

Riziká školských a komerčných softvérových projektov

MICHAL POLÁČIK

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

Abstrakt. Komerčné i školské softvérové projekty často prekračujú dohodnuté termíny, rozpočet, neobsahujú požadovanú funkcionálnosť alebo úplne zlyhávajú. Príčinou je nerozpoznanie alebo podcenenie vzniku možných neplánovaných situácií – rizík. Táto neznalosť je pre firmy príliš drahá z hľadiska peňazí i reputácie. Študenti kvôli nej strácajú čas a dostávajú horšie známky. Poukazuje to na dôležitosť kvalitného riadenia rizík softvérových projektov nielen vo firmách, ale aj v škole. Esej porovnáva prostredie univerzity s prostredím veľkých softvérových firiem. Skúma najväčšie riziká komerčných projektov z pohľadu skúsených projektových manažérov a spôsoby ako ich zmierniť. Cieľom práce je využiť získané poznatky na zvýšenie kvality školských projektov a zníženie stresu študentov.

Prečo riadiť riziká

70% všetkých komerčných softvérových projektov prekročí rozpočet, čas dodania, obsahuje menšiu funkcionálnosť ako sa dohodlo alebo úplne zlyhá [6].

Tiež vysokoškolskí študenti majú problémy pri riadení školských softvérových projektov. Často sa stáva, že:

- odovzdajú projekt oneskorene alebo vôbec
- odovzdaný produkt nefunguje
- prinesú dobré riešenie nesprávneho problému
- svoje programy často a vo veľkom rozsahu prerábajú
- venujú priveľa alebo primálo času jednotlivým vývojovým činnostiam

Zdá sa, že riadenie projektu nie je triviálna záležitosť. Aj keď sa projekt podrobne naplánuje, často sa vyskytne nejaká udalosť, ktorá spôsobí nedodržanie plánu.

Skúmaním možného výskytu neočakávaných udalostí a ich predchádzaním sa zaoberá manažment rizík.

Vymedzenie pojmov

Aby som predišiel nedorozumeniam pri čítaní eseje, rozhodol som sa uviesť na úvod vysvetlenie dôležitých pojmov, ktoré v texte často používam:

- *Riziko* – možnosť výskytu nepredvídanej udalosti spôsobujúcej škodu alebo stratu
- *Zákazník* – zadávateľ a hodnotiteľ projektu
- (*Koncový*) *užívateľ* – človek, ktorý bude produkt používať v praxi

Manažment rizík

Manažment rizík zahŕňa 4 procesy, ktoré prebehnú minimálne raz počas každej etapy projektu:

1. Identifikácia rizík
2. Kvantifikácia rizík
3. Zohľadnenie rizík
4. Sledovanie rizík počas celého projektu

Pre identifikáciu rizík sa používajú rôzne nástroje, najčastejšie zaškrtačacie zoznamy, diagramy a rozhovory so zainteresovanými ľuďmi. Cieľom je odhaliť potenciálne hrozby či príležitosti, ktoré by mohli výrazne ovplyvniť úspešnosť projektu, t.j. jeho cenu, čas dodania a prijatie zákazníkom.

V ďalšej fáze sa identifikovaným rizikám priradí pravdepodobnosť ich výskytu a potenciálnu škodu, ktorú môžu spôsobiť, resp. ich prínos, ak ide o príležitosť.

K rizikám je možné pristupovať tromi základnými spôsobmi:

1. Vyhnutie sa riziku, najčastejšie odstránením príčin jeho vzniku
2. Oslabenie rizika znížením pravdepodobnosti výskytu prípadne zoslabením škody, ktorú by mohlo spôsobiť.
3. Prijatie rizika a jeho dôsledkov, čo sa môže diať:
 - *Aktívne* – vypracovaním plánu, ako postupovať v prípade výskytu neočakávanej udalosti
 - *Pasívne* – zmierenie sa s dôsledkami

Nemenej dôležitou súčasťou manažmentu rizík je aj vykonávanie rizikového plánu, sledovanie a zohľadnenie zmien, ktoré znamenajú jeho aktualizáciu. Počas

trvania projektu vznikajú a zanikajú rôzne hrozby a rovnako aj ich pravdepodobnosť a škoda, ktorú môžu spôsobiť, sa zvykne meniť.

Školské prostredie

V eseji hľadám využitie poznatkov z riadenia rizík z veľkých softvérových firiem pre projekty realizované študentmi vysokých škôl. Preto považujem za vhodné najskôr analyzovať špecifiká školských projektov. Pre lepšiu predstavu, som študentské projekty rozdelil do 4 skupín:

1. *Zadania* slúžia na preverenie a praktické prehĺbenie vedomostí študenta. Ich výsledkom zvyčajne nie je ani prakticky použiteľná aplikácia ani výskum nových poznatkov. Vypracúva ich jeden, niekedy i viacerí študenti spoločne po dobu niekoľkých týždňov.
2. *Súťažné projekty* hodnotí komisia súťaže na základe stanovených kritérií (dizajn, inovatívnosť a i.). Výsledný produkt môže byť použiteľný v praxi, ale nie je to podmienkou. Vývoj trvá od niekoľkých týždňov po max. 1 rok. Existujú súťaže pre jednotlivcov i pre malé tímy (do 6 členov).
3. *Výskumné projekty* majú za cieľ priniesť nové vedecké poznatky. Nevyvíja sa žiaden systém pre konečného používateľa.
4. *Projekty pre prax* prinášajú produkt, ktorý bude mať svojich používateľov. Najčastejšie ide o projekty pre potreby školy, pre podporu výučby. Môže ísť aj o projekt na zákazku pre organizáciu či firmu alebo generický softvér.

Uvedené rozdelenie je len orientačné a nie každý školský projekt sa musí dať jednoznačne zaradiť do niektorej z uvedených skupín. Vo svojej eseji sa zameriavam najmä na posledný typ projektov.

Pri riadení softvérového projektu v škole treba brať do úvahy špecifiká tohto prostredia. Zákazníkom je väčšinou učiteľ, ktorý úspešnosť projektu hodnotí známku, resp. počtom bodov, ktoré udelí každému členovi tímu individuálne. Študentov nečaká žiadna finančná odmena. Výnimkou sú niektoré súťažné projekty v prípade najúspešnejších produktov. Vývojový tím tvorí zvyčajne 1 až 6 ľudí. Počas procesu vývoja nie je možné tím dopĺňať o nových členov. Študenti väčšinou nemajú veľa skúseností z praxe, či už odborných alebo manažérskych. Projekty trvajú od niekoľkých týždňov po maximálne 2 semestre. Termín odovzdania je pevný, preto nedodržanie časového plánu sa prejaví znížením dodanej funkcionality. Náklady na projekt nezvyknú byť finančné, iba časové.

Štyri kategórie rizík

Medzinárodná štúdia z roku 1998 [2] identifikovala riziká, ktorých sa zúčastnení projektoví manažéri z 3 krajín najväčšmi obávali. Autori štúdie na základe jej výsledkov a komentárov zúčastnených rozdelili riziká do štyroch skupín podľa

závažnosti a možnosti predchádzať im (pozri Tab. 1). Pozrime sa teraz na jednotlivé skupiny rizík bližšie.

		1	2
	Vysoká	Záujem zákazníka	Rozsah a požiadavky
Závažnosť rizika		4	3
	Nízka	Prostredie	Vývoj
		Nízka	Vysoká
		Úroveň kontrolovateľnosti rizika	

Tab. 1. Kategorizácia rizík do 4 kvadrantov podľa [2].

Záujem zákazníka

Najväčšie obavy zúčastnených expertov sa týkali záujmu zákazníka a používateľov o vyvíjaný produkt. Dôvodom je náročnosť presvedčiť zákazníka, aby prejavil viac záujmu. Autori štúdie odporúčajú predchádzať rizikám z tejto kategórie budovaním dôvery zákazníka voči softvérovej spoločnosti, napr. dodržiavaním vzájomných dohôd, a zohľadnením očakávaní koncových užívateľov.

Na úrovni školských projektov považujem riadenie rizík z tohto kvadrantu tiež za veľmi ťažké. Študenti, ktorí sa usilujú o čo najlepšiu známku, musia v prvom rade sledovať záujmy učiteľa. Považujem za nemožné donútiť učiteľa, aby sa viac zaujímal o softvérový produkt. Vedúci projektu má obmedzený čas, ktorý môže venovať konkrétnemu projektu. Preto by som sa usiloval využiť čas na konzultácie čo najefektívnejšie.

Vývojári by podľa môjho názoru nemali zabúdať na koncových užívateľov. Bez zohľadnenia ich potrieb a ponúknutím prívetivého užívateľského rozhrania síce môžu za projekt dostať dobrú známku, ale z pohľadu použiteľnosti produktu veľmi strácajú. Vždy ma mrzí, keď sa študenti venujú jeden či viac semestrov vývoju produktu pre koncové použitie, ktorý nakoniec nikto nevie alebo nechce používať. Preto považujem za potrebné oboznámiť sa s predstavami užívateľov o navrhovanom systéme.

Rozsah a požiadavky

2. kvadrant obsahuje riziká nesprávneho pochopenia požiadaviek a neschopnosti zohľadniť zmeny požiadaviek v priebehu vývoja.

Autori štúdie ponúkajú viacero riešení ako si s nimi poradiť:

- Výber metodológie vývoja, ktorá umožní dodatočné podrobnejšie špecifikovanie požiadaviek a ľahšie zapracovávanie zmien

- Rozdelenie dohodnutej funkcionality na esenciálnu, bez ktorej produkt nemá zmysel, a bonusovú, ktorá produkt obohatí, ale nie je nevyhnutná pre jeho fungovanie v praxi.
- Špecifikovanie, čo produkt nebude zahŕňať, pretože každá funkcionality navyše zvyšuje zložitosť systému, náklady na jeho vývoj a môže dokonca ohroziť aj dodržanie časového plánu a tým aj oneskoriť čas dodania.
- Oboznámenie zákazníka i koncových užívateľov o dopade zmien rozsahu projektu na rozvrh a náklady projektu.
- Zaistenie, že užívatelia a nie programátori smerujú vývoj. Preto je dôležité upovedomiť budúcich používateľov o ich dôležitej roli pri vývoji.

Myslím si, že (nielen) v školských projektov je kľúčové zistiť, čo zákazník chce. Tomu môže výrazne napomôcť vytvorenie biznis modelu a preskúmanie existujúcich biznis procesov problémovej oblasti. To podľa mňa vedie k hlbšiemu pochopeniu problematiky a je predpokladom pre hľadanie riešenia. Nasledujúci dialóg so zákazníkom by mal vyjasniť, ktorá biznis funkcionality sa bude automatizovať prostredníctvom plánovaného systému a čo systém riešiť nebude.

Myslím si, že voľbu metodológie vývoja by si mali študenti starostlivo zvážiť. Výber nevhodnej metodológie sa ukázalo ako riziko č. 1 v prieskume z roku 2004, ktorý skúmal riziká 720 softvérových projektov riadených skúsenými manažérmi zo 60 veľkých spoločností [4]. Vhodnosť použitia metodológie vývoja závisí najmä od toho, do akej miery sú zákazníkovi a vývojárom jasné požiadavky na systém, podľa predpokladanej pravdepodobnosti a frekvencie zmien v špecifikácii a pod. Štruktúrovaná metodológia je vhodnejšia, keď sú dopredu jasné požadované vlastnosti produktu, zatiaľ čo napr. rýchle prototypovanie uľahčuje identifikovanie požiadaviek užívateľa. Osobne považujem za ťažké získať ucelenú predstavu o produkte. Preto by som uprednostnil evolučné prístupy k vývoju softvéru, špirálový model, prototypovanie, agilné metódy a pod.

Považujem za vhodné konzultovať ako súčasný stav tak aj plán vývoja softvérového produktu s vedúcim projektu, aby sa predišlo odovzdaniu programu, ktorý síce funguje aj dobre vyzerá, ale rieši úplne niečo iné, ako zákazník chcel. Školské prostredie neumožňuje prítomnosť zákazníka na pracovisku, ako tento problém rieši extrémne programovanie.

Vývoj

Tento kvadrant zahŕňa všetky riziká spojené so samotným vývojom systému. Spomeniem najčastejšie problémy z tejto oblasti:

- Nedostatok vývojárov alebo ich nedostatočné schopnosti vzhľadom na potreby projektu
- Výber nevhodnej metodológie vývoja či jej nedostatočné uplatnenie
- Nepresné časové odhady

- Nejasné rozdelenie rolí a úloh pri vývoji

Odporúčania autorov štúdie zahŕňajú:

- Dôsledné dodržanie vývojového procesu a zvolenej metodológie vývoja
- Zmiernenie zložitosti systému jeho rozdeľovaním zhora nadol až po zvládnuteľné časti
- Jasné rozdelenie rolí a zodpovedností v tíme
- Vytvorenie rizikových plánov pre prípad krátkodobého i dlhodobého nedostatku ľudských zdrojov a príchodu nových technológií.

Túto skupinu rizík manažéri označili za menej závažnú, pretože cítili, že ju majú pod kontrolou. Toto však neplatí pre študentov vysokých škôl, ktorí sa len učia ako riadiť softvérový projekt. Ďalšie výskumy [3] tiež preukázali, že nezvládnutie riadenia vývoja vedie k niekoľkonásobnému zvýšeniu možných škôd u rizík z ostatných kvadrantov. Preto si myslím, že práve týmto rizikám by študenti mali venovať najväčšiu pozornosť. Zvládnutie zručností riadiť riziká vývoja už počas štúdia, keď ešte takpovediac „o nič nejde“, študenti podľa mňa veľmi ocenia, keď po skončení školy začnú zohrávať aktívnu úlohu pri riadení komerčného softvérového projektu.

Zdá sa mi, že v školských podmienkach - pri veľmi malom tíme (do 5 ľudí) a nemožnosti doplniť ho o ďalších členov, môže byť aj krátkodobý výpadok jedného člena tímu kritický pre dodržanie termínov. Je takmer nemožné vyhnúť sa situáciám, keď kľúčový člen tímu ochorie alebo sa dočasne nemôže venovať projektu, napr. z dôvodu pracovných povinností voči zamestnávateľovi. Preto považujem za dobré ak použité technológie a nástroje ovládajú aspoň dvaja tímoví kolegovia.

Dobré časové odhady podľa mňa vyžadujú určité skúsenosti z minulých podobných projektov. Riadenie projektov priveľmi odlišných od tých, s ktorými má manažér skúsenosti, prináša zo sebou veľa rizík. Najjednoduchším riešením je nechať si pre prípadné časové sklzy v rozvrhu dostatočnú časovú rezervu najmä pre činnosti na kritickej ceste (tie, ktorých oneskorením sa oneskorí celý projekt).

Podľa môjho názoru môže využitie príležitosti príchodu novej technológie výrazne uľahčiť vývoj. Na druhej strane ho môže aj výrazne sťažiť, keď sa podcenia riziká s tým spojené, najmä nedostatok ľudí, ktorí ju poznajú, jej nedostatočná znalosť, a prítomnosť neodhalených chýb či obmedzení v samotnej technológii.

Prostredie

Posledný kvadrant zahŕňa riziká z prostredia projektu vo vnútri i mimo organizácie. Typickými zástupcami z tejto oblasti sú zmeny na manažérskych postoch a konflikty medzi oddeleniami klientskej firmy, ktoré majú produkt používať. Tieto riziká patria medzi najťažšie riaditeľné a nevyskytujú sa často. Z týchto dôvodov odporúčam nevenovať im väčšiu pozornosť pri analýze rizík na úrovni školských projektov.

Záver

Riadenie projektu nie je jednoduché. Ak chce projektový manažér zvýšiť pravdepodobnosť dodržania plánu a predísť komplikáciám či dokonca úplnému zlyhaniu, mal venovať patričnú pozornosť aj riadeniu rizík.

V eseji som sa zameral najmä na identifikáciu hlavných rizík komerčných projektov a možnosti, ako im predchádzať. Prezentoval som svoj pohľad na to, ako by sa z nich mohli poučiť študenti pri školských projektoch.

Myslím si, že študenti by mali najviac energie venovať riadeniu rizík, nad ktorými majú najväčšiu kontrolu. Ide najmä o riziká týkajúce sa vývoja softvérového systému.

Použitá literatúra

1. Project Management Institute: *A Guide to the Project Management Body of Knowledge*. USA. Philadelphia, 1996.
2. Keil, M., Cule, P.E., Lyytinen, K., Schmidt, R.C.: *A framework for identifying software project risks*. Communications of the ACM, Vol. 41, No. 11 (1998), 76–83.
3. Wallace, L., Keil, M.: *Software project risks and their effect on outcomes*. Communications of the ACM, Vol. 47, No. 4 (2004), 68-73.
4. Tiwana, A., Keil, M.: *The One-Minute Risk Assessment Tool*. Communications of the ACM, Vol. 47, No. 11 (2004), 73–77.
5. Boehm, B., Port, D.: *Educating software engineering students to manage risk*. Proceedings of the 23rd International Conference on Software Engineering ICSE '01. Canada. Toronto, 2001. 591–600.
6. Johnson, J.: *Chaos in the New Millennium: The Ghost of Christmas Future*. The Standish Group, West Yarmouth, MA. 2000.

Annotation

School and commercial software project risks

Both school and commercial software project often suffer total failure, cost or schedule overruns, or deliver less functionality than promised. The root can be found in inability to recognize or in undervaluing possible occurrences of unplanned situations – risks. This ignorance leads to considerable money and reputation loss. It is also the cause, why students get worse marks. It points out to the importance of appropriate software project risk management not only in the business sphere, but also at university. Essay compares an academic environment with the environment of large enterprises. It examines greatest commercial project risks from the view of senior IT managers and possibilities to mitigate them. The purpose of this paper is to make use of acquired knowledge to improve quality of school projects and reduce students' amount of stress.