

Semestrální práce ke kurzu 4IT421 Zlepšování procesů budování IS	
Semestr	Zimní semestr 2016/2017
Autoři – jméno, příjmení, xname	Tomáš Fenyk, xfent00 Michal Hron, xhrom33 Petr Janata, xjanp107
Téma	User Stories are Placeholders for Requirements
Datum odevzdání	18. prosince 2016

Abstrakt

Tato práce pojednává o důležitosti správného zpracování User Stories v rámci agilních metodik vývoje softwaru. Nejprve práce seznamuje čtenáře s požadavky a User Stories při vývoji softwaru jako takovými a následně je krátce srovnává v rámci rigorózních a agilních metodik. Poté již probíhá zaměření na důležitost správného zpracování User Stories, a to zejména v rámci agilní metodiky Scrum. Závěrem je zmíněn také životní cyklus User Story.

Klíčová slova

Agilní metodiky, Scrum, User Story, uživatelské příběhy, rigorózní metodiky, vývoj softwaru, životní cyklus User Story

Obsah

1. Úvod	1
2. Požadavky v softwarovém projektu	2
2.1 Funkční požadavky	3
2.2 Nefunkční požadavky	3
3. User Story v agilním vývoji	3
4. Srovnání pohledu agilních a rigorózních metodik na požadavky	4
4.1 Rigorózní metodiky	4
4.2 Agilní metodiky	5
5. User Stories jako základ pro požadavky v agilních metodikách	5
5.1 Zpracování User Stories v rámci Sprint Planning Meetingu metodiky Scrum	6
5.1.1 Analýza požadavků v rámci Planning Meetingů	6
5.1.2 Použití "TWBDW" pro analýzu aneb akceptační kritéria u User Stories	7
5.1.3 Spolupráce při rozboru User Stories	8
5.1.4 Přípravenost User Story	8
5.1.5 Otázky při jednotlivých fázích zpracování User Story v rámci Sprint Planning Meetingu	9
5.2 Životní cyklus požadavku v agilní metodice Scrum	10
6. Závěr	12
Zdroje	13

1.Úvod

Cílem této semestrální práce je přiblížit a zdůraznit důležitost správného zpracování User Stories v rámci agilních metodik - práce se bude zabývat významem User Stories, jejich přínosy a nakonec životním cyklem v rámci vývoje. Kromě krátkého seznámení se s User Stories a požadavky tvoří nedílnou část této práce zaměřením na způsob analýzy požadavků v rámci Sprint Planning Meetingu agilní metodiky Scrum, kde bude využito poznatků z článku User Stories Are Placeholders for Requirements od Russe Lewise [Lewis 2016].

Požadavky hrají klíčovou roli při zpracovávání jakéhokoliv projektu. Při špatném porozumění požadavkům může dojít k negativním následkům v projektu, kterým je možno se vyhnout, pokud jsou aplikovány správné postupy. V této práci jsou pak vybrány některé tipy, které by mohly napomoci efektivnímu zpracování User Stories a z nich odvozených požadavků.

2. Požadavky v softwarovém projektu

Wiegers [2008, strana 28-29] říká, že žádná univerzální definice požadavku neexistuje. Nicméně například poradce Brian Lawrence definuje požadavek jako *“cokoli, co ovlivňuje rozhodování při návrhu”*.

Požadavek lze tedy definovat jako potřebu, o kterou usilujeme. Nemusí se to týkat pouze softwaru. Každý člověk se za běžného života denně setkává s požadavky týkajícími se těch nejběžnějších věcí. Na snídani si člověk objedná kávu - usiluje o kávu, jednoduché - ve skutečnosti ale očekává kávu v hrníčku, sklenici vody, cukr, vysokou teplotu kávy a v neposlední řadě i úsměv od obsluhy.

Zatímco příklad výše se může zdát poněkud triviální, v případě tvorby softwaru se požadavky až tolik neliší. Problém je vyšší mírou v tom, že mezi lidmi může vzniknout komunikační či znalostní bariéra, která vede ke špatnému pochopení požadavků. Mnohdy spolu musí komunikovat lidé, kteří rozumějí zcela odlišným věcem a co je důležitější příliš nerozumí tomu, co dělá ten druhý.

Dle dvou zahraničních výzkumů [Wiegers 2008, strana 26] z devadesátých let chyby způsobené při analýze požadavků jsou zodpovědné za 40 - 60% všech chyb. Velké množství problémů se softwarem je spojeno s nedostatky při:

- sběru požadavků
- dokumentaci požadavků
- schvalování a úpravách požadavků
- a neformálním získáváním informací - problém s mlčky předpokládanými, zamlčovanými nebo chybně definovanými požadavky

Jestliže totiž je kritická fáze práce s požadavky úspěšně zvládnuta, tak je dodán kvalitní zdravý systém a výsledkem jsou i spokojení zákazníci i vývojáři. Zatímco, když se tyto body pojmu se slabou důsledností, tak výsledkem může být mnoho nedorozumění a konfliktů, které v konečném důsledku povedou k neúspěchu projektu. To je ke zvýšenému nesouladu s plánem co se týče takzvaného magického trojúhelníku - to je dojde k výraznému posunu v 1 nebo více dimenzích kvality - rozsah / čas / cena. [Wiegers 2008, strana 26]

Požadavky lze rozdělit na dva základní typy [Roudenský a Havlíčková 2013, Kapitola 1]:

- Funkční požadavky

- Nefunkční požadavky

2.1 Funkční požadavky

Funkční požadavky popisují funkčnost služby nebo systému, tedy co by se mělo vykonávat.

2.2 Nefunkční požadavky

Mimofunkční neboli nefunkční požadavky popisují určité vlastnosti systému či omezující předpoklady.

Dle modelu kvality FURPS [Roudenský a Havlíčková 2013, Kapitola 2], dle kterého lze také klasifikovat požadavky, se za nefunkční požadavky považují atributy U, R, P, S¹:

- U - Použitelnost - soustředí se na vnímání systému člověkem
- R - Spolehlivost - soustředí se na selhání z hlediska běžného provozu
- P - Výkonnost - soustředí se na výkon systému
- S - Podporovatelnost - soustředí se na snadnost konfigurace, škálovatelnost, atp.

Jinými slovy nefunkční požadavky definují, za jakých podmínek se funkční požadavky mají vykonávat.

Krom uvedeného dělení se lze setkat samozřejmě i s jiným, například Wiegers [2008, strana 30] dělí požadavky na podnikatelské, uživatelské a funkční.

3. User Story v agilním vývoji

Cohn [2004, strana 4] říká, že User Story (neboli uživatelský příběh) je definována jako popis funkcionality, která bude uživateli či objednateli systému přinášet přidanou hodnotu. User Story plní však ještě další tři úkoly:

- Psaný popis příběhu slouží jako připomínka a pro plánování.
- Diskuze ohledně příběhu sloužící k získání detailů (příběhu).
- Testy sdělující a dokumentující detaily, které mohou být použity k rozhodnutí, kdy je příběh hotový.

¹ Nezmíněný atribut F znamená "Funkčnost", tedy funkční požadavky, které již jsou popsány.

User Stories též mohou a měly by být takzvaně INVEST [Lewis 2016], to znamená splňovat následující výčet vlastností:

- Independent (nezávislé)
- Negotiable (dosažitelné)
- Valuable (mít přidanou hodnotu)
- Estimatable (odhadnutelné)
- Small (malé)
- Testable (testovatelné)

User Story jsou psány zákazníky a vyjadřují své potřeby, které má systém naplnit. Zaměřují se především na hodnotu pro zákazníka a zákazník používá svou terminologii. Popis User Story má být stručný. [Buchalceková 2007]

Za běžný zápis požadavku jako User Story v agilních metodikách se dnes považuje formát [Cohn 2008] [Lewis 2016]:

“Jako <uživatelské role> chci <funkcionalitu> abych získal <nějakou hodnotu>.”

Obecně může být User Story prezentována jako požadavek z pohledu cílového zákazníka [Agile Business Consortium 2015], nebo jako krátké popisy funkcí z pohledu cílového zákazníka [Cohn 2016]. Rozdíl mezi User Story a požadavky je dosti rozostřený - User Story je možno považovat za formalizovaný způsob zápisu požadavků v agilních metodikách. Rozdílné jsou pak ale přístupy k požadavkům v agilních versus rigorózních metodikách.

4. Srovnání pohledu agilních a rigorózních metodik na požadavky

4.1 Rigorózní metodiky

Rigorózní metodiky podrobně definují jednotlivé dílčí kroky procesu řízení specifikace požadavku a doporučují jejich striktní dodržování. Definuje se zpravidla zadávání, zpracování a sledování požadavků. [Buchalceková 2007] Dle jejich výkladu je již jednou schválené zadání požadavku fixní a nelze jej v průběhu následující vývoje softwaru měnit, přičemž vývoj projektu je vykonáván v delších časových iteracích. [Kinská 2009, strana 63]

4.2 Agilní metodiky

Zřejmým rozdílem mezi agilním a tradičním vývojem softwaru je právě přístup k požadavkům. Agilní metodiky dávají důraz na spolupráci se zákazníkem (narozdíl od sjednávání kontraktu). Vychází totiž z přesvědčení, že požadavky není možné všechny specifikovat, dohodnout a podepsat na začátku projektu. Dle agilních metodik se proto definují jen 4 hrubé požadavky, které jsou upřesňovány a měněny na základě každodenních jednání. To podtrhává spoluúčast uživatelů. [Buchalceková 2007]

Správa požadavků je v podání rigorózních metodik o dokumentech. V rozsáhlých dokumentech ale může poměrně snadno dojít ke zkreslení znalosti, popis může být nejednoznačný nebo vágní a zpravidla se jedná o jednosměrný tok informací od autora ke čtenáři. Proti dokumentům staví agilní metodiky za efektivnější nástroj osobní komunikaci spolu s jednoduchými technikami pro záznam požadavků. Modely a dokumenty považují agilní metodiky pouze za vedlejší produkty vývoje softwaru a soustředí se především na spustitelný kód. [Buchalceková 2007]

5. User Stories jako základ pro požadavky v agilních metodikách

“Nezáleží na tom, jakou metodologii používáte (nebo předstíráte, že používáte) - agilní Scrum, XP, UP nebo vodopádový model - špatné požadavky vždy uškodí projektu více, než jakékoliv jiné selhání.” [Lewis 2016]

Význam User Stories pro vývoj projektu byla nastíněna v kapitole User Story v agilním vývoji. Bez kvalitně zpracovaných User Stories může dojít k mnoha nežádoucím efektům v rámci vývoje - například zpoždění projektu, odlišné porozumění požadavkům,... V některých případech dokonce mohou nastat situace, kdy projekty převyšují rozpočet zejména z důvodu chyb v požadavcích [Buchalceková 2007]. Některým těmto negativním efektům by se možná dalo předejít dostatečnou a efektivní analýzou jednotlivých příběhů před zahájením práce na dané části projektu, nejlépe rovnou při plánování průběhu projektu a zadávání požadavků.

V následujících kapitolách budou nastíněny možné tipy pro analýzu User Stories a požadavků v rámci agilní metodiky Scrum, které by mohly napomoci zabránění výskytu nežádoucích efektů týkajících se nesprávné interpretace User Stories a požadavků.

5.1 Zpracování User Stories v rámci Sprint Planning Meetingu metodiky Scrum

Scrum je agilní metodikou, která by měla být charakterizována slovy jako adaptivní, rychlá a postavená na samoorganizujících týmech [Buchalceková 2009, strana 142]. Vývoj v rámci Scrumu je pak často dělen na následující jednotlivé fáze - *fázi plánovací (Planning phase)*, *vytáčení (staging phase)*, *fázi vývoje (development phase)* a *fázi dodávky (release phase)* [Buchalceková 2009, strana 143]. Případně je pak možno setkat se s jednodušším rozdělením na pouhé fáze tři - konkrétně na *fázi zahájení*, *samotného vývoje* a *fázi ukončení* [Myslín 2016, strana 85 - 86] či jiná, avšak obsahově podobná rozdělení.

Samotné User Stories a z nich odvozené požadavky pak mohou vznikat v průběhu projektu při spolupráci s lidmi [Lewis 2016] - zde se mohou projevit výhody metodiky Scrum, která by na změny měla reagovat rychle a adaptovat se.

Zadané požadavky či jejich změny se po zpracování projeví položkami v Product Backlogu (Product Backlog je spravován Product Ownerem), ze kterého jsou poté jednotlivé položky vybrány pro zpracování ve Sprintu - tento výběr probíhá v rámci Sprint Planning Meetingu, přičemž z Product Backlogu se vybírají položky s nejvyšší prioritou [Buchalceková 2007] a je to právě Sprint Planning Meeting (či jeho podobné variace), který je kritickým okamžikem pro požadavky [Lewis 2016].

5.1.1 Analýza požadavků v rámci Planning Meetingů

Jednání u Sprint Planning Meetingu je možno kategorizovat na dvě části, kde v té první Product Owner popíše, které požadavky z Product Backlogu požaduje splnit v daném Sprintu, a kde ve druhé části probíhá rozebrání vybraných požadavků na jednotlivé úlohy (v originále "Tasks") a vyhodnocení jejich náročnosti a způsobu provedení. [Lacey aj. 2016]

V rámci přípravy a tvorby User Stories u plánovacích meetingů se často začíná s takzvanými "Epics", což jsou často povrchní příběhy a návrhy funkcionalit, které mohou obsahovat velké množství detailů a požadavků, které bude třeba zpracovat [Lewis 2016]. V počátku často není nutné věnovat pozornost jejich detailům, nicméně toto se mění v rámci postupu vývoje na projektu a dalších okolnostech. Obecně je v počátku možné ponechávat User Stories jako pouhé nadpisy do té doby, dokud se nedostanou na první příčky Product Backlogu a nejsou předmětem pro další Sprint Planning Meeting [Lacey aj. 2016], tedy požadavek na jejich detailnost stoupá s jejich důležitostí v rámci projektu a Product Backlogu.

Možnost stanovení rozsahu požadavků před počátkem jejich samotného vypracování v rámci chystaného Sprintu určuje smluvní podmínky mezi Product Ownerem a vývojovým týmem [Lewis 2016] - jedná se o ideální příležitost vyjasnit si, co a do jaké hloubky bude třeba zpracovat a zahladit případné pochybnosti a nepochopení zadání, který by mohly nastat při vývoji. Podobně jako jiné schůze vyžaduje i Sprint Planning Meeting dostatečnou přípravu, aby byl efektivní (do přípravy je možno zařadit například i stanovení akceptačních kritérií, kterým se bude věnovat kapitola níže).

Podstatným detailem pak může být chování manažera na jednotlivých schůzích, kde dle pozorování [Lewis 2016] může mít manažer přímý vliv na výkonnost týmu - pokud byl manažer schopen přidat hodnotu do diskuze v rámci meetingu, pak byl tým jako celek schopen dosáhnout lepších výsledků v rámci jednání. Naopak při jednání s týmy u manažerů, kteří se nebyli schopni dohodnout na rolích, způsobu a odpovědnostech, byly výsledky schůze neuspokojivé pro obě strany. Nejlepších výsledků se dosáhlo tehdy, pokud byly manažeri, Product Owneři a Scrum Masteri zaměřeni na celkový úspěch týmu. [Lewis 2016] Samozřejmě důkladná analýza požadavku a stanovení jeho specifikací jistě pomůže zabránit nedorozuměním v situaci, kdy by byl požadavek na konci Sprintu dokončen pouze z části, a to jen z toho důvodu, že došlo k nedorozumění při Sprint Planning Meetingu - velkým pomocníkem zde může být stanovení akceptačních kritérií a v rámci nich různých testů, které splnění požadavku potvrzují. Právě akceptačními kritérii bude věnována následující kapitola.

5.1.2 Použití “TWBDW” pro analýzu aneb akceptační kritéria u User Stories

Akceptační kritéria mohou pomoci jednoznačně určit, kdy je požadavek v rámci User Story splněn, a kdy nikoliv. [Lewis 2016] ve svém článku doporučuje pro tento účel použít formát “This Will Be Done When” (toto bude splněno, když), zkráceně TWBDW.

Jednoduše řečeno, chceme-li popsat nějakou User Story, je vhodné ji formalizovat a stanovit si kritéria pro její splnění. Níže je uveden příklad User Story ze článku [Lewis 2016]:

“Jako někdo zapojený do vývoje softwaru chci psát User Stories, aby každý jasně rozuměl tomu, co je třeba udělat.”

Dále je vhodné položit si následující otázky - “Jak toho konkrétně dosáhneme? Za jakých okolností bude tento požadavek splněn?” - a následně stanovit kritéria “TWBDW” [Lewis 2016]:

- User Story bude v rámci naší organizace dostupná každému, kdo ji chce vidět.

- Vývojový tým jí dostatečně dobře porozumí, takže bude schopen odhadnout práci, která bude třeba pro její vyřešení.
- Alespoň 10% našich požadavků bude psáno ve formě User Story.

Významným pomocníkem při stanovování akceptačních kritérií u User Stories může být faktor spolupráce. Ať už se jedná o zpracování User Story před Sprint Planning Meetingem, nebo během něj, je jistě více než vhodné jednotlivá kritéria projednávat s dalšími osobami, zejména pak klíčovými stakeholdery/zadavateli, Product Ownerem či expertem na dané téma - tato skupinová analýza může pomoci rozpracovat požadavek a analyzovat jeho další dimenze, kritéria a jeho celkový rozsah [Lewis 2016] - a tomuto tématu se věnuje následující kapitola. Obecně by User Stories neměly být jen tak předány vývojovému týmu, ale měly by být vloženy konverzace týmu s Product Ownerem [Pichler 2016].

5.1.3 Spolupráce při rozboru User Stories

Ve Scrumu je mimo jiné popisována část označená jako “Backlog Grooming” nebo “Backlog Refinement” [Pichler 2016][Lewis 2016]. Do češtiny je možno toto označení přeložit jako “uhlazení” nebo “urovnání” Backlogu. Právě v rámci této části je možno připravit a “uhladit” samotnou User Story pro další Sprint Planning Meeting v rámci jednání se stakeholdery - toto jednání může napomoci k vyvození správných závěrů, k pocitu “osobního vlastnictví” User Story a ke zvýšení úrovně kreativity týmu. Díky tomu může dojít ke snížení pracovní zátěže Product Ownera (využitím spolupráce týmu a klíčových stakeholderů) a zajištění připravenosti důležitých položek z Product Backlogu. [Pichler 2016][Lewis. 2016]

Z předchozích odstavců je patrné, že spolupráce při zpracování User Stories hraje klíčovou roli. Ať už se jedná o Backlog Refinement či Sprint Planning Meeting - komunikace a vzájemné dotazování se na jednotlivé aspekty User Story a požadavku napomáhá porozumění dané problematice. Důležité je však také poznat, kdy tato spolupráce přináší dobré výsledky a kdy je User Story ve výsledku připravena.

5.1.4 Připravenost User Story

“Připravená” User Story by měla být *srozumitelná, proveditelná a testovatelná* - definice těchto stavů je definována níže [Pichler 2010]:

User Story je *srozumitelná*, pokud všichni členové Scrum Teamu shodně porozuměli zadání - zde je možno navázat na pozorování a doporučení Russe Lewise [2016] zmíněném v předchozích odstavcích, kde je zmíněna podstata stanovení srozumitelných akceptačních kritérií a kolaborativní zpracování User Stories a jejich požadavků.

Testovatelnou se User Story stává, pokud existuje způsob, jak stanovit, zdali daná funkcionálita funguje tak, jak bylo očekáváno - toto je možno zajistit v předchozích kapitolách zmíněnými akceptačními kritérii a testy.

Proveditelnost User Story je možno ohodnotit kladně, pokud je možno ji zpracovat v rámci jedné iterace Sprintu. Jak již bylo zmíněno v rámci analýzy User Stories - je vhodné jednotlivé příběhy zpracovat a případně rozdělit na samostatné celky, které jsou v rámci Sprintu proveditelné.

Připravené User Stories jsou výsledkem právě výše zmíněného “uhlazování” Backlogu (Backlog Refinement procesu). Opět je důležité zajistit, že zejména důležité položky z Product Backlogu jsou projednávány také s vývojovým týmem, který tyto položky bude následně v průběhu Sprintu zpracovávat. [Pichler 2010]

5.1.5 Otázky při jednotlivých fázích zpracování User Story v rámci Sprint Planning Meetingu

Ve článku Russe Lewise a jeho pozorování [2016] byly zmíněny a doporučeny otázky, které je možno využít při zpracovávání User Story v rámci Sprint Planning Meetingu. Dále Lewis podotkl [2016], že pro zhodnocení toho, jak moc dobře si tým vede, stačí pozorovat průběh jeho Sprint Planning Meetingu. Otázky jsou rozděleny mezi dvě části meetingu zmíněné v kapitole “Analýza požadavků v rámci Sprint Planning Meetingu” a měly by napomoci kvalitnějšímu zpracování User Story.

V případě první části Sprint Planning Meetingu (takzvané “elaboration”) pak byly doporučeny tyto otázky [Lewis. 2016]:

- Byla User Story rozdělena na nezávislý a hodnotný příběh?
- Jak dobře analyzoval tým požadavky připojené k dané User Story?
- Byly identifikovány nějaké externí závislosti? Pokud ano, budou vedeny jako samostatné User Stories? Mohou způsobit nějaké problémy?
- Byly v důsledku týmového zpracování User Story zjištěny další příběhy a akceptační kritéria? Pomohlo to ve stanovení rozsahu tohoto příběhu?
- Je navrhované řešení v rámci plánování příběhu “reálné” pro všechny (včetně vás samotných)?
- Jsou jednotlivé úlohy (“Tasky”) stanovené při plánování řešení navrhovány přirozeně a jasně?

U stanovení náročnosti zpracování jednotlivých požadavků pak byly uvedeny tyto body [Lewis 2016]:

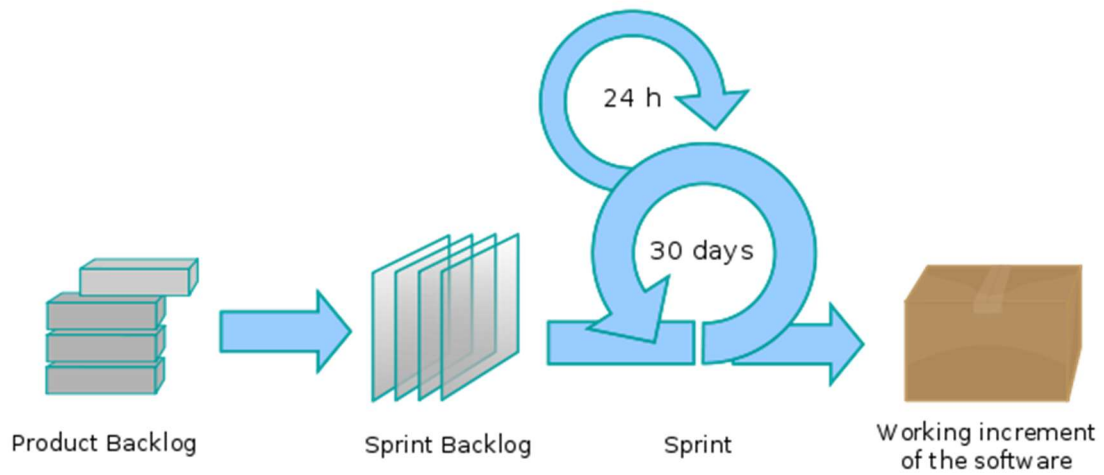
- Je dosahováno rozumného odhadu náročnosti společně - je každý do této debaty zapojen?
- Pokud jsou stanoveny časové odhady pro jednotlivé úkoly, přidávají vzhledem k
- Je každému srozuměn s jednotkou, která bude používána pro stanovení náročnosti?
- Pokud je stanoven časový odhad na jednotlivé úlohy, projeví se to v rozumném poměru na Sprintu s ohledem na odhad hodnoty příběhu?

Lewis se také věnoval samotnému ohodnocení týmové spolupráce, kde byla věnována pozornost tomu, jak je zacházeno s juniory jako členy týmu a zdali je důležité to, že všichni účastníci dostatečně porozuměli požadavku na té úrovni, aby byly schopni se plně zapojit (tedy zdali byly potřebná fakta dostatečně vysvětlena). Posledním bodem pak bylo celkové ohodnocení připravenosti týmu a toho, zdali očekávaný postup v rámci jednání odpovídal počátečním očekáváním.

5.2 Životní cyklus požadavku v agilní metodice Scrum

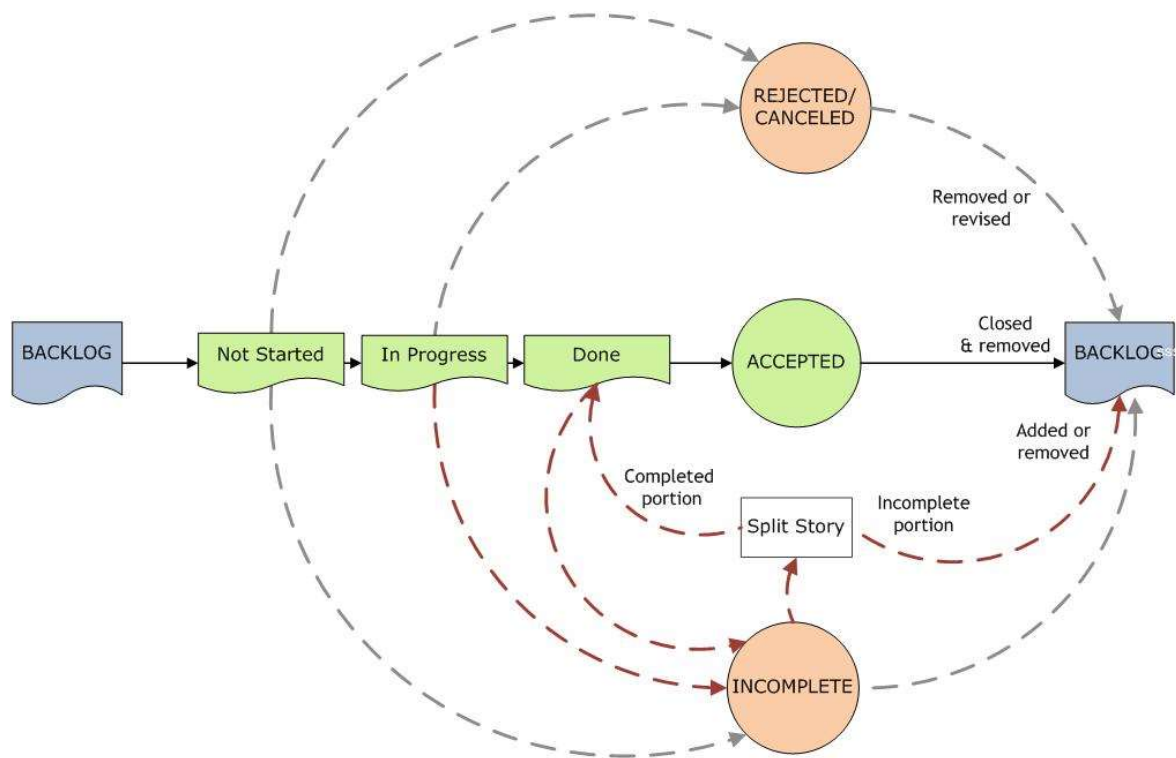
Životní cyklus User Story začíná tím, že je vytvořena a následně zařazena do Product Backlogu. V něm jsou User Stories řazeny dle priorit. Tím jak prioritizace User Story stoupá, pousouvá se i výše v Backlogu. Když se User Story dostane dostatečně vysoko, je dále analyzována a zpracována více do detailu. To může vést k tomu, že se User Story rozdělí na několik menších. Takto nově vzniklé User Stories jsou uloženy do Backlogu a jejich životní cyklus začíná.

Z User Stories v Product Backlogu jsou Product Ownerem vybírány User Stories pro následující Sprint (Product Owner je zodpovědný za položky v Product Backlogu). Ty se v rámci Sprintu zpracovávají (vybírány jsou dle důležitosti). Pokud následně projdou akceptačními testy, jsou považovány za splněné ("Done") a vyřazují se. Pokud User Story neprojde, vrací se zpět do Product Backlogu. Případně pokud je to možné, je User Story rozdělena na splněnou a nesplněnou část a do Backlogu se vrací jenom nesplněná část.



[Torak Inc. 2016]

State diagram for a user story



AgileDemystified.com

[SANTHANAM 2012]

6. Závěr

Cílem práce bylo přiblížit a zdůraznit důležitost správného zpracování User Stories v rámci agilních metodik - tento cíl se podařilo splnit. V práci byly v počátku krátce popsány vlastnosti požadavků a samotných User Stories a následně byla nastíněna důležitost kvalitního zpracování User Stories, zejména v rámci Sprint Planning Meetingu metodiky Scrum. Využito bylo poznatků a doporučení z několika zdrojů, přičemž primárním zdrojem tématu práce byl článek "User Stories are Placeholders for Requirements" od Russe Lewise [Lewis 2016].

V rámci zpracování User Stories je možno stanovit klíčové faktory, které se mohou výrazně projevit na kvalitě User Stories a z nich vycházejících požadavků - mezi tyto faktory je možno zařadit spolupráci a komunikaci mezi klíčovými osobami projektu. Kvalita samotných požadavků a User Stories může mít přímý dopad na úspěšnost projektu. Při práci na User Story je třeba si s klíčovými stakeholdery (zejména pak s Product Ownerem) stanovit, v jakém rozsahu bude vycházející požadavek řešen, a následně společně (nejlépe vždy se zapojením Scrum Teamu) prodiskutovat možná kritéria, která stanoví, kdy bude možno tento požadavek označit za splněný. Stanovení kritérií a analýza User Stories je klíčová (mimo jiné) při Sprint Planning Meetingu - zde může dojít k extenzivní výměně názorů mezi Scrum Teamem a Product Ownerem, což může napomoci správnému porozumění zadání. Dále pak může i mimo Sprint Planning Meeting (při přípravě) dojít k využití dalších osob (například expertů na dané téma) při analýze User Story - obecně spolupráce při analýze User Story se může zejména při zahrnutí Scrum Teamu projevit velmi pozitivně tak, že každý bude ve výsledku správně rozumět zadaným požadavkům.

Zdroje

- BUCHALCEVOVÁ, Alena. *Agilní metodiky a správa požadavků* [online]. 2007 [cit. 2016-12-14]. Dostupné z: <http://nb.vse.cz/~buchalc/clanky/tsw2007.pdf>
- BUCHALCEVOVÁ, Alena. *Metody budování informačních systémů*. Praha: Oeconomica, 2009. ISBN 978-80-245-1540-3.
- COHN, Mike. Advantages of the “As a User, I want” User Story template. Mountain Goat Software [online]. 2008 [cit. 2016-12-14]. Dostupné z: <https://www.mountangoatsoftware.com/blog/advantages-of-the-as-a-user-i-want-user-story-template>
- COHN, Mike. *User Stories Applied: For Agile Software Development*. Boston: Addison-Wesley, 2004. Addison-Wesley signature series. ISBN 03-212-0568-5.
- COHN, Mike. User Stories. *Mountain Goat Software* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <https://www.mountangoatsoftware.com/agile/user-stories>
- KINSKÁ, Marcela. *Správa a řízení požadavků a její implementace do projektů IS/ICT*. Praha, 2009. Diplomová práce. Vysoká škola ekonomická v Praze. Fakulta informatiky a statistiky. Dostupné z: <https://www.vse.cz/vskp/id/1150148>
- LACEY, Mitch a další. Sprint Planning Meeting. *Mitchlacey.com* [online]. 2010 [cit. 2016-12-14]. Dostupné z: <https://www.mitchlacey.com/intro-to-agile/scrum/sprint-planning-meeting>
- LEWIS, Russ. User Stories Are Placeholders for Requirements. *InfoQ* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <https://www.infoq.com/articles/user-stories-placeholders-requirements>
- MYSLÍN, Josef. *Scrum: průvodce agilním vývojem softwaru*. Brno: Compter Press, 2016. ISBN 978-80-251-4650-7.
- PICHLER, Roman. *10 Tips for Writing Good User Stories*. Romanpichler.com [online]. 2016 [cit. 2016-12-14]. Dostupné z: <http://www.romanpichler.com/blog/10-tips-writing-good-user-stories/>
- PICHLER, Roman. *Grooming the Product Backlog*. Romanpichler.com [online]. 2016 [cit. 2016-12-16]. Dostupné z: <http://www.romanpichler.com/blog/grooming-the-Product-Backlog/>
- PICHLER, Roman. *The Definition of Ready in Scrum*. Romanpichler.com [online]. 2010 [cit. 2016-12-14]. Dostupné z: <http://www.romanpichler.com/blog/the-definition-of-ready/>

- Requirements and User Stories. *Agile Business Consortium* [online]. 2015 [cit. 2016-12-14]. Dostupné z: <https://www.agilebusiness.org/content/requirements-and-user-stories>
- ROUDENSKÝ, Petr a Anna HAVLÍČKOVÁ. *Řízení kvality softwaru: Průvodce testováním*. Brno: Computer Press, 2013. ISBN 978-80-251-3816-8.
- SANTHANAM, Ram. *Life cycle of a User Story*. *AGILE DEMYSTIFIED* [online]. Chicago, Illinois, 2012 [cit. 2016-12-16]. Dostupné z: <https://agiledemystified.com/2012/02/25/user-story-life-cycle/>
- Torak Inc., *Getting started with Scrum*. Torak.com [online]. Phoenix, Arizona, 2016 [cit. 2016-12-16]. Dostupné z: <http://torak.com/resources/agile/scrum/>
- WIEGERS, Karl Eugene. *Požadavky na software*. Brno: Computer Press, 2008. ISBN 978-80-251-1877-1.