

IMPLEMENTASI ALGORITMA MEEUS DALAM PENENTUAN WAKTU SHALAT DAN PENCARIAN MASJID TERDEKAT

Dede Muhammad Isnaeni¹, Fitri Mintarsih², Feri Fahrianto³

Fakultas Sains dan Teknologi^{1,2,3}

Universitas Islam Negeri Syarif Hidayatullah Jakarta, Indonesia

ABSTRACT

In an age of emerging technologies such as the role of the communications media today is in need in answering every need, both physical and spiritual needs. For physical needs such as social media and so forth, for the spiritual needs as a religious lecture, digital Qur'an, and the timing of prayer. To be able to determine prayer times, a lot of methods and different opinions on the use of the algorithm determining the prayer time, one of which is the algorithm Meeus. Jean Meeus is a Belgian astronomer and mathematician born in 1928. He studied mathematics at the University of Leuven, Belgium, and graduated in 1953. He was interested in astronomy and mechanics of celestial bodies ball. Jean Meeus wrote many books on mathematical astronomy, such as the Canon of Solar eclipses, Elements of Solar eclipses 1951-2200, Canon of Lunar eclipses, Astronomical Formulae for Calculators, Astronomical Algorithms, Transits, Astronomical Tables of the Sun, Moon and Planets, Mathematical Astronomy Morsels . For his services in the field of astronomy, an asteroid found named asteroid 2213 Meeus. Meeus algorithm is an algorithm that is widely used for astronomical calculations because it is known with sufficient accuracy is high. Meeus algorithm itself is the reduction of VSOP87 algorithm which has a high degree of accuracy, of the thousands of tribes VSOP87 collection of algorithms to determine the position of the sun, then that counts is about hundreds of tribes large and important in this Meeus algorithm, while the tribes small is not taken into account. This does not reduce the accuracy of the calculation algorithm itself Meeus, therefore the author uses this Meeus algorithm in determining the time the five daily prayers. The method of searching a nearby mosque, the author uses the data base sourced from Google, this is due to the limitations of time, energy and ability to re-writer to record the mosque to be included in this study.

Keywords: *determination of the time of prayer, bookmark the nearest mosque, Smartphone, Android.*

ABSTRAK

Di zaman perkembangan teknologi seperti saat ini, peran media komunikasi saat ini sangat penting dalam menjawab setiap kebutuhan, baik kebutuhan jasmani dan rohani. Untuk kebutuhan fisik seperti media sosial dan lain sebagainya, untuk kebutuhan rohani sebagai ceramah agama, digital Qur'an, dan waktu shalat. Untuk dapat menentukan waktu shalat, banyak metode dan pendapat yang berbeda tentang penggunaan algoritma menentukan waktu shalat, salah satunya adalah algoritma Meeus. Jean Meeus adalah seorang astronom Belgia dan matematika lahir pada tahun 1928. Ia belajar matematika di Universitas Leuven, Belgia, dan lulus pada tahun 1953. Dia tertarik pada astronomi dan mekanika bola benda langit. Jean Meeus menulis banyak buku tentang matematika astronomi, seperti Canon gerhana Matahari, Elements Solar gerhana 1951-2200, Canon of Lunar gerhana, Astronomical Formula untuk Kalkulator, Astronomi Algoritma, Transit, Tabel Astronomi dari Matahari, Bulan dan Planet, Matematika Astronomi Morsels. Untuk jasanya di bidang astronomi, sebuah asteroid ditemukan bernama asteroid 2213 Meeus. Algoritma Meeus merupakan algoritma yang banyak digunakan untuk perhitungan astronomi karena diketahui dengan akurasi yang cukup tinggi. algoritma Meeus sendiri adalah pengurangan algoritma VSOP87 yang memiliki tingkat akurasi yang tinggi, dari ribuan suku VSOP87 koleksi algoritma untuk menentukan posisi matahari, maka yang penting adalah tentang ratusan suku besar dan penting dalam algoritma Meeus ini, sedangkan suku kecil tidak diperhitungkan. Ini tidak mengurangi akurasi dari algoritma perhitungan sendiri Meeus, oleh karena itu penulis menggunakan algoritma Meeus ini dalam menentukan waktu shalat lima waktu. Metode mencari masjid terdekat, penulis menggunakan data base yang bersumber dari Google, hal ini disebabkan keterbatasan waktu, tenaga dan kemampuan untuk kembali penulis untuk merekam masjid untuk dimasukkan dalam penelitian ini.

Kata kunci: *penentuan waktu shalat, penunjuk masjid terdekat, Smartphone, Android.*

1. Pendahuluan

Saat ini teknologi informasi semakin mengalami kemajuan yang sangat pesat. Teknologi informasi adalah suatu teknologi yang digunakan untuk mengolah data, termasuk memproses, mendapatkan, menyusun, menyimpan, memanipulasi data dalam berbagai cara untuk menghasilkan informasi yang berkualitas, efisien, tepat waktu dan menarik untuk dikonsumsi publik. Perkembangan Teknologi Informasi memacu suatu cara baru dalam kehidupan, dimulai dari cara penentuan waktu dalam menyusun agenda pekerjaan, pendidikan, kesehatan serta membantu dalam menentukan waktu kewajiban kita seperti halnya shalat.

Shalat adalah suatu kewajiban setiap individu muslim, pada zaman kemajuan teknologi informasi seperti sekarang ini, tentunya dapat membantu kualitas ibadah shalat, hal ini tentunya mendorong kita untuk lebih meningkatkan kualitas beribadah kepada Allah SWT. Kemajuan teknologi ini salah satunya dapat diterapkan pada cara kita menentukan waktu shalat, jika zaman dahulu penentuan waktu shalat dilihat dari cuaca, maka zaman sekarang dapat diketahui dengan hitungan matematis yang dapat diketahui dari handset android.

Algoritma meeus merupakan algoritma yang sudah banyak digunakan untuk perhitungan astronomi karena dikenal dengan keakuratannya yang cukup tinggi, sebagai contoh thesis yang ditulis oleh Farid Wajdi, mahasiswa IAIN walisongo yang berjudul Penerapan Algoritma Jean Meeus dalam Pengukuran Arah Kiblat dengan Theodolite, pada thesis ini dibahas bagaimana penerapan algoritma meeus dapat menentukan arah kiblat, adapun kekurangan dari thesis ini adalah penerapannya masih berbasis desktop, dan belum menerapkan basis mobile sehingga hanya dapat digunakan ketika kita sedang menggunakan komputer saja.

Lalu skripsi yang ditulis oleh Arif Agus P, mahasiswa Universitas Brawijaya jurusan fisika yang berjudul Pembuatan Aplikasi untuk Menentukan Fase dan Visibilitas Bulan dengan Menggunakan Algoritma Jean Meeus, sama seperti thesis diatas, skripsi ini masih bersifat desktop, sehingga hanya dapat digunakan ketika kita sedang menggunakan komputer saja.

Maka dalam hal ini penulis mencoba mengimplementasikan algoritma meeus dalam hal perhitungan waktu shalat berbasis mobile, sehingga pengguna dapat menggunakan algoritma meeus ini kapan saja dan dimana saja selama pengguna membawa handset android.

Algoritma meeus itu sendiri adalah reduksi dari algoritma VSOP87 yang memiliki tingkat akurasi tinggi, dari ribuan suku koreksi dari algoritma VSOP87 untuk menentukan posisi

matahari, maka yang diperhitungkan adalah sekitar ratusan suku-suku yang besar dan penting pada algoritma meeus ini, adapun suku-suku yang kecil tidak diperhitungkan.

Hal ini tidak mengurangi keakuratan dari perhitungan algoritma meeus itu sendiri, oleh karena itu penulis menggunakan algoritma meeus ini dalam menentukan waktu shalat lima waktu.

2. Penelitian Sejenis

A. Aplikasi Adzan dan Peningkat Shalat menggunakan Global Positioning System (GPS) berbasis Android 2.2

Sumber : Hasan Abdul Malik

Pada studi literatur pertama ini, fitur yang di tawarkan hanya berupa peningat waktu shalat dan pemilihan adzan saja, tanpa ada fitur lain yang mendukung kegiatan shalat.

B. Aplikasi Peningkat Shalat dan Arah Kiblat Menggunakan Global Positioning System (GPS) berbasis Android 1.6

Sumber : Muhammad Amiral

Pada studi literatur yang kedua ini, fitur yang ditawarkan sudah ditambah dengan penunjukkan arah kiblat, namun tidak menunjukkan masjid yang berada disekitar Penggunga.

C. Sistem Pemandu Pencarian Masjid Terdekat Berbasis Lokasi di atas Platform Android

Sumber : Fatimah Aljufri

Pada studi literatur selanjutnya, fitur yang terdapat pada aplikasi ini hanya menunjukkan masjid terdekat saja, tanpa memberitahukan waktu shalat kepada pengguna dan hanya dapat digunakan di daerah Kotamadya Yogyakarta saja.

D. Augmented Reality on Android Operating System-Based Device; Case Study: Mosque Finder

Sumber : Ary Mazharuddin S dan Diaz Hendrianto

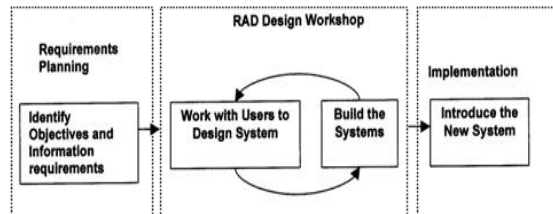
Fitur yang terdapat pada penelitian ini terbilang cukup lengkap sebagai pencari masjid, namun aplikasi ini tidak dapat memberikan rute menuju masjid yang dituju serta tidak menampilkan jadwal shalat.

3. Pembahasan

A. Metode Pengembangan Sistem

Dalam mengembangkan aplikasi ini peneliti menggunakan pendekatan metode pengembangan sistem *Rapid Application Development* (RAD) menurut teori Kendall & Kendall. Sedangkan tools yang digunakan adalah notasi *Unified Modelling Language* (UML) yang merupakan pemodelan berorientasi objek menurut Jefery L. Whitten adalah suatu pendekatan berorientasi objek

terhadap pengembangan sistem yang mencakup suatu metode pengembangan serta perangkat-perangkat lunak. Pengembangan sistem RAD memiliki tiga fase yakni penilaian, perancangan, dan penerapan yang melibatkan penganalisis dan pengguna. Gambar di bawah ini menggambarkan ketiga fase ini.



Gambar 3.1. Fase Rapid Application Development

1. Fase Perancangan Kebutuhan

Pada fase ini, penganalisis mengidentifikasi tujuan – tujuan aplikasi atau sistem untuk mengidentifikasi kebutuhan – kebutuhan data serta sistem yang dibutuhkan untuk membangun aplikasi tersebut.

Pada fase perancangan kebutuhan, yang dilakukan antara lain:

- a. Melakukan pembelajaran pada literatur yang membahas tentang algoritma meeus.
- b. Menganalisa sistem yang akan dibangun.
- c. Melakukan identifikasi masalah yang terjadi.

2. Fase Workshop Desain RAD

Fase ini adalah fase untuk merancang sistem yang akan dibuat atau memperbaiki (jika ada) sistem yang telah dibuat. Adapun fase ini terdiri dari dua hal, yakni *design system* dan *build system*.

Fase ini dicirikan dengan *workshop* karena layaknya sebuah *workshop* yang terdapat para partisipan yang berpartisipasi kuat dalam kelompok dan tidak terdapat aktifitas yang pasif. Partisipan tersebut dianalogikan sebagai kelompok-kelompok kecil (*Group Decision Support System*) yang dibentuk untuk membantu pengguna dalam menyetujui desain. Selama *workshop* desain RAD ini, pengguna merespon *working prototype* yang ada, menganalisa dan memperbaiki modul-modul yang dirancang menggunakan perangkat lunak berdasarkan respon pengguna.

a. Design System

- Desain Aplikasi

Pada perancangan aplikasi ini, didesain menggunakan *Unified Modeling Language (UML)*. Hal ini dilakukan karena UML dapat memudahkan dalam pengembangan sistem. Selain itu, UML juga lebih cocok digunakan dalam perancangan aplikasi *object oriented*.

UML (*Unified Modelling Language*) pertama kali diperkenalkan pada tahun 1990-an ketika Grady Booch, Ivar Jacobson dan James Rumbaugh mulai mengadopsi ide-ide serta kemampuan-kemampuan tambahan dari masing-

masing metodenya dan berusaha membuat metodologi terpadu yang kemudian dinamakan UML (*Unified Modelling Language*) (Nugroho, 2005:20).

Secara umum UML merupakan 'bahasa' untuk visualisasi, spesifikasi, konstruksi dan dokumentasi. Secara khusus, UML menspesifikasikan langkah-langkah penting dalam pengambilan keputusan analisis, perancangan, serta implementasi dalam sistem yang sangat bernuansa perangkat lunak (Nugroho, 2005:21).

Pendekatan UML memiliki nilai yang sangat baik dalam penyelidikan dan penelitian. Perangkat UML distandarkan sebagai peralatan untuk dokumen analisa dan perancangan dari sistem perangkat lunak. Peralatan UML termasuk diagram yang memberikan seseorang untuk menampilkan konstruksi dari sebuah sistem *object oriented*.

UML sendiri juga memberikan standar penulisan sebuah sistem *blue print*, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema *database*, dan komponen-komponen yang diperlukan dalam sistem *software*.

b. Diagram UML

UML menyediakan beberapa jenis diagram, diantaranya yang digunakan dalam penelitian ini adalah *Use Case Diagram*, *Sequence Diagram*, *Class Diagram*, dan *State Diagram*.

- Use Case Diagram.

Use Case diagram memperlihatkan himpunan *use case* dan aktor-aktor. Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku dari suatu sistem yang dibutuhkan serta diharapkan pengguna (Nugroho, 2005:19).

Di dalam *use case* terdapat teks untuk menjelaskan urutan kegiatan yang di sebut *use case specification*. *Use case specification* terdiri dari:

- Nama *Use Case*
Mencantumkan nama dari *use case* yang bersangkutan. Sebaiknya diawali dengan kata kerja untuk menunjukan suatu aktivitas.
- Deskripsi singkat
Menjelaskan secara singkat dalam satu atau dua kalimat tentang tujuan dari *use case* ini.
- Aliran normal
Ini adalah jantung dari *use case*. Menjelaskan interaksi antara *actor* dan sistem dalam kondisi normal, yaitu segala sesuatu berjalan dengan baik, tiada halangan atau hambatan dalam mencapai tujuan dari *use case*.
- Aliran alternatif
Merupakan perlengkapan dari *basic flow* karena tidak ada yang sempurna dalam setiap kali *use case* berlangsung. Di dalam *alternate flow* ini dijelaskan apa yang akan terjadi bila suatu halangan terjadi sewaktu *use case* berlangsung.

- Pre-condition
Menjelaskan persyaratan yang harus di penuhi sebelum *use case* bisa di mulai.
- Post-condition
Menjelaskan kondisi yang berubah atau terjadi saat *use case* selesai di eksekusi.

Activity Diagram

Activity diagram menggambarkan aliran fungsionalitas sistem. Pada tahap pemodelan bisnis, diagram aktivitas dapat digunakan untuk menunjukkan aliran kerja bisnis (*business workflow*). Dapat juga digunakan untuk menggambarkan aliran kejadian (*flow of event*) dalam *use case*. (Sholih, 2006 : 8).

Sequence Diagram

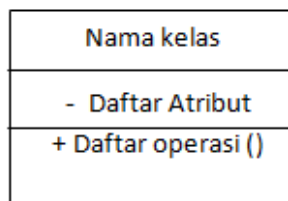
Sequence Diagram adalah diagram interaksi yang menekankan pada pengiriman pesan (*message*) dalam suatu waktu tertentu (Nugroho, 2005:19).

Kita dapat membaca diagram ini dengan melihat pada objek-objek dan pesan-pesan (*message*). Objek-objek yang berperan dalam aliran diperlihatkan pada kotak bersegi empat panjang yang melintas pada bagian atas diagram. Setiap objek memiliki garis hidup (*lifeline*), yang digambarkan sebagai garis vertikal di bawah nama suatu objek (Nugroho, 2005:92).

Class Diagram

Class Diagram mendeskripsikan jenis-jenis objek dalam sistem dan berbagai macam hubungan statis yang terdapat diantara mereka. *Class Diagram* juga menunjukkan properti dan operasi sebuah *class* dan batasan-batasan yang terdapat dalam hubungan-hubungan objek tersebut (Fowler, 2005:53).

Class didefinisikan sebagai kumpulan/himpunan objek dengan atribut/yang mirip, perilaku (operasi) yang mirip, serta hubungan dengan objek yang lain dengan cara yang mirip. (Nugroho, 2005 : 39).



Gambar 3.2. Notasi Class

Bagian paling atas pada notasi kelas digunakan sebagai nama kelas, dan secara opsional juga dapat disertakan stereotype-nya. Bagian tengah digunakan untuk mendeklarasikan atribut, dan bagian paling bawah digunakan mendeklarasikan operasi. (Sholih, 2006 : 103).

Class diagram umumnya tersusun dari elemen *class*, *interface*, *dependency*, *Generalization* dan *Association*. Relasi *dependency* menunjukkan bagaimana terjadi ketergantungan antar *class* yang ada. Relasi *Generalization* menunjukkan bagaimana suatu *class* menjadi *superclass* dari *class* lainnya dan *class* tersebut menjadi *subclass* dari *class* tersebut. Relasi *Association* menggambarkan navigasi antar *class*, berapa banyak obyek lain bisa berhubungan dengan satu obyek (*multiplicity* antar *class*), dan apakah satu *class* menjadi bagian dari *class* lainnya (*agregation*) (Hermawan, 2004:28).

Build System

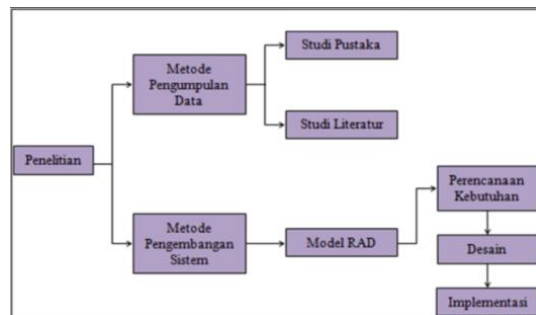
Pada tahap *build system* ini merupakan *output* dari *design system* yaitu pengembangan aplikasi yang telah dirancang sebelumnya. Pada fase ini membuat *pengguna interface* dan pengkodean program. Perancangan yang dilakukan meliputi halaman-halaman yang ada pada aplikasi.

3. Fase Implementasi

Setelah aplikasi selesai dibangun, dilakukan implementasi kepada pengguna untuk memperoleh hasil yang diharapkan, maka dilakukanlah uji coba atau *system testing*.

Pada tahap ini jika sistem dikembangkan belum sesuai dengan yang diharapkan, maka peneliti melakukan *revisi* terhadap aplikasi. Pada penelitian ini dilakukan pengujian aplikasi dengan metode *blackbox* secara mandiri.

B. Alur Berfikir



Gambar 3.3. Alur berfikir

4. Hasil

Seperti yang telah dibahas pada bab sebelumnya mengenai metode ini, peneliti memulai penelitian dengan melakukan pengumpulan data untuk menjadikan parameter awal penelitian.

A. Requirement Planning

1. Algoritma Meeus

Jean Meeus adalah astronom dan matematikawan kelahiran Belgia tahun 1928. Dia belajar matematika di Universitas Leuven, Belgia, dan lulus tahun 1953. Dia tertarik pada astronomi bola dan mekanika benda langit. Jean Meeus

menulis banyak buku matematika astronomi, seperti *Canon of Solar Eclipses, Elements of Solar Eclipses 1951-2200, Canon of Lunar Eclipses, Astronomical Formulae for Calculators, Astronomical Algorithms, Transits, Astronomical Tables of the Sun, Moon and Planets, Mathematical Astronomy Morsels. Atas jasanya dalam bidang astronomi, sebuah asteroid yang ditemukan diberi nama asteroid 2213 Meeus.*

B. Desain Workshop

1. Desain Sistem

Pada tahap ini, penulis akan merancang sistem pengingat waktu sholat dan penunjuk masjid terdekat. Adapun rancangannya meliputi rancangan identifikasi aktor, perancangan *use case* diagram, perancangan *activity* diagram, perancangan *sequence* diagram, perancangan *class* diagram, dan perancangan *user interface*.

- **Penentuan Aktor**

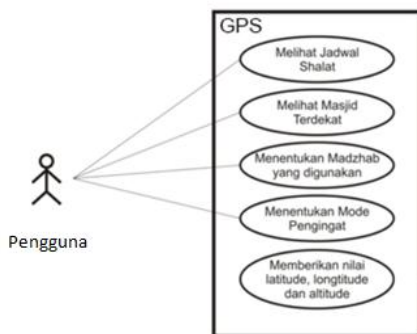
Pada tahap ini, penulis mendapatkan dua aktor yang terlibat pada aplikasi ini, yakni *user* dan *GPS*. *User* adalah aktor yang menggunakan *handphone* berbasis *Smartphone Android*, dimana *user* akan mendapatkan output dari sistem berupa pemberitahuan waktu sholat dan masjid yang jaraknya terdekat dari posisi *user*. Sedangkan *GPS* memiliki kewajiban untuk memberikan nilai koordinat *latitude*, *longtitude* dan *altitude* posisi *handphone* berada.

Tabel 4.1. Penentuan Actor

No	Aktor	Keterangan
1	User	User adalah aktor yang menggunakan <i>handphone</i> berbasis <i>smartphone android</i> . User akan mendapatkan output dari sistem berupa pemberitahuan waktu sholat dan masjid yang jaraknya terdekat dari posisi <i>user</i>

- **Use Case Diagram**

Use Case merupakan gambaran skenario dari interaksi antara user dengan sistem. Sebuah diagram *Use Case* menggambarkan hubungan antara user dan kegiatan yang dapat dilakukan terhadap aplikasi



Gambar 4.1. Diagram *Use Case*

Pada diagram diatas terdiri dari 1 aktor dan 5 *Use Case*. Di dalam diagram ini terdapat *extend* yang digunakan untuk menunjukkan bahwa satu *Use Case* merupakan tambahan fungsional dari *Use Case* lain.

Adapun *Use Case* sistem pada pengingat waktu sholat dan penunjuk masjid terdekat ini diantaranya :

1. Melihat Jadwal Sholat, merupakan *Use Case* yang menggambarkan bahwa *user* akan mendapatkan kemampuan untuk melihat jadwal sholat.
1. Melihat Masjid Terdekat, merupakan *Use Case* yang menggambarkan bahwa *user* akan mendapatkan kemampuan untuk mengetahui masjid terdekat dilokasi sekitar *handphone* berada.
2. Menentukan Mazhab yang digunakan, merupakan *Use Case* yang menggambarkan bahwa *user* diberikan keleluasaan untuk menentukan mazhab yang akan dipakainya dalam metode perhitungan waktu shalat.
3. Menentukan Mode Pengingat, merupakan *Use Case* yang menggambarkan bahwa *user* dapat mengatur mode pengingat waktu shalat, apakah dengan suara adzan atau hanya getar saja.
4. Memberikan nilai latitude, longtitude dan altitude, merupakan *Use Case* yang menggambarkan bahwa *GPS* memberikan nilai koordinat latitude, longtitude dan altitude posisi *handphone* berada.

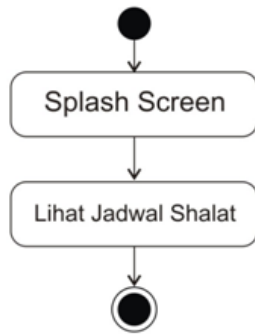
- **Test Case**

- **Test Case Melihat Jadwal Shalat.**
User berhasil melihat jadwal sholat.
- **Test Case Melihat Masjid Terdekat.**
User berhasil melihat masjid terdekat sesuai dengan posisi *handphone user*.
- **Test Case Menentukan Mazhab yang digunakan.**
User berhasil menentukan mazhab yang digunakan sesuai dengan keinginan *user*.
- **Test Case Menentukan Mode Pengingat.**
User berhasil memilih mode pengingat yang diinginkan, baik itu berupa suara adzan maupun hanya getar saja.
- **Test Case Memberikan Nilai Latitude, Longtitude dan Altitude.**
GPS berhasil memberikan nilai koordinat latitude, longtitude dan altitude sesuai dengan posisi *handphone*.

- **Activity Diagram**

Activity Diagram adalah teknik untuk mendeskripsikan logika procedural, proses bisnis, dan aliran kerj dalam banyak kasus. Adapun *Activity Diagram* pada program ini adalah:

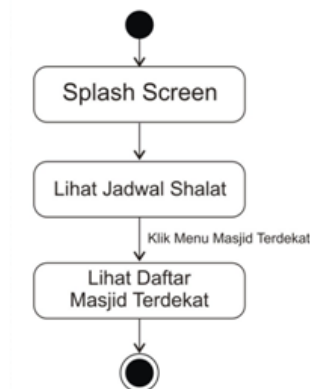
- **Activity Diagram Lihat jadwal shalat**



Gambar 4.2. *Activity Diagram* Lihat Jadwal Shalat

Activity diagram diatas menggambarkan bagaimana ketika aplikasi pertama kali dijalankan akan membawa *user* pada tampilan pembuka program Muaidzah ini selama 3 detik, lalu *user* akan langsung dihadapkan kepada jadwal waktu shalat.

- **Activity Diagram Lihat daftar masjid terdekat**

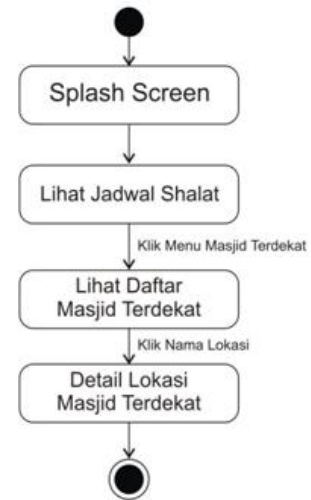


Gambar 4.3. *Activity Diagram* Lihat Daftar Masjid Terdekat

Activity diagram diatas menggambarkan ketika aplikasi pertama kali dijalankan akan membawa *user* pada tampilan pembuka program Muaidzah ini selama 3 detik, lalu *user* akan langsung dihadapkan kepada jadwal waktu shalat. Lalu *user* mengklik menu masjid terdekat, maka

akan ditampilkan daftar-daftar masjid terdekat dari posisi *user*.

- **Activity Diagram Detail lokasi masjid terdekat**



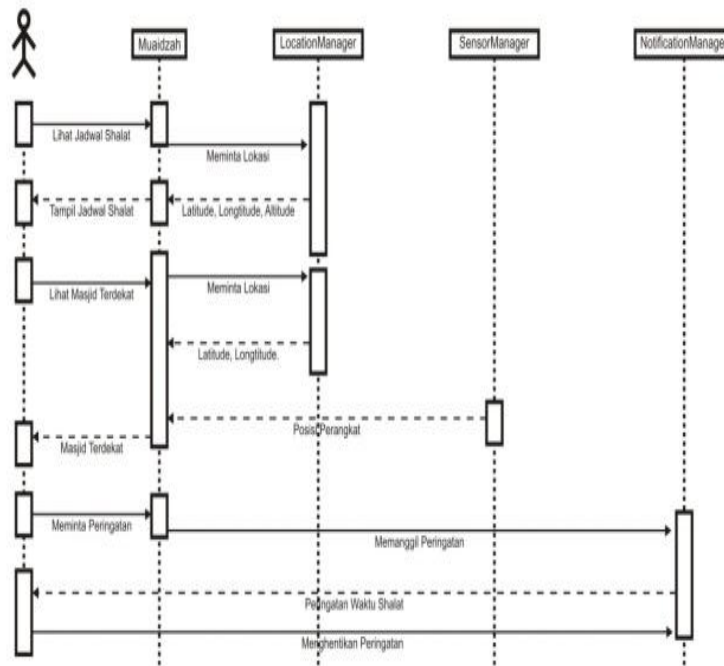
Gambar 4.4. *Activity Diagram* Detail Lokasi Masjid Terdekat

Activity diagram diatas menggambarkan bagaimana ketika aplikasi pertama kali dijalankan akan membawa *user* pada tampilan pembuka program Muaidzah ini selama 3 detik, lalu *user* akan langsung dihadapkan kepada jadwal waktu shalat. Lalu *user* mengklik menu masjid terdekat, maka akan ditampilkan daftar-daftar masjid terdekat dari posisi *user*.

Dengan mengklik nama lokasi yang dikehendaki, *user* akan dihadapkan dengan tampilan googlemap yang akan menunjukkan rute menuju masjid yang dituju.

- **Sequence Diagram**

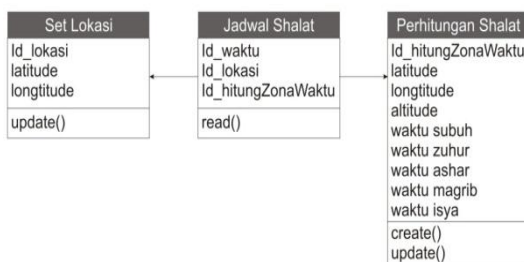
Diagram ini menggambarkan interaksi antar objek didalam dan disekitar sistem (termasuk pengguna, display dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence Diagram* terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek-objek terkait).



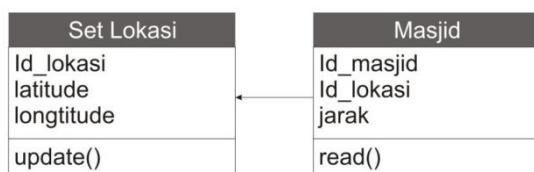
Gambar 4.5. Sequence Diagram

Class Diagram

Diagram ini digunakan untuk menggambarkan kumpulan dari class dan hubungannya. Diagram ini merupakan diagram yang paling umum ditemukan dalam pemodelan sistem berorientasi objek. Class menggambarkan keadaan suatu sistem, sekaligus layanan untuk memanipulasi keadaan metode atau fungsi sehingga class memiliki tiga area pokok, yaitu: nama, atribut, dan metode. Selain itu setiap class yang dapat menjadi sebuah form saat pembuatan program. *Class diagram* yang diusulkan dapat dilihat pada gambar berikut:



Gambar 4.6. Class Diagram Jadwal Shalat



Gambar 4.6. Class Diagram Masjid Terdekat

2. Build System

[1] Perancangan Antarmuka (Interface)

Perancangan *Interface* adalah bagian yang penting pada aplikasi, karena yang pertama kali dilihat dan berinteraksi dengan pengguna adalah tampilan antar muka (*interface*) aplikasi.

[2] Perancangan Algoritma Meeus

Pada tahap ini dilaksanakan implementasi perhitungan waktu shalat menurut algoritma Meeus yang digunakan oleh penulis. Adapun perhitungan dari algoritma Meeus itu sendiri akan dijabarkan dalam bentuk contoh kasus sebagai berikut :

Menentukan waktu shalat pada tanggal 20 Maret 2014 di Jakarta (L = -6,166667 derajat, B = 106,85 derajat, Z = 7, H = 50 meter). Sudut Subuh = 20 derajat. Sudut Isya' = 18 derajat. Ashar menggunakan madzhab Syafi'i (KA = 1).

Langkah – langkah :

1. Koordinat lintang daerah tersebut (L) = - 6,166667 derajat
2. Koordinat bujur daerah tersebut (B) = 106,85 derajat
3. Zona waktu daerah tersebut (Z) = 7
4. Ketinggian lokasi dari permukaan laut (H) = 50 meter
5. Julian Day untuk 20 Maret 2014 pukul 12.00 UT. Dari tanggal tersebut diperoleh nilai D = 20, M = 3, Y = 2014.
 - a. $A = INT(Y/100)$
 $A = INT(2014/100)$
 $A = 20$
 - b. $B (JD) = 2 + INT(A/4) - A$

$$B(JD) = 2 + \text{INT}(20/4) - 20$$

$$\underline{B(JD) = -13}$$

$$c. JD = 1720994,5 + \text{INT}(365,25*Y) + \text{INT}(30,6001(M + 1)) + B + D$$

$$JD = 1720994,5 + \text{INT}(365,25*2014) + \text{INT}(30,6001(3 + 1)) + (-13) + 20$$

$$\underline{JD = 2456729,0}$$

d. Convert menjadi JD lokal

$$JD \text{ lokal} = JD - Z/24$$

$$JD \text{ lokal} = 2456729,0 - 7/24$$

$$\underline{JD \text{ lokal} = 2456728,708}$$

Dari nilai JD tersebut, maka dapat dihitung sudut tanggal (T) untuk menghitung Delta dengan rumus :

$$T = 2 * \text{PI} * (JD - 2451545) / 365,25$$

$$T = 2 * 3,14159265359 * (2456728,708 - 2451545) / 365,25$$

$$\underline{T = 89,17234781}$$

6. Sudut deklinasi matahari (Delta), dapat dihitung dengan rumus :

$$\Delta = 0,37877 + 23,264 * \text{SIN}(57,297 * T - 79,547) + 0,3812 * \text{SIN}(2 * 57,297 * T - 82,682) + 0,17132 * \text{SIN}(3 * 57,297 * T - 59,722)$$

$$\Delta = 0,37877 + 23,264 * \text{SIN}(57,297 * 89,17234781 - 79,547) + 0,3812 * \text{SIN}(2 * 57,297 * 89,17234781 - 82,682) + 0,17132 * \text{SIN}(3 * 57,297 * 89,17234781 - 59,722)$$

$$\underline{\Delta = -0,190491501 \text{ derajat}}$$

7. *Equation of Time* (ET), untuk dapat menghitung ET, maka pertama kali yang harus dihitung adalah bujur rata-rata matahari (L0) dengan rumus :

$$L0 = 280,46607 + 36000,7698 * U$$

Dimana U:

$$U = (JD - 2451545) / 36525$$

$$U = (JD - 2451545) / 36525$$

$$U = (2456728,708 - 2451545) / 36525$$

$$\underline{U = 0,142141227}$$

Maka dapat dihitung

$$L0 = 280,46607 + 36000,7698 * U$$

$$L0 = 280,46607 + 36000,7698 * 0,142141227$$

$$\underline{L0 = 94,2069333}$$

Selanjutnya, *Equation of Time* dapat dihitung sebagai berikut :

$$1000 * ET = -(1789 + 237 * U) * \text{SIN}(L0) - (7146 - 62 * U) * \text{COS}(L0) + (9934 - 14 * U) * \text{SIN}(2 * L0) - (29 + 5 * U) * \text{COS}(2 * L0) + (74 + 10 * U) * \text{SIN}(3 * L0) + (320 - 4 * U) * \text{COS}(3 * L0) - 212 * \text{SIN}(4 * L0)$$

$$1000 * ET = -(1789 + 237 * 0,142141227) * \text{SIN}(94,2069333) - (7146 - 62 * 0,142141227) * \text{COS}(94,2069333) + (9934 - 14 * 0,142141227) * \text{SIN}(2 * 94,2069333) - (29 + 5 * 0,142141227) * \text{COS}(2 * 94,2069333) + (74 + 10 * 0,142141227) * \text{SIN}(3 * 94,2069333) + (320 - 4 * 0,142141227) * \text{COS}(3 * 94,2069333) - 212 * \text{SIN}(4 * 94,2069333)$$

$$1000 * ET = -7555$$

$$ET = -7555 / 1000$$

$$\underline{ET = -7,555}$$

8. Altitude matahari waktu subuh = 20 derajat.
Altitude matahari waktu Isya' = 18 derajat
9. Tetap panjang bayangan ashlar = 1 (Mazhab Syafi'i)
Dari data-data tersebut diatas, maka dapat waktu shalat sudah dapat dihitung

1. Waktu Zhuhur

$$\text{Zhuhur} = 12 + Z - B/15 - ET/60$$

$$\text{Zhuhur} = 12 + 7 - 106,85/15 - (-7,555/60)$$

$$\text{Zhuhur} = 12,0028$$

Kemudian nilai ini dikonversi ke jam : menit : detik

$$\text{Jam} = 12$$

$$\text{Menit} = 0,0028/1 * 60 = 0,168$$

$$\text{Menit} = 00$$

$$\text{Detik} = 0,168/1 * 60 = 10,08$$

$$\text{Detik} = 10$$

$$\text{Jam} : \text{menit} : \text{detik} = 12 : 00 : 10 \text{ WIB}$$

2. Waktu Ashar

$$\text{Ashar} = \text{Zhuhur} + (\text{Hour Angle Ashar})/15$$

$$\text{Ashar} = \text{Zhuhur} + \text{ACOS}(\text{COS}(\text{HA}))/15$$

$$\text{Ashar} = \text{Zhuhur} + \text{ACOS}(\text{SIN}(\text{Altitude}) - \text{SIN}(\text{Lintang}) * \text{SIN}(\Delta)) / \text{COS}(\text{Lintang}) * \text{COS}(\Delta) / 15$$

$$\text{Ashar} = \text{Zhuhur} + \text{ACOS}(\text{SIN}(\text{ARCCOT}(\text{KA} + \text{TAN}(\text{ABS}(\Delta - \text{Lintang})))) - \text{SIN}(\text{Lintang}) * \text{SIN}(\Delta) / \text{COS}(\text{Lintang}) * \text{COS}(\Delta)) / 15$$

$$\text{Ashar} = 12,0028 + \text{ACOS}(\text{SIN}(\text{ARCCOT}(1 + \text{TAN}(\text{ABS}((-0,107629) - (-0,003324704)))) - \text{SIN}(-0,003324704) * \text{SIN}(-0,107629) / \text{COS}(-0,003324704) * \text{COS}(-0,107629)) / 15$$

$$\text{Ashar} = 12,0028 + 47,60749529/15$$

$$\text{Ashar} = 15,17663$$

Kemudian nilai ini dikonversi ke jam : menit : detik

$$15,17663 = 15 : 10 : 35$$

3. Waktu Maghrib

$$\text{Maghrib} = \text{Zhuhur} + (\text{Hour Angle Maghrib})/15$$

$$\text{Maghrib} = \text{Zhuhur} + \text{ACOS}(\text{COS}(\text{HA}))/15$$

$$\text{Maghrib} = \text{Zhuhur} + \text{ACOS}(\text{SIN}(\text{Altitude}) - \text{SIN}(\text{Lintang}) * \text{SIN}(\text{Delta})) / [\text{COS}(\text{Lintang}) * \text{COS}(\text{Delta})]/15$$

$$\text{Maghrib} = \text{Zhuhur} + \text{ACOS}(\text{SIN}(0,8333 - 0,0347 * \text{SQRT}(H)) - \text{SIN}(-0,003324704) * \text{SIN}(-0,107629) / \text{COS}(-0,003324704) * \text{COS}(-0,107629))/15$$

$$\text{Maghrib} = 12,0028 + \text{ACOS}(\text{SIN}(0,8333 - 0,0347 * \text{SQRT}(50)) - \text{SIN}(-0,003324704) * \text{SIN}(-0,107629) / \text{COS}(-0,003324704) * \text{COS}(-0,107629))/15$$

$$\text{Maghrib} = 12,0028 + 91,09472347/15$$

$$\text{Maghrib} = 18,07578$$

Kemudian nilai ini dikonversi ke jam : menit : detik

$$18,07578 = 18 : 04 : 32$$

4. Waktu Isya

$$\text{Isya}' = \text{Zhuhur} + (\text{Hour Angle Isya}')/15$$

$$\text{Isya}' = \text{Zhuhur} + \text{ACOS}(\text{COS}(\text{HA}))/15$$

$$\text{Isya}' = \text{Zhuhur} + \text{ACOS}(\text{SIN}(\text{Altitude}) - \text{SIN}(\text{Lintang}) * \text{SIN}(\text{Delta})) / \text{COS}(\text{Lintang}) * \text{COS}(\text{Delta})/15$$

$$\text{Isya}' = 12,0028 + \text{ACOS}(\text{SIN}(-18) - \text{SIN}(-0,003324704) * \text{SIN}(-0,107629) / \text{COS}(-0,003324704) * \text{COS}(-0,107629))/15$$

$$\text{Isya}' = 12,0028 + 108,1165581/15$$

$$\text{Isya}' = 19,21057$$

Kemudian nilai ini dikonversi ke jam : menit : detik

$$19,21057 = 19 : 12 : 38$$

5. Waktu Subuh

$$\text{Subuh} = \text{Zhuhur} - (\text{Hour Angle Subuh})/15$$

$$\text{Subuh} = \text{Zhuhur} - \text{ACOS}(\text{COS}(\text{HA}))/15$$

$$\text{Subuh} = \text{Zhuhur} - \text{ACOS}(\text{SIN}(\text{Altitude}) - \text{SIN}(\text{Lintang}) * \text{SIN}(\text{Delta})) / \text{COS}(\text{Lintang}) * \text{COS}(\text{Delta})/15$$

$$\text{Subuh} = 12,0028 - \text{ACOS}(\text{SIN}(-20) - \text{SIN}(-0,003324704) * \text{SIN}(-0,107629) / \text{COS}(-0,003324704) * \text{COS}(-0,107629))/15$$

$$\text{Subuh} = 12,0028 - 110,157568/15$$

$$\text{Subuh} = 4,658962$$

Kemudian nilai ini dikonversi ke jam : menit : detik

$$4,658962 = 04 : 39 : 32$$

[b]. Perancangan Coding

Pada tahap ini dilaksanakan implementasi dari rancangan sistem yang dibuat. Bahasa yang digunakan pada pemrograman ini adalah bahasa Java. Untuk *editor* dan *unit test* digunakan Eclipse Galileo. Pada tahap *debugging* penulis menggunakan *sdk* yang telah diberikan.

Untuk menghitung waktu shalat, penulis menggunakan algoritma Meeus. Dengan perhitungan tersebut maka dapat ditentukan waktu shalat berdasarkan lokasi pengguna berada. Berikut potongan kode perhitungan waktu shalat dalam aplikasi ini.

C. *Implementation*

Agar aplikasi ini berjalan baik dan benar maka dibutuhkan perangkat yang mampu mendukung aplikasi ini baik dari segi perangkat lunak maupun perangkat keras. Untuk itu perlu diperhatikan kategori dari perangkat yang dapat menjalankan aplikasi ini.

1. Handphone dengan OS Android atau biasa disebut *Smartphone Android*.
2. *Smartphone Android* dengan minimal API 8.

5. Kesimpulan Dan Saran

A. Kesimpulan

Dari penelitian dan penulisan yang telah diuraikan, maka dapat ditarik kesimpulan bahwa : dengan memanfaatkan teknologi yang ada pada *Smartphone* khususnya yang berbasis sistem operasi Android, pengguna bisa dengan cepat mengetahui waktu shalat dan lokasi masjid yang berada paling dekat dengan posisi dia berada menggunakan sebuah aplikasi yang dirancang untuk dapat mengetahui waktu shalat dan masjid terdekat.

Oleh karena itu, dapat ditarik kesimpulan sebagai berikut :

- Penulis dapat menjelaskan keakurasian algoritma Meeus dalam penentuan waktu shalat.
- Aplikasi dapat mencari keberadaan masjid terdekat dengan memanfaatkan fasilitas GPS.

B. Saran

Aplikasi ini tentu saja masih jauh dari kata sempurna. Masih banyak hal yang dapat dilakukan untuk mengembangkan aplikasi ini agar menjadi lebih baik lagi, antara lain :

- Penulis mengharapkan aplikasi ini dapat dikembangkan lebih lanjut bukan hanya waktu shalat serta masjid terdekat, tetapi juga konten-konten islami lainnya agar dapat lebih bermanfaat.
- Diharapkan kedepannya dapat dikembangkan bukan hanya berjalan pada *platform Android* saja.

6. Referensi

- [1]. Amhar, Fahmi. 2002. Pengantar Memahami Astronomi Rukyat.
- [2]. Jogianto. 1999. Metodologi Penelitian Sistem Informasi. Yogyakarta: Andi.

- [3]. Kendall & Kendall. 2008. *System Analysis And Design*. London: Pearson International Edition 7th Edition.
- [4]. Meeus, Jean. 1991. *Astronomical Algorithm*, Willmann-Bell: Virginia.
- [5]. Mulyadi, Adi. 2010. *Membangun Aplikasi Android*. Yogyakarta: Multimedia Center Publishing.
- [6]. Riyanto. 2011. *Sistem Informasi Geografis Berbasis Mobile*. Yogyakarta: Gava Media.
- [7]. Safaat, Nazaruddin. 2012. *ANDROID Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*. Bandung: Informatika.
- <http://www.erasuslim.com/peradaban/ilmu-hisab/cara-menghitung-waktu-shalat.htm>
- <http://www.erasuslim.com/peradaban/ilmu-hisab/kalender-julian-kalender-gregorian-dan-julian-day.htm>