

# Agile project management

WI2022 Manajemen Proyek (Prodi IF)

Sem 1 2024/2025



# Agile principles

2

- Agile berarti mampu **bergerak** dengan **cepat** dan **mudah**.
- Agile menggunakan metode yang didasarkan pada pengembangan **iterative**/berulang dan **incremental**/bertahap, di mana requirements dan solusi berkembang melalui **kolaborasi**.
- Agile dapat digunakan untuk pengembangan perangkat lunak atau dalam lingkungan apa pun yang **requirements-nya tidak diketahui** atau **berubah dengan cepat**.
- Terkait Triple Constrains pada pendekatan agile, **Time** dan **Cost** bersifat **tetap**, namun **Scope fleksibel**, sehingga project sponsor atau produk owner dapat memprioritaskan dan memprioritaskan ulang pekerjaan yang ingin mereka selesaikan



# Agile manifesto

3

## 1. individuals and interactions over processes and tools.

peningkatan kinerja terjadi ketika mendapatkan individu dan interaksi yang benar dengan mengumpulkan development team dengan memberdayakan product owner.

## 2. working software over comprehensive documentation.

kegagalan testing dan memperbaiki defect selama sprint dapat memakan waktu, usaha dan biaya yang berlipat-lipat di sprint berikutnya. Apalagi jika sedang dideploy ke pasar dan development teamnya tidak lagi terlibat.

## 3. customer collaboration over contract negotiation.

product owner yang berdedikasi dan mudah diakses meningkatkan produktivitas berlipat karena development team mendapat klarifikasi langsung guna menyelaraskan prioritas customer dengan pekerjaan yang dilakukan.

## 4. responding to change over following a plan.

menerapkan pendekatan just-in-time, planning cukup untuk mendukung visi produk strategis dan roadmap. Adaptasi rencana dengan kenyataan di sepanjang development untuk menghindari fungsionalitas yang sia-sia dan memberikan produk yang menyenangkan customer.

# Agile principle

4

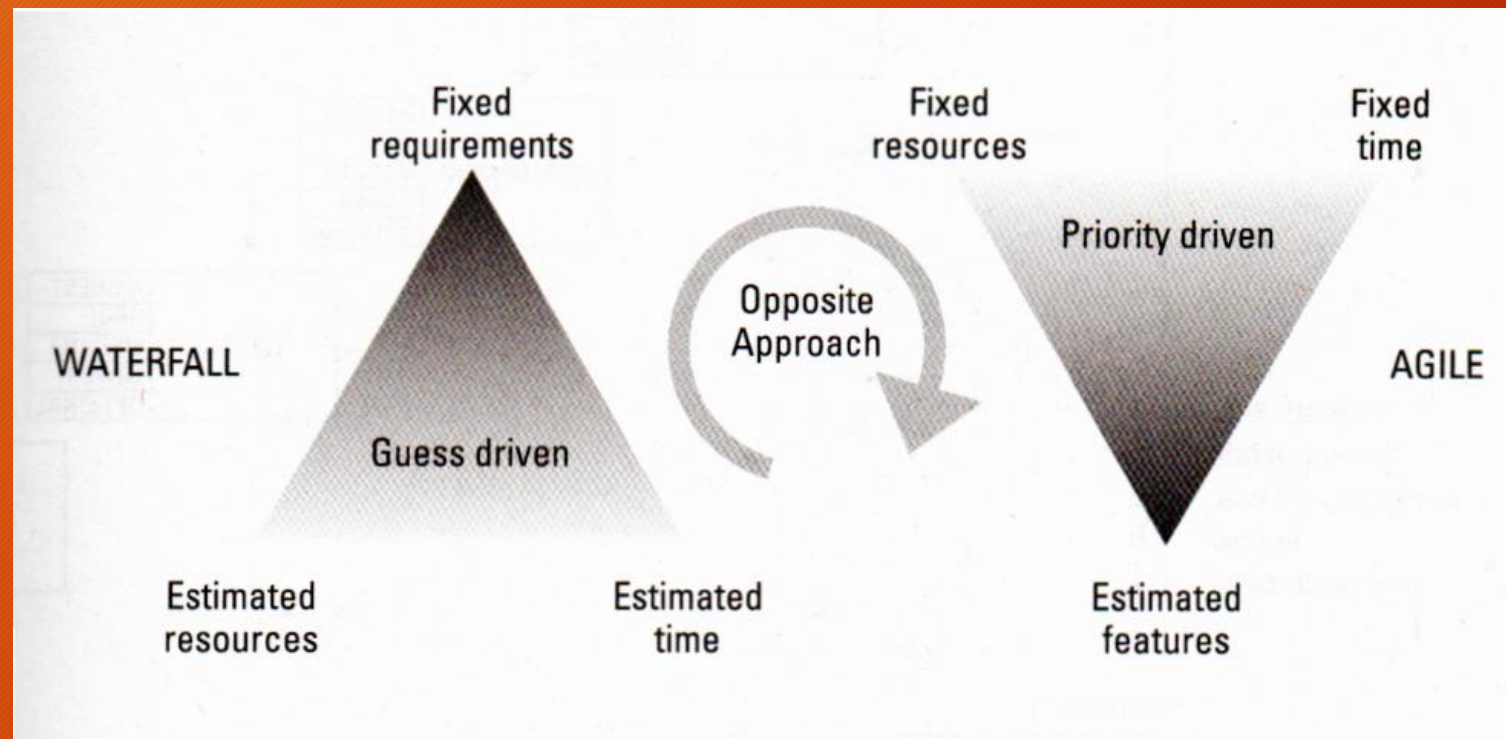
- Pedoman praktis Agile principle dapat digunakan untuk melihat apakah tim mengimplementasikan sesuai maksud agile movement.
- Empat kelompok implementasi Agile principle adalah:
  - **Customer satisfaction - C**  
Pendekatan Agile fokus pada kepuasan pelanggan
  - **Quality - Q**  
Memberikan kualitas produk yang meningkat setiap hari dari hasil terbaik tiap anggota tim.
  - **Teamwork - T**  
Teamwork (termasuk para customer dan stakeholder) sangat penting untuk agile product development.
  - **Product development - P**  
Pendekatan Agile berfokus pada planning dan executing pekerjaan untuk menghasilkan produk terbaik yang dapat dirilis.



# Waterfall vs Agile approaches

5

Variable triple constraint (scope/requirement, cost/resource, dan time) berkebalikan

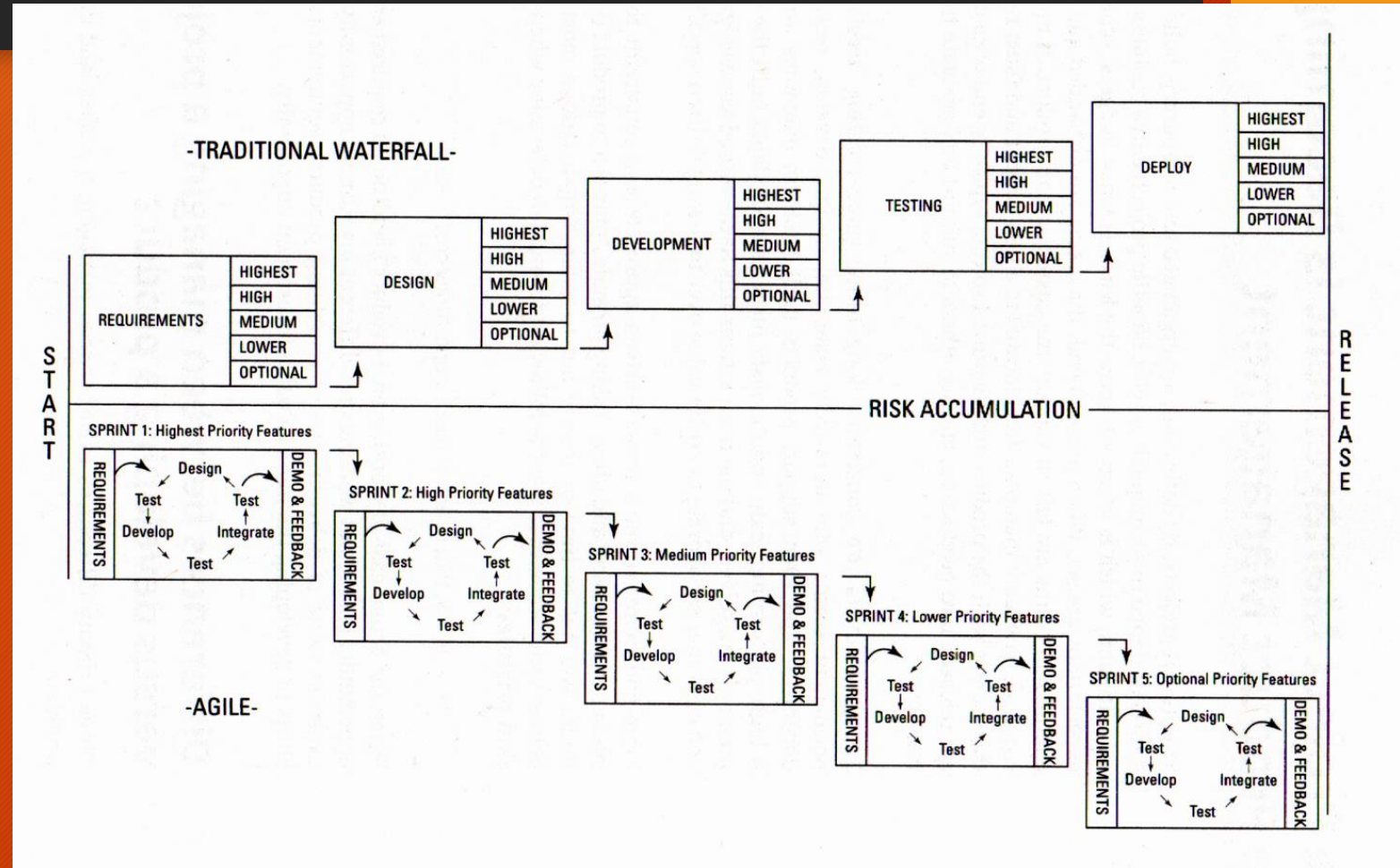


# Waterfall vs Agile approaches

6

**Waterfall** - setiap fase bergantung pada fase sebelumnya

**Agile** - development, testing, dan integrase sekelompok requirement dilakukan dalam siklus iterative yang singkat.





# Keunggulan Agile Approaches

7

- Fleksibilitas dan Stabilitas yang lebih besar.
- Mengurangi task (aktivitas) yang tidak produktif.
  - meeting, email, presentasi, dokumentasi proses
- Kualitas yang lebih tinggi dan Delivery lebih cepat.
- Peningkatan kinerja tim.
  - dukungan untuk tim, fokus, continuous improvement
- Kontrol yang lebih ketat.
- Menemukan kegagalan lebih cepat dan lebih murah recoverinya.

# Mengapa Agile Approaches lebih disukai?

8

- **Executive**
  - Efisien
  - Meningkatkan peluang Return of Investment
- **Product development dan Customers**
  - Meningkatkan adaptasi perubahan
  - Mendeliver value yang lebih besar
- **Management**
  - Kualitas yang lebih tinggi,
  - Mengurangi proses dan produk yang sia-sia,
  - Mengutamakan value
- **Development Teams**
  - memberdayakan development teams untuk menghasilkan karya terbaik secara wajar.



# (Big Three) Agile Approaches

9

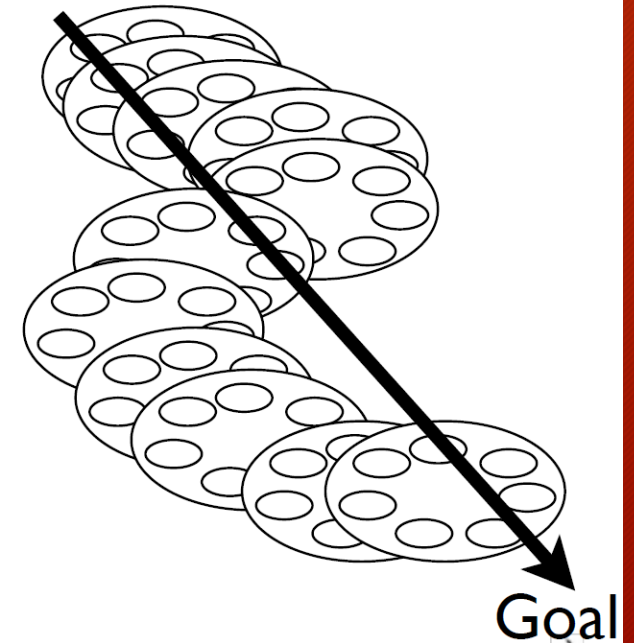
- **Lean**
  - memotong 'bagian yang tebal' saat lean muncul di manufaktur.
  - memahami lean dan pengembangan produk.
  - memahami kanban.
- **Scrum**
  - perjalanan panjang dengan sprint.
  - memahami peran scrum, artefak, dan seremoni (event).
- **Extreme Programming**
  - menemukan prinsip-prinsip Extreme Programming.
  - mengenal beberapa praktik Extreme Programming.

# Scrum

10



Start



- Cross-functional teams yang berkinerja tinggi dengan formasi scrum yang digunakan oleh tim rugby
- Kelompok kecil bertanggung jawab untuk mengambil bola dan menggerakkannya ke arah gawang



# Scrum Frameworks

11

## ROLES

- Product Owner
- Scrum Master
- Team

## CEREMONIES

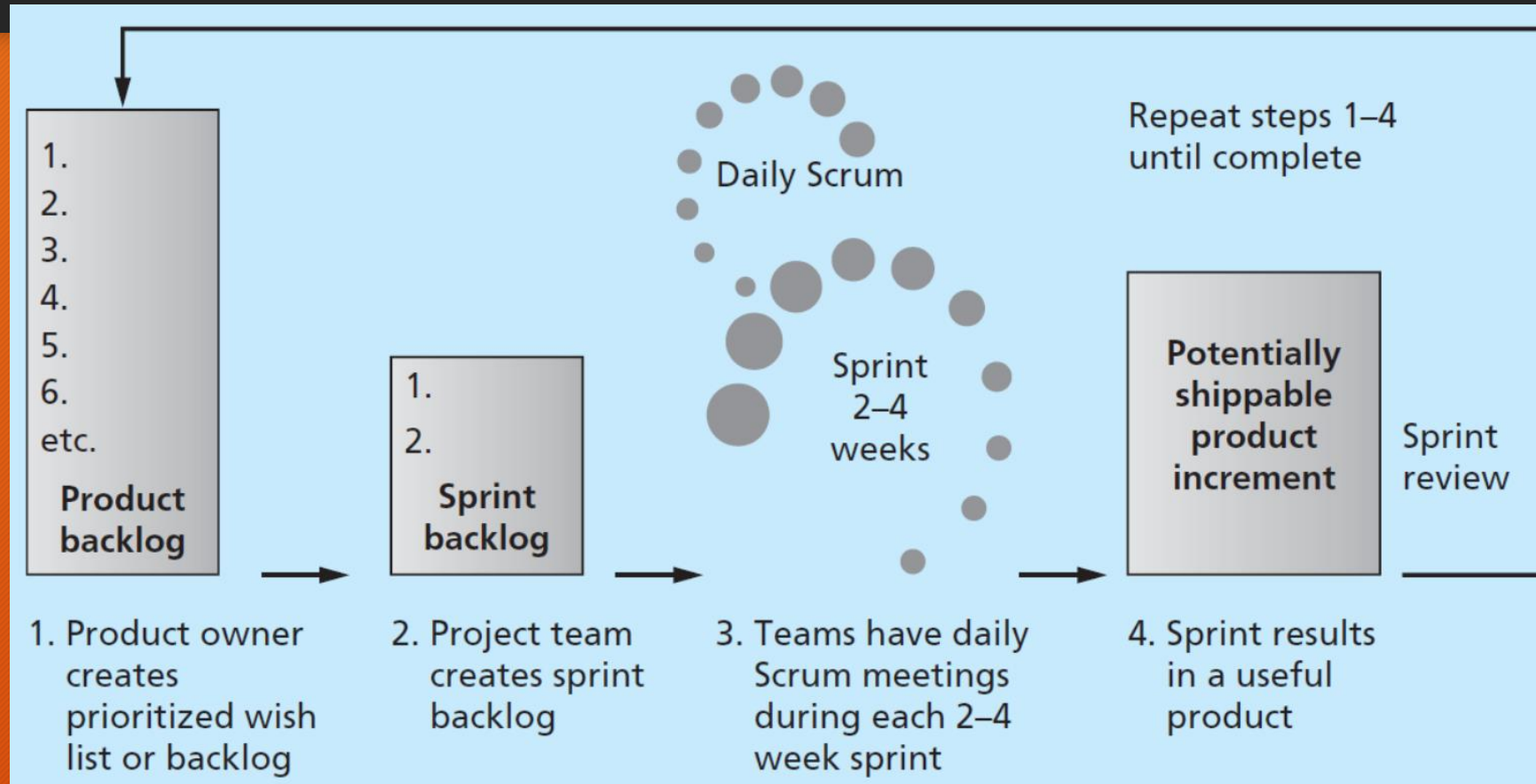
- Sprint Planning
- Sprint Review
- Sprint Restrospective
- Daily Scrum Meeting

## ARTIFACTS

- Product Backlog
- Sprint Backlog
- Burndown Charts

# Scrum Frameworks

12





# Scrum roles

13

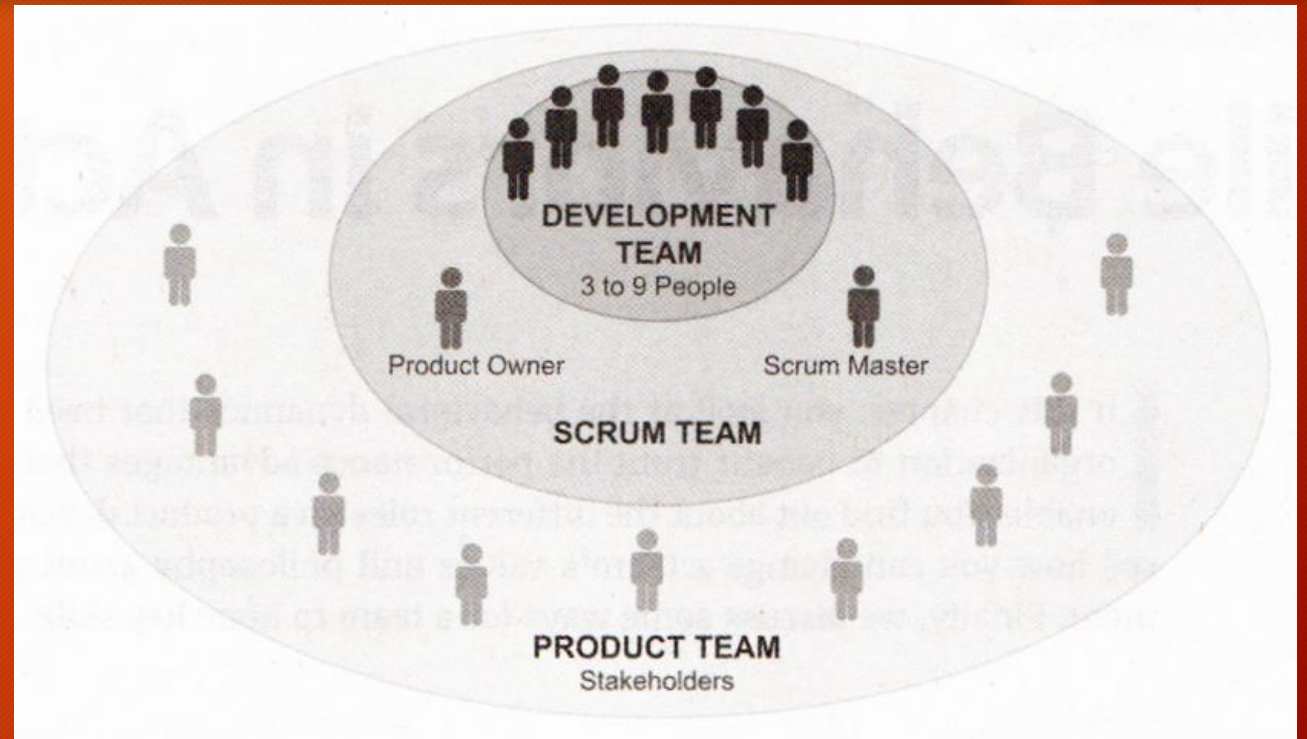
- **Product owner:** Orang yang bertanggung jawab atas *business value* proyek dan memutuskan pekerjaan apa yang harus dilakukan dan dalam urutan apa, seperti yang didokumentasikan dalam **product backlog**
- **ScrumMaster:** Orang yang memastikan bahwa tim produktif, memfasilitasi daily Scrum, memungkinkan kerja sama yang erat di semua peran dan fungsi, dan menghilangkan hambatan efektivitas tim.
- **Scrum team or development team:** Sebuah **cross-functional team** yang terdiri dari 5 - 9 orang yang mengatur diri mereka sendiri dan bekerja untuk menghasilkan hasil yang diinginkan pada setiap sprint. Sprint biasanya berlangsung selama 2 - 4 minggu, di mana pekerjaan tertentu harus diselesaikan dan siap untuk direview.

# Scrum roles

14

Peran penting lain untuk mendukung pengembangan:

- **Stakeholder:** dapat berasal dari departemen yang berbeda-beda yang berperan memberi masukan dan mempengaruhi hasil produk.
- **Agile mentor:** person yang memiliki pengalaman menerapkan prinsip, praktik, dan Teknik Agile untuk memberikan feedback dan saran bagi tim.





Atribut untuk menjadi tim scrum yang sukses:

- **Dedicated team** - setiap anggota tim scrum hanya bekerja pada tujuan produk yang diputuskan oleh tim scrum, bukan pada produk/tim lain.
- **Cross-functionality** - kemauan dan kemampuan untuk mengerjakan berbagai jenis tugas penciptaan produk.
- **Self-organization** - kemampuan dan tanggung jawab untuk melakukan pekerjaan pengembangan produk.
- **Self-management** - kemampuan dan tanggung jawab untuk tetap bekerja pada jalurnya.
- **Size-limited teams** - tidak lebih dari sembilan orang, untuk memastikan komunikasi yang efektif
- **Ownership** - inisiatif untuk bekerja dan bertanggung jawab atas hasil.

## 1. Product Backlog:

- Daftar requirement (fitur) yang diprioritaskan berdasarkan business value.
- Item dengan prioritas tertinggi harus di-breakdown cukup rinci agar tim dapat memperkirakan upaya yang diperlukan untuk mengembangkannya. Beberapa ahli menyarankan penjadwalan sekitar 10 hari kerja untuk setiap item. Ukuran dan kompleksitas pekerjaan menentukan perkiraannya.
- Konten (minimal):
  - Deskripsi requirement
  - Urutan prioritas requirement
  - Estimasi usaha pengerjaannya
- Product Owner bertanggung jawab untuk membuat dan memelihara/mengupdate product backlog
- Tim scrum memanfaatkan product backlog sesuai prioritas untuk merencanakan rilis setiap sprint.



## 2. Sprint Backlog:

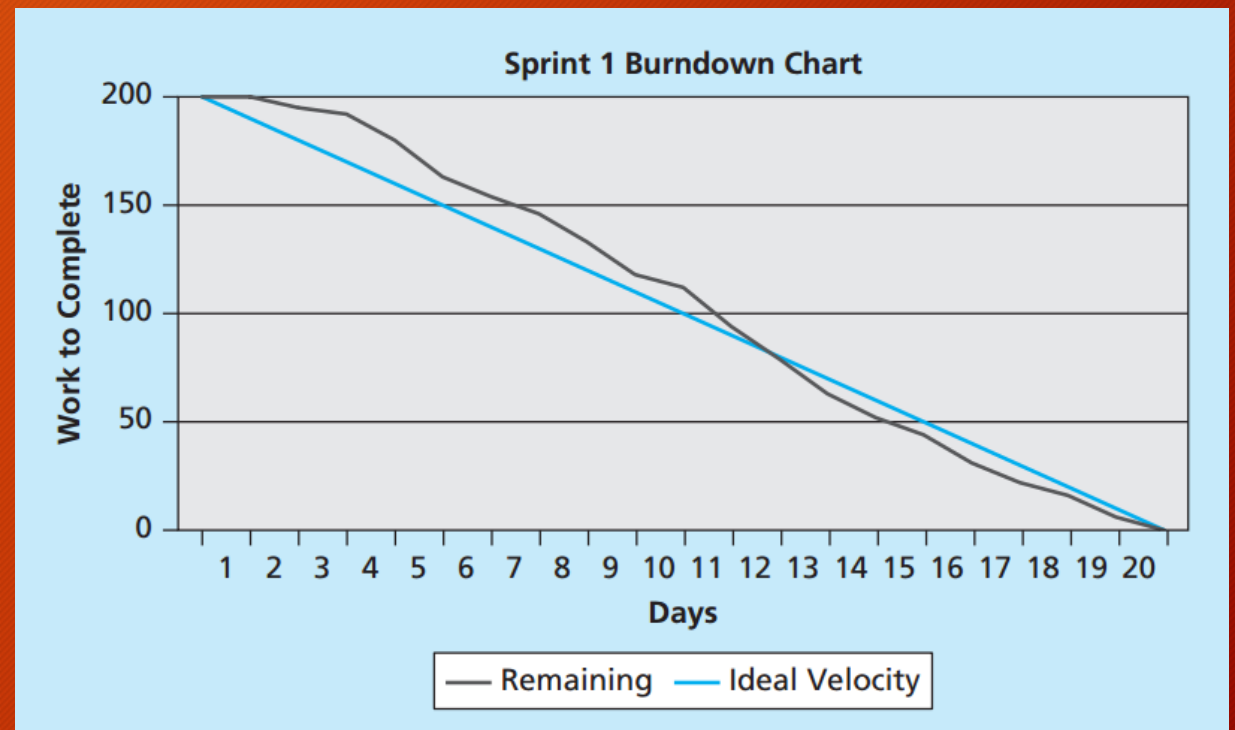
- Item dengan prioritas tertinggi dari product backlog yang harus diselesaikan oleh tim scrum untuk mencapai tujuan sprint tertentu.
- Saat sprint planning, Product owner dan Tim Scrum me-breakdown item dengan prioritas tertinggi menjadi task-task kecil yang membutuhkan waktu sekitar 12 - 16 jam untuk diselesaikan.
- Task dalam sprint backlog dapat diubah oleh tim development jika dinilai perlu untuk memastikan pencapaian tujuan sprint.

# Scrum Artefacts

18

## 3. Burndown Chart:

- Menampilkan status pekerjaan yang sedang berlangsung untuk sprint dan sisa pekerjaan kumulatif dalam sprint setiap harinya.
- Tim pengembangan telah menyelesaikan task yang mendekati tingkat pengerjaan (burn) yang merata dari alokasi jam yang tersedia.





Artefak penting lainnya:

## a. Product increment

- Produk yang dapat digunakan, dan berpotensi untuk dideliver ke user.
- Di setiap sprint, peningkatan produk mencakup fungsionalitas dari requirement yang telah elaborasi, didesain, dikembangkan, diuji, diintegrasikan, didokumentasikan, dan disetujui untuk memenuhi kebutuhan bisnis yang dimaksudkan.
- Produk yang diserahkan ke customer harus cukup lengkap menunjukkan fungsionalitas kerjanya dalam memenuhi tujuan bisnis customer.
- Mungkin perlu lebih dari satu sprint untuk menghasilkan fungsionalitas yang cukup berharga untuk diserahkan ke customer.

Artefak penting lainnya:

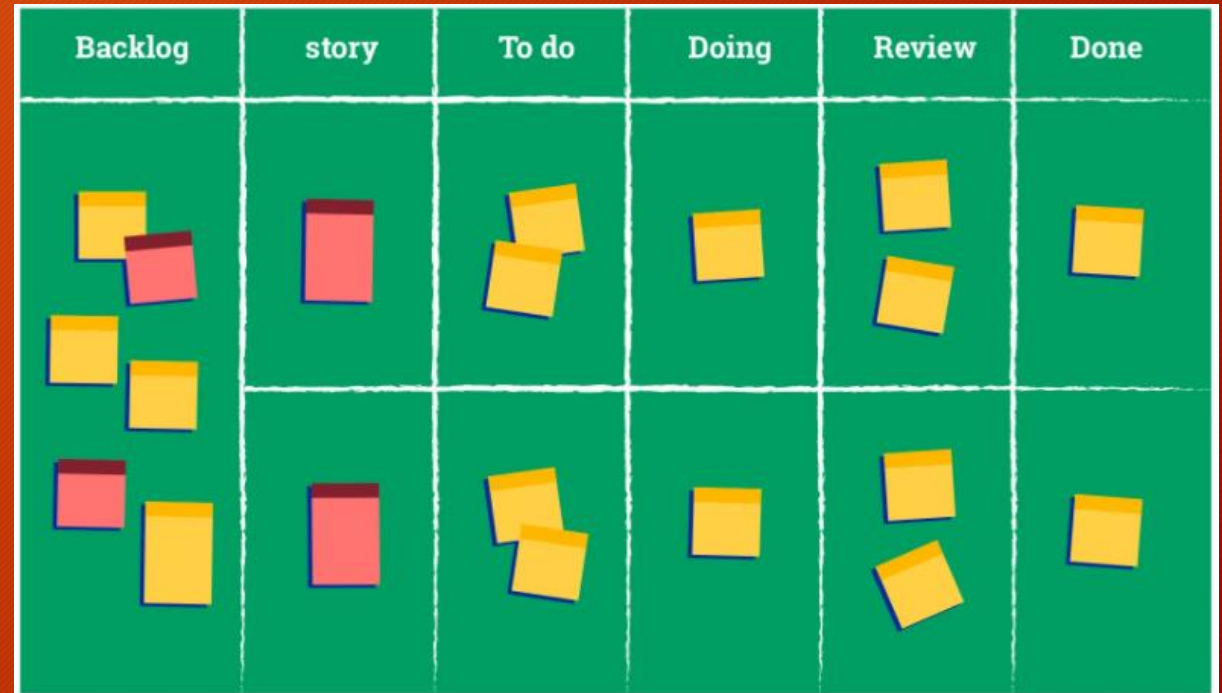
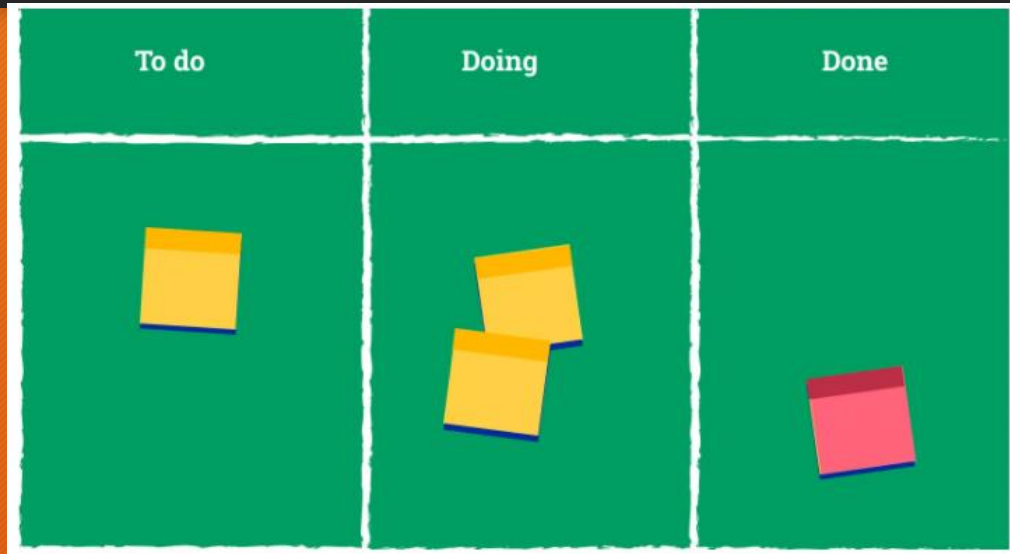
## b. Kanban Board

- Mencatat status unit-unit kerja (task dari WBS) yang dikerjakan di setiap sprint.
- Kanban dibagi menjadi beberapa kolom yang masing-masing menunjukkan status pengerjaan unit-unit kerja. Misalnya setidaknya terdapat kolom status: *To-Do*, *Doing (In Progress)*, *Done*.
- Setiap unit kerja diwakili dengan sebuah sticky note yang ditempel (diletakkan) pada kolom sesuai statusnya.
- Setiap sticky note dapat dibedakan warnanya berdasar kategori, misal warna dibedakan berdasar kategori jenis pekerjaan, atau dibedakan berdasar siapa yang mengerjakan unit kerja, dsb.



# Kanban Board (Sprint Board)

21



# Scrum Ceremonies

22

1. **Sprint Planning:** pertemuan dengan tim untuk memilih serangkaian pekerjaan dari product backlog yang akan diselesaikan selama sprint. Pertemuan ini memakan waktu sekitar empat jam hingga sehari penuh.
2. **Daily Scrum:** pertemuan singkat bagi tim pengembangan untuk berbagi kemajuan dan tantangan serta merencanakan pekerjaan untuk hari itu. Idealnya anggota tim berada di tempat yang sama, rapat biasanya berlangsung tidak lebih dari 15 menit, dan diadakan pada waktu dan tempat yang sama setiap harinya. Dapat juga dengan videoconferencing.
3. **Sprint Reviews:** pertemuan di mana tim mendemonstrasikan kepada product owner apa yang telah diselesaikannya selama sprint.
4. **Sprint Retrospectives:** pertemuan di mana tim mencari cara untuk meningkatkan produk dan proses berdasarkan review terhadap kinerja aktual tim pengembangan.



- **Tambahan penjelasan Daily Scrum:**
  - ScrumMaster menanyakan pekerjaan apa yang telah dilakukan sejak kemarin, pekerjaan apa yang direncanakan hari ini, dan hambatan atau hambatan apa yang mungkin menghambat upaya tim.
  - ScrumMaster mendokumentasikan hambatan-hambatan ini dan bekerja sama dengan stakeholder utama untuk menyelesaikannya setelah Daily Scrum.
  - Banyak tim menggunakan istilah permasalahan untuk hal-hal yang tidak harus diselesaikan dalam waktu 24 jam ke depan dan istilah penghambat untuk hal-hal yang perlu segera diatasi. Hal ini memungkinkan ScrumMaster untuk mempertahankan fokus pada item dengan prioritas tertinggi (blocker) terlebih dahulu baru kemudian mengelola penyelesaian masalah lainnya pada hari-hari berikutnya.

# Aktivitas Scrum berdasarkan Process Group

24

## Initiating

- Tentukan peran/roles
- Putuskan berapa banyak sprint yang akan dibuat untuk setiap rilis dan cakupan perangkat lunak yang akan di-deliver. (mirip dengan membagi proyek menjadi beberapa proyek yang lebih kecil)

## Planning

- Buat product backlog
- Buat sprint backlog
- Buat release backlog
- Rencanakan pekerjaan setiap hari di daily Scrum
- Dokumentasikan hambatan dalam bentuk daftar

## Executing

- Selesaikan task setiap hari selama sprint
- Hasilkan produk yang dapat di-deliver di akhir setiap sprint

## Monitoring & Controlling

- Selesaikan masalah dan hambatan
- Buat dan perbarui burndown chart
- Peragakan produk yang telah selesai pada sprint review meeting

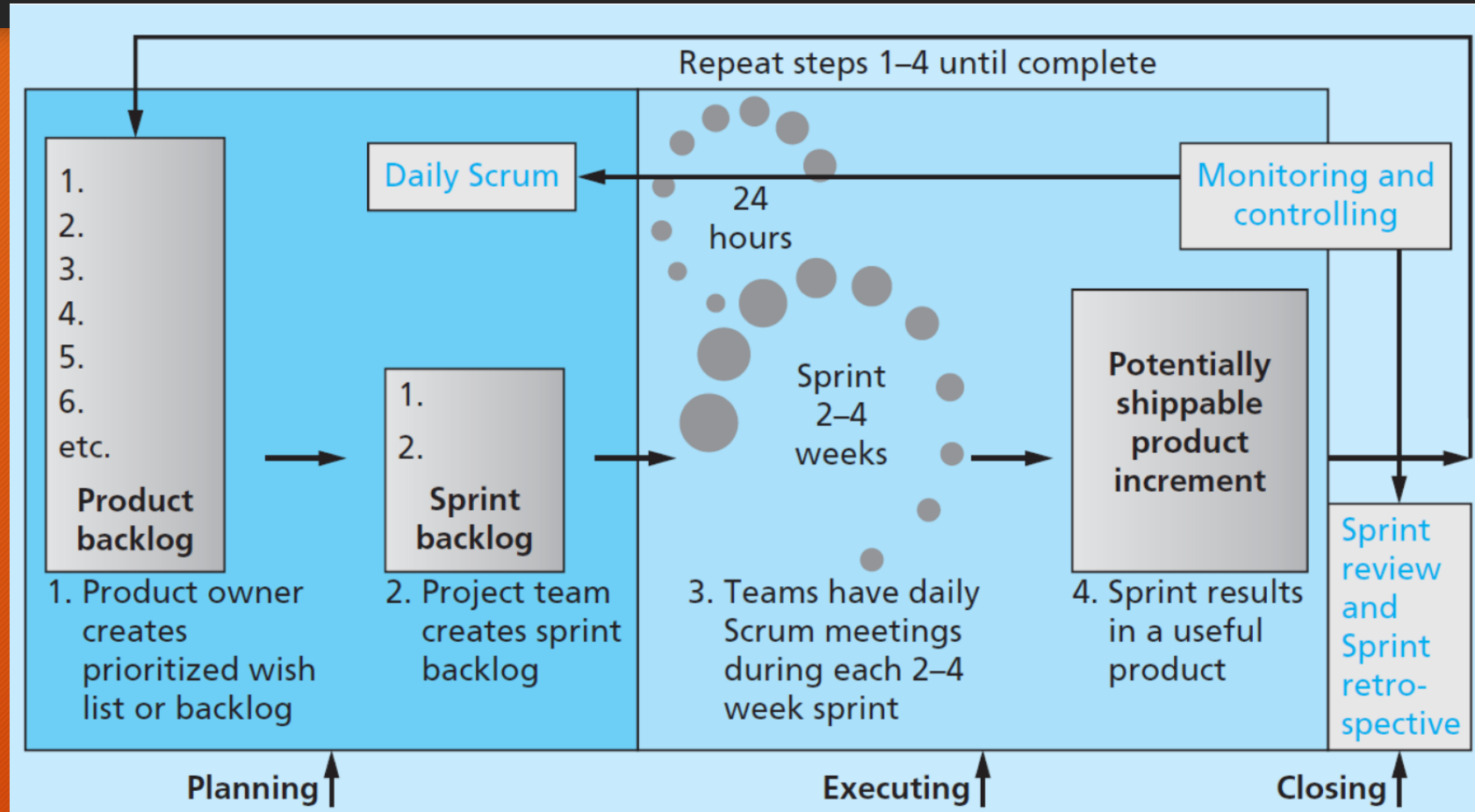
## Closing

- Renungkan cara meningkatkan produk dan proses selama sprint retrospective meeting.



# Process Group dalam Scrum

25



# A. Project Pre-Initiation and Initiation

26

- *Project pre-initiation:*
  - tentukan peran dan putuskan fungsi apa yang akan dideliver sebagai bagian dari setiap rilis,
  - berapa banyak sprint yang dibutuhkan untuk menyelesaikan rilis, dan
  - berapa banyak rilis software dideliver.
- Mirip dengan membagi proyek menjadi beberapa proyek yang lebih kecil
  - Project charter, daftar stakeholder, manajemen strategi stakeholder, dan kick-off meeting masih akan dibuat sebagai bagian dari inisiasi



## B. Planning

27

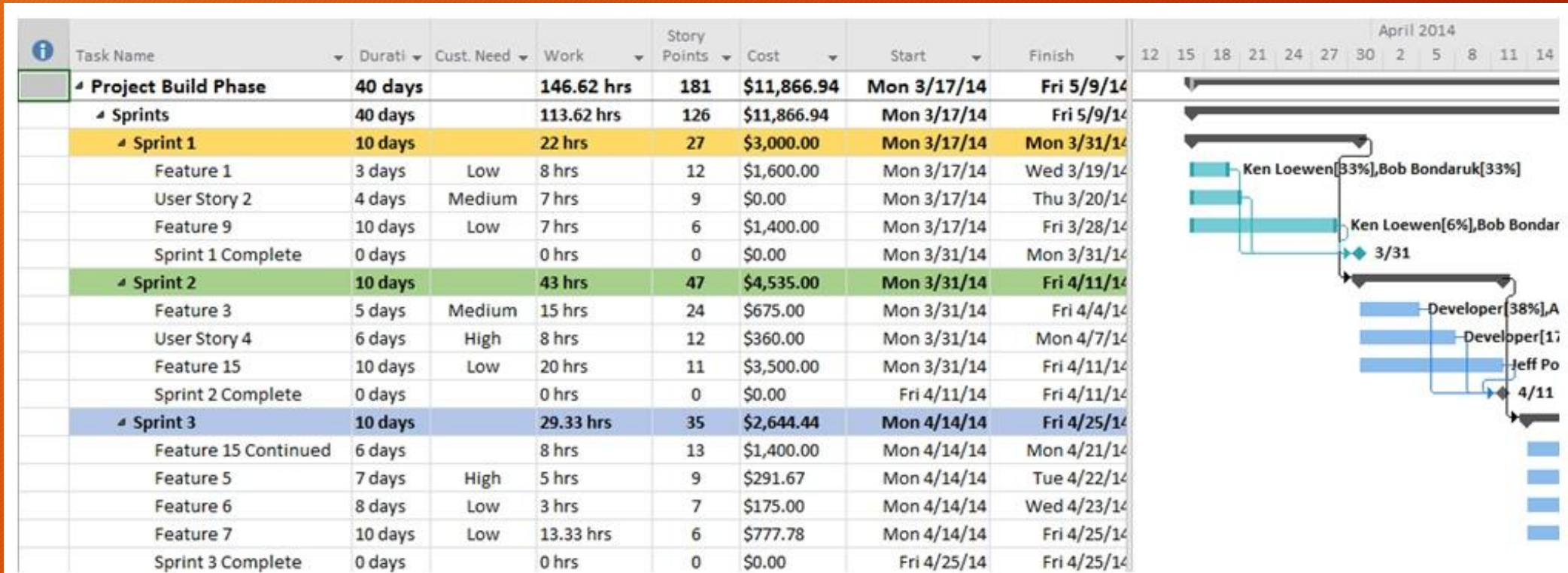
- Preliminary Scope statement
- Team contracts tidak mandatory
- Deskripsi pekerjaan global (tingkat tinggi) harus diidentifikasi dalam **product backlogs** dan **sprint backlogs**
  - Sebagai pengganti WBS, pekerjaan akan diidentifikasi dalam *product* dan *sprint backlogs*.
  - Tim membuat daftar *technical stories* dan task (rinci) terkait untuk diselesaikan setiap sprint.
  - Tim menetapkan perkiraan kecepatan/*velocity* (atau kapasitas) setiap *sprint* berdasarkan ketersediaan anggota tim setiap hari dalam suatu *sprint*
  - Perkiraan dapat diberikan sebagai:
    - jam kerja, dengan asumsi bahwa jam kerja adalah 8 jam per hari.
    - poin, dengan asumsi bahwa karyawan bekerja < 8 jam per hari. Misalnya, 6 jam per hari = 1 poin, jadi 36 jam kerja = 6 poin.

# Gantt chart

28

Process groups tidak mengikuti pola linier sederhana.

- Beberapa grup proses diulang untuk setiap sprint, menghasilkan beberapa rilis produk software yang dapat digunakan, seperti yang ditunjukkan oleh milestone.





## B. Planning - Steps (1)

29

- **Product backlog** dibuat untuk **memprioritaskan fungsi terpenting** dari hasil proyek dalam hal menambah nilai bagi bisnis
  - Product backlog dibuat setelah menganalisis hasil survei dan mendiskusikan opsinya
  - Pendekatan ini menggabungkan beberapa item dalam pernyataan lingkup, untuk fokus pada apa yang akan menambah nilai paling besar.
  - Memiliki person/anggota yang tepat akan berguna sebelum tim mengembangkan fitur penting.
  - Setelah mereview product backlog, Scrum team akan melakukan perencanaan dan memperbarui sprint backlog berdasarkan item-item yang dapat diselesaikan oleh tim dalam sprint pertama.
  - Kemudian tim akan mengadakan daily Scrum meeting pertama untuk merencanakan hari itu.

## B. Planning - Steps (2)

30

- Detilkan item dalam **sprint backlog** menjadi item pekerjaan yang lebih spesifik,
  - Dapat berupa **user stories** yaitu deskripsi singkat yang ditulis oleh pelanggan tentang apa yang mereka butuhkan dari sistem
  - Deskripsi ini panjangnya sekitar tiga kalimat.
  - Menjadi dasar untuk perkiraan waktu pada *sprint planning meeting*.
  - **User stories** harus dapat diuji dan cukup kecil sehingga pemrogram dapat menyelesaikan dan menguji kode mereka tepat waktu.

Misalnya, seseorang di area keuangan perusahaan dapat menulis user story: "***Kemampuan menagih pelanggan untuk beberapa produk dan layanan intranet.***" sebagai berikut:  
"Sebagai manajer keuangan, saya ingin situs kami menggunakan Perusahaan B untuk memproses pembayaran sehingga kami menghemat biaya transaksi dan layanan pelanggan."



## B. Planning - Steps (3)

31

- **Break down user stories ke technical stories.**
  - **User requirements** diterjemahkan dengan menggunakan **technical stories** ke dalam spesifikasi teknis yang diperlukan untuk membuat fungsionalitas pengguna.
  - **Technical stories** dapat berisi satu atau lebih **technical tasks** yang digunakan pengembang untuk memetakan kemajuan pada sprint board saat pekerjaan dilakukan selama sprint.
  - Sebuah **task** harus setara dengan beban **satu hari**.
  - Satu **item** dalam **sprint backlog** dapat didokumentasikan dalam satu atau lebih **user story**, satu **technical story**, dan **task**, berdasarkan kompleksitasnya.
  - Mungkin ada hubungan satu-ke-satu atau hubungan banyak-ke-satu jika item tersebut sangat kompleks.

## B. Planning - Steps (4)

32

- **Technical story** didetilkkan menjadi **task**; mirip dengan task dalam work breakdown structure.
- Beberapa organisasi menggunakan **release road map** untuk menyusun **semua pekerjaan** untuk seluruh rilis, yang dapat mencakup lebih dari satu sprint.
  - Road map ini biasanya direpresentasikan sebagai **bagan** yang terdiri dari **beberapa kolom**.
  - **Release road map** memberikan gambaran yang jelas tentang apa (ruang lingkup) yang akan terkandung dalam setiap sprint.
  - Tool ini memungkinkan Scrum team untuk dengan mudah mereview dan update perkiraan kecepatan dibandingkan dengan yang sebenarnya.
  - Release road map harus berasal dari data dalam Project Management Information System (PMIS) atau agile management system terkait, seperti Microsoft's Team Foundation Server, Rally, VersionOne, atau JIRA



## C. Execution

33

- Waktu dan uang paling banyak harus dikeluarkan untuk mengeksekusi rencana penciptaan produk.
  - Tim akan menyelesaikan tugas setiap hari
  - Dengan Agile, tim akan menghasilkan beberapa iterasi produk yang berpotensi dapat di-deliver.
- Misalnya, pada akhir sprint pertama, proyek perangkat lunak akan memiliki beberapa fungsionalitas yang tersedia
  - Users akan dapat mengakses template, sampel, dan orang yang ditunjuk untuk user stories, WBS, dan project schedules, sebagaimana tercantum dalam sprint backlog pertama.
  - Users juga dapat membuat saran untuk fungsionalitas yang ingin mereka lihat ketika sedang dikembangkan.
  - Iterasi pertama akan memberikan kemampuan untuk beberapa produk dan layanan bagi pelanggan
  - Dengan Scrum methods, bisnis dapat mengambil manfaat dari fitur-fitur baru ini beberapa bulan lebih awal daripada menggunakan pendekatan prediktif yang dijelaskan dalam kasus pertama.

## D. Monitoring and Controlling

34

- Dua tool utama untuk monitoring dan controlling dalam Scrum framework adalah
  - daily Scrum dan
  - sprint review
- Daily Scrums diadakan setiap pagi untuk merencanakan dan mengomunikasikan pekerjaan hari itu dan membahas segala risiko, masalah, atau penghambat
  - Daily Scrum mencakup diskusi singkat tentang masalah dan penghalang yang dihadapi oleh tim
  - ScrumMaster akan bekerja dengan stakeholders yang tepat untuk mengatasi masalah ini sebagai bagian dari monitoring dan controlling
  - ScrumMaster memiliki tugas utama untuk menghapus rintangan sehingga tim dapat berkinerja baik
  - ScrumMaster mendokumentasikan masalah dan hambatan, mirip list of prioritized risk



## D. Monitoring and Controlling (cont.)

35

- Kemajuan pekerjaan dalam sprint dapat **ditampilkan di kanban/sprint board** yang dikelola oleh ScrumMaster.
  - Kanban/Sprint board berisi *card* setiap tugas yang harus dikerjakan selama sprint
  - Setiap task berisi nomor kendali, task name, perkiraan waktu penyelesaian, peringkat atau nomor prioritas, dan anggota tim yang ditugaskan.
  - Saat tasks dibuka, dikerjakan, dan ditutup, cards dipindahkan secara fisik ke bagian yang sesuai di papan
  - Bagian-bagian ini termasuk Not Started, In Progress, Ready to Test, Tested, dan Closed
  - Developers memperbarui status tasks dalam bagian Not Started, In Progress, dan Ready to Test sections
  - Testers memperbarui status untuk tasks pada bagian Tested
  - Product owner bertanggung jawab untuk meninjau fungsionalitas, mengonfirmasi bahwa telah berfungsi seperti yang diharapkan, dan mengubah status task menjadi Closed

## D. Monitoring and Controlling (cont.)

36

- **Burndown chart** adalah artefak penting yang digunakan untuk menampilkan kemajuan secara grafis pada setiap sprint.
  - Burndown chart menunjukkan **kemajuan selama sprint** pertama untuk proyek tersebut, yang dijadwalkan berlangsung selama 4 minggu dan menghasilkan 5 item atau user storie yang tercantum dalam sprint backlog
  - selama perencanaan, setiap user story dipecah menjadi task-task spesifik, dan tim memperkirakan berapa lama waktu yang dibutuhkan untuk menyelesaikan setiap task untuk setiap user story
  - Selama daily Scrum meeting, pekerjaan yang diklaim anggota tim harus selesai dalam sehari
  - Tiap hari, tim harus memperkirakan kembali jumlah jam atau poin yang tersisa untuk setiap task
  - Beberapa task mungkin ditambahkan dan mungkin dihapus saat scope menjadi lebih jelas
  - Tidak peduli berapa jam telah dihabiskan; yang penting adalah berapa jam kerja tersisa untuk menyelesaikan user story setiap sprint



## D. Monitoring and Controlling (cont.)

37

- Burndown chart memplot jumlah estimasi jam atau poin yang tersisa setiap hari.
  - Ini juga menunjukkan garis burndown yang ideal, seolah-olah tim tersebut menyelesaikan jumlah pekerjaan yang sama setiap hari, atau 10 task setiap hari dalam contoh ini.
  - Chart tersebut dengan jelas menunjukkan apakah tim tersebut bekerja dengan baik dalam sprint tersebut atau apakah ada potensi masalah.
  - Burndown charts membantu tim memahami apakah user story mungkin tidak diselesaikan dalam sprint itu.
  - Chart tersebut memberikan indikator bahwa story berisiko dan harus dihapus.
  - Jika kemajuan lebih baik dari yang diantisipasi, burndown chart dapat menunjukkan bahwa story harus ditambahkan ke sprint.
- Di akhir setiap sprint, ScrumMaster memimpin **review meeting untuk mendemonstrasikan hasil sprint** pada tsb
  - Tim mendemonstrasikan kepada product owner apa yang telah diselesaikannya selama sprint.
  - Setelah mereview demonstrasi sprint, product owner akan mengupdate product backlog berdasarkan informasi terbaru dan kebutuhan bisnis, dan siklus sprint berikutnya akan dimulai.

## E. Closing

38

- Setelah sprint review, ScrumMaster memimpin **sprint retrospective**
  - Selama pertemuan singkat ini (sekitar setengah jam), tim merefleksikan apa yang terjadi selama sprint.
  - ScrumMaster biasanya meminta umpan balik anggota tim melalui email terlebih dahulu dan menyusun hasil ini sebelum pertemuan.
  - Fokus diskusi hanya pada hal-hal yang paling penting.
  - Retrospective mirip dengan lessons-learned report, tetapi berfokus pada periode waktu yang lebih singkat.
  - Sprint retrospective dimaksudkan untuk menjawab dua pertanyaan mendasar:
    - Apa yang **berjalan baik** selama sprint terakhir yang harus kita **terus lakukan**?
    - Apa yang bisa kita **lakukan secara berbeda** untuk meningkatkan produk atau proses?



# Resume

39

Agile Project Management	Traditional Project Management
Fokus pada interaksi dan <b>customer satisfaction</b>	Fokus pada <b>plans</b> dan <b>artifacts</b>
Respons terhadap perubahan melalui <b>tindakan adaptif</b>	Perubahan dikendalikan melalui <b>tindakan korektif</b>
Progressive elaboration, <b>rolling-wave planning</b>	<b>Up-front planning</b> yang monumental
Mengutamakan pelanggan, <b>time-boxed delivery</b>	Negosiasi Manager, <b>scope-based delivery</b>
Manajemen komitmen melalui <b>feature breakdown structure</b>	Manajemen aktivitas melalui <b>work breakdown structure</b>
Kolaborasi dalam tim yang self-disciplined dan <b>self-organizing</b>	Top-down <b>control</b>
Minimal set dari context-sensitive, praktik <b>generative</b>	<b>Prescriptive</b> , heavyweight methods
Essential, <b>value-focused metrics</b>	<b>Non-value added controls</b>

# Pemetaan PROJECT PLAN

40

Tugas 5 Scrum Planning - Sprint-1 Planning - Sprint-1 Executing

SCRUM PLANNING	
1. Product Vision	Goal Statement (output - MOV) - <i>scope statement</i>
2. Product Roadmap → Rencana iterasi/sprint	<ul style="list-style-type: none"><li>• <b>Lingkup proyek</b> dari sudut pandang <b>dekomposisi domain</b> (modul-modul produk) - <i>scope statement</i></li><li>• <b>Preliminary requirement</b> - <i>business case</i></li></ul>
3. Product Backlog	Tasks/aktifitas-aktifitas pada bagian <b>Executing</b> di WBS
SPRINT PLANNING	
1. Sprint Goal	(dari no 2 <b>product roadmap</b> di <b>scrum planning</b> )
2. Sprint Backlog	(pilih <b>tasks</b> di no 3 <b>product backlog</b> di <b>scrum planning</b> yang sesuai <b>sprint goal</b> )
3. Estimasi Task/Item Pekerjaan	( <b>dekomposisi</b> setiap <b>task</b> di <b>sprint backlog</b> , hingga <b>tasks teknis</b> )
KANBAN BOARD (Eksekusi Sprint)	
1. Kanban Board	(ambil dari <b>dekomposisi</b> no 3 <b>Task/Item Pekerjaan</b> di <b>sprint planning</b> )
2. Status Ketercapaian → Buat Burndown Chart	
3. Status di akhir Sprint	