

# Manažment rizík v softvérovom projekte

ANDREJ JANŽO

*Slovenská technická univerzita  
Fakulta informatiky a informačných technológií  
Ilkovičova 3, 842 16 Bratislava  
janzo00@student.fiit.stuba.sk*

**Abstrakt.** Analýza a manažment rizík v softvérových projektoch je stále na okraji záujmu manažmentu. Vysoký pomer neúspešných projektov dokazuje, že manažment by mal tento nezodpovedný prístup k rizikám vo vlastnom záujme prehodnotiť. Táto práca sa venuje manažérskemu systému ProRisk, ktorý popisuje postup pri manažmente rizík v softvérových projektoch a pokrýva celý životný cyklus projektu. Práca prechádza jednotlivými fázami systému pri tvorbe modelu rizík projektu, od identifikácie rizík, cez konštrukciu modelu, jeho kalibráciu, výpočet rizikových hodnôt, až po vytvorenie akčných plánov na zamedzenie výskytu rizík a monitorovací a udržiavací proces systému. Následne sa venuje zhodnoteniu systému a jeho výhod a nevýhod pre manažérov a organizáciu.

## Úvod

V súčasnej dobe nazývanej aj informačnej, sa vyvíja veľké množstvo softvérových produktov. Napriek tomu, že sa softvér vyvíja už niekoľko desaťročí, manažment sa rizikami nezaobera dostatočne. Podľa vykonaných prieskumov len jedna šestina všetkých softvérových projektov bola dokončená načas a v rámci plánovaného rozpočtu, jedna tretina projektov bola zrušená a vyše polovica bola odmietnutá [6]. Štúdia [3] ukázala, že 35 % zrušených projektov bolo vo fáze implementácie. Tento veľmi zlý pomer neúspešnosti projektov bol hlavne z dôvodu, že manažment sa nezaoberal varovaniami pred rizikami [5].

Na základe týchto prieskumov je zrejmé, že práca manažmentu bola veľmi slabá v oblasti analýzy a manažmentu rizík. Dobrá analýza a manažment rizík by mala byť základnou požiadavkou na manažment v každom väčšom softvérovom projekte. Väčšina softvérových projektov sa uskutočňuje v nepredvídateľnom prostredí, v ktorom existuje veľa pascí, ktoré môžu ovplyvniť úspešný výsledok projektu. Poprojektové analýzy odhaľujú, že veľa problémov, ktoré sa vyskytli boli v skutočnosti predpovedateľné.

Projekt sa považuje za úspešný, ak spĺňa požiadavky (merané funkcionalitou, spoľahlivosťou, udržiavateľnosťou, prenositeľnosťou, efektívnosťou, integráciou a prevádzky schopnosťou), je dodaný na čas a v rámci rozpočtu [7].

Manažment rizík sa zaoberá identifikáciou, čo môže ísť zle a ako môžu tieto prípady dopadať na projekt. Formálna analýza rizík je založená na pravdepodobnostnej teórii. Väčšina softvér vyvíjajúcich projektov zahŕňa veľký počet rizikových faktorov. Na rozdiel od triviálnych malých prípadov, je extrémne ťažké spraviť detailný odhad potrebných pravdepodobností. Preto je veľmi náročná úloha vymyslieť vhodnú metodológiu, ktorá môže byť aplikovateľná na praktické problémy.

Vývoj softvéru má relatívne krátku históriu skúseností na ktorých by sa dalo stavať. V iných disciplínach, starších (špeciálne väčšinou tradičné inžinierske oblasti), majú dobre zavedenú bázu znalostí, praktické manuály a profesionálne štandardy. Softvérové inžinierstvo, ako disciplína, ešte stále len prichádza k týmto termínom s otázkami. Ako výsledok toho je stále potrebné sa učiť ako majú byť riziká modelované a riadené v softvérových projektoch.

## Riziko v softvérovom projekte

Riziko znamená možnosť utrpieť stratu, poškodenie, nevýhodu alebo zničenie. Riziko sa samozrejme týka neistoty vzniku známych prípadov, ale aj prípadov, ktoré nie sú na počiatku identifikované ako dopadajúce na projekt [4]. Manažment rizík sa preto musí vyvíjať a učiť, adaptujúc sa na nové meniace sa znalosti ako postupuje projekt.

Hodnota rizika je zvyčajne definovaná ako efekt dopadu rizikovej udalosti a pravdepodobnosť, že udalosť nastane. Môžeme ho vyjadriť nasledovným vzorcom

$$V(A) = P(A) * C(A) \tag{1}$$

kde  $P(A)$  je pravdepodobnosť, že nastane udalosť  $A$ ,  $C(A)$  vyjadruje cenu dopadu alebo efektu rizika a  $V(A)$  je hodnota rizika pre udalosť  $A$ . Hodnota rizika je interpretovaná ako priemerná očakávaná strata nastania udalosti. Z takejto formy vyjadrenia rizika môžeme intuitívne prísť na to, že hodnota rizika je vyššia ak je vyššia pravdepodobnosť alebo cena dopadu. Ako výsledok, riziká s nízkou cenou dopadu ale vysokou pravdepodobnosťou môžu mať rovnakú hodnotu, ako riziká s vysokou cenou ale nízkou pravdepodobnosťou.

Pre malý počet rizikových udalostí je tento model dostatočne jednoduchý a poskytuje užitočnú podporu pre hodnotenie rizikových udalostí. Ako rastie komplexnosť je nutné zvážiť nezávislosť udalostí a kombinovať tieto udalosti na umožnenie efektívneho manažmentu, odhadu a riadenia. Musí sa zistiť veľký počet pravdepodobnostných hodnôt alebo rozdelenie pravdepodobností od manažmentu tímu. Tiež je potrebné odhadnúť podmienené pravdepodobnosti medzi udalosťami, dané tým, že niektoré nebudú úplne nezávislé od ostatných. Takáto úloha je veľmi komplexná, ak nie nemožná pre praktické projekty.

Ak je možné spraviť niektoré predpoklady o podmienených pravdepodobnostiach medzi rizikovými faktormi, potom je možné navrhnúť niektoré funkčné stratégie.

Napríklad, kombinovaný dopad skupiny rizikových faktorov môže byť odhadnutý pomocou vzorca

$$V_I = \sum_{i \in I} w_i \cdot V_i \quad (2)$$

pre prípady, ktoré sú v podstate nezávislé, alebo

$$V_I = \prod_{i \in I} w_i \cdot V_i \quad (3)$$

pre prípady, ktoré sú v podstate navzájom vylučujúce sa, alebo

$$V_I = \max_{i \in I} \{w_i \cdot V_i\} \quad (4)$$

pre fuzzy model zjednotenia následkov udalostí. Váhový koeficient  $w$  sa berie ako rizikový efekt vyskytnutej udalosti.

### Model rizík v softvérovom projekte

Taxonómia rizík softvérových projektov bola vyvinutá Inštitútom softvérového inžinierstva (SEI – Software Engineering Institute). Táto taxonómia definuje stromovú štruktúru hierarchie rizikových oblastí, náležite klasifikovanú na definovanie klastrov rizikových faktorov. Najvyššie tri levely sú zobrazené v tabuľke 1. Štvrtý level odpovedá skupine otázok, ktoré vyšetrujú potencionálne riziko a tak definujú rizikové faktory pre model. Príspevky každého klastra rizikových faktorov projektu môžu byť odhadované akumuláciou rizikových hodnôt na vrchole stromu. Relatívne príspevky každého faktora a klastra sú opísané váhovými faktormi, ktoré merajú každý efekt zvlášť. Tieto váhové koeficienty boli odhadované z veľkého počtu prípadových štúdií a môžu poskytovať štartovací bod pre projektový rizikový model.

Root Node [Level 0]	[Level 1]	[Level 2]
Project Risk	Software Development Risk	Requirements
		Design
		Code and Unit Test
		Integration and Test
		Engineering Specialities
	Development Environment	Development Process
		Development System
		Management Process
		Management Methods
	Program Constraints	Work Environment
		Resources
		Contract
		Program Interfaces

**Tab. 1** SEI Taxonómia

Taxonómia SEI je komplexná a potrebuje detailnú analýzu projektu a organizácie prevádzky. Z tejto analýzy procesov sa získajú potrebné detailné poznatky a hodnotenia rizík.

D.W.Karolak [2] navrhol viac formálny model, podobný taxonómii SEI, ale poskytujúci viac kvantitatívny a viac detailný prístup. Tento SERIM (Software Engineering Risk Management) model ponúka manažmentu rizík štyri prepojené stromy založené na 81 rizikových faktoroch. Tie odkazujú na rizikové perspektívy, ktorého tri najvyššie úrovne sú v tabuľke 2.

Rovnako ako SEI model, štvrtá úroveň je vyjadrená sadou otázok, ktoré vyšetrujú rizikové faktory podľa pravdepodobnosti výskytu. Rizikové faktory sú zdieľané vo všetkých perspektívach, rozdiely sú len vo váhových faktoroch. SERIM model je oproti SEI modelu komplexnejší, sú to v skutočnosti štyri modely, každý zodpovedajúci inej perspektíve.

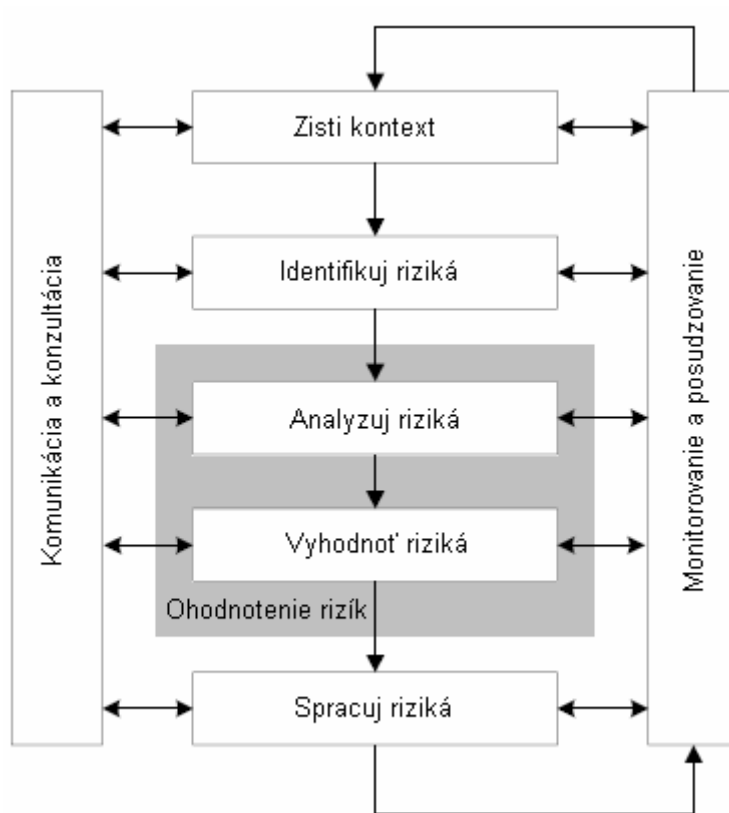
SERIM model potrebuje od používateľa (manažéra), aby poskytol odhady pravdepodobností pre každý rizikový prípad. Následne sa spočíta kombinovaná riziková hodnota pre každú úroveň stromu. Rizikový manažment je potom schopný porovnať príspevky rizík pre každú časť stromu medzi perspektívami a identifikovať kľúčové rizikové faktory.

Root Node [Level 0]	[Level 1]	[Level 2]
Risk Elements	Technical	(a) Organization
		(b) Estimation
		(c) Monitoring
		(d) Development Methodology
		(e) Tools
		(f) Risk Culture
		(g) Usability
		(h) Correctness
		(i) Reliability
		(j) Personnel
	Cost	(a) to (j) as above
	Schedule	(a) to (j) as above
Risk Categories	Process	(a) to (j) as above
	Product	(a) to (j) as above
Development Phases	Pre-requirements	(a) to (j) as above
	Requirements	(a) to (j) as above
	Design	(a) to (j) as above
	Coding	(a) to (j) as above
	Testing	(a) to (j) as above
	Delivery and Maintenance	(a) to (j) as above
Risk Activities	Identification	(a) to (j) as above
	Strategy and Planning	(a) to (j) as above
	Assessment	(a) to (j) as above
	Mitigation and Avoidance	(a) to (j) as above
	Reporting	(a) to (j) as above
	Prediction	(a) to (j) as above

Tab. 2. SERIM rizikové perspektívy

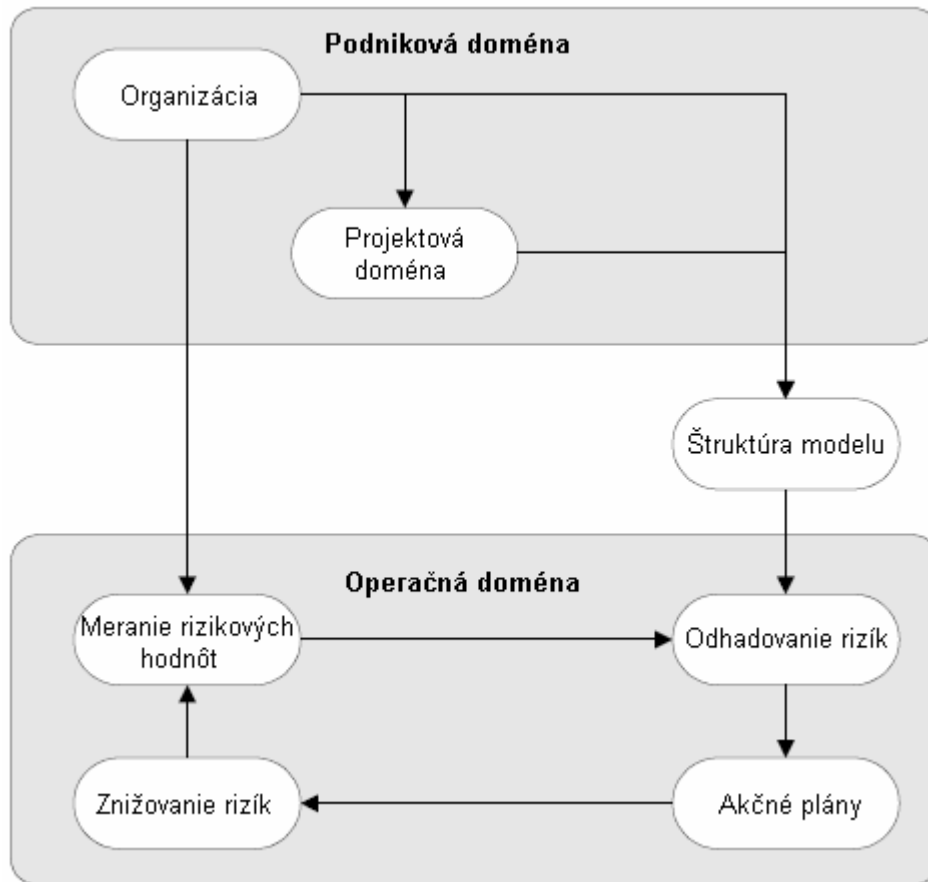
### System pre manažment rizík

Manažment rizík musí byť integrovanou súčasťou štruktúry projektového manažmentu, ak má byť efektívny. Kroky zobrazené na obrázku 1 musia byť obalené v procese, ktorý náležite identifikuje roly a zodpovednosti medzi organizáciou a projektovým tímom.



**Obr. 1.** Systém manažmentu rizík

ProRisk manažérsky systém postupuje podľa týchto krokov a rozširuje ich. Schéma tohto systému v kontexte typického softvérového projektu je na obrázku 2. Tento systém sa zameriava na primárne zložky projektu, podnikovú doménu a operačnú doménu. Podniková doména poskytuje možnosti na identifikáciu ekonomického prostredia, vedomostí a skúseností organizácie v ktorej sa projekt vyvíja. Operačná doména obsahuje modely na meranie, identifikovanie, odhadovanie, opisovanie a implementovanie plánov a obaľuje ich do spojených cyklických procesov, ktoré majú byť aplikované počas projektu.



Obr. 2. Manažérsky systém ProRisk

### Identifikácia podielnikov (stakeholders)

Podielníci v projekte sú jednotlivci, skupina ľudí alebo organizácií, ktoré majú úžitok z výsledku projektu (a dedukciou, utrpia stratu, ak projekt zlyhá). Majú jednu alebo viac perspektív a tieto perspektívy sú merané v ich vlastných jednotkách. Na efektívnu reprezentáciu ich záujmov, metodológia musí podporovať aj viac podielnikov aj rôzne spôsoby vyjadrenia záujmu.

### Identifikácia rizikových faktorov

Identifikácia podielnikov pomáha v identifikácii rizikových faktorov, ale aj tak je potrebné vedieť viac informácií. Sami podielníci, nie sú schopní vytvoriť kompletnú sadu rizikových faktorov, ktoré opíšu ich vlastnú perspektívu záujmu. Cieľom je identifikovať všetky veci, ktoré sa môžu zle vyvíjať a toto je úloha pre rizikový manažment. Na zistenie kompletnej sady rizikových faktorov je možné začať od nuly

a použiť brainstorming alebo Delphi metódu. Toto môže byť proces, ktorý pravdepodobne spotrebuje veľa času, ale v niektorých prípadoch to môže byť oprávnené. Získaním sady rizikových faktorov je rozumné začať s jedným z modelov a prispôbiť ho potrebám vlastného projektu.

### **Konštrukcia modelu stromu rizík**

Konštrukcia kompletného modelu stromu rizík je iteratívny a učiaci proces vyžadujúci zainteresovanosť všetkých podielnikov a manažmentu rizík, ktorí budú zodpovední za celkový model. Z počiatočného zoznamu rizikových faktorov, je ako prvé potrebné vytvoriť rizikové klastre. Všetky rizikové faktory v klastru by sa mali týkať podobných aspektov projektu. Kompletný strom rizík bude vytvorený ďalšou agregáciou klastrov pokiaľ nebude mať každá perspektíva jeden koreňový uzol, reprezentujúci celý projekt. Kompletný projekt môže byť reprezentovaný jedným alebo viac stromami alebo perspektívami.

### **Kalibrácia modelu**

Kalkulácia rizikovej hodnoty vyžaduje pravdepodobnosti rizikových udalostí a ich cena/dopad pre každú perspektívu ku ktorej patrí. Získavanie týchto hodnôt nie je triviálne. Kalibrácia modelu začína odhadovaním váhových faktorov. Každá riziková perspektíva musí byť spracovaná samostatne:

Cenová perspektíva: váhové faktory budú napríklad v eurách a budú vyjadrovať priemerné zvýšenie nákladov projektu, ak prípad nastane.

Rozvrhová perspektíva: váhové faktory budú v časových jednotkách (dni, týždne, atď.) a budú vyjadrovať priemerné zdržanie projektu, ak prípad nastane.

Ostatné rizikové perspektívy (napr. kvalita, reputácia) musia byť tiež merané ak sa použijú.

### **Odhadovanie pravdepodobností rizikových udalostí**

Toto je posledný krok vo fáze tvorby modelu a nutný krok pred tým ako sa použije model na čokoľvek užitočné. Každý rizikový faktor musí byť odhadnutý na jeho pravdepodobnosť výskytu. Obvykle bývajú tieto hodnoty reprezentované v rozsahu 0 až 1. Reprezentácia pomocou slov (XLow, VLow, Low, Med, High, VHigh, XHigh) môže používateľom umožniť myslieť vo viac opisných výrazoch ako číselné vyjadrenie.

### **Výpočet kombinovaných rizikových hodnôt**

ProRisk manažerský systém poskytuje viacero kombinácií modelu:

- Sumácia – kombinuje rizikové hodnoty sumáciou hodnôt pre každý klaster podľa vzorca (2)
- Súčin – kombinuje rizikové hodnoty súčinom hodnôt na najnižšej úrovni klastra podľa (3) a potom sumáciou najvyšších úrovní podľa (2)



- Maximum – založené na Murphyho zákone, ktorý hovorí, že ak sa niečo pokazí, tak to bude prípad, ktorý má najväčšiu rizikovú hodnotu, berúc tak do úvahy najhorší prípad.

Ak je model vytvorený a nakalibrovaný, môže sa použiť pri identifikácii kľúčových rizikových faktoroch.

### Vytvorenie akčných plánov

V ProRisk systéme, akčné plány poskytujú podporu pre dokumentáciu, manažment a na prehodnotenie rizikových prípadov počas vývoja projektu. Akčné plány vyžadujú:

- Delegovanie a zodpovednosť, t.j. kto je zodpovedný za ich vykonanie
- Opis, čo bolo spravené na redukciiu dopadov rizikových faktorov.
- Rozmiestnenie zdrojov na umožnenie dokončenia práce.
- Časový rozsah, opisujúci kedy má byť práca dokončená.

### Monitorovanie vývoja

Ako projekt postupuje, rizikové hodnoty sa menia z času na čas. Aby bolo možné primerane manažovať projekt, toto prechodné správanie sa musí zaznamenávať, aby bolo možné monitorovať vývoj rizikových vlastností. ProRisk manažérsky systém umožňuje časom vyvíjať model rizík, udržiavať snímky stavu modelu v časových epochách, umožňujúc prechodným zmenám byť zaznamenané a monitorované.

### Záver

Softvér vyvíjajúce projekty sú náročné na analýzu rizík, pretože obsahujú široké množstvo rizikových faktorov, ktoré sa dopredu ťažko odhadujú. Väčšina súčasne dostupných nástrojov nie je dostatočne škálovateľná na väčšie problémy alebo sú náročné na množstvo informácií potrebných na nastavenie a kalibráciu modelu.

Ponúknutý ProRisk systém môže byť aplikovaný aj na malé aj na komplexnejšie projekty. Umožňuje používateľom prispôbiť si model a špecializovať ho pre vlastné potreby. Vyžaduje však tiež detailnú analýzu organizácie a projektovej domény, čo zaberie dosť času a môže byť náročné najmä pre neskúsených, začínajúcich manažérov. Organizácia by preto pri zavádzaní nových postupov a po prijatí nového manažéra mala robiť dôsledné školenia. Na dobrý odhad pravdepodobností a kalibráciu modelu však treba aj určité skúsenosti v danej oblasti, a školenie je preto len začiatkom. Dobré výsledky v odhade rizík sa dajú robiť až po skúsenostiach z viacerých projektoch.

Dobré výsledky ProRisk systému môžu byť zaručené iba za nasledovných predpokladov:

- Rizikové faktory sú nezávislé alebo navzájom sa vylučujúce.
- Na výpočet zložených rizikových hodnôt sa použije vhodná metóda.

- Rizikové perspektívy sú vytvorené z odpovedajúcich rizikových hodnôt.
- Extrémne prípady sú z modelu vyňaté.

Problém môže byť v prvom predpoklade, keďže nie všetky riziká sú nezávislé a môžu ovplyvňovať iné riziká. S ďalšími tromi predpokladmi sa dá súhlasiť, ak použijeme zlú metódu, tak je zrejmé, že dostaneme zlé výsledky, alebo nie je možné v modeloch počítať s extrémnymi prípadmi, ktorých dôsledky by boli katastrofálne ale majú aj extrémne nízku pravdepodobnosť výskytu.

Skúseným manažérom môže ProRisk systém pri manažovaní rizík značne pomôcť, pretože pokrýva kompletný životný cyklus projektu a umožňuje analyzovať riziká a zároveň sa venovať konvenčným projektovým manažérskym aktivitám.

### 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. D. W. Karolak, Software Engineering Risk Management, *IEEE Computer Society*, 1997
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. Geoffrey G. Roy, A Risk Management Framework for Software Engineering Practice, *aswec*, p. 60, 2004 *Australian Software Engineering Conference (ASWEC'04)*
5. 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.
6. May, L.J.: Major causes of software project failures. <http://stsc.hill.af.mil/crosstalk/1998/jul/causes.asp>. 25.6.2002
7. Powell, P.L., Klein, J.H.: Risk management for information systems development. *Journal of information technology*, Vol. 11 (1996) 309-319.

### Annotation

#### *Risk management in software projects*

Risk analysis and management in software projects is still on marginal interests of management. High failure rate of projects proves, that management should reevaluate this irresponsible access to risk. This paper presents ProRisk Management framework, that describes risk management method in software projects and covers whole life-cycle of project. Paper goes through individual phases of the framework, from risk identification, through model construction, his calibration, computing risk values, till developing action plans to prevent risk occurrence and monitoring and maintaining process of the framework. Thereafter devotes to valuation of this framework and his benefits and disadvantages for managers and organization.