

KONTROLA BUDÚCNOSTI SCRUM PROJEKTOV

Dúfaj v najlepšie, očakávaj najhoršie.

Marek Hlaváč

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

Abstrakt. *Budúcnosť softvérových projektov je dôležitá, ale často krát zabúdaná a podceňovaná téma. Z pohľadu agilného vývoja v kontexte Scrum projektov ide o podstatnú a neodmysliteľnú súčasť, na ktorú sa prihliada počas každej fázy vývoja. Úspešná budúcnosť je zabezpečovaná prostredníctvom monitorovania a kontroly projektov. Cieľom monitorovania a kontroly je použitie vhodných techník a aplikácií adekvátnych metrík so zámerom predčasnej identifikácie rizík a ich skorého odstránenia. A práve toto je dôvod prečo existuje mnoho neúspešných projektov. V Scrum projektoch sa otvárajú monitorovaním nové možnosti výberu konkrétnych techník pre špecifické situácie. Dôvodom je zaznamenávanie podstatných hodnôt, ktoré nám umožnia sledovať stav a progres projektov vzhľadom na ich agilný charakter. Z tohto hľadiska je možné definovať otázky, ktorých odpovede sú nutnou podmienkou pri kontrole budúcnosti Scrum projektov. Aké sú spôsoby monitorovania Scrum projektov? Ako predčasne odhaliť potenciálne riziká? Načo si treba pri monitorovaní dávať pozor? V čom tkvie sila kontroly projektu?*

Kľúčové slová: *Scrum, monitorovanie, kontrola, softvérový projekt, progres*

Úvod

Monitorovanie softvérových projektov je jedna z tém softvérového inžinierstva, ktorá v prípade správneho zvládnutia zvyšuje šancu úspešného ukončenia projektov a ich následného nasadenia do prevádzky. Počas vývoja je vďaka skorej identifikácii problémov

2 Marek Hlaváč

možné predčasné vyhnutie sa konkrétnym úskaliam projektu, prípadne minimalizácie ich negatívnych účinkov na samotný projekt.

Monitorovanie zahŕňa viacero činností, ktorými je nutné sa zaoberať. Najdôležitejšou je výber správneho spôsobu vykonávania monitorovania so zámerom získania relevantných informácií, ktoré sa po dôkladnej analýze v procese kontroly projektu uplatňujú pri rozhodovaní a plánovaní ďalšieho vývoja. Nemenej dôležité je použitie vhodných nástrojov a metrík, ktoré sú prostriedkom monitorovania a hlavným ukazovateľom stavu projektov z viacerých uhlov pohľadu.

Agilné prístupy vývoja softvéru posúvajú techniky monitorovania a kontroly projektov o niečo ďalej. Je to zapríčinené prispôbovaním sa zmenám požiadaviek kladených na vyvíjaný softvér zákazníkmi. Zmeny majú vplyv ako aj na krátkodobé, tak aj dlhodobé ciele. Dôsledkom je dynamické prispôbovanie vývoja na základe nových skutočností.

Rozšírenou metódou agilného prístupu vývoja je Scrum, ktorého cieľom je rozčlenenie softvérových projektov na úlohy so zámerom zjednodušenia a ľahšieho porozumenia riešenému problému. Scrum otvára nové možnosti z hľadiska monitorovania projektu a projektového manažmentu. Na meniace sa podmienky v prostredí Scrumu je nutné nájsť a aplikovať dômyselnejšie spôsoby monitorovania a kontroly vzhľadom na budúcnosť projektov s cieľom odhaliť dôležité špecifiká progresu projektu a vhodne ich použiť pri ďalšom vývoji.

Kontrola v agilnom vývoji

Podľa [1] spočíva princíp agilného prístupu vývoja na niekoľkých princípoch, ktoré zásadne ovplyvňujú projekt a jeho charakter počas celého vývoja. Avšak, v agilnom prístupe nejde o ich dogmatické nasledovanie a plnenie. Dôraz sa kladie na samotný tím a jeho rozhodnutia pri výbere vhodnej cesty a progresívnej zmene jej smeru so zámerom odstránenia nedostatkov. Agilný prístup tak získava formu šablóny, ktorá kladie dôraz na zlepšenie vývojových procesov a zvyšuje kontrolu nad projektom, pričom umožňuje variabilne sa meniť na základe výstupov, ktoré je možné počas vývoja sledovať.

Scrumom proti prúdu

Scrum si získal v rámci manažmentu softvérových projektov množstvo priaznivcov a myslím si, že to je zaslúžené. Na rozdiel od tradičných prístupov k vývoju softvéru sa zameriava na podstatné vlastnosti a odbreňuje sa od zbytočných formalizmov [5]. Ruka v ruku s týmto odbremením je zvýšená pozornosť, ktorú je nutné zamerať na samotný progres projektu v každej jeho etape. Z pohľadu monitorovania sa tu otvárajú nové možnosti, pretože je možné sledovať veľké množstvo činností a s použitím vhodných metrík podporovať rozhodovanie a umocňovať tak kontrolu nad projektom.

Na základe skúseností získaných počas vývoja na tímovom projekte v študijnom prostredí, v ktorom používame Scrum si dovoľím tvrdiť, že z pohľadu kontroly projektu je Scrum ideálny pre menšie tímy skladajúce sa z počtu členov, ktorý nepresahuje viac ako desať ľudí. Dôvodom je, že malé tímy vytvárajú ideálne prostredie pre monitorovanie projektu, pretože spracovanie výstupov do rozumných uzáverov je uľahčené ich množstvom a viditeľnosťou pri použití konkrétnych monitorovacích techník.

Víťazstvo v šprinte nie je byť prvým, ale úspešným

Šprinty sú ťažiskom a základnou jednotkou práce v Scrum. V nich sa vykonáva najpodstatnejšia časť prác. Ich dôležitou vlastnosťou je časovo obmedzené vykonávanie. Na základe tejto vlastnosti môžeme aplikovať monitorovacie techniky, ktoré majú špecifický časový charakter.

Vzhľadom na monitorovanie sa mi ako najdôležitejšie fázy šprintu osvedčujú začiatok a koniec šprintu, resp. plánovanie úloh v šprinte a prezentácia produktu s retrospektívnym stretnutím. I keď to nemusí byť zjavné, tak práve tieto fázy sú z hľadiska monitorovania nanajvyš dôležité. Dôvodom je, že sa jedná o stretnutia, ktorých sa zúčastňujú všetci členovia tímu a roly vystupujúce pri vývoji Scrumom. To znamená, že môžeme jedinečným spôsobom konfrontovať získané hodnoty v tabuľkách a grafoch s reálnou situáciou. Ide o špecifický spôsob skúšky správnosti metódou kritickej konfrontácie.

Úspech šprintu je kapitola sama o sebe, keďže odpovedí na otázku čo je to úspešný šprint by mohlo byť mnoho. Avšak, niektoré odpovede majú niečo do seba. Podľa [3] celá veda spočíva v plánovaní šprintu, v ktorom je nutné prihliadať na tempo tímového vývoja, plnenie príslušných zodpovedností a ponechané otvorené dvierka pre ďalšie možnosti a nepredvídateľné situácie. Toto sa mi zdá ako správny spôsob, keďže počíta s optimistickým a zároveň aj pesimistickým scenárom a snaží sa tak vtesnať práce na šprinte niekde medzi.

Ďalším zo spôsobov, ktorý sa podľa môjho názoru dôležitosťou vyrovná stretnutiam je použitie samotného vyvíjaného produktu ako ukazovateľa stavu projektu [6]. Stav produktu nám môže prezradiť mnoho skrytých informácií, či už ide o úroveň kvality implementovaných funkcionalít alebo počet splnených požiadaviek. S týmto súvisí aj prezentácia produktu na konci každého šprintu, ktorá slúži ako akýsi regulátor vychýlenia sa od správneho smeru vývoja. Prezentácia zahŕňa účasť produktového vlastníka, to v konečnom dôsledku znamená, že ide o formu konfrontácie aktuálneho stavu projektu s konečným, keďže produktový vlastník je takpovediac jediný, ktorý má predstavu o finálnom produkte.

Komunikácia ako sprostredkovateľ tímovej kultúry

Dovolím si tvrdiť, že bez komunikácie by neexistoval žiaden softvérový projekt. Avšak, bez tímovej komunikácie by neexistoval žiaden Scrum. Základným elementom Scrumu je tímová komunikácia, ktorá sa deje v každom procese. A práve verbálna komunikácia je jedným z ďalších nástrojov, ktoré nám vo veľkej miere pomáhajú pri monitorovaní a kontrole projektov, či už ide o denný Scrum míting, plánovanie šprintu, prezentáciu produktu alebo retrospektívu po šprinte.

Denné Scrum mítingy, ktorých cieľom je stručná diskusia ohľadom postupu na projekte a úlohách za posledný deň nás môžu včas alarmovať pri zárodkoch problémov. V týchto prípadoch je nutné začať premýšľať ohľadom cieľu a náplne ďalšieho šprintu alebo v prípade veľkého problému ohľadom preplánovania úlohy, či celého šprintu.

Ako odhaliť problémy?

Mnoho softvérových projektov čaká neúspešný koniec, ktorý môže byť zapríčinený viacerými faktormi. Neúspešné projekty a riešenia rôznych problémových situácií však slúžia ako ideálne príklady, na ktorých je možné poučiť sa z chýb a následne v nových projektoch získané znalosti využiť so zámerom predčasnej identifikácie problémov. Ja si myslím, že prežiť problémy na vlastnej koži je rovnako dôležité ako skúmanie chýb na základe skúsenosti iných ľudí a projektov. I keď sa to môže javiť ako menej efektívna alternatíva, tak v budúcnosti nás môže pripraviť na tie najrôznejšie problémy a situácie.

Kamene úrazu

Prvou skupinou problémov, ktoré sú minimálne, ale časom naberajú na sile sú problémy spôsobené nesprávnym plánovaním, kedy sa jednotlivé úlohy buď podcenia, čo má za dôsledok nestihnúť ich úspešného dokončenia do konca termínu iterácie alebo sa dokončia no s následkami výraznej nefunkčnosti v ďalších iteráciách [2]. V rámci tímového projektu sa nám prihodila rovnaká situácia, keď sme si naplánovali úlohy príliš optimisticky, no nakoniec sme ich na poslednú chvíľu zvládli, avšak bez výraznejšieho testovania a overenia správnosti. Tento fakt nás stál v ďalšej iterácii vytvorenie dodatočných úloh so zapracovaním výrazných opráv.

Ďalšími problémami, ktoré nás postihli sa týkali nesprávneho určovania priorit jednotlivých úloh a súvisia s prekračovaním hraníc projektu. I keď tento fakt by mal byť minimalizovaný produktovým vlastníkom, tak môže nastať situácia kedy sa prioritujú úlohy, ktoré sú v konečnom dôsledku pre aktuálnu iteráciu nepodstatné a nemajú nič spoločné s jej cieľom. Podobná skúsenosť nás stretla na tímovom projekte, keď sme predčasne vytvorili návrh plánovania agentov a v ďalšej iterácii sme boli po prehodnotení situácie a úprave jadra aplikácie nútený návrh agenta od začiatku prerobiť.

Poslednou a podľa môjho názoru najdôležitejšou skupinou problémov sú problémy, ktoré vznikajú na základe neviditeľnosti softvéru, respektíve ťažkej identifikácie väzieb jednotlivých úloh v projekte. Tieto problémy sú implicitné a súvisia s povahou samotných projektov. Jedná sa o vážny problém aj z pohľadu Scrumu, keďže pri plánovaní nie je možné určiť a prioritizovať niektoré úlohy, po prípade presunúť ich plánovanie do ďalšej iterácie. Keď sme vytvárali prototyp riešenia na tímovom projekte, tak sme si pri plánovaní šprintu neuvedomili, že sú niektoré úlohy prepletené a jedna s druhou súvisí. Dôsledok bol ten, že na niektorých úlohách sa museli podieľať aj ľudia, ktorý pre ne neboli určené a celý vývoj sa tak výrazne spomalil až zasekol. Riešením je rozumná granularizácia úloh s prediskutovaním náplne úloh pri plánovaní šprintu.

Monitorujeme, aby sme boli lepší

Cieľom monitorovania je zvýšenie kontroly nad jednotlivými projektmi a minimalizácie negatívnych efektov počas vývoja. Avšak nie vždy je monitorovanie projektu v týchto bodoch úspešné, keďže stav projektu je dynamický a časovo nestály. Preto je podľa mňa potrebné zobrať do úvahy aj ďalší fakt, ktorým je prispôsobovanie monitorovania samotnému projektu, a nie naopak. To znamená, že čo nám pomohlo naposledy, nemusí nutne znamenať, že to bude nápomocné aj v budúcnosti a treba byť nato pripravený.

Ludia nie sú čísla

Podľa [4] je nutné pri monitorovaní projektov brať na vedomie, že výsledky prác jednotlivých členov Scrum tímu sú údaje, ktoré sú časom premenlivé. Princíp myšlienky je možné rozšíriť o upozornenie, že ľudia by sa mohli pri ich používaní popáliť. Keďže výsledky sú len čísla v grafoch, tak by sme sa nemali na namerané hodnoty spoliehať v plnej miere, ale je nutné, aby sme ich používali ako ukazovateľ a boli si vedomí ich vyjadrenia v reálnom kontexte pri plánovaní a rozdelení prác.

V tímovom projekte sa nám podarilo prekonať podobný problém. Pri plánovaní sme odhadli dve úlohy príliš optimisticky a v súčte boli pridelené jednej osobe, zhodou okolností práve mne. Po začatí prác ma čakal nemilý fakt. Bližším preskúmaním úloh som si uvedomil, že odhad bol veľmi nepresný a je možné, že čas reálnej práce sa natiahne takmer na dvojnásobok odhadu. V rámci časových možností nebolo možné obe úlohy úspešne dokončiť. Riešením tohto typu problémov je pri skorej identifikácii problému preplánovanie, po prípade rozdelenie úlohy a presunutie časti inému členovi tímu.

Každý sa učí na svojich chybách

Podľa môjho názoru je jednou z najdôležitejších fáz a vlastností Scrumu retrospektíva, ktorá je vykonávaná v rámci ukončenia každého šprintu. Jej dôležitosť spočíva v tom, že diskutuje aktuálne ukončený šprint so zámerom identifikácie a následného vyriešenia problémov, ktoré sa počas tohto šprintu objavili. Z hľadiska monitorovania ide o jedinečný prístup, v ktorom je možné zaznamenať značné množstvo dôležitých informácií.

Ak sa nad tým hlbšie zamyslíme, tak je možné retrospektívu označiť ako kolektívny spôsob učenia sa prostredníctvom tímovej inteligencie a nadobudnutých znalostí za účelom neustáleho zlepšovania sa a rastu. Z môjho pohľadu ide o ideálny spôsob aplikácie kontroly budúcnosti projektov, keďže je vykonávaná celým tímom, ktorý sa snaží riešiť problémy so zámerom ich odstránenia a zlepšenia projektových procesov.

Záver

Monitorovanie a kontrola projektov má vo softvérovom vývoji opodstatnený zmysel. Pri Scrume ide o elementárnu súčasť vývoja, ktorej zvládnutie ide ruka v ruku s úspechom projektu. Dobré zvládnutá predčasná identifikácia rizík, minimalizácia ich efektov a adaptívne prispôsobovanie vývoja pre konkrétne projekty hrá kľúčovú rolu pri kontrole budúcnosti v Scrum projektoch.

Na základe vlastnej skúsenosti viem, že v agilných projektoch sa mnohokrát zabúda práve na kontrolu a smerovanie projektu. Na úrovni tímu sa ignoruje stav projektu zo všeobecného hľadiska. Aplikujú sa praktiky, ktoré sú nepoužiteľné pre konkrétne projekty a často sa stáva, že je opomínané kritické zamyslenie, či objektívne zhodnotenie projektovej situácie.

Cieľom eseje je opísanie možností kontroly Scrum projektov so zámerom ich úspešného ukončenia, predstavenie základnej problematiky monitorovania v rámci Scrum projektov pri vykonávaní tímových procesov a uvedenie eventuálnych spôsobov, ktoré ponúkajú riešenie pre konkrétne problémy. Je dôležité si uvedomiť, že každý Scrum projekt je iný a čo platí pre jeden, nemusí nutne platiť pre iný.

Použitá literatura

1. Beck, K.: Manifesto for agile Software Development. [Online] 2001. [Datum: 24. 11 2011.] <http://agilemanifesto.org/>.
2. Dinakar, K.: Agile Development: Overcoming a Short-Term Focus in Implementing Best Practices. *Information Systems Journal*. 2009, s. 583.
3. Kniberg, H.: *Scrum and XP from the Trenches*. C4Media, 2007, s. 27.
4. Ktata, O., Lévesque, G. Designing and Implementing a Measurement Program for Scrum Teams: What do agile developers really need and want? *ACM International Conference Proceeding Series*. 2010, s. 104.
5. Rising, L., Janoff, N.S.: The Scrum Software Development Process for Small Teams. *Software, IEEE*. 2000, 17, s. 27.
6. Thomas, S.: Agile Project Monitoring and Control. [Online] 16. 06 2008. [Datum: 24. 11 2011.] <http://itsadeliverything.com/agile-project-monitoring-and-control>.

Annotation

Control the future of Scrum projects

The future of software projects is important, but often forgotten and underestimated topic. From the perspective of agile development in the context of Scrum projects it's a significant and indispensable part which is pointed out in each development phase. A successful future is assured through project monitoring and controlling. The aim of project monitoring and controlling is early identification of risks and their early removal by using appropriate techniques and metrics. This is the reason why there are so many project failures. For Scrum processes this opens up new possibilities in the selection of specific techniques for different situations. The point is to record essential values that allow us to monitor the status and progress of Scrum projects and their dynamic nature. From this point of view we can define the questions with competent answers which are a prerequisite for future control of Scrum projects. What are the ways of monitoring Scrum projects? How to detect potential risks? Why you should be careful in project monitoring? Where lies the power of project controlling?