

JE VINNÍKOM SCRUM?

*Projekt dopadol napriek použitiu agilnej metodiky zle.
Je ale vinníkom naozaj Scrum?*

Igor Aufricht

Slovenská technická univerzita
Fakulta informatiky a informačných technológií
Ilkovičova 3, 842 16 Bratislava
1234.ia[zavináč]gmail[.]com

Abstrakt. Softvérové projekty často končia neúspešne, zlyhajú alebo sú zrušené. Väčšina ukončených projektov prekročí určený rozpočet alebo časový plán. Takéto projekty stoja spoločnosti ročne miliardy dolárov. Agilné metodiky si ukladajú za cieľ vylepšenie procesu vývoja softvéru tak, aby projekt skončil úspešne a zákazník bol spokojný. Aj napriek implementácii agilných metodík však mnohé projekty končia neúspešne. Sú však na vine agilné metodiky? Momentálne neexistuje veľa zdrojov, ktoré by sa tejto problematike venovali. Ja však tvrdím, že zlyhanie mnohých agilných projektov nie je spôsobené agilnými metodikami per se, ale ich nevhodnou, prípadne nekompletnou implementáciou.

Kľúčové slová: zlyhanie projektov, agilné metodiky

Projekty neustále zlyhávajú

Softvérové inžinierstvo má za sebou dlhú históriu. Od štyridsiatych rokov, kedy sa objavili prvé digitálne počítače, cez rozmach osobných počítačov v osemdesiatych rokoch až po začiatok dvadsiateho prvého storočia, kedy sú počítače prakticky všade okolo nás. Človek by povedal, že po takej dlhej dobe už bude softvérový proces dotiahnutý k dokonalosti a projekty sa budú končiť úspešne, včas a s minimom nákladov.

Situácia je však diametrálne odlišná. Aj dnes, šesťdesiat rokov od vzniku prvých počítačov, veľké množstvo projektov končí neúspešne, prípadne s časovým sklzom a/alebo prekročeným rozpočtom. Zriedkavé nie sú ani projekty, ktoré nespĺňajú dohodnuté požiadavky alebo sú ťažko udržiavateľné.

Rôzne zdroje uvádzajú rôzne čísla [3]. Percento neúspešne ukončených projektov (teda takých, ktoré boli zrušené pred finalizáciou) sa pohybuje v závislosti od zdroja od

2 Igor Aufricht

15% po 30%. Zhruba polovica projektov prekročí rozpočet a ešte viac projektov skončí s časovým sklzom. Výsledkom sú straty rádovo miliárd dolárov ročne.

Agilné metodiky na pomoc

Agile Manifesto prezentuje pekný prehľad zámerov agilných metodík [1]. Autori dospeli k týmto hodnotám:

- Jednotlivci a interakcie pred procesmi a nástrojmi
- Fungujúci softvér pred vyčerpávajúcou dokumentáciou
- Spolupráca so zákazníkom pred vyjednávaním o zmluve
- Reagovanie na zmeny pred dodržovaním plánu

Vzhľadom na to, že hlavnými príčinami neúspechu projektov [4] sú nestihnuté termíny, prekročený rozpočet, nedostatočná komunikácia, prípadne nesplnenie projektových požiadaviek, zdá sa, že využitie agilných metodík takmer určite prinesie lepšiu úspešnosť projektov. Napriek tomu však veľa agilných projektov končí neúspechom [2].

Agilné metodiky nefungujú?

Osobne si myslím, že za mnohými neúspechmi projektov, ktoré o sebe tvrdia, že využívajú agilné metodiky, stojí nevhodná, resp. neúplná implementácia danej metodiky. Ako príklad by som si dovoľil zobrať jednu z najpoužívanejších agilných metodík dneška - Scrum.

Predstavme si rozhovor dvoch imaginárnych osôb. Ján Neborák je členom vývojárskeho tímu, ktorý sa nedávno pokúsil implementovať metodiku Scrum na vývoj projektu. Projekt skončil katastrofálne a Ján na základe vlastných skúseností zaujal negatívny postoj k agilným metodikám. Samo Agilista sa snaží zistiť nejaké podrobnosti:

„Použili sme SCRUM a ten nefungoval.“

„To sa mi nejako nezdá. Povedz mi niečo o vašom Product Ownerovi .“

„Product Ownera sme v tíme nemali. Ale mali sme analytika.“

„Mal analytik právo rozhodovať o zmenách požiadaviek?“

„Nie, ale veď tie boli dohodnuté vopred.“

„Takže ste nemali v tíme nikoho od zákazníka.“

„Nie, načo? Veď sme všetci vedeli, čo máme spraviť a kedy to odovzdať.“

„Aha. A ako ste vyvíjali?“

„Na začiatku mesiaca Scrum Master rozdelil úlohy a potom každý pracoval na svojich. Ale nefungovalo to, na konci mesiaca úlohy často neboli dokončené.“

„A skúšali ste rozdeliť úlohy na menšie a skrátiť šprinty?“

„Nie, veď potom by sa nič nestihlo spraviť.“

„Mávali ste denné stretnutia?“

„Áno, ale podľa mňa to bolo len zabíjanie času.“

„A ako teda projekt dopadol?“

„Projekt sme odovzdali neskoro, zákazník bol veľmi nespokojný a chcel nechať prepracovať polovicu systému. Keď sa dozvedel, koľko by to stálo, ukončil projekt. Tento Scrum nám teda veľmi nepomohol...“

Uvedený príklad je samozrejme extrémny, ale dovoľím si tvrdiť, že v mnohých prípadoch asi nebude mať až tak ďaleko od reality. Rád by som sa teraz pozrel na chyby, ktoré podľa môjho názoru a osobnej skúsenosti môžu ľahko spôsobiť neúspech Scrumového projektu.

Komunikácia so zákazníkom

Všetky agilné metodiky kladú veľký dôraz na úzku spoluprácu so zákazníkom. Klasické metodiky vývoja zahrňajú zákazníka do procesu iba v počiatočných fázach (špecifikácia, analýza, prípadne návrh) a ďalej sa zákazník dostane k produktu až po odovzdaní. To so sebou prináša hneď niekoľko rizík – nesplnenie požiadaviek, vytvorenie inej funkcionality, ako požaduje zákazník alebo možné pozlátenie systému.

Agilné metodiky vravia, že zákazník by mal byť zapojený do procesu vývoja neustále. V ideálnom prípade by mal byť človek od zákazníka členom tímu, v menej ideálnom by mal mať tím možnosť často komunikovať so zákazníkom. Niekedy však povedzme niektorý z manažérov usúdi, že mať v tíme človeka od zákazníka je v podstate iba plytvanie peňazí, keďže väčšinu času aj tak nič nerobí. Možno by stačilo, keby sa tím stretával so zákazníkom len raz týždenne. A už nám vznikajú prvé problémy.

Pri vývoji softvéru (hlavne pokiaľ ide o softvér na zákazku, najlepšie v nejakej špecifickej doméne ako je napr. bankovníctvo) vznikajú drobné nejasnosti prakticky denne. Vývojári sa nemajú na koho obrátiť a tak buď skúsia problém vyriešiť na základe vlastného úsudku (ktorý evidentne nebude vždy správny) alebo počkajú s riešením na najbližšie stretnutie so zákazníkom. Výsledkom je buď priame zdržanie alebo nepriame zdržanie spôsobené nutnosťou upravovať existujúce časti softvéru (o možných chybách, na ktoré sa príde až dodatočne, ani nehovoriac).

Možno si poviete, že také drobnosti nemôžu mať vplyv na výsledok projektu. Stačí však, aby takýchto prípadov bolo dostatočné množstvo, a dopad na projekt začne byť evidentný.

Dovoľte mi uviesť vlastnú skúsenosť z práce. Podieľam sa na vývoji jedného projektu, ktorý začal v roku 2008 a od tej doby prešiel mnohými zmenami a vylepšeniami. Spočiatku sme stretnutia so zákazníkom usporadúvali len vo fáze špecifikácie, prípadne návrhu (hlavne kvôli grafickému návrhu). Veľmi často vznikali počas vývoja nejasnosti, ktoré vyžadovali komunikáciu so zákazníkom, tá však musela počkať na najbližšie stretnutie. Neskôr, keď sa zákazník presťahoval do rovnakých priestorov ako naša firma, začali sme konzultovať so zákazníkom aj počas vývoja. Zákazník je teraz prakticky členom tímu a výsledok je omnoho lepší – pridanie novej funkcionality je rýchlejšie a menej chybové.

Komunikácia v rámci tímu

Podobne ako komunikácia so zákazníkom, aj komunikácia v rámci tímu je veľmi dôležitá. Opäť môžem poskytnúť aj vlastnú skúsenosť, tentoraz zo školského tímového projektu, ktorý v čase písania tejto práce práve prebieha. Na projekte pokračujeme po tíme z minulého roku a snažíme sa vyvíjať projekt metodikou Scrum. Lenže prvých niekoľko

šprintov dopadlo prakticky katastrofálne, keďže sme neukázali prakticky žiadny výsledok. Jednu z hlavných príčin vidím v nedostatočnej komunikácii v tíme.

Neraz sa počas týchto niekoľko týždňov opakovalo, že niektorý člen tímu mal s niečím problém a namiesto toho, aby skúsil problém vyriešiť s ostatnými, skúšal ho vyriešiť sám. Pritom iný člen tímu mu bol schopný pomôcť, ale o žiadnom probléme nevedel. Výsledkom bolo zbytočné zdržanie. Niekoľko krát sa tiež stalo, že dvaja ľudia pracovali na tom istom probléme, čo spôsobilo ďalšie zdržanie. Ďalším problémom je tiež iná „pracovná doba“ jednotlivých členov tímu, ktorá umožňuje len asynchrónnu komunikáciu (e-mail, fórum a pod.), čo spôsobuje ešte ďalšie prieťahy. Rozsiahlejšia komunikácia v rámci tímu by projektu jednoznačne prospela.

Metodika Scrum predpisuje denné stretnutia tímu, ktoré obsahujú debatu o tom, čo sa spravilo za predchádzajúci deň, čo sa má spraviť za aktuálny deň a čo bráni tomu, aby sa to naozaj spravilo. Tieto stretnutia sú podľa mňa veľmi šikovné a majú potenciál prudko zvýšiť výkonnosť celého tímu.

Granularita úloh

Oblasťou, kde sa spraví chyba ľahko, ale môže mať veľké následky, je plánovanie úloh. Konkrétne mám na mysli granularitu naplánovaných úloh. Hlavný problém vidím v tom, ak je granularita úloh príliš vysoká (môže to súvisieť aj s príliš dlhým šprintom – napríklad mesiac bude pravdepodobne príliš dlhá doba na jeden šprint). Najväčším rizikom tu je zlé odhadnutie náročnosti úlohy, čo má za následok časový sklz.

Tento krát môžem prispieť vlastnou skúsenosťou zo školského tímového projektu, ako aj z práce. Začnem tímovým projektom, ktorý som už spomínal. Okrem problémov s komunikáciou sa pod náš časový sklz podpísali aj problémy s naplánovaním úloh. Niektoré naplánované úlohy sa ukázali ako triviálne a iné zase namiesto plánovaného jedného šprintu zabrali šprinty tri. Výsledkom je nedodržanie časového harmonogramu.

Veľmi podobná situácia nastala viac krát v práci, kedy sa ma projektový manažér pýtal, ako dlho mi bude pridelená úloha trvať. Ja som sa pokúsil na základe popisu úlohy odhadnúť, ako dlho bude trvať. Asi uhádnete, ako to dopadlo. Áno, prakticky každý krát, kedy som si pred odhadom úlohu nerozdelil na drobné, bol môj odhad príliš optimistický a úlohu som nestihol dokončiť v termíne.

Pre efektívne naplánovanie úloh do šprintov je teda nevyhnutné úlohy rozdeliť na podúlohy. To je aj dôvod pre krátke trvanie šprintov. Čím dlhší je jeden šprint a čím vyššia granularita úloh, tým menej pružný je proces vývoja a tým vyššia je pravdepodobnosť, že plán je zle odhadnutý a projekt nedodrží časový harmonogram.

Prístup

Poslednou oblasťou, ktorej by som sa rád pozrel na zúbok, je prístup. Pod pojmom prístup myslím prístup jednotlivých členov tímu, ako aj manažmentu, k implementácii metodiky Scrum. Existujú v podstate dva extrémny a oba môžu mať na projekt katastrofálne následky.

Prvým extrémom je negativita. Jednak to môže byť negativita niektorých členov tímu (v horšom prípade aj celého tímu). Denné stretnutia sa odfláknu alebo sa vôbec nekonajú, úlohy sa neplnia, hodiny sa nevykazujú podľa reality, entuziazmus klesá, pracovné nasadenie takisto a výsledkom je neúspešný projekt. Tiež môže ísť o negativitu

manažmentu, ktorý neverí v úspech projektu vyvíjaného metodikou Scrum. Túto negativitu môže manažment ľahko preniesť na celý tím a aj pokiaľ ju neprenesie, často to skončí tím, že samotný manažment sabotuje vývoj projektu tým, že tímu nevychádza v ústrety alebo im dokonca hádže poľená pod nohy.

Druhým extrémom je určite prílišný entuziazmus a nasadenie Scrumu na projekty, na ktoré nie je vhodný. Môže ísť napríklad o projekty s obmedzeným rozpočtom, prípadne veľmi krátke projekty. Jednoducho povedané treba uvažovať trpezlivo.

Čo z toho vyplýva?

Projekty zlyhávali, zlyhávajú a stále zlyhávať budú. Niektoré projekty sú jednoducho určené na neúspech a nikto na tom nič nezmení. Avšak je dobré snažiť sa o to, aby projekty, ktoré majú šancu na úspešné ukončenie, aj úspešne skončili.

Jedným zo spôsobov, ako projekt úspešne skončiť, je použiť správnu metódu vývoja. V mnohých prípadoch je najvhodnejšou voľbou niektorá z agilných metód. Avšak je veľmi dôležité dobre si premyslieť, či je zvolená agilná metóda naozaj vhodná na daný projekt vzhľadom na jeho rozsah, dostupné zdroje, ľudí, rozpočet a čas.

A pokiaľ sa na vývoj projektu použije niektorá metóda, dodržanie jej odporúčaní je najlepšou cestou, ako doviest' projekt úspešne do cieľa. Teda v prípade Scrumu je najmäjšie tesne spolupracovať so zákazníkom, dobre si rozdeliť a naplánovať úlohy, usporadúvať denné stretnutia a potom stačí už len poctivo pracovať.

A aby som nezabudol - pozitívny prístup tiež nezaškodí.

Použitá literatúra

1. Beck, K., et al.: *Manifesto for Agile Software Development* (2001). <http://www.agilemanifesto.org> [cit. 2011-11-25]
2. Coram, M., Bohner, S.: The Impact of Agile Methods on Software Project Management. In: *Proc. of the 12th IEEE International Conference and Workshops on Engineering of Computer-Based Systems (ECBS '05)*. IEEE Computer Society, Washington, DC (2005), 363-370.
3. Dan Galorath on Estimating: *Software Project Failure Costs Billions. Better Estimation & Planning Can Help* (2008). <http://www.galorath.com/wp/software-project-failure-costs-billions-better-estimation-planning-can-help.php> [cit. 2011-11-25]
4. IT Cortex: *Failure Causes*. http://www.it-cortex.com/Stat_Failure_Cause.htm [cit. 2011-11-25]

Annotation

Is Scrum the culprit?

This paper talks about the causes of software project failures and why agile methods of software development help less than they should. Author argues that a lot of failed agile projects can be traced to incorrect or incomplete implementation of given agile methods and gives his insight into causes of the failures.