

# METODOLOGI SCRUM

## Introduksi

Bagi banyak pengembang industri perangkat lunak, metodologi Agile bukanlah sesuatu yang baru. Metode ini adalah jawaban langsung atas paradigma manajemen proyek tradisional yang dominan pada masanya, yaitu metode Waterfall. Pada tahun 2001, 17 pioner dari metodologi Agile bertemu di Snowbird Ski Resort, Utah, dan menyusun sebuah manifesto bersama mengenai metodologi ini. Manifesto ini kemudian menjadi prinsip dasar dari Agile, dengan penekanan pada komunikasi dan kolaborasi, fungsi perangkat lunak, dan fleksibilitas untuk beradaptasi dengan realitas bisnis yang muncul. Tetapi dari manifesto tersebut, tidak dinyatakan mengenai proses yang konkret, bagaimana saat tim pengembang harus bertemu dengan 3 persoalan yang pasti muncul: deadline, stakeholder, dan tekanan dalam proses *development*.

Muncullah subset-subset dari metodologi Agile, antara lain: Crystal Clear, Extreme Programming, Feature Driven Development, Dynamic Systems Development Method (DSDM), Scrum, dan lainnya. Jadi, Scrum merupakan salah satu subset dari metodologi Agile.

Scrum merupakan sebuah kerangka kerja untuk mengembangkan sebuah produk yang kompleks, di mana visi dari Scrum adalah produk yang bernilai tinggi secara kreativitas maupun produktivitas. Scrum didasari oleh teori kontrol empiris yang disebut empirisme, yang menekankan bahwa pengetahuan didapatkan dari pengalaman sehingga pembuatan keputusan sebaiknya didasari atas pengetahuan tersebut. Scrum menggunakan pendekatan secara *iterative* dan *incremental* untuk mengantisipasi ketidakpastian dalam proyek dan pengendalian resiko.

Tim pengembangan yang menerapkan Scrum, membagi dirinya dengan 3 peran:

1. Product Owner
2. Scrum Master
3. Tim Pengembang

Tim ini kemudian mengatur dirinya sendiri dalam proses, dengan merumuskan cara mereka sendiri untuk memecahkan masalah, tanpa diatur oleh pihak lain di luar anggota tim. Tim ini memiliki semua kompetensi yang dibutuhkan untuk menyelesaikan pekerjaan, tanpa mengandalkan pihak di luar anggota tim.

## A. Cara Kerja SCRUM

Scrum adalah sebuah kerangka kerja yang dilakukan secara berulang-ulang hingga memenuhi kondisi:

1. Mencapai batas deadline yang telah disepakati
2. Produk yang sedang dikembangkan telah memenuhi kebutuhan yang diinginkan
3. Hingga proyek tidak didanai lagi

### 1) *Sprint*

Perulangan atau iterasi pada Scrum memiliki batas waktu (*time-box*) selama 30 hari atau kurang. Dalam Scrum, putaran ini disebut *Sprint*. *Sprint* selalu sama dan konsisten sepanjang pengembangan produk. Apabila Tim Scrum memilih durasi *Sprint* selama 2 minggu, maka sepanjang pengembangan produk, panjang sebuah *Sprint* selalu konstan 2 minggu. Di akhir *Sprint*, Tim Pengembang harus menyelesaikan sebuah potongan produk (*product increment*) yang dapat digunakan oleh pengguna dan berpotensi untuk dirilis ke lingkungan produksi.

### 2) *Event*

*Sprint* merupakan pembungkus untuk semua *event* lain dalam Scrum. *Event Scrum* antara lain adalah:

#### a. *Sprint Planning*

Adalah kegiatan untuk merencanakan pekerjaan yang akan dilakukan oleh Tim Pengembang selama satu *Sprint*. Pada saat *Sprint Planning* dilakukan, *Product Owner* akan datang dengan *Product Backlog* yang sudah disusun berurutan. *Sprint Planning* terbagi dua bagian: 1. Bagian pertama membahas *Product Backlog Item* yang sudah terurut. 2. Bagian kedua membahas bagaimana mengerjakannya. Hasil dari *Sprint Planning* adalah *Sprint Backlog* dan *Sprint Goal*.

#### b. *Daily Scrum Meeting*

Setiap hari saat sebuah *Sprint* berakhir, Tim Pengembang akan mengadakan pertemuan untuk membahas strategi mereka untuk mencapai tujuan obyektif yang telah disepakati dalam *Sprint Planning*. *Daily Scrum Meeting* adalah sebuah

kesempatan bagi Tim Pengembang untuk dapat menginspeksi dan mengadaptasikan hasil pekerjaan mereka hingga hari ini. Tim pengembang yang baru mulai menggunakan Scrum akan menjawab tiga pertanyaan berikut:

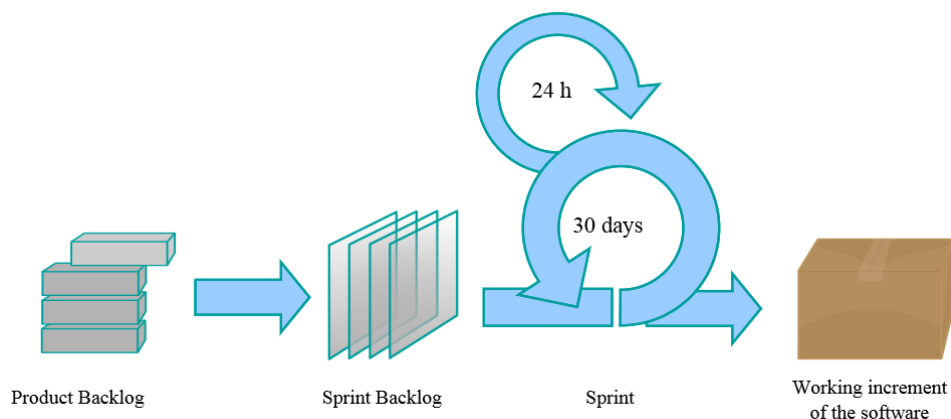
- Apa yang telah saya kontribusikan kemarin
- Apa yang akan saya kontribusikan hari ini
- Apa yang menghambat saya untuk menyelesaikan pekerjaan saya hingga hari ini.

c. ***Sprint Review***

Di akhir *Sprint*, Tim Pengembang bersama *Product Owner* dan *stakeholder* akan *review* potongan produk (*product increment*) yang telah dikembangkan dalam satu *Sprint*. Pada saat *Sprint Review*, seluruh pihak juga membahas perencanaan dan strategi untuk *Sprint* berikutnya.

d. ***Sprint Retrospectives***

Setelah *Sprint Review* berakhir, Tim Scrum akan langsung mengadakan pertemuan untuk membahas apa lagi yang dapat di-*improve* pada *Sprint* berikutnya. Kegiatan ini disebut *Sprint Retrospectives*. Ini adalah sebuah media *continuous improvement* bagi tim Scrum. Tim Scrum yang selalu ingin *improve*, tidak akan melewatkan seremoni tanpa terkecuali. Setelah *Retrospectives* selesai, Tim Scrum akan langsung melakukan *Sprint Planning* di keesokan harinya.



**Gambar 1.** Proses SCRUM

## B. Contoh Kasus SCRUM

### 1) Google AdWords

Proyek AdWords menggunakan Scrum dalam pengembangannya, dengan memiliki tim yang tersebar secara terdistribusi di lima lokasi dan menyatukan secara virtual semua produk Google disetiap dilakukan perilsan. Dalam perkembangannya, manajer dari proyek Google dibutuhkan untuk mengisi beberapa struktur tim. Hadirnya manajer ini pada tim Scrum bertujuan untuk membantu memberikan arahan untuk menyelesaikan prioritas pekerjaan yang tertinggi yang dirasa sulit diimplementasikan oleh tim. Dengan hal ini, tim tidak lagi membutuhkan ScrumMaster, dikarenakan tim sudah dapat berjalan dengan sendirinya.

### 2) SirsiDynix

Proyek SirsiDynix merupakan suatu proyek yang mengimplementasikan sistem perpustakaan warisan dengan lebih dari 12.500 daerah pengimplementasian. Di dalam pengerjaannya yang sudah berlangsung selama bertahun-tahun terdapat perubahan lingkungan bisnis yang menangani permintaan-permintaan baru di tengah pengerjaan proyek. Dan yang paling menyulitkan, industri pengembangan perangkat lunak perpustakaan sedang dalam fase konsolidasi.

Menjadi suatu keuntungan, Dynix memulai proyek dengan menggunakan proses Agile yang memungkinkan adaptasi terhadap perubahan permintaan selama proyek berlangsung. Dalam pengembangannya, proyek ini juga didukung oleh StarSoft yang memiliki sejarah penggunaan XP yang memberikan hasil yang baik dalam pengembangan perangkat lunak berbasis sistem. Kombinasi dari resiko yang tinggi, skala yang besar, perubahan permintaan, konsolidasi dan akuisisifaktor bisnis, dan pengalaman SirsiDynix dengan Scrum yang dikombinasikan dengan StarSoft yang menggunakan XP mengantarkan proyek ini untuk menggunakan implementasi Scrum yang terintegrasi. Scrum yang terintegrasi dilakukan dengan paradigma terdistribusi, sehingga dalam praktik pengembangan proyek ini, aktifitas dari Scrum dilakukan secara terdistribusi. Daily Scrum untuk seluruh pengembangan dilakukan dengan terintegrasi untuk seluruh *site*, begitu juga untuk pertemuan harian tim dari Product Owner yang dilakukan secara terintegrasi untuk seluruh *site*.

### **C. Keunggulan SCRUM**

- 1) Mudah menyesuaikan diri dengan perubahan yang terjadi pada realitas bisnis dengan cepat
- 2) Tim yang besar dipecah menjadi tim-tim kecil yang disebut Tim Scrum, sehingga memperlancar komunikasi, mengurangi biaya, dan saling memberdayakan satu sama lain
- 3) Dokumentasi dan pengujian selama proses pengembangan dilakukan secara terus menerus
- 4) Proses Scrum dapat menyatakan bahwa suatu proses pengembangan “sudah selesai” kapan saja

### **D. Kelemahan SCRUM**

Tim Pengembang dengan Scrum harus bersedia dan selalu siap untuk menerima perubahan yang terjadi di dalam proses pengembangan.

### **E. Tipe Proyek SCRUM**

Scrum direkomendasikan untuk tipe proyek yang tidak berorientasi pada *deadline*, dengan target produk yang kreatif dan berkualitas tinggi. Karena fokus Scrum adalah pada proses, dengan iteratif dan target per *Sprint*, diharapkan setiap potongan produk, akan menjadi lebih baik dari *Sprint* sebelumnya.

## DAFTAR PUSTAKA

*Scrum Methodology*. [online]. <http://scrummethodology.com/>, diakses tanggal 21 Oktober 2014.

Wikipedia. *Scrum (software development)*. [online].  
([http://en.wikipedia.org/wiki/Scrum\\_\(software\\_development\)](http://en.wikipedia.org/wiki/Scrum_(software_development))), diakses tanggal 22 Oktober 2014.

PT. Scrum Asia Pasifik. *Apa itu Scrum*. [online]. (<http://www.scrum.co.id/static/apa-itu-scrum>),  
diakses tanggal 21 Oktober 2014.

Pangaribuan, Andreas. 2012. *Metode Pengembangan Perangkat Lunak, Scrum*. [online].  
(<https://www.academia.edu/6288669/Scrum>), diakses tanggal 24 Oktober 2014.