Ako monitorovať softvérový projekt

BC. JAROSLAV TEŠLÁR

Slovenská technická univerzita
Fakulta informatiky a informačných technológií
Ilkovičova 3, 842 16 Bratislava
jaroslav.teslar@gmail.com

Abstrakt. Na priemerných softvérových projektoch pracujú desiatky ľudí niekoľko mesiacov. Pri takejto deľbe práce je prakticky nemožné ukončiť projekt časovo úspešne tak, aby navyše spĺňal všetky požiadavky zákaznika. Preto prirodzene vznikol mechanizmus, ktorý sa snaží čo najväčšou mierou zabezpečiť, aby sa stav projektu vyvíjal podľa plánu. To je umožnené zaznamenávaním vykonaných činností na projekte, ich dôkladnou analýzou a vyhodnotením dokončenej a zostávajúcej práce na projekte. Na základe určenia aktuálneho stavu projektu a jeho porovnaním s projektovým plánom sa tak dokážu v dobe trvania projektu nasadzovať zmeny vedúce k jeho úspešnému ukončeniu. Táto esej pojednáva práve o problematike monitorovania softvéro-vého projektu, zaoberá sa zaužívanými metódami a technikami monitorovania.

Úvod

Určiť, či niekto pracuje, je veľmi jednoduché. Ale ako môžem povedať, že je niekto pri svojej práci produktívny, či zaznamenáva progres? Pri krátkych automatických procesoch ako umývanie auta alebo maľovanie bytu je progres na práci viditeľný už po malej časovej jednotke. To však pri projektoch takých rozmerov, akým je vývoj softvéru, neplatí.

Každý softvér má svoj životný cyklus, ktorý zahŕňa špecifikáciu požiadaviek zákazníka, analýzu problematiky, návrh, implementáciu, testovanie, nasadenie do prevádzky a údržbu. Jeho forma je viditeľná až od fázy implementácie. V každej fáze môžu vznikať úskalia, ktorých riešenia je potrebné hľadať celé dni. Tímy pracujúce na softvérových projektoch pozostávajú z desiatok ľudí, ktorých práca na seba nadväzuje, závisí od seba a preto je veľmi dôležitá ich vzájomná komunikácia a synchronizácia. Aj tieto aspekty vedú k tomu, že vrámci takéhoto komplexného tímového projektu je prakticky nemožné zistiť, ako efektívne sa pracuje a či vkladané úsilie prispieva k progresu.

Monitorovanie sotvérového projektu je úloha pre manažérov. Neexisuje pre nich jednotná idea progresu a preto je na nich, aby posúdili, či idú veci k lepšiemu

Manažment v softvérovom inžinierstve, október 2007, s. 1-7.

alebo nie. V nasledujúcich častiach načrtnem spôsoby, ktorými sa manažéri môžu uberať, aby zabezpečili progres celého tímu a úspech projektu.

Progres projektu

Slovo progres poukazuje na postupný vývoj po vzostupnej línii. Znamená to, že ak má mať softvérový projekt progres, musia byť priebežne viditeľné výsledky alebo aspoň musí byť zrejmé, že projekt postupuje dopredu.

Podľa autora článku [2] existujú dva používané spôsoby, ako sa v procese softvérového projektu definuje progres.

Prvý spôsob je založený na tom, že sa naplánuje práca na projekte, s tým, že sa zvýraznia ciele, ktoré má projekt splniť. Úlohou projektového manažéra je zabezpečiť, aby každý člen tímu vedel, čo je jeho úlohou, t.j. aký presný výstup má vzniknúť z jeho úsilia na projekte. Potom každý člen tímu pravidelne sleduje svoju prácu a porovnáva ju s výsledkom, ku ktorému speje, určuje si svoj progres. Odhad termínu dokončenia projektu sa môže meniť, ale musí sa meniť spoločne pre všetkých a je potrebné ho stanovovať. Synchronizáciu odhahu dokončenia projektu zariadi manažér.

Druhý spôsob definície progresu je denné vedenie tímu. Projektový manažér je v pozícii, kedy má najviac informácii o celom projekte a jeho úlohách a cieľoch. Priebežne sa pozerá na už vykonanú prácu jednotlivých členov tímu a sleduje, ako ich práca plní stanovené ciele. Na základe svojich monitorovaní potom zvoláva stretnutia a diskusie, píše e-maily, v ktorých informuje, pripomína a motivuje každého tak, aby nikto nevykonával zbytočné aktivity. Je teda akýmsi koordinátorom projektu a tímového progresu.

Sledovanie progresu projektu

Projektoví manažéri sledujú progres tímu na projekte, podávajú priebežné správy, vytvárajú štatistiky. Pri efektívnom meraní progresu sa podľa zdroja [1] odporúča použiť metrika s aspoň nasledujúcimi vlastnosťami:

1. *objektivita:* metrika by mala byť založená na takých kritériách, ktoré sú pozorovateľné a overovateľné
2. *reálny čas:* metrika by mala poukazovať na to, v akom stave je projekt teraz a nie na to, čo sa stalo pred mesiacom
3. *viac úrovní:* viac úrovní údajov o projekte umožní manažérovi sledovať nižšie úrovne a izolovať problémové oblasti
4. *predikcia:* metrika by mala obsahovať odhad budúceho progresu

Ten istý zdroj uvádza dva spôsoby na meranie progresu:

1. **ukončené aktivity** – tento spôsob porovnáva aktuálny progres voči plánovanému pomocou stupňa ukončenia aktivít v projekte

2. **ukončené jednotky práce** – tento spôsob porovnáva aktuálny progres voči plánovanému pomocou ukončených jednotiek práce na projekte

Obe metriky majú samozrejme svoje silné a slabé stránky, na ktoré sa zameriam v nasledujúich častiach.

Meranie progresu založené na aktivitách

Progres projektu je možné merať v percentách ako podiel aktuálneho stavu aktivít a všetkých naplánovaných aktivít. Projekt má v pláne zadefinované činnosti a k nim pridelené dátumy začiatku a konca činností. Pre každé časové obdobie je vytvorená schéma progresu, ktorá znázorňuje činnosti a údaje o ich stave ukončenia. Stav ukončenia činnosti je udávaný v percentách, ktoré odhaduje projektový manažér.

Výhodou tejto metriky je, že pri dobrých odhadoch je zrejmé, ktoré činnosti sú pozadu vzhľadom na plán a potrebujú viac pozornosti. Odhady sú ale dôsledkom subjektívnych úsudkov a nemusia sa vždy zhodovať so skutočnou situáciou. Vylepšiť ich možno skrátením intervalov, v ktorom sa percentuálne schémy vytvárajú a dekompozíciou činností na menšie časti. Tak má manažér väčšiu šancu vidieť, ako sa jednotlivé malé činnosti blížia k úspešnému ukončeniu.

Grafické znázornenie progresu projektu

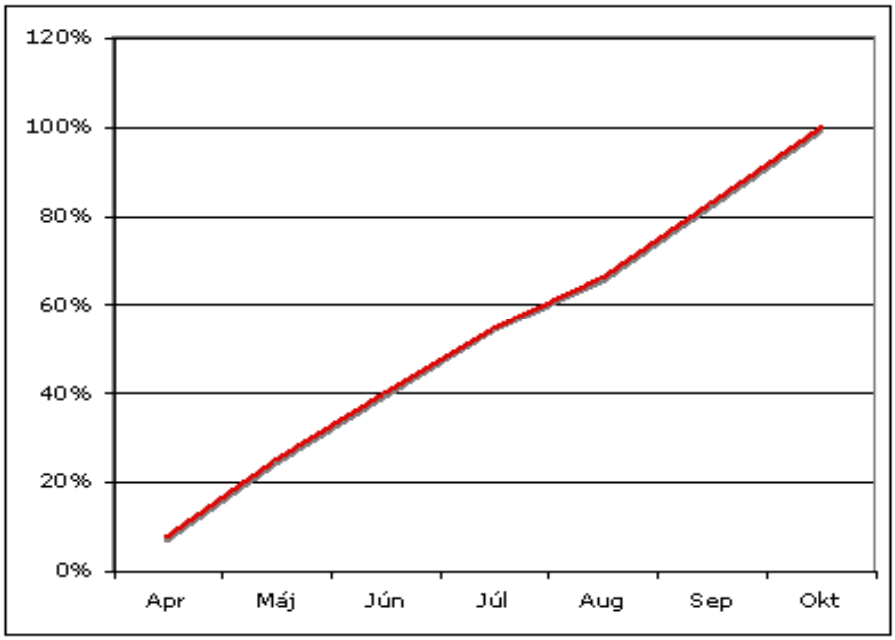
Zakládná schéma progresu podľa metriky založenej na aktivitách pozostáva z plánovej tabuľky aktivít, aktuálnej tabuľky aktivít a grafu progresu projektu. Plánová tabuľka znázorňuje v každom riadku činnosť a percentuálne ohodnotenie, ktoré vyjadruje predpokladaný stav ukokončenia činnosti v jednotlivých časových obdobiach. Aktuálna tabuľka zobrazuje odhadované dosiahnuté percentá činností podľa skutočného stavu projektu v určitom časovom období a vo všetkých obdobiach predtým.

Každá činnosť v projekte je naplánovaná na určitý čas a teda jej trvanie má určitú váhu v softvérovom projekte. Ak prenásobíme každú percentuálnu hodnotu v tabuľke plánu činností jej váhou, získame váhy každej činnosti v každom časovom období softvérového projektu. Súčet váh všetkých činností vrámci jedného časového obdobia nám tak určuje predpokladaný stav celého projektu v tomto časovom období. Ak tieto hodnoty zakreslíme do časového diagramu, získame graf predpokladaného progresu projektu. Analogicky sa dá postupovať s využitím aktuálnej tabuľky aj pre graf aktuálneho progresu. V prípade, že graf aktuálneho progresu bude v jednej sústave pod grafom plánovaného progresu, projekt je v časovom sklze a naopak.

Príklad plánovej tabuľky činností softvérového projektu je zobrazený v tabuľke č. 1. Projekt je naplánovaný na mesiace apríl-október a dekomponovaný na šesť činností: špecifikáciu požiadaviek, analýzu, návrh, implementáciu, testovanie a nasadenie. Číselné údaje v tabuľke znázorňujú ukončené percentá každej aktivity a v zátvorke sú súčasne uvedené príslušné váhy ukončenej časti činnosti vzhľadom na celý projekt. Predpokladaný progres celého projektu je znázornený na obrázku č. 1.

	Apr	Máj	Jún	Júl	Aug	Sep	Okt
Špecifikácia požiadaviek (12%)	50 (6)	100 (12)	100 (12)	100 (12)	100 (12)	100 (12)	100 (12)
Analýza (18%)	10 (2)	60 (11)	100 (18)	100 (18)	100 (18)	100 (18)	100 (18)
Návrh (18%)	0 (0)	0 (0)	55 (10)	100 (18)	100 (18)	100 (18)	100 (18)
Implementácia (24%)	0 (0)	0 (0)	0 (0)	30 (7)	60 (14)	100 (24)	100 (24)
Testovanie (18%)	0 (0)	0 (0)	0 (0)	0 (0)	20 (4)	60 (11)	100 (18)
Nasadenie (9%)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	100 (9)
Celkovo	(8)	(25)	(40)	(55)	(66)	(83)	(100)

Tab.1. Plánované ukončené percentá pre každú aktivitu a ich váhy za mesiac



Obr.1. Predpokladaný progres projektu

Meranie progresu založené na jednotkách práce

Druhá metrika progresu v projekte je viac zameraná na progres z perspektívy rozsahu projektu. Namiesto počítania percent ukončenia aktivít, sa počítajú ukončené jednotky práce na projekte.

Na začiatku sa teda určí predpokladaný celkový počet jednotiek práce, predpokladané dátumy začiatku a ukončenia projektu, a súčasne plánová línia reprezentujúca počet jednotiek ukončených v rôznych časových intervaloch. Typická plánová línia pre väčšinu softvérových projektov má tvar zakrúteného písmena S. Progres je najprv pomalý, ale potom viditeľne rastie, aby napokon zastal pri finálnych a najťažších jednotkách. Typickými jednotkami práce pre sofvérové projekty sú riadky kódu a funkčné bloky.

Riadky kódu

Riadky kódu je softvérová metrika, ktorá meria veľkosť softvéru, teda výstupu softvérového projektu, ako počet riadkov v texte zdrojového kódu programu. Poznáme dva hlavné typy riadkov kódu: fyzické a logické. Fyzické riadky kódu počítajú riadky zdrojového kódu programu vrátane komentárov a prázdnych riadkov, zatiaľčo logické riadky kódu sa snažia merať iba príkazy.

Riadky kódu poskytujú veľmi rýchly a automatický pohľad na veľkosť softvéru. Slúžia na intuitívne meranie vďaka faktu, že sú viditeľné a ich efekt je ľahko ukázateľný.

Táto softvérová metrika vznikla v dobe, keď boli používané riadkovo-orientované programovacie jazyky, ako napr. assembler a FORTRAN. Dnešné bežne používané jazyky povoľujú rôzne zápisy a formátovania kódu a jeden riadok textu nemusí automaticky zodpovedať jednému riadku kódu. Ďaľšie komplikácie pri použití tejto metriky vznikajú pre odlišné schopnosti programátorov. Skúsení programátori sú schopní napísať program s takou istou funkcionalitou s menším počtom riadkov, preto jeden program môže mať viac funkcií ako podobný rovnako dlhý program.

Riadky kódu sú tiež neefektívne pri porovnávaní programov napísaných v rôznych programovacích jazykoch. Programovacie jazyky poskytujú rôzne možnosti na vykonanie tých istých výpočtov. Extrémnym prípadom môže byť porovnanie zdrojového kódu nižšieho programovacieho jazyka (napr. assembler) s vyšším objektovým-orientovaným programovacím jazykom (napr. Java). Na vyrovnanie rozdielov medzi programovacími jazykmi už ale slúžia normalizovacie úpravy.

Najpodstatnejšou nevýhodou z hľadiska monitorovania celého softvérového projektu je to, že riadky kódu dokážu určiť produktivitu projektu poskytnutím výsledkov len fázy implementácie, ktorá znamená asi 30 – 35 % celého úsilia na projekte. Preto pre ostatné fázy je potrebné určiť iné jednotky práce.

Funkčné bloky

Funkčné bloky vyjadrujú rozsah softvérového projektu na základe funkcionality, ktorá je vnímaná používateľom softvéru. Veľkým kladom tejto metriky je implementačná nezávislosť, teda nezávislosť od programovacieho jazyka.

Funkčné bloky počítajú s nasledujúcimi funkcionalitami softvéru:

- Dátová funkcionalita
    - Interné logické súbory
    - Externé rozhrania
- Operačná funkcionalita
    - Externé vstupy
    - Externé výstupy
    - Externé dopyty

Funkčné bloky nie sú absolútne presné. Často nie sú dobre aplikovateľné na porovnanie viacerých organizácii. Ale ak sú používané vrámci projektov jednej organizácie, poskytujú veľmi dobrý spôsob merania veľkosti progresu na projekte a ich presnosť sa vtedy odhaduje do 5%. Preto sa stali najpoužívanejšou metrikou a svetovým štandardom na výpočet rozsahu softvéru a určenia produktivity práce na projekte.

Porovnanie a výber metriky

Hlavnou výhodou percentuálnej metriky na meranie progresu na softvérovom projekte je, že môže pridať percentá činnostiam, ktoré sú ešte len čiastočne vykonané. Naproti tomu, výhoda metriky založenej na jednotkách práce je, že ukazuje progres aktuálneho stavu projektu, ako sa lineárne pohybuje ku splneniu cieľov.

Nevýhodou merania progresu založeného na činnostiach je skresklenosť výsledných informácií, pretože projekt môže mať percentá znázorňujúce vykonanú prácu, hoci žiadne jednotky práce neboli úplne dokončené. Meranie jednotiek času však nemá žiadne výsledky, pokiaľ jednotky práce nie sú ukončené.

Najlepší spôsob monitorovania softvérového projektu je ideálne zvoliť na základe štyroch spomínaných kriterií: objektivita, reálny čas, viac úrovní, predikcia. Mal by byť určený celým projektovým tímom na workshope pred začatím samotných prác na projekte. Najskôr sa vhodne definujú činnosti aj jednotky práce pre samotný projekt a na základe nich a kritérii sa vyberie jedna metrika, ktorá sa použije na sledovanie progresu projektu.

Výber vhodnej metriky však stále nezaručuje, že monitorovanie projektu zabezpečí svoju funkciu a bude naozaj poukazovať na nedostatky a sklzy v softvérom projekte alebo na jeho správny chod. Každá metrika totiž stojí a padá na subjektívnych odhadoch projektového manažéra, teda vedúceho tímu, a menšou mierou aj na odhadoch jednotlivých členov. Podľa môjho názoru je preto v tomto aspekte dôležitá súdržnosť, komunikácia a synchronizácia tímu, manažérske a koordinačné schopnosti projektového manažéra ako vedúceho tímu.

Tím prechádza pri plnení jednotlivých projektov vývojom, nadobúda skúsenosti so spoluprácou v tíme, zaúča sa práci s viacerými problematikami. Vedúci tímu vníma každú osobnosť, vidí kvality a zamerania jednotlivcov, postupne zisťuje, čo od nich

môže očakávať. Prideľuje jednotlivé úlohy vhodným riešiteľom a vie lepšie určiť dobu riešenia úloh. Jeho subjektívne odhady sa čoraz viac blížia objektivite. Ak niektorí členovia tímu nestíhajú nasledovať časový harmonogram svojej úlohy, dôležité je ich neustále motivovanie a zdravá pracovná atmosféra. Ukážkovou motiváciou by mala byť precízna práca vedúceho tímu, pochvaly za správne rozhodnutia. V takomto tíme by malo monitorovanie nie len sledovať, ale aj zabezpečiť neustály progres projektu, dostatočnú produktivitu práce a efektívne priblížiť úspešné ukončenie projektu podľa projektového plánu.

Záver

Hlavnou úlohou monitorovania softvérového projektu je zaznamenávať a sledovať vývoj projektu a jeho stavu vzhľadom na plán projektu, t.j. progres projektu. Ako poukazuje táto esej, progres komplexných softvérových projektov je ťažko merateľný a existujú len približné metriky založené na subjektívnych odhadoch projektových manažérov a jednotlivých členov tímu. Dve základné metriky progresu rozdeľujú prácu na projekte na činnosti alebo na jednotky práce, ako napr. riadky kódu či funkčné bloky. Výber a použitie niektorej techniky nemusí ešte znamenať, že sledovanie progresu projektu bude úspešne informovať o aktuálnom stave projektu. Je ale jedným úspešným predpokladom, ktorý v kombinácii s dobrými skúsenosťami jednotlivých členov tímu, zdravej tímovej spolupráce a koordinačných zručností vedúceho tímu, vedie k reálnym výsledkom monitorovacieho procesu.

Použitá literatúra

1. IFPUG: *How to effectively track software progress*. [online]. 28. januára 2002. Dostupné na: <http://www.informit.com/articles/article.aspx?p=27356>
2. Scott Berkun: *Work vs. Progress* [online]. 15. augusta 2002. Dostupné na: <http://www.scottberkun.com/essays/45-work-vs-progress/>

Annotation

How to monitor a software project

Nowadays dozens of people work together many months on average software projects. Sometimes it is impossible to coordinate team work effectively and finish the project in time in the way it satisfies all the customer's requirements. That is why a mechanism has uprised, which has tried to provide a constant progress of the project. This mechanism includes recording finished activities in the project, analysing them and evaluating finished and not-finished work. Then the current state of the project can be set and it is compared to the project plan. Following that managers can make some changes to the project leading it to a successful finish. This essay deals with the topic of monitoring software projects and its standard methods.