

Sledovanie úloh z pohľadu manažéra a ostatných členov tímu

MAREK SOJKA

*Slovenská technická univerzita
Fakulta informatiky a informačných technológií
Ilkovičova 3, 842 16 Bratislava
xsojka@zavina.sk*

Abstrakt. Vývoj softvéru sa nezaobíde bez viacerých podporných prostriedkov, ktoré uľahčujú jeho proces. Medzi ich hlavné vlastnosti patrí aj výmena informácií medzi ľuďmi v tíme. K dobrému manažovaniu projektu je potrebné mať dostatočné množstvo údajov o jeho stave. Ich značná časť patrí sledovaniu úloh. V eseji sa zameriavam na analyzovanie podporných nástrojov, ktoré umožňujú manažérom mať o nich prehľad. Zamýšľam sa nad nimi z pohľadu viacerých rolí v projekte. Manažérske nároky na objem informácií o stave úloh sú často iné, aké by chceli produkovať zvyšné zložky tímu (analytici, programátori, tester). V práci sú rozdiskutované atribúty nástrojov z viacerých aspektov ľudí na projekte a ich dôležitosť z hľadiska monitorovania stavu.

Úvodom

V dnešnej dobe softvérového inžinierstva to projektoví manažéri nemajú ľahké. Dodanie produktu načas, v stanovenej kvalite a cene je čoraz menej častý jav. Dobrý manažér je taký, ktorý sa s týmto problémom dokáže vysporiadať. Prevencia je najlepší lekár, a preto je lepšie včas predvídať možné riziká projektu, ako neskôr hasiť veľký oheň. Na to potrebuje mať manažér prehľad, kde sa vývoj nachádza a aký bol pôvodný plán. Nemalú úlohu v tom zohráva správny výber softvérových nástrojov pre podporu riadenia projektu.

Pre manažovanie projektu existujú mnohé podporné prostriedky. Tie je výhodné použiť od projektov s jedným človekom až po tie gigantické. Ich opodstatnenosť je samozrejme priamo úmerná veľkosti projektu. Avšak už aj samotný človek máva mnohokrát problémy mať všetko na pamäti.

V nasledujúcich kapitolách budem pojednávať o nástrojoch pre sledovanie stavu úloh. Tie by mali odrážať stav projektu z hľadiska plnenia úloh v stanovenom termíne a s určenými ľudskými zdrojmi. Sú výzbrojou každého schopného manažéra a zdravého softvérového projektu. Skrátene ich budem nazývať monitorovacie nástroje. Predmetom tejto eseje nebude opis existujúcich riešení, aj keď mnohé poznatky mám

práve z používania jedného takéhoto nástroja. Budem písať o vlastnostiach a zamýšľať sa nad ich pridanou hodnotou pre projekt.

Základná charakteristika nástroja pre sledovanie úloh

Vychádzajú zo základných otázok sledovania postupu projektu [1]:

- Koľko práce sa vykonalo?
- Akú časť projektového plánu sa podarilo splniť?
- Koľko práce treba ešte splniť?
- Naznačujú dosiahnuté výsledky slabý postup a prípadnú potrebu zmeny plánu?

Tieto otázky by mal vedieť zodpovedať každý úspešný nástroj sledovania úloh. Vyhodnocuje ich zo zadávaných vstupov všetkých pracovníkov projektu. Ani nástroj však nie je vševedúci, a preto projektový manažér musí dohliadať aj na disciplínu vykazovania vykonanej činnosti jednotlivých ľudí. Len touto kombináciou dobrého nástroja a jeho relevantných vstupov môže mať manažér dobrý prehľad o stave projektu.

Zvoliť správnu granularitu úloh

Pod pojmom úloha si predstavujeme niečo, čo treba riešiť. Každá rola v tíme má určité úlohy. Tie sa líšia ako zameraním, tak aj svojou veľkosťou. Tú reprezentuje úsilie, potrebné na jej dokončenie, a čas. V nástrojoch pre manažovanie úloh, je to akosi základnou jednotkou. Môže vystupovať pod viacerými anglickými názvami.

V článku [2] sa popisuje konceptuálne prostredie pre vytváraný nástroj na sledovanie úloh v agilnom vývoji. Zavádza sa tu označenie Backlog pre zoznam úloh, ktoré sa majú vykonať. Vystupujú v ňom tri typy týchto zoznamov:

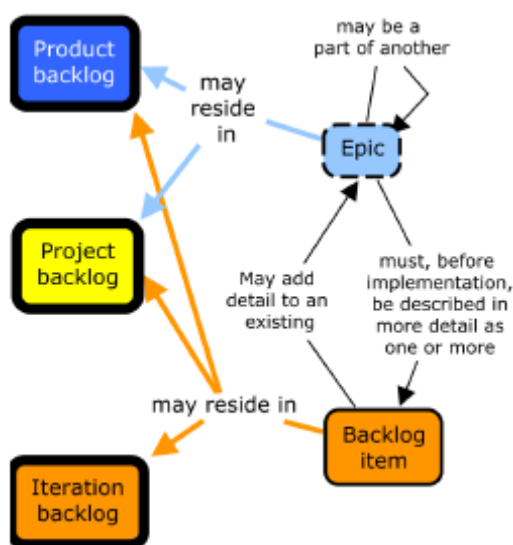
1. produktový zoznam
2. projektový zoznam
3. zoznam iterácie

Tieto sú zložené z položiek (Backlog item) a abstraktnejších položiek (epic). Tie nazvime ako dej. Zaujímavosťou pri tejto štruktúre je, že deje sa nachádzajú len v zozname produktovom a projektovom. Položky sa vyskytujú vo všetkých troch. Štruktúru možno vidieť na obrázku (Obr. 1.). Dej sa od položky líši v troch veciach:

1. môže obsahovať ďalší dej
2. je príliš veľký na to, aby bol rozumne nastavený jej čas potrebný na dokončenie

3. nenachádza sa v zozname iterácie

Príkladom položky môže byť úloha: Vytvorenie práv užívateľovi, a k nej dej by mal názov: Zabezpečenie systému.



Obr. 1. Príklad štruktúry úloh na projekte

Tento systém kategorizácie úloh je veľmi dobrý. Je však určený hlavne pre agilný vývoj. Ale podľa mňa je zbytočné mať navonok dva typy úloh. Radšej by som prijal len jednu jednotku – úlohu, ktorá môže obsahovať ďalšie podúlohy. Tie sa môžu ďalej tiež vnárať a takto vytvárajú stromovú štruktúru úloh. Kmeňové úlohy sa nachádzajú v jednom zozname pre celý projekt. Informáciu o iteráciách by som pridal ako nejakú vlastnosť úlohy.

Pri lepšom zamyslení sa dostávam k záveru, že obidva modely sú totožné. Rozdiel je len v tom, že moja koncová úloha sa v prípade potreby dokáže ďalej rozvíjať na menšie úlohy. Z pohľadu zadávateľa úloh to môže mať niekedy zmysel, keďže plánovanie úloh býva dynamické. Tými sú manažéri jednotlivých oddelení programátorov, návrhárov, analytikov, alebo testerov. Ako nevýhodu to môže brať projektový manažér. Toho môže mýliť informácia o narastajúcom počte úloh. Preto by mal manažér stanoviť aj nejaké pravidlá pre zadávanie nových úloh. Veľkou výhodou je keď sa dokážu integrovať aj do nástroja aby fungovali automaticky.

Pri spätnom pohľade na prvý model [2] vidieť pravidlo, že v zozname pre iteráciu je možné zadať len úlohu jednej úrovne. Nástroj by však mal dokázať podporiť aj iné pravidlá a obmedzenia pre úlohy. Najlepšie množinou vlastností, z ktorých sa dá vybrať na používanie len potrebné. Podľa mňa prvou požiadavkou pre nástroj na sledovanie úloh je jeho možnosť prispôbenia pre konkrétny projekt. Samozrejme

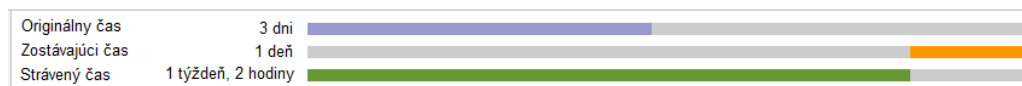
neplatí ak existuje nástroj s ideálnymi vlastnosťami pre konkrétny projekt. Takýchto relácií nástroja a projektu však nie je až tak veľa.

Času nie je nikdy dostatok

S úlohami je ruka v ruke spojený aj čas. Toho skutočne nieje v projekte dostatok. Projektový manažér ho musí mať stále pod kontrolou. Keď zoberieme úlohy na najvyššej úrovni a ohodnotíme ich časom potrebným na ich vykonanie, dostaneme krásny odhad času potrebného na dokončenie projektu, alebo jeho jednej iterácie. Píšem odhad, lebo presne to nikdy nebýva. Možno niekde v utópii. Úloha v podpornom nástroji by preto mala mať atribúty času minimálne dva - predpokladaný a čas vynaloženého úsilia na úlohu. Ten druhý sa postupne zväčšuje ako je vykonávaná daná úloha. Toto sa v nástrojoch označuje ako zaznamenávanie času (time tracking).

Podúlohy majú tiež tieto dva časové atribúty. Platí tu však že príslušný čas nadúlohy je súčtom času podúloh. Projektový manažér by sa mal vysporiadať s otázkou, či zaznamenávať vynaložený čas len pri najnižšej úrovni úloh. Podľa mňa by to malo byť pravidlom.

Malo by platiť, že vynaložený čas je menší, nanajvýš rovný tomu prvému. Tomu však nie je vždy tak (Obr. 2.). Nástroj by mal podporovať upozorňovanie príslušnej osoby, ak sa tento čas prekročí. Príslušnou osobou býva väčšinou zadávateľ úlohy. Projektový manažér by mal byť upozorňovaný v prvom rade o úlohách na najvyššej úrovni. Záleží to však dosť od veľkosti projektu. Pri menších môže byť upozorňovaný o každej úlohe.



Obr. 2. Príklad prekročenia pôvodného naplánovaného času – modrý a navýšenie zostatku času - oranžový

Podľa tendencie klesania rozdielu predpokladaného a vykonaného času možno vidieť rýchlosť vývoja. Nástroj by mal viesť tieto údaje vizuálne zobrazit'. Napríklad grafom klesania človekodní potrebných na dokončenie (Burndown), alebo formou kalendára a vykonanej práce v každom dni.

Ďalším údajom o čase pri úlohe je termín dokončenia. Zvyčajne sa zhoduje s termínom konca iterácie, ale môže byť aj iný. Napríklad ak sa jedná o nejaké vylepšenie pre konkrétneho klienta. Spolu k tomuto času má blízko aj priorita úlohy. Podľa týchto dvoch údajov sa programátor rozhoduje, ktorú úlohu riešiť skôr. Nástroj by mal podporovať širokú škálu hodnôt priority úlohy. Z vlastnej skúsenosti viem, že najhoršie je mať niekoľko úloh s rovnakou prioritou a dátumom dokončenia.

Odhady

Spraviť dobrý odhad času pre úlohy nie je jednoduché. Existujú na to viaceré metódy [1]:

- Posúdenie experta
- Odhad na základe analógie
- Simulácia
- Odhad na základe modelu

V mojej praxi som sa stretol len s prvými dvomi metódami. Nech už je odhad spravený akokoľvek dobre, môže nastať situácia, že na danú úlohu nestačí. Pre tieto prípady sa zavádza ďalší údaj pri úlohe – aktuálny odhad (Obr. 2.). Môže byť aj v tvare zostatku, čiže koľko času ostáva na dokončenie. Dobrý nástroj pre manažment úloh, by túto vlastnosť mal mať. Jej hodnotu nastavuje samotný vykonávateľ úlohy. Spočiatku je rovnaká ako originálny odhad. Správny prístup je tento údaj meniť akonáhle sa zistí skutočnosť predĺženia trvania úlohy. Toto aspoň očakávajú projektový manažéri. U vykonávateľov úloh, ktorí nemusia byť len programátori, nastáva zvyčajne tento jav: Čím nižší je pôvodný zostatok času, tým viac sa zvyšuje aktuálny. Stretol som sa už s tým. Najlepšie vidieť koľko času ešte treba, keď sa minie ten pôvodný. Podľa mňa sa dá zabrániť tomuto javu len veľmi jemnou granularitou úloh. Všetka vina však ostáva aj tak u vykonávateľov, lebo na druhej strane ani ich nebaví zaznamenávať čas pre “každé pohnutie myšou”.

Projektovému manažérovi tak ostáva len vysporiadať sa s touto skutočnosťou a dbať na to, aby odhady boli realistické a nie idealistické. Tento problém mu zatiaľ nepomôže riešiť žiadny nástroj na manažovanie.

Možné prekážky

Aj ten najlepší nástroj na monitorovanie času zlyháva, keď neexistuje, alebo je slabá disciplína zaznamenávania. Ľudská lenivosť sa prejavuje aj pri tejto činnosti. Manažér by mal zvoliť striktné pravidlá. Je jasné, že z jeho pohľadu je dobré mať zaznamenané kvantum informácií o vykonanej práci. Napríklad ktorým činnostiam sa venoval programátor pri plnení úlohy, koľko času strávil doplnkovou činnosťou ako integrovanie zdrojových kódov, či testovanie naprogramovanej časti. Alebo analytik môže vykazovať činnosti robenia skice analýzy, modelovania v CASE prostriedku alebo aj pitie kávy počas krátkej pauzy.

Ľudia v projekte sú rôzni, ale žiadneho nebaví robiť veci navyše, ktoré budú v konečnom dôsledku zbytočné. Ako som už písal, nástroj by sa mal dať nastaviť na viacero scenárov zaznamenávania. Problémom je však odhadnúť ten správny. V tomto majú výhodu sofistikovanejšie programy, v ktorých je možnosť v prípade potreby spraviť zmenu.

Hlavne užívateľsky priateľský

Obal predáva obsah. V dnešnej dobe je to tak aj u softvéru. No tu je to skôr orientované na prehľadnosť, ako na estetickú stránku. Nástroj na manažment času musí byť prehľadný, rýchly a nezaťažovať používateľa zbytočnosťami. Ved' predsa s ním pracuje denne každý člen projektu, z ktorých to väčšina nerobí z potešenia, ale preto, že to treba. Koniec koncov väčšina ma rovnaký postoj aj k chodeniu do zamestnania.

Niektoré činnosti späť so zaznamenávaním môžu byť aj automatizované. Napríklad aktuálny stav úlohy, v ktorom sa nachádza, môže meniť vývojové prostredie, v ktorom sa pracuje. Napríklad programátor si v ňom zadefinuje, že tieto zdrojové súbory patria tejto úlohe. Vtedy sa nastaví stav úlohy ako „výkonávaná“. A po úspešnej integrácii zdrojových kódov do nástroja pre verziovanie softvéru sa jej stav nastaví na „zatvorená“. Je tu možnosť aj nastavenie stavu na „otvorená“, keď sa zdrojové kódy nemenia určitý čas.

Námetov na zjednodušenia môže byť viacero. Dobré nástroje podporujú aj integráciu do viacerých vývojových prostredí a tak zjednodušujú prácu.

Pár slov na margo malého tímu

Tento semester sa venujem aj práci na tímovom projekte. Manažovanie šiestich ľudí je jednoduchšie ako pri veľkých projektoch. Výber prostriedku na sledovanie úloh nieje taký striktný. Kritéria výberu nie sú veľké. Je potreba mať nie zložitý nástroj na konfiguráciu a inštaláciu. Keď sa dokáže zabezpečiť pohodlný prístup každého člena k jednému súboru, môže samotným nástrojom byť aj Excel. Správne navrhnuté tabuľky sú zárukou úspechu. Generovanie rôznych grafov o stave projektu je tiež možné. Jedinou jeho nevýhodou je plnohodnotná práca s jedným súborom viacerým ľuďom súčasne.

Ďalšou jednoduchou alternatívou takéhoto nástroja je webová aplikácia prevádzkovaná niekde u jej tvorcov ktorá funguje na princípe „registruj a používaj“.

Záver

V práci som rozpracoval len niektoré vlastnosti podporného prostriedku na sledovanie času. Tých je oveľa viacej. Pre manažéra som naznačil dôležitosť výberu prostriedku a pre ľudí, ktorý zadávajú odpracovaný čas disciplínu tohto vykazovania. Všetko však ostáva v rukách projektového manažéra, ktorý si musí svojich ľudí manažovať tak, aby boli aj oni spokojní.

Úspešnosť manažéra nezávisí od dobrých podporných nástrojov, ale najúspešnejším manažérom pomáhajú správne podporné nástroje. Čo možno také, s ktorými sú spokojní všetci členovia projektu. A ešte úplne na záver pridám citát: *Keď*

si nezaznamenávame údaje o stave projektu, nevieme ho kontrolovať. A keď projekt nie je kontrolovaný, je mimo kontroly [3].

Použitá literatúra

1. Bieliková, M.: *Manažment v softvérovom inžinierstve*, 1999.
2. Rautiainen, K. T., Vähäniitty, J.: *Towards a conceptual framework and tool support for linking long-term product and business planning with agile software development*. <http://portal.acm.org/citation.cfm?doid=1370720.1370730> [cit. 2008-10-17]
3. Steve, M.: *Tool Support for Project Tracking*, Best practices, 1997.

Annotation

Time tracking from managers and other people on project views.

Software development proces needs many software tools, which helps it. Flowing of information between people in team is one of their main behaviours. For good project managing is necessary to have stock informations of its state. Large part of this informations belong to time tracking. My focus in this essay is oriented to helping tools for managers and their review of time in project. I contemplate to more views on its by different roles. Managers interests to capacity of state informations is different as have other people in team (analytics, developers, testers). In this paper are discussed tools from more aspects from people and their importance from view of state monitoring.