

Ken Schwaber & Jeff Sutherland

Panduan Scrum

Panduan Definitif untuk Scrum: Aturan Permainan

November 2020

Tujuan Panduan Scrum

Kami mengembangkan Scrum di awal 1990-an. Kami menulis Panduan Scrum versi pertama pada tahun 2010 untuk membantu orang di seluruh dunia memahami Scrum. Kami telah mengembangkan Panduan ini dengan berbagai pembaruan fungsional kecil sejak saat itu. Bersama-sama, kami berdiri di belakang Panduan ini.

Panduan Scrum berisi definisi Scrum. Setiap elemen kerangka kerja ini melayani tujuan spesifik yang penting untuk keseluruhan nilai dan hasil yang dicapai dengan Scrum. Mengubah desain atau ide-ide inti Scrum, mengabaikan elemen, atau tidak mengikuti aturan Scrum, akan menutupi masalah dan membatasi manfaat Scrum, bahkan berpotensi menghilangkan manfaatnya.

Kami mengikuti perkembangan penggunaan Scrum yang terus berkembang dalam ruang kompleks. Kami dengan rendah hati melihat Scrum diadopsi di banyak ruang yang pada dasarnya memiliki pekerjaan yang kompleks, di luar pengembangan produk perangkat lunak di mana Scrum berakar. Saat penggunaan Scrum menyebar, *Developers*, peneliti, analis, ilmuwan, dan spesialis lainnya melakukan pekerjaan itu. Kami menggunakan kata "*Developers*" di Scrum untuk tidak mengecualikan, tetapi untuk menyederhanakan. Jika Anda mendapatkan nilai dari Scrum, anggap diri Anda termasuk dari *Developers*.

Saat Scrum digunakan, pola, proses, dan wawasan yang selaras dengan kerangka kerja Scrum seperti dijelaskan dalam dokumen ini, mungkin untuk kemudian ditemukan, diterapkan dan dibuat. Penjelasan tentang hal tersebut melampaui tujuan dari Panduan Scrum karena hal tersebut peka konteks dan sangat berbeda antar penggunaan Scrum. Taktik untuk penggunaannya dalam kerangka kerja Scrum sangat bervariasi dan dijelaskan di tempat lain.

Ken Schwaber & Jeff Sutherland November 2020

© 2020 Ken Schwaber dan Jeff Sutherland

This publication is offered for license under the Attribution Share-Alike license of Creative Commons, accessible at <https://creativecommons.org/licenses/by-sa/4.0/legalcode> and also described in summary form at <https://creativecommons.org/licenses/by-sa/4.0/>. By utilizing this Scrum Guide, you acknowledge and agree that you have read and agree to be bound by the terms of the Attribution Share-Alike license of Creative Commons.

Tujuan Panduan Scrum	1
Definisi Scrum	3
Teori Scrum	3
Transparansi	3
Inspeksi	4
Adaptasi	4
Nilai-Nilai Scrum	4
Scrum Team	5
Developers	5
Product Owner	5
Scrum Master	6
Acara Scrum	7
Sprint	7
Sprint Planning	8
Daily Scrum	9
Sprint Review	9
Sprint Retrospective	10
Artefak Scrum	10
Product Backlog	10
Komitmen: Product Goal	11
Sprint Backlog	11
Komitmen: Sprint Goal	11
Increment	12
Komitmen: Definition of Done	12
Catatan Akhir	13
Ucapan Terima Kasih	13
Orang-orang	13
Sejarah Panduan Scrum	13
Terjemahan	13
Perubahan dari Panduan Scrum 2017 menjadi Panduan Scrum 2020	14
Catatan Penerjemah	15

Definisi Scrum

Scrum adalah kerangka kerja ringan yang membantu orang, tim, dan organisasi untuk menghasilkan nilai melalui solusi adaptif untuk masalah yang kompleks.

Singkatnya, Scrum membutuhkan seorang *Scrum Master* untuk memelihara lingkungan di mana:

1. *Product Owner* mengurutkan pekerjaan untuk masalah yang kompleks ke dalam *Product Backlog*.
2. *Scrum Team* mengubah sebagian pekerjaan yang terpilih menjadi nilai yang meningkat (*Increment Nilai*) selama suatu *Sprint*.
3. *Scrum Team* dan para pemangku kepentingannya memeriksa hasil dan menyesuaikan untuk *Sprint* berikutnya.
4. Ulangi

Scrum itu sederhana. Cobalah apa adanya dan tentukan apakah filosofi, teori, dan strukturnya membantu mencapai tujuan dan menciptakan nilai. Kerangka Kerja Scrum sengaja dibuat tidak lengkap, hanya mendefinisikan bagian-bagian yang diperlukan untuk menerapkan teori Scrum. Scrum dibangun berdasarkan kecerdasan kolektif orang-orang yang menggunakannya. Daripada memberikan instruksi terperinci, aturan Scrum memandu hubungan dan interaksi mereka.

Berbagai proses, teknik dan metode dapat digunakan di dalam kerangka kerja ini. Scrum merangkul seputar praktik yang ada atau menjadikannya tidak perlu. Scrum menunjukkan keefektifan relatif dari manajemen, lingkungan, dan teknik kerja saat ini, sehingga perbaikan dapat dilakukan.

Teori Scrum

Scrum didasarkan pada empirisme dan pemikiran lean. Empirisme menegaskan bahwa pengetahuan itu berasal dari pengalaman dan membuat keputusan berdasarkan apa yang diamati. Pemikiran lean mengurangi pemborosan dan fokus pada hal-hal yang terpenting.

Scrum menggunakan pendekatan iteratif dan inkremental untuk mengoptimalkan prediktabilitas dan mengendalikan risiko. Scrum melibatkan kelompok orang yang secara kolektif memiliki semua keterampilan dan keahlian untuk melakukan pekerjaan dan berbagi atau memperoleh keterampilan sebagaimana yang dibutuhkan.

Scrum menggabungkan empat acara formal untuk inspeksi dan adaptasi dalam suatu acara yang mewadahi, yaitu *Sprint*. Semua acara ini berhasil karena menerapkan pilar Scrum empiris yaitu **transparansi**, **inspeksi**, dan **adaptasi**.

Transparansi

Proses dan pekerjaan yang muncul harus dapat dilihat oleh mereka yang melakukan pekerjaan serta mereka yang menerima hasil pekerjaan. Dengan Scrum, keputusan penting didasarkan pada kondisi tiga artefak formal. Artefak yang memiliki transparansi rendah dapat mengarah kepada keputusan yang mengurangi nilai dan meningkatkan risiko.

Transparansi memungkinkan inspeksi. Inspeksi tanpa transparansi adalah menyesatkan dan sia-sia.

Inspeksi

Artefak Scrum dan kemajuan menuju tujuan yang disepakati harus sering diperiksa dan mendeteksi varian atau masalah yang mungkin tidak diinginkan secara seksama. Untuk membantu inspeksi, Scrum menyediakan ruang dalam bentuk lima acara.

Inspeksi memungkinkan adaptasi. Inspeksi tidak ada gunanya tanpa adaptasi. Acara scrum dirancang untuk memicu perubahan.

Adaptasi

Jika ada aspek proses yang menyimpang di luar batas yang dapat diterima atau jika produk yang dihasilkan tidak dapat diterima, proses yang diterapkan atau bahan yang diproduksi harus disesuaikan. Penyesuaiannya harus dibuat sesegera mungkin untuk meminimalkan penyimpangan lebih jauh.

Adaptasi menjadi lebih sulit ketika orang-orang yang terlibat tidak diberdayakan atau melakukan manajemen mandiri. Suatu *Scrum Team* diharapkan untuk beradaptasi seketika saat mempelajari sesuatu yang baru melalui inspeksi.

Nilai-Nilai Scrum

Keberhasilan penggunaan Scrum bergantung pada orang-orang yang menjadi lebih mahir dalam menjalankan lima nilai:

Komitmen, Fokus, Keterbukaan, Rasa Hormat, dan Keberanian

Scrum Team berkomitmen untuk mencapai tujuannya dan mendukung satu sama lain. Fokus utama mereka adalah pekerjaan dalam *Sprint* untuk membuat kemajuan sebaik mungkin menuju tujuannya. *Scrum Team* dan tim pemangku kepentingan saling terbuka tentang pekerjaan dan tantangannya. Anggota *Scrum Team* saling menghormati agar menjadi orang yang mampu, mandiri, dan dihormati oleh orang-orang yang bekerja dengan mereka. Anggota *Scrum Team* memiliki keberanian untuk melakukan hal yang benar, untuk mengatasi masalah yang sulit.

Nilai-nilai ini memberikan arahan kepada *Scrum Team* terkait dengan pekerjaan, tindakan, dan perilaku mereka. Segala keputusan yang dibuat, langkah yang diambil, dan cara menerapkan Scrum seharusnya memperkuat nilai-nilai ini, bukan mengurangi atau merusaknya. Anggota *Scrum Team* belajar dan mengeksplorasi nilai-nilai tersebut saat mereka memanfaatkan acara dan artefak Scrum. Saat nilai-nilai ini diterapkan oleh *Scrum Team* dan orang-orang yang bekerja dengannya, pilar empiris Scrum yaitu transparansi, inspeksi, dan adaptasi menjadi hidup dan membangun kepercayaan.

Scrum Team

Unit fundamental dari Scrum adalah sebuah tim kecil, sebuah *Scrum Team*. *Scrum Team* terdiri dari satu *Scrum Master*, satu *Product Owner*, dan *Developers*. Di dalam *Scrum Team*, tidak ada sub-tim atau hierarki. Ini adalah unit kohesif profesional yang berfokus pada satu tujuan pada satu waktu, **Product Goal**.

Scrum Team bersifat lintas fungsi, artinya anggota tim memiliki semua keterampilan yang diperlukan untuk menciptakan nilai di setiap *Sprint*. Mereka juga melakukan manajemen mandiri, artinya mereka secara internal memutuskan siapa melakukan apa, kapan, dan bagaimana.

Scrum Team cukup kecil untuk tetap gesit dan cukup besar untuk menyelesaikan pekerjaan penting dalam *Sprint*, biasanya 10 orang atau kurang. Secara umum, kami menemukan bahwa tim yang lebih kecil berkomunikasi dengan lebih baik dan lebih produktif. Jika *Scrum Team* menjadi terlalu besar, mereka harus mempertimbangkan untuk mengatur ulang menjadi beberapa *Scrum Team* yang kohesif, masing-masing berfokus pada produk yang sama. Oleh karena itu, mereka harus berbagi *Product Goal*, *Product Backlog*, dan *Product Owner* yang sama.

Scrum Team bertanggung jawab atas semua aktivitas terkait produk mulai dari kolaborasi dengan pemangku kepentingan, verifikasi, pemeliharaan, pengoperasian, eksperimen, penelitian dan pengembangan, dan hal lain yang mungkin diperlukan. Mereka disusun dan diberdayakan oleh organisasi untuk mengelola pekerjaan mereka sendiri. Bekerja dalam *Sprint* dengan kecepatan yang berkelanjutan meningkatkan fokus dan konsistensi *Scrum Team*.

Seluruh *Scrum Team* bertanggung jawab untuk menciptakan **Increment yang bernilai dan berguna di setiap Sprint**. Scrum menetapkan tiga akuntabilitas khusus dalam *Scrum Team*: *Developers*, *Product Owner*, dan *Scrum Master*.

Developers

Developers adalah anggota *Scrum Team* yang berkomitmen untuk membuat segala aspek dari *Increment* yang dapat digunakan, di setiap *Sprint*.

Keterampilan khusus yang dibutuhkan oleh *Developers* seringkali luas dan akan bervariasi tergantung pada domain pekerjaan. Namun, *Developers* selalu bertanggung jawab untuk:

- Membuat rencana untuk *Sprint*, *Sprint Backlog*;
- Menanamkan kualitas dengan mengikuti *Definition of Done*;
- Menyesuaikan rencana mereka setiap hari untuk mencapai *Sprint Goal*; dan,
- Saling bertanggung jawab sebagai profesional.

Product Owner

Product Owner bertanggung jawab untuk memaksimalkan nilai produk yang dihasilkan dari kinerja *Scrum Team*. Caranya dapat sangat bervariasi antar organisasi, *Scrum Team*, dan

individu.

Product Owner juga bertanggung jawab atas pengelolaan *Product Backlog* yang efektif, yang meliputi:

- Mengembangkan dan secara eksplisit mengkomunikasikan *Product Goal*;
- Membuat dan mengkomunikasikan *Product Backlog Item* dengan jelas;
- Memprioritaskan *Product Backlog Item*; dan,
- Memastikan *Product Backlog* transparan, terlihat dan dipahami.

Product Owner dapat melakukan pekerjaan di atas atau dapat mendelegasikan tanggung jawab kepada orang lain. Terlepas dari itu, *Product Owner* tetap bertanggung jawab.

Agar *Product Owner* berhasil, seluruh organisasi harus menghormati keputusan mereka. Keputusan ini terlihat dalam konten dan urutan *Product Backlog*, dan melalui *Increment* yang dapat diperiksa di *Sprint Review*.

Product Owner adalah satu orang, bukan komite. *Product Owner* dapat mewakili kebutuhan banyak pemangku kepentingan dalam *Product Backlog*. Mereka yang ingin mengubah *Product Backlog* dapat melakukannya dengan mencoba meyakinkan *Product Owner*.

Scrum Master

Scrum Master bertanggung jawab untuk menerapkan Scrum seperti yang didefinisikan dalam Panduan Scrum. Mereka melakukan ini dengan membantu semua orang memahami teori dan praktik Scrum, baik di dalam *Scrum Team* maupun organisasi.

Scrum Master bertanggung jawab atas efektivitas *Scrum Team*. Mereka melakukan ini dengan memungkinkan *Scrum Team* untuk meningkatkan praktiknya, dalam kerangka kerja Scrum.

Scrum Master adalah pemimpin sejati yang melayani *Scrum Team* dan organisasi yang lebih besar.

Scrum Master melayani *Scrum Team* dengan beberapa cara, termasuk:

- Melatih anggota tim dalam manajemen mandiri dan lintas fungsi;
- Membantu *Scrum Team* fokus pada pembuatan *Increment* bernilai tinggi yang memenuhi *Definition of Done*;
- Menghilangkan hambatan untuk kemajuan *Scrum Team*; dan,
- Memastikan bahwa semua acara Scrum berlangsung dan bersifat positif, produktif, dan tidak melebihi batas waktu.

Scrum Master melayani *Product Owner* dengan beberapa cara, termasuk:

- Membantu menemukan teknik untuk menjelaskan *Product Goal* dan mengelola *Product Backlog* yang efektif;

- Membantu *Scrum Team* memahami kebutuhan akan *Product Backlog Item* yang jelas dan ringkas;
- Membantu menetapkan perencanaan produk empiris untuk lingkungan yang kompleks; dan,
- Memfasilitasi kolaborasi dengan pemangku kepentingan sesuai permintaan atau kebutuhan.

Scrum Master melayani organisasi dengan beberapa cara, termasuk:

- Memimpin, melatih, dan membimbing organisasi dalam penerapan Scrum;
- Merencanakan dan memberi saran tentang implementasi Scrum dalam organisasi;
- Membantu karyawan dan pemangku kepentingan memahami dan menggunakan pendekatan empiris untuk pekerjaan yang kompleks; dan,
- Menghilangkan hambatan antara pemangku kepentingan dan *Scrum Team*.

Acara Scrum

Sprint adalah wadah untuk semua acara lainnya. Setiap acara di Scrum adalah kesempatan formal untuk memeriksa dan mengadaptasi artefak Scrum. Semua acara ini secara khusus dirancang untuk memungkinkan transparansi yang diperlukan. Kegagalan untuk menjalankan salah satu acara seperti yang ditentukan mengakibatkan hilangnya peluang untuk memeriksa dan beradaptasi. Acara digunakan di dalam Scrum untuk menciptakan keteraturan dan meminimalkan kebutuhan akan pertemuan yang tidak ditentukan di Scrum. Agar lebih optimal, semua acara diadakan pada waktu dan tempat yang sama untuk mengurangi kerumitan.

Sprint

Sprint adalah detak jantung Scrum, di mana ide diubah menjadi nilai.

Sprint adalah acara berdurasi tetap selama satu bulan atau kurang agar tercipta konsistensi. Segera setelah *Sprint* sebelumnya berakhir, langsung dimulailah *Sprint* yang baru.

Semua pekerjaan yang diperlukan untuk mencapai *Product Goal*, termasuk *Sprint Planning*, *Daily Scrum*, *Sprint Review*, dan *Sprint Retrospective*, terjadi dalam *Sprint*.

Selama *Sprint*:

- Tidak ada perubahan yang dilakukan yang akan membahayakan *Sprint Goal*;
- Kualitas tidak boleh menurun;
- *Product Backlog* disempurnakan sesuai kebutuhan; dan,
- Cakupan dapat diklarifikasi dan dinegosiasikan ulang dengan *Product Owner* karena adanya temuan baru.

Sprint memungkinkan prediktabilitas dengan setidaknya setiap bulan kalender memastikan Inspeksi dan adaptasi tentang progres menuju *Product Goal*. Ketika jangkauan *Sprint* terlalu panjang, *Sprint Goal* mungkin menjadi tidak valid, kompleksitas bisa meningkat, dan risiko

bisa meningkat. *Sprint* yang lebih pendek dapat digunakan untuk menghasilkan lebih banyak siklus pembelajaran dan membatasi risiko biaya dan upaya dalam kerangka waktu yang lebih kecil. Setiap *Sprint* dapat dianggap sebagai proyek pendek.

Berbagai cara tersedia untuk memperkirakan progres, seperti burn-down, burn-up, atau arus kumulatif. Meskipun terbukti bermanfaat, hal ini tidak menggantikan pentingnya empirisme. Dalam lingkungan yang kompleks, kita tidak mengetahui apa yang akan terjadi. Hanyalah hal-hal yang telah terjadi yang dapat digunakan untuk pengambilan keputusan berwawasan ke depan. *Sprint* dapat dibatalkan jika *Sprint Goal* sudah menjadi usang. Hanya *Product Owner* yang memiliki kewenangan untuk membatalkan *Sprint*.

Sprint Planning

Sprint dimulai dengan *Sprint Planning* yang mengatur pekerjaan yang akan dilakukan selama *Sprint*. Rencana yang dihasilkan ini dibuat secara kolaboratif oleh seluruh *Scrum Team*.

Product Owner memastikan bahwa peserta telah siap untuk membahas *Product Backlog Item* yang paling penting dan bagaimana hubungannya dengan *Product Goal*. *Scrum Team* juga dapat mengundang orang lain untuk menghadiri *Sprint Planning* untuk memberikan masukan.

Sprint Planning membahas topik-topik berikut:

Topik Pertama: **Mengapa *Sprint* ini berharga?**

Product Owner mengusulkan bagaimana produk dapat meningkatkan nilai dan kegunaannya di *Sprint* saat ini. Seluruh *Scrum Team* kemudian berkolaborasi untuk menentukan *Sprint Goal* yang mengkomunikasikan mengapa *Sprint* berharga bagi para pemangku kepentingan. *Sprint Goal* harus diselesaikan sebelum *Sprint Planning* berakhir.

Topik Kedua: **Apa yang Dapat Dilakukan dalam *Sprint* ini?**

Melalui diskusi dengan *Product Owner*, *Developers* memilih item dari *Product Backlog* untuk dimasukkan ke dalam *Sprint* saat ini. *Scrum Team* dapat menyempurnakan item-item terpilih selama proses ini, yang akan meningkatkan pemahaman dan kepercayaan diri.

Memilih seberapa banyak yang bisa diselesaikan dalam *Sprint* mungkin memberikan tantangan tersendiri. Namun, semakin *Developers* memahami kinerja mereka sebelumnya, kapasitas yang akan datang, dan *Definition of Done*, mereka akan semakin mantap akan perkiraan *Sprint* mereka.

Topik Ketiga: **Bagaimana pekerjaan yang dipilih diselesaikan?**

Untuk setiap *Product Backlog Item* yang terpilih, *Developers* merencanakan pekerjaan yang diperlukan untuk membuat *Increment* yang memenuhi *Definition of Done*. Hal ini biasanya dilakukan dengan menguraikan *Product Backlog Item* menjadi item pekerjaan yang lebih

kecil yang dapat diselesaikan dalam satu hari atau kurang. Hal ini dilakukan sepenuhnya atas kebijaksanaan *Developers*. Tidak ada orang lain yang memberi tahu mereka bagaimana cara mengubah *Product Backlog Item* menjadi *Increment* Nilai.

Sprint Goal, *Product Backlog Item* yang dipilih untuk *Sprint*, ditambah rencana untuk menyampaikannya bersama-sama disebut sebagai *Sprint Backlog*.

Sprint Planning memiliki batas waktu maksimum delapan jam untuk *Sprint* selama satu bulan. Untuk *Sprint* yang lebih pendek, acaranya biasanya lebih singkat.

Daily Scrum

Tujuan dari *Daily Scrum* adalah untuk memeriksa kemajuan menuju *Sprint Goal* dan menyesuaikan *Sprint Backlog* seperlunya, menyesuaikan rencana kerja yang akan datang.

Daily Scrum adalah acara 15 menit untuk para *Developers* dalam *Scrum Team*. Untuk mengurangi kerumitan, acara ini diadakan di waktu dan tempat yang sama setiap hari kerja dalam *Sprint*. Jika *Product Owner* atau *Scrum Master* secara aktif mengerjakan item di *Sprint Backlog*, mereka berpartisipasi sebagai *Developers*.

Developers dapat memilih struktur dan teknik apa pun yang mereka inginkan, selama *Daily Scrum* mereka berfokus pada kemajuan menuju *Sprint Goal* dan menghasilkan rencana yang dapat ditindaklanjuti untuk hari kerja berikutnya. Ini menciptakan fokus dan meningkatkan manajemen mandiri.

Daily Scrum meningkatkan komunikasi, mengidentifikasi hambatan, mendorong pengambilan keputusan yang cepat, dan akibatnya menghilangkan kebutuhan untuk pertemuan-pertemuan lainnya.

Daily Scrum bukan satu-satunya kesempatan bagi *Developers* untuk menyesuaikan rencana mereka. Mereka sering bertemu sepanjang hari untuk diskusi yang lebih mendetail tentang mengadaptasi atau merencanakan ulang segala pekerjaan dalam *Sprint*.

Sprint Review

Tujuan dari *Sprint Review* adalah untuk memeriksa hasil dari *Sprint* dan menentukan adaptasi selanjutnya. *Scrum Team* mempresentasikan hasil kerja mereka kepada pemangku kepentingan utama dan mendiskusikan kemajuan menuju *Product Goal*.

Selama acara ini berlangsung, *Scrum Team* dan pemangku kepentingan meninjau apa yang telah dicapai selama *Sprint* dan apa yang telah berubah di sekitar mereka. Berdasarkan informasi ini, peserta pertemuan berkolaborasi tentang apa yang harus dilakukan selanjutnya. *Product Backlog* juga dapat disesuaikan untuk memenuhi peluang baru. *Sprint Review* adalah sesi kerja dan *Scrum Team* harus menghindari membatasi hanya pada presentasi.

Sprint Review adalah acara terakhir kedua dari *Sprint* dan memiliki batas waktu maksimum empat jam untuk *Sprint* selama satu bulan. Untuk *Sprint* yang lebih pendek, acaranya

biasanya lebih singkat.

Sprint Retrospective

Tujuan dari *Sprint Retrospective* adalah untuk merencanakan cara untuk meningkatkan kualitas dan efektivitas.

Scrum Team mengkaji bagaimana berjalannya *Sprint* terakhir berkaitan dengan individu, interaksi, proses, alat, dan *Definition of Done* mereka. Elemen yang dikaji sering kali berbeda pada tiap domain pekerjaan. Asumsi yang menyesatkan mereka diidentifikasi dan asal-usulnya dieksplorasi. *Scrum Team* membahas apa yang berjalan dengan baik selama *Sprint*, masalah apa yang ditemuinya, dan bagaimana masalah itu telah/akan (atau tidak) diselesaikan.

Scrum Team mengidentifikasi perubahan yang paling berguna demi meningkatkan efektivitasnya. Perbaikan yang paling berdampak akan ditangani secepat mungkin. Hal tersebut bahkan dapat ditambahkan ke *Sprint Backlog* berikutnya.

Sprint Retrospective menutup *Sprint*. Batas waktunya maksimum tiga jam untuk *Sprint* selama satu bulan. Untuk *Sprint* yang lebih pendek, acaranya biasanya lebih singkat.

Artefak Scrum

Artefak Scrum mewakili hasil karya atau nilai. Mereka dirancang untuk memaksimalkan transparansi informasi utama. Sehingga, setiap orang yang memeriksanya memiliki dasar yang sama untuk adaptasi.

Setiap artefak berisi komitmen untuk memastikannya memberikan informasi yang meningkatkan transparansi dan fokus yang dapat diukur kemajuannya:

- Untuk *Product Backlog*, adalah *Product Goal*.
- Untuk *Sprint Backlog*, adalah *Sprint Goal*.
- Untuk *Increment*, adalah *Definition of Done*.

Komitmen ini ada untuk memperkuat empirisme dan nilai-nilai Scrum bagi *Scrum Team* dan pemangku kepentingan mereka.

Product Backlog

Product Backlog adalah daftar yang muncul dan tersusun tentang apa yang dibutuhkan untuk meningkatkan produk. Ini adalah satu-satunya sumber pekerjaan yang dilakukan oleh *Scrum Team*.

Product Backlog Item yang dapat dilakukan oleh *Scrum Team* dalam satu *Sprint* dianggap siap untuk dipilih dalam acara *Sprint Planning*. Mereka biasanya memperoleh tingkat transparansi ini setelah melakukan aktivitas pembahasan. Pembahasan *Product Backlog* adalah tindakan memecah dan mendefinisikan lebih lanjut *Product Backlog Item* menjadi item yang lebih kecil dan lebih akurat. Ini adalah aktivitas berkelanjutan untuk

menambahkan detail seperti deskripsi, urutan, dan ukuran. Atribut seringkali berbeda pada tiap domain pekerjaan.

Developers yang akan melakukan pekerjaan bertanggung jawab atas penentuan ukuran tersebut. *Product Owner* dapat memengaruhi *Developers* dengan membantu mereka memahami dan menentukan pilihan kompromi.

Komitmen: *Product Goal*

Product Goal menggambarkan keadaan produk di masa depan yang dapat dijadikan sebagai target untuk direncanakan oleh *Scrum Team*. *Product Goal* ada di *Product Backlog*. *Product Backlog* lainnya muncul untuk mendefinisikan "apa" yang akan memenuhi *Product Goal*.

Produk adalah kendaraan untuk menyampaikan nilai. Produk memiliki batasan yang jelas, pemangku kepentingan yang diketahui, pengguna atau pelanggan yang ditentukan dengan baik. Produk bisa menjadi layanan, produk fisik, atau sesuatu yang lebih abstrak.

Product Goal adalah tujuan jangka panjang *Scrum Team*. Mereka harus memenuhi (atau meninggalkan) satu tujuan sebelum mengambil tujuan berikutnya.

Sprint Backlog

Sprint Backlog terdiri dari *Sprint Goal* (mengapa), serangkaian *Product Backlog Item* yang dipilih untuk *Sprint* (apa), serta rencana yang dapat ditindaklanjuti untuk menghasilkan *Increment* (bagaimana).

Sprint Backlog adalah rencana oleh dan untuk *Developers*. Ini adalah gambaran waktu sebenarnya yang sangat terlihat dari pekerjaan yang direncanakan *Developers* selama *Sprint* untuk mencapai *Sprint Goal*. Karena itu, *Sprint Backlog* diperbarui sepanjang *Sprint* seiring dengan semakin banyaknya hal yang baru diketahui. *Sprint Backlog* ini harus memiliki perincian yang cukup sehingga dapat memeriksa kemajuan mereka di *Daily Scrum*.

Komitmen: *Sprint Goal*

Sprint Goal adalah satu-satunya tujuan dari *Sprint*. Meskipun *Sprint Goal* merupakan komitmen *Developers*, *Sprint Goal* memberikan fleksibilitas dalam hal pekerjaan yang dibutuhkan untuk mencapainya. *Sprint Goal* juga menciptakan koherensi dan fokus, mendorong *Scrum Team* untuk bekerja sama dan bukan dengan inisiatif terpisah.

Sprint Goal dibuat selama *Sprint Planning* dan kemudian ditambahkan ke *Sprint Backlog*. Saat *Developers* bekerja selama *Sprint*, mereka selalu mengingat *Sprint Goal*. Jika pekerjaan ternyata berbeda dari yang mereka harapkan, mereka bekerja sama dengan *Product Owner* untuk menegosiasikan ruang lingkup *Sprint Backlog* dalam *Sprint* tanpa mempengaruhi *Sprint Goal*.

Increment

Increment adalah pijakan menuju *Product Goal*. Setiap Penambahan merupakan tambahan untuk semua *Increment* sebelumnya dan diverifikasi secara menyeluruh, memastikan bahwa semua *Increment* berfungsi secara bersama. Agar bernilai, *Increment* harus dapat digunakan.

Dalam sebuah *Sprint*, dapat dibuat beberapa *Increment*. Keseluruhan *Increment* disajikan di *Sprint Review* sehingga mendukung empirisme. Namun, *Increment* dapat dikirimkan ke pemangku kepentingan sebelum akhir *Sprint*. *Sprint Review* tidak boleh dianggap sebagai gerbang untuk merilis nilai.

Suatu pekerjaan tidak dapat dianggap sebagai bagian dari *Increment* kecuali jika memenuhi *Definition of Done*.

Komitmen: Definition of Done

Definition of Done adalah deskripsi formal dari keadaan *Increment* jika memenuhi ukuran kualitas yang disyaratkan untuk produk.

Saat *Product Backlog Item* memenuhi *Definition of Done*, sebuah *Increment* lahir.

Definition of Done menciptakan transparansi dengan memberikan semua orang pemahaman bersama tentang pekerjaan apa yang diselesaikan sebagai bagian dari *Increment*. Jika *Product Backlog Item* tidak memenuhi *Definition of Done*, item tersebut tidak dapat dirilis atau bahkan disajikan di *Sprint Review*. Sebaliknya, dikembalikan ke *Product Backlog* untuk pertimbangan di masa mendatang.

Jika *Definition of Done* untuk sebuah *Increment* adalah bagian dari standar organisasi, semua *Scrum Team* harus mengikutinya sebagai standar minimal. Jika bukan standar organisasi, *Scrum Team* harus membuat *Definition of Done* yang sesuai untuk produk tersebut.

Developers diharuskan untuk memenuhi *Definition of Done*. Jika ada beberapa *Scrum Team* yang bekerja bersama dalam sebuah produk, mereka harus saling mendefinisikan dan mematuhi *Definition of Done* yang sama.

Catatan Akhir

Scrum bersifat gratis dan ditawarkan dalam Panduan ini. Kerangka kerja Scrum, seperti yang dijelaskan di sini, tidak dapat diubah. Meskipun dimungkinkan untuk hanya mengimplementasikan sebagian Scrum, hasilnya bukanlah Scrum. Scrum hanya ada secara keseluruhan dan berfungsi dengan baik sebagai wadah untuk teknik, metodologi, dan praktik lainnya.

Ucapan Terima Kasih

Orang-orang

Dari ribuan orang yang telah berkontribusi pada Scrum, kita harus memilih mereka yang berperan di awal: Jeff Sutherland bekerja dengan Jeff McKenna dan John Scumniotales, dan Ken Schwaber bekerja dengan Mike Smith dan Chris Martin, dan semuanya bekerja sama. Banyak orang lain yang berkontribusi di tahun-tahun berikutnya dan tanpa bantuan mereka Scrum tidak akan menjadi seperti sekarang ini.

Sejarah Panduan Scrum

Ken Schwaber dan Jeff Sutherland pertama kali bersama-sama mempresentasikan Scrum di Konferensi OOPSLA pada tahun 1995. Saat itu pada dasarnya mendokumentasikan pembelajaran yang diperoleh Ken dan Jeff selama beberapa tahun sebelumnya dan mempublikasikan definisi formal pertama Scrum.

Panduan Scrum mendokumentasikan Scrum sebagai dikembangkan, berkembang, dan dipertahankan selama lebih dari 30 tahun oleh Jeff Sutherland dan Ken Schwaber. Sumber lain memberikan pola, proses, dan wawasan yang melengkapi kerangka kerja Scrum. Hal tersebut dapat meningkatkan produktivitas, nilai, kreativitas, dan kepuasan dengan hasil.

Sejarah lengkap Scrum dijelaskan di tempat lain. Untuk menghormati tempat pertama di mana Scrum dicoba dan dibuktikan, kami mengakui Individual Inc., Newpage, Fidelity Investments, dan IDX (sekarang GE Medical).

Terjemahan

Panduan ini telah diterjemahkan dari versi asli berbahasa Inggris yang disediakan oleh pengembang di atas. Terjemahan ini dikelola oleh [Warkop Agile](mailto:warkopagile@gmail.com) (warkopagile@gmail.com). Informasi dan saran tentang terjemahan ini, dapat menghubungi *Contact Person*:

- Bie - warkopagile@gmail.com

Perubahan dari Panduan Scrum 2017 menjadi Panduan Scrum 2020

Makin Kurang Preskriptif

Tahun demi tahun, Panduan Scrum menjadi semakin preskriptif. Versi 2020 bertujuan untuk mengembalikan Scrum menjadi kerangka kerja yang minimal dan memadai dengan menghapus atau mengurangi bahasa yang preskriptif. Misalnya menghapus pertanyaan *Daily Scrum*, memperhalus bahasa di sekitar atribut *PBI*, memperhalus bahasa di sekitar *item retro* di *Sprint Backlog*, mempersingkat bagian pembatalan *Sprint*, dan banyak lagi.

Satu Tim, Berfokus pada Satu Produk

Tujuannya adalah untuk menghilangkan konsep tim terpisah dalam sebuah tim yang menyebabkan perilaku "*proxy*" atau "kami dan mereka" antara PO dan Tim *Developers*. Sekarang hanya ada satu *Scrum Team* yang berfokus pada tujuan yang sama, dengan tiga set akuntabilitas yang berbeda: PO, SM, dan *Developers*.

Pengenalan Tujuan Produk

Panduan Scrum 2020 memperkenalkan konsep *Product Goal* untuk memberikan fokus bagi *Scrum Team* menuju tujuan yang lebih besar dan berharga. Setiap *Sprint* harus membawa produk lebih dekat ke keseluruhan *Product Goal*.

Tempat untuk Sprint Goal, Definisi Selesai, dan Tujuan Produk

Panduan Scrum sebelumnya menjelaskan *Sprint Goal* dan *Definition of Done* tanpa benar-benar memberi mereka identitas. Mereka bukan artefak tapi agak melekat pada artefak. Dengan tambahan *Product Goal*, versi 2020 memberikan kejelasan yang lebih tentang hal ini. Masing-masing dari tiga artefak sekarang berisi 'komitmen'. Untuk *Product Backlog* adalah *Product Goal*, *Sprint Backlog* memiliki *Sprint Goal*, dan *Increment* memiliki *Definition of Done* (sekarang tanpa tanda kutip). Mereka ada untuk menghadirkan transparansi dan fokus terhadap kemajuan setiap artefak.

Manajemen Mandiri daripada Mengorganisir Diri Sendiri

Panduan Scrum sebelumnya menyebut *Development Team* sebagai tim yang mengatur dirinya sendiri, memilih siapa dan bagaimana melakukan pekerjaan. Dengan lebih fokus pada *Scrum Team*, versi 2020 menekankan pada *Scrum Team* yang melakukan manajemen mandiri atau mengelola diri sendiri, memilih siapa, bagaimana, dan apa yang akan dikerjakan.

Tiga Topik Perencanaan Sprint

Selain topik *Sprint Planning* "Apa" dan "Bagaimana", Panduan Scrum 2020 menekankan pada topik ketiga, "Mengapa", mengacu pada *Sprint Goal*.

Penyederhanaan Bahasa Secara Keseluruhan untuk Audiens yang Lebih Luas

Panduan Scrum 2020 telah menekankan pada penghapusan pernyataan yang berlebihan dan kompleks serta menghilangkan semua kesimpulan yang terkait pekerjaan TI (misalnya pengujian, sistem, desain, persyaratan, dll). Panduan Scrum sekarang kurang dari 13 halaman.

Catatan Penerjemah

Dokumen terjemahan ini telah melalui proses penerjemahan dan penyuntingan seakurat mungkin sesuai kemampuan penerjemah dan mengikuti aturan penerjemahan resmi. Beberapa kata tidak diterjemahkan untuk menghindari kerancuan dan pergeseran makna. Penerjemah melampirkan beberapa kata kunci dari dokumen sumber, dan terjemahannya pada dokumen ini. Untuk memahami lebih baik, penerjemah menyarankan agar memahami dokumen sumbernya langsung. Apabila ada bagian terjemahan yang dirasa kurang akurat, silakan menyarankan terjemahan yang lebih baik melalui warkopagile@gmail.com

Sumber	Terjemahan
Scrum	Scrum
Value	Nilai
Increment of Value	<i>Increment</i> Nilai
Scrum Framework	Kerangka Kerja Scrum
Self-Manage	Manajemen Mandiri
Scrum Values	Nilai-Nilai Scrum
Scrum Team	<i>Scrum Team</i>
Cross-Functional	Lintas Fungsi
Developers	<i>Developers</i>
Product Owner	<i>Product Owner</i>
Product Backlog Item	<i>Product Backlog Item</i>
Scrum Master	<i>Scrum Master</i>
Scrum Events / Event	Acara Scrum / Acara
The Sprint / Sprint	<i>Sprint</i>
Sprint Planning	<i>Sprint Planning</i>
Daily Scrum	<i>Daily Scrum</i>
Sprint Review	<i>Sprint Review</i>
Sprint Retrospective	<i>Sprint Retrospective</i>
Artifact / Artifacts	Artefak
Product Backlog	<i>Product Backlog</i>
Precise	Akurat
Product Goal	<i>Product Goal</i>
Sprint Backlog	<i>Sprint Backlog</i>
Real-time	Waktu sebenarnya
Sprint Goal	<i>Sprint Goal</i>
Increment	<i>Increment</i>
Definition of Done	<i>Definition of Done</i>

Jakarta, Februari 2020