

AUTOMATICS SOFTWARE ANALISYS

Arief Setyanto

Dosen STMIK AMIKOM Yogyakarta

Abstract

Dokumen awal yang dijadikan pedoman adalah dokumen View Point Oriented Document (VORD). Dokumen VORD merupakan hasil kerja dari seorang analis dalam mendefinisikan kebutuhan fungsional dari stake holder yang terdiri dari para calon pemakai perangkat lunak.

Keywords : VORD

Pendahuluan

Rekayasa perangkat lunak merupakan salah satu disiplin rekayasa yang memberikan hasil abstrak. Kondisi ini berbeda dengan disiplin rekayasa yang seperti elektronika mesin maupun sipil dimana hasil proses rekayasa memberikan bentuk nyata. Pengukuran volume sebuah hasil rekayasa yang abstrak menjadi tidak semudah pengukuran volume hasil rekayasa yang memberikan hasil konkret. Pengukuran volume pekerjaan rekayasa perangkat lunak yang abstrak hanya didasarkan kepada kegunaan dari hasil rekayasa. Volume dalam ukuran byte panjangnya kode program terkadang tidak linier dengan kegunaan dari hasil rekayasa perangkat lunak.

Bahan baku yang relatif tida diperlukan pada rekayasa perangkat lunak juga turut berperan dalam mempersulit pengukuran volume pekerjaan. Sementara rekayasa di bidang lain salah satu ukuran yang dapat dijadikan pedoman adalah jumlah bahan baku yang digunakan.

Sifat sifat perangkat lunak diatas menyebabkan pengukuran volume pekerjaan menjadi lebih sulit dilakukan. Akibatnya adalah perkiraan biaya pengembangan perangkat lunak juga akan menjadi tidak mudah dilakukan. Makalah ini akan mencoba memberikan satu alternatif analisis biaya pengembangan perangkat lunak yang akan dicoba di automasi. Dokumen awal yang dijadikan pedoman adalah

dokumen *View Point Oriented Document* (VORD). Dokumen VORD merupakan hasil kerja dari seorang analis dalam mendefinisikan kebutuhan fungsional dari stake holder yang terdiri dari para calon pemakai perangkat lunak.

Pembahasan Permasalahan Analisis Kebutuhan

Pekerjaan yang memiliki posisi yang cukup menentukan dalam urutan pekerjaan pembangunan perangkat lunak adalah proses analisis kebutuhan pengguna. Rekayasa kebutuhan (*requirement engineering*) merupakan tahapan pengumpulan kebutuhan layanan dari software yang akan dibangun oleh masing masing pengguna yang terlibat dalam sebuah sistem.

View Point Oriented Document (VORD)

Salah satu bentuk dokumen yang dapat digunakan untuk mencatat kebutuhan pengguna adalah VORD. Pembuatan dokumen ini diawali dengan mengoleksi user user yang akan berperan dalam menjalankan sistem perangkat lunak yang akan di bangun. Setelah mengumpulkan daftar pengguna yang selanjutnya akan disebut *view point*, kepadanya akan didokumentasikan service apa saja yang akan diberikan oleh software terhadap view point tersebut. Selanjutnya semua service dan semua view point akan diberikan definisi detailnya.

Penggunaan dokumen ini dapat membantu analisis untuk mengerjakan tugas rekayasa kebutuhan pengguna dengan lengkap. Walaupun pada prakteknya proses ini tidak selalu berjalan dengan baik karena ada saja beberapa service yang tercecer tidak ter-amati oleh analis. Kejadian ini merupakan sesuatu yang biasa terjadi dalam rekayasa perangkat lunak seperti disebutkan dalam beberapa buku teori.

SOLUSI

VORD Untuk Bahan Membuat USE CASE

Salah satu dokumen dalam satu set dokumen UML yang dibuat pada perancangan perangkat lunak berorientasi obyek – Object oriented analysis (OOA) adalah Use case. Dokumen ini mewakili gambaran yang akan diberikan kepada pengguna (AKTOR) tentang apa saja yang dapat dilakukan ketika perangkat lunak yang direncanakan telah dibuat.

Dengan modal dokumen VORD dapat dibuatkan sebuah proses automasi untuk menggenerate secara otomatis dokumen Use Case ini. Cara yang dapat dilakukan adalah dengan mengkonversi view point dalam VORD menjadi aktor. Serta services menjadi use case. Dengan mengubah dokumen VORD menjadi use case maka Service yang melibatkan lebih dari 1 view point hanya akan tergambar menjadi 1 use case saja. Dibawah ini akan disajikan contoh sederhana yang dapat menjadi gambaran proses konversi ini.

Pembobotan USE CASE

Untuk keperluan pengukuran volume pekerjaan setiap use case yang muncul diberi bobot. Pembobotan ini menjadi tugas analisis yang diberikan kepada setiap use case. Sebagai angka perkiraan tentunya angka yang disajikan tidaklah sepenuhnya tepat dengan kondisi pada pelaksanaan rekayasa. Sebagai pedoman untuk memudahkan pemberian angka bobot dapat di equivalenkan dengan jam orang kerja (JOK). Misalnya saja terdapat use case menghitung IPK, untuk merealisasikan use case ini diperlukan 10 JOK dari programmer untuk menuliskan program maka use case ini akan diberi bobot 10.

Berdasarkan use case yang ada dalam UML yang telah di otomasi tersebut dapat diperoleh jumlah total JOK. Jumlah jam orang kerja yang di butuhkan inilah yang akan dijadikan ukuran dari pekerjaan (volume pekerjaan). Perlu di ingat bahwa rekayasa perangkat lunak sebagai sebuah pekerjaan dan produk, biaya yang dibutuhkan selama proses rekayasa sepenuhnya akan di jadikan biaya

untuk membayar orang dan operasional selama proses itu berlangsung. Tidak ada biaya (jika ada tidak signifikan) yang diperlukan untuk membeli bahan baku.

Hasil pembobotan ini dapat berguna untuk memperkirakan jumlah sumber daya manusia yang dibutuhkan terlibat dalam tim pengembangan perangkat lunak. Pembobotan juga dapat digunakan untuk menghitung biaya produksi yang akan dihabiskan selama proses pembuatan program. Kedua faktor diatas selanjutnya akan menentukan berapa lama waktu yang dibutuhkan untuk membangun perangkat lunak tersebut. Penghitungan kasar ini akan menghasilkan waktu pengembangan dengan mengabaikan faktor urutan pekerjaan dalam jadwal pengembangan perangkat lunak.

Otomasi

Proses proses penyusunan dokumen VORD, konversi menjadi use case maupun pembobotan biasanya dilakukan secara manual oleh analis. Permasalahan menjadi cukup rumit ketika terjadi penambahan services, perubahan services maupun penambahan beberapa view point. Kejadian tersebut sangat mungkin terjadi karena teori – always incomplete- memang biasa terjadi di lapangan. Pada kondisi tersebut terpaksa analis akan menghitung dari awal lagi baik kebutuhan SDM, jadwal maupun estimasi biaya. Oleh karena aturan yang sudah cukup jelas, bentuk dokumen juga cukup jelas maka penulis mengusulkan pengembangan tools untuk melakukan otomasi proses ini.

Melihat susunan dokumen VORD maupun use case maka pada level pencatatan penulis mengusulkan di buat dokumen XML untuk menyimpan dokumen VORD maupun use case. Rencana dokumen XML untuk keperluan tersebut adalah sebagai berikut : User interface di rencanakan menggunakan notasi grafis agar pengguna dapat melakukan aktivitas perancangan seperti ketika menggunakan kertas dan pensil. Representasi output juga diusulkan dalam bentuk grafis. Adapun sebagian rencana antar muka adalah sebagai berikut :

Penutup

Makalah ini dibuat untuk menyelesaikan salah satu permasalahan permulaan dari proses rekayasa perangkat lunak. Otomasi proses pembuatan dokumen rekayasa ini diharapkan dapat menutup salah satu kelemahan software engineer yang di beberapa tempat sering teledor dalam mendokumentasikan hasil kerjanya. Keteledoran inilah yang kemudian menjadi penyebab mundurnya jadwal penyelesaian pekerjaan, membengkaknya biaya pengembangan dan kesulitan untuk melakukan penelusuran kembali proses rekayasa itu sendiri. Pengembangan ke berikutnya dapat dilakukan untuk melakukan otomasi proses selanjutnya seperti penyediaan dokumen diagram kelas, diagram kolaborasi, dokumen rencana testing dan lain sebagainya. Pengembangan ini diharapkan akan menciptakan tools yang cukup lengkap untuk membantu proses pengembangan perangkat lunak.

Daftar Pustaka

Sommerville, Ian, Software Engineering, Prentice hall 2000
Pressman Roger, Software Engineering,
Rekayasa Perangkat Lunak
Java