

An approach to facilitate software risk identification

A. Sana Khan

Abstract— Literature survey proved that risk identification is the first and most important step of software project risk management. Research is still in progress to introduce best methods for managing software project risks to increase the success ratio. This paper presents the results of industry survey based on the opinion from different software houses regarding the facts that could provide the foundation to facilitate software risk identification to enhance the effectiveness of software risk management, and could lay the foundation to incept data warehousing. The purpose of introducing a data warehouse for software risk management is to facilitate the vendors and clients of software projects to be aware of all the risks that could be identified in a project and mitigation actions to treat those risks, on the basis of past experiences by different companies regarding identified risks and practiced mitigation actions. Proposed idea is discussed in detail on the basis of survey results.

Index Terms – Software risk management, Risk identification, Risk mitigation, Data warehousing

I. INTRODUCTION

As it is a well-known fact that majority of software projects are getting failed and ability to effectively manage the risks is very important issue for software project success. Top IT organizations are practicing systematic risk management in software projects [1]. This paper is aimed to focus on two issues; in time risk identification and appropriate mitigation actions for reducing the probability/impact of identified risks. An approach is suggested for facilitating software risk management, by providing suggestions for in time identification of software project risks and following appropriate mitigation actions to effectively handle those risks.

In time risk identification is an issue, considered very important by different researchers [1] [2] [3]. As it is proved from literature survey [4] [5] [6] [7] [8] [9] that software projects are getting failed and risk management is the major contributor toward their failure, industry survey has been done to explore the facts that could enhance the effectiveness of software risk management by providing the facilitation for software risk identification. A risk suggestion data warehouse is also proposed.

II. INDUSTRY SURVEY

The industry survey has been performed to evaluate the following assumptions for the inception of a risk suggestion system

- Chance of risk repetition in similar kind of projects
- Chance of risk repetition when certain project parameters are similar Technology, Type, deadlines, No of resources and their expertise
- Mitigation actions can also be repeated for repeated risks
- Some automated tool can be used for risk identification
- Automated tool can be more useful than the tool only for data management?
- Automatic risk suggestion system on the basis of company's own data
- Risk suggestion system on the basis of combined data from different companies, while securing the sensitive information for more accurate risk suggestions

III. RESULTS

Questionnaire is used to carry out the industry survey. Project managers, team leads, software engineers from Pakistan's software industry are asked, to share their opinions regarding the eight questions. Questionnaire is emailed to different

Manuscript received September 30, 2008.

A. Sana Khan is with the COMSATS Information Technology center in COMSATS Institute of IT Abbottabad Pakistan phone: 92-0332-8920991; e-mail: (sanakhanre@yahoo.com).

software houses, and personal meetings were also conducted with project managers in some software houses. Evaluation is based on the results of 44 projects from 22 software houses.

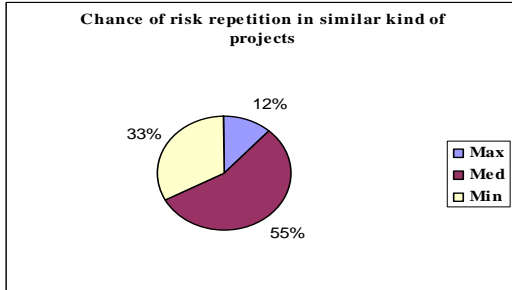


Fig. 1: Chance of risk repetition in similar kind of projects

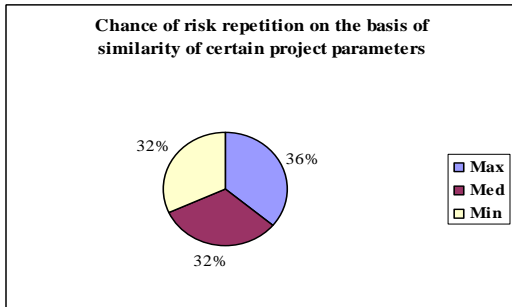


Fig. 2: Chance of risk repetition when certain project parameters are similar Technology, Type, deadlines, No of resources and their expertise

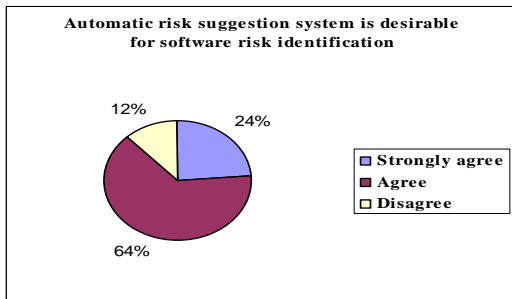


Fig. 3: Automatic risk suggestion system is desirable for software risk identification

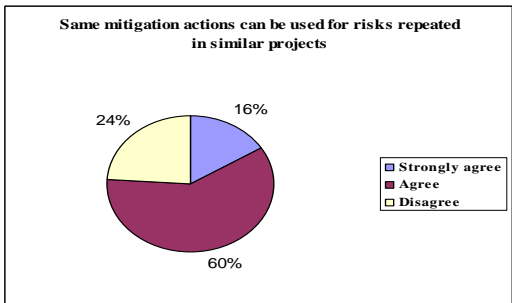


Fig. 4: Mitigation actions can also be repeated for repeated risks

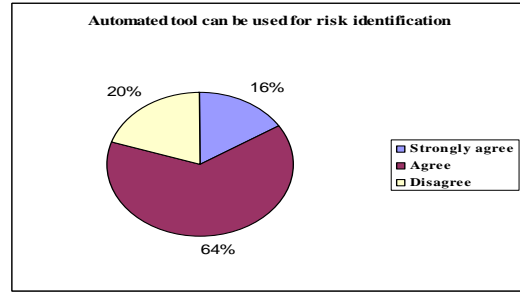


Fig. 5: Some automated tool can be used for risk identification

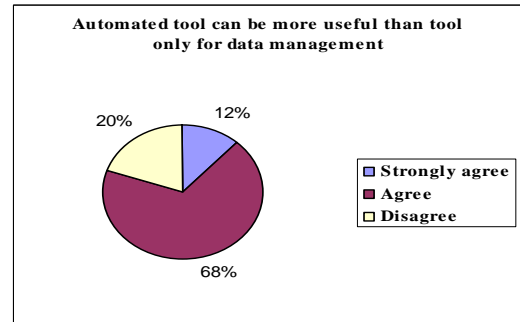


Fig. 6: Automated tool can be more useful than the tool only for data management?

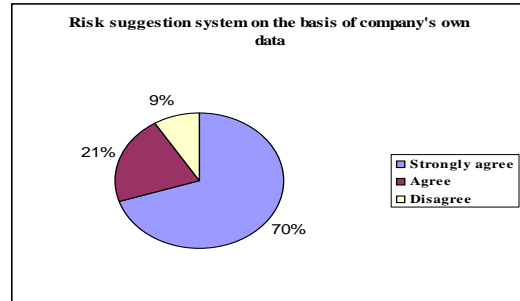


Fig. 7: Automatic risk suggestion system on the basis of company's own data

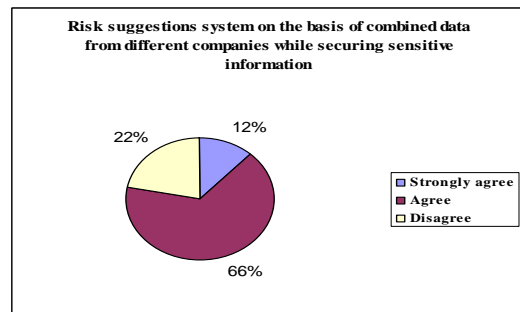


Fig. 8: Risk suggestion system on the basis of combined data from different companies, while securing the sensitive information for more accurate risk suggestions

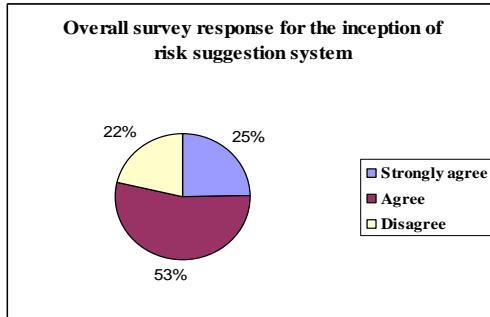


Fig. 9: Overall response

IV. PROPOSED IDEA

Proposed System will provide the suggestions regarding identified risks in previous projects on the basis of comparisons with following parameters.

Type

Application

Technology

Size/ duration

Budget

Total number of resources

Team size

Average Resource Experience

Project manager's experience

Company's experience

Geophysical (Environment) information

Project size, team size, project manager's experience, resource experience and technological aspects are considered as important risk areas of a software project [10]. Type, application, and size are considered as important parameters in other study [11].

A. Suggestions for risk identification

Proposed system will allow different companies to enter newly started project data (parameters listed above) to get the suggestions for possible risks previously experienced by other companies while doing the same type of projects. A list of risks (identified and/or occurred) will be suggested by the system on the basis of comparison with other projects on the basis of application similarity and having close similarity of other parameters. The suggested risk list may contain any type of risks concerning software projects experienced previously. Number of times a risk is previously identified / selected will help the user to decide the risk selection for his project.

User will select the risks that are considered compatible for a particular project, and the selected risks will be loaded in the company's own data base and in central warehouse, along with the increasing probability of risks (showing how many users previously have selected a particular risk suggested by the system), and if user wants to enter any new risk it will be loaded in company's own data base and in central warehouse, again with assigned probability. Purpose of the proposed system is to improve risk management decision-making by providing

suggestions regarding risk identification and mitigation on the basis of past experiences in software projects.

B. Suggestions for risk mitigation

Similarly list of mitigation actions will be displayed at the user request against any risk with the probability (showing how many users have previously selected that mitigation action); outcome on risk, impact of mitigation action, impact details and cost/ resource used for a particular mitigation action to help user decide easily to select a particular mitigation action against any risk. User will select the mitigation actions for his project risk(s) and the selected mitigation action will be stored in the source database and in the central warehouse. If a user enters any new mitigation action it will be loaded in company's own database and in central warehouse with the assigned probability. The whole process will be carried out while securing the company's personal or sensitive information. Company's personal or sensitive information will not be accessible to the other users. This automatic suggestion system will allow its user to identify the risks for a software project on the basis of past experience of the other users. System will present a list of risks after matching particular project specifications like application, project type, duration, team size, resources, avg resource experience, project manager's experience, company's experience and environment information, with all the projects that have some or most of the parameters similar to that project. User will get an appropriate view of all the risks that have been identified in the previous projects, and list of mitigation actions against a particular risk will also be available, that were practiced previously by different companies to handle / control a risk. From the available list of risks, user can select some or all of the risks for his project. Similarly from the available list of mitigation actions, user can select the mitigation actions on the basis of their impact, experienced previously along with other information.

OLAP will perform the analysis on the basis of project similarity (web, desktop, database etc), similarity of different project parameters, and will present the user with more concrete results regarding risks that have been identified previously in the projects. List of risks, with risk status, counter (no of times risk is previously identified/ selected), project phase in which the risk is identified along with other information will be presented to the users on the basis of comparisons with previous projects. System will also allow its user to obtain list of risks on the basis of requested parameter. For example, user can request list of risks in web applications only for those projects, in which .NET technology is used. User can request the risks from the system from different dimensions. Similarly, certain risk may have higher repetition in one type of project as compared to other type of projects. For example, if a user enters the data of a

web project, system will present user with all the risks that are identified previously in web projects, so the main purpose of proposing suggestions on the basis of application similarity is to allow the user decide easily to select a particular risk for his project

V. SUGGESTED ARCHITECTURE FOR THE PROPOSED METHODOLOGY

Suggested architecture contains a single ETL, that will extract data from multiple, distributed sources, and load data in central warehouse. Keeping in mind, the benefits of real time intelligence, Real time data warehouse will be used in the proposed architecture, because when ever user selects a risk or mitigation action, or enter any new risk or mitigation action it will be loaded in the company's own data base and at the same time must be loaded into data warehouse. Results will be displayed through OLAP that would be integrated into the operational systems of the clients. When ever user select any risk or mitigation action or insert a new risk or mitigation action, data warehouse will update the operational data bases about the inserted / selected risks and mitigation actions. Components of proposed architecture are

Source databases: Different heterogeneous data sources

ETL: Tool that will extract, transform, and load the data from heterogeneous data sources into data warehouse and the whole process is called extraction, transformation and load.

Data warehouse data base: Central repository where projects data with risk and mitigation actions information from all data sources will be stored, to make it available for end user analysis.

OLAP: Decision-making tool that will extract data from data warehouse, analyze it in different dimensions, to answer the queries of system's user.

A. Strengths of proposed frame work

- History is the foundation to forecast future, so the proposed system will provide an opportunity to learn from other experiences.
- Heterogeneity of data will be maintained.
- Flexible to provide required information (at detailed and summary level) to the user at request
- Personal or sensitive information of one user will not be accessible to other

B. Weakness

Proposed system is based on real time data warehouse architecture, so it could be a heavily loaded system, as each time insertion is performed in source databases, it also must be performed in data warehouse.

VI. DATA WAREHOUSE DATABASE DESIGN: SNOW FLAKE ARCHITECTURE

Proposed data base design is based on snowflake schema as there are two fact tables, and splitting the dimensions into sub dimensions to support On Line Analytical Processing normalizes dimension tables.

Fact tables: There are two fact tables, **Risks &**

Mitigation actions

Dimensions of Risks are Basic Risk, Owner, Day, Phase. Basic Risk, Owner, Day have sub dimensions

Basic Risk

Sub dimension

- **Application**

Sub dimensions

- **Type**
- **Technology**

Owner

Sub dimension

- **Vendor**

Phase

Day

Sub dimensions

- **Month**
- **Year**

Dimensions of Mitigation actions

1. Basic Mitigation action

2. Day

Sub dimensions

Month

Year

Primary keys: Risk Fact ID, MA Fact ID, MA ID, Risk ID, Application ID, Type ID, Tech ID, Owner ID, Vendor ID, Phase ID, Day ID, Month ID, Year ID

Surrogate Keys: Day ID, Phase ID, Owner ID, Risk ID, MA ID, Application ID

Risk Fact ID is a join key between two fact tables, as one mitigation action can be used to handle more than one risk, and one risk may have more than one mitigation actions against it.

Whenever a user selects a risk or mitigation action from presented results in end user application, its counter will be incremented in dimension tables and whenever a new risk or mitigation action will be inserted from the source databases, this information will be stored in fact and dimension tables.

Some dimensions like application, type, technology, phase, are to be populated before starting the risk / mitigation actions insertion and selection process.

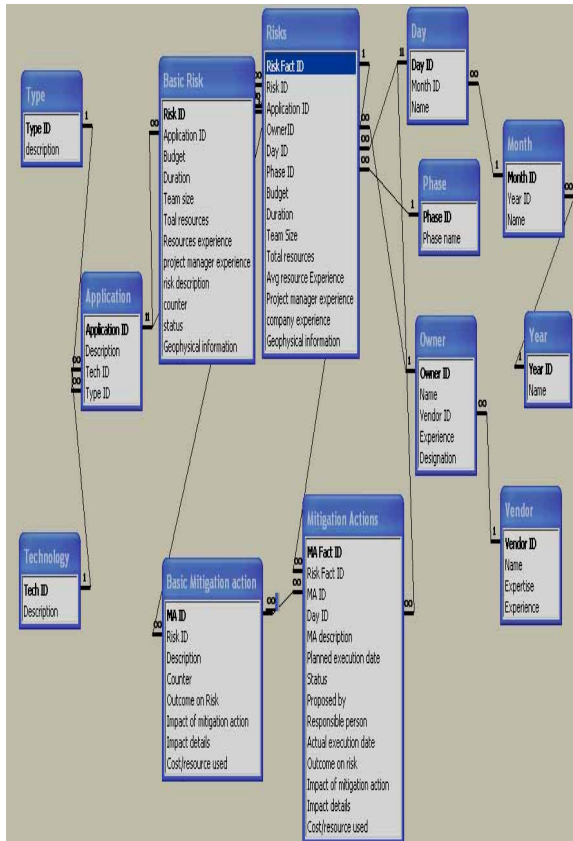


Fig 10: Data base design

VII. CONCLUSION

This research provides helpful information to improve software risk identification and mitigation to enhance the effectiveness of software risk management and lays the foundation for the inception of a risk suggestion system.

It is explored on the basis of industry survey that there exist risks repetition in projects by different companies and certain risks showed higher repetition in particular type of projects on the basis of project (application) similarity. Secondly on the basis of similarity of project parameters, percentage of risk repetition is evaluated and it is also concluded that mitigation actions can also be repeated for repeated risks in these projects. These results are the basis to lay the foundation for the inception of a risk suggestion system to facilitate software risk management by providing suggestions for risk identification and mitigation on the basis of experiences of different companies.

REFERNCES

- [1] Xiangnan Lu. , Yali Ge (2004), "Risk analysis in software development project with Owners and contractors" ", *Engineering Management Conference, 2004 IEEE International Volume 2*, 18-21 Oct. 2004 Page(s):789 – 793.
- [2] Miller J , Gorski J (2004) "Risk identification Patterns for software projects " *Foundations of Computing and Decision Sciences* Vol. 29, No. 1-2, pp. 115-131.
- [3] Schmidt, R., et al., Identifying software project risks (2001): An international Delphi study. *Journal of Management Information Systems*, 2001, 17(4): pp.5-36.
- [4] Powell, P.L., Klein, J.H. 1996. Risk Management for Information Systems Development. *Journal of Information Technology*, Vol 11, 309-319.
- [5] Linda Wallace and Mark Keil, "Software project risks and their affects on Outcomes", April 2004/vol 47 No.4 *Communication of ACM*.
- [6] Keshnee Padayachee (2002), "An Interpretive Study of Software Risk Management Perspectives", *ACM International Conference Proceeding Series*; Vol. 30, *Proceedings of SAICSIT 2002*, Pages 118 –127.
- [7] Stephen P. Masticola (2007), A Simple Estimate of the Cost of Software Project and the Breakeven effectiveness of Project risk management, 29th International Conference on Software Engineering Workshops (ICSEW'07), IEEE.
- [8] Tom Addison, Seema Vallabh (2002) Controlling Software Project Risks – an Empirical Study of Methods used by Experienced Project Managers", *Proceedings of SAICSIT 2002*, Pages 128 – 140.
- [9] Ernie Jordan and Luke Silcock (2005), *Beating IT risks: John Wiley & Sons ltd*.
- [10] Jiang, J., Klein, G. (2000), Software development risks to project effectiveness, *The Journal of Systems and Software*, Vol 52, No 1, pp. 3-10.
- [11] Paul L Bannerman (2007) , software project risks in public sector, *Proceedings of the 2007 Australian Software Engineering Conference (ASWEC'07)*, 389-398.



Sana Khan has done her MS in Computer Science with specialization in Software Engineering from COMSATS institute of information technology Abbottabad, Pakistan. She is currently working in COMSATS information technology centre and practicing Requirement engineering and Software testing. Her interest is in software quality assurance, Requirement engineering, and Empirical Software Engineering.