DEVELOPMENT OF A SOFTWARE RISK MANAGEMENT MODEL USING UNIQUE FEATURES OF A PROPOSED AUDIT COMPONENT

Ahdieh Sadat Khatavakhotan[1], Siew Hock Ow[2]

[1, 2] Department of Software Engineering, Faculty of Computer Science and Information Technology

University of Malaya, 50603, Kuala Lumpur, Malaysia

[1]Khotan@siswa.um.edu.my, [2]show@um.edu.my

ABSTRACT

One of the most exigent features of a risk is risk alteration that can exacerbate its consequences and make its management difficult. Therefore, good risk management models should be able to identify risks and monitor the changes to the risk as the project progresses. This feature is not emphasized in the current risk management models, and this has resulted in a high rate of failure in software risk management. This paper discusses the development of a software risk management model that uses features of an embedded audit componet as a verifier core. Special emphasis is on managing the risks of the risk management process which is done by remonitoring the risks and activities through the verifier core. The model includes four main phases – risk identification; measurement; assessment; and mitigation and contingency plan.

In order to evaluate the model, a six-month case study was conducted using the customer relationship management system of an industrial design company. The use of the proposed model produces the following results: more accurate risk classification (phase 1); more exact definition of the deviation rate from the established schedule (phase 2); the model adapts well to the changes to the risk factors, and makes better assessment of the consequences (phase 3); in implementing the mitigation and contingency plan, the dynamic verifier core successfully uncovers ignorable mistakes and also helps to reduce or lessen the consequences (phase 4). The proposed model has proven to be effective in reducing the unforeseen risks. This will improve the success rates of software projects.

Keywords: Software Risk Management, Risk Identification, Risk Measurement, Risk Assessment, Risk Mitigation, Dynamic Verifier Core.

1.0 INTRODUCTION AND BACKGROUND

Software Risk Management is a crucial activity in the software development process. It is different from the risk management of other products due to the non-physical nature of the deliverables. Indeed, risk management is a project within an IT project [1]. In software and IT projects, intangible assets like data, reputation, and liability are more vulnerable than physical resources such as devices and equipment. The three main penetrable

components of such projects are codes, data and documents. These components together with human resources, budgets, hardware, and scheduling constitute the vulnerable assets. Therefore, IT risk management needs special attention.

System development project failures had plagued the IT industry for years. In fact, reports have indicated that only 28% of software development projects are successful, down from previous estimates of 34% [2]. The alarming rate of IT projects failures reflects the relative unsuccessfulness in the risk management processes and highlights the pressing need to improve the current IT risk management models [3].

Another issue is the changing nature of the risks. Software engineers have had to deal with the rapid, diverse, and interdependency of the changes [4]. In this context, a lot of internal or external, controllable or uncontrollable, hidden or obvious factors are affecting the risk factors frequently and dynamically. These issues make the risk management process complicated. A reliable and practical risk management model should be able to handle all these changes. Rush and Vednere [5] stated that the complexity of IT risk management models makes it difficult for them to be applied. Castro, Gulías, and Abalde and Jorge [6] pointed that it is crucial to verify the risk management activities in every risk management model, but they did not propose any general solution. Martin [7] stressed the importance of having external supervision to verify the risk management activities. Alter and Sherer [8] conducted a comprehensive literature review on IT risk management models and found that the inapplicability of existing models results from the lack of clarity, practicality of use, incompleteness, and adaptability.

This paper discusses the development of a software risk management model that uses features of an embedded dynamic verifier core [9]. The proposed model consists of four phases, and embeds a core for identifying deviations after each phase. This will lead to a better outcome in the risk management process. The model was designed to be simple for implementation and it incorporated the good features of the current models. In the development of this model, there was special emphasis on managing the risks of the risk management process, which is done by remonitoring the risks and activities through the verifier core.

2.0 BRIEF REVIEW OF THE LITERATURE

In the last three decades, different risk management models have been proposed for evaluating and managing IT project risks. Most of the models emphasized the classification of the potential risks, identification of related risk factors, and initialization of risk factors for evaluation. Some of these models can identify, monitor, and control the risks in accordance with the software development process. Other models, however, act independently of the development process and manage the risks by classifying the potential risks, identifying the related risk factors, and initializing the risk factors for evaluation [10]. The main reference models and risk categories were proposed by Barry Boehm in 1991 [11]. He was one of the earliest IT and software risk experts to propose the risk management steps shown in Fig. 1.

Fig. 1. Software risk management steps [11]

As shown in Fig. 1, Boehm's model has two main phases, six sub-phases and their related steps, for risk management. He also presented ten important risk items (risk factors) from the operational, practical, resource, and scheduling aspects which are widely used in software researches. Table 1 shows the software risk items.

Most of the researches conducted are based on the above ten risk items. Some researchers also applied the COCOMO and the COCOMO II for their cost estimation. Most of the current risk management models, however, do not consider the risks and threats to the risk management process itself. They also ignore the changeable nature of the risk factors and the need for continuous revision of the risk management procedures.

Table 1. A prioritized list of top-ten software risk items [11]

No.	Risk Item	Risk Management Techniques
1	Personnel shortfalls	Staffing with top talent, job matching; teambuilding; morale building; cross-training; pre-scheduling key people
2	Unrealistic schedules and budgets	Detailed, multisource cost and schedule estimation; Design to cost; incremental development; software reuse; requirements scrubbing
3	Developing the wrong software functions	Organization analysis; mission analysis; ops-concept formulation; user surveys; prototyping; early users' manuals
4	Developing the wrong user interface	Task analysis; prototyping; scenarios; user characterization (functionality, style, workload)
5	Gold plating	Requirements scrubbing prototyping; cost-benefit analysis; design to cost
6	Continuing stream of requirement changes externally furnished components	High change threshold; information hiding; incremental development (defer changes to later increments)
7	Shortfalls in computer-science capabilities	Benchmarking; inspections; reference checking; compatibility analysis
8	Shortfalls in externally performed tasks	Reference checking; pre-award audits; award-fee contracts; competitive design or prototyping; teambuilding
9	Real-time performance shortfalls	Simulation; benchmarking; modeling; prototyping; instrumentation; tuning
10	Straining Computer-science capabilities	Technical analysis; cost-benefit analysis; prototyping; reference checking

3.0THE PROPOSED MODEL

The proposed software risk management model has four phases - risk identification, risk measurement, risk assessment, risk mitigation and contingency plan. This model enriched the main phases of Boehm risk model together with a wider range of risk categories (comparing Boehm's 10 top risks). DVC is added to the core of the model to improve the performance of a risk management process. The distinct responsibilities of each phase provide the functional independency. In the first phase, the preliminary risks are identified and the risk factor checklists are prepared. In the second and third phases, the risk factors' attributes are measured and assessed using qualitative or quantitative methods. The output of the third phase is a full evaluation report of the risks, which are ranked and prioritized. In the third phase, the Work Breakdown Structure (WBS) is used to obtain more accurate results. The fourth phase has two parts: (i) designing the mitigation and contingency plan, and (ii) implementing the aforementioned plans. Finally, the verifier core determines the appropriate time for the next iteration after the completion of the fourth phase.

Fig. 2 shows the proposed model for IT projects risk management. The unique feature of this model is the role played by a dynamic expert committee – to suggest necessary changes and eliminate deviations. These experts will be chosen based on their experience in the same field. Another criterion for selecting them is the success rate of their previous risk assessment in similar projects [9].

3.1. Phase 1: Risk Identification

In this phase, the potential risks to the project will be identified. This includes identifying the type and category of risks, which will be measured in the next phases. There are three main components in risk identification - technical risk, application-user risk, and business risk [12]. The risk identification steps are described below:

Step 1: Reviewing project documents and previous risks records. A review of the project documents, particularly the proposal and analysis reports, helps in identifying noteworthy risks [13]. The survey of well-documented risks in the previous phase or in similar projects or in the organization during the experimental stage can provide an assessment on the likelihood of risks [14].

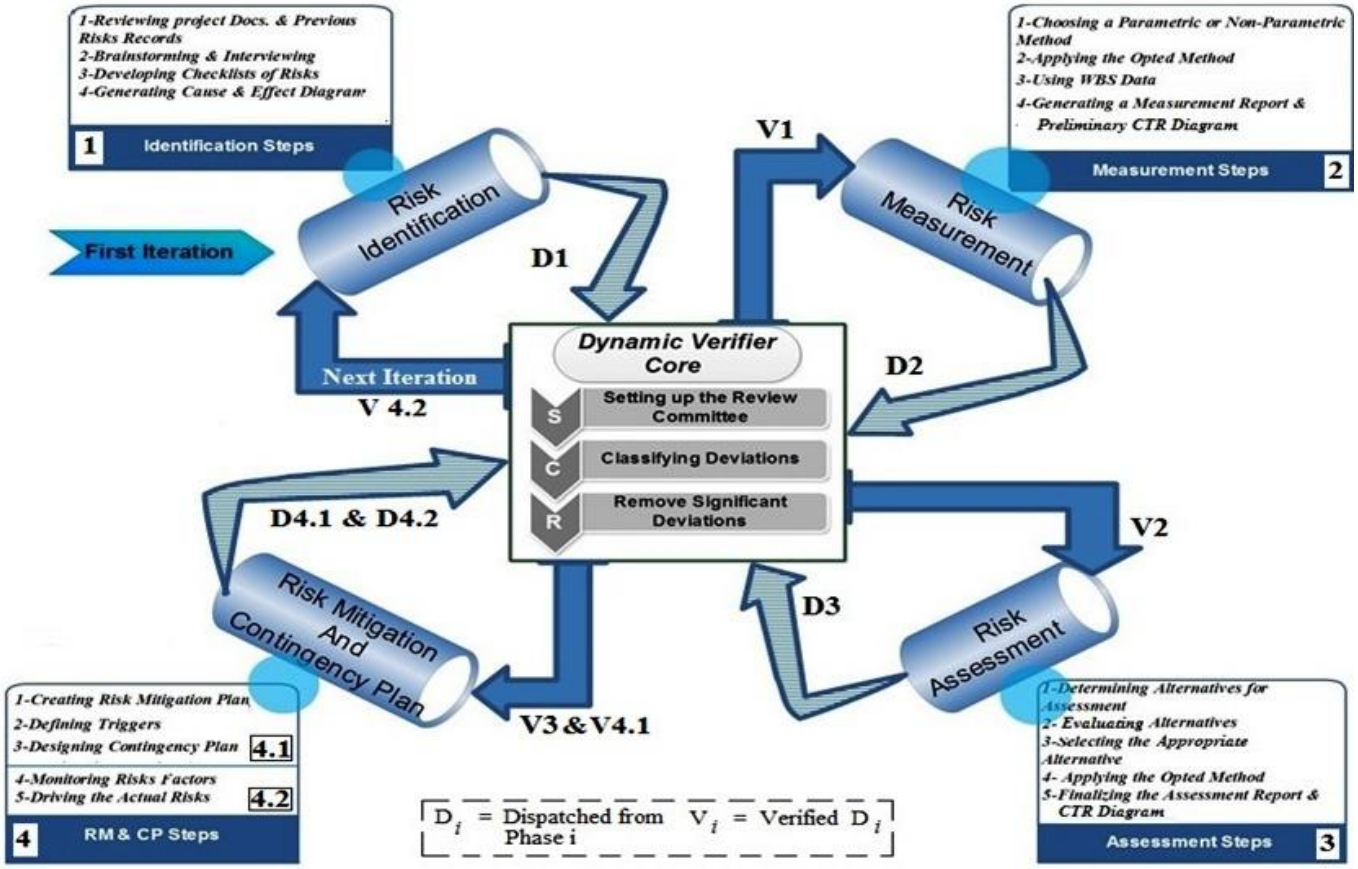
Step 2: Brainstorming and interviewing. The specialists in every section, especially the experienced ones, can provide a clearer picture of the risks, their components, and the threats to the resources. This step highlights the need to have brief but meaningful meetings with relevant people, especially the experts, who are well-informed about the present and the past risks.

Step 3: Developing checklists of risks. Simple and unambiguous checklists must include the results of studies in the previous steps, which provide information on the title of the risk, the type of threats, and the IT project assets which might be vulnerable to the threats (including business, technical, time, and management threats) [15].

Step 4: Generating the cause and effect diagram. This is the most important step because it involves the identification of the causes of a risk occurrence and its consequences or its influence on the risks [16].

3.2. Phase 2: Risk Measurement

This phase deals with the most important characteristic of a risk – determining or measuring to extent a risk can affect the different components of a project, work product, or end product [17]. For this purpose, both qualitative and quantitative methods should be applied. In the qualitative methods, a Likert scale is used to evaluate the risk attributes, while in the quantitative methods the parametric Mont-Carlo method or the non-parametric Simulation method could be used [18]. The professionals fill up a three and five point Likert scaled survey for the risks consequences and likelihood in this phase. However these surveys basically are extracted from the brainstorming, prior interviews with the experts and the related literature. Especially this method is also used for Work Breakdown Structure (WBS) for being more effective and better performance. This shows the combination of qualitative and quantitative methods. Phase 2 consists of four steps:



*Clock wised Diagram, WBS: Work Breakdown Structure, CTR: Cost Time Risk.

Fig. 2. The proposed model with the embedded dynamic verifier core

Step 1: Choosing a qualitative or quantitative measurement method. An appropriate measuring method will be selected after considering the characteristics of a risk which had been determined in the previous phase, the data obtained in the real environment, the time and budget limitation, and other constraints.

Step 2: Applying the selected method. The Work Breakdown Structure (WBS) method for measuring risks is used in the proposed model [19]. This is because the structure of the IT projects is based on the requirements specifications of each phase, rather than the physical components. In this way, the information obtained from each section of an IT project will be suitable for risk estimation and measurement [20].

Step 3: Using WBS data. The selected method in step 1 is applied for well-defined risks. The measures applied will update the risk factor attributes by a relative amount using the common unit. In the measuring process, each risk will be categorized either as Catastrophic, Critical, or Marginal [21]. Significant financial shortage or technical performance degradation are categorized as Catastrophic risks. Critical risks conclude minor delays in software modifications and some reduction in technical performance; however Marginal risks are minimal to small reduction in technical performance and financial resources [11].

Step 4: Generating a measurement report. The measured attributes of the risk factors are added to the information in the previous checklists, and they will be evaluated and reported with different units of time, and cost [22].

3.3. Phase 3: Risk Assessment

Risk assessment is the key to successful risk management [23]. In this phase, the risks will be rated and ranked based on the data and calculations done in the previous phase. In the event of any inaccurate assessment, there will be major difficulties in allocating the resources, in scheduling, and in the contingency plan [24]. The rank can be calculated using the following Equation (by multiplying Occurrence Probability (OC_Prob) and Impact Intensity (II)):

$\forall i \in \{RiskCategories\}$; $\forall j \in Risk.category_i$,

$$\text{Rank (Risk}i\text{)} = \sum_{j=1}^{Max} [\text{OC\_Prob}(\text{Risk}_{i,j}) * \text{II}(\text{Risk}_{i,j})] \quad (1)$$

Where: $0 \leq \text{OC\_Prob}(\alpha) \leq 1$

This qualitative measurement method can also be applied in a matrix form, known as “Cross Impact”. In the table, the term Consequence refers to the probability of a risk occurrence, while Likelihood refers to the Impact Intensity (see Equation (1)).

Table 2 shows the Cross Impact of Consequence and the likelihood of risks that are pertinent to research. The 5-level Likert scale is used in the evaluation of Likelihood; and Consequence is evaluated using a 3-level Likert scale.

Thus, the table shows that the two important risk criteria - Consequence and Likelihood – that determine how the risks will be ranked into either the Catastrophic, Critical, or Marginal category.

Table 2. Cross impact matrix

Likelihood	**Consequences**		
	High	**Medium**	**Low**
Very Likely	*Catastrophic*	*Catastrophic*	*Critical*
Likely	*Catastrophic*	*Critical*	*Critical*
Moderately	*Catastrophic*	*Critical*	*Marginal*
Unlikely	*Critical*	*Critical*	*Marginal*
Very Unlikely	*Critical*	*Marginal*	*Marginal*

3.4. Phase 4: Risk Mitigation and Contingency Plan

This phase uses the data extracted from the previous phase. This phase has two parts: (i) designing the mitigation and contingency plans in three steps, and (ii) implementing the aforementioned plans in two steps. The steps of phase 4 are as follows:

Step 1: Creating a risk mitigation plan. The mitigation plan is designed based on the information from the previous phases. This plan reduces the likelihood of risk occurrence, but if it occurs, it lessens the intensity of the adverse consequences of each risk [25].

Step 2: Defining triggers. Some criteria are defined together with the assessment routines during continuous monitoring in order to determine the exact time when a risk occurs [26].

Step 3: Designing a contingency plan. If a risk has occurred, the plan determines what measures to take to compensate the consequences, depending on the risk characteristics [27].

Step 4: Monitoring the risk factors. This step includes the routines, which regularly provide some information on risk supervision and its characteristics. Monitoring is one the most important stages in the risk management process, as it determines any decision to restart, in a model.

Step 5: Driving the actual risks. If a risk has occurred, the contingency plan will be executed. The checklists and reports are concurrently dispatched to the DVC to initiate the necessary modifications.

3.5. Dynamic Verifier Core

The proposed Dynamic Verifier Core (DVC) consists of three stages, as shown in Fig. 3. Each stage has some predefined responsibilities to facilitate interoperability.

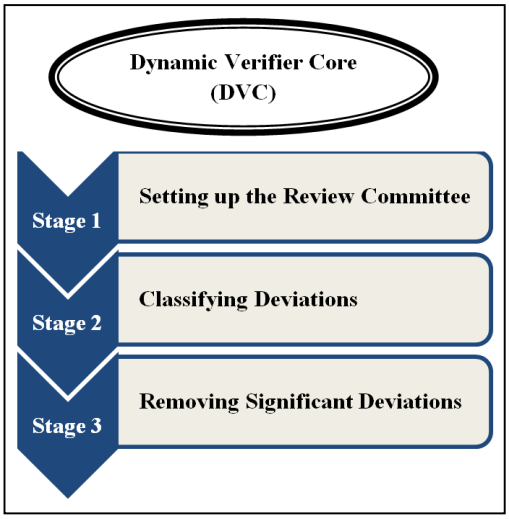


Fig. 3. Dynamic verifier core

After the completion of every phase, the information is dispatched to the DVC using pre-designed forms. A committee of experts had been set up to review the input (stage one). Any potential deviation discovered during the review was classified (stage two). Finally, the committee decided to eliminate the major deviations and update the modified checklist (stage three). In each phase, all major deviations were dealt with accordingly, and even the normal operations were improved. Tesch, Kloppenborg, and Frolick [28] diagnosed the deviations based on two criteria: i) any mistake that occurs during the functioning of the phases, and ii) changes in the real risk situation or the environmental changes. Another function of the DVC is to determine when to perform the next iteration of the model.

The proposed model has the following advantages: i) its simplicity, ii) the transparency of each phase and the gradual occurrence of the risk factors, iii) each phase is verified by the same people involved in the phase, as well as by the independent experts of the DVC committee, and iv) there is an external perspective to the tasks performed in each phase as they involved people who are not affected or influenced by the steps already completed. Thus, their judgments and suggestions have an independent perspective, v) it is a comprehensive model as it covers all stages from identification to implementation, to mitigation and contingency planning, and vi) embedding the DVC within the center of the model satisfies the relevant requirements of each phase independently.

4.0 CASE STUDY

In order to evaluate the proposed risk management model, it was used in a case study which was conducted over six months. The verifier team comprised the successful risk managers of other projects in the same company, while the researcher supervised and managed the flow of activities and the consequences based on the proposed model. The development of the Customer Relationship Management (CRM) system for a large industrial design company, "B.H", was used for the case study. The company has extensive experience in areas such as interior and exterior design, industrial design, graphics, multimedia, web-based software development, and website design. The CRM system is one of the current software systems being developed by this company. This project integrates various

activities such as buying and selling, marketing, and user involvement. The Spiral development method was used, and includes iterations to re-evaluate and restart the model at the end of each round.

The importance and sensitivity of the CRM system in a customer-based company motivated the researcher and the company's manager to re-manage the risks of the risk management process. The project is iterative in nature, thus, it will never stop. A comparison of the results from this case study would give a better understanding of the ability and efficiency of the model.

The main findings of the case study are tabulated and discussed in the sections below. The details of the meetings, the Cause and Effect forms, and the CTR diagrams remain confidential, as agreed between the researcher and the company.

4.1. Classification of Risk Categories and Risk Factors

The researcher developed a list of potential software risk categories and risk factors for the implementation of the model. The classification of the major software risk categories and their corresponding risk factors facilitates the risk management process and maintains uniformity in the model. Altogether, 15 categories of IT and software risks were selected for the evaluation of the proposed model. For each category of risk, the related risk factors were identified. Table 3 shows the 15 risk categories together with the 42 related risk factors. This table has four main columns - Category Code (C_Code); Category name (Category); Risk factor Code (RF_Code); and Risk Factors description (Risk Factors). The classified risk categories and risk factors create an initial framework of this model. The risk categories and risk factors, however, could be customized during the risk management process depending on any special circumstances of a project or a contract [29 , 30].

4.2. The implementation of the phases

- Phase 1: Risk Identification

In the first phase of the proposed model, the risk managers identified the risks by using the relevant risk categories and risk factors. Table 4 shows ten identified risks in phase 1 with the codes and descriptions of the effective risk factors [31]. In the experts' final reports for phase 1, some risk categories were found to be inapplicable . In addition, some risk factors were found to be redundant for the related categories, for example, the outsourcing risks and the responsibility of subcontractor to manage such risks. The report also pointed out that the training risk was ignored, and this is against the terms of the contract, which stipulate that the company should provide training for the staff to use the new system. The complexity risk was also ignored; because this professional company has clear guidelines on avoiding complexity in software development. On the other hand, the ethics risk was omitted in line with the terms of the contract and the friendly work environment in the company.

The maintenance risk was not specified because of the nature of the Spiral method used in the project development process, which involves iteration.

Table 3. Selected risk categories and the corresponding factors

No	C_Code	Category	RF_Code	Risk Factors
1	PM	Project Management	PM_1 PM_2 PM_3 PM_4	•Lack of effective project management skills/involvement •Poor project management •Introduction of new technology •Failure to manage end user expectations
2	US	User Side	US_1 US_2 US_3 US_4	•Not enough of user involvement •Lack of cooperation from users •Failure to gain user commitment •User's satisfaction
3	MN	Senior Management Support	MN_1 MN_2	•Lack of top-management commitment in the project •Lack of corporate leadership
4	SF	Personnel and Staffing	SF_1 SF_2 SF_3 SF_4 SF_5 SF_6 SF_7 SF_8	•Team arrangement •Not enough people with the right skills •Lack the required knowledge/skills among the project personnel •Lack of skilled personnel •Incompetent IS professionals in the team •Infrequent meetings of the project team •Excessive use of outside consultants •Staff satisfaction
5	TR	Train	TR_1	•Poor/inadequate user training
6	PR	Process Maturity	PR_1 PR_2 PR_3	•Lack of scientific methods •Poorly communicated goals/deliverables •Process related issues
7	TE	Technology	TE_1 TE_2 TE_3	•Technological newness •Innovations •Lack of technical specifications
8	CO	Complexity	CO_1	•Application size and complexity
9	ET	Ethic	ET_1	•Unethical behaviour
10	EN	Environmental	EN_1 EN_2 EN_3 EN_4	•Changing needs •Excessive and secondary requirements •Lack of frozen requirements •Changing scope/objectives
11	PN	Project Nature	PN_1 PN_2 PN_3 PN_4 PN_5	•Ignoring the obvious •Requirements creep •Misunderstanding the requirements •Conflict between user departments •Insufficient/inappropriate staffing
12	PP	Project Plan	PP_1 PP_2 PP_3 PP_4	•Lack of a documented project plan •Excessive schedule pressure •Deviation from timetable •Deviation from budget
13	MN	Maintainable	MN_1 MN_2 MN_3	•Maintenance plan •Maintenance cost •Maintenance time
14	SB	Subcontract	SB_1 SB_2 SB_3	•Time •Cost •Quality
15	SC	Security-Confidential	SC_1	•Security

The following are considered the risk factors of the identified risks:

Insufficient user involvement - Not applicable because of outsourcing

Lack of technical specifications - Not applicable because the technical information of the system and the experiences gained from previous successful projects are available

Changing needs - Not applicable because adaptation of the Spiral development methods allows the developers to make changes as the development process progresses

Table 4. Identified risks in Phase 1

No	C-Code	Category	RF-Code	Risk Factors
1	US	User Side	US_2 US_3 US_4	Lack of cooperation from users Failure to gain user commitment User satisfaction
2	TE	Technology	TE_1 TE_2	Technological newness Innovations
3	EN	Environmental	EN_3	Lack of frozen requirements
4	PN	Project Nature	PN_4	Conflict between user departments
5	PP	Project Plan	PP_3 PP_4	Deviation from timetable Deviation from budget
6	PR	Process Maturity	PR_3	Process related issues
7	MN	Maintainable	MN_1 MN_2	Maintenance plan Lack of corporate leadership
8	SB	Subcontract	SB_1 SB_3	Time Quality
9	SC	Security-Confidential	SC_1	Security
10	SF	Personnel and Staffing	SF_8	Staff satisfaction

Table 5. Verified identified risks

No.	Risk Category	Previous RF	New RF	Modified Corrections
1	User satisfaction	US_4	US_4_a	Client satisfaction in working with current system
			US_4_b	Staff satisfaction in working with new system
2	Project Plan	PP_3	PP_3	On-time prototype delivery (Spiral)
		PP_4	PP_4	Sufficiency of the assigned budget for the project
		-	MN_2	Maintenance costs
3	Department Collaboration	PN_4	PN_1_a	No existing information
			PN_1_b	Non-availability of information
			PN_1_c	Incorrect information

Table 6. Consequences of the risk factors

No.	RF_Code	**Consequences (C_S)**		
		Low	**Medium**	**High**
1	US_4_a US_4_b	Completely matched to user's views and requirements	Considered user's views and requirements partially	Did not match user's views and requirements
2	PP_3	Ignore deviation	A simple removal policy for deviation during the project	Major deviations that are impossible to be removed during the project
3	PP_4	Up to 5% deviation from the estimated budget	6% - 10% deviation from the estimated budget	More than 10% deviation from the estimated budget
4	PN_1_a PN_1_b PN_1_c	Incorrect information that can be modified before the end of the project	Incorrect information that can be identified before the end of the project	Incorrect information that cannot be identified before the end of the project
5	TE_1 TE_2	The possibility of handling by the end of designing each phase	The possibility of handling by the release date of each version	Postponing the release date of the final version
6	EN_3	Compatibility with requirements of each department	Conflicting with some requirements of some departments	Severely conflicting with most of the requirements of departments
7	MN_1	Completely maintainable through a contract	Limited maintenance contract	Did not consider maintenance procedures
8	SB_1	Ignore deviation	A simple removal policy for deviation during the project	Major deviations that are impossible to be removed during the project
9	SB_3	Compatible with system specifications	Partial compatibility with system specification	Conflicting with system specification
10	SC_1	Complete security protocol defined	Some security procedure defined	Security issues ignored
11	PR_3	Report and make minor changes	Change some goals of the system	Change the main goals and requirements of the system
12	US_2 US_3	Welcome feedback to improve the system	Resistance of some staff and departments to feedback on the system	Resistance of most of the staff and departments to feedback on the system

- The outcome of the DVC review of phase 1

The results from phase 1 were delivered to the independent DVC experts. As shown in Table 5, the verifiers made some major and minor changes to the identified risks based on their experiences and on the information derived from their review of the project documents. The changes made include the decomposition of the "User Satisfaction"

risk factor into two risk factors - the satisfaction of the clients with the system and the satisfaction of the staff with the system; and the "system maintenance" risk factor. Despite the identifiers' view regarding the inapplicability of the maintenance risk, these experts ranked the maintenance costs in the project plan category. Therefore, the project plan risk factors have been redefined more accurately, and are based on the methodology used. The related risk factor pertaining to the departments' cooperation, which was placed in the project nature category, was omitted and the corresponding risk factor "Ignoring the obvious" was replaced. The verifiers also decomposed the above-mentioned risk factor into three risk factors – non-existence of information; unavailability of information; and incorrect information. Finally, they suggested "Department collaboration" as the name for this category of risk factors.

- Phase 2: Risk Measurement

Table 6 shows the requirements and specifications of the consequences of each risk factor. The experts also had meetings with the company manager and the financial managers, and through the meetings they were able to identify three levels of risk factor consequences - Low, Medium, and High. In other words, the measurement criteria of the identified risks were determined.

- The outcome of the DVC review of phase 2

Table 7 shows the results of the verifiers' review of phase 2, based on information gathered from the project documents, interviews, and filled questionnaires. They made three changes in the measurement phase, and this is also reflected in Table 7. The most important change concerns the exact definition of the deviation rate from the estimated schedule, and not merely based on the qualitative description by the experts. This definition has also been used in the subcontract.

Table 7. Verified output of phase 3

No.	RF_Code	**Consequences (C_S)**		
		Low	**Medium**	**High**
1	PP_3	Up to 10% deviation from the estimated schedule	11%- 30% deviation from the estimated schedule	More than 30% deviation from the estimated schedule
2	PN_1_a PN_1_b PN_1_c	Incorrect information that has minimum effect on the accuracy of the system	Incorrect information that has been identified and removed till	Incorrect information that has serious effect on the accuracy of the system and not identified till the end of the project
3	SB_1	Up to 10% deviation from delivery time	11%- 30% deviation from delivery time	More than 30% deviation from delivery time

- Phase 3: Risk Assessment

In this phase, the verifiers considered the verified risk factors to determine the likelihood and the consequence of the risks. They referred to the project documents to obtain information on the project details, the terms of the outsourcing contract, the success rate of the outsourcing company in other similar projects, interviews, and filled questionnaires by the staff and clients. Table 8 shows the likelihood and consequences of the risk factors.

Table 8. Assessing the risk impact based on the consequences and likelihood of the risk factors

No	Risk factor	RF_Code	Likelihood	Consequence	Impact
1	Client satisfaction in working with current system	US_4_a	Likely	High	**CA**
2	Staff satisfaction in working with new system	US_4_b	Likely	Medium	**CR**
3	Technological newness	TE_1	Moderate	Low	**MA**
4	Innovations	TE_2	Moderate	Low	**MA**
5	Lack of frozen requirements	EN_3	Likely	Low	**CR**
6	On-time prototype delivery (Spiral)	PP_3	Moderate	Medium	**CR**
7	Sufficiency of the assigned budget for the project	PP_4	Moderate	Low	**MA**
8	Maintenance plan	MN_1	Likely	High	**CA**
9	Outsourcing - timeline	SB_1	Likely	High	**CA**
10	Outsourcing – quality	SB_3	Likely	Medium	**CR**
11	Security	SC_1	Very Likely	Medium	**CA**
12	Maintenance costs	MN_2	Very Unlikely	Medium	**MA**
13	No existing information	PN_1_a	Unlikely	Low	**MA**
14	Non-availability of information	PN_1_b	Unlikely	Low	**MA**
15	Incorrect information	PN_1_c	Unlikely	Low	**MA**

- The outcome of the DVC review of phase 3

The verifiers reviewed the results of the assessment and the impact of the risk factors. Almost 33% (one-third) of the risk factors had changed, as shown in Table 9. For example, “staff satisfaction” was assessed as Critical but the verifiers viewed this risk factor as Catastrophic. This is because the consequence had changed to High and as a result the total assessment was deemed to be Catastrophic. In addition, the consequence for the assigned budget of the project had changed to Critical from Marginal. The assessors ranked the related consequence and probability lower than the verifiers. For example, Maintenance was assessed as Catastrophic, but it was ranked lower to Critical. This is because the structure of the Spiral model follows the plan in each iteration, and in the final evaluation at the

last stage of the iteration, the consequence is considered to be Medium. This is similar to the outsourcing timeline. The verifiers believed that the professional outsourcing companies have adopted credible methods and have sufficient experiences to manage the timeline as stipulated in the contract, and any probable delay should not result in any serious consequences. However, they deemed any incorrect information to be Catastrophic, and this might have serious consequences for the project.

Table 9. The verified output of phase 3

No	Risk factor	RF_Code	Likelihood	Consequence	Impact
1	Staff satisfaction in working with new system	US_4_b	Likely	High	CA
2	Sufficiency in the assigned budget for the project	PP_4	Likely	Medium	CR
3	Maintenance plan	MN_1	Moderate	Medium	CR
4	Outsourcing – timeline	SB_1	Likely	Low	CR
5	Incorrect information	PN_1_c	Moderate	High	CA

- Phase 4: Risk Mitigation and Contingency Plan

The proposed model has a comprehensive mitigation plan to monitor, control, and manage the identified and assessed risks. Attention is given to the Catastrophic and Critical risks, and some triggers and monitoring procedures are defined for them. Table 10 shows the selected risks for mitigation. The plan was implemented and the reports were given to the verifiers for them to study and suggest change, where necessary. The final results of the proposed model are highlighted in the next section.

Table 10. Brief explanation of the monitoring and mitigation activities of the contingency plan

No	Risk factor	RF_ Code	Impact	Monitoring Activities	Mitigation Activities
1	Client satisfaction with the current system	US_4_a	CA	Getting regular feedback from the clients about the system	Getting feedback on any dissatisfaction and on ways of removing them and information on related activities
2	Staff satisfaction with the new system	US_4_b	CA	Receiving regular feedback from the staff about the system	Getting feedback on any dissatisfaction and on the ways of removing them and information on related activities
3	Lack of frozen requirements	EN_3	CR	Comparing the requirements and system specifications	Creating change matrix , mapping the change req. and system specifications, modifying the project plan and the contract amendment
4	On-time prototype delivery (Spiral)	PP_3	CR	Focusing on delivery time based on the estimated schedule and the actual delivery time	Regular monitoring of the progress of the prototype, and providing the equipment needed to facilitate the objective
5	Maintenance plan	MN_1	CR	Focusing on the tasks at the end of the cycle and comparing the achievement with the plan	Providing various documents for maintenance and for the staff before starting the activities
6	Outsourcing - timeline	SB_1	CA	Considering the estimated timelines and the elapsed time in the contract	Providing additional resources and the required facilities to meet the timelines
7	Outsourcing – quality	SB_3	CR	Comparing the quality of the products with the pre-defined standards of the contractor	Establishing the quality control committee to check on the quality and to propose practical ways to overcome any probable quality defects
8	Security	SC_1	CA	Regular checking of the data security principles	Gathering and providing data security protocols and documents for the contractors
9	Sufficiency of the assigned budget for the project	PP_4	CR	Instituting ways for reducing the budgets	Providing an urgent budget and taking relevant measures to have a budget for unforeseen costs
10	Incorrect information	PN_1_c	CA	Receiving regular acknowledgements from the experts about any inaccurate information	Developing a log system to gather all correspondences, and encouraging the staff to cooperate in giving the right information and in removing errors

5.0 RESULTS AND DISCUSSION

At the completion of the project, the information obtained on the occurred risks was studied. Table 11 shows the risks that occurred and the risks that did not occur. This table presents the risks that occurred, together with their degree of importance. The last two columns of the table compare the estimated rank of the risks produced by the model and the actual rank of the risks after occurrence. The comparison provides a clearer picture of the level of efficiency of the implemented mitigation activities in reducing the effects of the risks.

Table 11. Comparison of the expected and potential impact of risks

No	**Risk factor**	**Occurred**	**Mitigated**	**Expected Impact**	**Potential Impact**	**Real Impact After Mitigation**
1	Client satisfaction in working with current system	√	√	**CA**	**CA**	**MA**
2	Staff satisfaction in working with new system	√	√	**CA**	**CA**	**CR**
3	Lack of frozen requirements	√	√	**CR**	**CR**	**MA**
4	On-time prototype delivery (Spiral)	√	√	**CR**	**CR**	**MA**
	Maintenance plan	x	√	**CR**	**-**	-
5	Outsourcing – timeline	√	√	**CA**	**CA**	**MA**
6	Outsourcing – Quality	√	√	**CR**	**CR**	**MA**
7	Data security	√	√	**CA**	**CR**	**MA**
8	Sufficiency of the assigned budget for the project	x	√	**CR**	**-**	-
9	Incorrect information	√	√	**CA**	**CR**	**MA**
10	Lack of top-management commitment to the project	√	X	**-**	**CR**	**CR**

It is clear from the table that two risks did not occur (rows 5 and 9). This could be due either to the implementation of the mitigation measures and techniques or the absence of the right environmental condition for a risk to occur. In addition, one risk, "the support of the top manager" that occurred was neither observed in the identification phase nor in the DVC (row 11). This was due to the change in the company's management during the project. Reactive actions were applied to reduce the impacts of this risk. These issues should be assumed as a minor failure of this model; however they do not have major negative effect on total performance of the project [32].

Among all the identified risks, "data security" and "incorrect information" were both evaluated as Catastrophic, but were Critical, in practice (rows 8 and 10). By implementing the mitigation measures, the consequences of these two

risks were reduced to Marginal. None of the evaluated Critical or Marginal risks were deemed to be Catastrophic in the real environment.

Boehm' s model which is depicted in Literature Review section and is a skeleton of the proposed model, addressed less identified risks in this case study; because it covered less categories of risks [11]. In addition, the proposed model with DVC core identified more risks and improved their attributes. The changed or unforeseen risks are not in the list of the new risks identified by the DVC, and this reflects the importance of the DVC actions by the independent verifiers. An analysis of the results of the case study uncovers three ignorable mistakes and this reflects the efficiency of the model. It also shows that by applying the verifiers' decisions and the changes recommended by the independent experts, the risk management activities have been effective. As a result of this case study 6 risk factors are assessed correctly, in addition 2 risk factors are identified in a right track. This shows a significant success rate of the proposed model implementation [33]. The diagram in Fig. 4 shows the rate of the identified risks based on their ranking and occurrence, and it also compares the results.

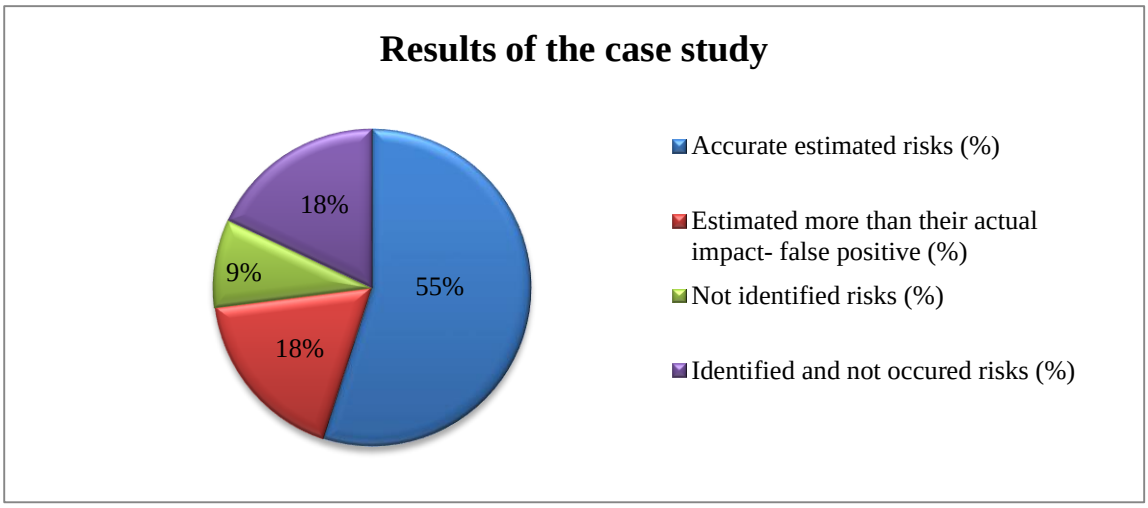


Fig. 4. Analysis of the results of the case study

6.0 CONCLUSION

There are two common threats during the risk management process - the probable errors or mistakes involved in each phase of the process; and the risk alteration during the risk activities. The proposed model incorporates features to address both threats. One of the main aims of this research is to verify the activities during the risk management process. This can be termed as managing the risks of the risk management process.

The proposed model reduces the unforeseen risks or risks that have already occurred by creating a verifier core that comprises risk managers and experts. The verifier core is dynamic as it can adapt to each phase, and this makes the management process efficient and up-to-date. The preliminary risk identification, risk measurement and assessment create the functional independency for each phase. The outcome of each phase, however, is verified by the DVC.

Applying DVC in this model promotes risk assessment 33% by improving risk factors comparing other risk models that do not use DVC [34].

In order to evaluate the efficiency of the model, it was implemented in the development process of a Customer Relationship Management system as a risky IT project. Independent experts were invited to be in a verifier committee and this has been useful in identifying new risks, and in suggesting changes or modifications to those risks. This had contributed greatly towards a favourable outcome for the mitigation and contingency plans. The researcher would like to suggest the application of the fuzzy logic concept in the assessment phase to make the estimation more accurate, in future studies.

.

ACKNOWLEDGMENT

This research was funded by the University of Malaya under the Postgraduate Research Grant (PPP), Account Number: PS027-2012A.

REFERENCES

[1] M. Benaroch, Y. Lichtenstein, K. Robinson, "Real Options In Information Technology Risk Management: An Empirical Validation For Risk-Option Relationship". *MIS Quarterly*, Vol. 30, No. 4, 2006, pp. 827-864.

[2] R. Pennington, B. Tuttle, "The Effects Of Information Overload On Software Project Risk Assessment". *Decision Sciences*, Vol. 38, No. 3, 2007, pp. 489-526. doi:10.1111/j.1540-5915.2007.00167.x

[3] B. W. Boehm, J. Bhuta, "Balancing Opportunities And Risks In Component-Based Software Development". *IEEE Software*, Vol. 2, 2008, pp. 56-63.

[4] B. W. Boehm, "Making A Difference In The Software Century". *IEEE Computer*, Vol. 41, No. 3, 2008, pp. 32-38.

[5] M. Rush, G. Vednere, "Calming The Data Storm: A Risk Management Model For Mitigating Risks". *Information Management Journal*, Vol. 42, No. 4, 2008, pp. 48-54.

[6] L. M. Castro, V. M. Gulías, C. Abalde, J. Jorge, "Managing The Risks Of Risk Management". *Journal of Decision Systems*, Vol. 17, No. 4, 2008, pp. 501-521. doi:10.3166/jds.17

[7] P. Martin, "Why Is Operational Risk Management Important?". *Journal of Securities Operations & Custody* Vol. 2, No. 4, 2010, pp. 324-332.

[8] S. Alter, S. A. Sherer, "A General But Readily Adaptable Model Of Information System Risk". *Communications of AIS*, Vol. 14, 2004, pp. 1-28.

[9] A. S. Khatavakhotan and S. H. Ow, "Improving IT Risk Management Process by an Embedded Dynamic Verifier Core: Towards Reducing IT Projects Failure," in 2012 Third International Conference on Intelligent Systems Modelling and Simulation, 2012, pp. 684-687.

[10] F. Tüysüz, C. Kahraman, "Project Risk Evaluation Using A Fuzzy Analytic Hierarchy Process: An Application To Information Technology Projects". *International Journal of Intelligent Systems*, Vol. 21, 2006, pp. 559–584. doi: 10.1002/int.20148

[11] B. W. Boehm, "Software Risk Management: Principles And Practices". *IEEE Software*, Vol. 8, No. 1, 1991, pp. 132-41. doi: 10.1109/52.62930

[12] S. G. Sutton, D. Khazanchi, C. Hampton, V. Arnold, "Risk Analysis In Extended Enterprise Environments: Identification Of Critical Risk Factors In B2B E-Commerce Relationships". *Journal of the Association for Information Systems* ,Vol. 9, No. 4, 2008, pp. 151-174.

[13] R. Pennington, B. Tuttle, "The Effects Of Information Overload On Software Project Risk Assessment". *Decision Sciences*, Vol. 38, 2007, pp. 489–526. doi: 10.1111/j.1540-5915.2007.00167.x

[14] K. C. Iyer, M. Sagheer, "Hierarchical Structuring Of PPP Risks Using Interpretative Structural Modelling". *Journal of Construction Engineering & Management*, Vol. 136, No. 2, 2010, pp. 151-159. doi:10.1061/(ASCE)CO.1943-7862.0000127

[15] D. Wu, D. L. Olson, "Introduction To The Special Section On Optimizing Risk Management: Methods And Tools". *Human & Ecological Risk Assessment*, Vol. 15, No. 2, 2009, pp. 220-226. doi:10.1080/10807030902760967

[16] J. S. Fraser, K. Schoening-Thiessen, B. J. Simkins, "Who Reads What Most Often? A Survey Of Enterprise Risk Management Literature Read By Risk Executives". *Journal of Applied Finance* Vol. 18, No. 1, 2008, pp. 73-91.

[17] B. Kloss-Grote, M. Moss, "How To Measure The Effectiveness Of Risk Management In Engineering Design Projects?". *Research in Engineering Design*, Vol. 19, No. 2/3, 2008, pp. 71-100. doi:10.1007/s00163-008-0049-y

[18] M. K. Khedr, "Project Risk Management Using Monte Carlo Simulation". *AACE International Transactions* 2006, pp. 2.1-2.10.

[19] J. Zhao, "Significance Of WBS In Contingency Modelling". *AACE International Transactions*, 2006, pp. 5.1-5.5.

[20] M. Better, F. Glover, G. Kochenberger, H. Wang, "Simulation Optimization: Applications In Risk Management". *International Journal of Information Technology & Decision Making*, Vol. 7. No. 4, 2008, pp. 571-587.

[21] A. Qazi, R. G. Raj, M. Tahir, M. Waheed, S. U. R. Khan, A. Abraham, "A Preliminary Investigation of User Perception and Behavioral Intention for Different Review Types: Customers and Designers Perspective," *The Scientific World Journal*, vol. 2014, Article ID 872929, 8 pages, 2014. doi:10.1155/2014/872929.

[22] P. Chitakornkijsil, "Disaster And Risk Management In A Global World". *International Journal of Organizational Innovation*, Vol. 3, No. 2, 2010, pp. 97-113.

[23] J. Barve, "COBIT For IT Risk Management In A Bank-A Case Study". *COBIT Focus 2010*, Vol. 3, 2010, pp. 1-6.

[24] R. Carmen, "The Enterprise Information System And Risk Management". *Annals of the University of Oradea, Economic Science Series*, Vol. 18, No. 4, 2009, pp. 1030-1034.

[25] S. Scandizzo, "Risk Mapping And Key Risk Indicators In Operational Risk Management". *Economic Notes*, Vol. 34, No. 2, 2005, pp. 231-256. doi:10.1111/j.0391-5026.2005.00150.x

[26] L. Yetman, "Project Management: Careful Planning Or Crystal Ball?". *Journal of the Quality Assurance Institute*, Vol. 20, No. 3, 2006, pp. 40-42.

[27] R. Prasad, "Schedule And Cost Risk Evaluation". *AACE International Transactions*, 2007, pp. 04.1-4.5.

[28] D. Tesch, T. J. Kloppenborg, M. N. Frolick, "IT Project Risk Factors: The Project Management Professional Perspective". *Journal of Computer Information Systems*, Vol. 47, No. 4 , 2007, pp. 61-69.

[29] S.-T. Lu, "Risk Factors Assessment for Software Development Project Based on Fuzzy Decision Making". *International Journal of Information and Electronics Engineering*, 2012.

[30] S. V. Shrivastava and U. Rathod, "Categorization of risk factors for distributed agile projects". *Information and Software Technology*, Vol. 58, Feb. 2015, pp. 373–387.

[31] B. Shahzad, "Identification of Risk Factors in Large Scale Software Projects", *International Journal of Knowledge Society Research*, Vol. 5, No. 1, 2014, pp. 1–11.

[32] Anon, "Failure Rate of Software". *Springer Series in Reliability Engineering*, 2008, pp. 225–236.

[33] S. K. Mathew and Y. Chen, "Achieving offshore software development success: An empirical analysis of risk mitigation through relational norms". *The Journal of Strategic Information Systems*, Vol. 22, No. 4, Dec. 2013, pp. 298–314.

[34] S. Islam, H. Mouratidis, and E. R. Weippl, "An empirical study on the implementation and evaluation of a goal-driven software development risk management model". *Information and Software Technology*, Vol. 56, No. 2, Feb. 2014, pp. 117–133.