

AKO STRACH OVPLYVŇUJE NAŠE PROJEKTY

Prečo bol Darth Vader skvelým šéfom? Nebál sa.

Ján Súkeník

Slovenská technická univerzita
Fakulta informatiky a informačných technológií
Ilkovičova 3, 842 16 Bratislava
sukenic08@student.fiit.stuba.sk

Abstrakt. *Softvérové projekty so sebou nesú veľký počet rôznych rizík. Tieto riziká bývajú aj dnes vo veľkej miere ignorované. Dôvodom tohto správania je strach. Ľudia sa o rizikách boja rozmýšľať a hovoriť, pretože je to považované za negatívne. Veľké percento softvérových projektov býva kvôli zatajeným alebo nenájdovým rizikám dokončených neskoro, alebo zrušených úplne. Práve hľadanie a vyhodnocovanie všetkých možných rizík počas realizácie softvérových projektov by mohlo znížiť percento neskoro dokončených a zrušených projektov. Riziká by sa dali odhaliť ešte vtedy, keď je ich odstránenie lacné. Esej sa zaoberá fenoménom nazvaným rizikám odporujúca kultúra v súvislosti so softvérovým inžinierstvom. Opisuje aj možné riešenie, ktoré je na prvý pohľad jednoduché a samozrejmé, jeho praktická aplikácia však vyžaduje zmenu v spôsobe akým myslíme.*

Kľúčové slová: *riziká, manažment projektov, rizikám odporujúca kultúra*

Úvod

Vývoj softvéru so sebou prináša množstvo rizík. Je jednoduché prečítať si tento fakt napísaný na papieri, napríklad v študentskej eseji. Omnoho zložitejšie je však uvedomiť si prítomnosť rizík počas riešenia naozajstných softvérových projektov a nejakým spôsobom proti nim bojovať.

Riziká nie sú charakteristické len pre oblasť softvérového inžinierstva. Vyskytujú sa aj v ostatných odboroch, napríklad stavebníctve, finančníctve, medicíne, atď. Tieto odbory s nimi počítajú ako s bežnou súčasťou všetkých procesov, snažia sa ich predpovedať

a zabraňovať im. Softvérovému inžinierstvu však aj dnes podobné myslenie často chýba a riziká v ňom bývajú pravidelne ignorované.

Prečo bývajú riziká ignorované?

Dôvodov, kvôli ktorým sa riziká zanedbávajú, existuje viacero. Asi najvýraznejším je fenomén, ktorý M. J. Carr [2] nazýva ako rizikám odporujúca kultúra. Je to celkom bežné správanie ľudí, ktoré súvisí so spôsobom akým myslíme. Neradi si totiž pripúšťame akékoľvek problémy a naivne veríme, že ich dokážeme kedykoľvek a bez následkov odstrániť. Takýto spôsob myslenia sa samozrejme objavuje v každej časti našich životov a tá profesionálna nie je výnimkou.

Čo je zlé na strachu z neznámeho?

Veľkou nevýhodou strachu z rizík sú jeho následky. Jedným z nich je trestanie tých ľudí, ktorí sa snažia odhaliť problémy v predstihu. Bývajú považovaní za negatívne mysliacich, obviňovaní z toho, že nemajú tímového ducha a že neveria v úspech projektu. Odmeňovaní bývajú naopak jednotlivci, ktorí rýchlo odstraňujú aktuálne problémy (teda riziká, ktoré sa už prejavili).

Ak si spomenieme na stredoškolské hodiny biológie, konkrétne na pána Ivana Petroviča Pavlova a jeho hladného psa, následky takéhoto správania musia byť každému jasné. Ak by v prostredí rizikám odporujúcej kultúry existovala nejaká snaha hľadať budúce problémy a v predstihu im zabraňovať, bude veľmi rýchlo potlačená. Do popredia sa dostanú rýchle a možno nie vždy úplne správne riešenia aktuálnych problémov. Bude im treba venovať stále viac a viac času, a budú stále viac a viac odpútať pozornosť od hlavného cieľa. Práve softvéroví inžinieri, resp. programátori by mali veľmi dobre vedieť, ako vie neustále nabaľovanie drobných zmien potopiť celkový výsledok.

Ak sú riziká naozaj také dôležité, nemal by byť okolo nich väčší rozruch?

Predpovedanie rizík a snaha zabrániť im je v odboroch iných ako softvérové inžinierstvo dobre známa, bežne sa používa a má dokázateľné a pozitívne prínosy. Viete si predstaviť burzového makléra, ktorý by sa nezaoberal vyhodnocovaním všetkých možných kombinácií jeho investícií? Viete si predstaviť doktora, ktorý nezváži výhody a nevýhody všetkých spôsobov liečby svojho pacienta?

V oblasti softvérového inžinierstva sa však dá nájsť len veľmi málo príkladov, ktoré by opisovali pozitívne alebo aj negatívne následky vyhodnocovania rizík. Ak niekde sú riziká bežnou a akceptovanou súčasťou vývoja softvéru, dotýční sa tým neradi chvália. Možno si uvedomujú, že riziká sú neoddeliteľnou súčasťou softvérového inžinierstva a považujú ich za svoje know-how, a teda za výhodu v konkurenčnom boji. Možno analýzu rizík len nechcú verejne priznať, napriek jej pozitívnym prínosom. Pripúšťali by tým, najmä pred svojimi zákazníkmi, že vedia, že ich projekty môžu zlyhať. Navyše dokonca vedia kde presne môže zlyhať. Tým sa opäť dostávame k spomínanej rizikám odporujúcej kultúre, tentokrát z pohľadu zákazníkov. Odhaľovanie a prevenciu rizík môžu aj oni vnímať ako negatívne myslenie a považovať ich za niečo nežiadúce.

Pozitíva a negatíva vyhodnocovania rizík sa navyše veľmi ťažko dokazujú. Úspechy aj neúspechy sa vždy pripisujú vykonaným akciám a konkrétnym ľuďom – vedeniu, manažérom, a niekedy aj zamestnancom. Málokedy sa však na ich príčiny pozerá ako na dôsledok rozhodnutí, ktoré vyplynuli z analýzy rizík.

Ako teda narábať s rizikami v softvérovom inžinierstve?

Pojem tzv. manažmentu rizík ako súčasť vývoja softvéru nie je žiadnou novinkou. Prvýkrát sa začal objavovať už začiatkom 90. rokov minulého storočia. Odvtedy vzniklo mnoho návodov, štandardov či smerníc, ktoré sa snažia túto aktivitu nejakým spôsobom formálne zadefinovať. Prehľadné zhrnutie týchto formálnych metód vytvoril P. L. Bannerman [1]. Rozdeľuje ich do niekoľkých kategórií, pričom jednotlivé kategórie sa môžu vzájomne dopĺňať.

Dajme formálnym metódam šancu

Formálne metódy vytvorili šikovní a skúsení ľudia, ktorí nad nimi určite strávili veľké množstvo času. Zakomponovali do nich svoje znalosti a praktické skúsenosti a ostatní už nemusia tieto skúsenosti získavať sami. Používanie formálnych metód pri hľadaní a vyhodnocovaní rizík preto nevnímam ako zlú vec. Práve naopak, je rozumné sa nimi inšpirovať. Slovo inšpirovať som použil zámerne, pretože si viem predstaviť situácie, kedy by spomínané štandardy mohli byť zbytočne zložité a tým spôsobovať viac škody ako úžitku. Najmä v prostredí malých tímov nie je potrebný až taký stupeň formality, aký tieto metódy špecifikujú.

Formálne metódy hľadania a vyhodnocovania rizík boli vytvárané aj s inými cieľmi. Jedným z nich je poskytnúť akýsi druh certifikácie. Vďaka tomu ja ako zákazník, ktorý si uvedomuje dôležitosť a hrozby rizík, viem o každej firme hneď rozhodnúť, či spĺňa moje požiadavky v tomto smere.

Pomôžu formálne metódy odstrániť strach z rizík? Nepomôžu!

Rizikám odporujúca kultúra je založená na strachu z neznámeho. Ľudia sa boja premýšľať a hovoriť o rizikách. Čo sa teda stane ak takýchto ľudí donútíme používať nejaký formálny systém hľadania a vyhodnocovania rizík? Určite sa strach z rizík zázračne nestratí. Budú mať akurát falošný pocit, že o riziká je postarané. Formálny systém nič nemení na tom, že nechápú hlavnú myšlienku hľadania a vyhodnocovania rizík – tú zatiaľ nepreznám, nájdete ju v závere. Kvôli tomu si nebudú vedieť používaný formálny systém ani nijako zjednodušiť alebo prispôbiť. Z toho vyplýva, že sa bude vykonávať množstvo činností navyše, ktoré budú úplne zbytočne krátiť reálny produktívny čas. Postupne sa možno kvôli tomu objaví potreba prijať ďalších ľudí. A výsledok? Úplne zbytočné predrazenie celého projektu. Nechufť ľudí venovať svoj čas veľkému množstvu formalít, a následná nekvalitne odvedená ostatná práca. Pozitívne prínosy? Žiadne. Zle vykonávané hľadanie a vyhodnocovanie rizík nie je o nič lepšie, ako keby sa nepoužívalo žiadne. Kvôli tomu sa pravdepodobnosť odhalenia rizík a tým aj pravdepodobnosť úspešného dokončenia projektu nezvýši takmer vôbec. Výsledok bude drahší, ale nebude lepší.

Existuje šanca, že aj ľudia z prostredia rizikám odporujúcej kultúry sa vďaka formálnym metódam postupne naučia, čo riziká naozaj sú. Ja si však myslím, že opačný prístup je vhodnejší a menej náročný. Najprv sa treba zbaviť strachu z rizík a tým samotnej rizikám odporujúcej kultúry, a až potom, resp. popri tom by sa mal zavádzať formálny proces hľadania a vyhodnocovania rizík.

Volám sa softvérové inžinierstvo a bojím sa rizík

Ak sa chceme zbaviť rizikám odporujúcej kultúry, musíme vykonať zmenu v spôsobe myslenia. K obyčajnému plneniu úloh treba pridať kritické myslenie a určitú formu odhadovania budúcnosti. Ako každý problém, aj tento sa dá rozdeliť na menšie časti.

Prvou časťou sú projektoví manažéri, resp. ľubovoľné iné elegantné názvy pre pozíciu, ktorá bola kedysi známa pod slovom šéf. Práve šéf je tým, ktorý sa musí rizík prestať báť ako prvý. Ak každú chvíľu bude mať na pamäti všetky zlé a dobré alternatívy, bude sa vedieť lepšie rozhodovať. Bude vedieť predpokladať pravdepodobnosti jednotlivých alternatív a ovplyvňovať podľa nich smerovanie projektu. Bude vedieť vykonávať akcie, ktoré znižujú pravdepodobnosti zlých alternatív, resp. znižujú rozsah ich následkov. Ak sa šéf nezbaví strachu z rizík, nezbaví sa ho ani nikto iný. Ďalším problémom je, že šéfovi nemá kto dôležitosť rizík vysvetliť a musí si ju uvedomiť sám.

Druhou časťou sú zamestnanci. Aj oni by mali myslieť podobne, ako je to nevyhnutné u šéfov. Tu sa však na dosiahnutie tohto efektu dajú použiť rôzne motivačné prostriedky. Príkladom je trestanie tých ľudí, ktorí mohli nejakému prejavému riziku včasne zabrániť. Okrem odmeňovania úspešného riešenia prejavých rizík treba ďalej odmeňovať aj tých, ktorí myslia na budúce riziká a podarí sa im minimalizovať ich pravdepodobnosť alebo možný dopad. Treba si teda vytréňovať zamestnancov, ktorí sa neboja hovoriť o rizikách. Ideálne riziká aj sami hľadajú a ponúkajú možné riešenia.

Tretiu časť tvoria zákazníci. Tým je tiež potrebné vysvetliť, ako môže hľadanie a vyhodnocovanie rizík zlepšiť šance ich projektu na úspech. Treba im ukázať, že tá trocha pesimistického myslenia nie je nežiadúca vec. Ak toto zákazník pochopí, môže sa sám stať súčasťou hľadania a vyhodnocovania rizík, a tým sám sebe zvýšiť šance na úspech. On sa pohybuje v oblasti, ktorej sa projekt týka, a má špecifické znalosti v tejto oblasti. Práve tieto znalosti môžu pomôcť nájsť najzávažnejšie riziká, ktoré nemusia súvisieť so softvérovým inžinierstvom, ale s praktickou realizáciou projektu. A v prípade, že zákazník dôležitosť rizík nepochopí, bude ho možno rozumnejšie nechať ísť ku konkurencii. Nech robí zbytočné problémy niekomu inému.

Ako by malo dobré hľadanie a riešenie rizík vyzerat'?

B. Koeller [3] vo svojom článku *Top 10 reasons why Darth Vader was an amazing project manager* opisuje ako ideálneho šéfa postavu Darth Vadera z filmov Georga Lucasa – Hviezdne vojny. Darth Vader vedel totiž vykonávať ukázkové hľadanie a vyhodnocovanie rizík, na základe ktorých potom robil svoje rozhodnutia a konkrétne akcie.

Hneď na začiatku prvého filmu môžeme vidieť, že po ukradnutí konštrukčných plánov Hviezdy smrti rebelmi Darth Vader identifikoval veľké riziko a okamžite sa snažil minimalizovať jeho dopad zajatím lode, ktorá viezla dané plány a neskôr zničením celej planéty. Ďalším rizikom bolo, keď sa Darth Vader dozvedel, že jeho syn Luke Skywalker

pomáha rebelom. Okamžite riziko riešil – snažil sa z Luke-a spraviť spojencu a získať ho na temnú stranu Sily. Spomínate si na časť, keď podplatil celé Cloud City, aby pripravil pascu do ktorej chytil všetkých hlavných hrdinov? Han Solo, Chewbacca, Leia a nakoniec aj Luke. Jednoducho geniálne. Darth Vader presne vedel, aké má pred sebou riziká, vybral najdôležitejšie a naplánoval riešenie, ktorým riziko odstránil. Viete nájsť naopak príklad, keď mal Darth Vader z rizika strach? Alebo keď si riziko nechcel pripustiť? Mal snáď niekedy strach spraviť rozhodnutie? Nie.

Úspešné odstránenie rizikám odporujúcej kultúry je teda nevyhnutným predpokladom pre zavedenie manažmentu rizík do softvérového inžinierstva. Až potom môže manažment rizík úspešne využívať existujúce formálne metódy (štandardy, smernice, atď.) vytvorené špeciálne pre tento účel. Stále však treba mať na pamäti hlavné princípy hľadania a vyhodnocovania rizík, ktoré spomína M. J. Carr [2]:

1. riziká treba hľadať vo všetkých aspektoch projektu,
2. riziká treba hľadať pravidelne, nie len jednorázovo,
3. každý projekt je úplne odlišný od ostatných, t.j. má vlastné, špecifické riziká.

Použitá literatúra

1. Bannerman, P.L.: Risk and risk management in software projects: A reassessment. *Journal of Systems and software*, Vol. 81, No. 12 (2008), 2118-2133.
2. Carr, M.J.: Counterpoint: Risk management may not be for everyone. *IEEE Software*, Vol. 14, No. 3 (1997), 21,24.
3. Koeller, B.: *Top 10 reasons why Darth Vader was an amazing project manager*. (2011)
<http://www.geekwire.com/2011/top-10-reasons-darth-vader-amazing-project-manager>
(Online, 25.11.2011)

Annotation

How fear affects our projects.

Risk management in software projects is often being neglected. Main reason for that may be the phenomenon called risk-averse culture. Essay describes this phenomenon. Author argues that only one available solution to this problem exists. The solution is then discussed, and guidelines for risk management in software projects are suggested.