

RANCANG BANGUN SISTEM *EXTREME PROGRAMMING* SEBAGAI METODOLOGI PENGEMBANGAN SISTEM

Ahmad Fatoni¹, Dhany Dwi²

Program Studi Sistem Komputer - Universitas Serang Raya Serang - Indonesia

ahmadfatoni101@gmail.com, dhanydwinugroho@gmail.com

Abstrak - Semakin meningkatnya kepuasan pelanggan, maka keinginan perusahaan untuk memberikan yang terbaik pun akan terus meningkat. Hal tersebut akan membuat sistem informasi dalam perusahaan harus terus dikembangkan setiap saat. Banyak sekali metodologi pengembangan sistem yang dapat digunakan. Salah satunya adalah *Extreme Programming*. *Extreme Programming* merupakan salah satu model proses dari *Agile Software Development* yang merupakan salah satu metodologi dalam pengembangan sistem berbasis *Software Development Life Cycle (SDLC)*. Tujuan penulisan ini adalah untuk memberikan informasi dasar mengenai konsep *Extreme Programming*, kelemahan dan kekurangan maupun penerapannya dalam dunia nyata. Metodologi penulisan *paper* yang digunakan adalah dengan mencari teori-teori yang mendukung yang diperoleh dari Metode Studi Kepustakaan (*Library Research*) dan Studi Web Browser (*Web Research*). Hasil yang dicapai adalah dengan adanya informasi mengenai *Extreme Programming* maka diharapkan tim *developer* dapat menentukan metodologi yang benar karena sudah mengetahui informasi dan kondisi seperti apa saat menerapkan *Extreme Programming*. Simpulan yang dapat diambil dari hasil penulisan ini adalah metodologi pengembangan sistem tidak hanya satu. Dan sebelum mengembangkan perangkat lunak, tim harus mempertimbangkan semua metodologi yang ada agar tidak terjadinya resiko kegagalan. *Extreme Programming* hanyalah satu dari sekian banyak metodologi yang ada.

Kata Kunci: *extreme programming*, metodologi, pengembangan sistem

I. PENDAHULUAN

Pengembangan sistem informasi adalah suatu aktivitas untuk menyempurnakan sistem atau perangkat lunak yang pernah dibuat sebelumnya. pengembangan sistem ini, tidak bisa dilakukan hanya dalam waktu satu atau dua hari saja. pengembangan sistem harus dilakukan dengan matang dan terencana. Tidak hanya itu, tidak menutup kemungkinan dalam mengerjakan proyek pengembangan sistem informasi akan terjadi perubahan waktu secara tiba-tiba yang akan membuat tim pengembang sistem informasi dihadapkan kedalam kondisi maupun situasi yang mendesak. Sebagai contoh, saat Anda menjadi seorang *project leader* pada sebuah proyek pengembangan perangkat lunak, dan proyek yang Anda lakukan sudah hampir selesai tetapi klien tiba-tiba mengubah kebijakan sesuai keinginannya, apakah hal tersebut hal yang menyenangkan? Tentu tidak. Dari awal proyek dilaksanakan, tentunya tim pengembangan sudah melakukan rencana untuk kedepannya untuk menghindarinya hal-hal yang tidak diinginkan. Untuk menanggulangi situasi seperti itu, maka terdapat salah satu cara untuk mengatasi perubahan situasi dan kondisi cepat, yaitu *Extreme Programming*.

II. DASAR TEORI

a. Konsep Sistem Informasi

Sistem adalah sekelompok elemen yang bekerja sama (terintegrasi) untuk mencapai suatu tujuan atau sasaran tertentu. Elemen-elemen dalam suatu sistem tidak dapat berdiri sendiri karena semua komponen ini saling membutuhkan dan saling berhubungan antara satu dengan yang lainnya demi mencapai tujuan.

Pengertian sistem informasi sangat beragam. Satzinger (2010, p6-7) berpendapat bahwa sistem informasi merupakan gabungan dari pengertian sistem dan informasi, sehingga dapat disimpulkan sistem

informasi adalah kumpulan-kumpulan komponen yang saling terhubung satu sama lain untuk mengumpulkan dan memproses data hingga menjadi suatu hasil yang tentunya dapat digunakan untuk menyelesaikan masalah. O'Brien (2008, p26) mengatakan bahwa sistem informasi terbagi menjadi banyak komponen, seperti sumber daya manusia, *hardware*, *software*, *data resources* dan *network research*.

b. Extreme Programming

Metode *Extreme Programming* sering juga dikenal dengan metode XP. Metode ini dicetuskan oleh Kent Beck, seorang pakar *software engineering*. *Extreme programming* adalah model pengembangan perangkat lunak yang menyederhanakan berbagai tahapan pengembangan sistem menjadi lebih efisien, adaptif dan fleksibel. Nilai dasar metode *extreme programming* :

- 1) *Communication* : Memfokuskan komunikasi yang baik antara *programmer* dengan *user* maupun antar *programmer*.
- 2) *Courage* : Pengembang perangkat lunak harus selalu memiliki keyakinan, keberanian dan integritas dalam melakukan tugasnya.
- 3) *Simplicity* : Lakukan semua dengan sederhana.
- 4) *Feedback* : Mengandalkan *feedback* sehingga dibutuhkan anggota tim yang berkualitas.
- 5) *Quality Work* : Proses berkualitas berimplikasi pada perangkat lunak yang berkualitas sebagai hasil akhirnya.

c. Hardware Resources

Komponen ketiga adalah *Software Resource*. *Software Resource* merupakan sekumpulan petunjuk untuk digunakan untuk memproses informasi. Contoh dari *software resources*

- Sistem *software*. Terdiri dari program pengoperasian sistem yang digunakan untuk

mengendalikan dan mendukung operasi pada sistem komputer.

- Aplikasi *software*. Program yang digunakan untuk memproses sebagian penggunaan komputer secara langsung dan dilakukan oleh *end users*.
- Prosedur. Petunjuk operasional yang ditujukan kepada user yang akan menggunakan sistem informasi tersebut.

d. Data Resources

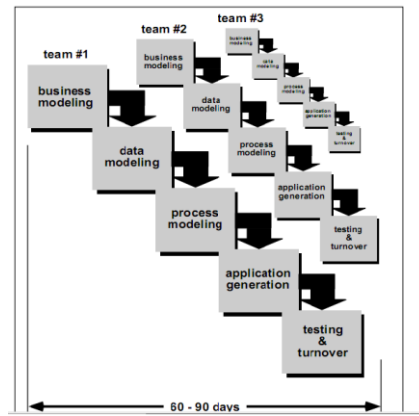
data resources. Sumber data yang diatur, disimpan dan diakses oleh manajemen sumber data ke dalam :

- Database yang menangani pemrosesan dan pengaturan data.
- Pengetahuan dasar yang menangani ilmu pengetahuan seperti fakta, aturan, contoh masalah mengenai praktik bisnis yang sukses.

e. Network Resources

Dan yang terakhir adalah *Network Resources*. *Network Resources* memperhatikan teknologi komunikasi dan jaringan yang ada pada sumber daya dalam sistem informasi. Namun yang diperhatikan hanya yang termasuk kedalam komponen dasar dari sumber daya tersebut saja. Yang termasuk di dalam *network resources*

- Media Komunikasi. Contohnya yaitu *twisted-pair wire*, kabel koaksial, dan *fiber-optical cable* dan *microwave*, *cellular* dan teknologi *wireless-satellite*
- Network Support. *Network Support* memperhatikan *hardware*, *software* dan teknologi data yang dibutuhkan untuk mendukung operasi dan penggunaan jaringan komunikasi.



Gambar 3. RAD

C. Agile Software Development

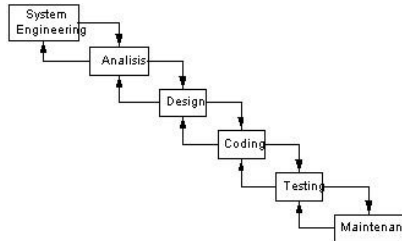
Merupakan model pengembangan sistem yang cepat dan lebih mementingkan interaksi dengan *user* daripada proses dan alat. Karena kepuasan klien merupakan prioritas utama bagi *agile software development*.



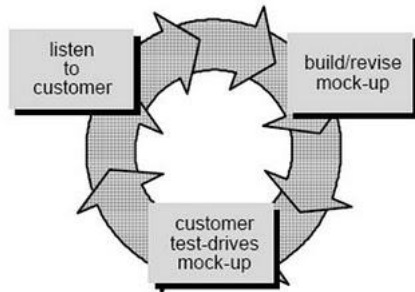
Gambar 4 Tahap model Agile Software Development.

III. DESAIN PENELITIAN

a. WAREFAL



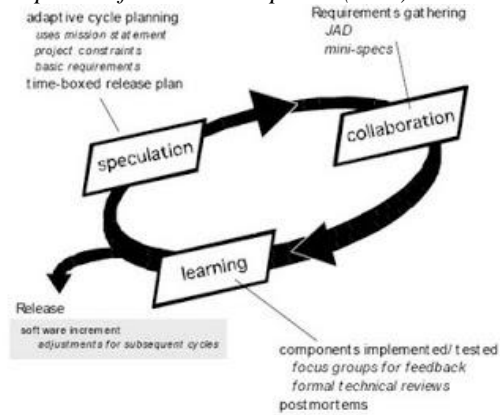
Gambar 1. Diagram Kontek waterfall



Gambar 2. Diagram Overview Prototype

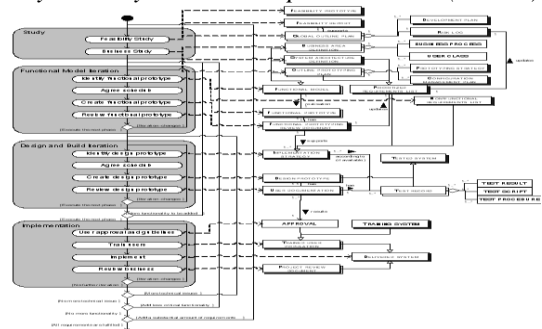
b. RAD

a. Adaptive Software Development (ASD)

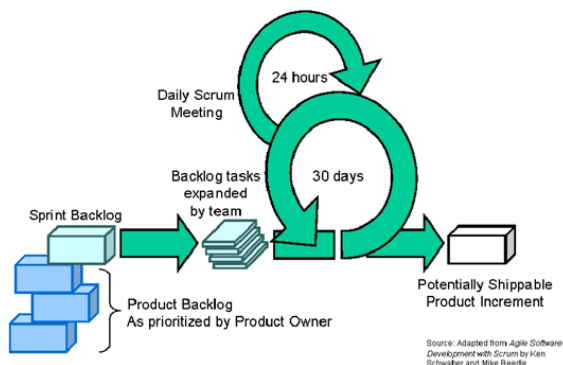


Gambar 5. Tahap model ASD

b. Dynamic Systems Development Method (DSDM)



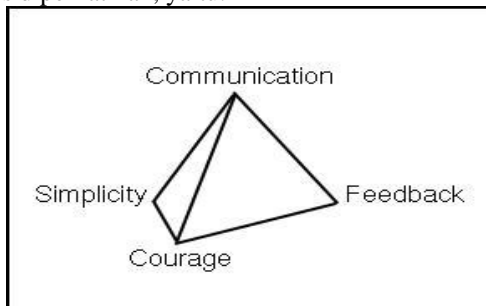
Gambar 6 Tahap model DSDM



Gambar 7 Scrum

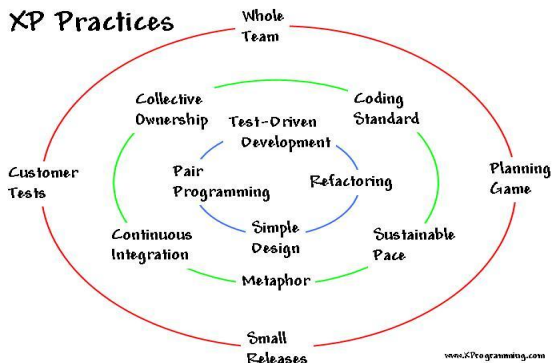
F. Nilai-Nilai Dasar *Extreme Programming*

Pada setiap metodologi pengembangan sistem tentunya ada beberapa hal dasar yang harus diperhatikan sebelum lanjut kedalam kerangka kerja agar tidak terjadi kesalahan pemilihan metodologi. Pada setiap tahapan proses pengembangan sistem yang menggunakan model proses *Extreme Programming* juga memiliki nilai-nilai mendasar yang harus diperhatikan, yaitu:



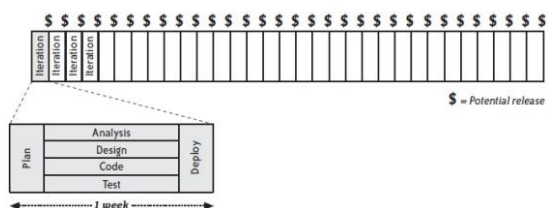
Gambar 8. Nilai Dasar *Extreme Programming*

G. Aspek Dasar *Extreme Programming*



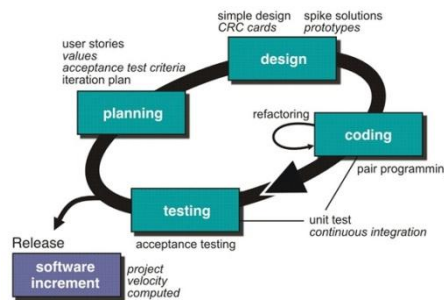
Gambar 9 Aspek Dasar *Extreme Programming*

Penerapan *Extreme Programming* bisa dilakukan di banyak perusahaan atau organisasi. Sebuah organisasi seperti *software house* atau pengembangan sebuah software dapat menerapkan *extreme programming* karena *extreme programming* mengedepankan *extreme programming life cycle*.



Gambar 10 *Extreme programming Life Cycle*

J. Kerangka Kerja *Extreme Programming*



Gambar 11 Kerangka Kerja *Extreme Programming*

IV. KESIMPULAN DAN SARAN

a. Kesimpulan

1. *Extreme Programming* berguna untuk mempercepat pekerjaan suatu tim dalam organisasi atau perusahaan. Karena dalam *Extreme Programming life cycle* menuntut ke suatu tim untuk menyelesaikan rangkaian aktivitas *Planning, Analys, Design & Code, Test, Deploy* dalam tempo waktu yang telah ditentukan.
2. *Extreme Programming* merupakan salah satu metodologi yang mendukung percepatan pembangunan suatu sistem.
3. *Extreme Programming* memprioritaskan komunikasi yang baik antar klien maupun antar sesama anggota tim.
4. Dalam penerapan *Extreme Programming* tanggung jawab per individu menjadi berkurang. Karena dalam mengembangkan suatu sistem, anggota tim selalu bekerja bersama-sama sehingga tanggung jawab per individu pun berkurang.
5. Semua anggota tim di tuntut untuk melakukan aktivitas *Planning, Analys, Design & Code, Test, Deploy*.
6. Tidak semua perusahaan atau organisasi bisa menerapkan metodologi *eXtreme Programming*, karena *eXtreme Programming* menuntut beberapa individu dengan spesialis berbeda (*Programmer, Designer, Analys, Tester*).

b. Saran

Extreme Programming bisa di terapkan pada perusahaan atau organisasi yang mempunyai tim khusus untuk pengembangan suatu sistem. Perusahaan atau organisasi yang ingin menerapkan *Extreme Programming* diharapkan untuk bisa memahami dan mengerti *Extreme Programming practice*, dan *scrum*. Jika perusahaan atau organisasi sudah mengerti *Extreme Programming practice* dan *scrum*, maka bisa mempelajari *scrum* terlebih dahulu lalu *full Extreme Programming*.

V. DAFTAR PUSTAKA

[1] Chan, Syahrial. 2010. *Membuat Aplikasi Client/Server dan Web dengan PowerBuilder 11.5*. Jakarta : PT Elex Media Komputindo.
 [2] Indrawan, Andri. 2008. ERD (Entitas Relationship Diagram). [Online]. Tersedia : <http://developdottxt>.

- wordpress.com/2008/04/03/erd-entitas-relationship-diagram/. [01 Februari 2014].
- [3] Jeb, Resthoe. 2012. Pengertian Kerja. [Online]. Tersedia: <http://resthoe.blogspot.com/2012/12/pengertian-kerja.html>. [23 Januari 2014].
- [4] Jeffrey L. Whitten & Lonnie D. Bentley. 2004. *System Analysis and Design Methods, 7th Edition*. New York : The McGraw-Hill Companies.
- [5] Jerry FitzGerald, Ardra F. FitzGerald & Warren D. Stallings, Jr. 1981. *Fundamentals of System Analysis, 2nd Edition*. New York : John Willey & Sons.
- [6] Pressman, Roger S. 2010. *Software Engineering*. Singapore : The McGraw-Hill Companies.
- [7] Robert J., Verzello & John Reuter III. 1982. *International Student Edition*. Tokyo : The McGraw-Hill Companies.
- [8] Iqbal, N., M.U. Hassan, A.R. Osman, M. Ahmad. 2013. A framework for partial implementation of PSP in Extreme programming. *International Journal of Engineering Research and Applications*. 3(2): 604-60