

Scrum

principy agilního managementu,
metodika Scrum

Agenda

- **Agile**
 - O čem to celé je?
- **Scrum**
 - Artefakty
 - Role
 - Procesy
- **User Stories**
 - I.N.V.E.S.T.
 - US vs UC
 - Odhady



Jak a hlavně proč vznikl Agile

- **Změna fungování světa**
 - dull vs dynamic business
- **Změna práce jako takové**
 - manufacturing vs development
- **Změna přístupu k zaměstnancům**
 - odklon od Taylorismu / Fordismu
 - Frederick W. Taylor
 - doers x thinkers
 - process over people
- **Změna fungování organizací**
 - „employee empowerment“
 - F. Laloux – posun na spektru od červených k tyrkysovým organizacím

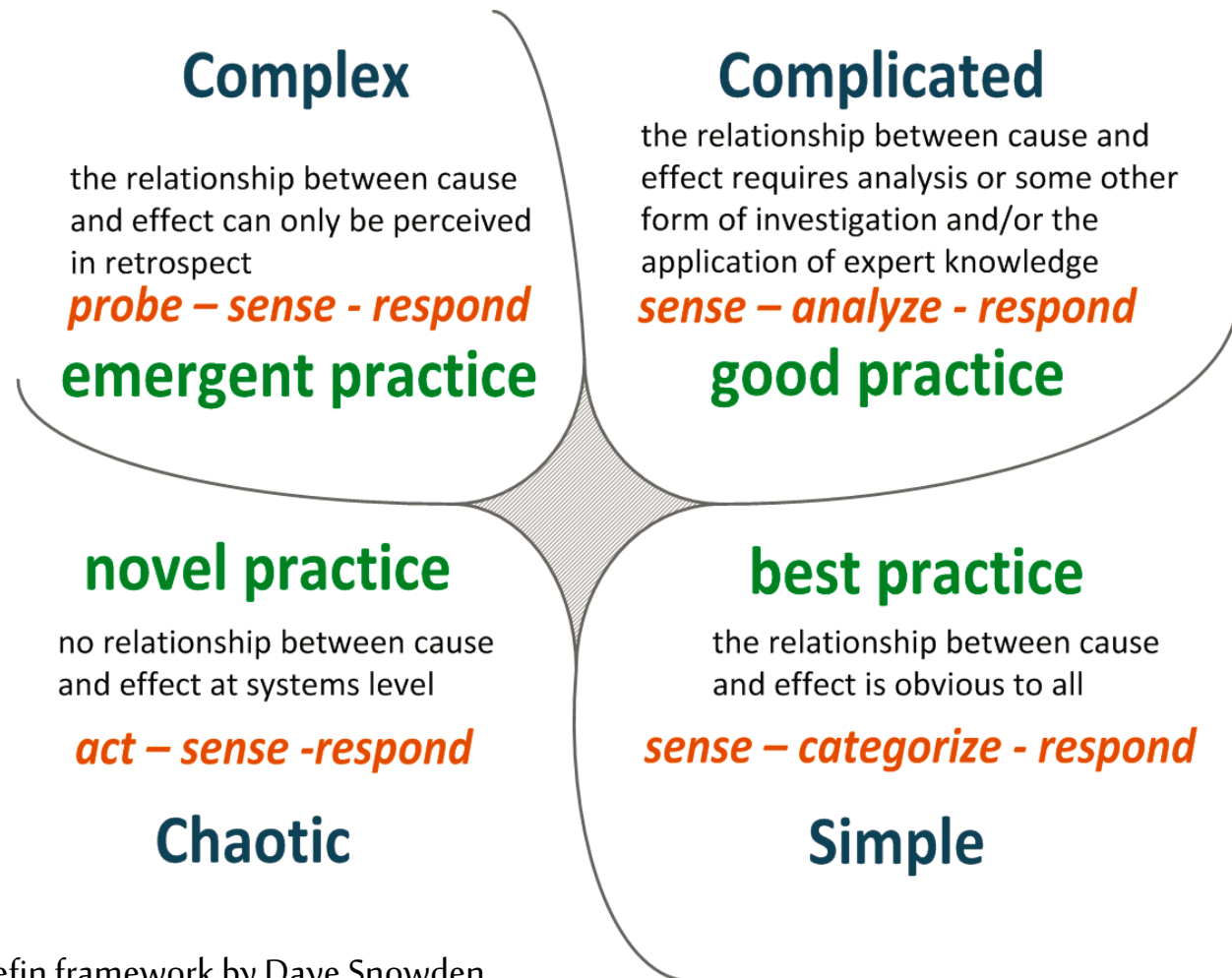


Co vůbec je Agile

- **Mindset** 😊
 - přístup k „organizaci práce“
- **Agile Manifesto (2001):**
 - **Individuals and interactions** over processes and tools
 - **Working software** over comprehensive documentation
 - **Customer collaboration** over contract negotiation
 - **Responding to change** over following a plan
 - **12 klíčových principů:** <http://agilemanifesto.org/principles.html>



Proč Agile?



Jak je na tom Agile?

- State Of Agile – Survey <http://stateofagile.versionone.com>
- jak je na tom Agile u nás?
- jak je na tom Agile u vás? ;-)
- kam Agile směřuje?



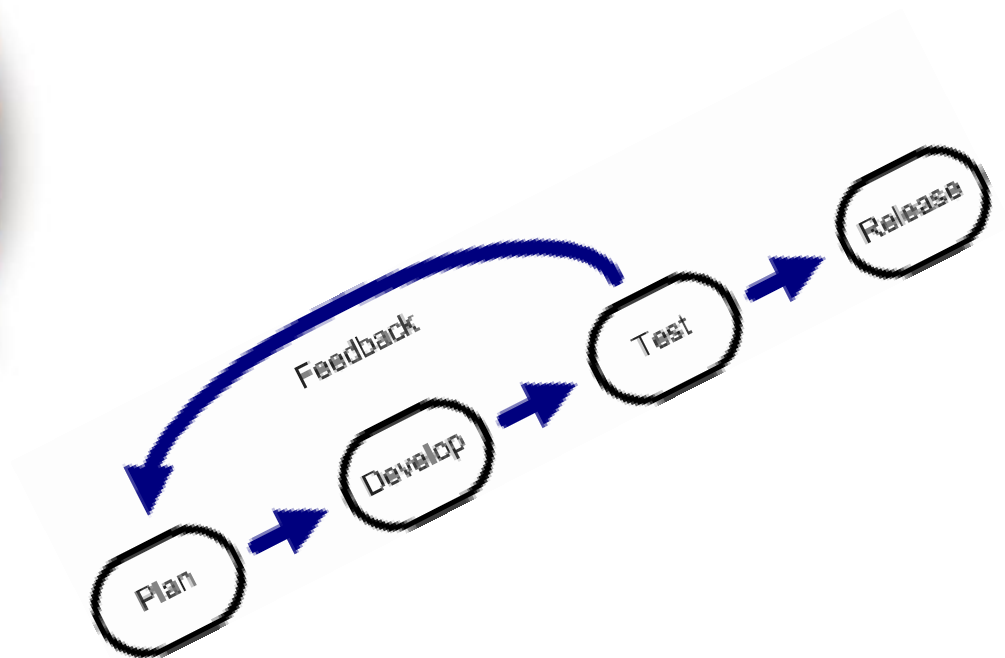
Agilní metodiky a praktiky

- KANBAN
- Scrum
- Lean
- TDD – Test Driven Development
- DevOps
- JIT
- KAIZEN
- ...



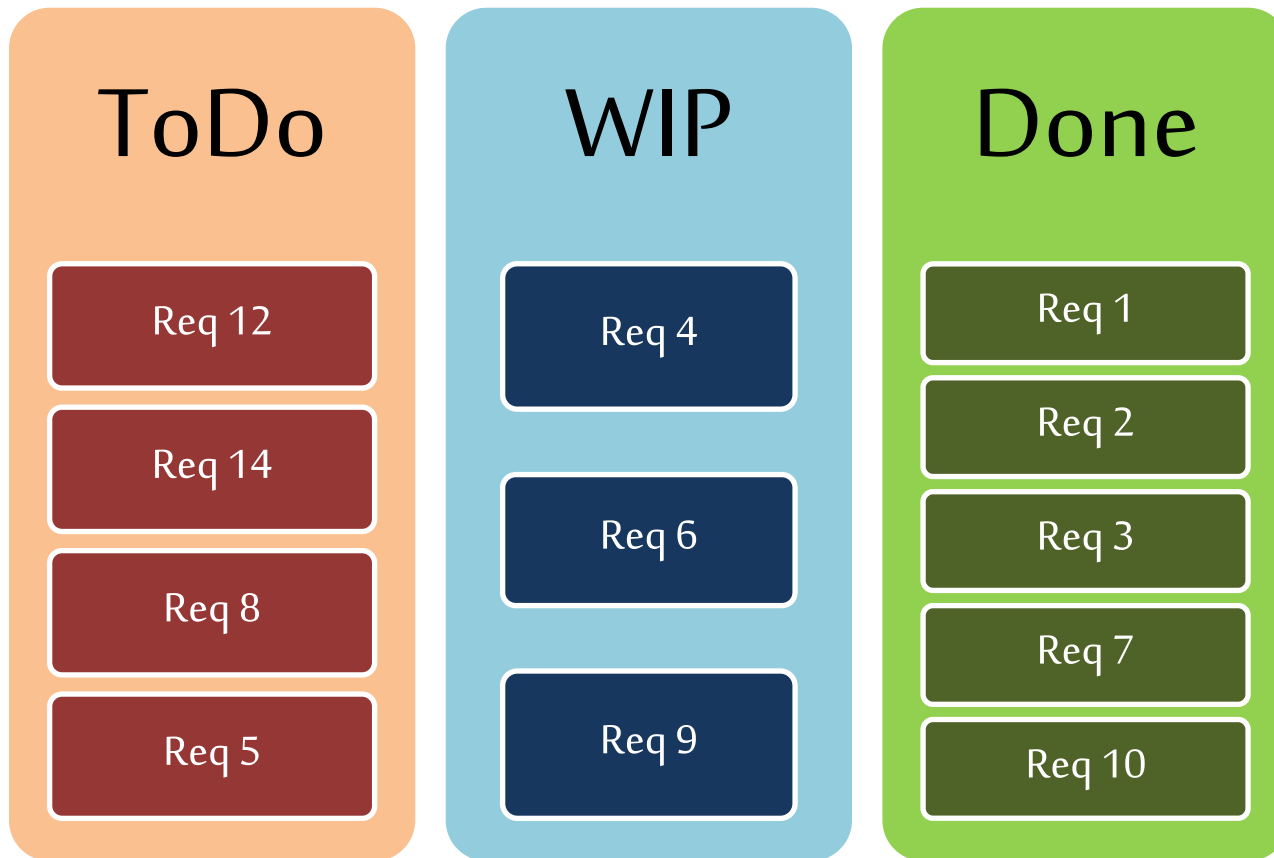
Scrum

- Vychází z **empirického procesu**

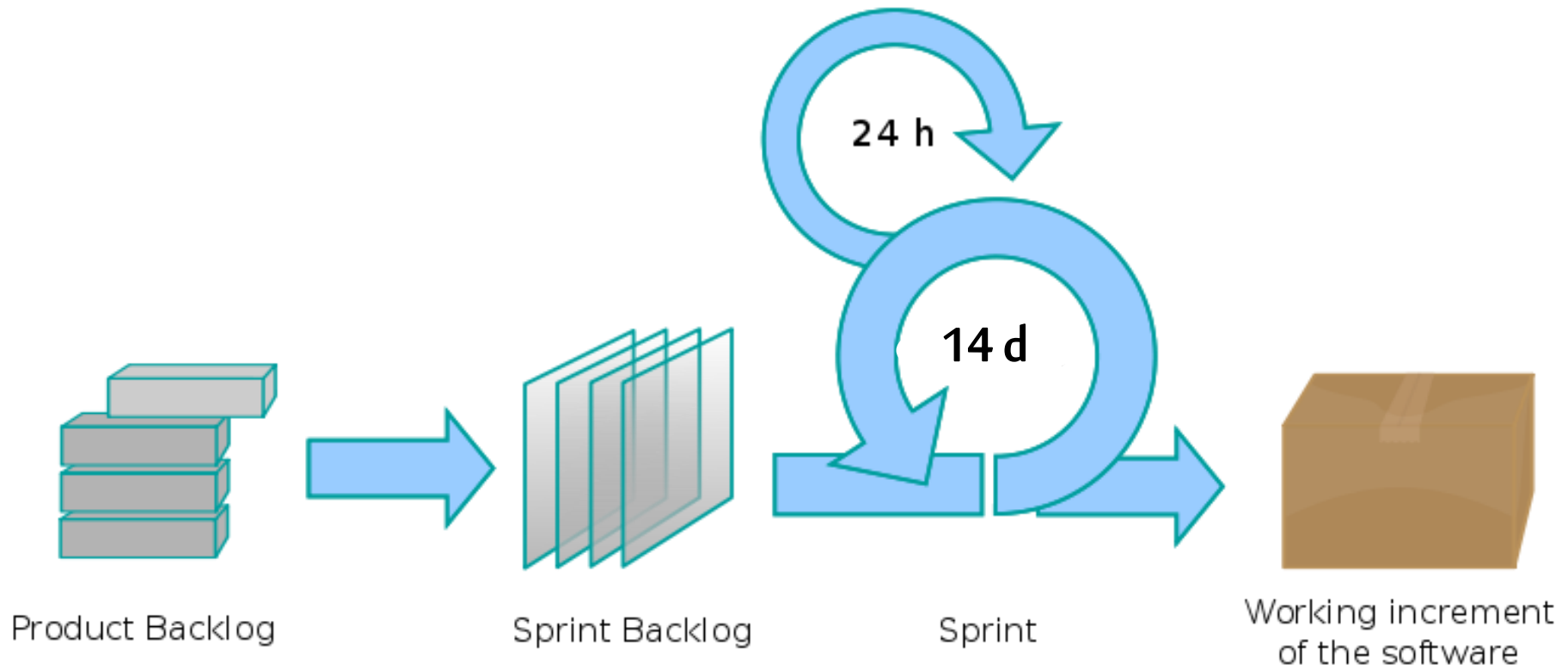


Scrum

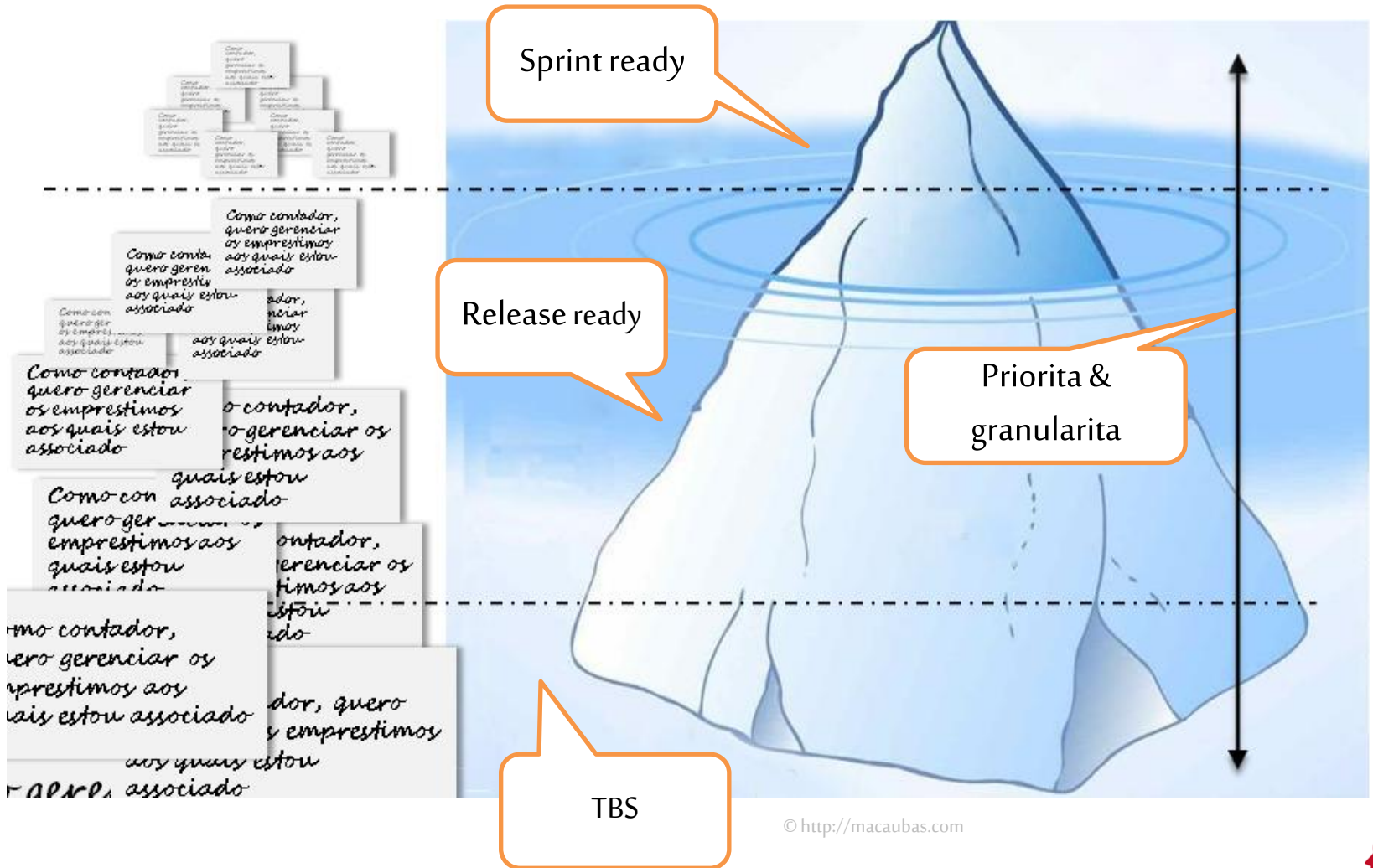
- Cyklický KANBAN ;-)



Scrum



Product Backlog



Product Backlog – D.E.E.P.

- Detailed appropriately
- Emergent
- Estimated (remaining effort)
- Prioritized

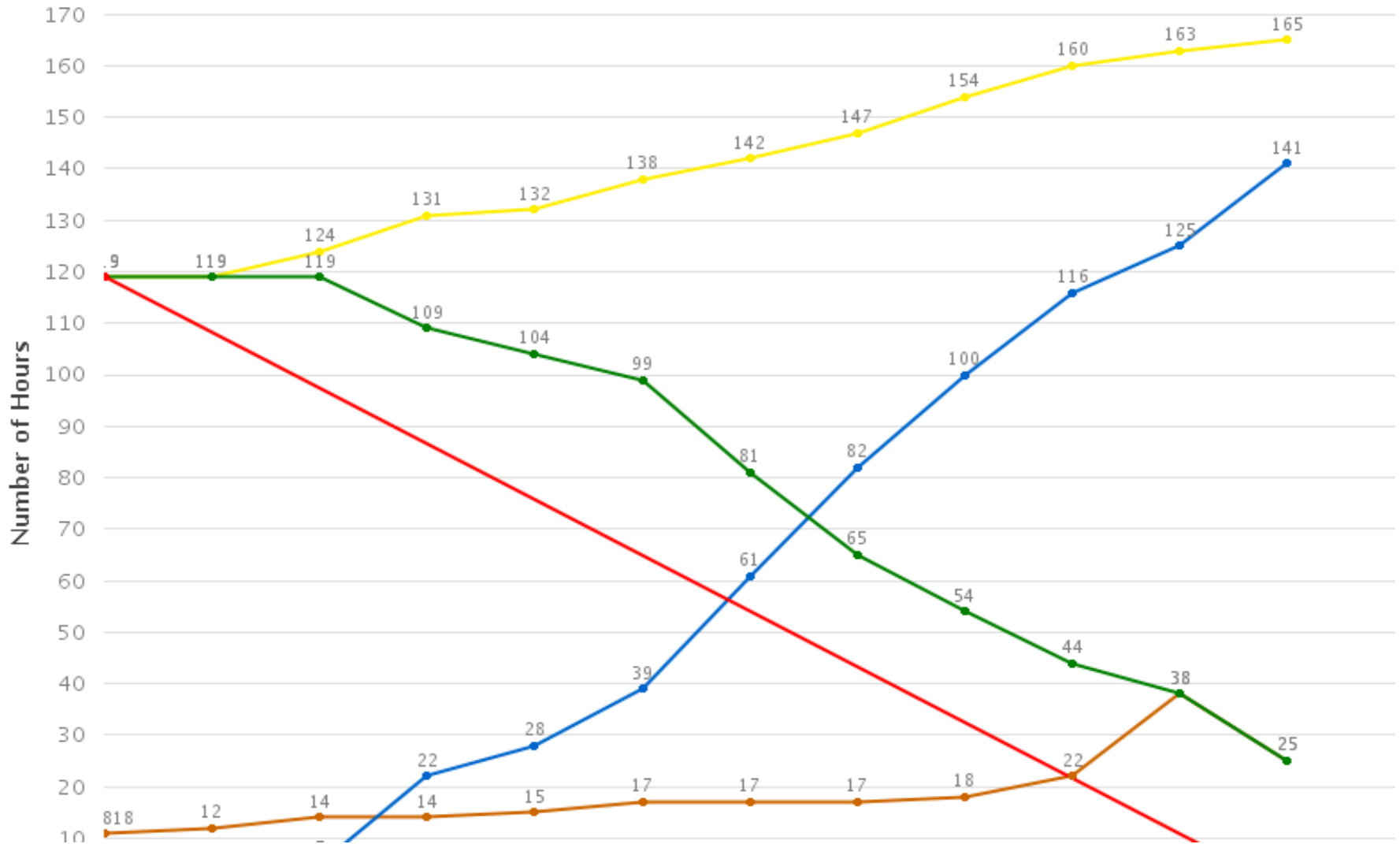


Sprint Backlog

- **Zodpovědnost teamu**
 - User Stories z PBL si **členové teamu sami řadí** do Sprint BL (dle priorit a toho, co zvládnou)
 - Sprint Planning Meeting
 - Sprint Goal, Forecast



Sprint Burndown Chart



Scrum Roles

- **Product Owner**

- Definuje produkt, prioritizuje úkoly, definuje milníky a scope
- Je zodpovědný za výsledný projekt (přidanou hodnotu)

- **Scrum Master**

- Je zodpovědný za procesy Scrumu, coaching teamu a PO
- Odstraňuje překážky procesů, podporuje kooperaci teamu

- **TEAM**

- Samoorganizující, určuje pracnost úkolů, určuje si sám co zvládne
- Zodpovědný za inkrement produktu – výsledek sprintu

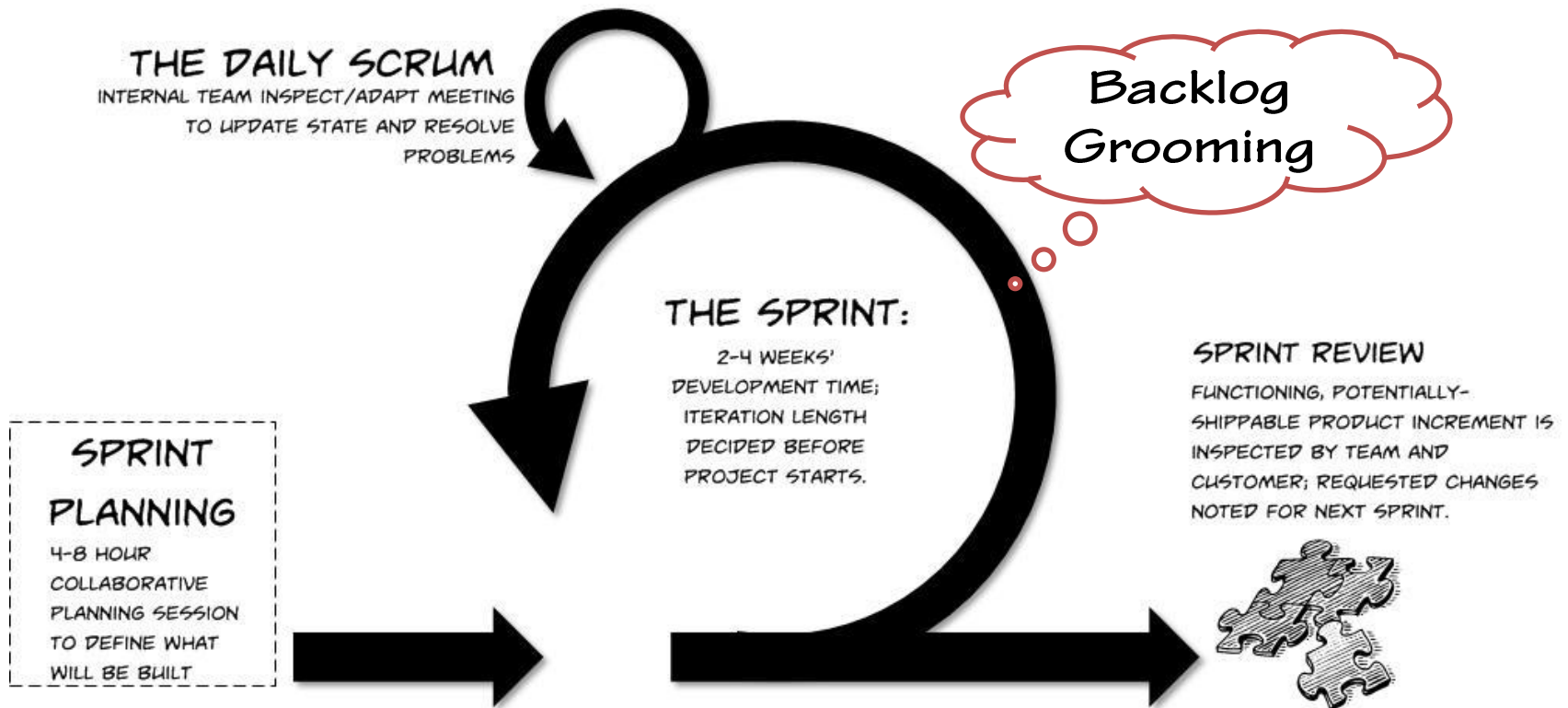


Plánování

- **Release planning**
 - Plán milníku
 - Hrubý seznam požadovaných featur
 - Odhad počtu sprintů
- **Sprint planning**
 - Goal / Forecast
 - Team commitment
 - Detailní seznam US s relevantními odhady
 - Rozpad na tasky (je li třeba)



Scrum Meetings



© derailleurconsulting.com

Scrum Meetings

- **Sprint Planning**
 - Sprint Goal and Forecast, upřesnění stories, revize odhadů, rozpad na tasky
- **Daily Scrum / standup**
 - Denní plán, akutní problémy
- **Sprint Review**
 - Vyhodnocení cílů sprintu – DEMO, naplnění Goals, Forecast
- **Retrospective**
 - Co se stalo a jak to udělat příště lépe? Problémy?
- **Product Backlog Grooming**
 - Údržba PBL, odhady, upřesnění stories



Sprint Planning Meeting

- **Team, PO, SM**, případně doménoví experti
- Cílem je stanovit **Sprint Goal** (a forecast)
- Výsledek:
 - PO ví co na konci sprintu dostane
 - DEV Team ví co má dělat
- Podmínky:
 - PO připravil Stories v BL a prioritizoval je
 - Team (s PO) US odhadl (Backlog Grooming)



Sprint Planning Meeting - průběh

- **Team** postupně odebírá položky z PBL
 - Diskutuje je s PO
 - Dekomponuje na tasky
 - Odhaduje náročnost tasků (US)
 - Zařazuje tasky do SBL
- Iterativně se opakuje dokud není vyčerpána kapacita teamu
 - Resp. VELOCITY – stanovená na základě předchozích sprintů
- Jen team plánuje co zvládne! (bez autonomie není odpovědnosti)



Sprint Planning - Special Tasks

- **Technical User Stories** – typicky např. „zprovoznění TST prostředí“ – úkoly CFG Managementu
- **Educational tasks** – např. školení, tech. hodinky atd.
- **Analytical tasks / Research** – někdy je nutný cílený výzkum pro pochopení dalšího úseku vývoje



STORY POINTS

- **Odhady náročnosti prací**
- **Story Point**
 - Bezrozměrné číslo
 - Je relativní
 - Vyjadřuje komplexnost dané feature / US / tasku
 - Vyjadřuje ale též míru neurčitosti / nejistoty!
 - **Výhody:**
 - Rychlost odhadu – odhad je jednodušší
 - Nezatížené „buffery“
 - Relativní odhady jsou pro lidi bližší
- **Jsou jen cestou, ne cílem ;-)**
 - Jak tedy fungují pokročilé teamy?



STORY POINTS – jak na to?

- **Triangulace** – „tahle feature je složitější než tamta (2SP), ale jednodušší než jiná (5SP) => náročnost 3SP
- **Analogie**
- **Expertní znalost**
- **Odhad nemusí být „přesný“ - je to jen odhad 😊**
 - Ve výsledku se kladné a záporné „nepřesnosti“ vyrovnají
 - Trocha úsilí přinese dostatečnou přesnost, další úsilí odhad již příliš nezlepší
- **Odhady určuje team, nikdo jiný!**



Planning Poker

- **Iterativní odhad** v teamu („moudrost davu“)
- **Pevné měřítko** – karty pro agilní plánování
0 – ½ – 1 – 2 – 3 – 5 – 8 – 13 – 20 – 40 – 100
- Typicky stačí ke konsenzu 2 - 3 iterace
- Musí být přítomen PO kvůli otázkám a upřesněním featur
- **Je to zejména „platforma“ pro komunikaci!**
- **Výhody:**
 - Není manipulováno dominantními členy teamu
 - Podporuje komunikaci (v rámci teamu a s PO)
 - Eliminuje různé vnímání časové náročnosti úkolu
 - Team si odhaduje sám, nikdo nic nediktuje – motivační faktor



Daily Scrum / Standup

- **Synchronizace teamu**
- **SM, Team** (případně další, ale nemluví)
- **3 otázky – odpovídá každý člen teamu**
 - **Co jsem dokončil** od minulého DS?
 - **Co dokončím** do dalšího DS?
 - Jaké mám **problémy / překážky**?
- **Updatuje se burndown chart** (pokud se tak neděje automaticky v SW)
- **do 15 minut, standup** – mluví jen členové teamu
 - případné diskuse až po DS



- Pravidlo č. 1



Sprint Review

- **SM, Team, PO, stakeholderi**
- Prezentace DEMA („potentialy shippable product“)
- PO akceptuje, nebo odmítne výsledek a dává feedback
- Revize Sprint Goal + Forecast
- Stakeholderi vidí **DEMO**
- Revize burndown chartu + kalkulace VELOCITY
- Pivoting „projektu“ => možné změny priorit v BL

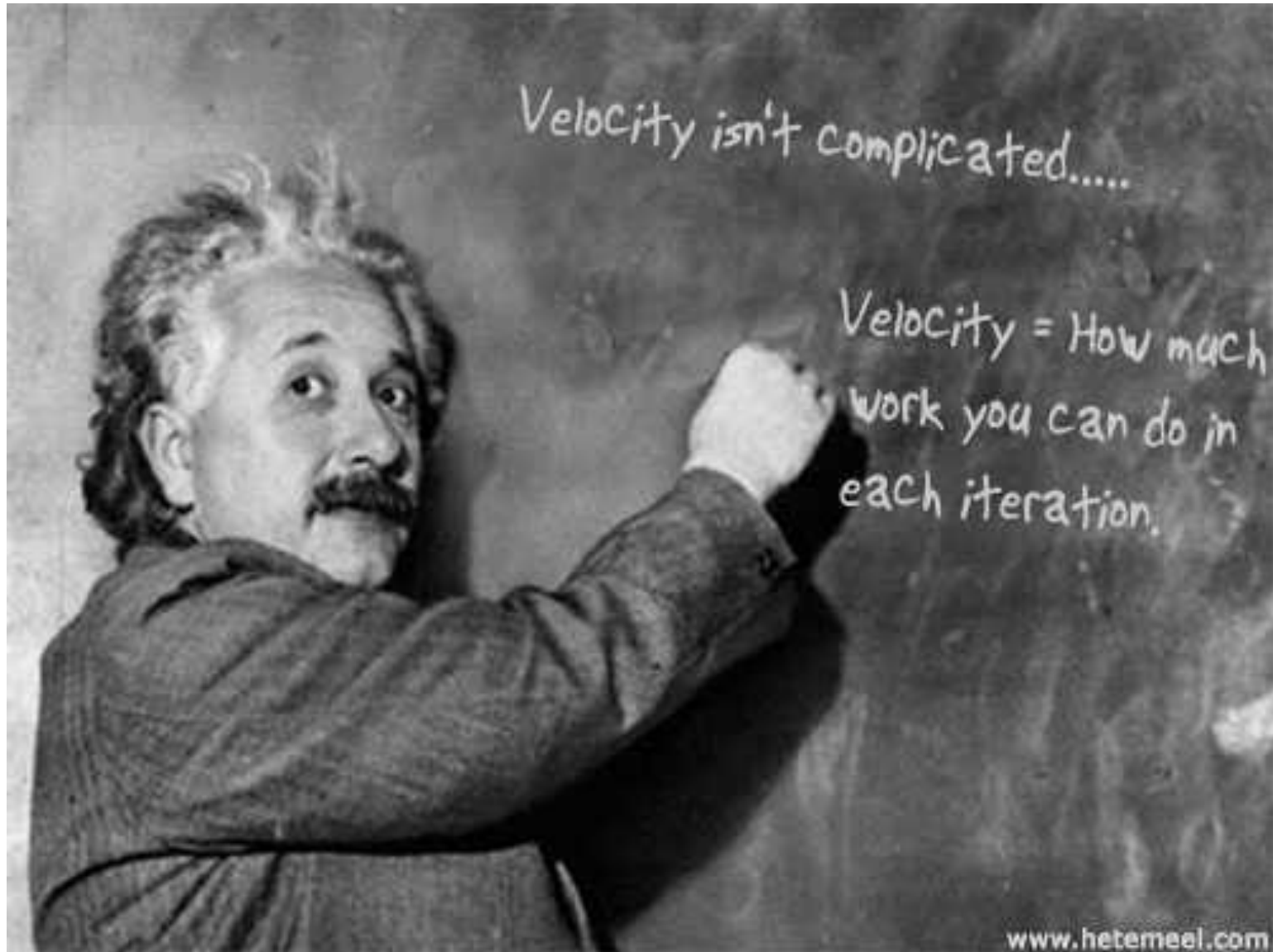


Retrospective

- **SM, Team**
- Lessons Learned, how to do better next time – KAIZEN!
- Sprint Weather
- Solve Conflicts / Correct dysfunctional behavior:
5 dysfunctions of the team
- Weather forecast
- ROTI
- vyžaduje koučování, kritické jsou soft skills u SM



Sprints & Velocity



www.hetemeel.com



Sprints & Velocity

- **Velocity** = SUMA SP **hotových** featur/US/tasků
 - (nebo hodin)
 - Nehotové tasky se vrací do BL – re-estimate?
- Př.
Release scope: 100SP
(estimated) Velocity: 25
Sprints required: 4
- **Scope** sprintu je **fixní** (a je jasný goal a forecast)
 - proč?



Plánování & Kapacita

- Čím je zatíženo časové plánování?
- „Časová“ kapacita člena teamu:
 - Celková: 100% (fulltime, „FT“)
 - *Plánovatelná: 80%*
 - SCRUM overhead (meetingy): 10%
 - Lookahead / preparation (BL grooming): 10%
 - Operativa – support: 10%
 - *Reálná kapacita na kreativní práci (plnění scope): 50%*
 - Ovšem meetingy, BL grooming atd. jsou „chtěné“ činnosti, ne „slack“! Bez nich nelze efektivně plnit scope.



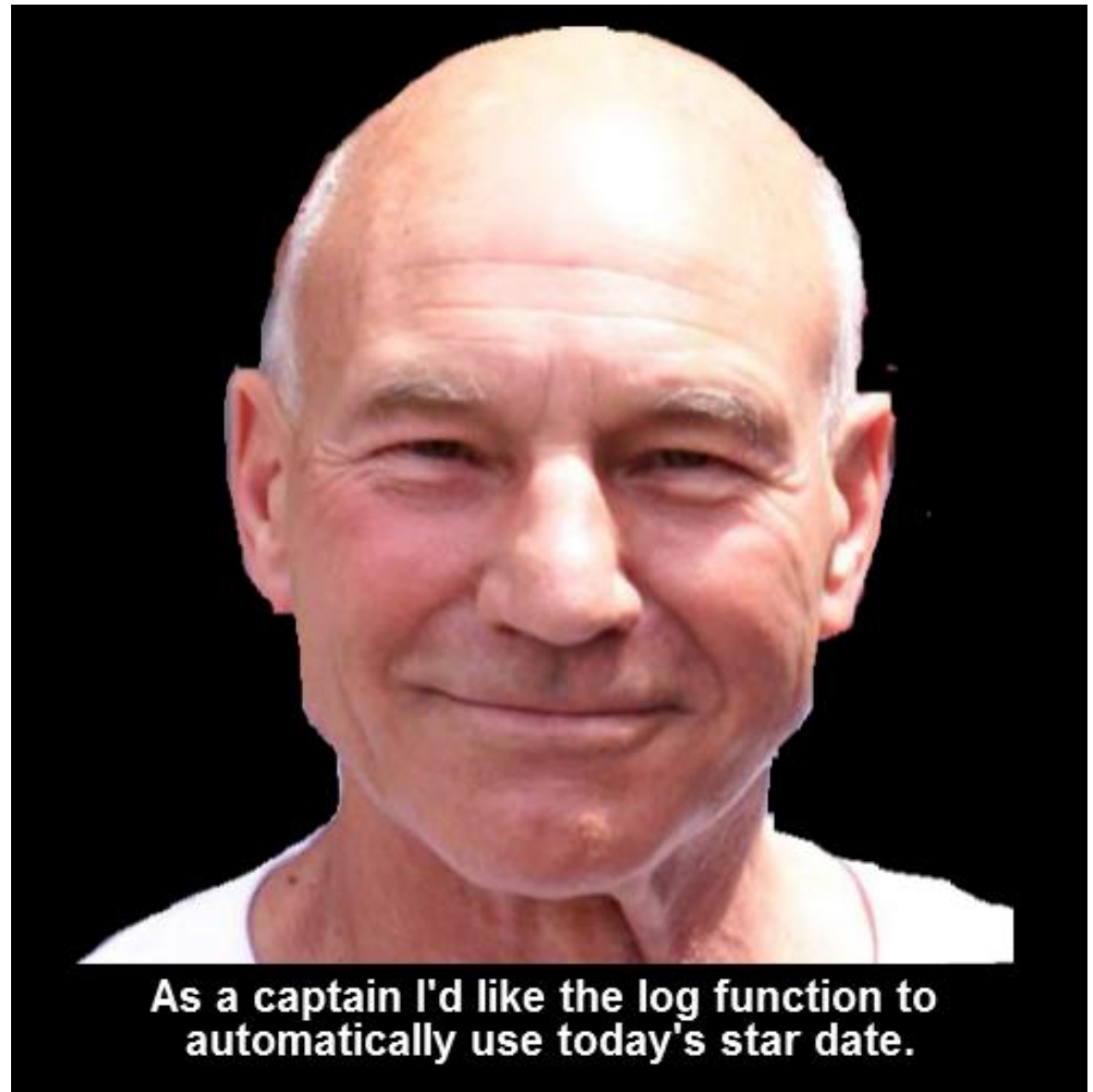
USER STORIES

Já, jakožto <Typ uživatele>,
bych chtěl, aby <Feature>
tak, že <Business value>.

Např: Já, jako *administrátor DBS* bych chtěl, abych mohl *hromadně měnit konfiguraci uživatelů systému a jejich oprávnění k přístupu do DB* tak, aby systém *převzal hodnoty z konfigurační šablony (CSV), ale zachoval si informaci o historickém nastavení a datu změn. Toto mi umožní provádět hromadné změny rychleji.*



Příklad z praxe



USER STORIES

- V přirozeném jazyku popsany požadavek
- **Kdo** – uživatelská role z pohledu businessu
(Admin systému, operátor helpdesku, pracovník frontline, ...)
 - Role podporují „hmatatelnost“
 - US oproti UC lépe znázorňují jak SW pomůže řešit reálný problém
- **Co** – business scénář, ne definice technolog. postupu řešení
- **Proč/Jak** – co je benefitem feature, přidanou hodnotou



USER STORIES

- **Krátké.** Nejlépe do dvou vět ;-)
 - „Placeholder“ pro pozdější komunikaci
- **Nemusí pokrývat** všechny **details**, není to dokumentace
- Popisují **přínos** nové featury **pro produkt**
 - Jednoznačná **přidaná hodnota**
- Popisují **akceptační kritéria**
 - Výchozí bod pro akceptační testy (ATDD)
- Mohou popisovat **omezení** (constraints)
- **Vertikální** (napříč technolog. vrstvami aplikace)



USER STORIES

[ID xxx-yyy]

[Title XYZ]

Size: N

As <User> I can <Feature/Function> so that <Business Value>

Acceptance Criteria

<User> can [operate/use] <Feature> so that [output] is [visible/complete/...].

Notes:

i.e. Does it need enhanced security?

Constraints:

Text field has to allow only numbers



US - I.N.V.E.S.T.

- Independent
- Negotiable
- Valuable
- Estimable
- Sized appropriately
- Testable



US - I.N.V.E.S.T.

- **Independent**
 - Umožní vyhnout se problémům při prioritizaci a odhadech
- **Negotiable**
 - Mohou se měnit v čase (dokud jsou v BL)
- **Valuable**
 - Představují přidanou hodnotu z hlediska businessu
- **Estimable**
 - Mají odhadnutelnou náročnost
 - Jasná akceptační kritéria
 - Pokud nejde odhadnout, nejde zařadit do SBL



US - I.N.V.E.S.T.

- **Sized appropriately**
 - US musí mít správnou granularitu: rozpad -> snazší odhady
 - Epics, velké US - se zvyšující se prioritou se rozpadnou na menší US
- **Testable**
 - Musí být jasná akceptační kritéria
 - TDD
 - ATDD
 - Performance, Stress, Failover testy mohou být samostatné US



US - Akceptační kritéria

- <User> can [*operate/use*] <Feature> so that [*output*] is [*visible/complete/...*]
- Umožní posoudit, zda je story implementována tak, jak PO / zákazník očekává
- Binární kritéria pro akceptaci US jako „hotové“
- Vodítka pro tvorbu akceptačních testů
 - Základní/kritické testy mohou být sepsány rovnou u US
- Podchycují možné nejasnosti ve formálním zadání US



US vs UC vs Doc.

- **US vs UC**

- UC je detailnější
- US se snáze dekomponují
- US není forma detailní dokumentace

- **US vs Dokumentace**

- US není / nenahrazuje dokumentaci
- Dokumentace je tak jako tak potřeba



Rozdělování USER STORIES

- Proč rozdělovat US:
 - Odhad přesahuje možnosti jednoho sprintu
 - Rozsah US neumožňuje odhad – příliš mnoho nejasností
- Po funkčních celcích (jednotlivé části CRUD např.)
- Po datech („Zákazník, lokace, zakázky,...“)
- Po rolích
- Komplexní US:
 - Story 1 – průzkum
 - Story 2 – implementace featur/-y



USER STORIES – časté chyby

- Popis úkolu/řešení, ne business scénář
- Horizontální rozdělení velkých stories na menší
- Závisající/provázané stories
- Příliš detailní – „goldplating“
- Špatně prioritizované
- Obsahují detailní popisy UI
- Chyby v Akceptačních kritériích



Dejte si s námi SCRUM

scRUM@cerebra.cz

CEREBRA s.r.o.

www.cerebra.cz

Pickova 1486/2

Praha Zbraslav 156 00

IČO: 27538702

