

Riziká softvérových projektov a možnosti zníženia ich dopadu

TOMÁŠ MOLNÁR

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

Abstrakt. Ako sa ukázalo, stupeň neúspechu softvérových projektov je v súčasnosti enormne vysoký a dopady na globálnu ekonomiku sú o to väčšie, čím viac spoločností sa púšťa do vývoja softvéru. Väčšina softvérových projektov sa vyvíja v nepredvídateľnom prostredí plnom nástrah, ktoré nepriaznivo vplývajú na výsledok projektu. Prieskumy zamerané na zhodnotenie projektov po ich ukončení však odhalili, že mnoho z týchto problémov bolo v skutočnosti možné predvídať. Táto práca sa preto zameriava na najzávažnejšie riziká, ktoré môžu nastať počas vývoja softvéru, a pravidlá používané na zníženie výskytu týchto rizík či minimalizáciu ich dopadu. Vychádzajúc zo skúseností projektových manažérov sa budem snažiť načrtnúť ich pohľad na svet rizík softvérových projektov.

Úvod

Projekt sa zvyčajne považuje za úspešný, ak spĺňa požiadavky naň kladené (správnosť, spoľahlivosť, udržiavateľnosť, prenositeľnosť, efektívnosť, integrita a použiteľnosť), je ukončený načas a v rámci rozpočtu [6]. Podľa [5] je iba jedna šestina všetkých projektov ukončená načas a v rámci rozpočtu, jedna tretina projektov bola zrušená a viac ako polovica nebola dodaná s plnou funkcionalitou. Podľa [4] je vysoký neúspech projektov spôsobený tým, že manažéri sa nedostatočne zaujímajú o posúdenie a manažovanie rizík softvérových projektov. Podľa štúdie [3] až 35% zo zrušených projektov bolo zrušených až vo fáze implementácie. Z toho vyplýva, že manažéri robia len veľmi málo pre odhaľovanie a ukončovanie projektov náchylných na neúspech. Tieto projekty potom „požierajú“ ďalšie a ďalšie zdroje a aj keď sú neskôr zrušené, investície vynaložené do ich udržania môžu viesť k úplnému vyčerpaniu finančných a ľudských zdrojov spoločnosti.

Manažment v softvérovom inžinierstve, máj 2005, s. 1-7.

Riziká a manažment rizík

Manažment rizík softvérových projektov sa snaží formalizovať vzťah medzi rizikami a výsledkom projektu. Zahŕňa techniky a návody na identifikáciu, analýzu a kontrolu softvérových rizík, zameriava sa na protopatrenia, ktoré obmedzia riziká v projekte, alebo znížia ich dopad. Manažment rizík by preto mal byť jednou zo základných súčastí procesu manažmentu každého softvérového projektu.

Analýzou rizík je možné zredukovať pravdepodobnosť výskytu rizík a tiež zredukovať priemerný počet vývojových cyklov počas životného cyklu projektu. Manažment rizík nie je vždy stopercentne úspešný, avšak jeho aplikovanie má zmysel v identifikovaní a odozve na potenciálne problémy v dostatočnom predstihu, vyhnúť sa krízovým situáciám.

„Aj keď nevieme predvídať budúcnosť s istotou, môžeme aplikovať štruktúrované pravidlá manažmentu rizík a nazrieť až za horizont udalostí, preskočiť nastražené pasce a spraviť opatrenia na minimalizáciu pravdepodobnosti vzniku alebo dopadu potenciálnych problémov.“ [2]

Riziká softvérových projektov

Nasledovné riziká boli identifikované viacerými výskumami a pravidelne sa objavujú v softvérových projektoch. Sú považované za najproblémovejšie a teda je nutné ich kontrolovať.

Jedným z rizík je *nejasný alebo zle pochopený účel či rozsah projektu*. Každý, kto je zainteresovaný na softvérovom projekte, má trochu odlišný pohľad na účel projektu. Napríklad používatelia potrebujú robustný, používateľsky príjemný systém s mnohými funkciami. Na druhej strane vývojári dúfajú v riešenie zaujímavých technických problémov a prácu so zaujímavými technológiami. Tieto rozličné očakávania vytvárajú elementárne konflikty, vedúce k nejasnému, či zle pochopenému účelu a rozsahu projektu.

Ďalším problémom sú *podhodnotené náklady a časové plány*. Všeobecne je veľmi náročné odhadnúť časové plány s akceptovateľnou presnosťou a dôslednosťou, preto treba tomuto riziku venovať vysokú pozornosť. Veľmi často sa spoločnosti púšťajú do projektov s podhodnotením ich veľkosti a komplexnosti, čo často krát vedie k úplnému kolapsu projektu.

Podľa uvedených výskumov je *nedostatočná angažovanosť vyššieho manažmentu v projekte* jedným z najkritickejších rizík projektu. Projekt je úspešný len v prípade, ak je najvyššou prioritou vyššieho manažmentu, podobne ako manažment projektu ovplyvňuje proces a postup projektu.

Podobne ako v predošlom prípade, aj riziko *nedostatočnej angažovanosti používateľov* je zaradené medzi veľmi kritické riziká. Ak používatelia nie sú dostatočne angažovaní v projekte vzniká riziko, že vývojári prevezmú definíciu detailnej funkčnosti a požiadaviek systému na seba, čo zväčša vedie k nespĺneniu definovaných požiadaviek a nespokojnosti zákazníka.

Za riziko sa tiež považujú *neadekvátne znalosti či schopnosti členov tímu*, pretože vývojový tím nemá vždy dostatočné znalosti technológie či doménovej oblasti, alebo jednoducho členovia tímu nemajú dostatočné skúsenosti na zvládnutie projektu podobného rozsahu. Nadhodnotené schopností ľudí môžu veľmi nepríjemne ovplyvniť výsledok projektu.

Ďalším rizikom je *nedostatočne efektívna metodológia manažmentu*. Často sa totiž stáva, že softvérové systémy nasadené do prevádzky sú používateľmi vnímané negatívne. Je to dôsledkom zlej metodológie projektu. Pozor však na používanie jednej metodológie dlhodobo – môže vytvoriť prostredie, kde daná metodológia je maximálne vhodná pre jeden typ projektu, ale vôbec nevyhovuje ostatným projektom.

Medzi riziká softvérových projektov tiež patria *zle pochopené požiadavky*. Zber a zaznamenávanie všetkých detailov od budúcich používateľov môže byť ťažké a časovo náročné. Dôsledkom toho je, že tím nie je dostatočne informovaný o požiadavkách potrebných na úspešné ukončenie projektu. To môže viesť k vývoju systému, ktorý je nepoužiteľný, pretože nebola vykonaná dostatočná analýza systému potrebná na vytvorenie kompletnej a presnej množiny požiadaviek.

Veľmi zaujímavým rizikom je takzvané *pozlátenie*, čo znamená, že vývojári a analytici často vymýšľajú dodatočné funkčnosti alebo zmeny, ktoré neboli definované v požiadavkách, no ktoré by podľa ich názoru zlepšili systém, alebo ho urobili viac atraktívnym pre používateľa. Tieto odchýlky môžu spôsobiť následnú nespokojnosť používateľov, predĺžovanie času vývoja a tým dodatočné náklady.

K najkritickejším rizikám patrí aj *neustála zmena požiadaviek*, pretože zmena požiadaviek používateľa ide ruka v ruku so zmenou požiadaviek systému. Na jednej strane zafixovaním určitej časti funkčnosti systému a dátumu dodania je možné začať vývoj tejto časti, avšak na strane druhej, požiadavky by možno nemali byť zafixované, pretože v dnešnom rýchlym svete obchodu zafixovanie požiadaviek nedokáže sledovať rýchle zmeny v obchodných praktikách. S fixným dizajnom má vývojár len malú flexibilitu pri zmenách špecifikácie. V každom prípade, neustále a nekontrolované zmeny v požiadavkách nevyhnutne vedú k posunom v časovom pláne projektu.

Riziko *vývoja nesprávnych funkcií systému* súvisí s rizikom zle pochopených požiadaviek. Je to veľmi nebezpečné riziko, spojené s dosahovaním správnej funkčnosti systému ako z pohľadu technického, tak z pohľadu používateľa. Pravdepodobnosť tohto rizika sa znižuje s pribúdajúcou praxou a vzdelaním manažéra.

Podobne ako riziko neadekvátnych znalostí či schopností členov tímu, aj *zavedenie novej technológie* môže veľmi nepríjemne ovplyvniť výsledok projektu. Riziko môže nastať v prípade ak sa použije nová technológia, ktorá nebola úspešná v iných spoločnostiach, alebo ak sa technológia zmení počas trvania projektu.

Takisto riziko *nezvládnutia manažmentu očakávaní koncových používateľov* je nezanedbateľné. Očakávania používateľov totiž rozhodujú o úspechu či neúspechu projektu. Ak dodaný systém nie je v súlade s očakávaniami, problém je na svete. Preto by mali byť dokumentované očakávania ustavične vyhodnocované voči funkčnosti systému.

Z prieskumu [1] vyplýva, že skúsený softvérový manažéri prisudzujú jednotlivým rizikám dôležitosť podľa grafu na obrázku 1:



Obr. 1. Dôležitosť jednotlivých rizík.

Legenda k obrázku 1:

1. Nejasný alebo zle pochopený účel či rozsah projektu
2. Podhodnotené náklady a časové plány
3. Nedostatočná angažovanosť vyššieho manažmentu v projekte
4. Nedostatočná angažovanosť používateľov
5. Neadekvátne znalosti či schopnosti
6. Nedostatočne efektívna metodológia manažmentu
7. Nesprávne pochopené požiadavky
8. Pozlátenie
9. Neustála zmena požiadaviek
10. Vývoj nesprávnych funkcií softvéru

Pravidlá zvládnutia softvérového projektu

Manažment softvérových projektov sa stal kritickým prvkom v mnohých spoločnostiach. S každým novým rizikom pri vývoji systému sa vynára potreba formálnych pravidiel na jeho zvládnutie. V nasledujúcom odseku si rozoberieme niektoré z nich.

Vytvorenie a dodržiavanie plánu vývoja je jedným zo základných pravidiel. Zoznam všetkých známych potenciálnych rizík by mal byť súčasťou každého projektového plánu. Kritické riziká by mali byť dokumentované a prehodnocované detailnejšie počas celého vývoja.

Pravidlo *kombinovať interné zhodnotenia a externé posudky* je aplikovateľné na riziká ako sú: neprimeraný počet ľudí na projekt, nepostačujúca metodológia, slabý odhad a nesprávna definícia rolí a zodpovedností členov tímu.

Angažovanie vyššieho manažmentu počas celého životného cyklu projektu hovorí o tom, že projektoví manažéri by mali periodicky zhodnotiť úroveň záujmu vyššieho manažmentu a používateľov aby podpora projektu nevyprchala.

Podobne aj *angažovanie používateľov počas celého životného cyklu* je vhodným opatrením proti určitým rizikám. Projektoví manažéri by nemali meniť rozsah projektu, bez konzultovania s používateľmi a bez revízie nákladov a časového plánu.

Ako samozrejme sa javí pravidlo *prideliť zodpovednosti jednotlivým členom tímu*, avšak nie v každom projekte sa dodržiava. Týmto opatrením je možné zmenšiť manažovanú jednotku v projekte a tak identifikovať viacero úloh, čo vedie k zvýšenému dohľadu nad jednotlivými aktivitami.

Ďalším pravidlom je *vytvoriť náhradný plán pre zvládnutie personálnych problémov*. Správny počet ľudí na projekte je pre projekt kritický. Vysoký počet ľudí vedie k vyššej komunikácii a problémom s koordináciou, čo napokon vedie k nižšej produktivite jednotky. Nedostatočný počet ľudí zas často vedie k oneskoreniu projektu, nestabilným prioritám a nedostatočnému testovaniu.

Pravidlo *zahrnutia pravidelných formálnych zhodnotení rizík* pozostáva z troch krokov: identifikácia rizík, ich analýza a priradenie priorit. Analýza rizík pomôže k pochopeniu prečo, a kde sa môže riziko vyskytnúť. Priradenie priorit umožní obetovať obmedzené zdroje tímu na najzávažnejšie riziká.

Rozdelenie projektu na lepšie kontrolovateľné časti by malo prispieť k zvýšeniu dohľadu nad projektom. Mnoho manažérov vie, že rozsiahle projekty by mali byť rozdelené na menšie časti, avšak len málo je rozdelených na časti efektívne.

Veľmi zanedbávaným pravidlom je *príprava používateľov na zmeny už počas vývoja*. Počas vývoja sa veľmi často zanedbáva efektívna prezentácia, načúvanie a komunikácia medzi vyšším manažmentom a používateľmi.

Aj pravidlo *odhadnúť dopad zmien v požiadavkách na náklady a časový plán* vedie k zmierneniu niektorých rizík. Manažér by mal informovať používateľov o dopade zmien v požiadavkách a vedieť rozlíšiť medzi potrebnou a nevyhnutnou funkčnosťou.

Ako ďalšie spomeniem pravidlo *stabilizácie požiadaviek a špecifikácie čo najskôr*. Manažér by mal zabezpečiť, že používatelia systému sú zaangažovaný v definovaní a stabilizácii detailných požiadaviek čo najskôr, teda už v úvodných fázach životného cyklu projektu.

Zhodnotenie postupu projektu a nastavenie ďalších krokov súvisí s prvým pravidlom – vytvorenie a dodržiavanie plánu vývoja. Postup projektu by mal byť často porovnávaný s projektovým plánom, a v prípade neočakávaných udalostí, alebo nezhode s plánom je nutné si povedať ako ďalej.

Efektivita opatrení

Niektoré z opatrení proti výskytu rizík či na ich úplnú elimináciu má dopad na viac než jedno riziko. Z [1] vyplývajú ako najefektívnejšie nasledovné opatrenia:

- Rozdeliť projekt na lepšie kontrolovateľné časti, eliminuje:
 - Nejasný alebo zle pochopený účel či rozsah projektu
 - Podhodnotené náklady a časové plány
 - Nedostatočne efektívna metodológia manažmentu
 - Vývoj nesprávnych funkcií softvéru
- Stabilizovať požiadavky a špecifikáciu čo najskôr, eliminuje:
 - Nejasný alebo zle pochopený účel či rozsah projektu
 - Nedostatočná angažovanosť používateľov
 - Nedostatočne efektívna metodológia manažmentu
 - Vývoj nesprávnych funkcií softvéru
- Zaangažovať vyšší manažment počas celého životného cyklu, eliminuje:
 - Nejasný alebo zle pochopený účel či rozsah projektu
 - Podhodnotené náklady a časové plány
 - Neustála zmena požiadaviek
- Zahnúť pravidelné formálne zhodnotenie rizík, eliminuje:
 - Podhodnotené náklady a časové plány
 - Nedostatočne efektívna metodológia manažmentu
 - Neustála zmena požiadaviek

Záver

Najčastejším rizikovým faktorom vyskytujúcim sa v softvérových projektoch je podľa [1] neustála zmena požiadaviek a podhodnotené náklady a časové plány. Podľa [1] je tiež zjavné, že 7 z desiatich najzávažnejších rizík je možné redukovať vhodnými opatreniami. Niektoré z rizík, ako napríklad podhodnotené náklady a časové plány alebo neporozumené požiadavky sa objavujú menej často s pribúdajúcimi skúsenosťami manažérov.

Ako je z tejto práce vidieť, manažmentom rizík v softvérových projektoch sa naozaj oplatí zaoberať. Aj z vlastných skúseností môžem povedať, že spomenuté riziká sú naozaj reálne a v projektoch sa vyskytujú často. Preto nie je nič udivujúce, že tak ako sa oplatí zaoberať identifikáciou rizík, oplatí sa robiť aj opatrenia na ich elimináciu prípadne zníženie ich dopadu. Osobne môžem potvrdiť, že *nejasný alebo zle pochopený účel či rozsah projektu* a *nesprávne pochopené požiadavky* patria k jedným z najväčších rizík projektov a veľmi často vedú k vývoju nesprávnej funkcionality, čím sa pri následnej oprave predlžuje čas potrebný na vývoj. Preto môžem s kľudným svedomím prehlásiť, že bez manažmentu rizík sa úspešný softvér (podľa definície v úvode) vytvoriť jednoducho nedá.

Použitá literatúra

1. Addison, T., Vallabh, S.: Controlling software project risks – an empirical study of methods used by experienced project managers. *Proceedings of SAICSIT*, (2002) 128-140.
2. Cule, P., Schmidt, R., Lyytinen, K., Keil, M.: *Corporate information systems management: The issues facing senior executives*. Down Jones-Irwin, USA, 2000.
3. Ewusi-Mensah, K., Przasnyski, Z.H.: On information systems abandonment: An exploratory study of organisational practise. *MIS quarterly*, Vol. 15, No. 1 (1991) 67-88.
4. Keil, M., Cule, P.E., Lyytinen, K., Schmidt, R.: A framework for identifying software project risks. *Communication of the ACM*, Vol. 41, No. 11 (1998) 77-83.
5. May, L.J.: Major causes of software project failures.
<http://stsc.hill.af.mil/crosstalk/1998/jul/causes.asp>. 25.6.2002.
6. Powell, P.L., Klein, J.H.: Risk management for information system development. *Journal of information technology*, Vol. 11 (1996) 309-319.

Annotation

Riziká softvérových projektov a možnosti zníženia ich dopadu

The failure rate of software projects has been proven to be very high, and the impact of failure on global economy is becoming worse as more companies venture into software development. Most software projects are developed in an unpredictable environment in which many pitfalls exist. That may affect the successful outcome of a project. Post-project evaluations reveal that many of those problems encountered were in fact predictable. This paper focuses on the most significant risks that may occur during development and the controls used to overcome or minimize the effect of these risks.