

Škálování agilního vývoje - Spotify & Lego

| Semestrální práce ke kurzu 4IT421 Zlepšování procesů budování IS | |
|--|---|
| Semestr | ZS 2018/2019 |
| Autoři | Ondřej Král krao00 Daniel Korous kord10 Kenan Dervišević derk01 |
| Téma | Scaling Agile - Spotify Model |
| Datum odevzdání | 21.12.2018 |

Abstrakt

Tato práce se zaměřuje na metodiky škálování agilního vývoje. Jejím cílem je popsat agilní přístupy a jejich škálování jak teoreticky, tak ve firmě Spotify. Dále jsme se zaměřili na představení agilního vývoje využitého ve společnosti LEGO a na osobnost Henrika Kniberga, který měl možnost se na zasazení těchto metodik v daných firmách výrazně podílet. V poslední části práce se zabýváme porovnáním agilních metodik ve Spotify i Lego a rekapitulujeme všechny výsledky naší práce.

Klíčová slova

škálování agilního vývoje, Spotify, Lego, Henrik Kniberg

Obsah

| | |
|------------------------------|----|
| Úvod | 2 |
| 1. Škálování agilního vývoje | 3 |
| 2. Osobnost Henrik Kniberg | 6 |
| 3. Spotify | 7 |
| 4. Lego | 10 |
| 5. Spotify vs. Lego | 13 |
| Závěr | 15 |
| Literatura | 16 |

Úvod

Tato semestrální práce je věnována tématu škálování agilního vývoje softwaru. Agilní vývoj v posledních letech zaznamenal obrovský boom díky své úspěšnosti, ze které se přirozeně řada společností snaží těžit, a nasadit tak agilní vývoj i ve své organizaci.

V našem článku nejprve řádně představíme agilní vývoj, pak problematiku škálování agilního vývoje a následně ji přiblížíme ve dvou světově známých podnicích: Spotify a LEGO. Tyto dvě společnosti spojuje autor hlavních zdrojů této semestrální práce, Henrik Kniberg, který v každé ze společností strávil několik let. Skrze jeho pohled následně porovnáváme a hodnotíme jejich přístup k agilnímu škálování.

1. Škálování agilního vývoje

Agilní řízení projektu je právě takové řízení projektu, které se vyznačuje svou připraveností a schopností rychle reagovat, má vynalézavý a přizpůsobivý charakter. (Doležal, 2016)

Agilní vývoj nejlépe popisuje pravděpodobně jeho manifest, který roku 2001 sepsali jeho autoři.

Agilní manifest zní:

„Objevujeme lepší způsoby vývoje software tím, že jej tvoříme a pomáháme při jeho tvorbě ostatním. Při této práci jsme dospěli k těmto hodnotám:

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

Jakkoliv jsou body napravo hodnotnější, bodů nalevo si ceníme více.“

(Beck et al., 2001)

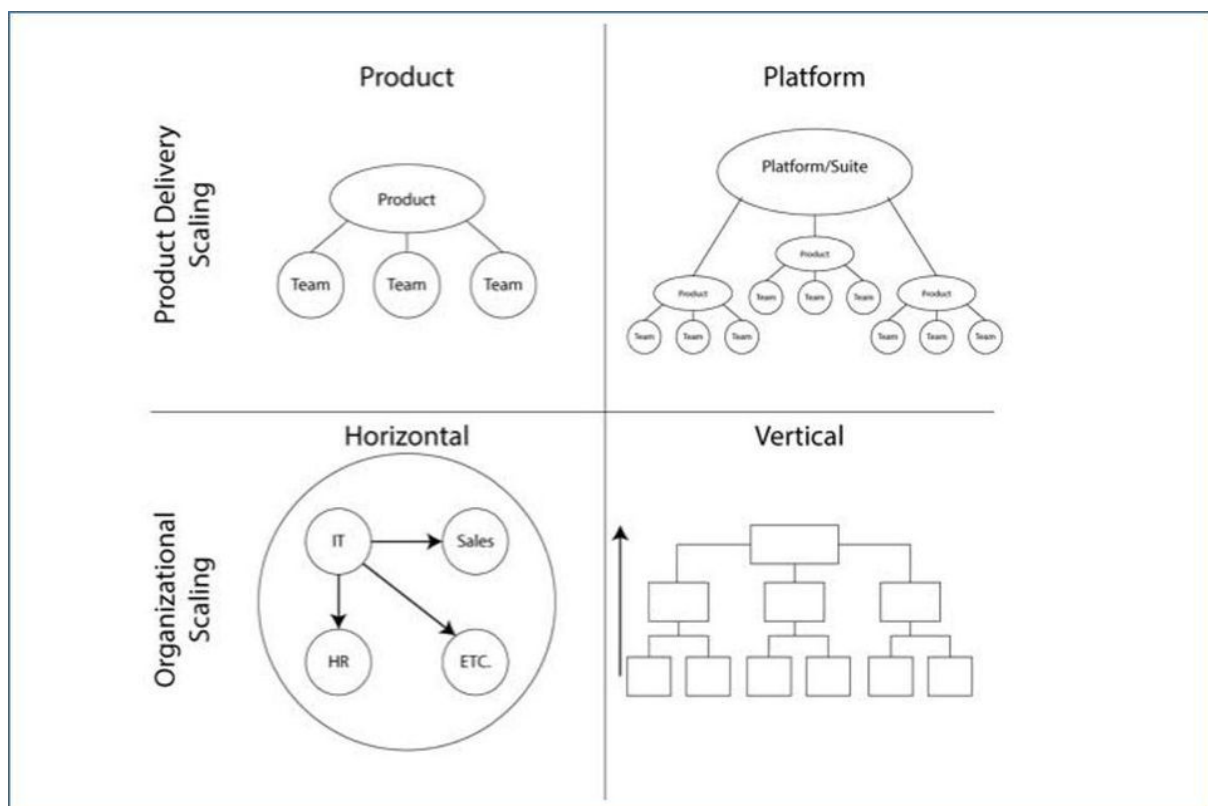
Neboť tradiční agilní vývoj však nelze zcela aplikovat na celé organizace, ale i tak se jeví jako vhodnější způsob řízení v budoucnu, například oproti vodopádovému vývoji, vznikla postupně myšlenka škálování agilní metodiky i pro více týmů s různým cílem a dokonce také pro velké organizace, či podniky a jejich oddělení jejichž hlavním cílem nemusí ani nutně být tvorba software.

Škálování agilní metodiky se dá dále rozdělit do několika konkrétnějších skupin podle typu. Hartman (Denning, 2016) jednotlivé skupiny dělí na Produktové, Platformní, Horizontální a Vertikální.

Produkt a Platforma jsou typy zaměřující se na škálování agilní metodiky skrze více týmů. V případě Produktu se jedná o více týmů pracujících na konkrétním produktu. V případě Platformy se pak tento typ týká škálování agilních metodik mezi týmy, jež pracují na různých produktech, které pak spadají do jedné platformy.

Horizontální škálování agilní metodiky je o tom, aby se koncepty agilního vývoje přesunuly i do jiných segmentů organizace než je IT a aby celá firma byla agilní i v případě řešení problémů spojených s byznysem a aby se agilně řešily také všechny další problémy a úkoly a celá firma tak fungovala na agilní metodice a myšlence.

Vertikální škálování se zaměřuje na aplikaci agilní metodiky v organizační struktuře směrem vzhůru a má za cíl přinést agilní myšlenku i mezi koordinátory a lídry, kteří řeší i naprosto jiné úkoly než ty spojené s vývojem softwaru nebo informačními technologiemi. Vertikální stejně jako Horizontální spadají do kategorie přístupů ke škálování celých organizací.



Podniky, které se však rozhodnou škálovat svůj agilní vývoj musí čelit několika výzvám. Mezi ně patří například zajištění kvalitní komunikace mezi týmy a jejich koordinace, do vývoje mohou také různě vstupovat další oddělení či partneři, kteří nefungují agilně, ale zároveň je potřeba zajistit možnost jejich funkční kolaborace. Díky neutuchajícímu zájmu o škálování agilní metodiky se postupně vykrystalizovalo několik metodik, které se snaží o efektivní transformaci podniků do agilního módu a podporovat je i po přechodu. (Hubený, 2018)

Mezi nejznámější z metodik patří například: Scaled Agile Framework (SAFe), Lean Management, Scrum of Scrums, Nexus, Large-Scale Scrum (LeSS), Disciplined Agile Delivery (DAD).

Mezi nejvýznamnější důvody, proč agilní vývoj v podniku škálovat pak patří především skutečnost, že firma potřebuje aplikovat agilní metodiku na svou kulturu a velikost. U zavádění agilního vývoje lze stále pozorovat hodnocení individuality a jejich interakcí, nabádání ke kolaboraci a redukování hierarchie. Software je obvykle vydáván postupně ve formě po menších částech v pravidelném cyklu, což vede k lepšímu pochopení potřeb zákazníka, většímu prostoru na jeho zpětnou vazbu a její zpracování do další verze softwaru. Dokumentace je stále důležitá, ale její priorita není již tak vysoká. Zákazník by také měl mít prostor lépe kolaborovat s manažery, vývojáři a designéry, což by také mělo pozitivně působit na jejich spolupráci, která bude méně připomínat vyjednávání. Hlavním cílem všech zastoupených by mělo být vytvořit kvalitní produkt. Další z velmi významných důvodů zavedení agilního škálování a myšlení v podniku je bezpochyby pozitivní vnímání reakce na chybu a její rychlá oprava za chodu nad stanoveným původním plánem a jeho slepým následováním. V zkratce se tedy dá říci, že škálování agilního vývoje se soustředí především na redukování nadbytečné byrokracie a času za účelem dodání kvalitního software. (Macdonagh, 2012)

Pro příklad si zde lehce rozepíšeme jak vypadá Scaled Agile Framework (SAFe), který mimo jiné sloužil také jako předloha při škálování agilního vývoje ve firmě Spotify.

Scaled Agile Framework (SAFe) je jednou ze základních platforem, které umožňují agilnímu vývoji být dobře škálovatelný pro podnikové systémy a software. Jedná se o jeden z nejběžněji používaných přístupů ke škálování agilu. Jejím cílem je sladit vývoj s dalšími obchodními cíli a stejně tak i jednotlivé týmy. SAFe podobně jako jiné frameworky jako je například Scrum, nabízí flexibilní vyvíjející se rámeček, ve kterém jsou splněny jednotlivé milníky důležité k dokončení obvykle většího projektu. SaFe se pro svou filozofii a postupy brzy stala velice oblíbeným zdrojem pro velké podnikové společnosti, jejichž týmy sdílejí vzájemnou závislost. Její popularita je také řízena svým strukturovaným a metodickým přístupem k harmonizaci a dokončování projektů. Na základě agilních zásad pak má SAFe pět hlavních složek, kterými se zabývá. Jedná se o architekturu, integraci, správu, financování a role. Tento proces zájmu se odehrává na třech úrovních. Těmi jsou tým, program a portfolio. Týmy bývají obvykle multidisciplinární. (Watts, 2017)

2. Osobnost Henrik Kniberg

Henrik Kniberg je odborným trenérem agilních a lean přístupů, známý zejména díky svému působení ve společnostech Spotify a Lego.

Je bývalým členem představenstva společnosti Agile Alliance a jeho zálibou je pomáhat společnostem uspět s technickou i lidskou stránkou vývoje softwaru. Během posledních 15 let působil jako CTO (Chief Technology Officer) ve 3 různých švédských IT společnostech, kde pomáhal v začátcích právě s implementací agilních a lean principů.



Henrik je velmi zkušený učitel, trenér, kouč, spisovatel a několikrát oceněný řečník. Jeho knihy mají přes půl milionu čtenářů, byly přeloženy do více než 12 jazyků a doposud slouží jako primární příručka pro zavedení agilních a lean přístupů pro stovky organizací po celém světě. Mezi nejznámější z nich patří například knihy *Scrum and XP from the Trenches*, *Kanban and Scrum: Making the most of both* nebo kniha *Lean from the Trenches: Managing Large-Scale Projects with Kanban*. Dále je autorem tzv. Scrum check-listu a řady naučných animovaných videí. (CRISP, 2018; Scrum Alliance, 2018)

Henrik Kniberg je původem ze Švédska, vyrostl však v japonském Tokyu ve čtvrti Roppongi, kde docházel na základní školu. Během svého dospívání vyzkoušel řadu hudebních nástrojů od piana, přes bicí, po kytaru a byl rozhodnutý stát se muzikantem (odtud lze vysledovat jeho motivaci pro působení ve společnosti Spotify). Po vojenské službě ve Švédsku se však přihlásil na technickou vysokou školu, kde vystudoval počítačové vědy a následně se dal na dráhu softwarového inženýrství. Své náhlé rozhodnutí zdůvodňuje nejistou uměleckou branží a také strachem o to, že mu hudba zevšední při ztrátě inspirace. Navíc zmiňuje, že programování a počítače ho ve škole vždy bavily. Nyní žije se svou ženou Sofií nedaleko Stockholmu na statku jménem Lövhagens Gård. Zde se také nachází celá jeho rodina včetně rodičů a jeho dvou sester. (Kniberg, 2018)

3. Spotify

V této kapitole je představeno řešení škálování agilního vývoje ve společnosti Spotify. Náhled do této organizace nám nabízí článek (Kniberg a Ivarsson, 2012) zveřejněný před několika lety, který však stále popisuje aktuální strukturu této společnosti.



Pro čtenáře, kteří by snad neznali společnost Spotify, mi ji dovoluňte stručně představit. Spotify je významná světová služba, která nabízí streamování či podcasting hudby od vybraných umělců a vydavatelství. Byla založena v roce 2006 a stejně jako Henrik Kniberg pochází ze Švédska. Posluchači mají dvě základní možnosti: služba je k dispozici na mobilních zařízeních i počítači buď zdarma, nebo v placené verzi na bázi měsíčního předplatného. Neplacená verze má omezenou funkcionalitu a mezi hudbu jsou náhodně vkládány reklamní spoty, nutno však podotknout, že verze zdarma nijak neomezuje dostupnost více než 25 milionů skladeb. Placená verze poté samozřejmě neobsahuje reklamní spoty a přidává bonusovou funkcionalitu. Zejména možnost využití služby zdarma přilákala základnu větší než 140 milionů uživatelů, přičemž zhruba polovina z nich (tedy 70 milionů uživatelů) si službu předplácí a využívá tak prémiové verze. (Koutský, 2018)

Jak jsme se mohli dozvědět v předchozí kapitole, Henrik získal blízký vztah k hudbě již během svého dospívání, tedy předtím než se dal na dráhu softwarového inženýrství. Znalosti z obou oborů mohl následně zužitkovat ve společnosti Spotify při zavedení škálování agilního vývoje a rozšiřování agilních principů, kde působil jako jeden z agilních koučů. Ve Spotify pracoval mezi lety 2012 - 2016 a měl možnost být u změny této organizace a aktivně se podílet na jejím růstu ze společnosti s pár stovkami zaměstnanců na společnost s několika tisíci zaměstnanci.

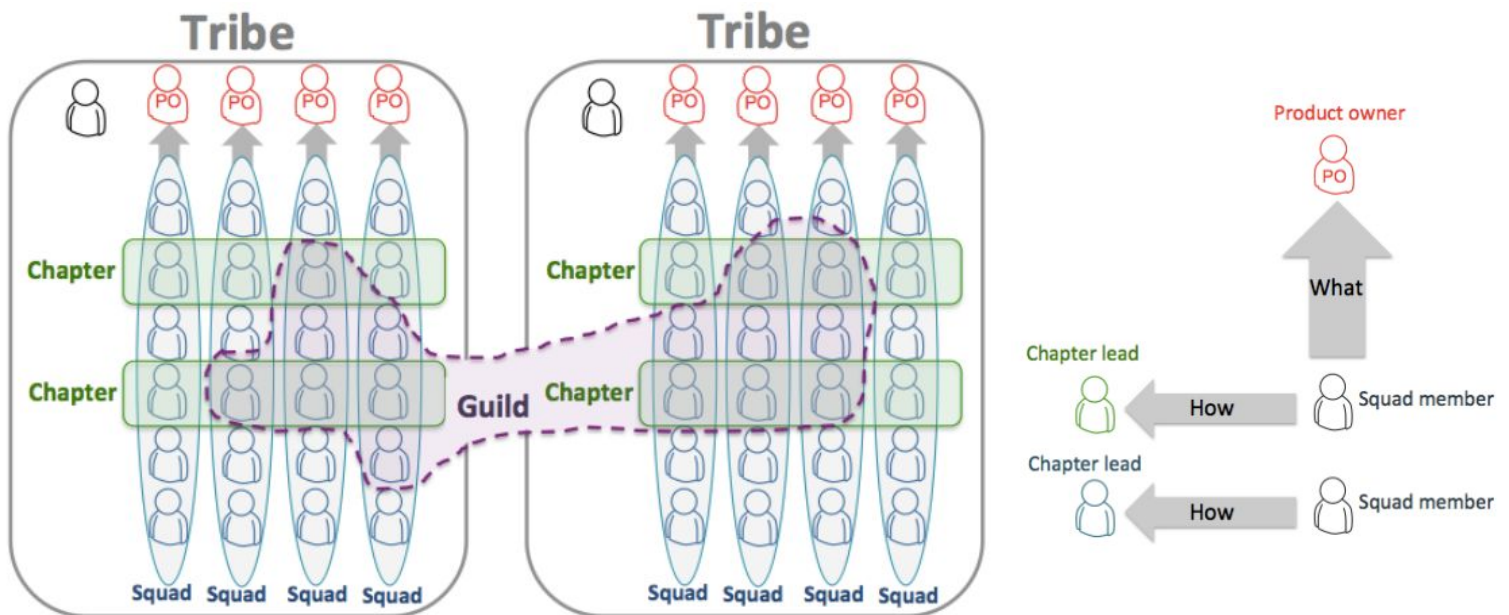
Pojďme si tedy blíže představit výsledky jeho počínání a strukturu organizace přizpůsobenou agilnímu vývoji, tak jak ji popsal v jednom z jeho článků (Kniberg a Ivarsson, 2012).

Základní prvkem celé organizace je tzv. Squad, který je podobný Scrum týmu a je navržen tak, aby působil jako takový mini-startup. Členové Squadu sedí společně v jedné kanceláři, mají veškeré potřebné dovednosti, schopnosti a nástroje pro návrh, vývoj, testování a nasazení software. Squad je do určité míry autonomní tým, který má vlastní způsob organizace a záleží jen na něm, který způsob si zvolí. Některé Squad týmy využívají čistý Scrum a jeho Sprints, některé využívají principů Kanban, některé mix obojího. Squad nemá určeného žádného vedoucího, každému Squadu je pouze přidělený Product Owner, který zodpovídá za prioritizaci vývoje a správu produktu daného týmu, nijak však nezasahuje do způsobu, jakým tým na úkolu pracuje. Dále jsou Squad týmům k dispozici agilní kouči dostupní ke konzultaci (paralela ke Scrum masterům). Zmíněným produktem daného týmu je myšlena jedna komplexní část software, která je dlouhodobou misí celého Squadu a jeho členové se tak mohou lehce stát experty právě v této oblasti. Na obrázku níže lze vidět možné rozdělení jednotlivých Squadů k různým částem software.



Rozdělení jednotlivých částí software je dále možno strukturalizovat pomocí tzv. Tribes. Tribe je nadřazený pojem pro Squad a typicky jeden Tribe obsahuje několik Squad týmů pracujících na produktu v podobné oblasti software. Tribe lze chápat také jako inkubátor mini-startupů, ke kterým jsme dříve přirovnali právě jednotlivé Squady. Každý Tribe má svého vedoucího, který se stará o zajištění co nejlepších podmínek pro práci jednotlivých Squadů. Tribe by dohromady neměl mít více než sto členů rozdělených do jednotlivých Squadů.

V tuto chvíli se lze pozastavit nad otázkou přílišné autonomie jednotlivých Tribes a Squadů. Představme si situaci, kdy se programátor jednoho Squad týmu potýká s problémem, kdy minulý týden vyřešil programátor jiného Squadu problém velice obdobný. Jaký smysl poté dává mít jednu společnou organizaci, jestliže v ní nedochází k horizontální koordinaci a sdílení zkušeností? Odpovědi na tyto otázky jsou právě poslední dva pojmy struktury organizace Spotify, kterými jsou tzv. Chapters a Guilds, které pomáhají spojovat organizaci dohromady v jeden celek.



Chapter je malá skupina lidí s podobnými schopnostmi, kompetencemi a pravomocemi, pracujících na obdobných pozicích v rámci různých Squadů, ale vždy stejného Tribe. Pod takovou Chapter si lze představit vývojáře, testery nebo grafiky spojující se napříč Squady za účelem sdílení zkušeností a diskuzí nad problémy. Každá Chapter má svého vedoucího pracovníka, který se stává přímým nadřízeným pro všechny členy této Chapter se všemi tradičními zodpovědnostmi jako je např. stanovení mzdy nebo zajištění školení, povolení dovolené apod. Každý takový vedoucí je však sám členem jednoho z Squadů a vykonává tak i své běžné povinnosti, čímž neztrácí kontakt s realitou. Oproti tomu Guild, který také spojuje lidi s podobnými schopnostmi a kompetencemi, vytváří spíše jakousi zájmovou skupinu. Guild není omezený jen na pracovníky na stejných pozicích, a dokonce ani na pracovníky v rámci stejného Tribe, navíc nemá žádného vedoucího. Přidat se do daného Guild může naprosto kdokoliv, kdo má o dané téma zájem a na ničem jiném nezáleží. Guild tak vytváří otevřený prostor skrze celou organizaci pro sdílení stejných zájmů, předávání zkušeností a vzájemnou diskuzi.

Každý zaměstnanec ve Spotify je tedy členem Squad týmu, v rámci kterého se dozví na čem má pracovat od daného Product Owenera, kdežto jak svoji práci dobře vykonávat mu poradí jeho Chapter, případně další Guilds, kterými může být členem.

4. Lego

Tato kapitola popisuje řešení agilního vývoje, která Lego Digital Solutions vedla k efektivnějšímu plánování, lepší koordinaci a vyšší spokojenosti zákazníků.

Lego hračka, kterou všichni známe a milujeme, začala s fyzickými hračkami a dodnes je to jejich hlavní produkt. Lego si uvědomil, že žijeme v digitální éře a snaží se překlenout tento rozdíl s Lego Digital Solutions. Lego DS si klade za cíl přinést technologii do společnosti vytvářením aplikací pro komunikaci mezi dětmi a rodiči. Agilní vývoj se stává normou a Lego, zejména Lego DS, to udržuje.



Na začátku, kdy bylo pouze 5 týmů, nebyly žádné problémy s koordinací a plánováním. Vzhledem k tomu, že počet lidí a týmů začal růst, problémy se začaly hromadit a bylo zřejmé, že je zapotřebí lepšího řešení. Dlouhodobé plánování a koordinace v rámci malých týmů byla stále v pořádku. Největším problémem byla střední část, která se většinou skládala z vlastníků produktů. Příliš často byli na schůzích všudypřítomně, zaplňování a neměli dostatečné zdroje k prioritizaci. To vedlo k větším problémům, jako je několik týmů pracujících na stejné věci, nedostatečná spokojenost zákazníků a nedostatečná schopnost připravit se na budoucnost, jako je vytváření výrobků, které by odolaly zkoušce času. Na konci roku 2014 se rozhodli provést některé změny. (KNIBERG a Brandsgård, 2016)

Jedním z řešení je uspořádání událostí Big Room Planning každých 8 týdnů. Tímto se zvyšuje efektivita týmů a ulehčuje proces upřednostňování vlastníků produktů. Na začátku tato událost trvala 2 dny, ale ve třetím čtvrtletí roku 2016 to bylo změněno pouze na jeden den. Skládá se z několika částí. Během Demo time se účastníci seznamují se všemi produkty, které byly uvolněny od poslední události. Bleskové rozhovory - Lightning talks se skládají z lidí, kteří dávají motivační rozhovory o všem, co se týká Lego Digital Solutions. Ty se pohybují od vážných, jako je bezpečnost a bezpečnost dětí online až po zábavné, jako jsou nadcházející hackatony a další

aktivity. Dalším bodem jsou Team breakouts. Chaotický a hlasitý, ale velmi účinný způsob, jak zakládat nové nápady do mysli akcionářů a plánovat práci, která bude provedena v příští iteraci. Ty se řídí takzvaným pravidlem o dvou nohách, což znamená, že pokud nepřispíváte nebo neučíte nic na místě, kde stojíte, pak se pohybujte 2 metry odtud a najděte místo, kde se učit budete. Plánované vytažení - Pull base planning je jméno pro velmi důležité jevy. Týmy se snaží pracovat, což znamená, že jsou si vědomi toho, co bylo učiněno během posledních iterací, chápají omezení a lépe předpovídají práci, kterou budou schopni dělat v nadcházející iteraci.

Během této akce existují důležité fyzické desky. Týmové desky začínají prázdné a postupně se naplňují během dvou dnů. Nakonec mají na příštích 8 týdnů high level plán. Zde je důležité poznamenat, že týmy nedodávají každých 8 týdnů. Někteří to dělají častěji, někteří méně. Riziková tabulka obsahuje seznam všech rizik. Některá rizika se během akce vyřeší, jiná potřebují, aby ho vyřešil top management. Závislostní deska je největší ze všech. Každý sloupec je tým, každý řádek je sprint (2 týdny). Poznámky představují závislosti. Zdá se, že nikdo nevládní závislost. Zprostředkovatelé to zpočátku rozvinuli, ale pak se týmy úplně sebeorganizují, vizualizují si vzájemnou závislost a používají desku jako formu protokolu handshake, aby zjistili, kdo s nimi musí mluvit. Zobrazují se pouze funkce se závislostmi. Nezávislé funkce jsou viditelné na příslušných týmových tabulích. Po setkání se deska převezme do kanceláře a přilepí se ke zdi. Scrum scrumů se provádí jednou nebo dvakrát týdně, aby se vyrazily vyřešené závislosti na desce. Stále existuje problém s týmy v Indii a pro ně snadný přístup k datům. Několik hodin po schůzce, facilitátor oznamuje, že je čas na veletrh návrhů (draft plan fair). To znamená, že pro jednoho člena z každého týmu jsou čtyři 7,5 minutové prezentace, aby mohl ostatním představit plán, zatímco ostatní členové týmu mohou jít a podívat se na jinou týmovou prezentaci. Podle jejich zkušeností tento formát funguje dobře, protože jen málo lidí potřebuje slyšet, co všechny týmy mají říct. Většinou se soustředují pouze na to, co je důležité pro jejich práci. Na konci dne většina lidí odchází, zatímco manažeři zůstávají. Rizikové desky se shromažďují v řadě poblíž velkého panelu závislostí a manažeři se shromáždí v půlkruhu. Všichni jsou ovlivňováni tím, co se děje s platformou, takže tam začínají. Majitel platformového produktu shrnuje svůj plán a vyvolává obtížné rozhodnutí co se musí dát pryč. Dále procházejí každou rizikovou tabule. Používají rámec ROAM pro zvládnutí rizik. Všechna rizika na tabuli musí být projednána a ROAM-ovaná: Resolved - vyřešena, Owned - vlastněna, Accepted - přijata, Mitigated - zmírněna.

Jednou z aktivit, které je třeba zdůraznit, je předběžné plánování PI, kde se majitelé produktů shromáždí a diskutují o prioritách funkcí pro nadcházející PI. Před každým plánováním PI existují tři takové předplánovací schůzky. Skutečným účelem události PI je, aby se týmy navzájem vyrovnaly. Hodnota takového typu plánování je přímo závislá na tom, kolik závislostí máte. Nejlepší je samozřejmě navrhnout strukturu vašeho týmu a architekturu, abyste minimalizovali závislosti a vyvarovali se nutnosti

plánovat takovým způsobem. Ale, pokud máte spoustu týmů pracujících na tomtéž produktu, pak je jediná cesta, jak jít dopředu. Trikem je vyhnout se pokušení vytvořit přesné a podrobné plány. Lepší je být spíše správně než určitě špatně.

Podrobnější details práce jsou diskutovány po plánování PI (během sprintu), ale kostra je vytvořena při velkém plánování. Další výhodou plánování PI je to, že vytváří radikální transparentnost a odhaluje vlastní složitost rozsáhlého vývoje softwaru. To může být trochu zstrašující pro externí stakeholdery, ale tím získávají lepší porozumění a větší respekt vůči týmům, protože jejich práce se ve skutečnosti snaží překonat celou už zmíněnou složitost vývoje.

Jaký dopad to dalo? Nic není dokonalé, ale celkový dopad byl překvapivě pozitivní.

- **Méně duplicitní práce**, Týmy se více vzájemně sladí, takže stráví méně času na nadbytečnou práci.
- **Méně problémů se závislostí**, Týmy ztrácejí méně času a spolupracují hladce s ostatními odděleními a stakeholdery.
- **Manažeři mohou aktualizovat priority a vyřešit překážky rychleji**, protože mají lepší představu o tom, co se skutečně děje.
- **Důvěra klientů se zlepšila**, protože mají lepší přehled o tom, na čem pracují týmy a proč.
- **Plánování je jednodušší a závazky jsou splněny častěji**, protože týmy a plánovači portfolia se dozvěděli, kolik práce se můžeme zavázne dodat a jaká je naše skutečná kapacita.



Důležité je zmínit, že se celkově zlepšila motivace členů týmu. Práce je větší zábava, když je méně zmatku a méně odpadu. A motivovaní lidé pracují lépe, takže je to pozitivní cyklus! Dalším vlivem, který bylo vidět, je, že další části LEGO navštíví setkání, získávají super inspiraci a začnou zkoumat, jak implementovat některé z těchto principů a postupů ve svých vlastních odděleních. Agilní přístup se ve skutečnosti šíří jako virus v rámci společnosti a vysoce viditelná povaha události plánování PI je katalyzátorem toho. (KNIBERG a Brandsgård, 2016)

5. Spotify vs. Lego

Spotify a LEGO spojuje v kontextu této práce především jedna věc, obě firmy se rozhodly pro škálování agilní metodiky v rámci jejich procesů vývoje a u obou jím byl mimo jiné pověřen Henrik Kniberg, nejhlavnější postava této seminární práce a také aplikování agilních metodik v těchto společnostech.

Spotify a LEGO se velmi liší již u svých produktů, firma Spotify se zaměřuje na svůj jediný produkt, kterým je streamovací služba s hudbou. LEGO je oproti tomu firma s mnohem delší tradicí a velikostí s kořeny ve výrobě fyzických hraček. Co je ovšem spojuje je definitivní snaha držet krok a touha využít k tomu agilních metodik.

Během škálování agilu do firmy Spotify se vycházelo ze stejných základů jako při tvorbě metodiky SAFe, která je však více deskriptivní. Nejnižší jednotka, na kterou se dá organizace Spotify v rámci agilního řízení rozpadnout, je Squad, z jednotlivých Squadů se pak stávají Tribes. Tato metodika podstatně více vychází ze standardní start-upové filozofie oproti filozofii uplatněné ve firmě LEGO.

Pro zajištění větší spolupráce ve firmě jsou dále ve Spotify zavedeny Chapters a Guildy, které v podstatě simulují jakési spolky a odborné skupiny s určitými zájmy a skill sety.

Ve firmě LEGO se při škálování agilních metodik vycházelo především ze SCRUMu, vzhledem k tomu, že se však na rozdíl od Spotify v LEGO Digital Solutions dělá na více produktech zároveň a část vývoje sídlí i v Indii, byla velká potřeba metodiku dostatečně ohnout a přizpůsobit firmě o takové velikosti jako je LEGO, metodika má však stále jisté vady z důvodu umístění části týmu v Indii, a tak i nadále zůstává prostor pro zlepšení.

Metodika použitá v LEGU podstatně více využívá role Manažerů, na které připadá podstatně větší břímě, než v případě Spotify, veškeré plánování na jednotlivé sprinty pro jednotlivé týmy pak v LEGU také probíhá na velké fyzické Kanbany.

Týmy si tuto metodiku chválí a jsou spokojeni, neboť práce je nyní přehlednější a vzniká méně chaosu na pracovišti a mezi zaměstnanci, díky tomu jsou v práci zaměstnanci také spokojenější a výkonnější. Sprinty probíhají v LEGU standardně na dva týdny.

Vedoucí v metodice pro LEGO také tráví více času plánováním a čištěním desek a starých tasku, ta je však vzhledem k velikosti firmy, struktuře a většímu množství produktů poměrně logické.

Jedním z největších rozdílů je také skutečnost, že LEGO používá Big Room plánování, zatímco v Spotify tomu tak není. Pro LEGO je to spíše důležitá část agilního vývojového procesu, ale Spotify se spoléhá na jiné techniky.

Existuje jiná organizace týmů, pokud jde o tyto dvě společnosti. Ve Spotify mají Chapters a Guilds, zatímco v LEGO mají vývojové týmy odpovědné za určité části procesu vývoje. LEGO má méně hierarchie než Spotify, což je vidět v přístupu, jak tyto dvě společnosti organizují své týmy a vedení týmů.

Je také možné vidět, že oba přístupy upřednostňují důvěru napříč týmy a rychlé sdílení všech cenných informací, jako jsou detaily o projektech, na kterých pracují v současné době. Všechno to se dělá, aby se zabránilo duplicitní práci, ztrátě času a plýtvání zdroji. Způsoby dosažení tohoto cíle jsou různé. Lego má přímý přístup v Big Room plánování, zatímco Spotify se snaží sdílet technický know-how přes Chapters a Guilds. Obě společnosti se snaží vyhnout "centrálním kontrolním systémům" a prosazovat transparentnost a sebeorganizaci. Lego řeší záležitosti týkající se závislosti během Big Room plánování. Spotify se pokouší použít oddělení pro snížení potřeby synchronizace týmů. Každý tým ve Spotify je zodpovědný za samostatný kus práci nebo určitou funkci softwaru.

Posílení morálky, poslouchání a oceňování názoru každého člověka je také jádrem obou přístupů. Obě společnost se přes škálování agile snaží co nejlépe využít hard a soft skilly svých pracovníků a zároveň udělat jim práci zajímavou aby byli co nejvíce efektivní. Škálování není snadný úkol, ale může být dosaženo, pokud existuje transparentnost, důvěra a rychlá zpětná vazba.

Závěr

Náš cíl popsání škálování agilního vývoje v obecnější rovině a poté detailně ve firmách Spotify a Lego, kde je představujeme pohledem Henrika Kniberga byl úspěšně splněn. V první části jsme se seznámili s agilní metodikou, o čem je, jak zní manifest agilního vývoje a o čem je vlastně škálování agilních metodik vývoje.

V další části textu jsme se poté podívali na historii Henrika Kniberga z trošku větší blízkosti a popsali si jeho život, to nám poté pomohlo k pochopení, proč si na příklad vybral některé ze svých zaměstnavatelů a získali jsme tak i větší přehled o jeho dosavadních zkušenostech.

Po úvodu do škálování agilních metodik a seznámení se s Henrikem Knibergem jsme zbytek času věnovali studiu jeho práce pro firmy Spotify a Lego. Na těchto jeho životních kapitolách bylo velmi zajímavé sledovat, jak přistupoval k agilnímu vývoji v jednotlivých případech a jak se z této metodiky dnes pomalu stává standardní způsob vývoje jak u mladších firem jako je Spotify, tak u zaběhlých velikánů jako je LEGO. Vzhledem k rozdílným produktům a charakteristice obou firem se uplatnil poměrně rozdílný způsob škálování agilních metodik vývoje, v obou případech se však setkal s úspěchem. Lego se dokonce rozhodlo přejímat vybrané metodiky agilního vývoje i do dalších částí a oddělení firmy, které nebyly v první části implementace zahrnuty. Oba přístupy jsou povedené a je zajímavé, jak se strukturálně odlišují.

V poslední kapitole jsme se poté věnovali právě rozdílům ve škálování agilních metodik mezi těmito dvěma výše zmíněnými společnostmi a snažili se popsat ty nejzásadnější rozdíly a co naopak jednotlivé přístupy spojuje. Během tvorby této práce jsme se neseťkali s žádným větším omezením nebo problémy, bohatým zdrojem pro nás byly jak informace z internetu, tak vybraná literatura. Za zmínku také stojí velmi dobře zpracovaný osobní blog Henrika Kniberga.

Škálování agilního vývoje a agilní vývoj obecně jinak považujeme za velmi zajímavou tematiku a myslíme si, že toto téma zůstane vysoce relevantní i v nadcházejících letech, kdy bude díky svým pozitivním výsledkům dále nabírat na své popularitě.

Literatura

BECK, Kent et al., 2001. *Agile Manifesto* [online]. [cit. 2018-12-10]. Dostupné z: <https://agilemanifesto.org/>.

CRISP, 2018. *Agile & Lean coaching portfolio* [online]. [cit. 2018-12-09]. Dostupné z: <https://www.crisp.se/konsulter/henrik-kniberg>

DENNING, Steve, 2017. *What does it mean to scale agile?* [online]. [cit. 2018-12-10]. Dostupné z: <https://www.forbes.com/sites/stevedenning/2016/04/15/what-does-it-mean-to-scale-agile>

DOLEŽAL, Jan et al., *Projektový management Komplexně, prakticky a podle světových standardů*. Grada Publishing, a.s., 2016. 418 s. ISBN 978-80-247-5620-2.

HUBENÝ, Michael, 2018. *Škálování agilního vývoje* [online]. [cit. 2018-12-10]. Dostupné z: https://is.muni.cz/th/px4ms/dp_michael-hubeny.pdf

KNIBERG, Henrik, 2018. *Henrik's homepage: My life story* [online]. [cit. 2018-12-09]. Dostupné z: <http://www.kniberg.com/henrik/lifeStory.jsp>

KNIBERG, Henrik a Eik Thyrsted Brandsgård. *Scaling Agile @ Lego: Planning as a Social Event*. [online]. 2016 [cit. 2018-10-09]. Dostupné z: <https://crisp.se/wp-content/uploads/2016/12/Agile@Lego.pdf>

KNIBERG, Henrik a Anders IVARSSON. *Scaling Agile @ Spotify: with Tribes, Squads, Chapters & Guilds* [online]. 2012, 14 [cit. 2018-10-09]. Dostupné z: <https://blog.crisp.se/wp-content/uploads/2012/11/SpotifyScaling.pdf>

KOUTSKÝ, Zdeněk, 2018. *Spotify si měsíčně předplácí více než 70 milionů uživatelů* [online]. [cit. 2018-12-09]. Dostupné z: <https://dotekomanie.cz/2018/01/spotify-si-mesicne-predplaci-vice-nez-70-milionu-uzivatelu/>

MACDONAGH, Mike, 2012. *What does "Agile at Scale" mean?*[online]. [cit. 2018-12-10]. Dostupné z: <https://mikemacd.wordpress.com/2012/05/02/what-does-agile-at-scale-mean/> .

SCRUM ALLIANCE, 2018. *Biography - Henrik Kniberg* [online]. [cit. 2018-12-09]. Dostupné z: <https://www.scrumalliance.org/community/profile/hkniberg>

WATTS, Stephen, 2017. *Scaled Agile Framework Explained* [online]. [cit. 2018-12-10]. Dostupné z: <https://www.bmc.com/blogs/scaled-agile-framework-safe-explained/>.