

**APLIKASI PENENTUAN WAKTU SHOLAT BERDASAR
LOKASI GEOGRAFIS PADA IPHONE**

LAPORAN TUGAS AKHIR

*Diajukan Sebagai Salah Satu Syarat
untuk Memperoleh Gelar Sarjana Teknik Informatika*



Disusun Oleh:

Nama : Muhammad Arief Widyananda

NIM : 06 523 299

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
YOGYAKARTA**

2010

HALAMAN JUDUL

**APLIKASI PENENTUAN WAKTU SHOLAT BERDASAR
LOKASI GEOGRAFIS PADA IPHONE**

LAPORAN TUGAS AKHIR

*Diajukan Sebagai Salah Satu Syarat
untuk Memperoleh Gelar Sarjana Teknik Informatika*



Disusun Oleh:

Nama : Muhammad Arief Widyananda

NIM : 06 523 299

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
YOGYAKARTA**

2010

LEMBAR PENGESAHAN PEMBIMBING

**APLIKASI PENENTUAN WAKTU SHOLAT BERDASAR
LOKASI GEOGRAFIS PADA iPHONE**

LAPORAN TUGAS AKHIR



Disusun oleh

Nama : Muhammad Arief Widyananda

NIM : 06 523 299

Yogyakarta, 8 Agustus 2010

Telah Diterima dan Disetujui dengan Baik Oleh:

Dosen pembimbing

Fathul Wahid, ST., M.Sc.

xv/43

LEMBAR PENGESAHAN PENGUJI

**APLIKASI PENENTUAN WAKTU SHOLAT BERDASAR
LOKASI GEOGRAFIS PADA iPHONE**

TUGAS AKHIR

Disusun oleh:

Nama : Muhammad Arief Widyananda

NIM : 06 523 299

Telah Dipertahankan di Depan Sidang Penguji Sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana Teknik Informatika
Fakultas Teknologi Industri Universitas Islam Indonesia.

Yogyakarta, 24 Agustus 2010

Tim Penguji

Fathul Wahid, S.T., M.Sc.

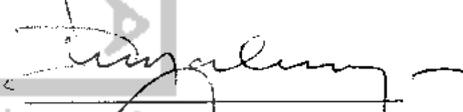
Ketua

R. Teduh Dirgahayu, S.T., M.Sc.

Anggota I

Zainudin Zuhri, S.T., M.IT.

Anggota II



Mengetahui,

Ketua Jurusan Teknik Informatika

Universitas Islam Indonesia



Yusuf Prayudi, S.Si., M.Kom.

**LEMBAR PERNYATAAN KEASLIAN
HASIL TUGAS AKHIR**

Saya yang bertanda tangan dibawah ini,

Nama : Muhammad Arief Widyananda

NIM : 06 523 299

Menyatakan bahwa seluruh komponen dan isi dalam laporan Tugas Akhir ini adalah hasil karya sendiri. Apabila dikemudian hari terbukti bahwa ada beberapa bagian dari karya ini adalah bukan hasil karya saya sendiri, maka saya siap menanggung resiko dan konsekuensi apapun.

Demikian pernyataan ini saya buat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 18 Agustus 2010



Muhammad Arief Widyananda

Halaman Persembahan

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

*Kupersembahkan Tugas Akhir ini
dengan setulus hatiku untuk*

*Ibu dan Bapakku tercinta,
Eni Zuhriyah dan Rochmadi*

*Kakak-kakakku,
Aditya Yulla SS, Nur Rahmi Ananda, dan Nur Aini Ananda*

*dan Seseorang yang selalu mendukungku,
Dinda Eling Kartikaning Sasmito*

*yang selalu memberikan Do'a, Semangat, Pengorbanan,
dan Dukungan selama Aku menyelesaikan studi*

Halaman Motto

*"Sesungguhnya sesudah kesulitan itu ada kemudahan"
(Q.S. Al-Inshirah: 6)*



*"Jangan tunda sampai besok,
apa yang bisa dikerjakan hari ini"*

*"Kamu, tidak lebih dari apa yang kamu bayangkan,
jadi bayangkanlah yang lebih baik untuk dirimu sendiri"*

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Assalamu'alaikum Wr.Wb

Alhamdulillah, segala puji bagi Allah SWT atas segala rahmat, hidayah dan inayah-Nya, sehingga penulisan laporan Tugas Akhir yang berjudul "**Aplikasi Penentuan Waktu Sholat Berdasar Lokasi Geografis pada iPhone**" dapat penulis selesaikan dengan baik. Shalawat serta salam semoga senantiasa tercurah atas Nabi Muhammad SAW, para kerabat, serta pengikutnya hingga hari kiamat nanti.

Laporan Tugas Akhir ini disusun sebagai salah satu syarat guna memperoleh gelar sarjana Jurusan Teknik Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia. Laporan Tugas Akhir juga sebagai sarana untuk mempraktekkan secara langsung ilmu dan teori yang telah diperoleh selama menjalani masa studi di Jurusan Teknik Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia.

Penyusunan laporan Tugas Akhir ini tidak lepas dari bimbingan, dukungan dan bantuan dari berbagai pihak. Oleh karena itu dalam kesempatan ini dengan segala kerendahan hati, penulis ingin menyampaikan ucapan terima kasih yang setulus-tulusnya kepada:

1. Allah SWT, Tuhan bagi seluruh alam yang melimpahkan rahmat dan karuniannya sehingga penulis selalu diberi kesehatan dan kemudahan selama pembuatan tugas akhir ini.
2. Ibu dan Bapak, Eni Zuhriyah dan Rochmadi, yang selalu dan terus memberikan do'a, semangat, dan dukungan.

3. Bapak Gumbolo Hadi Susanto, M.Sc., selaku Dekan Fakultas Teknologi Industri Universitas Islam Indonesia dan seluruh jajaran Dekanat Fakultas Teknologi Industri.
4. Bapak Yudi Prayudi, S.Si, M.Kom, selaku Ketua Jurusan Teknik Informatika. Terima kasih atas kemudahan dan dukungan yang telah diberikan.
5. Bapak Fathul Wahid, ST, M.Sc., selaku dosen pembimbing yang telah memberikan pengarahan, bimbingan, masukan, serta dorongan semangat selama pelaksanaan tugas akhir dan penulisan laporan.
6. Dosen-dosen Jurusan Teknik Informatika dan seluruh jajaran staff jurusan. Terima kasih atas semua ilmu pengetahuan, saran, motivasi serta bantuannya.
7. Bapak Drs. Sofwan Jannah, M.Ag., dosen Program Studi Syari'ah Fakultas Ilmu Agama Islam, terima kasih atas semua materi ilmu hisab yang telah diajarkan.
8. Bapak DR. Rinto Anugraha, dosen Teknik Fisika Universitas Gadjah Mada, terima kasih atas semua detail perhitungan ilmu hisab yang telah diberikan.
9. Kakak-kakakku dan keponakanku, Mas Adit, Mbak Nanda, Mbak Nunik dan ririn yang selalu memberikan semangat dan motivasi untuk selalu maju.
10. Seseorang yang selalu mendukungku, Dinda Eling KS, terima kasih atas bantuan, dukungan dan motivasi yang diberikan.
11. Teman-teman kontrakan, Riki "joe", Mas Aris, Arif "Jayonk", Ibnu "Ndes" terima kasih atas kebersamaan dan dukungan kalian.
12. Teman-teman "cah nakal", Adit, Miftah, Opek, Indra, Fikri, Nofal, Warsun, terima kasih atas tawa dan semangat kalian.
13. Teman-teman seperjuangan di Laboratorium Komputasi dan Sistem Cerdas, Liza "zha zha", Kisti "kizzy", Hendra "tukul", Henry "rierie", Robby "wrekudara", Mas Bowo, Mas Teddy, Mas Cookie, dan semua asisten KSC, terima kasih atas kebersamaan dan suasana kekeluargaan yang tercipta selama ini.
14. Teman-teman seperjuangan skripsi, Ari Yuda, dan Rohmad "Hida", terima kasih dukungan dan semangatnya.

15. Teman-teman Informatika 2006-2009, terimakasih atas kekompakan dan kebersamannya selama ini.
16. Semua pihak yang tidak dapat disebutkan satu persatu yang telah membantu sehingga laporan tugas akhir ini dapat terselesaikan.

Terima kasih kepada semua pihak yang telah membantu terselesainya penulisan laporan Tugas Akhir ini. Semoga Allah SWT melimpahkan rahmat dan hidayah-Nya serta membalas semua kebaikan yang telah diberikan.

Penulis menyadari bahwa dalam penyusunan laporan Tugas Akhir ini masih banyak terdapat kekeliruan dan kekurangan. Untuk itu penulis menyampaikan permohonan maaf sebelumnya serta sangat diharapkan kritik dan saran yang sifatnya membangun untuk penyempurnaan di masa mendatang.

Akhir kata semoga laporan ini dapat bermanfaat bagi kita semua. Amin.

Wassalamu 'alaikum Wr.Wb.

Yogyakarta, 15 Agustus 2010

Muhammad Arief Widyananda

الجامعة الإسلامية
الائتلاف

SARI

Ketepatan waktu sholat merupakan hal yang penting bagi umat Islam. Untuk itu dibutuhkan sebuah sistem penentu waktu sholat yang mudah digunakan. Sistem informasi penentuan waktu sholat yang sudah ada memiliki kelemahan dimana *user* harus melakukan pengaturan ulang terhadap lokasi geografis dan waktu setempat dimana *user* berada sekarang untuk mendapatkan waktu sholat. Dengan bantuan GPS (*Global Positioning System*) dapat diketahui lokasi dan waktu sehingga tidak dibutuhkan pengaturan oleh *user*. *iPhone* merupakan perangkat yang sudah dilengkapi GPS. Dengan demikian dapat dibuat sebuah aplikasi untuk menentukan waktu sholat jika diketahui letak geografis suatu lokasi pada perangkat *iPhone*. Pengembangan aplikasi yang dibuat dilakukan dengan mempelajari *library* Cocoa Touch yang telah disediakan oleh Apple. Pengujian dilakukan secara simulasi menggunakan *iPhone Simulator*. Aplikasi yang dibuat pada Tugas Akhir ini berupa aplikasi penentuan waktu sholat yang diimplementasikan pada *iPhone*. Aplikasi ini membantu *user* memperoleh informasi jadwal sholat dengan cepat dan mudah dan perhitungan yang cukup akurat.

Kata kunci :
Jadwal Sholat, *iPhone*, GPS.



TAKARIR

<i>Accelerometer</i>	Sensor percepatan
<i>Altitude</i>	Ketinggian tempat
<i>Application Programming Interface</i>	Kumpulan fungsi yang disediakan oleh suatu bahasa pemrograman
<i>Entity</i>	Obyek
<i>Framework</i>	Kerangka pemrograman
<i>Hour Angle</i>	Sistem koordinat ekuator
<i>Integrated Development Environment</i>	Aplikasi untuk membangun sebuah perangkat lunak
<i>Internet protocols</i>	Aturan yang digunakan oleh internet
<i>Library</i>	Kumpulan fungsi
<i>Method getter</i>	Fungsi untuk mendapatkan nilai variabel
<i>Method setter</i>	Fungsi untuk mengatur nilai variabel
<i>Multithreading</i>	Menjalankan banyak proses dalam satu waktu
<i>Sockets</i>	Salah satu jenis konektor
<i>User</i>	Pengguna



DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN PEMBIMBING	ii
LEMBAR PENGESAHAN PENGUJI	iii
LEMBAR PERNYATAAN KEASLIAN HASIL TUGAS AKHIR	iv
HALAMAN PERSEMBAHAN	v
HALAMAN MOTTO	vi
KATA PENGANTAR	vii
SARI	x
TAKARIR	xi
DAFTAR ISI	xii
DAFTAR TABEL	xiv
DAFTAR GAMBAR	xv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	1
1.3 Batasan Masalah	2
1.4 Manfaat Penelitian	2
1.5 Metodologi Penelitian	2
1.6 Sistematika Penulisan.....	2
BAB II LANDASAN TEORI	4
2.1 Penentuan Waktu Sholat.....	4
2.1.1 Penyediaan Data.....	4
2.1.2 Penyediaan Rumus	7
2.2 Penentuan Arah Kiblat	9
2.3 Perangkat <i>iPhone</i>	10
2.4 Bahasa Pemrograman Objective-C	10
2.5 Cocoa Touch.....	11
2.6 GPS.....	15
BAB III METODOLOGI	17
3.1 Analisis Kebutuhan Perangkat Lunak	17
3.1.1 Metode Analisis.....	17
3.1.2 Hasil Analisis	17

3.2	Perancangan Perangkat Lunak	19
3.2.1	Metode Perancangan	19
3.2.2	Hasil Perancangan	19
3.2.3	Perancangan Antarmuka.....	23
BAB IV HASIL DAN PEMBAHASAN		28
4.1	Implementasi Perangkat Lunak.....	28
4.1.1	Implementasi Proses.....	28
4.2	Analisis Kinerja Perangkat Lunak	36
4.2.1	Pengujian Normal.....	36
4.2.2	Pengujian Tidak Normal.....	41
BAB V KESIMPULAN DAN SARAN.....		42
5.1	Kesimpulan.....	42
5.2	Saran.....	42
DAFTAR PUSTAKA.....		43



DAFTAR TABEL

Tabel 4.1 Tabel Perbandingan Hasil Perhitungan Waktu Sholat Pada Tanggal 7 Agustus 2010.....	40
Tabel 4.2 Tabel Perbandingan Hasil Perhitungan Arah Kiblat.....	40



DAFTAR GAMBAR

Gambar 2.1 Posisi Matahari saat Waktu Sholat	7
Gambar 2.2 Hierarki Kelas pada Foundation Framework	13
Gambar 2.3 Hierarki Kelas pada <i>UIKit Framework</i>	14
Gambar 3.1 DFD Level 0	20
Gambar 3.2 DFD Level 1	21
Gambar 3.3 DFD Level 2 Proses perhitungan waktu sholat	22
Gambar 3.4 DFD Level 2 Proses ganti pengaturan	23
Gambar 3.5 Halaman lihat jadwal sholat	24
Gambar 3.6 Halaman lihat arah kiblat	24
Gambar 3.7 Halaman ganti pengaturan	25
Gambar 3.8 Halaman ganti pengaturan mazhab	25
Gambar 3.9 Halaman ganti pengaturan sudut Subuh	26
Gambar 3.10 Halaman ganti pengaturan sudut Isya'	26
Gambar 3.11 Halaman ganti pengaturan kota	27
Gambar 3.12 Halaman ganti tanggal	27
Gambar 4.1 Halaman lihat jadwal sholat	33
Gambar 4.2 Halaman lihat arah kiblat	35
Gambar 4.3 Halaman ganti pengaturan	36
Gambar 4.4 Pengaturan mazhab	37
Gambar 4.5 Pengaturan sudut subuh	37
Gambar 4.6 Pengaturan sudut isya	38
Gambar 4.7 Pengaturan kota	38
Gambar 4.8 Pengaturan tanggal	39
Gambar 4.9 Hasil perhitungan jadwal sholat	39
Gambar 4.10 Penanganan kesalahan masukan data kosong	41

BAB I

PENDAHULUAN

1.1 Latar Belakang

Ketepatan waktu sholat merupakan hal yang penting bagi umat Islam. Untuk itu dibutuhkan sebuah penentuan waktu sholat. Salah satu cara penentuan waktu sholat adalah dengan menggunakan perhitungan atau yang dikenal dengan ilmu hisab. Perhitungan waktu sholat sebenarnya dapat dilakukan jika mengetahui letak geografis suatu lokasi. Perhitungan ini sudah banyak dibuat dalam bentuk sistem informasi.

Sistem informasi penentuan waktu sholat sekarang ini sudah cukup banyak, namun memiliki kelemahan dimana jika *user* akan berpindah dari satu lokasi ke lokasi yang lain, *user* harus melakukan pengaturan ulang terhadap lokasi geografis dan waktu setempat dimana *user* berada sekarang. Hal ini akan merepotkan *user* pada saat *user* berpindah lokasi.

Global Positioning System (GPS), sebagai alat yang dapat menentukan lokasi dengan tepat, dapat digunakan sebagai alternatif agar *user* tidak lagi direpotkan oleh pengaturan ulang lokasi dan waktu setempat dimana *user* berada sekarang.

iPhone sebagai perangkat yang saat ini sudah banyak beredar di pasaran, sudah disertai dengan perangkat GPS. Dengan alasan tersebut, maka aplikasi ini cocok diimplementasikan pada perangkat *iPhone*.

1.2 Rumusan Masalah

Bagaimana membangun sebuah aplikasi untuk menentukan waktu sholat jika diketahui letak geografis suatu lokasi pada perangkat *iPhone*.

1.3 Batasan Masalah

Dalam melaksanakan suatu penelitian, diperlukan adanya batasan agar tidak menyimpang dari yang telah direncanakan sehingga tujuan yang sebenarnya dapat tercapai. Adapun batasan masalah dalam tugas akhir ini antara lain:

1. Perancangan dan simulasi dilakukan dengan menggunakan *iPhone Simulator*.
2. Rumus perhitungan waktu sholat yang digunakan mengacu kepada Pedoman Hisab Muhammadiyah.

1.4 Manfaat Penelitian

Hasil dari penelitian ini diharapkan dapat memberikan kemudahan kepada *user* untuk menentukan waktu sholat secara tepat dan cepat dengan memanfaatkan teknologi GPS.

1.5 Metodologi Penelitian

Penulis memperoleh data melalui studi kepustakaan, menggali informasi dari berbagai sumber yang layak untuk digunakan sebagai landasan dengan mempelajari dan mendalami materi-materi yang bersifat konseptual serta materi-materi yang bersifat terapan.

Pengembangan aplikasi ini dilakukan dengan mempelajari *library* Cocoa Touch yang telah disediakan oleh Apple.

Pengujian dilakukan secara simulasi menggunakan *iPhone Simulator*. Aplikasi yang akan dibangun menggunakan fasilitas GPS yang ada pada perangkat *iPhone* untuk menentukan letak geografis *user*.

1.6 Sistematika Penulisan

Sistematika penulisan dan garis besar isi laporan ini adalah sebagai berikut:

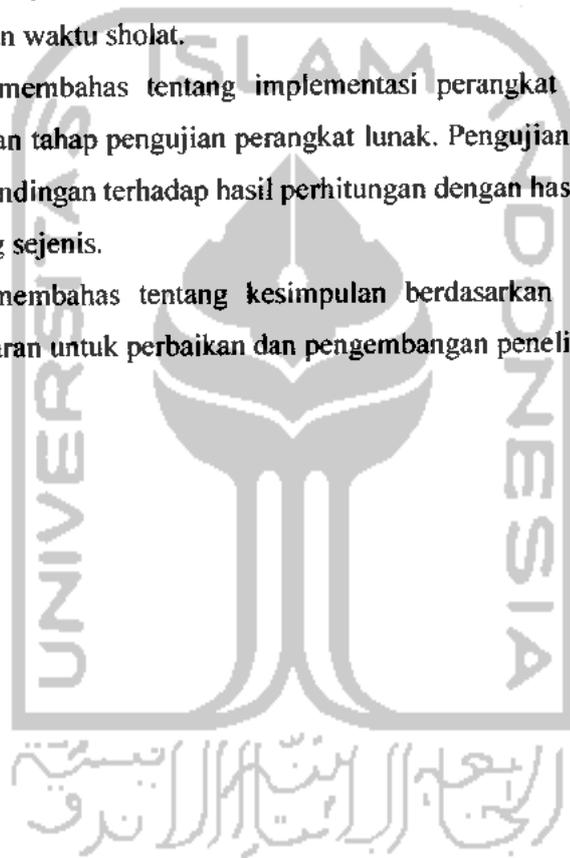
Bab I membahas tentang latar belakang kenapa dibutuhkan aplikasi penentuan waktu sholat ini, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian, dan sistematika penulisan.

Bab II membahas tentang teori-teori yang berhubungan dengan penelitian meliputi: Penentuan waktu shalat, *iPhone*, Objective-C, Cocoa Touch, dan pengertian GPS.

Bab III membahas tentang analisis kebutuhan perangkat lunak yang meliputi metode analisis yang akan dipakai, dan analisis sistem. Kemudian dilanjutkan dengan perancangan perangkat lunak yang meliputi pemodelan desain aplikasi penentuan waktu shalat.

Bab IV membahas tentang implementasi perangkat lunak. Kemudian dilanjutkan dengan tahap pengujian perangkat lunak. Pengujian dilakukan dengan melakukan perbandingan terhadap hasil perhitungan dengan hasil perhitungan dari aplikasi lain yang sejenis.

Bab V membahas tentang kesimpulan berdasarkan pembahasan hasil penelitian, dan saran untuk perbaikan dan pengembangan penelitian berikutnya.



BAB II LANDASAN TEORI

2.1 Penentuan Waktu Sholat

Perhitungan waktu sholat dapat diketahui melalui dua langkah, yaitu penyediaan data, dan penyediaan rumus-rumus yang digunakan. Setelah data dan rumus tersedia, proses selanjutnya adalah melakukan perhitungan dengan data tersebut (Majelis Tarjih dan Tajdid PP Muhammadiyah, 2009:54).

2.1.1 Penyediaan Data

Data-data yang digunakan untuk menghitung waktu sholat adalah sebagai berikut:

a. Koordinat Lintang Tempat (L)

Lintang tempat adalah jarak sepanjang meridian bumi diukur dari ekuator bumi (katulistiwa) sampai suatu tempat yang bersangkutan. Nilai dari lintang tempat adalah 0° sampai 90° . Lintang tempat bagi tempat-tempat yang berada di belahan bumi utara bertanda positif (+) dan tempat-tempat yang berada di belahan bumi selatan bertanda negatif (-). Dalam astronomi disebut *latitude* dan biasanya digunakan lambang ϕ (*phi*).

b. Koordinat Bujur Tempat (B)

Bujur tempat adalah jarak sepanjang equator bumi dihitung dari meridian yang melewati kota Greenwich sampai meridian yang melewati tempat bersangkutan. Dalam astronomi dikenal dengan nama *longitude* biasanya digunakan lambang λ (*lamda*). Nilai dari bujur tempat adalah 0° sampai 180° . Bagi tempat-tempat yang berada di sebelah barat Greenwich disebut Bujur Barat (BB) dan diberi tanda negatif (-). Sedangkan bagi tempat-tempat yang berada di sebelah timur Greenwich disebut Bujur Timur (BT) dan diberi tanda positif (+).

c. Zona Waktu Tempat (Z)

Zona Waktu (*Time Zone*) ini dibagi menjadi 24 zona waktu yang berbeda-beda, sesuai letak daerah tersebut. Waktu universal yang menjadi panutan adalah waktu GMT, waktu yang ada di Greenwich. Inggris.

d. Ketinggian Lokasi (H)

Ketinggian lokasi adalah ketinggian suatu lokasi dihitung dari permukaan laut. Hal ini menentukan waktu kapan terbit dan terbenamnya matahari. Tempat yang berada tinggi di atas permukaan laut akan lebih awal melihat matahari terbit serta lebih akhir melihat matahari terbenam, dibandingkan dengan tempat yang lebih rendah. Ketinggian lokasi ini biasanya menggunakan satuan meter (m).

e. Tanggal (D), Bulan (M), dan Tahun (Y)

Tanggal, bulan dan tahun akan sangat berpengaruh pada perhitungan waktu sholat. Kalender yang dianut sekarang adalah kalender gregorian, namun untuk menentukan waktu sholat digunakan kalender julian (JD). Untuk itu, dibutuhkan sebuah rumus untuk melakukan konversi dari kalender gregorian ke dalam kalender julian. Rumus tersebut dapat dituliskan sebagai berikut:

$$JD = 1720994,5 + (365,25 \times Y) + (30,6001 \times (M + 1)) + (2 + \frac{Y}{4} - \frac{Y}{100} + D + \frac{12}{24}) \dots\dots\dots(2.1)$$

Jika $M > 2$, maka nilai M dan Y tidak berubah. Jika $M = 1$ atau 2 , maka nilai M ditambah 12 sedangkan nilai Y dikurangi 1. Nilai JD di atas berlaku untuk pukul 12.00 GMT atau saat tengah hari di Greenwich. Adapun JD untuk pukul 12.00 waktu lokal, maka JD pukul 12.00 GMT waktu Greenwich tersebut harus dikurangi dengan $Z/24$ dimana Z adalah zona waktu lokal tersebut.

Setelah mendapatkan nilai dari JD tersebut, dapat dihitung sudut tanggal (T) dengan rumus sebagai berikut:

$$T = \frac{2 \times PHI \times (JD - 2451545)}{365,25} \dots\dots\dots(2.2)$$

f. Sudut Deklinasi Matahari (Delta)

Deklinasi matahari ialah jarak matahari dari lingkaran ekuator diukur sepanjang lingkaran waktu yang melalui matahari itu hingga ke titik pusat matahari tersebut.

$$\begin{aligned} \Delta = & 0,37877 + 23,264 \times \sin(57,297 \times T - 79,547) + 0,3812 \times \\ & \sin(2 \times 57,297 \times T - 82,682) + 0,17132 \times \sin(3 \times 57,297 \times T - \\ & 59,772) \dots \dots \dots (2.3) \end{aligned}$$

g. Perata Waktu (ET)

Perata waktu (*equation of time*) adalah selisih waktu antara waktu matahari hakiki dengan matahari rata-rata (pertengahan). Peredaran semua harian matahari dari arah timur ke barat itu tidaklah konstan, kadang-kadang cepat kadang-kadang lambat. Keadaan ini diakibatkan oleh percepatan bumi mengelilingi matahari tidak konstan karena bidang edarnya berbentuk elips.

Waktu matahari hakiki ialah waktu peredaran matahari senyatanya, sedangkan waktu matahari pertengahan adalah waktu peredaran semu matahari diandaikan ia beredar dengan konstan sebagaimana terlihat pada jam yang ada.

Rumus yang digunakan untuk menghitung ET adalah sebagai berikut:

$$\begin{aligned} ET = & \left(- \left(1789 + 237 \times \frac{(JD-245145)}{36525} \right) \times \sin \left(280,46607 + 36000,7689 \times \right. \right. \\ & \left. \left. \frac{(JD-245145)}{36525} \right) - \left(7146 - 62 \times \frac{(JD-245145)}{36525} \right) \times \cos \left(280,46607 + 36000,7689 \times \right. \right. \\ & \left. \left. \frac{(JD-245145)}{36525} \right) + \left(9934 - 14 \times \frac{(JD-245145)}{36525} \right) \times \sin \left(2 \times 280,46607 + \right. \right. \\ & \left. \left. 36000,7689 \times \frac{(JD-245145)}{36525} \right) - \left(29 + 5 \times \frac{(JD-245145)}{36525} \right) \times \cos \left(2 \times 280,46607 + \right. \right. \\ & \left. \left. 36000,7689 \times \frac{(JD-245145)}{36525} \right) + \left(74 + 10 \times \frac{(JD-245145)}{36525} \right) \times \sin \left(3 \times 280,46607 + \right. \right. \\ & \left. \left. 36000,7689 \times \frac{(JD-245145)}{36525} \right) + \left(320 - 4 \times \frac{(JD-245145)}{36525} \right) \times \cos \left(3 \times 280,46607 + \right. \right. \\ & \left. \left. 36000,7689 \times \frac{(JD-245145)}{36525} \right) - 212 \times \sin \left(4 \times 280,46607 + 36000,7689 \times \right. \right. \end{aligned}$$

$$\left(\frac{JD-245145}{36525}\right)/1000\dots\dots\dots(2.4)$$

h. Tetapan Panjang Bayangan Ashar (KA)

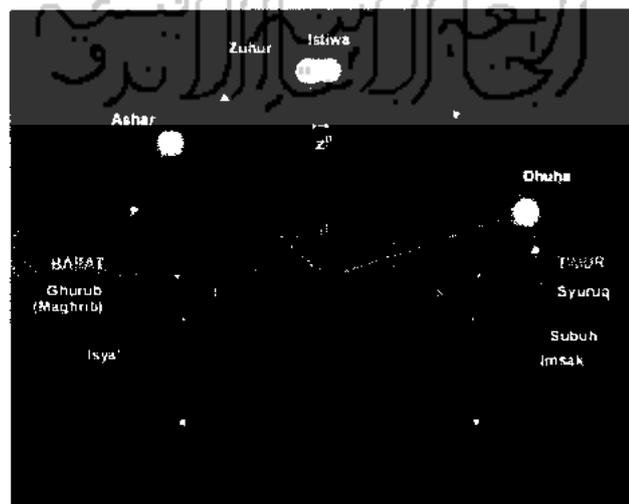
Tetapan panjang bayangan waktu sholat Ashar ini ada dua pendapat, yaitu pendapat mazhab Syafi'i dan pendapat madzhab Hanafi. Pendapat mazhab Syafi'i menyatakan panjang bayangan benda saat Ashar = tinggi benda + panjang bayangan saat Dhuhur. Sementara pendapat mazhab Hanafi menyatakan panjang bayangan benda saat Ashar = dua kali tinggi benda + panjang bayangan saat Dhuhur.

i. *Altitude* matahari waktu Subuh dan Isya'

Dalam referensi standar astronomi, sudut *altitude* untuk waktu Subuh dan Isya' adalah 18 derajat di bawah ufuk, atau sama dengan minus 18 derajat. Namun demikian ada beberapa pendapat mengenai sudut *altitude* matahari di bawah ufuk saat Subuh dan Isya. Diantaranya berkisar antara 15 hingga 20 derajat. Dengan demikian, perbedaan sudut yang digunakan akan menyebabkan perbedaan datangnya waktu Subuh dan Isya'.

2.1.2 Penyediaan Rumus

Posisi matahari pada saat waktu sholat, dapat dilihat pada Gambar 2.1.



Gambar 2.1 Posisi Matahari saat Waktu Sholat

Rumus-rumus yang digunakan untuk melakukan perhitungan waktu shalat adalah sebagai berikut:

a. Sholat Dhuhur

$$\text{Waktu sholat Dhuhur} = 12 + Z - \frac{B}{15} - \frac{ET}{60} \dots\dots\dots(2.5)$$

b. Sholat Ashar

$$\text{Waktu sholat Ashar} = \text{Dhuhur} + \frac{(\text{Hour Angle Ashar})}{15} \dots\dots\dots(2.6)$$

c. Sholat Maghrib

$$\text{Waktu sholat Maghrib} = \text{Dhuhur} + \frac{(\text{Hour Angle Maghrib})}{15} \dots\dots\dots(2.7)$$

d. Sholat Isya

$$\text{Waktu sholat Isya} = \text{Dhuhur} + \frac{(\text{Hour Angle Isya}')}{15} \dots\dots\dots(2.8)$$

e. Sholat Subuh

$$\text{Waktu sholat Subuh} = \text{Dhuhur} - \frac{(\text{Hour Angle Subuh})}{15} \dots\dots\dots(2.9)$$

Rumus-rumus diatas dibutuhkan nilai *hour angle* (HA), sedangkan rumus untuk menghitung nilai HA tersebut adalah sebagai berikut:

$$\cos(HA) = \frac{\sin(\text{altitude}) - \sin(L) \cdot \sin(\Delta)}{\cos(L) \cdot \cos(\Delta)} \dots\dots\dots(2.10)$$

$$HA = \arccos(\cos(HA)) \dots\dots\dots(2.11)$$

Nilai HA diatas dipengaruhi juga oleh nilai variabel *altitude*, rumus untuk menghitung nilai tersebut berbeda untuk setiap waktu shalat. Rumus-rumus yang digunakan adalah sebagai berikut:

a. Sholat Ashar

$$\text{altitude} = \arccot(KA + \tan(|\Delta - L|)) \dots\dots\dots(2.12)$$

Keterangan:

KA = 1, untuk Syafi'i.

KA = 2, untuk Hanafi.

b. Sholat Maghrib

$$\text{altitude} = 0,8333 - 0,0347 \cdot \sqrt{H} \dots\dots\dots(2.13)$$

c. Sholat Isya'

$$\text{altitude} = -(\text{sudut Isya}') \dots \dots \dots (2.14)$$

d. Sholat Subuh

$$\text{altitude} = -(\text{sudut Subuh}) \dots \dots \dots (2.15)$$

2.2 Penentuan Arah Kiblat

Pada saat ini metode yang sering digunakan dalam pengukuran arah kiblat ada tiga macam, yaitu memanfaatkan bayang-bayang kiblat, memanfaatkan arah utara geografis, memperhatikan ketika matahari tepat berada di atas Ka'bah (Majelis Tarjih dan Tajdid PP Muhammadiyah, 2009:41).

Penelitian kali ini akan menggunakan metode kedua, yaitu dengan memanfaatkan arah utara geografis. Pada metode ini, dibutuhkan data sebagai berikut:

- a. Koordinat Lintang Tempat (L)
- b. Koordinat Bujur Tempat (B)
- c. Koordinat Lintang Ka'bah (LK)
- d. Koordinat Bujur Ka'bah (BK)

Koordinat lintang dan bujur Ka'bah yang digunakan adalah $21^{\circ}25'21.07''$ untuk lintang, dan $39^{\circ}49'34.37''$ untuk nilai bujurnya (Jannah, 2009:11). Kemudian rumus yang digunakan adalah sebagai berikut (Anugraha, 2009):

$$\tan(K) = \frac{\sin(BK - B)}{\cos(L) \cdot \tan(LK) - \sin(L) \cdot \cos(B - BK)} \dots \dots \dots (2.16)$$

$$\cos(d) = \sin(L) \cdot \sin(LK) + \cos(L) \cdot \cos(LK) \cdot \cos(B - BK) \dots \dots (2.17)$$

Keterangan:

K = Arah kiblat

d = Jarak antara lokasi dan Ka'bah

2.3 Perangkat *iPhone*

iPhone adalah perangkat telepon genggam yang diproduksi oleh Apple untuk memenuhi kebutuhan konsumen akan sebuah perangkat yang menggabungkan antara perangkat komunikasi dan perangkat hiburan. *iPhone* adalah perangkat yang menggabungkan antara perangkat ponsel, *iPod*, dan perangkat untuk Internet (Apple Inc.).

iPhone generasi pertama dipasarkan pada tanggal 29 Juni 2007 di Amerika. *iPhone* terus berkembang, hingga pada tanggal 11 Juli 2008 Apple meluncurkan *iPhone* 3G, dan pada akhirnya pada tanggal 17 Juni 2009 meluncurkan generasi ketiganya, yaitu *iPhone* 3Gs.

Sampai saat ini, dua generasi terakhir itulah yang masih digemari oleh konsumen. Dua generasi tersebut memiliki fitur yang hampir sama, yaitu memiliki kamera, *bluetooth*, *Wi-Fi*, *accelerometer*, *voice memos*, *landscape keyboard*, dan GPS. Namun, *iPhone* 3Gs lebih unggul dengan fitur-fitur baru seperti *voice control*, *video recording*, *accessibility options*, kompas, dan ketahanan baterai yang melebihi *iPhone* 3G.

2.4 Bahasa Pemrograman Objective-C

Objective-C adalah sebuah bahasa pemrograman yang ditulis oleh Brad Cox pada awal tahun 1980. Bahasa pemrograman ini dirancang untuk pemrograman berorientasi obyek yang kompleks. Objective-C didefinisikan sebagai bahasa pemrograman berbasis C yang kecil, namun kuat. Pengembangan ini berdasarkan pada Smalltalk, sebuah bahasa pemrograman berorientasi obyek pertama. Objective-C dirancang untuk memberikan dukungan penuh terhadap pemrograman berorientasi obyek, namun dengan penulisan yang sederhana (Apple Inc., 2009:9).

Objective-C dipilih oleh Apple dalam pembuatan aplikasi pada *Mac*, *iPod* dan *iPhone* karena waktu yang digunakan untuk mengembangkan sebuah aplikasi menjadi lebih cepat jika dibandingkan dengan menggunakan bahasa lain. Hal ini

disebabkan karena terdapat beberapa kemudahan dalam penulisannya, dan memiliki *library* yang lengkap.

Perbedaan Objective-C dengan bahasa pemrograman lainnya adalah dalam pembuatan suatu kelas, diharuskan membuat minimal dua file, yaitu file *interface* dan file implementasinya. File *interface* berekstensi *.h* dan berisi deklarasi variabel dan fungsi dari kelas yang akan didefinisikan. File implementasi berekstensi *.m* dan berisi definisi dan implementasi dari fungsi-fungsi yang sudah dideklarasikan sebelumnya pada file *interface*.

Selain itu cara pembuatan *method setter* dan *method getter* pada Objective-C berbeda dengan bahasa pemrograman lainnya. Untuk pembuatan dua *method* tersebut digunakan kode program `@property` dilanjutkan dengan `@synthesize`. Berbeda dengan bahasa pemrograman lainnya yang mengharuskan penulisan *method* tersebut secara manual.

2.5 Cocoa Touch

Cocoa merupakan salah satu *Application Programming Interface* (API) dari Apple yang digunakan untuk pembuatan aplikasi berorientasi obyek untuk sistem operasi Mac OS X. Cocoa ini adalah salah satu dari lima API utama yang tersedia untuk Mac OS X. Empat API yang lain adalah Carbon, POSIX, X11, dan Java (Apple Inc., 2010:15).

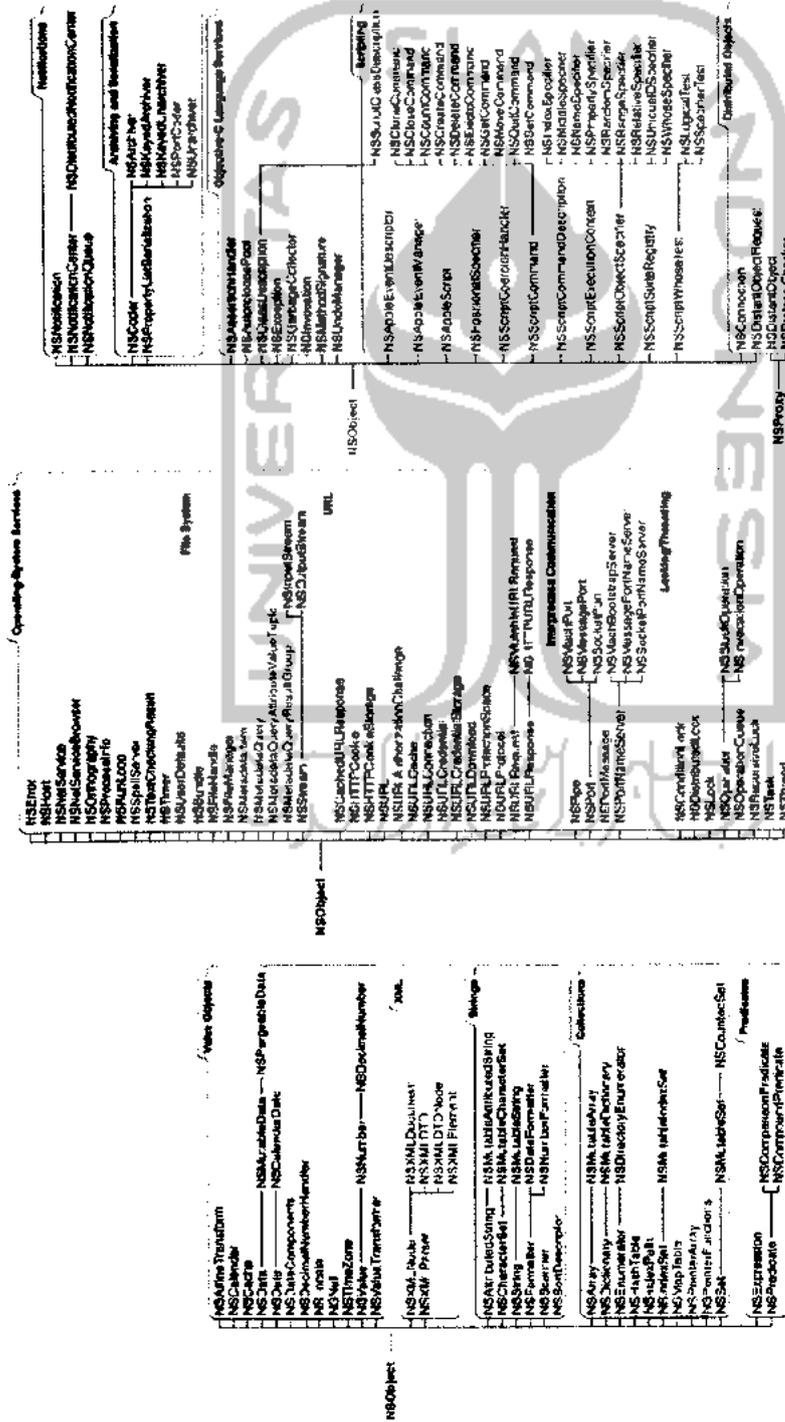
Cocoa Touch adalah sebuah pengembangan dari Cocoa yang dikhususkan untuk membangun aplikasi pada *iPhone*, *iPod Touch*, dan *iPad*. Cocoa Touch menyediakan lapisan abstraksi dari *iPhone OS*, sebuah sistem operasi untuk *iPhone*, *iPod Touch*, dan *iPad*. Cocoa Touch dikembangkan dengan menggunakan bahasa Objective-C.

Dalam penerapannya, Cocoa dibagi menjadi dua bagian, yaitu *Foundation framework* dan *UIKit framework*. Dua bagian ini tidak dapat dipisahkan, aplikasi tidak akan dapat dijalankan jika kita tidak menggunakan kedua bagian tersebut. Kelas, fungsi, tipe data, dan konstanta dalam *Foundation framework* menggunakan prefiks "NS", sedangkan dalam *UIKit framework* menggunakan prefiks "UI".

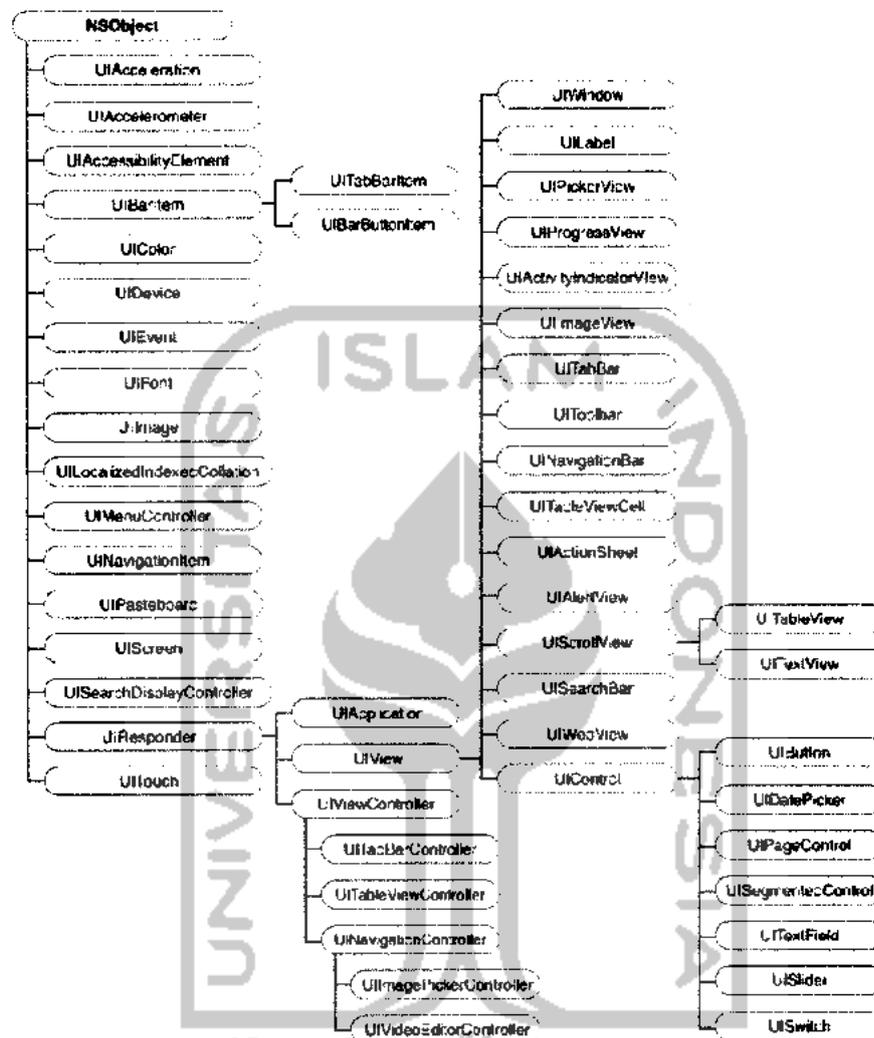
Foundation framework digunakan untuk mendefinisikan hal-hal umum dalam pemrograman, seperti manajemen memori dan memanipulasi obyek. Sedangkan *UIKit framework* digunakan untuk mendefinisikan dan memanipulasi *user interface*.

Kelas-kelas yang terdapat dalam *Foundation framework* dan *UIKit framework*, dapat dilihat pada gambar 2.2 dan gambar 2.3 di bawah ini:





Gambar 2.2 Hierarki Kelas pada Foundation Framework



Gambar 2.3 Hierarki Kelas pada *UIKit Framework*

Berikut ini adalah kelebihan-kelebihan yang dimiliki oleh Cocoa, yang akan memudahkan user dalam pembuatan aplikasi:

1. *User Interface Object*

Cocoa menyediakan banyak objek-objek untuk pembuatan antarmuka aplikasi. Objek-objek yang sudah tersedia dalam cocoa antara lain adalah *windows*, *text fields*, *image views*, *date pickers*, *progress indicators*, *table views*, *segmented controls*, *sliders*, *buttons*, dan *navigation bars*.

2. *Drawing and imaging*

Cocoa menyediakan fitur untuk menggambar dan melakukan manipulasi gambar secara manual. Hal ini dilengkapi dengan fitur untuk menjaga agar manipulasi gambar dapat berjalan dengan mudah, tanpa memikirkan adanya proporsional resolusi gambar, dan sebagainya.

3. *Performance*

Untuk menambah performa dari aplikasi yang akan dibangun, Cocoa menyediakan beberapa fitur yang mendukung untuk *multithreading*, *idle-time processing*, *lazy loading of resources*, *memory management*, dan *run-loop manipulation*.

4. *Networking*

Cocoa juga menyediakan fitur untuk berkomunikasi dengan *server* menggunakan koneksi *Internet Protocols*, maupun dengan koneksi *sockets*.

5. *Data Exchange*

Cocoa memudahkan *user* untuk pertukaran data dalam aplikasi maupun dengan aplikasi yang lainnya menggunakan metode *copy and paste*, maupun *drag and drop*.

2.6 GPS

Global Positioning System (GPS) adalah sistem navigasi berbasis satelit yang dapat mengirim dan menerima sinyal radio. Perangkat GPS menerima sinyal-sinyal kemudian memberikan sebuah informasi. Teknologi GPS berfungsi untuk menentukan lokasi, kecepatan, dan waktu (Garmin International, 2008:4).

GPS, secara resmi dikenal sebagai NAVSTAR (*Navigation Satellite Timing and Ranging*) *Global Positioning System*, dan awalnya dikembangkan untuk kebutuhan militer. Karena kemampuan navigasinya yang semakin populer dan dapat diakses menggunakan perangkat GPS yang kecil dan murah, pemerintah membuat sistem GPS yang disediakan untuk penggunaan secara umum.

Secara garis besar, prinsip kerja perangkat GPS adalah sebagai berikut (Garmin International, 2008:5):

1. Segmen kontrol melakukan pengecekan sinyal GPS, dan memberikan informasi kepada satelit GPS untuk mendapatkan akurasi maksimal.
2. Perangkat GPS akan mengumpulkan informasi dari satelit GPS.
3. Perangkat GPS melakukan pengecekan kesalahan dari satelit GPS, jika ternyata sinyal yang diterima oleh perangkat GPS mengalami gangguan.
4. Perangkat GPS akan menentukan lokasi, kecepatan, dan waktu.
5. Beberapa perangkat GPS dapat melakukan perhitungan lain seperti jarak perjalanan, jarak ke tujuan, waktu matahari terbit, waktu matahari terbenam, dan sebagainya.
6. Perangkat GPS akan menampilkan informasi pada layar.



BAB III METODOLOGI

3.1 Analisis Kebutuhan Perangkat Lunak

3.1.1 Metode Analisis

Tahap analisis digunakan untuk mengetahui semua kebutuhan dalam pengembangan perangkat lunak yang akan dibangun. Metode yang digunakan adalah metode analisis terstruktur dengan model aliran data. Jadi metode ini memiliki alur yang jelas dan teratur sehingga aplikasi yang akan dibangun adalah aplikasi yang detail dan terperinci.

3.1.2 Hasil Analisis

a. Analisis Kebutuhan Masukan Sistem

Berdasarkan analisis yang dilakukan, masukan yang dibutuhkan oleh sistem adalah sebagai berikut:

1. Koordinat lintang tempat.
2. Koordinat bujur tempat.
3. Zona waktu tempat.
4. Ketinggian lokasi.
5. Tanggal, bulan, dan tahun.
6. Tetapan panjang bayangan Ashar.
7. *Altitude* matahari waktu Subuh dan Isya.

b. Analisis Kebutuhan Proses

Proses-proses yang dilakukan oleh sistem adalah sebagai berikut:

1. Proses pengambilan data dari GPS.
2. Proses perhitungan waktu sholat, meliputi:
 - a. Proses perhitungan *equation of time*.
 - b. Proses perhitungan waktu sholat Dhuhur.

- c. Proses perhitungan waktu sholat Ashar.
 - d. Proses perhitungan waktu sholat Maghrib.
 - e. Proses perhitungan waktu sholat Isya'.
 - f. Proses perhitungan waktu sholat Subuh.
3. Proses perhitungan arah kiblat.
 4. Proses ganti pengaturan, meliputi:
 - a. Proses penggantian mazhab.
 - b. Proses penggantian ketinggian sudut matahari sholat Subuh.
 - c. Proses penggantian ketinggian sudut matahari sholat Isya'.
 - d. Proses penggantian pengaturan GPS.
 - e. Proses penggantian kota.
 - f. Proses penggantian tanggal.

c. Analisis Keluaran Sistem

Keluaran yang dihasilkan oleh sistem adalah jadwal sholat pada lokasi dan tanggal tersebut.

d. Kebutuhan Antarmuka

Antarmuka merupakan sarana yang menjadi penghubung antara pengguna dan sistem. Kebutuhan antarmuka yang dibutuhkan adalah antarmuka yang memudahkan *user* dalam melakukan akses terhadap sistem, nyaman serta meminimalkan terjadinya kesalahan.

e. Perangkat Lunak yang Dibutuhkan

Perangkat lunak yang digunakan dalam pembuatan aplikasi penentuan waktu sholat pada *iPhone* ini adalah sebagai berikut:

1. Mac OS

Mac OS adalah singkatan dari *Macintosh Operating System*. Mac OS adalah sistem operasi komputer yang dibuat oleh Apple khusus untuk komputer Macintosh dan tidak kompatibel dengan PC berbasis IBM.

2. **Xcode**

Xcode adalah sebuah *Integrated Development Environment* (IDE) yang digunakan untuk mengembangkan perangkat lunak pada Mac OS X, yang dikembangkan oleh Apple. Versi yang saat ini tersedia adalah 3.2.2 yang dilengkapi dengan *iPhone Software Development Kit* (SDK).

3. **Interface Builder**

Interface Builder adalah bagian dari Xcode, yang memudahkan pengguna untuk membuat antarmuka untuk aplikasi yang menggunakan *Graphical User Interface* (GUI). Antarmuka yang dihasilkan disimpan sebagai file *NeXT Interface Builder* (nib), atau untuk saat ini dikenal sebagai file xib.

4. **iPhone Simulator**

iPhone Simulator adalah aplikasi yang memudahkan pengguna untuk mensimulasikan aplikasi yang sedang dikembangkan sebelum melakukan instalasi aplikasi pada *iPhone* yang sebenarnya.

3.2 Perancangan Perangkat Lunak

3.2.1 Metode Perancangan

Metode perancangan yang digunakan adalah dengan *Data Flow Diagram* (DFD).

3.2.2 Hasil Perancangan

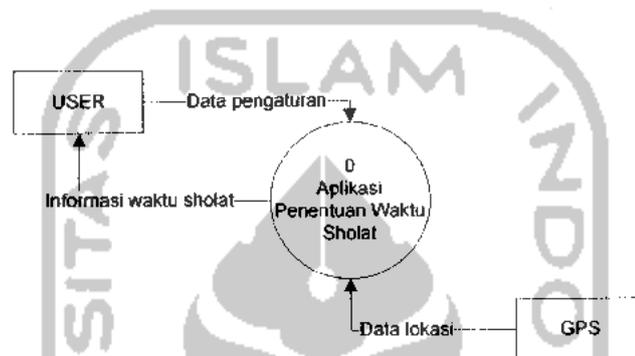
Hasil perancangan sistem yang akan dibangun adalah sebagai berikut:

a. **DFD**

Diagram yang menggunakan notasi untuk menggambarkan aliran data sistem disebut *Data Flow Diagram* (DFD). Dengan menggunakan diagram ini, dimungkinkan untuk menggambarkan sistem dari level yang paling tinggi menjadi level yang lebih rendah (dekomposisi), sehingga dapat memberikan kemudahan dalam pemahaman proses suatu sistem.

1) DFD Level 0

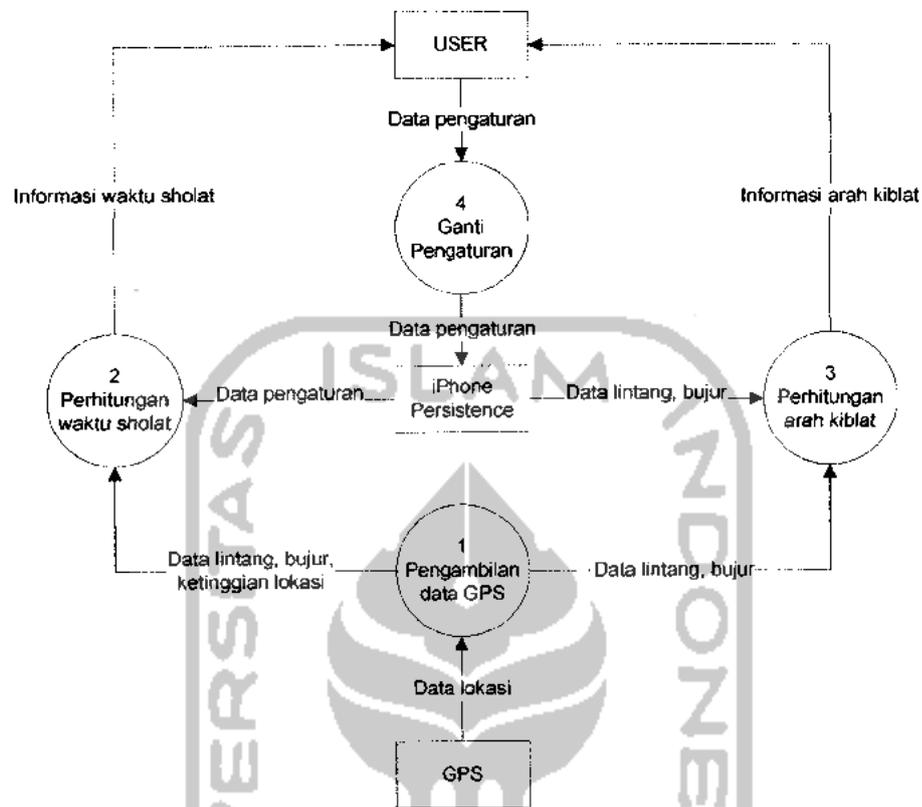
Perancangan ini dimulai dengan tahap perancangan diagram konteks yang sering disebut sebagai DFD level 0. Diagram ini merupakan bentuk paling global yang berisi tentang ruang lingkup kerja sistem dengan *entity-entity* luar yang berinteraksi dengan sistem tersebut. Penggunaan diagram arus data bertujuan untuk memudahkan dalam melihat arus data dalam sistem (lihat Gambar 3.1).



Gambar 3.1 DFD Level 0

2) DFD Level 1

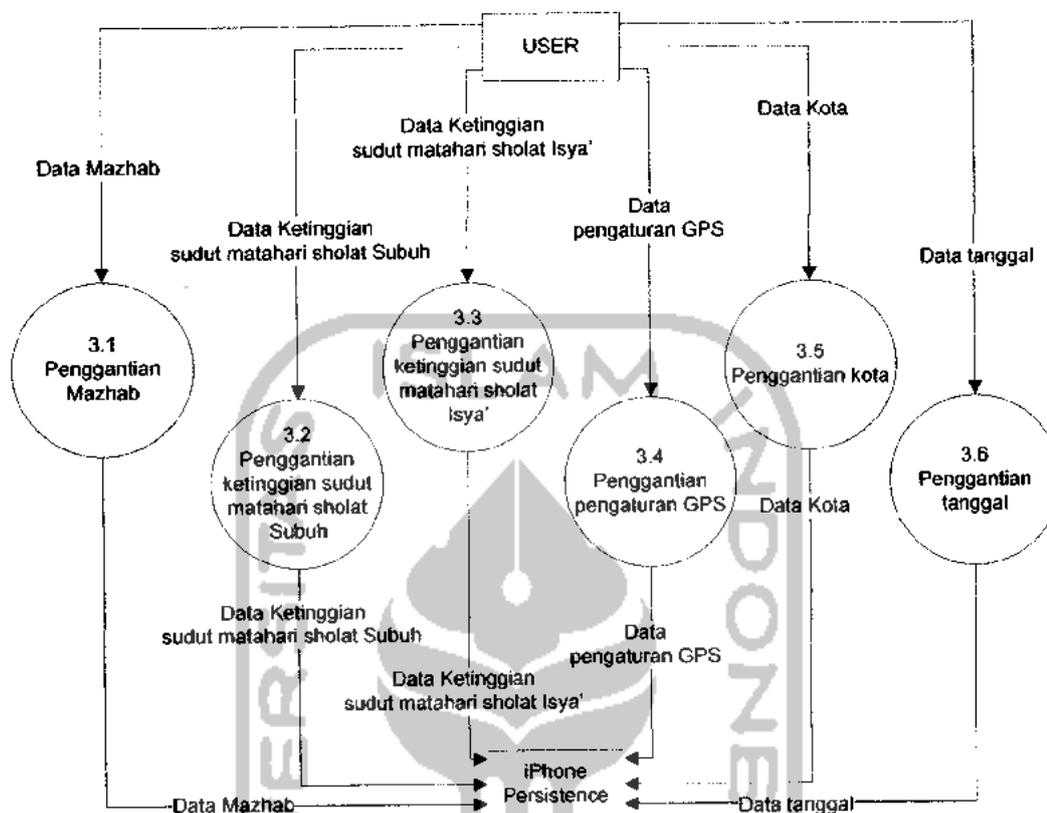
DFD level 1 menggambarkan proses yang terjadi dalam sistem. Pada DFD level 1 ada empat proses yaitu proses pengambilan data dari GPS, proses perhitungan waktu sholat, proses perhitungan arah kiblat, dan proses ganti pengaturan (lihat Gambar 3.2).



Gambar 3.2 DFD Level 1

3) DFD Level 2 Proses Perhitungan Waktu Sholat

DFD level 2 proses perhitungan waktu sholat terdiri dari enam proses utama, yaitu proses perhitungan *equation of time*, proses perhitungan waktu sholat Dhuhur, proses perhitungan waktu sholat Ashar, proses perhitungan waktu sholat Maghrib, proses perhitungan waktu sholat Isya', dan proses perhitungan waktu sholat Subuh (lihat Gambar 3.3).



Gambar 3.4 DFD Level 2 Proses ganti pengaturan

3.2.3 Perancangan Antarmuka

Perancangan antarmuka dari aplikasi penentuan waktu sholat ini dibuat dengan sederhana agar memudahkan pengguna dalam pengoperasiannya. Perancangan antarmuka aplikasi ini menggunakan menu *tab bar* yang terletak dibagian bawah halaman, dan terdiri dari delapan halaman sebagai berikut:

a. Halaman lihat jadwal sholat

Halaman ini merupakan halaman pertama yang dilihat oleh pengguna saat aplikasi dijalankan. Halaman ini menampilkan tanggal, waktu sholat dan mazhab yang digunakan (lihat Gambar 3.5).

Jadwal Sholat		
[Tanggal]		
Sholat Subuh	[jam]	
Sholat Dhuhur	[jam]	
Sholat Ashar	[jam]	
Sholat Maghrib	[jam]	
Sholat Isya	[jam]	
Perhitungan menggunakan rumus: [rumus]		
Jadwal Sholat	Arah Kiblat	Pengaturan

Gambar 3.5 Halaman lihat jadwal sholat

b. Halaman lihat arah kiblat

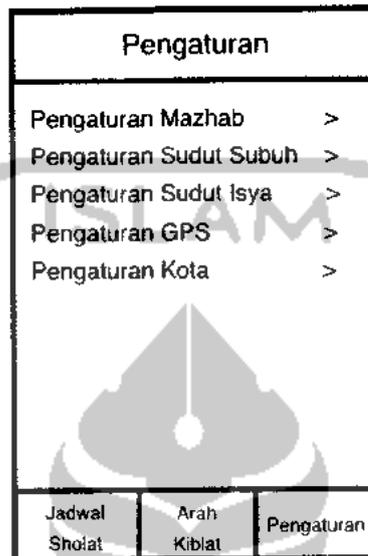
Halaman ini merupakan halaman saat pengguna memasuki menu lihat arah kiblat (lihat Gambar 3.6).

Arah Kiblat		
Koordinat Lintang	{lintang}	
Koordinat Bujur	{bujur}	
Arah Kiblat	{kiblat}	
Jarak ke ka'bah	{jarak}	
Jadwal Sholat	Arah Kiblat	Pengaturan

Gambar 3.6 Halaman lihat arah kiblat

c. Halaman ganti pengaturan

Halaman ini merupakan halaman saat pengguna memasuki menu ganti pengaturan (lihat Gambar 3.7).



Gambar 3.7 Halaman ganti pengaturan

d. Halaman ganti pengaturan mazhab

Halaman ini merupakan halaman saat pengguna memasuki menu ganti pengaturan mazhab (lihat Gambar 3.8).



Gambar 3.8 Halaman ganti pengaturan mazhab

e. Halaman ganti pengaturan sudut Subuh

Halaman ini merupakan halaman saat pengguna memasuki menu ganti pengaturan sudut Subuh (lihat Gambar 3.9).

< back Sudut Subuh

Sudut Subuh:

Keterangan:
Sudut subuh adalah ketinggian matahari pada saat kita melaksanakan sholat subuh. Nilai standar secara astronomis adalah 18 derajat.

Jadwal Sholat	Arah Kiblat	Pengaturan
---------------	-------------	------------

Gambar 3.9 Halaman ganti pengaturan sudut Subuh

f. Halaman ganti pengaturan sudut Isya'

Halaman ini merupakan halaman saat pengguna memasuki menu ganti pengaturan sudut Isya' (lihat Gambar 3.10).

< back Sudut Isya

Sudut Isya:

Keterangan:
Sudut isya adalah ketinggian matahari pada saat kita melaksanakan sholat isya. Nilai standar secara astronomis adalah 18 derajat.

Jadwal Sholat	Arah Kiblat	Pengaturan
---------------	-------------	------------

Gambar 3.10 Halaman ganti pengaturan sudut Isya'

g. Halaman ganti pengaturan kota

Halaman ini merupakan halaman saat pengguna memasuki menu ganti pengaturan sudut Isya' (lihat Gambar 3.11).

< back		Kota
List Kota:		
Kota A	<input type="checkbox"/>	
Kota B	<input type="checkbox"/>	
Kota C	<input type="checkbox"/>	
Kota D	<input type="checkbox"/>	
Kota E	<input checked="" type="checkbox"/>	
Kota F	<input type="checkbox"/>	
...		
Jadwal Sholat	Arah Kiblat	Pengaturan

Gambar 3.11 Halaman ganti pengaturan kota

h. Halaman ganti tanggal

Halaman ini merupakan halaman saat pengguna memasuki menu ganti pengaturan tanggal (lihat Gambar 3.12).

< back		Tanggal
Tanggal:	<input type="text"/>	
Bulan:	<input type="text"/>	
Tahun:	<input type="text"/>	
Jadwal Sholat	Arah Kiblat	Pengaturan

Gambar 3.12 Halaman ganti tanggal

BAB IV HASIL DAN PEMBAHASAN

4.1 Implementasi Perangkat Lunak

Implementasi perangkat lunak bertujuan untuk memastikan perangkat lunak yang dibangun dapat bekerja baik dan sesuai dengan tujuan dibangun. Sebelum perangkat lunak diimplementasikan, maka perangkat lunak harus bebas dari kesalahan. Kesalahan yang mungkin terjadi antara lain kesalahan tampilan ataupun kesalahan proses. Setelah perangkat lunak bebas dari kesalahan, kemudian dapat dilakukan pengujian dengan menjalankan perangkat lunak.

4.1.1 Implementasi Proses

Proses-proses yang terdapat dalam sistem ini adalah sebagai berikut:

a. Proses Pengambilan data GPS

Proses pengambilan data dari GPS dilakukan dengan melakukan import *library* “*Core Location*”, kemudian mengimplementasi *method* “*didUpdateToLocation*” untuk mendapatkan titik koordinat terakhir. Setelah mendapatkan titik koordinat maka dapat diketahui nilai bujur, lintang, dan ketinggian lokasi. Kemudian tiga nilai tersebut disimpan dalam suatu *array* yang nantinya akan digunakan untuk menghitung waktu shalat.

b. Proses Perhitungan Waktu Sholat

Proses perhitungan waktu shalat dibagi lagi menjadi enam proses, yaitu:

1. Proses perhitungan *equation of time*

Proses perhitungan ini adalah proses yang paling penting dalam perhitungan waktu shalat. Proses ini mengambil data dari *array* yang telah disediakan oleh proses pengambilan data GPS, atau dari data penyimpanan

daftar kota jika GPS tidak tersedia. Setelah mendapatkan data tersebut, proses ini menghitung nilai *equation of time* yang telah dibagi menjadi beberapa fungsi dalam pemrogramannya. Kode pemrograman dari proses ini adalah sebagai berikut:

```

-(double) hitungJulianDay:(int)tgl:(int)bln:(int)thn{
    float JD;
    if(bln == 1 || bln == 2){
        bln = bln + 12;
        thn = thn - 1;
    }
    int A = (int) thn/100;
    int B = 2 + (int) ((A/4) - A);
    JD = 1720994.5 + ((int) (365.25*thn)) + ((int) (30.6001 * (bln
    + 1))) + B + tgl + (12/24.0);

    return JD;
}

-(double) hitungJulianDayLocal:(double) JD_GMT:(int) z{
    double JDLocal = JD_GMT - (z/24.0);

    return JDLocal;
}

-(double) hitungEquationOfTime:(double) JD_Local{
    double U = (JD_Local - 2451545)/36525.0;

    double L0 = ((280.46607 + (36000.7698 * U)) * PHI / 180.0);

    double EOT = {(-1*(1789 + 237 * U))*sin(L0) - (7146 -
    62*U)*cos(L0) + (9934 - 14*U)*sin(2*L0) - (29 +
    5*U)*cos(2*L0) + (74 + 10*U)*sin(3*L0) + (320 -
    4*U)*cos(3*L0) - 212*sin(4*L0))/1000.0;
    return EOT;
}

```

2. Proses perhitungan waktu sholat Dhuhur

Proses berikutnya adalah menghitung waktu sholat Dhuhur dengan menggunakan rumus yang ada. Hasil dari perhitungan *equation of time* digunakan dalam perhitungan ini. Kode pemrograman untuk proses ini adalah sebagai berikut:

```

-(double) hitungWaktuDhuhur{
double JDGMT = [self hitungJulianDay: self.tanggal: self.bulan:
self.tahun];
double JDLocal = [self hitungJulianDayLocal: JDGMT: self.zona];
double EOTime = [self hitungEquationOfTime: JDLocal];
double B = self.bujur;

double dhuhur = 12 + self.zona - B/15 - EOTime/60;

return dhuhur;
}

```

3. Proses perhitungan waktu sholat Ashar

Proses perhitungan waktu sholat Ashar dipengaruhi oleh nilai dari hasil perhitungan waktu sholat Dhuhur, dan juga dipengaruhi oleh nilai tetapan sholat Ashar yang ditentukan oleh beberapa mazhab. Sebelum menghitung waktu sholat Ashar, proses ini terlebih dahulu menghitung *hour angle* untuk dijadikan salah satu parameter perhitungan. Kode pemrograman yang digunakan untuk menghitung waktu sholat Ashar adalah sebagai berikut:

```

-(double) hitungDeklinasiMatahari:(double) JD_Local{
double T = 2 * PHI * (JD_Local - 2451545) / 365.25;
double Delta = 0.37877 + 23.264 * sin((57.297 * T - 79.547) *
PHI /180.0)+0.3812 * sin((2 * 57.297 * T - 82.682) *
PHI/180.0)+0.17132*sin((3*57.297*T-59.722)*PHI/180.0);
double Delta_radian = Delta * PHI / 180.0;
}

```

```

        return Delta_radian;
    }

    -(double) hitungHourAngle:(double) altitude{
        double lintangRad = self.lintang * PHI / 180;
        double JDGMT = [self hitungJulianDay: self.tanggal: self.bulan:
            self.tahun];
        double JDLocal = [self hitungJulianDayLocal: JDGMT: self.zona];
        double Delta = [self hitungDeklinasiMatahari: JDLocal];
        double CosHA = (sin(altitude) - sin(Delta) * sin(lintangRad)) /
            (cos(Delta)*cos(lintangRad));
        double HA = acos(CosHA)*180/PHI;

        return HA;
    }

    -(double) hitungWaktuAshar:(double)dhuhur{
        double lintangRad = self.lintang * PHI / 180;
        double JDGMT = [self hitungJulianDay: self.tanggal: self.bulan:
            self.tahun];

        double JDLocal = [self hitungJulianDayLocal: JDGMT: self.zona];
        double Delta = [self hitungDeklinasiMatahari: JDLocal];

        double altitude = atan(1/([self absolute:(lintangRad-Delta)]));

        double HAAshar = [self hitungHourAngle:altitude];

        double ashar = dhuhur + (HAAshar / 15);

        return ashar;
    }
}

```

4. Proses perhitungan waktu sholat Maghrib

Proses perhitungan waktu Maghrib dipengaruhi oleh nilai dari hasil perhitungan waktu sholat Duhur, dan juga dipengaruhi oleh ketinggian suatu lokasi. Kode pemrograman untuk proses ini adalah sebagai berikut:

```

-(double) hitungWaktuMaghrib:(double) dhuhur{
double altitude = (-0.8333-0.0347*(sqrt([self
ketinggian]))) * PHI/180; //udah di radian
double HAMaghrib = [self hitungHourAngle:altitude];
double maghrib = dhuhur + (HAMaghrib / 15);

return maghrib;
}

```

5. Proses perhitungan waktu sholat Isya'

Proses perhitungan waktu sholat Isya' dipengaruhi oleh nilai dari hasil perhitungan waktu sholat Duhur, dan juga dipengaruhi oleh nilai sudut Isya'. Kode pemrograman dari proses perhitungan waktu sholat Isya' ini adalah sebagai berikut:

```

-(double) hitungWaktuIsya:(double) dhuhur{
double altitude = -([self sudutIsya]) * PHI/180;
double HAIIsya = [self hitungHourAngle:altitude];
double isya = dhuhur + (HAIIsya / 15);

return isya;
}

```

6. Proses perhitungan waktu sholat Subuh

Proses perhitungan waktu sholat Subuh dipengaruhi oleh hasil perhitungan waktu sholat Duhur, dan juga dipengaruhi oleh nilai dari sudut Subuh. Kode pemrograman dari proses ini adalah sebagai berikut:

```

-(double) hitungWaktuSubuh:(double) dhuhur{

```

```

double altitude = -([self sudutSubuh])*PHI/180;
double HASubuh = [self hitungHourAngle:altitude];
double subuh = dhuhur - (HASubuh / 15);

return subuh;
}

```

Setelah melakukan berbagai perhitungan waktu sholat, kemudian sistem akan menampilkan hasil dari perhitungan tersebut. Hasil ditampilkan dengan menggunakan objek dari kelas *UITableView* agar tampilan menjadi rapi. *UITableView* adalah kelas standar yang digunakan dalam pemrograman *iPhone* untuk menampilkan data seperti tabel satu kolom dan dapat dikonfigurasi sesuai dengan keinginan. *UITableView* juga dipakai oleh berbagai aplikasi-aplikasi standar *iPhone* seperti *iPod*, *settings*, *YouTube*, *Photos*, dan sebagainya (Apple Inc., 2009:212). Adapun hasil yang ditampilkan oleh sistem adalah seperti pada Gambar 4.1.

Sabtu, 7 Agustus 2010	
Sholat Subuh	04:54:06
Sholat Dhuhur	12:15:48
Sholat Ashar	03:37:43
Sholat Maghrib	06:37:45
Sholat Iya'	07:29:07

Gambar 4.1 Halaman lihat jadwal sholat

c. Proses Perhitungan Arah Kiblat

Proses perhitungan arah kiblat dilakukan dengan cara mengambil pengaturan lintang dan bujur tempat, kemudian menghitung arah kiblat dengan

rumus yang ada. Kode pemrograman yang digunakan dalam proses ini adalah sebagai berikut:

```

-(double) hitungArahKiblat{
    double BK = BujurKabah * PHI / 180;
    double BT = self.bujur * PHI / 180;
    double LK = LintangKabah * PHI / 180;
    double LT = self.lintang * PHI / 180;

    double tanK = sin(BK-BT) / (cos(LT)*tan(LK) -
        sin(LT)*cos(BT-BK));

    double K_radian = atan(tanK);

    double K = K_radian * 180 / PHI;
    if(K<0){
        K+=360;
    }

    return K;
}

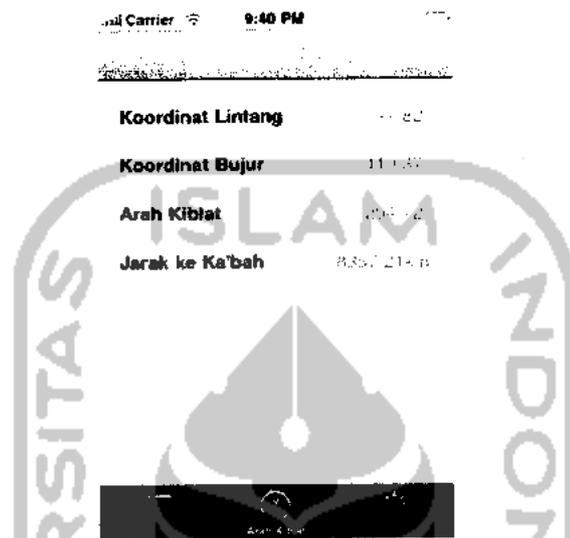
-(double) hitungJarakKabah{
    double BK = BujurKabah * PHI / 180;
    double BT = self.bujur * PHI / 180;
    double LK = LintangKabah * PHI / 180;
    double LT = self.lintang * PHI / 180;

    double cosD = sin(LT)*sin(LK) + cos(LT)*cos(LK)*cos(BT -
BK);
    double D = acos(cosD);
    double jarak = 6378.137 * D;

    return jarak;
}

```

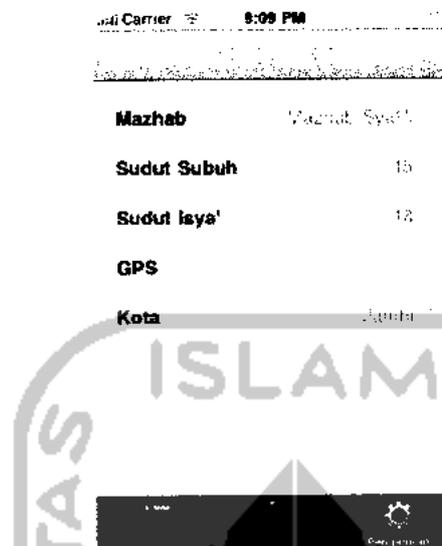
Setelah melakukan perhitungan tersebut, proses selanjutnya adalah menampilkannya kepada *user*. Tampilan dari proses ini adalah seperti pada Gambar 4.2.



Gambar 4.2 Halaman lihat arah kiblat

d. Proses Ganti Pengaturan

Proses ganti pengaturan adalah sebuah proses untuk menampilkan form-form pengaturan, kemudian menerima hasil pengaturan yang dimasukkan oleh *user*, dan menyimpannya kedalam file penyimpanan pada sistem. *UITableView* digunakan untuk menampilkan menu pengaturan, dan *UINavigationController* digunakan untuk mengatur menu-menu tersebut kedalam sebuah hirarki yang rapi. Tampilan dari halaman utama ganti pengaturan adalah seperti pada Gambar 4.3.



Gambar 4.3 Halaman ganti pengaturan

4.2 Analisis Kinerja Perangkat Lunak

Analisis kinerja perangkat lunak merupakan tahap dimana sistem dioperasikan pada tahap yang sebenarnya, sehingga akan diketahui apakah sistem yang telah dibuat benar-benar sesuai dengan yang direncanakan. Pada bagian ini juga dilakukan beberapa pengujian untuk penanganan kesalahan yang terdapat dalam aplikasi.

4.2.1 Pengujian Normal

Pengujian normal dilakukan dengan cara memberikan masukan pada aplikasi. Berikut beberapa pengujian yang dilakukan pada aplikasi penentuan waktu sholat.

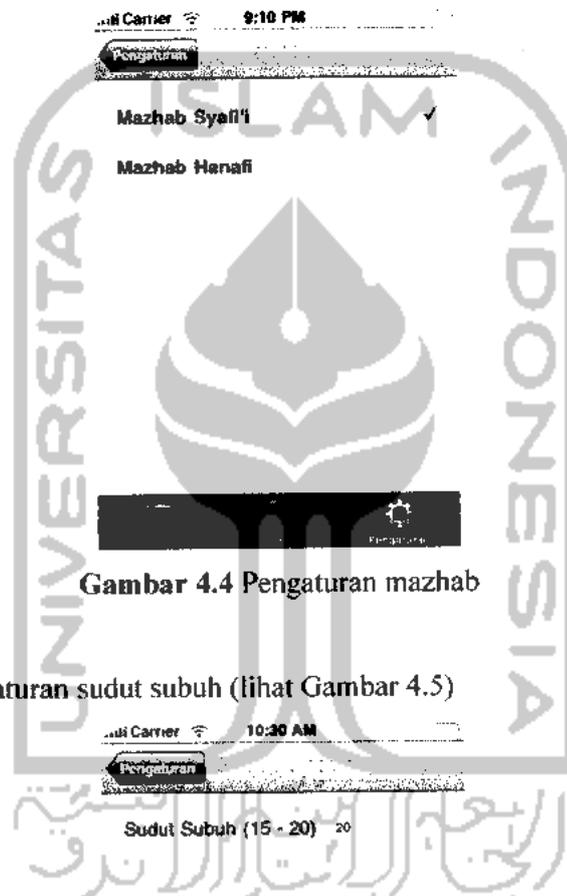
a. Pengujian Perhitungan Jadwal Sholat

Jadwal sholat yang akan diujikan adalah pada tanggal 7 Agustus 2010, dengan menggunakan mazhab syafi'i. Kemudian nilai sudut subuh yang digunakan adalah 20 derajat, dan sudut isya yang digunakan adalah 18 derajat. Pengujian tidak bisa dilakukan menggunakan fitur GPS dalam aplikasi

dikarenakan *iPhone Simulator* tidak terdapat perangkat GPS. Oleh karena itu, pengujian dilakukan dengan memilih satu kota, yaitu kota Jakarta.

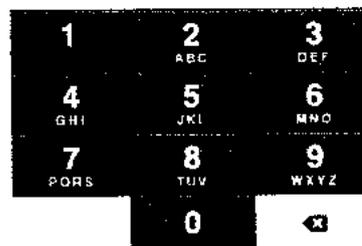
Berikut adalah tampilan dari pengaturan yang disesuaikan dengan studi kasus diatas:

1. Pengaturan mazhab (lihat Gambar 4.4)



Gambar 4.4 Pengaturan mazhab

2. Pengaturan sudut subuh (lihat Gambar 4.5)



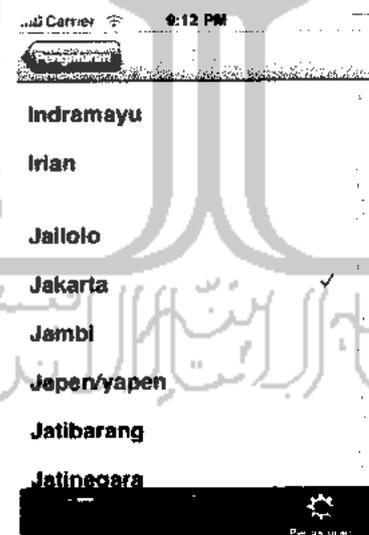
Gambar 4.5 Pengaturan sudut subuh

3. Pengaturan sudut isya (lihat Gambar 4.6)



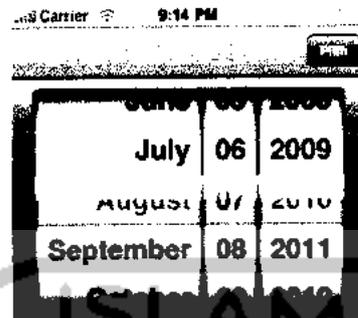
Gambar 4.6 Pengaturan sudut isya

4. Pengaturan kota (lihat Gambar 4.7)



Gambar 4.7 Pengaturan kota

5. Pengaturan tanggal (lihat Gambar 4.8)



Gambar 4.8 Pengaturan tanggal

Dari pengaturan tersebut, didapatkan hasil perhitungan jadwal sholat sebagai berikut (lihat Gambar 4.9).

Sabtu, 7 Agustus 2010	
Sholat Subuh	07:37:13
Sholat Dhuhur	11:36:32
Sholat Ashar	15:20:21
Sholat Maghrib	17:07:45
Sholat Iya'	19:36:31

Gambar 4.9 Hasil perhitungan jadwal sholat

Hasil dari perhitungan tersebut dibandingkan dengan hasil perhitungan dari aplikasi WinHisab yang dibuat oleh Departemen Agama Republik Indonesia (Badan Hisab dan Rukyat Departemen Agama Republik Indonesia, 1996), dan juga data dari Jakarta Islamic Centre (Jakarta Islamic Centre, 2010). Perhitungan menggunakan parameter yang sama untuk setiap sumber perhitungan, dan didapatkan hasil perbandingan seperti pada Tabel 4.1.

Tabel 4.1 Tabel Perbandingan Hasil Perhitungan Waktu Sholat
Pada Tanggal 7 Agustus 2010

	Subuh	Dhuhur	Ashar	Maghrib	Isya
Aplikasi Penentuan Waktu Sholat	4:42	11:58	15:20	17:57	19:06
WinHisab Departemen Agama RI	4:44	12:01	15:22	17:57	19:08
Jakarta Islamic Centre	4:51	11:58	15:20	17:55	19:06

Dari hasil perbandingan, didapatkan selisih kurang lebih tiga menit untuk setiap waktu sholat, maka dapat disimpulkan bahwa perhitungan yang diperoleh dari aplikasi penentuan waktu sholat ini cukup akurat.

b. Pengujian Perhitungan Arah Kiblat

Perhitungan arah kiblat yang akan diujikan adalah pada kota Yogyakarta, Aceh, dan San Francisco. Hasil dari perhitungan arah kiblat tersebut dibandingkan dengan hasil perhitungan dari aplikasi Qibla Locator (Qibla Locator, 2010). Perhitungan menggunakan parameter yang sama dan didapatkan hasil perhitungan seperti pada Tabel 4.2.

Tabel 4.2 Tabel Perbandingan Hasil Perhitungan Arah Kiblat

	Yogyakarta		Aceh		San Francisco	
	Arah Kiblat	Jarak	Arah Kiblat	Jarak	Arah Kiblat	Jarak
Aplikasi Penentuan waktu sholat	294,71°	8356,28 km	292,59°	6504,74 km	19,32°	13225,89 km
Qibla Locator	294,71°	8357 km	292,25°	6259 km	19,26°	13222 km

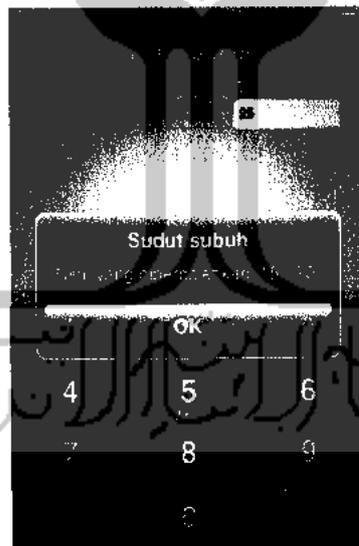
Dari hasil perbandingan, didapatkan nilai yang sama, maka dapat disimpulkan bahwa perhitungan arah kiblat yang diperoleh dari aplikasi ini cukup akurat.

4.2.2 Pengujian Tidak Normal

Pengujian tidak normal adalah dengan memasukkan konfigurasi yang tidak sesuai dengan keadaan pada umumnya. Pengujian yang dilakukan adalah sebagai berikut:

a. Penanganan Kesalahan Masukan Data Kosong

Penanganan kesalahan ini dilakukan untuk menangkap setiap kesalahan yang terjadi ketika *field* pada pengaturan isian data kosong atau tidak sesuai dengan rentang nilai yang diperbolehkan. Contoh penanganan kesalahan ini terdapat pada pengaturan sudut Subuh dan Isya'. Jika nilai dari sudut dikosongkan, maka akan muncul peringatan seperti pada Gambar 4.10.



Gambar 4.10 Penanganan kesalahan masukan data kosong

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Setelah dilakukan serangkaian pengujian pada aplikasi penentuan waktu shalat yang telah dibuat dalam penelitian Tugas Akhir ini, dapat disimpulkan bahwa:

1. Jadwal shalat dapat dihitung jika diketahui letak geografis suatu lokasi.
2. Aplikasi penentuan waktu shalat dapat diimplementasikan pada *iPhone*.
3. Dengan adanya aplikasi ini, *user* dapat memperoleh informasi jadwal shalat dengan cepat dan mudah.
4. Hasil perhitungan waktu shalat yang didapatkan cukup akurat.
5. Hasil perhitungan arah kiblat yang didapatkan cukup akurat.
6. Sistem mampu menangani kesalahan masukan data atau memiliki kemampuan "*error handling*" yang cukup baik.

5.2 Saran

Hasil dari penelitian dan pengujian pada perangkat lunak ini ditemukan beberapa keterbatasan pada sistem, untuk itu pengembangan penelitian dimasa yang akan datang dapat ditambahkan beberapa hal sebagai berikut:

1. Perlu adanya lisensi dari Apple, Inc. untuk memudahkan perancangan dan pengujian dikarenakan lisensi tersebut memungkinkan aplikasi yang sedang dibangun dapat diujikan pada *iPhone* yang sesungguhnya.
2. Penentuan arah kiblat perlu adanya tambahan fitur kompas yang ditentukan dengan posisi matahari.
3. Perlu adanya penambahan fitur untuk pengingat waktu shalat dengan alarm.

DAFTAR PUSTAKA

- Anugraha, Rinto. 2009. *Cara Menghitung Waktu Sholat*. Diakses dari <http://www.eramuslim.com/syariah/ilmu-hisab/cara-menghitung-waktu-shalat.htm>, pada tanggal 13 Mei 2010.
- Anugraha, Rinto. 2009. *Segitiga Bola dan Arah Kiblat*. Diakses dari <http://www.eramuslim.com/syariah/ilmu-hisab/segitiga-bola-dan-arrah-kiblat.htm>, pada tanggal 11 Agustus 2010.
- Apple, Inc. 2009. *The Objective-C Programming Language*. California: Author.
- Dave, M& Lamarche, J. 2009. *Beginning iPhone Development: Exploring the iPhone SDK*. USA: Apress.
- Garmin International. 2008. *GPS Beginner's Guide*. Kansas: Garmin Corporation.
- Badan Hisab dan Rukyat Departemen Agama Republik Indonesia. 1996. WinHisab 2.0.
- Jannah, Sofwan. 2009. Akurasi Pengukuran Arah Kiblat dengan Software Google Earth. Makalah disampaikan pada *Kunjungan Silaturahmi Dosen dan Mahasiswa Fakultas Syari'ah IAIN Raden Fatah Palembang*, Kampus FIAI UII. Yogyakarta, 27 Juli.
- Jakarta Islamic Centre. 2010. Jadwal Sholat untuk Jakarta. Diakses dari <http://islamic-center.or.id/agenda-mainmenu-28/jadwal-sholat>, pada tanggal 25 Juli 2010.

Tim Majelis Tarjih dan Tajdid PP Muhammadiyah. 2009. *Pedoman Hisab Muhammadiyah*. Yogyakarta: Majelis Tarjih dan Tajdid PP Muhammadiyah.

Qibla Locator. 2010. *Qibla Locator*. Diakses dari <http://www.qiblalocator.com/>, pada tanggal 11 Agustus 2010.

Suryantoro, Johan. 2009. *Daftar titik koordinat peta untuk kota / tempat di Indonesia*. Diakses dari <http://www.garisline.com/2009/11/daftar-titik-koordinat-peta-untuk-kota.html>, pada tanggal 12 Juli 2010.

