

Agilní metodiky pro distribuované projekty

Bc. Matej Paluš

xpalm29

Zimní semestr 2011

Obsah

1. ÚVOD	3
2. CHARAKTERISTIKA AGILNÝCH METODÍK	4
3. DISTRIBUTED SCRUM	8
3.1 CHARAKTERISTIKA DISTRIBUTOVANÉHO PROJEKTU	8
3.2 ÚVOD DO DISTRIBUTED SCRUM	9
3.3 DISTRIBUTED SCRUM V IBM	10
3.3.1 <i>Začiatok projektu</i>	11
3.3.2 <i>Prípravy pre Sprint Planning</i>	12
3.3.3 <i>Sprint Planning</i>	13
3.3.4 <i>Distribuované každodenné Scrum Meetingy</i>	14
3.3.5 <i>Efektívna spolupráca počas Sprintu</i>	15
3.3.6 <i>Koniec Sprint Reviews</i>	16
4. ZÁVER	18
5. ZDROJE	19

1. Úvod

Agilné metodiky predstavujú relatívne nový prístup pri riešení projektov. Vychádzajú z názoru, že najlepší spôsob ako preveriť funkcionality systému, je čo najrýchlejšie ho vyvinúť, odprezentovať ho zákazníkovi a na základe spätnej väzby ho upravovať. V poslednej dobe sa stávajú veľmi populárne u vývojových pracovníkov ale i manažérov projektov. Čím je však spôsobená stále sa zvyšujúca obľúbenosť agilných metodík? Sú naozaj práve tieto metodiky to najvhodnejšie pre riešenie daného projektu? Samozrejme, že záleží na type projektu, veľkosti či štruktúre. Agilné metodiky sa používajú hlavne pri menších projektoch, ktoré nie sú tak náročné na formálne prvky (mnohoriadkové dokumentácie). V súčasnosti sa však tieto metodiky pomaly presadzujú aj vo väčších projektoch s tým rozdielom, že sa kladie väčší dôraz na formálne prvky, ale stále to nie sú také vysoké nároky ako pri tradičných metodikách. A s väčšími projektmi prichádzajú aj väčšie tímy, ktoré sú v dnešnej dobe čím ďalej viac rozmiestnené po svete (distribuované). V prvej časti mojej práce sa zameriam na charakteristiku agilných metodík. V druhej a hlavnej časti sa chcem ale venovať metodike distribuovaného Scrumu (Distributed Scrum). V knihe *A Practical Guide to Distributed Scrum* je detailne popísaný celý cyklus vývoju projektu v distribuovanom prostredí a preto ponúknem vo svojej práci také rozsiahlejšie zhrnutie všetkého, čo táto kniha ponúka.

2. Charakteristika agilných metodík

Agilný softvérový vývoj je skupina vývojových metodík ktoré sú popísané v Manifeste agilného vývoja softvéru z roku 2001. Tento manifest založila skupina 17-tich softvérových inžinierov, iniciátorov vzniku týchto metód. Vychádzajú z názoru, že najlepší spôsob ako preveriť funkcionálnosť systému, je vyvinúť ho v čo najkratšom čase, odprezentovať ho zákazníkovi a na základe spätnej väzby ho upravovať.

V praxi to vyzerá tak, že celý vývoj je rozdelený do časových rámcov (iterácie popíšem neskôr). Tím ktorý má na starosti projekt musí odhadnúť jednotlivé požiadavky na funkcionálnosť a vyberie tie, ktorým sa bude venovať v rámci danej iterácie. Pri výbere sa riadi hlavne kapacitou tímu a časovou náročnosťou tak, aby sa úlohy zmestili do časového rámca. Nezabúda sa ani na biznis a prioritu. V každej iterácii musia vývojári doručiť nejakú biznis funkcionálnosť.

Samozrejme, že počas vývoja sa vyskytnú komplikácie a samotný tím tak nemusí stihnúť spraviť všetko včas. Namiesto prekročenia časového fondu na iteráciu a zvýšenia nákladov na inú „pomocnú“ silu, vyberie funkcionálnosť, ktorú len presunie do ďalšej iterácie, alebo ju nahradí jednoduchšou funkcionálnosťou. Lepším prípadom je, keď vývoj projektu napreduje rýchlejšie ako sa predpokladalo a stihne sa tak dokončiť celá funkcionálnosť ešte pred vyčerpaním časového fondu na danú iteráciu. V takomto prípade si tím vyberie ďalšiu požiadavku na funkcionálnosť a začne pracovať na nej. Je však vhodné dokončiť túto funkcionálnosť v danej iterácii, aby bol softvér zase použiteľný na konci celej iterácie. Všetko sa však musí priebežne konzultovať s klientom.

Prejdime však na samotný Manifest agilného vývoja softvéru. Základným pilierom týchto metodík sú tieto 4 hodnoty [1.]:

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

Časom sa z týchto 4 základných pilierov vytvorili ďalšie charakteristiky a princípy spoločné pre agilné metódy. Medzi tieto charakteristiky patria:

1. Najväčšou prioritou je uspokojiť zákazníka skorým a priebežným dodávaním hodnotného softvéru
2. Úpravy požiadaviek sú vítané aj neskôr počas vývoja. Agilné procesy podnecujú zmenu stať sa konkurenčnou výhodou zákazníka
3. Dodávať funkčný softvér často, každý týždeň až mesiac, s preferovaním kratšieho času
4. Spolupráca obchodníkov a vývojárov denno-denne, počas celého projektu
5. Tvorba projektov okolo motivovaných ľudí. Zabezpečiť im všetko potrebné, podpora ich potrieb a dôvera, že svoju prácu dokončia
6. Najúčinnjšou a najefektívnejšou metódou šírenia informácií v tíme je osobný rozhovor
7. Funkčný softvér je hlavným kritériom napredovania
8. Agilné procesy propagujú udržateľný vývoj. Sponzori, vývojári a užívatelia by mali byť schopní udržať tempo donekonečna
9. Neustála pozornosť k technickej excelentnosti a dobrému dizajnu zlepšuje agilitu
10. Základom je jednoduchosť – umenie maximalizovať množstvo práce, ktorú nie je potrebné urobiť
11. Najlepšia architektúra, požiadavky a dizajn sa vynárajú zo samoorganizovaného tímu
12. Tím v pravidelných intervaloch reflektuje, ako byť efektívnejší. Potom adekvátne prispôsobí svoje chovanie

Vo vyššie uvedených charakteristikách je spracovaná väčšina spoločných znakov agilných metód. Využitie týchto znakov sa však môže trochu líšiť v každej z metodík agilného vývoja. Veľmi dôležité sú časti o tom, že „Softvér vytvárajú ľudia pre ľudí“. Zapojenie ľudí je dôležitejšie ako kedykoľvek predtým, na ľudí, ktorí sú zapojení do procesu vývoja softvéru sa pozerá ako na samotný proces. V týchto metodikách sa nikdy neuprednostňuje dodržanie časového plánu, ktorý väčšinou ani neexistuje. Tím, ktorý pracuje na určitom projekte spolupracuje spolu dlhodobo a práve preto je dôraz kladený na ľudí. Dôležitým prvkom je utvorenie čo najužšieho vzťahu medzi jednotlivými členmi tímu. K tomuto pomáhajú aj časté stretnutia, napríklad v metodike SCRUM je takéto stretnutie na programe každý deň. Tieto stretnutia predstavujú najdôležitejšiu formu komunikácie. Fyzická komunikácia (tvárou v tvár) je jednoducho najefektívnejšia v projektoch, ktoré sa riešia pomocou týchto metodík.

Ďalšou charakteristikou agilných metodík je víťanie zmien v každej fáze životného cyklu vývoja softvéru. To znamená neustále zmeny k lepšiemu, k čomu je potreba úzka

spolupráca so zákazníkom. Žiadne vytvorenie softvéru a prezentovanie až finálneho produktu zákazníkovi. K dosiahnutiu tohto cieľa je preto potrebné časté vydávanie funkčných verzií, na ktorých si zákazník sám otestuje svoje požiadavky a zistí, ktoré požiadavky sú ešte aktuálne, a ktoré nie. Pri takomto testovaní zákazníkom sa predchádza chybným nepochopením funkcionality z pohľadu vývojárov softvéru. Tomuto sa musí prispôbiť aj samotný spôsob vývoja, ktorý je inkrementálny a iteratívny. Softvér sa preto vyvíja po menších celkoch a každý celok prechádza jednotlivými fázami samostatne. Každá verzia inkrementálne vytváraného softvéru musí byť funkčná (k jej funkčnosti nesmú byť potrebné neskôr vyvíjané časti). Hlavná myšlienka iteratívneho vývoju je, že na konci každej iterácie (ich dĺžka je obvykle niekoľko týždňov) je výstupom funkčný softvér, ktorý by ste mohli dodať zákazníkovi (softvér sa však odovzdáva až keď má dostatok požadovanej funkcionality). Inak povedané, opakovane vytvárate hotový softvér, s postupne rastúcou sadou funkcií.

Veľmi významnou charakteristikou agilných metodík je aj redukcia nadbytočnej práce, čo znamená, že vývojár sa zaoberá časťami, ktoré sú práve na rade, a nevenuje pozornosť iným, nepodstatnejším častiam. Tento postup je veľmi dôležitý aj pri zmene požiadavkov na softvér od zákazníka. Vývojár si nemôže byť istý, či zákazník neprehodnotí svoje požiadavky a preto sa každá časť vyvíja až keď je všetko dohodnuté so zákazníkom.

Agilné metodiky však nepredstavujú presné postupy ako sa má alebo nemá vyvíjať softvér, skôr obsahujú sadu odporúčaní, ktoré je možné prispôbiť a využiť len vyhovujúce časti a prispôbiť ich potrebám konkrétneho projektu.

Toľko k stručnému zhrnutiu základných princípov a charakteristík agilných metodík.

Otázkou však je prečo sú tieto metodiky momentálne úspešné v projektoch, kde klasický prístup k vývoju zlyháva? Hlavnou príčinou je dynamický vývoj, ktorý je spôsobený prudkými zmenami na poli informačných technológií. Predpoklad presného definovania a špecifikovania požiadaviek zo strany zákazníka je v agilných metodikách minulosťou. V súčasnosti sa jednoducho nedajú presne špecifikovať všetky požiadavky na projekt ešte pred jeho zahájením. Tie sa dynamicky menia počas celého životného cyklu projektu, či už kvôli zmene technológií, požiadaviek trhu alebo legislatívy. Každý projektový tím, tak musí byť schopný pružne reagovať na zmenu a vznik nových požiadaviek.

Projekty, využívajúce klasické metodiky momentálne stroskotávajú práve na tomto probléme, že vývojári, ktorí pracujú na softvéri niekedy aj rok, sa po uvedení na trh ukáže ako zastaraný. Požiadavky sa síce počas vývoja softvéru menili, ale tieto zmeny nie je možné

dostatočne zapracovať do projektu. Prekročenie časového plánu, či rozpočtu je tak veľmi obvyklým zjavom pri vyvíjaní softvéru v dlhšom horizonte. Klasické prístupy sa takisto snažia o zohľadnenie zmien a prípadne vzniku nových požiadaviek, to je však možné len pomocou naplánovania tejto zmeny či vzniku. Ani to nemusí byť stopercentne úspešné a záleží to hlavne od skúsenosti vývojára, ale aj náhody.

Agilné metodiky sa snažia riešiť tieto problémy tým, že vítajú zmeny požiadaviek a počítajú s nimi už od samého počiatku projektu. Zmena predstavuje v agilných metodikách základný princíp vývoja.

Nechcem však navodiť dojem, že agilné metodiky sú jediné a tie pravé na riešenie každého druhu projektu. Je potrebné rozlišovať typ projektu, veľkosť(rozsah), finančnú náročnosť, dĺžku projektu či predpokladať ako sa budú vyvíjať zmeny v priebehu projektu. Na základe toho sa už dá rozhodnúť akú metodiku pre daný projekt použijeme. Na veľké a náročné projekty použijeme skôr tradičné (klasické) metodiky ako RUP, Open atď. Na menšie, dynamické projekty je zas vhodnejšie použiť agilné metodiky, ktoré nevyžadujú takú náročnosť na dokumentáciu. V súčasnosti sa však obidva prístupy začínajú trochu prelínať. Klasické metodiky sa dajú aplikovať bez tak veľkého dokumentovania projektu a agilné metodiky používajú pri väčších projektoch viac formalizovaných dokumentov. Tie hlavné princípy jednotlivých metodík sa však prispôsobovať iným metodikám veľmi nemôžu a preto sa stále zachováva spomínaný rozdiel medzi obidvomi prístupmi k vývoju softvéru.

3. Distributed Scrum

3.1 Charakteristika distribuovaného projektu

Distribuovaný projekt je typ projektu, ktorý riešia členovia rozmiestnení po všetkých možných miestach na svete. Títo členovia teda nie sú spolu na jednom mieste a nemajú tak možnosť všetko vykomunikovať spolu a podľa toho sa zariadiť. Spolupráca jednotlivých členov je preto zabezpečená pomocou emailu, internetu a inými aplikáciami podporujúcimi komunikáciu na veľkú vzdialenosť. V distribuovaných projektoch neexistuje neformálna časť komunikácie ako tomu je v bežnom projekte. Komunikácia medzi členmi na odlišných miestach zásadným spôsobom klesá a znižuje sa aj jej kvalita. Tomto type projektov bežne nastáva situácia, keď jednotliví členovia sú z rôznych časových pásiem a pracujú tak v iných časoch. On-line stretnutia sú však potrebné aj napriek tomu, že ich zaistenie je niekedy veľmi náročnou úlohou.

Ďalšou vlastnosťou týchto projektov je to, že sú rozsahovo a aj počtom členov veľmi veľké. Synchronizácia členov je tak často veľmi zložitá. Keďže členovia sú často z rôznych kontinentov, dochádza aj ku kultúrnym rozdielom. Softvér je vyvíjaný ľuďmi a vývoj je tak silne viazaný na ľudskú povahu, ktorá je ovplyvnená aj kultúrou. Preto sú kultúrne rozdiely jedným z faktorov ovplyvňujúcich vývoj softvéru.

Na začiatku práce som spomínal aj 12 princípov, z ktorých sú najmä tieto 3 ovplyvnené distribuovaným vývojom:

1. Individuality a interakcie – výber správnych individualít, ktoré by boli schopné udržiavať efektívnu komunikáciu medzi sebou, je náročnejší bez priamej a dostatočnej komunikácie.
2. Spolupráca so zákazníkom – blízke vzťahy so zákazníkom sú ovplyvnené vzdialenosťami medzi jednotlivými členmi, čo má za následok menší objem a efektivitu komunikácie.
3. Odpoveď na zmenu – distribuované prostredie prináša väčšie zdržania pri reakcii na zmenu a tento čas sa tak predlžuje oproti bežným projektom.

3.2 Úvod do *Distributed Scrum*

Metodika Scrum používaná v distribuovaných projektoch sa trochu líši od tej bežnej. [4.] Pracovanie v distribuovanom tíme dáva organizáciám prístup k talentovaným ľuďom, ku ktorým by sa v lokálnom projekte nemali ako dostať. Navyše, keď organizácia pracuje s distribuovaným tímom, získava tak skúsenosti zo spolupráce s ľuďmi iných svetových trhov. Na konci sa potom môže ukázať práve táto skúsenosť ako nevyhnutná pre prania organizácie a jej rozširovanie v medzinárodnom meradle.

Okrem toho môže byť projekt dokončený rýchlejšie, keď ľudia z rôznych časových pásiem nepretržite pracujú na konkrétnom projekte. V konečnom dôsledku, organizácie môžu získať významné úspory z nákladov, ak pracujú v distribuovanom prostredí.

Ďalšie dôvody pre prácu v distribuovanom tíme:

- Flexibilita zníženia alebo zvýšenia počtu členov v tíme bez straty nevyhnutných informácií, ktoré sú potrebné pre kompletizáciu projektu
- Schopnosť znovu vytvorenia úspešného tímu v prípade potreby získania ďalších produktov.

Jedným z najdôležitejších faktorov, ktoré je potrebné prekonať pri zavádzaní metodiky Scrum v distribuovanom projekte je komunikácia. V bežnom prostredí sa kladie pri tejto metodike dôraz na každodenné stretnutia, ktoré sú súčasťou procesu tvorby nejakého projektu. V dôsledku toho je nevyhnutné mať face-to-face stretnutia počas významných udalostí projektu ako zahájenie či plánovanie projektu. Naplánovanie týchto stretnutí je však oveľa zložitejšie ako pri bežnom projekte.

Hoci komunikácia môže byť hlavným záujmom, v distribuovaných tímoch je to tímová spolupráca. Priamym následkom tak je, že Scrum tím zdieľa zodpovednosť so všetkým, čo jednotlivé tímy vyprodukujú. Tím je jednoducho hrdý na to, čo spravil. Navyše, kultúra v organizácii zabezpečuje, že individuality preukazujú svoj zmysel pre vlastníctvo a zaväzujú sa svojim tímom.

Najdôležitejším prvkom v metodike Scrum a jej aplikovaní v distribuovanom projekte sú komunikačné technológie. Členovia distribuovaného tímu musia mať prístup k príslušným komunikačným nástrojom, pretože tieto nástroje sú nevyhnutné pre zbavenie sa komunikačných bariér. Napríklad video konferencie, webové kamery alebo hands-free

súpravy sú obzvlášť nápomocné. IM softvér, zdieľanie obrazoviek či email sú tiež veľmi dôležitými komunikačnými nástrojmi.

Celkovo potom, efektívna implementácia metodiky Scrum (a kvalitný dosiahnutý produkt) sa môže podariť aj v distribuovanom tíme. Na dosiahnutie pocitu vlastníctva medzi všetkými členmi tímu je potrebné, zamerať sa na budovanie vzťahov a vytvorenie hlavných cieľov. Jedine tak sa žiadané výsledky dostavia.

3.3 Distributed Scrum v IBM

Distribuovaný Scrum je veľmi dobre a rozsiahlo popísaný v knihe *A Practical Guide to Distributed Scrum* [5.] , ktorú vydala firma IBM. Niekoľko odborníkov z tejto firmy popisuje množstvo detailov na ktoré treba dávať pozor pri aplikovaní metodiky do distribuovaného prostredia. V knihe sa venuje pozornosť takmer všetkému, na čo môžete naraziť pri distribuovanom Scrum. Pokúsim sa tak priblížiť každú jednu fázu projektu, tak ako to funguje v spoločnosti IBM. Fáze projektu, ktoré budem rozoberať sa dajú rozčleniť takto:

- Začiatok projektu (Identifikovanie problému, vytvorenie backlogu a release plánu)
- Prípravy pre Sprint planning (Sprint preplanning aktivity a prístupy pre Sprint preplanning meeting)
- Sprint planning (logistika, rozhodnutie čo sa spraví, rozhodovanie o tom ako sa to spraví)
- Distribuované každodenné Scrum meetingy (efektívne používanie 3 otázok na meetingu, spôsoby komunikácie pri denných meetingoch, rady pre distribuované meetiny)
- Efektívna spolupráca počas Sprintu (komunikácia počas Sprintu, vysporiadanie sa s novými požiadavkami uprostred Sprintu, integrácia, testovanie, Test-driven vývoj)
- Koniec Sprint reviews (kto sa zúčastňuje na Sprint reviews, výzvy ktorým tímy čelia, ďalšie výzvy pre distribuované tímy)

Firma IBM má obrovské skúsenosti v používaní metodiky distribuovaného Scrumu. Keďže má cez 400 000 zamestnancov po celom svete, nemá možnosť zhromaždiť všetkých na jedno miesto a jedinou šancou je tak využívanie distribuovaného vývoja. Na najväčších projektoch pracuje v IBM až okolo 600 zamestnancov a ich synchronizácia je mnohokrát veľmi náročnou, ale kľúčovou úlohou. Bežný menší projekt má v IBM približne 50 zamestnancov.

3.3.1 Začiatok projektu

Scrum zahrňuje 3 úrovne plánovania: Release Planning, Sprint Planning a Daily Scrums. Release sa zaoberá finálnym produktom a výsledkom, aký môže vedenie čakať. Sprint plannin hovorí o naplánovaní si jednotlivého Sprintu. Daily Scrums zas dáva príležitosť na každodenné prispôsobenie krátkodobých plánov.

Na začiatku projektu je však potrebné získať a zorganizovať všetky informácie, ktoré prichádzajú z rôznych zdrojov. Bežným nástrojom je Affinity Diagram. V ňom sa zoskupia informácie a potom sa môže jednoducho začať s analýzou a rozhodovaním o potrebných riešeniach. V praxi to vyzerá približne tak, že každý nápad sa napíše na kartičku, podobné nápady sa zoskupia vedľa seba, potom sa zoskupia jednotlivé témy a nakoniec sa tieto nápady potvrdia v náležitých skupinách. V tíme, ktorý je na jednom mieste ide o jednoduché vytvorenie takéhoto diagramu. V distribuovanom prostredí sa môžu vytvoriť lepkavé kartičky, ktoré sa potom jednoducho lepia na bielu tabuľu.

Ďalšia technika, ktorá sa musí prispôbiť v momente ak sa pracuje v distribuovanom tíme je Planning Poker. Je to technika, ktorá sa využíva pri agilnom odhadovaní. Každý člen tímu predstaví svoju definíciu požiadavku a potom ostatní členovia zhodnotia ako dôležitá je táto požiadavka. V distribuovanom tíme sú opäť iné spôsoby ako aplikovať túto techniku. ScrumMaster prečíta požiadavky a následne jednotlivé tímy spolu komunikujú cez skupinové chatové okno a dohadujú sa na významnosti požiadavky.

Ďalším dôležitým bodom pri plánovaní projektu v distribuovanom prostredí je brať do úvahy aj jednotlivé sviatky v rôznych krajinách. Všetko sa musí naplánovať hneď na začiatku, aby v neskoršej fáze projektu nevznikli problémy. Čím väčší projekt, tým náročnejšia je koordinácia distribuovaných tímov. Naplánovanie finálneho produktu a jeho požiadaviek je kľúčovým faktorom pri zahajovaní Scrum projektu.

3.3.2 Přípravy pre Sprint Planning

V tradičnom Scrume pozostáva príprava na Sprint Planning z 2 stretnutí, označovaných ako Sprint Planning meeting. Každé zo stretnutí má svoj presný zámer a úlohy. V prvom stretnutí vlastník produktu zopakuje priority najvyššieho stupňa a tím sa rozhodne koľko práce dokáže spraviť v Sprinte v závislosti na rýchlosti v predošlom Sprinte. V druhom stretnutí sa tímy dohadujú ako budú postupovať v spĺňaní jednotlivých požiadaviek v samotnom Sprinte.

Pri distribuovaných tímoch sa využíva jedna zaujímavá technika na zistenie, či každý člen tímu pochopil najdôležitejšie požiadavky. Scrum Master poprosí členov tímu o vysvetlenie kľúčových požiadaviek vlastnými slovami už pri stretnutí o plánovaní Sprintu. Týmto sa zaisťuje bežné porozumenie napriek jazykovým bariéram alebo kultúrnym rozdielom. Celým zmyslom týchto stretnutí je čo najrýchlejšie pochopenie kľúčových požiadaviek, a počas Sprintu sa už len zamerávať na ich priame riešenie.

Prítomnosť príprav pre Sprint Planning je o to dôležitejšia, pokiaľ sa jedná o projekt v distribuovanom tíme. V týchto stretnutiach sa účastníci snažia čo najlepšie odhadnúť priority a požiadavky potom naplánovať do nasledujúceho Sprintu. Na týchto stretnutiach sa môže tím rozhodnúť aj o tom, že niektoré požiadavky sú jednoducho nesplniteľné alebo sa môže dohodnúť na niektorých požiadavkách, o ktorých sa predtým nehovorilo.

Často sa využíva Full-Team Approach (Stretnutie, ktoré zahŕňa celý Scrum tím). Pre veľké distribuované tímy je však problém dostaviť sa na jedno miesto v jeden čas. Preto sa niekedy využívajú telekonferencie. Aj tu platí, že čím väčšia konferencia, tým horšia možnosť porozumenia medzi všetkými členmi tímu. Výhod takýchto stretnutí je mnoho. Od upevňovania ducha tímu počas stretnutia, cez vyššiu schopnosť adaptácie na projekt až po triedenie a návrh nových požiadaviek.

V distribuovaných projektoch je veľmi populárnym aj Preplanning Team Approach. To je stretnutie, ktorého sa zúčastňuje len vybraný člen tímu, ktorý ho potom reprezentuje. Pri rozsiahlych distribuovaných tímoch je toto veľkou šancou na to aby sa jednotliví zástupcovia tímov spolu stretli a prediskutovali všetky potrebné veci.

Ak to teda celé zhrnieme, vyššie spomenuté stretnutia nie sú oficiálne zaradené do metodiky Scrum, ale často ju dokážu spestriť a hlavne jednotlivé požiadavky potom dostať do

želanej podoby. Výhodou je aj udržovanie tímu v sústreďení pri prechode zo zahájenia projektu až po prvý Sprint.

3.3.3 Sprint Planning

Cieľom stretnutia Sprint Planning je dohoda tímu na čom sa bude pracovať v nasledujúcom Sprinte a ako sa ten cieľ dosiahne. Na plánovanie samotného Srintu sa často využívajú poznatky zo stretnutí, ktoré som popísal v minulej kapitole. Aj samotný Sprint Planning pozostáva z 2 stretnutí. V prvom sa tím rozhoduje čo bude robiť, v druhom stretnutí sa rozhoduje ako ten finálny cieľ spraviť.

O reprezentantoch tímu, ktorí sa zúčastňujú na stretnutiach som už hovoril. Na stretnutí Sprint Planning sa však musia zúčastniť všetci členovia tímu. V distribuovaných tímoch je však veľmi náročné nájsť čas, ktorý bude vyhovovať všetkým. Preto sa niektorí členovia tímu stretávajú aj mimo pracovných hodín, pokiaľ je to možné. Nemožnosť účasti všetkých členov tímu na stretnutí Sprint Planning sa opäť rieši pomocou komunikačných nástrojov. V dnešnej dobe to predsa len už nie je taký problém a dostupnosť internetu a nástrojov na prenos informácií a blíží k 100 %. Každý človek, ktorý sa zúčastňuje na takejto konferencii musí mať prístup k požiadavkám projektu, úlohám či odhadom. Využíva sa aj nástroj na zdieľanie obrazovky, kde jedna osoba sa stará o hlavnú obrazovku a pracuje na updatech, ktoré vyplývajú z diskusie.

Niekedy sa využíva aj systém s reprezentantmi. Tí musia na stretnutí Sprint Planning poskytnúť všetky odhady práce v Srintu a potom sa vrátiť do svojich tímov, kde tieto odhady prediskutujú.

Veľmi zaujímavou technikou je umiestnenie vízie produktu na viditeľné miesto. V bežnom tíme sa to dá vyriešiť jednoduchým napísaním na tabuľu. V distribuovaných tímoch sa to vyvesí na domovskú webovú stránku projektu. Všetci členovia tímu tak vidia túto víziu a zapamätajú si ju počas trvania celého projektu.

Ani identifikovanie úloh nie je v distribuovanom tíme tak jednoduché ako v bežnom. Hoci tímy v distribuovanom prostredí môžu spolu pracovať na diaľku, pridávanie kľúčových požiadaviek a úloh sa musí vždy náležite prediskutovať a tím to tak stojí mnoho času. Ďalším problémom oproti práci v bežnom tíme je aj nemožnosť čítania reči tela. Z toho sa dá vždy

Ľahšie pochopiť postoj daného člena tímu a to či sa plne stotožňuje s projektom. V distribuovanom tíme nastáva aj situácia keď je jednoducho ticho a ScrumMaster tak musí zobrať zodpovednosť za zaistenie dobrej komunikácie. Nikto by si totiž nemal interpretovať mlčanie ako súhlas. Členovia tímu by mali klásť otázky tak ako to potrebujú a slovné odpovede potom pomáhajú zlepšovať pochopenie.

3.3.4 Distribuované každodenné Scrum Meetingy

Veľký problém nastáva pri každodenných stretnutiach. V bežnom tíme ide o kľúčovú vec, pretože denné porady sú podstatou celej metodiky Scrum. Face-to-face kontakt môže byť len ťažko nahradený telekonferenciami, ktoré sú neosobné a členovia tímu nemusia dávať pozor počas celého stretnutia.

Tímy v distribuovanom prostredí sa však musia adaptovať aj na takéto podmienky a snažiť sa vyriešiť tieto každodenné stretnutia čo najefektívnejšie. Miera distribúcie tímu tiež ovplyvňuje finálne riešenie. Komunikácia tímu, keď všetci členovia sú z jednotného časového pásma a hovoria tým istým jazykom je o poznanie jednoduchšia ako keď je tím roztrúsený po celom svete, používajúc rôzne jazyky. V tímoch, ktorých sa pracovné hodiny neprekrývajú, je najťažšou úlohou naplánovanie 15 minútového Scrum meetingu, pretože niektoré tímy musia zapojiť aj čas, ktorý v práci bežne netravia

Členovia distribuovaného tímu sa môžu na Scrum meetingu zúčastňovať pomocou:

- Telekonferencie
- Videokonferencie
- Využívanie IM na komunikáciu
- Meeting skrz dokumentáciu
- Liasonov prístup

Prvé tri spôsoby asi netreba bližšie vysvetľovať. Meeting pomocou dokumentácie je najmenej efektívnejší spomedzi všetkých 4 spôsobov. Ide o to, že sa stretnú tímy, ktoré majú práve čas (nie je v ich časovom pásme napríklad hlboká noc) a vytvoria dokument z odpoveďami na 3 základné otázky Scrum Meetingu. Hlavnou výhodou tohto prístupu je, že nemusia byť všetci členovia tímu prítomní na porade. Avšak tieto stretnutia nie sú

interaktívne a členovia, ktorí práve nie sú prítomní, nemajú príležitosť vypočúť si komentáre od ostatných členov tímu na odpovede týchto 3 otázok. Takisto sa môže stať, že členovia, ktorí sa nezúčastňujú týchto stretnutí sa cítia byť menej členmi tímu alebo si myslia, že ostatní nedostatočne oceňujú ich príspevok do projektu. Je dôležité, aby všetci členovia Scrum tímu pracovali spoločne a robili kompromisy.

Liasonov prístup spočíva v rozdelení Scrum tímu do 2 rozdielnych Scrum meetingov, pričom každá polovica si určí čas, ktorý jej najviac vyhovuje. ScrumMaster potom zaistí uje zdieľanie informácií medzi týmito dvomi časťami. Aj táto metóda má však svoje nevýhody ako zhoršená komunikácia medzi tímami, ktorá sa dá prirovnať k detskej hre Telefón. (Jedna osoba povie druhej nejakú informáciu a tá sa pomaly mení a mení, až k poslednej osobe príde takmer rozdielna informácia).

K efektívnym denným poradám preto pomáha vhodne zvolený prístup tímov. Pokiaľ sa nájde dobré riešenie, môže distribuovaný tím fungovať minimálne tak správne ako bežný.

3.3.5 Efektívna spolupráca počas Sprintu

Vo veľkých distribuovaných tímoch je dodanie hodnotného riešenia skoro, jednou z najvýznamnejších zmien nových Scrum tímov. Vývoj, ktorý trval predtým niekoľko mesiacov kým sa uvidela pridaná hodnota, momentálne trvá niekoľko týždňov. Všetky tieto významné zmeny spôsobili, že tímy sú schopné dodať kvalitný softvér a predviesť ho zainteresovaným osobám s minimalizovaným rizikom neúspechu.

Tímy, ktorým sa pracovné hodiny takmer vôbec neprekrývajú, musia využívať primárne email alebo iné možnosti písanej dokumentácie. Písaná komunikácia je však pri distribuovaných tímoch miernou výhodou, pretože pre niektorých členov nemusí byť zvolený jazyk aj jazykom rodným. Pochopenie písanému slovu je tak jednoduchšie ako hovorenému.

Používanie správnych nástrojov je ďalšou dôležitou témou v metodike Distributed Scrum. Je síce pekné použiť 3D virtuálne stretnutie, ale ak každý člen tímu potrebuje 5 minút kým sa len dostane do aplikácie, tak to nemusí byť vôbec praktické a veľmi nápomocné. Každodenný Scrum Meeting má trvať 15 minút a preto jednoduchá telekonferencia, kde každý člen tímu môže odpovedať jednoducho na 3 otázky, je dostačujúca.

Niekedy sa stáva, že tímy prispôsobujú svoje pracovné hodiny tímom z iných časových pásiem. Pracujú tak na projekte „spolu“. Hrozí však, že tím, ktorý si takto upravil svoje pracovné hodiny, potom pracuje aj navyše, počas vlastných hodín. Je preto dôležité nastaviť hranice, ktoré budú členovia tímu dodržiavať.

Toto je ďalšia z kapitol, ktorá je veľmi detailne popísaná v knihe od IBM. Na mnoho stránkach je predstavených niekoľko prístupov, ktoré zefektívňujú prácu v distribuovanom tíme. Jednotlivé techniky sú vysvetlené tak ako prebiehajú v spoločnosti IBM a preto sú aj overené prakticky. Všetko je navyše podložené teoreticky a preto netreba mať pochybnosti, že niektoré veci v tejto kapitole sú len vymyslené nejakou skupinou ľudí.

3.3.6 Koniec Sprint Reviews

Primárnou úlohou Sprint Review stretnutia je posúdenie, či Scrum tím dosiahol cieľ, ktorý si určil na začiatku Sprintu. Na tomto stretnutí sa stretne vývojový tím, majiteľ produktu, ScrumMaster a ostatné zúčastnené strany. Samozrejme, že o cieľoch sa členovia tímu bavia aj počas Sprintu aby bolo zaistené porozumenie každého člena.

V veľkom distribuovanom prostredí je opäť dôležité nájsť správnu formu komunikácie s ostatnými zúčastnenými stranami. Tímy musia naplánovať stretnutia nejaký ten čas vopred. Nie je jednoduché, aby sa majiteľ produktu tak ľahko dostavil na toto stretnutie, kde mu bude predvedený výsledok práce.

Na začiatku tohto stretnutia majiteľ produktu zopakuje víziu, ktorú načrtnol v prvých plánovacích stretnutiach. Táto časť je významná pri veľkých distribuovaných tímoch. Každý z členov tímu si tak zopakuje a zosumarizuje vízie produktu na ktorom pracuje. Po zhrnutí strategickej vízie, by mal vlastník produktu zhrnúť aj objekty, ktoré sa využijú v Sprinte.

Aj pri tomto stretnutí je najväčším problémom naplánovanie. Stretnúť sa pomocou telekonferencie tak, aby všetky zúčastnené strany boli v práci počas svojich pracovných hodín, je nemožné. Členovia, ktorí žijú v rozdielnych časových pásmach majú jednoducho pracovnú dobu niekedy úplne obrátenú od ľudí, ktorí sú na druhej strane zemegule. Preto sa aj toto stretnutie rieši často pomocou „zdieľania bolesti“, čo znamená, že členovia ktorí nemajú pracovnú dobu v čase stretnutia, si ten čas nájdú a aj keby to bolo o polnoci, sa tohto stretnutia zúčastnia.

Predvádzanie implementovaných požiadaviek na konci Sprintu je teda často veľmi zložité. Vyžaduje si to veľkú náročnosť na koordináciu všetkých členov a samotného produktu. Demonštrácia týchto požiadaviek na konci Sprintu je však natoľko dôležitá, že jej vynechanie nie je možné.

4. Závěr

V mojej práci som sa venoval hlavne metodike Distributed Scrum. V úvode som charakterizoval trochu princípy agilných metodík a v ďalšej časti, som sa už plne venoval metodike Scrum a jej použiti v distribuovanom projekte.

Táto metodika pri použití v distribuovanom prostredí závisí hlavne na úrovni komunikácie medzi jednotlivými členmi tímu

V distribuovanom vývoji je najzložitejšou úlohou naplánovať jednotlivé stretnutia, ktoré sú potrebné pre úspešné aplikovanie tejto metodiky. Na začiatku projektu je potrebné si dohodnúť načasovanie a samotnú frekvenciu stretnutí. Veľkou výzvou v takomto type projektov je vyvinutie takého komunikačného kanálu, ktorý bude natoľko efektívny, že bude podporovať všetky potrebné činnosti.

V knihe, z ktorej som čerpal bol naozaj veľmi podrobne vysvetlený každý krok k úspešnému aplikovaniu tohto prístupu v distribuovanom prostredí. Mnoho techník je na 200 stranách do hĺbky rozobraných a napomáhajú tak ľahšie definovať distribuovanému tímu, ktorú z týchto techník použije práve na ten svoj špecifický projekt. Navyše 3 autori, ktorí sa spoločne podieľali na písaní tejto knihy majú mnoho praktických skúseností z aplikovania tejto metodiky v spoločnosti IBM.

5. Zdroje

[1.] *Agilemanifesto.org* [online]. 2001 [cit. 2011-12-11]. Manifest agilního vývoje software. Dostupné z WWW: <http://agilemanifesto.org/iso/cs/>

[2.] KOSA, Matúš. *Websupport.sk* [online]. 2011 [cit. 2011-11-14]. Agilná metodológia. Dostupné z WWW: <<http://blog.websupport.sk/2011/03/ws-agilne-agilna-metodologia/>>

[3.] DĚDEK, Vladimír. *Starec.blog.zive.cz* [online]. 2009 [cit. 2011-11-16]. Metodika Scrum. Dostupné z WWW: <http://starec.blog.zive.cz/2009/12/metodika-scrum-2/>

[4.] O'CONNELL, Feyza. *Scrumalliance.org* [online]. 2010 [cit. 2011-11-16]. Scrum Success in a Distributed Team Environment. Dostupné z WWW: <<http://www.scrumalliance.org/articles/165-scrum-success-in-a-distributed-team-environment>>

[5.] WOODWARD, Elizabeth ; SURDEK, Steffan ; GANIS, Matthew . *A Practical Guide to Distributed Scrum*. Boston : Pearson Education, Inc., 2010. 236 s.