

MONITOROVANIE PROJEKTU METÓDOU BODOV PRÍPADOV POUŽITIA

Oplatí sa postupovať podľa poučiek?

Peter Mačuga

Slovenská technická univerzita
Fakulta informatiky a informačných technológií
Ilkovičova 3, 842 16 Bratislava
xmacugap[zavináč]is.stuba.sk

Abstrakt. Esej ponúka subjektívny názor na techniku odhadovania resp. plánovania projektu- analýza vytvorenej hodnoty pomocou bodov prípadov použitia. Samotná technika je načrtnutá iba v hrubých obrysoch, na jej vysvetlenie je nepostačujúca. Eseji dominuje autorov negatívny pohľad na túto metódu podporený faktami, osobnými skúsenosťami a jeho subjektívnym názorom. Esej analyzuje a postupne vypracová, prípadne nachádza alternatívy ku kladným stránkam tejto techniky. Dokazuje, že vždy ešte záleží na ľuďoch, na ich skúsenostiach a pocitoch pri plánovaní (monitorovaní), nie je ich možné nahradiť žiadnymi vzorcami, postupmi.

Kľúčové slová: *esej, UCP, use case points, earned value analysis*

Úvod

Softvérové projekty v súčasnosti naberajú na robustnosti. Prirodzene to zvyšuje riziko zlyhania. Odoberatelia si toto riziko uvedomujú. Vedia, že projekt musí byť dobre navrhnutý, špecifikovaný. Tým sa ich úloha nekončí. Vo fáze vývoja softvéru musia svojím spôsobom zasahovať do vývoja, aby predišli neúspechu. Jedná sa konkrétne o monitorovanie projektu. Na to, aby si developeri a stakeholderi dorozumeli, používa sa ustálená metrika pre monitorovanie projektu. Je ich viacero, ale sú intuitívne pochopiteľné aj pre laikov v IT oblasti. Monitorovanie softvérového projektu sa prelína s inými oblasťami manažmentu softvérových projektov. Konkrétne ide o riadenie, plánovanie, manažment rizík. Monitorovanie projektu prispieva k včasnej identifikácii problémov,

2 Peter Mačuga

následne na základe výsledkov monitoringu niekedy treba prehodnotiť doterajší plán projektu, prípadne zmeniť priority úloh. Ako však určiť náročnosť úlohy (projektu)? Je zrejmé, že metriky založené na dĺžke programu neposkytujú objektívny pohľad. V eseji predstavím a zhodnotím metriku na monitorovanie projektu: body prípadov použitia (angl. *use case points*).

Prečo monitorovať?

Vývoj softvéru je náročný proces, na ktorý vplývajú rôzne faktory. Nemožno iba rozdeliť programátorské úlohy, pracovať a o 2 mesiace sa stretnúť a pokúsiť sa o integráciu jednotlivých častí. Proces vývoja softvéru musí byť niekým riadený, kontrolovaný. Týmito osobami sú projektoví manažéri. Okrem zabezpečenia personálnych a technických otázok dbajú o dodržiavanie plánu projektu, reagujú na zmeny požiadaviek zákazníkov, na zmeny v personálnom obsadení tímov. Vyššie spomenutý dohľad nad dodržiavaním plánu projektu sa nazýva *monitorovanie*. Monitorovanie projektu je teda dôležité na to, aby manažér mohol sledovať progres projektu alebo jeho jednotlivých častí.

Monitorovanie je proces, ktorý prináša výsledky hneď dva krát. Prvý krát je to pri monitorovaní samotného projektu (včas odhaľuje problémy čo sa týka splnenia plánu projektu alebo problémy s kvalitou softvéru), druhý krát je monitorovanie užitočné pri lepšom nastavení parametrov odhadov budúcich projektov. Monitorovaním získava manažér prehľad o výkonnosti zamestnanca alebo tímu. Budúce projekty budú môcť byť presnejšie naplánované.

Ako monitorovať?

Všetky spoločnosti zaoberajúce sa vývojom softvéru nahlas deklarujú svoju hlavnú zásadu: zákazník je na prvom mieste. Od toho sa odvíja hodnotenie progresu projektu, čiže koľko zo zákazníkovoých očakávaní sa zatiaľ naplnilo. Anglický ekvivalent sa nazýva „*earned value analysis*“, doslovne analýza vytvorenej hodnoty. V bežnom jazyku zrozumiteľný termín, ale na jeho objektívne zmeranie treba zaviesť jednotky. V čom bude meraná vytvorená hodnota? Počty riadkov kódu alebo počet implementovaných modulov v systéme poskytujú iba chabý odhad, ktorý sa reálne vôbec nemusí blížii skutočnej hodnote a najmä vynaloženému úsiliu na implementáciu danej funkcionality. Riešenie predstavujú body prípadov použitia.

Výpočet bodov prípadov použitia

Body prípadov použitia je metrika na analýzu vytvorenej hodnoty. Dá sa z nich spočítať koľko času treba na dokončenie projektu alebo jeho časti. Na jej použitie treba pri špecifikácii vytvoriť prípady použitia, následne cez niekoľko krokov ohodnotiť hráčov, samotné prípady použitia, vplyv technických a environmentálnych faktov [4].

Obrovskou výhodou je spomenutá náväznosť na plánovanie, na kontrolu dodržiavania plánu, možnosť ohodnotenia výkonnosti tímov za jednotku času. Znie to dobre, projektový manažér bude mať k dispozícii detailný prehľad o plnení plánu, výkonnosti pracovníkov.

Poučky spomínajú obmedzenia, nevýhody tejto techniky. Nevýhodou je jej robustnosť, koľko úsilia treba venovať na ohodnotenie navrhnutej funkcionality.

1. treba navrhnuť všetky relevantné prípady použitia. Tento návrh môže zabrať až 20% úsilia na projekte [1].
2. vytvorené prípady použitia musia byť na rovnakej úrovni granularity, aby výsledok predstavoval reálnu náročnosť úlohy (projektu). To sa dosahuje veľmi ťažko, najmä ak uvažíme že na tom bude pracovať viacero ľudí.
3. tvorba prípadov použitia predstavuje aktivitu bez výslednej hodnoty v produkte. Tento bod späť s bodom 1. Aktivita bez výslednej hodnoty je niečo, čo zákazník nešpecifikoval, za čo nezaplatí [2]. Preto je cieľom tieto aktivity obmedziť na minimum. Zdanlivo bez hodnoty pôsobí aj vytváranie prípadov použitia za účelom ohodnotenia náročnosti projektu. Kým pre zákazníka to nemá žiadnu hodnotu, vývojárom pomáha v organizácii práce.

Ako vlastne?

Po predstavení kladov a záporov tejto metriky je na projektových manažéroch, aby sa rozhodli, či chcú venovať desiatky percent času na vývoj produktu tvorbe a analýze prípadov použitia, alebo či oblasť odhadovania bude prenechaná na intuíciu, skúsenosti skúsených členov tímov. Táto intuícia sa v drvivej väčšine prípadov ukazuje ako správna [3].

Metódu bodov prípadov použitia nevyužívame ani v našom Tímovom projekte. Prečo? Pravdepodobne prevážili negatívne stránky metódy nad kladmi. Alebo možno preto, pretože človek je od prírody lenivý tvor a o programátoroch to platí dvojnásobne. Neradi píšeme dokumentácie, formality. Netrúfam si tým však tvrdiť, že nás zachráni skúsenosti pri plánovaní nášho tímového projektu, ale učíme sa. Argumenty proti metóde UCP nasledujú v podkapitolách.

Agilný vývoj

V posledných rokoch sa pri vývoji softvéru presadzuje agilný štýl vývoja softvéru. Agilné tímy si naplánujú na dvojtýždňový šprint, funkcionality ktorú treba vykonať a pri uzavretí šprintu bude plán vyhodnotený. Krátke (napríklad dvojtýždňové) šprinty zabezpečujú, že ak sa identifikuje problém (či už funkčný alebo problém čo sa týka nedodržania plánu), pokus o vyriešenie problému bude začlenený do nasledujúceho šprintu. Rovnako sa postupuje aj pri zmene požiadavky od zákazníka. Krátke šprinty (cyklus našpecifikovanie, plánovanie, implementácia, testovanie, vyhodnotenie) zabezpečujú pomerne rýchlu reakciu na vzniknuté problémy. Touto rýchlou reakciou by sa dal eliminovať jeden z argumentov za výpočet bodov prípadov použitia, konkrétne detailné naplánovanie, minimalizácia rizika časového sklzu. To nakláňa miskú váh proti technike bodov prípadov použitia.

A čo zákazníci?

Ako zákazníci budú participovať na monitorovaní projektu? Tu sa opäť vynára ako odpoveď agilný vývoj. Po ukončení šprintov má tím vždy pripravený softvérový prototyp s funkcionalitou dovedy vytvorenou. Prototyp vo vyšších fázach vývoja býva testovaný koncovými používateľmi. Zákazníci teda majú možnosť monitorovať priebeh vývoja softvéru aj takýmto spôsobom. Treba poznamenať, že toto nie je doménou iba agilného vývoja. Všetky ostatné metódy vývoja poskytujú funkčné prototypy softvéru po istom časovom období. Tým, že zákazníci dostávajú to, čo ich zaujíma spĺňame zásadu „customer first“ a môžu zhodnotiť súčasný stav.

Zatiaľ sa ako posledný z bodov PRE analýzu vytvorenej hodnoty pomocou bodov prípadov použitia drží možnosť určovať na základe vytvorenej hodnoty rýchlosť (kvalitu) tímu (pracovníka). V náväznosti na to v budúcnosti nebude možné vytvoriť dobrý plán.

Skúsenosti, intuícia

Skúsenosti boli už vyššie spomenuté ako prínos pre plánovanie [3]. Tento bod je uvedený, aby odrazil posledný oporný stĺp pre analýzu vytvorenej hodnoty pomocou bodov prípadov použitia. Nebude však predstavená alternatíva k určeniu „rýchlosti“ tímu, ale celkovo spochybnený výpočet bodov prípadov použitia.

Možnosť plánovať projekt na základe skúseností z predošlých projektov je akceptovaný štýl plánovania. Na chvíľu však prijmime postulát, že plánovanie sa robí na základe analýzy a výpočtu bodov prípadov použitia. Je to zložitý a zdĺhavý proces, ktorý okrem objektívnych vstupov používa na výpočet ľudsky (skúsenosťami) určené váhy hráčov, subjektívne ohodnotenie technických a environmentálnych faktorov a niekoľko bulharských konštánt. Každý jeden vzorec vo výpočte obsahuje nejaký subjektívne ohodnotenie. Konkrétne:

$$\begin{aligned}UUCP &= \sum \text{transakcia} + \sum \text{hráč} \\TCF &= 0,6 + (0,01 * TFaktor) \\EF &= 1,4 + (-0,03 * EFaktor)\end{aligned}$$

V prvom prípade sa váhy transakcií a hráčov určujú človekom (v tomto bode sa prejaví rôzna granularita zápisu prípadu použitia). Dve ďalšie rovnice sčítajú a násobia niečo s nejakými číslami. To niečo určí opäť človek.

Prečo sú dané presne vzorce, keď hodnoty do nich sú subjektívne určované skúsenosťami ľudí? Menej času zaberie plánovať projekt od začiatku iba na základe skúseností ako pokus objektívne niečo zrátať a pritom vo vzorcoch vystupuje subjektívne ohodnotenie niekoľkokrát.

Záver

Esej ukázala, že niet nad skúsenosti a názory ľudí. Nech sa bude pri určovaní vynaloženej hodnoty postupovať akokoľvek, pri všetkých technikách bude hrať kľúčovú úlohu ľudský faktor, názor podložený skúsenosťami. Nech je akokoľvek dobrý pokus o objektivizáciu a štandardizáciu metriky, bez človeka to ešte stále nejde.

A aká je odpoveď na otázku či postupovať podľa poučiek? Áno aj nie, nie vždy je to prínos a netreba slepo nasledovať učebnice. Prax nás naučí všetko čo budeme potrebovať. Nestretol som sa s tým, že by sa niekde reálne používala, niekto ju vymyslel, možno pre akademické diskusie, ale v praxi sa hľadí predovšetkým na skúsenosti vedúcich členov tímov.

Použitá literatúra

1. Cohn, Mike: *Agile Estimating and Planning*. Prentice Hall (2005). ISBN 0131479415.
2. Masateru Tsunoda, Tomoko Matsumura, Ken-ichi Matsumoto: *Modeling Software Project Monitoring with Stakeholders*. Springer-Verlag New York (2004). ISBN: 978-0-7695-4147-1.
3. Peter Naur: *Intuition in Software Development*. In *Proceedings of TAPSOFT, Vol.2'1985*. Springer-Verlag New York (1985). ISBN 3-540-15199-0 pp.60~79.
4. Mike Cohn: *Agile Estimating and Planning*. Prentice Hall (2006). ISBN 0-13-147941-5.

Annotation

Project monitoring with use case points

Essay provides a subjective view of the estimation techniques or project planning- earned value analysis with use case points. Technology itself is outlined only in broad outlines. Negative author's view dominates the essay, this view is supported by facts, personal experience and subjective opinion. Essay analyzes and gradually disproves, or finds alternatives all of the positive aspects of this technique. It proves that always depend on people, their feelings and experiences in software planning (monitoring), it is impossible to replace them with mathematical formulas or procedures.