AN EMPIRICAL STUDY OF SOFTWARE DEVELOPMENT FAILURES IN SRI LANKA

A. J. Perera[1] and S. P. P. Amaratunge[2]

ABSTRACT

Software development projects are considered as a unique entity in terms of project management due to the inherent attributes they encapsulate, thus making them distinct and susceptible to failure. Therefore, the importance of software project risk management is emphasized where the identification of the factors which affect software development projects to fail being the main component. This survey-based study focuses on identifying the factors that affect software development failures and the causes of these factors in Sri Lankan software development companies. Twenty-seven factors affecting software development failures were identified using a stringent scientific methodology and were tested to ascertain their validity, importance and the causes in terms of the Sri Lankan context using an interview based questionnaire. Twenty-five software development companies which belong to the software exporters' association were selected for data collection, and the gathered data were analyzed to ascertain the significance to the defined objectives using statistical tools. Misunderstanding of user requirements, poor project management skills of managers, and the lack of communication between the members of the software development team were identified as the top three factors affecting software development failures in Sri Lanka, while the lack of proper communication between the development team and the customer, poor project management knowledge of managers, and lack of proper planning were identified as the main phenomena which cause these factors to originate in software development projects in Sri Lanka.

INTRODUCTION

Information technology (IT) is one of the fastest growing industries in developed countries (Hartmanand and Ashrafi, 2002). IT projects can implement a rapidly expanding range of equipment, applications, services, and basic technologies that provide information to support the operation, management, analysis and decision-making functions within an organisation (Bacarini et al, 2004). In 1995, the spending of the United States (US) on software development projects reached $250 billion with 175,000 recorded software development projects. In that year, the US companies have paid an estimated $59 billion on

[1]Senior Team Leader, Cambio Software Engineering Pte. Ltd., Sri Lanka. Email: apjanaka@yahoo.com

[2]Professor, Department of Business Economics, University of Sri Jayewardenepura, Sri Lanka. Email: sppamaratunge@yahoo.com

cost overruns and another $81 billion on cancelled projects (Wallace and Keil, 2004). As implied by the statistics, software development project failures are a major issue and many studies can be found in literature which has attempted to reduce the number of project failures in the software development industry. Standish group report of 2004 specified that over 70% of the software development projects in the US were failures. Therefore, software development failures are important to be identified and mitigated in software development projects (Wallace and Keil, 2004).

Software development is becoming one of the most important industries in Sri Lanka and many foreign organizations outsource software development projects to Sri Lankan software development companies. Some of the major players in the software field, like Microsoft[3] and RedHat[4] have country offices in the island. Furthermore, there are many large software firms in the country and some multinational software companies[5] have opened development centres in Sri Lanka. Sri Lankan software development companies also face the issue of software development failures. Although reliable information on the costs associated with the total software development failures in Sri Lanka could not be found due to the non-availability of statistics, it is clear that Sri Lankan software companies also face the problem of failure in software development projects even though the extent of the implications is not known.

Software has four unique inherent attributes; complexity, conformity, changeability and reliability, which make software development a very difficult and complex task (Brooks, 1987). These attributes make software unique and impossible to mitigate against all risks. Therefore, there is no set method which can be followed in reducing software development failures. The best way is to mitigate risk so that the project will have a greater probability of success. Software development project managers have to identify the factors which might affect the failures of software development project and then try to reduce these factors in the software development process.

There are two types of investigations into factors of software development failures; some studies define individual factors while some amalgamate factors into defined groups by formulating frameworks which facilitate in the identification of the factors which cause software development failures. One of the main frameworks defined for identifying software development failures was compiled by Keil et al (1998) in their paper "A Framework for identifying software project risks". Keil et al defined four quadrants of type of factors which relate to software development failures. Customer mandate, scope and requirements, execution and environment were defined as the four quadrants. The validity of this study was emphasized by Wallace and Keil (2004) where they proved the validity of the framework defined by Keil et al (1998) by interviewing over five hundred software development project managers in the US. This study has used the above said two researches and has tried to attain two objectives; Ascertain the major factors[6] which effect software development project failures in Sri Lanka software development companies and to rank them according to the significance of the factor to software development project failures and

[3]Microsoft is a software development, which has developed the Windows Operating System.
[4]RedHat is another software company, which is the creator of ReHat Linux Operating System.
[5]IFS and Cambio are two examples of major software players in the world having development centres in Sri Lanka.
[6]Phenomena, which lead to software development projects to fail.

to ascertain the causes[7] of the factors of software development failures in Sri Lankan software development companies.

METHODOLOGY

A stringent and comprehensive methodology was followed when compiling this study to attain maximum validity and reliability. Initially, existing literature was collected and analyzed to comprehend the context of the problem faced. Literature was collected from various sources with scientific journals being the preferred sources. The found literature was individually analyzed and important factors[8] were identified and this resulted in the compilation of concepts related to software development failures. This process was undertaken in a methodical and scientific manner focusing on gaining maximum validity and reliability to the concepts which were identified. Two methods were used in concept identification: concepts were identified using the literature survey and concepts were identified via data gathering from industry professionals[9]. Data gathering through experts was essential in terms of the validity of the concepts since there were no previous studies done to find the factors which contribute to software development failures in Sri Lankan software companies.

List of twenty seven concepts were identified to be studied for the relationship they have in terms of software development project failures in Sri Lankan software development companies. This concept identification process was of four fold, integrating both data collection from industry professionals and getting input from existing literature. Initially, concepts were identified getting feedback from industry professionals, which resulted in the compilation of twenty four concepts. These twenty four concepts were then modified by three software project managers reducing the list to twenty one. This list of twenty one concepts was compared against two studies found in literature: risk categorization framework, a framework developed by Keil et al in their research paper published in Communication of ACM[10] 1998, and software project risks and their effect on outcomes by Wallace and Keil published in Communication of ACM in 2004, which were used to further modify the list of twenty one factors, based on which a list of twenty six concepts was identified. Finally, the pilot survey resulted in the addition of one more concept and a final list of twenty seven concepts was identified. Table 01 describes the filtering of concepts in the four stages to formulate the list of twenty seven concepts.

Concept identification leads to the operationalization of these concepts in to variables. The identified concepts were analyzed and converted into variables. There were two main kinds of variables identified; independent variables and dependent variables. Dependent variable of the study was identified as "Software development failures". Four main independent variables were identified; customer mandate, scope and requirements, execution, and environment. The twenty seven concepts identified were divided among these four quadrants and used to explain the four variables. Customer mandate consists of four factors; communication between the development team and users, change management

[7]Phenomena, which directly cause the factors to occur.

[8]Factors which affect software development projects to fail.

[9]Industry professionals are conceptualized as people working in software development industry with five years of working experience and have worked in more than two software development companies conducting software development related work.

[10]Communication of ACM is the monthly journal of association for computer machinery. This was launched in 1957 and is considered one of the main computer related journals in the world.

(customer side), lack of top management commitment (customer side) and lack of user participation, while scope and requirements consists of five factors; misunderstanding user requirements, changing user requirements, poorly defined project scopes, incorrect or conflicting user requirements, and undefined project success criteria. Execution consists of fifteen factors; lack of skilled staff, poor management skills, use of incorrect modelling techniques, insufficient budgets, bypassing lifecycle stages when running short of time or when considered not that important, lack of commitment of team members, inadequate testing, high employee turnover, lack of training of staff, lack of proper quality standards, lack of skilled staff, insufficient research ,lack of proper documentation, lack of industry/domain knowledge and communication between members of the development team with environment quadrant containing three factors; unstable organizational environment, corporate politics with negative effect on project and resources shifted from the project due to changes in organizational priorities.

Variables identified in the operationalization phase were used to develop a questionnaire for data collection. Conformity of the questionnaire to the objectives was of paramount importance and the entire design process was guided by the two objectives to gain high validity and reliability to the questionnaire. When designing the questionnaire, care was taken so that every question led to measure a certain variable or to get data relating to a variable. Furthermore, all questions were developed to be single directional[11] in order to enable smooth analysis of the collected data. Questionnaire designed focused on making the questions as simple as possible and to explain the needed information clearly to the respondent. Although the clarity of the questions is not emphasized due to the method of data collection being an interview based questionnaire, the clarity and the descriptiveness of the questions were maintained.

Operationalization divided the twenty seven concepts identified into four main concepts; customer mandate, scope and requirements, execution, and environment. The questionnaire was also designed in terms of this segmentation with the inclusion of four main sections to represent the four quadrants; customer mandate, scope and requirements, execution and environment. These sections included questions for each of the factors identified and included inside each of the sections. A single factor was represented by two questions in the questionnaire. First question was used to ascertain if the respondent agreed that the particular factor was a cause in software development failures in the respondent's organization. Five factor Likert scale, a famous attitudinal scale, was used to measure the responses of the first question with five options being given to the respondent; strongly agree, agree, neutral, disagree, and strongly disagree. The second question was designed to get the feedback of respondents who selected either strongly agree or agree in the first question. Second question asks the respondent to list down the causes for the particular factor to occur in organizations. Four options are given as selections, and a fifth option is given for respondents who think that the reason is not included in the questionnaire to describe their perceived reasons. The questionnaire had fifty four questions with each of the twenty seven factors having two questions each.

[11]Negative questions were not used.

Table 01: Four Step Concept Identification Process

Step 1 - (24 factors identified)	Step 2 - (21 factors identified)	Step 3 - (26 factors identified)	Step 4 - (27 factors identified)
Lack of proper Communication between stakeholders	Communication between development team and users	Communication between development team and users	Communication between development team and users
Lack of proper documentation	Communication between the team	Communication between the team	Communication between the team
Use of wrong technologies	Lack of proper documentation	Lack of proper documentation	Lack of proper documentation
Lack of skilled staff	Use of wrong technologies	Use of wrong technologies	Use of wrong technologies
Misunderstanding user requirements	Lack of skilled staff	Lack of skilled staff	Lack of skilled staff
Lack of teamwork	Misunderstanding user requirements	Misunderstanding user requirements	Misunderstanding user requirements
Changing User requirements	Changing User requirements	Changing User requirements	Changing User requirements
Poor Management skills	Poor Management skills	Poor Management skills	Poor Management skills
Change management (customer side)	Change management (customer side)	Change management (customer side)	Change management (customer side)
Coordination between stakeholders	Use of incorrect modeling techniques	Use of incorrect modeling techniques	Use of incorrect modeling techniques
Use of incorrect modeling techniques	Lack of top management commitment (customer side)	Lack of top management commitment (customer side)	Lack of top management commitment (customer side)
Lack of top management commitment (customer side)	Insufficient budgets	Insufficient budgets	Insufficient budgets
Insufficient budgets	Lack of industry knowledge	Lack of industry knowledge	Lack of industry knowledge
Lack of industry knowledge	Bypassing lifecycle stages	Bypassing lifecycle stages	Bypassing lifecycle stages
Poor prioritization of tasks	Commitment of team members	Commitment of team members	Commitment of team members
Bypassing lifecycle stages	Poorly defined project scopes	Poorly defined project scopes	Poorly defined project scopes
Lack of commitment of team members	Insufficient research	Insufficient research	Insufficient research
Poor monitoring and control of (upper) management on the status progress of the	Inadequate testing	Inadequate testing	Inadequate testing
Poorly defined project scopes	High employee turnover	High employee turnover	High employee turnover
Insufficient research	Lack of training of staff	Lack of training of staff	Lack of training of staff
Inadequate testing	Lack of proper quality standards	Lack of proper quality standards	Lack of proper quality standards
High employee turnover		Lack of user participation	Lack of user participation
Lack of training of staff		Incorrect/conflicting requirements	Incorrect/conflicting requirements
Lack of proper quality standards		Undefined project success criteria	Undefined project success criteria
		Unstable organizational environment	Unstable organizational environment
		Internal corporate politics	Internal corporate politics
			Resource allocation

Source: Author constructed

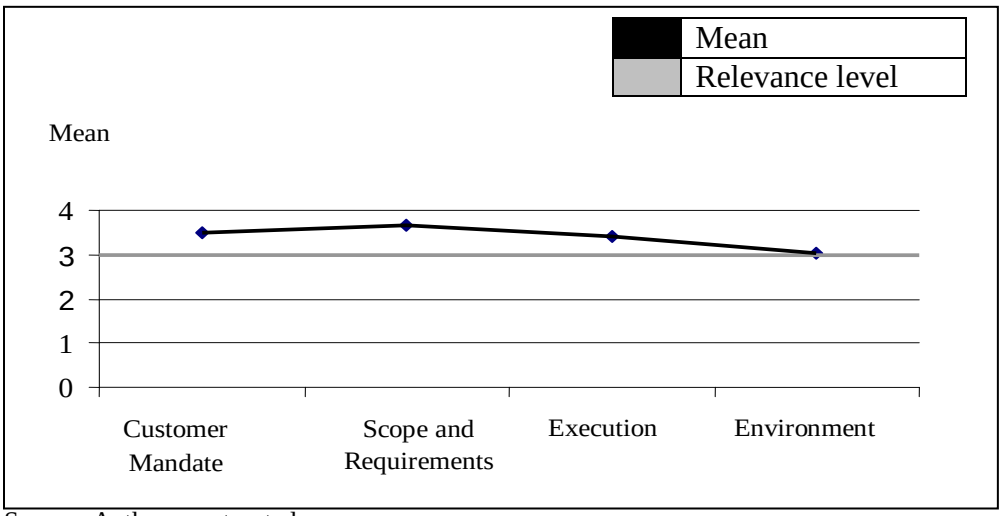
Unit of analysis was taken as organizations, aligning with the objective of finding the factors which affect software development failures in Sri Lankan software companies. Target population was identified as the set of software development companies in Sri Lanka. Only primary data were used, where data was collected in the field using an interview based questionnaire in a non-contrived environment. Data were gathered from twenty five software development companies in Sri Lanka who are members of the software exporters' association.

Data analysis was done after all data were collected and entered in to SPSS (version 13.0). Information was represented in SPSS in a meaningful and correct format to facilitate smooth analysis of the data. All data collected using the questionnaires were of the type Ordinal, and, representation of these data in SPSS was done, so that a factor will be represented by a single variable in SPSS. The questions used to query the relevance of the factor used the five point Likert scale and the scale values were represented as Strongly Agree – 5, Agree - 4, Neutral – 3, Disagree – 2, and Strongly Disagree – 1. The questions ascertaining the causes of the factors were represented by five variables per factor in SPSS. Each option in the questions was represented by a variable in SPSS and each of this contained a Boolean value (1,0); if the respondent selected a particular option as being a cause for a certain factor, it was represented by as one (1) in SPSS, and if the respondent did not select an option, it was represented by a zero (0).

ANALYSIS

The four quadrants, customer mandate, scope and requirements, execution and environment were all considered important in software development failures in Sri Lankan software companies. When analysing the mean values[12] of the relevant quadrants, it is clear that all four quadrants were important in terms of software development project failures (Figure 01). The environmental quadrant is the least important with a mean value of 3.04, while the scope and requirements quadrant had the highest mean value of 3.66. Even though scope and requirement quadrant was considered as the most important with a mean value of 3.66, the mean table alone did not show us the exact picture of the result since execution quadrant had fifteen factors and there were a few factors like lack of a sufficient budget which got a very low approval rating (2.76 mean value for factor) as a factor in software development failures, while factors such as the lack of project management skills of project managers got very high ratings (3.92 mean value for factor).

Figure 01: Summary of the Four Quadrants



Source: Author constructed

[12]Mean values of over three were considered as the factor being important in terms of software development failures.

When analysing the importance of factors for software development failures, it is clear that misunderstanding of user requirements is seen as the most important factor in software development failures (factor mean value of 4.0). Eighty-eight per cent of the respondents either agreed or strongly agreed that this was a factor in software development failures, while only 8% thought otherwise. Lack of proper communication between team members and Incorrect and conflicting user requirements, and Lack of domain knowledge of the team members were seen as the main causes for misunderstanding user requirements. Another important factor was Poor project management skills of project managers. This had factor mean value of 3.94 and 80% of the respondents thought that this was a factor in software development project failures in Sri Lanka. Lack of planning and poor project management knowledge of managers was seen as the main causes of this factor. Lack of communication between team members and poorly defined project scopes were also seen as important factors with high mean values, while factors, such as Internal corporate politics and high employee turnover were regarded less important relatively. There were two factors which had mean values less than 3.00: Lack of a sufficient budget and Lack of a stable environment in the company were given mean values 2.76 and 2.72. This is an important conclusion, especially when we consider the economic situation[13] of the country. It is clear that monitory problems or the stability of the companies do not figure as being important in factors of software development failures in Sri Lanka. Table 02 defines the list of factors and causes of these factors listed according to importance measured by its mean.

Table 02: Factor and Cause Summary of the Twenty Seven Factors

Factor	Mean	Main Cause
Misunderstanding of user requirements	4.00	Lack of proper communication between team members and customer
Poor project management skills of project managers	3.92	Poor project management knowledge of managers
Lack of communication between team members	3.88	Deficiencies in the communication mechanism in the company
Poorly defined project scopes	3.84	Lack of proper planning
Lack of documentation	3.64	Low priority given to documentation and documentation ignored when tight deadlines
Incorrect or conflicting user requirements	3.64	Customer not comprehending his/her requirements properly
Lack of change management in user side	3.60	Lack of knowledge in change management
Insufficient software testing	3.60	Lack of time for testing
Lack of management commitment from customer side	3.56	Not understanding the extent of the implications of the system
Use of wrong technologies	3.52	No technical feasibility study done

[13]Economic indicators like inflation rate and exchange rate fluctuations suggest that the economy is in a recession. Furthermore, the security situation in the country also paints a bleak picture of the economy and its stability.

Skill level of staff members	3.52	No trained staff for the required technologies
Dynamic user requirements	3.52	Customer not certain of the requirements
Lack of user participation	3.44	Problems in the customer organization and User involvement not integrated to the project plan
Lack of training for staff	3.40	Lack of emphasis shown on training
Lack of industry knowledge	3.40	Domain new to the team
Incorrect modeling techniques	3.36	Formal modeling techniques not used by the company
Communication between the development team and users	3.36	Deficiencies in the communication mechanism between users and the development team
Undefined success criteria	3.32	Poor project management
Lack of proper quality standards	3.32	Time constraints
High employee turnover	3.32	Lack of a good culture in the company
Internal corporate politics	3.24	Poor management skills of managers
Insufficient research done on technologies	3.24	No dedicated resources for R&D
Resources being shifted from the project	3.12	Lack of resources in other projects
By-passing development life cycle stages	3.12	Time constraints
Lack of commitment of the development team	3.04	Salary deficiencies and overworked employees
Lack of a sufficient budget	2.76	Allocating less money than actually needed
Lack of a stable environment in the company	2.72	Uncertainty about the future survival of the company

Source: Author constructed

Poor management practices, time constraints, the inability to comprehend the exact requirements of the software to be developed, and the lack of communication were seen as the main causes of twenty seven factors of software development failures.

Customer mandate quadrant found the lack of change management practices in the customer organization and communication gap between the development team and users as being the main causes, while the Scope and requirement quadrant emphasized that the inability to comprehend the exact requirements of the software to be developed as being the main cause. Identifying and comprehending the requirements properly was considered as an essential element in terms of software development failures and misunderstanding of requirements was also ranked the number one factor of software development failures. This implies that misunderstanding of user requirements is not only a major factor but also a cause for many other factors. In the execution quadrant, time constraints were seen as the main cause for the risk factors. It was found that when under tight time constraints, the

managers tend to avoid some important stages in the life cycle stages[14] and these cause software development project failures. Environment quadrant found that internal corporate politics and shifting of resources was the main cause for software development failures and these factors also indirectly lead to poor management practices as being the cause. Table 3 explains the factor and cause summary of each of the four quadrants according to the importance of the factors for software development project failures in Sri Lankan software development companies.

Table 03: Quadrant-wise Factor and Causes Summary

Quadrant / Factor	Main Cause/s	Mean	<= Neutral	> Neutral
Customer Mandate				
Lack of change management in user side	Lack of knowledge in change management	3.60	10	15
Lack of management commitment from customer side	Not understanding the extent of the implications of the system	3.56	10	15
Lack of user participation	Problems in the customer organization and User involvement not integrated to the project plan	3.44	11	14
Communication between the development team and users	Deficiencies in the communication mechanism between users and the development team	3.36	13	12
Quadrant mean		3.49		
Scope and Requirements				
Misunderstanding of user requirements	Lack of proper communication between team members and customer	4.00	3	22
Poorly defined project scopes	Lack of proper planning	3.84	6	19
Incorrect or conflicting user requirements	Customer not comprehending his/her requirements properly	3.64	10	15
Dynamic user requirements	Customer not certain of the requirements	3.52	11	14
Undefined success criteria	Poor project management	3.32	13	12
Quadrant mean		3.66		

[14]Software development life cycle stages.

Execution				
Poor project management skills of project managers	Poor project management knowledge of managers	3.92	4	21
Lack of communication between team members	Deficiencies in the communication mechanism in the company	3.88	5	20
Lack of documentation	Low priority given to documentation and Documentation ignored when tight deadlines	3.64	7	18
Insufficient software testing	Lack of time for testing	3.60	7	18
Use of wrong technologies	No technical feasibility study done	3.52	9	16
Skill level of staff members	No trained staff for the required technologies	3.52	9	16
Lack of industry knowledge	Domain new to the team	3.40	10	15
Lack of training for staff	Lack of emphasis shown on training	3.40	11	14
Incorrect modeling techniques	Formal modeling techniques not used by the company	3.36	12	13
High employee turnover	Lack of a good culture in the company	3.32	12	13
Lack of proper quality standards	Time constraints	3.32	12	13
Insufficient research done on technologies	No dedicated resources for R&D	3.24	12	13
By-passing development life cycle stages	Time constraints	3.12	13	12
Lack of commitment of the development team	Salary deficiencies and overworked employees	3.04	15	10
Lack of a sufficient budget	Allocating less money than actually needed	2.76	17	8
Quadrant mean		3.40		
Environment				
Lack of a stable environment in the company	Uncertainty about the future survival of the company	2.72	20	5
Internal corporate politics	Poor management skills of managers	3.24	13	12
Resources being shifted from the project	Lack of resources in other projects	3.12	14	11
Quadrant Mean		3.04		
Lack of a stable environment in the company	Uncertainty about the future survival of the company	2.72	20	5

Internal corporate politics	Poor management skills of managers	3.24	13	12
Resources being shifted from the project	Lack of resources in other projects	3.12	14	11
Quadrant mean		3.04		

Source: Author constructed

According to Table 03, all four factors in the customer mandate quadrant, lack of change management in user side, lack of management commitment from customer side, lack of user participation and Communication between the development team and users were all treated as being of important by the respondents in terms of factors affecting software development failures in Sri Lanka. Mean values are more than three for all factors, implying that they are considered by respondents as being important factors in software development project failures in Sri Lankan software development companies. Further analysis based on the total number of responses given for as agreed and strongly agreed against the number of responses given for neutral, disagreed and strongly disagreed defines that the Lack of change management in user side, Lack of management commitment from customer side, and Lack of user participation are considered by more than 50% of the respondents as either agreed or strongly agreed in terms of importance in software development failures, while Communication between the development team and users is considered by more than 50% of the respondents as either neutral, disagree or strongly disagree in terms of importance for software development failures.

When analysing the mean values of the responses for the five factors in the scope and requirements quadrant, we can see a minimum mean value of 3.32 for undefined success criteria and a maximum mean value of 4.00 for misunderstanding user requirements. Since all five factors are over the mean value of three, we can conclude that all factors are considered as important when considering software development failures. We can analyse the importance of these factors furthermore if we analyse this result based on the total number of responses given for as agreed and strongly agreed with the number of responses given for neutral, disagreed and strongly disagreed. If the user selects either agreed or strongly agreed, the respondent can be treated as agreeing that the relevant factor affects the software development project failures in Sri Lanka. It can be seen that misunderstanding user requirements had most responses above neutral, while undefined success criteria had the least responses. In fact the order of the factors when sorted according to the number of responses above neutral is same as in the mean value. But the important factor here is that undefined success criteria were considered important by less than 50% of the respondents. Therefore, the importance of undefined success criteria can be questioned although the mean value of the factor is above three. Therefore, we can conclude that in the scope and requirements quadrant, while all five factors are important for software development failure, misunderstanding user requirements is the main factor which affects the software projects failures, while undefined success criteria are the least important factor. Furthermore, undefined success criteria are believed by less than 50% as being important for software development failures.

Time constraints are seen as the main reason for the causes of the factors in the execution quadrant. Most software projects are under tight time constraints and when the time constraints are exceeded, the project managers, while trying to deliver the product quickly to the customer, make many mistakes. When the mean values of factors in quadrant three are analysed, it is observed that it varies between 3.92, poor project management skills

and 2.76, lack of a sufficient budget. This gives a fairly diverse list of factors varying according to importance. According to the mean values, the most important factor is the lack of project management skills of project managers, and the least important is the budgetary limitations. Lack of a sufficient budget is the only factor with less than three mean values. If these factors were analysed based on the total number of responses given for as agreed and strongly agreed with the number of responses given for neutral, disagreed and strongly disagreed, the most important factor becomes the lack of communication between team members while the least important factor remains the lack of a sufficient budget. The interesting factor we can see from this is, three factors by-passing life cycle stages; lack of commitment of the development team and the lack of a sufficient budget have less than 50% believing that they are important factors in software development failures in Sri Lankan software companies.

Environment quadrant has the least mean value out of the four quadrants. The fact that the mean value is 3.04 implies that although it is the lowest among the four quadrants, it is still important in terms of software development failures. Poor management skills and practices are seen as the major cause which affects the environmental factors in a software project. Internal corporate politics and resources shifting from projects are considered as important by respondents; surprisingly a lack of stable environment in the company is seen as not important relatively, which implies that people don't compromise their work even if the stability of their jobs are at risk. Only 20% said that the lack of a stable environment in the company had an effect on software development failures (strongly agree or agree). This might be attributed to the fact that people work harder when there is a risk to the company in order to save the company and their jobs. Nearly half the people think that internal corporate politics are a major factor in software development failures, while only 28% disagrees. Most people attribute this to poor management skills of the managers or the lack of motivation among team members which might also be a product of the poor management skills of the managers. Poor management skills among managers in the software field might be directly attributed to the fact that most of the teams, if not all, are led by technical people with little or no management experience. Shifting of resources is considered by nearly half the respondents as being an important issue while only 32% believe this is not an issue. Many believe resources are shifted because of lack of resources while poor management skills of managers are also attributed for this issue. The important thing here is only two respondents believed that lack of funds were to blame for this. This implies that lack of resources is not directly tied with budgets which again direct us to bad employee management policy which in turn directs us to poor management.

SUMMARY AND CONCLUSIONS

This study focused on ascertaining the factors which affect software development project failures in Sri Lankan software development companies and to figure out the causes of these factors, in line with the two objectives defined. Overall view of the identified factors affecting software development project failures in Sri Lankan software development companies implies that poor management practices and the lack of comprehension of requirements[15] are the two main factors which cause software development projects to fail. The knowledge of the software development project managers in terms of project management practices is seen as an essential component of success while the ability of the software development team to comprehend the customer requirements also features as an integral part of a successful software development project.

[15] The functionality that the software product should conform to.

With respect to the first objective; misunderstanding of user requirements, poor project management skills of project managers, lack of communication between team members and, poorly defined project scopes were identified as the top four factors which affect software development failures, while lack of a stable environment in the company and lack of a sufficient budget were seen as the two least important factors.

In accordance with the second objective; lack of proper communication between team members and customer, poor project management knowledge of managers, and the lack of proper planning were seen as the main causes of these factors of software development failures. The important observation here is that some factors, such as misunderstanding user requirements were identified as causes for the origination of other factors emphasizing the importance of these factors to the software development process. It can be implied that the software development companies should focus on improving project management practices, the timely delivery of software giving efficient time estimates, improve requirement gathering process and improve the communication mechanisms both internally and externally to reduce the occurrence of the factors which contribute to software development project failures.

When analysing the findings using the risk categorization framework, it can be concluded that factors in customer mandate, scope and requirements, execution of the development process and the environment were all considered as important in terms of factors which affect software development project failures in Sri Lankan software development companies.

Factors relating to scope and requirements quadrant were seen by respondents as the most important in terms of software development project failures in Sri Lankan software development companies. Misunderstanding of user requirements was considered the highest ranked factor which affects software development project failures. It was emphasized that software development companies should focus on fully comprehending the requirements of the customer and developing the software accordingly. Three causes, lack of proper communication between team members and customer, incorrect and conflicting user requirements, and lack of domain knowledge of the team members were identified by respondents as the three main causes that lead to misunderstanding of user requirements. These three causes identified were also identified as factors of software development failures. Incorrect and conflicting user requirements was ranked as the 6^{th} most important factor , Lack of domain knowledge of the team members ranked 15^{th} and lack of proper communication between team members and customer ranked 17^{th}. Therefore, it is clear that the project managers have to focus on improving communication between the customer and the development team and eradicate incorrect user requirements from being communicated to the development team. Also, it is clear that the project managers should formulate strategies to improve the domain knowledge of the development team. Further, three factors; poorly defined project scopes, dynamic user requirements and undefined success criteria were also considered important in terms of software development failures in the scope and requirements quadrant.

Lack of change management in the customer organization was seen as the most important factor in the customer mandate quadrant. Deficiencies in change management practices are seen as the main factor from the customer organization which will affect software development project failures. This finding illustrates that the change management techniques are extremely important when installing new software in a company since most software packages will change the work-flow of the company and the employees have to change their working methods in order to get the expected results from the software. Lack of knowledge in change management and the lack of emphasis or importance given to change

management were perceived as being the two main causes of not having proper change management strategies in the customer organization. Lack of management commitment from customer side, lack of user participation and communication between the development team and users were also considered as important factors which affect software development failures.

Poor project management skills were seen as the most important factor affecting software development failures in the execution quadrant. It is implied that the main factor affecting software development failures in the development process (execution) is the poor project management skills of project managers. Lack of planning, and poor project management knowledge of managers were seen as the main two causes of poor project management skills. Lack of communication between team members was also seen as one of the important factors in software development failures. The respondents saw deficiencies in the communication mechanism in the company as the main cause for the problems in communication. Further to these two factors; lack of software documentation, software testing and the knowledge and skill level of staff were also seen as major factors affecting software development failures.

Environmental factors were treated as the least important in terms of factors that affect software project failures with lack of a stable environment in the company ranking as the lowest among the selected factors. The highest ranked factor in the environment quadrant was the internal corporate politics, which implies that internal corporate politics is the most important factor outside the software development project[16] that will affect software development project failures. Poor project management skills and the uncertainty surrounding the future existence of the company were seen as the main causes of the factors in the environmental quadrant.

REFERENCES

Addison, Vallabh, T., Seema (2002). Controlling Software Project Risks, An Empirical Study of Methods Used By Experienced Project Managers. Proceedings of SAICSIT.

Amaratunge, S., (2007) Subject Manual of Research Methods.

Anthony, P., D'Costa. (2002). Software Outsourcing and Development Policy Implications: An Indian Perspective. International Journal of Technology Management, vol., 24, Numbers 7-8 /, 705 – 723.

Bacarini, Salm, G., Love, P. (2004). Management of Risks in IT Projects. Industrial Management and Data Systems, vol., 104, number 4.

Board of Investment of Sri Lanka (BOI). Web site www.boi.lk.

Boehm, B.W. (1991). Software Risk Management: Principles and Practices. Institute of Electrical and Electronics Engineers (IEEE).

Boehm, B.W. (May 1991). Spiral Model of Software Development and Enhancement. Institute of Electrical and Electronics Engineers (IEEE).

Brooks, F.P.Jr., (April/1987). No Silver Bullet: Essence and Accidents of Software Engineering. IEEE Computer; 20, 4, 10-19.

Carr, Marvin, J., Suresh, L., Konda, Monarch, I., Ulrich, C., Walker, C.F. (June 1993). Taxonomy Based Risk Identification. CMU/SEI-93-TR-6, Software Engineering Institute, Carnegie Mellon University.

CHAOS. The Standish Group Report of 1995.

[16]Factors external to software development projects, such as the economic condition of the country or the financial situation of the software development company.

Clemons, E.K. (January 1991). Investments in Information Technology. Communication of ACM vol., 34, no. 1.

Hartman, F., Ashrafi, R.A. (2002). Project Management in the Information Systems and Information Technologies Industries. Project Management Journal, Vol. 33 No. 3, pp. 4-14.

Johne, R., Lyytinen, K., (February 2000). Components Of Software Development Risk: How To Address Them ? A Project Manager Survey. IEEE, Transactions on Software Engineering, 26 (2).

Keil, M., Cule, P., Lyytinen, K., Schmidt, R. (Nov.1998). A Framework for Identifying Software Project Risks. Communication of ACM 41, 11, 76–83.

Kontio, J. (2002). Risk Management: What went wrong and what is the new agenda. 7th European Conference on Software Quality, Helsinki, Finland.

Lacity, C., Willcocks, L., Feeny, D.F., (1996). The Value Of Selective IT Sourcing. Sloan Management Review.

Laudon, K.C., Laudon, J.P. (2004). Management Information Systems : Managing the Digital Firm (Eighth Edition). India: Pearson Education (Singapore) Pvt. Ltd.

Lyytinen, K., Mathiassen, L., Ropponen, J. (1996). A Framework For Software Risk Management. Scandinavian Journal of Information Systems, 8(1):53–68.

May, L.J. (July 1998). Major Causes Of Software Development Project Failures. Cross Talk, the Journal of Defence Software Engineering.

Murthi, S. (September/October 2002). Preventive Risk Management for Software Projects. IT Professional, vol., 04, number 5.

Opatha, H.H.D.N.P., (2006) Subject Manual of Research Methods.

Sherer, A., Susan. (1995). The Three Dimensions of Software Risk. Proceedings of the 28th Annual Hawaii International Conference on System Sciences.

Wallace, L., Keil, M. (April 2004). Software Project Risks and Their Effect on Outcomes. Communication of ACM vol., 47, number 4.