

BAB I PENDAHULUAN

I.1 Latar Belakang

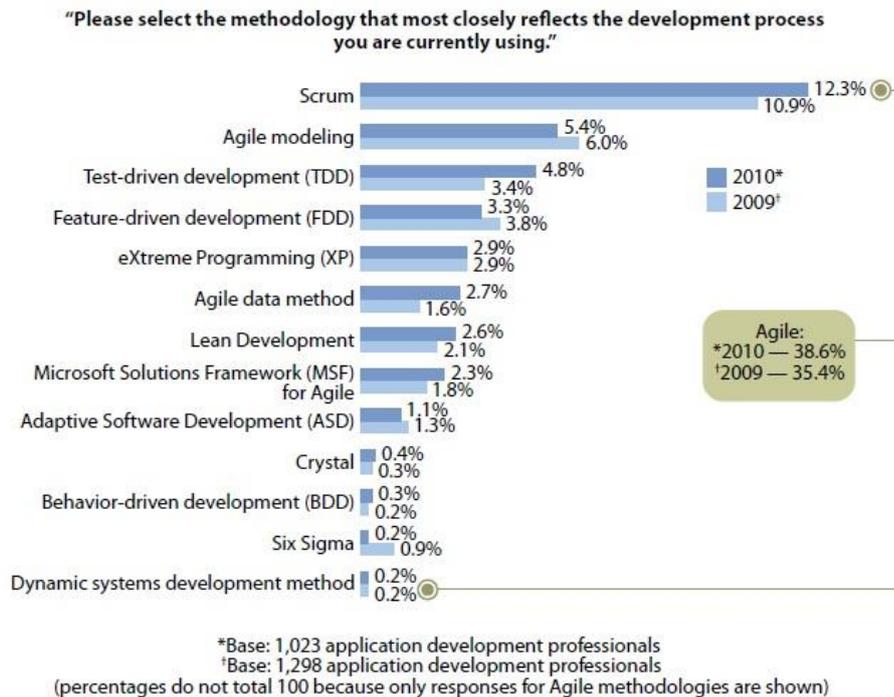
Proyek adalah sebuah usaha yang bersifat sementara untuk menghasilkan suatu produk atau layanan yang bersifat unik.¹ Dalam menjalankan proyek, terdapat batasan-batasan yang harus dikendalikan agar proyek berjalan sesuai dengan perencanaan awal. Proyek pengembangan perangkat lunak dapat dikatakan berhasil apabila memenuhi tiga batasan, yaitu sesuai dengan waktu pengerjaan proyek (*time*), sesuai dengan pembiayaan proyek (*cost*), dan menghasilkan produk yang sesuai dengan harapan (*scope*).² Dalam menjalankan proyek, tentu memerlukan sebuah mekanisme pengelolaan atau manajemen proyek agar pengkoordinasian kegiatan-kegiatan terselesaikan secara efektif dan efisien.

Saat ini terdapat banyak sekali metode ataupun *framework* manajemen proyek, salah satunya adalah *scrum*. *Scrum* merupakan sebuah *framework* manajemen proyek yang menggunakan konsep kolaborasi dalam pengerjaan proyek. Kolaborasi dalam *scrum* melibatkan *product owner* sebagai perwakilan dari *customer* dengan tim proyek *scrum*. Konsep kolaborasi yang terjadi di dalam *framework scrum* melibatkan seorang *scrum master* sebagai seorang fasilitator antar tim proyek dan inter tim proyek, hal ini menjadikan *scrum master* sebagai pihak yang mempertahankan pengerjaan proyek dari gangguan eksternal agar sesuai dengan *sprint backlog* dan *timeline*.

Penggunaan *scrum* sebagai *framework* manajemen proyek menurut survei oleh *Forrester Research* pada kuartal ketiga tahun 2011 dapat disimpulkan bahwa *scrum* memang banyak digunakan oleh para praktisi di dunia. Berikut adalah statistik penggunaan *scrum* sebagai *framework* manajemen proyek berdasarkan survey tersebut.

¹ Kathy Schwalbe, *Information Technology Project Management*, (Canada: Thomson Course Technology, 2006), page 4.

² Kathy Schwalbe, *Information Technology Project Management*, (Canada: Thomson Course Technology, 2006), page 7.



*Source: Forrester/Dr. Dobb's Global Developer Technographics® Survey, Q3 2010

†Source: Forrester/Dr. Dobb's Global Developer Technographics® Survey, Q3 2009

60109

Source: Forrester Research, Inc.

Gambar I. 1 Statistik Penggunaan *Framework Scrum*³

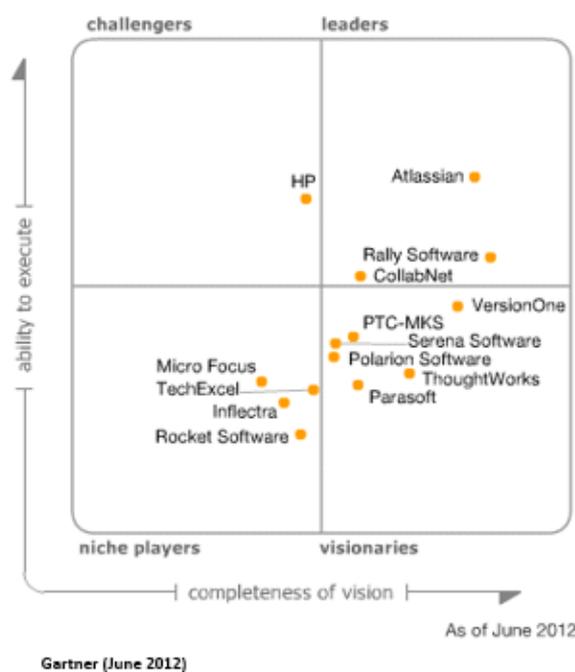
Framework scrum sering digunakan di dalam proyek dikarenakan mitigasi risiko terhadap kualitas *deliverable product* cukup baik dan adanya kejelasan mengenai kriteria produk dikatakan selesai. Hal tersebut dikarenakan *framework scrum* menggunakan konsep kolaborasi, sehingga risiko-risiko yang mungkin muncul pada tahap pengerjaan proyek telah terlebih dahulu didefinisikan pada pertemuan antara tim pengembang *scrum*, *scrum master*, dan *product owner*. Produk dapat dikatakan telah selesai apabila telah sesuai dengan *product backlog* yang telah disepakati oleh tim pengembang *scrum* dengan *product owner*.

Dalam pengelolaan proyek, fase proyek terbagi atas *initiating*, *planning*, *executing & controlling*, dan *closing*. Pada tahapan proyek tersebut, fase *executing & controlling* merupakan fase yang membutuhkan usaha paling besar, dikarenakan keseluruhan proses pengerjaan proyek bertumpu pada fase tersebut. Dibutuhkan

³ West Dave, Water-Scrum-Fall is the Reality for Most Organizations Today, (Cambridge: Forrester Research Inc., 2011), page 4.

sebuah alat bantu berupa aplikasi *electronic project management* yang dapat mengakomodir keseluruhan fungsi yang terdapat dalam fase *controlling* proyek. Sesuai dengan teori menurut Stephen R. Hagan dan Eric Teicholz, penggunaan *electronic project management* dalam pengelolaan proyek berpotensi untuk mengubah *current workflow* dan *relationship* dan menjadikan usaha yang signifikan dalam peningkatan produktivitas untuk industri.⁴

Saat ini telah terdapat *scrum tools* yang beredar di pasaran, berikut ini adalah data yang didapat dari hasil survey oleh *Gartner* pada tahun 2012.



Gambar I. 2 *Gartner Magic Quadrant for Application Life Cycle Management*⁵

Gambar I.2 menunjukkan kuadran aplikasi manajemen proyek berbasis *scrum* yang populer digunakan. Pada penelitian ini, diambil dua buah sampel aplikasi, yaitu “CollabNet” dan ”VersionOne” sebagai bahan *benchmark*. *Benchmark* dilakukan untuk mengetahui kelebihan dan kekurangan masing-masing aplikasi. Berikut adalah hasil *benchmark* yang telah dilakukan.

⁴ Stephen R. Hagan, Eric Teicholz, *e-PM (Electronic Project Management)*, (Massachusetts: Graphic Systems Inc, 2000), page.1

⁵ Murphy, Thomas E. *Application Life Cycle Management*. (Gartner Inc., 2012).

Tabel I. 1 Tabel *benchmark* antar aplikasi

No.	Fitur	CollabNet	VersionOne	Sprintometer	Ms. Project
1.	Kategori	Berbayar	Berbayar	Gratis	Berbayar
2.	Dukungan multi-peran	Ya	Tidak	Tidak	Tidak
3.	Inisiasi proyek	Tidak	Tidak	Tidak	Ya
4.	Perencanaan produk	Ya	Ya	Ya	Tidak
5.	Pengelolaan <i>product backlog</i>	Ya	Ya	Ya	Tidak
6.	Perencanaan <i>sprint</i>	Ya	Ya	Ya	Ya
7.	Pengelolaan <i>sprint backlog</i>	Ya	Ya	Ya	Ya
8.	Pengelolaan <i>release plan</i>	Ya	Ya	Tidak	Tidak
9.	<i>Sprint</i>	Ya	Ya	Ya	Ya
10.	Pertemuan harian	Ya	Tidak	Tidak	Tidak
11.	<i>Burndown chart</i>	Ya	Ya	Ya	Ya
12.	Grafik tingkat penyelesaian keseluruhan produk	Ya	Tidak	Tidak	Ya
13.	Grafik <i>earned value management</i>	Tidak	Tidak	Tidak	Ya
14.	<i>Sprint retrospective</i>	Ya	Ya	Tidak	Tidak
15.	Penutupan <i>sprint</i>	Ya	Ya	Tidak	Ya
16.	Penutupan proyek	Tidak	Tidak	Tidak	Ya

Tabel diatas adalah tabel perbandingan setiap aplikasi manajemen proyek berbasis metode *agile*, keseluruhan aplikasi yang dibandingkan memiliki dukungan terhadap *scrum*. Kesimpulan yang didapatkan dari hasil *benchmark* diatas adalah perlu adanya kombinasi dari setiap kelebihan dari masing-masing aplikasi diatas untuk dapat mengakomodir kekurangan dari setiap aplikasi.

Berdasarkan kesimpulan dari *benchmark* yang telah dilakukan, maka dilakukanlah fase pengembangan sistem, metode pengembangan sistem yang digunakan adalah dengan menggunakan metode *prototyping*. Metode *prototyping* digunakan karena waktu pengembangan sistem yang tergolong cepat dan kebutuhan sistem yang dinamis. Metode *prototyping* dapat mengakomodasi perubahan kebutuhan sistem yang dinamis dikarenakan adanya keterlibatan pengguna akhir dalam fase pengembangan sistem. Dengan demikian sistem yang dikembangkan dapat menutupi kekurangan dari masing-masing aplikasi dari hasil *benchmark* diatas.

I.2 Perumusan Masalah

Masalah yang akan dibahas pada tugas akhir ini adalah sebagai berikut.

- a. Bagaimana cara meningkatkan produktivitas dalam implementasi proyek berbasis *scrum*?
- b. Bagaimana membantu pengguna dalam mengelola waktu pengerjaan proyek, pembiayaan proyek, dan lingkup proyek?

I.3 Tujuan Penelitian

Tujuan yang ingin dicapai dalam tugas akhir ini adalah menghasilkan aplikasi *electronic project management* yang memiliki fungsi pengelolaan proyek berbasis *scrum* untuk mempermudah pengelolaan dan meningkatkan produktivitas dalam menjalankan proyek berbasis *scrum*.

I.4 Manfaat Penelitian

Manfaat yang dapat diambil dari penelitian tugas akhir ini adalah sebagai berikut.

- a. Memberikan sarana pengelolaan proyek berbasis *Scrum* melalui aplikasi *electronic project management*.
- b. Memberikan kemudahan akses informasi kepada seluruh anggota proyek dan *product owner* dalam menjalankan proyek.

I.5 Batasan Masalah

Beberapa batasan masalah yang digunakan dalam tugas akhir ini adalah sebagai berikut.

- a. Aplikasi yang dikembangkan ditujukan bagi penggunaan secara umum.
- b. Ruang lingkup pengembangan aplikasi hanya membahas sisi lingkup proyek, waktu proyek, dan pembiayaan proyek. Hal tersebut dikarenakan keterbatasan kemampuan peneliti dan waktu penelitian.
- c. Aplikasi yang dikembangkan tidak sampai tahap implementasi sistem, namun hanya sampai pada tahap pembuatan *prototype* yang akan diulas oleh *user* secara umum dan mengacu pada buku *Scrum Guide*.
- d. Aplikasi yang dibangun tidak mengakomodasi perhitungan prioritas *backlog* dan perhitungan *business value*.