

Čo je SCRUM?

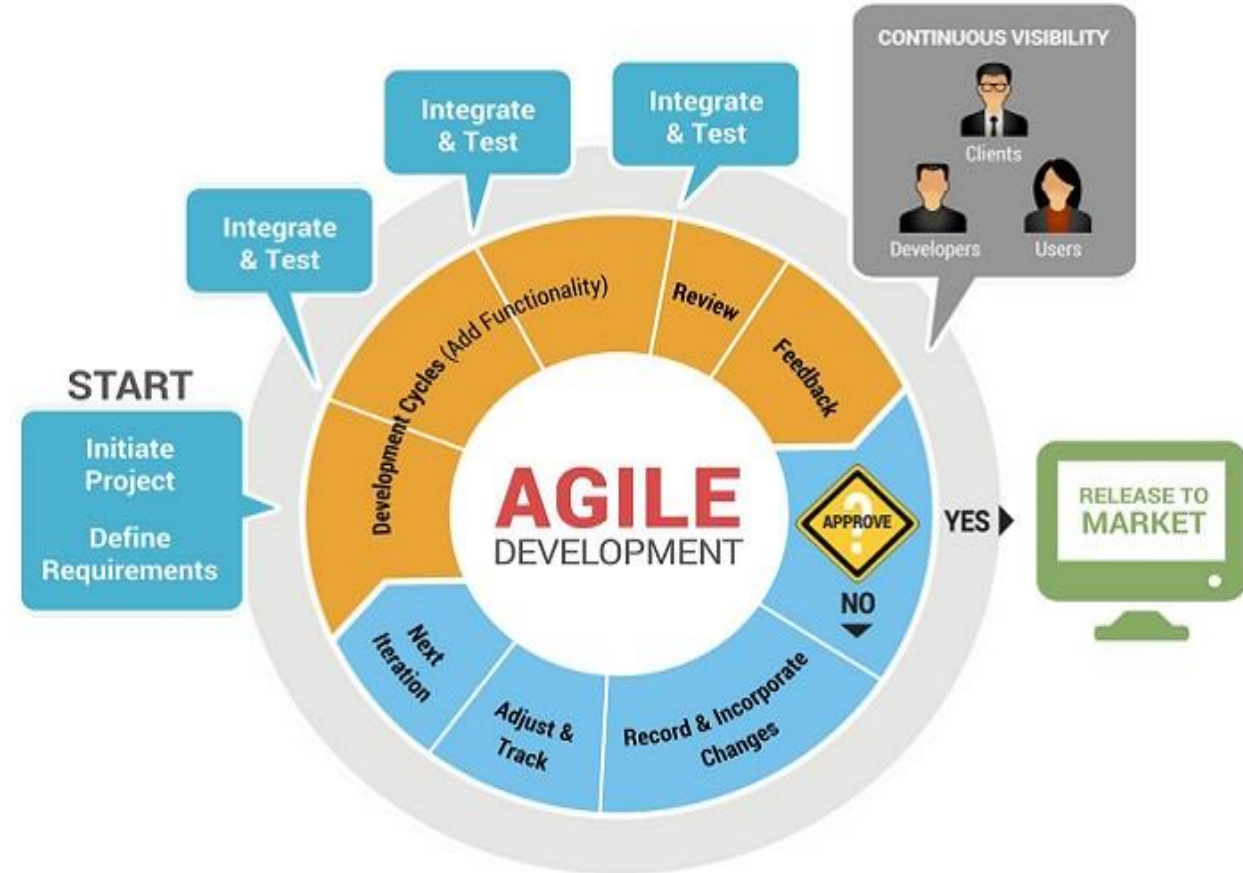
Agilný rámec pre vývoj a dodávku produktov



Vývoj softvéru, výskum, marketing

Agilné metodiky a agilný prístup

- Skupiny metód pôvodne určených pre vyvíjanie softvéru
- Založené na iteratívnom a inkrementálnom vývoji
- Umožňujú rýchly vývoj softvéru a zároveň dokážu reagovať na zmenu požiadaviek počas vývojového cyklu
- Správnosť systému overí jedine pomocou rýchleho vývoja, predloženia zákazníkovi a následných úprav podľa spätnej väzby (feedback)
- Nie je obmedzený len na programovanie, ale aj v Business Intelligence a v marketingovom plánovaní



Protiklad vodopádového modelu



Manifesto agilného vývoja softvéru

Ľudia a komunikácia sú viac než procesy a nástroje.
Funkčný softvér je viac než vyčerpávajúca dokumentácia.
Spolupráca so zákazníkom je viac než dojednávanie zmluvy.
Radšej reagovať na zmenu než sa držať plánu.

Aj keď časť napravo je dôležitá,
viac si ceníme ľavú časť.

Kent Beck
Mike Beedle
Arie van Bennekum
Alistair Cockburn
Ward Cunningham
Martin Fowler

James Grenning
Jim Highsmith
Andrew Hunt
Ron Jeffries
Jon Kern
Brian Marick

Robert C. Martin
Steve Mellor
Ken Schwaber
Jeff Sutherland
Dave Thomas

Agilemanifesto.org

Princípy Agilného Manifesta

1. Našou najvyššou **prioritou** je **uspokojiť zákazníka skorým a sústavným dodávaním hodnotného softvéru**
2. **Zmeny požiadaviek** sú **vítané** dokonca **aj** v **neskorých fázach vývoja**. Agilné procesy dokážu pretaviť **zmenu na konkurečnú výhodu zákazníka**
3. **Dodávame funkčný softvér často**, od niekoľkých týždňov po niekoľko mesiacov, **s uprednostnením čo najkratších intervalov**
4. **Ľudia z biznisu a vývojári** musia denne **spolupracovať počas celého projektu**
5. Postavte **projekty na motivovaných ľuďoch**. **Poskytnite** im **prostredie**, **podporu** a **dôverujte** im, že **svoju úlohu splnia**
6. **Najlepším spôsobom odovzdávania informácií** vývojovému tímu a v tíme je **osobný rozhovor**
7. **Základným ukazovateľom napredovania je funkčný softvér**
8. Agilné procesy podporujú **trvalo udržateľný rozvoj**. **Sponzori, vývojári a používatelia** by mali byť schopní **trvalo udržať konštantné tempo**
9. Sústavný **dôraz** na **technickú vyspelosť** a **kvalitný návrh podnecujú agilitu**
10. **Jednoduchosť – umenie vykonať naozaj len to potrebné**, je nevyhnutnosť
11. **Samoorganizované tímy** vytvárajú **najlepšie koncepty, požiadavky a návrhy** riešení
12. **Tím v pravidelných intervaloch vyhodnocuje sám seba s cieľom byť efektívnejší a prispôsobuje tomu svoje správanie**

Princípy Agilného Manifesta

Uplatňujeme nasledovné princípy:

Našou najvyššou prioritou je uspokojiť zákazníka
skorým a sústavným dodávaním
hodnotného softvéru.

Zmeny požiadaviek sú vítané dokonca aj v neskorých fázach vývoja.
Agilné procesy dokážu pretaviť zmenu
na konkurečnú výhodu zákazníka.

Dodávame funkčný softvér často,
od niekoľkých týždňov po niekoľko mesiacov,
s uprednostnením čo najkratších intervalov.

Ľudia z biznisu a vývojári musia denne spolupracovať
počas celého projektu.

Postavte projekty na motivovaných ľuďoch.
Poskytnite im prostredie, podporu a
dôverujte im, že svoju úlohu splnia.

Najlepším spôsobom odovzdávania informácií
vývojovému tímu a v tíme je osobný rozhovor.

Základným ukazovateľom napredovania je funkčný softvér.

Agilné procesy podporujú trvalo udržateľný rozvoj.
Sponzori, vývojári a užívatelia by mali byť schopní
trvalo udržať konštantné tempo.

Sústavný dôraz na technickú vyspelosť
a kvalitný návrh podnecujú agilitu.

Jednoduchosť – umenie vykonať naozaj len to potrebné, je nevyhnutnosť.

Samooorganizované tímy vytvárajú
najlepšie koncepty, požiadavky a návrhy riešení.

Tím v pravidelných intervaloch vyhodnocuje sám seba
s cieľom byť efektívnejší a prispôsobuje tomu svoje správanie.

Príklady agilných metodík

Extrémne
programovanie

Scrum

Vývoj riadený
vlastnosťami

Vývoj riadený
testami

Lean
development

Crystal
metodiky

Programs

(-) EXIN Agile Scrum (6)

Exam Language

English (6)

Brazilian Portuguese (4)

Dutch (4)

French (4)

German (4)

Latin American Spanish (4)

Chinese (2)

Czech (2)

Japanese (2)

Polish (2)

Spanish (1)

Product Type

Certification (6)

Module Level

Advanced (4)

Certifications



EXIN Agile Scrum Foundation

EXIN Agile Scrum Foundation offers professionals a unique certification that combines agile principles and scrum practices. The exam tests candidates on their understanding of key concepts and their a....

Training: Not mandatory
Module level: Foundation

[MORE INFORMATION](#) →



EXIN Agile Scrum Master

The EXIN Agile Scrum Master is a unique certification that combines agile principles and scrum practices with practical assignments. It tests the competences required to facilitate, coach and enable a....

Training: Mandatory
Module level: Advanced

[MORE INFORMATION](#) →



EXIN Agile Scrum Product

EXIN Agile Scrum Product Owner tests a candidate's skills and knowledge of agile principles and scrum practices.

ATTENTION

Scrum Alliance is now offering courses virtually due to the COVID-19 (coronavirus) situation.

Please visit [Course Search](#) for more details.

Transforming the World of Work®

Scrum Alliance® is a nonprofit organization that is guiding and inspiring individuals, leaders, and organizations with agile practices, principles, and values to help create workplaces that are joyful, prosperous, and sustainable.

[Certify Me](#)[At Your Service](#)

Certifications by Scrum Team Role

SCRUM MASTER TRACK



Certified ScrumMaster®

Intro course for those wishing to fill the role of Scrum Master or Scrum team member.

Prerequisite: none

[Find a course →](#)



Advanced Certified ScrumMaster

Advanced course for Scrum Masters who have one or more years of work experience in that role.

Prerequisite: CSM

[Find a course →](#)



Certified Scrum Professional®-ScrumMaster

Pinnacle course for experts wishing to

PRODUCT OWNER TRACK



Certified Scrum Product Owner®

Intro course for those who are closest to the "business side" of the project.

Prerequisite: none

[Find a course →](#)



Advanced Certified Scrum Product Owner

Advanced course for Product Owners who already have one year of experience on a Scrum team.

Prerequisite: CSPO

[Find a course →](#)



Certified Scrum Professional®-Product Owner

Pinnacle course for experts wishing to

DEVELOPER TRACK



Certified Scrum Developer®

Intro course for software developers (programmers) who are building software in a Scrum environment.

Prerequisite: none

[Find a course →](#)



Certified Scrum Professional®

Certified Scrum Professional for Developers

Prerequisite: active CSD

[COVID-19 response](#): How we are helping you and your teams during this time.



Welcome to the Home of Scrum!™

[WHAT IS SCRUM?](#)[GET CERTIFIED](#)[FIND TRAINING](#)

Scrum Alliance eBooks and Reports

If you've been curious about agile and Scrum practices, this is a good place to start. Read the resources below for valuable information.



The Elusive Agile Enterprise: A Forbes Insights Report

Agility is everywhere. Once seen as solely a software framework, agile and Scrum are exploding into a wide spectrum of industries from healthcare and marketing to finance and the arts. Jointly produced by Scrum Alliance and Forbes Insights, “The Elusive Agile Enterprise” report surveyed more than 1,000 C-level executives



The Scrum Guide

The Scrum Guide™, The Definitive Guide to Scrum: The Rules of the Game, contains the definition of Scrum, including roles, events, artifacts, and the philosophy that binds them together. Created and sustained by the founders of Scrum – Ken Schwaber and Jeff Sutherland – the guide is based on the principles on which agility

Agilné metodiky v právnom rámci

Fixed-Time fixed-Price

- Zmluva **definuje dielo**, ktoré má vzniknúť a byť dodaný za **pevne stanovenú sumu** a v **pevne stanovenom čase**
- Z **právneho hľadiska** ide o **Zmluvu o dielo**, ktorou sa „zhotoviteľ zaväzuje na svoj náklad a nebezpečenstvo vykonať pre objednávateľa dielo a objednávateľ sa zaväzuje dielo prevziať a zaplatiť cenu“

Time and Material

- Vyžaduje **Rámcovú zmluvu o poskytovaní služieb**
- Táto **zmluva nedefinuje dielo**, ale **iba služby**, ktoré bude **dodávateľ zákazníkovi poskytovať** (dizajn, programovanie, testovanie a pod.)
- Jednotlivé **služby** sú **poskytované na základe čiastkových objednávok**, ktoré typicky nasledujú napríklad po šprinte planning

Core activities

Processes · Requirements · Design ·
Engineering · Construction · Testing ·
Debugging · Deployment · Maintenance

Paradigms and models

Agile · Cleanroom · Incremental · Prototyping
· Spiral · V model · Waterfall

Methodologies and frameworks

ASD · DevOps · DAD · DSDM · FDD · IID ·
Kanban · Lean SD · LeSS · MDD · MSF ·
PSP · RAD · RUP · SAFe · **Scrum** · SEMAT ·
TSP · OpenUP · UP · XP

Supporting disciplines

Configuration management · Documentation ·
Software quality assurance (SQA) · Project
management · User experience

Practices

ATDD · BDD · CCO · CI · CD · DDD · PP ·
SBE · Stand-up · TDD

Tools

Compiler · Debugger · Profiler · GUI designer
· Modeling · IDE · Build automation · Release
automation · Infrastructure as code · Testing

Standards and Bodies of Knowledge

BABOK · CMMI · IEEE standards · ISO 9001
· ISO/IEC standards · PMBOK · SWEBOK ·
ITIL · IREB

1. **Iteratívny a inkrementálny spôsob** riadenia vývoja softvéru
2. Radený medzi **agilné metodiky**
3. **Flexibilná, holistická stratégia** produktového vývoja
4. **Počas projektu** môžu **zákazníci zmeniť názor** na **to, čo chcú** a **potrebujú** (Súhrn požiadaviek)
5. Empirický prístup, podľa ktorého **problém nemožno úplne pochopiť** alebo **definovať**
6. Nové nepredvídané úlohy **nemožno riešiť** len **lepším predvídaním** a **plánovaním**
7. **Sústredenie** na maximálnu schopnosť **tímu** rýchlo dodať časť s **najvyššou prioritou** a **reagovať na nové požiadavky**

AGILE V KOCKE

Agile funguje princípom niečo za niečo:
"Dôvera a malá hierarchia za výsledok."



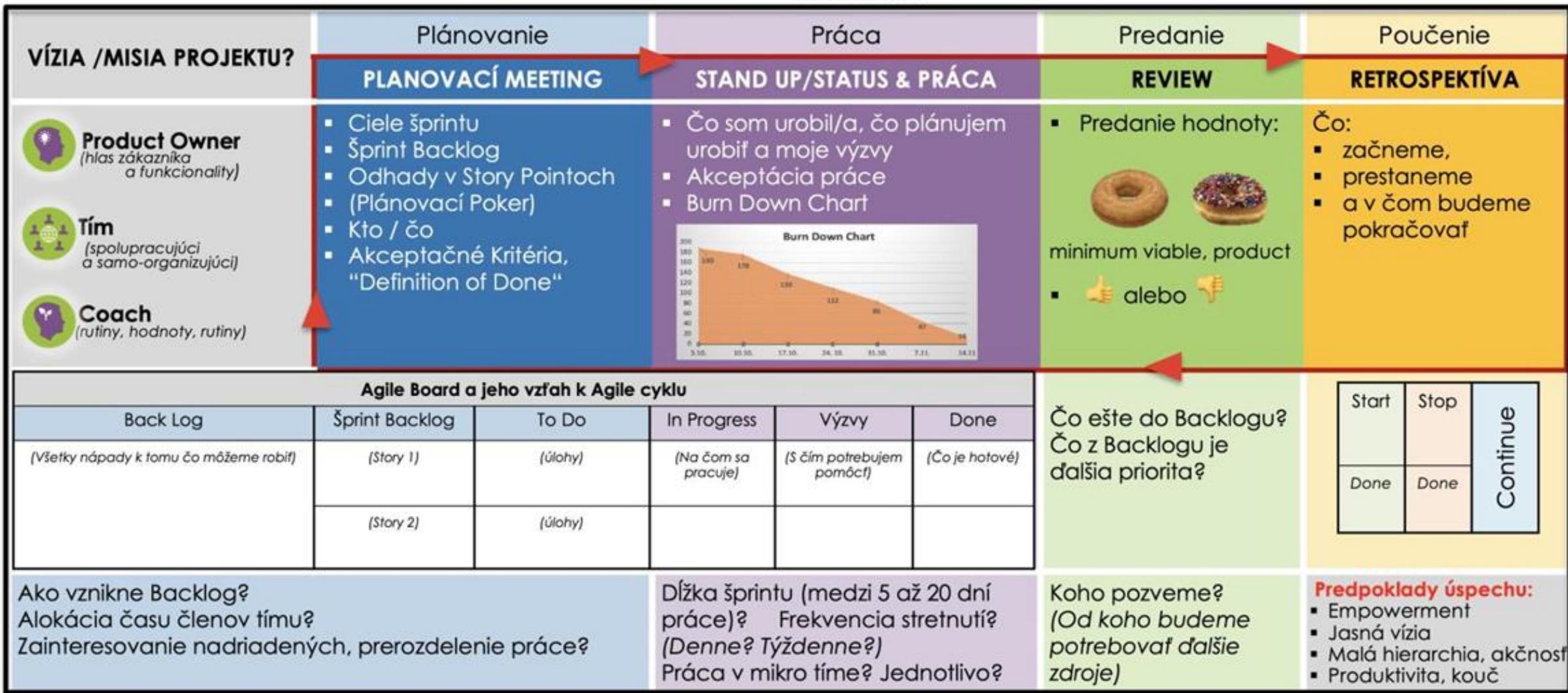
O
AGILE

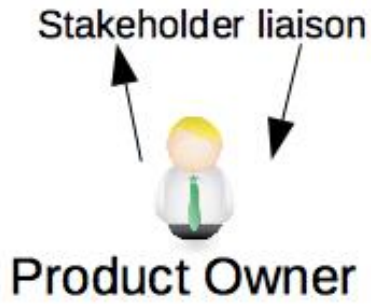
Agile je "**Lean**" verzia projektového manažmentu. Zameriava sa na predávanie hodnoty v **krátkych úsekoch** (šprintoch). Agile očakáva a **pracuje so zmenami**. Agile PM **eliminuje straty** (Nad plánovanie, zbytočný reporting). APM je podobný tomu ako si robíme **projekty sami pre seba medzi 17:00 až 9:00**. Pre využitie tejto metodológie je potrebné mať poskytnuté zdroje (ľudí).



AGILE
MANIFESTO

Jednotlivci a interakcia	pred	procesmi a nástrojmi
Fungujúci softvér	pred	vyčerpávajúcou dokumentáciou
Spolupráca so zákazníkom	pred	vyjednávaním o zmluve
Reagovanie na zmeny	pred	dodržiavaním plánu





Development Team

1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	

PBI's

Product Backlog



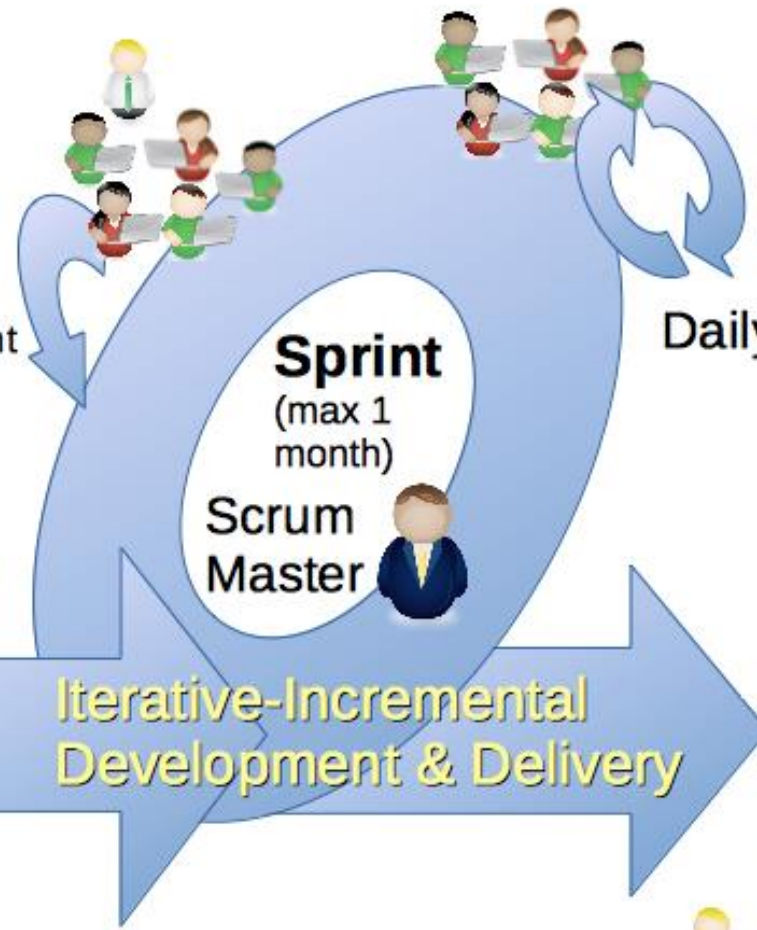
Sprint Planning

Topic 1: forecast PBI's
Topic 2: plan work (e.g. tasks)

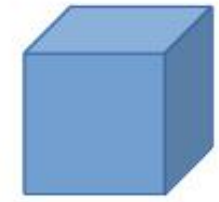


Sprint Backlog

Product Backlog Refinement



Iterative-Incremental Development & Delivery



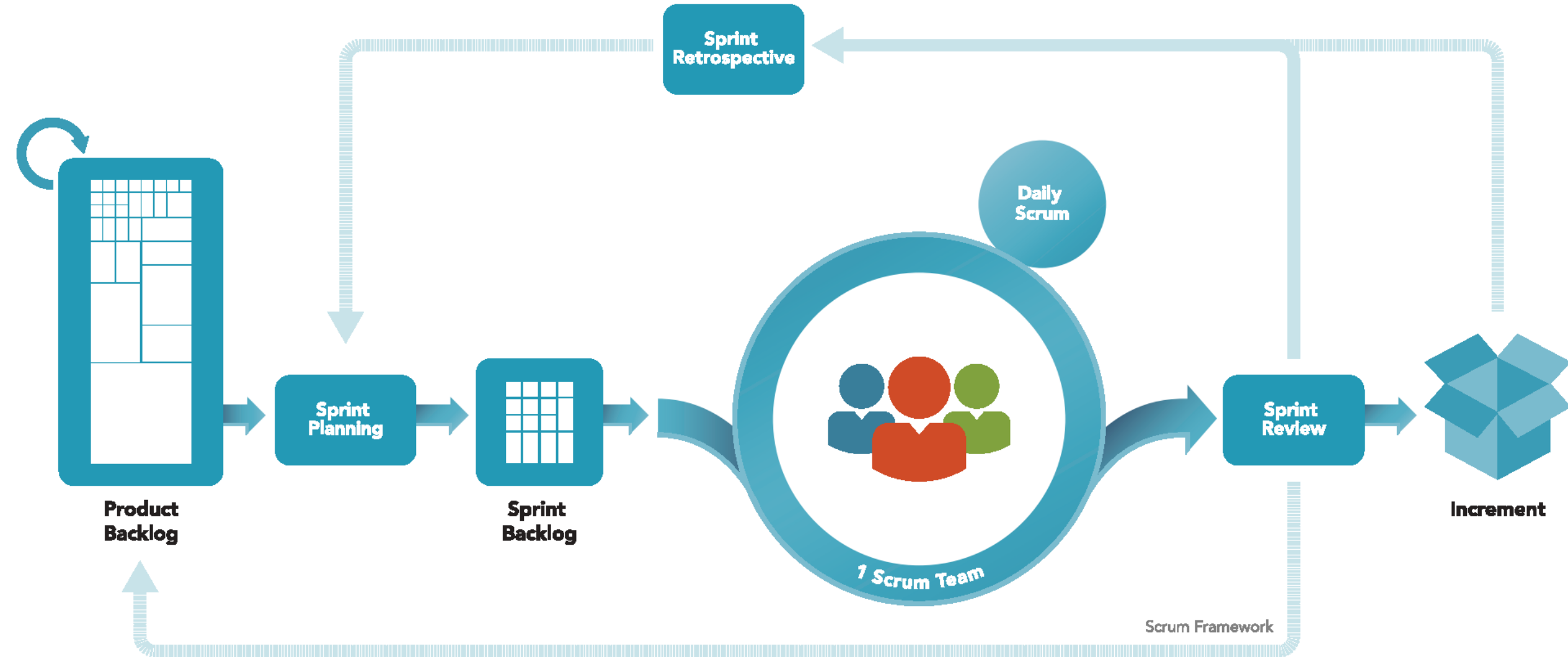
Potentially Releasable Increment

Sprint Review

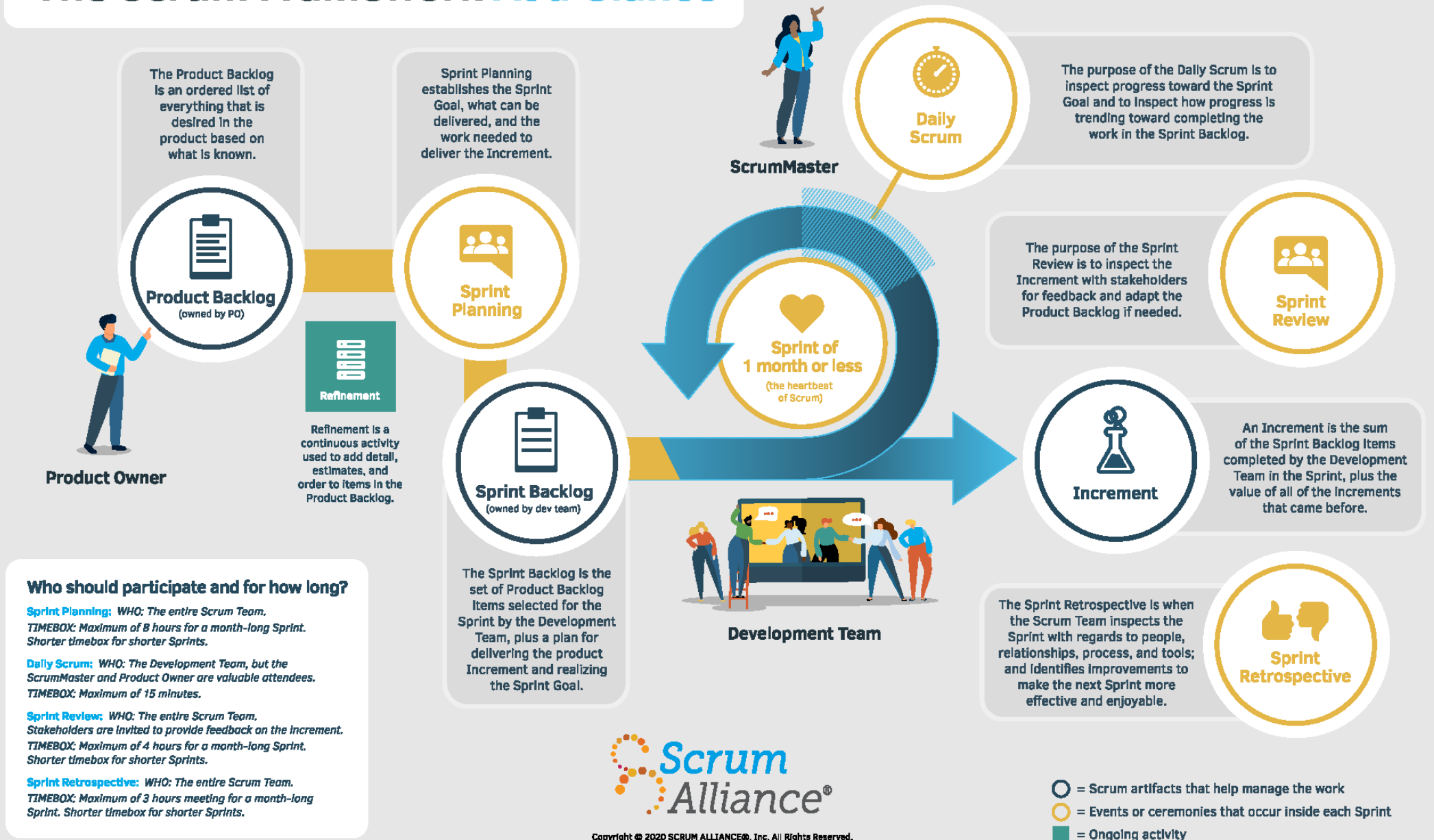
Sprint Retrospective

Scrum Framework

SCRUM FRAMEWORK

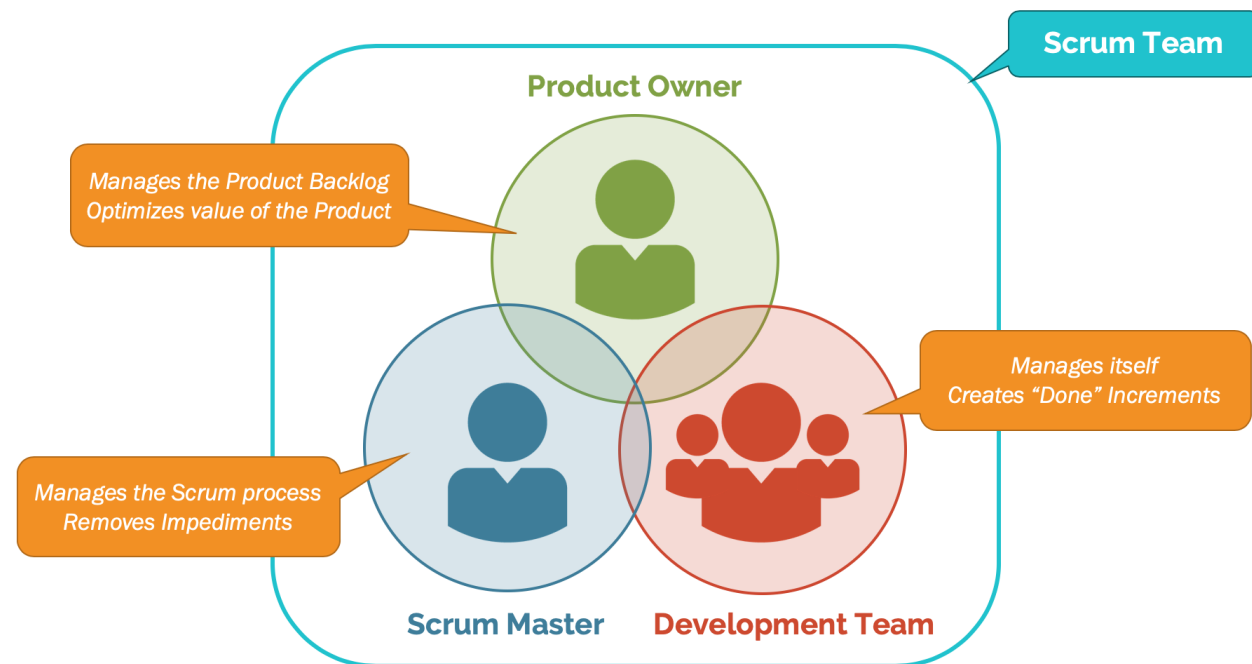


The Scrum Framework At a Glance



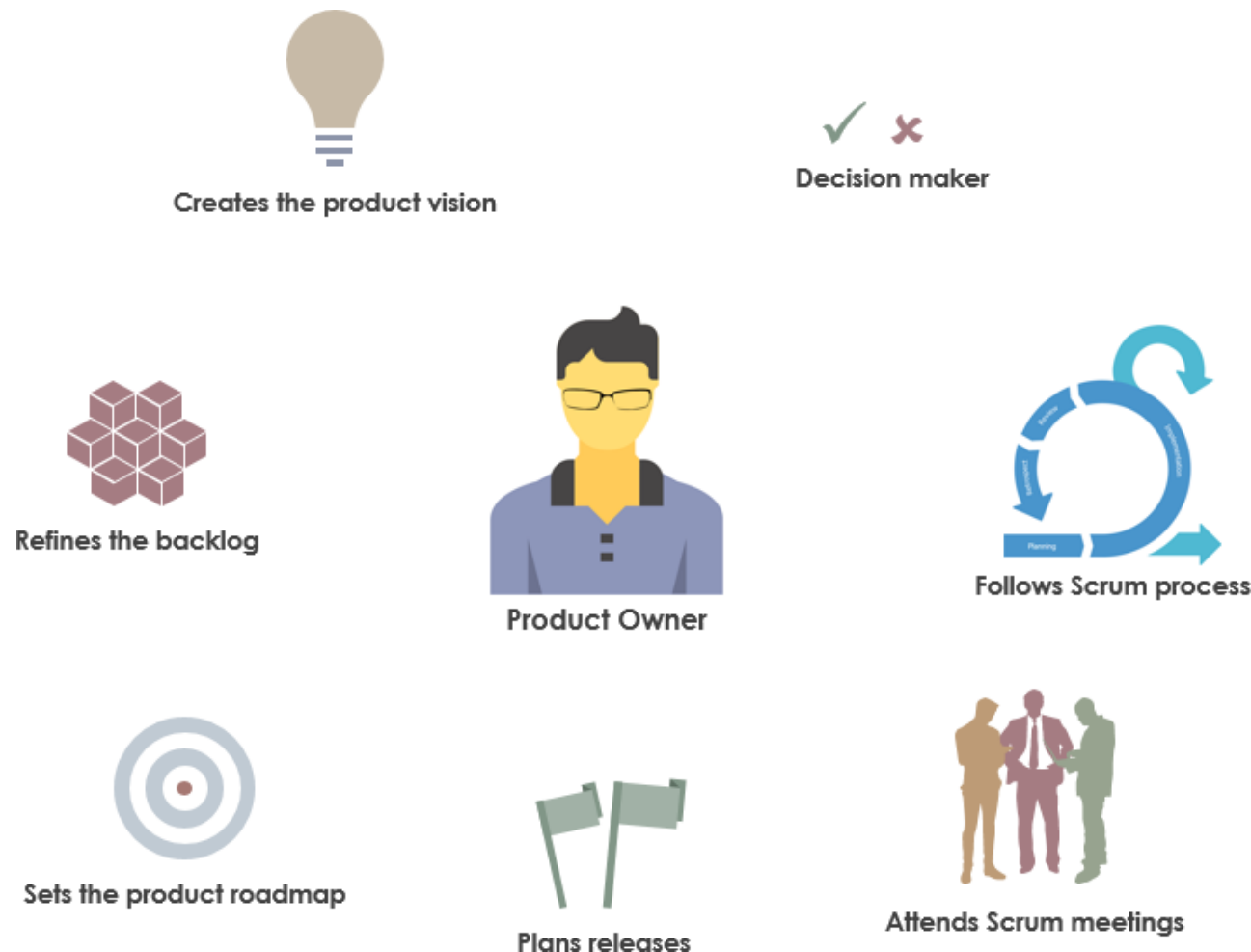
Role v SCRUMe

- Scrum rozlišuje **3 hlavné úlohy** a veľa pomocných rolí
- Hlavné role sa týkajú vzniku produktu (cieľe projektu) a dohromady sa označujú ako **Scrum team**
- V reálnych projektoch sú aj ďalšie role, Scrum však definuje iba tieto:
 1. **Vlastník produktu**
(product owner)
 2. **Vývojový team**
 3. **Scrum master**



Vlastník produktu (product owner)

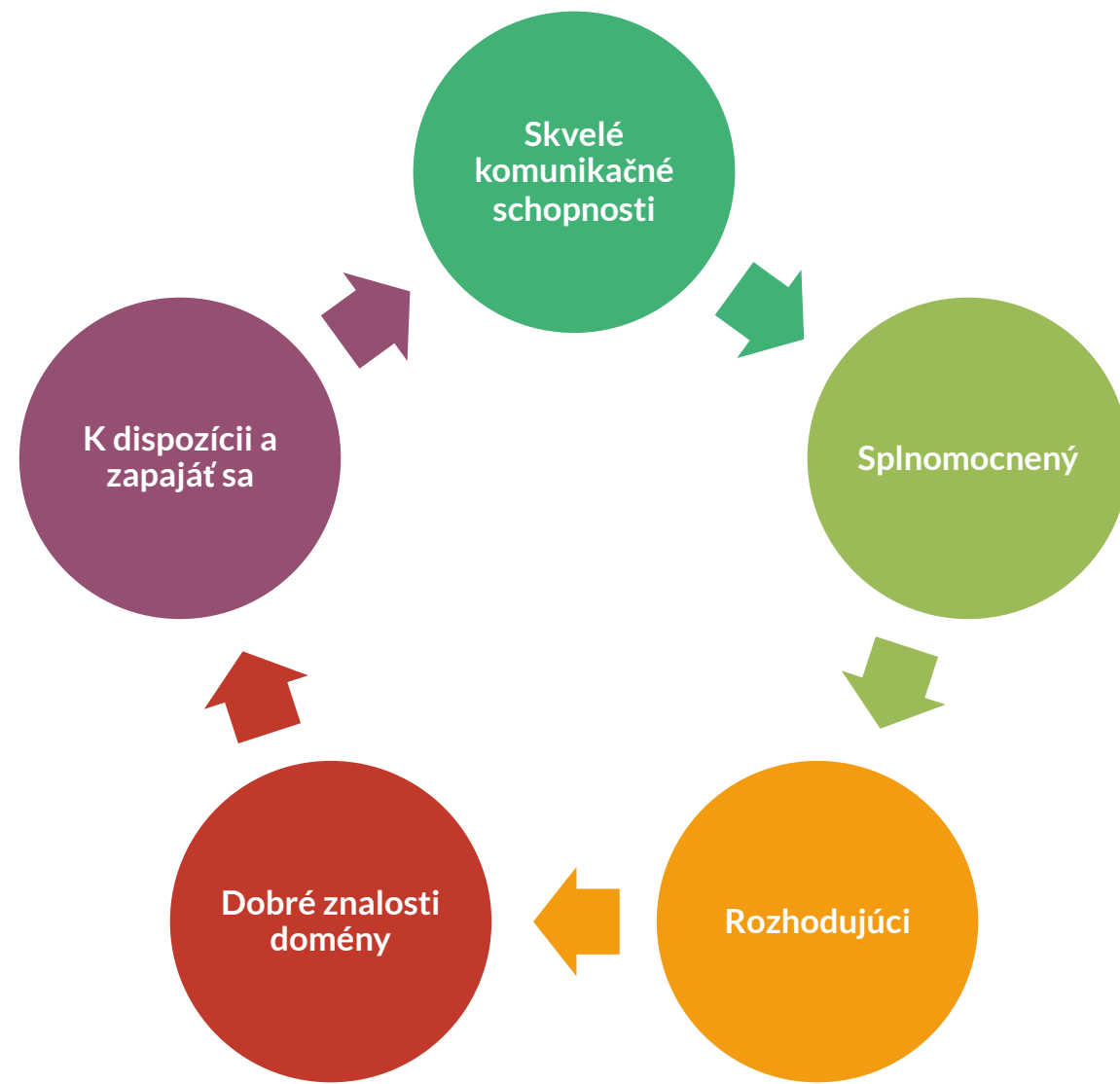
- **Vlastník produktu** reprezentuje **zainteresované** subjekty a je **hlasom** zákazníka
- Je **zodpovedný** za uistenie, že **team** do **biznisu** pridá hodnotu
- Vlastník produktu **píše články pre zákazníkov** (typicky skúsenosti používateľov), **hodnotí** a **priradzuje im priority**, a **pridáva** ich do **produktového backlogu**
- **Scrum tímy** majú **mať 1 vlastníka produktu**
- Táto **rola sa nemá spojiť so Scrum masterom**
- **Má byť na obchodnej strane projektu**, a **nikdy nemá interagovať** s členmi tímu o **technických aspektoch vývoja**
- Táto úloha je rovnaká ako **rola customer representative (reprezentant klientov)** v iných agilných frameworkoch



Vlastník produktu (product owner)

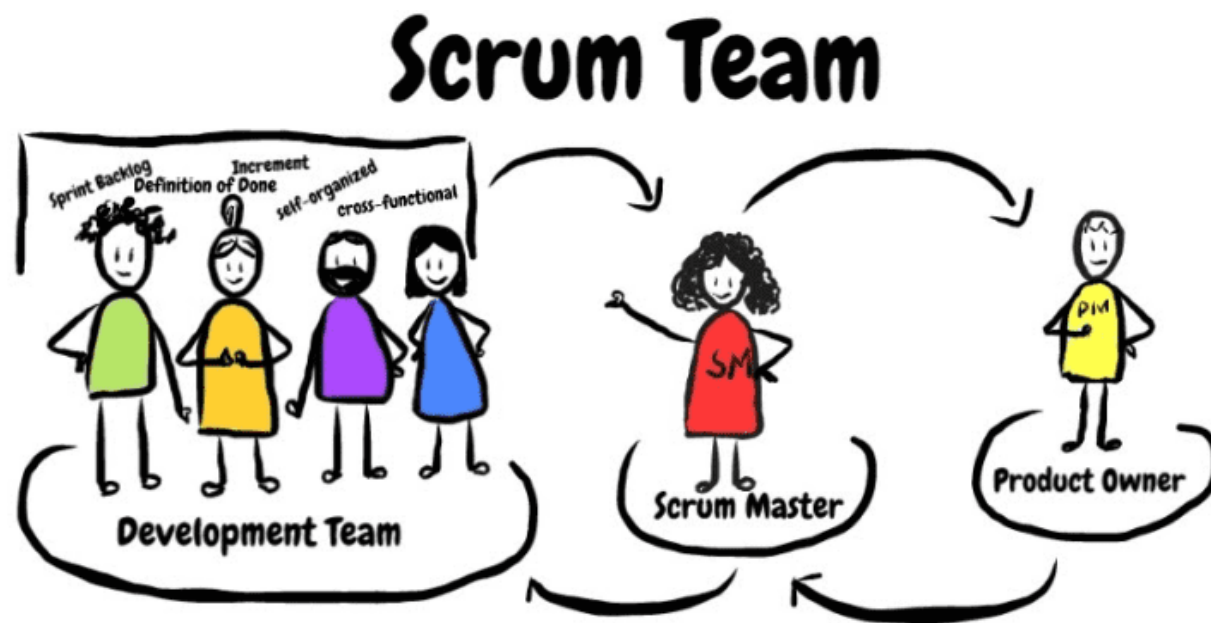
Komunikačné úlohy pre **zainteresované strany**:

- **Demonštrovať riešenie pre kľúčových stakeholderov**, ktorí neboli na iteračnom deme
- **Ohlasovať uvedenie nových verzií**
- **Komunikovať stav tímu**
- Organizovať míľnikové prehľady
- Vzdelávať v procese vývoja
- **Dohadovať priority, rozsah, financovanie a rozvrh**
- Uistiť sa, že produktové testy sú viditeľné, transparentné a jasné



Vývojový team (Development team)

- **Zodpovedný za dodanie potenciálne použiteľných inkrementov** (potentially shippable Increments - PSIs) produktu **na konci každého šprintu** (cieľ šprintu)
- **Zložený z 3-9 jednotlivcov**, ktorí robia aktuálnu prácu (analýza, dizajn, vývoj, test, technická komunikácia, dokument...)
- Sú **viacfunkčné**, so **všetkými zručnosťami** vytvoriť **produktový inkrement**
- Je **sebeorganizujúci**, aj keď tu **môže byť nejaký stupeň interakcie s projektovým manažmentom** (project management offices - PMOs)



SCRUM MASTER

22



What I think I do



What my mom
thinks I do



What my team
thinks I do



What my boss
thinks I do



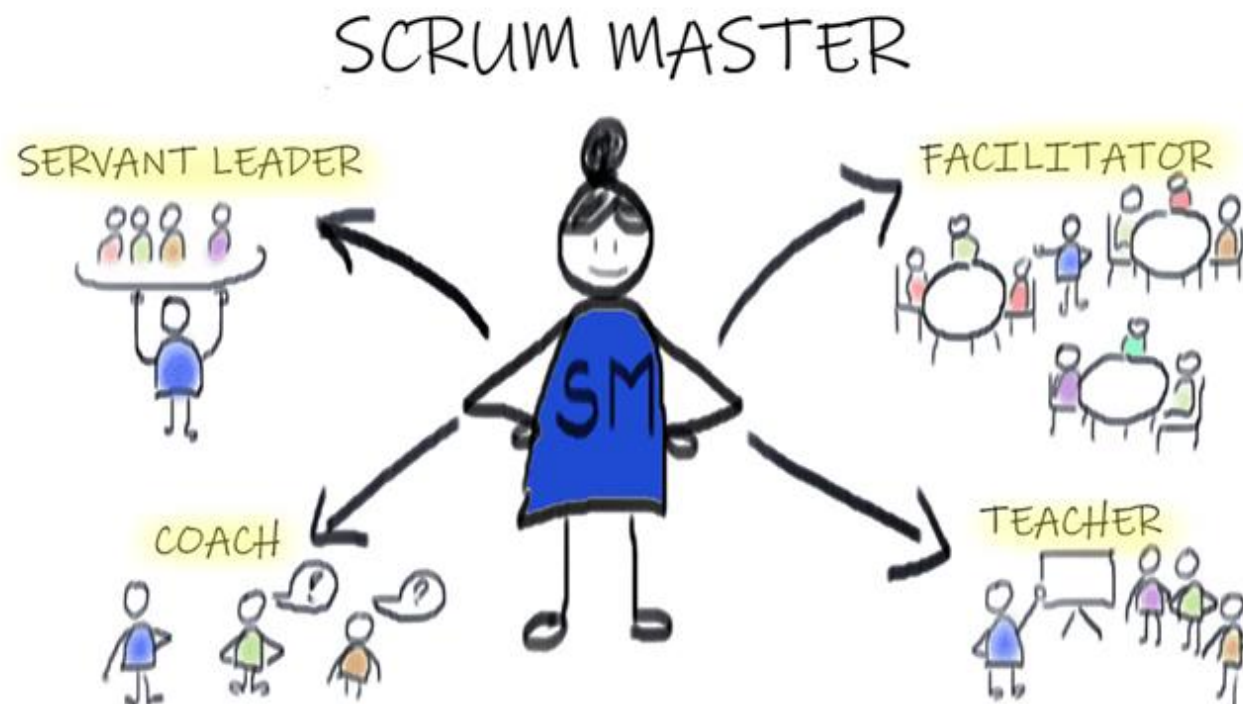
What society
thinks I do



What I really do

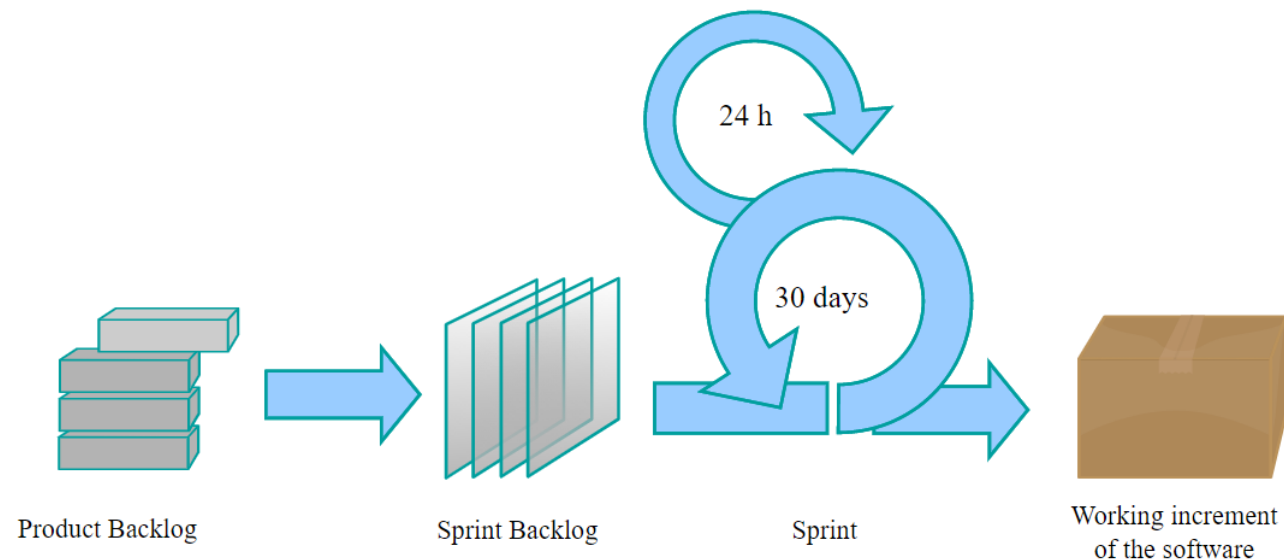
Scrum master

- **Zodpovedný** za **odstránenie prekážok** **teamu** na dodanie produktových cieľov
- **Nie je tradičný team leader** alebo **projektový manažér**, ale koná ako **sprostredkovateľ** medzi teamom a akýmkoľvek negatívnymi vplyvmi
- **Zaistuje**, že **Scrum proces je použitý** tak, ako bolo **naplánované** a členovia tímu **dodržiavajú dohodnuté procesy**
- Často **organizuje stretnutia** a **povzbudzuje tím** k zlepšeniu
- Táto rola sa niekedy označuje ako "**team facilitátor**" na zdôraznenie duálnej funkcie

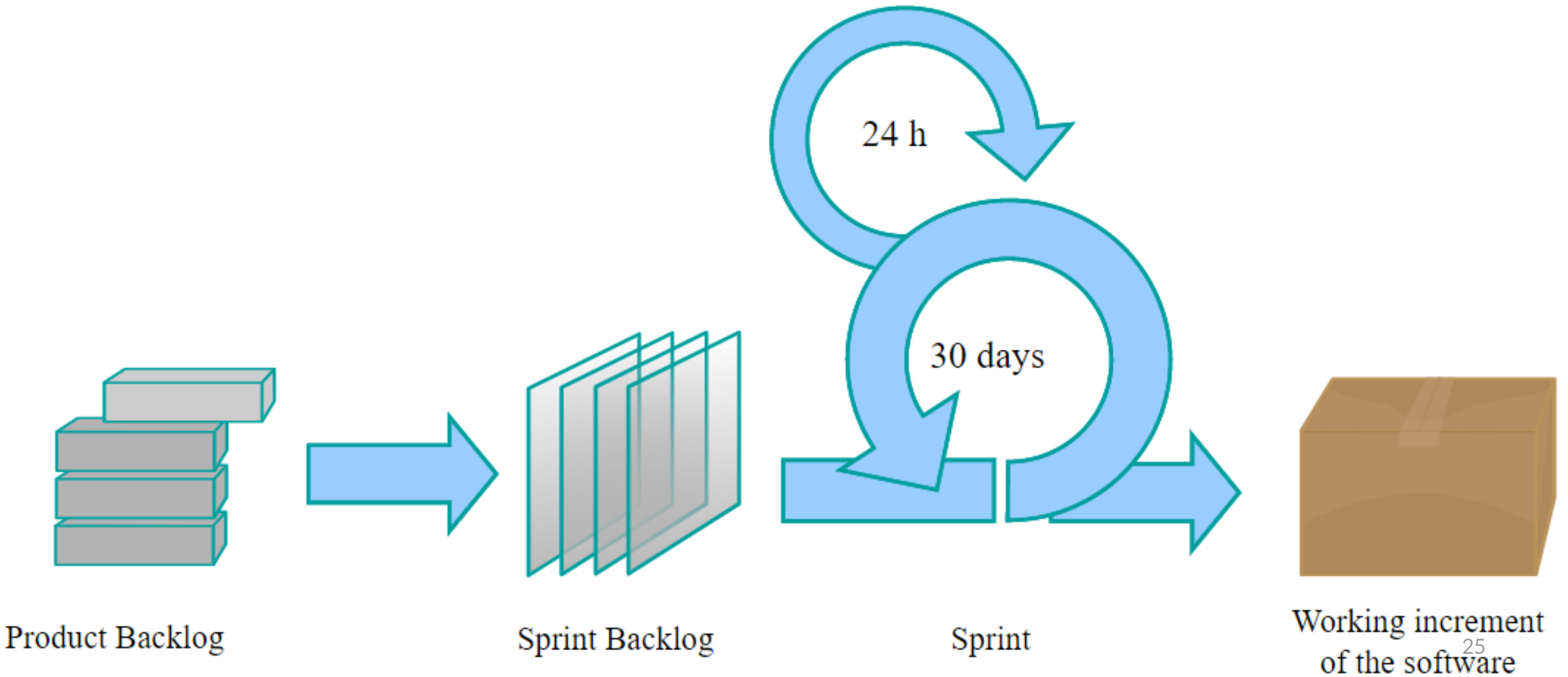


Šprint (Sprint)

- Sprint (alebo iterácia) je **základná jednotka vývoja**
- Časovo obmedzená snaha, teda je **obmedzený na špecifický čas**
- Doba je **určená dopredu** pre každý šprint a zvyčajne je to **1 týždeň až 1 mesiac, najčastejšie 2 týždne**
- **Začína** eventom **plánovanie šprintu**, cieľom ktorého je **definovať úlohy šprintu**, kde je **definovaná práca šprintu** a **odhadnutý záväzok** pre **cieľ šprintu**
- **Končí** **recenziou šprintu** (šprint **review**) a **retrospektívou**, kde je **reportovanie progresu** pre **stakeholderov** a definujú sa **úlohy** na **zlepšenie** pre **d ďalšie šprinty**
- Scrum zdôrazňuje spracovaný produkt na konci šprintu, ktorý je naozaj hotový, v prípade softvéru to môže znamenať, že softvér bol integrovaný, kompletne testovaný, zdokumentovaný a potenciálne môže byť dodaný.



Šprint (Sprint)



Udalosti (Eventy)

Plánovanie šprintu

Na **začiatku šprintu**, tím organizuje **šprint planning event** - plánovacie udalosť:

- **Vybrať, aká práca** sa **urobí**
- **Pripraviť úlohy** šprintu, ktoré **určia čas** na úlohy **pre celý tím**
- **Definovať a diskutovať koľko práce** je **potrebné urobiť počas šprintu**
- **4-hodinový limit pre 2-týždenný šprint**, pomerne pre inú dĺžku
- **V 1. polovici sa celý tím** (vývojový tím, scrum master a product owner) dohodnú, ktoré **produktové položky zo zásobníka** (backlog) budú v **danom šprinte realizovať**
- **V 2. polovici vývojový tím určí prácu** (úlohy) potrebné na **dodanie položiek backlogu** a **zapíše ich do sprintového backlogu**

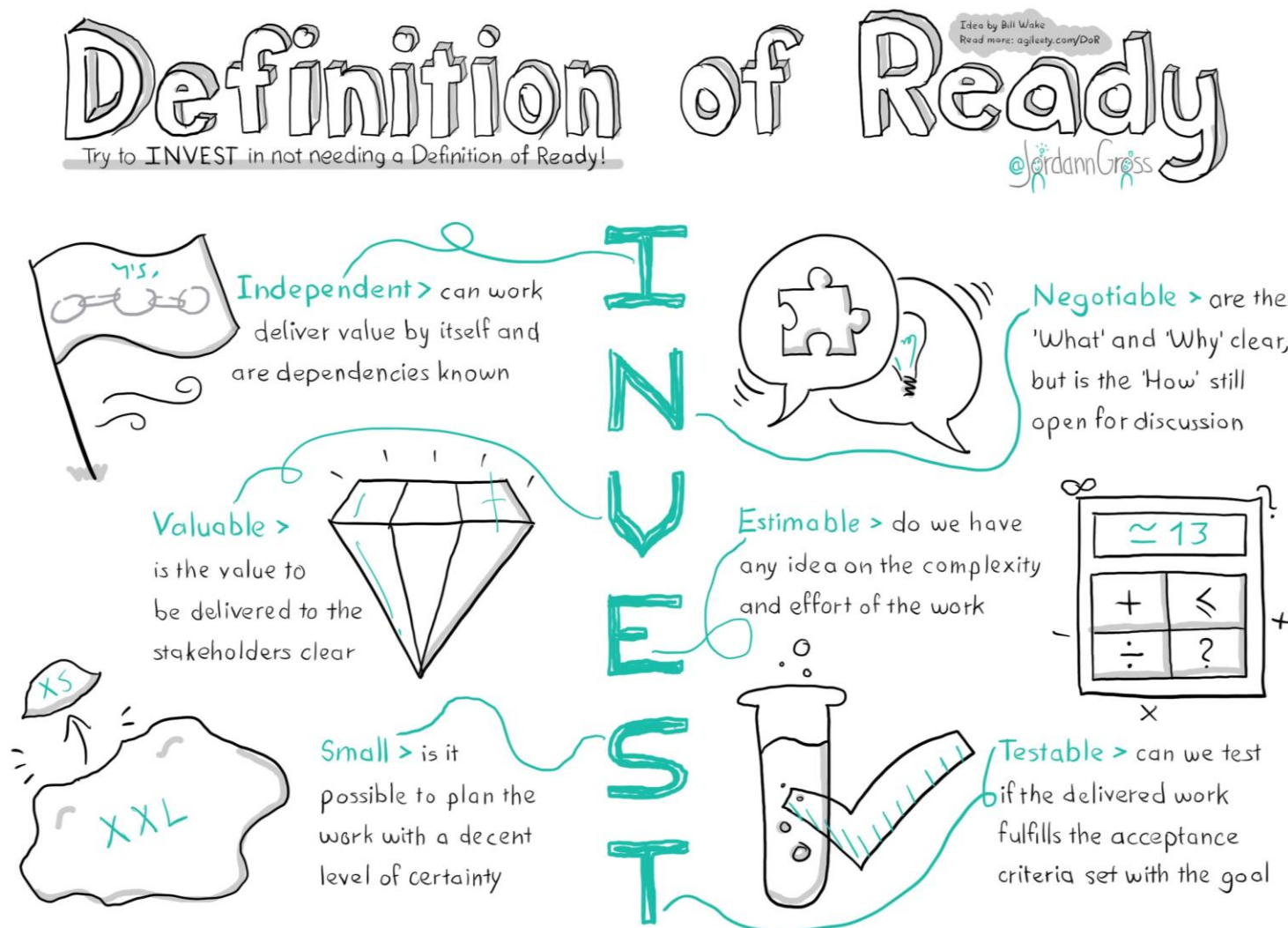
Denný Scrum

Každý deň počas šprintu team **organizuje denný Scrum** (alebo stand-up) so špecifickými zásadami:

- **Všetci** členovia softvérového tímu prídu **pripravení**
- **Začne sa presne načas** aj keď niektorí členovia chýbajú
- **Prebehne každý deň** na **rovnakom mieste** a v **rovnaký čas**
- **Dĺžka** je obmedzená na **15 minút**
- Každý je vítaný, aj keď **zvyčajne hovoria len teamové role scrumu**

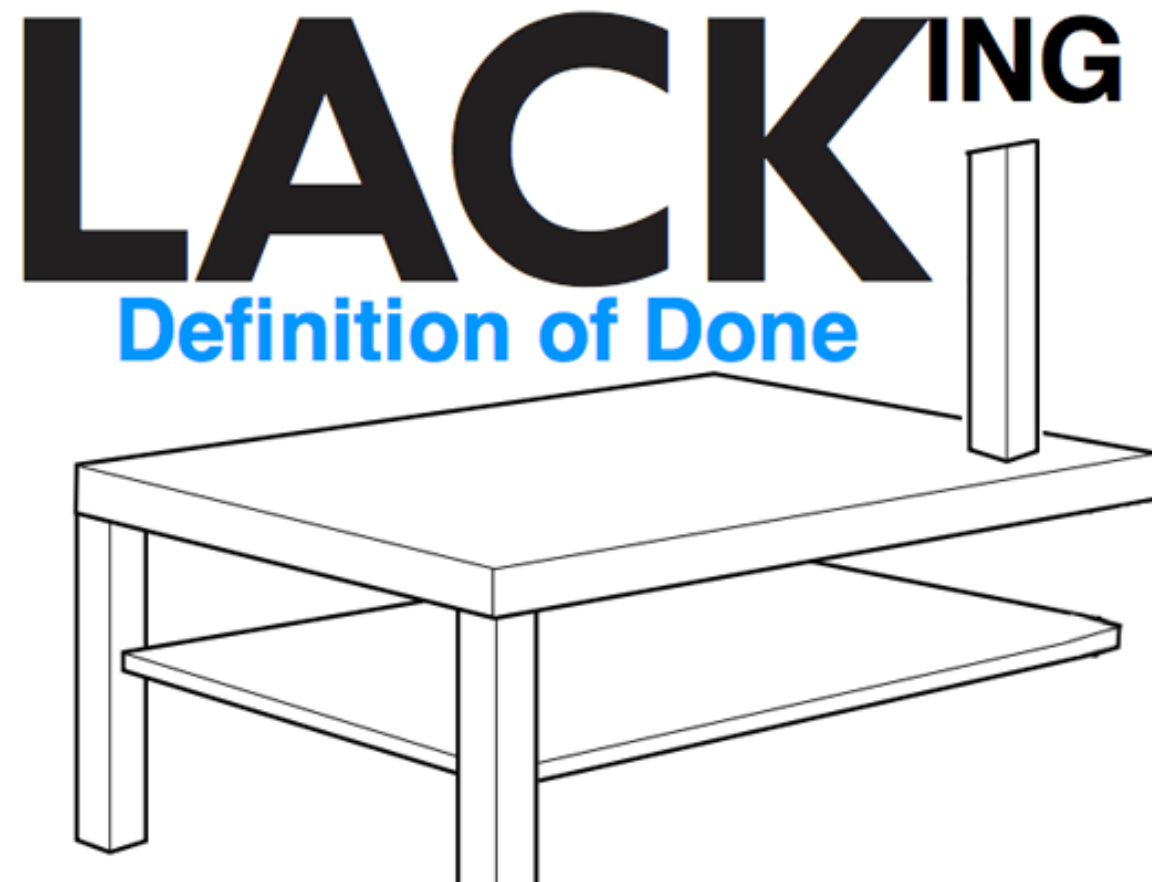
Definícia pripravenosti (Definition of ready DoR)

- Počiatkové kritériá na určenie, či sú špecifikácie a vstupy dostatočne nastavené na spustenie pracovnej položky, t. j. používateľského príbehu

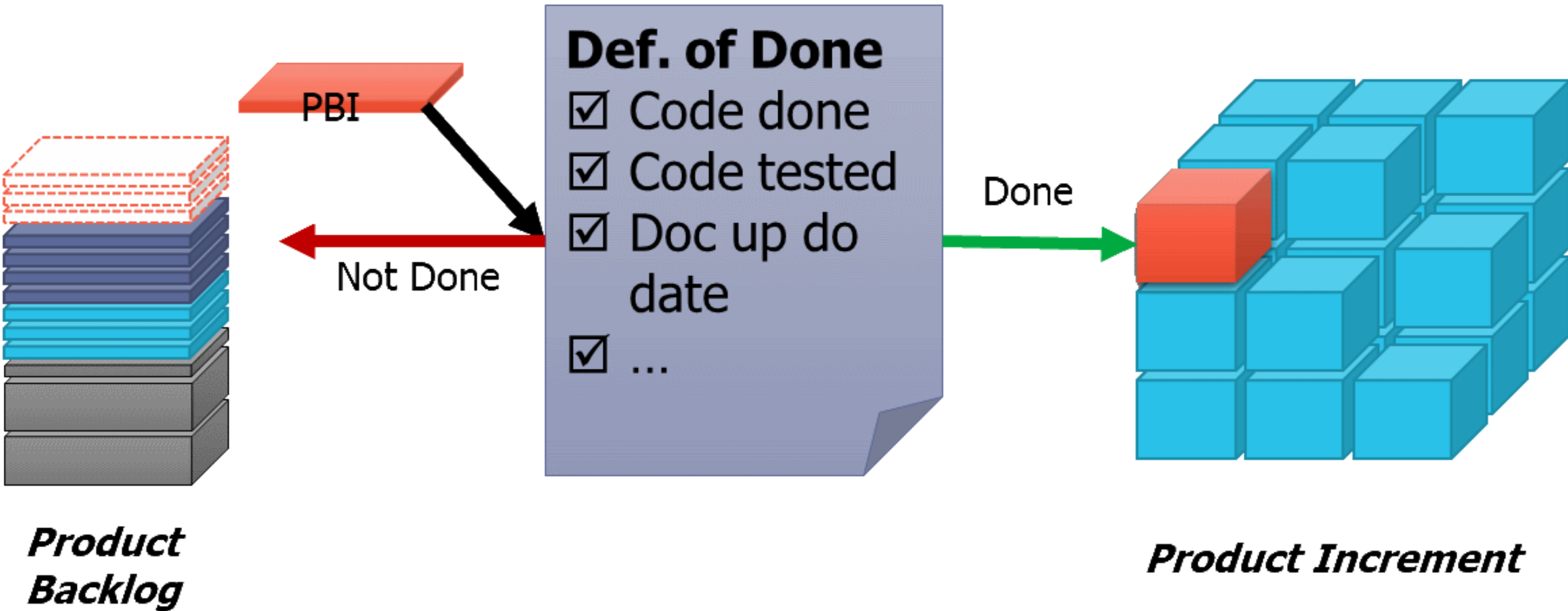


Definícia kedy máme hotovo (Definition of done DoD)

- Kritériá ukončenia na určenie, či je položka nevybaveného produktu úplná
- V mnohých prípadoch DoD vyžaduje, aby boli všetky regresné testy úspešné
- Definícia hotového sa môže líšiť od jedného scrum tímu k druhému, ale musí byť konzistentná v rámci jedného tímu



Definition of done DoD



Definition of done DoD

Definition of Done (DoD)

The below examples might be included in a User Story DoD:

- Unit tests passed
- Code reviewed
- Acceptance criteria met
- Functional Tests passed
- Non-Functional requirements met
- Product Owner accepts the User Story

User Story: Definition of Done Checklist	Sprint: Definition of Done Checklist	Release: Definition of Done Checklist
✓ Code builds with no error	✓ Satisfied DoD for each user story in the sprint	✓ Satisfied DoD for each sprint in the release
✓ Unit testing is complete	✓ Marketing feedback is implemented	✓ Production environment is ready
✓ Code review is complete	✓ Legal / compliance review is complete	✓ CI / CD verified and working
✓ Localization & translation is complete	✓ User help guide created or updated	✓ User help guide localized
✓ Localization testing passed	✓ Training video created or updated	✓ Training video localized
✓ Browser and / or device compatibility testing is complete	✓ Refactoring is complete	✓ Rollback process is documented
✓ Regression testing is complete	✓ Configuration or build changes documented	✓ Smoke testing scenarios are ready
✓ Automation tests are written and passed	✓ Performance testing is complete	✓ Customer Support team is trained
✓ Acceptance criteria is met	✓ Security testing is complete	✓ Release communications are sent
✓ Signed off by Product Owner	✓ Sprint marked as ready for deployment	✓ All stakeholders signed off for the release

Definition of done DoD

Definition of "done"



1. Clothes sorted?
2. Picked the right detergent/fabric softener?
3. Pre-treated stains?
4. Set washer temperature and cycle (depending on the clothes load)
5. Loaded the washer?
6. Checked when laundry done to unload?
7. Load dryer
8. Set dryer temperature and cycle
9. Unload dryer
10. Hang or fold laundry
11. Carry clothes in laundry basket to room
12. Organize clothes in closet

To Do

Doing

Done

USE
KANBAN

TRY
KANBAN
TOOL

LEARN
ABOUT
KANBAN

GET SOME
STICKY
NOTES!

GET A
WHITE -
BOARD



Product Backlog ✕

Organize Job Fair

Place Job Ad.

General Activity

Find Job

Find Job

Manage Vacancy

Manage Vacancy

Recruit Candidate

Recruit Candidate

General User Task

Browse Jobs

Post Resume

Receive Job Alert

Post Vacancy

Amend Vacancy

Cancel Vacancy

Search Candidates

Contact Candidates

1.0



Log system events

WIP

0 / 2

Perform general search

WIP

0 / 1

Upload resume

WIP

0 / 4

Subscribe for 'Job Alert'

WIP

0 / 1

Submit job vacancy

WIP

0 / 1

Edit vacancy

WIP

0 / 1

Remove a vacancy

WIP

0 / 1

Review application list

WIP

0 / 1

Send PM to seeker

WIP

0 / 1

Perform advanced search

WIP

0 / 3

Register as Employer

WIP

0 / 3

Confirm vacancy removal

WIP

0 / 1

2.0



Archive system data per hour

0 / 1

Build resume online

0 / 3

Receive SMS notification for Job Alert

0 / 1

Approve vacancy submission

0 / 4

Approve vacancy amendment

0 / 3

Receive notification for expired post

0 / 1

Perform advanced candidate

0 / 1

Invite for interview

Unscheduled



Save search settings within session

Complete job seeker questionnaire

Duplicate vacancy

Bookmark a candidate

User Stories

Theme

Epic

Epic

User
Stories

User Stories

User Stories

Task

Task

Task

Task

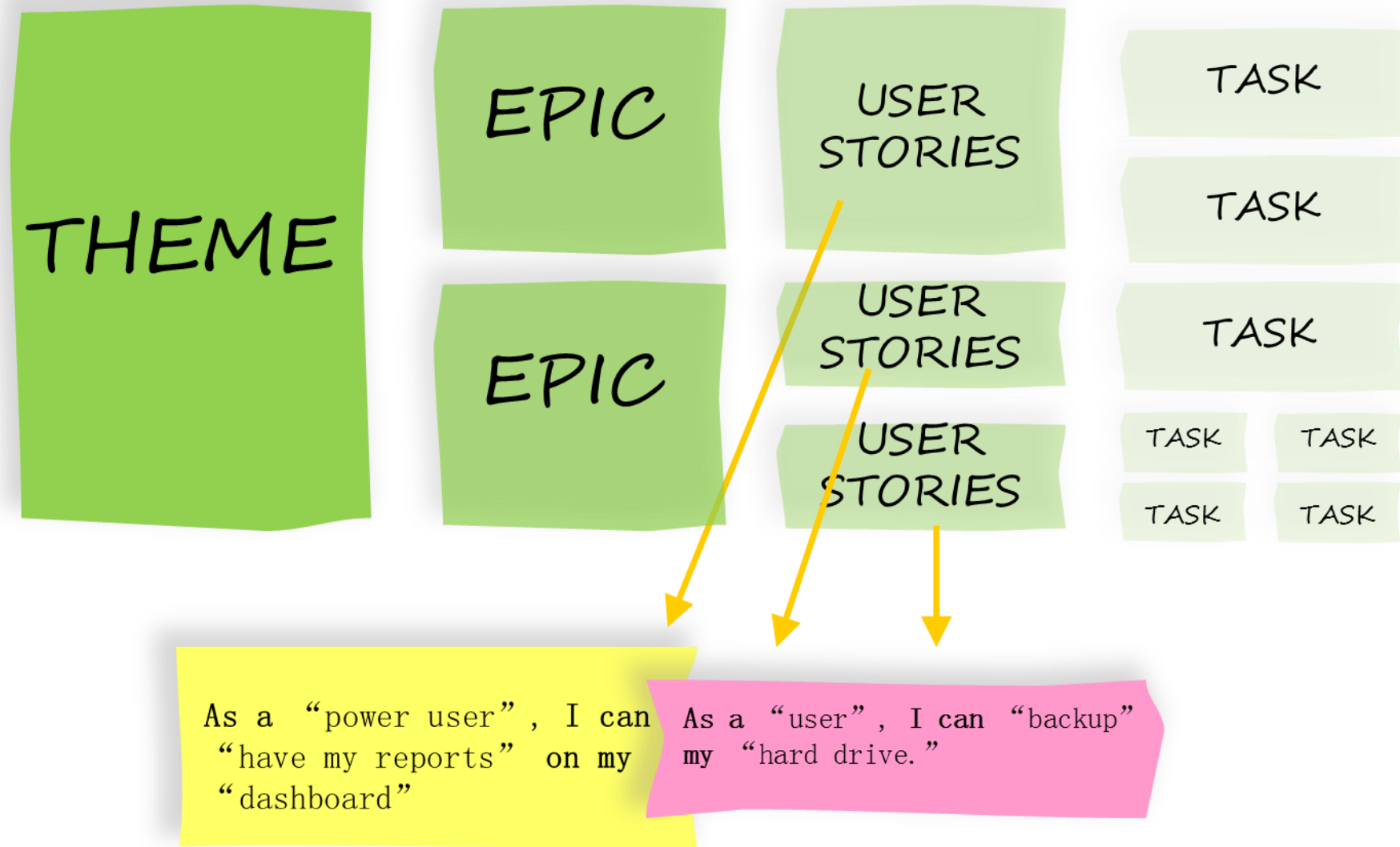
Task

Task

Task

USER STORIES

35



Writing a user story

- 1 Define your **end user**
- 2 Specify what **they want**
- 3 Describe **the benefit**
- 4 Add **acceptance criteria**

How to write an agile user story

- 1** Define your **end user**
Who will be using your product?

As a parent,

- 3** Describe **the benefit**
What will your user gain
from using your product?

so that I know he is safe
without disturbing him.



- 2** Specify what **they want**
What solution are you offering?

I want to check on my
sleeping baby without
entering his room,

- 4** Add **acceptance criteria**
What determines this story as
'done'?

e.g. Alert to be sent to the
registered smartphone
if problem is detected.

Top tips to create a good user story

- ✎ Develop a persona profile to visualise your end user
- ✎ Always write from your end user's perspective
- ✎ Avoid adding technical details too early
- ✎ Try not to add too many acceptance criteria
- ✎ Keep stories brief, breaking them up if you need to
- ✎ Make sure they meet your "definition of done"

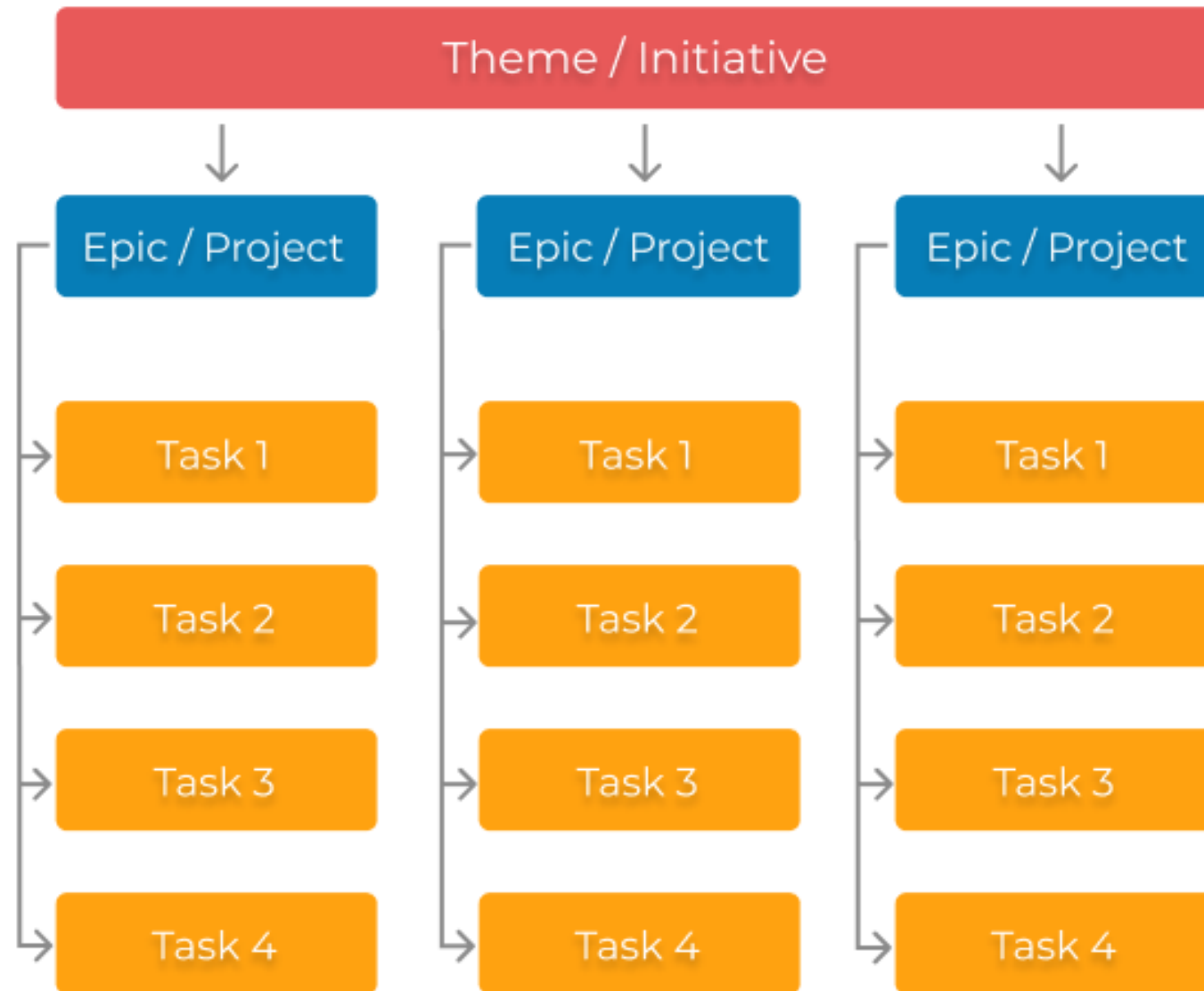
As a <user role>

I want <goal>

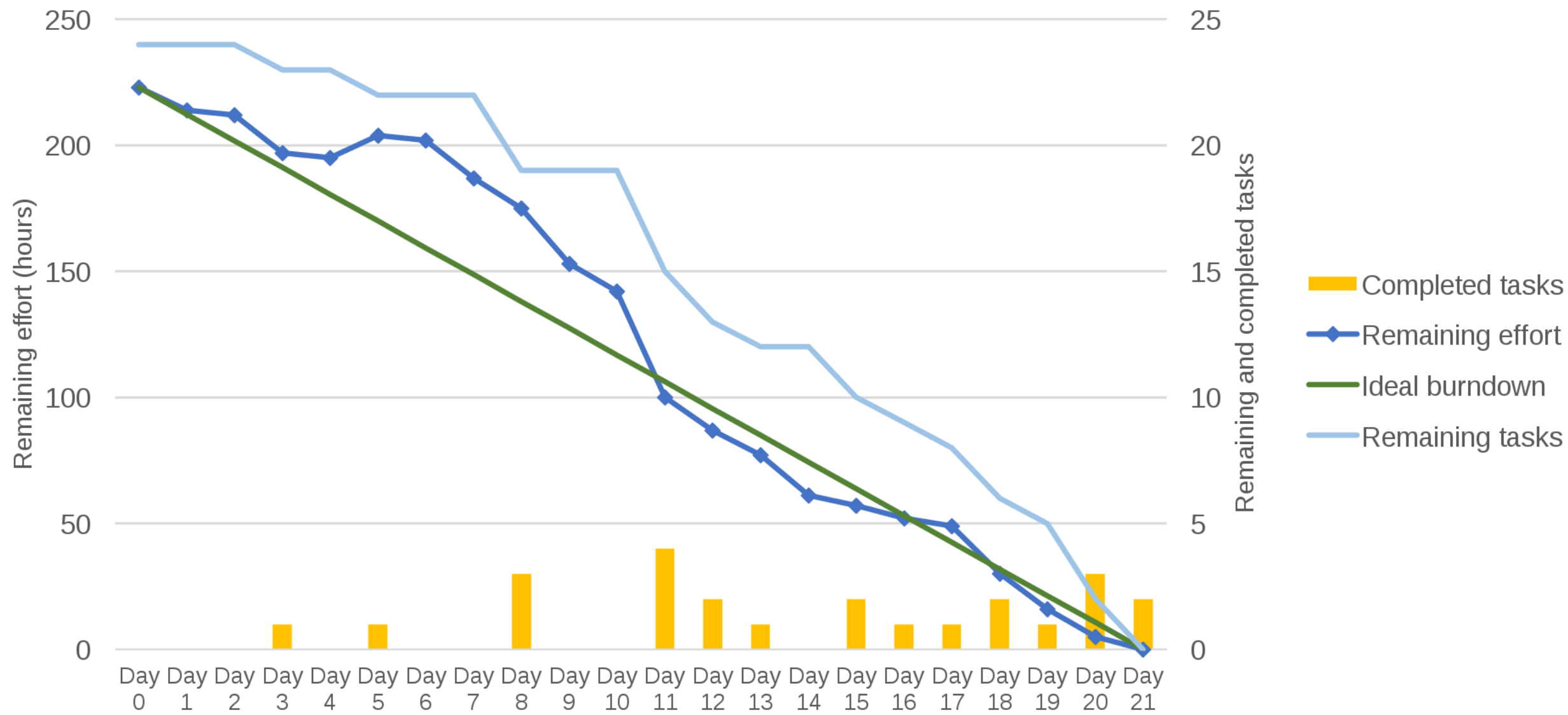
so that <benefit>.

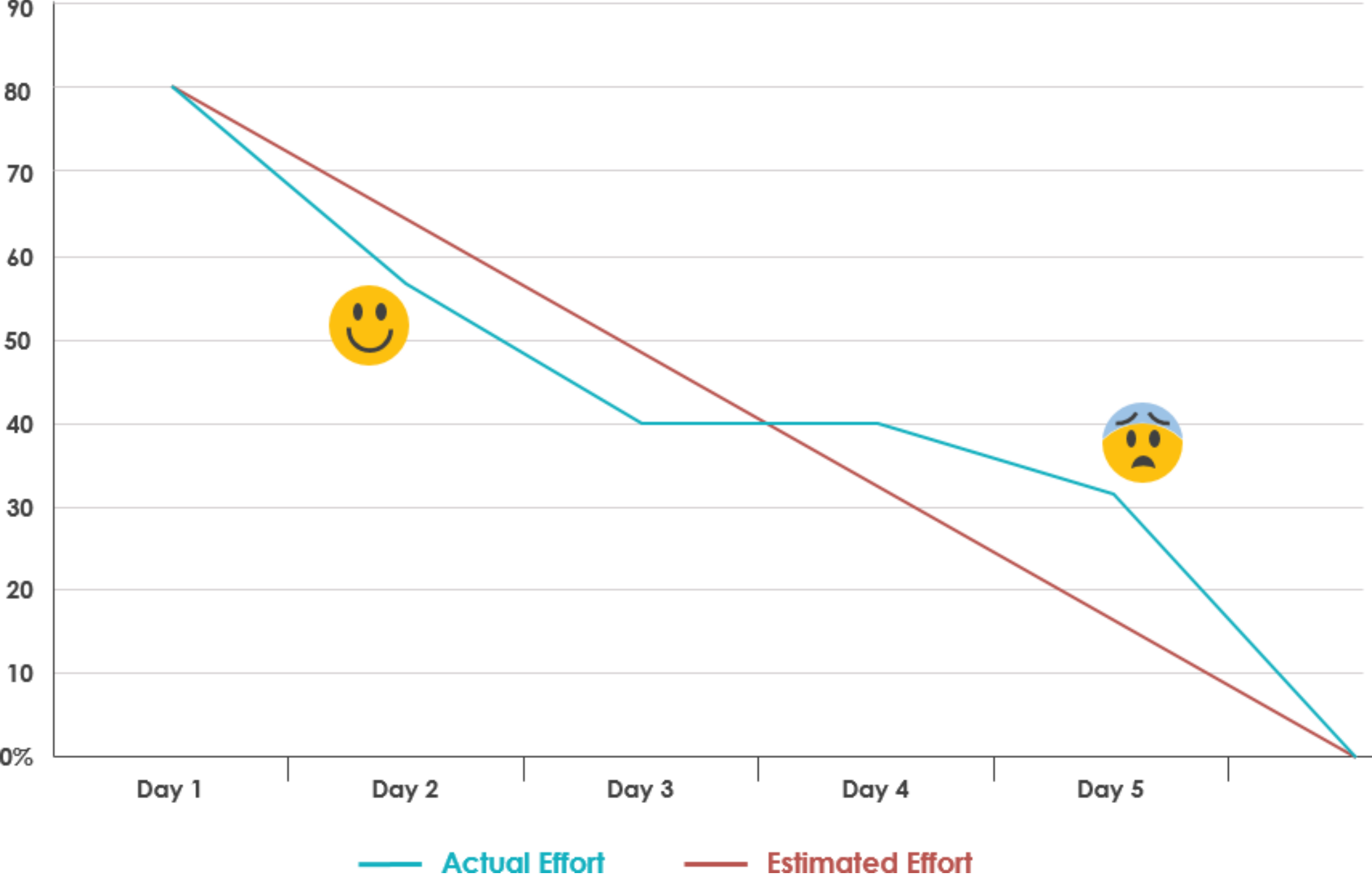
Stories	Not started	In progress	Done
<div>Story #1</div>			<div>Task A</div> <div>Task B</div> <div>Task C</div>
<div>Story #2</div>	<div>Task A</div>	<div>Task C</div>	<div>Task B</div>
<div>Story #3</div>	<div>Task B</div> <div>Task D</div>		<div>Task A</div> <div>Task C</div>

Agile Tasks Breakdown



Sample Burndown Chart







Results

Progress Over Time

Personal Improvement Backlog

Community

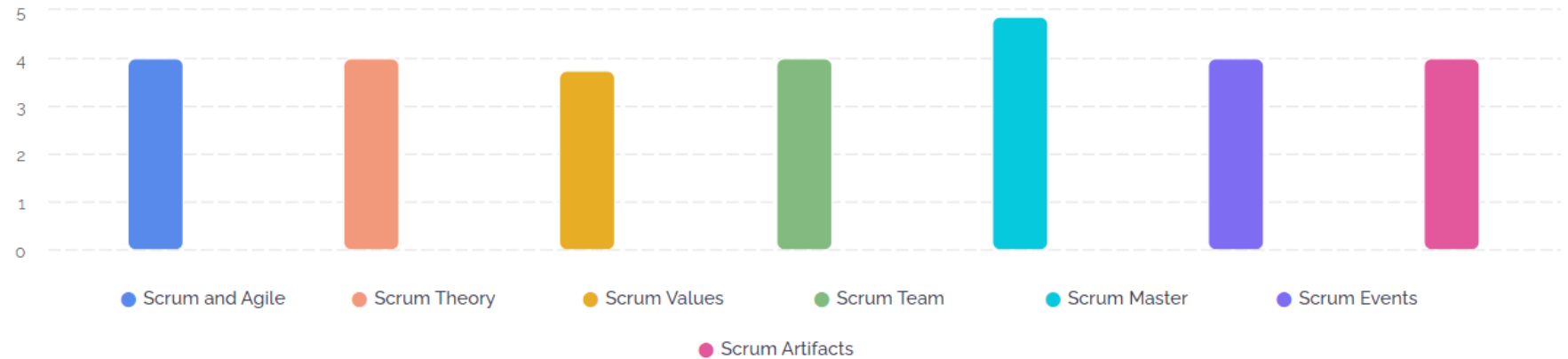
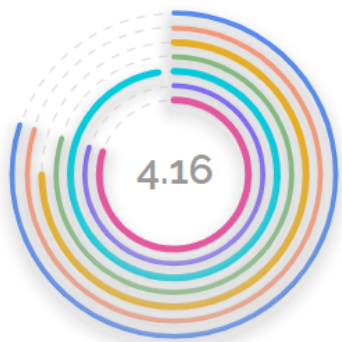


4.16

27.10.2020



Results



Feedback

Recommendations

Here are **the key** questions and **related resources** we recommend you **explore further**

Scrum Values

The team members personally commit to achieving the goals of the Sprint.

