

**APLIKASI PERHITUNGAN *LOSS* SAMBUNGAN SKSO DENGAN MENGGUNAKAN
HANDPHONE BERBASIS JAVA (J2ME)**
*APPLICATION FOR SKSO CONNECTION LOSS CALCULATION WITH HANDPHONE BASED ON
JAVA (J2ME)*

Adnan Purwanto, Anggun Fitriani Isnawati, Bayu Hatmoko
^{1,2,3} Program Studi Diploma III Teknik Telekomunikasi AKATEL Purwokerto
[1jeandoel@yahoo.com](mailto:jeandoel@yahoo.com), [2anggun_fitriani@yahoo.com](mailto:anggun_fitriani@yahoo.com)

ABSTRAK

Dunia Teknologi Informasi (TI) telah berkembang pesat salah satunya adalah bahasa pemrograman. Ada beberapa bahasa pemrograman, diantaranya adalah Java. Program Java yang digunakan adalah Java 2 *Micro Edition* (J2ME) digunakan untuk menjalankan dan mengembangkan aplikasi-aplikasi Java pada perangkat semacam *handphone*, PDA (Personal Digital Assistance) Palm, dan poket PC. Hasil studi pemanfaatan *handphone* sebagai media perencanaan di perhitungan *link budget* di ilmu telekomunikasi menunjukkan hasil ada peningkatan efisiensi dan kecepatan waktu dalam proses perhitungan. Dalam penelitian ini akan dilakukan hal yang sama, namun untuk perhitungan *loss* sambungan.

Kata kunci : Java 2 *Micro Edition*, *loss*, *handphone*

PENDAHULUAN

Dunia Teknologi Informasi (TI) telah berkembang pesat salah satunya adalah bahasa pemrograman. Ada beberapa bahasa pemrograman, diantaranya adalah Java.

Java adalah bahasa pemrograman yang dapat berjalan di semua *platform* tanpa harus mengubah kode sedikitpun, asalkan sistem yang digunakan terdapat JRE (*Java Runtime Environment*). Hal ini yang menjadi kekuatan Java sebagai bahasa pemrograman yang dapat menyelesaikan masalah-masalah terutama dalam perhitungan yang sulit dan membutuhkan waktu yang lama. Aplikasi menggunakan Java sangatlah luas dan dibagi menjadi tiga sub bagian yaitu J2EE (*Java 2 Enterprise Edition*), J2SE (*Java 2 Standard Edition*), dan J2ME (*Java 2 Micro Edition*). Dari ketiga sub bagian yang akan digunakan dalam program java ini

adalah J2ME. Karena, *software* ini yang dapat diaplikasikan dalam *handphone*.

Fungsi *handphone* sekarang berbeda dengan dahulu yang fungsi *handphonenya* hanya sebatas telepon dan sms. Tetapi, sekarang sudah bisa memainkan musik, kamera, game dan aplikasi. Sehingga *handphone* masa sekarang sangat berpengaruh pada perkembangan *programmer* yang ingin menciptakan sebuah program dan juga program ini dapat digunakan sewaktu-waktu dalam melakukan perhitungan sistem komunikasi serat optik yang dapat diakses melalui *handphone*.

Dari hasil penelitian yang pernah dilakukan oleh Urip Imam Saputra dengan judul "Perhitungan Link Budget pada Sistem Komunikasi Satelit" dan Beny Priyo Utomo dengan judul "Perhitungan *Link Budget* pada Sistem Komunikasi Serat Optik". Menunjukkan

hasil yang dapat diperoleh yaitu sisi efektif dan efisiensi dari penggunaan *handphone* sebagai media penerapan aplikasi. Hasil lain yang didapat adalah dapat memberikan akurasi dan ketepatan seperti *calculator programmable*

Berdasarkan nilai lebih tersebut diatas, penulis mencoba mengangkat judul **“APLIKASI PERHITUNGAN LOSS SAMBUNGAN SKSO DENGAN MENGGUNAKAN HANDPHONE BERBASIS JAVA (J2ME)”**.

METODOLOGI PENELITIAN

1. Frekuensi Normal V (Parameter-V)^[5]

Berdasarkan penjalaran gelombang elektromagnetik dalam inti, dapat diturunkan besaran yang disebut frekuensi normal (*V-number / parameter-V*), yaitu:

$$V = \frac{2\pi a}{\lambda} \sqrt{n_1^2 - n_2^2} = \frac{2\pi a}{\lambda} n_1 \sqrt{2\Delta} \dots(1)$$

dimana:

V = frekuensi normal atau parameter V

a = jari-jari inti (μm)

λ = panjang gelombang (μm)

n_1 = *core*

n_2 = *cladding*

Parameter-V merupakan suatu nilai yang berkaitan dengan kondisi *cut off* dari suatu serat optik. Merupakan parameter yang menentukan seberapa banyak modus penjalaran yang mampu di-*support* oleh suatu fiber.

2. Loss Sambungan

Secara ideal sambungan antar dua potong serat optik harus mampu menyalurkan daya

optik secara sempurna sehingga tidak mengurangi kinerja sistem. Tetapi pada kenyataannya selalu terjadi ketidaksempurnaan yang mengakibatkan reduksi daya optik pada saat melalui sambungan.

Beberapa masalah yang biasa dihadapi dalam sambungan adalah sebagai berikut: (i) pantulan celah, (ii) separasi antara dua ujung, (iii) kualitas ujung tidak sempurna, dan (iv) perbedaan diameter.

a. Loss karena pantulan celah

Pantulan ini terjadi karena sinar melalui celah antara dua serat optik. Berdasarkan hukum pemantulan Fresnel, untuk optik *step index*, koefisien pantulan celah adalah:

$$R = \left(\frac{n_1 - n_c}{n_1 + n_c} \right)^2 \dots\dots\dots(2)$$

dengan n_1 = indeks bias inti dan n_c = indeks bias celah.

Efisiensi celah itu, selanjutnya adalah:

$$\eta_F = (1 - R)^2 = \left[\frac{4n_1 n_c}{(n_1 + n_c)^2} \right]^2 \dots\dots\dots(3)$$

dimana:

η_F = efisiensi fresnel

n_1 = core atau indeks bias inti

n_c = indeks bias celah

sehingga persamaan *loss*-nya adalah:

$$\mathcal{L}_F = -10 \log \eta_F \dots\dots\dots(4)$$

dimana:

\mathcal{L}_F = *loss* fresnel (dB)

η_F = efisiensi fresnel

Untuk memperkecil *loss* itu, disisipkan cairan untuk penyesuaian indeks bias (*matching liquid*) sehingga $n_c \approx n_1$

b. *Loss* karena salah sambung

Loss karena salah sambung sangat mudah terjadi dalam serat optik *singlemode* karena diameternya sangat kecil. Berbagai macam salah sambung dalam serat optik *singlemode* dikemukakan berikut ini.

Dalam perumusan efisiensinya, w_o adalah *mode-field* diameter (MFD) dari modus fundamental atau yang lebih dikenal dengan istilah *spot size*, yang memenuhi persamaan berikut:

$$w_o = 0,65a + 1,62aV^{-3/2} + 2,88aV^{-6} \dots\dots(5)$$

di mana a = jari-jari inti dan V = frekuensi normal.

a) Separasi

$$\eta_c = \frac{1}{(Z^2 + 1)}; Z = \frac{z\lambda}{2\pi w_o^2} \dots\dots(6)$$

dimana: η_c = efisiensi pada separasi

Z = kesalahan separasi

z = separasi (μm)

λ = panjang gelombang (μm)

w_o = *mode-field* diameter (μm)

b) Lateral

$$\eta_c = e^{-(u)^2}; u = \frac{d}{w_o} \dots\dots(7)$$

dimana: η_c = efisiensi pada lateral

u = kesalahan lateral

d = pergesaren sumbu (μm)

w_o = *mode-field* diameter (μm)

c) Sudut

$$\eta_c = e^{-(T)^2}; T = \frac{\pi n_2 w_o \theta}{\lambda} \dots\dots(8)$$

dimana: η_c = efisiensi pada sudut

T = kesalahan sudut

n_2 = *cladding*

w_o = *mode-field* diameter (μm)

θ = sudut (derajat)

λ = panjang gelombang (μm)

d) Diameter

$$\eta_c = 4 \left(\frac{w_{o2}}{w_{o1}} + \frac{w_{o1}}{w_{o2}} \right)^{-2} \dots\dots(9)$$

dimana: η_c = efisiensi pada diameter

w_{o1} = *mode-field* diameter serat optik pertama (μm)

w_{o2} = *mode-field* diameter serat optik kedua (μm)

Secara umum dapat dinyatakan bahwa jika terjadi beberapa salah sambung secara serentak, maka total efisiensi salah sambung adalah perkalian dari efisiensi-efisiensi itu :

$$\eta_T = \eta_1 \times \eta_2 \times \dots \dots\dots(10)$$

dimana: η_T = total efisiensi

η = efisiensi-efisiensi

Dari persamaan tersebut, dapat diperoleh *loss* sambungan tersebut adalah:

$$\mathcal{L}_T = -10 \log \eta_T \dots\dots(11)$$

dimana: \mathcal{L}_T = *loss* sambungan (dB)

η_T = total efisiensi

Java 2 Micro Edition (J2ME)

J2ME adalah satu set spesifikasi dan teknologi yang fokus kepada perangkat konsumen. Perangkat ini memiliki jumlah memori yang terbatas, menghabiskan sedikit

daya dari baterai, layar yang kecil dan *bandwidth* jaringan yang rendah. Dengan perkembangbiakan perangkat *mobile* konsumen dari telepon, PDA, kotak permainan ke peralatan-peralatan rumah, Java menyediakan suatu lingkungan yang *portable* untuk mengembangkan dan menjalankan aplikasi pada perangkat ini.

Program J2ME, seperti semua program Java adalah diterjemahkan oleh VM. Program-program tersebut dikompilasi ke dalam *bytecode* dan diterjemahkan dengan Java *Virtual Machine* (JVM). Ini berarti bahwa program-program tersebut tidak berhubungan langsung dengan perangkat karena J2ME menyediakan suatu interface yang sesuai dengan perangkat.

1. *Connected Limited Device Configuration*

CLDC adalah perangkat dasar dari J2ME, spesifikasi dasar yang berupa *library* dan API yang diimplementasikan pada J2ME, seperti yang digunakan pada telepon seluler, *pager*, dan PDA. Perangkat tersebut dibatasi dengan keterbatasan memori, sumber daya, dan kemampuan memproses. Spesifikasi CLDC pada J2ME adalah spesifikasi minimal dari *package*, kelas, dan sebagian fungsi Java *Virtual Machine* yang dikurangi agar dapat diimplementasikan dengan keterbatasan sumber daya pada alat-alat tersebut.

2. *Mobile Information Device Profile (MIDP)*

MIDP adalah spesifikasi untuk sebuah profil J2ME. MIDP memiliki lapisan di atas CLDC, API tambahan untuk daur hidup aplikasi,

antarmuka, jaringan, dan penyimpanan persisten.

HASIL DAN PEMBAHASAN

A. Perhitungan Parameter *Loss Sambungan* SKSO

1. Menghitung Parameter-V

Dengan menggunakan rumus 2.1 maka dapat dicari besarnya redaman total dengan cara:

$$V = \frac{2\pi a}{\lambda} \sqrt{n_1^2 - n_2^2}$$

$$V = \frac{2\pi 3,5}{1,55} \sqrt{1,48^2 - 1,46^2}$$

$$= 3,440367826$$

Tabel dibawah ini merupakan perhitungan parameter-V dengan nilai yang berbeda.

Tabel 4.2 Perhitungan parameter-V

No	λ	n_1	n_2	a	Parameter-V	
					Manual	Program
1	1,55	1,48	1,46	3,5	3,440367826	3,438
2	1,45	1,48	1,47	3	2,232770723	2,232
3	1,3	1,48	1,46	2,5	2,929983588	2,928
4	1,7	1,48	1,47	3,5	2,221825768	2,221
5	1,2	1,48	1,46	2	2,53931911	2,538

Jika menggunakan *handphone emulator* maka diperoleh hasil:



Gambar 4.1 Perhitungan Parameter-V

2. Menghitung Sambungan pada separasi dan MFD

Langkah kedua yaitu menghitung sambungan pada separasi, tetapi untuk mencari sambungan ini terlebih dahulu mencari MFD. Maka perhitungan manual sebagai berikut.

$$w_0 = 0,65a + 1,62aV^{-3/2} + 2,88aV^{-1}$$

$$w_0 = (0,65 \cdot 3,5) + (1,62 \cdot 3,5 \cdot 3,44^{-3/2}) + (2,88 \cdot 3,5 \cdot 3,44^{-6})$$

$$= 3,17059703 \mu\text{m}$$

Tabel dibawah ini merupakan perhitungan *mode field diameter* dengan nilai yang berbeda.

Tabel 4.3 Perhitungan *Mode field diameter*

No	Parameter-V	Jari-jari inti	<i>Mode field diameter</i>	
			Manual	Program
1	3,438	3,5	3,17059703	3,17
2	2,232	3	3,477333736	3,477
3	2,928	2,5	2,444774398	2,445
4	2,221	3,5	4,07198998	4,072
5	2,538	2	2,12287422	2,123

Mencari sambungan separasi maka diperoleh hasil:

$$Z = \frac{z\lambda}{2\pi w_0^2}$$

$$Z = \frac{1 * 1,55}{2\pi * 3,17^2}$$

$$= 0,024548971$$

Mencari efisiensi sambungan separasi maka diperoleh hasil:

$$\eta_c = \frac{1}{(Z^2 + 1)}$$

$$\eta_c = \frac{1}{(0,024548971^2 + 1)}$$

$$= 0,99939771$$

Tabel dibawah ini merupakan perhitungan efisiensi separasi dengan nilai yang berbeda.

Tabel 4.4 Perhitungan Efisiensi separasi

No	W_0	z	a	Efisiensi separasi	
				Manual	Program
1	3,17	1	1,55	0,999397711	0,999
2	3,477	2	1,45	0,998544591	0,998
3	2,445	2	1,3	0,995231348	0,995
4	4,072	3	1,7	0,997609382	0,998
5	2,123	3	1,2	0,98409685	0,984

Jika menggunakan *handphone emulator* maka diperoleh hasil:



Gambar 4.2 Perhitungan *Mode Field Diameter* (1) dan Efisiensi Sambungan Separasi (2)

Dalam memperoleh hasil tersebut, apabila dilakukan perhitungan secara manual maka akan diperoleh waktu ± 7 menit sedangkan menggunakan aplikasi akan diperoleh waktu $\pm 0,5$ menit. Hal ini berarti bahwa aplikasi ini memiliki efisiensi waktu dalam perhitungan sebesar ± 14 kali lebih cepat, dibandingkan dengan perhitungan secara manual.

3. Menghitung Sambungan Lateral

Kemudian Nilai sambungan lateral didapat dengan menggunakan rumus 2.7 sebagai berikut:

$$u = \frac{d}{w_o}$$

$$u = \frac{1,25}{3,17}$$

$$u = 0,394321766$$

Mencari efisiensi lateral maka diperoleh hasil:

$$\eta_c = e^{-(u)^2}$$

$$\begin{aligned} \eta_c &= e^{-(0,394321766)^2} \\ &= 0,855995931 \end{aligned}$$

Tabel dibawah ini merupakan perhitungan Efisiensi lateral dengan nilai yang berbeda.

Tabel 4.5 Perhitungan Efisiensi lateral

No	Mode field diameter	Pergeseran sumbu	Efisiensi Lateral	
			Manual	Program
1	3,17	1,25	0,855995931	0,856
2	3,477	1	0,920612372	0,921
3	2,445	1,5	0,686342512	0,686
4	4,072	0,75	0,966644971	0,967
5	2,123	1,75	0,506880953	0,507

Jika menggunakan *handphone emulator* maka diperoleh hasil:



Gambar 4.3 Perhitungan Efisiensi Lateral

Dalam memperoleh hasil tersebut, apabila dilakukan perhitungan secara manual maka akan diperoleh waktu ± 3 menit sedangkan menggunakan aplikasi akan diperoleh waktu $\pm 0,5$ menit.

4. Menghitung Sambungan Sudut

Kemudian nilai sambungan sudut didapat dengan menggunakan rumus 2.8 sebagai berikut:

$$T = \frac{\pi n_2 w_o \theta}{\lambda}$$

$$T = \frac{\pi * 1,46 * 3,17 * 2}{25}$$

$$= 1,16319353$$

Mencari efisiensi lateral maka diperoleh hasil:

$$\eta_c = e^{-(T)^2}$$

$$\begin{aligned} \eta_c &= e^{-(1,16319353)^2} \\ &= 0,258458746 \end{aligned}$$

Tabel dibawah ini merupakan perhitungan efisiensi sudut dengan nilai yang berbeda.

Tabel 4.6 Perhitungan Efisiensi sudut

No	W_o	n_2	Sudut	λ	Efisiensi sudut	
					Manual	Program
1	3,17	1,46	2	25	0,25845874	0,259
2	3,477	1,47	2	30	0,31792543	0,318
3	2,445	1,46	3	30	0,28431860	0,284
4	4,072	1,47	1	25	0,56790046	0,568
5	2,123	1,46	1	35	0,92551488	0,926

Jika menggunakan *handphone emulator* maka diperoleh hasil:



Gambar 4.4 Perhitungan Efisiensi Sudut

Dalam memperoleh hasil tersebut, apabila dilakukan perhitungan secara manual maka akan diperoleh waktu ± 3 menit sedangkan menggunakan aplikasi akan diperoleh waktu $\pm 0,5$ menit.

5. Menghitung Sambungan Diameter

Kemudian Nilai sambungan Diameter didapat dengan menggunakan rumus 2.9 tetapi harus mencari parameter- v dan *mode field* diameter yang kedua sebagai berikut.

Mencari parameter- v sebagai berikut:

$$V = \frac{2\pi a}{\lambda} \sqrt{n_1^2 - n_2^2}$$

$$V = \frac{2\pi 3}{1,55} \sqrt{1,48^2 - 1,46^2}$$

$$= 2,948886708$$

Mencari *mode field* diameter sebagai berikut:

$$w_0 = 0,65a + 1,62aV^{-3/2} + 2,88aV^{-6}$$

$$w_0 = (0,65*3) + (1,62*3*2,948^{-3/2}) + (2,88*3*2,948^{-6})$$

$$= 2,923326004$$

Mencari efisiensi sambungan pada diameter maka diperoleh hasil:

$$\eta_c = 4 \left(\frac{w_{o2}}{w_{o1}} + \frac{w_{o1}}{w_{o2}} \right)^{-2}$$

$$\eta_c = 4 \left(\frac{2,923}{3,17} + \frac{3,17}{2,923} \right)^{-2}$$

$$= 0,993448129$$

Tabel dibawah ini merupakan perhitungan efisiensi diameter dengan nilai yang berbeda.

Tabel 4.7 Perhitungan Efisiensi diameter

No	w_0	a_1	a_2	λ	n_1	n_2	Efisiensi diameter	
							Manual	Program
1	3,2	3,5	3	1,5	1,48	1,46	0,99345	0,993
2	3,5	3	2,5	1,4	1,48	1,47	0,99942	0,999
3	2,4	2,5	3,5	1,3	1,48	1,46	0,96438	0,964
4	4,1	3,5	2	1,7	1,48	1,47	0,96356	0,963
5	2,1	2	3,5	1,2	1,48	1,46	0,91218	0,912

Jika menggunakan *handphone emulator* maka diperoleh hasil



Gambar 4.5 Perhitungan Efisiensi Diameter

Dalam memperoleh hasil tersebut, apabila dilakukan perhitungan secara manual maka akan diperoleh waktu ± 6 menit sedangkan menggunakan aplikasi akan diperoleh waktu $\pm 0,5$ menit.

6. Menghitung Efisiensi Freshnel

Kemudian yang selanjutnya adalah mencari nilai efisiensi freshnel dengan menggunakan rumus 2.3 maka diperoleh hasil perhitungannya adalah

$$\eta_F = (1 - R)^2 = \left[\frac{4n_1n_c}{(n_1 + n_c)^2} \right]^2$$

$$\eta_F = \left[\frac{4 * 1,48 * 1}{(1,48 + 1)^2} \right]^2$$

$$= 0,926481368$$

Tabel dibawah ini merupakan perhitungan efisiensi fresnel dengan nilai yang berbeda.

Tabel 4.8 Perhitungan Efisiensi fresnel

No	Core	Indeks bias celah	Efisiensi fresnel	
			Manual	Program
1	1,48	1	0,926481	0,926
2	1,48	1.25	0,985854	0,986
3	1,48	1.5	0,999909	1
4	1,48	1.75	0,986074	0,986
5	1,48	2	0,955843	0,956

Jika menggunakan *handphone emulator* maka diperoleh hasil:



Gambar 4.6 Perhitungan Efisiensi Fresnel

Dalam memperoleh hasil tersebut, apabila dilakukan perhitungan secara manual maka akan diperoleh waktu ± 2 menit

sedangkan menggunakan aplikasi akan diperoleh waktu $\pm 0,5$ menit.

7. Menghitung Total Efisiensi

Kemudian yang selanjutnya adalah mencari nilai total *efisiensi* dengan menggunakan rumus 2.10 maka diperoleh hasil perhitungannya adalah

$$\eta_T = \eta_1 \times \eta_2 \times \dots$$

$$\eta_T = 0,999 * 0,856 * 0,259 * 0,993 * 0,926$$

$$= 0.203656958$$

Tabel dibawah ini merupakan perhitungan total efisiensi dengan nilai yang berbeda.

Tabel 4.9 Perhitungan Total Efisiensi

No	Eff. fresnel	Eff. separasi	Eff. Lateral	Eff. sudut	Efisiensi diameter	Total Efisiensi	
						Manual	Program
1	0,926	0,999	0,856	0,259	0,993	0.2036	0,203
2	0,986	0,998	0,921	0,318	0,999	0.2879	0,288
3	1	0,995	0,686	0,284	0,964	0.1869	0,187
4	0,986	0,998	0,967	0,568	0,963	0.5205	0,52
5	0,956	0,984	0,507	0,926	0,912	0.4028	0,403

Jika menggunakan *handphone emulator* maka diperoleh hasil:



Gambar 4.7 Perhitungan Total Efisiensi

Dalam memperoleh hasil tersebut, apabila dilakukan perhitungan secara manual maka akan diperoleh waktu ± 1 menit sedangkan menggunakan aplikasi akan diperoleh waktu $\pm 0,5$ menit.

8. Menghitung Loss Sambungan

Kemudian yang selanjutnya adalah mencari nilai *loss* sambungan dengan menggunakan rumus 2.11 maka diperoleh hasil perhitungannya adalah

$$\begin{aligned} \xi_T &= -10 \log \eta_T \\ \xi_T &= -10 \log 0,203 \\ &= 6,92 \text{ dB} \end{aligned}$$

Tabel dibawah ini merupakan perhitungan *loss* sambungan dengan nilai yang berbeda.

Tabel 4.10 Perhitungan Loss sambungan

No	Total efisiensi	Loss sambungan	
		Manual	Program
1	0,203	6,925039621	6,92
2	0,288	5,406075122	5,4
3	0,187	7,281583935	7,28
4	0,52	2,839966564	2,84
5	0,403	3,946949539	3,95

Jika menggunakan *handphone emulator* maka diperoleh hasil:



Gambar 4.8 Perhitungan Loss Sambungan

Dalam memperoleh hasil tersebut, apabila dilakukan perhitungan secara manual maka akan diperoleh waktu ± 1 menit sedangkan menggunakan aplikasi akan diperoleh waktu $\pm 0,5$ menit.

KESIMPULAN

1. Menghitung *Loss* sambungan sistem komunikasi serat optik dapat dibuat dengan menggunakan bahasa pemrograman Java dengan *handphone* yang memiliki konfigurasi CLDC 1.1 dan profil MIDP 2.0, karena spesifikasi ini yang dibutuhkan aplikasi dalam melakukan proses *floating point* (perhitungan).
2. Hasil perhitungan secara manual akan sama dengan hasil perhitungan menggunakan aplikasi program. Namun ada beberapa perhitungan aplikasi dibandingkan perhitungan secara manual akan terdapat perbedaan nilai angka di belakang koma. Karena nilai sudah dibulatkan dengan pembulatan tiga angka dibelakang koma.
3. Susunan program yang dibuat dari *software* menggunakan *netbeans* 6.5 dan proses pengkompilasi dengan *Java Development Kit* (JDK).
4. Operasi perhitungan pada J2ME hanya mendukung operasi penambahan dan pengurangan. Sehingga perlu direktori tambahan yang dinamakan *oMath*. Jenis *oMath* yang digunakan adalah *oMathFP* dan *oMathDbl*. Kedua *oMath* tersebut berfungsi

untuk perhitungan selain penambahan dan pengurangan.

Mobile, Penerbit Informatika, Bandung, 2006.

DAFTAR PUSTAKA

- 1) Anonymus. oMath.
<http://svn.omath.org/core/trunk/javasource/org/omath/ui/OmathApplet.java>.
 (diakses tanggal 20 Mei 2009, pukul 12.00)
- 2) _____. Spesifikasi *Handphone*.
<http://mobilezoo.biz/homepagemobile.php?manuf=sonyericsson&handset=K550j>
 (diakses tanggal 13 Agustus 2009, pukul 20.00)
- 3) Avestro, J, 2007. Pengembangan Aplikasi *Mobile*.pdf.
<ftp://192.168.0.109/Modul%20Kuliah/Module%20JENI%202/JENI-J2ME-Bab01-Pengembangan%20Aplikasi%20Mobile.pdf> (diakses tanggal 26 April 2009, pukul 12.30)
- 4) Budi Raharjo, Imam Heryanto dan Arif Haryono, *Tuntunan Pemrograman Java Untuk Handphone*, Penerbit Informatika, Bandung, 2007.
- 5) Isnawati, Anggun Fitriani. *Diktat Sistem Komunikasi Serat Optik*, Akatel Sandhy Putra Purwokerto. Purwokerto. 2005
- 6) M. Shalahuddin dan Rosa A. S, *Pemrograman J2ME Belajar Cepat Pemrograman Perangkat Telekomunikasi*