

Monitorovanie softvérového projektu

TOMÁŠ BARTALOS

*Slovenská technická univerzita
Fakulta informatiky a informačných technológií
Ilkovičova 3, 842 16 Bratislava*

Tomas[.]Bartalos[zavináč]zoznam[.]sk

Abstrakt. Účelom článku je oboznámiť čitateľa prečo je potrebné vynakladať značné úsilie a finančné prostriedky na monitorovanie softvérového projektu. Čitateľ sa tiež môže zahĺbiť do problematiky monitorovania softvéru a taktiež sa môže dozvedieť o problémoch a úskaliach, ktoré vznikajú pri monitorovaní. Článok tiež opisuje úlohy projektového manažéra, ako člena tímu, ktorý je zodpovedný za monitorovanie a riadenie projektu. Podrobne vysvetľuje prečo sú faktory ako cena, čas a funkcionálna klúčovými pri monitorovaní, aké praktiky a postupy sa pri monitorovaní používajú. Článok používa asociácie z reálneho sveta aby ponúkol aj čitateľovi, ktorý nemá znalosti z oblasti softvérového inžinierstva prehľad o monitorovaní softvérového projektu.

Úvod

Vytváranie softvérových produktov zaznamenáva v poslednom období veľký rozmach. Na zefektívnenie a skrátenie času potrebného na vývoj nevytvárajú softvér jednotlivci, ale skupina ľudí – tím. Vývoj v tíme má oproti vývoju jednotlivca svoje špecifiká, na ktoré nemôžeme zabúdať. Pri tímovej spolupráci je nevyhnutné vyriešiť komunikáciu medzi členmi tímu, riadiť a plánovať ich prácu a monitorovať výsledky tejto práce. Monitorovanie projektu je podľa môjho názoru jednou z najdôležitejších úloh vykonávanou počas životného cyklu softvéru. Táto úloha je špecifická tým, že je vykonávaná paralelne s každou fázou životného cyklu od prvej analýzy až po odovzdanie hotového produktu zákazníkovi. Za monitorovací proces je zodpovedný jeden člen tímu, ktorý ho vykonáva a ktorý podáva pravidelné hlásenia. Tento člen tímu je väčšinou projektový manažér. Výsledky monitoringu prinášajú dôležité informácie o tom, v akej fáze vývoja sa projekt nachádza, aké sú problémy pri vývoji a v neposlednom rade informácie o vynaložených finančných prostriedkoch na daný projekt. Prácu projektového manažéra môžem prirovnať k riadeniu dopravného prostriedku v neznámom prostredí. Šofér neustále kontroluje či neodbočil z trasy, či má optimálnu spotrebu, a či dôjde do cieľa na čas. Čo i len malé zaváhania, a nesprávne odbočenie z trasy môže mať za následok časový sklz, zvýšenie predpokladanej spotreby, a v krajných prípadoch aj riziko nedôjdenia do cieľa. Tie isté faktory, ktoré

determinujú úspech vodiča sú najčastejšie sledované aj v softvérových produktoch. Dobrý projektový manažér dokáže dôjsť do cieľa bezpečne a načas aj s menej vybaveným vozidlom.

Prečo je potrebné monitorovať projekt

Softvérový tím je skupina odborníkov, ktorí sa spoločnými silami snažia vytvoriť čo najkvalitnejší produkt. Je to v skutočnosti tak? Ľudia sú rôzni, a zo skúsenosti viem, že v každej skupine ľudí sa nachádzajú aj takí, ktorí radi profitujú z práce ostatných. Sú to členovia tímu, ktorí svojou lenivosťou znižujú produktivitu, a kazia výkonnosť tímu. Je to jeden dôvod, prečo je potrebný kvalitný tímový monitoring. Na druhej strane tím, ktorý pozostáva zo samých snaživých členov tiež nie je bez správneho monitoringu perspektívnym tímom. V tomto prípade sa môže ľahko stať, že budú dvaja členovia robiť nevedomky na tej istej úlohe, čo je plytvanie ľudskými zdrojmi. Procesy, ktoré nastávajú v softvérovom tíme môžem prirovnať k procesom prebiehajúcich pri fungovaní futbalového mužstva. Prácu programátora, alebo dokumentaristu môžem prirovnať k práci hráča na futbalovom ihrisku. Jeho činnosť je skutočne viditeľná, a relatívne aj merateľná. Môže nastať situácia, že sa hráč spoľahne na svojich spoluhráčov, a prestane sa snažiť. Môže však nastať aj taká situácia, že hráč sa celý zápas snaží, dáva zo seba maximum (veľké množstvo napísaného zdrojového kódu), a napriek tomu zápas prehrá. Môže to byť z dôvodu slabšej tímovej spolupráce, alebo z dôvodu neskúsenosti hráča. Z tohto príkladu je viditeľné, že aj v prípade, keď sa hráči snažia nie je možné bez správneho monitorovania a riadenia prísť k úspechom. Na to aby bol futbalový tím úspešný potrebuje niekoho, kto dokáže monitorovať, viesť a stmeliť prácu jednotlivých hráčov na ihrisku – futbalového trénera. Činnosť futbalového trénera môžem prirovnať k činnosti projektového manažéra. Tréner sa snaží svojich hráčov viesť k tomu, aby boli čo najlepšie pripravený na vzniknuté problémy na ihrisku, a podávali čo možno najvyšší výkon (riadenie). Psychologickými nástrojmi sa hráčov snaží presvedčiť, aby hrali čo najlepšie nie preto, že hrať musia a dostanú za to finančnú odmenu, ale preto, že ich to baví, a hrať chcú (motivácia). Snaží sa s hráčmi neustále komunikovať, diskutovať o ich neúspechoch a hľadať spoločné riešenia (komunikácia). V prípade, že hráč urobí závažnú chybu, tréner sa ho nesnaží zosmiešniť, ale analyzovať vzniknutú situáciu, a ukázať ako chybe predísť a riešiť ju (podpora). Podľa môjho názoru by tréner tieto spomínané činnosti bez správneho monitoringu nedokázal vykonávať. Tréner musí monitorovať a vyhodnocovať výkony hráčov, a až podľa týchto informácií môže plánovať do budúcnosti. Rovnaká je potreba monitorovania aj v softvérovom projekte. Projektový manažér musí neustále zisťovať výkonnosť jednotlivých členov tímu, celkový stav projektu, a podľa týchto informácií podporovať a riadiť členov tak ako som to spomínal u futbalového trénera.

Aby sme mohli efektívne monitorovať projekt, musíme si najprv odpovedať na otázku, prečo chceme softvér monitorovať. Softvér chceme monitorovať preto, že chceme [5]:

- Riadiť vývoj projektu tak, aby sa čo najviac približoval projektovému plánu
- zistiť možné problémy ešte pred tým ako nastanú
- nasadiť potrebné opatrenia

Monitorovanie projektu neslúži len na kontrolu práce ako takej, ale taktiež na overenie kvality práce jednotlivých členov tímu. V prípade, že členovia tímu produkujú výstupy, ktoré sú nepresné, a chybové, je potrebné mať správny monitoring, ktorý tieto chyby odhalí, a upozorní na ne jednotlivých členov. V prípade, že chybu neodhalíme včas náklady na jej nápravu exponenciálne narastajú. Podľa tabuľky 1 vidíme, že je nevyhnutné odhaliť chybu v softvérovom projekte v čo najskoršej etape.

Tab. 1. náklady na opravu zistenej chyby [1]

<i>Etapa v ktorej sa chyba odhalí</i>	<i>Náklady na opravu [človeko-hodiny]</i>
<i>Špecifikácia</i>	2
<i>Návrh</i>	5
<i>Implementácia</i>	15
<i>Akceptačné testovanie</i>	50
<i>Údržba</i>	150

Podľa Johnsona [4] v roku 1995 vynaložili americké firmy približne 59 miliárd dolárov na predražené projekty a ďalších 81 miliárd stratili na neúspešných projektoch. Ďalší výskum, ktorého sa zúčastnilo 72 vyvíjaných informačných systémov v 23 amerických firmách ukázal, že priemerne 36% projektov prekročilo odhadnutý počet pracovných hodín a 22% z týchto projektov bolo odovzdaných neskoro – po plánovanom termíne [3]. Myslím, že z týchto údajov jasne vyplýva potreba kvalitnejšieho monitorovania a riadenia softvérových produktov.

Faktory ktoré v projekte monitorujeme

V úvode eseje som načrtol myšlienku asociácie práce projektového manažéra a šoféra riadiaceho dopravný prostriedok. Vodič neustále kontroloval či nedobočil z trasy, má dostatok paliva na cestu, má optimálnu spotrebu, a či nemá časový sklz. V tomto príklade si môžeme všimnúť tieto činitele, ktoré ovplyvňujú úspech vodiča:

- čas
- finančné prostriedky (spotreba, dostatok paliva)
- funkcionálna (bezpečné dôjdenie do cieľa)

Všetky tieto 3 faktory sa navzájom ovplyvňujú a závisia od seba. Ak napríklad vodič nesprávne odbočí z trasy, musí plytvať palivom na návrat k pôvodnej trase. Vodič má v tomto prípade časový sklz, ktorý chce vyrovnať rýchlejšou jazdou. Takouto

jazdou sa však zvyšuje spotreba dopravného prostriedku, a taktiež je väčšia pravdepodobnosť, že vodič spraví ďalšiu chybu. Cyklickým opakovaním takýchto chýb môže dôjsť k tomu, že sa vodič nedostane bezpečne do cieľa. Môj názor je, že ak by vodič hneď zo začiatku venoval väčšiu pozornosť monitorovaniu spomenutých faktorov, alebo by investoval do podporných nástrojov na monitorovanie (GPS) nemusel by vynakladať zbytočné finančné prostriedky, a časové straty. Rovnaká situácia je aj v prípade riadenia softvérových projektov. Projektový manažér musí neustále vyhodnocovať a monitorovať stav softvérového projektu. Základné a dlhodobé požiadavky, ktoré projektový manažér sleduje sú tie isté ako pri dopravnom prostriedku – čas, cena, funkcionálnosť. Ak sa pri vývoji nájde v projekte chyba, je podľa môjho názoru lepšie zachovať chladnú hlavu, snažiť sa chybu svedomito napraviť a nie chybu povrchne odstrániť a bezhlavo sa ponáhľať, aby sme produkt odovzdali na čas. Pretože ako sme videli v časti „Prečo je potrebné monitorovať projekt“ nápravy na chybu odhalenú pri prevádzke sú neúmerne vyššie, ako náklady na chybu odhalenú v skorších etapách životného cyklu projektu.

Úloha projektového manažéra je sledovať a tlmočiť požiadavky zákazníka na produkt. Samozrejme typickou požiadavkou zákazníka je mať najlacnejší produkt za čo najkratší čas. Táto požiadavka je podľa môjho názoru nerealizovateľná, a v prípade, že sa niekomu takýto produkt podarí vytvoriť, bude určite trpieť jeho funkcionálnosť, a bude obsahovať nespočetne veľa chýb. Preto sa väčšinou na začiatku projektu určia parametre, ktoré sú fixné, a tie, ktoré sú variabilné a môžu sa teda počas vývoja projektu meniť. Vo väčšine prípadov je fixným parametrom funkcionálnosť, a relatívne variabilnými sú cena a čas. Zákazník má na začiatku predstavu čo chce, aby program robil, no je pomerne náročné vytvoriť odhad aký čas, a koľko finančných prostriedkov bude potrebné vynaložiť na vývoj takéhoto programu.

Dnes už však existujú aj nové metódy tvorby softvéru, ktoré do značnej miery urýchľujú vývojový proces. Príklad takejto metodiky je agilné programovanie. Táto metodika programovania dopredu počíta s tým, že sa funkcionálnosť softvérového produktu bude počas vývoja meniť. Jedna z čŕt takéhoto programovania je programovanie podľa princípu skratky YAGNI (nebudeš to potrebovať) – implementuje sa len to, čo je nevyhnutné, teda aj keď viem, že v budúcnosti budem potrebovať istú funkcionálnosť programu, dnes ju ešte neimplementujem, lebo viem, že sa v budúcnosti môže zmeniť. Výhoda tohto prístupu programovania je, že mám za relatívne krátky čas hotový produkt, so základnou funkcionálnosťou. Fixnými požiadavkami sú v tomto prípade čas a cena, a variabilnou je funkcionálnosť, ktorá sa môže počas vývoja meniť.

Spomenuté faktory sa budem snažiť monitorovať aj počas vývoja školského tímového projektu. Základné rozdiely vo faktoroch medzi školským tímovým projektom a firemnými projektmi:

- Funkcionálnosť – správna funkcionálnosť školského produktu je nevyhnutnosťou, podľa môjho názoru sa však v produkte neobjavia skryté chyby (bugy), ktoré sa prejavujú v produktoch v ostrej prevádzke, kde ich používa tisíce ľudí.

- Hodnotenie (Peniaze) – odmena v školskom projekte je získanie dobrej známky a absolvovanie predmetu. Finančná motivácia vo firemných projektoch je podľa mňa silnejšou zbraňou, ktorá dokáže lepšie motivovať členov tímu. Pozitívne hodnotím snahy na fakulte FIIT STU odmeňovať najlepšie projekty aj inými nehmotnými výhodami.
- Čas - neodovzdanie školského projektu na čas môže mať za následok neabsolvovanie predmetu, čo je na druhej strane silná motivácia pre študentov.
- Skúsenosti - členovia školského tímu majú väčšinou menšie skúsenosti s riadením a vývojom softvérových projektov, ako ich kolegovia v „ostrej“ prevádzke.

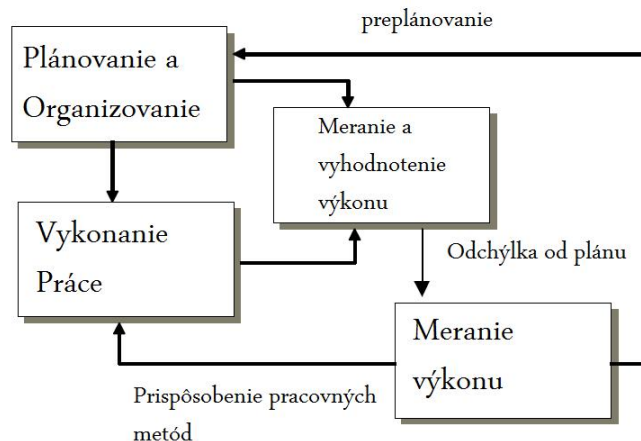
Zo spomenutých skutočností vyplýva, že fixný faktor vo vývoji školských projektov je čas, a variabilnými sú hodnotenie a funkcionálnosť.

Ako monitorovať projekt

Toto je veľmi rozsiahla téma, ktorou sa zaoberá množstvo projektových manažérov. Samozrejme existuje množstvo prístupov, ktoré dokážu monitorovať rôzne aspekty softvérového projektu, každý má svoje výhody a nevýhody. Keď som sa sám zamýšľal nad otázkou ako efektívne monitorovať projekt, napadlo ma niekoľko ďalších otázok, na ktoré som sa snažil odpovedať. Z akých procesov pozostáva monitoring softvérového projektu? V prvom rade monitorovanie nie je len kontrolovanie stavu softvérového projektu, ale taktiež snaha o analýzu, detekciu a možnú nápravu vzniknutej chyby, alebo nesprávnych postupov. Monitorovací proces sa skladá z troch krokov[2]:

- Zistenie stavu, v ktorom sa projekt nachádza
- Porovnanie stavu s plánom
- V prípade, že sa stav s plánom nezhoduje, zistenie príčin, a snaha o nápravu

Jedna z úloh projektového manažéra je teda porovnávanie aktuálneho stavu projektu s plánom, a následne uskutočňovanie nápravných krokov. Na obrázku 1 je zobrazená schéma takéhoto typu monitorovania projektu.



Obr. 1. Schéma monitorovania projektu [5]

Ako správne monitorovať softvérový projekt? Jeden z procesov pri monitorovaní je monitorovanie činností a postupov členov tímu, ktoré posúvajú stav projektu dopredu. Členovia tímu môžu mať rozličné role, pričom výsledok práce každej z rolí môže byť viac, či menej viditeľná. Napríklad práca člena tímu v roli vedúceho nemusí byť vo výslednom produkte fyzicky viditeľná vôbec (nemusel napísať ani jeden riadok zdrojového kódu, alebo dokumentácie), ale bez jeho organizačných a riadiacich schopností by podľa môjho názoru softvérový produkt nemohol byť odovzdaný v danom čase a kvalite, a pravdepodobne by bol odsúdený na neúspech. Práca programátora, alebo dokumentaristu je už prácou viditeľnou a relatívne merateľnou. Pri diskutovaní rôznorodosti činností jednotlivých členov tímu, je vidieť, že každý okruh činností je potrebné monitorovať iným spôsobom. Prácu programátora, alebo dokumentaristu môžeme relatívne merať priebežne na základe napísaných zdrojových kódov, alebo dokumentácie. Pri pracovných činnostiach ako analytik, manažér kvality, manažér plánovania, projektový manažér prichádzam k problému nemerateľnosti. Ich práca je priebežne ťažko merateľná, a chyby sa prejavujú až v neskorších etapách vývoja, keď sú náklady na nápravu vysoké. Preto podľa mňa neexistuje univerzálny a komplexný spôsob ako monitorovať softvérový projekt.

Každý projekt sa dá monitorovať z rôznych pohľadov (aspektov). V jednom z prístupov môžeme sledovať dve základné veličiny, ktoré určujú stav projektu:

- Práca, ktorú členovia tímu vykonávajú
- Pokrok, ktorý sa touto prácou dosahuje

Určenie miery vykonanej práce sa zdá byť na prvý pohľad jednoduché. No nie vždy je tomu tak. Ak kopeme jamu, je miera vykonanej práce jednoducho merateľná. Práca a postup je priamo úmerná objemu vykopanej jamy, prípadne môžeme zohľadniť tvrdosť pôdy. Avšak pri niektorých činnostiach nie je vykonaná práca dlhú dobu

viditeľná. Napríklad pri písaní tohto odborného článku pozostáva veľké množstvo práce z vyhľadania vhodných zdrojov a štúdium danej problematiky. Táto časť práce je však skrytá, a jej výsledok bude viditeľný až po napísaní článku.

Druhá veličina, pomocou ktorej dokážeme monitorovať projekt je pokrok práce. Pokrok je množstvo práce, ktorou daná práca prispela k úspešnému dokončeniu projektu. Zistenie správneho pokroku nie je jednoduché, a v mnohých prípadoch je veľmi ťažko merateľné. Hlavne vo veľkých spoločnostiach býva problém vyjadriť, o akú časť nás istá práca posunula k dokončeniu úlohy. Ak učiníme nejakú prácu, ešte nie je isté, že sme s ňou vykonali pokrok v projekte. Jedna úloha môže byť napríklad odstrániť vodu z bazény a druhá naplniť bazén vodou. Úloha vyprázdenia bazény by v prvom prípade znamenala posunutie k ukončeniu úlohy, v druhom prípade by však tá istá práca znamenala vzdialenie sa od ukončenia úlohy, pretože by bolo potrebné spraviť dodatočnú prácu, aby sme sa dostali aspoň do stavu, v ktorom sme boli na začiatku práce. Meranie postupu je preto podľa môjho názoru veľmi efektívny spôsob monitorovania softvérového projektu. Pretože vyjadruje nielen skutočnosť, že člen tímu niečo robí, ale že to robí aj dobre. Jeden člen tímu môže napísať množstvo strán zdrojového kódu, zatiaľ čo iný člen tímu s lepším postupom dokáže na jednu stranu napísať viac ako ten prvý.

Ďalší problém, ktorý sa vynára pri snahe monitoringu projektu vychádza priamo z povahy softvéru. Softvér je [5]:

- Nehmatateľný – softvér je na rozdiel od hardvéru abstraktný. Nevieme teda fyzicky zobrazit' jeho stav.
- Zložitý – veľké softvérové produkty vytvára množstvo vývojárov, a teda neexistuje člen tímu, ktorý by rozumel celému produktu, čo spôsobuje problémy pre manažérov.
- Nestály na požiadavky – vývoj veľkého softvérového produktu trvá dlhší čas, preto požiadavky, ktoré mal zákazník na začiatku vývoja sa môžu dynamicky meniť.

Všetky spomínané vlastnosti softvéru spôsobujú projektovým manažérom problémy pri monitorovaní.

Typickým syndrómom nesprávneho monitoringu je syndróm „90% hotovo”. Tento syndróm vzniká z faktu, že ľudia riešia najprv úlohy, ktoré sú im známe, a vedia ich vyriešiť, ale úlohy, ktoré sú im neznáme riešia ako posledné. Pri vyriešení známych úloh majú pocit, že softvér je už takmer hotový, a treba v ňom dotiahnuť už len pár detailov. Tento názor je však väčšinou klamlivý, a práve doladenie týchto detailov trvá väčšinou najdlhší čas.

Záver

V eseji som objasnil, prečo je monitorovanie nevyhnutným procesom pri každej fáze vývoja životného cyklu softvéru. Zameral som sa na hlavné faktory, ktoré sú sledované pri monitorovaní projektov. Taktiež som sa snažil vyjadriť svoj názor, že je veľmi

ťažké komplexne monitorovať celý projekt, a že softvérový projekt je tak špecifický proces, že neexistuje univerzálny spôsob, ktorým ho dokážeme monitorovať. Opísal som tiež rôzne spôsoby, ktorými dokážeme monitorovať jednotlivé aspekty projektu. Zameral som sa na úskalía a problémy, ktoré vznikajú pri monitoringu. Po prečítaní článku by malo byť zrejmé, že informácie nadobudnuté pri monitorovaní sú dôležitým podkladom pre projektového manažéra, ktorý sa na základe týchto informácií rozhoduje o ďalšom plánovaní a riadení projektu. V eseji som sa snažil jednotlivé manažérske úlohy a vlastnosti prirovnať ku skutočným procesom, ktoré sú pre čitateľa známe z každodenného sveta. Tým som chcel dosiahnuť to, aby aj čitateľ, ktorý nemá znalosti o problematike softvérového inžinierstva mal po prečítaní článku základnú myšlienku o tom prečo je monitorovanie nevyhnutné a ako prebieha.

Použitá literatúra

1. **Bieliková, M.** *Softvérové inžinierstvo - Princípy a manažment*. STU, Vazovova 5, Bratislava, 2000.
2. **Chorvat, J.** *Project management nation: Tools, Techniques and Goals for the new practicing IT project manager*, 1998
3. **Genuchten, M.** Why is Software Late? An Empirical Study of Reasons for Delay in Software Development, *IEEE Transactions on Software Engineering*. June 1991.
4. **Johnson, J.** The Dollar Drain of IT Project Failures, *Application Development Trends*. 1995.
5. **Jurison, J.** Software Project Management: *The Manager's View*. *Communications of the Association for Information Systems*. September 1999.

Annotation

Monitoring of software project

The purpose of this essay is to inform reader why is it so necessary to make effort and give finances on monitoring of software project. Reader can understand the problematic of software monitoring and also gain knowledge about problems and troubles which appears during monitoring. Essay also discusses the tasks of software manager, who is a member of the team and is in charge of software monitoring and control. It describes in detail why factors as money, time and functionality are the key factors in software monitoring and which techniques and methods are used in monitoring. Real world associations are used in the article to allow better comprehension also to people who have just a little knowledge about software engineering.