

HĽADANIE STRATENÉHO ČASU (MONITOROVANIE A MANAŽMENT ČASOVÉHO PLÁNU)

„Čas sú peniaze, peniaze hýbu svetom.“

Vladimír Martinka

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

Abstrakt. Je všeobecne známe, že pri vývoji softvéru je jednou z najväčších výziev správne odhadnúť množstvo času potrebného na jeho výrobu, testovanie a distribúciu. Samozrejme, keďže tvorba softvéru je dynamický proces, je takmer nemožné dostať dostatočne presný odhad hneď pri začiatkovej fáze plánovania. V tejto eseji by som sa chcel zamerať na monitorovanie odhadovaného a reálneho času vynaloženého na vývoj softvérového produktu. Chcel by som najmä vyzdvihnúť význam času ako primárneho zdroja vývojárskych firiem a nutnosť jeho manažmentu. Rozoberiem niektoré metódy, ktoré môžu projektoví manažéri použiť na sledovanie času stráveného nad jednotlivými časťami softvéru a vplyv týchto zistení na ďalšie plánovanie vývoja daného produktu.

Kľúčové slová: Monitorovanie Času, Čas, Manažment Projektu, Časový Plán, Plánovanie na základe dôkazov

Je manažment času nutný?

Vraví sa že čas sú peniaze a taktiež že peniaze hýbu svetom. Čas sa naozaj dá vnímať ako celosvetová univerzálna mena a ako každá cenina je nanajvýš vhodné ním nemrhať a správne ho investovať.

Keďže človek je tvor chamtivý, každý chce využiť čas čo najefektívnejšie – čo najviac zarobiť, čo najlepšie sa zabaviť a podobne. A samozrejme ako všetko čoho je obmedzené

množstvo, nie je ho nikdy dosť a veľmi ľahko sa človek dá zvieť k jeho neefektívnemu využívaniu.

Môžeme si teraz položiť otázku, prečo je manažment času dôležitý? Najprv si musíme uvedomiť dôležitý fakt, že v IT sektore predstavuje čas vedúci aspekt pri tvorbe softvéru spolu s kvalitou a cenou. Avšak pozrime sa na tieto tri vlastnosti softvéru bližšie.

Cena softvéru sa dá vyjadriť v *človekohodinách* a *človekodňoch* použitých pri vývoji softvéru a kvalita sa za normálnych okolností priamo odvíja od času stráveného nad konkrétnym softvérovým produktom. Z tohto je zrejmé, že vzhľadom na fakt, že čas úzko súvisí s ostatnými dvoma kľúčovými aspektmi softvéru, nie je bezpečné ho nechať plynúť bez povšimnutia. V opačnom prípade je možné, že sa „zobudíme“ niekoľko dní pred dátumom dodania softvérového produktu (alebo prototypu) so zistením, že softvér nespĺňa požiadavky, čo býva vo väčšine prípadov veľmi nepríjemné vysvetľovať nespokojnému zákazníkovi (najmä ak ide o produkt, ktorý musí byť nasadený do istého dátumu) a môže viesť k poškodeniu reputácie firmy prípadne až k strate zákazky.

Manažment času v praxi

V tomto bode je zrejmé, že manažment času nie je záležitosť, ktorú sa oplatí brať na ľahkú váhu. Ako však funguje pri reálnych projektoch?

Každý manažér, ktorému aspoň trochu záleží na bezproblémovom priebehu jeho projektu, musí mať k dispozícii nejaký časový plán. Klasický plán pozostáva výhradne z úloh, ktoré je potrebné spraviť, aby bol projekt hotový, a taktiež by mal mapovať ich závislosti – teda ktoré majú byť splnené najskôr.

Vytvorenie *časového* plánu softvérového projektu je však odveký problém. Tu použijem analógiu so stolárom. Stolár vie odhadnúť pomerne presne, koľko dosiek a času potrebuje na výrobu poličky, dokonca ak takú poličku už vyrábal, vie to odhadnúť s minimálnou odchýlkou. Výroba softvéru je však principiálne odlišná záležitosť, pretože každý kus softvéru je jedinečný. Pri softvéri začneme stavať, avšak nevieme, či to bude polička, dom alebo Eiffelova veža – jednoducho povedané nevieme jednoznačne určiť, v akom stave sa projekt nachádza a koľko presne času bude vyžadovať jeho dokončenie.

Aby sme teda zistili, či je nejaký vyvíjaný komponent stolička alebo mrakodrap, je nevyhnutné, aby sme mali čo najlepší prehľad o procese vývoja daného produktu. Samozrejme, pre toto je nutné mať prístup k informáciám o priebehu projektu, takže sa už jedná o nejakú formu monitorovania.

Podľa môjho názoru je najdôležitejší a často ignorovaný práve monitorovanie času počas vývoja softvérového produktu, pretože čas považujem za spoločný menovateľ ostatných aspektov softvéru (ako kvalita a cena).

Časový plán projektu

Pri vývoji softvéru (a nielen softvéru) teda často nie všetko ide podľa pôvodného plánu. Z toho vyplýva potreba v určitých intervaloch plán (samozrejme aj časový) prispôbovať momentálnemu stavu projektu. Je zrejmé, že monitorovanie a manažment času úzko súvisia práve s časovým plánom projektu.

Zoberme si najprv projekt ako taký – na začiatku každého projektu sú požiadavky alebo objednávka. Za predpokladu, že máme vytvoriť úplne nový softvér (a teda nemáme na čom stavať) sa situácia komplikuje, pretože o oblasti, ktorej sa softvér týka, máme často veľmi málo informácií a taktiež súčasťou objednávky býva nasadenie softvéru, ktoré je do istej miery závislé aj na objednávateľovi. Štandardný postup v tomto prípade je preskúmanie cieľového prostredia v spolupráci so zadávateľom a výsledné vytvorenie ponuky. Táto ponuka už obsahuje rozsah projektu, špecifikované požiadavky a požadovaný termín dodania.

Na základe týchto údajov už je možné vytvoriť plán – najprv je potrebné rozbiť projekt na podúlohy, ktorým je pridelená nejaká priorita a čas, kedy majú byť dokončené. Na základe čoho však vieme odhadnúť, kedy bude nejaký komponent dokončený? Ako bolo spomenuté, je náročné dopredu určiť rozsah práce potrebný na danom komponente.

Pri tvorbe plánu máme dva hlavné zdroje s ktorými musíme pracovať: ľudia a čas. Takže jediné čo nám ostáva, je prideliť ľuďom prácu na jednotlivých komponentoch v určitých intervaloch tak, aby bol vývoj softvéru čo najefektívnejší (korešpondujúc so špecializáciami vývojárov a zložitou podúloh).

To, koľko bude trvať vývoj daného komponentu, závisí výhradne od ľudí, ktorí na ňom pracujú. Je dôležité si uvedomiť, že z pohľadu manažéra nemáme možnosť priamo ovplyvniť čas, kedy budú jednotlivé komponenty projektu hotové (ak vývojárom priamo nepomôžeme). Čas môžeme ovplyvniť iba nepriamo a to niekoľkými spôsobmi:

1. Personálny manažment

Zdanlivo najjednoduchší spôsob ako urýchliť vývoj nejakého komponentu je jednoducho priradiť mu viac ľudí (človekohodín). Toto je však dvojsečná zbraň, pretože sa často stáva, že človek preradený na inú úlohu má problémy prispôsobiť sa a trvá mu to nejaký čas, ktorý je tým pádom stratený (mohol ho využiť na prácu na svojom pôvodnom projekte a pod.)

2. Upravenie obsahu komponentu

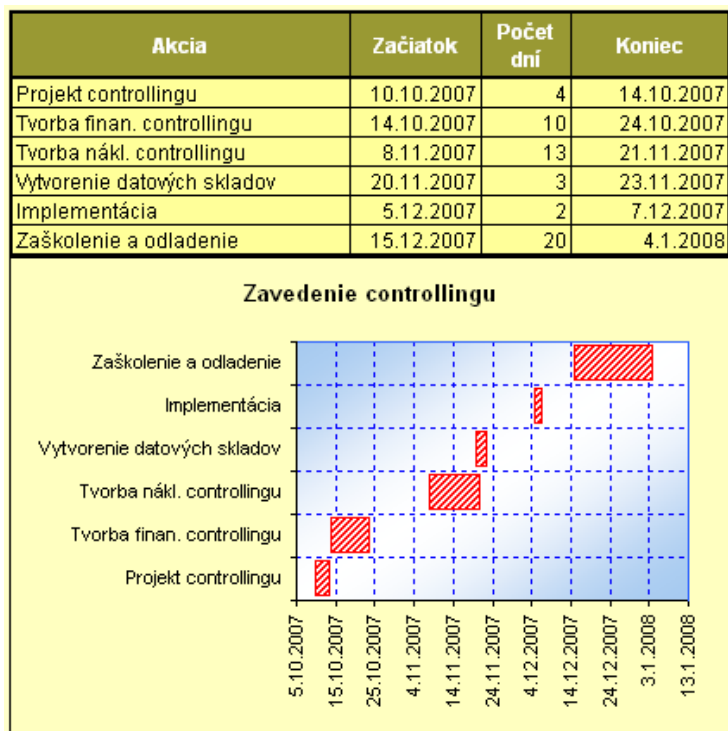
Niekedy je jednoduchšie vypustiť niektoré funkcionality projektu, alebo dokonca úplne zrušiť vývoj niektorých komponentov za účelom stiahnutia dodacej lehoty. Tieto môžu byť neskôr dorobené a nasadené už do fungujúceho softvéru. Problém je, že v tejto fáze je vývoj a najmä nasadzovanie nových funkcionalít oveľa nákladnejšie ako keby boli už v pôvodnom produkte.

3. Prehodnotenie priorít

Ak vývojár pracuje na niekoľkých úlohách zároveň, je možné ho usmerniť, na ktorú úlohu sa má prednostne sústrediť.

Každopádne, odhad času ukončenia jednotlivých komponentov by nemal byť určený manažérom, ale tímom pracujúcim na danej úlohe, pretože majú väčšinou lepší prehľad o danej oblasti ako manažér projektu [3]. Konečne, ak máme k dispozícii odhady trvania jednotlivých podúloh a ich pridelenie zamestnancom, môžeme hovoriť o časovom pláne,

s ktorým môžeme ďalej pri vývoji pracovať. Je nanajvýš vhodné si časový plán graficky reprezentovať, napríklad Ganttovým diagramom (je možné použiť napríklad aj MS Excel).



Obr. 1 Príklad Ganttovho diagramu

Monitorovanie času

Časový plán je počas vývoja softvéru nutné pravidelne upravovať. Najčastejšie sa plán prehodnocuje na pravidelných stretnutiach k projektu. Aby mohol manažment rozhodnúť o zmenách v pláne, musí mať dostatok správnych informácií o priebehu projektu. Práve tu nastupuje potreba monitorovania práce na projekte.

Pod monitorovaním času si môžeme predstaviť najmä sledovanie, do akej miery sú jednotlivé komponenty softvérového produktu dokončené. Keďže na každom komponente pracuje nejaký tím vývojárov, intuitívne základný spôsob získavania informácií o stave projektu sú hlásenia jednotlivých tímov o stave aktuálnych úloh.

Hlavnou úlohou monitorovania vývoja softvérového produktu je včas identifikovať problémy, ktoré nastávajú počas vývojového procesu. Problémy môžu mať rôzny charakter – chyby, problém so získaním licencie, choroba, výpoveď zamestnanca a pod., avšak väčšinou sa prejavujú na posunutí dátumov dokončenia jednotlivých komponentov produktu (ak vzájomne súvisia, môže sa skomplikovať aj vývoj ostatných). Preto ak zistíme, že pri súčasnom tempe nestíhame termín dodania, je zrejme, že niekde vznikol problém, ktorý buď treba riešiť, alebo ho zapracovať do časového plánu.

Ďalej je tu problém formy monitorovania času. Za normálnych okolností stačí, ak sa vyhodnotí stav každého rozpracovaného komponentu na pravidelných stretnutiach k projektu, kde sa upravuje časový plán. Tu ale narážame znova na známy problém: počas vývoja softvéru môže byť uvádzanie stavu, v ktorom sa projekt nachádza pomerne zavádzajúce. Existuje niekoľko spôsobov ako sa dá merať stav projektu v závislosti od údajov, ktoré tím manažérovi poskytuje:

1. Podľa fázy v ktorej sa projekt nachádza

Tím hlási v akej fáze sa momentálne ich projekt nachádza (napr. Návrh, Prototyp, Implementácia, Vyladovanie)

Toto je pomerne triviálna metóda a poskytuje príjemcovi pomerne málo informácií o stave a priebehu projektu.

2. Podľa výsledkov (riadky kódu, prototyp...)

Toto už dáva manažérovi hmatateľnejšiu správu o stave projektu, avšak relevantnosť je pomerne malá, keďže počet riadkov kódu nie vždy zodpovedá vynaloženému úsiliu (čo môže byť v niektorých prípadoch pre vývojára stresujúce a mať negatívny dopad na kvalitu produktu) a vôbec nehovorí o tom, kedy bude projekt dokončený.

3. Projektový denník

V ňom si každý pracovník eviduje koľko času venoval projektu a koľko času venoval ostatným činnostiam. Monitorovanie stavu projektu na báze denníkov je bohaté na informácie, avšak táto výhoda sa stáva taktiež nevýhodou, pretože príliš podrobné zápisy môže byť náročné pravidelne kontrolovať. Je však užitočné kombinovať ho s inými jednoduchšími formami monitorovania projektu, pretože umožňuje podrobnejší náhľad na stav vývoja daného komponentu, čo je užitočné v prípade, že sa v ňom vyskytne nejaká komplikácia.

4. Podľa vynaloženého času

Ďalším zaujímavým spôsobom, ktorý máme k dispozícii na monitorovanie času, je sledovanie vynaloženého času. Každý zamestnanec si eviduje čas vynaložený na danú konkrétnu úlohu. Na prvý pohľad sa to môže zdať podobné ako hlásenie počtu riadkov kódu, avšak hlavný rozdiel je v tom, že jednoducho neexistuje spôsob ako odhadnúť, koľko riadkov kódu bude daný komponent mať.

Na druhej strane máme k dispozícii odhad, kedy má byť úloha dokončená – a to práve časový plán! Tým pádom na základe plánu (ktorý si odhadoval sám vývojár ktorý podáva správu) môžeme predpokladať, že bude úloha hotová v rámci odhadovaného času.

Plánovanie na základe dôkazov

Joel Spolsky vo svojom článku „Evidence based scheduling“ [3] predstavuje zaujímavý prístup k manažmentu času v softvérových projektoch. Je to variácia monitorovania podľa vynaloženého času, avšak okrem pohľadu do minulosti, teda už dokončených projektov,

sa pokúša predpovedať budúcnosť – konkrétne pravdepodobnosti, s ktorými bude projekt dokončený v danom čase.

Je pomerne jednoduchá a funguje nasledovne: počas vývoja si každý vývojár odhaduje čas za ktorý spraví danú úlohu a zaznamenáva si skutočné časy koľko na úlohách skutočne pracuje (vrátane odstraňovania chýb). Pomery týchto výsledkov mu dáva tzv. „rýchlosti“, čiže čísla vyjadrujúce ako dlho robí na úlohách vzhľadom na jeho pôvodné odhady. Po nejakom čase je týchto „rýchlostí“ dosť na to, aby sa dal vytvoriť nejaký profil daného vývojára a dalo sa odhadnúť, koľko mu budú trvať ešte nevykonané úlohy v pláne. Ak toto spravíme pre každého vývojára a každú podúlohu, dokážeme spraviť pravdepodobnostný model dátumu dodania softvérového produktu.

Táto metóda manažmentu časového plánu je jednoznačne zaujímavá. Zdanlivo dáva pomerne presné výsledky a jej krása tkvie najmä v jej jednoduchosti použitia. Podľa môjho názoru je však v každom prípade vhodné viesť aj podrobnejší záznam o priebehu vývoja ako napríklad horeuvedený projektový denník a na pravidelné projektové stretnutia vybrať niektorý z jednoduchších modelov monitorovania času.

Záver

Mnoho manažérov a vývojárov má negatívny postoj k odhadom času a tvorbe časových plánov, keďže odhady sa v oblasti vývoja softvéru často výrazne mýlia. Argumentujú tým, že nie je možné vytvoriť presný plán, ak sú v pláne položky s vysokou mierou neurčitosti a tým pádom je plán nepoužiteľný.

Je potrebné si ale uvedomiť, čo sú zdroje vývojárskej firmy – odpoveď je zrejmá, sú to ľudia – najmä vývojári ktorí firme venujú časť svojho času. Na čo tento čas firma použije je smerodajné k jej úspechu, a preto je manažment času pre ňu jednoznačne zaujímavý.

To, či je firma úspešná závisí do veľkej miery na tom, či dokáže dodržať stanovené termíny a taktiež čím rýchlejšie a efektívnejšie vytvorí jeden produkt, tým väčší má zisk (samozrejme ak má produkt aj primeranú kvalitu).

Je paradoxné, že oblasť, ktorá sa najviac špecializuje na použitie techniky, je jedna z najcitlivejších na ľudské zdroje. Platí teda, že čím kvalitnejších zamestnancov má firma k dispozícii, tým má väčšiu hodnotu čas, ktorý venuje zákazníkom, a preto je vždy dobrou investíciou naučiť sa manažovať čas tak, aby sa ho stratilo čo najmenej.

Použitá literatúra

1. Osgood, N.: *Basics of Cost and Schedule Monitoring*. Dostupné na internete: <http://ocw.mit.edu/NR/rdonlyres/Civil-and-Environmental-Engineering/1-040Spring-2004/6301B6AC-06F2-444F-AC42-5550FD406C9D/0/114costschdmtrs4.pdf>
[cit: 2009-10-8]
2. Reh, F. J.: *Project management 101 : Managing Time and Schedule*. Dostupné na internete: <http://management.about.com/cs/projectmanagement/a/PM101c.htm>
[cit: 2009-10-8]

3. Spolsky, J.: *Evidence Based Scheduling*. Dostupné na internete:

<http://www.joelonsoftware.com/items/2007/10/26.html>

[cit: 2009-10-8]

Annotation

In Search of Lost Time

It is common knowledge, that while developing software, one of the greatest challenges is to accurately estimate the amount of time required for its creation, testing and distribution. Naturally, software development is a dynamic process, and it is almost impossible to get sufficiently accurate estimation right in the first planning phase. In this essay, I would like to concentrate on monitoring estimated and real time required for software product development. I would like to emphasize the importance of time as a primary resource for developer firms and necessity of its management. I will analyze several methods, which project managers can use for monitoring time spent on individual parts of software and influence of findings on further planning and development of given product.