

AKÉ NÁSTRAHY V SEBE SKRÝVA MANAŽMENT RIZÍK?

V jednoduchosti je krása.

Michal Lihocký

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

Abstrakt. Manažment rizík je činnosť, ktorá napriek tomu, že je všeobecne vnímaná za dôležitú, je často vykonávaná len v minimálnej miere či dokonca vôbec, čo predstavuje problém, ktorý je predmetom tejto eseje. Čo teda predstavuje prekážku pri zavádzaní manažmentu rizík do vývoja? S akými ďalšími problémami sa pri vykonávaní manažmentu rizík môžeme stretnúť? Aké sú najčastejšie riziká pri vývoji softvérových projektov? Na čo sa oplatí zamerať? Existuje nejaký prístup či metóda, ktoré sa javia byť riešením spomínaných problémov? Je takýto prístup použiteľný aj v našom projekte v rámci predmetu Tímový projekt? Práve nad týmito otázkami som sa pokúsil zamyslieť v tejto eseji.

Kľúčové slová: vývoj softvéru, riziká, manažment rizík, cieľmi riadený manažment rizík

Dôležitosť manažmentu rizík

Existuje mnoho dôvodov, kvôli ktorým môže projekt skončiť neúspechom a pri vývoji projektu je preto nesmierne dôležité venovať pozornosť problémom a možným situáciám, ktoré by mohli mať negatívny dopad na priebeh alebo samotný výsledok projektu. Predchádzaním takýchto problémov sa zaoberá práve manažment rizík, ktorého realizácia v praxi je však stále problémová. Ale ako je vlastne možné, že ľudia si uvedomujú dôležitosť manažmentu rizík, no napriek tomu existuje množstvo projektov, v ktorých sa manažment rizík vykonáva len v minimálnej miere, či dokonca vôbec?

Na začiatku je vhodné si uvedomiť problémy, ktoré nastávajú už pri samotnom zavádzaní rizík do vývoja projektu. Zavedenie manažmentu rizík však ani zďaleka nestačí.

Problémové je taktiež aj jeho samotné vykonávanie. Preto je vhodné venovať pozornosť otázkam akými sú: Aké sú najčastejšie problémy pri zavádzaní a vykonávaní manažmentu rizík? Na čo sa oplatí zamerať? Existujú nejaké osvedčené prístupy v tejto oblasti? Existuje nejaký prístup, ktorý by sa javil ako východiskový pri týchto problémoch?

Problémy pri zavádzaní rizík do vývoja

Odzaly, Greer a Sage vypracovali empirickú štúdiu [1] zaoberajúcu sa prekážkami, kvôli ktorým je zavedenie manažmentu rizík problémové. Na základe skúseností projektových manažérov z rôznych spoločností autori identifikovali najčastejšie problémy spojené s manažmentom rizík. Podľa môjho názoru je porozumenie týchto prekážok kľúčové pre efektívne zavedenie manažmentu rizík do vývoja softvéru.

Autori poukázali na to, že najväčšou prekážkou je vysoká cena manažmentu rizík v porovnaní s jeho návratovou hodnotou. Ale ako máme odhadnúť jeho návratovú hodnotu? Osobne si myslím, že ak by aj existovali metódy na vyčíslenie návratovosti manažmentu rizík, boli by veľmi subjektívne. Dokázať jeho skutočnú hodnotu je takmer nemožné. Veď len ťažko vieme dokázať, že práve naše úsilie v súvislosti možných problémov v budúcnosti predstavovalo ten kľúčový faktor, ktorý viedol k zmierneniu dopadu či prevencii určitých rizík. Ľudia tak manažment rizík následne považujú za činnosť, do ktorej sa neoplatí investovať zdroje, či už ide o čas, ľudí alebo peniaze. Preto si myslím, že manažment rizík by mal byť lacný a časovo nenáročný, no zároveň dostatočne efektívny na to, aby si jeho hodnotu uvedomovali nielen manažéri, ktorí sú zaň zodpovední, ale aj všetci ostatní, ktorí sú akýmkoľvek spôsobom zapojení do vývoja.

Štúdia poukazuje aj na to, že manažment rizík je veľmi náročný, ak sa vykonáva ako samostatná aktivita nezávisle od projektu. K tomuto názoru sa taktiež prikláňam a ako jednoduché riešenie tohto problému vidím zapojenie do manažmentu rizík všetkých ľudí, ktorí sa podieľajú na vývoji softvéru, čo podľa mňa v konečnom dôsledku povedie k vnímaniu manažmentu rizík ako prirodzenej súčasť vývoja. Myslím si, že ľudia, ktorí pracujú na jednotlivých častiach projektu, sú práve tí, ktorí majú o danej časti najlepší prehľad, a teda práve oni sú tí, ktorí by mali manažment rizík vykonávať. Veď už len samotná identifikácia rizík sa zdá byť aktivita, do ktorej by mali byť zapojení všetci zúčastnení. Veď kto iný vie lepšie identifikovať možné problémy s dodržaním plánu projektu ako osoba, ktorá je zodpovedná za plánovanie? Kto iný vie lepšie identifikovať možné problémy s použitými technológiami ako samotní vývojári, ktorí ich používajú? Kto iný vie lepšie identifikovať problémy s rozdelením práce či nedostatkom personálu ako osoba, ktorá je zodpovedná za pridelovanie úloh? Štúdia uvádza, že manažment rizík, ktorý sa samostatne vykonáva nezávisle od vývoja projektu, môže byť náročný. Podľa mňa však takto vykonávaný manažment rizík môže byť nie len náročný, no dokonca až zbytočný z hľadiska jeho skutočného prínosu pre projekt. Pokiaľ do aktivít, akými sú identifikácia rizík či návrh možných stratégií a riešení pri ich odstraňovaní, nie sú zapojení ľudia, ktorých sa to bezprostredne týka, je podľa mňa takýto manažment rizík bezvýznamný.

Existujú však aj psychologické problémy, s ktorými sa stretávame pri zavádzaní manažmentu rizík. Ako príklad takéhoto problému autori uvádzajú považovanie identifikácie možných rizík za negatívne myslenie, ktoré môže byť vnímané ako

domotivujúce a teda nežiadúce. Osobne považujem za dôležité, aby boli ľudia v tomto smere motivovaní a to nie len pri identifikácii a analýze rizík, ale taktiež aj pri navrhovaní preventívnych opatrení či úprav v snahe zmierniť možný dopad týchto rizík. Taktiež si myslím, že pri identifikácii možných negatívnych situácií v budúcnosti je veľmi dôležitý spôsob komunikácie jednotlivcov. Ak by ľudia len konštatovali, ako čo môže zlyhať, atmosféra v takomto prostredí by bola vnímaná negatívne. Rovnako tak negatívne by bol potom však vnímaný aj samotný manažment rizík, ktorému by ľudia za to mohli pripisovať vinu. Podľa mňa je preto dôležité, aby pri identifikácii takýchto problémov bola pozornosť zameraná viac na zmiernenie dopadu či odstránenie týchto problémov ako problémy samotné.

Ak si aj poradíme so spomínanými problémami a podarí sa nám zaviesť manažment rizík do vývoja, ešte to ani zďaleka nemusí nič znamenať. Na množstvo ďalších problémov narazíme aj počas samotného vykonávania manažmentu rizík. Ešte dôležitejšie ako to, či sa bude manažment rizík vykonávať, je ako sa bude vykonávať, pričom je dobré poznať známe osvedčené prístupy a rovnako tak aj chyby, ktorých sa dopustili iní. Na začiatku je dobré si uvedomiť aké sú najčastejšie a najdôležitejšie riziká, s ktorými sa môžeme stretnúť, čo nám môže pomôcť najmä pri identifikácii rizík.

Časté riziká pri vývoji softvérových projektov

Addison a Vallabh vo svojej štúdii [2] skúmali rôzne riziká pri vývoji softvérových projektov, ktoré následne zoradili podľa dôležitosti a pravdepodobnosti výskytu. Poukázali na to, že tri najčastejšie riziká sú: *nerealistický plán a rozpočet, neustále meniace sa požiadavky a nejasné alebo neporozumené ciele a účel projektu*. Osobne si myslím, že na tieto tri riziká treba pri zavádzaní manažmentu rizík do nového projektu skutočne myslieť, keďže môžu veľmi nepriaznivo ovplyvniť vývoj celého projektu.

V súvislosti s prvým rizikom je potrebné dávať pozor na to, aby sa nepodceňoval rozsah a zložitosť, a to nie len projektu ako takého, ale aj čiastkových úloh, ktoré sa v rámci vývoja vyskytujú. Autori poukázali aj na to, že fixný rozvrh môže spôsobiť ľuďom stres a ľudia pod tlakom nie vždy pracujú efektívnejšie. S týmto problémom absolútne súhlasím a na základe vlastných skúseností môžem povedať, že ľudia, ktorí cítia potrebu niečo dokončiť v krátkom časovom termíne majú tendenciu sa zamerať len na hlavné požiadavky a zanedbajú určité potencionálne problémy. Takýto softvér potom obsahuje chyby, ktoré sa nemusia odhaliť včas, čo predstavuje „časovanú bombu“ a teda obrovské riziko. Neskôr v situáciách, kedy je potrebné rýchlo opraviť určité chybné správanie softvéru, sa niekedy vytvorí oprava, ktorá nie je v súlade s pôvodným návrhom tohto softvéru. Vytvorením každej ďalšej takejto opravy sa softvér stáva menej udržiavateľným a taktiež sa to negatívne prejavuje aj na jeho rozšíriteľnosti. To taktiež predstavuje riziko, ktorého dopad sa prejaví najmä vo fáze údržby softvéru. Vyhradenie primeraného času na jednotlivé úlohy považujem za kľúč k efektívnej práci a kvalitným výsledkom.

Ako ďalšie veľmi časté riziko autori identifikovali neustále meniace sa požiadavky. S týmto však nie tak celkom súhlasím. Áno, niekedy sa môže stať, že sa počas dlhšieho obdobia vykonáva určitá práca a ak sa medzičasom zmenili požiadavky, tak sa na takýto problém môže prísť veľmi neskoro a práca ľudí tak bola zbytočná. Ale dá sa všeobecne povedať, že ide o veľmi časté riziko? Vezmime si taký agilný vývoj softvéru. Veď je to

prístup, ktorý je práve neustále meniacim sa požiadavkám prispôsobený. Pravdepodobnosť toho, že zmena požiadaviek bude mať negatívny dopad na projekt je preto nižšia v porovnaní so štandardnými metodológiami vývoja softvéru. Negatívnemu dopadu meniacim sa požiadavkám zo strany zákazníka vieme predchádzať najmä vďaka častej komunikácii so zákazníkom, častom dodávaní funkčného softvéru, častej komunikácii v rámci tímu a prototypovaniu. Ale za zmeny požiadaviek však nemusí byť zodpovedný len zákazník. Problémy môže spôsobiť napríklad aj zmena legislatívy, ktorej dopadu sa niekedy zabrániť jednoducho nedá. To sa však týka najmä projektov, pri ktorých požiadavky na určitú funkcionálnu vyplývajú priamo zo zákona. Pri iných projektoch je však pravdepodobnosť výskytu takéhoto rizika ďaleko nižšia.

Posledným z uvedených rizík je riziko neporozumenia cieľov a účelu projektu. A to je práve problém, ktorý môže predstavovať riziko aj pri agilnom vývoji a teda aj pri práci na projekte v rámci predmetu Tímový projekt, kde vyvíjame agilne podľa metodológie Scrum. Neporozumenie účelu jednotlivých častí projektu sa odzrkadľuje v podobe konfliktu záujmov, keďže rôzni ľudia môžu mať rôzne očakávania. A aj keď sa tým stretáva veľmi často a je jasné, kto by mal na čom pracovať, stále tu ostáva riziko, že predstava daného člena o danej práci je odlišná od predstavy ostatných členov. A hoci sa na takýto problém príde pomerne skoro, môže to mať veľmi negatívny dopad na vývoj projektu. Preto si myslím, že je veľmi dôležité, aby tím na spoločných stretnutiach nediskutoval len o tom, čo je potrebné spraviť, ale taktiež aj na to, ako sa to spraví. Myslím si, že diskusia o tom, čo všetko práca na určitých úlohách zahŕňa, napomôže aj pri skorej identifikácii problémov, ktoré môžu pri práci nastať. A práve skoré odhaľovanie možných problémov môže byť pre projekt obrovským prínosom.

Autori tejto štúdie však neskúmali len riziká, ale rovnako tak aj aktivity, ktoré sa ukazujú ako najvhodnejšie pri ich prevencii. Na čo sa teda pri vykonávaní manažmentu rizík uplatí zamerať? Aké sú najčastejšie prístupy pri prevencii spomínaných rizík?

Najčastejšie prístupy pri prevencii a zmierňovaní dopadu rizík

Addison a Vallabh vo svojej práci [2] identifikovali taktiež aj aktivity, ktoré napomáhajú pri prevencii spomínaných rizík. Myslím si, že uvedenie si týchto aktivít môže značne napomôcť pri navrhovaní konkrétneho spôsobu vykonávania manažmentu rizík. Ako tri najefektívnejšie aktivity autori identifikovali: *rozdelenie projektu na kontrolovateľné časti, snaha o skoré ujasnenie požiadaviek a zahrnutie pravidelného formálneho posudzovania rizík do vývoja projektu.*

Rozdelenie projektu na kontrolovateľné časti, snaha o skoré ujasnenie požiadaviek, no taktiež aj častá komunikácia so zákazníkom a pravidelné zhodnotenie postupu práce, sú však aktivity, ktoré sú priamou súčasťou metodológie vývoja Scrum. Tímy, ktoré vyvíjajú podľa tejto metodológie, tak vykonávajú tieto aktivity ako prirodzenú súčasť vývoja, nie ako aktivity primárne určené na prevenciu určitých rizík. Ani zďaleka to však neznamená, že takéto tímy sa s týmito rizikami už nemusia zaoberať. Znamená to len to, že pravdepodobnosť ich výskytu je nižšia.

Pre nás je však zaujímavá najmä posledná zo spomínaných aktivít a to *zahrnutie pravidelného formálneho posudzovania rizík do vývoja projektu.* Pod posudzovaním sa myslí identifikácia rizík, následne ich analýza a na záver ich ohodnotenie, najmä z pohľadu

pravdepodobnosti výskytu a rozsahu možného dopadu. Autori tvrdia, že takéto posudzovanie rizík by malo prebiehať na začiatku projektu a pri vyhodnocovaní jednotlivých fáz vývoja. S týmto názorom sa plne stotožňujem a v takto prebiehajúcim vykonávaní manažmentu rizík vidím viacero ďalších výhod. Pri metodológii Scrum je práca rozdelená do takzvaných šprintov, pričom na konci každého šprintu väčšinou prebieha jeho zhodnotenie. Zhodnotenie šprintu býva väčšinou predmetom spoločného stretnutia tímu, čo vytvára ideálne podmienky pre identifikáciu a analýzu rizík a to nie len z pohľadu manažéra, ale rovnako tak z pohľadu jednotlivých členov tímu. Vďaka formálnosti tejto činnosti tím prirodzene udržiava zoznam najčastejších rizík, z ktorého vyplýva na čo by sa mal tím zamerať aby týmto rizikám predišiel. Osobne si myslím, že členovia tímu si tieto riziká najviac uvedomia práve na takýchto stretnutiach.

Aj keď poznáme určité odporúčané postupy a aktivity, ktoré nám môžu napomôcť pri vykonávaní manažmentu rizík, to ako konkrétne je vhodné manažment rizík vykonávať je stále otáznou. Aký prístup sa teda oplatí použiť?

Cieľmi riadený manažment rizík

Shareeful Islam sa vo svojej práci [3] zameril na nájdenie prístupu, ktorým by bolo možné jednoducho a systematicky zintegrovat' manažment rizík do vývoja projektu tak, aby to značne zlepšilo výstup projektu. Navrhol model manažmentu rizík, ktorý sa javí ako východiskový vzhľadom na spomínané problémy a taktiež je v súlade so spomínanými prístupmi pri prevencii rizík.

Tento model je založený na skorom odhaľovaní možných rizík, spoločnej komunikácii členov tímu, a to najmä na spoločných stretnutiach, a integrácii manažmentu rizík ako priamej súčasť vývoja softvéru. Osobne si myslím, že agilná metodológia vývoja Scrum je na uplatnenie tohto modelu ideálna a tento model by sa teda dal využiť aj na našom projekte v rámci predmetu Tímový projekt, kde vyvíjame softvér práve podľa metodológie Scrum.

Navrhovaný model rozdeľuje manažment rizík do 4 vrstiev: *vrstva cieľov*, *vrstva rizík*, *vrstva ohodnotenia rizík* a *vrstvu opatrení*. Osobne si myslím, že to najlepšie na tomto modeli je jeho jednoduchosť. Základná myšlienka spočíva vo vykonávaní identifikácie rizík ako priamej súčasť identifikácie a analýze cieľov. Vykonávanie manažmentu rizík v tomto prípade prebieha v štyroch krokoch: *stanovenie cieľov*, *identifikácia rizík a možných problémov*, *analýza a ohodnotenie rizík* a *návrh riešení jednotlivých rizík*.

V našom projekte na predmete Tímový projekt prebieha stanovovanie čiastkových cieľov ako súčasť plánovania, ktoré je založené na kolektívnych odhadoch a teda vykonávané na spoločných stretnutiach tímu. Pri časových odhadoch jednotlivých úloh je však presný odhad veľmi problémový a podceňovanie náročnosti úloh je v tejto súvislosti riziko, ktorého dopad môže mať na projekt zdravujúce následky. Z vlastných skúseností so stanovovaním takýchto cieľov môžem povedať, že v prípadoch kedy sa nestrávil dostatočne veľa času s identifikáciou možných problémov to často končilo nerealistickými plánmi. Niekedy sa pri riešení úloh vyskytlo toľko problémov, že boli úspešne splnené len vďaka bravúrnym, ba až nemožným, výkonom vývojárov. V takomto extrémnom prípade však ide už doslova o hazard, čo nemusí mať vždy šťastný koniec. Pri riešení projektu takémuto riziku nechceme čeliť a preto je veľmi vhodné stráviť s analýzou rozsahu

a náročnosti úloh viac času. Zapojenie identifikácie rizík do tejto činnosti je v tomto smere kľúčové.

Uvedený prístup rieši dokonca aj spomínané problémy, s ktorými sa stretávame pri zavádzaní manažmentu rizík do projektu. Myslím si, že manažment rizík v takejto podobe by nebol vnímaný ako subjektívny a zbytočný, ale by bol, práve naopak, podporovaný všetkými členmi tímu. Pevne verím, že aplikácia tohto prístupu by bola v konečnom dôsledku pre náš projekt skutočným prínosom.

Záver

Aj napriek problémom, s ktorými sa stretávame pri zavádzaní manažmentu rizík do vývoja, ide o činnosť, ktorou sa rozhodne oplatí zaoberať a to aj v projektoch menšieho rozsahu. V takom prípade je však potrebné brať ohľad na množstvo úsilia a času, ktoré náš manažment rizík môže stáť a zavedenie tejto činnosti do projektu patrične prispôbiť. Osobne si myslím, že v snahe dopracovať sa k čo najúspešnejšiemu výsledku nášho projektu v rámci predmetu Tímový projekt, sa ako východiskový prístup javí pravidelné vykonávanie manažmentu rizík na spoločných stretnutiach tak, aby bol vnímaný ako súčasť vývoja. Zároveň by malo ísť o činnosť, ktorá bude jednoduchá a časovo nenáročná, čo podľa môjho názoru povedie k uvítaniu tohto prístupu aj ostatnými členmi tímu. Uvedený prístup, vychádzajúci z cieľmi riadeného manažmentu rizík, sa pre náš projekt javí ako ideálny. Verím, že pri vykonávaní manažmentu rizík v takejto podobe pôjde o činnosť, ktorej vykonávanie bude pre projekt skutočným prínosom.

Použitá literatúra

1. Odzaly, E.E., Greer, D., Sage, P.: Software Risk Management Barriers: an Empirical Study. In: *Proc. of International Symposium on Empirical Software Engineering and Measurement*, October 2009, vol. 3, pp. 418-421
2. Addison, T., Vallabh, S.: Controlling Software Project Risks – an Empirical Study of Methods used by Experienced Project Managers. In: *Proc. of South African Institute for Computer Scientists and Information Technologists*, September 2002, pp. 128-140
3. Islam, S.: Software Development Risk Management Model – A Goal Driven Approach. In: *Proc. of ESEC/FSE Doctoral Symposium*, August 2009, pp. 5-8

Annotation

What kinds of traps are hidden within risk management?

Risk management is generally considered as an important part of software development, yet it happens very often that teams don't support it. But how come that people are not willing to spend any resources to support risk management although they realize its importance? What prevents them to do so? What are main barriers in introducing risk management? Even after we successfully introduce risk management within a project, there are still many problems that we will face. What are the other problems we may face regarding risk management? Are there any recommended best

practices in this field? Is there any concrete approach or model that would seem to be applicable in our project we develop within Team Project course? These are the main problems discussed in this essay.