BAB 4
IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi Sistem

Implementasi sistem merupakan tahap penterjemah perancangan berdasarkan hasil analisis ke dalam suatu bahasa pemrograman tertentu serta penerapan perangkat lunak yang di bangun pada lingkungan yang sesungguhnya. Implementasi dan pengujian dilakukan bertujuan untuk mengetahui hasil dari sistem yang di bangun. Implementasi sistem dimulai dari pengunduhan aplikasi oleh pengguna dan di gunakan oleh pengguna untuk melakukan pointing antenna.

4.1.1 Implementasi Perangkat Keras

Implementasi perangkat keras merupakan spesfikasi perangkat keras yang digunakan dalam mengimplementasikan sistem yang dibangun. Dalam hal ini perangkat keras yang dimaksud adalah *smartphone* yang digunakan oleh pengguna. Berikut adalah spesifikasi perangkat keras yang dibutuhkan untuk menjalankan aplikasi *antenna pointer*:

Tabel 4.1 Implementasi Perangkat *Smartphone* Pengguna

No	Komponen	Spesifikasi
1	*Processor*	Quad-core 1.8 GHz
2	*RAM*	2 GB
3	*Storage*	16 GB
4	*Network*	GSM / HSPA / LTE
5	Fitur / Sensor	GPS, Magnetometer

Pada Tabel 4.1 Implementasi Perangkat Keras Pengguna menunjukan spesifikasi perangkat keras *smartphone* yang digunakan salah satu pengguna dalam menjalankan aplikasi Antenna Pointer yang sudah diunduh. Perangkat kunci yang dibutuhkan agar aplikasi *antenna pointer* dapat berjalan dengan baik adalah perangkat *smartphone* yang memiliki dukungan koneksi internet baik itu HSPA ataupun LTE, dukungan fitur GPS, dan dukungan sensor magnetometer.

4.1.2 Implementasi Perangkat Lunak

Implementasi perangkat lunak merupakan spesifikasi perangkat lunak yang digunakan dalam mengimplementasikan sistem yang dibangun. Dalam hal ini perangkat lunak yang dimaksud adalah sistem opreasi yang digunakan oleh pengguna. Implementasi perangkat lunak yang digunakan oleh pengguna adalah sebagai berikut:

Tabel 4.2 Implementasi Perangkat Lunak Pengguna

No	Perangkat Lunak	Spesifikasi
1	Sistem Operasi	Android versi 5.0 (Lollipop)

Pada Tabel 4.2 Implementasi Perangkat Lunak Pengguna menunjukan spesifikasi perangkat lunak atau sistem operasi yang digunakan oleh pengguna adalah Android versi 5.0 (Lollipop). Perangkat *smartphone* yang memiliki sistem operasi Android versi 5.0 sudah dapat memasang dan menggunakan aplikasi *antenna pointer* dengan baik.

4.1.3 Implementasi Basis Data

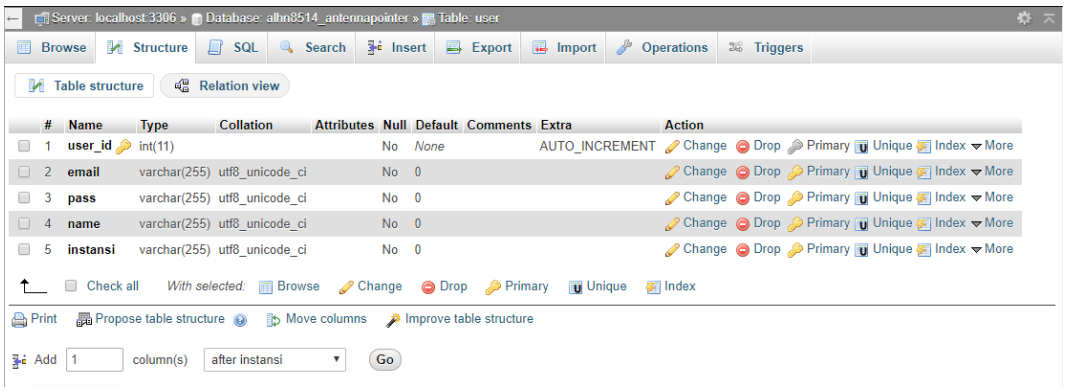
Implementasi basis data merupakan gambaran dalam pembuatan basis data pada aplikasi yang di bangun. Basis data yang dibangun menggunakan Bahasa SQL (*Structured Query Language*). Dan dalam implementasinya, DBMS yang digunakan untuk implementasi basis data ini adalah MySQL versi 5.6.39 (*MySQL Community Server*).

Seperti yang sudah dibahas pada bab sebelumnya mengenai basis data yang dibangun, pada aplikasi antenna pointer ini memiliki sebuah basis data yang terdiri dari dua tabel yaitu tabel *user* dan tabel *history*. Tabel *user* digunakan untuk menyimpan data-data pengguna mulai dari *account credentials*, nama, dan instansi asal dari pengguna tersebut. Sedangkan tabel *history* digunakan untuk menyimpan data-data riwayat instalasi antenna yang sudah dilakukan oleh pengguna mulai dari data deskripsi, foto, sampai dengan tanggal dan waktu pencatatan riwayat. Berikut adalah implementasi basis data pada sistem yang dibangun:

Tabel 4.3 Implementasi Basis Data

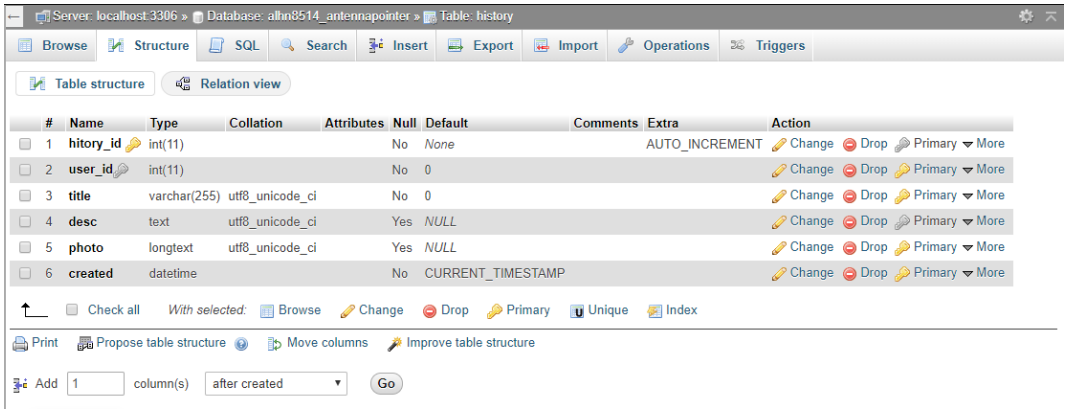
No	Nama Tabel	SQL
1	*user*	CREATE TABLE `user` (`user_id` int(11) NOT NULL, `email` varchar(255) COLLATE utf8_unicode_ci NOT NULL DEFAULT '0', `pass` varchar(255) COLLATE utf8_unicode_ci NOT NULL DEFAULT '0', `name` varchar(255) COLLATE utf8_unicode_ci NOT NULL DEFAULT '0', `instansi` varchar(255) COLLATE utf8_unicode_ci NOT NULL DEFAULT '0') ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;
2	*history*	CREATE TABLE `history` (`hitory_id` int(11) NOT NULL, `user_id` int(11) NOT NULL DEFAULT '0', `title` varchar(255) COLLATE utf8_unicode_ci NOT NULL DEFAULT '0', `desc` text COLLATE utf8_unicode_ci, `photo` longtext COLLATE utf8_unicode_ci, `created` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;

Berikut adalah gambar struktur tabel dari tabel *user* yang sudah dibuat sesuai dengan *query* yang dibuat pada Tabel 4.3 Implementasi Basis Data:



Gambar 4.1 Implementasi Tabel *user*

Berikut adalah gambar struktur tabel dari tabel *history* yang sudah dibuat sesuai dengan *query* yang dibuat pada Tabel 4.3 Implementasi Basis Data:



Gambar 4.2 Implementasi Tabel *history*

4.1.4 Implementasi Antarmuka

Implementasi antarmuka bertujuan untuk mengubah hasil analisis menjadi bentuk nyata. Antarmuka merupakan bagian terpenting pada pembangunan aplikasi *mobile android* untuk *pointing antenna*. Implementasi antarmuka pada pembangunan aplikasi ini terdiri dari antarmuka aplikasi pengguna atau teknisi yang akan melakukan instalasi dan *pointing antenna*. Berikut adalah implementasi antarmuka dari sistem yang dibangun:

Tabel 4.4 Implementasi Antarmuka Pengguna

No	Nama Antarmuka	Keterangan
1	Halaman *Intro Screen*	Menampilkan panduan cara penggunaan aplikasi, halaman ini hanya akan muncul sekali pada saat aplikasi pertama kali di install
2	Halaman *Pointing Antenna*	Menampilkan maps dan form untuk perencanaan *pointing antenna*
3	Halaman Hasil *Pointing Antenna*	Menampilkan grafik hasil *pointing antenna* berikut dengan elevasi permukaan tanah
4	Halaman Menentukan Arah Kompas	Menampilkan maps untuk menentukan arah yang akan dituju oleh kompas
5	Halaman Tampilan Kompas	Menampilkan kompas yang menunjuk ke arah yang sudah ditentukan sebelumnya

6	Halaman *Login*	Menampilkan form *login* berupa *email* dan *password*
7	Halaman Pendaftaran	Menampilkan form pendaftaran berupa data diri untuk registrasi akun
8	Halaman *Reset Password*	Menampilkan form *reset password* berupa alamat *email* dari pengguna
9	Halaman Melihat List *History*	Menampilkan list history instalasi antenna yang sudah dilakukan oleh pengguna
10	Halaman Detail *History*	Menampilkan detail dari instalasi antena yang sudah dilakukan oleh pengguna
11	Halaman Tambah *History*	Menampilkan form untuk menambah history instalasi antenna berupa foto dan data-data tambahan lainnya.

Dari Tabel 4.4 Implementasi Antarmuka Pengguna berikut adalah *screenshot* atau gambar dari antarmuka aplikasi yang sudah diterapkan:

1. Implementasi Halaman *Intro Screen*

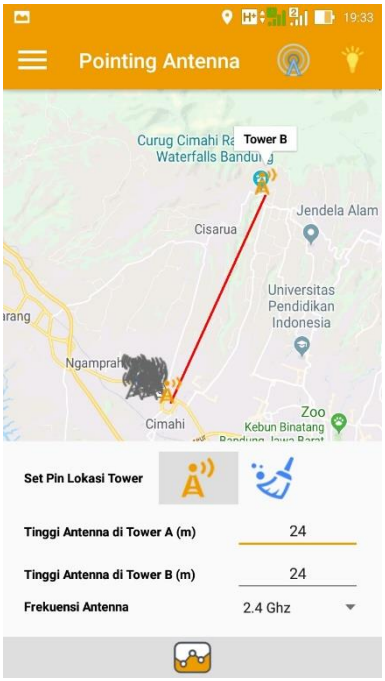


Gambar 4.3 Implementasi Halaman *Intro Screen*

Halaman *Intro Screen* ini berisi informasi mengenai panduan dan cara penggunaan aplikasi. Terdapat empat *slide* didalam halaman *intro screen* ini meliputi *slide* pengenalan awal atau sambutan dan tiga sisanya merupakan slide yang menjelaskan langkah-langkah penggunaan aplikasi mulai dari perencanaan, instalasi, dan dokumentasi.

2. Implementasi Halaman *Pointing Antenna*

   Berikut adalah gambar implementasi dari halaman *pointing antenna*:



Gambar 4.4 Implementasi Halaman *Pointing Antenna*

3. Implementasi Halaman Hasil Pointing Antenna

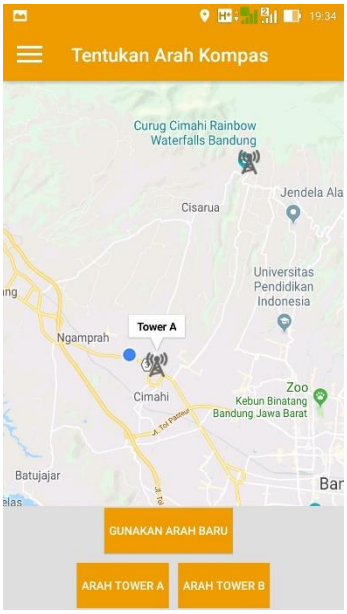
   Berikut adalah gambar implementasi dari halaman hasil *pointing antenna*:



Gambar 4.5 Implementasi Halaman Hasil *Pointing Antenna*

4. Implementasi Halaman Menentukan Arah Kompas

   Berikut gambar implementasi dari halaman menentukan arah kompas:



Gambar 4.6 Implementasi Halaman Menentukan Arah Kompas

5. Implementasi Halaman Tampilan Kompas

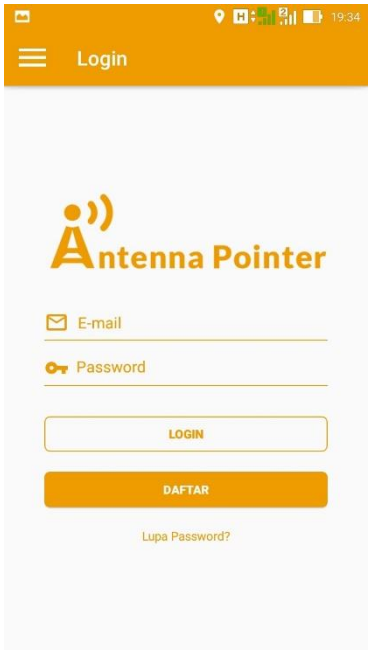
   Berikut adalah gambar implementasi dari halaman tampilan kompas:



Gambar 4.7 Implementasi Halaman Tampilan Kompas

6. Implementasi Halaman Login

   Berikut adalah gambar implementasi dari halaman login:



Gambar 4.8 Implementasi Halaman Login

7. Implementasi Halaman Pendaftaran

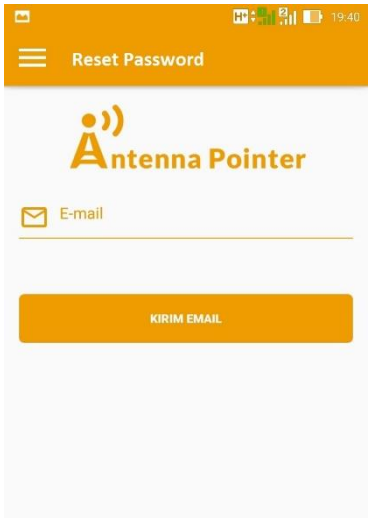
   Berikut adalah gambar implementasi dari halaman pendaftaran:



Gambar 4.9 Implementasi Halaman Pendaftaran

8. Implementasi Halaman *Reset Password*

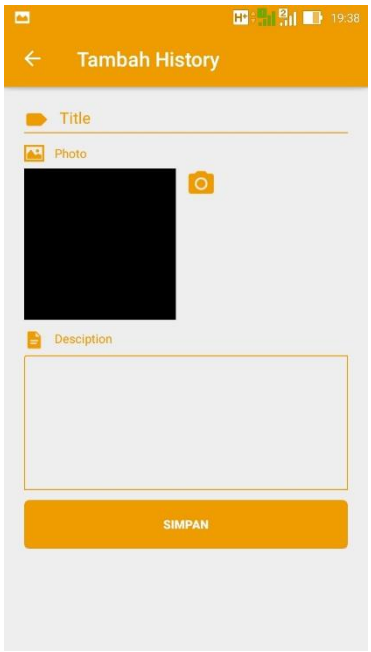
   Berikut adalah gambar implementasi dari halaman *reset password*:



Gambar 4.10 Implementasi Halaman *Reset Password*

9. Implementasi Halaman Tambah *History*

   Berikut adalah gambar implementasi dari halaman tambah *history*:



Gambar 4.11 Implementasi Halaman Melihat List *History*

4.1.5 Implementasi Teknologi

Implementasi teknologi merupakan penerapan dari hasil analisis teknologi kedalam bentuk nyata dari aplikasi atau sistem yang dibangun. Implementasi teknologi ini merupakan salah satu bagian yang sangat penting dalam penelitian dan pembangunan aplikasi *antenna pointer* ini. Tanpa realiasasi dari implementasi teknologi, sistem yang dibangun tidak dapat berfungsi sama sekali karena teknologi *mobile* yang diterapkan merupakan kunci utama agar semua fungsi pada aplikasi dapat berjalan. Berikut adalah implementasi teknologi pada aplikasi antenna pointer:

1. GPS

   Sesuai dengan pembahasan analisis teknologi GPS pada bab sebelumnya, teknologi ini digunakan untuk mendapatkan koordinat pengguna yang kemudian dimanfaatkan pada google maps untuk mencari tower terdekat. Berikut adalah *function* dari penggunaan teknologi GPS untuk mendapatkan lokasi pengguna saat ini dan dapat digunakan lagi untuk mencari tower terdekat atau fitur kompas:

```
public void getMyLocationGps(){
    //ringProgressDialog.show();

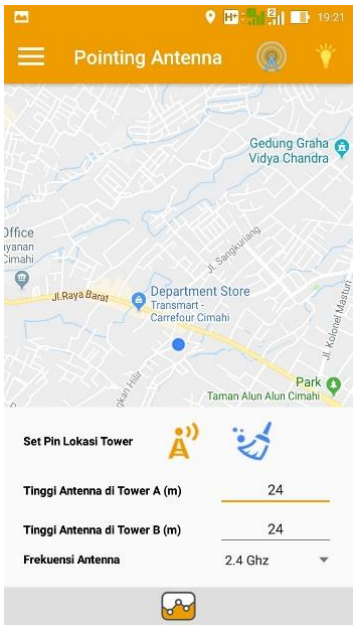
    GPSTracker gpsTracker=new GPSTracker(getActivity(), (location) → {
          try {
              gpsLocation = location;
              if (gpsLocation != null) {
                  if (mStatusGps==0) {
                      mStatusGps = 1;
                      setup.setGps(gpsLocation);
                      addPathMyLocation(gpsLocation);
                     layout_loding.setVisibility(View.GONE);
                  }
              }
          }catch (Exception e){

          }

    });
//      getActivity().runOnUiThread(new Runnable() {
//          @Override
//          public void run() {
//              ringProgressDialog = ProgressDialog.show(getActivity(), "Memuat lokasi saat ini", "Info ...", true);
//              ringProgressDialog.setCancelable(false);
//          }
//      });
//      gpsLocation = gpsTracker.getLocation(LocationManager.GPS_PROVIDER);
//      if (gpsLocation != null) {
//          addPathMyLocation(gpsLocation);
//      } else {
//          Toast.makeText(
```

Gambar 4.12 *Function* Penggunaan Teknologi GPS

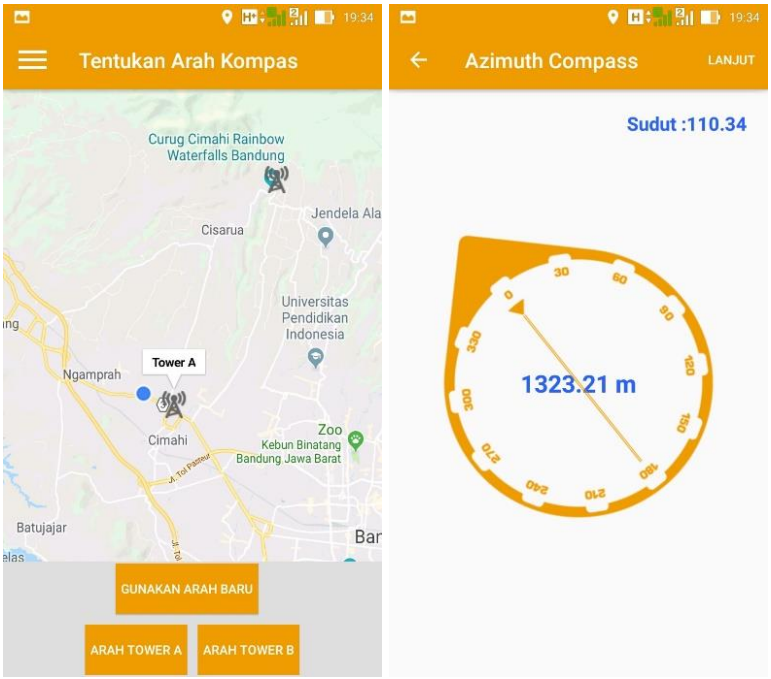
Sesuai dengan Gambar 4.12 Implementasi Teknologi GPS Pada Google Maps dapat dihasilkan output dari penggunaan GPS pada gambar berikut:



Gambar 4.13 Hasil Implementasi Teknologi GPS Pada Google Maps

Pada Gambar 4.13 Hasil Implementasi Teknologi GPS Pada Google Maps terlihat dot (titik) berwarna biru yang menunjukkan lokasi pengguna saat ini. Dot berwarna biru tersebut secara dinamis akan berubah posisi apabila pengguna bergerak ato berganti lokasi. Selanjutnya jika pengguna memanfaatkan fitur mencari tower terdekat maka koordinat yang sudah didapat dari GPS ini akan diolah kembali untuk mendapatkan tower-tower terdekat disekitar pengguna.

Selain dimanfaatkan pada fitur mencari tower terdekat dan google maps, penggunaan dari GPS ini dimanfaatkan juga untuk fitur kompas pada aplikasi *antenna pointer*. Koordinat pengguna dari GPS diperlukan untuk menghitung sudut azimuth kompas dan mengarahkan kompas dari lokasi pengguna ke titik lokasi yang dituju. Berikut adalah gambar hasil penggunaan teknologi GPS pada fitur kompas:



Gambar 4.14 Hasil Implementasi Teknologi GPS Pada Kompas

2. Google Maps API

Sesuai dengan pembahasan analisis teknologi Google Maps API pada bab sebelumnya, teknologi ini digunakan untuk memvisualisasikan peta agar lebih memudahkan pengguna dalam proses pointing antenna. Berikut adalah gambar *setting* dan *function* dari penggunaan Google Maps API:

```
<resources>
    <!--
    TODO: Before you run your application, you need a GooAntennaPointerMaps API key.

    To get one, follow this link, follow the directions and press "Create" at the end:

    https://console.developers.google.com/flows/enableapi?apiid=maps_android_backend&keyType=CLIENT_SIDE_ANDROID&r=F2:BD:E4:78:25:BB:86:6

    You can also add your credentials to an existing key, using these values:

    Package name:
    F2:BD:E4:78:25:BB:86:68:2F:98:BC:86:F0:A9:AA:38:3F:EF:5C:32

    SHA-1 certificate fingerprint:
    F2:BD:E4:78:25:BB:86:68:2F:98:BC:86:F0:A9:AA:38:3F:EF:5C:32

    Alternatively, follow the directions here:
    https://developers.google.com/maps/documentation/android/start#get-key

    Once you have your key (it starts with "AIza"), replace the "google_maps_key"
    string in this file.
    -->
    <string name="google_maps_key" templateMergeStrategy="preserve" translatable="false">AIzaSyBxKwfgZas2I2eP9H1P7t7DCCfLa8Z53Ao</string>
</resources>
```

Gambar 4.15 Setting Google Maps API Key

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"

    xmlns:tools="http://schemas.android.com/tools"
    xmlns:app="http://schemas.android.com/apk/res-auto">
    <fragment xmlns:android="http://schemas.android.com/apk/res/android"
        xmlns:map="http://schemas.android.com/apk/res-auto"
        xmlns:tools="http://schemas.android.com/tools"
        android:id="@+id/map"
        android:name="com.google.android.gms.maps.SupportMapFragment"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        tools:context="com.antennapointer.MapsActivity" />

    <RelativeLayout
        android:gravity="bottom"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        >
        <LinearLayout
            android:id="@+id/menu_bottom"
            android:background="@color/withe"
            android:layout_width="match_parent"
            android:layout_height="225dp"
            android:orientation="vertical"
           >
```

Gambar 4.16 *Layout* Penggunaan Google Maps API

Dari setting dan layout yang sudah diterapkan tersebut maka secara otomatis google maps akan tampil pada aplikasi *antenna pointer* sesuai dengan layout yang sudah ditentukan. Berikut adalah gambar hasil output dari pengguaan Google Maps API:



Gambar 4.17 Hasil Implementasi Teknologi Google Maps API

3. Google Elevation API

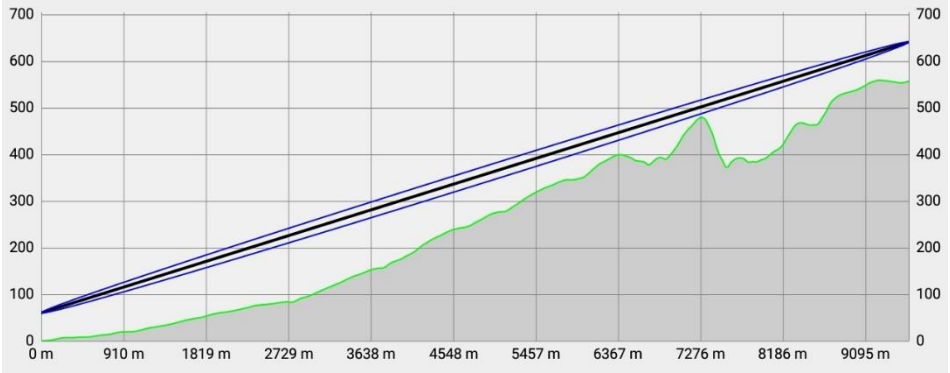
Sesuai dengan pembahasan analisis teknologi Google Elevation API pada bab sebelumnya, teknologi ini digunakan untuk memvisualisasikan kontur tanah atau elevasi permukaan bumi kedalam bentuk grafik agar lebih memudahkan pengguna dalam melakukan *pointing antenna*. Berikut adalah *function* dari implementasi Google Elevation API:

```
xAxis.setTextColor(R.color.colorWeight);
xAxis.setPosition(XAxis.XAxisPosition.BOTTOM);
xAxis.setDrawGridLines(true);
xAxis.setDrawAxisLine(true);
xAxis.setTextColor(Color.BLACK);
xAxis.setTextSize(10f);
YAxis leftAxis = mChart.getAxisLeft();
leftAxis.removeAllLimitLines(); // reset all limit lines to avoid overlapping lines
leftAxis.setStartAtZero(true);
leftAxis.setDrawLimitLinesBehindData(true);
leftAxis.setDrawAxisLine(true);
leftAxis.setDrawGridLines(true);
leftAxis.setTextSize(10f);
leftAxis.setTextColor(Color.BLACK);
mChart.getAxisRight().setEnabled(true);
mChart.animateXY( durationMillisX: 2000,  durationMillisY: 2000, Easing.EasingOption.EaseInOutQuart, Easing.EasingOption.EaseInOutQuart);
jsonUrl = "https://maps.googleapis.com/maps/api/elevation/json?path="+mPointA+"|"+mPointB+"&samples="+String.valueOf(msamples+1);
StringRequest getData = new StringRequest(jsonUrl, (resp) → {
        Gson gson= new Gson();
        FraznelModel fraznelModel= gson.fromJson(resp,FraznelModel.class);
         mResults= fraznelModel.getResults();
        FresnelZone fresnelZone=new FresnelZone();
        LinkedList<ChartData> am=fresnelZone.Calculate(wavelength,samples, distance, heightStart, heightEnd,mResults);
        setData(am,fresnelZone.getNilai_min());

}, (error) → {
        Toast.makeText( context: LineChartActivity.this,  text: "Check Internet Connection", Toast.LENGTH_LONG).show();
});
```

Gambar 4.18 *Function* Penggunaan Google Elevation API

Dari Gambar 4.18 Function Penggunaan Google Elevation API akan didapatkan larik data elevasi permukaan bumi di setiap titik lokasi. Kemudian data tersebut diolah dan ditampilkan dalam bentuk grafik agar lebih mudah dibaca oleh pengguna. Berikut adalah hasil dari implementasi Google Elevation API yang ditampilkan dalam bentuk grafik:



Gambar 4.19 Implementasi Teknologi Google Elevation API

4. OpenCelliD API

Teknologi OpenCelliD API digunakan untuk mendapatkan data tower-tower terdekat dari lokasi pengguna. Penggunaan teknologi ini memanfaatkan GPS untuk mendapatkan lokasi pengguna secara *realtime*. Berikut adalah *function* dari implementasi OpenCelliD API:

```
private class GetData extends AsyncTask<String, Void, String> {
    @Override
    protected String doInBackground(String... params) {

        return PostData(gpsLocation.getLatitude(),gpsLocation.getLongitude(), rad: 20, lim: 15);
    }

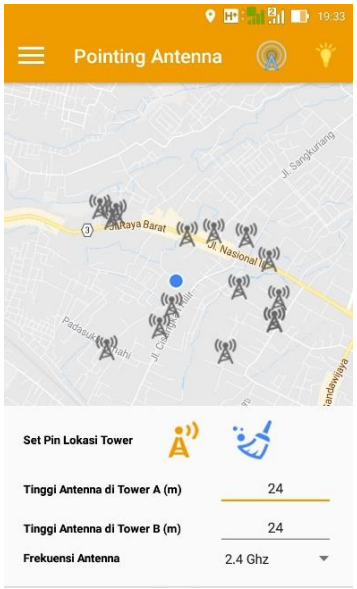
    @Override
    protected void onPostExecute(String s) {
        if (CekData(s)>0) {
            try {
                Gson gson=new Gson();
                LatitudeModel data=gson.fromJson(s, LatitudeModel.class);
                datamyTower=data.getData();
                showTower();
                layout_loding.setVisibility(View.GONE);
            }catch (Exception e){

            }


        }else{
            Toast.makeText(getActivity(), text: "Silahkan Cek server",Toast.LENGTH_SHORT).show();
        }
    }
```

Gambar 4.20 Implementasi Teknologi OpenCelliD API

Dari *function* tersebut kemudian dalam aplikasi antenna pointer akan ditampilkan data tower terdekat dengan bentuk icon tower di maps:



Gambar 4.21 Implementasi Teknologi OpenCelliD API

4.1.5 Implementasi Metode Perhitungan *Fresnel Zone*

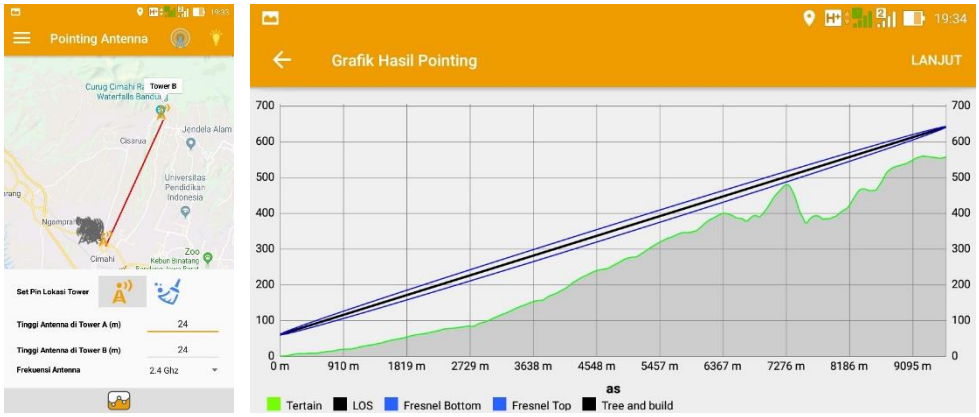
Implementasi metode perhitungan *fresnel zone* merupakan penerapan dari hasil analisis perhitungan *fresnel zone* kedalam bentuk algoritma yang diterapkan pada aplikasi atau sistem yang dibangun. Implementasi metode atau algoritma perhitungan *fresnel zone* ini sangat penting untuk menjalankan fungsionalitas utama aplikasi yaitu *pointing antenna*. Berikut adalah implementasi metode perhitungan *fresnel zone* pada aplikasi *antenna pointer*:

```
public static final double FresnelWave( double wavelength, double d1, double d2) {
    double fresnel = Math.sqrt((wavelength*d1*d2)/(d1+d2));
    return fresnel;
}

public final LinkedList Calculate(double wavelength, double samples, double distance, double heightStart, double heightEnd,List<Resul
    double mSoilstart=Result.get(0).getElevation();
    double mSoilend=Result.get(Result.size()-1).getElevation();
    double ds       = distance / samples;
    double diff     = ((heightEnd+mSoilend) - ( heightStart+mSoilstart))/(samples);
    nilai_min= getMin(Result);
    LinkedList<ChartData> linkedList = new LinkedList<>();
    if (wavelength != 0) {
        for (int i = 0; i <= samples; i++) {
            double dist     = ds * i;
            double disdist  = distance - dist;
            double fz       = FresnelWave(wavelength, dist, disdist);
            double direct   = heightStart + (diff * i);
            String km       = String.valueOf(Math.round(dist)) + " m";
            linkedList.add(new ChartData(km,  yValue: Result.get(i).getElevation()-nilai_min,  yValue2: mSoilstart+direct-nilai_min,  yVal
        }
    }
    return linkedList;
```

Gambar 4.22 Implementasi Algoritma Perhitungan *Fresnel Zone*

Untuk penerapannya pada aplikasi *antenna pointer* yang dibangun, algoritma perhitungan *fresnel zone* ini akan dieksekusi oleh sistem ketika pengguna sudah melengkapi data masukan pada *form* dan maps *pointing antenna*. Sistem mengolah data-data tersebut dengan algoritma perhitungan *fresnel zone* yang kemudian ditampilkan dalam bentuk grafik yang menggambarkan sebuah elips dari *fresnel zone* antara dua titik *pointing antenna*.



Gambar 4.23 Implementasi Metode Perhitungan *Fresnel Zone*

4.2 Pengujian Sistem

Pada tahap pengujian sistem, pembangunan aplikasi *antenna pointer* berdasarkan dari perancangan fungsionalitas dan analisis sistem pada bab sebelumnya. Pengujian yang dilakukan bertujuan untuk mengetahui apakah aplikasi *antenna pointer* yang dibangun telah sesuai dengah hasil yang diinginkan atau tidak. Pengujian yang dilakukan dalam penelitian ini meliputi pengujian fungsional (*functional testing*) dan *acceptance testing* dengan pengujian beta.

4.2.1 Pengujian Fungsional

Pengujian fungsional yang dilakukan menggunakan metode *blackbox* yang berfokus pada persyaratan fungsionalitas dari sistem [16]. Pengujian *blackbox* adalah metode pengujian perangkat yang berfokus pada persyaratan fungsional perangkat lunak. Pengujian ini memungkinkan analisis sistem memperoleh kumpulan kondisi input yang akan mengerjakan seluruh keperluan fungsional program. Pada *blackbox testing*, pengguna atau penguji memperlakukan sebuah program sebagai sebuah kotak hitam (*blackbox*) yang tidak diketahui apa isi dari kotak tersebut, pengguna hanya memberikan berbagai macam data input yang kemudian menerima data output untuk disesuaikan dengan fungsional sistem. [17]

4.2.1.1 Skenario Pengujian Fungsionalitas Aplikasi

Berikut adalah skenario pengujian fungsionalitas aplikasi yang dibangun menggunakan metode *blackbox*. Tabel berikut menerangkan komponen atau fitur-fitur yang akan diuji.

Tabel 4.5 Skenario Pengujian Fungsionalitas Aplikasi

No	Komponen yang Di Uji	Skenario	Pengujian
1	Halaman *Intro Screen*	Menampilkan panduan cara penggunaan aplikasi	*Blackbox*
2		Meminta data tower terdekat	*Blackbox*

	Halaman *Pointing Antenna*	Input data titik tower A dan B, ketinggian antena dan frekuensi antena	*Blackbox*
		Rekomendasi ketinggian antenna di setiap titik tower	*Blackbox*
		Input data kosong	*Blackbox*
3	Halaman Kompas	Memilih titik dari arah kompas	*Blackbox*
4	Halaman *Login*	Input data *email* dan *password* yang benar	*Blackbox*
		Input data *email* dan *password* yang salah	*Blackbox*
		Input data kosong	*Blackbox*
5	Halaman Pendaftaran	Input data secara lengkap dan benar	*Blackbox*
		Input *email* yang sudah teregistrasi	*Blackbox*
		Input data kosong	*Blackbox*
6	Halaman *Reset Password*	Input data *email*	*Blackbox*
		Input *email* kosong	*Blackbox*
7	Halaman Melihat List *History*	Menampilkan list *history* instalasi antena	*Blackbox*
		Klik opsi hapus pada list *history* instalasi antena	*Blackbox*
8	Halaman Detail *History*	Menampilkan detail *history* instalasi antena	*Blackbox*
11	Halaman Tambah *History*	Input data secara lengkap dan benar	*Blackbox*
		Input data kosong	*Blackbox*

4.2.1.2 Skenario dan Hasil Pengujian Fungsionalitas Aplikasi

Hasil pengujian menampilkan hasil dari pengujian aplikasi yang dilakukan sesuai dengan rencana dan skenario pengujian. Berikut hasil dari pengujian yang telah dilakukan:

1. Halaman Pointing Antenna

Tabel 4.6 Pengujian *Blackbox* Halaman *Pointing Antenna*

Skenario Pengujian	Data Masukan	Hasil Yang Diharapkan	Data Keluaran	Hasil Pengujian
Mendapatkan data tower terdekat	(-6.886565, 107.61528)	Tampil tower terdekat pada maps	Tampil tower terdekat pada maps	(√) Berhasil ( ) Gagal
Input koordinat tower A dan B, tinggi antenna dan frekuensi antena untuk perhitungan fresnel zone	(-6.886565, 107.61528), (-6.861134, 107.59300), tinggi 30m, frekuensi 2.4 GHz	Tampil grafik hasil perhitungan fresnel zone dan elevasi permukaan bumi	Tampil grafik hasil perhitungan fresnel zone dan elevasi permukaan bumi	(√) Berhasil ( ) Gagal
Rekomendasi ketinggian antenna	30m di tower A, 30m di tower B	Tampil rekomendasi ketinggian antenna	Tampil rekomendasi ketinggian antenna	(√) Berhasil ( ) Gagal
Input data kosong untuk perhitungan fresnel zone	Form field tidak diisi / tidak lengkap	Perhitungan fresnel zone tidak diproses dan tidak tampil grafik	Perhitungan fresnel zone tidak diproses dan tidak tampil grafik	( ) Berhasil (√) Gagal

2. Halaman Kompas

Tabel 4.7 Pengujian *Blackbox* Halaman Kompas

Skenario Pengujian	Data Masukan	Hasil Yang Diharapkan	Data Keluaran	Hasil Pengujian
Memilih lokasi yang akan dituju kompas	(-6.861134, 107.59300)	Kompas mengarah ke lokasi yang dipilih	Kompas mengarah ke koordinat lokasi UPI	(√) Berhasil ( ) Gagal

3. Halaman Login

Tabel 4.8 Pengujian *Blackbox* Halaman *Login*

Skenario Pengujian	Data Masukan	Hasil Yang Diharapkan	Data Keluaran	Hasil Pengujian
Input data *username, password* yang benar	*username*: firman58, *password*: F4P@55	Berhasil login dan diarahkan ke halaman melihat *history*	Berhasil login dan diarahkan ke halaman melihat *history*	(√) Berhasil ( ) Gagal
Input data *username, password* yang salah	*username:* test *password*: 123	Tampil pesan gagal login	Tampil pesan gagal login	( ) Berhasil (√) Gagal
Input data kosong	*username:* firman58, *password* tidak diisi	Tampil pesan untuk melengkapi data	Tampil pesan untuk melengkapi data	( ) Berhasil (√) Gagal

4. Halaman Pendaftaran

Tabel 4.9 Pengujian *Blackbox* Halaman Pendaftaran

Skenario Pengujian	Data Masukan	Hasil Yang Diharapkan	Data Keluaran	Hasil Pengujian
Input data secara lengkap dan benar	nama: hadi, *email*: hadi-@gmail .com, *password*: 12, instansi: TKJ	Registrasi berhasil dan akun baru sudah terbuat	Registrasi berhasil dan akun baru sudah terbuat	(√) Berhasil ( ) Gagal
Input *email* yang sudah teregistrasi	nama: hadi, *email*: hadi.gmail .com, *password*: 12, instansi: TKJ	Tampil pesan gagal registrasi	Tampil pesan gagal registrasi	( ) Berhasil (√) Gagal
Input data kosong	Semua field tidak diisi	Tampil pesan untuk melengkapi data	Tampil pesan untuk me-lengkapi data	( ) Berhasil (√) Gagal

5. Halaman Tambah *History*

Tabel 4.10 Pengujian *Blackbox* Halaman *Reset Password*

Skenario Pengujian	Data Masukan	Hasil Yang Diharapkan	Data Keluaran	Hasil Pengujian
Input data secara lengkap dan benar	judul: antena UPI, photo instalasi, deskripsi: freq 5GHz,	Data history berhasil ditambahkan dan disimpan	Data history berhasil ditambahkan dan disimpan	(√) Berhasil ( ) Gagal
Input data kosong	judul: tidak diisi, photo: tidak diisi, deskripsi: tidak diisi	Tampil pesan untuk melengkapi data	Tampil pesan untuk me-lengkapi data	( ) Berhasil (√) Gagal

4.2.2 Pengujian Beta

Pengujian beta yang mana merupakan bagian dari *acceptance testing* atau *User Acceptance Testing* (UAT) merupakan pengujian yang dilakukan secara objektif yang dilakukan secara langsung oleh pengguna aplikasi. Hal ini dilakukan untuk dapat mengetahui sejauh mana aplikasi yang dibangun dapat membantu memudahkan dan dapat menyelesaikan masalah yang sudah dijelaskan pada identifikasi masalah dan sesuai dengan harapan dari pengguna.[17]

4.2.2.1 Skenario Pengujian Beta

Kuesioner pengujian merupakan teknik pengolahan data yang merupakan langkah dalam mengolah data yang telah didapat untuk dijadikan hasil penelitian sehingga dapat ditarik kesimpulan. Kuesioner yang akan di ajukan kepada koresponden mempunyai pertanyaan dengan dengan pilihan jawaban 5 buah menggunakan skala likert 1 sampai 5. Berikut adalah detail skala likert:

Tabel 4.11 Penilaian Skala Likert

Kategori	Nilai
Sangat Setuju	5
Setuju	4
Kurang Setuju	3
Tidak Setuju	2
Sangat Tidak Setuju	1

a. Untuk menghitung jumlah maksimum skor kriterium jumlah penilaian = 5, dan jumlah responden 30 yaitu dengan rumus:

$$Kriterium = Jumlah\ Penilaian\ \times Jumlah\ Responden$$

Maka nilai dari kuesioner yang dibuat 5 x 150 = 150

b. Untuk mengetahui jumlah jawaban dari responden dalam bentuk persentase, di gunakan rumus sebagai berikut:

$$P = \frac{total\ nilai}{skor\ ideal} \times 100\%$$

Keterangan:

P = Nilai persentase yang dicari

Total nilai = Jumlah frekuensi dikalikan nilai yang di tetapkan jawaban

Skor Ideal = Nilai tertinggi dilakukan dengan jumlah sampel

c. Skor yang telah di dapat selanjutnya dimasukan ke dalam bentuk interpresentasi skor sebagai berikut:



Gambar 4.24 *Interval Rating Scale*

Berikut ini adalah pertanyaan kuesioner yang akan diajukan kepada responden yang menggunakan aplikasi ini. Pertanyaanya adalah sebagai berikut:

Tabel 4.12 Daftar Pertanyaan Kuesioner Pengguna

No	Pertanyaan
1	Apakah aplikasi *antenna pointer* ini dapat memudahkan anda dalam melakukan *pointing antenna*?
2	Apakah aplikasi *antenna pointer* ini dapat membantu memudahkan anda dalam mencari tower terdekat untuk pemasangan antenna?
3	Apakah aplikasi *antenna pointer* ini dapat membantu menjadi alat subtitusi dari keterbatasan GPS dan kompas ketika melakukan *pointing antenna*?
4	Apakah aplikasi *antenna pointer* ini dapat membantu memudahkan dokumentasi dari instalasi antenna yang sudah dilakukan?
5	Apakah dengan adanya aplikasi *antenna pointer* ini dapat lebih meningkatkan efisiensi waktu ketika pointing dan instalasi antenna?
6	Apakah aplikasi *antenna pointer* ini mudah untuk dipelajari dan digunakan?

4.2.2.2 Perhitungan Hasil Kuesioner

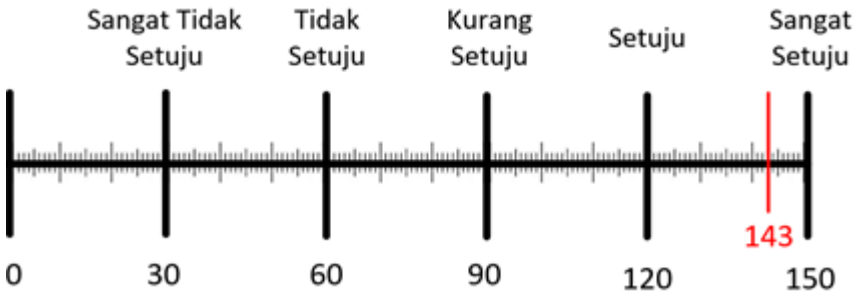
Berikut ini adalah hasil persentase masing-masing jawaban yang sudah dihitung nilainya dengan menggunakan rumur-rumus diatas. Kuesioner ini telah diujikan kepada 30 orang pengguna.

1. Apakah aplikasi antenna pointer ini dapat memudahkan anda dalam melakukan pointing antenna?

Tabel 4.13 Hasil Kuesioner Pertanyaan 1

Kategori Jawaban	Skor	Frekuensi Jawaban	Total Skor	Nilai Presentase	Keputusan
Sangat Setuju	5	23	115	(143 / (30x5)) x 100% = **95.3%**	Mendekati Sangat Setuju
Setuju	4	7	28		
Kurang Setuju	3	0	0		
Tidak Setuju	2	0	0		
Sangat Tidak Setuju	1	0	0		
Jumlah		30	143		

Berdasarkan perhitungan diatas, total skor yang didapat adalah sebanyak 143 skor dengan skala kategori jawaban diantaranya sangat setuju dan setuju. Sedangkan hasil dari nilai presentasi responden adalah 95.3% dari nilai yang diharapkan sebesar 100%. Maka dapat disimpulkan bahwa aplikasi ini sudah dapat memenuhi tujuannya untuk memudahkan pengguna dalam melakukan *pointing antenna*. Berikut gambar *interval rating scale* untuk pertanyaan nomer 1:



Gambar 4.25 Interval Rating Scale Pertanyaan 1

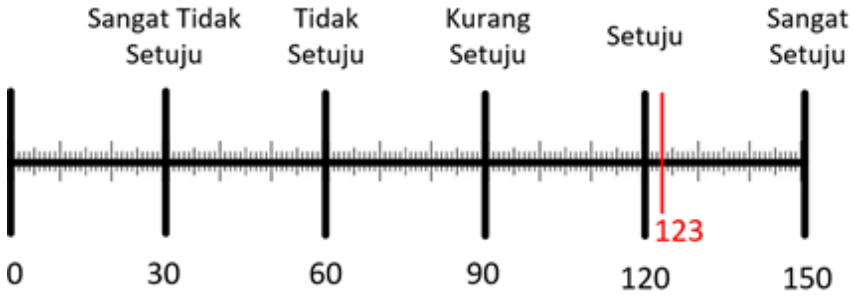
2. Apakah aplikasi antenna pointer ini dapat membantu memudahkan anda dalam mencari tower terdekat untuk pemasangan antenna?

Tabel 4.14 Hasil Kuesioner Pertanyaan 2

Kategori Jawaban	Skor	Frekuensi Jawaban	Total Skor	Nilai Presentase	Keputusan
Sangat Setuju	5	3	15	(123 / (30x5)) x 100% = **82%**	Setuju
Setuju	4	27	108		
Kurang Setuju	3	0	0		
Tidak Setuju	2	0	0		
Sangat Tidak Setuju	1	0	0		
Jumlah		30	123		

Berdasarkan perhitungan diatas, total skor yang didapat adalah sebanyak 123 skor dengan skala kategori jawaban diantaranya sangat setuju dan setuju. Sedangkan hasil dari nilai presentasi responden adalah 82% dari nilai yang diharapkan sebesar 100%. Maka dapat disimpulkan bahwa

aplikasi ini sudah dapat memenuhi tujuannya untuk memudahkan pengguna dalam mencari tower terdekat untuk pemasangan antena. Berikut gambar *interval rating scale* untuk pertanyaan nomer 2:



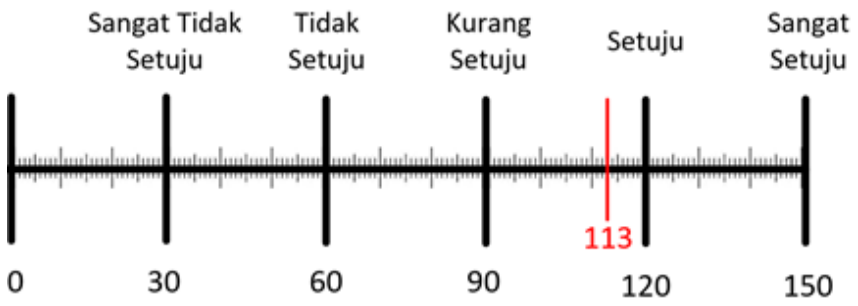
Gambar 4.26 Interval Rating Scale Pertanyaan 2

3. Apakah aplikasi antenna pointer ini dapat membantu menjadi alat subtitusi dari keterbatasan GPS dan kompas ketika melakukan pointing antenna?

Tabel 4.15 Hasil Kuesioner Pertanyaan 3

Kategori Jawaban	Skor	Frekuensi Jawaban	Total Skor	Nilai Presentase	Keputusan
Sangat Setuju	5	1	5	(113 / (30x5)) x 100% = **75.3%**	Mendekati Setuju
Setuju	4	21	84		
Kurang Setuju	3	8	24		
Tidak Setuju	2	0	0		
Sangat Tidak Setuju	1	0	0		
Jumlah		30	113		

Berdasarkan perhitungan diatas, total skor yang didapat adalah sebanyak 113 skor dengan skala kategori jawaban diantaranya sangat setuju, setuju dan kurang setuju. Sedangkan hasil dari nilai presentasi responden adalah 75.3% dari nilai yang diharapkan sebesar 100%. Maka dapat disimpulkan bahwa aplikasi ini sudah dapat memenuhi tujuannya untuk menjadi alat pengganti dan mengatasi keterbatasan dari perangkat kompas dan GPS. Berikut gambar *interval rating scale* untuk pertanyaan nomer 3:



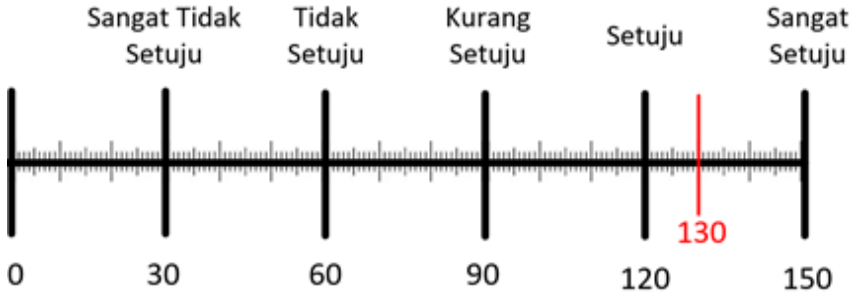
Gambar 4.27 Interval Rating Scale Pertanyaan 3

4. Apakah aplikasi antenna pointer ini dapat membantu memudahkan dokumentasi dari instalasi antenna yang sudah dilakukan?

Tabel 4.16 Hasil Kuesioner Pertanyaan 4

Kategori Jawaban	Skor	Frekuensi Jawaban	Total Skor	Nilai Presentase	Keputusan
Sangat Setuju	5	11	55	(130 / (30x5)) x 100% = **86.7%**	Setuju
Setuju	4	18	72		
Kurang Setuju	3	1	3		
Tidak Setuju	2	0	0		
Sangat Tidak Setuju	1	0	0		
Jumlah		30	130		

Berdasarkan perhitungan diatas, total skor yang didapat adalah sebanyak 130 skor dengan skala kategori jawaban diantaranya sangat setuju, setuju dan kurang setuju. Sedangkan hasil dari nilai presentasi responden adalah 86.7% dari nilai yang diharapkan sebesar 100%. Maka dapat disimpulkan aplikasi ini dapat memudahkan dokumentasi dari instalasi.



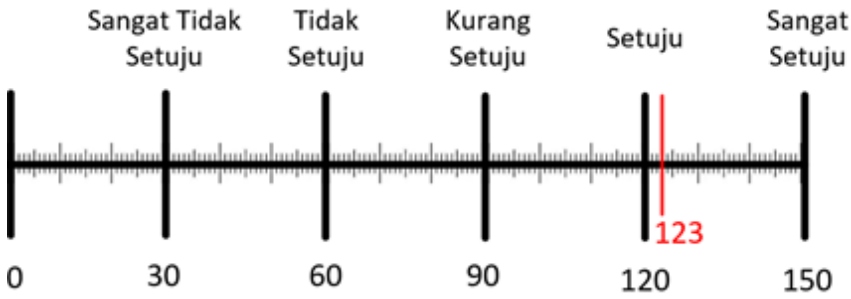
Gambar 4.28 Interval Rating Scale Pertanyaan 4

5. Apakah dengan adanya aplikasi antenna pointer ini dapat lebih meningkatkan efisiensi waktu ketika melakukan pointing dan instalasi antenna?

Tabel 4.17 Hasil Kuesioner Pertanyaan 5

Kategori Jawaban	Skor	Frekuensi Jawaban	Total Skor	Nilai Presentase	Keputusan
Sangat Setuju	5	3	15	(123 / (30x5)) x 100% = **82%**	Setuju
Setuju	4	27	108		
Kurang Setuju	3	0	0		
Tidak Setuju	2	0	0		
Sangat Tidak Setuju	1	0	0		
Jumlah		30	123		

Berdasarkan perhitungan diatas, total skor yang didapat adalah sebanyak 123 skor dengan skala kategori jawaban diantaranya sangat setuju dan setuju. Sedangkan hasil dari nilai presentasi responden adalah 82% dari nilai yang diharapkan sebesar 100%. Maka dapat disimpulkan bahwa aplikasi ini sudah dapat memenuhi tujuannya untuk lebih meningkatkan efisiensi waktu ketika melakukan pointing dan instalasi antena. Berikut gambar *interval rating scale* untuk pertanyaan nomer 5:



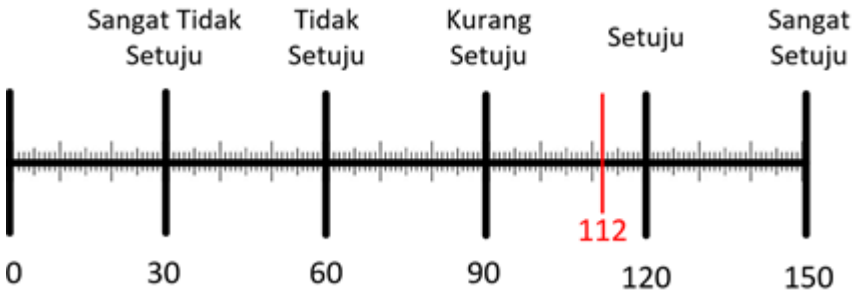
Gambar 4.29 Interval Rating Scale Pertanyaan 5

6. Apakah aplikasi antenna pointer ini mudah untuk dipelajari dan digunakan?

Tabel 4.18 Hasil Kuesioner Pertanyaan 6

Kategori Jawaban	Skor	Frekuensi Jawaban	Total Skor	Nilai Presentase	Keputusan
Sangat Setuju	5	2	10	(112 / (30x5)) x 100% = **74.7%**	Mendekati Setuju
Setuju	4	18	72		
Kurang Setuju	3	10	30		
Tidak Setuju	2	0	0		
Sangat Tidak Setuju	1	0	0		
Jumlah		30	112		

Berdasarkan perhitungan diatas, total skor yang didapat adalah sebanyak 112 skor dengan skala kategori jawaban diantaranya sangat setuju, setuju dan kurang setuju. Sedangkan hasil dari nilai presentasi responden adalah 74.7% dari nilai yang diharapkan sebesar 100%. Maka dapat disimpulkan bahwa aplikasi ini sudah dapat memenuhi tujuannya agar mudah dipelajari dan digunakan oleh pengguna. Berikut gambar *interval rating scale* untuk pertanyaan nomer 5:



Gambar 4.30 Interval Rating Scale Pertanyaan 6

4.2.3 Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian *blackbox* yang dapat dilihat pada Tabel 4.6 Skenario dan Hasil Hasil Pengujian Fungsionalitas Aplikasi dapat diketahui bahwa *output* dari aplikasi *antenna pointer* sudah sesuai dengan skenario dan fungsionalitas aplikasi yang diharapkan. Dan berdasarkan tabel tersebut juga dapat dinyatakan bahwa aplikasi sudah lolos dalam pengujian *blackbox*. Aplikasi *antenna*

pointer juga sudah dapat digunakan secara publik kepada masyarakat khususnya teknisi antenna. Sedangkan berdasarkan hasil pengujian beta yang menggunakan skala likert, dapat dilihat hasil rekapitulasi sebagai berikut:

Tabel 4.19 Rekapitulasi Hasil Pengujian Skala Likert

No	Pertanyaan	Skor	Presentase	Keputusan
1	Apakah aplikasi *antenna pointer* ini dapat memudahkan anda dalam melakukan *pointing antenna*?	143	95.3%	Sangat Setuju
2	Apakah aplikasi *antenna pointer* ini dapat membantu memudahkan anda dalam mencari tower terdekat untuk pemasangan antenna?	123	82%	Setuju
3	Apakah aplikasi *antenna pointer* ini dapat membantu menjadi alat subtitusi dari keterbatasan GPS dan kompas ketika melakukan *pointing antenna*?	113	75.3%	Setuju
4	Apakah aplikasi *antenna pointer* ini dapat membantu memudahkan dokumentasi dari instalasi antenna yang sudah dilakukan?	130	86.7%	Setuju
5	Apakah dengan adanya aplikasi *antenna pointer* ini dapat lebih meningkatkan efisiensi waktu ketika pointing dan instalasi antenna?	123	82%	Setuju
6	Apakah aplikasi *antenna pointer* ini mudah untuk dipelajari dan digunakan?	112	74.7%	Setuju

Dari rekapitulasi hasil pengujian beta dengan skala likert diatas dapat diketahui bahwa setiap poin pertanyaan memiliki skor yang sangat bagus yaitu rata-rata skor 120 dengan presentase 82.7%. Sehingga dari hasil pengujian tersebut dapat disimpulkan bahwa pembangunan aplikasi *antenna pointer* sudah sesuai dengan tujuan yang diharapkan. Mayoritas dari responden setuju bahwa aplikasi *antenna pointer* yang dibangun dapat membantu menyelesaikan permasalahan yang tercantum pada identifikasi masalah dan tujuan dari penelitian dapat tercapai.