Proceedings of the Pakistan Academy of Sciences 52 (4): 285–300 (2015)
Copyright © Pakistan Academy of Sciences
ISSN: 0377 - 2969 (print), 2306 - 1448 (online)

Pakistan Academy of Sciences

Research Article

Systematic Literature Review Protocol for Green Software Multi-sourcing with Preliminary Results

Muhammad Salam, and Siffat Ullah Khan*

Department of Computer Science and Information Technology
University of Malakand, Lower Dir, Pakistan

Abstract: Green Software Engineering is a research area gaining reputation rapidly in order to create green software. Software development multi-sourcing is a modern Global Software Engineering (GSE) pattern for creating high quality software at minimum cost and time at low wages countries by contracting out the software development work to multiple vendors located at multiple geographical locations across the globe. Green Software multi-sourcing is a modern research area which lies at the intersection of both the aforementioned research areas (Green Software Engineering and Software Multi-Sourcing). The goal of this paper is to present the systematic literature review (SLR) protocol which has been developed, validated and in the process of implementation. The implementation of the SLR protocol will yield in the identification of success/risk factors, and real-world green practices to be considered by multi-sourcing vendor organizations that can contribute towards the development of green and sustainable software. Systematic Literature Review procedure has been used as research practice. SLR is a new research method in software engineering which is dissimilar from conventional literature analysis. SLR is more comprehensive and is grounded on a pre-defined protocol. We have developed a Systematic Literature Review (SLR) protocol and are currently in the process of its implementation. The anticipated conclusions of the SLR execution will be the identification of success/risk factors and practices that can contribute towards the development of green software in multi-sourced projects.

Keywords: Green software development, green software multi-sourcing, systematic literature review protocol, green software multi-sourcing models

1. INTRODUCTION

This study is inspired by two interests: the need for sustainable and green software development and the development of a model for measuring vendors' green capabilities in software multi-sourced projects. Green computing is an emerging research field and several solutions have been proposed for this purpose [1].

Green computing can be demarcated as the employment of maximizing the proficient use of computing assets to minimize its deleterious influence on environment [2]. The environmental effects and concerns due to information and communication technologies (ICTs) have flagrantly raised these days [3-6]. Business organizations and ICT firms consider green computing as an integral part of their overall business strategies [7]. From both business and consumers perspective energy efficiency is on the highest priority. The reasons reported include, increasing energy outlays and ecological change, originated by mounting global warming and greenhouse gas [8].

Literature reveals that in recent years a number of struggles have been made while creating green and sustainable software. Several researchers has focused on hardware oriented solutions for environmental sustainability whereas others have worked on software oriented solutions [9]. Imtiaz and Mahmoud [10] have proposed green model having two levels, for sustainable software engineering, the first level presents guidelines for green software development and the

Received, April 2015; Accepted, November 2015
* Corresponding author: Siffat Ullah Khan; Email: siffatullah@uom.edu.pk

second level is consist of methods occupied by the software during its execution to support green computing. Erdelyi [11] has presented green rules for developing long-living software in the industrial automation domain. Appasami and Suresh [12] have presented some general guidelines and tips for green computing. GREENSOFT model for sustainable software has been presented by [13], the model supports software engineers/administrators/software users in building, sustaining, and using software in a more ecological way. Shenoy and Earatta [14] have presented a model for green software development in which they suggest some guidelines and practices that support sustainability. Some green software quality factors are presented by [15] which contribute in green software development. Mahaux and Canon [16] have argued that requirements engineering is critical for sustainable software development in the whole software life cycle. Other studies [17, 18] also argue that requirements engineering play important role in sustainable software development. According to Imtiaz and Mahmoud [10] software manages and operates the underlying hardware and therefore has indirect effect on the environment. Similarly software oriented practices can utilize hardware resources efficiently and limit the need for new hardware due to updates. Sedef et al., [19] came up with a guide for green software developers, in the context of environmental sustainability and software quality criteria.

The latest literature in the domain reveals that there is scarcity of software engineering models and tools that can well define how software can be designed and maintained in an eco-friendly manner [14]. Software engineering researchers have developed keen interest in green software development due to the demand of environmental sustainability [20]. Academia and software houses consider green software engineering as a strength due to its vital role in environmental sustainability and energy consumption [1]. The ICT power utilization is increasing day by day. Consequently, a number of hardware oriented solutions exist, but the involvement of green software is still not well matured [21]. A diversity of research on green computing has generally focused on ecological sustainability in terms of computer hardware [10]. Green Software has been defined in the literature as, that Software, whose direct and indirect negative influences on economy/humanity/ human beings/environment due to its creation, deployment, and usage minimal and/or which has a positive consequence on sustainable development [21].

In voluminous software development tasks, developers treat sustainability as postscripts because developers/designers are focused by time-to-market force and are often less conversant about green software practices and techniques [22]. Creating energy efficient and green software is becoming popular subject rapidly [23]. Various aspects of green computing and green ICT have been explored by researchers and scholars in general [11, 24-31]. However, very petite attention has been compensated to the development of green software in multisourcing environment. Multi-sourcing is a contemporary paradigm in software outsourcing domain which offers the benefits of using multiple vendors for the development of software in a shorter time span. The outsourced software is developed in modules/components by various software vendors to expedite the development process [32].

Consequently, Green Software Multi-sourcing has developed a new area of research interest in Global Software Engineering domain in order to develop green and sustainable software in multi-sourced software development projects. Research in this area will assist multi-sourcing industry to deliver green software in general and to gauge vendors' readiness for the development of greener software in particular. For this purpose we have proposed Green Software Multi-sourcing Readiness Model. The proposed structure of the model has been published [33]. In current paper we have presented/exposed our systematic literature Review (SLR) protocol. The preliminary/pilot results of the SLR protocol are also presented.

2. OBJECTIVES

Our main objective is to develop Green Software Multi-souring Readiness Model (GSMRM), to assist multi-sourcing vendor organizations in assessing their readiness for developing green software in multi-sourced projects. The preliminary structure of the model has been published [33].

The proposed model will be developed in three phases i.e. first phase is protocol development and its implementation, second phase

is empirical validation of protocol findings and the third phase is model development and evaluation phase as shown in Fig.1.

In this paper we have presented the first segment/phase of our ongoing research i.e. systematic literature review protocol. We have produced the SLR protocol for identification of the success factors, risk factors and real-world green practices to be considered by multi-sourcing vendor organizations that can contribute towards the development of green and sustainable software in multi-sourced software development projects. While designing our protocol, we studied a number of SLR protocols for guidance [34-36].

3. RESEARCH METHODOLOGY

In order to fulfil the aim of our research/study project, the research work will be carried out in three phases/stages. In the first stage success factors, risk factors and their practices, regarding the development of green and sustainable software, will be investigated via Systematic Literature Review process. A systematic literature review (SLR) process is a new research method in the software engineering domain which is used for the identification, assessment and clarification of existing literature on a specific research topic/research question [37]. According to [38, 39] SLR has become an imperative research methodology in software engineering. SLR is difficult to conduct as compared to ordinary literature review. However the results generated by SLR are comparatively unbiased and to some extent repeatable because it utilizes a pre-defined and validated protocol [40].

In second phase we will conduct empirical studies in software multi-sourcing industry in order to authenticate the SLR outcomes and to find any new factors/features apart from the findings of the SLR. An analogous method has been used by other software engineering experts and researchers [41, 42].

The third phase is the development of the proposed model. The outcomes of the SLR and empirical study will provide inputs to the development of the GSMRM. A similar approach has been used by other software engineering researchers[41]. The proposed GSMRM model will be validated through conduction of five case studies in software development multi-sourcing industry. A similar approach has been used by other researcher [43]. The current study discusses the first phase/stage of our research study i.e. Systematic Literature Review Protocol Development. In current paper we have only presented first phase of our ongoing research i.e. SLR protocol with its preliminary results.

4. SYSTEMATIC LITERATURE REVIEW PROTOCOL

The SLR protocol is a set of activities that have to be completed in order to response the research questions listed in the *Section 4.1*. Our review protocol is made of seven components as shown in Fig. 2. Each component is explained in the following sub-sections:

4.1 Research Questions

We have formulated the following Research Questions (RQs).

RQ 1: What are the success factors, as identified in the literature, to be addressed by multi-sourcing vendor organizations which can contribute towards the development of green and sustainable software?

RQ 2: What are the risk factors, as identified in the literature, to be avoided by multi-sourcing vendor organizations which can contribute towards the development of green and sustainable software?

RQ 3: What are the real-world practices, as identified in the literature, to be adopted by multi-sourcing vendor organizations which can contribute towards the development of green and sustainable software?

4.2 Building Search Term

We have used population, intervention and outcomes of relevance for designing search term from our research questions.

Population: Software Multi-sourcing Vendor Organization

Intervention: Success Factors, Risk Factors and Real-world practices

Outcomes of relevance: Green Software Multi-sourcing Readiness Model (GSMRM)

Experimental Design: SLR, Empirical studies, case studies, expert's opinions and theoretical studies.

4.3 Search strategy/policy

The following search strategy is used for the construction of search terms.

a) We will identify major search terms from population, intervention and outcome of the formulated research questions.

b) We will find the alternate (synonyms) for these major search terms.

c) Appropriate keywords will be considered from relevant papers.

d) Use 'AND' for the concatenation of major terms and 'OR' for concatenation of alternative spellings and synonyms.

Results for a) Green software, practices, success factors, risks factors, multi-sourcing.

Results for b)

Green software :("green computing" OR "green IT" OR "green software" OR "sustainable software")

Practices: ("practices" or "methods")

Success Factors: ("factors" OR "elements" OR "parameters")

Risks Factors: ("risks" OR "barriers" OR "problems")

Multi-sourcing: ("multi-sourcing" OR "multisourcing" OR "multi vendors")

Results for c)

("Green software", "sustainable software", "green computing", "green ICT", "green IT")

Results for d)

Track 1: ("Green software development" OR "green software" OR "sustainable software" OR "green IT" OR "green computing") AND ("practices" OR "methods") AND ("factors" OR "elements" OR "parameters") AND ("risks" OR "barriers" OR "problems") AND ("multi-sourcing" OR "multi vendors outsourcing")

Track 2: ("Green computing" OR "green IT" OR "green software" OR "sustainable software") AND ("practices" OR "methods") AND ("factors" OR "elements" OR "parameters") AND ("risks" OR "barriers" OR "problems")

Where Track1 represents search string designed specifically to retrieve literature regarding green software in software development multi-sourcing context, whereas Track 2 represents search string designed with the intent to retrieve literature regarding green software in general context. The results of search string mentioned in Track 1 were very poor and almost negligible as shown in Table 1. Therefore we decided, after thorough discussions with fellows of software engineering research group (SERG-UOM) at the university, to move towards the implementation of Track 2 as shown in Table 2.

The factors, to be identified through the SLR, will be validated through empirical studies in multi-sourcing software industry in order to know whether these findings are applicable specifically, or can be adopted, in software multi-sourcing environment. A similar approach has already been used by other researchers [44]. Further very limited numbers of empirical research studies have been conducted GSD context in general and software multi-sourcing in particular [45].

Table 1. Track 1 results.

Digital Libraries	Total publication Found
Science Direct	00
ACM	00
IEEE Xplore	00
Springer Link	02
Google Scholar	14

Table 2. Track 2 results.

Digital Libraries	Total publication Found
Science Direct	1494
ACM	696
IEEE Xplore	33
Springer Link	1869
Google Scholar	3390

4.4 Resources Searched

We have searched the following digital libraries using 4.3(d) as a search strings.

- Science Direct (http://www.sciencedirect.com/)
- ACM (http://dl.acm.org/)
- IEEE Xplore (http://ieeexplore.ieee.org/)
- Springer Link (http://link.springer.com/)
- Google Scholar (https://scholar.google.com.pk/)

4.5 Publication Selection

The publication selection criteria are used to select relevant literature from the execution of search string. We have defined the following inclusion/exclusion criteria in order to extract relevant research papers from available literature.

4.5.1 Inclusion Criteria

- Research papers that are relevant to our research questions.
- Research work that describe green software in multi sourcing environment/General context.
- The research papers/ articles /books/review paper is in English Language.
- The article/paper is available in full text.
- Research papers that describes success factors, risk factors and practices of green software development in multi-sourcing/General context.

4.5.2 Exclusion Criteria

- Article/papers/books etc. that do not fulfil inclusion criteria as mentioned above will be excluded

4.5.3 Primary Selection of Relevant Literature

Primary selection of relevant literature has been performed by reviewing the abstract, title and keywords of the papers. The purpose of primary selection is to eliminate those outcomes which have no relevance to our research questions. The primary selected research papers have been checked against the aforementioned publication selection criteria by reading full text of the selected papers.

4.6 Publication Quality Assessment

The main drive of quality valuation is to check and assess the quality of finally selected papers. The quality checklist contains the following questions:

a. Is the objective of the research is clearly defined?
b. Is the outcomes of the research is connected to the objective of the research?
c. Whether the term green software in multi sourced projects/General software development context is discussed clearly?
d. Is it clear how the factor/practice was identified?
e. Each of the above interrogations will be marked as 'YES', 'NO', or 'N.A'.

Apart from the above quality criteria we will also use the following quality criterion which is based on our own experience. We have used the same criteria in our previous research [46].

Criteria for A-quality papers: In this category we list those papers which fulfill the following criteria's:

- Paper published in impact factor Journal
- Having clear methodology
- Having sample size of the following condition
  a. Case Study ≥ 3
  b. Interviews ≥ 12
  c. Survey ≥ 50
  d. Literature Review ≥ 50

Criteria for B-quality papers: In this category we list those papers which fulfill the following criteria's:

- Paper published in well reputed conference
- Having clear methodology
- Having sample size of the following condition
  a. Case Study = 2
  b. Interviews: $\geq 5$and ≤ 11
  c. Survey: ≥ 30 and ≤ 49
  d. Literature Review: ≥ 30 and ≤ 49

Criteria for C-quality papers: In this category we list those papers which fulfill the following criteria's:

- Experienced reports/ articles, published in less reputed venues (Journal, Conference)
- Having clear methodology
- Having sample size of the following condition:

Table 3. SLR Protocol preliminary results.

S. No	Digital library	Search String Used	Date Constraint	Total Publications Found	Primary/ Initial Selection	Final Selection
1	Science Direct	Track 2	All Years	1494	22	09
2	ACM	Track 2	All Years	696	25	11
3	IEEE Xplore	Track 2	All Years	33	05	02
4	Springer Link	Track 2	All Years	1869	17	06
5	Google Scholar	Track 2	All Years	3390	22	13
6		**Total**		**7,482**	**91**	41
7		**Publication through Snowballing: 33** **Total Publications through SLR: 41** **Total Finally Selected Publications: (N=74)**				

Table 4. Data extraction form.

Data to be Extracted	Description/Values
Extraction Form S.No	
Paper ID	
Date Of Review	
Title Of The Paper	
Reference	
Year (Data Collection / Publication Year)	
Database/Digital Library/Online Resource	
Publication Quality (A/B/C/Other)	
Population (Sample/Target)	
Study Strategy (OLR, SLR, Survey, Interview, Case Study, Experiment, Experience Report etc)	
Company Type (National, Multi-National (MNC),NA)	
Company Size (Small, Medium, Large, NA)	
Company SPI Status (CMMI,ISO etc, NA)	
Country (Data collection / Author)	
Multi-Sourcing Context /General Context	
Success Factors	
Risks Factors	
Practices	

a. Case Study: 1
b. Interviews ≤ 5
c. Survey: ≥ 1 and ≤ 29
d. Literature Review: ≥ 1 and ≤ 29

4.7 Data Extraction and Synthesis Process

The practice of mining data from finally selected research papers/articles is called extraction of data. The fundamental goal of our SLR protocol is to mine data (success factors, risk factors and green software practices) which fulfills our framed research questions that is; RQ1, RQ2 and RQ3. We have successfully extracted data from 74 publications. The review was mainly undertaken by a single researcher (principal author), however secondary researcher (Research Supervisor) was

Table 5. List of factors for green and sustainable software development.

S. No.	Success Factor	N= 74	References
1	Green software design and efficient coding	57	[α 1][α 3][α 4][α 5][α 6][α 9][α 10][α 11][α 12][α 16][α 17][α 18][α 19][α 21][α 22][α 23][α 24][α 25][α 36][α 37][α 38][α 40][α 41][α 43][α 45][α 48][α 52][α 53][α 74][α 58][α 60][α 62][α 63][α 65][α 66][α 67][α 68][α 71][α 26][α 29][α 30][α 32][α 33][α 33][α 2][α 7][α 8][α 20][α 34][α 39][α 55][α 57][α 27][α 28][α 29][α 32]
2	Power-saving software strategies	55	[α 2][α 5][α 1][α 6][α 7][α 8][α 10][α 11][α 12][α 13][α 14][α 15][α 17][α 18][α 21][α 22][α 23][α 24][α 25][α 35][α 36][α 37][α 39][α 40][α 41][α 43][α 44][α 46][α 47][α 48][α 49][α 50][α 51][α 52][α 53][α 54][α 55][α 56][α 73][α 58][α 60][α 61][α 63][α 65][α 66][α 68][α 72][α 26][α 29][α 30][α 32][α 33]
3	Low carbon emission throughout the software development process	45	[α 2][α 5][α 1][α 6][α 8][α 9][α 10][α 11][α 13][α 15][α 17][α 18][α 22][α 25][α 36][α 37][α 40][α 43][α 44][α 46][α 48][α 52][α 54][α 55][α 57][α 58][α 59][α 60][α 63][α 64][α 65][α 66][α 67][α 68][α 70][α 71][α 72][α 26][α 28][α 29][α 30][α 32][α 33]
4	Efficient Resource Utilization	44	[α 2][α 3][α 4][α 6][α 8][α 9][α 10][α 11][α 12][α 14][α 21][α 22][α 23][α 24][α 25][α 34][α 35][α 37][α 38][α 40][α 41][α 42][α 43][α 44][α 47][α 50][α 51][α 52][α 53][α 54][α 55][α 73][α 58][α 60][α 61][α 62][α 63][α 64][α 65][α 68][α 29][α 30][α 32]
5	Paperless communication	41	[α 2][α 5][α 1][α 6][α 8][α 11][α 12][α 23][α 25][α 34][α 36][α 40][α 43][α 45][α 48][α 50][α 51][α 52][α 53][α 54][α 55][α 56][α 57][α 74][α 58][α 59][α 60][α 62][α 64][α 66][α 65][α 67][α 71][α 72][α 26][α 28][α 31][α 32][α 33]
6	Filtration of requirements through green evaluator	40	[α 1][α 3][α 4][α 5][α 6][α 8][α 9][α 10][α 11][α 12][α 16][α 19][α 20][α 21][α 22][α 24][α 34][α 36][α 38][α 41][α 42][α 43][α 46][α 53][α 58][α 59][α 61][α 63][α 64][α 65][α 67][α 71][α 26][α 28][α 29][α 30][α 32]
7	E-Waste Management	40	[α 1][α 5][α 6][α 20][α 22][α 25][α 43][α 48][α 52][α 58][α 60][α 62][α 64][α 65][α 67][α 71][α 29][α 32][α 21][α 23][α 25][α 35][α 37][α 40][α 43][α 44][α 46][α 47][α 48][α 49][α 50][α 52][α 57][α 58][α 63][α 65][α 67][α 30][α 33][α 1][α 2][α 3[α 5]
8	Green utilization of the software	20	[α 5][α 11][α 13][α 17][α 18][α 20][α 21][α 22][α 24][α 25][α 43][α 46][α 54][α 58][α 65][α 70][α

Contd.....

Table 5 (Contd.)

S. No.	Success Factor	N= 74	References
			71][α 26][α 29][α 32]
9	Sustainable maintenance of the software	19	[α 5][α 11][α 13][α 17][α 18][α 20][α 21][α 22][α 24][α 25][α 43][α 46][α 54][α 58][α 65][α 70][α 71][α 26][α 29][α 32]
10	Sustainable testing	17	[α 1][α 4][α 5][α 6][α 12][α 13][α 16][α 19][α 21][α 22][α 34][α 43][α 46][α 53][α 29][α 30][α 32]
11	Use of cloud for software distribution	16	[α 1][α 2][α 3][α 5][α 12][α 16][α 20][α 22][α 43][α 53][α 58][α 61][α 65][α 71][α 32]
12	Use of agile strategies	15	[α 3][α 6][α 16][α 21][α 24][α 34][α 43][α 53][α 59][α 27][α 28][α 38][α 48][α 13]
13	Software reusability	14	[α 5][α 1][α 6][α 7][α 16][α 18][α 34][α 38][α 42][α 53][α 61][α 63][α 68][α 33]
14	Green graphical user interface	14	[α 4][α 8][α 12][α 14][α 16][α 18][α 21][α 36][α 53][α 61][α 26][α 29][α 33]
15	Software flexibility and legacy systems support	14	[α 4][α 5][α 6][α 8][α 9][α 10][α 22][α 23][α 38][α 39][α 58][α 63][α 71][α 29]
16	Green project management	11	[α 23][α 34][α 35][α 43][α 53][α 56][α 57][α 73][α 64][α 71][α 33]
17	Provision of green infrastructure	10	[α 1][α 22][α 25][α 37][α 48][α 59][α 62][α 63][α 65][α 29]
18	Green application development environment	09	[α 6][α 19][α 21][α 25][α 36][α 39][α 61][α 68][α 32]
19	Efficient estimation strategies	03	[α 38][α 45][α 13]
20	Green economy	01	[α 69]

Table 6. List of finally selected research papers in the SLR.

Final ID	Paper Title	Tracing Number	Year	Database
α 1	Green software development model: an approach towards sustainable software development	Snowballing 1	2011	IEEE
α 2	Exploration of green computing	Snowballing 2	2013	CiteSeer
α 3	Green software engineering process: moving towards sustainable software product design.	Snowballing 3	2013	Google scholar
α 4	Towards software sustainability guidelines for Long-living Industrial Systems	Snowballing 4	2011	CiteSeer
α 5	Green RM: reference model for sustainable software development	Snowballing 5	2013	Google scholar
α 6	A green model for sustainable software engineering	Snowballing 6	2013	Google

Contd.....

Table 6 (Contd.)

Final ID	Paper Title	Tracing Number	Year	Database
				Scholar
α 7	Green software: greening what and how much?	Snowballing 7	2014	IEEE
α 8	Evaluation of software product quality attributes and environmental attributes using ANP decision framework	Snowballing 8	2014	Google scholar
α 9	Deploying and provisioning green software	Snowballing 9	2014	IEEE
α 10	Green software development and design for environmental sustainability	Snowballing 10	2013	Google scholar
α 11	Exploring initial challenges for green software engineering: summary of the first GREENS workshop.	Snowballing 11	2013	ACM
α 12	A review on green software development in a cloud environment regarding software development life cycle: (SDLC) perspective.	Snowballing 12	2013	Google scholar
α 13	Toward sustainable software engineering (nier track)	Snowballing 13	2011	Google Scholar
α 14	Towards a software product sustainability model	Snowballing 14	2013	Google scholar
α 15	A systematic literature review on green software metrics	Snowballing 15	2013	Google Scholar
α 16	Developing sustainable software solutions for Bioinformatics by the “Butterfly ” paradigm	Snowballing 16	2014	Google Scholar
α 17	Green web engineering-measurements and findings	Snowballing 17	2012	Google Scholar
α 18	Analysis and principles of green UI design for web portals	Snowballing 18	2014	Google scholar
α 19	Approach to improve energy efficiency of information systems	Snowballing 19	2014	Google Scholar
α 20	Enhancing software engineering processes towards sustainable software product design	Snowballing 20	2010	Google scholar
α 21	Special factors of development of green software Supporting eco sustainability	Snowballing 21	2013	Google scholar
α 22	Sustainable development, sustainable software, and sustainable software engineering	Snowballing 22	2011	IEEE
α 23	Measuring the sustainability performance of software projects	Snowballing 23	2010	IEEE
α 24	A systematic literature review on sustainability studies in software engineering	Snowballing 24	2014	Google scholar
α 25	Optimization of operating systems towards green computing	Snowballing 25	2011	Google scholar
α 26	Principles and holistic design of green web portal	Snowballing 26	2013	Google scholar
α 27	Green agile maturity model for global software development vendors	Snowballing 27	2014	Google scholar
α 28	Green and sustainable technologies in Software Engineering	Snowballing 28	2015	Google

Contd.....

Table 6 (Contd.)

Final ID	Paper Title	Tracing Number	Year	Database
				scholar
α 29	Greening an existing software system using the GPU	Snowballing 29	2013	Google scholar
α 30	Green computing and software defects in open source software: an empirical study.	Snowballing 30	2014	Google scholar
α 31	Understanding green software development: a conceptual framework	Snowballing 31	2015	IEEE
α 32	Lifecycle energy assessment of mobile applications.	Snowballing 32	2013	Google scholar
α 33	Sustainability issues in human computer interaction design	Snowballing 33	2014	Google scholar
α 34	Balancing software product investments	ACM1_page1	2009	ACM
α 35	Integrating sustainability into undergraduate computing education	ACM4_page2	2010	ACM
α 36	The software perspective for energy-efficient mobile applications development	ACM9_page4	2012	ACM
α 37	Fostering green it	ACM10_page5	2013	ACM
α 38	Sustainability evaluation of software architectures: a systematic review	ACM11_page6	2011	ACM
α 39	Want to green application software? – mind the target hardware	ACM17_page10	2013	ACM
α 40	An agenda for 'Green' information technology and Systems research	ACM21_page14	2011	ACM
α 41	Green software services: from requirements to business models	ACM25_page22	2013	ACM
α 42	Sustainable software development	ACM28_page25	2004	ACM
α 43	Green software engineering with agile methods	ACM29_page28	2013	ACM
α 44	The evolution of green ICT practice: UK higher education institutions case study	ACM31_page35	2011	ACM
α 45	Green IT Maturity: developing a framework based on practices and actions	GS8_page1	2013	Google scholar
α 46	Green wall: a methodology for sustainable development using green computing	GS10_page1	2014	Google scholar
α 47	Green IT adoption and sustainable value creation	GS14_page2	2014	Google scholar
α 48	Green computing technologies towards the development of ICT: a critical study	GS15_page2	2013	Google scholar
α 49	Green computing: practice of efficient and eco-friendly computing resources	GS17_page2	2009	Google scholar
α 50	Impact of green computing in it industry to make ecofriendly	GS19_page3	2014	Google

Contd.....

Table 6 (Contd.)

Final ID	Paper Title	Tracing Number	Year	Database
	environment			scholar
α 51	A research agenda on managerial intention to green it adoption: from norm activation perspective	GS20_page3	2014	Google scholar
α 52	Green Computing: an essential trend for secure future	GS22_page3	2013	Google scholar
α 53	Summary of the First Workshop on Sustainable Software for Science: Practice and Experiences	GS23_page3	2014	Google scholar
α 54	Green software and green software engineering-definitions, measurements, and quality aspects	GS27_page5	2013	Google scholar
α 55	Approaches to green computing to reduce global warming	GS28_page5	2013	Google scholar
α 56	A green IS taxonomy	GS30_page8	2014	Google scholar
α 57	A maturity model for green ICT: The case of the SURF green ICT maturity model,	GS31_page20	2014	Google scholar
α 58	The GREENSOFT model: a reference model for green and sustainable software and its engineering	SD1_page1	2011	Science Direct
α 59	A categorization of green practices used by Dutch data centers	SD3_page1	2013	Science Direct
α 60	Exploring the role of IT for environmental sustainability in China: An empirical analysis	SD7_page1	2013	Science Direct
α 61	Is software ‘‘green’’? Application development environments and energy efficiency in open source applications	SD8_page1	2012	Science Direct
α 62	Risk identification in green IT practice	SD9_page1	2013	Science Direct
α 63	Energy efficiency and low carbon enabler green IT framework for data centers considering green metrics	SD10_page1	2012	Science Direct
α 64	Designing IT systems according to environmental settings: A strategic analysis framework	SD12_page1	2011	Science Direct
α 65	Promoting green ICT in China: A framework based on innovation system approaches	SD17_page1	2012	Science Direct
α 66	Green WSUS	SD19_page2	2012	Science Direct
α 67	Green computing practices as a part of the way to the sustainable development.	SL1_page1	2013	Springer Link
α 68	Software level green computing for large scale systems	SL7_page1	2012	Springer Link
α 69	Green challenges to system software in data centers	SL17_page5	2011	Springer Link
α 70	Green software and green IT: an end users perspective	SL2_page1	2011	Springer Link
α 71	A Model and selected instances of green and sustainable	SL3_Page1	2010	Springer

Contd.....

Table 6 (Contd.)

Final ID	Paper Title	Tracing Number	Year	Database
	software			Link
α 72	Unpacking green IS: a review of the existing literature and directions for the future	SL10_page1	2012	Springer Link
α 73	Estimating the energy consumption of executing software processes	IEEE1_page1	2013	IEEE
α 74	Green IT-outsourcing assurance model	IEEE2_page1	2013	IEEE

approached for assistance in case of an issue about the data mining.

We have successfully performed inter-rater reliability test after data extraction. We have used analogous methodology in former research [47]. We have extracted data from finally selected 74 publications using data extraction form as shown in Table 4. In order to properly synthesize the extracted data, we will use data synthesis process by presenting the required data (success factors, risk factors and practices) in tabular format according to the formulated research questions i.e. RQ1, RQ2, RQ3.

5. PRELIMINARY RESULTS OF THE PROTOCOL

The protocol preliminary results are presented in Table 3. The protocol has retrieved 7,482 research papers/articles. We have selected 111 research papers as a primary/initial selection by reading title of the paper, and examining abstract. After removing duplicate papers we got 91 papers. By applying our pre-defined quality criteria, our publication sample size was squeezed to 74 research papers (Table 3). These include 41 papers from SLR protocol search strategy and 33 papers through Snowballing. Snowballing approach is normally used to complement the SLR search results by searching through the reference/bibliography of a specific research paper or the citations to the paper in order to identify additional/new research papers. We have used the guidelines [48] for conducting snowballing methodology. We have preliminary extracted a list of 20 factors; these factors are shown in Table 5, where Table 6 presents the details of finally selected papers. The identified factors support software development organizations to develop green and sustainable software. We have studied a number of studies for the design and development of the presented SLR protocol [49-56].

6. CONCLUSIONS

Software development multi-sourcing is an emerging global software engineering (GSE) paradigm for producing high quality software at minimum cost and time, by hiring more than one software vendors. Consequently, green software multi-sourcing has raised a novel research area in the context of GSE domain which aims to produce greener software in multi-sourced software development projects. Research in this area will assist multi-sourcing industry to deliver green software in general and to gauge multisourcing vendors' readiness for the development of greener software in particular. In this paper we have presented our SLR protocol to identify the success factors, risk factors and green practices from the available literature on green software development. During implementation of the protocol, while using our search string defined in Track 2 in various search libraries, we have identified a publication sample of (N=7482) research papers out of them (N=91) have been selected as the primary sample by reviewing through title and abstract of the papers. After reviewing through full text of the primary sample of publications, finally we selected 74 papers as our final sample. We have shown these search results and publications selection in Table 3.

We have completed the data extraction phase of the protocol, by extracting data from the finally selected papers on our pre-defined data extraction form as shown in the Table 4.

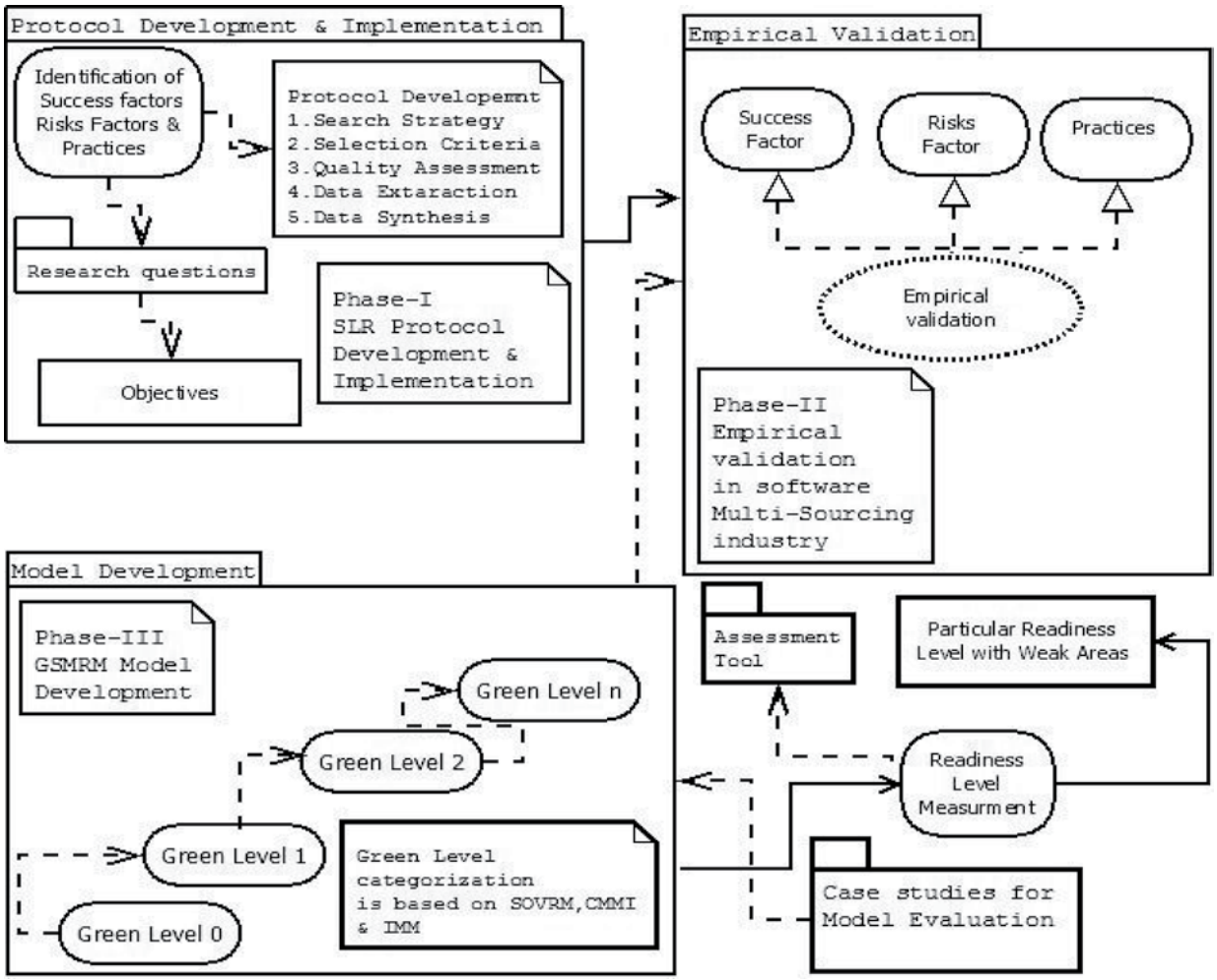

Fig. 1. Phases in our Proposed GSMRM model.

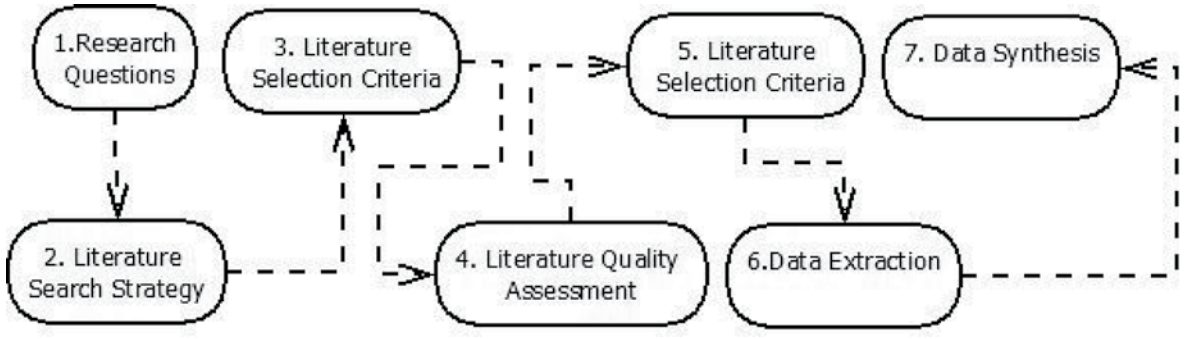

Fig. 2. Components of SLR protocol.

Currently we are in the process of SLR data synthesis in order to synthesize the extracted data. We have preliminary extracted a list of 20 factors (Table 5); Table 6 presents the details of finally selected papers. The identified factors support software development organizations to develop green and sustainable software. Further our ultimate goal is the development of Green Software Multi-Sourcing Readiness Model (GSMRM) as shown in Fig.1.

7. ACKNOWLEDGEMENTS

We are grateful to all fellows of the Software-Engineering-Research-Group (SERG-UOM) at University of Malakand, Pakistan for their review comments in general and for protocol validation in particular. We are also thankful to the anonymous reviewers of the 19th International Conference on Evaluation and Assessment in Software Engineering (EASE 2015) for their valuable comments.

8. REFERENCES

1. Cai, S., X. Chen & I. Bose. Exploring the role of IT for environmental sustainability in China: An empirical analysis. *International Journal of Production Economics* 146: 491-500 (2013).
2. Harmon R.R., & N. Auseklis. Sustainable IT services: Assessing the impact of green computing practices. In: *International Conference on Management of Engineering & Technology*, PICMET, Portland, p. 1707 - 1717 (2009).
3. Gu, Q., P. Lago., H. Muccin., & S. Potenza. A categorization of green practices used by Dutch data centers. *Procedia Computer Science* 19:770-776 (2013).
4. Harmon, R., H. Demirkan., N. Auseklis & M. Reinoso. From green computing to sustainable IT: Developing a sustainable service orientation. In: *System Sciences (HICSS), 43rd Hawaii International Conference*, p. 1-10 (2010).
5. Davidson, E., E. Vaast & P. Wang. The greening of IT: how discourse informs IT sustainability innovation. In; *IEEE 13th Conference on Commerce and Enterprise Computing (CEC*), p. 421-427 (2011).
6. Harris, J. *Green Computing and It Best Practices on Regulations and Industry Initiatives, Virtualization, Power Management, Materials Recycling and Telecommuting*. Emereo Pty Ltd, London, UK (2008).
7. Harbla, A., P. Dimri., D. Negi & Y. S. Chauhan. Green computing research challenges: a review. *International Journal of Advanced Research in Computer Science and Software Engineering* 3(10): 1-13 (2013).
8. Moshnyaga, V.G. An assessment of software lifecycle energy. In: *Power and Timing Modeling, Optimization and Simulation (PATMOS), 23rd International Workshop*, Karlsruhe, p. 112-119 (2013).
9. Sierszecki, S., K. Sierszecki., T. Mikkonen., M. Steffens., T. Fogdal., & J. Savolainen. Green software: greening what and how much?. *Software, IEEE* 31: 64-68 (2014).
10. Mahmoud, S.S. & I. Ahmad. A green model for sustainable software engineering. *International Journal of Software Engineering and Its Applications* 7(4): 55-74 (2013).
11. Erdelyi, K. Special factors of development of green software supporting eco sustainability.In: *Intelligent Systems and Informatics (SISY), IEEE 11th International Symposium,* Subotica, p. 337-340 (2013).
12. Appasami, G. & S. Joseph. Optimization of operating systems towards green computing. *International Journal of Combinatorial Optimization Problems and Informatics* 2(3):39-51 (2011).
13. Naumann, S., M. Dick., E. Kern & T. Johann. The GREENSOFT Model: a reference model for green and sustainable software and its engineering. *Sustainable Computing: Informatics and Systems* 1(4): 294-304 (2011).
14. Shenoy, S. S., & R. Eeratta. Green software development model: an approach towards sustainable software development. *India Conference (INDICON), India Annual IEEE,* p. 1-6 (2011).
15. Taina, J. Good, bad, and beautiful software-In search of green software quality factors. *CEPIS UPGRADE* 12(4): 22-27 (2011).
16. Mahaux, M. & C. Canon. Integrating the complexity of sustainability in requirements engineering. In: *First International Workshop on Requirements for Sustainable Systems.* Belgium, p. 31-33 (2012).
17. Mahaux, M., P. Heymans & G. Saval. Discovering sustainability requirements: an experience report. In: *Requirements Engineering: Foundation for Software Quality SE-3, Lecture Notes in Computer Science*, 6606 ed. Springer Berlin Heidelberg, p.19-33 (2011).
18. Winschiers, H. & B. Paterson. Sustainable software development. In: 2*004 Annual Research Conference of the South African Institute Of Computer Scientists and Information Technologists on IT Research in Developing Countries*. South Africa, p. 274-278 (2004).
19. Koçak, S.A., G.I. Alptek & A.B. Bener. Evaluation of software product quality attributes and environmental attributes using ANP decision framework. In: *Third International Workshop on Requirement Engineering for Sustainable Systems,* p. 37-44 (2014).
20. Bozzelli, P., Q. Gu & P. Lago. *A Systematic Literature Review on Green Software Metrics*. VU University, Amsterdam (2013).
21. Dick, M., J. Drangmeister, E. Kern & S. Naumann. Green software engineering with agile methods. In: *Green and Sustainable Software (GREENS), 2nd International* Workshop, Netherlands, p. 78-85 (2013).
22. Durdik, Z., B. Klatt., H. Koziolek., K. Krogmann., J. Stammel & R. Weiss. Sustainability guidelines

for long-living software systems. In: *28th IEEE International Conference Software Maintenance (ICSM)*, Trento, p. 517-526 (2012).
23. Sabharwal, M.R. Software power optimization analysis and optimization for energy-efficient software. In: *Proceedings of the 17th IEEE/ACM international symposium on Low-power electronics and design*, Fukuoka, p. 63-64 (2011).
24. Khan, R.U. & S. U. Khan. Green IT-outsourcing assurance model. In: *Global Software Engineering Workshops (ICGSEW), IEEE 8th International Conference*, Bari, p. 84-87 (2013).
25. Chauhan, N.S. & A. Saxena. A green software development life cycle for cloud computing. *IT Professional* 15(1): 28-34 (2013).
26. Johann, T., M. Dick., E. Kern & S. Naumann. Sustainable development, sustainable software, and sustainable software engineering: an integrated approach. In: *Humanities, Science & Engineering Research (SHUSER), International Symposium*, Kuala Lumpur, p. 34-39 (2011).
27. Dustdar, S., F. Li., H. Truong, S. Sehic, S. Nastic, S. Qanbar, M. Vögler & M. Claeßens. Green software services: from requirements to business models. In: *Green and Sustainable Software (GREENS), 2nd International Workshop* USA, p. 1-7 (2013).
28. Liangli, M., C. Yanshen., S. Yufei & W. Qingyi. Virtualization maturity reference model for green Software. In: *International Conference on Control Engineering and Communication Technology (ICCECT),* Liaoning, p. 573-576(2012).
29. Sahin, C., F. Cayci., J. Clause., F. Kiamilev., L. Pollock & K. Winbladh. Towards power reduction through improved software design. In: *Energytech IEEE*, Cleveland, p. 1-6 (2012).
30. Noureddine, A., A. bourdon., R. Rouvoy., & L. Seinturier. Runtime monitoring of software energy hotspots. In: *27th IEEE/ACM International Conference on Automated Software Engineering. Essen,* Germany, Essen, p. 160-169 (2012)
31. Khan, M.A., C. Hankendi., A. K. Coskun & M. C. Herbordt. Software optimization for performance, energy, and thermal distribution: Initial case studies. In: *Green Computing Conference and Workshops (IGCC),* Orlando, p. 1-6 (2011).
32. Kehal, H. Outsourcing and offshoring. In: *The 21st Century: A Socio-Economic Perspective*. Lgi Global (2006).
33. Salam, M. & S.U. Khan. Green software multi-sourcing readiness model (GSMRM) from vendor's perspective. *Science international (Lahore)* 26:1421-1424 (2014).
34. Turner, M., & S. *Charters. Protocol for a Systematic Literature Review of the Technology Acceptance Model and Its Predictive Capabilities.* Keele University, UK (2006).
35. Sarah, B., N. Baddoo., T. Hall., H. Robinson & H. Sharp. *Protocol for a Systematic Literature Review of Motivation in Software Engineering.* University of Hertfordshire UK and the Open University, UK, (2006).
36. Khan, S.U., M. Niazi & N. Ikram. *Systematic Literature Review Protocol for Software Outsourcing Relationships Trust (SORT)*. School of Computing & Maths, Keele University, UK (2009).
37. Kitchenham, B. *Procedures for Performing Systematic Reviews*. Keele University, UK (2004).
38. Zhang, H., M.A. Babar & P. Tell. Identifying relevant studies in software engineering. *Information and Software Technology* 53(6): 625-637 (2011).
39. Kitchenham, B. & S. Charters. *Guidelines for performing Systematic Literature Reviews in Software Engineering.* Keele University and Durham University Joint Report EBSE 2007-001 (2007).
40. Ilyas, M. & S.U. Khan. Software integration model for global software development. In: *5th International Multitopic Conference (INMIC),* Islamabad, p. 452-457 (2012).
41. Khan, S.U. & M.K. Niazi. *Systematic Literature Review Protocol for Software Outsourcing Vendors Readiness Model (SOVRM).* School of Computing and Maths, Keele University TR/08-0,1 UK (2008).
42. Ahmad, A. & S.J. Kolla. *Effective Distribution of Roles and Responsibilities in Global Software Development Teams*. School of Computing Blekinge Institute of Technology SE-371 79 Karlskrona Sweden, MSE-20XX-XX (2011).
43. Arslan, M, & M.A. Riaz. *A Roadmap for Usability and User Experience Measurement during early phases of Web Applications Development.* School of Computing, Blekinge Institute of Technology Box 520 SE - 372 25 Ronneby Sweden (2010).
44. Korkala, M. & F. Maurer. Waste identification as the means for improving communication in globally distributed agile software development. *Journal of Systems and Software* 95: 122-140 (2014).
45. Sriram, R. & S.K. Mathew. Global software development using agile methodologies: a review of literature. In: *Innovation and Technology (ICMIT), IEEE International Conference,* Sanur Bali, Indonesia, p. 389-393 (2012).
46. Ali, I. & S.U. Khan. Usability of agile methods in software development outsourcing: a systematic literature review protocol. *Science International (Lahore)* 26(5): 2023-2027 (2014).
47. Khan, S.U. & M.K. Niazi., & N. Ikram. Software development outsourcing relationships trust: a systematic literature review protocol. *Evaluation*

and Assessment in Software Engineering, EASE 2010, Keele University, UK 1-5 (2010).

48. Wohlin, C. Guidelines for snowballing in systematic literature studies and a replication in software engineering. In: *Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering. London, England*, p. 1-10 (2014).
49. Niazi, M., S. Mahmood, M. Alshayeb & A. Hroub. Empirical investigation of the challenges of the existing tools used in global software development projects. *IET Software* 9(5): 135-143 (2015).
50. Verner, J.M., O.P. Brereton, B.A. Kitchenham., M. Turner & M. Niazi. Risks and risk mitigation in global software development: A tertiary study. *Information and Software Technology* 56(1): 54-78 (2014).
51. Verner, J.M., O.P. Brereton, B.A. Kitchenham, M. Turner & M. Niazi. Systematic literature reviews in global software development: a tertiary study. In: *16th International Conference on Evaluation & Assessment in Software Engineering (EASE)* Spain, p. 2-11 (2012).
52. Niazi, M. Software Process Improvement: A road to success. In: *Product-Focused Software Process Improvement: Springer,* p. 395-401 (2006).
53. Niazi, M. An instrument for measuring the maturity of requirements engineering process. *The 6th International Conference on Product Focused Software P*rocess Improv*ement, LNCS, Oulu, Finland*, p. 13-16 (2005).
54. Khan, S.U., & M. Niazi & R. Ahmad. Empirical investigation of success factors for offshore software development outsourcing vendors. *Journal of IET Software* 6:1-15 (2012).
55. Niazi,M. & D. Wilson., D. Zowghi & B. Wong. A model for the implementation of software process improvement: An empirical study. In: *Product Focused Software Process Improvement:* Springer, p. 1-16 (2004).
56. Niazi, M., D. Wilson & D. Zowghi. A model for the implementation of software process improvement: a pilot study. In: *Third International Conference on Software Quality (QSIC03), IEEE Computer Society,* Dallas, USA, p. 196-203 (2003).