

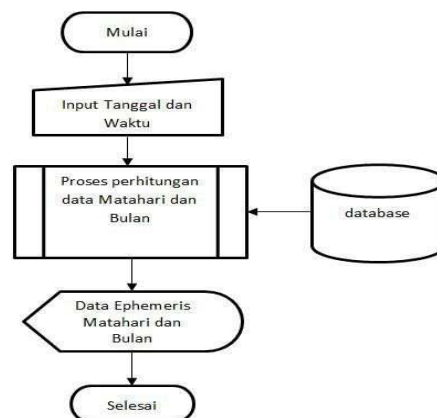
BAB III

DESAIN, RANCANGAN DAN IMPLEMENTASI PROGRAM

EPHEMERISAYA

Dalam skripsi ini penulis menyusun sebuah aplikasi perhitungan data *ephemeris* Matahari dan Bulan dengan nama *Ephemeris Hisab-Rukyah*, disingkat *EphemerisSaya*. Aplikasi *EphemerisSaya* merupakan aplikasi perhitungan data *ephemeris* Matahari dan Bulan untuk keperluan perhitungan ilmu falak metode hisab kontemporer sistem *ephemeris* menggunakan algoritma Jean Meeus sebagai basis perhitungannya. Bahasa pemrograman yang digunakan untuk menyusun aplikasi ini adalah *PHP*, sedangkan sistem *database* yang digunakan adalah *MySQL*.

Antarmuka aplikasi *EphemerisSaya* dibuat dengan konsep *user friendly*. Pengguna hanya perlu meng-*input* data tanggal dan waktu untuk melakukan perhitungan data *ephemeris*. Adapun diagram alir rancangan proses kerja aplikasi *EphemerisSaya* secara umum adalah sebagai berikut:



Gambar 3.1 *Flowchart* Aplikasi *EphemerisSaya* Secara Umum

Dari gambar tersebut dapat dilihat alur proses pengoperasian aplikasi *Ephemeris* secara umum. Pertama, pengguna memasukkan *input* data tanggal dan waktu. Selanjutnya data akan diproses menggunakan algoritma Jean Meeus yang telah disusun menggunakan bahasa pemrograman *PHP*, di mana beberapa perhitungannya memerlukan data-data yang telah tersimpan di dalam *MySQL database*. Setelah itu hasil perhitungan berupa data *ephemeris* Matahari dan Bulan akan di-*deploy* melalui *browser* di *gadget* pengguna.

A. Algoritma Perhitungan Data *Ephemeris* Matahari dan Bulan Metode Jean Meeus

Aplikasi ini menggunakan algoritma Jean Meeus dalam proses perhitungannya, sebagaimana dijelaskan sebelumnya. Adapun alur proses perhitungan algoritma secara lengkap adalah sebagai berikut:

1. Perhitungan *Julian Day*, Selisih antara *Universal Time* dan *Dynamical Time* (ΔT), *Julian Day Ephemeris*, *Julian Centuries* (T) dan *Julian Millenia* (\mathcal{T})

a. Perhitungan *Julian Day*

Input data yang diperlukan untuk melakukan perhitungan data *ephemeris* yakni data tanggal dan waktu. Data tanggal berisi tanggal, bulan dan tahun, data waktu berisi jam, menit dan detik. Selanjutnya, data tanggal terlebih dahulu diubah ke dalam sistem Julian, sedangkan data waktu diubah ke dalam UT atau GMT (misal 08.00 WIB = 01.00 GMT).

Berikut ini algoritma untuk mengubah tanggal dan tahun ke sistem *Julian Day* (JD):¹

Jika bulan > 2, maka M = bulan, dan Y = tahun

Jika bulan = 1 atau 2, maka M = bulan+12, dan Y = tahun-1

A = INT (Y/100)

B = 2-A+INT(A/4)

JD = INT (365,25 * (Y + 4716)) + INT (30,6001 * (M+1)) + B
+ tanggal + (jam + menit/60 + detik/3600) /24 - 1524,5

b. Perhitungan ΔT

Langkah selanjutnya adalah menghitung ΔT . Terdapat banyak rumus yang diperkenalkan oleh Jean Meeus untuk mencari ΔT . Sebagaimana diterangkan pada bab sebelumnya, nilai ΔT dari tahun ke tahun senantiasa berubah, sehingga beberapa rumus perhitungan ΔT yang terdapat pada buku *Astronomical Algorithms* milik Jean Meeus, kadang menghasilkan data *error* untuk perhitungan ΔT pada tahun tertentu.² Adapun rumus yang terbaru yang Meeus perkenalkan melalui *website* NASA yakni rumus logika *polynomial expression for ΔT* .³

Rumus *polynomial expression for ΔT* merupakan rumus yang disusun oleh Jean Meeus dan Fred Espenak. Rumus ini disusun untuk mengatasi masalah *error* hasil perhitungan pada beberapa rumus-rumus

¹ Jean Meeus, *Astronomical Algorithms*, First Edition, Virginia: Willman-Bell, Inc., 1991, hlm. 61

² *Ibid.* hlm. 71-75

³ <http://eclipse.gsfc.nasa.gov/SEcat5/deltatpoly.html>, diakses pada tanggal 9 Januari 2013 pukul 11.55 WIB.

sebelum. Adapun cangkupan perhitungan dari rumus ini adalah dari tahun -1999 s/d +3000, atau sekitar lima millenium.⁴ Berikut ini adalah logika perhitungan *polynomial expression for ΔT* :⁵

Jika tahun ≤ -500 , maka:

$$\Delta T = -20 + 32 * u^2$$

$$\text{Di mana } u = (y-1820)/100$$

Jika tahun diantara -500 s/d +500 maka:

$$\Delta T = 10583.6 - 1014.41 * u + 33.78311 * u^2 - 5.952053 * u^3 - 0.1798452 * u^4 + 0.022174192 * u^5 + 0.0090316521 * u^6$$

$$\text{Di mana } u = y/100$$

Jika tahun diantara +500 s/d +1600, maka:

$$\Delta T = 1574.2 - 556.01 * u + 71.23472 * u^2 + 0.319781 * u^3 - 0.8503463 * u^4 - 0.005050998 * u^5 + 0.0083572073 * u^6$$

$$\text{Di mana } u = (y-1000)/100$$

Jika tahun diantara +1600 s/d +1700, maka:

$$\Delta T = 120 - 0.9808 * t - 0.01532 * t^2 + t^3 / 7129$$

$$\text{Di mana } t = y - 1600$$

Jika tahun diantara +1700 s/d +1800, maka:

$$\Delta T = 8.83 + 0.1603 * t - 0.0059285 * t^2 + 0.00013336 * t^3 - t^4 / 1174000$$

$$\text{Di mana } t = y - 1700$$

Jika tahun diantara +1800 s/d +1860, maka:

⁴ *Ibid.*

⁵ *Ibid.*

$$\Delta T = 13.72 - 0.332447 * t + 0.0068612 * t^2 + 0.0041116 * t^3 - 0.00037436 * t^4 + 0.0000121272 * t^5 - 0.0000001699 * t^6 + 0.00000000875 * t^7$$

Di mana $t = y - 1800$

Jika tahun diantara 1860 s/d 1900, maka:

$$\Delta T = 7.62 + 0.5737 * t - 0.251754 * t^2 + 0.01680668 * t^3 - 0.0004473624 * t^4 + t^5 / 233174$$

Di mana $t = y - 1860$

Jika tahun di antara 1900 s/d 1920, maka:

$$\Delta T = -2.79 + 1.494119 * t - 0.0598939 * t^2 + 0.0061966 * t^3 - 0.000197 * t^4$$

maka $t = y - 1900$

jika tahun di antara 1920 s/d 1941, maka:

$$\Delta T = 21.20 + 0.84493 * t - 0.076100 * t^2 + 0.0020936 * t^3$$

Di mana $t = y - 1920$

Jika tahun di antara 1941 s/d 1961, maka:

$$\Delta T = 29.07 + 0.407 * t - t^2 / 233 + t^3 / 2547$$

Di mana $t = y - 1950$

Jika tahun di antara 1961 s/d 1986, maka:

$$\Delta T = 45.45 + 1.067 * t - t^2 / 260 - t^3 / 718$$

Di mana $t = y - 1975$

Jika tahun di antara 1986 s/d 2005, maka:

$$\Delta T = 63.86 + 0.3345 * t - 0.060374 * t^2 + 0.0017275 * t^3 + 0.000651814 * t^4 + 0.00002373599 * t^5$$

Di mana $t = y - 2000$

Jika tahun di antara 2005 s/d 2050, maka:

$$\Delta T = 62.92 + 0.32217 * t + 0.005589 * t^2$$

Di mana $t = y - 2000$

Jika tahun diantara 2050 s/d 2150, maka:

$$\Delta T = -20 + 32 * ((y-1820)/100)^2 - 0.5628 * (2150 - y)$$

Jika tahun ≥ 2150 , maka:

$$\Delta T = -20 + 32 * u^2$$

Di mana $u = (y-1820)/100$

c. Menghitung *Julian Day Ephemeris*, *Julian Centuries* (T) dan *Julian Millenia*(\mathcal{T})⁶

Langkah selanjutnya adalah menghitung JDE, T dan \mathcal{T} , berikut rumus yang dipergunakan:

$$JDE = JD + \Delta T$$

$$T (TD) = (JDE - 2452545) / 36525$$

$$\mathcal{T} = T / 10$$

2. Perhitungan Data *Ephemeris* Matahari

a. Koreksi Posisi Planet Bumi

Untuk mengetahui data *ephemeris* Matahari, sebelumnya dilakukan perhitungan posisi orbit planet Bumi dari Matahari. Terdapat 3 data posisi untuk masing-masing planet, yakni: L untuk bujur heliosentris planet, B untuk lintang heliosentris planet dan R untuk jarak planet-Matahari.⁷

⁶ Jean Meeus, *loc.cit.*

⁷ *Ibid.* hlm. 205

Untuk perhitungan data posisi planet Bumi terdapat 6 kelompok koreksi untuk data L Bumi, 2 kelompok koreksi untuk data B Bumi dan 5 kelompok koreksi untuk data R Bumi. Data tabel koreksi *Periodic Terms* untuk planet Bumi dapat dilihat pada lampiran I. Adapun untuk menggunakan data koreksi pada tabel tersebut menggunakan rumus:

$$A * \cos(B + C * T)$$

Koreksi pertama yakni koreksi bujur heliosentris Bumi (L), terdapat 6 kelompok koreksi pada tahap ini, yakni L0, L1, L2, L3, L4 dan L5. Di mana pada masing-masing kelompok koreksi terdapat suku-suku koreksi dengan jumlah yang berbeda-beda. Masing-masing suku koreksi tersebut dihitung menggunakan rumus yang telah disebutkan di atas. Selanjutnya suku-suku koreksi tersebut dijumlahkan pada masing-masing kelompok koreksi.

Terakhir, untuk mendapatkan bujur heliosentris Bumi digunakan rumus:

$$L = (L_0 + L_1 * T + L_2 * T^2 + L_3 * T^3 + L_4 * T^4 + L_5 * T^5) / 10^8$$

L merupakan bujur ekliptika Bumi diukur dari Matahari (heliosentris). Adapun untuk mengetahui bujur ekliptika Matahari dari Bumi (geosentris) menggunakan rumus berikut:

$$\Theta = L + 180, \text{ hasil dalam derajat.}$$

Setelah itu, menghitung koreksi untuk Θ dengan rumus:

$$\Theta^t = \Theta - 0,09033'', \text{ hasil dalam derajat.}$$

Hasilnya merupakan *true geometric longitude* matahari.

Koreksi kedua adalah koreksi lintang heliosentris Matahari (B), terdapat 2 kelompok koreksi pada tahap ini, yakni B0 dan B1. Proses koreksinya sama dengan koreksi L. Selanjutnya untuk mengetahui B menggunakan rumus berikut:

$$B = ((B0 + B1 * T) / 10^8) * 1, \text{ hasil dalam bentuk radian, kemudian} \\ \text{dikonversi ke dalam detik busur.}$$

Setelah itu menghitung koreksi B dengan rumus:

$$\lambda' = \Theta - 1,397 * T - 0,00031 * T^2$$

$$\Delta B = 0,03916 * (\text{COS } \lambda' - \text{SIN } \lambda'), \text{ di mana untuk pemrograman} \\ \text{PHP, } \lambda' \text{ dirubah ke dalam bentuk radian terlebih dahulu.}$$

Selanjutnya untuk mengetahui *apparent latitude* Matahari menggunakan rumus:

$$\beta = B + \Delta B$$

Koreksi ketiga adalah koreksi jarak Matahari-Bumi (R). Terdapat 5 kelompok koreksi. Adapun proses koreksinya sama dengan koreksi L dan B. Selanjutnya untuk mengetahui *true geocentric distance* menggunakan rumus:

$$R = (R0 + R1 * T + R2 * T^2 + R3 * T^3 + R4 * T^4) / 10^8$$

b. Menghitung *Nutasi* dan *True Obliquity*

Langkah pertama yakni menghitung *mean obliquity* dengan rumus:

$$E_o = 23^\circ 26' 21,448'' + (-4680,93 * U - 1,55 * U^2 + 1999,25 * U^3 - 51,38 * U^4 - 249,67 * U^5 - 39,05 * U^6 + 7,12 * U^7 + 27,87 * U^8 + 5,79 * U^9 + 2,45 * U^{10})/3600, \text{ hasil dalam derajat.}$$

Di mana $U = T/100$

Untuk mengetahui *true obliquity* dan *nutasi*, diperlukan perhitungan koreksi $\Delta\epsilon$ dan $\Delta\psi$ menggunakan tabel *terms of the 1980 IAU Theory of Nutations*, sebagaimana terdapat pada lampiran II dan III. Namun sebelum menghitung $\Delta\psi$ dan $\Delta\epsilon$ dengan tabel tersebut, diperlukan perhitungan *multiple arguments* terlebih dahulu. Berikut rumus perhitungan *multiple arguments* untuk $\Delta\psi$ dan $\Delta\epsilon$:

$$D = 297,85036 + 445267,11148 * T - 0,0019142 * T^2 + T^3/189474, \text{ hasil dalam derajat.}$$

$$M_o = 357,52772 + 35999,05034 * T - 0,0001603 * T^2 - T^3/300000, \text{ hasil dalam derajat.}$$

$$M^c = 134,96298 + 477198,867398 * T + 0,0086972 * T^2 + T^3/56250, \text{ hasil dalam derajat.}$$

$$F = 93,27191 + 483202,017538 * T - 0,0036825 * T^2 + T^3/327270, \text{ hasil dalam derajat.}$$

$$\Omega^c = 125,04452 - 1934,136261 * T + 0,0020708 * T^2 + T^3/450000, \text{ hasil dalam derajat.}$$

Langkah selanjutnya adalah menghitung koreksi $\Delta\psi$ pada tabel yang terdapat pada lampiran II menggunakan rumus:

$$(\text{Coefficient 1} + \text{Coefficient 2} * T) * \sin(\text{Multiple Arguments})$$

Setelah itu seluruh koreksi dijumlahkan, kemudian digunakan untuk menghitung *Nutasi* ($\Delta\psi$).

$$\Delta\psi = \text{jumlah koreksi}/10000/3600, \text{ hasil dalam derajat.}$$

Selanjutnya perhitungan berpindah ke tabel koreksi $\Delta\epsilon$ pada lampiran III.

Koreksi dihitung menggunakan rumus:

$$(\text{Coefficient 1} + \text{Coefficient 2} * T) * \cos(\text{Multiple Arguments})$$

Setelah itu, seluruh koreksi dijumlahkan untuk kemudian digunakan untuk menghitung $\Delta\epsilon$ menggunakan rumus:

$$\Delta\epsilon = \text{koreksi}/10000/3600, \text{ hasil dalam derajat.}$$

Setelah itu menghitung *true obliquity* menggunakan rumus:

$$\epsilon = E_0 + \Delta\epsilon, \text{ hasil dalam derajat.}$$

Selanjutnya menghitung koreksi *aberasi* dengan rumus:

$$\text{Aberasi} = (-20,4898'' / R)/3600, \text{ hasil dalam derajat.}$$

Terakhir, yakni menghitung *apparent longitude* Matahari berdasarkan hasil perhitungan di atas dengan rumus:

$$\lambda = \Theta^t + \Delta\psi + \text{Aberasi}, \text{ hasil dalam bentuk derajat.}$$

c. Menghitung *Right Ascension* dan *Deklinasi Matahari*⁸

Rumus *apparent right ascension*:

⁸*Ibid.* hlm. 89

$\alpha = \text{Atan} ((\text{Sin } \lambda * \text{Cos } \varepsilon - \text{Tan } \beta * \text{Sin } \varepsilon) / \text{Cos } \lambda)$, hasil dalam derajat.

Rumus *apparent declination*:

$\text{Sin } \delta = \text{Sin } \beta * \text{Cos } \varepsilon + \text{Cos } \beta * \text{Sin } \varepsilon * \text{Sin } \lambda$, hasil dalam derajat.

d. Menghitung *Equation of Time*⁹

Sebelum menghitung *equation of time*, terlebih dahulu dilakukan perhitungan bujur rata-rata Matahari menggunakan rumus:

$$L_o = 280,4664567 + 360007,6982779 * T - 0,03032028 * T^2 + T^3 / 49931 - T^4 / 15299 - T^5 / 1988000, \text{ hasil dalam derajat dan di modulus-kan.}$$

Rumus perhitungan *equation of time*:

$$Eq = L_o - 0,0057183 - \alpha + \Delta\psi * \text{COS } \varepsilon, \text{ hasil dalam menit waktu.}$$

e. Menghitung Semi Diameter Matahari¹⁰

$$Sd = 15' 59,63'' / R, \text{ hasil dalam derajat.}$$

3. Perhitungan Data *Ephemeris* Bulan

a. Perhitungan Koreksi Posisi Bulan¹¹

Langkah pertama dalam perhitungan posisi Bulan yakni menghitung *Moon's mean longitude* terlebih dahulu menggunakan rumus berikut:

⁹ *Ibid.* hlm. 171-175

¹⁰ *Ibid.* hlm. 359-361

¹¹ *Ibid.* hlm. 307-308

$$L' = 218,3164591 + 481267,88134236 * T - 0,0013268 * T^2 + T^3/538841 - T^4/65194000$$

Selanjutnya dilanjutkan pada perhitungan rumus-rumus *argument* berikut:

Rumus perhitungan *mean elongation of the Moon*:

$$D = 297,8502042 + 445267,1115168 * T - 0,00163 * T^2 + T^3/545868 - T^4/113065000$$

Rumus perhitungan *Sun's mean Anomaly*:

$$M = 357,5291092 + 35999,0502909 * T - 0,0001536 * T^2 + T^3/24490000$$

Rumus perhitungan *Moon's mean Anomaly*:

$$M' = 134,9634114 + 477198,8676313 * T + 0,008997 * T^2 + T^3/69699 - T^4/14712000$$

Rumus *Argumen Bujur Bulan* (jarak rata-rata Bulan yang dihitung dari titik perpotongan lintasan bulan dengan *ekliptika*):

$$F = 93,2720993 + 483202,0175273 * T - 0,0034029 * T^2 - T^3/3526000 + T^4/863310000$$

Setelah itu dilanjutkan pada perhitungan 3 argumen tambahan dengan rumus sebagai berikut:

$$\begin{aligned} A1 &= 119,75 + 131,849 * T \\ A2 &= 53,09 + 479264,29 * T \\ A3 &= 313,45 + 481266,484 * T \end{aligned}$$

Hasil perhitungan diatas dalam bentuk derajat dengan nilai yang belum terbatas, untuk mengubahnya menjadi nilai diantara 0-360°, hasil perhitungan terlebih dahulu di-*modulus*-kan. Adapun untuk digunakan dalam perhitungan pemrograman *PHP* nilai derajat diubah menjadi radians.

Langkah selanjutnya untuk menghitung posisi bulan yakni menghitung koreksi periodik berdasarkan tabel koreksi periodik yang terdapat pada lampiran IV untuk koreksi bujur Bulan, lampiran V untuk

koreksi lintang Bulan dan lampiran VI untuk koreksi jarak Bumi-Bulan. Rumus yang digunakan untuk menghitung koreksi pada tabel koreksi periodik bujur dan lintang Bulan adalah sebagai berikut:

$$\text{Koreksi} = \text{Coefficient} * \text{Sin (Multiple Arguments)}$$

Namun Jika nilai M pada tabel $\neq 0$ ($0 < M$ atau $M > 0$) maka:

$$\text{Koreksi} = \text{Coefficient} * E * \text{Sin (Multiple Arguments)}$$

Di mana E adalah eksentrisitas orbit Bumi

$$\text{dan } E = 1 - 0,002516 * T - 0,0000074 * T^2$$

Sedangkan untuk menghitung koreksi pada tabel koreksi periodik jarak Bumi-Bulan menggunakan rumus:

$$\text{Koreksi} = \text{Coefficient} * \text{Cos (Multiple Arguments)}$$

Namun Jika nilai M pada tabel $\neq 0$ ($0 < M$ atau $M > 0$) maka:

$$\text{Koreksi} = \text{Coefficient} * E * \text{Cos (Multiple Arguments)}$$

Di mana E adalah eksentrisitas orbit Bumi

$$\text{dan } E = 1 - 0,002516 * T - 0,0000074 * T^2$$

Selanjutnya seluruh koreksi dari masing-masing tabel dijumlah, di mana hasil penjumlahan tersebut yakni Σl untuk hasil koreksi periodik bujur Bulan, Σb untuk hasil koreksi periodik lintang Bulan dan Σr untuk koreksi periodik jarak Bumi-Bulan. Untuk mendapatkan hasil koreksi tambahan bujur Bulan menggunakan rumus:¹²

¹² *Ibid.* hlm. 312

$$\text{Kor. tambahan } \Sigma l = (\Sigma l + 3958 * \sin(A1) + 1962 * \sin(L' - F) + 318 * \sin(A2)) / 1.000.000$$

Untuk mendapatkan hasil koreksi tambahan lintang Bulan menggunakan rumus:¹³

$$\text{Kor. tambahan } \Sigma b = (\Sigma b - 2235 * \sin(L') + 382 * \sin(A3) + 175 * \sin(A1 - F) + 175 * \sin(A1 + F) + 127 * \sin(L' - M') - 115 * \sin(L' + M')) / 1.000.000$$

Untuk menghitung koreksi tambahan untuk jarak Bumi-Bulan menggunakan rumus:¹⁴

$$\text{Kor. tambahan } \Sigma r = \Sigma r / 1.000, \text{ hasil dalam satuan kilometer.}$$

b. Perhitungan *True Longitude*, *True Latitude* dan *True Geocentric Distance of The Moon*

Rumus yang digunakan untuk menghitung koordinat Bulan (*true longitude*, *latitude* dan *true geocentric distance of the Moon*) yakni:¹⁵

$$\begin{aligned} \lambda &= L' \text{ (dalam format derajat) + kor. tambahan } \Sigma l, \text{ hasil dalam derajat.} \\ \beta &= \text{kor. tambahan } \Sigma b, \text{ hasil dalam derajat} \\ \text{TGD} &= 385000,56 + \text{kor. tambahan } \Sigma r, \text{ hasil dalam kilometer.} \end{aligned}$$

c. *Nutasi* dan *True Obliquity*

Perhitungan *nutasi* dan *true obliquity* pada perhitungan data Bulan adalah sama dengan perhitungan *nutasi* dan *true obliquity* pada perhitungan data Matahari.

¹³ *Ibid.*

¹⁴ *Ibid.*

¹⁵ *Ibid.*

d. *Apparent Longitude* Bulan (λ)

Rumus yang digunakan yakni:¹⁶

$$\text{Apparent Longitude} = \text{True Longitude} + \Delta\psi$$

e. *Apparent Right Ascension* dan *Deklinasi* Bulan

Rumus *apparent right ascension*:¹⁷

$$\alpha = \text{Atan} ((\text{Sin } \lambda * \text{Cos } \varepsilon - \text{Tan } \beta * \text{Sin } \varepsilon) / \text{Cos } \lambda)$$

Rumus *apparent declination*:¹⁸

$$\text{Sin } \delta = \text{Sin } \beta * \text{Cos } \varepsilon + \text{Cos } \beta * \text{Sin } \varepsilon * \text{Sin } \lambda$$

f. Perhitungan *Horizontal Parallaks* Bulan

Untuk menghitung *horizontal parallaks* Bulan, dapat menggunakan rumus berikut:¹⁹

$$\text{Sin HP}_c = 6378,14 / \text{TGD bulan, hasil dalam satuan derajat.}$$

g. Menghitung Semi diameter Bulan

Ada beberapa rumus yang bisa digunakan untuk menghitung semi diameter Bulan, salah satunya yakni menggunakan rumus:²⁰

$$\text{Sd}_c = 358473400 / \text{Jarak Bumi-Bulan}$$

h. Menghitung *Fraction Illumination* Bulan²¹

Untuk menghitung *fraction illumination* Bulan (FIB), diperlukan data Matahari, sebab nilai FIB tergantung dari besar-kecilnya sudut elongasi Bulan. Data Matahari yang dibutuhkan untuk perhitungan FIB

¹⁶ *Ibid.* hlm. 313.

¹⁷ *Ibid.*

¹⁸ *Ibid.*

¹⁹ *Ibid.* hlm. 263.

²⁰ *Ibid.* hlm. 360-361.

²¹ *Ibid.* hlm. 315-317

yakni α_o , δ_o dan R (*deklinasi*, *right ascension* dan jarak Bumi-Matahari). Adapun untuk langkah pertama yang dilakukan yakni dengan menghitung sudut *elongasi* Bulan terlebih dahulu dengan rumus:

$$\text{Cos } el = \text{Sin } \delta_o * \text{Sin } \delta_c + \text{Cos } \delta_o * \text{Cos } \delta_c * \text{Cos } (\alpha_o - \alpha_c)$$

Langkah selanjutnya yakni menghitung *phase angle of the Moon*. Namun Sebelum menghitung *phase angle*, terlebih dahulu nilai R diubah dari satuan AU ke dalam satuan kilometer, di mana 1 AU= 149.597.871 km. Adapun rumus untuk menghitung *phase angle of the Moon* yakni:

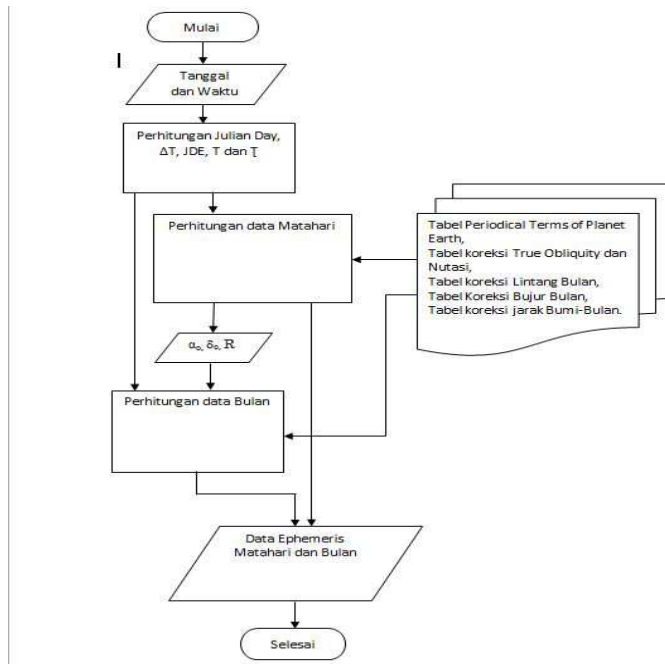
$$\text{Tan } i = (R * \text{Sin } el) / (\text{TGD bulan} - R * \text{Cos } el)$$

Terakhir yakni menghitung FIB dengan rumus:

$$\text{FIB} = (1 + \text{cos } i) / 2$$

4. Diagram Alir Perhitungan Data *Ephemeris* Matahari dan Bulan Algoritma Jean Meeus

Berdasarkan proses perhitungan yang telah dijelaskan di atas, secara umum alur proses perhitungan data *ephemeris* Matahari dan Bulan dalam algoritma Jean Meeus tersebut dapat digambarkan sebagai berikut:



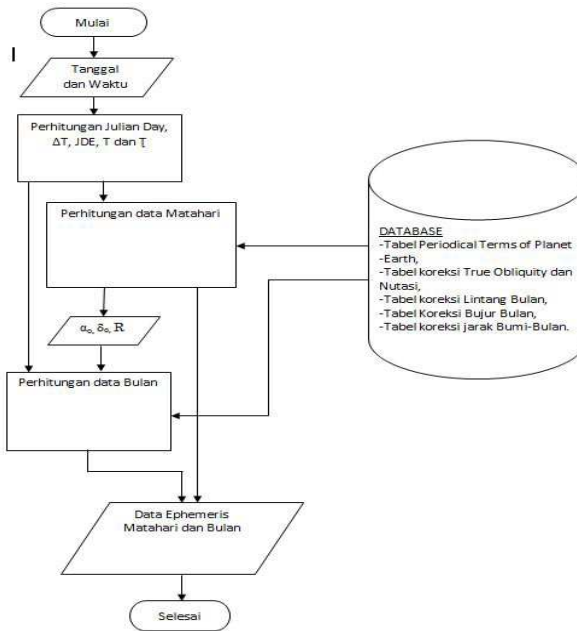
Gambar 3.2 *Flowchart* Perhitungan Data Matahari dan Bulan Algoritma Jean Meeus

B. Desain dan Rancangan Program

1. Rancangan Perangkat Lunak

a. Rancangan Proses Perhitungan

Rancangan proses perhitungan aplikasi *EphemerisSaya* mengikuti alur perhitungan algoritma Jean Meeus sebagaimana telah dijelaskan sebelumnya. Adapun diagram alir proses perhitungan aplikasi *EphemerisSaya* adalah sebagai berikut:



Gambar 3.3 *Flowchart* Alur Perhitungan Aplikasi *EphemerisSaya*

b. Rancangan Penyusunan *Database*

Database dalam suatu aplikasi berguna untuk menyimpan data-data dengan jumlah banyak yang diperlukan dalam proses perhitungan, sehingga pengguna tidak perlu meng-*input* secara manual data-data tersebut setiap kali melakukan perhitungan. Dalam aplikasi *EphemerisSaya*, *database* yang digunakan adalah berupa tabel-tabel koreksi yang diperlukan dalam perhitungan data Matahari dan Bulan algoritma Jean Meeus.

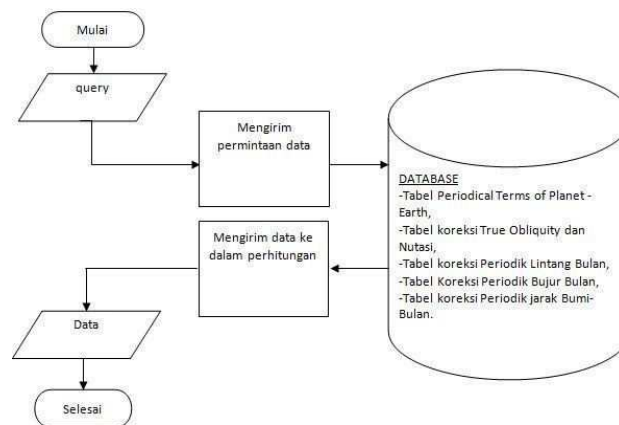
Terdapat 6 macam tabel koreksi yang digunakan dalam perhitungan data Matahari dan Bulan algoritma Jean Meeus. Tabel-tabel tersebut yakni: tabel *periodic terms for planet Earth*, tabel koreksi nutasi, tabel koreksi *true obliquity*, tabel koreksi periodik bujur Bulan,

tabel koreksi periodik lintang Bulan, tabel koreksi periodik jarak Bumi-Bulan. Adapun isi tabel-tabel tersebut secara lengkap dapat dilihat pada lampiran I, II, III, IV, V dan VI.

Sebagaimana disebutkan sebelumnya, bahwa jenis *database* yang digunakan pada aplikasi ini adalah *MySQL*. Oleh karena itu, untuk dapat dipergunakan di dalam aplikasi, maka tabel-tabel koreksi yang terdapat dalam lampiran I s/d VI terlebih dahulu dimasukkan ke dalam sistem *MySQL database*.

c. Rancangan Pengambilan *Database*

Untuk menggunakan data yang berada di dalam *MySQL database* diperlukan bahasa perintah (*query*) *SQL*, yakni bahasa perintah standar yang digunakan untuk permintaan data dari *database* berjenis *SQL*. Adapun alur permintaan data tersebut secara umum dapat digambarkan sebagai berikut:



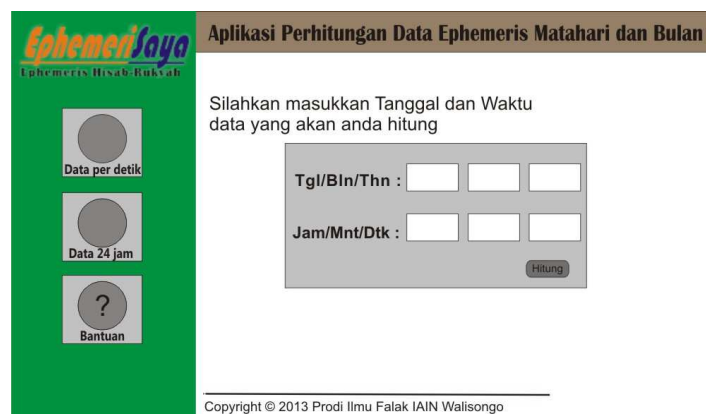
Gambar 3.4 Flowchart Alur Proses Pembacaan *Database*

2. Desain Antarmuka

Aplikasi *EphemerisSaya* dirancang secara *fleksibel* untuk dapat diakses melalui *desktop browser* maupun *mobile browser*. Oleh karena itu, untuk kenyamanan pengguna maka dibutuhkan 2 macam desain antarmuka aplikasi, yakni: antarmuka versi *web* untuk pengguna yang mengakses menggunakan *personal computer* dan antarmuka versi *mobile web* bagi pengguna yang mengakses aplikasi melalui perangkat *mobile*.

a. Desain Antarmuka *Web*

Secara umum desain antarmuka aplikasi yang digunakan dalam tampilan *web* dibagi menjadi tiga macam bagi, yakni tampilan halaman utama, tampilan halaman *output* dan tampilan halaman bantuan. Karena tampilan antarmuka *web* diperuntukkan bagi pengguna yang mengakses aplikasi melalui *desktop browser*, maka desain tampilan dirancang sedemikian rupa agar mempermudah pengguna ketika menjalankan aplikasi melalui *desktop browser*.



Gambar 3.5 Desain Antarmuka Halaman Utama/Halaman Input Versi *Web*

Halaman Utama adalah halaman yang secara *default* muncul ketika pertama kali program aplikasi diakses oleh pengguna. Pada halaman ini terdapat *form input* data yang diperlukan untuk perhitungan data ephemeris Matahari dan Bulan. Pada versi *web*-nya terdapat dua opsi *input*, yakni *input* untuk perhitungan data *ephemeris* untuk ketelitian per detik dan *input* untuk perhitungan data *ephemeris* dalam 1 hari, dengan ketelitian per jam.

Pada *form input* opsi pertama, data yang diperlukan yakni data tanggal, bulan, tahun, jam, menit dan detik. Sedangkan pada *form input* opsi kedua, data yang harus dimasukkan hanya ada tiga, yakni data tanggal, bulan dan tahun.



Gambar 3.6 Desain Antarmuka Halaman *Output* Data Versi *Web*

Halaman *output* data adalah halaman yang berfungsi untuk menampilkan hasil perhitungan aplikasi *EphemerisSaya* berdasarkan *input* data yang dimasukkan sebelumnya. Sebagaimana halaman utama yang memiliki dua opsi *input* data, untuk halaman penampilan *output* data

juga menyediakan dua macam halaman penampil data, yakni halaman penampil data *ephemeris* hasil perhitungan untuk ketelitian per detik dan halaman penampil data *ephemeris* hasil perhitungan untuk 1 hari.

Halaman bantuan, halaman ini berisi keterangan mengenai panduan penggunaan aplikasi dan informasi tentang aplikasi.



Gambar 3.7 Desain Antarmuka Halaman Bantuan Versi *Web*

b. Desain Antarmuka *Mobile-Web*

Desain antarmuka aplikasi *EphemerisSaya* untuk versi *mobile-web* dirancang lebih minimalis, menyesuaikan keterbatasan memori dan tampilan yang sering menjadi kendala pada *mobile browser*. Menu perhitungan data *ephemeris* untuk 24 jam tidak diikutsertakan pada versi ini, sehingga hanya menu perhitungan data *ephemeris* untuk ketelitian per detik yang disediakan dalam versi ini. Namun demikian desain antarmuka aplikasi tetap dirancang sedemikian rupa untuk mempermudah pengguna ketika mengaksesnya melalui *mobile browser*, tanpa mengabaikan sisi artistiknya.

Secara umum terdapat tiga macam tampilan pada versi *mobile-web*, yakni halaman utama/*home*, halaman hasil perhitungan dan halaman bantuan/*help*.



Gambar 3.8 Desain Versi *Mobile-Web*: Halaman Home (*Kiri*), Halaman Tampil Hasil Perhitungan (*Tengah*) dan Halaman Bantuan (*Kanan*)

C. Implementasi Rancangan Program *EphemerisSaya*

Setelah perancangan desain aplikasi *EphemerisSaya* sebagaimana telah dijelaskan di atas, tahap selanjutnya yakni pengimplementasian desain rancangan program tersebut ke dalam bahasa pemrograman *PHP* dan *MySQL*.

1. Lingkungan Implementasi

Proses implementasi rancangan program *EphemerisSaya* ke dalam bahasa pemrograman *PHP* dan *MySQL* untuk menghasilkan sebuah aplikasi *web*, memerlukan beberapa perangkat. Terdiri dari perangkat lunak sebagai media yang digunakan dalam proses pemrograman dan perangkat keras sebagai alat untuk menjalankan perangkat lunak tersebut. Perangkat

Lunak yang dibutuhkan untuk proses pemrograman aplikasi *Ephemerisaya* antara lain:

- a. *Microsoft Windows 7 Ultimate* sebagai sistem operasi yang digunakan penulis untuk merancang aplikasi *Ephemerisaya*.
- b. *Notepad++* sebagai *default editor* yang digunakan penulis dalam proses *coding*.
- c. *Notepad2* sebagai editor opsional yang digunakan pula oleh penulis dalam proses *coding*.
- d. *Microsoft Excel 2010*, sebagai media penyusunan data-data yang akan digunakan sebagai *database*.
- e. *XAMPP 1.8.1* yakni aplikasi *localhost* yang berguna sebagai lingkungan penyusunan program aplikasi sebelum di-upload ke *webhosting*.
- f. *phpMyAdmin* yakni program *administrator database* yang termasuk satu paket *localhost XAMPP 1.8.1*. Aplikasi ini berfungsi untuk merancang dan menyimpan *database* dalam bentuk *MySQL*, dengan aplikasi ini pula *database* yang sebelumnya telah disusun menggunakan *Microsoft Excel 2010*, akan di-*import* ke dalam bentuk *MySQL*.
- g. *Macromedia Dreamweaver 8*, merupakan aplikasi yang digunakan untuk membuat desain tampilan halaman *web*.

h. *Mozilla Firefox* dan *Google Chrome*, aplikasi browser yang digunakan untuk menampilkan program.

Spesifikasi perangkat keras yang digunakan penulis untuk menyusun program *EphemerisSaya* yakni *Notebook Axiio Neon Series MNC-M740S* dengan *processor* Pentium Dual-Core T4500 2.30 GHz dan *memory* RAM 1 GB.

2. Implementasi Perangkat Lunak

Pada bagian ini akan dijelaskan mengenai proses implementasi desain rancangan aplikasi *EphemerisSaya* menggunakan bahasa pemrograman *PHP* dan basis data *MySQL*. Adapun tahap-tahap implementasi rancangan aplikasi secara keseluruhan adalah sebagai berikut:

a. Implementasi Proses Perhitungan

Proses perhitungan data *ephemeris* pada aplikasi ini menggunakan teknik pemrograman berorientasi obyek /*Object Orientation Programming* (OOP). Sebagaimana diterangkan pada bab sebelumnya bahwa teknik OOP digunakan untuk pemrograman yang memiliki keterikatan data dan kompleksitas proses. Hal tersebut bertujuan untuk efisiensi dan fleksibilitas penulisan program karena teknik ini mencegah pengulangan penulisan data yang sama dan antar obyek dalam program dapat saling berhubungan.²²

Secara umum dalam teknik OOP program akan dibagi menjadi obyek-obyek yang nantinya dapat terhubung satu sama lain. Obyek-obyek

²² Betha Sidik, *Pemrograman Web dengan PHP*, Bandung: Penerbit Informatika, 2012, hlm. 507-508

tersebut terbagi ke dalam *property* dan *method*. Adapun dalam pemrograman *PHP*, *class* berfungsi sebagai *property* sedangkan *method*-nya adalah berupa fungsi-fungsi yang ada di dalam *class* tersebut.²³

Implementasi teknik OOP dalam pemrograman proses perhitungan *EphemerisSaya* yakni dengan membagi proses perhitungan menjadi 3 *class* utama, yakni *class JulianDay*, *class RumDasMat*, dan *class RumDasBul*. *Class JulianDay* berisi rumus-rumus perhitungan Julian Day, Delta T, JDE, T dan Υ , *class RumDasMat* berisi rumus-rumus perhitungan data *ephemeris* Matahari dan *class RumDasBul* berisi rumus perhitungan data *ephemeris* Bulan. Adapun rumus-rumus tersebut ditulis dalam bentuk fungsi-fungsi yang dimasukkan ke dalam masing-masing *class*.

Sebagai gambaran penulisan proses perhitungan aplikasi *EphemerisSaya* menggunakan teknik OOP, di bawah ini merupakan contoh penulisan rumus perhitungan *apparent latitude* Matahari dan perhitungan *true geocentric distance* Matahari yang masing-masing dibentuk menjadi *function Betha* dan *function R*, yang mana keduanya berada di dalam *class RumDasMat*.

```
class RumDasMat {
    //APPARENT LATITUDE MATAHARI
    function Betha ($Bhe,$deltaB) {
        $Betha=$Bhe-$deltaB;
        return $Betha;}
    //True Geosentric Distance Matahari
    function R ($Tau,$R0,$R1,$R2,$R3,$R4) {
        $R=($R0+$R1*$Tau+$R2*(pow($Tau,2))+$R3*(pow($Tau
        ,3))+$R4*(pow($Tau,4)))/(pow(10,8));
        return $R;}}
```

²³ *Ibid.* hlm. 517-519

Penulisan secara lengkap *source code* dari *class JulianDay*, *class RumDasMat*, dan *class RumDasBul*, dapat dilihat pada lampiran VII.

Proses pemrograman selanjutnya yakni dengan pembuatan *inheritance class* atau *class* turunan. *inheritance class* berfungsi sebagai *property* penyalaras proses perhitungan antar fungsi yang ada dalam 3 *class* utama. Pada *class* ini berisi variabel-variabel dari fungsi yang telah ada pada 3 *class* utama. Variabel dari fungsi-fungsi tersebut dipanggil kembali untuk kemudian diproses secara urut mengikuti alur perhitungan algoritma Jean Meeus. Sebagaimana pada penulisan *class* utama, *inheritance class* juga dibagi menjadi 3 buah *class* yang mana masing-masing *class* tersebut saling terhubung, ketiga buah *inheritance class* tersebut yakni *class HitJulDay*, *class HitDatMat* dan *class HitDatBul*. Adapun penulisan secara lengkap *source code* dari *inheritance class* tersebut dapat dilihat pada lampiran VIII.

b. Implementasi *Database*

Proses pemrograman selanjutnya yakni berupa implementasi penyusunan *database*. Data-data yang terdapat pada tabel-tabel koreksi pada lampiran I s/d VI, masing-masing ditulis ulang menggunakan aplikasi *Microsoft Excel*. Setelah penulisan selesai, tabel-tabel tersebut kemudian disimpan dengan format *CSV*.

Tahap selanjutnya yakni peng-*import*-an tabel-tabel *database* tersebut ke dalam format *MySQL database*. Proses peng-*import*-an tersebut menggunakan aplikasi *phpMyAdmin* yang telah disediakan dalam paket

program *localhost XAMPP*. Sebelum melakukan peng-*import*-an, terlebih dahulu dibuat sebuah *database* pada *phpMyAdmin* dengan nama *dbephemeris.sql*. Setelah itu tabel-tabel *database* yang telah disusun sebelumnya dengan format *CSV* di-*import* ke dalam file *dbephemeris.sql* dengan format *MySQL*.

Oleh karena itu, secara umum pada file *database dbephemeris.sql* terdapat enam file tabel. Keenam file tabel tersebut yaitu: file *appendix.sql* yang berisi tabel *periodical terms of planet Earth*, file *delta_psi.sql* berisi tabel koreksi *nutasi*, file *delta_epsilon.sql* berisi tabel koreksi *obliquity*, file *suku_periodik_bujur_ekliptik.sql* yang berisi tabel koreksi periodik bujur ekliptika Bulan, file *suku_periodik_lintang_ekliptik.sql* yang berisi tabel koreksi periodik lintang ekliptika Bulan dan file *suku_periodik_jarak_bumi_bulan.sql* yang berisi tabel koreksi periodik jarak Bumi-Bulan.

c. Implementasi Pengaksesan *Database*

Bahasa pemrograman yang digunakan dalam penyusunan aplikasi *EphemerisSaya* adalah *PHP*, sedangkan *database* yang digunakan adalah *MySQL*. Untuk itu diperlukan suatu proses untuk dapat mengakses *database* dalam file *dbephemeris.sql* untuk dapat digunakan digunakan dalam proses perhitungan aplikasi tersebut.

PHP telah menyediakan fungsi-fungsi untuk keperluan interaksi antar keduanya. Beberapa fungsi konektivitas *PHP* ke *MySQL database* yang digunakan dalam pemrograman aplikasi *Ephemerisaya* antara lain:²⁴

- 1) *mysql_connect* (*string server, string username, string password*), merupakan fungsi untuk membuka koneksi ke suatu *server MySQL*.
- 2) *mysql_close* (*resource link_identifier*), adalah fungsi untuk menutup koneksi dari suatu *server MySQL* yang telah dibuka.
- 3) *mysql_select_db* (*string database_name, resource link_identifier*), adalah fungsi untuk memilih *database* yang akan digunakan.
- 4) *mysql_query* (*string query*), adalah fungsi untuk menuliskan perintah permintaan eksekusi yang akan dijatuhkan kepada *database* yang telah dipilih. Pada fungsi ini dimasukkan bahasa perintah (*query*) *SQL*. *SQL Query* yang dipakai dalam aplikasi ini adalah perintah untuk menampilkan data dengan parameter berupa variabel dari *field* tertentu, yakni dengan menggunakan perintah **SELECT** dan klausa **WHERE**.²⁵
- 5) *mysql_fetch_array* (*resource result*), adalah fungsi untuk mengambil *record* dari suatu *database* untuk kemudian menampilkannya ke dalam *array* asosiatif atau pun *array* numerik.

²⁴ Kasiman Peranginangin, *Aplikasi Web dengan PHP dan MySQL*, Yogyakarta: Penerbit Andi, 2006, hlm. 417-418

²⁵ *Ibid.* hlm. 400-403

Adapun implementasi proses pengaksesan *database* dalam aplikasi ini adalah sebagai berikut:

1) Implementasi pembukaan koneksi *database*

```
<?php
//membuka database
$konek= mysql_connect ("localhost","root","");
if (!$konek) {
die ( "Tak Terkoneksi".mysql_error());
}
else{
echo "Terkoneksi<br>";
}
?>
```

2) Implementasi pemilihan *database*

```
<?php
//memilih database
mysql_select_db ("dbephemeris",$konek);
?>
```

3) Implementasi *query* untuk membaca *database* dan menampilkannya dalam *array* asosiatif

```
<?php
function Arrai ($NamaTabel,$series) {
//membaca database
$IsiKue="SELECT D,M,Mkomple,F,koef FROM $NamaTabel
WHERE series=$series";
$Kueri = mysql_query($IsiKue);
//menampilkan dalam array
$Arrai = mysql_fetch_array($Kueri);
return $Arrai;
}
?>
```

4) Implementasi penutupan koneksi *database*

```
<?php
mysql_close($konek);
?>
```

d. Implementasi Input Data

Sebelum proses perhitungan aplikasi *Ephemeris* Saya dijalankan, pengguna terlebih dahulu harus memasukkan data-data yang dibutuhkan.

Data-data yang dibutuhkan dalam proses perhitungan yakni berupa data tanggal, bulan, tahun, jam, menit dan detik.

Secara *default* pada bahasa pemrograman *PHP* tidak menyertakan fasilitas *form input* data, mengingat fungsi dasar dari bahasa pemrograman ini adalah sebagai *preprocessor*/pengolah data saja. Adapun untuk proses inputisasi data dapat menggunakan fasilitas *form* yang telah disediakan dalam bahasa *HTML* untuk kemudian isi dari *form input* data tersebut dikirimkan ke dalam bahasa *PHP*.

Adapun implementasinya dalam pemrograman ini yakni dengan membuat *form HTML* dalam file *input.htm* yang terdiri dari *form* tanggal, *form* bulan, *form* tahun, *form* jam, *form* menit dan *form* detik. Setelah itu data yang telah di-*input* ke dalam *form-form* tersebut dikirim ke dalam file *tampildata.php*, yang berfungsi sebagai penangkap data.

Data-data yang dikirim tersebut pada file *tampildata.php* ditangkap dengan *method get* kemudian dimasukkan ke dalam variabel-variabel yang telah disediakan. Variabel-variabel tersebut nantinya digunakan dalam proses perhitungan data *ephemeris*. Berikut ini *source code* dari file *input.htm*:

```
<html><style type="text/css"><!--.style1 {font-family: Arial,
Helvetica, sans-serif}--></style>
<body>
Hitung data Sun Moon Ephemeris
<form action="tampildata.php" method="get">
<table width="400" border="0" bordercolor="#FFFFFF"
bgcolor="#CCCCCC">
<tr>
<tdwidth="155">
```

```

        <span class="style1"> Jam/Menit/Detik </span>
    </td>
    <td width="8">:</td>
    <td width="164">
        <input name="JAM" type="text" value="00" size="2"> :
        <input type="text" name="MENIT" value="00" size="2"
        maxlength="2"/> : <input type="text" name="DETIK"
        value="00" size="2" maxlength="2"/> <span
        class="style1"> GMT </span>
    </td>
</tr>
<tr>
    <td>
        <span class="style1"> Tanggal/Bulan/Tahun </span>
    </td>
    <td>:</td>
    <td>
        <input type="text" name="TANGGAL" value="00" size="2"
        maxlength="2"/> / <input type="text" name="BULAN"
        value="00" size="2" maxlength="2"/> / <input
        type="text" name="TAHUN" value="0000" size="2"
        maxlength="4"/>
    </td>
</tr>
<tr>
    <td colspan="3" align="right">
        <input type="submit" name="hitung" value="hitung">
    </div>
    </td>
</tr>
</table><p>&nbsp;</p></form>
</body></html>

```

Berikut ini *source code* proses penangkapan data dari file *tampildata.php*:

```

<?php
    $tahun=$_GET[TAHUN];
    $bulan=$_GET[BULAN];
    $tgl=$_GET[TANGGAL];
    $jam=$_GET[JAM];
    $menit=$_GET[MENIT];
    $detik=$_GET[DETIK];
?>

```

e. Implementasi Penampilan Data

Perintah penampilan data pada *PHP* yakni menggunakan operator **echo** atau **print**. Adapun implementasinya dalam proses penampilan hasil perhitungan pada program ini yakni dengan menuliskan **echo** atau **print**

pada variabel data-data *ephemeris* hasil perhitungan. Berikut ini *source code* untuk menampilkan data *ephemeris* Matahari dan Bulan yang terdapat pada file *tampildata.php*:

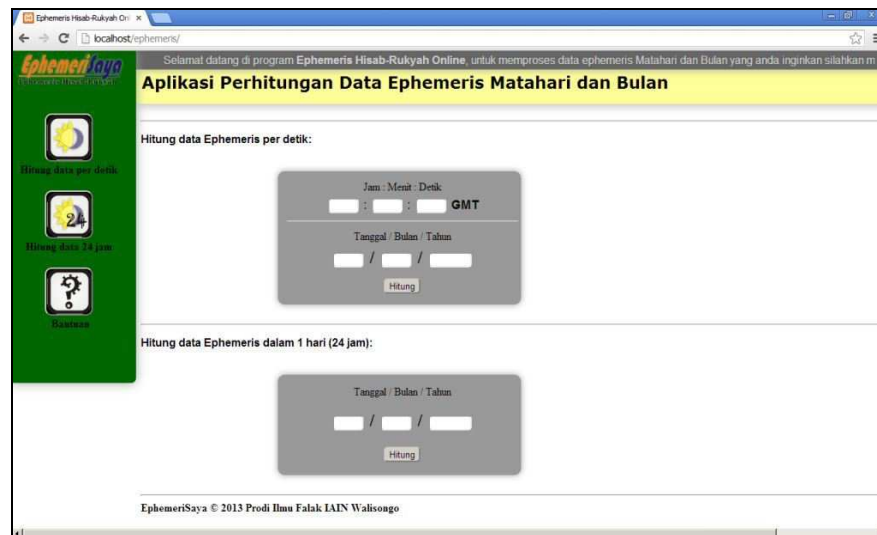
```
<?php
echo "<br>DATA MATAHARI
    <br>Apparent Longitude Matahari = $LAMDAadms
    <br>Apparent Latitude Matahari = $BETHA
    <br>True Obliquity = $Epsidms
    <br>Apparent Right Ascension Matahari = $Alphadms
    <br>Apparent Declination Matahari = $DELTAadms
    <br>Equation of Time = $EQhms
    <br>Semi Diameter Matahari = $SDdms
    <br>True Geocentric Distance Matahari = $TGD
    <br>";
echo "<br>DATA BULAN
    <br>Apparent Longitude Bulan = $ApparLongBuldms
    <br>Apparent Latitude Bulan = $ApparLatBuldms
    <br>Apparent Right Ascension Bulan = $RABuldms
    <br>Apparent Declination Bulan = $dekBuldms
    <br>Semi Diameter Bulan = $Sdbuldms
    <br>True Geocentric Distance Bulan = $TGD
    <br>Horisontal Parallax = $HPdms
    <br>FIB = $FIB
    <br>";
?>
```

f. Implementasi Antarmuka

Dalam penyusunan antarmuka aplikasi *EphemerisSaya*, penulis menggunakan bahasa penyusunan *web* yakni *HTML* dan *CSS*. *HTML* merupakan bahasa yang digunakan untuk menyusun halaman *website*, sedangkan *CSS* dipergunakan untuk menyusun *desain* tampilan *website*. Adapun untuk proses pengimplementasian desain antarmuka aplikasi ini, penulis menggunakan program *Macromedia Dreamweaver* sebagai editor penulisan *HTML* dan *CSS*.

Sebagaimana pada desain rancangan antarmuka yang telah dijelaskan sebelumnya, aplikasi *EphemerisSaya* memiliki dua versi

antarmuka yakni antarmuka versi *web* dan versi *mobile-web*. Pada masing-masing versi dibagi menjadi tiga macam tampilan yakni tampilan halaman utama (*home*), halaman *output* data dan halaman bantuan (*help*). Berikut ini merupakan antarmuka aplikasi versi *web*.



Gambar 3.9 Antarmuka Halaman Utama Versi *Web*

Jam	Ecliptic Longitude	Ecliptic Latitude	Apparent Right Ascension	Apparent Declination	True geocentric Distance	Semi Diameter Matahari	True Obliquity	Equation of Time
0 GMT	270° 17' 20.94"	-0.76°	270° 18' 54.53"	-23° 26' 6.38"	0.98369461 AU	0° 16' 15.54"	23° 26' 6.76"	1 m 36.06 d
1 GMT	270° 19' 53.67"	-0.76°	270° 21' 40.98"	-23° 26' 6.02"	0.98369215 AU	0° 16' 15.54"	23° 26' 6.76"	1 m 34.82 d
2 GMT	270° 22' 28.4"	-0.76°	270° 24' 27.44"	-23° 26' 5.61"	0.98368971 AU	0° 16' 15.54"	23° 26' 6.76"	1 m 33.58 d
3 GMT	270° 24' 59.13"	-0.76°	270° 27' 13.81"	-23° 26' 5.16"	0.98368727 AU	0° 16' 15.54"	23° 26' 6.76"	1 m 32.34 d
4 GMT	270° 27' 31.86"	-0.76°	270° 30' 0.37"	-23° 26' 4.65"	0.98368485 AU	0° 16' 15.55"	23° 26' 6.76"	1 m 31.1 d
5 GMT	270° 30' 4.59"	-0.75°	270° 32' 46.83"	-23° 26' 4.09"	0.98368244 AU	0° 16' 15.55"	23° 26' 6.76"	1 m 29.86 d
6 GMT	270° 32' 37.33"	-0.75°	270° 35' 33.29"	-23° 26' 3.49"	0.98368003 AU	0° 16' 15.55"	23° 26' 6.76"	1 m 28.62 d
7 GMT	270° 35' 10.06"	-0.75°	270° 38' 19.76"	-23° 26' 2.84"	0.98367764 AU	0° 16' 15.55"	23° 26' 6.76"	1 m 27.38 d
8 GMT	270° 37' 42.8"	-0.75°	270° 41' 6.22"	-23° 26' 2.13"	0.98367526 AU	0° 16' 15.55"	23° 26' 6.77"	1 m 26.13 d
9 GMT	270° 40' 15.54"	-0.74°	270° 43' 52.69"	-23° 26' 1.38"	0.98367289 AU	0° 16' 15.56"	23° 26' 6.77"	1 m 24.89 d
10 GMT	270° 42' 48.28"	-0.74°	270° 46' 39.16"	-23° 26' 0.58"	0.98367052 AU	0° 16' 15.56"	23° 26' 6.77"	1 m 23.65 d
11 GMT	270° 45' 21.02"	-0.74°	270° 49' 25.62"	-23° 25' 59.73"	0.98366817 AU	0° 16' 15.56"	23° 26' 6.77"	1 m 22.41 d
12 GMT	270° 47' 53.76"	-0.74°	270° 52' 12.09"	-23° 25' 58.83"	0.98366582 AU	0° 16' 15.57"	23° 26' 6.77"	1 m 21.17 d
13 GMT	270° 50' 26.5"	-0.73°	270° 54' 58.56"	-23° 25' 57.88"	0.98366349 AU	0° 16' 15.57"	23° 26' 6.77"	1 m 19.93 d
14 GMT	270° 52' 59.24"	-0.73°	270° 57' 45.03"	-23° 25' 56.89"	0.98366117 AU	0° 16' 15.57"	23° 26' 6.77"	1 m 18.69 d
15 GMT	270° 55' 31.99"	-0.73°	271° 0' 31.5"	-23° 25' 55.84"	0.98365886 AU	0° 16' 15.57"	23° 26' 6.77"	1 m 17.44 d
16 GMT	270° 58' 4.73"	-0.73°	271° 3' 17.97"	-23° 25' 54.74"	0.98365655 AU	0° 16' 15.57"	23° 26' 6.77"	1 m 16.2 d
17 GMT	271° 0' 37.48"	-0.72°	271° 6' 4.44"	-23° 25' 53.59"	0.98365425 AU	0° 16' 15.58"	23° 26' 6.77"	1 m 14.96 d
18 GMT	271° 3' 10.23"	-0.72°	271° 8' 50.91"	-23° 25' 52.4"	0.98365197 AU	0° 16' 15.58"	23° 26' 6.78"	1 m 13.72 d

Gambar 3.10 Antarmuka Halaman Penampil Data Perhitungan 24 Jam Versi *Web*

Jam:Menit:Detik : : GMT
Tanggal/Bulan/Tahun / /

Hitung

Data Matahari dan Bulan pada tanggal 29/02/2012 pukul 08:30:00 GMT

Julian Day 2455986.2541667 hari
Delta T 67.6652841 detik

Data Matahari

Jam	Ecliptic Longitude	Ecliptic Latitude	Apparent Right Ascension	Apparent Declination	True Geocentric Distance	Semi Diameter Matahari	True Obliquity	Equation of Time
08 : 30 : 00 GMT	340° 9' 34.51"	0.29°	341° 40' 58.07"	-17° 45' 29.4"	0.99071551 AU	0° 16' 8.62"	23° 26' 13.04"	-12 m 27.52 d

Data Bulan

Jam	Apparent Longitude Bulan	Apparent Latitude Bulan	Apparent Right Ascension	Apparent Declination	Horizontal Parallax	Semi Diameter Bulan	Distance of The Moon	Fraction Illumination Bulan
08 : 30 : 00 GMT	62° 29' 43.91"	0° 36' 57.84"	60° 17' 45.6"	21° 15' 42.25"	0° 54' 28.19"	0° 14' 49.67"	402929.788 km	0.434657

EphemerisSaya © 2013 Prodi Ilmu Falak IAIN Walisongo

Gambar 3.11 Antarmuka Halaman Penampil Data Perhitungan Per Detik Versi Web

BANTUAN

Panduan penggunaan aplikasi

- Masukkan data pada salah satu form pengisian data pada halaman utama
- Form pertama digunakan untuk menghitung data Ephemeris Matahari dan Bulan per detik
- Form kedua digunakan untuk menghitung data Ephemeris Matahari dan Bulan dalam 1 hari (24 jam)
- Perlu diingat, format waktu yang digunakan adalah GMT (Greenwich Mean Time), bukan waktu lokal.
- Proses perhitungan menggunakan Algoritma Jean Meeus.
- Saran dan kritik, via email < yakubndeso@yahoo.com > atau SMS phone di: < 0857 2172 8664 >

Tentang Aplikasi

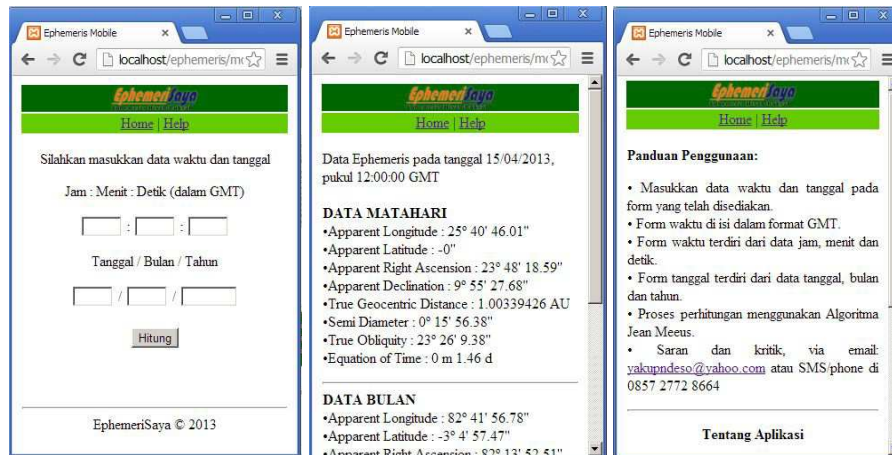
Aplikasi Ephemeris Hisab-Rukyah Online - EphemerisSaya v.1.0 © 2013
oleh M. Yakub Mubarak < yakubndeso@yahoo.com >

Prodi Ilmu Falak
Fakultas Syariah dan Ekonomi Islam
IAIN Walisongo

EphemerisSaya © 2013 Prodi Ilmu Falak IAIN Walisongo

Gambar 3.12 Antarmuka Halaman Bantuan Versi Web

Adapun antarmuka versi *mobile web* dari aplikasi ini adalah sebagai berikut:



Gambar 3.13 Gambar Antarmuka Versi *Mobile Web*, Halaman Utama (*kiri*), Halaman Penampil Data (*tengah*) dan Halaman Bantuan (*kanan*)

Antarmuka tampilan aplikasi dirancang untuk dapat berjalan secara *fleksibel*, di mana antarmuka yang ditampilkan di layar *browser* akan menyesuaikan jenis perangkat dan *browser* yang digunakan oleh pengguna. Apabila aplikasi diakses melalui perangkat komputer maka antarmuka yang muncul adalah antarmuka versi *web* dan jika diakses menggunakan perangkat *mobile* maka antarmuka yang muncul adalah versi *mobile web*. Hal ini dimungkinkan karena pada aplikasi telah disertakan sistem analisis pengenalan jenis perangkat dan *browser* yang digunakan pengguna. Adapun *source code* yang digunakan untuk merancang sistem penganalisa jenis perangkat dan *browser*, selengkapnya dapat dilihat pada lampiran IX.