

Semestrální práce ke kurzu 4IT421 Zlepšování procesů budování IS	
Semestr	LS 2017
Autoři	- Luboš Matřha, xmatl41 - Michal Skramuský, xskrm24 - Jiří Škoda, xskoj35
Téma	Improving Scrum with the Kanban-Ace Framework
Datum odevzdání	14. 5. 2017

Abstrakt

Tato práce se věnuje Kanban-Ace Frameworku, který byl vyvinut primárně pro podporu slabých stránek metodiky Scrum. V práci jsou na začátek představeny hlavní principy a nástroje Scrumu, další část se pak věnuje samotné metodice Kanban Ace a Kanban Ace Frameworku, jejich představení, vývoji, principům a výhodám, jež nabízí. Hlavní část práce se pak zaměřuje na spojení Kanban-Ace se Scrumem, vysvětlení výhod plynoucích z této kombinace a porovnání této kombinace s jinými spojeními agilních metodik se Scrumem.

Klíčová slova

Scrum, Kanban, Kanban Ace, Kanban-Ace Framework, Agile, vizualizace, Kanban-Ace board, Akashi Bridge, Scrumban, Lean thinking

Obsah

1 Úvod.....	1
2 Scrum	1
2.1 Scrum tým	2
2.2 Události ve Scrumu	2
2.3 Scrum artefakty	3
2.4 Hodnoty Scrumu	3
3 Kanban Ace.....	3
3.1 Kanban a jeho historie.....	4
3.2 Open Kanban, Kanban Ace a Kanban-Ace Framework	4
4 Spojení Kanban-Ace Framework a Scrum.....	6
4.1 Zachování rolí	6
4.2 Sprinty a meetingy	7
4.3 Vizualizace	8
4.4 Lean thinking.....	10
4.5 Porovnání se Scrumban.....	10
5 Závěr.....	12
Seznam zdrojů	13

1 Úvod

Agilní přístup v dnešní době již neoddělitelně patří k vývoji softwaru a obecně k přístupům k řešení jakéhokoli projektu. Neexistuje však univerzální metodika, která by neměla své slabiny a nedokonalosti. Tato práce se nebude zabývat tím, jak vyvinout takovouto metodiku, ale tím, jak vylepšit již existující široce využívanou metodiku a podpořit její slabé stránky. Dle Josepha Hurtada (Joseph, 2016) totiž problematika vývoje softwaru a jeho řízení přesahuje přístup jen jedné metodiky, ale měla by být kombinací více přístupů, jež se vhodně doplňují a podporují. Proto také v roce 2016 přišel s novým rámcem, který nazval Kanban-Ace Framework a který rozšiřuje a vylepšuje hojně užívanou metodiku Scrum.

Cílem této práce bude nejdříve stručně popsat metodiku Scrum, její principy, hodnoty a nástroje, jež využívá k dosažení cílů. Dále představit metodiku Kanban Ace a z ní odvozený rámec Kanban-Ace Framework, jejich principy a silné stránky. Hlavním cílem práce je pak popis spojení metodiky a frameworku a výhody, jež toto spojení přináší a jeho porovnání s jinými přístupy.

2 Scrum

Scrum je agilním způsobem řízení projektu používaným ve vývoji software. Vznikl v 90. letech minulého století. V publikovaném článku New New Product Development Game, který napsali Hirotaka Takeuchi a Ikujiro Nonaka do Harvard Business Review, byly popsány nové možnosti řízení projektových týmů. Tuto metodiku ukazovali na příkladu z automobilového průmyslu. Ve vědecké práci popisovali, jak je tak vysoký výkon ve vývoji nových a komplexních produktů dosažen tím, když jsou týmy vedeny jako malé samostatné jednotky lidí, které nehrne dopředu úkol, ale cíl.

Název Scrum vznikl z pojmu v ragby, kde znamená znovuzahájení hry po nechtěném přerušení. Za zakladatele tohoto iterativního a inkrementálního rámce lze považovat Kena Schwabera, který použil myšlenky Takeuchiho a Nonaky při vývoji software. V roce 1995 Schwaber s Jeffem Sutherlandem prezentovali Scrum na americké konferenci OOPSLA 95. (Pollack, 2015)

2.1 Scrum tým

Scrum tým je tvořen třemi rolemi, které se zároveň doplňují.

Product Owner neboli vlastník produktu je odpovědný za prioritní seznam úkonů. Zodpovídá za to, co se bude v dalším sprintu implementovat a další detaily. Zastupuje stakeholdery a zákazníka. Je hlavním prostředníkem mezi vývojářským týmem a zákazníkem. Jeho nejdůležitějšími povinnostmi je komunikace a udržování product backlogu.

Development Team neboli vývojářský tým nese zodpovědnost za doručení správně odvedené práce za vymezený čas. Tým by měl být maximálně soběstačný.

Scrum Master jedná jako zprostředkovatel mezi Product Ownerem a týmem. Scrum master se hlavně stará o vývojáře a zajišťuje, aby fungovalo všechno, jak má a vývojáře nic nerušilo od práce. Organizuje schůzky a zajišťuje správný chod sprintu. Scrum masteri jsou často zaměňováni za projektové manažery. Ve scrumu ale manažeři nemají žádné místo, protože tým by se měl řídit sám. (Cervone, 2011)

2.2 Události ve Scrumu

Sprint neboli „srdce Scrumu“ je časový úsek (který trvá obvykle měsíc a méně). V tomto období se vytvoří použitelný kus produktu (např. část aplikace). Během sprintu se nemění cíle, které byly na začátku zadány

Plánování sprintu – na začátku sprintu se naplánuje jaká práce se udělá. Maximálně 8 hodin. Scrum master zajistí, aby se všichni zúčastnili a rozuměli proč se schází. Plánování odpovídá na otázky, co za produkt může být doručen na konci sprintu? a jak bude přerozdělena práce? Vytyčí se cíl sprintu.

Denní Scrum (Standup) je patnáctiminutové časové okénko pro vývojářský tým, kde si mohou synchronizovat své aktivity s ostatními a vytvořit si plán na další pracovní den. Denní Scrum se pořádá každý den na stejném místě ve stejný čas.

Na sprint review se zhodnotí, jaká práce byla odvedena a jaká práce se nestihla. Hotová práce se předvede stakeholderům jako tzv. demo.

Na sprint retrospective se tým se Scrum masterem ohlédne na minulý sprint a zhodnotí, co se dělalo správně a co špatně, kde se nestíhalo apod. Po zhodnocení se zavedou vylepšující změny a začíná nový sprint. (Pollack, 2015)

2.3 Scrum artefakty

Product backlog je seznam požadavků, které má tým přetvořit v hotový produkt. Obsahuje specifikaci, opravy defektů, nefunkční požadavky a další. Musí být splněno všechno z tohoto seznamu, aby byl doručen hotový produkt. Za product backlog zodpovídá product owner a sám prioritizuje úkoly, které je potřeba udělat co nejdříve.

Sprint Backlog je seznam práce, kterou musí vývojářský tým udělat za daný sprint. Sprint backlog se postupně vyvíjí z product backlogu, dokud v product backlogu nezůstane žádná další práce k dodělení. Úkoly ze sprint backlogu si vývojáři rozdělují sami mezi sebou.

Increment je vytvořený celek z hotových částí product backlogu. Na konci sprintu musí být increment hotov. Když je increment hotov, musí být použitelný, bez chyb rovnou poslaný do produkce. (Cervone, 2011)

2.4 Hodnoty Scrumu

Novinka z července 2016, kdy Ken Schwaber a Jeff Sutherland představili hodnoty Scrumu. Celkem je pět hodnot: Courage, Commitment, Focus, Openness, and Respect tzn odvaha, odhodlání, soustředění, otevřenost a respekt. Hodnoty scrumu by měly být viditelné a každá firma by si je měla spolu se Scrumem adaptovat. K adaptaci je popsáno pár pomůcek např. napsat si hodnoty na zeď a každému členovi týmu říci, ať napíše ke každé hodnotě, jak jí předvede během dne. Další možností je si ve sprint retrospective přidat hodnoty a zvlášť je ohodnotit.

3 Kanban Ace

Kanban Ace je počinem institutu AgileLion, jež sídlí v Torontu v Kanadě. Členové tohoto institutu mají několikaleté zkušenosti s využíváním Kanbanu v praxi z mnoha realizovaných projektů. Institut vznikl v roce 2012, kdy se nynější členové rozhodli, že jejich nabyté vědomosti a zkušenosti začnou šířit mezi veřejnost, jelikož se domnívali, že jejich vlastní myšlenky a „best practices“ by mohly být užitečné a přínosné na poli agilních metodik pro vývoj softwaru. Nyní na svých stránkách institut nabízí velké množství online kurzů, hodin a materiálů, které jsou zaměřené na metodiky, jež institut vyvíjí sám na základě zkušeností

svých členů. Tyto metodiky vychází především z hlavních myšlenek a principů Kanbanu. Proto na začátek, ještě než se začnu věnovat samotné metodice Kanban Ace, zabrousím trochu do historie a stručně popíšu vývoj Kanbanu, aby bylo jasné, kam tuto metodiku zařadit.

3.1 Kanban a jeho historie

Začátky Kanbanu se datují do poloviny 20. století. Kanban nebo častěji označovaný jako kanban s malým „k“ je synonymem pro výrobní techniku automobilky Toyota, se kterou přišla v roce 1945. V roce 1978 pak Taichi Ohno napsal publikaci s názvem *Toyota's Production System* (TPS), ve které byla celá výrobní technika popsána. Z této techniky pak vychází principy a hnutí *Lean*. Nicméně až v roce 1988, kdy byla tato publikace přeložena do angličtiny, se začala rozšiřovat do okolního světa. Poté začaly vznikat její různé adaptace a mnohé techniky a metodiky byly inspirovány jejími principy. Kanban, jak je znám dnes, taktéž vychází z hlavních principů a technik TPS a dále je rozšiřuje o další principy jako například o teorii úzkých míst a o mnoho dalších přístupů, které se týkají vývoje softwaru a IT. Poprvé byla tato metodika představena v publikaci v roce 2009 pojmenované Scrumban, kterou napsal Corey Ladas a o rok později pak David Anderson přišel s další důležitou knihou nazvanou *Kanban: Successful Evolutionary Change for Your Technology Business*, která taktéž výrazně přispěla k formování této metodiky. Hlavní myšlenkou Kanbanu je skloubení *Lean* a *Agilních* přístupů, přičemž se zaměřuje především na řízení vývoje a dodávky softwaru. (Joseph, 2016) (Ambler, 2010)

Nicméně v roce 2013 se Kanban rozdělil na dva směry, když se výše zmíněný David Anderson rozhodl, že svou verzi Kanbanu, kterou nazval „Kanban Method“, pojme trochu jiným způsobem, než tomu je u ostatních agilních metodik. Zaměřil se spíše na management a na možnosti jeho vylepšování. Na druhé straně pak zůstal klasický Kanban, jež se zaměřuje především na vývoj softwaru a IT obecně a zůstává tak blíže agilním metodikám. Mezi klasický Kanban se řadí například metodiky nazvané Scrumban a Leanban od výše zmíněného autora Coreya Ladase a mimo jiné pak také Kanban Ace a Open Kanban, z něhož Kanban Ace vychází. (Joseph, 2016)

3.2 Open Kanban, Kanban Ace a Kanban-Ace Framework

Kanban Ace je agilní metodika stejně jako například Scrum či Extreme Programming. Vychází z metodiky nazvané Open Kanban, jež je také počinem AgileLion institutu. Open Kanban je ultra lehká open-source metodika, která je postavená na hlavních myšlenkách Kanbanu a slouží především ke studijním účelům, kdy si na ní začátečníci mohou osvojit

principy, hodnoty a techniky Kanbanu. Nicméně může posloužit také jako odrazový můstek pro vývoj nových metodik, jelikož je zcela zdarma a volně ke stažení. Přesně tak vznikl také Kanban Ace, jež přebírá základ z Open Kanbanu a dále na něm staví a rozšiřuje ho. (AgileLion Institute, 2013)

Kanban Ace je na internetových stránkách AgileLion institutu charakterizován jako způsob, jak docílit agility při vývoji softwaru, a přitom se zaměřovat na dodávání hodnoty zákazníkům. To tedy znamená, že staví na hlavních principech Manifestu Agilního vývoje softwaru, jimiž jsou:

- ***„Jednotlivci a interakce před procesy a nástroji***
- ***Fungující software před vyčerpávající dokumentací***
- ***Spolupráce se zákazníkem před vyjednáváním o smlouvě***
- ***Reagování na změny před dodržováním plánu***“ (Manifest Agilního vývoje software)

A stejně tak jsou pro tuto metodiku klíčové *Lean* principy, což znamená, že se zaměřuje na efektivní řízení procesů a jejím cílem je uspokojit zákazníka přinášením optimální hodnoty, a přitom se snažit snižovat odpad na minimum. Dále pak také přináší své principy, metody, techniky a hodnoty, které tuto metodiku dělají unikátní a použitelnou nejen pro vývoj softwaru a oblasti IT, ale i pro mnoho dalších oblastí, jako například management, marketing, obchod, design a jiné. Dle institutu je tedy tato metodika vhodná jak pro vývojáře, tak pro manažery, podnikatele, konzultanty či projektové manažery. Dále také uvádí čtyři hlavní principy, resp. vlastnosti této metodiky, které představují její silné stránky.

- Skloubí *Lean* a Agilní přístupy, je jednoduchá a efektivní a dá se aplikovat na jakýkoliv projekt či organizaci.
- Není to jen pouhá metodika, ale celý rámec pro řízení práce, jež přináší nové přístupy pro IT management.
- Je pro ni klíčová technická dokonalost a řízení kvality, tzn. musí být doručovány pouze kvalitní produkty a řešení bez chyb.
- Drží krok s aktuálními trendy a novými vědomostmi a praktikami z oblasti *Lean* a agilních metodik. (AgileLion Institute)

Nejnovější myšlenkou autora této metodiky Josepha Hurtada pak byla změna Kanban Ace na framework. Tedy již neprezentuje Kanban Ace jako samostatnou metodiku, ale spíše jako rámec, který definuje hodnoty, principy a techniky, kterými přistupovat k již existujícím

agilním metodikám a pomáhat tak vylepšovat jejich slabé stránky. Jelikož je Joseph Hurtado certifikovaný Scrum master a je si vědom toho, že Scrum je globálně nejpoužívanější metodikou pro vývoj softwaru a taktéž si je vědom jeho slabin, rozhodl se jako první přizpůsobit rámec Kanban Ace právě pro tuto metodiku. Chce tedy podpořit slabá místa Scrumu, ale zároveň zachovat jeho hlavní principy. K tomu používá přístup nazvaný podle japonského mostu *Akashi*, který představuje nástroje a techniky odvozené z Kanbanu k podpoření slabých stránek Scrumu. Nehodlá však skončit jen u Scrumu, ale jeho cílem je z Kanban Ace Frameworku udělat globální univerzální nástroj, jež bude užitečným přínosem pro agilní vývoj softwaru. Taktéž hodlá napsat o Kanban-Ace Frameworku knihu, která by měla vyjít již během tohoto roku a která bude obsahovat mnoho dalších užitečných nápadů, přístupů a inovací. (Joseph, 2016)

4 Spojení Kanban-Ace Framework a Scrum

Kanban-Ace Framework je zamýšlený jako rámec pro doplnění jiné agilní metodiky o nejlepší praktiky Kanbanu. Při propojení Kanban-Ace Framework a Scrum tedy nedochází k velkým změnám v tom, jak Scrum funguje a vývojáři, kteří do té doby pracovali podle Scrumu i nadále mohou využívat jeho nástroje. Důvodem proč propojením těchto dvou metodik nedochází k velké změně Scrumu je ten, že jejich zaměření na úroveň řízení je odlišné. Zatímco Scrum je především zaměřen na úroveň projektového řízení, což znamená, že slouží k plánování jednotlivých etap projektů – sprintů, k definování rolí, k plánování meetingů a k prioritizaci jednotlivých úkolů v rámci sprintů, tak Kanban-Ace Framework je spíše zaměřen na úroveň řízení exekutivy. Slouží k omezení množství rozdělané práce, k optimalizaci práce na úkolech, zeštíhlování procesů a měření a řízení toků. (Joseph, 2016)

4.1 Zachování rolí

Jak již bylo řečeno při kombinaci těchto dvou metodik je zachováno to nejlepší ze Scrumu, což se týká i definovaných rolí. Je považováno za velkou výhodu mít v rámci metodiky definovanou roli, která zodpovídá za komunikaci se zákazníkem, předávání požadavků, prioritizaci úkolů v rámci jednotlivých sprintů a demonstraci hotové práce za sprint zákazníkovi. Kvůli těmto výhodám byla role Product ownera v této kombinaci zachována. Dále byla zachována i role Scrum mastera jehož vlastnosti jako znalost a prosazování metodiky, technologická podpora týmu a koordinátor týmu jsou taky velmi potřebné. Role vývojářského týmu je v obou metodikách stejná, a tedy také zůstává. Z těchto důvodů tedy kombinace těchto dvou metodik, převzala role ze Scrumu, protože přinášejí mnoho výhod.

4.2 Sprinty a meetingy

Velikou výhodou Scrumu jsou oddělené iterace. Díky nim je možné naplánovat přijatelné množství práce, které má pro zákazníka největší přidanou hodnotu na několik týdnů dopředu a po několika týdnech je zákazníkovi předvést jako funkční produkt. Zároveň však také ze Scrumu dělají ne tak úplně agilní metodiku. Požadavky je možné přidávat do sprintu jen na jeho začátku, během plánování a za celou dobu sprintu již není možné přijímat a implementovat žádné další požadavky, což ze Scrumu dělá metodiku, která sice není vodopádová, ale vlastně se skládá z velkého množství malých vodopádů. Ve fázi projektu, kdy je software vyvíjen, postupně se přidávají funkcionality, a ještě není nasazen na produkci, je tento přístup žádoucí a velice výhodný. Avšak v případě toho, že se aplikace nasazuje na produkční prostředí či již nasazena je, tak se vývojový tým často setkává s tím, že přichází velké množství požadavků s vysokou prioritou, které je potřeba odbavit velmi rychle, například ve formě hotfixů a není možné je řešit pomocí sprintů. Tuto problematiku právě řeší propojení s Kanban-Ace Frameworkem. Při jejich kombinaci je možné využívat jak oddělené iterace, tak i průběžné nasazování v závislosti na tom, v jaké části projektu se vývojový tým zrovna nachází a jaký přístup je optimálnější. Průběžné nasazování má výhodu v tom, že je možné nasadit verzi software kdykoli, kdy je připravená, což může být několikrát týdně, ale třeba i několikrát denně. Kromě toho, že tento přístup dává možnost rychle promítnout změnu na produkci, tak také v kombinaci s managementem kvality dodává jistou dávku jistoty, že provedená úprava či přidaná funkcionality bude bez chyb a kompatibilní. Toto je řešeno tím, že si každý úkol v rámci svého životního cyklu projde fází integrace, testování a demonstraci na demo prostředí.

Další věc, která je ze Scrumu zachována jsou meetingy. Meetingy jsou důležité pro plánování a prioritizaci práce pro nadcházející sprint a pro informování o průběhu práce během sprintu a případné řešení problémů, které by potenciálně mohly vézt k záseku ve vývoji. Přirozeně plánování práce pro sprint nedává smysl, pokud tým vyvíjí v režimu průběžného nasazování, kdy se úkoly zpracovávají přímo. Avšak denní stand-up meetingy jsou využívány i v tomto režimu, jelikož je potřeba mít přehled o stavu úkolů a řešit problémy ve vývojářském týmu. Při průběžném nasazování se do jisté míry využívá i sprint review, které slouží ke zkontrolování úprav či vytvořené funkcionality. Tento typ meetingu je však o něco zeštíhlen a dochází k němu výrazně častěji, jelikož jak bylo zmíněno v tomto režimu se nasazují změny třeba i několikrát denně. Sprint retrospective také při průběžném nasazování není povinný, jelikož není přesně dáno, za jakou dobu by se měl provádět, nicméně potřebné

informace o poučení a zlepšení mohou být zaznamenávány během vývoje například online wiki. (Joseph, 2016)

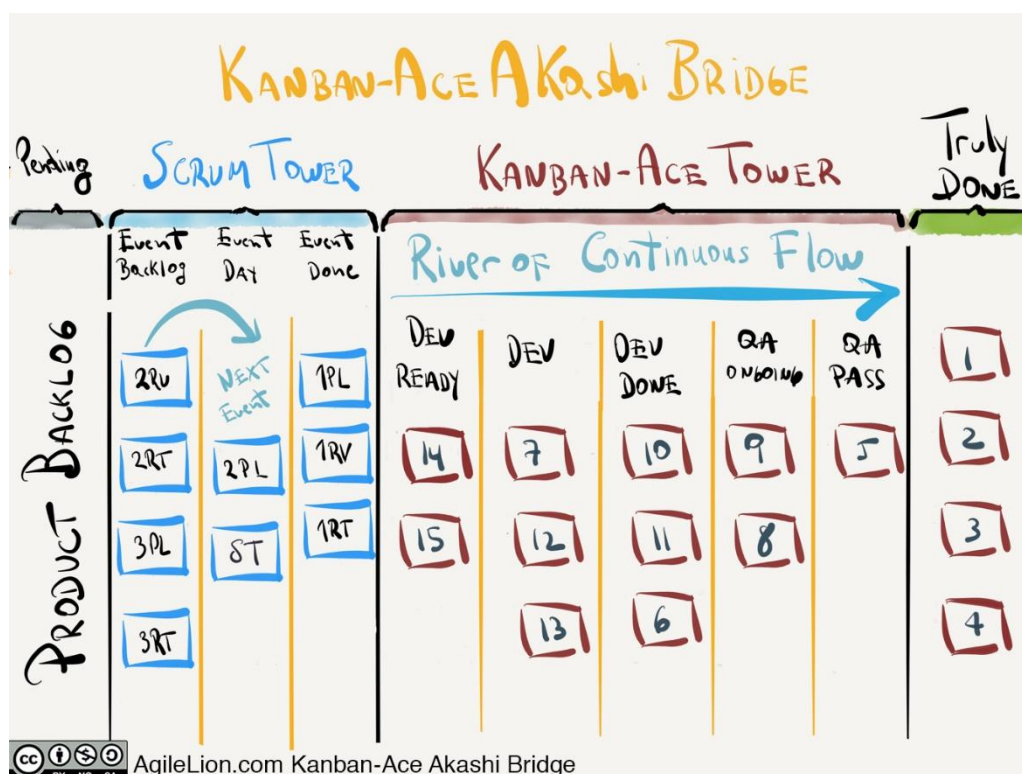
4.3 Vizualizace

Forma vizualizace, jakou používá Scrum a jakou používá Kanban-Ace Framework je podobná. V rámci Kanban-Ace Framework se sice nepoužívá nic jako burn down chart, ale Scrum board a Kanban-Ace board jsou si velice podobné.

Scrum board obsahuje sloupčky, které reprezentují jednotlivé stavy úkolů v rámci sprintu. Úplně nalevo je sloupec, který reprezentuje sprint backlog. Úkoly, které musejí být udělány v rámci probíhajícího sprintu, aby byl sprint považován za dokončený. Jakmile se úkol dostane přes stavy jako "in progres", "testing" až do stavu úplně napravo, tak je považován za hotový a již se s ním nemanipuluje, ledaže by se v něm našla nějaká chyba a byl by vrácen k opravě.

Kanban-Ace board má také sloupce, které reprezentují stavy jednotlivých úkolů, ale tok práce je vnímán jinak. Na rozdíl od Scrum board, Kanban-Ace board neznázorňuje jen úkoly, které je potřeba udělat pro splnění iterace, a které jsou ukončené, když se dostanou do stavu zobrazeného na pravé straně tabule, ale znázorňuje opakující se činnosti, které se znovu zařadí na levou stranu tabule a bude se na nich dělat až to bude potřeba. Dalším rozdílem je, že na Kanban-Ace board se znázorňuje, kdo všechno na daném úkolu pracuje. Díky tomu lze pak lépe sledovat, kde se hromadí práce a je případně za potřebí pomoc vedoucího týmu. Důležitou částí Kanban-Ace board je omezení rozdělané práce. Nad sloupcem či sloupci může být zobrazeno číslo znázorňující, kolik může být maximálně v daném stavu rozdělaných úkolů. Toto omezení zvyšuje efektivitu práce tím, že jednotliví vývojáři nemusí řešit větší množství úkolů najednou a neztrácejí čas přepínáním se mezi těmito úkoly.

Kombinací Scrum board a Kanban-Ace board vzniká nová metoda vizualizace, která se nazývá Akashi Bridge. Tato metoda umožňuje zobrazit najednou globální i detailní pohled na průběh projektu, potažmo na průběh probíhajícího sprintu.



Obrázek 1: Kanban-Ace Akashi Bridge (Zdroj: www.infoq.com/articles/kanbanace-improve-scrum)

Zleva na obrázku 1 lze vidět sloupec "Pending". Tento sloupec obsahuje všechny úkoly, které bude potřeba během celého projektu udělat, aby mohl být projekt úspěšně dokončen. Další sloupec s názvem "Scrum Tower" obsahuje informace o plánovaných a proběhlých meetingách. Tento sloupec týmu dává možnost vidět jaké meetingy jsou plánovány, aby se na ně mohli připravit a také dává managementu globální pohled na celý projekt. Na obrázku 1 lze vidět, že již proběhl první sprint plan a stejně tak i sprint retrospective a review. To znamená, že je již dokončen první sprint a tento den se koná sprint plan pro druhý sprint. Ve třetím sloupci, který se jmenuje "Kanban-Ace Tower", lze vlastně vidět Kanban-Ace board. Na obrázku 1 sice nelze vidět využití omezení rozdělané práce ani zobrazení toho, kdo na daných úkolech pracuje, nicméně i v Akashi Bridge jsou tyto prvky obsaženy. Tento sloupec znázorňuje detailní pohled na průběh projektu, potažmo zobrazuje současný stav sprintu, na kterém tým dělá. Posledním sloupcem je "Truly Done", který obsahuje úkoly Product backlogu, jenž již byly úspěšně dokončeny.

Díky této metodě vizualizace je tedy možné mít zároveň přehled o tom, jak na tom je celý projekt, ale zároveň lze sledovat postup u jednotlivých úkolů a případně řešit problémy či záseky s nimi souvisejícími. (Joseph, 2016)

4.4 Lean thinking

Jak již bylo několikrát zmíněno Kanban-Ace Framework přináší do kombinace se Scrumem nejlepší praktiky Kanbanu. Jednou z nich je i lean thinking.

Jedním z klíčových pravidel lean thinking je omezení zbytečného plýtvání časem. Za plýtváním časem se dá považovat i nadbytečné množství meetingů, které jsou součástí Scrum. Vývojáři nepotřebují každý den řešit postup týkající se jejich práce a pro některé to může být dokonce frustrující a mohou mít pocit, že jsou příliš kontrolováni. Kanban-Ace tedy přichází s možností omezení každodenních stand-up meetingů. Tyto meetingy stačí pořádat jen několikrát za týden a tím pádem je možné ušetřit čas pro vývoj. Zároveň jejich konání lze plánovat a znázorňovat na Akashi Bridge a tím pádem mít nad jejich konáním lepší kontrolu.

Další omezení plýtváním časem je omezení rozdělané práce. Je známo, že když vývojář dělá na více úkolech najednou nebo dokonce více projektech najednou, tak jeho efektivita velmi klesá. Je to dáno tím, že nakonec stráví více času přepínáním mezi jednotlivými úkoly či projekty. Pro každý projekt může mít specifické prostředí, které si musí připravit či prostě jen dlouho trvá, než se v problematice znovu zorientuje. Omezení rozdělané práce je potom součástí Kanban-Ace board potažmo Akashi Bridge.

Důležitá věc, kterou s sebou Kanban-Ace Framework přináší je optimalizace pracovních toků dle potřeb zákazníka. Nejjednodušší způsob optimalizace je na základě prioritizace úkolů Product ownerem na základě toho, které z nich přinesou zákazníkovi nejvyšší přidanou hodnotu, potažmo které v současné době nejvíce potřebuje. Tento způsob využívá Scrum, ale Kanban-Ace Framework dává možnost přizpůsobit i způsob jakým budou požadavky odbavovány a jakým způsobem a jak často budou zákazníkovi dodávány. Toho je dosaženo jednoduše tak, že lze přepínat mezi režimem vývoje v oddělených iteracích a režimem průběžného nasazování. Dále lze například měnit délku sprintů potažmo frekvenci testování a nasazování nových verzí. (AgileLion Institute, 2013) (Ambler, 2010)

4.5 Porovnání se Scrumban

V této části práce bude porovnávána kombinace Kanban-Ace Framework a Scrum se Scrumban. Tyto dvě hybridní metodiky budou porovnávány podle 4 kritérií:

1. Převzetí rolí ze Scrumu
2. Přístup k iteracím
3. Přístup ke Scrum meetingům

4. Vizualizace

Co se týče převzetí rolí, tak se dá úspěšně tvrdit, že obě metodiky je do určité míry přebírají. Rozdíl je však v tom, jakým způsobem k tomu přistupují. Scrumban ke Scrumovým rolím přistupuje tak, že nejdůležitější je vývojářský tým a případné další role je možné používat, ale nejsou nutné. Kanban-Ace naopak k rolím přistupuje tak, že je správné je zachovat ve stejném smyslu jako ve Scrumu, maximálně je potřeba lehce upravit jejich úlohu.

Kanban-Ace v kombinaci se Scrumem nabízí možnost využívat jak iterativní vývoj v podobě sprintů, tak průběžné nasazování, které spíše bližší řízení toků práce Kanbanu. Scrumban naopak iterace odstraňuje a rovnou je nahrazuje průběžným nasazováním.

Ke Scrum meetingům mají tyto metodiky vcelku dost odlišný přístup. Scrumban považuje denní stand-up meetingy za důležité především z hlediska plánování dalšího postupu, získávání informací o stavu projektu a popřípadě návrhů na zlepšení. Ostatní meetingy nejsou považovány za tak důležité z velké části kvůli tomu, že Scrumban nepodporuje iterativní vývoj. Kanban-Ace považuje za redundantní denní stand-up meetingy a snaží se je omezit ve prospěch času na vývoj. Ostatní meetingy považuje za více důležité, ale jen za předpokladu, že je projekt v režimu iterativního vývoje.

Scrumban ani Kanban-Ace Framework nepřebírají ze Scrumu nástroje jako je burn down chart. Obě dvě metodiky však využívají Kanban board. Velkou výhodou kombinace Kanban-Ace Frameworku a Scrumu oproti Scrumban je rozšířená metoda vizualizace, která přebírá prvky vizualizace z obou metodik, a to Akashi Bridge. Tato metoda dává možnost náhledu na projekt jak z globálního, tak z detailního pohledu. V případě Scrumban by však takováto vizualizace nemusela být tolik užitečná, jelikož nepoužívá iterace a prosazuje především stand-up meetingy na denním režimu, které není potřeba znázorňovat.

Z tohoto porovnání tedy vyplývá, že Scrumban se spíše více přiklání ke způsobu vývoje Kanbanu o čemž vypovídá absence iterací. V tomto směru je kombinace Kanban-Ace a Scrumu více kompromisní. Nabízí možnost vývoje po iteracích, ale zároveň i vývoj pomocí průběžného nasazování. Obecně by se dalo tvrdit, že Scrumban je daleko víc upravený pomocí principů Kanbanu, zatímco Kanban-Ace Framework slouží jako doplnění Scrumu o nejlepší praktiky Kanbanu. (Pahuja, 2016)

5 Závěr

Všechny vytyčené cíle byly naplněny. Dle našeho názoru propojení Kanban-Ace Frameworku se Scrumem funguje dobře, protože samotný Kanban-Ace Framework je tvořen jako doplňující rámec k jiné agilní metodice. Výhodou tedy je, že Scrum stále slouží jako metodika, podle které se vývojáři řídí a mohou nadále využívat jeho nástroje. Scrum je však používán spíše na projektové řízení, kdežto Kanban-Ace Framework slouží k řízení exekutivy. Role ze Scrumu jsou zachovány a stejně tak i Sprinty a meetingy, naopak Kanban-Ace přidává průběžné nasazování, díky čemuž lze rychleji promítnout změny na produkci. Dále je představena nová metoda vizualizace Akashi Bridge, která kombinuje Scrum board a Kanban-Ace board a umožňuje tak najednou zobrazit pohled na průběh projektu globálně i detailně.

Omezením bylo to, že Kanban-Ace Framework je velmi mladý a autor zatím neměl čas vydat knihu ani jinou publikaci, jež by detailně popisovala jeho hodnoty a principy. Nyní se totiž zaměřuje na šíření svých myšlenek a znalostí především pomocí online kurzů, takže zdroje s informacemi o Kanban-Ace Frameworku jsou zatím jen omezené.

Seznam zdrojů

AgileLion Institute. 2013. Open Kanban - An Open Source, Ultra Light, Agile & Lean Method. *AgileLion*. [Online] AgileLion Institute, 16. Srpen 2013. [Citace: 29. Duben 2017.] <http://agilelion.com/agile-kanban-cafe/open-kanban>.

—. Why Learn Kanban Ace? *AgileLion*. [Online] AgileLion Institute. [Citace: 29. Duben 2017.] <http://agilelion.com/why-learn-kanban-ace>.

Ambler, Scott W. 2010. Dr. Dobb's. [Online] 2010. [Citace: 11. Březen 2017.] <http://www.drdoobbs.com/architecture-and-design/introduction-to-kanban/225702051?cid=Ambysoft>.

CERVONE, H. Frank, 2011. Understanding agile project management methods using Scrum. *OCLC Systems & Services; Bradford* [online]. 27(1), 18–22. ISSN 1065075X. Dostupné z: doi:<http://dx.doi.org.zdroje.vse.cz/10.1108/10650751111106528>

Joseph, Hurtado. 2016. Improving Scrum with the Kanban-Ace Framework. *InfoQ*. [Online] AgileLion, 19. Říjen 2016. [Citace: 28. Duben 2017.] <https://www.infoq.com/articles/kanbanace-improve-scrum>.

Kniberg, Henrik, a další. 2010. *Kanban and Scrum: making the most of both*. místo neznámé : S.l.: C4Media. InfoQ enterprise software development series, 2010. ISBN 978-0-557-13832-6.

Manifest Agilního vývoje software. *Manifest Agilního vývoje software*. [Online] [Citace: 12. Květen 2017.] <http://agilemanifesto.org/iso/cs/manifesto.html>.

Pahuja, Savita. 2016. What is Scrumban? *Agile Alliance*. [Online] Agile Alliance, 2016. [Citace: 5. Květen 2017.] <https://www.agilealliance.org/what-is-scrumban/>.

Pollack, Lisa. 2015. *How project management turned into a Scrum*. London, UK : Financial Times, 2015. ISSN 03071766.