

Semestrální práce ke kurzu 4IT421 Zlepšování procesů budování IS	
Semestr	LS 2016/2017
Autoři – jméno, příjmení , xname	David Haška – xhasd04 Jan Černý – xcerj136 Petr Adámek – xadap18
Téma	An Introduction to Modern Agile
Datum odevzdání	12.3.2017

Abstrakt

Tato práce se zabývá popisem a vysvětlení fungování všech principů a praktik Modern Agile. Dále se tato práce zabývá porovnáním Modern Agile s Manifesto for Agile Software Development.

Cílem bylo popsat nový způsob vývoje produktu pomocí Modern Agile a následné porovnání s Manifesto.

Cíl práce byl naplněn úspěšným popsáním všech principů a praktik včetně porovnání s Manifesto. Popis se zaměřil především na praktické přirovnání k ukázkám.

Klíčová slova

agilní, metodiky, vývoj, testování, principy, praktiky, disciplína

Obsah

1. Úvod.....	1
2. Modern Agile	2
2.1. Co je na Modern Agile tak moderního?	2
3. Disciplíny, principy a praktiky.....	3
3.1. Disciplíny	3
3.2. Dvanáct principů a praktik.....	5
4. Zavádění Modern Agile do praxe	9
5. Porovnání s Manifesto for Agile Software Development	10
6. Budoucnost Agilních metodik	11
7. Závěr	12
8. Literatura.....	13

Seznam obrázků

Obrázek 1	Modern Agile, 4 principy	3
Obrázek 2	Přehled principů a praktik v návaznosti na disciplíny.....	6
Obrázek 3	Modern Agile radarový graf	9

1. Úvod

Agilní metodiky umožňují pružnou a flexibilní práci. Tento soubor principů se zabývá novým způsobem pojetí vývoje pomocí agilních metodik. Tato práce čerpá především z konference o Modern Agile, ze které následně vychází všechny zdroje o tomto tématu.

Cílem této práce je popsat nový způsob vývoje produktu pomocí Modern Agile a následné porovnání s Manifesto for Agile Software Development.

Cíle dosáhneme pomocí rešerše a analýzy článku o Modern Agile a také videa z konference o těchto principech. Analýza umožní pochopit a podrobně popsat celý soubor principů a praktik.

2. Modern Agile

Máme dost metodik a principů, jak řídit vývoj produktu, tak proč zvolit právě Modern Agile? Důvod je jednoduchý. Máme zde jasný a stručný popis pouze 4 principů a 12 dobrovolných praktik, které stmelují tyto principy dohromady. Vývoj probíhá velice rychle, protože se kladě veliký důraz na jednoduchost. Zpravidla zde nejsou žádné role či zodpovědnosti, zbytečné postupy typu stand-up setkání, rámce nebo nutnost certifikátů pro vykonávání dle těchto principů.

2.1. Co je na Modern Agile tak moderního?

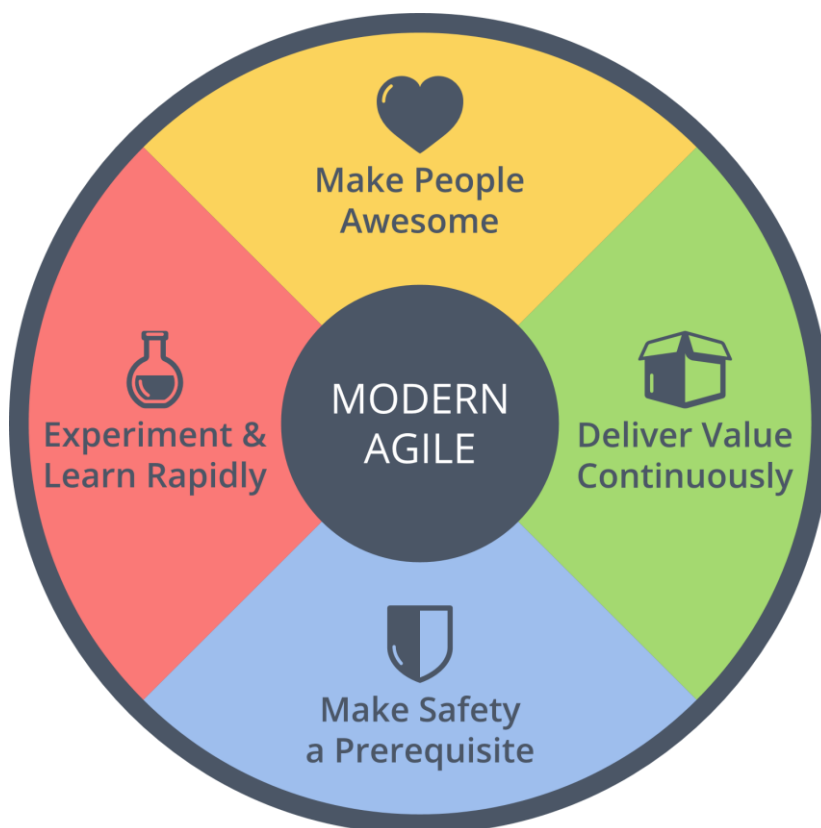
Moderní na něm je to, že doopravdy vychází z minimalismu, tedy odstranění postupů a nástrojů, které zbytečně prodlužují a zpomalují vývoj produktu. Z hlediska významu slova moderní se tedy jedná o pokrokové principy, protože žádné jiné předtím nedefinovalo vývoj takto jednoduchým způsobem. Dokumenty nehrají žádnou roli a klade se důraz na jednoduchost, rychlost a bezpečnost.

3. Disciplíny, principy a praktiky

Modern Agile je definován čtyřmi disciplínami. Disciplíny snižují riziko podnikání, zlepšují efektivnost zdrojů a odstraňují omezení zaměstnanců. Tyto disciplíny stanovují oblasti, do kterých tematicky spadají principy a praktiky. Principy a praktiky přesněji určují, jak splňovat Modern Agile a získat z něj tak výhody, které přináší.

3.1. Disciplíny

Obrázek 1 Modern Agile, 4 principy



Modern Agile, 4 principy (online, ModernAgile.org, 2016)

3.1.1. Pomozte lidem stát se excelentními (Make Users Awesome)

Produkt či službu můžeme chápat jako ekosystém na jejímž konci je vyřešený problém zákazníka a tím pádem i spokojený zákazník.

Na tomto stojí princip „Pomozte lidem stát se excelentními“. Tento princip je inspirovaný blogem Kathy Sierra. Cílem je činit našeho zákazníka úžasného naším produktem či službou nikoliv mít dobrý produkt či společnost. To uděláme tak, že najdeme problém, které je drží zpět a uděláme takové změny, aby náš zákazník dosáhl úžasných výsledků.

To je ovšem jenom část ekosystému. Součástí ekosystému jsou i lidé, kteří službu či produkt vytvářejí. Aby dělali svoji práci dobře musí být spokojení i oni. Moderní agilní vývoj bere tedy v potaz celý ekosystém – prodejce, vývojáře, zákazníky.

K tomu se váže i práce na zaměstnancích. Je potřeba, aby se vývojář stále zlepšoval. Zlepšování je dobré, pokud známe cíl a víme, jak k němu jít. Cíl bez cesty nemá smysl. Například Japonsko roku 1964 za cíl vytvářet super rychlé vlaky, ale nevěděli jak ani neměli technologie.

Další je pak přemýšlet, zda inkrementálně zlepšovat nebo po velkých částech, protože velké zlepšení nás nutí i začít jinak přemýšlet.

Amazon vzal roku 1997 princip tzv. „Zákazníkovu posedlost“ a začal ho používat jako jeden ze svých vůdčích principů. To ukazuje, že pokud uděláš zákazníky skvělými, mají tendenci se přirozeně stát propagátorem vašich produktů nebo služeb.¹ (online, Kerievsky, 2016)

Přínos této disciplíny:

- Přirozenou reklamu produktu či služby
- Dlouhodobou udržitelnost
- Kvalitní produkt či službu

3.1.2. Bezpečné prostředí je prerekvizitou (Make safety a prerequisite)

Pocit bezpečí je základní potřeba člověka. Pokud není tato potřeba naplněna, lidé nepracují naplno. Modern Agile stanovuje jako jeden ze základních kamenů výkonosti pocit bezpečí. Pocitem bezpečí je myšlen jako ochrana času lidí, jejich zdraví, peníz, reputace a vztahů.

Lidé se bojí změn, říct své názory a myšlenky a také se bojí dělat chyby. Tento strach snižuje jejich výkon. Pokud má podniková kultura v sobě strach, žádný nový proces nebo praktiky nepomohou.

Tento strach není přímo strach ze selhání, ale především strach z viny. Vina nemá žádný pozitivní výsledek a nijak nepomáhá.

Chyba dle Modern Agilu má být chápána jako výsledek nezmapovaného problému v prostředí. Tato chyba má být pak jako podklad pro zlepšení. Budoucí zlepšení může vést ke zmenšení rizik a chyb a ke snížení ztrát.

„Bezpečné prostředí je prerekvizitou“ znamená stanovení bezpečnostních pravidel před zahájením samotné práce. Pokud se bezpečnost nezlepší, učíme se ze selhání a získané zkušenosti aplikujeme do budoucích pravidel pro další práce.

Příklad z praxe:

„Paul O’Neill, CEO ve firmě Alcoa, jednou řekl „Priority se mění, když se mění naše zaměření, ale bezpečnost nikdy. Bezpečnost je podmínkou, nikoliv prioritou.“ Lidé nepracují dobře, když žijí ve strachu. Když je někdo vyhozen nebo penalizován, ostatní se začnou bát, že je potká stejný osud a produktivita klesá.“²(online, Kerievsky, 2015)

Přínos této disciplíny:

- Větší výkonnost zaměstnanců
- Lepší kvalita výsledků
- Neustálé zlepšování a udržitelnost

¹ Amazon has made “Customer Obsession” a guiding principle since 1997 and it shows. If you make customers awesome, they tend to be natural promoters of your products or services.

² Paul O’Neill, former CEO of Alcoa, once said, "Priorities change while our focus on safety never changes. Safety is a prerequisite, not a priority." People don't function well when they live in fear. If someone is fired or penalized for a mistake, everyone becomes afraid of a similar fate and productivity plummets.

3.1.3. Zkoušet a rychle se učit (Experiment a learn rapidly)

Roku 1959 byla vypsána soutěž na vytvoření letadla poháněné lidskou silou schopné se pohybovat do všech 8 směrů. 18 let to nikdo nezvládl, až do doby, kdy to začal řešit MacCready. Jeho první myšlenkou bylo, že problém je, že nevíme, jaký je vlastně problém. Následně navrhl proces, díky kterému mohl rychle a bezpečně opakovat problém letu. Vytvořili lehký prototyp, který byl bezpečně ukotven a lítal nízko nad zemí. Díky tomu mohl on a jeho tým rychle testovat jejich poznatky, učit se z výsledků a výsledné zkušenosti aplikovat do nových testů. Zatím co ostatním trvalo dlouho (v týdnech až měsících) než uskutečnili další test on mohl testovat v řádu hodin. Tento proces mu umožnil splnit zadání do 1 roku.

Zkoušení a rychlé učení nás chrání před ztrátou času a umožňuje rychlejší objev úspěchu. Možnost zkoušet nám umožňuje pocit možnosti selhat. K úspěšnému podnikání stačí jen pár úspěchů, které mohou pramenit z řady neúspěchů

Přínos této disciplíny:

- Rychlejší nalezené řešení
- Ušetření nákladů
- Větší důvěra a prostor pro selhání

3.1.4. Průběžně doručovat hodnotu (Deliver value continuously)

Disciplína „průběžně doručovat hodnotu“ znamená, vytvářet malé samostatné části, které lze dodat, nasadit nebo aplikovat. Nejedná se pouze o dodání nějaké hodnoty pro zákazníka, ale může se jednat i pouze o dodávku prototypu. Na prototypu pak můžeme ověřovat co se zákazník líbí a co ne, zda daný nápad má uplatnění pro firmu nebo dokonce, zda námi navrhnutá cesta bude mít požadované výsledky nebo je třeba ji změnit.

Průběžné doručování hodnoty umožňuje flexibilnější práci. Zákazník tuto flexibilitu může pak vidět při rychlé opravě chyb. V interním systému firmy se flexibilita může jevit, lepším zapracováním nového zaměstnance.

Problém u nových zaměstnanců je, že firma nezná jeho schopnosti. Vzniká mezi „nováčkem“ a firmou nedůvěra, která potřebuje čas na to, aby zmizela. Při aplikaci této disciplíny je „nováček“ přiřazen k mentorovi, který dělá pouze na malých částech (nová funkcionality, oprava chyb) a na nich může „nováčkovi“ ukázat co a jak má dělat. Díky tomu, je pak nováček, místo týdnů a měsíců, samostatně pracovat v řádech dní. Takto pracují zaměstnanci ve firmě AirBnB.

Tato disciplína může napomáhat při optimalizaci zdrojů. Průběžná dodávka způsobuje relativně stálé používání zdrojů a díky tomu můžeme snížit náklady o přebytečné. Příkladem bych uvedl zrušení pronájmu nadbytečných server.

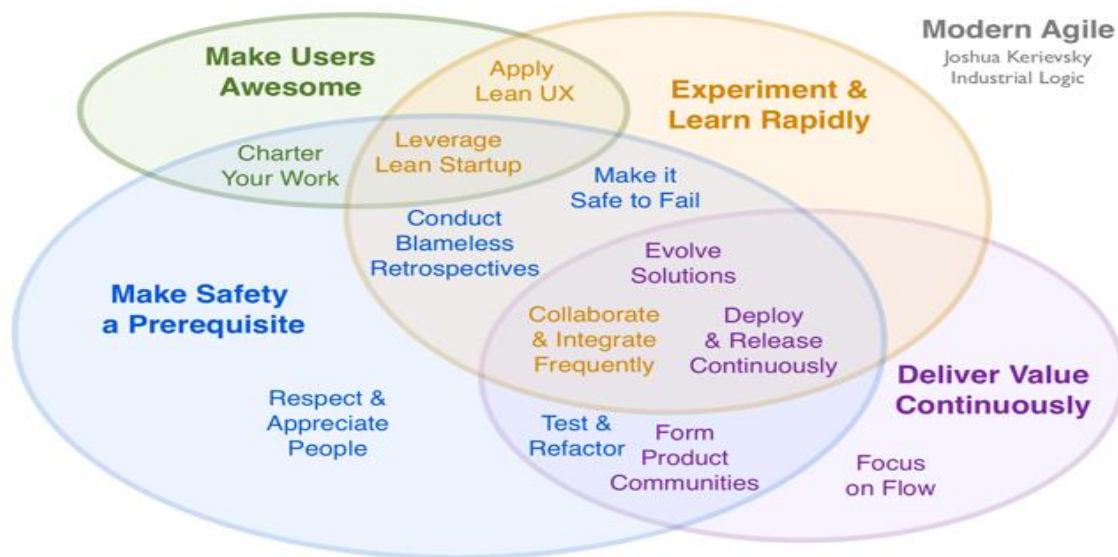
Přínos této disciplíny

- Rychlé náhledy na budoucí výsledky
- Spokojený zákazník
- Rychlé zaučení nováčků
- Možnost optimalizace zdrojů
- Odhalení možných špatných výsledků

3.2. Dvanáct principů a praktik

Principy a praktiky jsou součástí Modern Agile. Je to konkrétnější praktikování disciplín, přičemž mohou náležet do jedné nebo více disciplín.

Obrázek 2 Přehled principů a praktik v návaznosti na disciplíny



Zdroj (online, Kerievsky, 2015)

3.2.1. Zmapuj si svojí práci (Charter Your Work)

Spadá do disciplín: *Pomozte lidem stát se excelentními a Bezpečné prostředí je prerekvizitou*

Musíte si stanovit, co pro vás znamená udělat člověka úžasným, abyste to mohli splnit. Mapování nám umožňuje vyčistit a upřesnit vizi, naše poslání a dovoluje nám testovat naše výsledky. Nemapujeme pouze na začátku, ale musíme mapovat po celou dobu, abychom neztratili námi zvolenou cestu a zůstali na ni.

3.2.2. Využijte „Lean Startup“ (Leverage Lean Startup)

Spadá do disciplín: *Pomozte lidem stát se excelentními, Bezpečné prostředí je prerekvizitou, Zkoušet a rychle se učit*

Technika Lean Startup nám umožňuje objevit co máme pro zákazníka vytvářet a zároveň nám ukazuje jak rychle a jednoduše naše hypotézy ověřit předtím, než vyrobíme plně funkční řešení. Tato technika je možná aplikovat na malé i velké projekty.

3.2.3. Používej „Lean UX“ (Apply Lean UX)

Spadá do disciplín: *Pomozte lidem stát se excelentními a Zkoušet a rychle se učit*

Lean UX vychází z metody Lean Startup. Na rozdíl od Lean Startupu nám nepomáhá vyhledat problém zákazníka a vytvářet řešení, ale ověřuje použitelnost našeho řešení. Rychlá, levná a metoda, která je řízená výsledky analýz.

3.2.4. Spolupracujte a často propojujte řešení (Collaborate & Integrate Frequently)

Spadá do disciplín: *Zkoušet a rychle se učit, Bezpečné prostředí je prerekvizitou, Průběžně doručovat hodnotu*

Slabá spolupráce v týmu nebo mezi týmy vytváří riziko pro firmu. Pokud máme šikovné pracovníky, kteří nejsou příliš „týmovými hráči“, vzniká koncentrace vědomostí. Další členové týmu nejsou schopni opravovat větší chyby a řešit některé problémy. Zároveň to způsobuje zatížení jednoho člověka a nezatížení pro ostatní, které může vést k nespokojenosti a ztrátě produktivity.

Řešení pro tyto problémy je například párové programování, kdy dva lidé pracují společně a vyměňují se v řešení problémů. Dalším řešením je „Mobbing“ pro více lidí. Skupina opět pracuje dohromady a členové se střídají v pozici, ve které je vykonáváno jejich řešení.

Obě tyto metody zvyšují průběh učení jednotlivých členů, zvyšuje rychlost řešení problému a snižuje riziko závislosti na omezeném počtu lidí. Také umožňuje aplikaci nových technologií jelikož, může najmout odborníka, který zaučí zbylé zaměstnance.

3.2.5. Bezpečné selhání (Make it Safe to Fail)

Spadá do disciplín: *Bezpečné prostředí je prerekvizitou, Zkoušet a rychle se učit.*

Zpravidla se jedná o to, že nejlepší firmy podporují firemní kulturu takovým způsobem, že se nikdo z lidí nebojí selhání nebo také nesouhlasu. Dalo by se říci, že jde v podstatě o bezpečné experimentování v rámci zlepšování hodnot ve firmě.

Známý konzultant v prostředí managementu, pan Dave Snowden tvrdí, že pomocí experimentů v malém měřítku, můžeme docílit hodnotných lekcí v kontextu složitého systému.

Společnosti, které pracují s praktikou bezpečného selhání jsou většinou vzdělávací organizace. Ukázkovým příkladem je firma Etsy, kde nově přijatý zaměstnanec omylem smazal kritický CSS³ soubor a snížil tím významně rychlost načítání webových stránek. (online, Kerievsky, 2015)

3.2.6. Testování a "refactoring" (Test & Refactor)

Spadá do disciplín: *Bezpečné prostředí je prerekvizitou, Průběžně doručovat hodnotu.*

Testování a "refactoring" je součástí všech z agilních metod. Testování je velmi důležitou součástí vývoje každého produktu. V jedné nezávislé studii se došlo k závěru, že testování v Modern Agile došlo ke snížení chybovosti o 80%, což vede k jakýmsi pocitu bezpečí, kde vývojáři se mohou více soustředit na samotný vývoj, aniž by produkovali spousty chyb. Testování se provádí na základě automatických a manuálních testů. Vývoj řízený testy je tedy samozřejmostí.

"Refactoring" jako nedílnou součástí zjednodušování kódu je systematický proces zlepšování změn v kódu bez ovlivnění funkčnosti. V zásadě se jedná o zpřehlednění kódu drobnými změnami. (online, Hájek, 2013)

V průběhu vývoje a testování zjišťujeme mnoho nedostatku, které má právě vylepšit proces "Refactoring". Obojí, tedy testování a "refactoring" jsou nezbytné pro rychlý vývoj v týmech a proto je nedílnou součástí Modern Agile, který má být moderní a v moderním světě má být vše, co nejdříve.

3.2.7. Respektování a oceňování lidí (Respect & Appreciate People)

Tato praktika patří do disciplíny: *Bezpečné prostředí je prerekvizitou*

Pocit bezpečí je zde založen na tom, že jsou všichni lidé ve firmě a v rámci firemní kultury respektování a oceňování bez ohledu na pozici či roli ve firmě, vzdělání, věku, rasy nebo pohlaví. Pocit bezpečí může

³ Zkratka Cascading Style Sheet a jedná se o jazyk, který určuje způsob zobrazení webových stránek napsaných v HTML

vyvolat třeba vrácení peněz za službu, kterou jsme nemohli využívat z důvodu nedostupnosti, třeba výpadek serveru.

Paul O'Neill, bývalý generální ředitel společnosti Alcoa, uvedl, že firma, která chce dosahovat skvělých výsledků, respektive výkonů od svých zaměstnanců, tak musí mít firemní kultura, která je nastavena tak, že jsou lidé všude a každý den respektování a oceňování.

Oceňování může probíhat na základě výkonů v podobě finančních odměn, ale také jiných materiálních či nemateriálních odměn, které způsobí to, že jsou lidé stále motivováni.

3.2.8. Conduct Blameless Retrospectives

Patří do disciplín: Bezpečné prostředí je prerekvizitou, Zkoušet a rychle se učit.

Retrospektiva je způsob v agilních metodách, jak zjistit, v čem jsme byli dobří a v čem naopak ne. Důvod je ten, že nechceme, aby se věci, které se nám nelíbily a naopak, abychom mohli prosazovat věci, které se nám líbily a které můžeme využít v náš prospěch v rámci úspěšného vývoje produktu.

3.2.9. Zaměření na plynulost (Focus on Flow)

Patří do disciplín: Průběžně doručovat hodnotu.

Modern Agile preferuje plynulý průběh všemi časovými bloky (jako např.: sprinty, iteracemi). K tomu využívá Kanbanu, který umožňuje plynulý průběh prací. Neurčuje se tedy pracnost jednotlivých úkonu, jaké množství času bude potřeba k danému úkolu, kolik práce člověk stihne, ale pracuje se kontinuálně.

3.2.10. Nasazovat a uvolňovat řešení průběžně (Deploy & Release Continuously)

Patří do disciplín: Průběžně doručovat hodnotu, Bezpečné prostředí je prerekvizitou

Díky průběžnému nasazování řešení se můžeme velmi brzo testovat produkce (např. A/B testování), to umožňuje bezpečně doručovat hodnotu uživatelům, jak nejrychleji je to jen možné.

3.2.11. Vytvářet komunitní produkty (Form Product Communities)

Patří do disciplín: Průběžně doručovat hodnotu, Bezpečné prostředí je prerekvizitou

Bez správných lidí, nemůže společnost vytvořit kvalitní softwarové řešení, které by dělalo uživatele úžasnými. Správní lidé vytvářejí komunitu vašich produktů. Komunita vývojářů není rozdělena do front-end, back-end, middleware a UX týmů, každý s vlastními nedodělkami. Komunita má plnou zásobu nápadů, díky kterým je připravena vyvíjet primitivní řešení a začlenit je do sofistikované nabídky, díky níž se stanou uživatelé úžasní a pomůžou organizaci uspět. Vytvoření skupiny produktů pomáhá lidem získat povědomí o tom, kdo co dělá a jak jsou nezbytné pro dosažení klíčových výsledků. Definování a zdokonalování produktové komunity se často stává během pronájmu.

3.2.12. Vyvíjet řešení (Evolve Solutions)

Patří do disciplín: Průběžně doručovat hodnotu, Podmínka bezpečí, Zkoušet a rychle se učit.

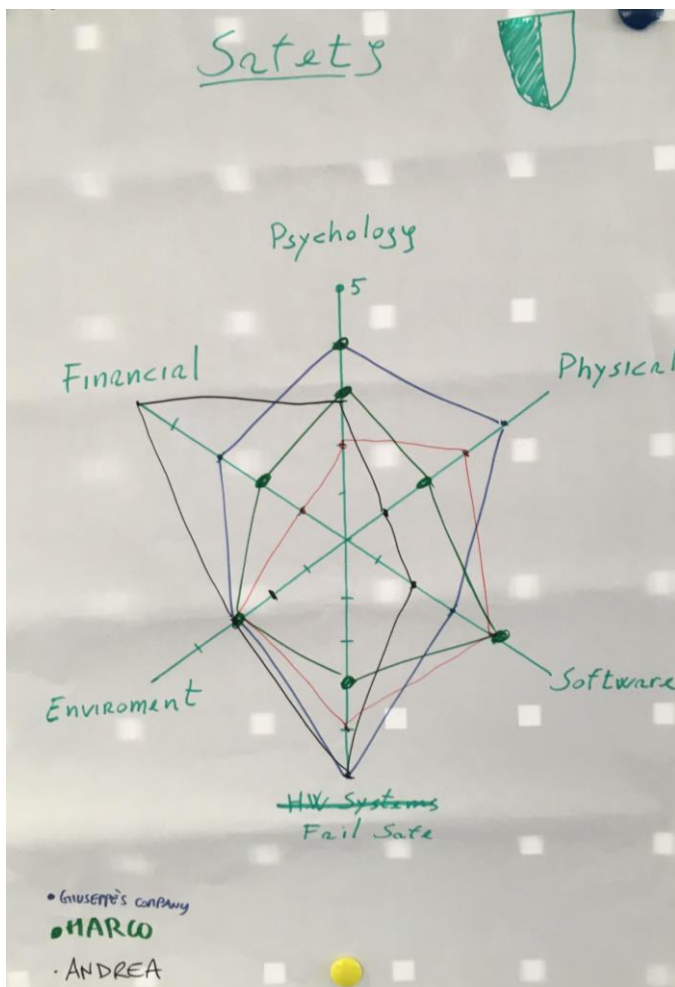
Pokud jde o objevování rychlých řešení, je nutné si nejdříve zorganizovat lidi a týmy, vytvořit plán co nejdříve udělat, jak spolupracovat, integrovat, vyvinout a uvést do praxe. Například společnosti založené na principech Modern Agile jsou schopni uvést produkty velmi rychle a díky tomu jsou schopni naopak dostat rychle zpětnou vazbu, ze které se můžou poučit.

4. Zavádění Modern Agile do praxe

Jak praktikovat principy do praxe je nejlépe vidět na příkladu napsání knihy. Pokud se budete řídit principy Modern Agile, rychle přijdete na to, jestli kniha bude pro lidi přínosná nebo je dělat „skvělými“. Nejednalo by se o bezpečnou investici času, pokud by o knihu nebyl zájem. Takže pokud chcete zjistit, jestli je skutečně zájem o téma, o kterém chcete psát, musíte si to nejdříve na něčem otestovat a následně se z toho poučit. Abyste to mohli zjistit, musíte lidem dát něco přečíst, zjistit jejich názor, v případě knihy je nejlepší volbou začít psát o daném tématu blog, psát články nebo napsat pár kapitol. Tyto texty musíte dostat mezi lidi a snažit se od nich získat zpětnou vazbu. Čím více budete psát a budete mít více kapitol, stránek do své knihy, tím víc si budete dávat záležet na bezpečnosti, když budete dělat s někým rozhovor, tak si ho nahrajete na dva diktafony, budete si vytvářet zálohy.

Jednou z dalších možností, která může pomoci v zavádění principů Modern Agile je využití *Modern Agile radarového grafu*, díky němu lze pak měřit stav v principu podmínka bezpečí (Make safety a prerequisite). Kde si tým může stanovit kritéria podle, který bude hodnotit u bezpečnosti to mohou být například fyzická, psychická, softwarová, environmentální, finanční bezpečnost. Na radarovém grafu je stanovena škála 0-5 pro každý element a tým rozhodne jakého skóre kde kdo dosáhl, výborně se pak na takovém grafu dá pozorovat zlepšení.

Obrázek 3 Modern Agile radarový graf



Modern Agile radarový graf (online, Kerievsky, 2015)

5. Porovnání s Manifesto for Agile Software Development

Celou dobu se v rámci této seminární práce mluvilo o nějakém vývoji produktu. Zvolení pojmu produkt, a nikoliv software nebo aplikace je naschvál, protože principy a praktiky Modern Agile se soustředí na vývoj kteréhokoliv produktu. V zásadě můžeme zmínit knihu a její následný prodej, ale je důležitý říci, že to lze aplikovat také na vývoj software či aplikace.

V samotném procesu celého vývoje se tyto soubory principů tolik neliší. Můžeme zde najít spousty podobných rysů nebo dokonce principů či praktik, které se liší pouze názvoslovím. Můžeme tedy říci, že si tyto soubory jsou velice podobné. Příkladem může být disciplína Zkoušet a rychle se učit, která odpovídá principu "Reagování na změny" v souboru principů Manifesto for Agile Software Development. (online, Kerievsky, 2016b)

Hlavní a nejdůležitější rozdíl mezi Modern Agile a Manifesto for Agile Software Development je ten, že Manifesto je určen pouze pro vývoj aplikací a software, jak již vyplynulo z předchozího odstavce.

6. Budoucnost Agilních metodik

Zatímco agilní metodiky se modernizují, tak hlavní agilní uživatelé ve vývojářském průmyslu se stále zaseknutí v líných procesech, jsou svázáni byrokratickými nástroji, frameworky a certifikacemi.

Podle Joshui Kerievského (2016b), mladí uživatelé si zatím myslí, že Agilní metodiky jsou jen pro jejich otce. Otcové si zase myslí, že agilní metodika je o rychlosti a příběhu, ale ne o potřebách konečného uživatele. Programátoři, kteří si oblíbili jednu agilní metodiku ji pak opustí pro Software Craftsmanship, který jim přijde užitečnější. Designéři zase preferují Design Thinking a podnikatelé dávají přednost Lean Startupu. (Kerievsky, 2016b)

Mezitím, Agile Industrial Complex (Instituce kolem Agilní metodiky a školení v agilních metodikách v USA) pracuje na úpravách a vývoji metodiky, snaží se aktualizovat materiály, zkoušky a příklady, aby byly ze současného světa. Není tedy na škodu přijít i s jinou možností pojetí agilních metodiky, jako je právě třeba Modern Agile.

7. Závěr

Při psaní práce jsme se potýkali s omezeným množstvím zdrojů, hlavním problémem bylo, že většina zdrojů opírala o článek autora myšlenky. Tím, že myšlenka Modern Agile je poměrně mladá neexistuje mnoho zdrojů, zkušeností na toto téma od jiných autorů, kteří by se problematikou zabírali.

Je čas se vyslat zprávu o odlišnosti Modern Agile a zároveň poslat zastaralou metodiky a zkušenosti do zaslouženého důchodu. Je čas vrátit Agilní metodice lehkost, dostat ji zpět do praxe a zpátky na výsluní, přesně tam kam patří. Je čas na Modern Agile.

8. Literatura

KERIEVSKY, Joshua. An Introduction to Modern Agile. InfoQ: Software Development News [online]. 2016a [cit. 2017-05-08]. Dostupné z: https://www.infoq.com/articles/modern-agile-intro?utm_source=infoqWeeklyNewsletter&utm_medium=WeeklyNL_EditorialContent_culture-methods&utm_campaign=10252016news

KERIEVSKY, Joshua., Modern Agile. [Online] . 2016b. [cit. 2017-05-08]. Dostupné z: <https://www.agilealliance.org/resources/videos/modern-agile/>

KERIEVSKY, Joshua. Modern Agile - Industrial Logic. Industrial Logic - Modern Agile Consultancy [online]. 2015 [cit. 2017-05-08]. Dostupné z: <https://www.industriallogic.com/blog/modern-agile/>

HÁJEK, Lukáš. Refaktorování: Code refactoring [online]. 2013, 31 [cit. 2017-05-10]. Dostupné z: <http://kfe.fjfi.cvut.cz/~hajeklu2/files/OOP/Hajek-Refactoring.pdf>

CARDOZA, C.,. Agile 2016: Agile is old, we need to make it modern - See more at: <http://sdtimes.com/agile-2016-agile-old-need-make-modern/#sthash.vWXoP8TM.dpuf>. [Online]. 2016 [cit. 2017 2 10] Dostupné z: <https://www.slideshare.net/AgileSparks/modern-agile-joshua-kerievsk>

FRANCINO, Y., 2017. Modern Agile and Heart of Agile: A new focus for agile development. [Online] 2017 [cit. 10 2 2017]. Dostupné z: <https://techbeacon.com/modern-agile-heart-agile-new-focus-agile-development>