AUTOMATED RISK ANALYSIS SOFTWARE MODEL FOR ENHANCED SOFTWARE DEVELOPMENT

Ghazala Nasreen#,*,[1],Muhammad Asam #,*,[2] Salman Afsar #,*,3

#,*,1 Department of Computer Science, National College of Business Administration And Economics, Lahore, Pakistan

#,*,2 Department of Computer Science, GC University, Lahore, Pakistan

#,*,3 Department of Computer Science, University of Agriculture, Faisalabad, Pakistan

Corresponding author: ghazalah.mahar@yahoo.com

ABSTRACT: *This paper presents Risk Covered Software Model (RiCSoM) that is a software development model which also performs the efficient risk management from the preliminary phase in an automated way and ensures a successful software project. Although, the other software models do not perform risk analysis from preliminary phase but in the mid or end phases, which results lot of failures due to customer's changing requirements. The model also detects commonly occurring risk factors in different phases of software development (SD) to mitigate risk. In this model, risk analysis process starts from the preliminary phase that helps to avoid transferring risk to the next phases, decreasing the cost of risk mitigation at the end of the project. It provides a framework that even a beginner developer can develop successful software without concern of its nature and size unlike other models.*

Keywords--- Risk analysis, Risk management, Process model, Risk assessment.

INTRODUCTION

With the emergence of software engineering technology, lots of software are being developed but still there are lot of failures. About 1/3rd of the software are completed successfully and others with over budget and time due to flaws. Most of the software fail due to over budget, delay in the delivery of the software's and so on. The main risk factor is changing requirements causing loss or project failure [1]. Software development has a lot of strategic opportunities but still subjected to high levels of uncertainty and risk [2]. Risk is itself not bad but its consequences may be negative so it should be reduced or mitigated and we experience it by doing this. There are many risk analysis tools for prediction of potential risk. These tools generally provide help to update risks and making of the reports by compiling records and the analysis [3].Current risk analysis tools are specific to any system and the software development model which is used for development, so different risk analysis techniques are used for different models [4]. There is a need of risk analysis tool independent of software project nature, size and the software development model used, either agile or iterative. Agile model uses lot of testing but still there exists lot of failure chances. Spiral model (Roger Pressman) a traditional risk analysis model uses risk analysis technique but suits for small projects due to its cost and expertise. For the effective risk management there should be intelligent risk analysis model giving framework for risk free software development.

RISK ANALYSIS AND RISK MANAGEMENT IN SOFTWARE DEVELOPMENT

Risk Analysis

It may be difficult to measure potential loss and uncertainties in project. There may be high chance of error in measuring these two elements. Dealing with high potential loss is different from dealing with low potential loss [5].Generally there are two types of risk strategies. Traditional is reactive and used for generic problems, but risk is proactive that deals with agile problems [6].

RISK MANAGEMENT

Risk management is recognized as means of mitigating software failures [7]. It's the activity that helps the project managers to recognize and manage the risk of cost overruns in software and helps the team and managers to make better decisions.

LITERATURE REVIEW

If we need a success of software then it is necessary that a software development model should be correctly selected. Although many models have been presented at the beginning of the waterfall model [7].Each phase of the SDLC (Software Development Life Cycle) is vulnerable to the different kinds of risk factors. Risk management is to identify and to understand these risks in preliminary stage successfully [8].

Automated Threat Modeling introduced the identification trees and mitigation trees data structures to identify the threats in the software designs during design phase of software development and advise mitigation techniques [9]. Firdose2014, worked on risk management and the state of risk for most-liked software development models; (i.e. waterfall, spiral and agile development) [10].A model was proposed using Bayesian Belief Network technology which focuses on the top risks of the software for risk assessment during software development projects [11].Evaluating risks to the schedule is really complex. Different strategies for the schedule of risk analysis of software development were presented [12].

Analysis of the risk impacts across all the stages of software development is categorized in projects as short term and the long term projects. This analysis mode helps to enable the developer to manage the risks effectively [13].Risk is a potential problem, it may occur or not but to identify the risk is a good idea.

In the software development projects everyone is involved i.e. software managers, engineers and other stakeholders in risk analysis and management must participate also. But as long as the software industry is growing software risks have been increasing too [14].

RiCSoM Model

RiSCoM is independent of the software size and nature and it develops software using risk analysis technique in an

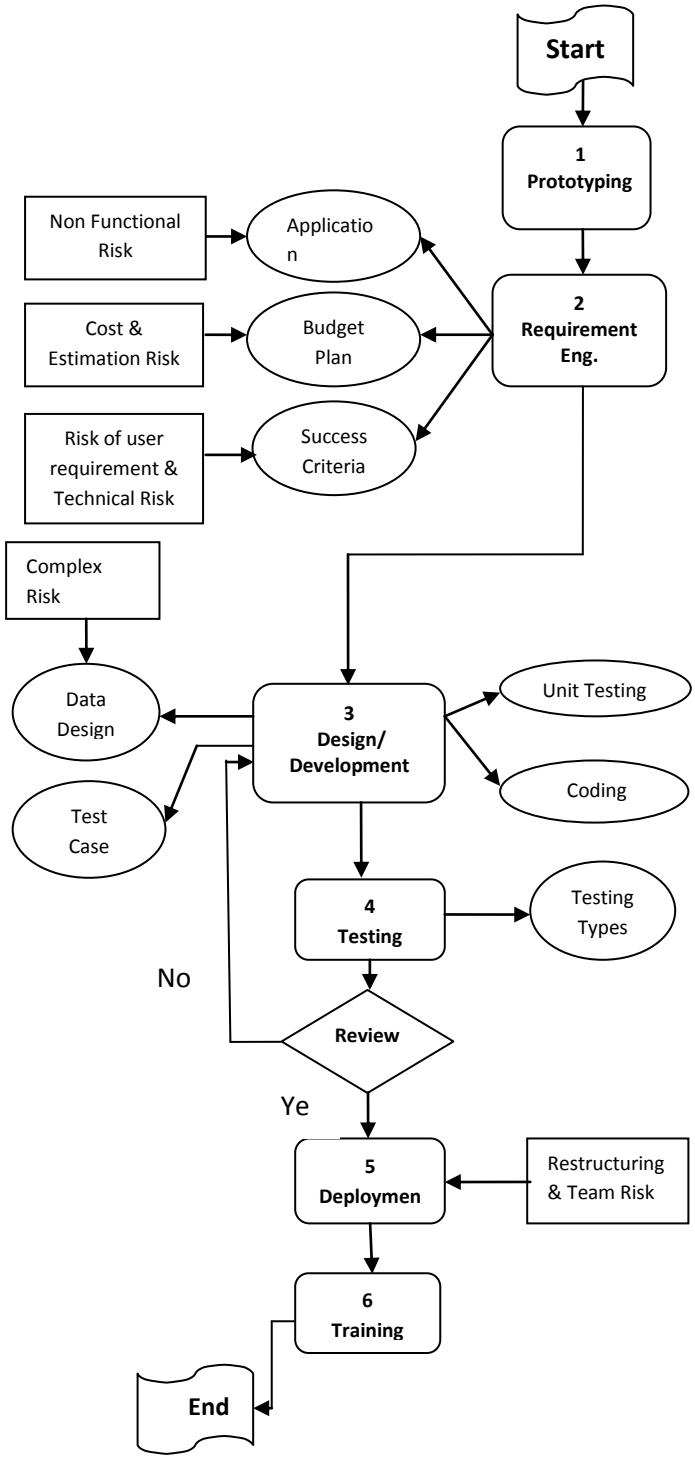

Fig_1: RiCSoM

automated way. Prototyping gathers maximum possible user requirements and maximizes the success chances. Risk analysis used in preliminary phase and throughout the software development, mitigates lots of risk factors ignored by other risk analysis tools. In the requirement engineering phase, if **non functional risk** is ignored it can affect the software performance and security, etc. Similarly the estimation risk and technical risk results the problem of over budget and time or project failure. **Complex risk** is very important while designing software.

Review helps to analyze the software and if the testing fails the model iterates back to design phase to remove problems which ensures software quality. The restructuring and team risk decreases by start to end, involvement of customer in software development.

OTHER RISK ANALYSIS TECHNIQUES

Most of the risk analysis models just generate reports for decision making for expertise to perform risk management which is costly. Another risk analysis model called "Automated Threat Modeling" uses risk analysis only in design phase of software development[15]. Other models use risk analysis in the mid or end phase, ignoring the start phase risks. Spiral model is risk analysis model, but used mostly for small projects due to its cost. Agile technique uses testing for risk management, but still lot of failures.

Collecting the Indicators

In this research work we have reviewed and selected many researchers related to the software development and categorized risk into high, medium and low level. The summarization of risks on the basis of selected risks was done. It helped to develop questionnaire for survey. Descriptive and analytical research design was adopted for research. It's a qualitative type of research in which questionnaire, literature review and interviews were conducted with expert developers with 2 to 9 year experience.

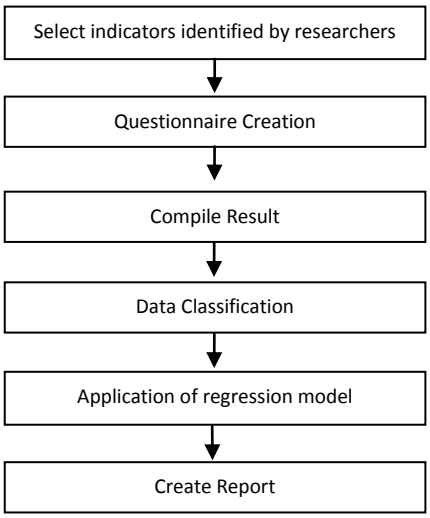

Fig_1: Process for Factors Prediction

Table-1: Classifying Risk Categories

Rank	Risk Category	Rank	Risk Category	Rank	Risk Category
1.	User	10	Team	19	Design
2.	Requirement	11	Organizational Environment	20	Developer
3.	Cost	12	Scope change	21	Development
4.	Schedule	13	Communication	22	Process
5.	Quality	14	Overdrawn budget	23	Stakeholder
6.	Business	15	Unrealistic requirement	24	Resource availability
7.	People	16	Change management	25	Culture
8.	Project complexity	17	Estimation	26	Time dimension
9.	Planning & control	18	Project management		

SUMMARY OF INDICATORS

From the taxonomy given the main groups of the risks category were selected for the research. The classification is given below Table-2:

Table-2: Main Groups for the risk category of software development

Rank	Risk Category:
1	Non Functional
2	Cost
3	Estimation
4	User
5	Requirement
6	Technical
7	Complex
8	Team

DATA COLLECTION AND VALIDATION

The collection of data was done from the experts with the similar opinion direction like agreed, probably agreed or disagreed. If one of them answered not-agreed, then the option of disagreed was selected. So the factor caused the risk for software project came from the same direction of the questionnaire.

FACTORS EXTRACTION AND RESULTS

It is the method to separate and group the all factors from quality analysis and develop the questionnaire with the objective to select the factor with the risk and analyze from these risk factors for the prediction of the risk.

APPLICATION MODULES

The result was analyzed by using ordinal regression of the respondents composed of 30 questions. The summary of all the answers of the questions of the respondents from different software houses was analyzed by SPSS (Statistical Package for Social Sciences). The result and analysis given by the tool is given below.

Table-3: Case Processing Summary

		N	Marginal Percentage
Satisfaction level	Extremely Unsatisfied	4	
	Unsatisfied	29	6.7%
	Satisfied	20	48.3%
	Extremely Satisfied	7	33.3%
Valid		60	11.7%
Missing		0	100.0%
Total		60	

The marginal percentage lists the proportion of valid observations found in each of the outcome variable's groups. Of the 60 subjects, 4 and 29 categorized as extremely unsatisfied and unsatisfied, therefore marginal percentage for unsatisfied responses is 55% that shows that more respondents are unsatisfied as compare to satisfy.

Table-4: Model Fitting Information

Model	-2 Log Likelihood	Chi-Square	Df	Sig.
Intercept Only	136.470 105.978	30.492	6	.000

Link function: Logit.

The above small p-value=0.000 from the LR test, <0.05, would lead us to conclude that at least one of the regression coefficients in the model is not equal to zero.

Table-5: Goodness-of-Fit

	Chi-Square	Df	Sig.
Pearson	184.909	165	.138
Deviance	104.591	165	1.000

Link function: Logit.

The above result shows that the p-value is greater than 0.05, therefore the hypothesis that model is of good fit not rejected here.

Table-6: Pseudo R-Square

Cox and Snell	.398
Nagelkerke	.443
McFadden	.221

Link function: Logit.

Here the Pseudo R-square (Nagelkerke= 44.3%) indicates that six independent variables explain a relatively moderate proportion of the variation in dependent variable (satisfaction levels) which is 44.3% and model is adequate enough. There may some other important predictors of Satisfaction levels that have an effect on the dependent variable.

Table-7: Parameter Estimates

	Estimate	Std. Error	Wald	df
Threshold [Satisfaction = 1]	-.194	2.310	.007	1
[Satisfaction = 2]	3.702	2.350	2.481	1
[Satisfaction = 4]	6.387	2.474	6.667	1
Location Estimation Risk	.563	.391	2.068	1
Tech_Risk	.144	.394	.134	1

Table-7.1: Parameter Estimates

	95% Confidence Interval	
Sig.	**Lower Bound**	**Upper Bound**
.933	-4.721	4.333
.115	-.905	8.309
.010	1.539	11.236
.150	-.204	1.330
.714	-.628	.917

LINK FUNCTION: LOGIT.

In ordinal regression model, if there are "*k*" categories of dependent variable, then a total of "*k* – 1" models are estimated. The results of the above table show that complex risk has a positive, statistically significant, regression relationship on satisfaction level.

CONCLUSION

The purpose of research work was to develop a software model which performs risk management and to find the common occurring risk factors, causing hurdles in successful software development. Preliminary risk classification in this model improves software development's efficiency, cost and time. Prediction of risks was obtained from software developers and it will help in decision making and carry out risk management properly. As the future work, more models are suggested by using other techniques like neural networks and artificial intelligence techniques to mitigate risks of cloud computing in banking organizations.

REFERENCES

[1] Erdogmus. H. N. D. *Risk Assessment and Management Options*. The New England Journal of Medicine. (2015) 78-89.

[2] D. Hartman. Interview: *Jim Johnson of the Standish success frequency on spiral and agile model*. International Journal of Innovative Research in Computer and Communication Engineering. . (2006) 301-310.

[3] Tom Patterson, *The Use of Information Technology in Risk Management. AICPA. (2015) 33-39.*

[4] Majid. *Risk Assessment and Management Options*. The New England Journal of Medicine. (2013) 78-89.

[5} Dr. E. Wallmuller, *Risk Management for IT and Software Projects*, Science Direct. (2014).

[6] Johnson. D. L. *Risk Management and The Small Software* (2009). http://www.sei.cmu.edu/iprc/sepg2006/johnson.pdf

[7] W. Ma, L. Liu, W. Feng, Y. Shan, and F. Peng. *Analyzing project risks within a cultural and organizational setting*, in ICSE Workshop on Leadership and Management in Software Architecture, (2009) 6–14.

[8] Gao,Y. *Research on the Rule of Evolution of Software Development Process Model*. IEEE Software. 78(12) (2010) 466-470.

[9] Hijazi. *et al. Risk factors in software development phases*, European Scientific Journal. (2014) 213-232.

[10] Maxime F. *et. Al, Automating risk analysis of software design models.* Hindawi. (2014).

[11] Firdose. *et al, Risk factors in software development phases*,European Scientific Journal. Firdose. (2014).

[12] Frydman, *et al.,Automating risk analysis of software design models*, Hindawi. (2014).

[13] Suri and K. Narula. *Strategies for evaluating risks to project schedule*, International Journal of Computer Applications .0975 – 8887 (2013).

[14] Bhujang K.B. and Suma, *Risk Impact Analysis across the Phases of Software Development*, Lecture Notes on Software Engineering, Vol. 2. (2014).

[15] Bazaz, Y. *et al.,Comparative Study of Risk Assessment Models Corresponding to Risk Elements*, IEEE-International Conference on Advances In Engineering, Science And Management (2014) 61-66.