

ANALISIS RESIKO PADA PENGEMBANGAN PERANGKAT LUNAK YANG MENGGUNAKAN METODE WATERFALL DAN PROTOTYPING.

Fandli Supandi, Wahit Desta P, Yuni Ambar S. dan Mat Sudir

Program Magister Teknik Informatika,
Universitas Amikom Yogyakarta

Email: fandli.supandi10@gmail.com

Abstrak

Pada pengembangan perangkat lunak tentunya kita selalu menggunakan tahapan – tahapan dalam proses pengembangannya, salah satu metode yang paling sering digunakan sebagai tahapan dalam pengembangan perangkat lunak adalah metode System Development Life Cycle (SDLC). Dalam SDLC itu sendiri terdapat beberapa metode di dalamnya diantaranya yaitu metode waterfall dan prototyping. Dalam pengembangan perangkat lunak yang menggunakan metode waterfall dan prototyping juga terkadang berakhir dengan kegagalan, hal tersebut salah satunya dipengaruhi oleh antisipasi terhadap resiko yang mungkin terjadi saat pengembangan perangkat lunak tersebut tidak baik dikarenakan analisis terhadap resiko yang mungkin terjadi belum tepat. Oleh karena hal tersebut maka dalam makalah ini penulis akan menjabarkan analisis resiko yang mungkin terjadi pada pengembangan perangkat lunak yang menggunakan metode waterfall dan prototyping berdasarkan karakteristik dari masing – masing metode tersebut.

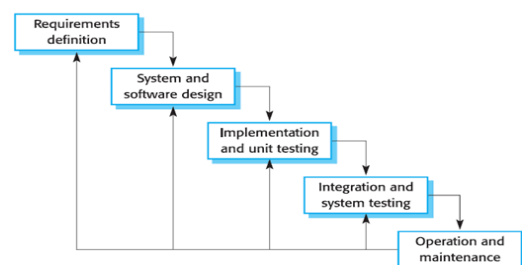
Kata-kata kunci: perangkat lunak, waterfall, prototyping, analisis resiko

1. Pendahuluan

Perkembangan teknologi seperti sekarang ini sangat bermanfaat dan telah mengalami perkembangan sangat pesat khususnya dalam bidang informatika. Dimana dengan perkembangan ini, dimana banyak sekali perangkat lunak yang dikembangkan maupun dibuat. tentunya dalam pengembangan perangkat lunak tersebut memiliki tahapan – tahapan dalam proses pembuatannya dimana yang sering kita kenal dengan System Development Life Cycle (SDLC) yang diantaranya yaitu metode waterfall dan metode prototyping yang membuat proses pengembangan perangkat lunak tersebut menjadi lebih terstruktur dengan baik.

Namun pengembangan suatu proyek perangkat lunak yang sudah menggunakan metodepun sering kali mengalami kegagalan, hal tersebut dikarenakan kurangnya antisipasi terhadap resiko – resiko yang dapat terjadi dalam penggunaan metode tersebut. Dari permasalahan diatas maka pada kali ini penulis akan memaparkan analisis resiko yang dapat terjadi dalam penggunaan metode waterfall dan prototyping.

Metode air terjun atau yang sering disebut metode waterfall sering dinamakan siklus hidup klasik (*classic life cycle*), dimana hal ini menggambarkan pendekatan yang sistematis dan juga berurutan pada pengembangan perangkat lunak, dimulainya dengan spesifikasi kebutuhan pengguna lalu berlanjut melalui tahapan-tahapan perencanaan (*planning*), permodelan (*modeling*), konstruksi (*construction*), serta penyerahan sistem ke para pelanggan/pengguna (*deployment*), yang diakhiri dengan dukungan pada perangkat lunak lengkap yang dihasilkan (Pressman, 2012). Tahapan metode waterfall dapat dilihat pada gambar di bawah ini. [1]



Gambar 1. metode waterfall.

2. Landasan Teori

2.1. Metode waterfall

Kelebihan metode waterfall [2]:

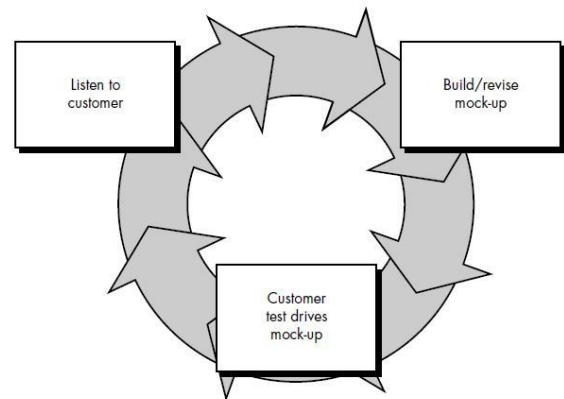
1. Kualitas dari sistem yang dihasilkan akan baik. Ini dikarenakan oleh pelaksanaannya secara bertahap.
2. Metode ini merupakan salah satu metode yang sangat baik digunakan. Dikarenakan dalam prosesnya semakin rinci dan di tinjau kembali setiap tahapannya saat pengujian system.
3. Document pengembangan system sangat terorganisir (*disusun dan diatur di suatu kesatuan*), karena setiap fase harus terselesaikan dengan lengkap sebelum melangkah ke fase berikutnya. Jadi setiap fase atau tahapan akan mempunyai dokumen tertentu

Kekurangan metode *waterfall*[2]:

1. Diperlukan majemen yang baik, karena proses pengembangan tidak dapat dilakukan secara berulang sebelum terjadinya suatu produk.
2. Kesalahan kecil akan menjadi masalah besar jika tidak diketahui sejak awal pengembangan yang berakibat pada tahapan selanjutnya.
3. Pelanggan sulit menyatakan kebutuhan secara eksplisit (*gambang, tegas, terus terang*) sehingga tidak dapat mengakomodasi (*menyediakan sesuatu untuk memenuhi kebutuhan*) ketidakpastian pada saat awal pengembangan.
4. Pelanggan harus sabar, karena pembuatan perangkat lunak akan dimulai ketika tahap desain sudah selesai. Sedangkan pada tahap sebelum desain bisa memakan waktu yang lama.
5. Pada kenyataannya, jarang mengikuti urutan sekuensial (*runtunan*) seperti pada teori. Iterasi (*perulangan*) sering terjadi menyebabkan masalah baru.

2.2. Metode Prototyping

Metode Prototype merupakan suatu paradigma baru dalam metode pengembangan perangkat lunak dimana metode ini tidak hanya sekedar evolusi dalam dunia pengembangan perangkat lunak, tetapi juga merevolusi metode pengembangan perangkat lunak yang lama yaitu sistem sekuensial yang biasa dikenal dengan nama SDLC atau waterfall development model.[1]



Gambar 2. Metode Prototyping

Dalam Model Prototype, prototype dari perangkat lunak yang dihasilkan kemudian dipresentasikan kepada pelanggan, dan pelanggan tersebut diberikan kesempatan untuk memberikan masukan sehingga perangkat lunak yang dihasilkan nantinya betul-betul sesuai dengan keinginan dan kebutuhan pelanggan. Perubahan dan presentasi prototype dapat dilakukan berkali-kali sampai dicapai kesepakatan bentuk dari perangkat lunak yang akan dikembangkan.

Kelebihan metode *prototyping*[2]:

1. Dapat mengakomodir (*menyediakan sesuatu untuk memenuhi kebutuhan*) perangkat lunak yang spesifikasi kebutuhannya belum rinci.
2. Dapat diimplementasikan di dalam konteks setiap model perangkat lunak.
3. Lebih menghemat waktu dalam pengembangannya.
4. Pelanggan berperan aktif dalam pengembangan sistem.

Kekurangan metode *prototyping*[2]:

1. Pelanggan kadang tidak melihat atau menyadari bahwa perangkat lunak yang ada belum mencantumkan kualitas perangkat lunak secara keseluruhan dan juga belum memikirkan kemampuan pemeliharaan untuk jangka waktu yang lama.
2. Pengembang biasanya ingin cepat menyelesaikan proyek, sehingga dalam proses pengerjaannya seringkali mengesampingkan kualitas dari proses tersebut.

3. Hasil dan Pembahasan

Berdasarkan penjelasan mengenai metode *waterfall* dan *prototyping* diatas dan menimbang dari kelebihan dan kekurangan dari tiap metode maka penulis menyimpulkan beberapa analisis terhadap resiko serta pengaruhnya terhadap proyek, teknikal, serta bisnis.[3]

3.1. Analisis Resiko Metode *Waterfall*

Resiko	Pengaruh	Penjelasan
Sumberdaya manusia	Proyek	- SDM yang berpengalaman meninggalkan proyek sebelum waktu proyek selesai. - SDM baru tidak tahan dengan tekanan proyek sehingga meninggalkan proyek sebelum proyek selesai.
Perubahan Manajemen	Proyek	Jika terjadi perubahan manajemen dalam proyek.
Kebutuhan sistem	Proyek	Kebutuhan sistem berubah akibat perubahan manajemen, atau kebutuhan berubah ditengah pengerjaan proyek dikarenakan hal tertentu.
Penundaan	Proyek dan Teknikal	Pengerjaan dalam setiap tahapan yang tidak sesuai jadwal.
Spesifikasi kebutuhan	Proyek dan Teknikal	Kesalahan dalam mengidentifikasi spesifikasi kebutuhan awal, sehingga membuat proses akan lebih lama.
Case Tool dibawah kinerja	Proyek dan Teknikal	Case Tool yang mendukung proyek tersebut tidak bekerja seperti yang diharapkan.
Biaya	Proyek dan Bisnis	Kenaikan biaya akibat keterlambatan dalam pengerjaan setiap tahap proyek.
Perubahan Teknologi	Bisnis	Teknologi yang mendasari pembuatan sistem diganti ke teknologi baru.
Persaingan bisnis produk	Bisnis	Produk yang kompetitif dipasarkan sebelum proyek selesai.
Ketidaksiuaian	Bisnis	Produk yang dihasilkan tidak sepenuhnya sesuai dengan strategi bisnis yang diinginkan oleh pelanggan

3.2. Analisis Resiko Metode *Prototyping*

Resiko	Pengaruh	Penjelasan
Sumberdaya manusia	Proyek	SDM yang berpengalaman meninggalkan proyek sebelum waktu proyek selesai.
Kebutuhan sistem	Proyek dan Teknikal	Kebutuhan sistem tidak terdefinisi dengan baik dikarenakan komunikasi yang

	teknikal	buruk antara pengembangan dan pelaksanaan.
Meremehkan kurun	Proyek dan Teknikal	Ukuran sistem yang diremehkan atau ada beberapa hal ketika menganalisis diremehkan sehingga menimbulkan kesalahan pada hasil proyek.
Case Tool dibawah kinerja	Proyek dan Teknikal	Case Tool yang mendukung proyek tersebut tidak bekerja seperti yang diharapkan.
Perubahan Teknologi	Bisnis	Kesalahan dalam mengidentifikasi spesifikasi kebutuhan awal, sehingga membuat proses akan lebih lama.
Case Tool dibawah kinerja	Proyek dan Teknikal	Case Tool yang mendukung proyek tersebut tidak bekerja seperti yang diharapkan.
Perubahan Teknologi	Bisnis	Teknologi yang mendasari pembuatan sistem diganti ke teknologi baru.
Persaingan bisnis produk	Bisnis	Produk yang kompetitif dipasarkan sebelum proyek selesai.

4. Kesimpulan dan Saran

Berdasarkan karakteristik serta kelebihan dan kekurangan dari metode *waterfall* dan *prototyping* maka penulis menyimpulkan beberapa resiko yang mungkin terjadi dalam pengembangan proyek perangkat lunak serta pengaruhnya terhadap proyek, teknikal. Serta bisnis. Sehingga dapat terhindar dari kegagalan proyek perangkat lunak.

Diharapkan agar penulis yang lainnya dapat memaparkan analisis resiko yang mungkin terjadi pada metode yang lainnya yang juga merupakan bagian dari System Development Life Cycle (SDLC).

5. Ucapan terima kasih

Terima kasih kami ucapkan kepada ibu Prof. Dr. Ema Utami, S.Si., M.Kom yang telah membimbing kami dalam penyusunan makalah ini. Juga kepada Universitas Amikom Yogyakarta dan Seminar Nasional Dinamika Informatika (SENADI) Universitas PGRI Yogyakarta.

Referensi

- [1] Roger, S. Pressman, Ph.D., 2012, Rekayasa Perangkat Lunak (Pendekatan Praktisi) Edisi 7 : Buku 1, Yogyakarta: Andi.

- [2] [2] Pressman R S, 2010, "Rekayasa Perangkat Lunak", Yogyakarta, Andi.
- [3] [3] Primawati A, Dkk, 2013, "Analisis manajemen proyek perangkat lunak sistem informasi akuntansi pada biro sistem informasi pt. X". Fakultas Teknik Matematika & IPA, Pogram Studi Teknik Informatika, Universitas Indraprasa PGRI