

# Plánovanie rizík – poistenie zo všetkých pohľadov

IVAN TOMOVIČ

*Slovenská technická univerzita  
Fakulta informatiky a informačných technológií  
Ilkovičova 3, 842 16 Bratislava  
tomovic[.]i[zavináč]centrum[.]cz*

**Abstrakt.** Dnes, keď je pri vývoji softvéru dôležitá každá sekunda a klient chce vidieť výsledky okamžite, na chyby nie je priestor. Tlak prichádza z každej strany a prináša riziká, ktoré ohrozujú projekt. Riziko je často chápané ako nebezpečenstvo vzniku určitého druhu straty. Manažment rizík a príprava vhodných bezpečnostných riešení je komplexným problémom. Analýza a riadenie rizík pomáha predchádzať alebo aspoň minimalizovať situácie, ktoré by mohli ohroziť časový plán alebo čo je horšie, celý softvérový projekt. V práci sa zaoberám hlavne tým, ako odstrániť príčiny vzniku rizikových udalostí alebo aspoň obmedziť ich negatívne dôsledky na softvérový projekt. Ďalej sú tu rozobrané rôzne príčiny vzniku chýb a rizík, ktoré najčastejšie zapríčiňujú neúspech projektu a ich rozdelenie do skupín, ako napríklad predvídateľné, teda aj ovplyvniteľné a neovplyvniteľné a čo vlastne riziko znamená. Nakoniec sú spomenuté rôzne osvedčené modely vývoja softvéru.

## Čo je to riziko a jeho príčiny

Podľa autorov knihy [5] je riziko historický výraz pochádzajúci údajne zo 17. storočia, kedy sa objavil v súvislosti s lodnou plavbou. Výraz „risico“ pochádza z taliančiny a označoval úskalie, ktorému sa museli plavci vyhnúť. Následne sa tým vyjadrovalo „vystavenie nepriaznivým okolnostiam“. V starších encyklopédiách nájdeme pod týmto heslom vysvetlenie, že sa jedná o odvahu, či nebezpečenstvo, prípadne že „riskovať“ znamená odvážiť sa niečoho. Až neskôr sa objavuje i význam v zmysle novej straty. Dnes už vieme, že nebezpečenstvo predstavuje niečo iné a v teórii rizika súvisí s hrozbou. Podľa dnešných výkladov sa rizikom obecné rozumie nebezpečenstvo vzniku škody, poškodenia, straty či zničenia, prípadne neúspechu. Neexistuje jedna obecné uznávaná definícia a pojem riziko je definovaný rôzne [5]:

1. Pravdepodobnosť, či možnosť vzniku straty, neúspechu.
2. Variabilita možných výsledkov alebo neistota ich dosiahnutia.

*Manažment projektov softvérových a informačných systémov, október 2008, s. 1-10.*

3. Odlišnosť skutočných a očakávaných výsledkov.
4. Pravdepodobnosť akéhokoľvek výsledku odlišného od výsledku očakávaného.
5. Situácia, kedy kvantitatívny rozsah určitého javu podlieha istému rozdeleniu pravdepodobnosti.
6. Nebezpečenstvo negatívnej odchýlky od cieľa (tzv. čisté riziko).
7. Nebezpečenstvo chybného rozhodnutia.
8. Možnosť vzniku straty alebo zisku (tzv. špekulatívne riziko).
9. Neurčitosť spojená s vývojom hodnoty aktíva (tzv. investičné riziko).
10. Stredná hodnota stratovej funkcie.
11. Možnosť, že špecifická hrozba využije špecifickú zraniteľnosť systému.

Z hľadiska problematiky riadenia rizík bude užitočné vychádzať z chápania rizika ako možnosti, že s určitou pravdepodobnosťou dôjde k udalosti, ktorá sa líši od predpokladaného stavu, či vývoja. Riziko by nemalo byť znižované, respektíve redukované na obyčajnú pravdepodobnosť, lebo zahrňuje ako samotnú pravdepodobnosť, tak aj kvantitatívny rozsah danej udalosti. S rizikom sú tesne prepojené dva pojmy [5]:

- Pojem neurčitého výsledku, o ktorom sa implicitne uvažuje vo všetkých definíciách rizika: výsledok musí byť neistý. Ak máme hovoriť o riziku, musia existovať aspoň dve varianty riešenia. Keď s istotou vieme, že dôjde k strate, nedá sa hovoriť o riziku. Výsledok je tu istý a riziko neexistuje. (Riziko je späté s rozhodnutím, kedy a do akého základného prostriedku investovať.)
- Aspoň jeden z možných výsledkov je nežiaduci. Vo všeobecnom zmysle slova môže ísť o stratu, keď istá časť projektu je stratená.

Sú rôzne príčiny vzniku rizík:

- príčiny predvídateľné a ovplyvniteľné – veľkosť, rozsah a komplexnosť projektu, projektová a podniková kultúra, kvalifikácia, termíny, špecifikácia úloh, osobné vplyvy, motivácia, nadriadenosť projektových cieľov, finančné zázemie, atď.
- príčiny neovplyvniteľné – politické a podmienky, technický pokrok, dostupnosť zdrojov, odberateľské správanie, mentalita, atď.

Ak je dostupné dostatočné množstvo relevantných informácií, ktoré umožňujú presne odhadovať pravdepodobnosť a stratu, potom na odhad dôležitosti, resp. závažnosti rizika, je použitý klasický prístup pomocou tzv. určenia vplyvu rizika na očakávanú škodu podľa vzorca [1]:

Očakávaná škoda = pravdepodobnosť nastania udalosti \* škoda pri vzniku

## Čo je to projekt a jeho riadenie

Myslím si, že projekt je riadiace prostredie, ktoré je vytvorené, aby zabezpečilo dodanie špecifikovaného produktu. Projekt musí mať jasne definovaný začiatok, koniec a ako má vypadáť finálny produkt. Nie každá činnosť sa dá označiť ako projekt, napríklad prevádzka informačného systému spravidla nie je projektom. V rámci riadenia projektu treba venovať pozornosť najmä tomuto okruhu otázok:

- Organizácia – určenie práv a povinností
- Plánovanie, kontrola postupu a kvality prác
- Práca s ľuďmi – je potrebné zostaviť vhodný tím ľudí pre daný projekt
- Riadenie a zavádzanie zmien v projekte

Vecí, ktoré treba sledovať, je rozhodne viac, ale podľa mňa tieto body sú najdôležitejšie. Z týchto bodov by som sa rád vyjadril k bodu číslo 3, lebo si myslím, že práca s ľuďmi je ťažkou a dôležitou, ak nie najdôležitejšou činnosťou. Napríklad medzi programátormi je bežne až rádový rozdiel v ich produktivite. Takže z tohto vyplýva, že vyhľadaním špičkových odborníkov, ich správnym zaškolením a motiváciou je možné dosiahnuť až rádové zvýšenie produktivity pri vývoji softvéru, čo už nie je zanedbateľná časť.

Myslím si, že motivácia veľmi dôležitý faktor a slúži na to, aby ľudia ochotne robili to, čo v skutočnosti aj tak urobiť musia. Bežne, keď človek vykonáva nejakú činnosť, ktorú musí robiť, tak nepracuje na „plné obrátky“, ale využíva iba určitú časť svojej kapacity (tzv. „potter line“) a práve motiváciou sa jeho účinnosť zvyšuje. Frederick Herzberg uvádza dva druhy motivačných prvkov:

1. Pozitívne prvky, ktoré dvíhajú produktivitu nad „potter line“, čo je 40-60% kapacity človeka. (Ak sú prítomné, človek je schopný pracovať viac, ako „býva zvykom“.)
  - možnosť osobného rastu vykonaním danej práce
  - uspokojenie z dosiahnutého výsledku
  - práca ako výzva naplno využiť svoje schopnosti
  - pocit zodpovednosti
  - záujem vedenia, ocenenie dosiahnutých výsledkov
2. Negatívne prvky, ktoré, ak nie sú prítomné, spôsobujú pokles produktivity pod „potter line“. Rozdiel oproti pozitívnym motivačným faktorom je ten, že aj maximálnym uspokojovaním týchto potrieb sa obvykle nedá dosiahnuť dlhodobé prekročenie „potter line“.
  - nevhodné pracovné prostredie (nedostatočný priestor, nevhodné osvetlenie, teplota v miestnosti, nástroje – hardvérové, softvérové, ...)
  - nedostatočná bezpečnosť

- neuspokojujúce postavenie v rámci spoločnosti
- zlé vzťahy v rámci tímu
- nedostatočné finančné ohodnotenie

## Ciele analýzy rizík

Často je plán projektu založený na predpoklade, že všetko pôjde tak, ako má a nič nepredvídaného sa nestane. Bohužiaľ, toto je málokedy pravda, a preto je potrebné možným neočakávaným udalostiam vopred venovať pozornosť. Tomuto sa hovorí analýza rizika. Inými slovami, ide o prevedenie rizika na zvládnuteľný problém. Cieľom analýzy rizík je dopredu odhadnúť možné ohrozenia priebehu projektu a pripraviť si reakcie na tieto ohrozenia. Pred začatím riešenia projektu je dobré zodpovedať si tieto otázky:

- termíny – stihneme dodať aplikáciu včas?
- funkcie – zvládneme implementovať všetko, čo si zákazník požadoval?
- náklady – nebudeme mať vyššie náklady, ako je cena?
- legislatíva – ktoré zákony a predpisy môžu obmedziť či ohroziť dokončenie a dodanie aplikácie?
- konkurencia – nedodá konkurencia podobný produkt skôr a lacnejšie?
- hardvér – budeme vedieť náš produkt prevádzkovať na špecifikovanom počítačovom vybavení u zákazníka?

## Extrémne programovanie

Asi najznámejším predstaviteľom z metodík, ktoré sa radia k „agilným“ je extrémne programovanie (xProgramming alebo aj XP). Za pôvodcu myšlienky extrémneho programovania je považovaný Kent Beck.

Bruce Eckel vo svojej knihe [2] píše, že extrémne programovanie dáva návod na to, ako proces tvorby softvéru nielen urýchliť a skvalitniť výsledný produkt, ale aj ako zabezpečiť to, aby sa znížil počet neúspešných projektov a taktiež dáva návody na to, ako má tím vývojárov – programátorov pracovať, ako čo najskôr docieľiť dodanie „medziproduktu“ na testovanie používateľom, ako písať vlastný kód (programovanie vo dvojiciach) a zabezpečiť jeho „kolektívne vlastníctvo“ a podobne. Súčasťou tímu musí byť zástupca budúceho používateľa.

Myslím si, že v skratke môžeme o extrémnom programovaní povedať, že je to taký prístup k tvorbe počítačových programov, pri ktorom sa využíva jednoduchosť, komunikatívnosť a spätná väzba. Je určený pre malé tímy, ktoré potrebujú vyvinúť softvér rýchlo a najmä v podmienkach meniacich sa požiadaviek.

Podľa B. Eckela je základom extrémneho programovania dvanásť odporúčaní. Uvediem ich stručnú charakteristiku:

- Plánovanie - v procese plánovania sa zadávateľovi umožňuje ohodnotiť požadované vlastnosti (črty) tvoreného programového diela z obchodného hľadiska. Programátori robia odhad ceny, ktorou sa zabezpečenie tejto vlastnosti bude podieľať na celkovej cene za vytvorenie diela. To umožňuje stanoviť priority a rozhodnúť, čo je skutočne potrebné vyriešiť a čo je druhoradé. Takémuto procesu plánovania sa tiež hovorí plánovacia hra. Význam takéhoto plánovania je v tom, že umožňuje efektívne vedenie projektu k úspechu.
- Rýchly výstup - programátorský tím zabezpečí prvý funkčný vzor tvoreného systému v čo najkratšom čase a v krátkych cykloch robí jeho rozširovanie so zohľadnenými pripomienkami používateľa.
- Metafora - programátorský tím používa spoločné názvoslovie a spoločný opis systému. Je to základ efektívnej komunikácie v tíme.
- Jednoduchosť – výsledný program by mal byť čo najjednoduchší, ale pritom musí spĺňať aktuálne požiadavky. Mal by obsahovať čo možno najmenej kódu určeného "pre budúcnosť". Namiesto toho by mala byť pozornosť venovaná jeho "obchodnej hodnote".
- Testovanie – programátorský tím po celý čas vývoja venuje zvláštnu pozornosť overovaniu vytvoreného kódu. Testy sa pripravujú dokonca skôr než vlastný program. Používatelia potvrdzujú opodstatnenosť testov, lebo im zaručujú, že ich požiadavky budú splnené.
- Prebudovávanie – počas celého vývoja sa programové riešenie vylepšuje. Dosahuje sa to "vyčisťovaním" programu – odstraňujú sa duplicity, zjednodušuje sa riešenie, ale zachováva sa kompletnosť. Využíva sa k tomu nepretržitá komunikácia v rámci riešiteľského tímu, ale aj komunikácia s používateľom.
- Párové programovanie – programátori píšu kódy programov v dvojiciach pri jednom stroji. Mnohé experimenty potvrdili, že v dvojiciach sa vytvára lepší softvér a rýchlejšie, než v prípade samostatnej práce jednotlivých programátorov.
- Kolektívne vlastníctvo – všetky zdrojové kódy patria všetkým programátorom. To umožňuje tímu rýchlo napredovať v riešení, pretože je možné úpravy a zmeny robiť bez zbytočných priesporov.
- Neustála integrácia – softvérový produkt sa kompiluje a spája veľakrát počas jedného dňa. Umožňuje sa tým udržať v strehu všetkých programátorov a rýchle napredovať v riešení.

- 40 hodinový pracovný týždeň – unavení programátori robia viac chýb. Programátorský tím nesmie pracovať extenzívne a v nadčasoch. Programátori sa musia udržiavať svieži, zdraví a efektívni. Taký bude potom aj celý tím.
- Strana používateľa – projekt je vedený určeným zástupcom, ktorý je oprávnený stanovovať požiadavky, priority a odpovedá na otázky kladené programátormi. Zlepšuje to komunikáciu s používateľom a redukuje papierovú dokumentáciu, ktorá je často najnákladnejšou zložkou softvérového projektu.
- Štandardy kódovania – pre efektívnu prácu v tíme, kde sa pracuje v dvojiciach a je spoločné vlastníctvo kódu, je dôležité písať zdrojový kód rovnakým spôsobom a podľa rovnakých zásad (štábna kultúra)

## PRINCE2

PRINCE2 (Projekty v kontrolovaných prostrediach) je procesná metóda pre efektívne riadenie projektov. PRINCE2 je v podstate štandard používaný vo veľkej miere vládou Spojeného kráľovstva a je široko uznávaný v súkromnom sektore, v Spojenom kráľovstve ako aj medzinárodne [4].

### Stručný opis metodiky

Vznikla na základe potreby britskej vlády pre metodiku projektového riadenia a je ňou aj dodnes používaná. Zároveň PRINCE2 používajú aj iné vlády EÚ (napr. Dánska), NATO, OSN, ale aj veľké korporáčne organizácie ako napríklad Deutsche Post, DHL, Fujitsu Siemens, Logica, Philips Electronics, Sun Microsystems. Certifikáty PRINCE2 už boli vydané viac ako 250.000 projektovým vedúcim na celom svete [3].

### Hlavné charakteristiky projektu

Projekt je riadený obchodným prípadom, ktorý sa pravidelne zhodnocuje. Je jasne definovaná organizačná štruktúra a je založený na procesoch, komponentoch, technikách. Ďalej sú tu kontrolovateľné etapy a plánovanie, ktoré je zamerané na výstupy/produkty.

### Procesy

Projekt PRINCE2™ je riadený ôsmimi procesmi a ich podprocesmi, ktoré siahajú od začatia projektu a vytvorenia Obchodného prípadu, cez procesy plánovania a riadenia projektu až do ukončenia projektu [3]:

- Zahájenie projektu - projekt je zahájený po obdržaní mandátu projektu s jednoduchým popisom finálneho produktu. Na začiatku zahájenia sa zvolí projektový vedúci a predseda projektového výboru a potom celá organizačná štruktúra projektu. Pre každý projekt je nutné zohľadniť, či je projekt

realizovateľný a stojí za vynaložené úsilie. Je nutné nadefinovať spôsob, akým sa ku projektu pristúpi

- Nastavenie projektu – na začiatku procesu je dôležité od zákazníka zistiť očakávania kvality a na základe nich popísať procesy vedúce k zaisteniu kvality. Ďalej je nutné identifikovať koncové produkty a naplánovať ich dodanie. Následne projektový vedúci spresní obchodný prípad, zoznam rizík, založí kontrolné mechanizmy projektu a dokumenty podáva na schválenie Projektovému výboru, ktorý autorizuje projekt
- Smerovanie projektu - siaha od začiatku až po koniec projektu. Projektový výbor smeruje projekt a schvaľuje všetky zásadné rozhodnutia na projekte, ktoré nie sú v rámci kompetencie projektového vedúceho
- Plánovanie - skladá sa z definovania a analýzy produktov, závislostí, termínovania a analýzy rizík, ktoré existujú v pláne. Po prejdení všetkých uvedených podprocesov plánovania je vytvorený plán etapy, celého projektu, náhradný plán či plán samostatnej úlohy
- Dohľad nad etapou - je závislý na rôznych iných procesoch, v ktorom projektový vedúci zadáva úlohy vedúcemu tímu, zisťuje pokrok v ich spĺňaní, zachytáva a hodnotí body projektu a následne ich rieši. Projektový vedúci posieľa správy o stave projektu
- Dohľad nad dodaním produktu - vedúci tímu na začiatku procesu prijme úlohu od projektového vedúceho s jasne definovanými požiadavkami a popisom produktu a očakávanej kvality. Keď danú úlohu dokončí, doručí ju projektovému vedúcemu, ktorý mu potvrdí prevzatie jej vyhotovenia
- Riadenie prechodu medzi etapami - zvyčajne začína pri konci aktuálnej etapy počas procesu dohľadu nad etapou. V prípade, že jedna etapa končí, treba naplánovať a začať novú etapu. Druhý prípad je, že projektový výbor posúdi otvorený bod projektu za natoľko závažný, že je nutné vypracovať plán realizácie výnimky. Vtedy sa etapa musí naplánovať, aktualizovať projektový plán, obchodný prípad a prehľad rizík. Etapa musí byť nakoniec schválená projektovým výborom
- Ukončenie projektu - projektový vedúci podá projektovému výboru návrh na ukončenie projektu a výbor ho musí schváliť. Projektový vedúci po ukončení projektu odovzdá klientovi odporúčenie následných krokov a nakoniec spolu s celým projektovým tímom zhodnotia úspešnosť a priebeh projektu

### **Komponenty**

V projektovom riadení sú komponenty rozhodujúcimi článkami. PRINCE2 má osem komponentov [3]:

- Obchodný prípad - je riadiacou silou projektu. Obsahuje hlavné dôvody pre uskutočnenie projektu a výhody, ktoré projekt prinesie

- Organizácia - Sú tu zastúpené 3 kľúčové zložky:
- biznis - projektový vedúci,
- používatelia - hlavný používateľ
- organizácia zodpovedná za dodávku - hlavný dodávateľ
- Plány - Určujú spôsob, akým je možné dosiahnuť stanovené ciele.
- Kontroly - Zaisťujú, aby stav projektu korešpondoval s obchodným prípadom a aby boli dodané požadované produkty v stanovenej kvalite a dohodnutom čase a cene
- Riadenie rizík - Snaží sa riziká identifikovať a proaktívne na ne reagovať tak, aby projekt mal možnosť uspieť
- Kvalita v prostredí projektu - v procese nastavenia projektu si projektový vedúci a hlavný používateľ presne zadefinujú kvalitu a počas trvania projektu kontrolujú jej napĺňovanie
- Riadenie konfigurácií - identifikuje, sleduje a ochraňuje produkty daného projektu
- Kontrola zmien - zmeny je v projektovom prostredí nutné kontrolovať pomocou zisťovania ich dopadu na projekt, dôležitosti a ceny

### **Techniky**

PRINCE2 definuje tri techniky [3]:

- Produktovo-orientované plánovanie - je ho možné použiť v akomkoľvek druhu plánovania na projekte. Počas plánovania vznikajú štyri produkty:
  - Popis hotového produktu na projekte
  - Hierarchická štruktúra produktu
  - Popisy každého z produktov
  - Diagram produktových tokov
- Kontrola zmien - ku zmenám môže na projekte dochádzať rôznymi spôsobmi. Zmeny produktu musia byť schválené na zodpovednej úrovni a následne musí byť upravený popis hotového produktu
- Revízie kvality - revízia kvality má za úlohu overiť, či produkt spĺňa kritériá kvality. V kontexte IT typicky hovoríme o testovaní.



## Záver

Zabezpečenie kvality všetkých produktov (teda nielen kvality výsledného systému) je dôležité, pretože náklady súvisiace s opravou chyby v systéme sú tým vyššie, čím neskôr sa chyba odhalí. Ak vznikne chyba v špecifikácii a odhalí sa počas testovania, náklady na jej odstránenie sú rádovo vyššie ako náklady na odstránenie tejto istej chyby počas kontroly špecifikácie požiadaviek. Preto vznikli a stále vznikajú rôzne metodiky na dosiahnutie toho, aby bol projekt úspešne dokončený.

Projekty sa od seba líšia zameraním, ale aj svojou rozsiahlosťou. Pre rôzne typy projektov je vhodná iná metodika na dosiahnutie úspechu a preto je potrebné sa rozhodnúť pre tú správnu. Napríklad z vyššie spomenutých metodík je extrémne programovanie vhodné najmä na menšie projekty s menším počtom ľudí v tíme (5-15), kde je potrebné vývoj ukončiť v čo najkratšom čase a požiadavky sa menia „zo dňa na deň“. Podľa prieskumu firmy IBM z roku 2000 až vyše 50% vývojárov odpovedalo, že XP je dobré a iba necelých 10% má s ním zlé skúsenosti.

PRINCE2 je flexibilná metodika a tým pádom je prispôsobiteľná na akýkoľvek typ projektu, ale jej nevýhodou pre malé projekty je jej obsiahlosť (zahŕňa všetky dôležité súčasti projektového riadenia). Je založená na procesoch a zahŕňa celý životný cyklus projektu, jasne definuje roly a štruktúru v projektovom riadení, stanovuje adekvátne úrovne zodpovedností. Merateľné produkty a zdroje sú jasne definované.

## Použitá literatúra

1. Bieliková, M.: Softvérové inžinierstvo. Princípy a manažment. Vydavateľstvo STU, Bratislava 2000.
2. Eckel B., Allison Ch.: Myslíme v jazyku C++, Vydavateľstvo Grada, 2006
3. Harris P.: PRINCE2 Planning & Control Using Microsoft Project, EastwoodHarris Pty Ltd., 2005
4. S&T – Projektový manažment [online]. [cit. 2008-10-14]. Dostupný z WWW: <[http://www.snt.sk/Content.Node/riesenia\\_a\\_sluzby/globalne\\_sluzby/project\\_management/25857.sk.php](http://www.snt.sk/Content.Node/riesenia_a_sluzby/globalne_sluzby/project_management/25857.sk.php)>
5. Smejkal V., Rais K.: Řízení rizik ve firmách a jiných organizacích (Druhé, aktualizované a rozšířené vydání), Vydavateľstvo Grada, 2006

## Annotation

### *Risk management- secure your project from every side*

Today, when every second of time is important and client wants to see the results immediately, there is no place for mistakes anymore. There is a pressure from every side, bringing risks which endanger the project. Risk is often seen as a danger of some kind of loss. Risk management as well as the preparation of adequate safety solutions is a complex problem. Analysis and risk management helps to prevent or at least minimise situations which could endanger the time schedule, or worse so the software project itself. The paper deals with risk

occasions' elimination or at least minimisation of their negative impact on the software project. Moreover, different mistakes and risks' sources which most frequently cause a project failure are analyzed and divided into groups, e.g. foreseeable and controllable or the uncontrollable on the other side as well is what the meaning of a risk itself. Finally, different approved models of software development are mentioned.