

BAB III

METODE SLAMET HAMBALI DAN APLIKASI KIBLAT SIKU-SIKU

A. Metode Slamet Hambali

1. Biografi Singkat Slamet Hambali

Slamet Hambali dilahirkan pada tanggal 5 Agustus 1954 di sebuah desa bernama Bajangan, Kecamatan Bringin, Kabupaten Semarang, Jawa Tengah. Sang ayah KH. Hambali telah mengajarkan ilmu falak sejak kecil padanya.¹ Hal yang membuat ia tertarik belajar ilmu falak yaitu adanya anggapan yang menyatakan bahwa seorang ahli falak itu dapat mengetahui kapan daun akan jatuh dari tangkainya, dan hal lain yang sebenarnya tidak ada sama sekali.²

Slamet Hambali merupakan anak kedua dari lima bersaudara. Seluruh saudaranya masih menetap di Salatiga. Kakaknya bernama H. Ma'sum menemani sang ibu di Salatiga sementara adik-adiknya bernama Siti Fatimah, Siti Mas'udah dan Mahasin juga masih berada di kawasan Salatiga. Slamet Hambali melakukan kegiatan mengajar di Semarang dengan segala aktifitas yang semakin padat di beberapa lembaga negara sehingga Slamet Hambali masih menetap di Semarang. Sejak tahun 1988 ia menetap di Semarang, tepatnya di kawasan perumahan Pasadena

¹ Slamet Hambali, *Ilmu Falak Arah Kiblat Setiap Saat*, Semarang: Pustaka Ilmu, 2012. hlm. 173.

² Barokatul Laili, *Analisis Metode Pengukuran Arah Kiblat Metode Slamet Hambali*, Skripsi Sarjana Fakultas Syariah IAIN Walisongo Semarang, 2013, hlm. 57. Pengambilan data-data biografi merujuk pada sumber sekunder dikarenakan pernyataan Slamet Hambali saat hendak diwawancarai agar dapat mengambil data biografi pada tulisan-tulisan terdahulu.

Krapyak Semarang Barat. Ia tinggal bersama istrinya, Hj. Isti'anah dan dua putrinya Rusda Kamalia dan Jamilia Husna. Selama tiga tahun Slamet Hambali juga pernah dipercaya menjadi ketua RT di lingkungan perumahannya.³

Mengenai jenjang pendidikan yang ia tempuh, dimulai dari Sekolah Rakyat Sambirejo, namun hanya berhenti sampai tingkat tiga dan selesai pada tahun 1966 setelah kembali dilanjutkan ke SR Rembes. Selanjutnya Slamet Hambali mulai masuk pondok pesantren di daerah Bancaan di bawah asuhan KH. Isom sekaligus melanjutkan pendidikannya di MTs NU Salatiga. Pada tahun 1969, ia lulus Madrasah Tsanawiyah kemudian melanjutkan Madrasah Aliyah di tempat yang sama dan lulus pada tahun 1977.⁴

Pada masa remaja itu juga ia pernah nyantri di sebuah pondok pesantren yang diasuh oleh KH Zubair Umar Al-Jaelany. Dari sinilah kemahirannya dalam ilmu falak mulai berkembang. Dibawah bimbingan langsung kyai Zubair, ia belajar falak dengan mendalami sebuah kitab falak bernama *Al-Khulashoh Al-Wafiyyah*, karangan sang kyai. Ia lulus S1 dari Fakultas Syariah IAIN Walisongo pada tahun 1979 M. dan S2 IAIN Walisongo tahun 2011M.⁵

³ Barokatul Laili, *Analisis Metode Pengukuran Arah Kiblat Metode Slamet Hambali...*, *op. cit.*, hlm 58.

⁴ *Ibid.*

⁵ Slamet Hambali, *Ilmu Falak Arah Kiblat Setiap Saat...*, *loc.cit.*

2. Metode Segitiga Siku-siku Slamet Hambali

Metode Slamet Hambali seperti yang dijelaskan pada bab pendahuluan merupakan metode pengukuran arah kiblat yang menggunakan segitiga siku-siku dengan memanfaatkan bayangan Matahari setiap saat. Terdapat dua jenis pengukuran yang ditawarkan dalam menentukan arah kiblat yaitu menggunakan satu segitiga siku-siku dan dua segitiga siku-siku. Metode tersebut menggunakan peralatan dan metode klasik yang sangat sederhana dan praktis karena hanya menggunakan tongkat yang berdiri tegak lurus yang memanfaatkan bayangan Matahari dan segitiga siku-siku sehingga diharapkan dapat digunakan oleh masyarakat.⁶

Hal ini tentu berbeda dengan kasus alat bantu theodolit yang harganya tergolong mahal sehingga tidak semua orang dapat membeli dan memilikinya. Theodolit sejauh ini memang dianggap sebagai alat dengan akurasi yang tinggi di antara metode-metode yang sudah ada dalam penentuan arah kiblat. Theodolite menentukan suatu posisi dengan tata koordinat horizon dan menentukan arah kiblat dengan hasil data yang akurat karena dapat menunjukkan sudut hingga satuan detik busur. Theodolite digital memiliki tingkat akurasi yang tinggi maksimal tingkat kesalahannya adalah 5".⁷

⁶ Lihat selengkapnya Slamet Hambali, *Ilmu Falak; Arah Kiblat Setiap Saat ..., op. cit.*

⁷ *Ibid.* hlm. 62.

Adapun metode pemikiran tentang arah kiblat Slamet Hambali adalah sebagai berikut :⁸

1. Menghitung arah kiblat (B) dan azimuth kiblat di lokasi yang akan diukur arah kiblatnya. Untuk mendapatkan arah kiblat digunakan rumus:

$$\text{Cotan B} = \text{cotan } \phi_k \sin \phi_x : \sin C - \cos \phi_x \text{ cotan } C^9$$

Untuk mempercepat dan mempermudah dalam melakukan hisab arah kiblat, rumus tersebut disederhanakan menjadi:

$$\text{Cotan B} = \tan \phi_k \cos \phi_x : \sin C - \sin \phi_x : \tan C^{10}$$

Keterangan:

B : adalah arah kiblat dihitung dari titik Utara atau selatan, jika perhitungan positif arah kiblat dihitung dari titik Utara, dan jika hasil perhitungan negatif dihitung dari titik Selatan. B juga bisa disebut busur arah kiblat atau sudut arah kiblat.

ϕ_k : adalah lintang Kakbah yaitu $21^{\circ} 25' 20,98''$.

ϕ_x : adalah lintang setempat yang akan dihitung arah kiblatnya.

C : adalah jarak bujur terdekat dari Kakbah ke Timur atau ke Barat sampai dengan bujur tempat yang akan diukur arah kiblatnya.

Rumus ini sangat sederhana karena data lintang Kakbah dan lintang tempat yang akan diukur arah kiblatnya langsung digunakan tanpa melalui 90° dikurangi lintang Kakbah dan 90° dikurangi lintang tempat.

⁸ *Ibid.* hlm. 144-146.

⁹ Lihat juga Slamet Hambali, *Ilmu Falak Penentuan Awal Waktu Shalat dan Arah Kiblat Seluruh Dunia*, Semarang: Pasca Sarjana IAIN Walisongo, 2011. hlm. 182.

¹⁰ *Ibid.*

Melalui rumus diatas kita akan menghasilkan arah kiblat suatu tempat yang kemudian akan dihitung azimuth kiblatnya.¹¹ Sebelum melakukan perhitungan azimuth kiblat sebaiknya terlebih dahulu harus mempersiapkan data-data lintang dan bujur tempat maupun Kakbah. untuk mendapatkan data garis bujur dan garis lintang yang akurat bisa menggunakan *Global Positioning System (GPS)*¹² atau *Google Earth*.

2. Menghitung arah Matahari (A) dan azimuth Matahari di lokasi pada saat pengambilan arah bayangan Matahari dengan menggunakan rumus :

$$\text{Cotan A} = \tan \delta m \cdot \cos \phi^x \div \sin t - \sin \phi^x \div \tan t^{13}$$

Keterangan:

- A = Arah Matahari.
 δm = Deklinasi Matahari.¹⁴
 ϕ^x = Lintang Tempat.
t = Sudut Waktu Matahari.

Sebelum melakukan perhitungan tersebut, kita perlu mempersiapkan data-data yang menunjang dalam perhitungan tersebut yakni data lintang (ϕ^x) dan bujur tempat (BB/BT) selain itu juga *equation*

¹¹ Untuk mendapatkan azimuth kiblat dapat digunakan rumus sebagai berikut:

- Jika B (arah kiblat) = UT; maka azimuth kiblatnya tetap
 Jika B (arah kiblat) = ST; maka azimuth kiblatnya adalah $180^\circ + B$
 Jika B (arah kiblat) = SB; maka azimuth kiblatnya adalah $180^\circ - B$
 Jika B (arah kiblat) = UB; maka azimuth kiblatnya adalah $360^\circ - B$

Selengkapnya lihat Slamet Hambali, *Ilmu Falak; Arah Kiblat Setiap Saat...*, *op. cit.*, hlm. 70.

¹² GPS merupakan alat ukur dalam menentukan koordinat dari suatu tempat dengan bantuan satelit. Lihat selengkapnya Slamet Hambali, *Ilmu Falak, Penentuan Awal Waktu Shalat & Arah Kiblat Seluruh Dunia*, Semarang:Program Pasca Sarjana IAIN Walisongo Semarang, 2011. hlm. 219-225.

¹³ Slamet Hambali, *Ilmu Falak; Arah Kiblat Setiap Saat...*, *op. cit.*, hlm. 70.

¹⁴ Deklinasi Matahari adalah busur pada lingkaran waktu yang diukur mulai titik perpotongan antara lingkaran waktu dengan lingkaran ekuator. Selengkapnya lihat Muhyiddin Khazin, *Ilmu Falak Dalam Teori dan Praktik*, cet III, Yogyakarta: Buana Pustaka, 2004.

of time (e),¹⁵ deklinasi Matahari, serta menghitung sudut waktu Matahari (t). Untuk mendapatkan hasil yang akurat diperlukan data yang akurat pula. Untuk mendapatkan data-data yang akurat bisa menggunakan tabel *ephemeris*, almanak nautika, astro info dan *astronomical algorithms*.

3. Menghitung sudut kiblat dari bayangan Matahari (Q) dengan rumus :

$$Q = \text{azimuth kiblat} - \text{azimuth Matahari}$$

Langkah-langkah yang perlu dilakukan untuk mendapatkan azimuth kiblat antara lain sebagai berikut:¹⁶

- a. Jika azimuth kiblat dikurangi azimuth Matahari sisanya tidak lebih dari 90° , maka sisa tersebut langsung ditetapkan sebagai sudut kiblat dari bayangan Matahari. Posisi arah kiblat berada di sebelah kanan bayangan Matahari jika sisanya positif dan arah kiblat berada di sebelah kiri bayangan Matahari jika sisanya negatif.
- b. Jika azimuth kiblat dikurangi (azimuth Matahari + 180°) sisanya tidak lebih dari 90° , maka sisa tersebut langsung ditetapkan sebagai sudut kiblat dari bayangan Matahari. Posisi arah kiblat berada di sebelah kanan bayangan Matahari jika sisanya positif dan arah kiblat berada di sebelah kiri bayangan Matahari jika sisanya negatif.

¹⁵ *Equation of Time* ialah perata waktu untuk menyesuaikan antara waktu matahari sebenarnya dan waktu matahari rata-rata. Selengkapnya baca Rinto Anugraha, *Mekanika Benda Langit*, materi perkuliahan Studi Fisika fakultas MIPA Universitas Gajah Mada. hlm. 76-78.

¹⁶ Slamet Hambali, *Ilmu Falak; Arah Kiblat Setiap Saat...*, op. cit. hlm. 86.

- c. Jika azimuth kiblat dikurangi ($\text{azimuth Matahari} - 180^\circ$) sisanya tidak lebih dari 90° , maka sisa tersebut langsung ditetapkan sebagai sudut kiblat dari bayangan Matahari. Posisi arah kiblat berada di sebelah kanan bayangan Matahari jika sisanya positif dan arah kiblat berada di sebelah kiri bayangan Matahari jika sisanya negatif.
- d. Jika $(360^\circ + \text{azimuth kiblat})$ dikurangi azimuth Matahari sisanya positif kurang dari 90° , maka ditetapkan langsung sebagai sudut kiblat dari bayangan Matahari dan posisi arah kiblat berada di sebelah kanan bayangan Matahari.
- e. Jika $(360^\circ + \text{azimuth kiblat})$ dikurangi ($\text{azimuth Matahari} + 180^\circ$) sisanya positif kurang dari 90° , maka ditetapkan langsung sebagai sudut kiblat dari bayangan Matahari dan posisi arah kiblat berada di sebelah kanan bayangan Matahari.

B. Aplikasi Kiblat Siku-siku

1. Deskripsi Umum Aplikasi

Aplikasi perhitungan arah kiblat metode satu segitiga siku-siku Slamet Hambali diberi nama aplikasi Kiblat Siku-siku. Aplikasi ini merupakan aplikasi android untuk mempermudah dalam penentuan arah kiblat menggunakan segitiga siku-siku karya pemikiran Slamet Hambali.

Aplikasi Kiblat Siku-siku menjadikan Matahari sebagai acuan penentuan arah kiblat, sehingga diperlukan proses perhitungan azimuth kiblat dan azimuth bayangan Matahari maupun azimuth Matahari itu sendiri. Pada proses perhitungan posisi matahari akan digunakan proses perhitungan Jean Meeus pada buku *Astronomical Algorithms*¹⁷ dengan tingkat akurasi yang berselisih maksimal satuan kecil detik busur pada perhitungan. Guna menunjang keakuratan juga digunakan buku yang telah diterjemahkan¹⁸ dan buku Rinto Anugraha berjudul *Mekanika Benda Langit*¹⁹ yang juga digunakan dalam kuliah Fisika Fakultas UGM. Ketiga buku tersebut digunakan untuk memudahkan dalam memahami perhitungan dan pengaplikasian.

Aplikasi Kiblat Siku-siku merupakan aplikasi penentu arah kiblat pertama yang menggunakan bantuan segitiga siku-siku dalam proses penentuan arah kiblatnya. Kiblat Siku-siku juga dilengkapi dengan kompas sebagai alat bantu lain dalam penentuan arah kiblat yang disesuaikan dengan hasil azimuth kiblat yang tertera pada aplikasi.

Pada bab Pendahuluan telah dijelaskan sepintas mengenai aplikasi Kiblat Siku-siku yang merupakan alat bantu dalam penentuan arah kiblat pemikiran Slamet Hambali dengan menggunakan segitiga siku-siku. Guna mempermudah perhitungan arah kiblat maka dikembangkan sebuah aplikasi Kiblat Siku-siku bagi pengguna *smartphone* android dalam

¹⁷ Jean Meeus, *Astronomical Algorithms*, Virgia : Willman-Bell. Inc, 1991.

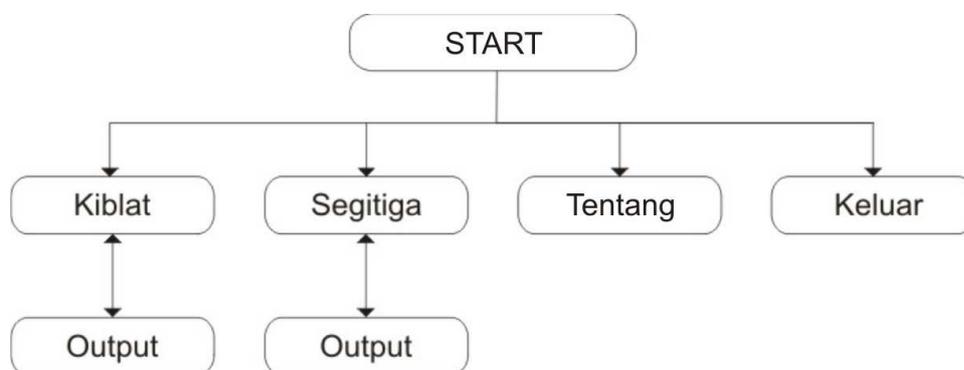
¹⁸ Jean Meeus, *Astronomical Algorithms*, Virgia : Willman-Bell. Inc, 1991. Diterjemahkan oleh Dr. Ing. Khafid sebagai Modul Kuliah Astronomi, IAIN Walisongo.

¹⁹ Rinto Anugraha, *Mekanika Benda Langit*, materi perkuliahan Studi Fisika Fakultas MIPA Universitas Gajah Mada.

penentuan arah kiblat menggunakan metode satu segitiga siku-siku Slamet Hambali.

Selain itu aplikasi Kiblat Siku-siku pun memanjakan para penggunanya karena didesain secara otomatis dalam mengambil data-data yang diperlukan seperti koordinat tempat, waktu dan tanggal saat itu. Dalam proses penentuan arah kiblat menggunakan segitiga siku-siku pengguna tetap melakukan proses penentuan arah kiblat dengan langsung terjun ke lapangan dengan data-data yang telah tertera pada aplikasi. Hal ini akan meminimalisir ketidakakuratan yang dihasilkan oleh aplikasi jika tetap dijalankan secara otomatis.

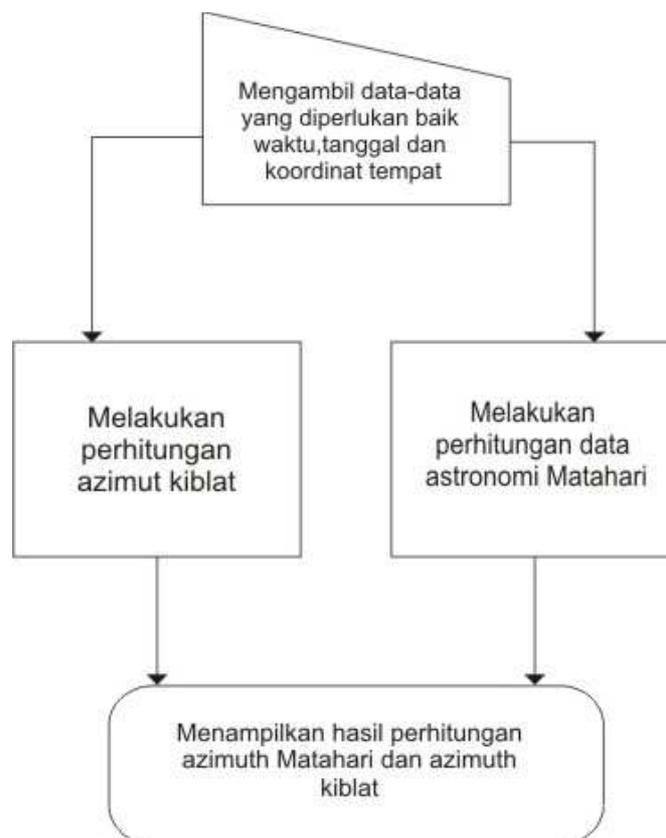
Kompas merupakan salah satu fitur yang ada pada aplikasi ini sebagai alat bantu bagi pengguna yang tidak ingin repot-repot untuk menentukan arah kiblat dengan metode Slamet Hambali. Dengan kompas dan data azimuth kiblat yang terdapat pada aplikasi, pengguna dapat memperkirakan arah mana yang merupakan arah kiblat namun dengan akurasi yang sangat rendah sesuai dengan perkiraan pengguna *smartphone* tersebut.



Gambar 3.1 Stuktur navigasi aplikasi

Struktur navigasi pada gambar 3.1 merupakan alur dari aplikasi Kiblat Siku-siku. Pada tampilan awal aplikasi akan menyediakan fitur menu yang akan dipilih oleh pengguna dan kemudian akan dieksekusi langsung setelah pengguna memilih pilihan tersebut dan akan berpindah ke *activity* yang lain yang telah dipilih.

Dari gambar 3.2 dapat dilihat alur aplikasi perhitungan Kiblat Siku-siku secara umum. Proses pertama yang dilakukan ialah pengambilan input data yang secara otomatis dilakukan oleh aplikasi dengan bantuan GPS pada android. Selanjutnya mengambil data tanggal dan waktu guna melakukan proses perhitungan data astronomis Matahari yang dilanjutkan dengan perhitungan beda azimuth.

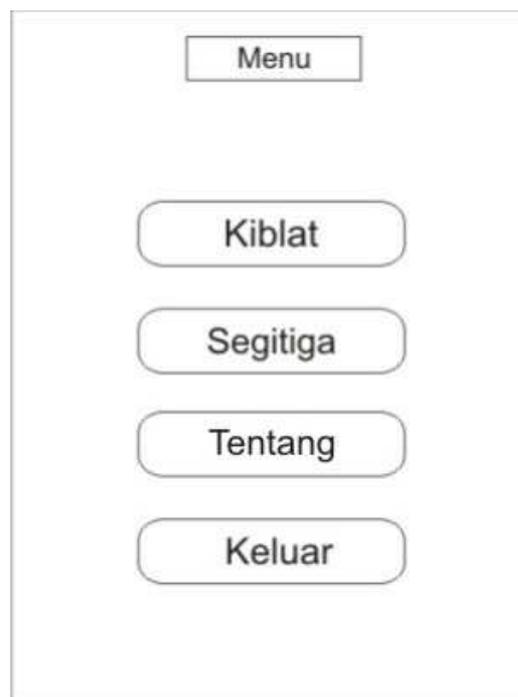


Gambar 3.2 Alur sistem Aplikasi Kiblat Siku-siku secara umum.

Aplikasi Kiblat Siku-siku dirancang dengan menggunakan bahasa pemrograman *android*. Bahasa pemrograman android merupakan pengembangan bahasa pemrograman *Java Micro 2 Edition (J2ME)*. Pada android terdapat elemen-elemen yang telah disempurnakan, contohnya ialah *Dalvik Virtual Manager* yang dirancang dan dikostumisasi guna memastikan bahwa beberapa *feature-feature* berjalan lebih efisien pada perangkat *mobile*.²⁰

2. Perancangan Antar Muka (*User Interface*)

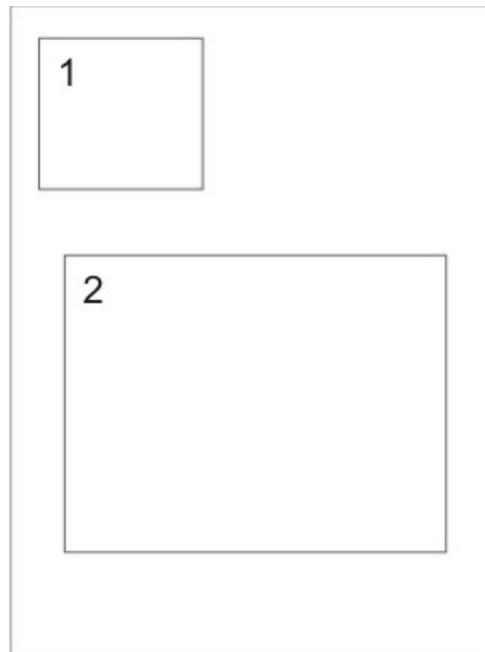
a. Perancangan Tampilan Utama



Gambar 3.3 Tampilan Utama Menu Aplikasi

²⁰ Nazaruddin safaat. *Android "Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android"*, Bandung: Informatika, 2012. hlm. 4.

b. Perancangan Tampilan Kiblat



Gambar 3.4 Tampilan Class.Kiblat

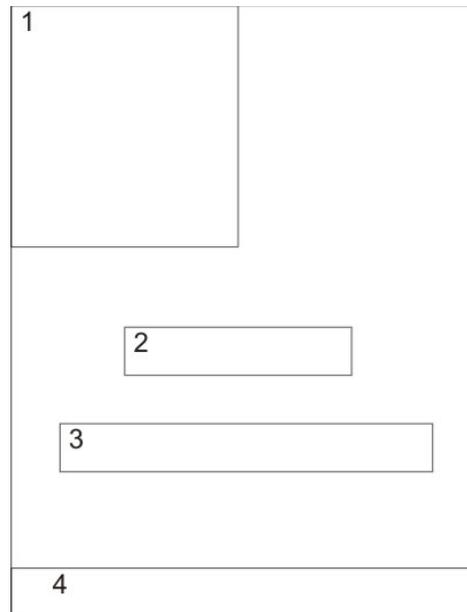
1. Teks

Pada bagian ini akan menampilkan teks-teks yang berisi data lintang dan bujur tempat pengamat menggunakan GPS dan menampilkan azimuth kiblat serta azimuth kompas.

2. Fitur Kompas

Pada bagian ini akan menampilkan fitur kompas dengan *Image View* yang dapat berubah sesuai dengan azimuth yang ditampilkan pada *mobile*.

c. Perancangan Tampilan Segitiga



Gambar 3.5 Tampilan Perhitungan Segitiga Siku

1. Teks

Pada bagian ini akan menampilkan teks-teks yang berisi data lintang dan bujur tempat pengamat menggunakan GPS serta data-data azimuth dan beda sudut untuk penentuan arah kiblat menggunakan segitiga siku-siku.

2. Teks

Menampilkan teks Panjang Garis yang dapat dirubah sesuai kebutuhan pengguna dalam menentukan arah kiblat

3. Teks

Menampilkan hasil perhitungan Panjang Sisi.

4. Tombol

Bagian ini menampilkan tombol yang berfungsi sebagai petunjuk penggunaan aplikasi Kiblat Siku dalam pengukuran arah kiblat.

3. Spesifikasi Perangkat Keras dan Lunak

Dalam penerapan dan perancangan aplikasi, terdapat beberapa hal yang diperlukan. Perangkat keras dan perangkat lunak merupakan hal-hal yang diperlukan dalam perancangan tersebut.

a. Spesifikasi Perangkat Keras

Dalam perancangan yang telah dijelaskan sebelumnya dibutuhkan beberapa hal seperti perangkat keras yakni untuk penyajian aplikasi. Adapun alat-alat yang dibutuhkan adalah:

1) *Mobilephone* Dengan Sistem Operasi Android

Mobilephone digunakan untuk menjalankan program aplikasi yang telah dikembangkan. Adapun handphone yang digunakan adalah Andromax I dengan OS Android *Ice Cream Sandwich* (Android 4.0) sebagai contoh dalam menjalankan Aplikasi Kiblat Siku-siku.

2) Laptop Dell Inspiron N4030

Laptop Dell Inspiron N4030 digunakan untuk merancang aplikasi Kiblat Siku dengan spesifikasi chipset Intel Pentium 2.13Ghz, 2048MB RAM, ATI Mobility Radeon HD 5430.

b. Spesifikasi Perangkat Lunak

Adapun perangkat lunak yang digunakan dalam proses pembuatan aplikasi perhitungan arah Kiblat Metode Slamet Hambali adalah sebagai berikut :

- 1) *Java Development Kit (JDK)* dan *Java Runtime Environment (JRE)* untuk instalasi bahasa pemrograman Java.
- 2) *Integrated Development Environment (IDE) Eclipse Juno* yang berfungsi untuk mengolah bahasa pemrograman serta perhitungan yang dibutuhkan, sehingga memudahkan dalam pengembangan aplikasi.
- 3) *Android Software Development Kit (Android SDK)* yang berfungsi untuk menyediakan emulator dan perlengkapan lain dalam pembuatan sebuah aplikasi.
- 4) *Android Development Tools (ADT)* untuk memberikan kemudahan dalam pengembangan aplikasi, membuat tampilan aplikasi yang menarik, menambahkan komponen dan menyiapkan aplikasi yang telah dikembangkan untuk di distribusikan.
- 5) *Microsoft Windows 7* sebagai sistem operasi yang digunakan untuk merancang Aplikasi Kiblat Siku.
- 6) *Corel Draw X4* untuk membuat tampilan yang menarik.
- 7) *Android VDM (Virtual Device Manager)* untuk emulator aplikasi Kiblat Siku-siku.

C. Implementasi Perancangan Aplikasi

Pada pembahasan ini akan dijelaskan mengenai implementasi perancangan Aplikasi Perhitungan *Kiblat Siku-siku* dengan menggunakan *Eclipse Juno 4.0*. Sebelumnya akan terlebih dahulu menjelaskan beberapa komponen penting lainnya dalam pemrograman.

1. Pengambilan Tanggal dan Waktu

Pada aplikasi *Kiblat Siku* ini mengambil data tanggal dan waktu secara *realtime* sesuai dengan yang terdapat pada perangkat *mobile* android. Dikarenakan android merupakan sistem operasi yang sangat canggih, setting tanggal dan waktu pada perangkat *mobile* android biasanya telah otomatis diatur secara *realtime* kecuali jika pengguna telah merubah settingan *default* tersebut.

Data-data tanggal dan waktu berupa jam, menit dan detik pada perangkat dapat diambil untuk kemudian diimplementasikan pada proses perhitungan selanjutnya. Selain data tersebut juga bisa langsung dapat mengambil data *time zone* untuk wilayah tertentu misalnya +7 GMT untuk Indonesia. Adapun *coding* untuk pengambilan data-data tersebut antara lain adalah sebagai berikut:

```
import java.util.Calendar;
import java.util.Date;
Calendar cal = Calendar.getInstance();
cal.setTime(new Date());
double tz = cal.get(Calendar.ZONE_OFFSET);
int thn = cal.get(Calendar.YEAR);
int bln = cal.get(Calendar.MONTH)+1;
int tgl = cal.get(Calendar.DATE);
jm = cal.get(Calendar.HOUR_OF_DAY);
mnt = cal.get(Calendar.MINUTE);
dtk = cal.get(Calendar.SECOND);
```

2. Pengambilan Koordinat Lokasi

Perangkat *mobile* yang memiliki sistem operasi android telah dilengkapi perangkat GPS untuk memudahkan dalam mengetahui koordinat tempat dari pengguna android. Oleh sebab itu aplikasi Kiblat Siku-siku juga menggunakan GPS guna pengambilan koordinat lokasi pengguna *mobile* untuk memudahkan pengguna dan mendapatkan hasil yang berakurasi tinggi. Adapun *coding* dalam pengaktifan GPS dan pengambilan koordinat tempat pengamatan antara lain sebagai berikut:

```
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;

lm2=(LocationManager) getSystemService(Context.LOCATION_
SERVICE);
locListener = new MyLocationListener();
lm2.requestLocationUpdates(LocationManager.GPS_PROVIDER,
600, 150,locListener);
double latTempat= loc.getLatitude();
double lonTempat= loc.getLongitude();
```

3. Perhitungan Azimuth Kiblat

Implementasi perhitungan azimuth kiblat merupakan penulisan *coding* rumus arah kiblat. Rumus arah kiblat sendiri telah dijelaskan sebelumnya pada bab dua. Data yang perlu disiapkan terlebih dahulu adalah data lintang dan bujur Kakbah, serta data letak geografis tempat akan secara otomatis terinput menggunakan GPS. Adapun data koordinat Kakbah ialah $21^{\circ} 25' 20,98''$ LU dan $39^{\circ} 49' 34.22''$. Setelah

didapatkan data-data tersebut baru diimplementasikan dengan coding sebagai berikut :

```
// azimuth kiblat
double kiblat, arah;
    double SB=(lngKabah-lonTempat);
    double sinsbmd= Math.sin(SB * Math.PI/180);
    double cossbmd= Math.cos(SB * Math.PI/180);
    double sinLT = Math.sin(latTempat * Math.PI/180);
    double cosLT = Math.cos(latTempat * Math.PI/180);
    double tanLM = Math.tan(latKabah * Math.PI/180);
    double penyebut= (cosLT * tanLM) - sinLT * cossbmd;

    kiblat = Math.atan2(sinsbmd, penyebut) * 180/Math.PI;
    arah= kiblat<0 ? kiblat+360 : kiblat;

int drjt21=(int)(arah);
int mnt22=(int)((Math.abs(arah)-Math.abs(drjt21))*60);
int dtk24=(int)((((Math.abs(arah)-Math.abs(drjt21))*60)-
Math.abs(mnt22))*60);
AKTxt.setText(" "+ drjt21 + "° " + mnt22 + "' " + dtk24 + "" "
+"Degrees" );
```

4. Pembuatan Kompas

Kompas pada aplikasi Kiblat Siku-siku merupakan alat bantu dalam menentukan arah kiblat bagi pengguna yang ingin menentukan arah kiblat dimanapun dan kapanpun dengan hadirnya perhitungan azimuth kiblat dan azimuth pada kompas. Sehingga memungkinkan untuk berijtihad dengan mengira-ngira arah kiblat yang sebenarnya dengan bantuan kompas dan perhitungan azimuth kiblat tersebut.

Coding untuk pembuatan kompas antara lain sebagai berikut:

```
import android.hardware.Sensor;
import android.hardware.SensorEvent;
import android.hardware.SensorEventListener;
import android.hardware.SensorManager;
import android.view.animation.Animation;
import android.view.animation.RotateAnimation;
```

```

mSensorManager = (SensorManager)
getSystemService(SENSOR_SERVICE);
mSensorManager.registerListener(this,
mSensorManager.getDefaultSensor(Sensor.TYPE_ORIENTATION),
    SensorManager.SENSOR_DELAY_GAME);
mSensorManager.unregisterListener(this);
public void onSensorChanged(SensorEvent event) {
    float degree = Math.round(event.values[0]);

    tvHeading.setText("Azimuth: " +
Float.toString(degree) + " Degrees");
    RotateAnimation ra = new RotateAnimation(
        currentDegree,
        -degree,
        Animation.RELATIVE_TO_SELF, 0.5f,
        Animation.RELATIVE_TO_SELF,
        0.5f);

    ra.setDuration(210);
    ra.setFillAfter(true);
    image.startAnimation(ra);
    currentDegree = -degree;

```

5. Perhitungan Azimuth Matahari

Menentukan azimuth Matahari membutuhkan perhitungan yang panjang karena diperlukan data-data pendukung dalam penentuan azimuth Matahari salah satunya ialah deklinasi Marahari, sudut waktu Matahari dan lain sebagainya.

Adapun penjabaran langkah-langkah dalam perhitungan azimuth Matahari dan *coding* programnya antara lain sebagai berikut :

a. Menentukan Julian Day

Julian day adalah perhitungan yang berlanjut berupa pecahan yang hitung mulai tahun -4712. *Julian day* dimulai tepat pada siang hari yaitu pukul 12 *Universal Time* (UT). Julian Day

selalu berkorespondensi dengan *Dynamical Time* atau *Julian Ephemeris Day (JDE)*.²¹

Julian Day (JD) tersebut juga dapat didefinisikan sebagai banyaknya hari yang dilalui sejak Senin tanggal 1 Januari tahun 4713 sebelum Masehi pada pertengahan hari atau pukul 12:00:00 UT (Universal Time). Tahun 4713 SM sama dengan tahun -4712.

²² Adapun *coding* untuk menentukan Julian Day dalam program ini adalah:

```
//julian day
int month, years, x, y;
double tah, DT, JD;
if (bln <= 2){
    month = bln + 12;
} else {month = bln+1;}
if (bln <= 2){
    years = thn - 1;
} else {years = thn;}

//perhitungan
waktu = jm+mnt/60+dtk/3600;
x = ( years / 100);
int io = x/4;
y = 2+io-x;
JD=1720994.5+(int)(365.25*years)+(int)(30.60001*
(month+1))+y+tgl+waktu/24-timezone;
```

b. Menentukan *Greenwich Sideral Time (GST)* dan *Local Sideral Time (LST)* untuk waktu *Universal Time (UT)*

Waktu yang digunakan sehari-hari merupakan waktu *solar time* yang sama dengan 24 jam. Sementara itu, satu *sidereal day* sama dengan 23 jam 56 menit 4 detik *solar time*. Waktu untuk

²¹ Jean Meeus, *Astronomical Algorithms...*, *op. cit.*, hlm. 59.

²² Rinto Anugraha, *Mekanika Benda Langit ...*, *op. cit.*, hlm. 8.

menunjukkan *sidereal time* ialah Greenwich Sidereal Time (GST), sedangkan waktu untuk *solar time* adalah UT.²³ Adapun *coding* untuk menentukan GST dan LST (Local Sidereal Time) suatu tempat ialah sebagai berikut:

```
double GST1x = (6.6973745583 + 2400.0513369072*T1 +
0.0000258622*T1*T1)%24;
    double GST1;
    if(GST1x<0){
        GST1=(double)(GST1x+24);
    }else{
        GST1=(double)GST1x;}
        double GST3x = (GST1 + (waktu-
tz/3600/1000) * 1.00273790935)%24;
    if(GST3x<0){
        GST3=(double)GST3x+24;
    }else{
        GST3=(double)GST3x;}
double LST = (GST3 + (lonTempat/15));
```

c. Menentukan Deklinasi Matahari

Deklinasi Matahari adalah busur pada lingkaran waktu yang diukur mulai titik perpotongan antara lingkaran waktu dengan lingkaran ekuator ke arah utara atau selatan sampai ke titik pusat benda langit.²⁴ Adapun *coding* untuk menentukan deklinasi Matahari ialah sebagai berikut :

```
double sinLambda = Math.sin(Lambda * Math.PI/180);
double cosLambda = Math.abs(Math.cos(Lambda*Math.PI/180));
double cosEpsilon = Math.cos(Epsilon * Math.PI/180);
double X = Math.abs(cosEpsilon * sinLambda);
double Alpha = Math.atan(X / cosLambda)*180/Math.PI;
double sinEpsilon = Math.sin(Epsilon * Math.PI/180);
double Dek = Math.asin(sinEpsilon*sinLambda)*180/Math.PI;
```

²³ *Ibid.* hlm. 23.

²⁴ Muhyiddin Khozin, *Ilmu Falak Dalam Teori dan Praktik*, cet III, Yogyakarta: Buana Pustaka, 2004, hlm. 51.

d. Menentukan *Equation of Time*

Equation of time merupakan perata waktu untuk menyesuaikan waktu matahari yang sesungguhnya. *Equation of time* dapat bernilai positif (+) dan dapat bernilai negatif (-)²⁵.

```
double eq1 = -(1789+237* T) * sinlo - (7146-62*T) * coslo;
double eq2 = (9934-14*T) * sinlo2 - (29+5*T) * coslo2 +
(74 + 10*T)*sinlo3;
double eq3 = (320-4*T) * coslo3 - 212* sinlo4;
double eq = ((eq1 + eq2 + eq3)/1000)/60;
```

e. Menentukan Azimuth Matahari

Dalam menentukan azimuth Matahari ini menggunakan metode *coding* yang hampir sama dalam menentukan metode menghitung azimuth kiblat. Adapun *coding* menghitung azimuth Matahari antara lain:

```
// Menghitung Azimuth Matahari
double arh2, hasilku;
double t= ((waktu + eq -12 -(BT-lonTempat)/15))*15;

double tanDek = Math.tan(Dek * Math.PI/180);
double sinLin = Math.sin(latTempat * Math.PI/180);
double cosLin = Math.cos(latTempat * Math.PI/180);
double sint= Math.sin(t * Math.PI/180);
double cost= Math.cos(t * Math.PI/180);
double penyeb= (cosLin * tanDek) - sinLin * cost;
double hasilku1 = Math.atan2(sint, penyeb) *
180/Math.PI;
hasilku = hasilku1 * -1;
arh2= hasilku<0 ? hasilku+360 : hasilku;
int drjt78=(int)(arh2);
int mnt79=(int)((Math.abs(arh2)-
Math.abs(drjt78))*60);
int dtk80=(int)((((Math.abs(arh2)-
Math.abs(drjt78))*60)-Math.abs(mnt79))*60);
```

²⁵ Rinto Anugraha, *Mekanika Benda Langit ..., op. cit.*, hlm. 76-78.

```

        AMTxt.setText(" " + drjt78 + "° " + mnt79 +
        "' " + dtk80 + "' " + "Degrees" );

```

f. Menentukan Sudut Kiblat dan Panjang Sisi

Pada dasarnya sudut kiblat dari bayangan Matahari ialah jarak antara azimuth kiblat dengan azimuth Matahari.²⁶ Hal ini seperti yang telah dijelaskan sebelumnya. Adapun *coding* menghitung azimuth Matahari ialah sebagai berikut:

```

double Qs;
if (arah-arh2>0 && arah-arh2<90){
    Qs=arah-arh2 ;
} else if (arah-arh2<0 && Math.abs(arah-arh2)<90){
    Qs=arah-arh2 ;
} else if(arah-(arh2+180)>0 && arah-(arh2+180)<90){
    Qs=arah-(arh2+180) ;
} else if(arah-(arh2+180)<0 && Math.abs(arah-
(arh2+180))<90){
    Qs=arah-(arh2+180);
} else if(arah-(arh2-180)>0 && arah-(arh2-180)<90){
    Qs=arah-(arh2-180) ;
} else if(arah-(arh2-180)<0 && Math.abs(arah-(arh2-
180))<90){
    Qs=arah-(arh2-180);
} else if ((arah+360)-arh2>0 && (arah+360)-arh2<90){
    Qs=(arah+360)-arh2 ;
} else { Qs= (arah+360)-(arh2+180);}

int drjt31=(int)(Qs);
int mnt32=(int)((Math.abs(Qs)-Math.abs(drjt31))*60);
int dtk34=(int)((((Math.abs(Qs)-
Math.abs(drjt31))*60)-Math.abs(mnt32))*60);
TX.setText(" " + drjt31 + "° " + mnt32 + "' " + dtk34
+ "' " + "Degrees" );
int panjang= Integer.parseInt
(bedaTxt.getText().toString());
double hasilx = Math.tan(( Math.abs ( Qs ) ) *
Math.PI / 180) * panjang ;
pnjgTxt.setText(String.valueOf(hasilx));

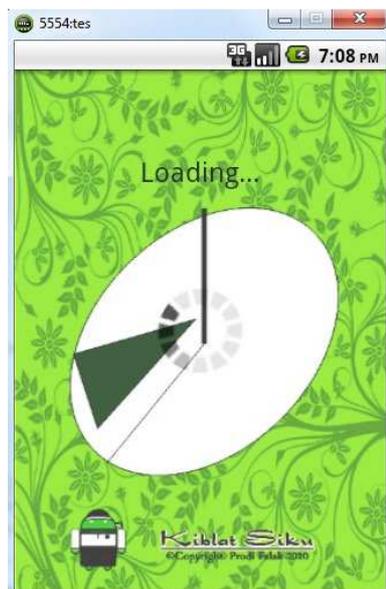
```

²⁶ Lihat selengkapnya Slamet Hambali, *Ilmu Falak; Arah Kiblat Setiap Saat...*, op. cit. hlm. 86.

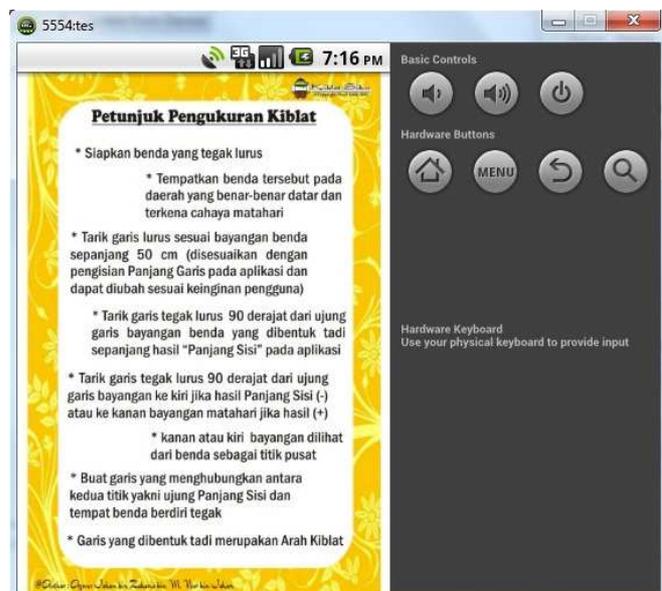
D. Implementasi Antarmuka Aplikasi Kiblat Siku

Bagian ini merupakan bagian uji hasil implementasi desain antarmuka setelah semua bahasa pemrograman ditulis dan dijalankan pada emulator android sehingga secara otomatis program Eclipse meng-*compile* aplikasi menjadi file *.apk* yang dapat diaplikasikan langsung pada *smartphone*. Karena aplikasi Kiblat Siku-siku menggunakan GPS dalam penggunaan aplikasi, sangat dianjurkan bagi pengguna untuk menyalakan fitur GPS pada *smartphone* android pengguna dan mengizinkan internet untuk mendapatkan sinyal GPS yang lebih cepat dan akan menghabiskan daya baterai lebih cepat. Fitur kompas pada aplikasi juga tidak akan berjalan dengan baik apabila fitur *compass* pada *smartphone* tidak mendukung untuk dijalkannya kompas.

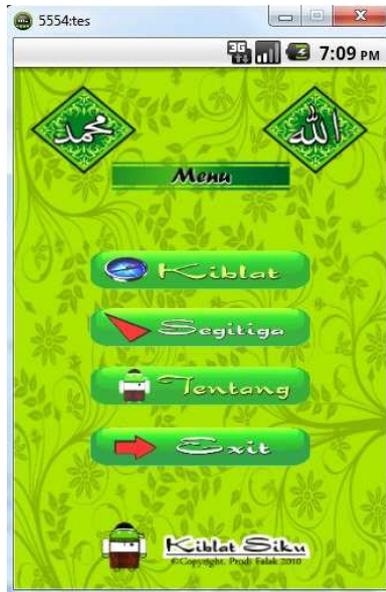
Adapun implementasi dari aplikasi Kiblat Siku-siku antara lain sebagai berikut :



Gambar 3.6 Desain antarmuka tampilan *splashscreen* pada tampilan pembuka.



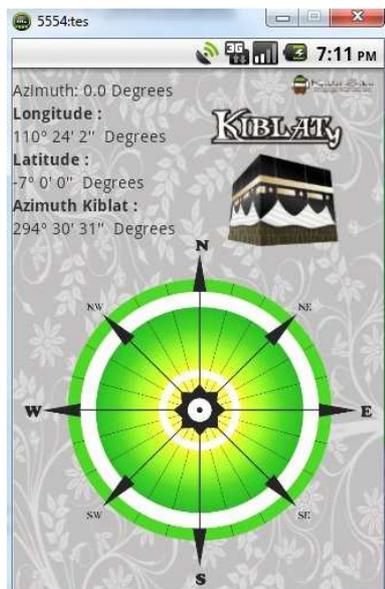
Gambar 3.7 Desain antarmuka tampilan petunjuk penggunaan aplikasi KiblatSiku



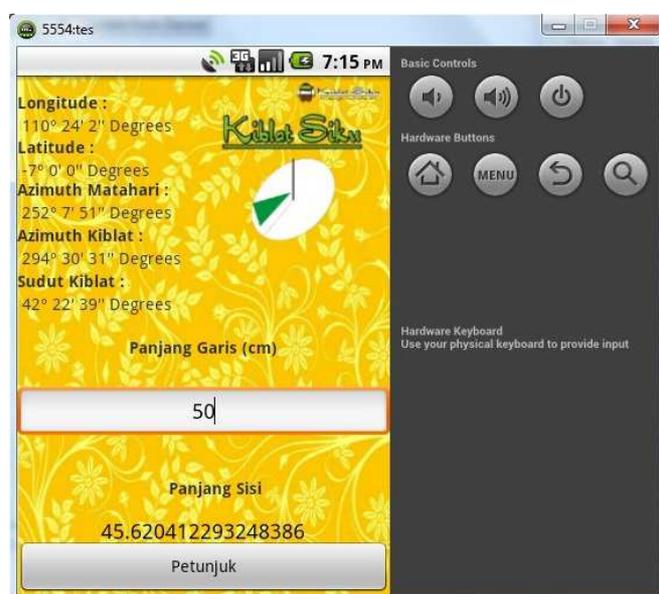
Gambar 3.8 Desain antarmuka tampilan List Menu pada tampilan pembuka.



Gambar 3.9 Desain antarmuka tampilan menu Tentang Metode Slamet Hambali.



Gambar 3.10 Desain antarmuka tampilan menu Kiblat dan fitur kompas pembantu penentu kiblat.



Gambar 3.11 Desain antarmuka tampilan hasil perhitungan Kiblat Siku Metode Slamet Hambali

Gambar 3.6 di atas merupakan hasil antarmuka dari tampilan *splashscreen* pada aplikasi Kiblat Siku-siku saat aplikasi baru dijalankan. Gambar 3.7 merupakan antarmuka tampilan yang menunjukkan cara penentuan arah kiblat menggunakan *list* menu “Segitiga” untuk metode pengukuran menggunakan satu segitiga siku-siku. Pada gambar 3.8 terlihat hasil tampilan utama menu pada aplikasi Kiblat Siku-siku sebagai hasil dari rancangan awal pada gambar 3.3.

Pada gambar 3.9 terdapat tampilan antarmuka yang akan ditemukan apabila memilih *list* menu “Tentang”. Pilihan ini akan menunjukkan biografi singkat Slamet Hambali dan metode Slamet Hambali. Fitur kompas pada *list* menu “Kiblat” terlihat pada gambar 3.10. Fitur kompas ini sebagai tambahan untuk menentukan arah kiblat bagi pengguna aplikasi yang tidak memilih untuk menggunakan metode segitiga siku-siku. Gambar 3.11 merupakan tampilan perhitungan dari aplikasi Kiblat Siku-siku guna mempermudah proses perhitungan arah kiblat dan posisi Matahari.