

## RANCANG BANGUN PERANGKAT LUNAK PENENTU ARAH KIBLAT, PENGHITUNG WAKTU SHALAT DAN KONVERSI KALENDER HIJRIYYAH BERBASIS *SMARTPHONE* ANDROID

*M. Didik R. Wahyudi*

Jurusan Teknik Informatika FST UIN Sunan Kalijaga  
*E-Mail* : m.didik@uin-suka.ac.id

### **ABSTRACT**

*As muslims, it becomes our duty to continuously improve the quality of worship. 5 time praying on time with the right mecca position is a key requirement of a Muslim. The rapid increase in the use of Android smartphones and demands to improve the quality of worship is getting better, then it takes an application that is able to meet those needs. This Android application can meet the needs in the hopes of improving the quality of worship. In addition, that Muslims are familiar with Hijriyyah calendar, then the application is also equipped with a built Hijriyyah calendar. By using this application, is expected to help Muslims to worship ALLAH SWT.*

**Keyword:** *Islamic Applications, Compass Qibla, Adhan Alarm, Calendar Hijriyyah*

### **PENDAHULUAN**

Perkembangan teknologi yang semakin maju dan canggih, mengakibatkan adanya pergeseran bentuk dan fungsi komputer. Komputer hadir dalam bentuk yang lebih sederhana yaitu komputer tablet dan *smartphone*. Komputer tablet adalah suatu komputer portabel lengkap yang seluruhnya berupa layar sentuh datar, tidak menggunakan keyboard dan mouse sebagai media input. Sedangkan *smartphone* adalah telepon genggam yang mempunyai kemampuan layaknya sebuah komputer.

Pemakaian komputer tablet dan *smartphone* untuk melakukan akses internet mencapai angka sekitar 10% dari keseluruhan pemakai internet. Besarnya pertumbuhan pemakai perangkat mobile inilah yang memicu perkembangan aplikasi berbasis mobile. Aplikasi berbasis mobile yang didominasi oleh Android dan iOS semakin beragam, mulai dari aplikasi hiburan, perkantoran serta aplikasi untuk keperluan

ibadah.

Sebagai seorang muslim, sholat lima waktu adalah ibadah wajib. Agar sholat tersebut memenuhi syarat syahnya sholat maka diantaranya harus menghadap kiblat dan sudah memasuki waktu sholat secara tepat. Sebagai seorang muslim sudah saatnya untuk lebih terbiasa dengan penganggalan hijriyyah. Kalender hijriyyah ini merupakan kalender yang digunakan sebagai rujukan penanggalan oleh umat muslim, disamping kalender masehi. Kalender hijriyyah ini dipergunakan untuk penentuan hari – hari besar dalam Islam. Dengan tujuan agar umat Islam lebih mudah untuk memahami dan mengerti hari – hari besar dalam agama Islam, maka dilakukanlah penelitian mengenai konversi tanggal masehi kedalam hijriyyah dengan perangkat mobile android ini.

## TINJAUAN PUSTAKA

### GPS (Global Positioning System)

GPS (Global Positioning System) adalah sistem untuk menentukan posisi di permukaan bumi dengan bantuan sinkronisasi sinyal satelit. Sistem ini menggunakan 24 satelit yang mengirimkan sinyal gelombang mikro ke Bumi. Sinyal ini diterima oleh alat penerima di permukaan, dan digunakan untuk menentukan posisi, kecepatan, arah, dan waktu. Sistem yang serupa dengan GPS antara lain GLONASS Rusia, Galileo Uni Eropa, IRNSS India. GPS dikembangkan oleh Departemen Pertahanan Amerika Serikat, dengan nama lengkapnya adalah NAVSTAR GPS. NAVSTAR adalah nama yang diberikan oleh John Walsh, seorang penentu kebijakan penting dalam program GPS. Cara kerja GPS *receiver* adalah setiap satelit GPS memancarkan sinyal-sinyal gelombang mikro. GPS receiver menggunakan sinyal satelit yang diterima untuk melakukan triangulasi posisi dengan cara mengukur lama perjalanan waktu sinyal dikirimkan dari satelit, kemudian mengalikannya dengan kecepatan cahaya untuk menentukan secara tepat berapa jauh dirinya dari satelit. Dengan mengunci minimum 3 sinyal dari satelit yang berbeda, maka GPS receiver dapat menghitung posisi tetap sebuah titik yaitu koordinat posisi lintang dan bujur (*Latitude & Longitude*).

### Bearing

Bearing adalah nilai derajat arah yang direpresentasikan menjadi sudut. Sudutnya diukur dari utara maupun selatan, ke arah timur ataupun barat, setiap sudut tersebut menghasilkan derajat kurang dari 90°. Kuadran yang dipakai ditunjukkan dengan huruf N (utara) atau S (selatan) mendahului sudutnya dan E (timur) atau W (Barat) mengikutinya.

### Azimuth

*Azimuth* adalah sudut yang diukur searah jarum jam dari sembarang meridian acuan, apabila berlawanan dengan arah jarum jam dan arah berlawanan tersebut melewati titik acuan,

maka nilai sudut pergeseran tersebut dapat direpresentasikan dengan *minus* (-). Dalam pengukuran tanah datar, *Azimuth* biasanya diukur dari utara, tetapi para ahli astronomi, militer dan *National Geodetic Survey* memakai selatan sebagai arah acuan.

### Algoritma Kuwaiti

Algoritma kuwaiti adalah algoritma hasil penelitian dari Top developer microsoft yang tergabung dalam Microsoft's Middle East Products Division (MEPD), yang dibentuk khusus untuk mencari atau memecahkan penanggalan hijriyah, kalender hijriyah tersebut dirasa sangat penting dan umumnya dipakai di saudi arabia maupun kota – kota besar seperti kuwait, sehingga algoritma kuwaiti ini sering dipakai didalam aplikasi produk dari microsoft diantaranya sistem operasi Arabic locales support, Microsoft Office, COM, Visual Basic, VBA, and SQL Server 2000.

### Sistem Operasi Android

Merupakan sistem operasi yang dikembangkan oleh google dengan menggunakan kernel linux. Sistem operasi ini bersifat open source, sehingga memudahkan para developer untuk bebas mengembangkan aplikasi dalam sistem operasi android ini. Android SDK (Software Development Kit) menyediakan alat dan API (Application Programming Interface) yang diperlukan untuk memulai mengembangkan aplikasi pada platform Android menggunakan bahasa pemrograman Java.

Dengan tersedianya library didalam pemrograman android tersebut memudahkan developer untuk mengembangkan aplikasi ke arah pengendalian sensor device android. Dua penjelasan library sensor yang ada dalam pemrograman android diantaranya adalah *sensor\_magnetic\_field* dan *sensor\_accelerometer* digunakan untuk menentukan pitch (sumbu y) dan roll untuk (z). sehingga menghasilkan nilai *true north orientation azimuth* (sumbu x), dan menghasilkan arah utara *compass*.

## METODE PENELITIAN

Metode yang digunakan dalam penelitian ini adalah metode studi pustaka yaitu meneliti dan mengkaji literatur-literatur baik berupa buku, paper maupun jurnal yang berhubungan dengan materi penelitian. Hasil yang diperoleh akan disimulasikan dalam bentuk program komputer.

Tahap-tahap yang akan dilakukan terdiri dari :

### 1. Tahap Inisiasi

Pada tahap ini dilakukan penelusuran dan pengkajian literatur-literatur yang berhubungan dengan sistem operasi Android, pemrograman JAVA, rumus konversi kalender masehi ke hijriyyah, rumus penentuan arah kiblat dan rumus penentuan waktu sholat 5 waktu.

### 2. Tahap Investigasi

Pada tahap ini dilakukan investigasi terhadap kebutuhan sistem yang akan dipergunakan untuk membangun aplikasi ini dan kebutuhan sistem untuk implementasi sistem ini. Investigasi meliputi penentuan versi platform/ sistem operasi, desain user interface, proses perhitungan rumus serta karakter jenis output yang dibutuhkan.

### 3. Tahap Prototype

Pada tahap ini mulai dibangun perangkat lunak untuk penentu arah kiblat, kompas kiblat dan konversi kalender hijriyyah.

### 4. Tahap Validasi/Pengujian

Pada tahap ini dilakukan pengujian terhadap prototype aplikasi yang sudah dibuat untuk memastikan bahwa aplikasi tersebut sudah sesuai dengan yang dibutuhkan. Pengujian dilakukan pada desain interface, akurasi perhitungan, kompatibilitas yaitu berupa instalasi prototype pada *smartphone* yang berbeda merk dan konfigurasi perangkat keras.

## HASIL DAN PEMBAHASAN

### MENENTUKAN ARAH KOMPAS MENUJU LOKASI KIBLAT

Penentuan arah kiblat pada perangkat android memanfaatkan *Sensor.Type\_Accelerometer* Dan *Sensor.Type\_Magnetic\_Field* untuk menentukan arah utara kompas, sehingga apabila device android tersebut diputar – putar kekiri ataupun kekanan, maka arah utara kompas pada device android tersebut tetap pada posisi utara yang benar, *true north* nantinya akan digunakan untuk penentuan arah kiblat menurut pergeseran azimuth yang didapat.

```
public void onSensorChanged(SensorEvent event) {
    final float alpha = 0.97f;
    synchronized (this) {
        if (event.sensor.getType() == Sensor.TYPE_ACCELEROMETER) {
            mGravity[0] = alpha * mGravity[0] + (1 - alpha)
            * event.values[0];
            mGravity[1] = alpha * mGravity[1] + (1 - alpha)
            * event.values[1];
            mGravity[2] = alpha * mGravity[2] + (1 - alpha)
            * event.values[2];
        }
        if (event.sensor.getType() == Sensor.TYPE_MAGNETIC_FIELD)
        {
            mGeomagnetic[0] = alpha * mGeomagnetic[0] + (1 - alpha)
            * event.values[0];
            mGeomagnetic[1] = alpha * mGeomagnetic[1] + (1 - alpha)
            * event.values[1];
            mGeomagnetic[2] = alpha * mGeomagnetic[2] + (1 - alpha)
            * event.values[2];
        }
        float R[] = new float[9];
        float I[] = new float[9];
        boolean success = SensorManager.getRotationMatrix(R, I, mGravity, mGeomagnetic);
        if (success) {
            float orientation[] = new float[3];
            SensorManager.getOrientation(R, orientation);
            azimuth = (float) Math.toDegrees(orientation[0]); //ambil azimuth
            loc2.setLatitude(21.422505); //Latitude Ka'bah
            loc2.setLongitude(39.826169); //Longitude Ka'bah
            azimuth = azimuth + loc1.bearingTo(loc2); //azimuth + bearing
            AdjustArrow();
        }
    }
}
```

Gambar 1 : Skrip program untuk penentuan arah kompas menuju qiblat

### Kalkulasi *bearing* antara posisi lokasi *AppCompassHijri* dengan posisi lokasi *ka'bah*

Pengukuran bearing dilakukan dengan menetapkan arah utara kompass sebagai posisi acuan semula, sehingga perhitungan derajat sebuah garis yang berada didalam pergeseran kiri atau kanan arah jarum jam dapat dihitung mengacu arah utara sebagai sumbu utama.

Di dalam pemrograman android sudah didukung *library* dan *method* untuk pengukuran *bearing* menurut dua lokasi yang berbeda, sehingga dapat ditarik garis lurus dan garis tersebut dapat diambil derajat pergeserannya mengacu arah utara kompas. Method **bearingTo** adalah method yang digunakan untuk kalkulasi

dua lokasi yang berbeda. Kode dapat dilihat pada kode program berikut ini.

```
loc2.setLatitude(21.422505); //Latitude Ka'bah
loc2.setLongitude(39.826169); //Longitude Ka'bah
azimuth = azimuth + loc1.bearingTo(loc2); //azimuth + bearing
```

Gambar 2 : Skrip program untuk Kalkulasi *bearing* posisi lokasi device ke lokasi ka'bah

### Kalkulasi *bearing to azimuth*

Dari hasil bearing sebelumnya, maka dapat dilakukan konversi bearing menuju azimuth, dikarenakan pada perangkat mobile android itu sendiri mengakomodasi azimuth untuk menentukan arah perputaran *compass*. Bearing to azimuth ini mengkalkulasi hasil antara arah true north menuju bearing yang sudah didapat sebelumnya, dapat dilihat pada kode program berikut ini.

```
private void adjustArrow() {
    if (arrowView == null) {
        Log.i(TAG, "arrow view is not set");
        return; }
    Animation an = new RotateAnimation(-currentAzimuth, -azimuth,
    Animation.RELATIVE_TO_SELF, 0.5f, Animation.RELATIVE_TO_SELF, 0.5f);
    currentAzimuth = azimuth;
    an.setDuration(500);
    an.setRepeatCount(0);
    an.setFillAfter(true);
    arrowView.startAnimation(an); }
```

Gambar 4 : Skrip program untuk menghitung Redirect *arrow* mata panah ke arah ka'bah

### KONVERSI KALENDER MASEHI KE KALENDER HIJRIYAH

Konversi kalender masehi kedalam hijriyah ini menggunakan algoritma kuwaiti, hasil penelitian dari top developer microsoft. Algoritma kuwaiti dapat melakukan perhitungan konversi kalender berdasarkan selisih astronomi (pendekatan menurut astronomi) dan perhitungan konversi kalender berdasarkan selisih civil (pendekatan menurut keputusan pemerintah). Hasil dari perhitungan astronomi dan civil tersebut menghasilkan

```
float orientation[] = new float[3];
SensorManager.getOrientation(R, orientation);
azimuth = (float) Math.toDegrees(orientation[0]); //ambil azimuth
loc2.setLatitude(21.422505); //Latitude Ka'bah
loc2.setLongitude(39.826169); //Longitude Ka'bah
azimuth = azimuth + loc1.bearingTo(loc2); //azimuth + bearing
```

Gambar 3 : Skrip program untuk Kalkulasi *bearing to azimuth*

### Redirect *arrow* mata panah ke arah ka'bah

Hasil dari true north tersebut akan dikalkulasikan dengan hasil azimuth sehingga menimbulkan pergeseran *true north* menjadi *true qibla* (arah kiblat yang benar) yang kemudian hasil dari *true qibla* tersebut digunakan untuk menghasilkan nilai rotation matrix, nilai rotation matrix ini digunakan untuk memutar mata panah menuju lokasi kiblat. Modul AdjustArrow(); digunakan untuk memutar arah anak panah menuju lokasi ka'bah. Berikut ini isi kode program modul AdjustArrow();:

selisih satu hari. *Class* yang digunakan untuk metode perhitungan dapat dilihat pada kode program berikut ini.

```
package com.examples.android.calendar;
import java.text.DateFormat;
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.Calendar;
public class DateHigri {
    static double z;
    static double gmod(double n,double m)
    {return ((n % m) + m) % m; }
```

```

static double[] kuwaiticalendar(boolean
adjust, String tanggal, int metodetanggal) {
    Calendar today;
    if(tanggal == "kosong"){ today = Calendar.
getInstance(); }
    else{   DateFormat df = new
SimpleDateFormat("yyyy-MM-dd");
        today = Calendar.getInstance();
        try { today.setTime(df.
parse(tanggal));}
        catch (ParseException e) {
            e.printStackTrace();}
    int adj=0;
    if(adjust){adj=0; }
    else{adj=1;}
    if (adjust) {int adjustmili = 1000 * 60 * 60
* 24 * adj;
        long todaymili = today.
getTimeInMillis() + adjustmili;
        today.setTimeInMillis(todaymili);}
    double day = today.get(Calendar.DAY_OF_
MONTH);
    double month = today.get(Calendar.
MONTH);
    double year = today.get(Calendar.YEAR);
    double m = month + 1;
    double y = year;
    if (m < 3) { y -= 1;
        m += 12; }
    double a = Math.floor(y / 100.);
    double b = 2 - a + Math.floor(a / 4.);
    if (y < 1583)
        b = 0;
    if (y == 1582) { if (m > 10) b = -10;
        if (m == 10) { b = 0;
            if (day > 4) b = -10;} }
    double jd=Math.floor(365.25 *
(y+4716))+Math.floor(30.6001*(m+1))+da
y+b-1524;
    b = 0;
    if (jd > 2299160) { a = Math.floor((jd -
1867216.25) / 36524.25);
        b = 1 + a - Math.floor(a / 4.);}
    double bb = jd + b + 1524;
    double cc = Math.floor((bb - 122.1) /
365.25);
    double dd = Math.floor(365.25 * cc);
    double ee = Math.floor((bb - dd) /
30.6001);
    day = (bb - dd) - Math.floor(30.6001 * ee);
        month = ee - 1;
    if (ee > 13) { cc += 1;
        month = ee - 13; }
    year = cc - 4716;

```

```

double wd = gmod(jd + 1, 7) + 1;
double iyear = 10631. / 30.;
double epochastro = 1948084;
double epochcivil = 1948085;
double shift1 = 8.01 / 60.;
if (metodetanggal == 0){ z = jd - epochcivil;
}
else{ z = jd - epochastro; }
double cyc = Math.floor(z / 10631.);
z = z - 10631 * cyc;
double j = Math.floor((z - shift1) / iyear);
double iy = 30 * cyc + j;
z = z - Math.floor(j * iyear + shift1);
double im = Math.floor((z + 28.5001) /
29.5);
if (im == 13)
    im = 12;
double id = z - Math.floor(29.5001 * im -
29);
double[] myRes = new double[8];
myRes[0] = day; // calculated day (CE)
myRes[1] = month - 1; // calculated month
(CE)
myRes[2] = year; // calculated year (CE)
myRes[3] = jd - 1; // julian day number
myRes[4] = wd - 1; // weekday number
myRes[5] = id; // islamic date
myRes[6] = im - 1; // isDateHigrilamic
month
myRes[7] = iy; // islamic year
return myRes; }
public static String writeIslamicDate(String
tanggal, int metodetanggal) {
    String[] wdNames = {"Minggu", "Senin",
"Selasa", "Rabu", "Kamis",
"Jum'at", "Sabtu"};
    String[] iMonthNames = {"Muharram",
"Safar", "Rabi'ul Awwal",
"Rabi'ul Akhir", "Jumadal Ula", "Jumadal
Akhira", "Rajab",
"Sha'ban", "Ramadan", "Shawwal", "Dhul
Qa'ada", "Dhul Hijja"};
    // This Value is used to give the correct
day +- 1 day

    boolean dayTest=true;
    double[] iDate = kuwaiticalendar(dayTest,
tanggal, metodetanggal);
    String outputIslamicDate = wdNames[(int)
iDate[4]] + ", " + (int) iDate[5]
+ " "+ iMonthNames[(int) iDate[6]] + " "
+ (int) iDate[7] + " Hijriyah";
    return outputIslamicDate;
}}

```

**MENENTUKAN WAKTU SHOLAT**

Penentuan waktu sholat menggunakan library class bahasa pemrograman java diambil dari praytimes.org. Library class tersebut dikembangkan oleh Hussain Ali Khan seorang developer di praytimes.org. Source code tersebut bebas digunakan dengan modifikasi

maupun tidak, dengan syarat didalam source tersebut masih ada keterangan diambil dari website praytimes.org. Hasil dari library class tersebut dapat melakukan kalkulasi penentuan waktu sholat dengan beberapa cara perhitungan, diantaranya ditunjukkan dalam tabel berikut ini.

Tabel 1 : Metode Perhitungan

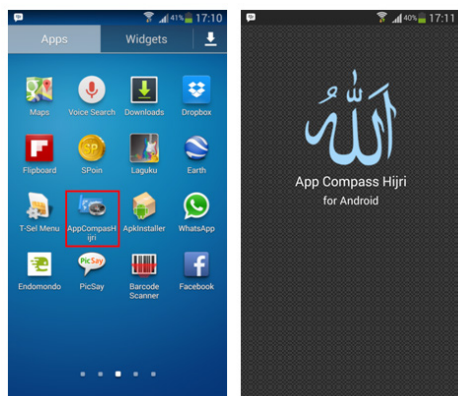
No.	Cara Perhitungan Menurut	Digunakan dinegara
1.	Muslim World League	Europe, Far East, parts of US
2.	Islamic Society of North America	North America (US and Canada)
3.	Egyptian General Authority os Survey	Africa, Syria, Lebanon, Malaysia
4.	Umm al-Qura University, Makkah	Arabian Peninsula
5.	University of islamic Sciences, Karachi	Pakistan, Afganistan, Bangladesh, India
6.	Institute of Geophysics, University of Tehran	Iran, Some Shia communities
7.	Shia Ithna Ashari, Leva Research Insitute, Qum	Some Shia communities worldwide

Sumber : [http://praytimes.org/wiki/Calculation\\_Methods](http://praytimes.org/wiki/Calculation_Methods)

Library class ini juga mengakomodasi perhitungan waktu sholat kedalam madzhab syafi'i ataupun hanafi. Perhitungan waktu sholat tersebut memerlukan titik lokasi GPS berupa bujur dan lintang. Titik lokasi ini digunakan untuk penyesuaian waktu sholat dimana AppCompassHijri dijalankan. Penentuan waktu sholat ini mengacu pada class yang sudah disediakan oleh praytimes.org.

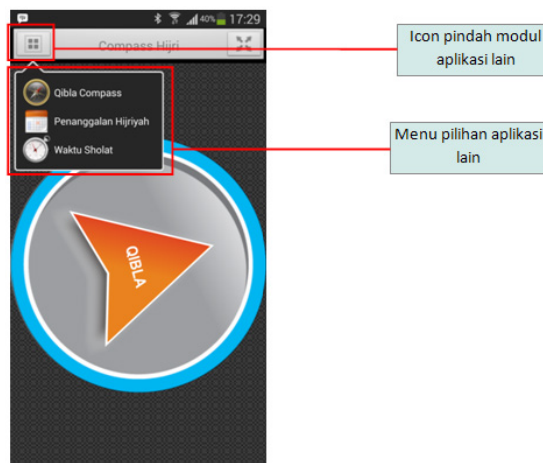
**CARAMENGGUNAKAN APPCOMPASSHIJRI**

Setelah diinstall dalam perangkat smart-phone, maka akan muncul icon AppCompas Hijri seperti dalam gambar berikut ini :



Gambar 5 : Aplikasi AppCompasHijri sesudah diinstall dan tampilan awal ketika dijalankan

Ketika aplikasi dibuka, secara default, akan langsung menampilkan kompas kiblat. Jika hendak membuka modul aplikasi yang lain seperti kalender hijriyyah dan waktu sholat, dapat menyentuh bagian kanan atas, sebagaimana yang ditunjukkan dalam gambar berikut ini :



Gambar 6 : Pemilihan menu aplikasi

Setelah memilih aplikasi yang hendak dijalankan, maka pemakai dapat mempergunakan aplikasi ini. Modul qibla compass ini memerlukan titik koordinat GPS sebagai penentu awal dimana akan dilakukan pengukuran

bearing, maka modul GPS dalam *smartphone* harus di aktifkan terlebih dahulu. Penggunaan kalender hijriyah tidak memerlukan modul khusus. Sensor gravitasi ataupun GPS tidak berpengaruh didalam penanggalan hijriyah ini, kalender hijriyah ini hanya melakukan konversi kalender masehi yang ada didalam *handphone* kedalam penanggalan hijriyah, yang terpenting adalah kalender masehi didalam *handphone* masih tetap berjalan dan tersetting secara benar. Sehingga menghasilkan konversi kalender masehi ke dalam hijriyah secara tepat. Tampilan kalender hijriyah dapat dilihat pada Gambar 7.

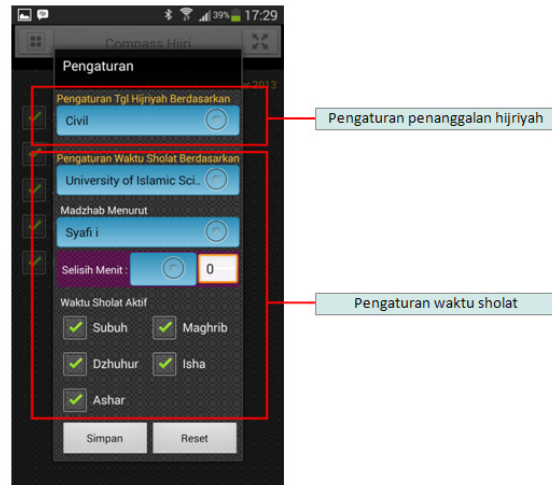


Gambar 7 : Tampilan kalender hijriyah

Modul konversi kalender hijriyah dan penentuan waktu sholat menyediakan fasilitas kalibrasi yang dibutuhkan untuk penyesuaian secara manual. Fasilitas ini diperlukan jika terjadi perhitungan berbeda dan memiliki selisih menit tertentu. Fasilitas kalibrasi tersedia pada menu pengaturan.

Di dalam menu pengaturan ini tersedia pengaturan – pengaturan yang dapat mengatur konfigurasi dari beberapa sub aplikasi

AppCompassHijri, sub aplikasi yang dapat di atur didalam pengaturan tersebut diantaranya sub aplikasi penanggalan hijriyah dan waktu sholat.



Gambar 8 : Menu Pengaturan Aplikasi

## KESIMPULAN

Aplikasi islami berbasis android, **AppCompassHijri** ini sangat bermanfaat untuk menunjang ibadah. Perangkat GPS diperlukan guna mendukung akurasi arah dan waktu sholat 5 waktu. Hasil pengujian yang dilakukan, aplikasi ini relatif mendekati dari hasil sebenarnya, bahkan hasil pengujian penanggalan hijriyah dapat dikatakan akurat dan tepat. Sehingga, AppCompassHijri dapat digunakan untuk membantu pencarian arah kiblat, penentuan waktu sholat dan konversi kalender hijriyah ke masehi.

Untuk mendapatkan hasil yang akurat dan maksimal, penggunaan AppCompassHijri ini harus memenuhi beberapa kriteria penggunaan, diantaranya :

1. Digunakan di lokasi yang mudah dijangkau sinyal GPS,
2. Digunakan di lokasi yang mudah dijangkau sinyal operator selular/*internet service provider*;
3. Penggunaan *qibla compass* harus dalam keadaan layar menghadap keatas dan seimbang dengan gravitasi bumi.

## DAFTAR PUSTAKA

- Ananda, diakses : 8 November 2013 00:13 WIB, *Cara Kerja GPS*, <http://dhitovans.blogspot.com/2011/04/cara-kerja-gps.html>.
- Anonymous, diakses : 8 Oktober 2013 11:45 WIB, *The Kuwaiti Algorithm*, [http://www.staff.science.uu.nl/~gent0113/islam/islam\\_tabcal\\_others.htm](http://www.staff.science.uu.nl/~gent0113/islam/islam_tabcal_others.htm).
- Anonymous, diakses : 7 November 2013 23:55 WIB, *Sensor Manager*, <http://developer.android.com/reference/android/hardware/SensorManager.html>
- Anonymou, 2012, diakses : 8 November 2013 00:14 WIB, . 2012. "Bagaimana GPS itu Bekerja". <http://garmin.co.id/2012/04/bagaiman-gps-itu-bekerja/>
- Cahyadsn, diakses : 7 November 2013 23:56 WIB, <http://www.pkpu.or.id/adzan3/monthly.php?id=308>
- DiMarzio, J.F., 2008, *Android : A Programmer's Guide*, Mc Graw Hill
- Hussain Ali Khan, diakses : 7 November 2013 23:55 WIB, *Code Java Translation*, <http://praytimes.org/code/git/?a=tree&p=PrayTimes&hb=HEAD&f=v1/java>.
- Majlis Tarjih dan Tajdid PP Muhammadiyah, 2009, *Pedoman Hisab Muhammadiyah*, Pimpinan Pusat Muhammadiyah
- Lee, Wei-Meng, 2011, *Beginning Android Application Development*, Willey Publishing
- \_\_\_\_\_, diakses tanggal 8/04/2013, *10% of web browsers are using smartphones or tablets*, [http://www.phonearena.com/news/10-of-web-browsers-are-using-smartphones-or-tablets\\_id36246#1-](http://www.phonearena.com/news/10-of-web-browsers-are-using-smartphones-or-tablets_id36246#1-)
- \_\_\_\_\_, diakses tanggal 8/04/2013, *Komputer Tablet*, [http://id.wikipedia.org/wiki/Komputer\\_tablet](http://id.wikipedia.org/wiki/Komputer_tablet)
- \_\_\_\_\_, diakses tanggal 8/04/2013, *Smartphone*, <http://id.wikipedia.org/wiki/Smartphone>
- \_\_\_\_\_, diakses : 17/11/2013 jam 14:44, *Eclipse (perangkat lunak)*, [http://id.wikipedia.org/wiki/Eclipse\\_\(perangkat\\_lunak\)](http://id.wikipedia.org/wiki/Eclipse_(perangkat_lunak))
- \_\_\_\_\_, diakses : 17/11/2013 jam 11:48, *Kalender Hijriyah*, [http://id.wikipedia.org/wiki/Kalender\\_Hijriyah](http://id.wikipedia.org/wiki/Kalender_Hijriyah)
- \_\_\_\_\_, diakses : 17/11/2013 jam 15:19, *Android (sistem operasi)*, [http://id.wikipedia.org/w/index.php?title=Android\\_sistem\\_operasi](http://id.wikipedia.org/w/index.php?title=Android_sistem_operasi)
- \_\_\_\_\_, diakses : 17/11/2013 jam 19:35, *Telepon Cerdas*, [http://id.wikipedia.org/w/index.php?title=Telepon\\_cerdas](http://id.wikipedia.org/w/index.php?title=Telepon_cerdas)