dijalankan, tentunya seorang peneliti harus memiliki wawasan yang luas terkait objek yang akan diteliti. Jika tidak, maka dapat dipastikan dalam presentasi yang besar bahwa penelitian tersebut akan gagal.

### **III.6 Metode Analisis Data/Pengujian Sistem**

Teknik pengujian yang dilakukan terhadap aplikasi sistem informasi ini adalah pengujian dengan metode white box dan black box. Pengujian white box didasarkan pada pengecekan terhadap detail perancangan seperti pengujian tombol-tombol yang ada pada sistem informasi sedangkan pengujian black box dilakukan melalui pengujian fungsi-fungsi yang ada pada sistem informasi.



### III.7 Jadwal Penelitian

Adapun rencana perkiraan penelitian sebagai berikut:

Tabel III.1 Jadwal Penelitian

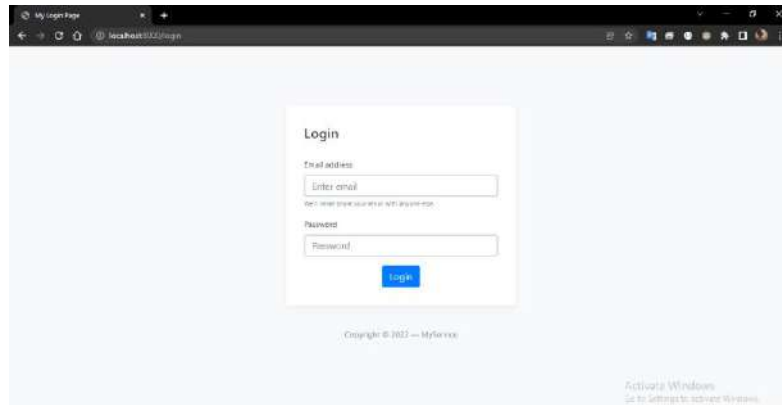
No.	Kegiatan	Waktu (Pekan)													
		1	2	3	4	5	6	7	8	9	10	11	12	13	14
1.	Perencanaan	■	■	■											
2.	Pengumpulan data				■	■	■								
3.	Analisa dan perancangan							■	■	■	■				
4.	Implementasi dan pengujian											■	■		
5.	Dokumentasi													■	■

## BAB IV PEMBAHASAN DAN PEMBAHASAN

### IV.1 Hasil

#### IV.1.1 *Interface Website Database Aplikasi*

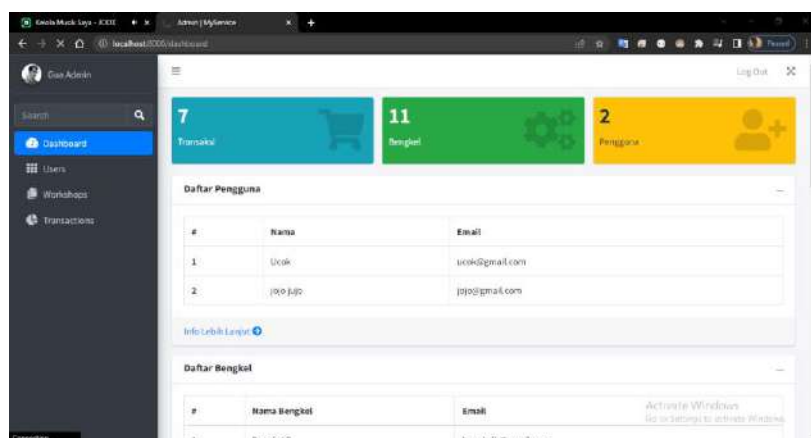
##### 1. Tampilan Halaman Login Admin



Gambar IV.1 Tampilan Halaman Login Admin

Pada gambar diatas menampilkan halaman admin ketika website pertama kali diakses. Untuk melakukan kelola data pada sisi admin harus dilakukan Login ke dalam website terlebih dahulu. Admin dapat Login menggunakan Email dan Password. Email dan Password harus sesuai dengan data yang ada pada database.

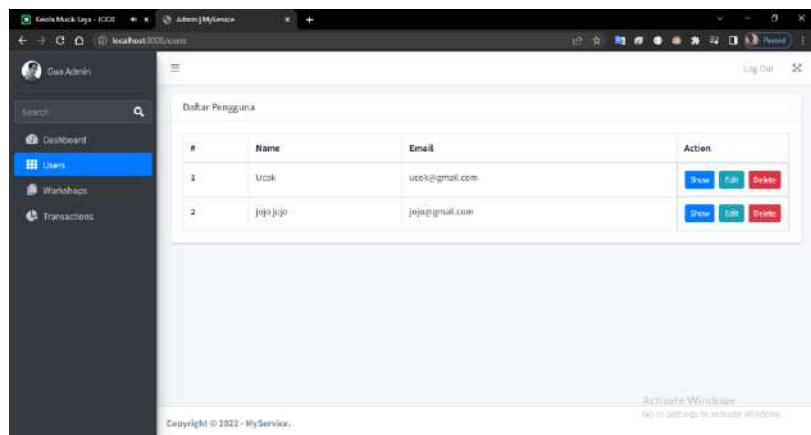
##### 2. Tampilan Halaman Dashboard



Gambar IV.2 Tampilan Halaman Dashboard

Pada gambar diatas menampilkan halaman dashboard ketika admin selesai melakukan login. pada halaman dashboard tampil informasi jumlah transaksi, jumlah bengkel, jumlah pengguna, serta menampilkan beberapa detail data transaksi, bengkel, dan pengguna.

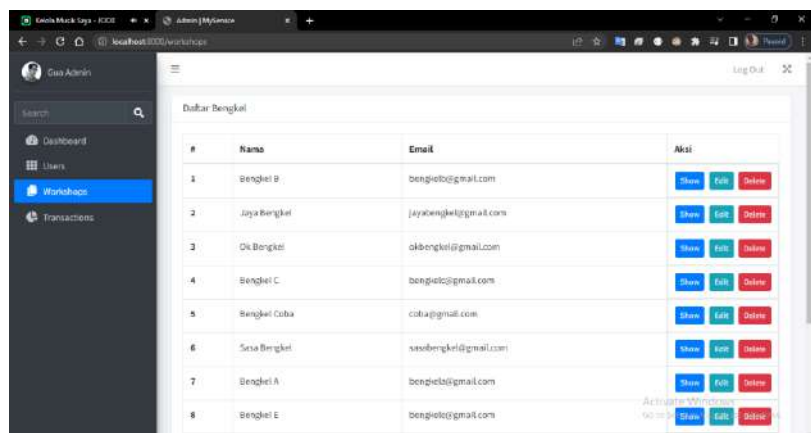
### 3. Tampilan Halaman Daftar User



Gambar IV.3 Tampilan Halaman Daftar User

Pada gambar diatas menampilkan halaman daftar user atau pengguna, pada halaman daftar ini admin dapat mengelola pengguna seperti melihat detail data pengguna, mengedit data pengguna, dan menghapus pengguna.

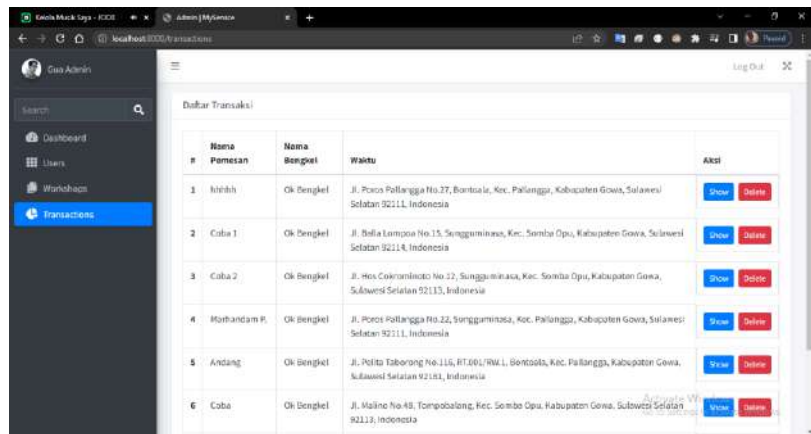
### 4. Tampilan Halaman Daftar Bengkel



Gambar IV.4 Tampilan Halaman Daftar Bengkel

Pada gambar diatas menampilkan halaman daftar bengkel, pada halaman daftar ini admin dapat mengelola pengguna seperti melihat detail data bengkel, mengedit data bengkel, dan menghapus bengkel.

## 5. Tampilan Halaman Daftar Transaksi



No Pemesan	Nama Bengkel	Waktu	Aksi	
1	hrihh	Ok Bengkel	Jl. Porek Palangka No.37, Bontosa le, Kec. Palangka, Kabupaten Gowa, Sulawesi Selatan 92114, Indonesia	<a href="#">View</a> <a href="#">Delete</a>
2	Coba 1	Ok Bengkel	Jl. Bolla Lompoe No.15, Sunggaminasa, Kec. Sembu Opu, Kabupaten Gowa, Sulawesi Selatan 92114, Indonesia	<a href="#">View</a> <a href="#">Delete</a>
3	Coba 2	Ok Bengkel	Jl. Hns Cakramanico No.22, Sunggaminasa, Kec. Sembu Opu, Kabupaten Gowa, Sulawesi Selatan 92113, Indonesia	<a href="#">View</a> <a href="#">Delete</a>
4	Morhandam R.	Ok Bengkel	Jl. Porek Palangka No.22, Sunggaminasa, Kec. Palangka, Kabupaten Gowa, Sulawesi Selatan 92114, Indonesia	<a href="#">View</a> <a href="#">Delete</a>
5	Andang	Ok Bengkel	Jl. Polta Taborong No.116, RT.061/RW.1, Bontosa, Kec. Palangka, Kabupaten Gowa, Sulawesi Selatan 92181, Indonesia	<a href="#">View</a> <a href="#">Delete</a>
6	Coba	Ok Bengkel	Jl. Maline No.48, Tompobalang, Kec. Sembu Opu, Kabupaten Gowa, Sulawesi Selatan 92113, Indonesia	<a href="#">View</a> <a href="#">Delete</a>

Gambar IV.5 Tampilan Halaman Daftar Transaksi

Pada gambar diatas menampilkan halaman daftar transaksi, pada halaman daftar ini admin dapat mengelola pengguna seperti melihat detail data transaksi, mengedit data transaksi, dan menghapus transaksi.

## IV.1.2 *Interface* Pada Aplikasi Android Dari Sisi Pengguna

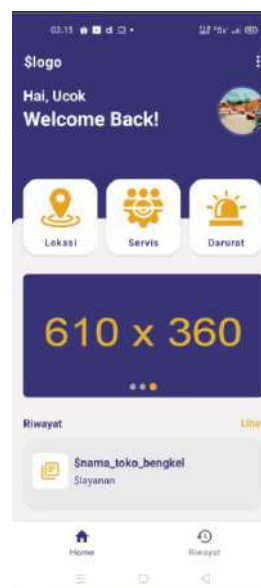
### 1. Tampilan Splash Screen Aplikasi



Gambar IV.6 Tampilan Spalsh Screen Aplikasi Pengguna

Gambar diatas menampilkan halaman splash screen ketika aplikasi pertama kali dibuka. Aplikasi menampilkan gambar icon aplikasi dan nama aplikasi.

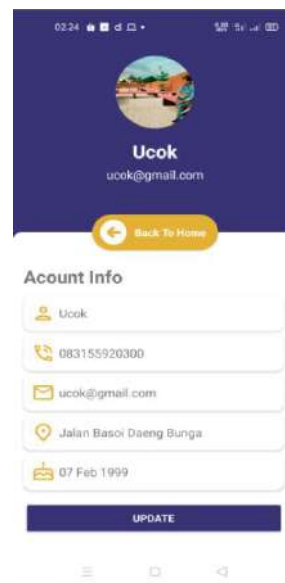
### 2. Tampilan Halaman Utama Aplikasi



Gambar IV.7 Tampilan Halaman Utama Aplikasi Pengguna

Gambar diatas menampilkan halaman utama pada aplikasi, dihalaman ini menampilkan fitur utama yaitu cek lokasi, service, dan darurat. Serta menampilkan foto profile pengguna, image slider untuk banner iklan atau promo, transaksi terbaru, dan bottom navigation.

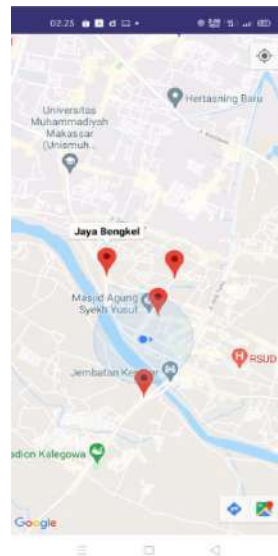
### 3. Tampilan Halaman Profile Pengguna



Gambar IV.8 Tampilan Halaman Profile Pengguna

Gambar diatas menampilkan halaman profile pengguna pada aplikasi, dihalaman ini menampilkan foto profile pengguna, menampilkan data-data pengguna dan fitur edit untuk mengubah data pengguna.

#### 4. Tampilan Halaman Fitur Lokasi



Gambar IV.9 Tampilan Halaman Fitur Lokasi

Gambar diatas menampilkan halaman fitur lokasi dimana aplikasi akan menampilkan titik-titik lokasi bengkel terdekat dari lokasi pengguna dengan menggunakan metode Haversine Formula.

#### 5. Tampilan Halaman Fitur Service

14:27

Nama

Jenis Kendaraan

Merek Kendaraan

Tahun

Transmisi

Lokasi Penjemputan

Mengetahui Permasalahan

Tidak Mengetahui Permasalahan

Gambar IV.10 Tampilan Halaman Fitur Service

Gambar diatas menampilkan halaman fitur service motor atau mobil dimana terdapat inputan nama, jenis kendaraan, merek kendaraan, tahun, transmisi kendaraan, lokasi penjemputan. Serta terdapat tombol mengetahui permasalahan dan tidak mengetahui permasalahan.

## 6. Tampilan Halaman Confirm Location



Gambar IV.11 Tampilan Halaman Confirm Location

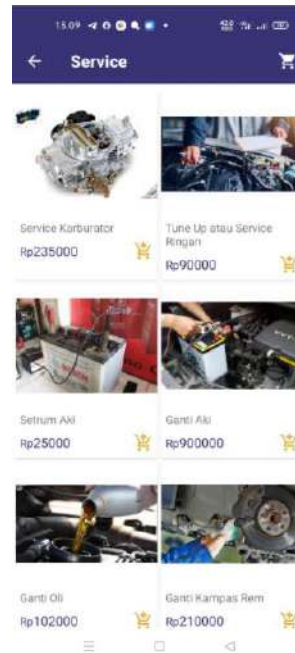
Gambar diatas menampilkan halaman confirm location dimana pengguna akan menentukan lokasi penjemputan kendaraan serta secara otomatis akan menampilkan alamat dari titik lokasi.





Pada gambar diatas diatas menampilkan halaman daftar (tidak mengetahui permasalahan) bengkel dimana pada halaman ini akan menampilkan list-list bengkel serta menampilkan total harga service.

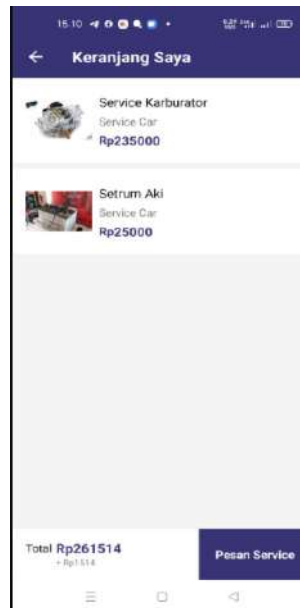
#### 9. Tampilan Halaman Daftar Service



Gambar IV.14 Halaman Daftar Service

Pada gambar diatas menampilkan halaman daftar service dimana pada halaman ini akan menampilkan list-list service yang akan dipilih pengguna.

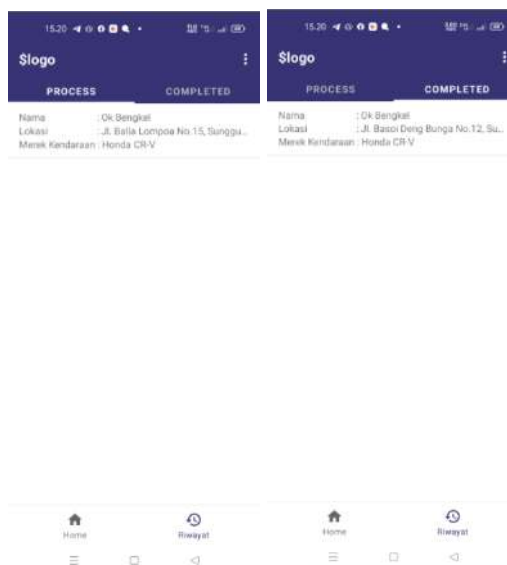
## 10. Tampilan Halaman Daftar Pesanan (Keranjang)



Gambar IV.15 Halaman Daftar Pesanan (Keranjang)

Pada gambar diatas menampilkan halaman daftar pesanan dimana pada halaman ini menampilkan daftar pesanan yang telah dipilih dihalaman daftar service serta dihalaman ini menampilkan total harga yang harus dibayar.

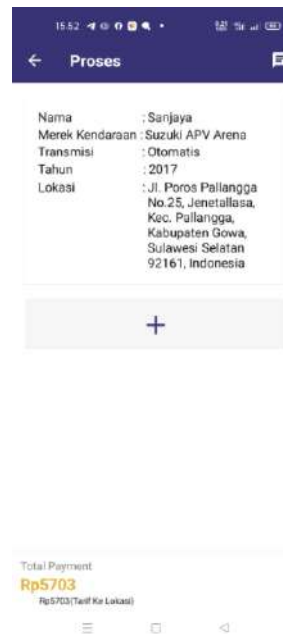
## 11. Tampilan Halaman Riwayat



Gambar IV.16 Halaman Riwayat

Gambar diatas menampilkan halaman history dimana terdapat dua tab yaitu process dan completed. Tab process akan menampilkan daftar pesanan service yang masih dalam pengerjaan sedangkan tab completed menampilkan daftar pesanan service yang sudah selesai.

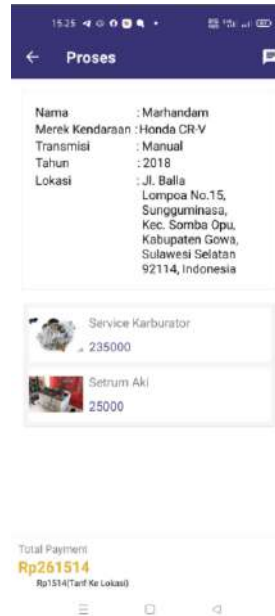
## 12. Tampilan Halaman Detail Proses Pengguna (Tidak Mengetahui Permasalahan)



Gambar IV.17 Halaman Detail Proses Pengguna (Tidak Mengetahui Permasalahan)

Pada gambar diatas menampilkan halaman detail proses (tidak mengetahui permasalahan) dimana pada halaman tersebut menampilkan data pesanan, icon tambah untuk menambah service, dan menampilkan total harga service. Ketika mengklik icon tambah maka akan otomatis dialihkan ke halaman daftar service, disini pengguna dapat memilih service sesuai arahan dari pihak bengkel atau montir yang telah memeriksa kendaraan pengguna karena studi kasusnya pengguna tidak mengetahui permasalahan kendaraan.

13. Tampilan Halaman Detail Proses Pengguna (Mengetahui Permasalahan)



Gambar IV.18 Halaman Detail Proses Pengguna (Mengetahui Permasalahan)

Gambar diatas menampilkan halaman detail process dimana terdapat fitur chat, tampilan data pesanan, daftar service yang telah dipesan serta total harga yang harus dibayar.

14. Tampilan Halaman Fitur Chat Dari Sisi Pengguna



Gambar IV.19 Tampilan Halaman Fitur Chat Dari Sisi Pengguna

Gambar diatas menampilkan halaman firut chat dari sisi pengguna dimana fitur ini digunakan untuk saling berkomunikasi melalui text antara pihak pemesan dan pihak bengkel.

#### IV.1.3 *Interface* Pada Aplikasi Android Dari Sisi Bengkel

##### 1. Tampilan Splash Screen Aplikasi Bengkel



**\$LOGO**



Gambar IV.20 Tampilan Spalsh Screen Aplikasi Bengkel

Gambar diatas menampilkan halaman splash screen ketika aplikasi pertama kali dibuka. Aplikasi menampilkan gambar icon aplikasi dan nama aplikasi.

## 2. Tampilan Halaman Utama Aplikasi Bengkel



Gambar IV.21 Tampilan Halaman Utama Aplikasi Bengkel

Gambar diatas menampilkan halaman utama pada aplikasi, dihalaman ini menampilkan fitur utama yaitu profile, service order, dan service price. Selain itu terdapat toggle on/off yang merepresentasikan bengkel tersebut siap mendapatkan orderan service atau tidak dan tombol logout.

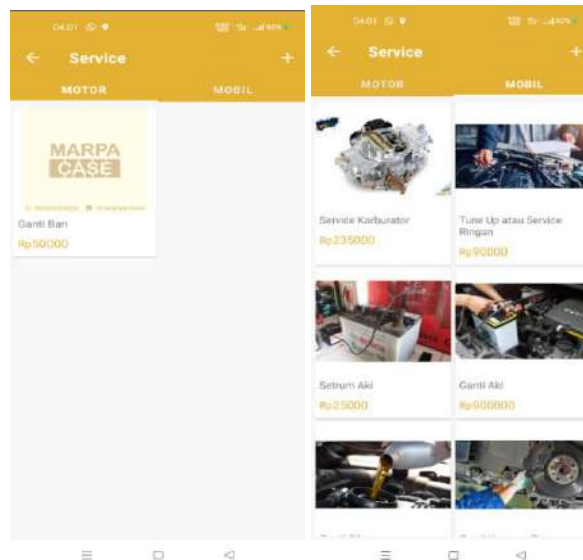
## 3. Tampilan Halaman Profile Bengkel



Gambar IV.22 Tampilan Halaman Profile Bengkel

Gambar diatas menampilkan halaman profile bengkel pada aplikasi, dihalaman ini menampilkan data-data bengkel.

#### 4. Tampilan Halaman Service Pada Bengkel



Gambar IV.23 Tampilan Halaman Service

Gambar diatas menampilkan halaman service dimana aplikasi akan menampilkan daftar service motor dan mobil yang tersedia dibengkel tersebut.

#### 5. Tampilan Halaman Service Order



Gambar IV.24 Tampilan Halaman Service Order



Gambar diatas menampilkan halaman service order dimana terdapat tiga tab yaitu confirm, process dan completed. Tab confirm akan menampilkan list pesanan service yang akan diterima tau ditolak, tab process akan menampilkan daftar pesanan service yang masih dalam pengerjaan sedangkan tab completed menampilkan daftar pesanan service yang sudah selesai.

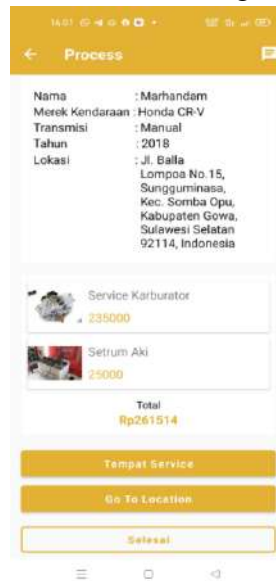
#### 6. Tampilan Halaman Detail Confirm Bengkel



Gambar IV.25 Tampilan Halaman Detail Confirm Bengkel

Gambar diatas menampilkan halaman detail process dimana pihak bengkel akan menerima pesanan service atau menolak pesanan service.

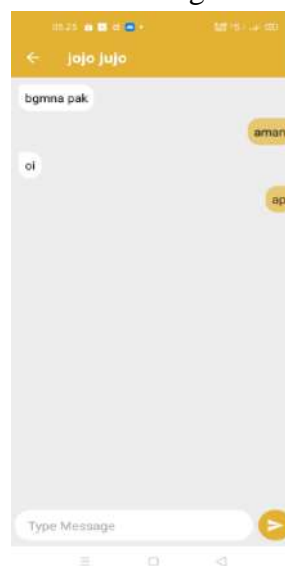
## 7. Tampilan Halaman Detail Process Bengkel



Gambar IV.26 Tampilan Halaman Detail Process Bengkel

Gambar diatas menampilkan halaman detail process, dimana terdapat fitur chat, tombol go to location, serta menampilkan data pesanan dan daftar service yang telah dipesan. Tombol go to location akan otomatis beralih ke google maps untuk menentukan tracking.

## 8. Tampilan Halaman Fitur Chat Bengkel



Gambar IV.27 Tampilan Halaman Fitur Chat Bengkel

Gambar diatas menampilkan halaman fitur chat dari sisi bengkel dimana fitur ini digunakan untuk saling berkomunikasi melalui text antara pihak pemesan dan pihak bengkel.

## IV.2 Pembahasan

### IV.1.1 Pengujian Black Box

Black box testing merupakan pengujian perangkat lunak dari sisi fungsional tanpa memperhatikan sisi desain dan kode pada program. Pengujian dilakukan untuk mengetahui apakah fungsi masukan (input) dan fungsi keluaran (output) dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.

#### 1. Pengujian Halaman *Login Admin*

Tabel IV.1 Pengujian Halaman *Login Admin*

<b>Kasus dan Hasil Uji</b>			
<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Input email	Melakukan validasi inputan	Berhasil melakukan validasi	Sesuai
Input password	Melakukan validasi inputan	Berhasil melakukan validasi	Sesuai
Klik login	Mengarahkan admin ke halaman dashboard	Berhasil mengarahkan admin ke halaman dashboard	Sesuai

Pada tabel IV.1 merupakan tabel pengujian halaman login admin, Pada tabel diatas terdapat beberapa pengujian diantaranya adalah input email, input password dan mengklik tombol login dimana hasil pengujian dapat berjalan dan berhasil sesuai yang diharapkan.

## 2. Pengujian Menu Navigation Sidebar pada Admin

Tabel IV.2 Pengujian Menu Navigation Sidebar

<b>Kasus dan Hasil Uji</b>			
<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Search menu	Melakukan pencarian menu	Berhasil melakukan pencarian menu	Sesuai
Klik dashboard	Menampilkan halaman dashboard	Berhasil menampilkan halaman dashboard	Sesuai
Klik Users	Menampilkan halaman users	Berhasil menampilkan halaman users	Sesuai
Klik Workshops	Menampilkan halaman workshops	Berhasil menampilkan halaman workshop	Sesuai
Klik Transactions	Menampilkan halaman transactions	Berhasil menampilkan halaman transactions	Sesuai

Pada tabel IV.2 merupakan tabel pengujian menu navigation sidebar pada admin, Pada tabel diatas terdapat beberapa pengujian diantaranya adalah melakukan pencarian menu, tombol dashboard, users, workshops, transactions, dan advertisement dimana hasil pengujian dapat berjalan dan berhasil sesuai yang diharapkan.

### 3. Pengujian Halaman Kelola Pengguna pada Admin

Tabel IV.3 Pengujian Halaman Kelola Pengguna

<b>Kasus dan Hasil Uji</b>			
<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Klik show	Menampilkan halaman detail	Berhasil menampilkan halaman detail	Sesuai
Klik edit	Menampilkan halaman edit	Berhasil menampilkan halaman edit	Sesuai
Klik delete	Menghapus pengguna dari daftar pengguna	Berhasil menghapus pengguna	Sesuai

Pada tabel IV.3 merupakan tabel pengujian halaman Kelola pengguna pada admin, Pada tabel diatas terdapat beberapa pengujian diantaranya adalah melakukan pengujian tombol show, edit dan delete, dimana hasil pengujian dapat berjalan sesuai yang diharapkan.

### 4. Pengujian Halaman Kelola Bengkel pada Admin

Tabel IV.4 Pengujian Halaman Kelola Bengkel

<b>Kasus dan Hasil Uji</b>			
<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Klik show	Menampilkan halaman detail	Berhasil menampilkan halaman detail	Sesuai
Klik edit	Menampilkan halaman edit	Berhasil menampilkan halaman edit	Sesuai
Klik delete	Menghapus bengkel dari daftar bengkel	Berhasil menghapus bengkel	Sesuai

Pada tabel IV.4 merupakan tabel pengujian halaman Kelola bengkel pada admin, Pada tabel diatas terdapat beberapa pengujian diantaranya adalah melakukan pengujian tombol show, edit dan delete, dimana hasil pengujian dapat berjalan sesuai yang diharapkan.

5. Pengujian Halaman Kelola Transaksi pada Admin

Tabel IV.5 Pengujian Halaman Kelola Transaksi

<b>Kasus dan Hasil Uji</b>			
<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Klik show	Menampilkan halaman detail	Berhasil menampilkan halaman detail	Sesuai
Klik delete	Menghapus transaksi dari daftar transaksi	Berhasil menghapus transaksi	Sesuai

Pada tabel IV.5 merupakan tabel pengujian halaman Kelola transaksi pada admin, Pada tabel diatas terdapat beberapa pengujian diantaranya adalah melakukan pengujian tombol show dan delete, dimana hasil pengujian berjalan sesuai yang diharapkan.

6. Pengujian Halaman Utama atau Home Pengguna Pada Android

Tabel IV.6 Pengujian Halaman Home Pengguna

<b>Kasus dan Hasil Uji</b>			
<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Klik lokasi	Menampilkan halaman lokasi	Berhasil menampilkan halaman detail	Sesuai

Klik service	Menampilkan halaman service	Berhasil menampilkan halaman service	Sesuai
Klik darurat	Menampilkan halaman darurat	Berhasil menampilkan halaman darurat	Sesuai
Klik foto profile	Menampilkan halaman profile	Berhasil menampilkan halaman profile	Sesuai
Klik home	Menampilkan halaman home	Berhasil menampilkan halaman home	Sesuai
Klik riwayat	Menampilkan halaman riwayat	Berhasil menampilkan halaman riwayat	Sesuai

Pada tabel IV.6 merupakan tabel pengujian halaman home pengguna pada android, Pada tabel diatas terdapat beberapa pengujian diantaranya adalah melakukan pengujian tombol lokasi, service, darurat, dan foto profile, home, riwayat dimana hasil pengujian dapat berjalan.

#### 7. Pengujian Halaman Profile Pengguna Pada Android

Tabel IV.7 Pengujian Halaman Profile Pengguna

<b>Kasus dan Hasil Uji</b>			
<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Klik foto profile	Menampilkan halaman gallery	Berhasil menampilkan halaman gallery	Sesuai
Klik back	Kembali ke halaman home	Berhasil kembali ke halaman home	Sesuai
Input nama	Melakukan validasi inputan	Berhasil melakukan validasi	Sesuai

Input telepon	Melakukan validasi inputan	Berhasil melakukan validasi	Sesuai
Input email	Melakukan validasi inputan	Berhasil melakukan validasi	Sesuai
Input alamat	Melakukan validasi inputan	Berhasil melakukan validasi	Sesuai
Input tanggal lahir	Melakukan validasi inputan	Berhasil melakukan validasi	Sesuai
Klik update	Melakukan update pada database	Berhasil melakukan pembaharuan	Sesuai

Pada tabel IV.7 merupakan tabel pengujian halaman profile pengguna, Pada tabel diatas terdapat beberapa pengujian diantaranya adalah melakukan pengujian tombol back, foto profile, update dimana hasil pengujian dapat berjalan sesuai yang diharapkan. Terdapat juga pengujian inputan nama, telepon, email, alamat, tanggal lahir dimana hasil pengujian dapat berjalan sesuai yang diharapkan.

#### 8. Pengujian Halaman pada Fitur Lokasi Pengguna Pada Android

Tabel IV.8 Pengujian Halaman pada Fitur Lokasi Pengguna

<b>Kasus dan Hasil Uji</b>			
<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Klik titik lokasi saya	Menampilkan titik biru	Berhasil menampilkan titik biru	Sesuai
Klik marker	Menampilkan nama bengkel dan tombol rute	Berhasil menampilkan nama bengkel dan tombol rute	Sesuai



Klik rute	Mengalihkan ke Maps	Berhasil mengalihkan ke Maps	Sesuai
-----------	---------------------	------------------------------	--------

Pada tabel IV.8 merupakan tabel pengujian halaman fitur lokasi pada pengguna, Pada tabel diatas terdapat beberapa pengujian diantaranya adalah melakukan pengujian tombol lokasi, service, darurat, dan foto profile, dimana hasil pengujian dapat berjalan sesuai dengan apa yang diharapkan.

9. Pengujian Halaman Service dan Darurat Pada Android

Tabel IV.9 Pengujian Halaman Service dan Darurat Pada Android

<b>Kasus dan Hasil Uji</b>			
<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Input nama	Melakukan validasi inputan nama	Berhasil melakukan validasi inputan	Sesuai
Input Jenis Kendaraan	Melakukan validasi inputan	Berhasil melakukan validasi inputan	Sesuai
Input Tipe Kendaraan	Melakukan validasi inputan	Berhasil melakukan validasi inputan	Sesuai
Input Tahun Kendaraan	Melakukan validasi inputan	Berhasil melakukan validasi inputan	Sesuai
Input transmisi	Melakukan validasi inputan	Berhasil melakukan validasi inputan	Sesuai
Input lokasi penjemputan	Melakukan validasi inputan	Berhasil melakukan validasi inputan	Sesuai
Klik icon lokasi	Menampilkan halaman place picker	Berhasil menampilkan	Sesuai

		halaman place picker	
Klik mengetahui permasalahan	Menampilkan halaman daftar bengkel	Berhasil menampilkan halaman daftar bengkel	Sesuai
Klik tidak mengetahui permasalahan	Menampilkan halaman daftar bengkel	Berhasil menampilkan halaman daftar bengkel	Sesuai

Pada tabel IV.9 merupakan tabel pengujian halaman service pada android, Pada tabel diatas terdapat beberapa pengujian diantaranya adalah melakukan pengujian input nama, jenis kendaraan, input tipe kendaraan, input tahun kendaraan, input transmisi kendaraan, input lokasi penjemputan melalui tombol, menekan tombol mengetahui dan tidak mengetahui permasalahan, dimana hasil pengujian dapat berjalan sesuai dengan apa yang diharapkan.

#### 10. Pengujian Halaman Daftar Bengkel

Tabel IV.10 Pengujian Halaman Daftar Bengkel

<b>Kasus dan Hasil Uji</b>			
<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Klik item bengkel	Menampilkan halaman daftar service	Berhasil menampilkan halaman daftar service	Sesuai

11. Pengujian Halaman Daftar Service Pada Android

Tabel IV.11 Pengujian Halaman Daftar Service

<b>Kasus dan Hasil Uji</b>			
<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Klik item service	Memasukkan item service ke keranjang	Berhasil memasukkan item service ke keranjang	Sesuai
Klik icon keranjang	Menampilkan halaman keranjang	Berhasil menampilkan halaman keranjang	Sesuai

12. Pengujian Halaman Keranjang Pada Android

Tabel IV.12 Pengujian Halaman Keranjang Saya

<b>Kasus dan Hasil Uji</b>			
<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Klik pesan service	Menyimpan data pesanan service ke database	Berhasil menyimpan data pesanan service ke database	Sesuai

13. Pengujian Halaman Riwayat Pengguna Pada Android

Tabel IV.13 Pengujian Halaman Riwayat Pengguna

<b>Kasus dan Hasil Uji</b>			
<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Klik tab process	Menampilkan halaman process	Berhasil menampilkan halaman process	Sesuai
Klik tab completed	Menampilkan halaman completed	Berhasil menampilkan halaman completed	Sesuai

Klik item process	Menampilkan halaman detail process	Berhasil menampilkan halaman detail process	Sesuai
Klik item completed	Menampilkan halaman detail completed	Berhasil menampilkan halaman detail completed	Sesuai

Pada tabel IV.13 merupakan tabel pengujian halaman riwayat pengguna pada android, Pada tabel diatas terdapat beberapa pengujian diantaranya adalah melakukan pengujian tombol tab process dan completed, menekan salah satu item dimana hasil pengujian dapat berjalan sesuai dengan apa yang diharapkan.

#### 14. Pengujian Halaman Detail Proses Pengguna Pada Android

Tabel IV.14 Pengujian Halaman Detail Proses Pengguna

<b>Kasus dan Hasil Uji</b>			
<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Klik chat	Menampilkan halaman chat	Berhasil menampilkan halaman chat	Sesuai
Klik icon tambah	Menampilkan halaman daftar service	Berhasil menampilkan halaman daftar service	Sesuai
Klik back	Kembali ke halaman sebelumnya	Berhasil kembali ke halaman sebelumnya	Sesuai

Pada tabel IV.14 merupakan tabel pengujian halaman detail proses pengguna pada android, Pada tabel diatas terdapat beberapa pengujian diantaranya adalah melakukan pengujian

tombol chat, tombol tambah, tombol back dimana hasil pengujian dapat berjalan sesuai dengan apa yang diharapkan.

#### 15. Pengujian Halaman Chat Pengguna Pada Android

Tabel IV.15 Pengujian Halaman Chat Pengguna

<b>Kasus dan Hasil Uji</b>			
<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Input message	Melakukan validasi inputan	Berhasil melakukan validasi	Sesuai
Klik send	Mengirim message ke pihak bengkel serta melakukan proses write pada database	Berhasil mengirim message ke pihak bengkel serta melakukan proses write pada database	Sesuai
Klik back	Kembali ke halaman sebelumnya	Mengirim message ke pihak bengkel serta melakukan proses write pada database	Sesuai

Pada tabel IV.15 merupakan tabel pengujian halaman chat pengguna pada android, Pada tabel diatas terdapat beberapa pengujian diantaranya adalah melakukan pengujian tombol send chat, tombol back dan validasi input message dimana hasil pengujian dapat berjalan sesuai dengan apa yang diharapkan.

16. Pengujian Halaman Utama atau Home Bengkel Pada Android

Tabel IV.16 Pengujian Halaman Home Bengkel

<b>Kasus dan Hasil Uji</b>			
<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Klik sign out	Keluar dari akun dan Menampilkan halaman login	Berhasil Keluar dari akun dan Menampilkan halaman login	Sesuai
Klik on/off	Menyimpan state pada penyimpanan local (shared preferences) dan database	Berhasil menyimpan state pada penyimpanan local (shared preferences) dan database	Sesuai
Klik on/off darurat	Menyimpan state pada penyimpanan local (shared preferences) dan database	Berhasil menyimpan state pada penyimpanan local (shared preferences) dan database	Sesuai
Klik profile	Menampilkan halaman profile	Berhasil menampilkan halaman profile	Sesuai
Klik pesanan service	Menampilkan halaman service order	Berhasil menampilkan halaman service order	Sesuai
Klik service	Menampilkan halaman service price	Berhasil menampilkan halaman service price	Sesuai

Pada tabel IV.16 merupakan tabel pengujian halaman home pengguna pada android, Pada tabel diatas terdapat beberapa pengujian diantaranya adalah melakukan pengujian

tombol sign out, profile, service order, service dan tombol on/off dimana hasil pengujian dapat berjalan kecuali tombol darurat.

17. Pengujian Halaman Profile Bengkel pada Android

Tabel IV.17 Pengujian Halaman Profile Bengkel

<b>Kasus dan Hasil Uji</b>			
<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Input nama bengkel	Melakukan validasi inputan	Berhasil Melakukan validasi	Sesuai
Input email	Melakukan validasi inputan	Berhasil Melakukan validasi	Sesuai
Input address	Melakukan validasi inputan	Berhasil Melakukan validasi	Sesuai
Input phone	Melakukan validasi inputan	Berhasil Melakukan validasi	Sesuai
Input operational Hour	Melakukan validasi inputan	Berhasil Melakukan validasi	Sesuai
Input alamat	Melakukan validasi inputan	Berhasil melakukan validasi	Sesuai
Klik checkbox motorcycle	Checkbox tercentang	Checkbox berhasil tercentang	Sesuai
Klik checkbox car	Checkbox tercentang	Checkbox berhasil tercentang	Sesuai
Klik update	Melakukan update pada database	Berhasil melakukan pembaharuan	Sesuai

Pada tabel IV.17 merupakan tabel pengujian halaman profile bengkel, Pada tabel diatas terdapat beberapa pengujian

diantarnya adalah melakukan pengujian tombol update dimana hasil pengujian dapat berjalan sesuai yang diharapkan. Terdapat juga pengujian inputan nama bengkel, telepon, email, alamat, jam operasi dimana hasil pengujian dapat berjalan sesuai yang diharapkan.

18. Pengujian Halaman Service Bengkel pada Android

Tabel IV.18 Pengujian Halaman Service Bengkel

<b>Kasus dan Hasil Uji</b>			
<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Klik tab motor	Menampilkan halaman service untuk motor	Berhasil menampilkan halaman service untuk motor	Sesuai
Klik tab mobil	Menampilkan halaman service untuk mobil	Berhasil Menampilkan halaman service untuk mobil	Sesuai
Klik icon tambah	Menampilkan halaman tambah service	Berhasil menampilkan halaman tambah service	Sesuai
Klik item service	Menampilkan halaman detail service	Berhasil menampilkan halaman detail service	Sesuai

Pada tabel IV.18 merupakan tabel pengujian halaman service price bengkel pada android, Pada tabel diatas terdapat beberapa pengujian diantaranya adalah melakukan pengujian tombol tab motorcycle dan car, mengklik tombol tambah, dan menekan item service dimana hasil pengujian dapat berjalan sesuai dengan apa yang diharapkan.



19. Pengujian Halaman Tambah Service dan Detail Service

Tabel IV.19 Pengujian Halaman Tambah Service dan Detail Service

<b>Kasus dan Hasil Uji</b>			
<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Input nama service	Melakukan validasi inputan	Berhasil melakukan validasi inputan	Sesuai
Input harga jasa service	Melakukan validasi inputan	Berhasil melakukan validasi inputan	Sesuai
Input harga service	Melakukan validasi inputan	Berhasil melakukan validasi inputan	Sesuai
Input kendaraan	Melakukan validasi inputan	Berhasil melakukan validasi inputan	Sesuai
Input merek kendaraan	Melakukan validasi inputan	Berhasil melakukan validasi inputan	Sesuai
Klik pilih gambar	Manampilkan halaman galery	Berhasil menampilkan halaman galery	Sesuai
Klik upload service	Menyimpan data service ke database	Berhasil menyimpan data service ke database	Sesuai

20. Pengujian Halaman Service Order Begkel pada Android

Tabel IV.20 Pengujian Halaman Service Order Bengkel

<b>Kasus dan Hasil Uji</b>			
<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Klik tab confirm	Menampilkan halaman confirm	Berhasil menampilkan halaman confirm	Sesuai

Klik tab process	Menampilkan halaman process	Berhasil menampilkan halaman process	Sesuai
Klik tab completed	Menampilkan halaman completed	Berhasil menampilkan halaman completed	Sesuai
Klik item confirm	Menampilkan halaman detail confirm	Berhasil menampilkan halaman detail confirm	Sesuai
Klik item process	Menampilkan halaman detail process	Berhasil menampilkan halaman detail process	Sesuai
Klik item completed	Menampilkan halaman detail completed	Berhasil menampilkan halaman detail completed	Sesuai

Pada tabel IV.21 merupakan tabel pengujian Halaman service order bengkel pengguna pada android, Pada tabel diatas terdapat beberapa pengujian diantaranya adalah melakukan pengujian tombol tab confirm, process dan completed, menekan salah satu item dimana hasil pengujian dapat berjalan sesuai dengan apa yang diharapkan.

21. Pengujian Halaman Detail Confirm Bengkel pada Android

Tabel IV.21 Pengujian Halaman Detail Confirm Bengkel

<b>Kasus dan Hasil Uji</b>			
<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Klik accept	Menyimpan data pesanan ke database	Berhasil menyimpan data pesanan ke database	Sesuai

Klik reject	Menghapus data pesanan di database	Berhasil menghapus data pesanan di database	Sesuai
-------------	------------------------------------	---	--------

Pada tabel IV.21 merupakan tabel pengujian halaman detail confirm begkel pada android, Pada tabel diatas terdapat 2 pengujian diantaranya adalah melakukan pengujian tombol accept dan reject dimana hasil pengujian dapat berjalan sesuai dengan apa yang diharapkan.

## 22. Pengujian Halaman Detail Process Bengkel pada Android

Tabel IV.22 Pengujian Halaman Detail Process Bengkel

<b>Kasus dan Hasil Uji</b>			
<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Klik chat	Menampilkan halaman chat	Berhasil menampilkan halaman chat	Sesuai
Klik tempat service	Menampilkan dialog konfirmasi tempat service	Berhasil menampilkan dialog konfirmasi tempat service	Sesuai
Klik rute lokasi	Menampilkan rute melalui Maps	Berhasil Menampilkan rute melalui Maps	Sesuai
Klik selesai	Meyimpan data pesanan service pada database	Berhasil meyimpan data pesanan service pada database	sesuai

Pada tabel IV.22 merupakan tabel pengujian halaman detail process bengkel pada android, Pada tabel diatas terdapat beberapa pengujian diantaranya adalah melakukan pengujian tombol chat, tempat service, go to location, selesai dimana hasil pengujian dapat berjalan sesuai dengan apa yang diharapkan.

23. Pengujian Halaman Chat Bengkel pada Android

Tabel IV.23 Pengujian Halaman Chat Bengkel

<b>Kasus dan Hasil Uji</b>			
<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Input message	Melakukan validasi inputan	Berhasil melakukan validasi	Sesuai
Klik send	Mengirim message ke pihak pemesan serta melakukan proses write pada database	Berhasil mengirim message ke pihak pemesan serta melakukan proses write pada database	Sesuai
Klik back	Kembali ke halaman sebelumnya	Mengirim message ke pihak bengkel serta melakukan proses write pada database	Sesuai

Pada tabel IV.23 merupakan tabel pengujian halaman chat bengkel pada android, Pada tabel diatas terdapat beberapa pengujian diantaranya adalah melakukan pengujian tombol send chat, tombol back dan validasi input message dimana hasil pengujian dapat berjalan sesuai dengan apa yang diharapkan.

24. Rekapitulasi Hasil Pengujian Black Box

Tabel IV.24 Rekapitulasi Hasil Pengujian Black Box

<b>Kesimpulan dan Hasil Uji</b>			
<b>Interface yang diuji</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Halaman Login Admin	Dapat tampil halaman dengan fungsi	Berhasil tampil halaman dengan fungsi login dan	Sesuai

	login dan pengecekan user admin	pengecekan admin berjalan dengan baik	
Menu Navigation Sidebar pada Admin	Menampilkan halaman dashboard, users, workshops, transakctions, atau advertisement	Berhasil pindah halaman dashboard, users, workshops, transakctions, atau advertisement	Sesuai
Halaman Kelola Pengguna pada Admin	Melakukan kelola pengguna seperti show, edit dan delete.	Berhasil melakukan kelola pengguna seperti show, edit dan delete.	Sesuai
Halaman Kelola Bengkel pada Admin	Melakukan kelola bengkel seperti show, edit dan delete.	Berhasil melakukan kelola bengkel seperti show, edit dan delete.	Sesuai
Halaman Kelola Transaksi pada Admin	Melakukan kelola transaksi seperti show, edit dan delete.	Berhasil melakukan kelola transaksi seperti show, edit dan delete.	Sesuai
Halaman Utama atau Home Pengguna Pada Android	Menampilkan halaman lokasi, service dan darurat	Berhasil menampilkan halaman lokasi, service dan darurat	Sesuai
Halaman Profile Pengguna Pada Android	Mengambil data dari database dan menampilkan data pengguna	Berhasil mengambil data dari database dan menampilkan data pengguna	Sesuai
Halaman pada Fitur Lokasi Pengguna Pada Android	Menampilkan titik-titik lokasi bengkel terdekat	Berhasil menampilkan titik-titik lokasi bengkel terdekat	Sesuai

Halaman Service Pada Android	Melakukan valisasi input nama, jenis kendaraan, merek kendaraan, tahun, transmisi, lokasi penjemputan, serta mengujian tombol mengetahui permasalahan dan tombol tidak mengetahui permasalahan	Berhasil melakukan valisasi input nama, jenis kendaraan, merek kendaraan, tahun, transmisi, lokasi penjemputan, serta berhasil meguji tombol mengetahui permasalahan dan tombol tidak mengetahui permasalahan	Sesuai
Halaman Daftar Bengkel	Manampilkan daftar bengkel	Berhasil menampilkan daftar bengkel	Sesuai
Halaman daftar service	Memasukkan item service ke keranjang dan menampilkan halaman keranjang	Berhasil memasukkan item service ke keranjang dan menampilkan halaman keranjang	Sesuai
Halaman keranjang	Menyimpan data pesanan service ke database	Berhasil menyimpan data pesanan service ke database	Sesuai
Halaman Riwayat Pengguna Pada Android	Menampilkan list pesanan pada tahapan proses atau selesai	Berhasil menampilkan list pesanan pada tahapan proses atau selesai	Sesuai
Halaman Detail Proses	Menampilkan data pemesan,	Berhasil menampilkan	Sesuai

Pengguna Pada Android	listpesanan service, dan total harga service	data pemesan, list pesanan service, dan total harga service	
Halaman Chat Pengguna Pada Android	Menampilkan daftar pesan dan dapat mengirim pesan	Berhasil menampilkan daftar pesan dan mengirim pesan	Sesuai
Halaman Utama atau Home Bengkel Pada Android	Dapat menampilkan halaman dengan fungsi pindah activity atau halaman ke profile, service order, dan service price. Menyimpan state on/off pada penyimpanan local dan firebase database	Berhasil menampilkan halaman dengan fungsi pindah activity atau halaman ke profile, service order, dan service price. Berhasil menyimpan state on/off pada penyimpanan local dan firebase database	Sesuai
Halaman Profile Bengkel pada Android	Mengambil data dari database dan menampilkan data bengkel serta menjalankan fungsi update untuk memperbaharui data bengkel	Berhasil mengambil data dari database dan menampilkan data bengkel serta menjalankan fungsi update untuk memperbaharui data bengkel	Sesuai
Halaman Service	Menampilkan daftar service	Berhasil menampilkan	Sesuai

Bengkel pada Android	motor dan mobil	daftar service motor dan mobil	
Halaman tambah service dan detail service	Menyimpan data service dan mengupdate data service pada database	Berhasil menyimpan data service dan mengupdate data service pada database	Sesuai
Halaman Service Order Bengkel pada Android	Menampilkan daftar service order pada tab confirm, process atau completed	Berhasil menampilkan list pesanan pada tahapan proses atau selesai	Sesuai
Halaman Detail Confirm Bengkel pada Android	Menampilkan konfirmasi Tindakan apakah orderan service diterima atau ditolak. Jika diterima orderan service akan disimpan didatabase, jika ditolak akan dihapus	Berhasil menampilkan konfirmasi Tindakan apakah orderan service diterima atau ditolak. Jika diterima orderan service akan disimpan didatabase, jika ditolak akan dihapus	Sesuai
Halaman Detail Process Bengkel pada Android	Menampilkan data pemesan service, nomor telepon, lokasi, dan inputan nota	Berhasil menampilkan data pemesan service, nomor telepon, lokasi, dan inputan nota	Seusai
Halaman Chat Bengkel pada Android	Menampilkan daftar pesan dan dapat mengirim pesan	Berhasil menampilkan daftar pesan dan mengirim pesan	Sesuai



Halaman Nota Bengkel pada Android	Menyimpan list nota berdasarkan inputan yang dimasukkan pada database	Berhasil menyimpan list nota berdasarkan inputan yang dimasukkan pada database	Sesuai
-----------------------------------	---	--	--------

Pada tabel diatas dapat dilihat hasil pengujian yang dilakukan dengan menggunakan metode pengujian Black Box dengan menguji setiap fungsi pada halaman baik pada website database dan pada aplikasi android. Terdapat 25 interface yang dilakukan pengujian menggunakan metode Black Box dan di dapatkan hasil pengujian sesuai dengan hasil yang diharapkan.

#### IV.1.2 Mengintegrasikan Website dengan Firebase Realtime Database

Agar website dapat terhubung dengan Firebase Realtime Database diperlukan sebuah SDK yaitu Firebase Admin SDK For PHP. Firebase Admin SDK For PHP adalah sebuah SDK (Software Development Kit) untuk dapat mengakses firebase service menggunakan bahasa pemograman PHP. Pengkonfigurasiian dapat menggunakan perintah composer require krait/firebase-php.

```
composer require krait/firebase-php
```

Gambar IV.28 Perintah Konfogurasi Firebase Admin SDK For PHP

Autentikasi akun layanan dan perizinan akses layanan Firebase membutuhkan file private key dalam format JSON. untuk mendapatkan file private key dapat diakses melalui Firebase Console.

createAuth() digunakan untuk apapun yang berhubungan dengan sistem autentikasi, createDatabase() digunakan untuk melaukan proses CRUD (Create, Read, Update, dan Delete) pada firebase realtime database dan createStorage() digunakan untuk

menyimpan dan mengakses file atau gambar pada firebase realtime database

```
public function __construct(){
    $factory = (new Factory)
    ->withServiceAccount(__DIR__.'/config-firebase.json')
    ->withDatabaseUri('https://e-service-694c5-default-rtdb.firebaseio.com/');

    $this->auth = $factory->createAuth();
    $this->database = $factory->createDatabase();
    $this->storage = $factory->createStorage();
}
```

Gambar IV.29 Script Integrasi Antara Website dan Firebase Realtime Database

#### IV.1.3 Script Penggunaan Metode Haversine Formula pada Android

##### 1. Menentukan titik awal pengguna

```
private var resultLocation: AddressData? = null
```

Gambar IV.30 Deklarasi Variabel resultLocation

Pada gambar diatas terdapat pendeklarasian variable dengan nama resultLocation dengan tipe AddressData. Variable ini digunakan untuk menampung data titik lokasi pengguna.

```
private fun getLastLocationData() {
    val locationRequest = LocationRequest()
    locationRequest.priority = LocationRequest.PRIORITY_HIGH_ACCURACY
    locationRequest.interval = 0
    locationRequest.fastestInterval = 0
    locationRequest.numUpdates = 1
    fusedLocationProviderClient = LocationServices.getFusedLocationProviderClient(requireActivity())

    if (ActivityCompat.checkSelfPermission(
        requireContext(),
        Manifest.permission.ACCESS_FINE_LOCATION
    ) != PackageManager.PERMISSION_GRANTED || ActivityCompat.checkSelfPermission(
        requireContext(),
        Manifest.permission.ACCESS_COARSE_LOCATION
    ) != PackageManager.PERMISSION_GRANTED
    ) {
        // TODO: Consider calling
        // ActivityCompat#requestPermissions
        // here to request the missing permissions, and then overriding
        // public void onRequestPermissionsResult(int requestCode, String[] permissions,
        // int[] grantResults)
        // to handle the case where the user grants the permission. See the documentation
        // for ActivityCompat#requestPermissions for more details.
        return
    }
    fusedLocationProviderClient.requestLocationUpdates(
        locationRequest, locationCallback, Looper.myLooper() as Looper
    )
}

private val locationCallback = object : LocationCallback() {
    override fun onLocationResult(p0: LocationResult) {
        val lastLocation: Location = p0.lastLocation
        Log.d("MyService.SimpleHome", "Your last location: ${lastLocation.latitude}, ${lastLocation.longitude}")
        intentPlacePicker(lastLocation)
    }
}
```

Gambar IV.31 Script Untuk Mendapatkan Lokasi Terakhir

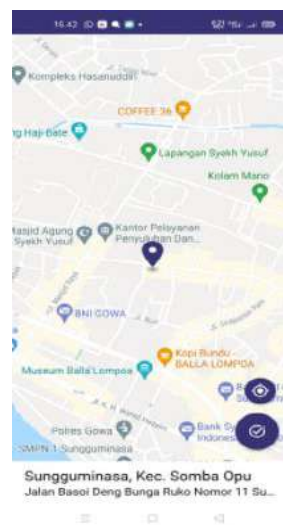
Pada gambar diatas menampilkan script function `lasLocationData`. Function ini akan dijalankan untuk mendapatkan lokasi terakhir GPS pada smartphone. Hasil nilainya akan ditampung pada variable `lastLocation`.

```
private fun intentPlacePicker(location: Location){
    val mIntent = PlacePicker.IntentBuilder()
        .setLatLng(location.latitude, location.longitude) // Initial Latitude and Longitude the Map will load into
        .setMapZoom(16.0f) // Map Zoom Level. Default: 14.0
        .setAddressRequired(true) // Set If return only Coordinates if cannot fetch Address for the coordinates. Default: True
        .setMarkerDrawable(R.drawable.ic_marker) // Change the default Marker Image
        .setMarkerImageImageColor(R.color.purple_200)
        .setFillColor(R.color.purple_200)
        .setPrimaryTextColor(R.color.black) // Change text color of Shortened Address
        .setSecondaryTextColor(R.color.black) // Change text color of full Address
        .setBottomViewColor(R.color.white) // Change Address View Background Color (Default: White)
        .setMapType(MapType.NORMAL)
        .build(requireActivity())
    startActivityForResult(mIntent, Constants.PLACE_PICKER_REQUEST)
}

override fun onActivityResult(requestCode: Int, resultCode: Int, data: Intent?) {
    if (requestCode == Constants.PLACE_PICKER_REQUEST) {
        if (resultCode == Activity.RESULT_OK) {
            resultLocation = data?.getParcelableExtra(Constants.ADDRESS_INTENT)!!
            Toast.makeText(activity, text = "$resultLocation", Toast.LENGTH_SHORT).show()
        }
    } else {
        super.onActivityResult(requestCode, resultCode, data)
    }
}
}
```

Gambar IV.32 Script Mengambil Lokasi Pengguna Secara

Pada gambar diatas menampilkan sebuah function `intentPlacePicker`, dimana function ini akan digunakan untuk memindahkan activity (halaman) ke halaman Place Picker seperti gambar dibawah.



Gambar IV.33 Halaman Place Picker

Pada gambar diatas terdapat marker yang dapat digeser atau dicustom sesuai lokasi pengguna. Variable resultLocation akan berisi nilai sesuai titik pada marker yang ditentukan.

## 2. Mengitung jarak dengan menggunakan metode Haversine Formula

```
object CalculateDistance {  
    //Metode Haversine Formula  
    fun getDistance(  
        currentLatitude: Double,  
        currentLongitude: Double,  
        destinationLatitude: Double,  
        destinationLongitude: Double  
    ): Double {  
        val earthRadius = 6371  
        val dLatitude = Math.toRadians(destinationLatitude - currentLatitude)  
        val dLongitude = Math.toRadians(destinationLongitude - currentLongitude)  
        val a = sin(x: dLatitude / 2) * sin(x: dLatitude / 2) +  
            cos(Math.toRadians(currentLatitude)) *  
            cos(Math.toRadians(destinationLatitude)) *  
            sin(x: dLongitude / 2) * sin(x: dLongitude / 2)  
        val c = 2 * atan2(sqrt(a), sqrt(x: 1 - a))  
        return earthRadius * c  
    }  
}
```

Gambar IV.34 Script Implementasi Metode Haversine Formula

Pada gambar diatas terdapat object CalculateDistance yang didalamnya terdapat function getDistance. Function ini lah yang menerapkan penggunaan metode Haversine Formula dimana function getDistance menerima 4 parameter yaitu currentLatitude, currentLongitude, destinationLatitude, dan destinationLongitude. currentLatitude dan currentLongitude adalah nilai latitude dan longitude lokasi pengguna, sedangkan destinationLatitude dan destinationLongitude adalah nilai latitude dan longitude lokasi bengkel.

### 3. Mengambil Semua Data Lokasi Bengkel

```
private val listWorkshop= ArrayList<Workshop>()
private fun readWorkshopData(){
    val workshopListener = object : ValueEventListener {
        override fun onDataChange(snapshot: DataSnapshot) {
            if (snapshot.exists()){
                for (i in snapshot.getChildren()){
                    val data = i.getValue(Workshop::class.java) as Workshop
                    val calculateDistance = resultLocation?.let { @AddressData
                        CalculateDistance.getDistance(
                            it.latitude, it.longitude, data.latitude!!, data.longitude!!)
                    }

                    if (calculateDistance != null) {
                        if (calculateDistance <= 1.5){
                            if(data.motorcycle == true){
                                Log.d( tag= "MyService-$simpleName", msg= "id workshop: ${data.id} - $calculateDistance")
                                ListWorkshop.add(data)
                            }
                        }
                    }
                }

                if (ListWorkshop.isNotEmpty()) {
                    Log.d( tag= "MyService-datal", listWorkshop.toString())
                    readStateData(ListWorkshop)
                } else {
                    dialogLoading.dismiss()
                    Toast.makeText(requireContext(), text= "Can't Find Workshop", Toast.LENGTH_LONG).show()
                }
            } else {
                dialogLoading.dismiss()
                Toast.makeText(context, text= "Data Not Found", Toast.LENGTH_SHORT).show()
            }
        }
    }

    @Override fun onCancelled(error: DatabaseError) {
        dialogLoading.dismiss()
        Log.d( tag= "MyService-$simpleName-error", error.message)
    }
}
@ReadAllData(db.workshopRef, workshopListener)
```

Gambar IV.35 Script Untuk Mengambil Semua Data Lokasi Bengkel

Pada gambar diatas menampilkan script untuk mengambil semua data lokasi bengkel. Di script ini melakukan perulangan untuk menghitung jarak antara lokasi pengguna dengan bengkel, jika jaraknya kurang dari sama dengan 3 Km maka data akan ditampung pada variable listWorkshop dan akan ditampilkan dengan memanggil variable tersebut.

#### IV.1.4 Pengujian Metode Haversine

Pengujian akan membandingkan perhitungan manual dan perhitungan pada source code pada aplikasi (Gambar IV.35) dengan menggunakan Metode Haversine Formula. Titik awal berada di Universitas Fajar Makassar dan titik tujuan di titik bengkel A, bengkel B, bengkle C, bengkel D, dan bengkel E. Setiap

bengkel memiliki lokasi yang berbeda sehingga setiap bengkel memiliki nilai latitude dan longitude yang berbeda. Nilai inilah yang menjadi acuan untuk pengujian Metode Haversine Formula.

1. Perhitungan Universitas Fajar ke Bengkel A

Universitas Fajar:

$$\text{Lat1} = -5.1492722$$

$$\text{Lng1} = 119.4500857$$

Bengkel Service Mobil Panggilan 24 Jam Makassar (Jl.

Veteran Selatan No.67):

$$\text{Lat2} = -5.1607714$$

$$\text{Lng2} = 119.4220339$$

$$\Delta \text{lat} = \frac{\pi}{180} \times ((-5.1607714) - (-5.1492722))$$

$$\Delta \text{lat} = -2.0069890134532108E - 4$$

$$\Delta \text{lng} = \frac{\pi}{180} \times (119.4220339 - 119.4500857)$$

$$\Delta \text{lng} = -4.895962711109465E - 4$$

$$a = \sin^2 \frac{-2.0069890134532108E-4}{2} \times \cos(-5.1492722) \times$$

$$\cos(-5.1607714) \times \sin^2 \frac{-4.895962711109465E-4}{2}$$

$$= 6.951234438071952E - 8$$

$$c = 2 \times \text{atan2} (\sqrt{6.951234438071952E - 8},$$

$$\sqrt{1 - 6.951234438071952E - 8})$$

$$= 5.273038819935803E - 4$$

$$d = 6371 \times 5.273038819935803E - 4$$

$$= 3.35 \text{ Km}$$

2. Perhitungan Universitas Fajar ke Bengkel B

Universitas Fajar:

$$\text{Lat1} = -5.1492722$$

$$\text{Lng1} = 119.4500857$$

Bengkel Mobil Rizky (Jl. Maccini Gusung No.101)

$$\text{Lat2} = -5.1398677$$

$$\text{Lng2} = 119.4268102$$

$$\Delta\text{lat} = \frac{\pi}{180} \times ((-5.1398677) - (-5.1492722))$$

$$\Delta\text{lat} = 1.6413948950381514E - 4$$

$$\Delta\text{lng} = \frac{\pi}{180} \times (119.4268102 - 119.4500857)$$

$$\Delta\text{lng} = -4.06234110047886E - 4$$

$$\begin{aligned} a &= \sin^2 \frac{1.6413948950381514E-4}{2} \times \cos(-5.1492722) \times \\ &\quad \cos(-5.1398677) \times \sin^2 \frac{-4.06234110047886E-4}{2} \\ &= 4.766025516426517E - 8 \end{aligned}$$

$$\begin{aligned} c &= 2 \times \text{atan2} (\sqrt{4.766025516426517E - 8}, \\ &\quad \sqrt{1 - 4.766025516426517E - 8}) \\ &= 4.366245798002299E - 4 \end{aligned}$$

$$\begin{aligned} d &= 6371 \times 4.366245798002299E - 4 \\ &= 2.78 \text{ Km} \end{aligned}$$

### 3. Perhitungan Universitas Fajar ke Bengkel C

Universitas Fajar:

$$\text{Lat1} = -5.1492722$$

$$\text{Lng1} = 119.4500857$$

Bengkel Berlian (Jl. Tinumbu VII)

$$\text{Lat2} = -5.1180832$$

$$\text{Lng2} = 119.4296878$$

$$\Delta\text{lat} = \frac{\pi}{180} \times ((-5.1180832) - (-5.1492722))$$

$$= 5.443507404045176E - 4$$

$$\begin{aligned}\Delta lng &= \frac{\pi}{180} \times (119.4296878 - 119.4500857) \\ &= -3.560105154926574E - 4\end{aligned}$$

$$\begin{aligned}a &= \sin^2 \frac{5.443507404045176E-4}{2} \times \cos(-5.1492722) \times \\ &\quad \cos(-5.1180832) \times \sin^2 \frac{-3.560105154926574E-4}{2} \\ &= 1.0551160255727028E - 7\end{aligned}$$

$$\begin{aligned}c &= 2 \times \operatorname{atan2}(\sqrt{1.0551160255727028E - 7}, \\ &\quad \sqrt{1 - 1.0551160255727028E - 7}) \\ &= 6.496510025180287E - 4\end{aligned}$$

$$\begin{aligned}d &= 6371 \times 6.496510025180287E - 4 \\ &= 4.13 \text{ Km}\end{aligned}$$

#### 4. Perhitungan Universitas Fajar ke Bengkel D

Universitas Fajar

Lat1 = -5.1492722

Lng1 = 119.4500857

Bengkel panggil makassar maros (Jl. Gardenia Raya)

Lat2 = -5.1512782

Lng2 = 119.4692274

$$\begin{aligned}\Delta lat &= \frac{\pi}{180} \times ((-5.1512782) - (-5.1492722)) \\ &= -3.501130479500154E - 5\end{aligned}$$

$$\begin{aligned}\Delta lng &= \frac{\pi}{180} \times (119.4692274 - 119.4500857) \\ &= 3.34085689428852E - 4\end{aligned}$$

$$\begin{aligned}a &= \sin^2 \frac{-3.501130479500154E-5}{2} \times \cos(-5.1492722) \times \\ &\quad \cos(-5.1512782) \times \sin^2 \frac{3.34085689428852E-4}{2}\end{aligned}$$



$$= 2.7984905198797705E - 8$$

$$c = 2 \times \operatorname{atan2}(\sqrt{2.7984905198797705E - 8}, \sqrt{1 - 2.7984905198797705E - 8})$$

$$= 3.3457379132173124E - 4$$

$$d = 6371 \times 3.3457379132173124E - 4$$

$$= 2.13 \text{ Km}$$

##### 5. Perhitungan Universitas Fajar ke Bengkel E

Universitas Fajar

$$\text{Lat1} = -5.1492722$$

$$\text{Lng1} = 119.4500857$$

Moses Jaya Motor (Lorong 2, Kapasa, Kec. Tamalanrea)

$$\text{Lat2} = -5.1185736$$

$$\text{Lng2} = 119.5051679$$

$$\Delta \text{lat} = \frac{\pi}{180} \times ((-5.1185736) - (-5.1492722))$$

$$= 5.357916457527316E - 4$$

$$\Delta \text{lng} = \frac{\pi}{180} \times (119.5051679 - 119.4500857)$$

$$= 9.613657492420394E - 4$$

$$a = \sin^2 \frac{5.357916457527316E - 4}{2} \times \cos(-5.1492722) \times \cos(-5.1185736) \times \sin^2 \frac{9.613657492420394E - 4}{2}$$

$$= 3.0097400692073424E - 7$$

$$c = 2 \times \operatorname{atan2}(\sqrt{3.0097400692073424E - 7}, \sqrt{1 - 3.0097400692073424E - 7})$$

$$= 0.001097222014208349$$

$$d = 6371 \times 0.001097222014208349$$

$$= 6.99 \text{ Km}$$

IV.1.5 Perbandingan Jarak Pada Pengujian metode Haversine secara manual

Tabel IV.25 Hasil Pengujian Metode Haversine

No	Lokasi Awal (Latitude, Longitude)	Lokasi Akhir (Latitude, Longitude)	Hasil	
			Perhitungan Manual	Perhitungan Otomatis
1	Universitas Fajar Makassar (-5.1492722, 119.4500857)	Bengkel Service Mobil Panggilan 24 Jam Makassar, Jl. Veteran Selatan No.67 (-5.1607714, 119.4220339)	3.35 Km	3.35 Km
2		Bengkel Mobil Rizky, Jl. Maccini Gusung No.101 (-5.1398677, 119.4268102)	2.78 Km	2.78 Km
3		Bengkel Berlian, JL. Tinumbu VII (-5.1180832, 119.4296878)	4.13 Km	4.13 Km
4		Bengkel panggil makassar maros, Jl. Gardenia Raya (-5.1512782, 119.4692274)	2.13 Km	2.13 Km
5		Moses Jaya Motor, Lorong 2, Kapasa, Kec. Tamalanrea (-5.1185736, 119.5051679)	6.99 Km	6.99 Km

Berdasarkan tabel diatas dapat diperhatikan bahwa hasil dari perhitungan manual sama dengan hasil perhitungan otomatis yang berada di source code aplikasi dengan menggunakan rumus metode Haversine Formula. Ini membuktikan bahwa rumus metode Haversine Formula pada aplikasi berjalan dengan baik.

IV.1.6 Perbandingan Jarak Pada Aplikasi Dengan *Website*  
[www.movable-type.co.uk](http://www.movable-type.co.uk)

Tabel IV.26 Hasil Pengujian Metode Haversine

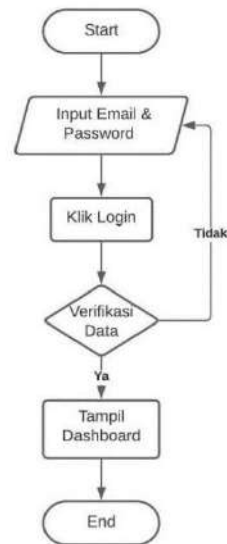
No	Lokasi Awal	Lokasi Tujuan	Jarak Pada Aplikasi	Jarak Pada <i>Website</i>	Selisih Jarak
1.	Universitas Fajar Makassar	Bengkel Service Mobil Panggilan 24 Jam Makassar (Jl. Veteran Selatan No.67)	3.3 Km	3.3 Km	0 Km
2.		Bengkel Mobil Rizky (Jl. Maccini Gusung No.101)	2.7 Km	2.7 Km	0 Km
3.		Bengkel Berlian (Jl. Tinumbu VII)	4.1 Km	4.1 Km	0 Km
4.		Bengkel panggil makassar maros (Jl. Gardenia Raya)	2.1 Km	2.1 Km	0 Km
5.		Moses Jaya Motor (Lorong 2, Kapasa, Kec. Tamalanrea)	6.9 Km	6.9 Km	0 Km
Total Selisih Jarak					0 Km

Berdasarkan tabel diatas dapat diperhatikan bahwa selisih jarak yang didapatkan oleh aplikasi menggunakan Metode Haversine memiliki jarak sama dengan jarak yang ada pada website [www.movable-type.co.uk](http://www.movable-type.co.uk). Berdasarkan tabel diatas dapat dihitung nilai rata-rata selisih jarak dari titik awal ke 5 lokasi tujuan dimana hasilnya adalah 0 Km yang membuktikan penggunaan Metode Haversine Formuula berjalan dengan baik pada aplikasi.

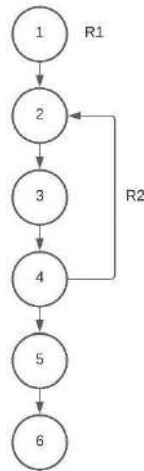
#### IV.1.7 Pengujian White Box

White Box Testing merupakan teknik menguji aplikasi dengan menganalisa kode program pada sisi logika apakah program memiliki kesalahan atau tidak. Jika kode telah di produksi kedalam output dan tidak memenuhi persyaratan maka kode akan di kompilasi hingga mencapai hasil yang diharapkan. Pada pengujian white box akan digunakan teknik basis path.

##### 1. Pengujian White Box Login Pengguna, Bengkel, dan Admin



Gambar IV.36 Flowchart Login Pengguna, Bengkel, dan Admin



Gambar IV.37 Flowgraph Login Pengguna, Bengkel, dan Admin

Dari *flowgraph login* pemain diatas terdapat **6 node**, **6 edge** dan **2 region**. Kompleksitas Siklomatis (pengukuran kuantitatif kompleksitas logis suatu program) dari *flowgraph* dapat diperoleh dengan perhitungan berikut.

$$V(G) = E - N + 2$$

Dimana:

E = jumlah *edge* pada *flowgraph* ditandakan dengan gambar panah

N = jumlah *node* pada *flowgraph* ditandakan dengan gambar lingkaran

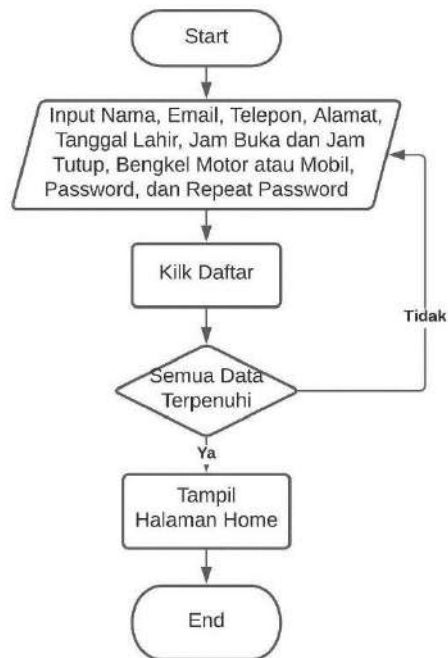
Sehingga kompleksitas siklomatis,

$$V(G) = 6 - 6 + 2 = 2$$

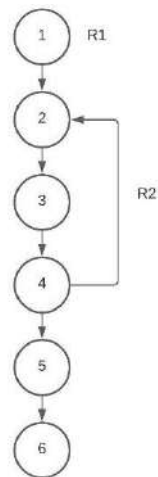
Tabel IV.27 Skenario *Test Case Login* Pengguna, Bengkel, dan Admin

Path	1
Jalur	1-2-3-4-5-6
Skenario	<ul style="list-style-type: none"> <li>- Start</li> <li>- <i>Input email dan password</i></li> <li>- Klik <i>login</i></li> <li>- Data benar</li> <li>- Menampilkan data akun</li> <li>- End</li> </ul>
Hasil pengujian	Berhasil
Path	2
Jalur	1-2-3-4-2-3-4-5-6
Skenario	<ul style="list-style-type: none"> <li>- Start</li> <li>- <i>Input email dan password</i></li> <li>- Klik <i>login</i></li> <li>- Data Salah</li> <li>- Kembali <i>input email dan password</i></li> <li>- Klik <i>login</i></li> <li>- Data benar</li> <li>- Menampilkan <i>dashboard</i> Admin</li> <li>- End</li> </ul>
Hasil pengujian	Berhasil

## 2. Pengujian White Box Daftar Akun Bengkel



Gambar IV.38 Daftar Akun Bengkel



Gambar IV.39 Flowgraph Daftar Akun Bengkel

Dari flowgraph login pemain diatas terdapat **6 node**, **6 edge** dan **2 region**. Kompleksitas Siklomatis (pengukuran kuantitatif kompleksitas logis suatu program) dari flowgraph dapat diperoleh dengan perhitungan berikut.

$$V(G) = E - N + 2$$

Dimana:

E = jumlah edge pada flowgraph ditandakan dengan gambar panah

N = jumlah node pada flowgraph ditandakan dengan gambar lingkaran

Sehingga kompleksitas siklomatis,

$$V(G) = 6 - 6 + 2 = 2$$

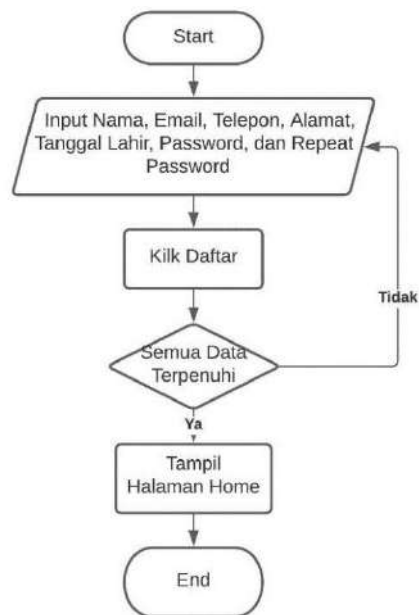
Tabel IV.28 Skenario Test Case Daftar Akun Bengkel

Path	1
Jalur	1-2-3-4-5-6
Skenario	<ul style="list-style-type: none"> <li>- Start</li> <li>- Input Nama, <i>Email</i>, Telepon, Alamat, Tanggal Lahir, Jam Buka dan Jam Tutup, Bengkel Motor atau Mobil, <i>Password</i>, dan <i>Repeat Password</i></li> <li>- Klik <i>sign up</i></li> <li>- Semau data terpenuhi</li> <li>- Menampilkan halaman <i>home</i></li> <li>- End</li> </ul>
Hasil pengujian	Berhasil
Path	2
Jalur	1-2-3-4-2-3-4-5-6
Skenario	<ul style="list-style-type: none"> <li>- Start</li> <li>- Input Nama, <i>Email</i>, Telepon, Alamat, Tanggal Lahir, Jam Buka dan Jam Tutup, Bengkel Motor atau Mobil, <i>Password</i>, dan <i>Repeat Password</i></li> <li>- Klik <i>sign up</i></li> <li>- Semau data tidak terpenuhi</li> <li>- Input Nama, <i>Email</i>, Telepon, Alamat, Tanggal Lahir, Jam Buka dan Jam Tutup, Bengkel Motor atau Mobil, <i>Password</i>, dan <i>Repeat Password</i></li> <li>- Klik <i>sign up</i></li> <li>- Semau data terpenuhi</li> </ul>

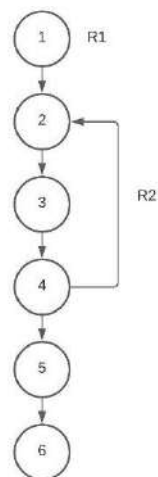


	- Menampilkan halaman <i>home</i> - End
Hasil pengujian	Berhasil

### 3. Pengujian White Box Daftar Pengguna



Gambar IV.40 Flowchart Daftar Pengguna



Gambar IV.41 Flowgraph Daftar Pengguna

Dari flowgraph login pemain diatas terdapat **6 node**, **6 edge** dan **2 region**. Kompleksitas Siklomatis (pengukuran

kuantitatif kompleksitas logis suatu program) dari flowgraph dapat diperoleh dengan perhitungan berikut.

$$V(G) = E - N + 2$$

Dimana:

E = jumlah edge pada flowgraph ditandakan dengan gambar panah

N = jumlah node pada flowgraph ditandakan dengan gambar lingkaran

Sehingga kompleksitas siklomatis,

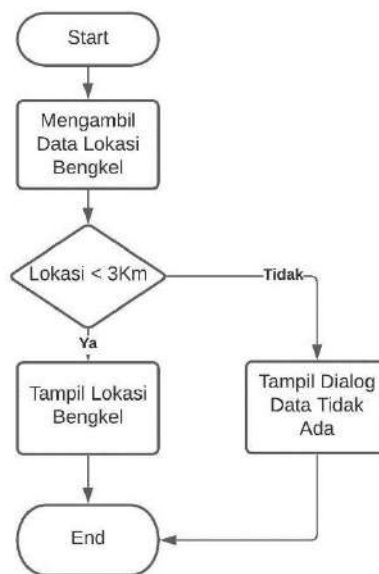
$$V(G) = 6 - 6 + 2 = 2$$

Tabel IV.29 Skenario *Test Case* Daftar Pengguna

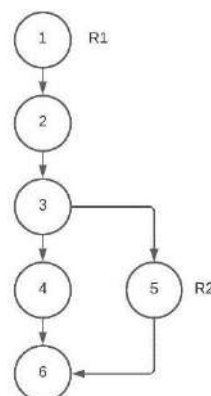
Path	1
Jalur	1-2-3-4-5-6
Skenario	<ul style="list-style-type: none"> <li>- Start</li> <li>- <i>Input</i> nama, email, telepon, alamat, tanggal lahir, <i>password</i>, dan <i>repeat password</i></li> <li>- Klik <i>sign up</i></li> <li>- Semau data terpenuhi</li> <li>- Menampilkan halaman <i>home</i></li> <li>- End</li> </ul>
Hasil pengujian	Berhasil
Path	2
Jalur	1-2-3-4-2-3-4-5-6
Skenario	<ul style="list-style-type: none"> <li>- Start</li> <li>- <i>Input</i> nama, email, telepon, alamat, tanggal lahir, <i>password</i>, dan <i>repeat password</i></li> <li>- Klik <i>sign up</i></li> <li>- Semau data tidak terpenuhi</li> <li>- <i>Input</i> nama, email, telepon, alamat, tanggal lahir, <i>password</i>, dan <i>repeat password</i></li> <li>- Klik <i>sign up</i></li> <li>- Semau data terpenuhi</li> <li>- Menampilkan halaman <i>home</i></li> <li>- End</li> </ul>

Hasil pengujian	Berhasil

#### 4. Pengujian White Box Titik Lokasi Bengkel Terdekat



Gambar IV.42 Flowchart Titik Lokasi Bengkel Terdekat



Gambar IV.43 Flowgraph Titik Lokasi Bengkel Terdekat

Dari *flowgraph* login pemain diatas terdapat **6 node**, **6 edge** dan **2 Region**. Kompleksitas Siklomatis (pengukuran

kuantitatif kompleksitas logis suatu program) dari *flowgraph* dapat diperoleh dengan perhitungan berikut.

$$V(G) = E - N + 2$$

Dimana:

E = jumlah *edge* pada *flowgraph* ditandakan dengan gambar panah

N = jumlah *node* pada *flowgraph* ditandakan dengan gambar lingkaran

Sehingga kompleksitas siklomatis,

$$V(G) = 6 - 6 + 2 = 2$$

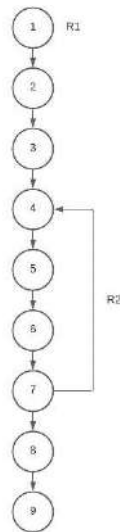
Tabel IV.30 Skenario Test Case Titik Lokasi Bengkel Terdekat

Path	1
Jalur	1-2-3-4-6
Skenario	<ul style="list-style-type: none"> <li>- Start</li> <li>- Mengambil data lokasi bengkel di database</li> <li>- Data Ada</li> <li>- Menampilkan titik lokasi bengkel terdekat</li> <li>- End</li> </ul>
Hasil pengujian	Berhasil
Path	2
Jalur	1-2-3-5-6
Skenario	<ul style="list-style-type: none"> <li>- Start</li> <li>- Mengambil data lokasi bengkel di database</li> <li>- Data tidak ada</li> <li>- Menampilkan dialog bahwa lokasi bengkel tidak ada disekitar pengguna</li> <li>- End</li> </ul>
Hasil pengujian	Berhasil

5. Pengujian White Box Pemesanan Service di Fitur Service dan Darurat (Tidak Mengetahui Permasalahan)



Gambar IV.44 Flowchart Pemesanan Service di Fitur Service dan Darurat (Tidak Mengetahui Permasalahan)



Gambar IV.45 Flowgraph Pemesanan Service di Fitur Service dan Darurat (Tidak Mengetahui Permasalahan)

Dari *flowgraph login* pemain diatas terdapat **9 node**, **9 edge** dan **2 Region**. Kompleksitas Siklomatis (pengukuran kuantitatif kompleksitas logis suatu program) dari *flowgraph* dapat diperoleh dengan perhitungan berikut.

$$V(G) = E - N + 2$$

Dimana:

E = jumlah *edge* pada *flowgraph* ditandakan dengan gambar panah

N = jumlah *node* pada *flowgraph* ditandakan dengan gambar lingkaran

Sehingga kompleksitas siklomatis,

$$V(G) = 9 - 9 + 2 = 2$$

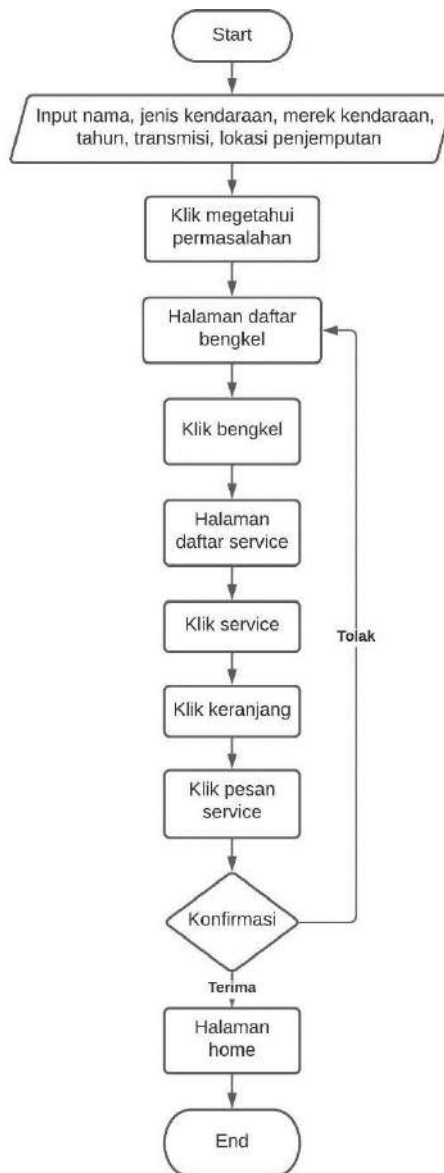
Tabel IV.31 Skenario Test Case Pemesanan Service di Fitur Service dan Darurat (Tidak Mengetahui Permasalahan)

Path	1
Jalur	1-2-3-4-5-6-7-8-9

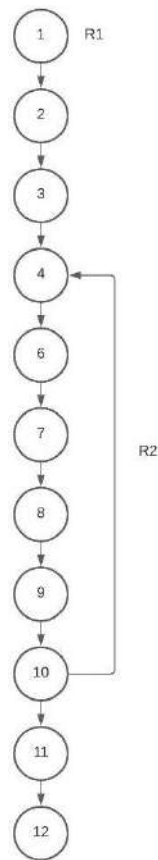
Skenario	<ul style="list-style-type: none"> <li>- Start</li> <li>- <i>Input</i> nama, jenis kendaraan, merek kendaraan, tahun, transmisi, lokasi penjemputan</li> <li>- Klik mengetahui permasalahan</li> <li>- Menampilkan halaman daftar bengkel</li> <li>- Pilih bengkel yang diinginkan</li> <li>- Menampilkan tarif harga ke lokasi berdasarkan jarak</li> <li>- Klik pesan service</li> <li>- Pesanan service diterima</li> <li>- Menampilkan halaman home</li> <li>- End</li> </ul>
Hasil pengujian	Berhasil
Path	2
Jalur	1-2-3-4-5-6-7-4-5-6-7-8-9
Skenario	<ul style="list-style-type: none"> <li>- Start</li> <li>- <i>Input</i> nama, jenis kendaraan, merek kendaraan, tahun, transmisi, lokasi penjemputan</li> <li>- Klik mengetahui permasalahan</li> <li>- Menampilkan halaman daftar bengkel</li> <li>- Pilih bengkel yang diinginkan</li> <li>- Menampilkan tarif harga ke lokasi berdasarkan jarak</li> <li>- Klik pesan service</li> <li>- Pesanan service ditolak</li> <li>- Menampilkan halaman daftar bengkel</li> <li>- Pilih bengkel yang diinginkan</li> <li>- Menampilkan tarif harga ke lokasi berdasarkan jarak</li> <li>- Klik pesan service</li> <li>- Pesanan service diterima</li> <li>- Menampilkan halaman home</li> <li>- End</li> </ul>
Hasil pengujian	Berhasil

6. Pengujian White Box Pemesanan Service di Fitur Service dan Darurat (Mengetahui Permasalahan)





Gambar IV.46 Flowchart Pemesanan Service di Fitur Service dan Darurat (Mengetahui Permasalahan)



Gambar IV.47 Flowgraph Pemesanan Service di Fitur Service dan Darurat (Mengetahui Permasalahan)

Dari *flowgraph login* pemain diatas terdapat **12 node**, **11 edge** dan **2 Region**. Kompleksitas Siklomatis (pengukuran kuantitatif kompleksitas logis suatu program) dari *flowgraph* dapat diperoleh dengan perhitungan berikut.

$$V(G) = E - N + 2$$

Dimana:

E = jumlah *edge* pada *flowgraph* ditandakan dengan gambar panah

N = jumlah *node* pada *flowgraph* ditandakan dengan gambar lingkaran

Sehingga kompleksitas siklomatis,

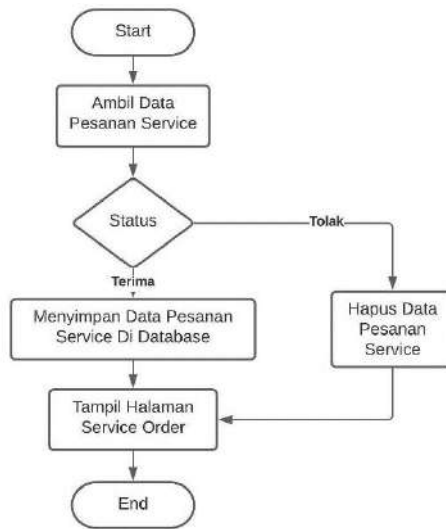
$$V(G) = 12 - 11 + 2 = 3$$

Tabel IV.32 Skenario Test Case Pemesanan Service di Fitur Service dan Darurat (Mengetahui Permasalahan)

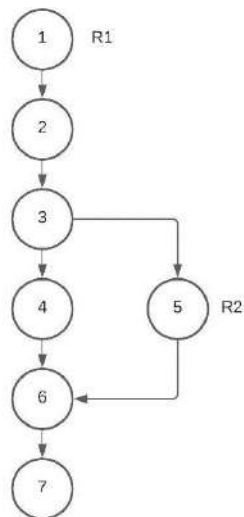
Path	1
Jalur	1-2-3-4-5-6-7-8-9-10-11-12
Skenario	<ul style="list-style-type: none"> <li>- Start</li> <li>- <i>Input</i> nama, jenis kendaraan, merek kendaraan, tahun, transmisi, lokasi penjemputan</li> <li>- Klik mengetahui permasalahan</li> <li>- Menampilkan halaman daftar bengkel</li> <li>- Pilih bengkel yang diinginkan</li> <li>- Menampilkan halamann daftar service</li> <li>- Klik service</li> <li>- Klik keranjang</li> <li>- Klik pesan service</li> <li>- Pesanan service diterima</li> <li>- Menampilkan halaman home</li> <li>- End</li> </ul>
Hasil pengujian	Berhasil
Path	2
Jalur	1-2-3-4-5-6-7-8-9-10-4-5-6-7-8-9-10-11-12
Skenario	<ul style="list-style-type: none"> <li>- Start</li> <li>- <i>Input</i> nama, jenis kendaraan, merek kendaraan, tahun, transmisi, lokasi penjemputan</li> <li>- Klik mengetahui permasalahan</li> <li>- Menampilkan halaman daftar bengkel</li> <li>- Pilih bengkel yang diinginkan</li> <li>- Menampilkan halamann daftar service</li> <li>- Klik service</li> <li>- Klik keranjang</li> <li>- Klik pesan service</li> <li>- Pesanan service ditolak</li> <li>- Menampilkan halaman daftar bengkel</li> <li>- Pilih bengkel yang diinginkan</li> <li>- Menampilkan halamann daftar service</li> <li>- Klik service</li> <li>- Klik keranjang</li> <li>- Klik pesan service</li> <li>- Pesanan service diterima</li> </ul>

	- Menampilkan halaman home - End
Hasil pengujian	Berhasil

7. Pengujian White Box Konfirmasi Pesanan Service Pada Sisi Bengkel



Gambar IV.48 Flowchart Konfirmasi Pesanan Service Pada Sisi Bengkel



Gambar IV.49 Flowgraph Konfirmasi Pesanan Service Pada Sisi Bengkel

Dari flowgraph login pemain diatas terdapat **7 node**, **7 edge** dan **2 Region**. Kompleksitas Siklomatis (pengukuran kuantitatif kompleksitas logis suatu program) dari flowgraph dapat diperoleh dengan perhitungan berikut.

$$V(G) = E - N + 2$$

Dimana:

E = jumlah edge pada flowgraph ditandakan dengan gambar panah

N = jumlah node pada flowgraph ditandakan dengan gambar lingkaran

Sehingga kompleksitas siklomatis,

$$V(G) = 7 - 7 + 2 = 2$$

Tabel IV.33 Skenario Test Case Konfirmasi Pesanan Service Pada Sisi Bengkel

Path	1
Jalur	1-2-3-4-6-7
Skenario	<ul style="list-style-type: none"> <li>- Start</li> <li>- Mengambil data pesanan service di database</li> <li>- Status diterima</li> <li>- Menyimpan data pesanan service di database</li> <li>- Tampil halaman <i>service order</i></li> <li>- End</li> </ul>
Hasil pengujian	Berhasil
Path	2
Jalur	1-2-3-5-6-7
Skenario	<ul style="list-style-type: none"> <li>- Start</li> <li>- Mengambil data pesanan service di database</li> <li>- Status ditolak</li> <li>- Menghapus data pesanan service di database</li> <li>- Tampil halaman <i>service order</i></li> <li>- End</li> </ul>
Hasil pengujian	Berhasil

## 8. Rekapitulasi Hasil Pengujian *White Box*

Tabel IV.34 Hasil Pengujian *White Box*

<b>Kesimpulan dan Hasil Uji</b>				
<b>Fungsi</b>	<b>Kompleksitas Siklomatis, V(G)</b>	<b>Region (R)</b>	<b>Path</b>	<b>Hasil</b>
Login Pengguna, Bengkel, dan Admin	2	2	2	Berhasil
Daftar Akun Bengkel	2	2	2	Berhasil
Daftar Pengguna	2	2	2	Berhasil
Mencari titik Lokasi Bengkel Terdekat	2	2	2	Berhasil
Pemesanan Service pada fitur Service dan Darurat (Tidak Mengetahui Permasalahan)	2	2	2	Berhasil
Pemesanan Service pada fitur Service dan Darurat (Mengetahui Permasalahan)	3	2	2	Berhasil
Konfirmasi Pesanan Service Pada Sisi Bengkel	2	2	2	Berhasil

Berdasarkan tabel rekapitulasi hasil pengujian white box diatas, didapatkan hasil dengan program berhasil menjalankan scenario test case untuk pengujian fungsi logika bisnis pada

sistem yang setidaknya dilakukan sebanyak minimal 1 kali pengujian

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **V.1 Kesimpulan**

1. Aplikasi pemesanan service kendaraan pada bengkel Berbasis Android yang dibangun berhasil melakukan sistem pemesanan service kendaraan dari pengguna ke bengkel terdekat. Aplikasi android dibangun menggunakan IDE Android Studio dengan Bahasa pemograman Kotlin dan basis data aplikasi menggunakan Firebase Database.
2. Sistem pemetaan dibuat dengan menggunakan batuan maps yang akan menampilkan peta digital. Dipeta inilah dibuat sebuah fitur place picker yang digunakan untuk menginput titik lokasi bengkel pada saat registrasi. Data longitude dan latitude lokasi akan disimpan di firebase database sebagai acuan untuk lokasi bengkel.
3. Aplikasi berhasil menghitung jarak dari titik pengguna ke titik lokasi bengkel dengan menggunakan metode Haversine Formula lalu akan diolah dan hanya akan mengambil lokasi bengkel terdekat dari titik pengguna atau pemesan.

#### **V.2 Saran**

Dari hasil penelitian aplikasi pemesanan service kendaraan pada bengkel berbasis android dengan menggunakan metode Haversine Formula untuk menemukan lokasi bengkel terdekat dari lokasi pengguna atau pemesan.

1. Peneliti berharap agar dapat dilakukan pengembangan tentang sistem pemesanan service kendaraan antara bengkel dan pengguna.
2. Melakukan pencarian bengkel terdekat dengan menggunakan metode atau algoritma selain metode yang peneliti gunakan.



## DAFTAR PUSTAKA

- Arifin, M. F., Marisa, F., & Wijaya, I. D. (2016). Implementasi Google Speech Untuk Penentuan Level Pembelajaran Iqro' Berbasis Android. 16-21.
- Damanik, C. K. (2019). ANALISIS PENGEMBANGAN USAHA BENGKEL MOTOR DI JALAN. 1-15.'
- Dennis.A, W. B. (2012). *Systems Analysis & Design. 5th edition. P.504. John Wiley & Sons.*
- Farid, & Yunus, Y. (2017). ANALISA ALGORITMA HAVERSINE FORMULA UNTUK PENCARIAN. *ILKOM Jurnal Ilmiah*, 353-355.
- Fatansyah. (2015). *Basis Data*. Bandung: Informatika Bandung.
- Hasibuan, M. S. (2011). *Manajemen Sumber Daya Manusia*. Jakarta: PT Bumi Askara.
- Ifan Purnawan, S., Marisa, F., & Wijaya, I. D. (2018). Aplikasi Pencarian Pariwisata Dan Tempat Oleh Oleh Terdekat Menggunakan Metode Haversine. *Jurnal Informatika Merdeka Pasuruan*, 9-16.
- Jony. (2015). Aplikasi Informasi Akademik Berbasis Android . *Jurnal SISFOKOM*, 42-48.
- Kurniawan, H. (2015). Perancangan Sistem Informasi Bengkel Mobil Berbasis Web. 636-641.
- Pasaribu, A. F., Darwis, D., Irawan, A., & Ade, S. (2019). SISTEM INFORMASI GEOGRAFIS UNTUK PENCARIAN LOKASI. *Jurnal TEKNOKOMPAK*, 1-6.
- Rozy, M. F., Nugroho, A. P., & Nurcholis, M. (2017). Aplikasi Pelayanan dan Pengolahan Data Bengkel Secara Elektronik Berbasis Web. *Jurnal Informatika Merdeka Pasuruan*, 54-69.
- Rubiono, G., & Yasi, R. M. (2017). Sosialisasi Manajemen Limbah Oli Bengkel Mobil. 5-9.
- Sanadi, E. A., Achmad, A., & Dewiani. (2018). Pemanfaatan Realtime Database di Platform Firebase. 20.26.
- Siagian, & Sondang, P. (2012). *Manajemen Sumber Daya Manusia*. Jakarta: Bumi Aksara.

Xianjun Chen, Z. J. (2017). Restful API Architecture Based on Laravel Framework . *IOP Conf. Series: Journal of Physics: Conf. Series 910 (2017)*.

Yulianto, Ramadiani, & Harsa Kridalaksana, A. (2018). PENERAPAN FORMULA HAVERSINE PADA SISTEM INFORMASI . *Jurnal Ilmiah Ilmu Komputer*, 14-21.

Yulianto, W. (2015). MENENTUKAN JARAK TERDEKAT HOTEL DENGAN METODE HAVERSINE FORMULA.