

Impact of Poor Requirement Engineering in Software Outsourcing: A Study on Software Developers' Experience

I. Perera

Indika Perera

Department of Computer Science and Engineering,
University of Moratuwa, Sri Lanka
E-mail: indikaperera@uom.lk

Abstract: The software Requirement Engineering (RE) is one of the most important and fundamental activities in the software life cycle. With the introduction of different software process paradigms, the Requirement Engineering appeared in different facets, yet remaining its significance without a doubt. The software development outsourcing is considered as a win-win situation for both developed and developing countries. High numbers of low paid, yet talented workforce in developing countries could be employed for software outsourcing projects with the demanding power of the outsourcer to decide the projects, their scope and priorities with the intention of profit maximization. This study was conducted to analyze the impact of poor Requirement Engineering in outsourced software projects from the developers' context (sample size $n = 57$). It was identified that the present outsourcing scenario has created to have frequent requirement changes, shrunk design and stretched development phases, and frequent deliverables, which have to be accommodated by the software developer with extra effort and commitment beyond the project norms. The results reveal important issues and open policy level discussions while questioning our insights on the outsourcing benefits as a whole.

Keywords: Requirement Engineering, Software Outsourcing, Project Management, Software Developer Productivity

1 Introduction

With the development of Internet and other communication media, the entire world is becoming a single village with the reach of each other at arms length. Many explain this scenario with the combination of other concepts as the Globalization. Some views Globalization with a positive look, where as others have strong critiques on its impact to individual nations and ultimately to the global economy. The development of the software and computing technologies has triggered the Globalization and world's operations to a large scale. In fact, the software outsourcing process gains its significance in the global economy with more and more projects being outsourced [1]. Software applications are complex and intangible products, which are difficult to manage. Hence, software lifecycle management becomes one of the key research areas in Software Engineering. Due to the nature of the software, software researchers and practitioners are focused to improve the software processes, which are used to develop software [2]. The underline assumption is that there is a direct correlation between the quality of the process and the quality of the developed software [3]. A software process can be considered as a set of tools, methods, and practices, where we use to produce a software product [4]. Requirement Engineering is one of the key process phases in the software development process; in most cases, essential enough to determine the success or failure of the project. Therefore, having an error free RE

activity and specification can be more significant for the project success than any other activity. Software requirement errors can cost up to 70-85% of additional cost of reworking in an average software project [5]. This shows the importance of managing the Requirement Engineering works productively especially in an outsourced project where the cost and profit concerns are high motives [18]. Furthermore, it is a known fact that many small to medium scale software companies practice flexible, lightweight, Agile like informal processes to suit their development; more or less for the sake of having a process practice to convince the outsourcers and win the projects [6]. However, in practice, these organizations tend to follow ad-hoc set of activities than any established process, which are camouflaged as the Agile practice. These informalities make the RE phase more complex, resulting the developers' programming work unnecessarily tedious and unproductive. This research has tried to study on the issues of poor RE practices with outsourced software projects and how it affects to junior level software developers' productivity and work life balance. The rest of the paper presents the research work and outcomes with the following organization. Section 2 will discuss the background literature related to this study. Section 3 discusses the research problem in brief. Thereafter, the section 4 is included with the experiment methodology and section 5, the analysis section explains the analysis done based on the collected results. The conclusion with the possible future directives is included thereafter along with the acknowledgements. The references will compile the paper.

2 Background

This section contains a comprehensive synopsis of the referred literature for this study in the areas of Software Requirement Engineering and Software Outsourcing. It is important to mention that, there have very few research works been done to study how the current software outsourcing practices affect to the software developers work.

2.1 Software Requirements Engineering

Software Requirement Engineering is a mature and one of the key life cycle activities in any software process paradigm. As Nuseibeh and Easterbrook explained, software systems' Requirements Engineering (RE) is the process of discovering that purpose, by identifying stakeholders and their needs, and documenting these in a form that is amenable to analysis, communication, and subsequent implementation [7]. More precisely, Zave clearly defines the Requirement Engineering as; "Requirements Engineering is the branch of software Engineering concerned with the real world goals for, functions of, and constraints on software systems. It is also concerned with the relationship of these factors to precise specifications of software behavior, and to their evolution over time and across software families." [8]. Sometimes, it is impossible to have complete and finalized set of requirements at the beginning of a project. This allows requirement changes to happen during the latter stages of the project and create conflicts with the software process been practiced. To overcome such scenarios Agile like flexible processes have been introduced and well established in the software industry. Agile process welcomes frequent requirement changes even at late stages of the project. With frequent deliverables, Agile process measures its progress through the norm of working software [9]. However, the most important point to remember, even if someone practices the Agile process is, that still he needs to perform a certain set of Requirement Engineering activities to ensure smooth and correct process flow [17]. Therefore, whatever the process practice used in the development, following proper Requirement Engineering activities is a must, even though the practices may have different facets.

2.2 Software Outsourcing

There are number of definitions for the concept outsourcing in general and specifically in the context of software development. Rajkumar and Mani explain offshore outsourcing as: "When the supplier of software development is from another country than the firm that decides to outsource information systems" [10]. The software development outsourcing has, in the recent past, taken up global dimensions in which companies from the developed countries sub contract software lifecycle function to the developing countries [11]. Importantly, outsourcing has become an appealing option to organizations operating within the global economy for a variety of reasons [12]. One of the major and foremost reasons to outsource software development is the lower cost and relative high profit margins for outsourcer [13]. Another advantage of software development outsourcing is the opportunity for the outsourcer to focus more on their core businesses and strategic activities [1]. Sometimes, this makes the outsourcing company to act solely as an intermediary organization, except for legal and contractual works. Furthermore, there are some issues with outsourcing to countries with vast heterogeneous socio-economic and cultural factors. Having local representatives may be a key option to overcome cultural and linguistic barriers in offshore outsourcing [14], [15]. However, this can be an extra cost item and will not help developers if the representatives are not technically sound. Nevertheless, the outsourcing attracts both developed and developing countries due to their benefits from different perspectives. Developing countries trying to secure more and more outsourcing projects, aiming the monetary gains, without conducting grass root level research on how it impacts to their workforce, industry and society at large.

3 Research Problem

The research problem which this study has focused mainly based on the software developers' experience in an outsourcing environment. There have been number of business and economic forums to attract outsourcing projects for developing countries, but not visible effort been done for assessing the impact of developing an outsourced industry on various contexts, except the monetary gains and employment opportunities. Since the majority of studies done to examine how to manage the outsource projects form the view point of the outsourcer, there is a significant vacuum of knowledge to understand the issues faced by the end developers due to the poor Requirement Engineering. As the outsourcing process consists with two parties mainly; the outsourcer and the service/solution provider, there are different issues for both parties depending upon their context. In many cases, the software developers are employed by the solution providers to engage in outsourced projects. This research primarily focuses on the problems that software developers encounter during their project activities, due to the poor software requirement practices of the projects. It is a known fact that many software outsourced projects get schedule overrun and extended developer time. One research found out that 60-80% of software projects get schedule or effort overrun and the average overrun varies from 30-40% of the initial project scope [16]. Why this level of scope variation during the project happens even with mature, experienced staff is a concern need to be examined with extensive studies. There can be other factors, which cause the over time work. Furthermore, the software developers' attitude towards such over work situation need to be assessed. This was a major objective of this study. Another questioning area is whether the process of passing requirements to the developer either directly from the main client or from the outsourcer, is effective and efficient enough to facilitate the development process. With the technological advancement and the studies to alleviate geographical barriers, have come up with various methods of communication for outsourcing [15]. In fact there are number of informal communication approaches used in the software outsourcing

and this study has analyzed the developer perception towards each approach and their relative significance to hinder the development activities. Identifying Key Success Factors for software process practices may be helpful for the improvement of current processes [19]. This study will also help to provide necessary guidance for possible improvements to the Requirement Engineering activities in outsourced software projects.

4 Research Methodology

The research methodology was based on a comprehensive survey conducted using the Sri Lankan outsourcing companies and their employees. Since the main objectives of the study and the research problem are generic to any outsourcing company and country as a whole the survey and its findings could easily be generalized with minor alterations. For this study, 57 software developers from 8 Small to Medium software developing enterprises were used. All these developers are engaged in outsourced projects for more than one year. To avoid any extreme cases in the study and to reduce skill variances, the developers who have industrial experience for less than 3 years were selected. The study was on voluntary basis and to strengthen the trust on the confidentiality of the participants' responses the study was conducted on individual basis without the intervention of the participants' employing organization. The study had two major phases for data collection. First all the participants were individually contacted through e-mail and telephone to explain the objective of the study and the confidentiality of their responses. The study was designed flexible enough to conduct by the participants as their own experiment. For the first phase, a worksheet was given to each participant to fill every day for 3 weeks. The worksheet was simple to fill and it has only four fields to be entered, based on the following measurement parameters.

N - Number of requirement changes/emerge for a given day

T_S - Start time of the work

T_F - Finish time of the work

T_M -Time spent on meetings/discussions to clarify/understand requirements for a given day.

These are basic parameters for this study and those were used to analyze the average daily commitment of the developer and as a validating factor for the second phase data. The second phase was based on a questionnaire which was distributed electronically through e-mail and the participants had to fill their one and send back. Strong assurances were given on the confidentiality of their responses in order to have more accurate information. The following section describes the collected data and their analysis

5 Analysis

During the first phase of the study, the above mentioned 4 parameters were recorded and the distribution of average time spent by each developer on meetings/ discussions to clarify requirements was calculated using the following model (1). μ_{M_j} is the mean time spent for the j^{th} developer, and T_{M_i} is the time spent on the i^{th} day up to n , i.e. the number of days he worked during the experiment.

$$\mu_{M_j} = \frac{\sum_{i=1}^n T_{M_i}}{n} \quad (1)$$

The obtained values are sampled for displaying purpose and shown in the Table 1.

Table 1: The distribution of average daily times spent on meetings/discussions for clarifying requirements during the development

Time Spent on Meetings for Requirements	Number
less than 0.5 hours	5
0.5 - 1 hours	9
1 - 1.5 hours	13
1.5 - 2 hours	14
2 - 2.5 hours	11
2.5 - 3 hours	3
3 - 3.5 hours	1
more than 3.5 hours	1

To obtain the population mean value μ_M of T_M , the following model (2) was used and m represents the number of participants in the sample.

$$\mu_M = \frac{\sum_{j=1}^m \frac{\sum_{i=1}^n T_{Mij}}{n}}{m} \tag{2}$$

The obtained μ_M value for the entire population is 1.571 hours. The value shows that on average a developer has to spend more than one and half hours of their development time, daily to clarify requirements that they are developing. The main problem with this is that these times are not included in the project schedules and the developers have to get these times from their own expense as the deliverable deadlines are fixed.

Similarly with the same meaning of the notations as above, the population mean value for N , i.e. μ_N was calculated with the following model in (3).

$$\mu_M = \frac{\sum_{j=1}^m \frac{\sum_{i=1}^n N_{ij}}{n}}{m} \tag{3}$$

The obtained value for μ_N was 2.84. This indicates that nearly three requirements alterations happen per day on average during the development phase. There are many reasons for such significant change to occur. The first fact is that everybody wants to have the working software so quickly and success of getting future work depends on the delivery of current work. Because of this rapid nature of project development, the outsourcer cuts down the time on Requirement Engineering, design, and testing phases of the life cycle. However, when the frequent deliverable is done the outsourcer frequently improves his expected functionalities and gives frequent alterations to the planed requirements. Moreover, due to the high demanding power, the outsourcer gets what he requests over multiple times of changing the same requirements.

Finally the work pattern of the participants were analyzed with the model described in (4).

$$\mu_{W_j} = \frac{\sum_{i=1}^n \{T_{F_i} - T_{S_i} - K\}}{n} \tag{4}$$

In this model T_{F_i} and T_{S_i} are the finish time and the start time of a developer on i^{th} day, respectively. K is an average constant to represent the summation of none working times during a given day. It was considered to be 1.5 hours based on preliminary studies. With that model the following sample of work patterns were derived for a given week. In fact there is a reason to analyze this for a week specially. All the developers have in their employment contract that they are supposed to work on weekdays i.e. Monday to Friday and they have the opportunity of holidays including Saturday, Sunday and any other declared holidays by the government. However, it was observed that most of the participants have worked in Saturdays, special holidays, and in some cases Sundays to meet their deadlines. The mean value (μ_W) for the population, was

Table 2: The distribution of average working hours per week

Average working hours per week	Number
35 - 40 hours	2
40 - 45 hours	7
45 - 50 hours	16
50 - 55 hours	21
55 - 60 hours	10
60 - 65 hours	1

calculated using the model described in equation (5); hence, $\mu_W = 51.76$ hours, which indicates an average developer spends nearly 12 hours of additional work than their contractual norm of 40 hours per week. That means they put effort on additional work load on average worth of 29.4% of their monthly salary. In fact when the results finding were informed to the people who participated, they were astonished, and one of their response is worth to mention here. *"... I can't believe this. This means up to now I have worked nearly one million rupees work just for free ... I should have thought of this before ..."* [Participant's response]

$$\mu_W = \frac{\sum_{j=1}^m \frac{\sum_{i=1}^n \{T_{F_i} - T_{S_i} - K\}}{n}}{m} \quad (5)$$

5.1 Questionnaire Analysis

The questionnaire was consisted with 16 questions. Some of the questions were simple Yes/No type and some have to be answered with further details. Out those 16 questions, 6 were used to condition the participant's mind to answer correctly, and also used to validate the accuracy of their responses for the main questions. For this analysis, only the relevant questions, i.e. the 10 main questions were used.

Q1. Do you get a sufficient Requirement Specification for the project?

For this question 41 participants said 'No', 12 said 'Yes' and 4 said Don't Know. Therefore as a percentage, 71.92% believes that they do not get a sufficient requirement specification for their development.

Q2. What are the most frequent methods you use to communication on Requirements? Please rank them

The Table 3 summarizes the responses for the Question 2. Only 13 developers are working with proper Requirements Specification in which only 5 have the opportunity to consider it as the

Table 3: The distribution of average working hours per week

Method of Communication	1 st	2 nd	3 rd	4 th	Total
Formatted Req. Spec. as E-mail attachment	5	6	2	0	13
E-mail messages	24	21	8	4	57
Instant Messaging (IM) / Chat	19	18	11	9	57
Voice Conference / Telephone	6	10	24	11	51
Video Conference	3	2	12	19	36
Face to Face communication	0	0	0	4	4
Travel to Client places for Discussions	0	0	0	1	1

main form of requirement communication. Lack of proper Requirement Specification makes it enormously vulnerable to requirement alternations and scope creep, during the later stages of the project. Also only 1 participant has the opportunity to go to client premises and experience their expectations where as only four have the opportunity to meet the clients for discussions. Email messages considered as the main form of communication followed by instant messaging or chat. Also everybody (all 57) use E-mail and IM/chatting with different significance. Voice conferencing and Video conferencing are used as secondary and tertiary forms of communication when extra clarification is needed on requirements, in generally.

Q3. What are the reasons you think for requirement issues in your projects?

Table 4: The distribution of average working hours per week

Reason	Number
Poor Technical Knowledge at Outsourcer	37
Optimistic Scheduling	29
Scope Creep and Stretching Requirements	42
Poor Communication	17
Informal Ways of Practice	31
Management Issues	24
Practice of "Do What They Say"	44

The Table 4 indicates developers' responses on what they believe as the causes of requirement issues. There were multiple reasons given and most of the participants have concluded that the current practices in the outsourced industry are mainly responsible for these RE issues. As mentioned in the Background section, the Agile practices has significant impact on this issue as when it is practiced without a proper knowledge on how to use, the Agile process becomes a worse practice and creates a chaotic environment. According to other findings of this study it shows that developers have to accommodate such chaos within their activities and meet the deadlines and outcomes as expected. However, these results open a huge necessity for a discussion to improve the outsourcing industry in the developing countries.

Q4. Do you experience spending more time on assigned tasks than they were scheduled? If yes what is the reason for that?

As the Figure 1 shows, a significant impact to schedule overrun is due to the requirement issues. The other significant factors are technical difficulties, which may be due to lack of training and development opportunities for developers, and the Management issues specially having optimistic schedules to delight the outsourcer. The most important fact is that all of the participants ex-

perience to spend more time than they were scheduled, making them to work more time than the employment norms, without any compensation for the extra effort.

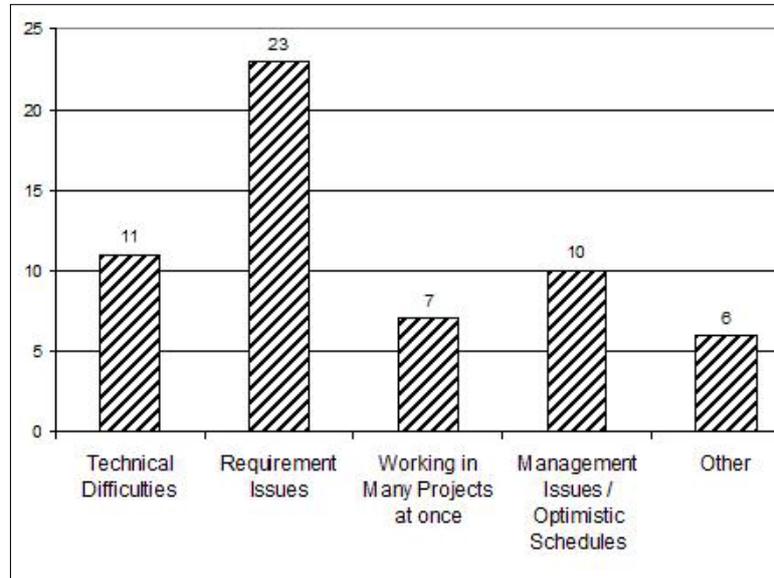


Figure 1: Reasons make the schedule overruns

Q5. Do you experience difficulties for understanding requirements from foreigners? Comment on this

Out of the 57 participants, 49 has responded and 37 had issues with verbal communication due to the accent of the foreigners. 42 of the participants complained about email and instant messages with insufficient information to take a decision; As a result, they have been spending more time and effort to clarify requirements. More importantly, the natural language based e-mails or chat messages, instead of a proper requirement specification make it further difficult to understand what the client is actually expecting.

Q6. If you are provided with complete and final requirement specification at the beginning of development, will it helps to your work?

23 people said it will help them lot and 9 were not sure about whether it helps them or not. 25 have indicated that it will not help as they will never getting a finalized requirement specification at any time of the project. This shows how much significant the issue of ever changing and poor Requirement Engineering process in the outsourced industry.

Q7. Are you the direct person to get requirements from client? If not how many to be passed for requirements to reach you?

8 participants have direct access with their clients to get requirements, 17 participants get through one person, 29 participants get through two persons, and 3 participants gets through three persons. The most obvious scenario was to have one person (non-technical) at the out-source end and one person (technical) at the development end to form 2 person layer to process requirements. Due to the non-technical managerial person at the client end, the requirements process deters its effectiveness, making developers to work more on requirements fine tuning and error fixing during the development phase.

Q8. How good are you at work and life balancing?

The idea of this question is to assess the health and social impact from the outsourcing work to the developers. There was a 5 level likert scale given to rank them considering their extra activities, exercise time, recreational activities and good life style practices. Results are shown in figure 2. 45.61% of the participants are not pay attention as required, for their social, health and other extra activities. When they spend more time on their work they get less time to attend those activities. The impact of this may not visible for short term but there are huge social, economical and health consequences in the long run which may not be easily compensated by their wee bit of extra salary they get compared to other disciplines. The good point is still the majority are happy about their work life balance, but there should be a proper mechanism to improve the situation.

Q9. Are you properly compensated for your effort?

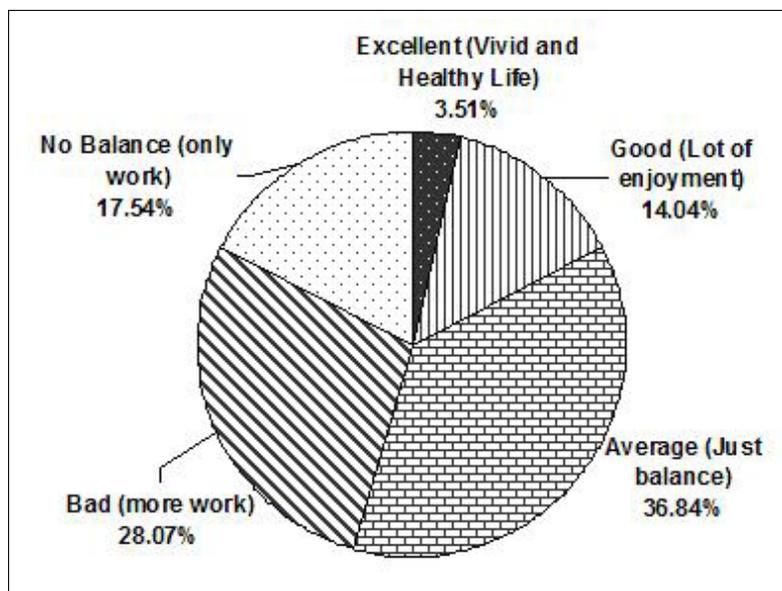


Figure 2: Work - Life balance of the developers

For this question the 21 participants (36.84%) said Yes and 26 participants (45.61%) have said No. 3 people said they are not sure about it. 7 participants did not respond to this question. In fact, out of the participants who responded, more than 50% is unhappy about their compensation and believe that they are under paid and over utilized.

Q10. What is the impact from time zone difference for your work times?

This was an open ended question where developers have to express their feeling about the impact from time zone difference. Idea of this question was to assess whether their work patterns have been affected from that. In fact, many have responded in similar fashion indicating there is a significant impact from the time zone difference. Most of the outsourcing projects are from United States of America and European Union member countries, where there is a significant time difference with the Sri Lankan time, around 11-13 hours and 4-6 hours, respectively. What the participants have indicated was that when the outsourcer wants to change a requirement or impose a new one, the developers have to communicate with them which will mostly the evening or late night in Sri Lankan Local time. In the following day they have to implement those from the morning. Therefore, invariably they have to spend at least 11-12 hours per day despite what

was agreed in the contract as 8 hours. And these don't count for any overtime payment or additional benefit.

The above individual analyses of questions provide a clear view on how the poor Requirement Engineering practices affect to software developers in outsourced projects. The questions were focused into different aspects of significance relevant to the research problem. Even though the fact that, responds for these questions covers a larger spectrum of issues related to the problem, there should be further researches with more focus to these significant areas identified in this analysis.

5.2 Study Limitations

In this study, there were some research limitations which are worth to mention. Since this study was involved with human activities, this had the experimental limitation of different skill levels between the participants. However this was general scenario of the software industry and has no significant impact for this study's outcome.

The second and the most crucial limitation with this study was the difference between the requirement changes. Some requirements were very simple and some were not. Though it was really hard to eliminate this, in the general situation every project has both difficult requirement changes and simple changes. Therefore, in the large population conditions ($n > 30$), this behaves with normal distribution and nullifies the impact as the standard error is 0.

Another limitation with the study was the truncation errors of the collected data. Literally, what happened was, the developers were confident on expressing their values with integer figures of hours or in minutes over the decimal or fractional values. For an example they might have said actual time as 7.15 but the time may be 7.11 However, since there is no comparison done based on their given values, the impact for the study outcome was negligible. Furthermore, these errors are also normally distributed with the standard error 0 in the large population samples.

6 Conclusion

Despite the outcome of this research, there are some possible future studies relevant to this research, which can be considered as further extensions. This study mainly focused on the Requirement Engineering issues in the development phase of the outsourced software project life cycle. There are other important life cycle phases in the outsourced projects such as System Testing, System Design, System Implementation, etc. Then again, there can be similar future studies to examine the social, economic and health impact from malpractices of software outsource industry in the context to employees, countries, and to industry as in whole. Furthermore, impact on software outsourcing due to different cultural characteristics, social and demographic parameters, and geographical and ethnic factors can be another fruitful research area.

Domain specific examine of the impact of poor Requirement Engineering on developers is another possible study as a further extension to this study, which will indeed provide an in depth insight for the prevailing issues. Due to the limited resources this study was conducted in a limited environment. Though the results prove impressive observations, it is expected, and encouraged the other scholars to conduct further researches based on the findings of this study to formulate a global knowledgebase on outsourcing and impact to its stakeholders due to various parameters. For that level of generalization, essentially there should more similar researches have to be conducted within other major countries within the global software outsourcing industry.

This study was initiated to examine an unaddressed issue in the outsourcing industry. Though the study does not cover each and every issue experienced by the developers, it provides sufficient

results and analyses to understand the gravity of the issue faced by developers engaged in outsourced industry. It is high time for policy makers to do further research on the identified issues and formulate standardization mechanisms to strengthen outsource industry in many developing countries while facilitating the developer community. In fact, the new computational paradigms such as Cloud Computing can cause significant impact towards reforming how outsourcing works at present. It is possible to have influential impact on present software outsourcing countries, when emerging solutions offering outsourcing both hardware and software [20]. It is therefore, a must to investigate possible improvements with new technologies to enhance the outsourcing industry, both individual country level and global level.

With all this regard it is fair to believe that, this study will create a significant paradigm shift towards rethink of the outsourced software development process. In conclusion, it is evident that this research is one of the significant achievements in the present Software Engineering field. This study's outcome will direct the policy implications to benefits the stakeholders of the software outsource industry, thus assist to improve the social, economical and health levels of software developers while avoiding crisis situation within the industry, in the long run.

Acknowledgement

Author would like to thank the people who helped for this study in various forms, especially, the developers who participated in this study for their effort and contribution to make this a success.

Bibliography

- [1] R. E. Ahmed, Software Maintenance Outsourcing: Issues and Strategies. *Computers and Electrical Engineering*, Vol. 32(6), pp. 449-453, 2006.
- [2] G.I.U.S. Perera, M.S.D. Fernando, Enhanced Agile Software Development - Hybrid Paradigm with LEAN Practice, in *Proc. of IEEE 2nd ICIIS conference* , Peradeniya, pp.239-244, 2007
- [3] A. Fuggetta, Software process: a roadmap, in *Proc. of the conference on The future of Software Engineering, ICSE*, Limerick, p.25-34, 2000
- [4] W.S. Humphrey, *Managing the Software Process*, SEI, Pearson Education, India, pp.03, 2006
- [5] K.E. Wiegers, *Software Requirements*, 2nd Ed. Redmond, Wash, Microsoft Press, 2003
- [6] G.I.U.S. Perera, M.S.D. Fernando, Bridging the gap - Business and information systems: A roadmap, in *Proc. of 4th ICBM conference*, pp. 334-343, 2007
- [7] B. Nuseibeh, S. Easterbrook, "Requirement Engineering: A Roadmap", *Proc. of the conference on The future of Software Engineering, 22nd ICSE*, Limerick, p. 35-46, 2000
- [8] P. Zave, Classification of Research Efforts in Requirements Engineering. *ACM Computing Surveys*, Vol. 29(4), pp. 315-321, 1997
- [9] K. Beck, *et. al.*, Manifesto for agile software development, 2001, available at <http://agilemanifesto.org/>, [accessed on 07th December 2008]
- [10] T.M. Rajkumar, R.V.S. Mani, Offshore Software Development. The View from Indian Suppliers, *Information Systems Management*, pp. 63-73, 2001

-
- [11] E. Carmel, Taxonomy of New Software Exporting Nations, *Electronic Journal of Information Systems in Developing Countries*, Vol. 13(2), p. 1-6, 2003
- [12] A. Yalaho, Plugging Into Offshore Outsourcing Of Software Development: A Multiple Case Study, *Issues in Information Systems*, Vol. 8(2), pp. 499-515, 2007
- [13] N. Levina, J.W. Ross, From the Vendor's Perspective: Exploring the Value Proposition in Information Technology Outsourcing, *MIS Quarterly*, Vol. 27(3), pp. 331-365, 2003
- [14] C. T. Coward, Looking Beyond India: Factors that Shape the Global Outsourcing Decisions of Small and Medium Sized Companies in America, *Electronic Journal of Information Systems in Developing Countries*, Vol. 13(11), pp. 1-12, 2003
- [15] E. Carmel, R. Agarwal, Tactical Approaches for Alleviating Distance in Global Software Development, *IEEE Software*, Vol. 18(2), pp. 22-29, 2001
- [16] K. Molíkken, M. Jørgensen, A Review of Surveys on Software Effort Estimation. *In Proc. of the International Symposium on Empirical Software Engineering*, pp. 223-231, 2003
- [17] I. Perera, Impact of using agile practice for student software projects in computer science education, *International Journal of Education and Development using Information and Communication Technology*, Vol. 5(3), Online [<http://ijedict.dec.uwi.edu/viewarticle.php?id=755>], 2009
- [18] G.I.U.S. Perera, Fernando M.S.D., Rapid Decision Making for Post Architectural Changes in Agile Development - A Guide to Reduce Uncertainty, *International Journal of Information Technology and Knowledge Management*, Vol. 2(2), In Press, 2009
- [19] G.I.U.S. Perera, "Key Success Factors for e-Learning Acceptability: A Case Based Analysis on Blended Learning End-User Experience", *In Proc. of IEEE International Advance Computing Conference, IACC'09*, pp. 2379-2384, 2009
- [20] I. Perera, Reshaping the Computation with Clouds: an Analysis on Opportunities and Issues of Cloud Computing, *Journal of Advances in Computational Sciences and Technology*, 2(3), pp.305-324, 2009