Journal of Computing and Information Technology - CIT 23, 2015, 2, 123–139
doi:10.2498/cit.1002457

Modelling and Evaluating Software Project Risks with Quantitative Analysis Techniques in Planning Software Development

Abdelrafe Elzamly[1,2] and Burairah Hussin[1]

[1]Information and Communication Technology, Universiti Teknikal Malaysia Melaka (UTeM), Malaysia
[2]Department of Computer Science, Faculty of Applied Sciences, Al-Aqsa University, Gaza, Palestine

Risk is not always avoidable, but it is controllable. The aim of this paper is to present new techniques which use the stepwise regression analysis to model and evaluate the risks in planning software development and reducing risk with software process improvement. Top ten software risk factors in planning software development phase and thirty control factors were presented to respondents. This study incorporates risk management approach and planning software development to mitigate software project failure. Performed techniques used stepwise regression analysis models to compare the controls to each of the risk planning software development factors, in order to determine and evaluate if they are effective in mitigating the occurrence of each risk planning factor and, finally, to select the optimal model. Also, top ten risk planning software development factors were mitigated by using control factors. The study has been conducted on a group of software project managers. Successful project risk management will greatly improve the probability of project success.

Keywords: software project management, risk management, planning software development, software risk factors, risk management techniques, stepwise regression analysis techniques, quantitative techniques

1. Introduction

Despite much research and progress in the area of software project management, software development projects still fail to deliver acceptable systems on time and within budget. Much of the failure could be avoided by managers' pro-active maintenance and dealing with risk factors rather than waiting for problems to occur and then trying to react. Project management and risk management have been proposed as a solution to preserve the quality and integrity of a project by reducing cost escalation [1]. Due to the involvement of risk management in monitoring the success of a software project, analyzing potential risks, and making decisions about what to do about potential risks, the risk management is considered the planned control of risk. Integrating formal risk management with project management is a new phenomenon in software engineering and product management community. In addition, risk is an uncertainty that can have a negative or positive effect on meeting project objectives. Risk management is the process of identifying, analyzing and controlling risk throughout the life of a project to meet the project objectives.

However, an intelligent performance analysis approach is adapted for decision making to select the optimization techniques to apply in real world problem solving approach particularly related to industrial engineering problems [2]. In addition, the goal of risk management is early identification and recognition of risks and then actively changing the course of actions to mitigate and reduce the risk [3]. In the process of understanding the factors that contribute to software project success, risk is becoming increasingly important. This is a result of the size, complexity and strategic importance of many of the information systems currently being developed. Today, we must think of risk as a part of software project lifecycle which is important for a software project survival. On the other hand, risk management aims to read risks as improvement opportunities and provide inputs to growth plans [4].

In our paper, we identified software planning risk factors and risk management techniques that guide software project managers to understand and mitigate risks in software analysis development projects. However, Software Development Life Cycle, according to [5], is the process of creating or altering systems, and the models and methodologies that people use to develop these systems. Also, it includes the following phases: planning, analysis, design, implementation, and maintenance. In addition, we focused on the planning phase. During this phase, the group that is responsible for creating the system must first determine what the system needs to do for the organization (new requirements gathering) and evaluate existing system/software. Risk management is a practice of controlling risk and the practice consists of processes, methods and tools for managing risks in a software project before they become problems [6].

This study will guide software managers to apply software risk management practices with real world software development organizations and verify the effectiveness of the modelling techniques on a software project. We hope that the approaches will succeed in the software risk management methodology, which will improve the probability of software project success. **The objectives of this study are:** to identify the software risk factors of planning software development in the Palestinian software development organizations, to rank the software risk factors in planning software development according to their importance, severity and occurrence frequency based on data source, to identify the activities performed by software project managers, to model and evaluate the identified risks in the planning of software development.

According to Taylor, we should apply techniques consistently throughout the software project risk management process [7] Risk management is a practice of controlling risk and the practice consists of processes, methods, and tools for managing risks in a software project before they become problems [6]. Therefore, Boehm talked about value-based risk management, including principles and practices for risk identification, analysis, prioritization, and mitigation [8].

2. Literature Review

Previous studies had shown that risk mitigation in software projects are classified into three categories – namely, qualitative, quantitative and mining approaches. Firstly, quantitative risk is based on statistical methods that deal with accurate measurement about risk or lead to quantitative inputs that help to form a regression model to understand how software project risk factors influence project success. Furthermore, qualitative risk techniques lead to subjective opinions expressed or self-judgment by software manager using techniques, namely scenario analysis, Delphi analysis, brainstorming session, and other subjective approaches to mitigate risks. Elzamly and Hussin [9] improved quality of software projects of the participating companies while estimating the quality – affecting risks in IT software projects. The results show that there were 40 common risks in software projects of IT companies in Palestine. The amount of technical and non-technical difficulties was very large. Khanfar, Elzamly, et al. [10], the new technique used the chi-square (χ^2) test to control the risks in a software project. However, we also used new techniques which are the regression test and effect size test proposed to manage the risks in a software project and reduce risk with software process improvement [11]. Furthermore, we used the new stepwise regression technique to manage the risks in a software project. These tests were performed using regression analysis to compare the controls to each of the risk factors to determine if they are effective in mitigating the occurrence of each risk factor implementation phase [12]. In addition, we proposed the new mining technique that use the fuzzy multiple regression analysis techniques to manage the risks in a software project. However, these mining tests were performed using fuzzy multiple regression analysis techniques to compare the risk management techniques to each of the software risk factors to determine if they are effective in mitigating the occurrence of each software risk factor [13]. Further, the fuzzy regression analysis modelling techniques are used to manage the risks in a planning software development project. Top ten software risk factors in planning phase and thirty risk management techniques were presented to respondents [14]. In addition, we identified and managed the maintenance risks in a software development project by using fuzzy multiple regression analysis [15]. Also, we proposed new mining techniques that use the fuzzy multiple regression analysis techniques with fuzzy concepts to manage the software risks in a software project. Top ten software risk factors in analysis phase and thirty risk management techniques were presented to respondents. However, these mining tests were performed using fuzzy multiple regression analysis techniques to compare the risk management techniques with each of the software risk factors in order to determine if they are effective in reducing the occurrence of each software risk factor [16]. Also, the paper aimed to present a new mining technique to identify the risk management techniques that are effective in reducing the occurrence of each software implementation risk [17]. In addition, we proposed the new quantitative and mining techniques to compare the risk management techniques to each of the software maintenance risks in order to identify and model if they are effective in mitigating the occurrence of each software maintenance risk in software development life cycle [18]. Furthermore, we presented the stepwise multiple regression analysis technique and Durbin Watson technique to reduce software maintenance risks in a software project [19].

The authors continue the effort to enrich the managing software project risks considering mining and quantitative approach with large data set. Two techniques are introduced, namely stepwise multiple regression analysis and fuzzy multiple regression to manage the software risks [20]. This paper aims to present new techniques to determine if fuzzy and stepwise regression are effective in mitigating the occurrence of software risk factor in the implementation phase [21]. Finally, risk management methodology that has five phases: risk identification (planning, identification, prioritization), risk analysis and evaluation (risk analysis, risk evaluation), risk treatment, risk controlling, risk communication and documentation relied on three categories or techniques such as risk qualitative analysis, risk quantitative analysis and risk mining analysis throughout the life of a software project to meet the goals [22]. Although there are many methods in software risk management, software development projects have a high rate of risk failure. Thus, if the complexity and size of the software projects are increased, managing software development risk becomes more difficult [23]. There are several software risk management approaches, models, and frameworks according to a literature review.

3. Top 10 Software Risk Factors in Planning Software Development Phase

We displayed the top ten software risk factors in planning software development project lifecycle that is most commonly used by researchers when studying the risk in software projects. However, the list consists of the 10 most serious risks to a project ranked from one to ten, each risk's status, and the plan for addressing each risk. These factors need to be addressed and thereafter they need to be controlled. Consequently, we presented 'top-ten' based on survey Boehm's 10 risk items 1991 on software risk management [24], the top 10 risk items according to a survey of experienced project managers, Boehm's 10 risk items 2002 and Boehm's 10 risk items 2006-2007, Miler [25], The Standish Group survey [26], Addison and Vallabh [27], Addison [28], Khanfar, Elzamly et al. [10], Elzamly and Hussin [11], Elzamly and Hussin [9], Aloini et al. [29], Han and Huang [30] [31], Aritua et al. [32], Schmidt et al. [33], Mark Keil et al. [34], Nakatsu and Iacovou [35], Chen and Huang [36], Mark Keil et al. [37], Wallace et al. [38], Sumner [39], Boehem and Ross [40], Ewusi-Mensah [41], Pare et al. [42], Houston et al. [43], Lawrence et al. [44], Shafer and Officer [45], hoodat and Rashidi [23], Jones et al. [46], Jones [47], Taimour [48], and other scholars, researchers in software engineering, to obtain software risk factors and risk management techniques. These software project risks are illustrated in Table 1.

4. Risk Management Techniques

Through reading the existing literature on software risk management approach and methodology, we listed thirty control factors that are considered important in reducing and modeling

Dimension	No	Software risk factors	Frequency
Planning software development	1	Low key user involvement	14
	2	Unrealistic schedules and budgets	14
	3	Unclear / misunderstood / unrealistic / change scope and objectives (goals)	8
	4	Insufficient/inappropriate staffing	8
	5	Lack of senior management commitment and technical leadership	8
	6	Poor /inadequate planning and strategic thinking	7
	7	Lack of effective software project management methodology	6
	8	Change in organizational management during the software project	5
	9	Ineffective communication software project system	3
	10	Absence of historical data (templates)	2
	Total frequency		**75**

Table 1. Illustrates top ten software risk factors in software projects according to researchers.

the software risk factors identified in planning software development; these controls are:

C1: Using of requirements scrubbing, C2: Stabilizing requirements and specifications as early as possible, C3: Assessing cost and scheduling the impact of each change on requirements and specifications, C4: Developing prototyping and having the requirements reviewed by the client, C5: Developing and adhering a software project plan, C6: Implementing and following a communication plan, C7: Developing contingency plans to cope with staffing problems, C8: Assigning responsibilities to team members and rotate jobs, C9: Having team-building sessions, C10: Reviewing and communicating progress to date and setting objectives for the next phase, C11: Dividing the software project into controllable portions, C12: Reusable source code and interface methods, C13: Reusable test plans and test cases, C14: Reusable database and data mining structures, C15: Reusable user documents early, C16: Implementing/utilizing automated version control tools, C17: Implementing/utilizing benchmarking and tools of technical analysis, C18: Creating and analyzing process by simulation and modeling, C19: Providing scenarios and methods and using the reference checking, C20: Involving management during the entire software project lifecycle, C21: Including formal and periodic risk assessment, C22: Utilizing change control board and exercising quality change control practices, C23: Educating users on the impact of changes during the software project, C24: Ensuring quality-factor deliverables and task analysis, C25: Avoiding having too many new functions on software projects, C26: Incremental development (deferring changes to later increments), C27: Combining internal evaluations by external reviews, C28: Maintaining proper documentation of each individual's work, C29: Providing training in the new technology and organizing domain knowledge training, C30: Participating of users during the entire software project lifecycle.

5. Empirical Strategy

Data collection was achieved through the use of a structured questionnaire and historical data to assist in estimating the quality of software through determining risks that were common to the majority of software projects in the analyzed software companies. Top ten software risk factors and thirty control factors were presented to respondents. The method of sample selection referred to as 'snowball' and distribution of personal regular sampling was used. This procedure is appropriate when members of homogeneous groups (such as software project managers, IT managers) are difficult to locate. Seventy six software project managers participated in this study. The project managers that participated in this survey came from specific, mainly software project management in software development organizations.

Respondents were presented with various questions, which used scales 1–7. For presentation purposes in this paper and for effectiveness, the point scale is as follows: For choices being labeled 'unimportant' equals one, and 'extremely important' equals seven. Similarly, seven frequency categories were scaled into 'never' equals one and 'always' equals seven. All questions in software risk factors were measured on a seven-point Likert scale from unimportant to extremely important and software control factors were measured on a seven-point Likert scale from never to always. However, to describe "Software Development Company in Palestine" that has in-house development software and a supplier of software for local or international market, we depended on Palestinian Information Technology Association (PITA) Members' webpage at PITA's website [PITA 2012 `www.pita.ps/`], Palestinian investment promotion agency [PIPA 2012 `http://www.pipa.gov.ps/`] to select top IT managers and software project managers. In order to find the relation among risks that the software projects confront and the counter measures that should be taken to reduce risks, many researchers used different statistical methods. In this paper, we used correlation analysis and regression analysis models based on stepwise selection method and Durbin-Watson Statistic. In general, the software risk management methodology includes five phases named as risk identification, risk analysis and evaluation, risk treatment, risk controlling, risk communication and documentation that contribute to the success of any undertaking software project. We started our risk management methodology by risk identification of all possible software risks in planning software development. Many risks possibilities are taken from the past literature; however, we take into consideration only the top ten risk softwares for planning software development based upon previous work. At the same time, we also identify the possible risk management techniques from the past literature. Also we finished with 30 risk management techniques that come from software project risk management involving risk analysis and evaluation, risk treatment, risk controlling, risk communication and documentation, in order to incorporate the quantitative approach and software risk management methodology to mitigate planning software risks.

5.1. Correlation Analysis

Clearly, the preceding analysis states that there are correlations between determining variables besides correlation between risk factors and all determining control factors. However, the equation of Correlation Coefficient is the following:

$$r=\frac{n\left[\sum(x_i,y_i)\right]-\left(\sum x_i\right)\left(\sum y_i\right)}{\sqrt{\left[n\left(\sum x_i^2\right)-\left(\sum x_i\right)^2\right]\left[n\left(\sum y_i^2\right)-\left(\sum y_i\right)^2\right]}} \tag{1}$$

5.2. Regression Analysis Model

Regression modeling is one of the most widely used statistical modeling techniques for fitting a response (dependent) variable Y as a function of predictor (independent) variables X_i (multiple regression).

$$Y = \beta_0 + \beta_1 X_1 + \ldots + \beta_n X_n + \varepsilon \tag{2a}$$

Indeed, software risk factor is a dependent variable while control factors are independent variables. A linear equation between one dependent and many independent variables may be expressed as:

$$\hat{Y} = b_0 + b_1 X_1 + b_2 X_2 + \ldots + b_n X_n \tag{2b}$$

where $b_0, b_1, b_2, \ldots$ and b_n are regression coefficients; $X_1, X_2, \ldots$ and X_n are the independent variables, and Y is the dependent variable. The values of $b_0, b_1, b_2, \ldots$ and b_n of the multiple regression equation may be obtained by solving the linear equations system [49].

5.3. Stepwise Regression (Adds and Removes Variables)

According to [50], [51], stepwise regression method (SRM) combines and alternates between forward selection and backward elimination. At each step, the best remaining variable is added, provided it passes the 5% significance level, and then all variables currently in the regression are checked to see if any can be removed, using the greater than 10% significance criterion. In addition [51], the MSRA method is a stepwise optimization process of the multiple regression analysis method. Also, a stepwise-regression method is applied which systematically adds and removes model components based on statistical test to automatically identify the risks for a large scale data in operation [52]. Therefore [50], SRM is particularly useful when we need to predict a dependent variable from a (very) large set of independent variables.

5.4. Coefficient of Determination

Coefficient of determination (R^2) is the proportion of variation in the observed values of the response variable Y that is explained by the regression $\hat{Y}$ [49]:

$$R^2 = \frac{\text{RSS}}{\text{TSS}} = \frac{\sum(\hat{y} - y_{avg})^2}{\sum(y - y_{avg})^2} \quad (3)$$

According to [49], regression sum of squares (RSS) is the variation in the observed values of the response variable Y that is explained by the regression $\hat{Y}$, while total sum of squares (TSS) is the variation in the observed values of the response variable Y.

5.5. Durbin-Watson Statistic (D)

Durbin-Watson statistic is an index that tests for autocorrelation (the relationship between values separated from each other by a given time lag) in the residuals (prediction errors) from a statistical regression analysis (`http://www.investopedia.com/terms/d/durbin-watson-statistic.asp/2013/2/26`). Consequently, we will avoid using independent variables that have errors with a strong positive or negative autocorrelation, because this can lead to an incorrect forecast for the dependent variable. However, the value D always lies between 0 and 4 the defined D-W statistic as:

$$D = \frac{\sum(e_i - e_{i-1})^2}{\sum e_i^2}, \quad \text{for } N \text{ and } K-1 \text{ df} \quad (4)$$

where N is the number of observations.

5.6. Importance of Risk Factors in Planning Software Development Phase

All respondents indicated that the software risks of "ineffective communication software project system" were the highest software risk factors and important ones. In fact, the risk factors sorted in descending order of respective means were identified as important which resulted in the following ranking of importance of the listed risks (in order of importance): Risk 9, Risk 10, Risk 3, Risk 1, Risk 6, Risk 8, Risk 7, Risk 2, Risk 4, Risk 5.

Risk	N	Mean	Std. Deviation	%
R9	76	3.934	0.806	78.684
R10	76	3.868	0.806	77.368
R3	76	3.842	0.801	76.842
R1	76	3.803	0.749	76.053
R6	76	3.789	0.736	75.789
R8	76	3.711	0.877	74.211
R7	76	3.697	0.766	73.947
R2	76	3.684	0.716	73.684
R4	76	3.658	0.946	73.158
R5	76	3.618	0.848	72.368
Total	76	3.761	0.543	75.211

Table 2. Mean score for each risk factor (planning software development).

5.7. Ranking of Importance of Risk Factors for Project Managers' Experience

Table 3 shows the overall ranking of importance of each planning risk factor for the three categories of project managers' experience.

Phase	Risk	Experience 2-5 years	Experience 6-10 years	Experience > 10 years
Planning software development	R1	R9	R3	R10
	R2	R1	R10	R1
	R3	R6	R9	R5
	R4	R10	R6	R3
	R5	R3	R1	R9
	R6	R8	R8	R7
	R7	R5	R7	R6
	R8	R2	R2	R8
	R9	R4	R4	R4
	R10	R7	R5	R2

Table 3. The overall risk ranking of each risk factor.

5.8. Frequency of Occurrence of Controls

Table 4 shows the mean and the standard deviation for each control factor. The results of this paper show that most of the controls are used most of the time and rather often.

Control	Mean	Std. Deviation	% percent
C29	4.408	0.803	88.15789
C30	4.368	0.907	87.36842
C20	4.184	0.668	83.68421
C27	4.171	0.755	83.42105
C21	4.171	0.7	83.42105
C19	4.158	0.612	83.15789
C28	4.158	0.767	83.15789
C25	4.132	0.718	82.63158
C26	4.118	0.653	82.36842
C23	4.105	0.741	82.10526
—	—	—	—
C13	3.868	0.754	77.36842

Table 4. Mean score for each control factor.

5.9. Relationships Between Risks and Control Variables

Regression technique was performed on the data to determine whether there were significant relationships between control factors and risk factors. These tests were performed using stepwise regression analysis model to compare the controls to each of the risk planning software development factors to determine and evaluate if they are effective in mitigating the occurrence of each risk factor. Significant relationships between risks and controls, being important for the optimal models, are presented in the continuation. This study presents the model for software risk management within planning software development process.

R1: Risk of 'Low Key User Involvement' compared to controls.

Table 5, Table 6, Table 7 and Table 8 show that the obtained significant values (Sig.) are all less than the selected significant level of

C1	C2	C3	C5	C6
.336**	.281*	.283*	.433**	.524**
C10	C7	C8	C9	C11
.373**	.323**	.438**	.460**	.384**
C12	C14	C15	C19	C20
.271*	.250*	.264*	.251*	.309**
C21	C22	C24		
.443**	.370**	.285*		

*Correlation is significant at the 0.05 level (2-tailed).
**Correlation is significant at the 0.01 level (2-tailed).

Table 5. Illustrates correlations between respective controls and R1.

Model	R	R Square	Durbin-Watson
1	.524[a]	.275	
2	.591[b]	.349	1.729

a. Predictors: (constant), C6
b. Predictors: (constant), C6, C21

Table 6. Illustrates multiple correlations R, and R square.

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	12.469	1	12.469	28.000	.000[a]
	Residual	32.952	74	.445		
	Total	45.421	75			
2	Regression	15.856	2	7.928	19.576	.000[b]
	Residual	29.565	73	.405		
	Total	45.421	75			

a. Predictors: (constant), C6 b. Predictors: (constant), C6, C21
c. Dependent variable: R1

Table 7. Illustrates an Analysis of Variance (ANOVA[c]).

Model		Unstandardized Coefficients	Standardized Coefficients	t	Sig.
		b	Beta		
1	(constant)	2.429		5.307	.000
	C6	.476	.524	5.292	.000
2	(constant)	1.369		2.403	.019
	C6	.381	.419	4.140	.000
	C21	.295	.293	2.892	.005

Dependent variable: R1

Table 8. Illustrates the model coefficients, respective t values and their significance (coefficients[a], coefficients[b]).

$\alpha = 0.05$, so there is a positive correlation between controls 1, 2, 3, 5, 6, 7, 8, 9, 10, 11, 12, 14, 15, 19, 20, 21, 22, 24 and Risk 1 (Table 5). However, the results show that Controls 6 and 21 have a positive impact value of $r_6 = 0.524$ and $r_{21} = 0.443$ respectively with Risk 1. The multiple correlation R is 0.591, the value of R^2 is 0.349 in the best model (stable model, Tables 6 and 7). This means that the Model 2 explained 34.9 % of the variability of dependent variable Risk 1. Furthermore, the Durbin-Watson statistic (D) is 1.729 and the table gives the critical values based on K = 2 (regressors), N = 76, $\alpha = 0.05$ ($d_U = 1.680$, $d_L = 1.571$); there is evidence of no autocorrelation ($d_U < D < 2 + d_L$: No autocorrelation). However, we will avoid independent variables that have errors with a strong positive and negative correlation in the stepwise multiple regression model, because this can lead to an incorrect prediction based on independent variables.

R2: Risk of 'Unrealistic Schedules and Budgets' compared to controls.

Table 9, Table 10, Table 11, and Table 12 show that the obtained significant values are less than the selected significant value of $\alpha = 0.05$, so there is a positive correlation between Controls 1, 3, 5, 6, 7, 8, 9, 10, 11, 24, 25, and Risk 2, respectively. In addition, Control 1 has an impact on Risk 2, and the results show that Control 1 has a positive impact value of R=0.389 and the value of R^2 is 0.151. This means that the model (Table 10) explained 15.1% of the variability of response Risk 2. Furthermore, Durbin-Watson statistic (D) is 2.227, and $d_U = 1.652$, $d_L = 1.598$ based on K = 1, N = 76, $\alpha = 0.05$; there is evidence of no autocorrelation ($d_U < D < 2 + d_L$: No autocorrelation).

C1	C3	C5	C6	C7	C8
.389**	.297**	.330**	.355**	.243*	.229*
C9	C10	C11	C24	C25	
.232*	.349**	.251*	.331**	.238*	

*Correlation is significant at the 0.05 level (2-tailed).
**Correlation is significant at the 0.01 level (2-tailed).

Table 9. Illustrates correlations between respective controls and R2.

Model	R	R Square	Durbin-Watson
1	.389[a]	.151	2.227

a. Predictors: (constant), C1

Table 10. Illustrates multiple correlation R, and R square.

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	5.817	1	5.817	13.204	.001[a]
	Residual	32.604	74	.441		
	Total	38.421	75			

a. Predictors: (constant), C1
b. Dependent variable: R2

Table 11. Illustrates an Analysis of Variance (ANOVA[b]).

Model		Unstandardized Coefficients	Standardized Coefficients	t	Sig.
		b	Beta		
1	(constant)	2.844		5.553	.000
	C1	.376	.389	3.634	.001

a. Dependent variable: R2

Table 12. Illustrates the model coefficients, respective t values and their significance (coefficients[a]).

R3: Risk of 'Misunderstood / Unrealistic Scope and Objectives (Goals)' compared to controls.

Table 13, Table 14, Table 15, and Table 16 show that the obtained significant values are less than the selected significant value of $\alpha = 0.05$, so there is a positive correlation between Controls 1, 6, 10, 19, 30, and Risk 3, respectively. In addition, Control 6 has an impact on Risk 3, and

C1	C6	C10	C19	C30
.254*	.264*	.247*	.264*	.235*

*Correlation is significant at the 0.05 level (2-tailed).

Table 13. Illustrates correlations between respective controls and R3.

Model	R	R Square	Durbin-Watson
1	.264[a]	.070	1.812

a. Predictors: (constant), C6

Table 14. Illustrates multiple correlation R, and R square.

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	4.702	1	4.702	5.538	.021[a]
	Residual	62.825	74	.849		
	Total	67.526	75			

a. Predictors: (constant), C6 b. Dependent variable: R3

Table 15. Illustrates an Analysis of Variance (ANOVA[b]).

Model		Unstandardized Coefficients	Standardized Coefficients	t	Sig.
		b	Beta		
1	(constant)	3.455		5.469	.000
	C6	.292	.264	3.634	.021

a. Dependent variable: R3

Table 16. Illustrates the model coefficients, respective t values and their significance (coefficients[a]).

the results show that Control 6 has a positive impact value of R=0.264 and the value of R^2 is 0.070. This means that the model (Table 14) explained 7.0% of the variability of response Risk 3. Furthermore, Durbin-Watson statistic (D) is 1.812, and ($d_U = 1.652$, $d_L = 1.598$) based on K = 1, N = 76, $\alpha = 0.05$; there is evidence of no autocorrelation because of the rule ($d_U < D < 2 + d_L$: No autocorrelation).

R4: Risk of 'Insufficient / Inappropriate Staffing' compared to controls.

Table 17, Table 18, Table 19, and Table 20 show that the obtained significant values are less than the selected significant value of $\alpha = 0.05$, so there is a positive correlation between Controls 1, 3, 5, 6, 7, 8, 9, 10, 11, 24, 28, and Risk 4, respectively. In addition, Control 6 has an impact on Risk 4, and the results show that Control 6 has a positive impact value of $R = 0.374$ and the value of R^2 is 0.140. This means that the model (Table 18) explained 14.0% of the variability of response Risk 4. Furthermore, Durbin-Watson statistic (D) is 1.745, and ($d_U = 1.652$, $d_L = 1.598$) based on K = 1, N = 76, $\alpha = 0.05$; there is evidence of no autocorrelation because of the rule ($d_U < D < 2 + d_L$: No autocorrelation).

C1	C3	C5	C6	C7	C8
.285*	.266*	.313**	.374**	.247*	.291*
C9	C10	C11	C24	C28	
.276*	.309**	.249*	.225*	.263*	

Table 17. Illustrates correlations between respective controls and R4.

Model	R	R Square	Durbin-Watson
1	.374[a]	.140	1.745

a. Predictors: (constant), C6

Table 18. Illustrates multiple correlation R, and R square.

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	9.898	1	9.898	12.031	.001[a]
	Residual	60.879	74	.823		
	Total	70.776	75			

a. Predictors: (constant), C6 b. Dependent variable: R4

Table 19. Illustrates an Analysis of Variance (ANOVA[b]).

Model		Unstandardized Coefficients	Standardized Coefficients	t	Sig.
		b	Beta		
1	(constant)	2.544		4.091	.000
	C6	.424	.374	3.634	.001

a . Dependent variable: R4

Table 20. Illustrates the model coefficients, respective t values and their significance (coefficients[a]).

R5: Risk of 'Lack of Senior Management Commitment and Technical Leadership' compared to controls.

Table 21, Table 22, Table 23, and Table 24 show that the obtained significant values are less than the selected significant value of $\alpha = 0.05$, so there is a positive correlation between Controls 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 16, 18, 19, 20, 21, 22, 24, 25, 28, and Risk 5, respectively. Controls 6 and 16 have an impact on Risk 5. In addition, the results show that Controls 6, 16 have a positive impact value of 0.433, 0.329 respectively, multiple correlation is R=0.498 and value of R^2 is 0.248. This means that the model (Table 22) explained 24.8% of the variability of response Risk 5. Furthermore, Durbin-Watson statistic is 2.427, and ($d_U = 1.680$, $d_L = 1.571$)

C1	C2	C3	C4	C5
.370**	.272*	.309**	.233*	.390**
C6	C7	C8	C9	C10
.433**	.293*	.293*	.308**	.307**
C11	C16	C18	C19	C20
.277*	.329**	.254*	.256*	.231*
C21	C22	C24	C25	C28
.294*	.243*	.232*	.286*	.283*

Table 21. Illustrates correlations between respective controls and R5.

Model	R	R Square	Durbin-Watson
1	.433[a]	.187	
2	.498[b]	.248	2.427

a. Predictors: (constant), C6
b. Predictors: (constant), C6, C16

Table 22. Illustrates multiple correlations R, and R square.

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	10.799	1	10.799	17.045	.000[a]
	Residual	46.885	74	.634		
	Total	57.684	75			
2	Regression	14.311	2	7.156	12.043	.000[b]
	Residual	43.373	73	.594		
	Total	57.684	75			

a. Predictors: (constant), C6
b. Predictors: (constant), C6, C16
c. Dependent variable: R5

Table 23. Illustrates an Analysis of Variance (ANOVA[c]).

Model		Unstandardized Coefficients	Standardized Coefficients	t	Sig.
		b	Beta		
1	(constant)	2.410		4.415	.000
	C6	.443	.443	4.129	.000
2	(constant)	1.077		1.414	.162
	C6	.391	.382	3.683	.000
	C16	.319	.252	2.431	.018

Dependent variable: R5

Table 24. Illustrates the model coefficients, respective t values and their significance (coefficients[a], coefficients[b]).

based on K = 2, N = 76, $\alpha = 0.05$; there is evidence of no autocorrelation ($d_U < D < 2+d_L$: No autocorrelation).

R6: Risk of 'Poor /Inadequate Software Project Planning and Strategic Thinking' compared to controls.

Table 25, Table 26, Table 27, and Table 28 show that the obtained significant values are less than the selected significant value of $\alpha = 0.05$, so there is a positive correlation between Controls 3, 4, 5, 6, 14, and Risk 6, respectively. Controls 5, 14, and 27 have an impact on Risk 6. In addition, the results show that Controls 5, 14

C3	C4	C5	C6	C14
.267*	.249*	.268*	.228*	.254**

Table 25. Illustrates correlations between respective controls and R6.

Model	R	R Square	Durbin-Watson
1	.268[a]	.072	
2	.351[b]	.123	
3	.423[c]	.179	2.006

a. Predictors: (constant), C5 b. Predictors: (constant), C5, C14
c. Predictors: (constant), C5, C14, C27

Table 26. Illustrates multiple correlations R, and R square.

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	3.170	1	3.170	5.740	.019[a]
	Residual	40.869	74	.552		
	Total	44.039	75			
2	Regression	5.411	2	2.705	5.112	.008[b]
	Residual	38.629	73	.529		
	Total	44.039	75			
3	Regression	7.884	3	2.628	5.234	.003[c]
	Residual	36.155	72	.502		
	Total	44.039	75			

a. Predictors: (constant), C5 b. Predictors: (constant), C5, C14
c. Predictors: (constant), C5, C14, C27 d. Dependent variable: R6

Table 27. Illustrates an Analysis of Variance (ANOVA[d]).

Model		Unstandardized Coefficients	Standardized Coefficients	t	Sig.
		b	Beta		
1	(constant)	3.499		6.352	.000
	C5	.256	.268	2.396	.019
2	(constant)	2.394		3.145	.002
	C5	.232	.243	2.201	.031
	C14	.245	.227	2.058	.043
3	(constant)	2.715		3.594	.001
	C5	.292	.306	2.750	.008
	C14	.377	.350	2.894	.005
	C27	-.247	-.277	-2.219	.030

Dependent variable: R6

Table 28. Illustrates the model coefficients, respective t values and their significance (coefficients[a], coefficients[b], coefficients[c]).

have a positive impact value of 0.268 and 0.254, respectively, multiple correlation is R=0.423 and the value of R^2 is 0.179. This means that the model (Table 26) explained 17.9% of the variability of response Risk 6. Also, Durbin-Watson statistic (D) is 2.006, and ($d_U = 1.709$, $d_L = 1.543$) based on $K = 3$, $N = 76$, $\alpha = 0.05$; there is evidence of no autocorrelation ($d_U < D < 2 + d_L$: No autocorrelation).

R7: Risk of 'Lack of an Effective Software Project Management Methodology' compared to controls.

Table 29, Table 30, Table 31, and Table 32 show that the obtained significant values are less than the selected significant value of $\alpha = 0.05$, so there is a positive correlation between Controls 24, 25 and Risk 7, respectively. In addition, Control 24 has an impact on Risk 7, and the results show that Control 24 has a positive impact value of R=0.394 and the value of R^2 is 0.155. This means that the model (Table 30) explained 15.5% of the variability of response Risk 7. Furthermore, Durbin-Watson statistic (D) is 1.933, and ($d_U = 1.652$, $d_L = 1.598$) based on $K = 1$, $N = 76$, $\alpha = 0.05$; there is evidence of no autocorrelation because of the rule ($d_U < D < 2 + d_L$: No autocorrelation).

r	C24	C25
R7	.394**	.294*

Table 29. Illustrates correlations between respective controls and R7.

Model	R	R Square	Durbin-Watson
1	.394[a]	.155	1.933

a. Predictors: (constant), C24

Table 30. Illustrates multiple correlation R, and R square.

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	7.387	1	7.387	13.582	.000[a]
	Residual	40.245	74	.544		
	Total	47.632	75			

a. Predictors: (constant), C24 b. Dependent variable: R7

Table 31. Illustrates an Analysis of Variance (ANOVA[b]).

Model		Unstandardized Coefficients	Standardized Coefficients	t	Sig.
		b	Beta		
1	(constant)	2.327		3.566	.001
	c24	.469	.394	3.685	.000

a. Dependent variable: R7

Table 32. Illustrates the model coefficients, respective t values and their significance (coefficients[a]).

R8: Risk of 'Change in Organizational Management During the Software Project' compared to Controls.

R	C17
R8	.255*

Table 33. Illustrates correlations between respective controls and R8.

Model	R	R Square	Durbin-Watson
1	.255[a]	.065	
2	.374[b]	.140	
3	.444[c]	.197	
4	.496[d]	.246	1.883

a. Predictors: (constant), C17 b. Predictors: (constant), C17, C27
c. Predictors: (constant), C17, C27, C25
d. Predictors: (constant), C17, C27, C25, C6

Table 34. Illustrates multiple correlations R, and R square.

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	3.760	1	3.760	5.165	.026[a]
	Residual	53.872	74	.728		
	Total	57.632	75			
2	Regression	8.075	2	4.037	5.947	.004[b]
	Residual	49.557	73	.679		
	Total	57.632	75			
3	Regression	11.368	3	3.789	5.898	.001[c]
	Residual	46.263	72	.643		
	Total	57.632	75			
4	Regression	14.198	4	3.550	5.803	.000[d]
	Residual	43.433	71	.612		
	Total	57.632	75			

a. Predictors: (constant), C17
b. Predictors: (constant), C17, C27
c. Predictors: (constant), C17, C27, C25
d. Predictors: (constant), C17, C27, C25, C6
e. Dependent variable: R8

Table 35. Illustrates an Analysis of Variance (ANOVA[e]).

Model		Unstandardized Coefficients	Standardized Coefficients	t	Sig.
		b	Beta		
1	(constant)	3.269		5.093	.000
	C17	.282	.255	2.273	.026
2	(constant)	4.268		5.802	.000
	C17	.389	.352	3.060	.003
	C27	-.295	-.290	-2.521	.014
3	(constant)	3.457		4.320	.000
	C17	.344	.311	2.741	.008
	C27	-.397	-.391	-3.243	.002
	C25	.306	.268	2.264	.027
4	(constant)	2.723		3.196	.002
	C17	.320	.290	2.608	.011
	C27	-.441	-.434	-3.637	.001
	C25	.287	.251	2.166	.034
	C6	.236	.230	2.151	.035

Dependent variable: R8

Table 36. Illustrates the model coefficients, respective t values and their significance (coefficients[a], coefficients[b], coefficients[c], coefficients[d]).

Table 33, Table 34, Table 35, and Table 36 show that the obtained significant values are less than the selected significant value of $\alpha = 0.05$, so there is a positive correlation between Control 17 and Risk 8. In addition, Controls 6, 17, 25, and 27 have an impact on Risk 8, multiple correlation value R=0.496, and the value of R^2 is 0.246. This means that the model (Table 34) explained 24.6% of the variability of response Risk 8. Furthermore, Durbin-Watson statistic (D) is 1.883, and ($d_U = 1.543$, $d_L = 1.709$) based on K = 4, N = 76, $\alpha = 0.05$; there is evidence of no autocorrelation ($d_U < D < 2+d_L$: No autocorrelation).

R9: Risk of 'Ineffective Communication Software Project System' compared to Controls.

Table 37, Table 38, Table 39, and Table 40 show that the obtained significant values are less than the selected significant value of $\alpha = 0.05$, so there is a positive correlation between Controls 24, 27, 25, 5, and Risk 9. Controls 24, 27, 25, and 5 have an impact on the Risk 9. In addition, multiple correlation value R=0.500, and the value of R^2 is 0.250. This means that the model (Table 38) explained 25.0% of the variability of response Risk 9. Furthermore, Durbin-Watson statistic (D) is 1.687, and ($d_U = 1.739$, $d_L = 1.515$) based on K = 4, N = 76, $\alpha = 0.05$; there is evidence of inconclusive ($d_L < D < d_U$: Inconclusive).

R	C24	C26
R9	.272*	.233*

Table 37. Illustrates correlations between respective controls and R9.

Model	R	R Square	Durbin-Watson
1	.272[a]	.074	
2	.387[b]	.150	
3	.448[c]	.201	
4	.500[d]	.250	1.687

a. Predictors: (constant), C24 b. Predictors: (constant), C24, C27
c. Predictors: (constant), C24, C27, C25
d. Predictors: (constant), C24, C27, C25, C5

Table 38. Illustrates multiple correlations R, and R square.

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	3.819	1	3.819	5.892	.018[a]
	Residual	47.970	74	.648		
	Total	51.789	75			
2	Regression	7.772	2	3.886	6.445	.003[b]
	Residual	44.017	73	.603		
	Total	51.789	75			
3	Regression	10.402	3	3.467	6.032	.001[c]
	Residual	41.378	72	.575		
	Total	51.789	75			
4	Regression	12.951	4	3.238	5.919	.000[d]
	Residual	38.838	71	.547		
	Total	51.789	75			

a. Predictors: (constant), C24 b. Predictors: (constant), C24, C27
c. Predictors: (constant), C24, C27, C25
d. Predictors: (constant), C24, C27, C25, C5 e. Dependent variable: R9

Table 39. Illustrates an Analysis of Variance (ANOVA[e]).

Model		Unstandardized Coefficients	Standardized Coefficients	t	Sig.
		b	Beta		
1	(constant)	3.233		4.539	.000
	C24	.338	.272	2.427	.018
2	(constant)	4.090		5.353	.000
	C24	.461	.371	3.233	.002
	C27	-.283	-.293	-2.560	.013
3	(constant)	3.560		4.530	.000
	C24	.360	.289	2.448	.017
	C27	-.365	-.378	-3.186	.002
	C25	.285	.263	2.139	.036
4	(constant)	2.762		3.245	.002
	C24	.275	.221	1.853	.068
	C27	-.428	-.444	-3.708	.000
	C25	.341	.315	2.571	.012
	C5	.249	.241	2.159	.034

Dependent variable: R9

Table 40. Illustrates the model coefficients, respective t values and their significance (coefficients[a], coefficients[b], coefficients[c], coefficients[d]).

R10: Risk of 'Absence of Historical Data (Templates)' compared to Controls.

Table 41, Table 42, Table 43, and Table 44 show that the obtained significant values are less than the selected significant value of $\alpha = 0.05$, so there is a positive correlation between Controls 1, 3, 4, 5, 6, 7, 8, 10, 29 and Risk 10, respectively. In addition, Control 4 has an impact on Risk 10, and the results show that Control 4 has a positive impact value R=0.379 and the value of R^2 is 0.143. This means that the model (Table 42) explained 14.3% of the variability of response Risk 10. Furthermore, Durbin-Watson statistic (D) is 1.882, and ($d_U = 1.652$, $d_L = 1.598$) based on $K = 1$, $N = 76$, $\alpha = 0.05$; there is evidence of no autocorrelation because of the rule ($d_U < D < 2 + d_L$: No autocorrelation).

C1	C3	C4	C5	C6
.312**	.354**	.379**	.349**	.331**
C7	C8	C10	C29	
.370**	.253*	.234*	.256*	

Table 41. Illustrates correlations between respective controls and R10.

Model	R	R Square	Durbin-Watson
1	.379[a]	.143	1.882

a. Predictors: (constant), C4

Table 42. Illustrates multiple correlations R, and R square.

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	6.983	1	6.983	12.391	.001[a]
	Residual	41.702	74	.564		
	Total	48.684	75			

a. Predictors: (constant), C4 b. Dependent variable: R10

Table 43. Illustrates an Analysis of Variance (ANOVA[b]).

Model		Unstandardized Coefficients	Standardized Coefficients	t	Sig.
		b	Beta		
1	(constant)	3.023		5.690	.000
	C4	.370	.379	3.520	.001

a. Dependent variable: R10

Table 44. Illustrates the model coefficients, respective t values and their significance (coefficients[a]).

5.10. Software Risk Factors Identification Checklists and Control Factors (Risk Management Techniques)

Table 45 shows risk factors identification checklist with risk management techniques based on a questionnaire of experienced software project managers. We can use the checklist in software projects to identify and mitigate risk factors on lifecycle software projects by risk management techniques.

6. Conclusions

The concern of our paper are the modelling risks of planning software development. The results show that all risk planning software projects were very important, and important in software project manager's perspective, whereas all Controls are used most of the time, and rather often. This study incorporates risk management approach and planning software development to mitigate software project failure by using stepwise multiple regression technique. These tests were performed using regression analysis (stepwise regression), in order to compare the Controls to each of the risk factors, to determine and evaluate if they are effective in mitigating the occurrence of each risk factor, and, finally, to select the optimal model. Only significant relationships between risks and controls were reported. In addition, we determined the positive correlation between risk factors and risk management techniques, then measured impact risk in the software project lifecycle. We used correlation analysis, regression analysis, models based on the stepwise selection method (add and remove), and Durbin-Watson statistic. How-

No	Software Risk Factors	Risk Management Techniques
1	Low key user involvement.	C6: Implementing and following a communication plan, C21: Including formal and periodic Risk assessment.
2	Unrealistic schedules and budgets.	C1: Using of requirements scrubbing.
3	Misunderstood/unrealistic scope and objectives (goals).	C6: Implementing and following a communication plan.
4	Insufficient/inappropriate staffing.	C6: Implementing and following a communication plan.
5	Lack of senior management commitment and technical leadership.	C6: Implementing and following a communication plan, C16: Implementing/utilizing automated version Control tools.
6	Poor/inadequate software project planning and strategic thinking.	C5: Developing and adhering a software project plan, C14: Reusable database and data mining structures, C27: Combining internal evaluations by external reviews.
7	Lack of an effective software project management methodology.	C24: Ensuring that quality-factor deliverables and task analysis.
8	Change in organizational management during the software project.	C17: Implement/utilize benchmarking and tools of technical analysis, C27: Combining internal evaluations by external reviews, C25: Avoiding having too many new functions on software projects, C6: Implementing and following a communication plan.
9	Ineffective communication software project system.	C24: Ensuring quality-factor deliverables and task analysis, C27: Combining internal evaluations by external reviews, C25: Avoiding having too many new functions on software projects, C5: Developing and adhering a software project plan.
10	Absence of historical data (templates).	C4: Develop prototyping and have the requirements reviewed by the client.

Table 45. Software risk planning development factors were mitigated by risk management techniques.

ever, we reported the control factors in risk management approach were mitigated on risk planning software development factors in Table 45. Through the results, we found out that some controls don't have impact, so the important controls should be considered by the software development companies in Palestinian. In addition, we cannot obtain historical data from database before using certain techniques. As future work, we intend to apply these study results on a real-world software project to verify the effectiveness of the new techniques and approach on a software project. We can use other techniques to manage and mitigate software project risks, such as neural network, genetic algorithm, Bayesian statistics, and other artificial intelligence techniques.

7. Appendix

Appendix illustrates models with an intercept (from Savin and White)

Durbin-Watson Statistic: 1 Percent Significance Points of d_L and d_U and 5 Percent Significance Points of d_L and d_U.

	K	1		2		3	
N	**Significance**	**d_L**	**d_U**	**d_L**	**d_U**	**d_L**	**d_U**
75	1%	1.448	1.501	1.422	1.529	1.395	1.557
75	5 %	1.598	1.652	1.571	1.680	1.543	1.709

	K	4		5			
N	**Significance**	**d_L**	**d_U**	**d_L**	**d_U**		
75	1 %	1.368	1.586	1.340	1.617		
75	5 %	1.515	1.739	1.487	1.770		

Acknowledgments

This study is supported by Faculty of Information and Communication Technology, Technical University of Malaysia Malacca (UTeM), Malaysia and Al-Aqsa University, Palestine.

References

[1] K. SCHWALBE, Information Technology Project Management. *Sixth. Course Technology*, Cengage Learning, 2010, pp. 490.

[2] P. VASANT, Hybrid Mesh Adaptive Direct Search Genetic Algorithms and Line Search Approaches for Fuzzy Optimization Problems in Production Planning. In *Handbook of Optimization Intelligent Systems Reference Library*, **38**, I. ZELINKA, V. SNÁŠEL, A. ABRAHAM, EDS., Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 779–799.

[3] J. MILER AND J. GÓRSKI, Supporting Team Risk Management in Software Procurement and Development Projects. In *4th National Conference on Software Engineering*, 2002, pp. 1–15.

[4] C. R. PANDIAN, *Applied software Risk management: A guide for software project managers*. Auerbach Publications is an imprint of the Taylor & Francis Group, 2007, pp. 246.

[5] J. HOFFER, J. GEORGE AND J. VALACICH, *Modern Systems Analysis and Design*, 6th ed. Prentice Hall, 2011, pp. 575.

[6] J. SODHI AND P. SODHI, IT Project Management Handbook. Management Concepts, 2001, pp. 264.

[7] J. TAYLOR, Managing Information Technology Projects: Applying Project Management Strategies to Software, Hardware and Integration Initiatives. AMACOM ©2004, pp. 274.

[8] B. BOEHM, Value-based software engineering. *ACM SIGSOFT Softw. Eng. Notes*, **28**(2), pp. 1–12, Mar. 2003.

[9] A. ELZAMLY AND B. HUSSIN, Estimating Quality-Affecting Risks in Software Projects. *Int. Manag. Rev. Am. Sch. Press*, **7**(2), pp. 66–83, 2011.

[10] K. KHANFAR, A. ELZAMLY, W. AL-AHMAD, E. EL-QAWASMEH, K. ALSAMARA, S. ABULEIL, Managing Software Project Risks with the Chi-Square Technique. *Int. Manag. Rev.*, **4**(2), pp. 18–29, 2008.

[11] A. ELZAMLY, B. HUSSIN, Managing Software Project Risks with Proposed Regression Model Techniques and Effect Size Technique. *Int. Rev. Comput. Softw.*, **6**(2), pp. 250–263, 2011.

[12] A. ELZAMLY, B. HUSSIN, Managing Software Project Risks (Implementation Phase) with Proposed Stepwise Regression Analysis Techniques. *Int. J. Inf. Technol.*, **1**(4), pp. 300–312, 2013.

[13] A. ELZAMLY AND B. HUSSIN, Managing Software Project Risks (Design Phase) with Proposed Fuzzy Regression Analysis Techniques with Fuzzy Concepts. *Int. Rev. Comput. Softw.*, **8**(11), pp. 2601–2613, 2013.

[14] A. ELZAMLY, B. HUSSIN, Managing Software Project Risks (Planning Phase) with Proposed Fuzzy Regression Analysis Techniques with Fuzzy Concepts. *Int. J. Inf. Comput. Sci.*, **3**(2), pp. 31–40, 2014.

[15] A. ELZAMLY, B. HUSSIN, Identifying and Managing Software Project Risks with Proposed Fuzzy Regression Analysis Techniques: Maintenance Phase. In *2014 Conference on Management and Engineering (CME2014)*, 2014(Cme), pp. 1868–1881.

[16] A. ELZAMLY, B. HUSSIN, Managing Software Project Risks (Analysis Phase) with Proposed Fuzzy Regression Analysis Modelling Techniques with Fuzzy Concepts. *J. Comput. Inf. Technol.*, **22**(2), pp. 131–144, 2014.

[17] A. ELZAMLY, B. HUSSIN, Modelling and mitigating Software Implementation Project Risks with Proposed Mining Technique. *Inf. Eng.*, **3**, pp. 39–48, 2014.

[18] A. ELZAMLY, Evaluation of Quantitative and Mining Techniques for Reducing Software Maintenance Risks. *Appl. Math. Sci.*, **8**(111), pp. 5533–5542, 2014.

[19] A. ELZAMLY, B. HUSSIN, Mitigating Software Maintenance Project Risks with Stepwise Regression Analysis Techniques. *J. Mod. Math. Front.*, **3**(2), pp. 34–44, 2014.

[20] A. ELZAMLY, B. HUSSIN, A Comparison of Stepwise And Fuzzy Multiple Regression Analysis Techniques for Managing Software Project Risks: Analysis Phase. *J. Comput. Sci.*, **10**(10), pp. 1725–1742, 2014.

[21] A. ELZAMLY, B. HUSSIN, A Comparison of Fuzzy and Stepwise Multiple Regression Analysis Techniques for Managing Software Project Risks: Implementation Phase. *Int. Manag. Rev.*, **10**(1), pp. 43–54, 2014.

[22] A. ELZAMLY, B. HUSSIN, An Enhancement of Framework Software Risk Management Methodology for Successful Software Development. *J. Theor. Appl. Inf. Technol.*, **62**(2), pp. 410–423, 2014.

[23] H. HOODAT, H. RASHIDI, Classification and Analysis of Risks in Software Engineering. *Eng. Technol.*, **56**(32), pp. 446–452, 2009.

[24] B. BOEHM, Software Risk Management: Principles and Practices. *IEEE Softw.*, (January), pp. 1–10, 1991.

[25] J. MILER, A Method of Software Project Risk Identification and Analysis. Gdansk University of Technology, 2005.

[26] CHAOS, The Standish Group Report. 1995.

[27] T. ADDISON, S. VALLABH, Controlling Software Project Risks – an Empirical Study of Methods used by Experienced Project Managers. In *Proceedings of SAICSIT*, 2002, pp. 128–140.

[28] T. ADDISON, E-commerce project development Risks: Evidence from a Delphi survey. *Int. J. Inf. Manage.*, **23**(2003), pp. 25–40, 2003.

[29] D. ALOINI, R. DULMIN, V. MININNO, Risk management in ERP project introduction: Review of the literature. *Inf. Manag.*, **44**(6), pp. 547–567, Sep. 2007.

[30] W. HAN, S. HUANG, An empirical analysis of Risk components and performance on software projects. *J. Syst. Softw.*, **80**(1), pp. 42–50, Jan. 2007.

[31] S. HUANG, W. HAN, Exploring the relationship between software project duration and Risk exposure: A cluster analysis. *Inf. Manag.*, **45**(3), pp. 175–182, Apr. 2008.

[32] B. ARITUA, N. J. SMITH, D. BOWER, What Risks are common to or amplified in programmes: Evidence from UK public sector infrastructure schemes. *Int. J. Proj. Manag.*, **29**(3), pp. 303–312, Apr. 2011.

[33] R. SCHMIDT, K. LYYTINEN, M. KEIL, P. CULE, Identifying Software Project Risks: An International Delphi Study. *J. Manag. Inf. Syst.*, **17**(4), pp. 5–36, 2001.

[34] M. KEIL, P. CULE, K. LYYTINEN, R. SCHMIDT, A Framework for Identifying Software Project Risks. *Commun. ACM*, **41**(11), pp. 76–83, 1998.

[35] R. NAKATSU, C. IACOVOU, A comparative study of important Risk factors involved in offshore and domestic outsourcing of software development projects: A two-panel Delphi study. *Inf. Manag.*, **46**(1), pp. 57–68, Jan. 2009.

[36] J. CHEN, S. HUANG, An empirical analysis of the impact of software development problem factors on software maintainability. *J. Syst. Softw.*, **82**(6), pp. 981–992, Jun. 2009.

[37] M. KEIL, A. TIWANA, A. BUSH, Reconciling user and project manager perceptions of IT project Risk: A Delphi study. *Inf. Syst. J.*, **12**(2), pp. 103–119, 2002.

[38] L. WALLACE, M. KEIL, A. RAI, How Software Project Risk Affects Project Performance: An Investigation of the Dimensions of Risk and an Exploratory Model. *Decis. Sci.*, **35**(2), 2004.

[39] M. SUMNER, Risk factors in enterprise-wide/ERP projects. *J. Inf. Technol.*, **15**(4), pp. 317–327, Dec. 2000.

[40] B. BOEHM, R. ROSS, Theory-W Software Project Management: Principles and Examples. *IEEE Trans. Softw. Eng.*, **15**(7), pp. 902–916, 1989.

[41] KWEKU EWUSI-MENSAH, *Software Development Failures: Anatomy of Abandoned Projects*. The MIT Press, p. 276, 2003.

[42] G. PARÉ, C. SICOTTE, M. JAANA, D. GIROUARD, Prioritizing Clinical Information System Project Risk Factors: A Delphi Study. In *Proceedings of the 41st Hawaii International Conference on System Sciences*, pp. 1–10, 2008.

[43] D. HOUSTON, G. MACKULAK, J. COLLOFELLO, Stochastic simulation of Risk factor potential effects for software development Risk management. *J. Syst. Softw.*, **59**(3), pp. 247–257, Dec. 2001.

[44] B. LAWRENCE, K. WIEGERS, C. EBERT, The Top Risks of Requirements Engineering, 2001.

[45] D. SHAFER AND C. OFFICER, Software Risk: Why must we keep learning from experience? Where Does Risk Occur in DP Software Development? In *Dynamic Positioning Conference*, pp. 1–19, 2004.

[46] C. Jones, G. Glen, G. Anna, D. Miller, Strategies for improving systems development project success. *Issues Inf. Syst.*, **XI**(1), pp. 164–173, 2010.

[47] C. Jones, *Applied Software Measurement Global Analysis of Productivity and Quality, Third.* No. Third Edition. McGraw-Hill Companies, 2008, p. 662.

[48] A.-N. Taimour, Why IT Projects Fail, 2005.

[49] C. Martin, J. Pasquier, C. M., A. T., Software Development Effort Estimation Using Fuzzy Logic: A Case Study. In *Proceedings of the Sixth Mexican International Conference on Computer Science (ENC'05)*, pp. 113–120, 2005.

[50] X. Jin, X. Xu, Remote sensing of leaf water content for winter wheat using grey relational analysis (GRA), stepwise regression method (SRM) and partial least squares (PLS). In *First International Conference on Agro-Geoinformatics (Agro-Geoinformatics)*, pp. 1–5, 2012.

[51] Y. Lan, S. Guo, Multiple Stepwise Regression Analysis on Knowledge Evaluation. In *International Conference on Management of E-Commerce and E-Government*, pp. 297–302, 2008.

[52] N. Zhou, J. Pierre, D. Trudnowski, A Stepwise Regression Method for Estimating Dominant Electromechanical Modes. *IEEE Trans. Power Syst.*, **27**(2), pp. 1051–1059, 2012.

Received: August, 2014
Revised: October, 2014
Accepted: November, 2014

Contact addresses:

Abdelrafe Elzamly
Department of Computer Science
Faculty of Applied Sciences
Al-Aqsa University P.O.BOX: 4051
Gaza
Palestine
e-mail: abd_elzamly@yahoo.com

Burairah Hussin
Fakulti Maklumat & Komunikasi
Universiti Teknikal Malaysia Melaka
Locked Bag 1752, Durian Tunggal
P.O.BOX: 76109 Durian Tunggal
Melaka Malaysia
e-mail: burairah@utem.edu.my

Abdelrafe Elzamly is currently a Ph.D. student of Information and Communication Technology at the Technical University Malaysia Malaka (UTeM). Born on November 30, 1976 in Gaza, Palestine, he received his B.Sc. degree in 1999 from AI-Aqsa University, Gaza and his Master degree in Computer Information Systems in 2006 from the University of Banking and Financial Sciences. Since 1999 he has been working as a full time lecturer of Computer Science at AI-Aqsa University. Also, from 1999 to 2007 he worked as a part time lecturer at the Islamic University in Gaza. Between 2010 and 2012 he worked as a Manager in the Mustafa Center for Studies and Scientific Research in Gaza. His research interests are in Risk management, quality software, software engineering and data mining.

Burairah Hussin received his Ph.D. degree in Management Science-Condition Monitoring Modelling, from the University of Salford,UK in 2007. Before that, he received a M.Sc. degree in Numerical Analysis and Programming from the University of Dundee, UK in 1998 and a B.Sc. degree in Computer Science from the University of Technology Malaysia in 1996. He currrently works as an Associate Professor at the Technical University Malaysia Malaka (UTeM). He also worked as the Deputy Dean (Research and Postgraduate) at the Faculty of Information and Communiction Technology, Technical University of Malaysia Malaka (UTeM). His research interests are in data analysis, data mining, maintenance modelling, artificial intelligence, Risk management, numerical analysis, and computer network advising and development.