

# MODEL I-CASE UNTUK PENGEMBANGAN REKAYASA PERANGKAT LUNAK PERPUSTAKAAN DIGITAL BERBASIS OPEN SOURCE

**Yodhi Yuniarthe**

Jurusan Teknik Informatika, STMIK Mitra, Bandar Lampung  
E-mail : yodhi@ymail.com

## ABSTRAK

Sebagai bagian akhir dari proses rekayasa perangkat lunak, I-CASE (*Integrated Computer Aided Software Engineering*) memegang peran penting dalam membentuk peralatan sistem yang menentukan kualitas perangkat lunak. Makalah ini menguraikan tentang suatu kajian secara konseptual mengenai model I-CASE yang didalamnya terdapat sub komponen yang berinteraksi dalam membangun peralatan proses (*process tools*) dalam CASE sehingga dapat disesuaikan dengan prinsip-prinsip dalam perangkat lunak. Studi kasus dalam penelitian ini adalah pada aplikasi perangkat lunak berbasis open source bagi perpustakaan di lingkungan perguruan tinggi (PT) telah mengantarkan pada kesuksesan secara teknis dimana program aplikasinya memiliki kompetensi untuk dikembangkan sehingga layak untuk dilakukan penelitian lebih lanjut. Perpustakaan digital (*digital library*) merupakan produk dari perpustakaan terbarukan saat ini. Masalah saat ini pada perpustakaan PT adalah pada sentralisasi dimana fasilitas perpustakaan membutuhkan integrasi data yang terkoordinasi agar pemustaka dapat mengakses sumber referensi yang memuaskan. Oleh karena itu, Penelitian ini memfokuskan bagaimana menyiapkan model I-CASE dalam rangka mengantisipasi kebutuhan perangkat lunak bagi perpustakaan digital tersebut. Metodologi penelitian yang digunakan adalah studi literatur yang komprehensif mengenai model arsitektural saat ini dan teknik pembaganan sistem yang menguraikan dokumentasi interface pemakai dengan kebutuhan sistem yang akan dibangun. Perkembangan open source yang begitu pesat menyebabkan kemudahan perancangan aplikasi, tetapi dapat menimbulkan perancangan yang samaunya karena sifat open source yang source codenya gratis (*free*). Hasil pembahasan diharapkan dapat memberikan masukan kepada para pengembang khususnya programmer menyiapkan rancangan program perangkat lunak berbasis open source dengan model I-CASE yang mendukung aplikasi pemakai dengan beragam pilihan sehingga mendukung otomasi perpustakaan.

**Kata kunci :** I-CASE, open source, process tools, digital library, model arsitektural

## 1. PENDAHULUAN

Perpustakaan digital saat ini telah mulai banyak diterapkan di lingkungan perguruan tinggi. Perkembangan teknologi informasi di bidang *open source* telah meningkatkan jumlah pengguna dan pemakaian system yang lebih fleksibel.

Paling tidak ada beberapa perubahan di dunia universitas yang akan mempengaruhi dunia perpustakaan juga di Indonesia, yaitu: 1). Tingkat Konsumerisme : jumlah mahasiswa terus bertambah, demikian pula keragaman dalam latarbelakang dan tujuan pendidikan mereka. 2). Pendidikan tinggi yang konsumtif – para mahasiswa dan orang tua memiliki harapan semakin tinggi tentang mutu pendidikan yang dikaitkan dengan kesempatan kerja, sedemikian rupa sehingga universitas pun menjadi semakin berorientasi kepada pelanggan (*customer oriented*). 3). Pembiayaan siswa : para mahasiswa diharapkan membayar lebih banyak dan dengan demikian menjadi kontributor penting bagi pendidikan mereka sendiri. 4). Rancangan kursus dan metode pembelajaran : cenderung menuju pendidikan berbasis semester dan menggunakan modul-modul yang dapat ditawarkan secara terpisah. menumbuhkan kecenderungan belajar secara otonom (*autonomous learning*) dengan memanfaatkan berbagai teknologi informasi, baik dalam pengajaran maupun pengujian. 5). Akuntabilitas pendidikan tinggi : setiap institusi dituntut untuk lebih menekankan aspek kualitas, sementara harus juga memikirkan sumber dana yang beragam untuk mendukung butir pertama, yaitu pendidikan yang bersifat massal. 6). Pendanaan : universitas semakin bersaing untuk meningkatkan

efektivitas pendanaan, dan mungkin juga akan menimbulkan tuntutan efisiensi, termasuk dalam hal perpustakaan.

Berdasarkan hal ini, maka perlu kiranya dilakukan studi lebih lanjut bagaimana peningkatan sumber daya organisasi pada perpustakaan digital yang mendukung teknologi informasi dengan model open source terkini melalui peralatan terpadu dari perangkat lunak komputer.

## 2. TINJAUAN PUSTAKA

### 2.1. Konsep Perpustakaan Digital

Teknologi informasi tidak terbatas hanya pada komputer tetapi terdiri dari beberapa kumpulan teknologi yang menghubungkan semua computer ke dalam jaringan untuk melakukan pertukaran informasi jarak jauh dan lintas batas perpustakaan. Internet menyediakan konektivitas global dan *flatfom* fleksible agar aliran informasi berjalan tanpa hambatan di sepanjang lini perpustakaan dan antara perpustakaan dengan pengguna maupun penerbit. Dengan kemampuan teknologi informasi yang semakin berkembang, berkembang pulalah bentuk perpustakaan digital tersebut dalam berbagai media diantaranya : [1].

#### 2.1.1 SGML

Standard Generalized Mark-up Language (SGML) adalah kumpulan dari kode-kode yang memberikan salah satu dalam bagian komponen-komponen (judul, formula, paragraf diagram dan lain-lain). Dokumen SGML dapat disimpan lebih efisien dan dapat diperoleh kembali dengan komponen individual. Lebih penting lagi, SGML melindungi gambar dari sebuah dokumen, mengijinkan untuk dapat ditampilkan pada layar video dengan cara yang sama dimana penulis atau penerbit memperuntukkannya untuk melihat ketika dicetak pada kertas.

#### 2.1.2. Perpustakaan Video Digital

Video menggabungkan bunyi dan gambar dan usaha lainnya tipe dari dokumen yang kompleks. Media video dapat mengambil secara terpisah dan mengartikan secara terpisah penggunaan alat-alat yang berbeda untuk komponen yang berbeda lalu dirakit kembali. Heterogenitas dari sumber-sumber daya informasi dan sistem komunikasi adalah alami dan konsekuensinya tidak dapat dihindarkan dari perkembangan didalam sebuah kompetisi dan ruang lingkup teknologi yang kreatif. Perbedaan pendekatan memproduksi perbedaan sistem dan praktek.

Pada tingkat pemula dengan relatif di dalam evolusi dari teknologi perpustakaan digital merupakan kebutuhan sangat vital dimana proyek berusaha keras untuk pendekatan penyatuan dengan fungsional secara standar dan protokol yang digunakan, sekalipun pada mulanya tidak sepenuhnya didayagunakan. Ketelitian mendesain dari kemampuan memperluas dalam perpustakaan digital akan memfasilitasi kemajuan penelitian selanjutnya dan pengertian dari pengaruh yang kuat dari pendekatan baru pada komunitas user tanpa memerlukan tindakan untuk menggantikan penginstalan dasar. Kemampuan dalam mengoperasikan keseragaman yang nyata dan transparansi yang berbeda-beda, tempat penyimpanan distribusi informasi. Sangat kompleks, sistem multi komponen akan ditampakkan ke *user* sebagai salah satu kemudahan tunggal.

Kemampuan dalam mengoperasikan harus juga dicapai dalam dimensi lain seperti : 1). Waktu (kemampuan dalam pengoperasian dari sistem lama ke sistem baru). 2). Bahasa (kemampuan dalam pengoperasian multi bahasa). 3). Sintax (mendistribusikan pencarian silang tempat penyimpanan heterogen dan pelayanan. 4). Arti kata (*user* dapat mengakses kelas-kelas dari objek digital yang sama di dalam pengertian dimana mereka memiliki permintaan kata demi kata).

Kemampuan perpustakaan digital harus ditingkatkan untuk mencapai keunggulan dalam pengoperasian dan tidak dapat dihindarkan peningkatan persyaratan permintaan pelayanan oleh user. Keberadaan protokol internet (misalnya http dengan basis World Wide Web) biasanya mengetahui ketidakcukupan dalam

pencarian harus memindahkan melebihi dasar tertentu dari penyebaran protokol dan sistem-sistem tanpa ancaman keadaan yang terdapat di dalam pengaksesan.

### 2.1.3. Dokumen Digital

Keinginan dalam merubah bentuk dokumen ke dalam bentuk yang lebih interaktif merupakan suatu perubahan yang memungkinkan user menikmati sajian informasi dalam bentuk yang berbeda dari sekarang. Satu petunjuk ke masa depan unit konseptual ditemukan didalam ide-ide yang berkembang dari suatu dokumen. Dokumen fisik dapat mengambil beberapa bentuk tetapi dikarakteristikan oleh atribut dasar dari suatu isi dan struktur bagaimana isi ditunjukkan. Struktur mempertinggi arti dengan mensuplai informasi kontekstual. Dokumen juga dapat dikarakteristikan dengan tipe dan gaya. Dokumen yang ada dalam bentuk digital memperoleh hak kekayaan lainnya yaitu format digital. Pemilihan format digital untuk sebuah dokumen memiliki potensial tantangan yang positif maupun negatif secara fungsi dan kegunaan. Isi, struktur dan format dapat dibicarakan secara bebas untuk memperbesar perluasan fungsinya.

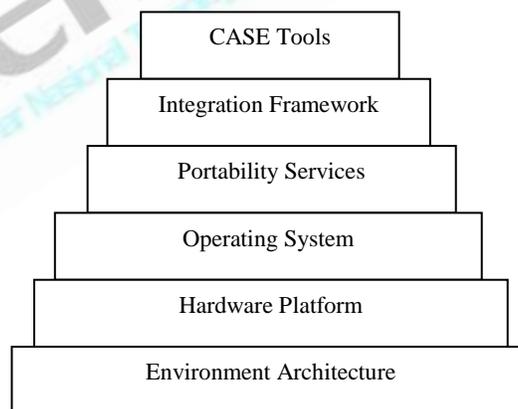
### 2.1.4. Optical Character Recognition (OCR)

OCR selalu difungsikan sebagai alternatif penyimpanan untuk kunci utama, teks dalam bentuk digital. Sebagai definisi, OCR adalah metode pemasukan data pada komputer yang digunakan adalah teknologi scan dan analisa gambar. Untuk identifikasi atau pembaca karakter dalam bentuk tekas sebagai kunci atau keyword. Kode-kode tersebut juga dilihat bentuk digital jika ingin menyiapkan dokumen yang akan diubah bentuknya menjadi bentuk digital. Terlebih dahulu dokumen itu discan dan dicetak dahulu agar tidak terjadi kesalahan.

## 2.2. Konsep CASE

### 2.2.1. Bangunan Blok CASE

Computer Aided Software Engineering (CASE) merupakan sebuah alat tunggal yang mendukung suatu aktivitas rekayasa perangkat lunak yang menunjukkan peralatan, sebuah database, orang-orang, perangkat keras, system operasi, standard an komponen lainnya. Peralatan CASE menambah kotak peralatan perekayasa perangkat lunak, karena CASE menyediakan perekayasa dengan kemampuan untuk mengotomasikan aktivitas manual dan memperbaiki wawasan perekayasaan. Peralatan CASE membantu meyakinkan kualitas yang dirancang sebelum produk dibangun. [2].



Gambar-1 : Blok Bangunan CASE [2]

### 2.2.2. Taksonomi dari Peralatan CASE

Peralatan CASE dapat diklasifikasikan berdasarkan beberapa aspek yang dapat disebutkan, antara lain : 1). Peralatan rekayasa informasi. 2). Peralatan manajemen dan permodelan proses. 3). Peralatan perencanaan

proyek. 3). Peralatan analisis resiko. 4). Peralatan manajemen proyek 5). Peralatan pelacakan kebutuhan. 6). Peralatan manajemen dan alat ukur. 7). Peralatan dokumentasi 8). Peralatan perangkat lunak sistem 9). Peralatan jaminan kualitas. 10). Peralatan manajemen basis data. 11). Peralatan manajemen konfigurasi perangkat lunak. 12). Peralatan rancangan dan analisis. 13). Peralatan PRO/SIM. 14). Peralatan pengembangan dan rancangan antarmuka. 15). Peralatan prototyping. 16). Peralatan pemrograman. 17). Peralatan pengujian dan integrasi. 18). Peralatan analisis static. 19). Peralatan analisis dinamis dan lain-lain. [2].

### 2.2.3. Lingkungan I CASE

Ada beberapa keuntungan I CASE, antara lain : 1). Transfer yang perlahan dari informasi (model, program, dokumen, data) dari satu peralatan ke peralatan lainnya dan dari satu langkah rekayasa perangkat lunak ke langkah berikutnya. 2). Sebuah pengurangan dalam usaha yang diperlukan untuk menjalankan segala aktivitas yang meliputi manajemen konfigurasi perangkat lunak. 3). Sebuah peningkatan dalam pengendalian proyek yang dicapai melalui perencanaan, monitoring dan komunikasi yang lebih baik. 4). Perbaikan koordinasi diantara anggota staf yang bekerja pada sebuah perangkat lunak yang besar. [2].

### 2.2.4. Elemen I CASE

Ada beberapa elemen yang menyusun I-CASE, antara lain : 1). Blok fondasi yang terdiri dari, arsitektur, platform perangkat keras dan perangkat lunak, sistem operasi. 2). Jasa portabilitas. 3). Kerangka kerja integrasi. 4). Peralatan CASE. [2].

## 2.4. Pengembangan Open Source

*Open source* merupakan istilah dalam bahasa Inggris. *Open* berarti buka, sedangkan *source* berarti sumber. Jadi, secara harafiah, *open source* berarti sumber terbuka, atau lebih tepatnya kode sumber (*source code*) yang dibuka untuk umum/publik. Dalam dunia komputer, secara sederhana, semua yang berjalan ketika pengguna memencet tombol *power* adalah program/perangkat lunak/*software*.

Seluruh program yang berjalan di komputer dibuat melalui bahasa pemrograman tertentu. Terdapat beragam pilihan bahasa pemrograman, misalnya: Visual Basic, Pascal, Delphi, C, Java, dan lain sebagainya. Masing-masing bahasa pemrograman mempunyai kelebihan dan kekurangan. Seseorang yang ingin membuat program yang dapat berjalan di komputer harus menentukan bahasa pemrograman yang akan dipakai dan kemudian menuliskannya menurut kaidah yang dibuat oleh bahasa pemrograman yang telah dipilih. Hasil penulisan ini disebut *source code*. *Source code* kemudian akan di kompilasi (*compile*) oleh *compiler* bahasa pemrograman tersebut untuk menghasilkan sebuah *file* yang dapat dieksekusi/dijalankan oleh komputer.

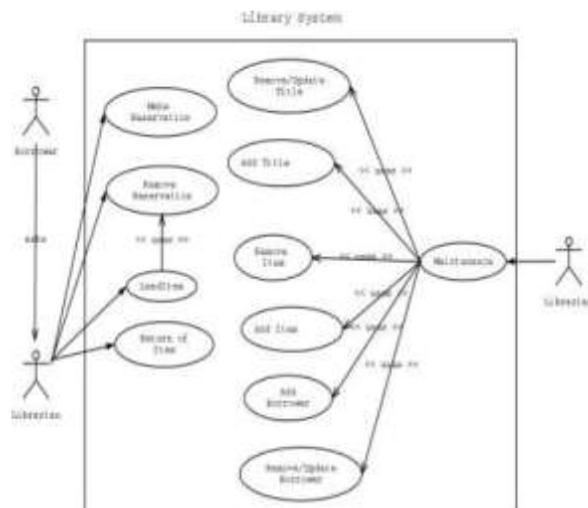
Teknologi OSS telah menjadi building blocks dari aplikasi/layanan Internet, antara lain : 1). Mail server: sendmail, qmail, postfix. 2). Web server: apache. 3). ftp server: wu-ftp, proftpd 4). dB server: mysql, postgres. 5). Blog software: wordpress. 6). Wiki software: mediawiki. 7). Content management system (CMS).

Proyek-proyek Open Source telah ada sejak 1960-an, tetapi cenderung mengalami popularitas yang baru ditemukan saat ini disebabkan oleh meningkatnya penggunaan internet sebagai media komunikasi dan informasi. Proses rekayasa perangkat lunak meliputi: 1). Pedefinisian Persyaratan. 2). Rancangan tingkat sistem. 3). Rancangan lengkap. 4). Implementasi 5). Integrasi, 6). Bidang Pengujian dan 7). Dukungan. Perangkat lunak model open source mendukung semua aspek dari proses ini dan menyediakan insinyur dengan metodologi untuk mengikuti untuk menghasilkan produk berkualitas tinggi menerapkan persyaratan klien. [3].

### 3. PENGEMBANGAN MODEL DAN PEMBAHASAN

#### 3.1. Model I-CASE

Pengembangan perpustakaan digital menjadi keharusan dimana sistem perpustakaan digital memberikan fleksibilitas kepada pemakai tentang akses informasi referensi yang dibutuhkan.



Gambar-2 : Library System Use-Case Diagram [3].

Proyek-proyek Open Source, seperti proyek-proyek *proprietary*, membutuhkan tingkat analisis persyaratan dan pemodelan untuk mengimplementasikan solusi. UML adalah pendekatan definitif untuk membangun model pembangunan berbasis proses yang menggabungkan suara dan arsitektur yang kuat. Spesifikasi ini memungkinkan pengembang menggunakan notasi standar untuk model komponen sistem, perilaku, dan pengguna. Persyaratan analisis dilakukan menyangkut masalah perangkat lunak atau sistem, kemudian dimodelkan melalui UML dan disajikan sebagai solusi konseptual. Spesifikasi UML tidak menentukan metodologi yang tepat atau proses yang harus digunakan untuk menyelesaikan masalah, tetapi menguraikan pemahaman analisis masalah untuk terjemahan mudah oleh tim desain. [3].

UML mendefinisikan notasi dan semantik untuk jenis berikut solusi masalah: 1). Interaksi pengguna : menggambarkan batas dan interaksi antara pengguna dan sistem. 2). Interaksi atau Kolaborasi Model : menjelaskan bagaimana objek dalam sistem berinteraksi untuk menyelesaikan tugas. 3). Dynamic Model : legiatan grafik menggambarkan alur kerja sistem akan capai. 4). Logika atau Kelas Model : menjelaskan kelas dan objek sistem. 5). Komponen Fisik Model : menjelaskan perangkat lunak, dan kadang-kadang perangkat keras sistem. 6). Fisik Deployment Model : menjelaskan arsitektur sistem fisik dan penyebaran komponen pada perangkat keras sistem. [3].

UML menyediakan model perangkat lunak Open Source dengan kemampuan untuk berkembang dari aplikasi solusi sederhana untuk penggunaan pribadi untuk aplikasi skala besar menyelesaikan persyaratan ukuran sistem industri. Ini adalah proses yang UML menyediakan pengembangan organisasi yang memungkinkan mereka untuk kembali fokus pengembangan usaha mereka terhadap pemahaman mudah masalah ini kompleks. [3].

#### 3.2. Analisis Perancangan Sistem

Pola pengembangan model open source diasumsikan pada keinginan pengguna untuk dapat menggunakan berbagai macam perangkat lunak. Pola pengembangan ini merupakan pendekatan pragmatis untuk mereduksi biaya atau meningkatkan kualitas dari perangkat lunak. Singkatnya, pembuat program

memberikan *source code* untuk digunakan pengguna, dan pengguna memiliki kebebasan untuk mengubah dan memodifikasi kode program untuk meningkatkan kualitas program. Pola *sharing software* seperti ini tentu hanya dapat terjadi diantara pengguna yang juga memiliki kemampuan teknis untuk mengubah atau memodifikasi kode program, atau yang disebut sebagai pengguna yang “prosumen” (produsen-konsumen). Model pengembangan seperti ini yang menjadi ide dasar dari *Free/Open Source Software*.

Pembuatan sebuah perangkat lunak open source biasanya terdiri 3 fase, yaitu: [4].

#### 1. Fase awal (*the initial development phase*)

Pembuatan program dapat dilakukan oleh *initial developer* (individu atau perusahaan) untuk kebutuhan kerjanya. Hasil program tersebut kemudian diberikan kepada orang lain. Jika ada yang tertarik dengan programnya, *initial developer* akan memberikan program dan *source code*-nya untuk dikembangkan lagi.

#### 2. Fase pertumbuhan (*the growth phase*)

Pada fase ini, program yang dibuat oleh pengembang awal telah dimodifikasi melalui kontribusi orang lain. Kontribusi ini membuat program lebih generic dan fungsional.

#### 3. Fase pengorganisasian (*organization phase*)

Tahap ini dapat tercapai ketika program yang dibuat dan dikembangkan dengan kontribusi yang luas dari berbagai pihak telah menjadi standar baku untuk digunakan dalam suatu komunitas atau kelompok. Pada tahap ini, biaya pemeliharaan sistem menjadi tanggung jawab komunitas.

### 4. SIMPULAN

Pengembangan model I CASE pada open source perpustakaan digital merupakan suatu topik yang sangat menarik perhatian karena berasal dari permasalahan kepraktisan atas penggunaan perangkat lunak dengan alat bantuan yang terintegrasi dengan kebutuhan pemakai. Model I CASE dengan pendekatan UML (Unified Modelling Language) dapat menjadi solusi alternatif dalam merancang perpustakaan digital berbasis open source. Oleh karena itu, diharapkan model I CASE dengan memperhatikan aspek kepraktisan dapat menjadi perhatian bagi perekayasa perangkat lunak khususnya berkaitan dengan perpustakaan digital berbasis open source.

### DAFTAR PUSTAKA

- [1] <http://journal.unair.ac.id/filerPDF/PERPUSTAKAAN%20DIGITAL.pdf>, diakses tanggal 12 Maret 2011
- [2] Roger S. Pressman, “Software Engineering : A Practitioner’s Approach”. Fourth Edition, Mc Graw-Hill Companies., 1997, page 806-814.
- [3] <http://ldp.org/LDP/LG/issue67/gilliam.html>, diakses tanggal 14 Maret 2011.
- [4] .Adi Indrayanto, Budi Rahardjo, Andika Triwidada, Zaki Akhmad, Indra, Syarifudin, “Panduan Penelitian Open Source Software”, Versi 1.00, 2007, hal. 31-32, the Creative Commons Attribution Attribution-NonCommercial-NoDerivs V2.5, (<http://creativecommons.org>), diakses tanggal 16 Maret 2011.