

BAB IV

HASIL DAN PEMBAHASAAN

4.1 Hasil Penelitian

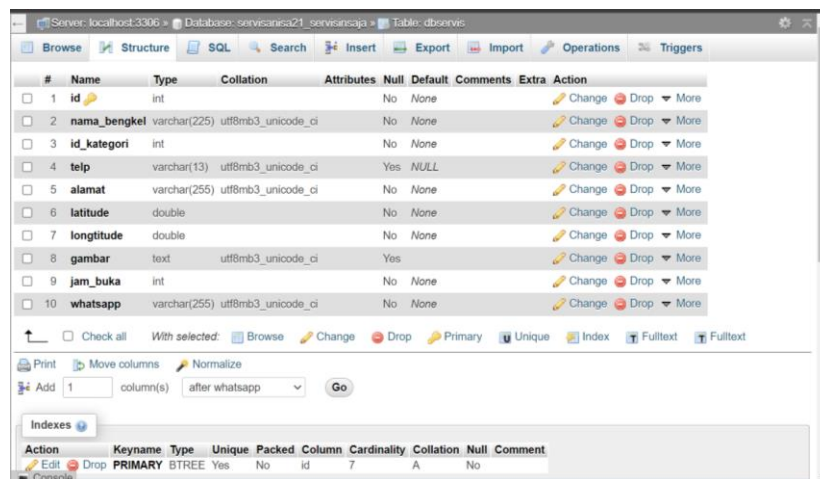
Setelah dilakukan tahapan rancangan, Penelitian ini menghasilkan sebuah aplikasi *marketplace* jasa servis berbasis android yang terdiri atas beberapa menu yang dapat dipilih oleh *user*. Dengan aplikasi ini, *user* dapat menemukan jasa servis dan mendapatkan rincian terkait jasa servis

4.2 Hasil Perancangan Database

Pada perancangan database aplikasi terdapat 5 tabel yaitu sebagai berikut :

1. Tabel servis

Pada table servis terdapat 10 atribut seperti yang terlihat pada gambar 4.1, untuk menampung data rincian penyedia jasa servis

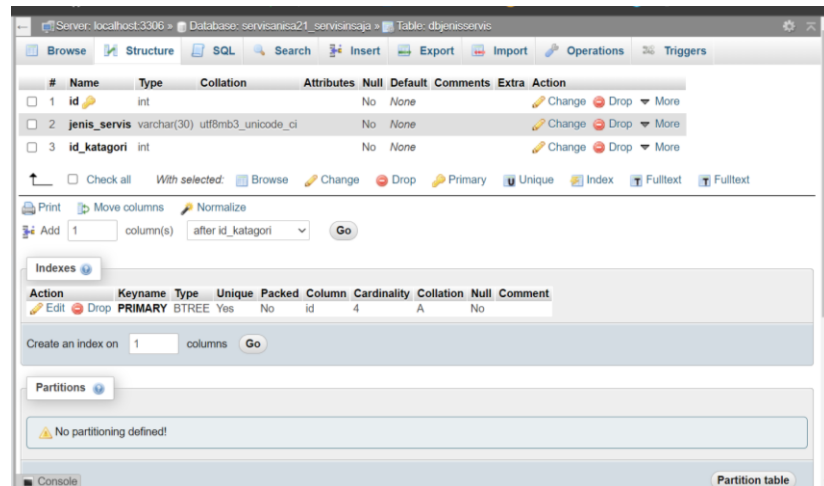


| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|----|--------------|--------------|--------------------|------------|------|---------|----------|-------|------------------|
| 1 | id | int | | | No | None | | | Change Drop More |
| 2 | nama_bengkel | varchar(225) | utf8mb3_unicode_ci | | No | None | | | Change Drop More |
| 3 | id_kategori | int | | | No | None | | | Change Drop More |
| 4 | telp | varchar(13) | utf8mb3_unicode_ci | | Yes | NULL | | | Change Drop More |
| 5 | alamat | varchar(255) | utf8mb3_unicode_ci | | No | None | | | Change Drop More |
| 6 | latitude | double | | | No | None | | | Change Drop More |
| 7 | longitude | double | | | No | None | | | Change Drop More |
| 8 | gambar | text | utf8mb3_unicode_ci | | Yes | | | | Change Drop More |
| 9 | jam_buka | int | | | No | None | | | Change Drop More |
| 10 | whatsapp | varchar(255) | utf8mb3_unicode_ci | | No | None | | | Change Drop More |

Gambar 4.1 Tabel servis

2. Tabel jenis servis

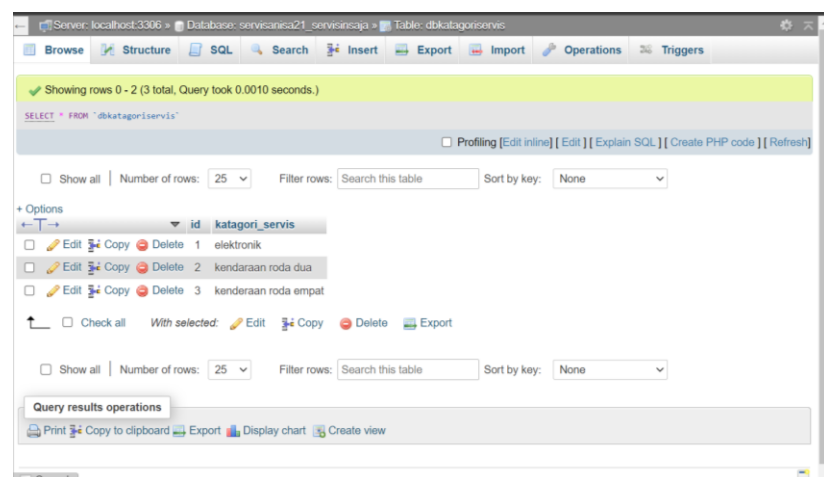
Pada tabel jenis servis terdapat 3 atribut yang terlihat pada gambar 4.2, tabel ini untuk menampung data jenis servis



Gambar 4.2 Tabel jenis servis

3. Tabel kategori servis

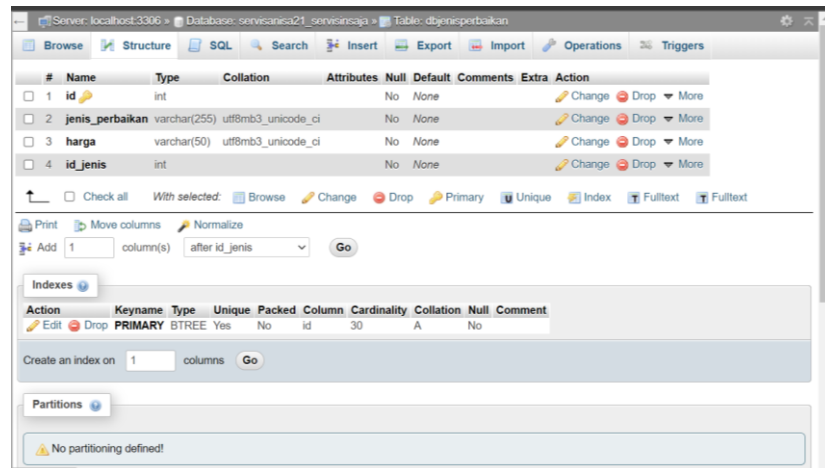
Pada tabel jenis servis terdapat 3 atribut yang terlihat pada gambar 4.3, tabel ini untuk menampung data kategori kendaraan.



Gambar 4.3 Tabel kategori servis

4. Tabel jenis perbaikan

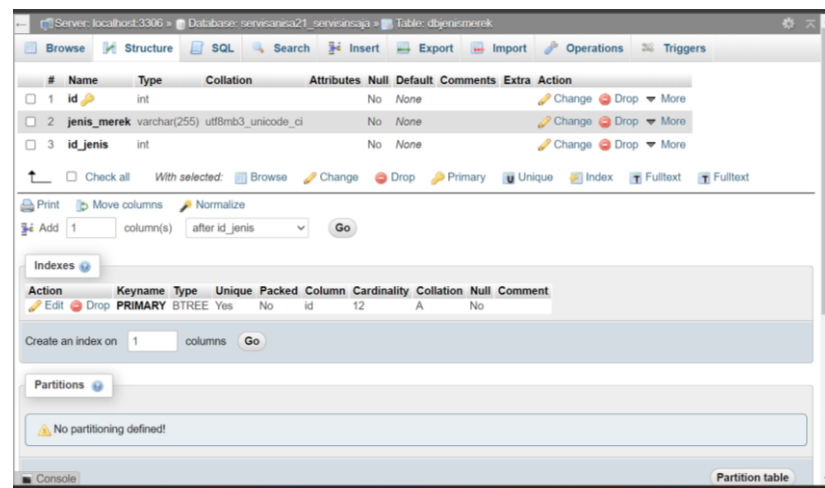
Pada tabel jenis servis terdapat 4 atribut yang terlihat pada gambar 4.4, tabel ini untuk menampung data jenis perbaikan yang dapat dilayani oleh penyedia jasa.



Gambar 4.4 Tabel jenis perbaikan

5. tabel jenis merek

Pada tabel jenis servis terdapat 3 atribut yang terlihat pada gambar 4.5, tabel ini untuk menampung data jenis merek barang yang dapat diservis.



Gambar 4.5 Tabel jenis merek

4.3 Hasil *Interface* Aplikasi

Hasil *Interface* aplikasi *marketplace* jasa servis ini sebagai berikut :

4.3.1 *Interface Halaman Splash Screen*

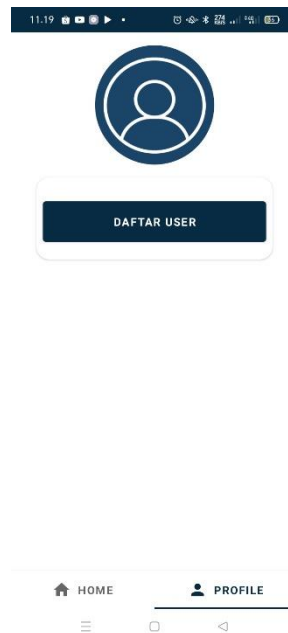
Halaman *Splash Screen* merupakan halaman awal yang menampilkan logo jasa Servisinsaja selama 5 detik, lalu halaman akan berpindah ke halaman login



Gambar 4.6 *Interface Halaman Splash Screen*

4.3.2 *Interface Halaman Profile*

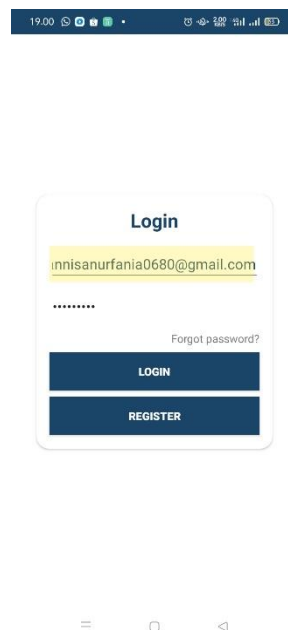
Pada Halaman profile ini *user* dapat *login* atau *sign up*. *User* dapat melakukan pencarian jasa servis tanpa harus *login*, namun untuk melihat halaman detail pada setiap jasa servis *user* di haruskan untuk *login* terlebih dahulu.



Gambar 4.7 *Interface* Halaman *Profil*

4.3.3 *Interface* Halaman *Login*

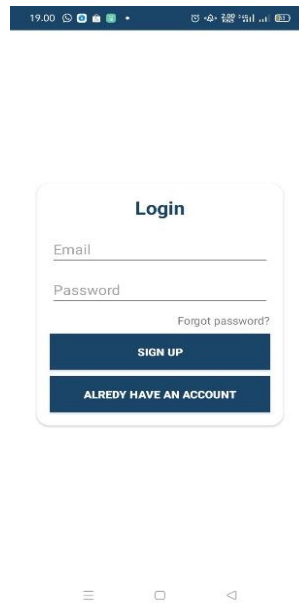
Pada halaman *login*, *user* diharuskan untuk melakukan login terlebih dahulu dengan memasukan email dan password agar dapat masuk ke halaman detail jasa servis



Gambar 4.8 *Interface* Halaman *Login*

4.3.4 *Interface Halaman Registrasi*

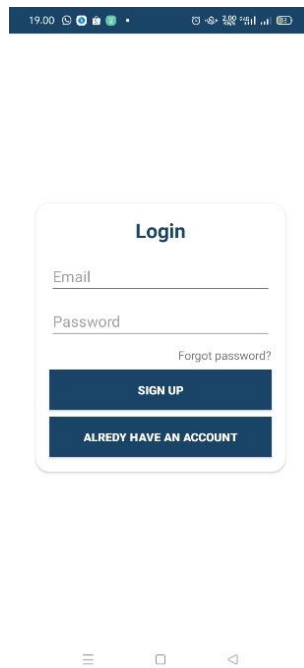
Bagi *user* yang belum memiliki akun, maka *user* harus registrasi terlebih dahulu dengan mendaftarkan email dan password. Setelah registrasi selesai *user* dapat *login* menggunakan akun yang sudah terdaftar



Gambar 4.9 *Interface Halaman Registrasi*

4.3.5 *Interface Halaman Forget Password*

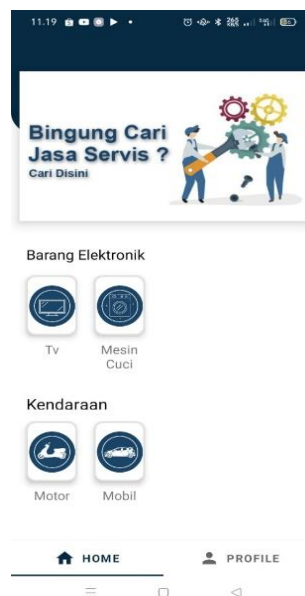
Jika *user* lupa *password*, *user* dapat memanfaatkan fitur *Forget Password* hanya dengan memasukan *e-mail*, maka pesan *Authentication* akan dikirim pada *e-mail* untuk mengubah *password*



Gambar 4.10 *Interface* Halaman *Forget Password*

4.3.6 *Interface* Halaman Utama

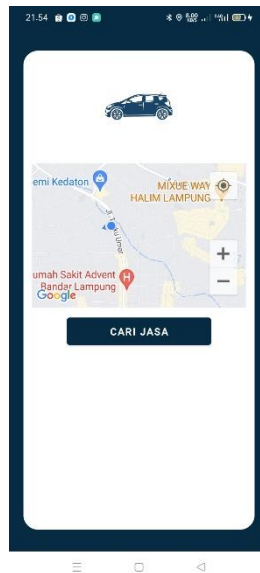
Pada halaman utama ini memiliki beberapa menu pilihan sesuai kebutuhan *user* dalam melakukan pencarian servis yang diinginkan. Terdapat 4 pilihan jasa servis yaitu tv, mesin suci, mobil, dan motor.



Gambar 4.11 *Interface* Halaman Utama

4.3.7 *Interface* Halaman Form Pencarian

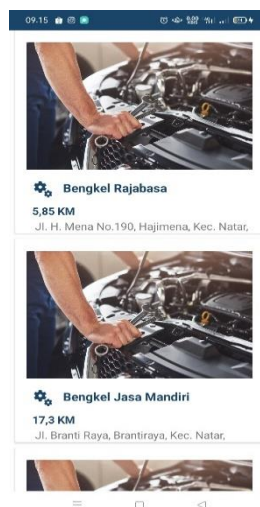
Pada halaman ini *user* wajib memasukan pilihan terhadap merek dan jenis perbaikan. Selanjutnya *user* dapat melakukan pencarian melalui button pencarian jasa



Gambar 4.12 *Interface* Halaman Form Pencarian

4.3.8 *Interface* Halaman Daftar Jasa

Pada halaman ini menampilkan hasil pencarian jasa servis terdekat dengan *user*



Gambar 4.13 *Interface* Halaman daftar jasa

4.3.9 Interface Halaman Detail

Pada halaman ini menampilkan rincian informasi mengenai jasa servis seperti alamat dan jam buka. Selain itu terdapat button yang terhubung ke whatsapp, maps, dan telephone






Gambar 4.14 Interface Halaman Detail

4.4 Hasil Pengujian Perangkat Lunak

Pada tahap ini dilakukan uji kemampuan dan keefektifan suatu sistem. Metode yang digunakan untuk pengujian yaitu menggunakan *Black-box* testing yang bertujuan menemukan kesalahan fungsi dan menyertakan *user* sebagai penguji sebuah sistem. Pengujian dilakukan menggunakan beberapa perangkat *mobile* dengan spesifikasi yang berbeda.

4.4.1 Pengujian Instalasi

Tabel 4.1 Spesifikasi Device

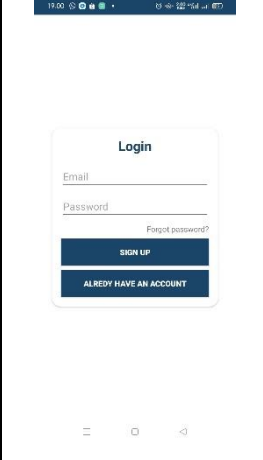
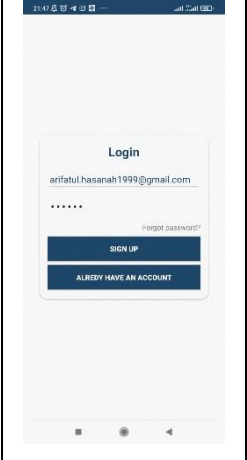




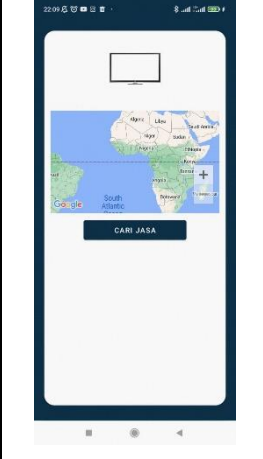
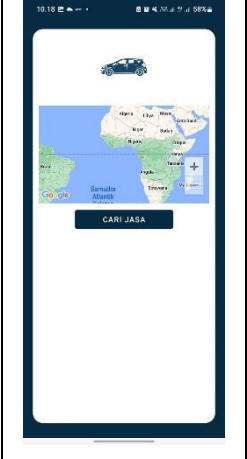

| No | Perangkat | Spesifikasi | Hasil Pengujian | Keterangan |
|----|--------------------------|--|--|------------|
| 1 | Oppo A16 | Processor : MediaTek Helio G35 RAM: 3GB OS : Android 11 Layar : 6,52" |  | Berhasil |
| 2 | Xiomi Nete 10 Pro | Processor : Qualcomm® Snapdragon™ 732G RAM: 8 GB OS : Android 11 Layar : 6,57" |  | Berhasil |
| 3 | Xiomi PocoPhone F1 | Processor : Qualcomm SDM845 Snapdragon 845 RAM: 8 GB OS : Android 10 Layar : 6,18" |  | Berhasil |

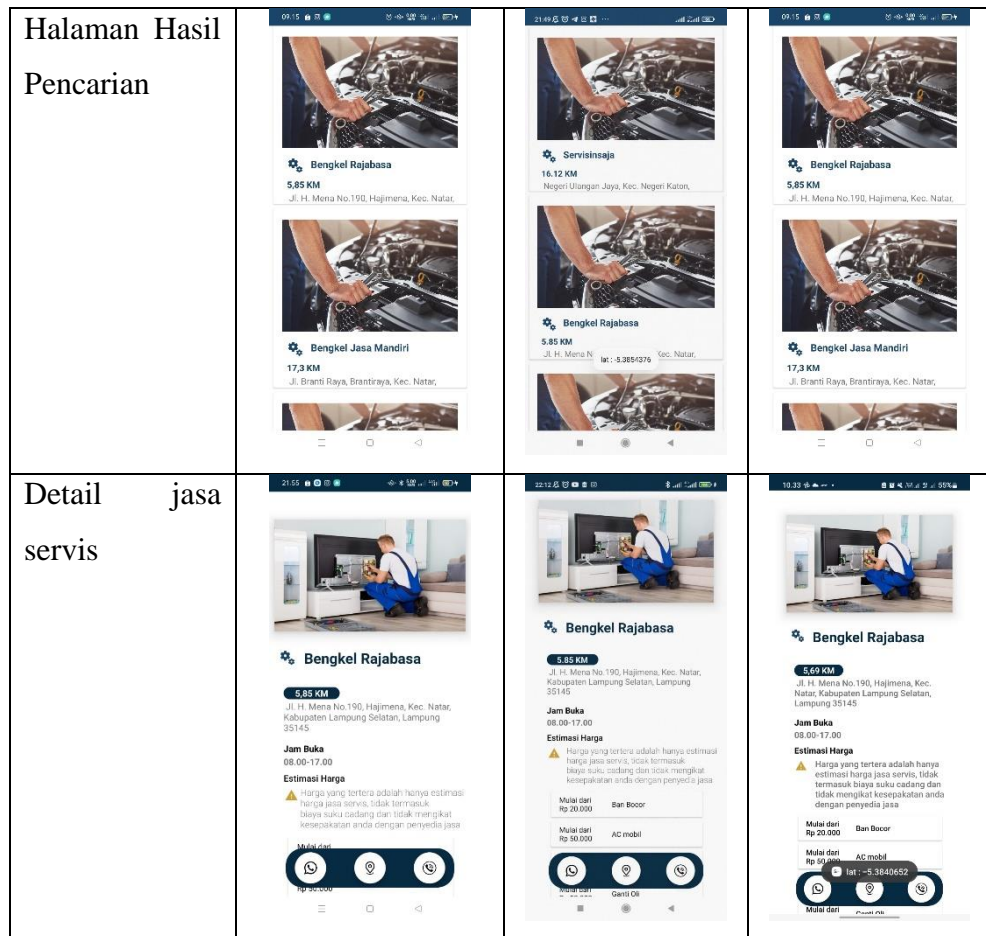
4.4.2 Pengujian Interface

Hasil pengujian menggunakan metode Black Box Testing terdiri dari 3 komponen yaitu uji fungsi menu dan tombol, uji interface, uji kinerja loading dan tingkah laku, uji struktur dan database. Pengujian aplikasi *marketplace* menggunakan metode *Haversine* berbasis mobile dilakukan dengan 3 device yang spesifikasi dan ukuran layar yang berbeda, berikut hasil pengujian interface yang telah dilakukan :

Tabel 4.2 Pengujian Interface

| Proses | ScreenShot Interface | | |
|---------------|----------------------|-----------------|-----------------|
| | <i>Device 1</i> | <i>Device 2</i> | <i>Device 3</i> |
| Splash Screen | | | |
| Login | | | |

| | | | |
|---------------------------|---|--|---|
| <p>Registrasi</p> |  |  |  |
| <p>Menu Utama</p> |  |  |  |
| <p>Menu Form Cari</p> |  |  |  |



4.5 Pembahasan Algoritma Haversine

Aplikasi *Marketplace* jasa servis ini dikembangkan dengan menggunakan algoritma *Haversine* sebagai pencarian jasa terdekat. Algoritma *Haversine* dapat melakukan pencarian jarak terdekat dengan menentukan jarak antara kedua titik, dengan variabel yaitu menggunakan garis lintang (*longitude*) dan garis bujur (*latitude*). Berikut perhitungan dari Algoritma *Haversine* :

a. Lokasi *User* Kantor Direksi PTPN7

Lokasi : Jl. Teuku Umar No.300, Kedaton, Kec. Kedaton, Kota Bandar Lampung, Lampung 35152

Lat 1 : -5,3850964594175

Long 1 : 105,260651245179

b. Lokasi Tujuan Bengkel Rajabasa

Lokasi : Jl. H. Mena No.190, Hajimena, Kec. Natar, Kabupaten Lampung Selatan, Lampung 35145

Lat2 : -5,35288406273102

Long2 : 105,219086372156

c. Merubah derajat ke radian

$$\begin{aligned}\text{Lat1} &= -5.377301952922488 \times 0.0174532925 \text{ radian} \\ &= -0,093987666 \text{ radian}\end{aligned}$$

$$\begin{aligned}\text{Long1} &= 105.2496308689621 \times 0.0174532925 \text{ radian} \\ &= 1,837144988 \text{ radian}\end{aligned}$$

$$\begin{aligned}\text{Lat2} &= -5.352694973089969 \times 0.0174532925 \text{ radian} \\ &= -0,093425454 \text{ radian}\end{aligned}$$

$$\begin{aligned}\text{Long2} &= 105.21910942449055 \times 0.0174532925 \text{ radian} \\ &= 1,836419544 \text{ radian}\end{aligned}$$

d. Mencari nilai X dan Y

$$\begin{aligned}Y &= (\text{Lat2} - \text{Lat1}) \\ &= -0,093422151028 - 0,093851623845 \\ &= -0,190273774873\end{aligned}$$

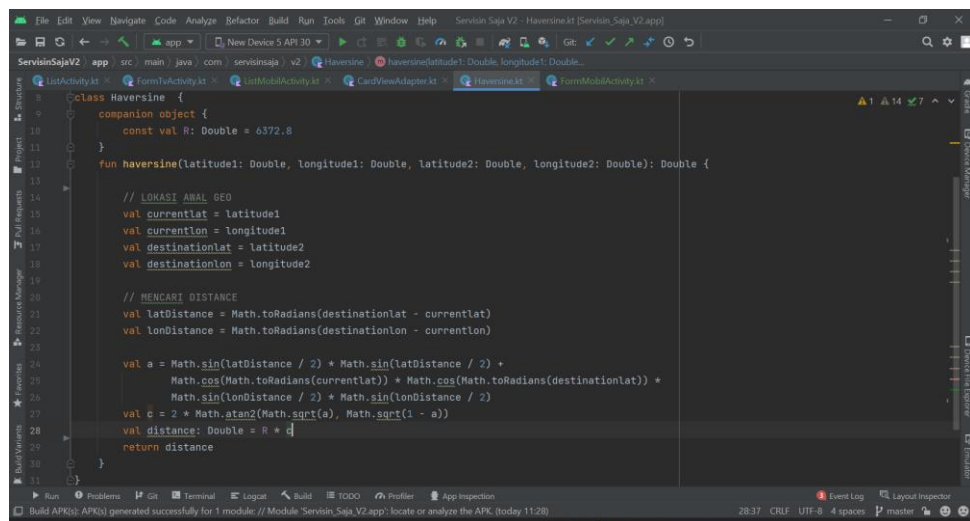
$$\begin{aligned}X &= (\text{long2} - \text{long1}) \times \cos ((\text{lat1} + \text{lat2})/2) \\ &= (0,000725444) \times \cos (-0,18741312 / 2) \\ &= 0,000725444 \times 0,995612752 \\ &= 0,000722261\end{aligned}$$

e. Mencari nilai d (jarak)

$$\begin{aligned}d &= \text{sqrt} ((x*x) + (y*y)) * R \\ &= \text{sqrt} (5,21661 + 3,16082) \times 6372 \\ &= 5,831269089 \text{ KM}\end{aligned}$$

Sehingga berdasarkan contoh diatas di temukan jarak antara lokasi *user* terkini dengan penyedia jasa berjarak

Pada aplikasi *marketplace* jasa servis ini, *Algoritma Haversine* diimplementasikan untuk proses perhitungan jarak berdasarkan jarak antara kedua titik. Sistem akan mengambil data *current location user* dari GPS dan mengambil data *destination location* jasa dari database, sehingga di dapatkan jarak antara *user* dan penyedia jasa. *Source code* penerapan *algoritma Haversine* pada program dapat dilihat pada gambar



```

class Haversine {
    companion object {
        const val R: Double = 6372.8
    }
    fun haversine(latitude1: Double, longitude1: Double, latitude2: Double, longitude2: Double): Double {
        // LOKASI AWAL GEO
        val currentlat = latitude1
        val currentlon = longitude1
        val destinationlat = latitude2
        val destinationlon = longitude2

        // MENCARI DISTANCE
        val latDistance = Math.toRadians(destinationlat - currentlat)
        val lonDistance = Math.toRadians(destinationlon - currentlon)

        val a = Math.sin(latDistance / 2) * Math.sin(latDistance / 2) +
            Math.cos(Math.toRadians(currentlat)) * Math.cos(Math.toRadians(destinationlat)) *
            Math.sin(lonDistance / 2) * Math.sin(lonDistance / 2)
        val c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1 - a))
        val distance: Double = R * c
        return distance
    }
}

```

Gambar 4.15 *Alogritma Haversine*

4.6 Kelebihan dan Kelemahan Aplikasi

Berdasarkan algoritma yang diterapkan, hasil yang didapat dari pengujian terhadap aplikasi terdapat kelebihan dan kekurangan antara lain :

4.6.1 Kelebihan

- Aplikasi memberikan informasi mengenai jasa servis terdekat dan estimasi harga dalam membantu *user* menemukan jasa servis sesuai kebutuhan
- Aplikasi ini terdapat rincian jasa servis yang terhubung dengan WhatsApp, Telephone, dan Google Maps
- Aplikasi ini mudah diakses karena berbasis mobile

4.6.2 Kekurangan

- a. Aplikasi ini hanya dapat dijalankan secara *online*
- b. Aplikasi ini tidak dapat dijalankan atau tidak mendukung pada sistem android *versi* 5.0 ke bawah