

## PENGUKURAN KUALITAS PERANGKAT LUNAK SISTEM E-LEARNING MENGUNAKAN METRIC FUNCTION ORIENTED

Ritzkal<sup>1\*</sup>, Arief Goeritno<sup>2</sup>, Eko Hadi P<sup>1</sup>

<sup>1</sup>Program Studi Teknik Informatika, Fakultas Teknik, Universitas Ibn Khaldun Bogor

<sup>2</sup>Program Studi Teknik Elektro, Fakultas Teknik, Universitas Ibn Khaldun Bogor

Jl. KH. Sholeh Iskandar KM.2, Kedung Badak, Tanah Sereal, Kota Bogor 16162 Jawa Barat

Email: [ritzkal@ft.uika-bogor.ac.id](mailto:ritzkal@ft.uika-bogor.ac.id)

### Abstrak

Pencapaian kualitas perangkat lunak dapat dinilai melalui pengukuran. Ada banyak atribut-atribut perangkat lunak yang dapat diukur. Jumlah atribut yang digunakan dalam pengukuran tergantung pada banyaknya informasi yang ingin diperoleh melalui pengukuran. Contohnya, ketika seorang manager proyek ingin memperoleh informasi mengenai tingkat keandalan dari perangkat lunak yang dikembangkan maka atribut-atribut yang diukur adalah seperti jumlah kesalahan yang mungkin terjadi dalam kurun waktu tertentu, jumlah fungsi, jumlah baris kode, kerumitan, dan ujicoba yang dilakukan untuk memastikan tingkat kesalahan yang mungkin terjadi selama proses pengembangan perangkat lunak. Pada akhirnya informasi-informasi tersebut akan digunakan untuk mendukung fungsi-fungsi manajemen seperti perencanaan, pengorganisasian, pelaksanaan, dan pengendalian. Banyaknya standar kualitas yang ada saat ini menunjukkan bahwa begitu pentingnya pencapaian sebuah kualitas. Pentingnya kualitas perangkat lunak memang masih diperhadapkan pada banyaknya standar kualitas yang ada. Para ahli rekayasa perangkat lunak tampaknya masih memerlukan sedikit waktu untuk menetapkan sebuah standar tunggal yang dapat menggugurkan standar-standar kualitas yang banyak ini. Di sisi lain, pencapaian kualitas perangkat lunak tidak bisa menunggu sampai standar tunggal itu ditetapkan. Tujuan dari penelitian yaitu mendapatkan nilai kualitas perangkat lunak sistem e-learning. Penelitian ini dilakukan dengan teknik kuantitatif. Penelitian ini adalah penerapan dari teori-teori mengenai kualitas perangkat lunak, dan dilakukan sebagai pembuktian teori-teori tersebut. Secara skematik, Metode penelitian dilakukan dengan beberapa tahapan kumpulan data, komputasi numeric (metric) dan komputasi numeric (indikator). Hasil yang diperoleh berupa pengumpulan data pada perangkat lunak, perhitungan komputasi numerik dan perhitungan indicator kualitas ISO 9126 yang meliputi indikator kualitas fungsional, indicator kualitas reliabilitas, indicator kualitas usabilitas, indicator kualitas efisiensi, indicator kualitas maintainabilitas, dan indicator kualitas portabilitas. Berdasarkan hasil tersebut, maka ditarik kesimpulan sesuai tujuan penelitian dengan pengujian indikator kualitas ISO 9126 menyatakan perangkat lunak tersebut baik.

*Kata kunci:* Iso 9126, komputasi numeric dengan metric dan komputasi numeric dengan indikator.

### 1. PENDAHULUAN

Pencapaian kualitas perangkat lunak dapat dinilai melalui pengukuran. Ada banyak atribut-atribut perangkat lunak yang dapat diukur. Jumlah atribut yang digunakan dalam pengukuran tergantung pada banyaknya informasi yang ingin diperoleh melalui pengukuran. Contohnya, ketika seorang manager proyek ingin memperoleh informasi mengenai tingkat keandalan dari perangkat lunak yang dikembangkan maka atribut-atribut yang diukur adalah seperti jumlah kesalahan yang mungkin terjadi dalam kurun waktu tertentu, jumlah fungsi, jumlah baris kode, kerumitan, dan ujicoba yang dilakukan untuk memastikan tingkat kesalahan yang mungkin terjadi selama proses pengembangan perangkat lunak. Pada akhirnya informasi-informasi tersebut akan digunakan untuk mendukung fungsi-fungsi manajemen seperti perencanaan, pengorganisasian, pelaksanaan, dan pengendalian.

Banyaknya standar kualitas yang ada saat ini menunjukkan bahwa begitu pentingnya pencapaian sebuah kualitas. Pentingnya kualitas perangkat lunak memang masih diperhadapkan pada banyaknya standar kualitas yang ada. Para ahli rekayasa perangkat lunak tampaknya masih memerlukan sedikit waktu untuk menetapkan sebuah standar tunggal yang dapat menggugurkan standar-standar kualitas yang banyak ini. Di sisi lain, pencapaian kualitas perangkat lunak tidak bisa menunggu sampai standar tunggal itu ditetapkan. Tujuan dari penelitian yaitu mendapatkan

nilai kualitas perangkat lunak sistem e-learning. Penelitian ini dilakukan dengan teknik kuantitatif. Penelitian ini adalah penerapan dari teori-teori mengenai kualitas perangkat lunak, dan dilakukan sebagai pembuktian teori-teori tersebut. Secara skematik

## 2. METODOLOGI

Metode penelitian dilakukan dengan beberapa tahapan kumpulan data, komputasi numerik (metric) dan komputasi numerik (indikator). Hasil yang diperoleh berupa pengumpulan data pada perangkat lunak, perhitungan komputasi numerik pada metric dan perhitungan komputasi numerik dengan indikator kualitas ISO 9126 yang meliputi indikator kualitas fungsional, indikator kualitas reliabilitas, indikator kualitas usabilitas, indikator kualitas efisiensi, indikator kualitas maintainabilitas, dan indikator kualitas portabilitas

## 3. HASIL DAN PEMBAHASAN

### 3.1. Hasil Pengamatan

Berdasarkan hasil pengamatan, maka didapatkanlah data-data seperti yang diperlihatkan oleh Tabel 1 berikut ini.

Nama Data	Item	Jumlah
Inputan Pengguna	Entri Pengguna Baru	8
	Entri Pokok Bahasan	
	Entri Soal	
	Pengaturan KBM	
	Pengaturan Komposisi Soal	
	Pengubahan Password	
	Pengubahan Pokok Bahasan	
	Pengubahan Akun Pengguna	
Output Pengguna	Daftar Pengguna	18
	Daftar Matapelajaran Aktif	
	Daftar Pokok Bahasan	
	Daftar Soal	
	Tampilan Jumlah Soal Aktif	
	Tampilan Detail Soal	
	Tampilan Pengaturan KBM	
	Tampilan Pengaturan Komposisi	
	Tampilan Arsip	
	Lembar Soal Ujian	
	Lembar Kunci Jawaban	
	Laporan Pengiriman Data Ke Server	
	Pesan Error Login	
	Laporan Pengubahan Data Ke Server	
	Daftar Wajib Soal	
	Informasi Log Pengguna	
	List Guru ( <i>dropdown list</i> )	
	Konfirmasi Penghapusan Data Pengguna	
Permintaan Pengguna	Menu Beranda	28
	Menu Entri Pokok Bahasan	
	Menu Filter Pokok Bahasan	
	Menu Entri Soal	
	Menu Filter Soal	
	Menu Generate Soal	
	Menu Arsip	
	Menu Pengaturan Kuliah	
	Menu Pengaturan Komposisi Soal	
	Menu Pengaturan Akun Pengguna	
	Menu Penggantian Password	
	Tombol Logout	
Tombol Login		
Tombol Ubah Pokok Bahasan		

	Tombol Ubah Akun Pengguna	
	Tombol Hapus Akun Pengguna	
	Tombol Aktikan Soal	
	Tombol Nonaktifkan Soal	
	Tombol Aktifkan Pokok Bahasan	
	Tombol Nonaktifkan Pokok Bahasan	
	Paging Daftar Soal	
	Paging Arsip Soal	
	Filter Soal	
	Filter Pokok Bahasan	
	Tombol Menampilkan Soal	
	Tombol Generate Soal	
	Tombol Tampilkan Lembar Soal	
	Tombol Tampilkan Kunci Jawaban	
File	File Arsip	
	File guru	
	File Jadwal Mengajar	
	File kelas	
	File Soal	
	File Jawaban	
	File Matapelajaran	14
	File Pengaturan Komposisi Soal	
	File Pokok Bahasan	
	File walikelas	
	File Pengaturan Jadwal	
	File Kelas	
	File Pengguna	
	File Wajib Soal	
Interface Eksternal	TCP/ IP	1
Modul Program Saat Ini	Entri Soal	
	Filter Soal	
	Generate Soal	
	Pengarsipan Soal	7
	Pengaturan Tahun Ajar	
	Pembuatan Silabus	
	Manajemen Akun Pengguna	
Pengaturan Komposisi Soal		
Modul Terganti	Generate Soal	2
	Manajemen Akun Pengguna	
Modul Terhapus	-	0
Bahasa Pemrograman	PHP 5	1
Kegagalan	Format Output Lembar Soal Ujian	
	Terjadinya Soal Ganda	
	Pelembaran variabel Session pada output Log Pengguna	4
	Tombol update pada record pengguna terakhir tidak berfungsi karena kesalahan kalang ( <i>looping</i> )	

### 3.2. Perhitungan Komputasi Numeric

Perhitungan komputasi numeric pada metric merupakan metric peangkat lunak berorientasi fungsi ditarik berdasarkan sebuah pengukuran fungsionalitas yang disampaikan oleh aplikasi sebagai suatu nilai normalisasi. Karena fungsionalitas tidak dapat diukur secara langsung, maka fungsionalitas harus ditarik secara tidak langsung dari pengukuran langsung lainnya. Metrik berorientasi fungsi dibuat oleh Alan J. Albrecht (1979) yang disebut dengan function point. Saat ini ada banyak variasi cara perhitungan function point setelah metode ini dikembangkan dan direvisi oleh International Function Point User Group (IFPUG) sejak tahun

1986. Namun pada penelitian ini penulis akan memfokuskan penggunaan function point yang dibuat oleh Albreth. Function point ditarik dengan menggunakan sebuah hubungan empiris berdasarkan pengukuran langsung domain informasi perangkat lunak yang dapat dihitung serta perkiraan kompleksitas perangkat lunak. Function point dihitung dengan menggunakan rating scale seperti yang ditunjukkan oleh Gambar 1.

**Tabel 2. Komputasi numeric**

Param Pengukuran	Jumlah	X	Faktor Pembobolan			
			Sederhana	Rata - Rata	Kompleks	
J. Input Pengguna	8	X	3			= 24
J. Output Pengguna	18	X	4			= 74
J. Permintaan Pengguna	28	X		5		= 140
J. File	14	X		10		= 140
J. Interface External	1	x	6			= 6
<b>Total</b>						<b>382</b>

**3.3. Perhitungan indikator kualitas ISO 9126**

Setelah data dikumpulkan maka langkah berikutnya adalah mencari indikator kualitas ISO 9126 yaitu fungsionalitas, reliabilitas, usabilitas, efisiensi, pemeliharaan, dan portabilitas

**3.3.1 Fungsionalitas**

Indikator fungsionalitas dapat ditarik dari function point. Perhitungan function point membutuhkan data berupa input pengguna, output pengguna, permintaan pengguna, file, dan interface eksternal. Setiap data tersebut harus dinilai kompleksitasnya secara umum yaitu sederhana, sedang atau kompleks. Dengan data dan penilaian tersebut, maka perhitungan function pointnya adalah sebagai berikut.

**Tabel 3. Indikator fungsionalitas**

**Notasi Domain Informasi**

Param Pengukuran	Jumlah	X	Faktor Pembobolan			
			Sederhana	Rata - Rata	Kompleks	
J. Input Pengguna	8	X	3			= 24
J. Output Pengguna	18	X	4			= 74
J. Permintaan Pengguna	28	X		5		= 140
J. File	14	X		10		= 140
J. Interface External	1	x	6			= 6
<b>Total</b>						<b>382</b>

**Faktor perubah kompleksitas**

Faktor perubah kompleksitas didapatkan dari penilaian keempatbelas atribut yang terdapat pada perangkat lunak. Keempatbelas atribut tersebut digunakan sebagai faktor untuk menormalisasi perhitungan function point. Perhitungan faktor perubah kompleksitas function point adalah sebagai berikut

**Tabel 4. Faktor perubah kompleksitas**

No	Faktor Peubah	Nilai
1	Backup dan recovery	0 1 2 3 4 5
2	Komunikasi data	0 1 2 3 4 5
3	Distribusi pemrosesan data	0 1 2 3 4 5
4	Performa	0 1 2 3 4 5
5	Konfigurasi operasional	0 1 2 3 4 5
6	Inputan onlie (on-line entry)	0 1 2 3 4 5
7	Perubahan data on-line (on-line update)	0 1 2 3 4 5
8	Tingkat transaksi data	0 1 2 3 4 5
9	Efisiensi pengguna	0 1 2 3 4 5
10	Kompleksitas pemrosesan	0 1 2 3 4 5
11	Penggunaan kembali (reusability)	0 1 2 3 4 5
12	Konversi dan Instalasi	0 1 2 3 4 5
13	Penggandaan instalasi	0 1 2 3 4 5

14	Fasilitas perubahan (Facilitate Change)	0 1 2 3 4 5
<b>TOTAL</b>		<b>30</b>

**Function Point**

Diketahui :  $Jumlah\ Total = 382$   
 $\sum Fi = 30$

Rumus untuk mencari function point adalah sebagai berikut.

$$FP = Jumlah\ Total \times (0,65 + 0,01 \sum Fi)$$

$$FP = 382 \times (0,65 + (0,01 \times 30))$$

$$FP = 362,9$$

$$FP_{Max} = 1,35 \times Jumlah\ Total$$

$$FP_{Max} = 1,35 \times 382$$

$$FP_{Max} = 515,7$$

Dengan keterangan:

- $FP$  = Function Point
- $FP_{Max}$  = Function Point Maksimum
- $Jumlah\ Total$  = Nilai Total domain informasi
- $\sum Fi$  = Jumlah harga penyesuaian kompleksitas

Berdasarkan function point dan function point maksimum, maka tingkat pencapaian fungsionalitas perangkat lunak e-learning adalah sebagai berikut.

Diketahui:  $FP = 362,9$   
 $FP_{Max} = 515,7$

$$Fungsionalitas = \frac{FP}{FP_{Max}}$$

$$Fungsionalitas = \frac{362,9}{515,7}$$

$$Fungsionalitas = \frac{362,9}{515,7}$$

**Fungsionalitas = 0,7037**

Jadi, nilai fungsionalitas perangkat lunak e-learning adalah sebesar 0,7037.

**3.3.2 Reliabilitas**

Indikator reliabilitas (keandalan) perangkat lunak didapatkan dari metrik *rate of failure occurrence (ROCOF)*. Untuk menghitung *ROCOF* dibutuhkan variabel function point (*FP*), dan jumlah kegagalan (*Failure*).

Diketahui:  $FP = 362,9$   
 $Failure = 4$

$$ROCOF = \frac{Kegagalan}{Function\ Point}$$

$$ROCOF = \frac{4}{362,9}$$

$$ROCOF = \frac{4}{362,9}$$

$$ROCOF = 0,110$$

Reliabilitas dapat ditarik menggunakan rumus berikut.

$$Reliabilitas = 1 - ROCOF$$

$$Reliabilitas = 1 - 0,110$$

**Reliabilitas = 0,9889**

Dengan demikian maka reliabilitas perangkat lunak e-learning adalah sebesar 0,9889. Nilai 0,9889 mengindikasikan bahwa dalam 1.000 kali pengoperasian, diperkirakan perangkat lunak aplikasi e-learning mampu bekerja dengan benar selama 988 kali. Dengan kata lain dalam 1000 kali pengoperasian, aplikasi e-learning diperkirakan mengalami kegagalan fungsi sebanyak 11 kali.

**3.3.3 Usabilitas**

Indikator usabilitas perangkat lunak didapatkan dari metrik *speed of operation*. Semakin *speed of operation* mendekati nilai 0 menandakan usabilitas semakin meningkat. Sebaliknya, semakin *speed of operation* mendekati nilai 1 maka usabilitas semakin menurun.

Diketahui:  $Function\ Point\ (FP) = 362,9$   
 $Inputan\ Pengguna = 8$   
 $Permintaan\ Pengguna = 28$

$$Speed\ of\ Operation = \frac{Inputan\ Pengguna + Permintaan\ Pengguna}{Function\ Point}$$

$$Speed\ of\ Operation = \frac{8 + 28}{362,9}$$

$$Speed\ of\ Operation = 0,0992$$

$$Usabilitas = 1 - Speed\ of\ Operation$$

$$Usabilitas = 1 - 0,0992$$

**Usabilitas = 0.9008**

Jadi tingkat usabilitas perangkat lunak E-Learning adalah sebesar 0,9008. Nilai 0,9008 mengindikasikan bahwa pekerjaan yang seharusnya diselesaikan dalam 1.000 kali kerja, sekarang digantikan oleh sistem sebanyak 900 pekerjaan sehingga pekerjaan tersebut dapat diselesaikan hanya dengan 100 kali kerja. Atau lebih jelasnya, pekerjaan yang seharusnya diselesaikan dalam waktu 40 jam (1 minggu kerja), akan selesai dalam waktu 4 jam (setengah hari kerja).

**3.3.4 Efisiensi**

Efisiensi berkaitan dengan performa, sumber daya yang diperlukan, dan penghematan yang diperoleh dari penggunaan produk. Untuk mendapatkan indikator efisiensi maka diperlukan beberapa variabel untuk menghitungnya. Variabel-variabel tersebut adalah total manfaat, total biaya, dan usaha yang diperlukan untuk membangun perangkat lunak.

**Metric Perkiraan**

Diketahui:  $FP = 362,9$   
 $E = -13,39 + 0,0545 FP$   
 $E = -13,39 + (0,0545 \times 362,9)$   
 $E = 6,38805\ person-month$

**3.3.5 Pemeliharaan**

Pada perangkat lunak, semakin banyak perubahan yang terjadi pada perangkat lunak menandakan bahwa pemeliharaan akan semakin sulit untuk dilakukan. Untuk pencarian nilai maintainabilitas, metrik yang digunakan adalah metrik *Software Maturity Index (SMI)*. Semakin SMI mendekati nilai 1 maka produk akan semakin stabil. Sebaliknya semakin SMI menjauhi nilai 1 maka produk akan semakin tidak stabil. Variabel-variabel yang dibutuhkan untuk mencari nilai SMI adalah jumlah modul saat ini (MT), jumlah penambahan modul (Fa), jumlah modul yang mengalami perubahan (Fc), dan jumlah modul yang sudah dihapus sejak perancangan awal (Fd).

Diketahui:  $MT = 7$   
 $Fa = 0$   
 $Fc = 2$   
 $Fd = 0$

$$SMI = \frac{MT - (Fa + Fc + Fd)}{MT}$$

$$SMI = Pemeliharaan$$

$$Pemeliharaan = \frac{MT - (Fa + Fc + Fd)}{MT}$$

$$Pemeliharaan = \frac{7 - (0 + 2 + 0)}{7}$$

**Pemeliharaan = 0,7142**

Indikator pemeliharaan sebesar 0,7142 menandakan bahwa dari setiap 10 modul program, terdapat 7 modul yang diperkirakan stabil sehingga tidak membutuhkan perubahan yang berarti pada waktu pemeliharaan dan 3 modul lainnya yang diperkirakan akan mengalami perubahan pada tahap pemeliharaan.

**3.3.6 Portabilitas**

Berdasarkan tinjauan pustaka, portabilitas perangkat lunak dibagi menjadi tiga level yaitu portabilitas *source code*, *intermediate code*, dan *runnable code*. Setiap bahasa pemrograman pasti memiliki salah satu dari sifat portabilitas tersebut. Bahasa pemrograman yang digunakan untuk membangun aplikasi e-learning adalah bahasa pemrograman PHP. Perangkat lunak yang dibangun dengan bahasa pemrograman PHP saat ini masih merupakan perangkat lunak yang *portable* secara *source code*. PHP memiliki kode translator berupa mesin PHP yang terletak di web server Apache / IIS. Mesin PHP menerjemahkan kode-kode

program sehingga dapat dibaca oleh *web browser* client seperti Microsoft Internet Explorer, Mozilla FireFox, Safari, Opera dan sebagainya. Sesuai dengan tinjauan pustaka mengenai portabilitas, indikator portabilitas *source code* =1, *intermediate code*= 0,66, dan *runnable code*=0,33. Karena sifat portabilitas yang dimiliki oleh perangkat lunak aplikasi e-learning adalah portabilitas *source code*, maka indikator portabilitas = 1.

**Generalisasi kualitas perangkat lunak**

Setelah mendapatkan keenam indikator kualitas ISO 9126 yaitu fungsionalitas, reliabilitas, usabilitas, efisiensi, pemeliharaan, dan portabilitas maka langkah terakhir adalah melakukan generalisasi. Generalisasi yang dimaksud adalah penilaian secara keseluruhan dari pencapaian kualitas perangkat lunak sistem E-Learning.

No	Karakteristik	Nilai
1	Fungsionalitas	0,70
2	Reliabilitas	0,99
3	Usabilitas	0,90
4	Efisiensi	0,11
5	Maintainabilitas (Pemeliharaan)	0,75
6	Portabilitas	1
<b>TOTAL KUALITAS</b>		<b>4,45</b>

Idealnya, nilai setiap indikator kualitas harus sama dengan 1 sehingga mendapatkan nilai total kualitas= 6. Dengan kata lain, kualitas perangkat lunak terbaik tercapai ketika nilai total kualitas = 6. Kenyataannya perangkat lunak aplikasi e-learning telah mencapai nilai kualitas sebesar 4.45. Jika pencapaian tersebut dikonversi ke dalam nilai persen maka kualitas perangkat lunak aplikasi e-learning adalah sebagai berikut.

$$\begin{aligned} \text{Diketahui: Kualitas Maksimum} &= 6 \\ \text{Pencapaian Kualitas} &= 4,40 \\ \text{Persentase Pencapaian Kualitas} &= \frac{\text{Pencapaian Kualitas}}{\text{Kualitas Maksimum}} \times 100\% \\ &= \frac{4,45}{6} \times 100\% \\ \text{Persentase Pencapaian Kualitas} &= 74,19\% \end{aligned}$$

Jadi, kualitas perangkat lunak aplikasi e-learning secara keseluruhan adalah sebesar 74,19% dengan pengorbanan awal sebesar Rp6.077.400,00. Dengan didapatkannya nilai kualitas sebesar 74,19%, maka pihak manajemen dapat menentukan untuk menggunakan perangkat lunak aplikasi e-learning, atau mencari alternatif perangkat lunak sejenis yang kualitasnya mungkin dapat melebihi aplikasi e-learning seandainya kualitas tersebut dianggap kurang memuaskan.

**4. KESIMPULAN**

Untuk menentukan kualitas produk perangkat lunak, maka diperlukan suatu pengukuran. Pengukuran kualitas melibatkan variabel, instrumen, dan standar. Dengan pengukuran yang menggunakan instrumen berupa metrik, dan standar ISO 9126, perangkat lunak aplikasi e-learning mencapai kualitas sebesar 74,19%. Karakteristik kualitas tertinggi dari aplikasi e-learning dicapai oleh karakteristik reliabilitas dengan nilai 0,9888 dan karakteristik terendah dicapai oleh karakteristik efisiensi dengan nilai 0,1024. Sebagai pembuktian hipotesis, berdasarkan pengukuran yang telah dilakukan, maka disimpulkan bahwa hasil pengukuran dari beberapa metrik berbanding lurus dengan kualitas perangkat lunak dan beberapa hasil pengukuran metrik lainnya berbanding terbalik dengan kualitas. Metrik-metrik yang berbanding lurus dengan kualitas perangkat lunak adalah *function point*, *Software Maturity Index (SMI)*, dan metrik portabilitas. Sedangkan metrik-metrik lainnya yang digunakan dalam penelitian ini berbanding terbalik dengan pengukuran kualitas perangkat lunak sehingga untuk menormalkannya digunakan pengurangan terhadap bilangan 1.

Berdasarkan hasil penelitian maka aplikasi e-learning yang akan diterapkan dianggap cukup layak digunakan dengan pencapaian kualitas sebesar 74,19%. Namun demikian, keputusan untuk menggunakan atau mencari alternatif pengganti perangkat lunak ini adalah hak

dari pihak manajemen. Pada akhirnya penelitian ini sudah selesai dilaksanakan dengan pencapaian tujuan untuk mengetahui tingkat pencapaian kualitas perangkat lunak menggunakan metode metrik *function-oriented* dan standar kualitas ISO 9126.

#### DAFTAR PUSTAKA

- Arfan, Aulia. 2010. *Implementasi Pengukuran Kualitas pada Perangkat Lunak Menggunakan Metode Lines of Code (LOC) dan Function Point (FP)*. Departemen Ilmu Komputer. Skripsi. Universitas Sumatera Utara.
- Hariyanto, Bambang, 2004. *Rekayasa Sistem Berorientasi Objek*. Informatika, Bandung.
- ISO/IEC 9126.1991. *Information Technology-Software Product Evaluation- Quality Characteristics and Guidelines for Their Use*. 1<sup>st</sup> ed. International Organization for Standardization.
- Longstreet, David. *Function Point Analysis Training Course* [pdf], <http://www.softwaremetrics.com/Function%20Point%20Training%20Booklet%20New.pdf> (5 Juli 2011).
- Nazir, Moh. 2005. *Metode Penelitian*. Ghalia Indonesia, Bogor.
- Pressman, Roger S. 2007. (terjemahan) *Rekayasa Perangkat Lunak Pendekatan Praktisi (Buku 1)*. Ed.2. Andi, Yogyakarta.
- Hutabarat, Bernaridho I. *Portabilitas:Memahami Makna yang Tepat tentang Portabilitas*. PC Media,ed.9, 2011.
- Simamarta, Janner. 2010. *Rekayasa Perangkat Lunak*. Andi, Yogyakarta.
- Sommerville, Ian. 2007. *Software Engineering*. 8<sup>th</sup> ed. Pearson Education Limited, Harlow.
- Whitten, Jeffery L., Bentley, Lonnie D. and Dittman, Kevin C. 2004. (terjemahan) *Metode Desain dan Analisis Sistem*. Ed.6. Andi, Yogyakarta.
- Wijarnako, Wahyu. 2005. *Implementasi Metrik pada Pengembangan Perangkat Lunak*. Jurusan Teknik Elektro. Skripsi. Universitas Gajah Mada.