

Tinjauan Proses Elisitasi Kebutuhan Perangkat Lunak Menggunakan Metode Agile

Ni Made Satvika Iswari

Sekolah Teknik Elektro dan Informatika, Institut Teknologi Bandung, Bandung, Indonesia
satvika.iswari@gmail.com

Diterima 16 Mei 2012

Disetujui 30 Mei 2012

Abstract—Proses elisitasi kebutuhan perangkat lunak dengan menggunakan metode *agile* didasari oleh kebutuhan yang terus berubah seiring dengan berjalannya proses pembangunan perangkat lunak. Agent Based Goal Elicitation (ATABGE) merupakan sebuah teknik elisitasi kebutuhan yang menggunakan pendekatan *agile*. Teknik ini menggunakan *initial goals* kemudian mendekomposisinya menjadi *subgoals* yang dilakukan oleh *stakeholder*. Namun teknik ini perlu ditinjau kembali pada bagian Pre-Sprint, yang menyatakan bahwa *initial goals* ditentukan oleh analyst, padahal ini akan menyebabkan kebutuhan yang didapatkan menjadi tidak murni dari *stakeholder*. Selain itu, pada tahap Sprint, para *stakeholder* sepertinya perlu untuk diberikan prioritas wewenang untuk menentukan hasil. Secara keseluruhan, penggunaan metode *agile* juga memiliki beberapa kekurangan, yaitu kesulitan dalam menentukan estimasi biaya sejak awal proyek dan kesulitan dalam menentukan arsitektur perangkat lunak yang akan dibangun.

Index Terms—*requirement engineering, elicitation, agile, ATABGE, goals, agent*

I. PENDAHULUAN

Saat ini, beberapa pendekatan pembangunan perangkat lunak telah diajukan dan digunakan secara luas, salah satunya adalah proses *agile*. Seluruh proses *agile* yang diajukan bertujuan untuk menyelesaikan masalah kebutuhan (*requirement*) yang terus berubah seiring berjalannya pengembangan perangkat lunak. Pada tulisan ini, akan dibahas mengenai sebuah teknik elisitasi kebutuhan dengan menggunakan pendekatan *agile*, yang bernama Agent Based Goal Elicitation (ATABGE).

Teknik ATABGE didasari mekanisme dan pendekatan *agile* untuk mendapatkan *goal*/kebutuhan dari *stakeholder*. Pada proses tersebut, *goal* akan didekomposisi menjadi sub – sub *goals* dengan melibatkan peran *stakeholder* secara aktif sampai dengan penentuan Agent yang terlibat.

Namun, proses elisitasi kebutuhan dengan

menggunakan metode *agile* menimbulkan beberapa kesulitan. Pada tulisan ini akan dibahas juga mengenai kesulitan – kesulitan proses elisitasi dengan menggunakan metode *agile* tersebut. Selain itu, akan dibahas pula prinsip – prinsip dari teknik ATABGE yang kiranya perlu untuk ditinjau kembali.

II. REQUIREMENT ENGINEERING

Pengukuran utama dari keberhasilan sebuah perangkat lunak adalah apakah perangkat lunak yang dibangun sesuai dengan tujuan yang dimaksudkan pada perangkat lunak tersebut. *Requirement engineering* merupakan proses menemukan tujuan tersebut, dengan mengidentifikasi *stakeholders* dan kebutuhannya, dan mendokumentasikannya dalam bentuk yang dapat dianalisis, dikomunikasikan, dan diimplementasikan. Terdapat beberapa kesulitan dalam proses *requirement engineering* ini, yaitu *stakeholders* (termasuk *costumer*, pengguna, dan pengembang perangkat lunak) dapat berjumlah banyak dan terdistribusi. Tujuan masing-masing *stakeholder* tersebut dapat bervariasi dan dapat terjadi konflik, bergantung pada sudut pandang lingkungan dimana mereka bekerja dan pekerjaan yang ditangani. Tujuan tersebut dapat eksplisit atau mungkin sulit diartikulasi, dan hal ini dapat dibatasi oleh berbagai faktor di luar kendali para *stakeholder* tersebut[1].

Dalam pembangunan perangkat lunak, *requirement engineering* memegang peranan yang sangat penting. Hal ini dikarenakan keberhasilan perangkat lunak yang dibangun dapat diukur dengan parameter apakah perangkat lunak tersebut sesuai dengan kebutuhan yang telah didefinisikan di awal. Apabila pengembang kurang tepat dalam mendefinisikan kebutuhan di awal, maka perangkat lunak yang dihasilkan tidak dapat dikatakan berhasil. Kalaupun perlu dilakukan perbaikan-perbaikan terhadap bagian yang belum terpenuhi dari perangkat lunak tersebut, maka diperlukan biaya tambahan yang nilainya tentu lebih besar dari biaya perkiraan awal pembangunan perangkat lunak.

Stakeholder yang memegang peranan penting dalam pendefinisian kebutuhan perangkat lunak dapat terdiri dari beberapa orang. Setiap orang tersebut tentu memiliki pandangannya masing-masing terhadap perangkat lunak yang akan dibangun. Dalam tahap *requirement engineering* inilah, seluruh kebutuhan dari *stakeholder* tersebut dikumpulkan untuk kemudian dilakukan analisis dengan tujuan membangun perangkat lunak yang tepat guna, sesuai dengan kebutuhan.

III. PROSES ELISITASI DALAM REQUIREMENT ENGINEERING

Proses elisitasi merupakan tahap pertama dalam *requirement engineering*, yaitu proses identifikasi kebutuhan perangkat lunak yang berasal dari berbagai sumber, yaitu wawancara, *workshop*, *workflow* dan analisis tugas, analisis dokumen, dan mekanisme lainnya.

Elisitasi sebenarnya bukan kata yang terbaik dalam menggambarkan proses ini, karena kata elisitasi berarti bahwa kebutuhan perangkat lunak tersebut sudah ada dalam pemikiran *stakeholder*, sehingga pengembang hanya perlu menanyakannya saja. Padahal kebutuhan perangkat lunak terbentuk melalui proses kolaborasi antara *stakeholder* dengan pengembang perangkat lunak. Dan merupakan tugas dari pengembang perangkat lunak untuk membantu *stakeholder* menemukan kebutuhan perangkat lunak berdasarkan apa yang memang diperlukan, bukan pada apa yang diinginkan.

Terdapat beberapa teknik dalam elisitasi kebutuhan, yaitu:

1. Wawancara
2. Workshop
3. Survey pasar
4. Pemeriksaan *problem reports*
5. Observasi
6. Etnografi

IV. AGILE TECHNIQUE FOR AGENT BASED GOAL ELICITATION (ATABGE)

Agile Technique for Agent Based Goal Elicitation (ATABGE) merupakan sebuah teknik elisitasi kebutuhan yang menggunakan prinsip dan pendekatan yang gesit (*agile*). Teknik ini diajukan oleh A. M. Sen dan K. Hemachandran dari Departement of Computer Science, Assam University, Silchar, Assam, India.

Teknik ini menggunakan asumsi bahwa *goals* yang ingin dicapai belum pernah didokumentasikan atau dielisisasi secara eksplisit dari *stakeholders*, sehingga analisis harus bekerja dari diagram atau dokumen – dokumen mengenai perusahaan yang bersangkutan. ATABGE berfokus pada identifikasi awal dan abstraksi goal dari para *stakeholder*. Proses yang dilakukan dalam metode ATABGE ini berkisar antara 1 sampai 4 minggu, yang disebut Sprint.

Prinsip dari teknik *agile* adalah mengadopsi dokumentasi seminimal mungkin untuk mengekspresikan kebutuhan (*requirement*) dalam bahasa yang natural dalam setiap sesi Sprint. Selanjutnya, tantangan dalam elisitasi dari kebutuhan non-fungsional dilakukan dengan mendorong partisipasi maksimum dari *stakeholders* dalam sesi Sprint.

Dalam metode ATABGE, terdapat 3 buah aktivitas utama yang dilakukan. Aktivitas yang pertama adalah Pre-Sprint, yaitu aktivitas awal yang dilakukan sebelum melakukan proses elisitasi *goals*. Aktivitas berikutnya adalah Sprint, yang merupakan proses utama dalam melakukan proses elisitasi *goals*. Kemudian yang terakhir adalah Post Sprint, yang merupakan aktivitas penutup dari kegiatan elisitasi yang dilakukan.

A. Pre-sprint Activity

Pada sesi Pre-Sprint, analyst mula – mula akan memeriksa dokumen – dokumen perusahaan, seperti “Interview Transcripts”, “Objectives of Enterprises”, “Corporate Policies”, “Corporate Vision”, “Mission Statement of the Enterprise”, dan “Work Flow Diagrams”. Berdasarkan pemeriksaan terhadap dokumen – dokumen tersebut, analisis kemudian akan menyiapkan *initial list* dari goal yang akan dicapai dari proses elisitasi yang dilakukan.

Setelah *initial goals* tersebut telah siap, para *stakeholders* akan diundang dalam proses Pre-Sprint. Pada proses ini, jumlah *stakeholders* yang berpartisipasi dan jumlah *initial goal* akan dihitung. Para *stakeholders* kemudian akan diberikan *initial goals* tersebut dan diminta untuk memberikan nilai prioritas untuk setiap *goals* serta alasannya dalam sebuah Activity Card Compiler. Activity Card Compiler tersebut kemudian diolah dan menghasilkan First Compilation Table, yang menggambarkan *initial goals* yang telah diberikan nilai prioritasnya oleh setiap *stakeholder*.

B. Sprint Activity

Setelah First Compilation Table terbentuk, maka dapat dilakukan *Sprint Activity* yang terbagi menjadi 4 buah bagian, yaitu Develop, Wrap, Review, dan Adjust.

Pada tahap Develop, *stakeholder* akan mengambil goal pada urutan pertama yang mereka pilih dari *initial goals*. Setelah itu, mereka diminta untuk menentukan sub-goal dan agent yang terlibat. Goal dan Agent merupakan kunci utama dalam model ini. Agent dalam hal ini dapat berupa manusia, *input/output devices*, komponen perangkat lunak, dan sebagainya.

Pada tahap Wrap, sub-goal yang telah ditentukan pada tahap sebelumnya akan dihubungkan dengan predesernya. Tahap ini akan dimasukkan dalam Activity Card Compiler.

Kemudian pada tahap Review, setiap Activity Card yang telah dikompilasi akan didiskusikan oleh seluruh anggota dengan cara menunjukkan jumlah dan deskripsi dari sub-goal yang berhasil dielisisasi. Analyst kemudian akan memasukan seluruh informasi tersebut ke dalam program Activity Card Compiler.

Selanjutnya. Pada tahap Adjust, seluruh hasil pada tahap Review akan dideliver ke Activity Card masing – masing *stakeholder*. Kemudian, isi dari Activity Card seluruh *stakeholder* menjadi sama sesuai dengan hasil yang telah dicapai sejauh ini.

Tahap Sprint ini akan terus berulang sampai tidak ada lagi sub-goal baru lagi yang ingin dicapai.

C. Post Sprint Activity

Proses elisitasi telah berakhir. Pada tahap ini, tim Requirement Engineer akan mendokumentasikan hasil goal dan sub-sub goalnya yang telah dielisisasi dengan menggunakan prosedur yang telah disepakati.

V. TINJAUAN METODE ELISITASI

Proses *agile* yang dilakukan pada *requirement engineering* pada dasarnya bertujuan untuk memformulasikan gambaran jelas mengenai sistem dimana kebutuhan akan selalu berubah.

Adapun prinsip dan pendekatan untuk mendukung proses *agile* adalah sebagai berikut:

- *Iterative Development*
- *Increments of Work*
- *Collaboratorium*
- *Adaptation*

- *Emergence*

Sebelum proses *agile* dikembangkan, kebanyakan proyek menggunakan model *waterfall* atau model pengembangan perangkat lunak lainnya dalam melakukan proses pengembangan perangkat lunak. Pada pengembangan dengan menggunakan model ini, proses pengumpulan kebutuhan (*requirement*) akan dilakukan di awal dengan usaha yang cukup besar. Namun, ternyata metode tersebut hampir pasti penuh dengan kesalahan dan membutuhkan waktu yang lama. Pada metode tersebut, *stakeholder* akan cenderung untuk mengubah kebutuhannya seiring dengan apa yang dia lihat di pasaran.

Oleh sebab itu, dikembangkanlah sebuah proses yang gesit (*agile*), yang bertujuan untuk menangani kebutuhan yang sering berubah – ubah seiring dengan berjalannya waktu. Namun, proses *agile* dikembangkan bukan tanpa kekurangan. Pada subbab ini, akan dijelaskan beberapa hal yang menjadi kesulitan dalam melakukan proses elisitasi kebutuhan menggunakan metode *agile*. Selain itu, akan dibahas pula mengenai konsep – konsep ATABGE yang dirasa perlu untuk ditinjau kembali.

A. Kesulitan – Kesulitan dalam Proses Elisitasi Kebutuhan Menggunakan Teknik *Agile*

Proses elisitasi kebutuhan dengan menggunakan metode *agile* akan membutuhkan waktu yang tidak dapat diperkirakan sebelumnya. Hal ini menyebabkan segala estimasi dan pengembangan arsitektur perangkat lunak menjadi lebih sulit dilakukan. Tanpa mengetahui wujud final dari produk atau kebutuhan pasar, proses estimasi menjadi hal yang tidak mungkin untuk dilakukan dengan benar.

Adalah hal yang cukup menenangkan mengetahui bahwa perubahan kebutuhan yang dilakukan pada saat elisitasi kebutuhan dilakukan demi kompetisi untuk menghasilkan produk terbaik. Namun, dengan menggunakan metode *agile*, kita hanya dapat melakukan perhitungan mengenai biaya yang dibutuhkan untuk sebuah siklus saja, untuk estimasi biaya selanjutnya tidak dapat dilakukan dengan baik karena tidak diketahuinya kebutuhan pada siklus berikutnya.

Salah satu hal yang dapat dilakukan untuk mengatasi hal ini adalah dengan melakukan pembayaran atas proyek per siklus *agile* yang dijalani, yang sebenarnya akan sama dengan pembayaran seluruh proyek dengan menggunakan metode yang lama. Namun, tetap saja, akan ada waktu dimana jumlah biaya proyek secara

keseluruhan dibutuhkan, misalnya dalam kontrak tim. Dalam proses elisitasi yang dilakukan, *stakeholder* akan menyampaikan apa yang dibutuhkanannya sebisa yang mereka dapat lakukan, sehingga estimasi biaya terpaksa dilakukan dengan menyesuaikan dengan probabilitas biaya dari perubahan yang akan datang.

Sama halnya dengan arsitektur yang dipilih oleh tim selama siklus awal. Pilihan tersebut bisa jadi salah karena kebutuhan selanjutnya sulit untuk diprediksi. Setelah adanya perubahan kebutuhan, yang dapat dilakukan adalah mengerjakan ulang arsitektur tersebut.

B. Tinjauan Teknik ATABGE

ATABGE, seperti yang telah dijelaskan sebelumnya, merupakan salah satu teknik elisitasi kebutuhan dalam *requirement engineering* yang menerapkan metode *agile*. Teknik ini terdiri dari 3 buah tahap, yaitu Pre-sprint, Sprint, dan Post Sprint Activity.

Pada tahap Pre-Sprint, disebutkan bahwa *initial goals* ditentukan sejak awal oleh *analyst* setelah meninjau dokumen – dokumen yang berhubungan. Dalam prinsip *requirement elicitation*, dikemukakan bahwa proses elisitasi merupakan proses kolaborasi, dimana *stakeholder* dan *analyst* bersama – sama membangun kebutuhan sesuai dengan apa yang diharapkan oleh *stakeholder*. Adapun tujuan *analyst* dalam hal ini adalah membantu *stakeholder* untuk menemukan apa yang benar – benar mereka butuhkan. Apabila *initial goals* telah ditentukan di awal oleh *analyst*, dikhawatirkan *stakeholder* akan terarah sehingga kebutuhan yang kemudian dilontarkan menjadi tidak murni lagi, atau dengan kata lain tidak menggambarkan apa yang sebenarnya *stakeholder* inginkan.

Kemudian, tinjauan dilakukan untuk tahap Sprint, yaitu tahap dimana para *stakeholder* menentukan subgoals serta deskripsi dan prioritasnya dalam sebuah Activity Card. Pada tahap ini, Activity Card dari para *stakeholder* akan diproses menjadi sebuah model yang mewakili prioritas kebutuhan dari para *stakeholder*. Pada tahap ini, posisi dan kedudukan dari para *stakeholder* dibuat seolah – olah sejajar, sehingga prioritas pendapat dari masing – masing *stakeholder* dianggap sama. Padahal, masing – masing *stakeholder* tentu memiliki bagian dan wewenangnya masing – masing dalam proyek yang dijalankan. Hal ini akan menimbulkan permasalahan ketika terjadi perbedaan pandangan mengenai prioritas suatu subgoal tertentu. *Stakeholder* yang memiliki andil dalam subgoal tersebut tentu lebih memahami mengenai prioritas

yang sebenarnya, namun berbeda dengan *stakeholder* yang kurang paham namun memberikan prioritas yang berbeda sehingga memengaruhi hasil akhir dari proses elisitasi kebutuhan.

VI. KESIMPULAN

Pada proses elisitasi kebutuhan, *analyst* akan mengumpulkan kebutuhan dan informasi dari *stakeholder*, dan menyiapkan Software Requirement Specification untuk pengembangan perangkat lunak yang selanjutnya dilakukan. Proses elisitasi kebutuhan tersebut merupakan proses yang cukup menantang, karena seringkali kebutuhan dari *stakeholder* terus berubah seiring berjalannya pengembangan perangkat lunak. Dengan demikian, diajukanlah Agile Technique for Agent Based Goal Elicitation (ATABGE), yang menggunakan pendekatan metode *agile*.

Teknik ATABGE ini memiliki 2 buah kunci utama, yaitu *goal* dan *agent*. Selama proses elisitasi dengan menggunakan teknik ATABGE, *stakeholder* akan dilibatkan secara aktif mulai dari mendekomposisi *goal* sampai menentukan *agent*. Namun, agaknya perlu ditinjau kembali pada tahap Pre-Sprint, dimana dikatakan bahwa *initial goals* ditentukan sepenuhnya oleh *analyst*. Hal ini akan menimbulkan kebutuhan yang tidak murni dari *stakeholder*. Selain itu, pada tahap Sprint, masing – masing *stakeholder* nampaknya perlu untuk diberikan prioritas pendapat, agar *stakeholder* yang memiliki wewenang lebih dapat menentukan keputusan dengan pertimbangan yang lebih tinggi.

Penggunaan metode *agile* juga memiliki beberapa kekurangan, yaitu kesulitan dalam menentukan estimasi biaya sejak awal proyek dan kesulitan dalam menentukan arsitektur perangkat lunak.

DAFTAR PUSTAKA

- [1] Nuseibeh, Bashar, Steve Easterbrook. 2000. *Requirements Engineering: A Roadmap*. ICSE '00 Proceedings of the Conference on The Future of Software Engineering
- [2] Johnson, Dr. Suzette, et al. 2009 *Requirement Engineering in an Agile Environment*. Northrop Grumman Corporation
- [3] Paetsch, Fraukem et al. 2003. *Requirement Engineering and Agile Software Development*. University of Calgary.
- [4] Sen, A.M, et al. 2010. *Elicitation of Goals in Requirement Engineering using Agile Methods*. 34th Annual IEEE Computer Software and Applications Conference Workshops.
- [5] Tomayko, James E. 2002. *Engineering of Unstable Requirements Using Agile Methods*. Carnegie Mellon University.