

PERBANDINGAN MODEL WATERFALL DAN PROTOTYPING UNTUK PENGEMBANGAN SISTEM INFORMASI

RANI SUSANTO, ANNA DARA ANDRIANA
Program Studi Teknik Informatika, Fakultas Teknik dan Ilmu Komputer
Universitas Komputer Indonesia

System Development Life Cycle atau yang lebih dikenal dengan istilah SDLC adalah metodologi umum yang digunakan untuk mengembangkan sistem informasi. SDLC terdiri dari beberapa fase yang dimulai dari fase perencanaan, analisis, perancangan, implementasi hingga pemeliharaan sistem. Konsep SDLC ini mendasari berbagai jenis model pengembangan perangkat lunak untuk membentuk suatu kerangka kerja untuk perencanaan dan pengendalian pembuatan sistem informasi. Model-model SDLC yang sering digunakan antara lain *Waterfall* dan *Prototyping*. Pembahasan mengenai model pengembangan perangkat lunak ini terdapat di salah satu materi di mata kuliah Analisis dan Desain Sistem Informasi yang ada di program studi Teknik Informatika yaitu di Bab 2 Pengembangan Sistem Informasi.

Hasil perbandingan dari kedua model tersebut menyatakan bahwa model *waterfall* lebih cocok untuk sistem atau perangkat lunak yang bersifat generik, artinya sistem dapat diidentifikasi semua kebutuhannya dari awal dengan spesifikasi yang umum. Sesuai dengan karakteristik model ini, contoh topik Tugas Akhir/Skripsi yang cocok jika menggunakan model ini adalah tugas akhir/skripsi yang memiliki tujuan untuk membangun sebuah sistem dari awal yang mengumpulkan kebutuhan sistem yang akan dibangun sesuai dengan topik penelitian yang dipilih sampai dengan produk tersebut diuji. Sedangkan *prototyping* lebih cocok untuk sistem atau perangkat lunak yang bersifat *customize*, artinya *software* yang diciptakan berdasarkan permintaan dan kebutuhan (bahkan situasi atau kondisi) tertentu. Sesuai dengan karakteristik model ini contoh topik Tugas Akhir/Skripsi yang cocok jika menggunakan model ini adalah tugas akhir/skripsi yang memiliki tujuan untuk mengimplementasikan sebuah metode atau algoritma tertentu pada suatu kasus.

Keywords : *System Development Life Cycle, Waterfall, Prototype*

PENDAHULUAN

System Development Life Cycle atau yang lebih dikenal dengan istilah SDLC adalah metodologi umum yang digunakan untuk mengembangkan sistem informasi. SDLC terdiri dari beberapa fase yang dimulai dari fase perencanaan, analisis, perancangan, implementasi hingga pemeliharaan sistem. Konsep SDLC ini mendasari berbagai jenis

model pengembangan perangkat lunak untuk membentuk suatu kerangka kerja untuk perencanaan dan pengendalian pembuatan sistem informasi. Model-model SDLC yang sering digunakan antara lain *Waterfall* dan *Prototyping*.

Model *Waterfall* adalah salah satu model SDLC yang sering digunakan atau sering disebut juga dengan model konvensional

atau *classic life cycle*. Model ini menggunakan pendekatan sistematis dan urutan dimulai dari level kebutuhan sistem lalu menuju ke tahapan analisis, desain, *coding*, *testing/verification* dan *maintanance*. Selain *waterfall* terdapat model yang bernama *Prototyping*, pendekatan model ini dimulai dari pengumpulan kebutuhan, pembangunan *prototype*, pengkodean, implementasi hingga pemeliharaan. Model *Waterfall* dan *Prototyping* memiliki pendekatan yang hampir sama sehingga dalam proses pengembangan sistem perlu diketahui kasus atau sistem yang memiliki karakteristik seperti apa yang harus menggunakan masing-masing model tersebut. Pembahasan mengenai model pengembangan perangkat lunak ini terdapat di salah satu materi di mata kuliah Analisis dan Desain Sistem Informasi yang ada di program studi Teknik Informatika yaitu di Bab 2 Pengembangan Sistem Informasi.

Berdasarkan permasalahan tersebut maka perlu dilakukan suatu perbandingan dari masing-masing model perangkat lunak tersebut yang dapat membantu dalam menentukan model yang paling sesuai untuk proses pengembangan perangkat lunak.

TINJAUAN PUSTAKA

1. Jenis Penelitian

Jenis-jenis penelitian dapat diklasifikasikan berdasarkan tujuan, metode, tingkatan eksplanasi dan analisis data.

- a. Berdasarkan Tujuan : penelitian dasar (*basic research*) dan penelitian terapan (*applied research*).
- b. Berdasarkan Tingkat Eksplanasi : Metode Deskriptif, Komparatif dan Asosiatif
- c. Berdasarkan Analisis dan Jenis Data : Kuantitatif, Kualitatif, Gabungan

2. System Development Life Cycle

SDLC (*Systems Development Life Cycle*, Siklus Hidup Pengembangan Sistem) atau

Systems Life Cycle (Siklus Hidup Sistem), dalam rekayasa sistem dan rekayasa perangkat lunak, adalah proses pembuatan dan perubahan sistem serta model dan metodologi yang digunakan untuk mengembangkan sistem-sistem tersebut. Konsep ini umumnya merujuk pada sistem komputer atau informasi. SDLC juga merupakan pola yang diambil untuk mengembangkan sistem perangkat lunak, yang terdiri dari tahap-tahap: rencana (*planning*), analisis (*analysis*), desain (*design*), implementasi (*implementation*), uji coba (*testing*) dan pengelolaan (*maintenance*) [1]. Dalam rekayasa perangkat lunak, konsep SDLC mendasari berbagai jenis metodologi pengembangan perangkat lunak. Metodologi-metodologi ini membentuk suatu kerangka kerja untuk perencanaan dan pengendalian pembuatan sistem informasi, yaitu proses pengembangan perangkat lunak. Terdapat 3 jenis metode siklus hidup sistem yang paling banyak digunakan, yakni: siklus hidup sistem tradisional (*traditional system life cycle*), siklus hidup menggunakan prototyping (*life cycle using prototyping*), dan siklus hidup sistem orientasi objek (*object-oriented system life cycle*). [1]

SDLC juga merupakan metodologi umum dalam pengembangan sistem yang menandai kemajuan usaha analisis dan desain. Fase-fase yang terdapat didalam SDLC meliputi :

- a. Perencanaan Sistem (*Systems Planning*)
- b. Analisis Sistem (*Systems Analysis*)
- c. Perancangan Sistem (*Systems Design*)
- d. Implementasi Sistem (*Systems Implementation*)
- e. Pemeliharaan Sistem (*Systems Maintenance*)

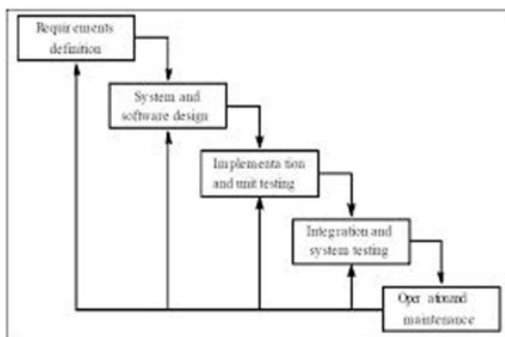
Konsep SDLC mendasari model pengembangan perangkat lunak lainnya. Model pengembangan perangkat lunak tersebut antara lain *waterfall*, *prototype*, *iterative*, *spiral*, *rapid application development* (RAD) dan lainnya. Dalam

penelitian ini hanya akan dibahas dua model yaitu *Waterfall* dan *Prototyping*.

3. Model Waterfall

Model *waterfall* adalah model pengembangan perangkat lunak yang paling sering digunakan. Model pengembangan ini bersifat linear dari tahap awal pengembangan system yaitu tahap perencanaan sampai tahap akhir pengembangan system yaitu tahap pemeliharaan. Tahapan berikutnya tidak akan dilaksanakan sebelum tahapan sebelumnya selesai dilaksanakan dan tidak bisa kembali atau mengulang ke tahap sebelumnya.

Tahapan dari model pengembangan *waterfall* ini digambarkan pada gambar 1. Model Pengembangan *Waterfall*.



Gambar 1. Model Pengembangan *Waterfall*

4. Model Prototyping

Model *prototyping* merupakan suatu teknik untuk mengumpulkan informasi tertentu mengenai kebutuhan-kebutuhan informasi pengguna secara cepat. Berfokus pada penyajian dari aspek-aspek perangkat lunak tersebut yang akan nampak bagi pelanggan atau pemakai. Prototipe tersebut akan dievaluasi oleh pelanggan/pemakai dan dipakai untuk menyaring kebutuhan pengembangan perangkat lunak. Tahapan

model pengembangan *prototyping* ini digambarkan pada gambar 2. Model Pengembangan *Prototyping*.



Gambar 2. Model Pengembangan Prototyping

TUJUAN DAN MANFAAT PENELITIAN

1. Tujuan Penelitian

Tujuan dari penelitian ini adalah :

- Mengetahui perbandingan antara Model Waterfall dan Prototyping berdasarkan karakteristiknya
- Mempermudah dalam pemilihan model sesuai dengan topik Tugas Akhir/Skripsi yang dipilih
- Menjadi acuan bahan ajar dari mata kuliah yang ada di program studi Teknik Informatika yaitu Analisis dan Desain Sistem Informasi atau disebut juga dengan ADSI tepatnya pada Bab 2 Pengembangan Sistem Informasi.

2. Manfaat Penelitian

Manfaat dari penelitian ini yaitu :

- Menjadi acuan bahan ajar di mata kuliah Analisis dan Desain Sistem Informasi untuk Bab 2 Pengembangan Sistem Informasi.
- Membantu mahasiswa dalam memilih model yang sesuai dengan topik tugas akhir/skripsi yang dipillihnya.

METODE PENELITIAN

Pendekatan yang digunakan pada penelitian ini adalah pendekatan kuantitatif dengan menggunakan metode deskriptif, yaitu melakukan studi komparatif untuk membandingkan fenomena-fenomena yang ditemukan dan membuat klasifikasi yang bersumber pada suatu standar. Adapun langkah-langkah penelitian meliputi :

- a. Memilih dan merumuskan masalah
- b. Menelusuri sumber-sumber kepustakaan
- c. Melakukan observasi dan wawancara
- d. Menginterpretasikan kondisi penelitian dengan data yang diperoleh
- e. Melakukan analisis data
- f. Membuat perbandingan model yang diinginkan sesuai dengan karakteristik model pengembangan dan sistem yang sedang berjalan
- g. Membuat laporan penelitian ilmiah.

HASIL DAN PEMBAHASAN

Berdasarkan hasil analisis dan perbandingan yang telah dilakukan, maka didapatkan hasil.

Tabel 1. Perbandingan Model *Waterfall* dan *Prototyping*

| Tahapan Pengembangan perangkat Lunak | <i>Waterfall</i> | <i>Prototyping</i> |
|--|--|---|
| Perencanaan Sistem (<i>Systems Planning</i>) | Berawal dari kebutuhan | Berawal dari kebutuhan |
| Analisis Sistem (<i>Systems Analysis</i>) | Kebutuhan data harus dianalisis diawal secara lengkap dan menyeluruh | Kebutuhan data dapat ditambah ataupun dikurangi sesuai dengan kebutuhan <i>user</i> , ketika dilakukan <i>testing</i> . |

| Tahapan Pengembangan perangkat Lunak | <i>Waterfall</i> | <i>Prototyping</i> |
|--|--|---|
| | Perubahan data ataupun <i>funksional</i> akan merubah keseluruhan proses pada tahapan berikutnya. | Perubahan dapat dilakukan selama sistem atau perangkat lunak masih dalam bentuk <i>prototype</i> |
| Perancangan Sistem (<i>Systems Design</i>) | Testing dilakukan ketika semua tahapan pada model sudah selesai. | Testing dapat dilakukan ketika <i>prototype</i> telah dibangun, sehingga hasil <i>testing</i> dapat merubah rancangan sistem. |
| | Tidak dapat memberikan gambaran yang jelas mengenai sistem yang dibangun, karena sistem bisa dilihat jika semua tahapan telah dilakukan. | Memberikan <i>prototype</i> sebagai gambaran sistem yang akan dibangun, sehingga <i>user</i> dapat melihat dan berinteraksi langsung dengan gambaran sistem. <i>User</i> berperan aktif dalam pengembangan sistem Sistem yang dibangun akan sesuai dengan keinginan <i>user</i> |

| Tahapan Pengembangan perangkat Lunak | Waterfall | Prototyping |
|--|---|--|
| Implementasi Sistem (Systems Implementation) | Menerapkan proses perancangan yang baik | Tidak menerapkan proses perancangan yang baik |
| | Evaluasi dilakukan ketika sistem telah dibangun | Evaluasi dilakukan ketika <i>prototype</i> telah dibangun. |
| | Mengedepankan kebutuhan fungsional sistem | Mengedepankan aspek kenyamanan <i>user</i> |
| Pemeliharaan Sistem (Systems Maintenance) | Dilakukan sesuai kesepakatan | Dilakukan sesuai kesepakatan |

KESIMPULAN DAN SARAN

1. Kesimpulan

Berdasarkan analisis dan perbandingan yang telah dilakukan, maka dapat disimpulkan bahwa :

- Dapat diketahui karakteristik dari kedua model pengembangan perangkat lunak *Waterfall* dan *Prototype*
- Model pengembangan *Waterfall* cocok digunakan untuk sistem atau perangkat lunak yang bersifat generik, artinya sistem dapat diidentifikasi semua kebutuhannya dari awal dengan spesifikasi yang umum serta sesuai untuk tugas akhir/skripsi yang memiliki tujuan untuk membangun sebuah sistem dari awal yang mengumpulkan kebutuhan sistem yang akan dibangun sesuai dengan topik penelitian yang dipilih sampai dengan produk tersebut diuji
- Model pengembangan *Prototype* lebih cocok untuk sistem atau perangkat

lunak yang bersifat *customize*, artinya *software* yang diciptakan berdasarkan permintaan dan kebutuhan (bahkan situasi atau kondisi) tertentu dan sesuai untuk tugas akhir/skripsi yang memiliki tujuan untuk mengimplementasikan sebuah metode atau algoritma tertentu pada suatu kasus.

2. Saran

Kajian di dalam penelitian ini masih memiliki banyak kekurangan, berikut adalah hal-hal yang dapat dilakukan untuk penelitian lebih lanjut.

- Melakukan perbandingan model pengembangan perangkat lunak yang lainnya sebagai pelengkap bahan ajar Analisis dan Desain Sistem Informasi di Program Studi Teknik Informatika.
- Menjadikan hasil perbandingan model perangkat lunak yang lainnya sebagai bahan acuan yang dapat digunakan untuk topik-topik tugas akhir/skripsi di program studi Teknik Informatika.

DAFTAR PUSTAKA

- Britton, Carol; Jill Doake (2001). *Object-Oriented Systems Development*. McGraw-Hill. hlm. 27. ISBN 0-07-709544-8.
- HM., Yogyanto, Analisis dan Disain Sistem Informasi : Pendekatan Terstruktur, Penerbit Andi Offset, Yogyakarta, 1995.
- Ritchie, Rose and Bernie Michalik. "Converting a project from a waterfall to an iterative approach." developerWorks. 15 May 2006. <<http://www128.ibm.com/developerworks/rational/library/may06/ritchie/index.html>>.
- Sommerville, Ian, *Software Engineering*, Addison-Wesley, 2007

