JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 33, 1465-1480 (2017)
DOI: 10.6688/JISE.2017.33.6.5

Identification of Patterns in Failure of Software Projects

BASIT SHAHZAD[1], KINZA MEHR AWAN[2,*], M. IKRAM-ULLAH LALI[3] AND WAQAR ASLAM[4]
[1]*Department of Engineering*
National University of Modern Language
Islamabad, 44000 Pakistan
[2]*Department of Computer Science and Information Technology*
University of Sargodha
Sargodha, 40100 Pakistan
[3]*Department of Software Engineering*
University of Gujrat
Gujrat, 50700 Pakistan
[4]*Department of Computer Science and Information Technology*
The Islamia University of Bahawalpur
Bahawalpur, 63100 Pakistan
E-mail: {basit.shahzad@gmail.com; kinza_awan786@yahoo.com[*]*;*
ikramlali@gmail.com; waqar.aslam@iub.edu.pk}

Software development is not an easy job to manage, in result, many projects end in failure. It has been acknowledged that despite of all mitigation techniques, the successful completion rate of the software till 2016, by the Standish group reports remains 30-40%. With an increase in the scope and investment of the software projects, this failure rate is quite high. In outsourced projects the failure rate is even more significant. The aim of this research is to explore the fundamental reasons of software failure in outsourced and in-house software projects. We have found new patterns to identify the causes of failure in software projects. To address our research questions, systematically we identified different articles from the literature that provide the evidences for causes of failure in software. We have identified thirty-seven different risks of inhouse and thirty-nine risks of outsourced software projects.

Keywords: software risks, causes of failure, inhouse and outsource software development, impact of software risks, categories of risks, SLR on failures

1. INTRODUCTION

Exponential increase in ubiquitous demand of software applications by the users in their routine activities have resulted in rapid evolution in the software industry. However, complexity and risks in software industry are also increased with its rapid growth and expansion [1]. As said by the CIO.com [2], "*Managing an Information Technology job is like juggling chunks of Jelly*".

Despite of all advancement in software development technologies yet successful completion rate of software projects around the globe is just 30%-40% [3]. According to Standish Group reports from 1994-2016 [4], success rate of software development has inclined just from 16% to 32%. This insignificant success rate is always a question for practitioners and researchers. Furthermore, if the software development is outsourced,

Received January 8, 2017; revised February 9, 2017; accepted February 14, 2017.
Communicated by Kashif Saleem.

then the success rate will be even more compromised [5].

Earlier to conclude any software as successful or failed let's first determine the definition of success criteria. According to ISO/ISE, 12207, 2008, p.5, "*an effort with well-planned start and finish dates to develop a product or service in compliance with specified requirement and defined resources*" [6], in line with the cited standard and the traditional practice, criteria to measure the success is a combination of meeting schedule, budget, specified requirements and quality levels. Whereas, de Bakker *et al.* and Glass [7, 8] concluded that apart from conventional success criteria factors there are many other factors which are risk to software development. These risks may serve as a root cause to failure and unfortunately most of them are not properly acknowledged to help identification and mitigation [9].

Somehow these risks can be significantly controlled through project management as it is supporting enough in software development for success through its principles and practices [10]. In addition to this, many other project management umbrella activities should be kept in practice to attain success [11]. Software development methodologies help the project managers and their teams to decide the feasible activities, associated tasks flow, tools and skills that purposely contribute in quality and success against technical, organizational and environment risks [12]. Each of known software development methodology has its own working guidelines but many companies adopted hybrid approach to their solution by applying more than one methodology [13]. Therefore, when it's time to select a software development methodology either for inhouse or outsourcing, companies should rely on something that is best according to conditions [14].

We conducted a systematic literature review to identify the software risks that may cause failures. We discover the patterns in failure of software development from existing literature. The structure of this article is as follows. Section 2 describes the basic concept, section 3 presents the research questions and the structure of review process. In section 4, we examine and analyze the selected primary studies and summarize the results. Section 5 discusses the validity of research study following by section 6 concludes the lesson learned.

2. BASIC CONCEPTS

We conducted a pilot search for Systematic Literature Review (SLR) and recognized that the identification of risks in software development is a complex and highly concerned. Gibbert *et al.* [15] suggests to explain the basic concepts discussed in SLR for ensuring the soundness of research before initiating the study. So, we define the following three basic concepts in our SLR;

i. Inhouse software development versus outsource software development
ii. Risks in software development
iii. Categories of software risks

First, we discuss the difference between the inhouse and outsourced software development. Traxler, [16] stated that an organization in an in-house software development scenario builds its own team for developmental tasks. It works on a pre-defined schedule

and cost for the desired quality. Its success completely depends upon the resources owned by the organization. [17]. It offers many benefits, a full time dedicated team in a single place, hence they can develop and carry change with mutual understanding [18]. The downside of this approach is a lot of time is required to hire and train a developer who fits in the organization's culture. In some cases, if organizations have limited technical resources or time then they seek help from outsourcing.

In outsourcing an organization hires an outer service to accomplish its developmental tasks. An organization can opt for outsourcing due to limited, specialized skills, tools and services [17]. With this, company is capable to bring external resources for defined time schedule. Sometimes the outsourcing services are hired from overseas. Hence no office politics, chit chat and noise affects organization's owned human resources. Disadvantages of outsourcing also exist. One of the major problems is to find an economical and excellent quality service vendor. As hiring a high profile service vendor often comes with high cost and limited time availability due to their busy schedules [18]. List of outsourcing issues includes cultural, linguistic, legal and time shift problems.

The second concept is to elaborate the software risks. They are the integral part of software development in its each activity. Risks have a substantial effect on the outcome of a software project, making risk management a key part of software development. Risk is a situation that can influence the project goals and outcomes, depending upon the probability of occurrence and impact of loss [19]. Software risk management provides a holistic view for risk mitigation in a cyclic process; risk identification, analysis, monitoring, controlling and tracking [20].

Third concept highlights the understanding of the categories of software risks. Based upon the literature survey [21-27], we categorize, the risks in three dimensions; environment/social, technical and organizational risks.

(i) Environmental/Social risks

Nowadays, as software development has become more global, environmental factors influence more as development is carried out in union of multiple off-site teams [21, 25, 27]. Sometimes organization strategy changes with the external factors, like market trends, users taste, unreliable vendors, competitors, *etc.* Hence these factor hosts many risks in the course of software development [26].

(ii) Technical risks

The most discussed and recorded risks in literature are technical risks. Accordingly, more risk mitigation approaches are proposed for these risks. The technical risks are caused from unskillful and incompetent technical staff, unrealistic estimation of schedules and cost, lack of strategy and processes [25]. Poor understanding, assessment and simulation of tools also add a lot in these risks [26].

(iii) Organizational risks

Organization domain focuses upon the approach used by development teams and ensures to meet the planned work break down activities to develop a winning project [22]. Its risk impacts upon the interpersonal relationship between stakeholders, organizational politics, lack of communication adequate skills.

3. RESEARCH QUESTION AND REVIEW PROCESS

To proceed, this review, we composed three research questions in section 3.1. Later sections from sections 3.2 to 3.6, define systematic review process, to answers our first two research questions. The guidelines suggested in [28] are considered while planning this study. Answer to the third research question is figured out from the results of first two questions.

3.1 Research Questions

In pilot searches, we witnessed that multiple research studies have been conducted which explore various risk factors in software development. But there is no concrete evidence that separately figure out the risks in outsourced and inhouse software projects. To overcome these limitations, we formulated the following questions;

*RQ*1: What are the risk factors that effect in-house software development?
*RQ*2: What are the risk factors that effect out-source software development?
*RQ*3: How the risk factors of in-house software development and outsource software development are different?

3.2 Data Source and Search Strategy

We performed search queries using the scientific database and their search engines from the publishers. Therefore, we selected the following available scientific databases;

i. ACM Digital Library (portal.acm.org/dl.cfm);
ii. Elsevier Science Direct (www.sciencedirect.com) ;
iii. IEEE Electronic Library (ieeexplore.ieee.org);
iv. SpringerLink (www.springerlink.com); and
v. Emerald (www.emeraldinsight.com).

Some of the known databases in our knowledge but not included because we don't have open access to them.

According the Kitchenham and Menezes's three viewpoint guidance (population, intervention & outcome) for extracting data, following list of terms are formulated concord to our research question later used in search queries for the selection of articles [29, 30].

a) For *RQ*1: inhouse, software development, risk, failure, factor, effect, cause
b) For *RQ*2: outsource, software development, risk, failure, factor, effect, cause

We also used synonyms of the selected terms in the search strings. The search strings have been composed using the logical expressions between the search terms. Each search string has been carefully formulated according to the constraints of specific databases.

The search strategy states the list of instructions and checks that are obeyed and observed during the collection of data for primary studies. Hence, this research work has following inclusion and exclusion criteria;

1. The selected data for primary studies belongs to the domain of software engineering, in context of risk management.
2. Only full length conference or journal published articles and book chapters are considered in primary studies to justify an evidence based review.
3. The selected data is only in English language to overcome the limitation of linguistic awareness.
4. No published data before 2010 was considered to include only current trends.
5. Only considered those articles that were either free or openly available through the courtesy of Higher Education Commission of Pakistan.

3.3 Article Selection

The title and the abstract of the retrieved article is read by the researcher for its biasness and relevance else the full text article is read for its validity and soundness as desired. If the same article has been extracted from different databases in response to the requested query, it's mandatory to remove the duplication. It's also ensured that the results of the selected article should be from some evidence based research.

3.4 Quality Assurance Procedures

Some more quality assurance constraints have been followed to improve the relevance and minimizing the biasness. Quality assessment criteria has been designed by keeping in mind the guidance for internal and external validity as given by [29, 32]. Appendix A, represents the quality assessment form, used in this systematic literature review. In quality assessment form, data has been collected for each study to ensure its substantial worth.

3.5 Data Extraction Plot

This section describes the useful information that has been gathered from each selected primary study. Appendix B depicts the form that initially documents the publication details from each primary study which has been ranked successful, after evaluation through the quality assessment criteria.

In Appendix C, the data extraction plot is exhibited, which extracts causes of software risks from each primary study. Then, all the extracted data has been examined for holistic assessment to detect the patterns of failure in software development.

3.6 Data Composition Plan

Finally, results have been compiled, from the collected data. As data has been extracted from different studies each of that has its own methodology. Therefore, the results are exhibited in tabular format.

4. RESULTS AND DISSCUSSION

Execution of search strings and retrieval of data from the databases has been com-

pleted in three months, June 2016 to August 2016. The number of steps followed in data collection can be seen in Fig. 1. This yields 2116 articles in response to RQ1 and 2391 relevant studies for RQ2, from the data sources.

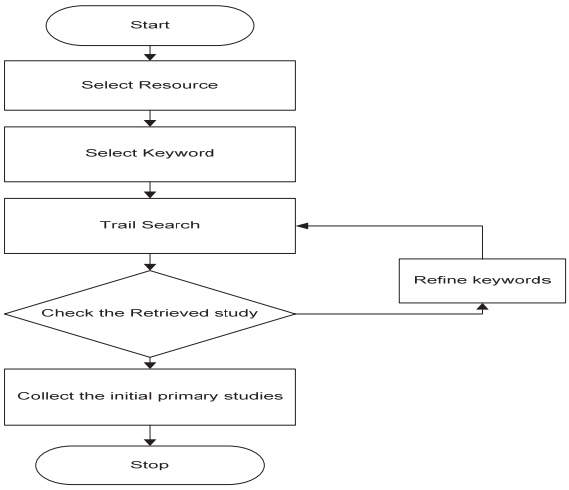

Fig. 1. Search strategy.

Data selection criteria mentioned in section 3.3 has been applied to 4507 selected research articles. After reading abstract 128 articles are chosen. Later 72 studies have been shortlisted after reading full length articles. Further 39 more studies have been discarded as they didn't satisfy our quality evaluation criterion. List of retrieved and selected articles are mentioned in Table 1.

Table 1. List of retrieved results from the selected data sources.

S/n	Data Source	Initially Scanned		1st Filtered		2nd filtered		Finally selected	
		SS1	SS2	SS1	SS2	SS1	SS2	SS1	SS2
1	IEEE Xplore	779	564	24	29	12	20	5	10
2	Google Scholar	1142	1438	15	21	24	10	12	4
3	ACM digital library	113	218	09	11	0	0	0	`0
4	Science Direct	58	103	5	08	1	5	1	1
5	Springer	24	68	2	4	0	0	0	0
	Total	**2116**	**2391**	**55**	**73**	**37**	**35**	**18**	**15**

Eighteen research articles relevant to RQ1, and fifteen research articles relevant to RQ2, have been selected according to the Quality assessment procedure (see Appendix A). Each research study after evaluation is awarded a quality assessment score. Research articles mentioned in Table 2 refers the primary studies selected for this SLR.

Table 2. Selected research articles for primary study in response to *RQ*1 & *RQ*2.

ID #	Articles Reference to the Results for Research Question # 1	ID #	Articles Reference to the Results for Research Question # 2
[1]	Hashim *et al.*, (2013, October)	[1]	Persson & Mathiassen (2010)
[2]	Sonchan, & Ramingwong, (2014, May)	[2]	Chadli & Idri (2015, October)
[3]	Antinyan *et al.*, (2014, October)	[3]	Khan *et al.*, (2014)
[4]	Sipayung,&Sembiring, (2015, November).	[4]	Colomo-Palacios *et al.*, (2012)
[5]	Rommel & Gutierrez (2012)	[5]	Ikediashi *et al.*, (2012)
[6]	Sundararajan *et al.*, (2013)	[6]	Ahmed *et al.*, (2014)
[7]	Abdullah *et al.*, (2015)	[7]	Khan & Khan (2014, April)
[8]	Shahzad & Al-Mudimigh (2010, July)	[8]	Ali & Khan (2014, August)
[9]	Christiansen *et al.*, (2006)	[9]	Liu & Yuliani (2016)
[10]	Elzamly & Hussin, (2014)	[10]	Khan & Azeem (2014)
[11]	Arnuphaptrairong, T. (2011, March)	[11]	Zafar *et al.*, (2011, December)
[12]	Shahzad & Safvi, (2010)	[12]	da Silva *et al.*, (2010, August)
[13]	Elzamly *et al.*, (2016)	[13]	Iqbal *et al.*, (2012, July)
[14]	Hoermann *et al.*, (2010)	[14]	Nurdiani *et al.*, (2011, August)
[15]	Hoermann *et al.*, (2011)	[15]	Verner *et al.*, (2012)
[16]	Stefanova & Georgiev (2014)		
[17]	López & Salmeron (2012)		

Data from the primary studies have been extracted in the specified format in two different predesigned forms. The First form (see Appendix B) use to extracts the general information from the primary studies. In second form, the details of the risk factors exclaimed in each selected primary study have been tabulated (See Appendix C).

4.1 Data Composition

This section presents the risk factors affecting the software development, with respect to their frequency of occurrence in the selected primary studies. Further percentage of each identified risk factor has been calculated to assign it a severity level, according to following criteria [24, 33, 34]:

i. Ignorable 1%-09%
ii. Insignificant 10%-18 %
iii. Tolerable 19%-49%
iv. High 51%-75%
v. Catastrophic 76%-100%

Based upon the selected primary studies, we extracted risk factors, depicted in Table 3, as the main cause of failure in inhouse software development.

Furthermore, our study reveals risks factors listed in Table 4, are the main cause of software failure in outsourcing in software development.

Table 3. Identified risk factors in inhouse development.

s/n	Risk factor	Primary study	Frequency *N*=18	Percentage	Severity Level
1	Overrun Budget & resources	1, 2, 3, 4, 5, 7, 8, 9, 10 (A), 12, 13, 14, 15, 16	14	78%	Catastrophic
2	Unrealistic estimated schedule	1, 2, 3, 4, 5, 7, 8, 9, 10 (A), 12, 13, 16, 17	13	72%	High
3	Technical complexity (new tools)	2, 4, 5, 6, 8, 9, 10 (B), 12, 13, 14, 15, 17	12	67%	High
4	Creeping scope	1, 2, 3, 4, 5, 6, 9, 10 (B), 11, 13, 16, 17	12	67%	High
5	Lack of technical skills	2, 4, 5, 6, 7, 8, 9, 10 (A), 12, 13, 16, 17	12	67%	High
6	Lack of top management involvement	2, 4, 5, 6, 7, 10 (A), 11, 13, 14, 15, 17	11	61%	High
7	Incomplete requirement specification	1, 2, 3, 4, 8, 10 (B), 12, 13, 14, 17	10	56%	Tolerable
8	staff attrition / turnover	2, 4, 6, 8, 9, 10 (A), 12, 13, 16, 17	10	56%	Tolerable
9	Lack of user involvement	2, 4, 5, 6, 9, 10 (A), 11, 13, 17	9	50%	Tolerable
10	Unclear and Ambiguous requirement	3, 5, 6, 9, 10 (B), 11, 13, 14	8	44%	Tolerable
11	Team collaboration	2, 4, 5, 7, 9, 10 (A), 13, 17	8	44%	Tolerable
12	No proper WBS	5, 7, 8, 10 (B), 13, 14, 15, 17	8	44%	Tolerable
13	Inadequate organization infrastructure	2, 4, 5, 10 (A), 12, 13, 15	7	39%	Tolerable
14	Inadequate organizational strategies	5, 9, 10 (A, B), 13, 14, 15	7	39%	Tolerable
15	Poor project management skills	5, 9, 10 (A, B), 11, 13, 17	7	39%	Tolerable
16	Lack of commitment	5, 6, 11, 13, 14, 15, 17	7	39%	Tolerable
17	Conflict between stakeholders	5, 6, 7, 9, 14, 16, 17	7	39%	Tolerable
18	Improper utilization of reusable components	3, 5, 6, 8, 12, 13	6	33%	Tolerable
19	Improper project planning	1, 9, 10 (A, B), 13, 17	6	33%	Tolerable
20	Poor/Lack of automation tools	2, 4, 5, 6, 9, 17	6	33%	Tolerable
21	Insufficient testing	2, 3, 4, 8, 13, 17	6	33%	Tolerable
22	Expectation gap	5, 6, 11, 14, 15, 17	6	33%	Tolerable
23	Lack of Communication	1, 2, 4, 5, 16, 17	6	33%	Tolerable
24	Gold plated requirement	1, 6, 10 (A, B), 13	5	28%	Tolerable
25	Inappropriate selection of process	2, 4, 9, 14, 15	5	28%	Tolerable

26	Lack of HR training	5, 9, 12, 13	4	22%	Tolerable
27	inadequate knowledge base	8, 10 (A), 12, 13	4	22%	Tolerable
28	improper CCB	1, 13, 14	3	17%	Tolerable
29	untraceable require-ments	3, 5, 10 (B)	3	17%	Tolerable
30	Faulty codes	3, 12, 13	3	17%	Tolerable
31	Lack of QA activates	13, 14, 15	3	17%	Tolerable
32	User resist to change	5, 9, 17	3	17%	Tolerable
33	Poor documentation	13, 14, 17	3	17%	Insignificant
34	Poor design	3, 13	2	11%	Insignificant
35	Natural disasters	8, 12	2	11%	Insignificant
36	Inadequate Risks mitigation	14, 15	2	11%	Insignificant
37	Unclear contracts	14	1	6%	Insignificant

Table 4. Identified risk factors in outsource development.

s/n	Risk factor	Primary study	Frequency N=15	Percentage	Severity Level
1	Variation in culture	1, 2, 3, 4, 5, 7, 10, 12, 13, 14, 15	11	73%	Catastrophic
2	Lack of adequate skills set	1, 2, 6, 7, 8, 9, 10, 11, 12, 13, 14	11	73%	Catastrophic
3	Lack of task mutual clarity & under-standing	1, 2, 3, 5, 9, 10, 11, 13, 14, 15	10	67%	High
4	lack of development task coordination	1, 2, 4, 7, 9, 10, 11, 12, 14, 15	10	67%	High
5	lack of synchronous teams communication	2, 3, 4, 7, 9, 11, 12, 13, 14, 15	10	67%	High
6	Scarce mutual Process understanding	1, 2, 4, 6, 7, 10, 11, 12, 14, 15	10	67%	High
7	Insufficient knowledge capturing across other site	1, 2, 4, 7, 9, 10, 11, 14, 15	9	60%	High
8	Lack of trust across site	1, 2, 3, 4, 10, 11, 12, 14, 15	9	60%	High
9	Lack of mutual pro-ject management	4, 5, 6, 7, 9, 10, 12, 14, 15	9	60%	High
10	Time zone difference	1, 2, 3, 11, 12, 13, 14, 15	8	53%	High
11	No proper WBS	1, 4, 7, 9, 10, 11, 12, 15	8	53%	High
12	Language difference	1, 2, 7, 10, 12, 13, 14, 15	8	53%	High
13	Deviation in work environment	1, 2, 3, 10, 11, 12, 14, 15	8	53%	High
14	Difference in legal norms/laws	2, 4, 5, 6, 7,12, 13, 14	8	53%	High
15	Variation in mutual	5, 7, 9, 10, 11, 12, 14,	8	53%	High

	management support	15			
16	Conflict in team goal across site	1, 2, 5, 9,12, 13, 15	7	47%	Tolerable
17	Difference in stake-holder commitment	1, 2, 7, 9, 13, 14, 15	7	47%	Tolerable
18	Limited social inter-action	1, 2, 4, 5, 10, 14, 15	7	47%	Tolerable
19	Variation in staff experience	2, 5, 9, 11, 13, 14, 15	7	47%	Tolerable
20	Variation of product quality Achieved	4, 5, 7, 8, 10, 12, 15	7	47%	Tolerable
21	Poor contract documentation	5, 7, 8, 9, 10, 13, 15	7	47%	Tolerable
22	Conflict in budget	5, 7, 8, 9, 12, 13, 15	7	47%	Tolerable
23	Breach of intellectual property	5, 6, 10, 12, 13, 14, 15	7	47%	Tolerable
24	Limited face to face conferencing	1, 2, 3, 11, 14, 15	6	40%	Tolerable
25	Variation in organi-zation infrastructure	2, 10, 11, 12, 14, 15	6	40%	Tolerable
26	Delay feed back	2, 3, 10, 11, 13, 14	6	40%	Tolerable
27	Fear of external competitors risks	5, 6, 7, 8, 13, 15	6	40%	Tolerable
28	Absence of task ownership	5, 9, 10, 11, 12, 14	6	40%	Tolerable
29	Risk of poor esti-mated time	9, 10, 12, 13, 15	5	33%	Tolerable
30	inappropriate task distribution	2, 12, 14, 15	4	27%	Tolerable
31	Security issues	5, 6,11,14	4	27%	Tolerable
32	Improper risk mitiga-tion approaches	5, 8, 12, 15	4	27%	Tolerable
33	Poor outsourcing vendor selection	5, 7, 9, 13	4	27%	Tolerable
34	Chance of confiden-tial leaks	5, 10, 15	3	20%	Tolerable
35	Political instability	5, 7, 15	3	20%	Tolerable
36	Natural disasters	5, 13	2	13%	insignificant
37	Staff turnover	13, 14	2	13%	insignificant
38	Compatibility issues	1	1	7%	Ignorable
39	Conflict in labor wages	6	1	7%	Ignorable

4.1 Difference between the Risk Factors of Inhouse and Outsource Projects

The top ten risks from Tables 3 and 4 according to their frequency of occurrence are extracted. Critical analysis is carried out to discover the difference in risks of both approaches according to their dimension to answer our third research question, as shown in Table 5.

Table 5. Top ten risks factors in inhouse & outsourced software projects.

	Inhouse software development		Outsource software development	
S/n	Risk Dimension	Risk Factor	Risk Factor	Risk Dimension
1	Organizational Risks	Overrun Budget & resources	Variation in culture	Environmental/ Social/ Risks
2	Organizational risks	Unrealistic estimated schedule	Lack of adequate skills set	Technical risks
3	Technical risks	Technical complexity (new tools)	Lack of task mutual clarity & under-standing	Organizational risks
4	Technical risks	Creeping scope	lack of development task coordination	Organizational risks
5	Technical risks	Lack of technical skills	lack of synchronous teams communication	Organizational risks
6	Organizational risks	Lack of top manage-ment involvement	Insufficient mutual Process understand-ing/coordination	Organizational risks
7	Technical risks	Incomplete require-ment specification	Insufficient knowledge capturing across other site	Organizational risks
8	Organizational risks	staff attrition / turnover	Lack of trust across site	Organizational risks
9	Organizational risks	Lack of user involvement	Lack of mutual project management	Organizational risks
10	Technical risks	Unclear and ambigu-ous requirement	Time zone difference	Environmental/ Social/ Risks

The comparison reveals that key risks to outsourced projects occur in organizational dimension followed by social and technical dimension. Whereas in inhouse development main cause of failure are organization and technical risks. It has been concluded, that no significant cause of failure in inhouse development belongs to social dimension.

5. VALIDITY OF LITERATURE REVIEW

Our review study is conducted in a systematic designed process as shown in Fig. 1. Whereas during the construction of search strings and selection of primary studies some assumptions has been considered, depending upon the common sharing of concepts among the authors. The documented procedure of review process, guaranteed the validity of the search, choice and evaluation of articles. There may exist some threats to soundness in subjective evaluation of the articles. However, these threats are weakened by reasoning the analysis process from multiple authors.

Another threat to the validity of review process is that, it is based upon the results provided by the different search engines. The search is systematic and can be repetitive but each time the produced results may not be same, due to the expanding nature of digital databases [31, 35].

Here it is possible that we may be overlooked few relevant articles in results. However, we involved multiple authors in selection process, thus reducing this possibility.

6. CONCULSION

Software development is a difficult process and necessitates efficient handling of the available resources and tools. Poor software development planning and associated skills invites multiple risk factors. Efficient principles and practices of project and risk management guides a lot to overcome software risks. It is firmly believed that successful application of risk management lies in the identification of all possible know & hidden risks. Therefore, in this article, we separately identified software risk list for outsourced and inhouse development. Hence our focus is to handle risks through their early identification rather than applying costly mitigation techniques.

We conducted an extensive systematic literature review to achieve high validity. According to the selection criteria, we extracted 4507 different articles from multiple digital databases. After applying inclusion/exclusion and quality assessment criteria, and different refinement approaches, finally we get 33 relevant articles to answer our first two research questions. After critical examination of these articles, we have identified 37 risks of inhouse and 39 of outsourced software development. In our research, we assigned frequency to each identified risk factor, depending upon the magnitude of its evidence extracted from the literature. Percentage of occurrence of each risk, has been also calculated to rank it severity level. Later, Top ten risk factors of each development approach are acutely analyzed. Hence concluding that major causes of failure in inhouse development exists in organizational and technical dimensions whereas risks in outsourced development exists in social and organizational dimensions.

Utmost effort has been applied to identify maximum possible risk factors from the available literature. These identified software risks are expected to offer a better understanding to identify and handle the major causes of failure of a software project. The list of identified risk factors may grow in future with larger samples of primary studies. This study may support other researchers to improve the paradigm of risk management in software development. It also assists them to introduce better mitigation techniques for each identified risk.

REFERENCES

1. A. A. Keshlaf, "WeDRisk: an approach to managing web and distributed software development risks," in *EBSE*, 2007-001, Keele University.
2. D. L. Pells, "Seven good reasons for the rapid growth of project management in IT and why that trend will continue," *PM World Today*, Vol. 10, 2008, pp. 1-14.
3. J. L. Eveleens and C. Verhoef, "The rise and fall of the chaos report figures," *IEEE Software*, Vol. 27, 2010, p. 30.
4. Manifesto, Chaos, *Think Big*, *Act Small*, The Standish Group International Inc., 176, 2013p.
5. P. Savolainen, J. J. Ahonen, and I. Richardson, "Software development project success and failure from the supplier's perspective: A systematic literature review," *International Journal of Project Management*, Vol. 30, 2012, pp. 458-469.
6. K. Bakker, A. Boonstra, and H. Wortmann, "Does risk management contribute to IT project success? A meta-analysis of empirical evidence," *International Journal of Project Management*, Vol. 28, 2010, pp. 493-503.

7. R. L. Glass, "Frequently forgotten fundamental facts about software engineering," *IEEE Software*, Vol. 18, 2001, pp. 112-111.
8. N. Agarwal and U. Rathod, "Defining 'success' for software projects: An exploratory revelation," *International Journal of Project Management*, Vol. 24, 2006, pp. 358-370.
9. K. E. Papke-Shields, C. Beise, and J. Quan, "Do project managers practice what they preach, and does it matter to project success?" *International Journal of Project Management*, Vol. 28, 2010, pp. 650-662.
10. R. Zheng, R. Yin, and Y. Tao, *Practical Software Engineering*, Tsinghua University Press, Beijing, Vol. 4, 1997, pp 27-28.
11. K. Schwaber and M. Beedle, *Agilè Software Development with Scrum*, Upper Saddle River, Prentice Hall, NJ, 2002, pp. 94-100.
12. E. M. Simão, "Comparison of software development methodologies based on the SWEBOK," Doctoral dissertation, Department of Informatics, University of Minho, 2011.
13. J. M. Verner and W. M. Evanco, "In-house software development: what project management practices lead to success?" *IEEE Software*, Vol. 22, 2005, pp. 86-93.
14. M. Gibbert, W. Ruigrok, and B. Wicki, "What passes as a rigorous case study," *Strategic Management Journal*, Vol. 29, 2008, pp. 1465-1474.
15. D. Traxler, "Outsourcing v. hiring in-house: Pros and cons," *Practical Ecommerce*, 2012.
16. Clydebuilt Business Solutions Ltd., "Developing in-house vs. off the shelf," 2012, http://www.clydebuiltsolutions.com.
17. S. Fayman, "Should you hire an in-house developer or outsource overseas?" 2013, https://www.kissmetrics.com.
18. R. Prikladnicki and J. L. N. Audy, "Managing global software engineering: A comparative analysis of offshore outsourcing and the internal offshoring of software development," *Information Systems Management*, Vol. 29, 2012, pp. 216-232.
19. B. W. Boehm, "Software risk management: principles and practices," *IEEE Software*, Vol. 8, 1991, pp. 32-41.
20. Q. Khan and S. Ghayyur, "Software risks and mitigation in global software development," *Journal of Theoretical and Applied Information Technology*, Vol. 22, 2010, pp. 58-69.
21. B. Boehm and R. Turner, "Management challenges to implementing agile processes in traditional development organizations," *IEEE Software*, Vol. 22, 2005, pp. 30-39.
22. T. Chow and D. B. Cao, "A survey study of critical success factors in agile software projects," *Journal of Systems and Software*, Vol. 81, 2008, pp. 961-971.
23. B. Shahzad, I. Ullah, and N. Khan, "Software risk identification and mitigation in incremental model," in *Proceedings of IEEE International Conference on Information and Multimedia Technology*, 2009, pp. 366-370.
24. A. Sharma, S. Sengupta, and A. Gupta, "Exploring risk dimensions in the Indian software industry," *Project Management Journal*, Vol. 42, 2011, pp. 78-91.
25. H. I. Mathkour, B. Shahzad, and S. Al-Wakeel, "Software risk management and avoidance strategy," in *Proceedings of International Conference on Machine Learning and Computing*, Vol. 3, 2011, pp. 477-481.

26. M. Keil, P. E. Cule, K. Lyytinen, and R. C. Schmidt, "A framework for identifying software project risks," *Communications of the ACM*, Vol. 41, 1998, pp. 76-83.
27. Keele, Staffs, "Guidelines for performing systematic literature reviews in software engineering," Technical Report, Ver. 2.3, EBSE. 2007.
28. B. Kitchenham, "Procedures for performing systematic reviews," Keele, UK, Keele University, Vol. 33, 2004, pp. 1-26.
29. J. Menezes Jr, C. Gusmão, and H. Moura, "Defining indicators for risk assessment in software development projects," *Clei Electronic Journal*, Vol. 16, 2013, p. 11.
30. M. Jørgensen and M. Shepperd, "A systematic review of software development cost estimation studies," *IEEE Transactions on Software Engineering*, Vol. 33, 2007, pp. 33-53.
31. D. Evans, "Hierarchy of evidence: a framework for ranking evidence evaluating healthcare interventions," *Journal of Clinical Nursing*, Vol. 12, 2003, pp. 77-84.
32. T. W. Kwan, "A risk management methodology with risk dependencies," Doctoral dissertation, Department of Computing, The Hong Kong Polytechnic University, 2010.
33. C. R. Pandian, *Applied Software Risk Management: A Guide for Software Project Managers*, CRC Press, NY, 2006, pp. 46-47.
34. P. Brereton, B. A. Kitchenham, D. Budgen, M. Turner, and M. Khalil, "Lessons from applying the systematic literature review process within the software engineering domain," *Journal of Systems and Software*, Vol. 80, 2007, pp. 571-583.
35. B. Kitchenham, R. Pretorius, D. Budgen, O. P. Brereton, M. Turner, M. Niazi, and S. Linkman, "Systematic literature reviews in software engineering – a tertiary study," *Information and Software Technology*, Vol. 52, 2010, pp. 792-805.

APPENDIX-A

Quality Assessment Form

S/n	Quality Assessment Question	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[10]	[11]	[12]	[13]	[14]	[15]	[16]	[17]
1	**Research Problem:**																		
	Does the study explicitly stated its problem?	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	Does it recommend future work?	1		1	1	1	1						1	1	1				
2	**Literature Reference**																		
	Do the study presents a satisfactory literature review?	1	1		1	1	1	1	1		1	1	1	1	1	1	1		
3	**Research Methodology**																		
	Is the research comprehends on a scientific research methodology?	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	Is evaluation technique is stated?	1	1	1	1		1	1	1	1	1	1	1		1	1	1	1	1
	Is any statistical technique is applied?		1		1	1	1				1	1			1		1		1
4	**Outcomes**																		
	Has a handful evidence provided after the analysis?	1	1	1		1	1	1	1	1	1	1	1	1	1	1	1	1	1
	Does the extracted evidence justify the conclusion?	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	Total Score	7	7	6	7	7	8	6	6	5	7	7	7	6	8	6	7	5	6

APPENDIX-B

Study ID	Study title	Authors	Journal/ Conference	Publication Year	Keywords	Date of review conduction	Organization type
[1]	Hashim et al., (2013, October).	Rabia Hashim	Conference	2013	*Risk Factors; Risk Estimate; Risk Uncertainty; Project Failure*	20-6-16	Academic
[2]	Sonchan, & Ramingwong, (2014, May).	Pontakon Sonchan	conference	2014	*Risk Management; Software Project Management; Risk Identification; Risk Analysis; Delphi Study.*	20-6-16	Academic

APPENDIX-C

Study ID	Causes of Failure	Method of Study	Year of Study	Country of study	Database	Quality Assessment
[1]	Overrun Budget & resources, unrealistic estimated schedule, Creeping scope, Incomplete requirement specification, gold plated requirement, Improper project planning, Lack of Communication, improper change control board	SLR	2013	Pakistan	IEEE	7
[2]	Overrun Budget & resources, Technical complexity(new tools) , unrealistic estimated schedule, Creeping scope, Lack of top management involvement, Incomplete requirement specification, Lack of technical skills, staff attrition /	Content analysis and Delphi Study	2014	Thailand	IEEE	7

Basit Shahzad earned his Ph.D. from University Technology Petronas, Malaysia, and his MSc degree from National University of Science and Technology, Islamabad, Pakistan. Dr Shahzad, is a Visiting Scientist at University of Cambridge, UK and Visiting Fellow of Macquarie University, Australia. Dr Shahzad is a Collaborating Researcher with the Hagenberg Centre for Software Competence, Austria. His research and teaching career span over 16 years. Dr. Shahzad is currently at Faculty of Engineering and Computer Science, NUML, Islamabad as an Assistant Professor. Before this he has served King Saud University, Riyadh and has worked as an Assistant Professor at COMSATS Institute of Information Technology, Islamabad. Dr. Shahzad has numerous publications in journals and conferences of international repute and has a very active research profile. He has editorial role in several conferences and journals of high repute and has edited numbers of special issues in significant journals in the areas of software engineering, social networks, and mobile healthcare. Dr. Shahzad is reviewer of several high impact journals and is in teh program committee of several distinguished conferences. Dr. Shahzad values 'research' a lot and his career objectives revolve around conducting and supervising research in Information Systems (Enterprise Architecture, Software cost and risk modeling, Mitigating technological risks in modern banking), Advancements in Research Methodologies (Quantitative, Qualitative, Mixed method), mobile healthcare, and Social Networks (Theory, Empirical Social Network Analysis, Micro-blogging based analysis) *etc*.

Kinza Mehr Awan is working as a Lecturer in the Department of Computer Science and IT, University of Sargodha, Pakistan. She did her BS is Software Engineering and MS in Computer Science from the University of Sargodha, Pakistan. She is active young researcher in the area of software engineering, requirement engineering, risk management and software testing.

M. IkramUllah Lali is working as an Associate Professor at the Department of CS, University of Gujrat, Pakistan. He obtained Masters in Software Engineering and Ph.D. CS degrees from COMSATS, Islamabad, Pakistan. His areas of interests include machine learning, social network data analysis, formal methods

and software testing. He is author of more than 40 research articles which have been published in conferences and reputed journals.

Waqar Aslam received the Ph.D. degree in Computer Science from The Eindhoven University of Technology, The Netherlands. He is an Assistant Professor at the Department of Computer Science and IT, The Islamia University of Bahawalpur, Pakistan. His research interests include performance modeling of (distributed) software architectures, effort/time/cost estimation of software development in (distributed) Agile setups, performance modeling and QoS of wireless/computer networks and social network data analysis.