



# SCRUM

2010

Únor 2010

*Scrum: Vyvinuli a udržují Ken Schwaber a Jeff Sutherland*

# Poděkování

## Úvod

Scrum vychází z nejlepších zkušeností a zvyklostí, které odvětví přijalo za své a již desítky let je používá a prověřuje. Je tedy empirickou teorií procesu. Jim Coplien jednou k tomu Jeffovi poznamenal: „Scrum se bude líbit všem. Je to přesně to, co už teď děláme, kdykoliv jsme přitlačeni ke zdi“.

## Lidé

Z tisíců lidí, kteří přispěli ke vzniku Scrumu, bychom chtěli připomenout ty, kteří významně pomohli v prvních deseti letech. Jako první to byli Jeff Sutherland, který spolupracoval s Jeffem McKennou, a Ken Schwaber s Mikem Smithem a Chrisem Martinem. Scrum byl poprvé formálně představen a publikován na konferenci OOPSLA 1995. V průběhu následujících 5 let dále významně přispěli Mike Beedle a Martine Devos. A pak všichni další, bez pomoci kterých by Scrum nebyl zdokonalen do takové podoby, v jaké ho známe dnes.

## Historie

Historie Scrumu může být považována ve vývoji software za dlouhou. Pokud bychom chtěli připomenout místa, kde byl poprvé vyzkoušen a zdokonalen, musíme zmínit firmy Individual, Inc., Fidelity Investments a IDX (dnes GE Medical).

## Překlad

*Průvodce byl přeložen dle původní anglické verze, kterou poskytli Ken Schwaber a Jeff Sutherland. Na překladu se podíleli Vladimír Čech, Jiří Marschal, Zdeněk Měrka, Štěpán Roh, Jiří Rusňák a Michal Vallo. Jazykovou korekturu udělali Ondřej Chylík a Jiří Marschal. Práce koordinoval Michal Vallo.*

## Účel

Scrum se používá k vývoji komplexních produktů od počátku 90. let. Tento dokument popisuje, jak používat Scrum k vytváření produktů. Scrum není proces nebo technika na vytvoření produktu. Je to rámec, který umožňuje aplikovat různé procesy a techniky. Úlohou Scrumu je zviditelnit relativní efektivnost Vašich vývojových postupů tak, abyste se mohli zlepšovat, **a poskytuje Vám rámec pro vývoj komplexních produktů.**

## Teorie Scrumu

Scrum je založen na teorii řízení empirických procesů a využívá iterační a inkrementální přístup k optimalizaci předvídatelnosti a řízení rizika. Každá implementace řízení empirického procesu stojí na třech pilířích.

### Prvním pilířem je transparentnost

Transparentnost zajišťuje, aby všechny aspekty procesu, které mají vliv na výsledek, byli lidem, kteří výsledky řídí, stále viditelné. A nejenom že musí být viditelné, ale musí být také na první pohled srozumitelné. To znamená: jestliže někdo, kdo proces kontroluje, považuje něco hotové, musí se to plně shodovat s tím, jak je „hotovo“ definované.

### Druhým pilířem je kontrola

Různé aspekty procesu se musí kontrolovat tak často, aby se daly odhalit nepřijatelné odchylky v procesu. Frekvence kontroly se musí zvážit, protože akt kontroly způsobuje změny ve všech procesech. Hlavočlánkem nastává, když požadovaná frekvence kontroly převyšuje toleranci procesu ke kontrole. Toto naštěstí při vývoji software nebývá pravidlem. Dalším faktorem je zručnost a pozornost lidí, kteří provádějí kontrolu výsledků.

## Třetím pilířem je adaptace

Jestliže revizor na základě kontroly rozhodne, že jeden nebo více aspektů procesu jsou mimo přijatelné hranice a že výsledný produkt bude nepřijatelný (neakceptovatelný), pak tento revizor musí přizpůsobit proces nebo zpracovávané materiály. Změna musí být provedena co nejdříve, aby byla minimalizována budoucí odchylka.

Ve Scrumu jsou definovány tři body pro kontrolu a přizpůsobení (adaptaci). Prvním je Daily Scrum, denní meeting, který slouží ke kontrole postupu prací vůči cílům Sprintu a k provádění korekcí vedoucích k optimalizaci hodnoty, která bude vytvořena v následujícím pracovním dni. Dalším bodem jsou Sprint Review (hodnocení) a Sprint Planning (plánování), které slouží ke kontrole postupu prací vůči cílům Release (vydání nové verze) a provádění korekcí vedoucích k optimalizaci hodnot vytvářených v příštím Sprintu. Posledním bodem je Sprint Retrospective (retrospektiva). Tento meeting se využívá pro zhodnocení právě dokončeného Sprintu a ke stanovení, jaké změny učiní příští Sprint produktivnějším, více naplňujícím a zábavnějším.

## Obsah Scrumu

Rámec Scrumu se skládá ze skupiny **Scrum týmů** a s nimi spojených rolí, **časových rámců** (Time-Boxes), **artefaktů** a **pravidel**.

Scrum týmy jsou sestaveny tak, aby optimalizovaly flexibilitu a produktivitu. K dosažení tohoto cíle je využívána samoorganizace týmů, multi-funkčnost jejich členů a práce v iteracích. Každý Scrum tým obsahuje tyto tři role: 1) **Scrum Master** je zodpovědný za pochopení procesu a jeho dodržování, 2) **Vlastník produktu** (Product Owner) je zodpovědný za dosažení maximální hodnoty vytvořené Scrum týmem a 3) **Tým**, který práci realizuje. Tým se skládá z vývojářů se všemi potřebnými dovednostmi, které zajistí přetvoření požadavků vlastníka produktu v potenciálně nasaditelný inkrement produktu po ukončení Sprintu.

Scrum využívá časové rámce k vytvoření pravidelnosti. Časově ohraničené elementy Scrumu jsou **meeting pro plánování release** (Release Planning Meeting), **plánovací meeting pro Sprint** (Sprint Planning Meeting), **Sprint** (samotný vývoj), **Daily Scrum**, **Sprint Review** (zhodnocení), **Retrospektiva** (Sprint Retrospective).

Srdcem Scrumu je Sprint, což je iterace v délce jednoho měsíce či méně. Délka iterace je pevná po celou dobu vývoje. Všechny Sprints používají stejný Scrum Framework a vytvářejí přírůstek (inkrement) konečného produktu, který je potenciálně nasaditelný. Každý sprint začíná okamžitě po ukončení předchozího.

Scrum využívá čtyři hlavní artefakty. **Produktový backlog** (seznam požadavků na produkt) je prioritizovaný seznam všeho, co má produkt obsahovat. **Sprint Backlog** je seznam úkolů, které mají v rámci jednoho sprintu proměnit část produktového backlogu v přírůstek potenciálně nasaditelného produktu. Burndown slouží jako ukazatel zbývajících práce v čase. **Release Burndown** slouží k měření velikosti ještě nerealizovaných (zbývajících) položek produktového backlogu v čase v rámci plánu vydání (release). **Sprint Burndown** měří velikost zbývajících položek sprint backlogu v čase v rámci probíhajícího sprintu.

**Pravidla** spojují dohromady časové rámce, role a artefakty Scrumu. Jejich aplikace je popsána napříč celým tímto dokumentem. Například je ve Scrumu definováno pravidlo, že během Daily Scrum mohou mluvit pouze členové týmu, tj. lidé, kteří se zavázali přetvořit produktový backlog na přírůstek. Způsoby implementace Scrumu, které nejsou až tak pravidlem, jako spíše doporučením, jsou popsány v rámečcích „Tip“.

#### Tip

*Pokud nejsou pravidla stanovena, očekává se od „uživatelů“ Scrumu, že vymyslí, jak postupovat. Nezkoušejte nalézt dokonalé řešení, protože problém se často velmi rychle změní.*

*Místo toho něco zkuste a sledujte, jak to funguje. Přirozený mechanismus Scrumu „kontroluj a přizpůsobuj“ Vás už navede.*

## Role ve Scrumu

Scrum tým se skládá ze Scrum Mastera, vlastníka produktu a týmu. Členové týmu se ve Scrumu označují jako „prasata“. Vlastník produktu je „prase“ v souvislosti s produktovým backlogem (je za něj zodpovědný). Tým je „prase“ v souvislosti s prací, která je v rámci sprintu realizována. Scrum Master je „prase“ v souvislosti s procesy Scrumu (je zodpovědný za jejich pochopení týmem a jejich dodržování). Všichni ostatní jsou ve Scrumu označováni jako „kuřata“. „Kuřata“ nesmějí říkat „prasatům“, jak mají dělat svoji práci. Toto pojmenování vychází z následujícího příběhu:

*Kuře a prase byli spolu, když kuře řeklo:  
„Otevřeme si restauraci!“*

*Prase chvíli přemýšlelo, a pak se zeptalo:  
„Jak ji nazveme?“*

*Kuře odpovědělo: „Šunka a vejce.“*

*Na to prase odpoví: „Ne, děkuji, já půjdu  
s kůží na trh, ale ty se pouze zapojíš.“*

## Scrum Master

Scrum Master je zodpovědný za dodržování hodnot, postupů a pravidel Scrumu týmem. Scrum Master pomáhá týmu a organizaci v adaptaci Scrumu (přijetí za své). Scrum Master učí tým koučováním a vedením vyšší produktivě a tvorbě kvalitnějšího produktu. Scrum Master pomáhá týmu pochopit a využívat samoorganizaci a multi-funkčnost. Scrum Master pomáhá týmu dosahovat co nejlepších výsledků v prostředí, které ještě nemusí být zcela optimalizované na vývoj komplexních produktů. Tato aktivita

### Tip

*Scrum Master spolupracuje s klienty a managementem na vyhledání a jmenování vlastníka produktu. Scrum Master učí vlastníka produktu, jak má dělat svoji práci. Od vlastníka produktu se očekává, že ví, jak řídit a optimalizovat hodnotu pomocí Scrumu. Pokud tomu tak není, je to odpovědnost Scrum Mastera.*

### Tip

*Scrum Master může být členem týmu, např. vývojář, který řeší úlohy sprintu. Tento stav však často vede ke konfliktům, např. když musí volit mezi odstraňováním problémů a řešením úloh. Scrum Master nikdy nesmí být současně i vlastníkem produktu.*

Scrum Mastera se nazývá „odstraňování překážek“. Úloha Scrum Mastera v týmu je být podpůrným vedoucím.

## Vlastník produktu

Vlastník produktu je jedinou osobou zodpovědnou za správu produktového backlogu a za to, aby se úsilí vynaložené týmem co nejlépe zúročilo. Udržuje produktový backlog a zajišťuje, že je každému přístupný. Každý ví, které položky mají nejvyšší prioritu, takže každý ví, na čem se bude pracovat.

Vlastník produktu je jedna osoba, ne výbor. Výbory mohou vlastníkovému produktu radit či jej ovlivňovat, ale lidé, kteří chtějí změnit prioritu nějaké úlohy, musí přesvědčit vlastníka produktu.

Společnosti adoptující Scrum mohou zjistit, že tento systém ovlivňuje jejich metody pro nastavování priorit a požadavků v čase.

Aby vlastník produktu mohl uspět, je potřeba, aby každý v organizaci respektoval jeho rozhodnutí. Nikdo nemá dovoleno říkat týmu, aby pracoval na základě odlišně nastavených priorit, a týmům není dovoleno naslouchat nikomu, kdo tvrdí něco jiného. Rozhodnutí vlastníka produktu jsou zrcadlena v obsahu a prioritizaci produktového backlogu. To vyžaduje od vlastníka produktu, aby dělal to nejlepší, co je v jeho silách. Role vlastníka produktu je jak náročná, tak i naplňující.

### Tip

*Při komerčním vývoji může být vlastníkem produktu produktový manažer. Při interním vývoji to může být i manažer odpovědný za podnikovou funkci, která je vývojem automatizována.*

### Tip

*Vlastník produktu může být členem týmu, popř. i vyvíjet. Tato dodatečná zodpovědnost však může zasahovat do schopnosti vlastníka produktu pracovat se zájmovými stranami (stakeholders). Nicméně vlastník produktu nikdy nesmí být Scrum Masterem.*

## Tým

V rámci každého sprintu přeměňují týmy vývojářů produktový backlog v přírůstek potenciálně nasaditelné funkčnosti. Týmy jsou vícefunkční (cross-functional); členové týmu musí mít všechny dovednosti potřebné k vytvoření přírůstku funkčnosti. Členové týmu mají často specializované dovednosti, např. programování, kontrola kvality, obchodní analýza, architektura, návrh uživatelských rozhraní nebo návrh databází. Jenže dovednosti, které členové týmu sdílejí – tedy schopnost přistoupit k požadavku a přeměnit ho v použitelný produkt – jsou obvykle důležitější než ty ostatní. Lidé, kteří odmítají programovat, protože jsou architekti nebo návrháři, nejsou vhodní pro tým. Každý se musí zapojit, i kdyby to mělo znamenat naučit se něco nového nebo si připomenout něco staršího. V týmech nejsou žádné role a to bez výjimek. V týmech nejsou ani žádné podtýmy vyčleněné na nějaké určité oblasti jako je např. testování nebo obchodní analýza.

Týmy se organizují samy. Nikdo – ani Scrum Master – neříká týmu jak přeměňovat produktový backlog v přírůstek nasaditelné funkčnosti. Tým na to přijde sám. Každý člen týmu použije svoje znalosti a zkušenosti k vyřešení všech problémů. Výsledná synergie zlepšuje výkonnost a efektivitu celého týmu.

Optimální velikost týmu je sedm lidí, plus-minus dva. Pokud má tým méně než pět lidí, interakce se snižuje, což ve výsledku vede k nižšímu přírůstku produktivity. Navíc může tým během Sprintu narazit na omezení daná nedostatkem dovedností a selhat v dodání nasaditelné části produktu. Více jak devět členů jednoduše vyžaduje nadměrnou koordinaci. Velké týmy jsou příliš komplexní na to, aby se daly řídit empiricky. Nicméně jsme narazili na několik úspěšných týmů, které se vymykaly horní i dolní hranici počtu členů. Role vlastníka produktu a Scrum Mastera se nepočítají, pakliže taktéž nejsou „prasátky“ pracujícími na úlohách ze sprint backlogu.

Složení týmu se po skončení sprintu může změnit. Pokaždé, když se



tak stane, vytrácí se produktivita nabytá při týmové samoorganizaci. Při změnách v týmu je třeba velké opatrnosti.

## Časově ohraničené činnosti

Časově ohraničenými činnostmi ve Scrumu jsou schůze pro **naplánování release** (Release Planning Meeting), **Sprint, plánování sprintu** (Sprint Planning Meeting), **Review** (hodnocení sprintu), **Retrospektiva** (Sprint Retrospective) a **Daily Scrum** (denní schůze).

### Meeting pro plánování vydání (release)

Účelem plánování vydání je ustanovit plán a cíle, kterým scrum týmy a zbytek organizace rozumějí a mohou je dále komunikovat. Plánování vydání odpovídá na otázky typu: „Jak přeměnit vizi v úspěšný produkt tím nejlepším způsobem? Jak uspokojit či překonat očekávání zákazníka a návratnost investic?“ Plán vydání obsahuje cíl vydání, nejdůležitější části produktového backlogu, největší rizika a přehled vlastností a funkčností, které budou součástí vydání. Také stanovuje pravděpodobné datum dokončení a předpokládané náklady, pakliže se nic nezmění. Organizace může zkoumat postup a měnit plán vydání mezi jednotlivými sprinty.

Plánování vydání je naprosto volitelné. Pakliže Scrum týmy začnou s prací bez této schůze, absence jeho artefaktů se časem projeví jako překážka, kterou je nutno odstranit. Práce na odstranění překážky se zařadí do produktového backlogu.

Produkty se za pomoci Scrumu tvoří iterativně, v každém sprintu se vytvoří přírůstek produktu počínaje tím nejhodnotnějším a nejrizikovějším. Výstupem dalších sprintů jsou další přírůstky. Každý přírůstek je potenciálně nasaditelný kus celého produktu. Když se vytvoří dost přírůstků na to, aby měl produkt dostatečnou hodnotu pro své investory, je produkt vydán.

Většina organizací již má proces plánování release a ve většině

těchto procesů se plánuje na počátku a v průběhu se již plán nemění. Při plánování release podle Scrumu se určuje celkový cíl a možné výsledky, což typicky zabere ne více než 15-20% času, který se spotřebuje při tvorbě tradičního plánu vydání. Avšak plánuje se i při každém review meetingu a plánování sprintu, jakož i při každém Daily Scrumu. Celkově se při plánování release podle Scrumu stráví zřejmě více času než při tradičním způsobu.

Plánování release vyžaduje odhadování a prioritizaci produktového backlogu. K tomu slouží množství technik, které leží mimo rámec Scrumu, ale dokáží být velmi užitečné.

## Sprint

Sprint je iterace. Sprints jsou časově ohraničené. Scrum Master během sprintu dohlíží na to, aby se nezměnilo nic, co by ohrozilo cíl sprintu. Jak složení týmu, tak kvalitativní měřítka se během sprintu nemění. Sprint se skládá z plánování sprintu, fáze vývoje, review meetingu a retrospektivy. Sprints následují jeden za druhým a bez přerušení na sebe navazují.

### Tip

*Má-li tým pocit, že přecenil své síly, pak se dohodne s vlastníkem produktu na odstranění či zredukování části produktového backlogu vybraného pro daný Sprint.*

*Má-li tým pocit, že své síly podcenil, pak může spolu s vlastníkem produktu vybrat dodatečné části produktového backlogu.*

Projekt se používá jako nástroj k dosažení něčeho; při vývoji software je to vybudování produktu nebo systému. Každý projekt se skládá ze zadání, co se vybuduje, z plánu, jak se to vybuduje, práce vynaložené v souladu s plánem a výsledného produktu. Každý projekt má horizont, tedy časové ohraničení, v rámci kterého je plán užitečný. Je-li horizont příliš dlouhý, může se změnit zadání, objeví se příliš mnoho nových proměnných, riziko může být příliš vysoké atd. Scrum je rámec pro projekt, jehož horizont není delší než jeden měsíc a který je natolik složitý, že by delší horizont byl již příliš

riskantní. Předvídatelnost projektu musí být řízená alespoň na měsíční bázi a riziko, že se projekt vymkne kontrole nebo se stane nepředvídatelným, se musí omezit minimálně jednou měsíčně.

#### Tip

*Začíná-li tým se Scrumem, pak mu dvoutýdenní Sprints dovolí učit se bez rizika upadnutí do nejistoty. Sprints takovéto délky je možno synchronizovat s ostatními týmy spojením dvou přírůstků v jeden.*

Před vypršením časového ohraničení je možno sprint zrušit. Pouze vlastník produktu může sprint zrušit, ačkoliv tak může učinit pod vlivem některé zájmové strany, týmu či Scrum Mastera. Za jakých okolností je možné sprint zrušit? Vedení může být nuceno zrušit sprint, jestliže se cíl sprintu stane zastaralým. To se může stát, jestliže společnost změní své směřování, změní se podmínky na trhu nebo technologie. Všeobecně se má sprint zrušit, jestliže už za daných okolností nemá význam. Avšak díky krátkému trvání sprintu to má málokdy smysl.

Po zrušení sprintu projdou všechny dokončené a „hotové“ části produktového backlogu hodnocením a jsou akceptovány, jestliže jsou potenciálně nasaditelným přírůstkem. Všechny zbylé části produktového backlogu jsou vráceny zpět do produktového backlogu s původními odhady. Veškerá práce na nich udělaná se považuje za ztracenou. Ukončení sprintu stojí zdroje, protože každý musí znovu projít plánovací schůzi před začátkem dalšího sprintu. Pro tým jsou ukončení sprintu často traumatická a velmi neobvyklá.

## Meeting pro plánování sprintu

Během meetingu pro plánování sprintu se plánuje iterace. Pro jednoměsíční sprint je časově omezen na osm hodin. Pro kratší sprints je to poměrně méně (například pro dvou týdenní sprint by to měly být čtyři hodiny). Meeting pro plánování sprintu se skládá ze dvou částí. V první části je rozhodnuto, co se ve sprintu bude dělat. Ve druhé části (pro měsíční sprint časově omezené na čtyři hodiny)

tým řeší, jak bude postupovat, aby přidal danou funkcionalitu do přírůstku produktu (product increment) během sprintu.

Meeting pro plánování sprintu má dvě části: část „Co?“ a část „Jak?“. Některé Scrum týmy tyto části spojují. V první části se Scrum tým zabývá otázkou „Co?“. Vlastník produktu představuje týmu požadavky z produktového backlogu s nejvyšší prioritou. Pracují společně, aby se dohodli, které funkcionality budou vyvinuty během následujícího sprintu. Vstupem meetingu jsou produktový backlog, poslední přírůstek produktu, kapacita týmu a předcházející výkonnost týmu. Část backlogu, kterou tým vybírá do sprintu je výhradně na týmu. Pouze tým může stanovit, co dokáže udělat v následujícím sprintu.

Vybráním produktového backlogu je vytvořen cíl sprintu. Cíl sprintu je záměr, který bude dosažen prostřednictvím realizace produktového backlogu. Je to přehled, který ukazuje týmu směr a říká proč je přírůstek vytvářen. Cíl sprintu je podmnožina cíle releasu.

Důvodem, proč mít cíl sprintu, je dát týmu určitý manévrovací prostor v rámci funkcionality. Například cílem pro nadcházející sprint může také být: „Automatizovat funkcionalitu modifikace uživatelského účtu prostřednictvím bezpečného, zotavitelného transakčního middleware.“ Při práci tým udržuje tento cíl v paměti. Aby splnil cíl, implementuje funkcionalitu a technologii. Pokud se ukáže, že práce bude těžší, než tým předpokládal, pak tým jedná s vlastníkem produktu a funkcionalitu implementuje pouze částečně.

Ve druhé části meetingu pro plánování sprintu se tým zabývá otázkou „Jak?“. Během této druhé části meetingu (pro měsíční sprint časově omezené na čtyři hodiny) tým zvažuje, jakým způsobem bude řešit úkoly z produktového backlogu, vybrané během plánování sprintu („Co?“) do výsledného přírůstku. Tým obvykle začíná návrhem práce. Při návrhu tým identifikuje úkoly. Tyto úkoly jsou drobnými částmi práce, potřebnými k převedení produktového backlogu na fungující software. Úkoly by měly být děleny tak, aby

mohly být udělány za méně než jeden den. Tento seznam úkolů se nazývá Sprint Backlog. Tým se samo-organizuje za účelem vykonání práce ve sprint backlogu, buď během meetingu pro plánování sprintu nebo v průběhu sprintu.

Vlastník produktu je přítomný během druhé části meetingu pro plánování sprintu, aby vyjasnil produktový backlog a aby pomohl dělat kompromisy. Pokud tým zjistí, že má příliš mnoho nebo příliš málo práce, může znovu vyjednat produktový backlog s vlastníkem produktu. Tým může také přizvat k účasti další lidi, aby poskytli technickou nebo odbornou radu. Na této schůzce si nový tým často poprvé uvědomí, že buď uspěje nebo neuspěje jako tým, ne jako jednotlivci. Tým si uvědomí, že musí spoléhat na sebe. Když si to uvědomí, začne se samo-organizovat, aby získal charakteristiku a chování skutečného týmu.

#### Tip

*Obvykle se na meetingu pro plánování sprintu vymyslí pouze 60-70% z celého Sprint Backlogu. Zbytek se připraví pro pozdější rozpracování anebo si ponechá větší odhady, které se dekomponují později v průběhu sprintu.*

## Sprint Review (hodnocení)

Na konci sprintu se koná Sprint Review meeting. Je to meeting pro měsíční sprint omezený čtyřmi hodinami. Pro sprinty s kratším trváním, přidejte poměrně méně času (například dvoutýdenní sprint by měl dvouhodinové sprint review). Během sprint review Scrum tým a zástupci zájmových stran (stakeholderi) jednají o tom, co bylo právě uděláno. Na tomto základě a změnách v produktovém backlogu během sprintu řeší, co jsou další věci, které by měly být udělány. Je to neformální meeting s předvedením funkcionality, určený k posílení dohody o tom, co dělat dál.

Meeting zahrnuje alespoň následující prvky. Vlastník produktu odliší, co bylo a co nebylo uděláno. Tým prodiskutuje, co šlo během sprintu

dobře, na jaké problémy narazil a jak tyto problémy řešil. Potom tým předvede práci, která byla udělána, a odpovídá na otázky. Následně vlastník produktu probírá produktový backlog tak, jak je. Dále určí pravděpodobná data dokončení za předpokladu různých výkonností. Celá skupina pak jedná o tom, co bylo předvedeno a co to znamená s ohledem na další práci. Sprint Review poskytuje cenný vstup pro následný meeting pro plánování sprintu.

## Retrospektiva

Po Sprint Review a před dalším meetingem pro plánování sprintu uskuteční Scrum tým ještě jeden meeting zvaný retrospektiva. Jedná se o meeting časově omezený pro měsíční sprint na tři hodiny (pro sprinty s kratším trváním, přidejte poměrně méně času). Na tomto meetingu Scrum Master povzbuzuje tým, aby v mezích Scrum rámce přizpůsobil svůj vývojový proces tak, aby byl pro příští sprint efektivnější a příjemnější. Techniky užitečné při retrospektivě popisuje mnoho knih.

Účelem retrospektivy je prověřit, jak fungoval poslední sprint s ohledem na lidi, vztahy, procesy a nástroje. Prověření by mělo identifikovat nejdůležitější body, které se podařily a také ty, které mohou průběh sprintu zlepšit, jestliže budou dělány jinak. To zahrnuje složení Scrum Týmu, organizaci meetingů, nástroje, definici „hotovo“, metody komunikace a procesy pro transformaci položek produktového backlogu do něčeho „hotového“. Na konci retrospektivy by měl mít Scrum tým identifikovaná proveditelná opatření, která realizuje v příštím sprintu za účelem zlepšení. Tyto změny se stanou adaptací empirické kontroly.

## Daily Scrum

Každý tým se denně setkává na 15-minutovém kontrolním a adaptačním meetingu nazývaném Daily Scrum. Daily Scrum se po celou dobu sprintu koná ve stejný čas a na stejném místě. Během meetingu každý člen týmu jasně popíše:

1. Co dokončil od posledního meetingu;
2. Co hodlá dělat před příštím meetingem;
3. A jaké překážky mu v tom brání.

Daily Scrum zlepšuje komunikaci, eliminuje další meetingy, identifikuje a odstraňuje překážky ve vývoji, zdůrazňuje a podporuje rychlá rozhodnutí a zlepšuje úroveň znalostí každého člena týmu o projektu.

Scrum Master zajišťuje, aby se tento meeting konal. Tým je zodpovědný za vedení Daily Scrumu. Scrum Master učí tým udržet Daily Scrum krátký prosazováním pravidel a zajištěním, že lidé mluví stručně. Scrum Master také prosazuje pravidlo, že „kuřatům“ není dovoleno mluvit nebo jinak zasahovat do Daily Scrumu.

Daily Scrum není status meeting. Není pro každého, ale pro lidi převádějící položky produktového backlogu do přírůstku (tým). Tým se zavázal k cíli sprintu a k vybraným položkám produktového backlogu. Daily Scrum je kontrola postupu k cíli sprintu (tři otázky). Někdy se konají navazující schůzky, aby se provedlo přizpůsobení nadcházející práce ve sprintu. Záměrem je optimalizovat pravděpodobnost, že tým splní svůj cíl. Toto je klíčový kontrolní a přizpůsobovací meeting v empirickém procesu Scrumu.

## Artefakty Scrumu

Artefakty Scrumu zahrnují produktový backlog, release burndown, sprint backlog a sprint burndown.

### Produktový backlog a release burndown

Požadavky na produkt, který tým(y) vyvíjí, jsou uvedeny v produktovém backlogu. Vlastník produktu je zodpovědný za obsah, dostupnost a prioritizaci backlogu. Produktový backlog není nikdy úplný. Úvodní část k vývoji pouze rozplánovává zpočátku známé a dobře pochopené požadavky. Produktový backlog se vyvíjí tak, jak se vyvíjí produkt a prostředí, ve kterém bude používán. Produktový backlog je dynamický, stále se mění, aby obsahoval to, co produkt potřebuje, aby byl vhodný, konkurenceschopný a užitečný. Dokud existuje produkt, existuje produktový backlog.

Produktový backlog představuje všechno nezbytné k vývoji a nasazení úspěšného produktu. Je to seznam všech vlastností, funkcí, technologií, rozšíření a oprav chyb, které představují všechny změny, které budou provedeny v produktu

#### Tip

*Položky produktového backlogu jsou většinou User Stories. Lze použít i případy užití (Use Case), ty jsou ale vhodnější při vývoji životně nebo mission-critical software.*

v příštích vydáních. Položky produktového backlogu mají popis, prioritu a odhad. Priorita je řízena rizikem, hodnotou a nezbytností. Ke stanovení těchto atributů slouží různé techniky.

Produktový backlog je seřazen podle priority. Položky produktového backlogu s nejvyšší prioritou řídí okamžité vývojové aktivity. Čím vyšší priorita, tím je položka naléhavější, tím více o ní bylo přemýšleno a tím více je shoda ohledně její hodnoty. Položky backlogu s vyšší prioritou jsou jasnější a mají více detailních informací než položky backlogu s nižší prioritou. Lepší odhady jsou založeny na větší jasnosti a více detailech. Čím nižší priorita, tím méně detailů, až je téměř nemožné položky určit.



Jak je produkt používán, jeho hodnota roste a trh poskytuje zpětnou vazbu, produktový backlog se vyvíjí do rozsáhlejšího a podrobnějšího seznamu. Požadavky se nikdy nepřestanou měnit. Produktový backlog je živý dokument. Změny v byznys požadavcích, podmínkách trhu, technologiích a personálním zajištění způsobují změny v produktovém backlogu. K minimalizaci předělávání jsou detailně rozpracovány pouze položky s nejvyšší prioritou. Položky produktového backlogu, které zaměstnají týmy na několik příštích sprintů, jsou rozpracovány a rozděleny tak, že každá položka může být dokončena během jednoho sprintu.

#### Tip

*Scrum Tým často stráví 10 % každého Sprintu přípravou produktového backlogu, aby naplnil jeho v textu uvedenou definici. Když je připraven na tuto úroveň granularity, položky na vrchu produktového backlogu (nejvyšší priorita, největší hodnota) jsou rozloženy tak, aby se vešly do jednoho sprintu. Tyto položky byly analyzovány a důkladně promyšleny během přípravného procesu. Když se koná plánování sprintu, položky produktového backlogu s nejvyšší prioritou jsou dobře srozumitelné a jednoduše vybratelné.*

#### Tip

*Akceptační testy se často používají jako další atribut položky produktového backlogu. Často mohou nahradit podrobný textový popis testovacím scénářem, který říká, co položka produktového backlogu musí splňovat, aby mohla být označena za dokončenou.*

Více Scrum týmů často pracuje společně na stejném produktu. K popisu připravované práce na produktu se používá pouze jeden produktový backlog. Pak se použije atribut produktového backlogu, který seskupuje položky. Seskupování se může dít dle množiny vlastností, technologie nebo architektury a je často používáno jako způsob organizace práce Scrum týmem.

Release Burndown graf zaznamenává součet zbývajících odhadovaného úsilí v čase. Odhadované úsilí je v libovolných jednotkách práce, na kterých se Scrum tým a organizace dohodly. Jednotkami času jsou obvykle sprinty.

Odhady pro položky produktového backlogu jsou stanovovány poprvé během plánování release a poté kdykoliv, když vzniknou nové položky. Během přípravy produktového backlogu jsou odhady přezkoumány a revidovány. Nicméně mohou být aktualizovány kdykoliv. Tým je odpovědný za všechny odhady. Vlastník produktu může ovlivnit tým tím, že mu pomůže porozumět a vybrat kompromisy, ale finální odhad je vytvořen týmem. Vlastník produktu neustále udržuje produktový backlog a Release Backlog Burndown aktualizovaný. Linie trendu se kreslí na základě změn zbývajících práce.

#### Tip

*V některých organizacích je do backlogu přidáváno více práce než je dokončováno. To může vytvořit linii trendu, která je stagnující nebo dokonce stoupající vzhůru. Pro kompenzaci tohoto jevu a udržení transparentnosti, může být vytvořena nová úroveň, když se práce přidává nebo ubírá. Úroveň by měla přidávat nebo ubírat pouze významné změny a měla by být dobře dokumentována.*

#### Tip

*Během prvních dvou až tří sprintů může být linie trendu nespolehlivá, to pokud tým dříve nepracoval dohromady, nezná dobře produkt nebo nerozumí dobře vybrané technologii.*

## Sprint Backlog a Sprint Burndown

Sprint backlog je složen z úkolů, jejichž plněním tým transformuje položky produktového backlogu do „hotového“ přírůstku. Mnohé z těchto úkolů se vytvářejí v průběhu meetingu pro plánování sprintu. Jedná se o veškerou činnost, kterou tým identifikuje jako nezbytnou pro splnění cíle daného sprintu. Jednotlivé úkoly ze sprint backlogu musí být dekomponovány, a to na takovou úroveň, aby bylo možné sledovat plnění úkolu na denní bázi, tj. na Daily Scrumu. Obvyklá pracnost úkolu ze sprint backlogu je jeden den nebo menší.

Sprint backlog se formuje během sprintu a tým ho upravuje po celou dobu sprintu. Jakmile dojde na dílčí úkoly, tým může zjistit, že je

jich potřeba více, nebo naopak méně, nebo že vybrané úkoly zaberou více nebo méně času, než se původně odhadovalo. Pokud je požadována další práce, tým ji přidá do sprint backlogu. Podle toho, jak je na úkolech pracováno, nebo když jsou dokončeny, provádí se aktualizace odhadu jejich zbývajících pracností. Pokud se některý úkol ukáže jako zbytečný, je z backlogu odstraněn. Jedině tým může změnit sprint backlog v průběhu sprintu. Jedině tým může měnit obsah jednotlivých úkolů nebo odhady pracností. Sprint backlog tak představuje jasný, transparentní a aktuální obraz práce, kterou chce tým v průběhu sprintu dokončit. Sprint backlog je jen a výlučně záležitostí týmu.

Sprint backlog burndown je graf, který ukazuje množství práce, která v průběhu času zbývá k dokončení sprintu. K vytvoření tohoto grafu potřebujete každý den v průběhu sprintu určit, kolik práce k dokončení backlogu ještě zbývá, což je součet odhadů pracností za všechny úkoly tohoto backlogu. Sledujte hodnoty těchto součtů za každý den a zaznamenávejte je do grafu. Spojením jednotlivých bodů grafu získáte křivku ukazující zbývajících práci v čase. Tato křivka může týmu pomoci řídit postup dokončování práce na sprintu. Trvání není ve Scrumu důležité. Středem pozornosti jsou pouze tyto proměnné: práce a datum.

Jedno z pravidel Scrumu se vztahuje k účelu každého sprintu – a tím je vytvořit takový přírůstek funkcionality, který je potenciálně nasaditelný. Jinak řečeno, je úzce spojen s pracovní definicí pojmu „hotovo“.

#### Tip

*Pokud je to možné, nakreslete graf ručně na velký list papíru (nebo tabuli) a vystavte ho na pracovišti týmu tak, aby byl na očích. Tým se raději dívá na velký, viditelný graf na zdi, než aby si prohlížel takový graf v Excelu nebo jiném nástroji.*

## Hotovo

Scrum požaduje od týmů, aby v každém sprintu vytvořily přírůstek funkcionality produktu. Tento přírůstek musí být možno nasadit, když se vlastník produktu rozhodne, že chce tuto funkcionalitu okamžitě zprovoznit. Aby tomu tak bylo, musí přírůstek představovat kompletní část produktu. Musí být postě „hotov“. Každý přírůstek musí být kompatibilní se všemi předchozími přírůstky, důkladně otestován tak, aby se zabezpečilo, že vše bude společně fungovat.

Při vývoji produktů může být prohlášení „je to hotové“ chápáno různými lidmi různě. Pro někoho to může znamenat, že produkt je přinejmenším dobře naprogramován, refaktorován, otestován jednotkovými testy, zkompilován a akceptován. Někdo jiný si může myslet, že byl jen dokončen kód. Pokud všichni nevědí, jak je definováno „hotovo“, zbývající dva pilíře empirického řízení procesů nefungují. Pokud někdo prohlásí něco za „hotové“, všichni musí rozumět tomu, co „hotové“ znamená.

„Hotové“ znamená to, co tým míní v okamžiku, kdy potvrzuje závazek dokončit určitý úkol v rámci sprintu. Některé produkty neobsahují dokumentaci, to znamená, že také definice „hotovo“ neobsahuje dokumentaci. Kompletně „hotový“ přírůstek pokrývá vše – od analýzy, návrhu, refaktoringu, programování, dokumentace a testování až po všechny položky backlogu tohoto přírůstku. Testování zahrnuje unit testy, systémové, uživatelské a regresní testy, stejně jako nefunkcionální testy typu výkonost, stabilita, bezpečnost a integrace. „Hotovo“ zahrnuje také např. internacionalizaci. Ne všechny týmy jsou schopny zahrnout do své definice „hotovo“ vše, co je k implementaci potřeba. To musí být jasné každému vlastníkovi produktu. Tuto zbývající práci je třeba dokončit předtím, než se produkt nasadí a začne používat.

### Tip

*Nedokončená práce se často akumuluje v produktovém backlogu pod stejnojmennou položkou nebo pod položkou „Implementační práce“. Když se tato položka udržuje aktuální, zůstává product backlog burndown přesnější a zachovává si tak vypovídací schopnost.*

## ZÁVĚREČNÉ ZAMYŠLENÍ

Některé organizace nejsou schopné dodat kompletní přírůstek v rámci jednoho sprintu. Možná ještě nemusejí mít vytvořenou infrastrukturu pro automatické testování tak aby provedly veškeré testování. V těchto případech jsou přírůstky rozděleny do dvou kategorií na "hotovou" práci a "nehotovou" práci. "Nehotová" práce je část z přírůstků, která musí být dodělána později. Vlastník produktu přesně ví, co bude kontrolovat na konci sprintu, protože přírůstek odpovídá definici "hotovo" a vlastník produktu této definici rozumí. "Nehotová" práce je přidána zpět jako položka produktového backlogu s označením "nedodělaná práce" (undone work) tato se pak hromadí a přesně odráží v Release Burndown grafu. Tato technika vnáší transparentnost do procesu tvorby vydání. Kontrola a přizpůsobení ve Sprint Review je jen tak přesná, jak přesná je tato transparentnost.

Například jestliže není tým schopen provést testování výkonnosti, regresí, stability bezpečnosti a integrační testování pro každou položku v produktovém backlogu, pak je propočítán proporcionální podíl práce oproti práci, co může být udělána (analýza, návrh, refaktoring, programování, dokumentace, jednotkové a uživatelské testování). Řekněme, že poměr práce "hotovo"/"nehotovo" je šest jednotek ku čtyřem jednotkám. Pokud tým dokončí šest položek z produktového backlogu (tým odhaduje na základě toho, že ví, co a jak má udělat), přidá po skončení zpět do produktového backlogu čtyři položky jako "nedodělaná práce".

Sprint za sprintem se "nehotová" práce z každého přírůstku kumuluje a musí jí být věnována pozornost před vydáním produktu. Tato práce je kumulována lineárně, ačkoliv ve skutečnosti má povahu nějakého druhu exponenciální kumulace, která je závislá na charakteru každé organizace. Do konce každého vydání jsou přidávány „sprinty před vydáním“ tak, aby byla dokončena tato "nehotová" práce. Počet sprintů je neodhadnutelný jestliže kumulace "nehotové" práce má nelineární charakter.