European Journal of Scientific Research
ISSN 1450-216X Vol.46 No.2 (2010), pp.221-232

http://www.eurojournals.com/ejsr.htm

The Impact of Global Software Development Factors on Effort Estimation Methods

Mohammad Muhairat
Department of Software Engineering, Al-Zaytoonah University of Jordan, Amman, Jordan
E-mail: drmohairat@alzaytoonah.edu.jo

Saleh Aldaajeh
Department of Software Engineering, Al-Zaytoonah University of Jordan, Amman, Jordan
E-mail: saleh.aldajah@yahoo.ca

Rafa E. Al-Qutaish
Department of Management Information Systems
Al-Ain University of Science & Technology - Abu Dhabi Campus, Abu Dhabi, UAE
E-mail: rafa@ieee.org

Abstract

Outsourcing software development work activities has brought many benefits to software development projects, such as, reduced development cost and time. Managing the application of this strategy is a key characteristic in its own or in its implications. Accurate effort estimation is crucial to software development projects success, especially in globally distributed projects. Therefore, it is necessary to identify and to investigate the underlying factors which influence the accuracy of effort estimation methods. In this paper, we will investigate the COCOMO II, SLIM and ISBSG effort estimation methods. Furthermore, the ISBSG method supports the experts' judgment estimation for effort as a candidate effort estimation method representing the expert judgment with accuracy in estimating the required amount of effort to accomplish a given project within the context of globally distributed projects.

Keywords: COCOMO II, ISBSG, SLIM, Outsourcing, Effort Estimation Methods

1. Introduction

Most of today's software development organizations seek to save time, reduce cost, and increase quality of their software products. Therefore, they invest in developing parts or the entire software products by contracting their work with a third party such as a team, partner, or an organization (Smite, 2006).

Globally distributed environment pervade today's software development industry. This strategy works by transmitting the common way of developing a software product (in-house) to a software life-cycle activities that are distributed among members who are separated by some boundary such as: contextual, organizational, cultural, temporal, geographical, or political (Betz and Mäkiö, 2007). Although the strategy of Globalization encloses many benefits which supports the development of software product in a cost effective way, this strategy faces many challenges which may hinder the success of globally distributed software development projects (Ågerfalk *et al.*, 2008; Carmel, 1999; Conchúir *et al.*, 2009; Feeny *et al.*, 2006; Šmite and Borzovs, 2008). As Kile *et al.* (2005) denoted in their study - which observing the rate of projects' success in globally distributed environment - that 60% of these projects were failed to deliver within time, budget, and desired quality. Thus, managing the globally distributed environment is a key characteristic, in its own, or in its implications. However, in order to successfully plan software development projects' activities, it is important to sustain a high level of accuracy to effort estimation methods (Shepperd *et al.*, 1996).

There are several effort estimation methods which can be used to estimate the required amount of effort to successfully deliver a software product such as: COnstructive COst MOdel (COCOMO), Software LIfe-cycle Model (SLIM), Experts' judgment, etc (Boehm, 1981; Panlilio-Yap, 1992; Shepperd *et al.*, 1996). The aforementioned methods are focused on embracing the different aspects which may influence the forecasting process of the required effort for delivering the software products (Boehm and Valerdi, 2008).

Experts' judgment is one of the methods by which assessors conduct their effort estimation via using their expertise and their logical reasoning to estimate the required amount of effort needed to develop a software product. The accuracy of this method is mainly depends on the skills, knowledge, and experience of the assessors to estimate the required among of effort to complete a given project. Unfortunately, most of the statistical effort estimation methods were designed and developed at the time when Globalization is still a new trend and is poorly explored (Smite, 2007). Due to the aforementioned challenges associated with this strategy, effort estimation methods are lacking from accuracy (Conchúir *et al.*, 2009). Therefore, the environment in which the software product is being developed must be taken into account as one of the important factors to software development projects' success (Conchúir *et al.*, 2009). Furthermore, it is required to elevate the level of accuracy for effort estimation methods used in this context (Ågerfalk *et al.*, 2005).

This paper investigates the influence of the different factors which affect the effort estimation methods accuracy in the context of globally distributed software development projects. Furthermore, it provides recommendations on the suitability of effort estimation methods based on the treated factors.

However, this paper is organized as follows: Section 2 further describes the scope of this research. Section 3 presents the research aims and objectives, research questions and systematically explains the research methodology framework for conducting the results. Section 4 and 5 shows and elaborates on the research outcomes. Section 6 illustrates the authors' recommendations for improving effort estimation process in the context of globally distributed environment. Finally, the paper is concluded in section 7.

2. Background and Motivation

This paper aims at exploring the accuracy of effort estimation methods in the context of globally distributed environment. Since globally distributed environment encloses many challenges to software development projects' success, section 2.1 briefly discusses the associated challenges of the application of this strategy to projects' success. Effort estimation impersonates a focal role in determining the success of projects' planning. Section 2.2 provides a brief discussion on the most conventionally effort estimation methods used in software industry. This research is based on empirical data extracted from different software projects where their life-cycle activities are globally distributed. Section 2.3 further describes the illustrated case studies.

2.1. Globally Distributed Environment's Influence, Factors, and Challenges

There are many peculiarities such as geographical diversity, temporal diversity, cultural diversity, linguistic and legislative diversity, etc, which distinguish this environment from the typical in-house development environment. Nevertheless, this environment has a great influence on software development projects' success. For instance, the temporal diversity may result in poor communication, and therefore, lower level of social interaction and richness of the information exchanged.

Globally distributed software development projects' success is challenged from different aspects. These challenges influences the software project at different levels, especially on the individuals' efficiency and the time consumed on developing a software product (Kormeren and Parvianen, 2007). According to Kormeren and Parvianen (2007), globally distributed team members' productivity decreases up to 50% compared to the level of co-located team members' productivity. Furthermore, in most cases, the delivery of software products developed in a globally distributed environment takes twice and a half more time than the software products developed in a co-located environment (in-house) (Hersleb and Mockus, 2005).

In most cases, the factors encountered in the globally distributed environment are investigated from two aspects: social and professional aspects (Bartelt *et al.*, 2009). For example, communication methods between distributed teams are investigated from social interaction and information exchanged richness. Additionally, communication methods, tools, and techniques are investigated on the bases of their professional aspects as well such as quality, delay in transmitting, etc, (Bartelt *et al.*, 2009). Nevertheless, globally distributed software development projects' success is never isolated to a certain factor or drive. There are several risks associated with this environment that is needed to be taken into consideration in order to plan, execute, and deliver projects' outcomes within time, budget, and the desired level of quality.

2.2. Effort Estimation

Developing software products in a cost effective way is the overwhelming objective for many organizations. Furthermore, the accurate estimation of the required amount of effort for projects completion is an ultimate goal. Many research studies indicated that projects without realistic planning and accurate estimation are often exceed their allocated budget and the proposed completion time (Boehm *et al.*, 2000; Nguyen *et al.*, 2008; Wittig, 1997).

Effort estimation methods can be roughly categorized into two main categories: Mathematically-based which relies on mathematical formulas for constructing and representing the required amount of effort, and experience-based which depends mainly on the experience for supplying the needed information to performing the effort estimation process. The mathematically-based effort estimation methods COCOMO II and SLIM are the most conventionally used to estimate the required amount of effort for developing a software product (Boehm, 1981; Kemere, 2008).

Experienced-based effort estimation methods can be support by history of completed projects. For example, The International Software Benchmarking Standards Group (ISBSG) provides information for more than four thousand completed software development projects. The ISBSG contains a large repository that helps in performing analyses, benchmarking, and comparisons of different trends in software projects (ISBSG, 2009). The underlying benefits of the ISBSG can be illustrated as a simulation approach which can be used to help the assessors to use facts to consolidate their assessment for the required time to complete a given software development project (Boehm *et al.*, 2000). Furthermore, ISBSG provides three different anticipation values representing the minimum amount of time, estimated time the project is more likely to complete, and the maximum time a given project may consume to finish.

2.2.1. COCOMO II

COCOMO method is first published in Software Engineering Economics book by Boehm (1981). This method is widely used for estimating cost and schedule for projects.

COCOMO II structure for estimating necessary effort and duration of projects is well described. This method mainly uses project's size. For example source Lines Of Codes (SLOC) or Function Points (FP) (Boehm *et al.*, 2000). Projects' cost is derived directly from Person Month (PM) effort. The (PM) represents the number of hours that a person spend to complete a given task presented in a calendar month. COCOMO II deals with variety of factors that influence projects' effort estimation. It has 17 cost drivers (for post architecture model) and 5 scale factors (Boehm *et al.*, 2000). There are three sub models for COCOMO II: Application Composition Model, Post Architecture Model, and Early Design Model. COCOMO II includes scale factors in order to steer the effort estimation team to make better approximation based on the influencing factors. These factors are related to organizational and team characteristics. Each scale factor has values from range of very low to extra high rating level. The weight of scaling factors could diver according to organizations and projects. The followings are the equations which COCOMO II proposed to estimate the required effort:

$$PM = A \times Size^{E} \times \prod_{i=1}^{17} EM^{i}$$

Where:

- $A = 2.94$ (for COCOMO II), Size is estimated by Kilo Source Lines Of Code (*KSLOC*) measure or unit, Cost drivers can be found in (Boehm *et al.*, 2000).
- $E = B + 0.01 \times \sum_{i=1}^{5} Scale\ Factor$.
- *EM* represents the Effort Multiplier, $B = 0.91$ for COCOMO II (Boehm *et al.*, 2000).

$$Duration = C \times PM^{D+0.2\times(E-B)}$$

Where:

- C = 3.67, D = 0.28, and B = 0.91.
- PM_{NS} is effort in PM excluding the Required Development Schedule (SCED) cost driver, and it is defined as the following:

$$PM_{NS} = A \times SizeB \times \prod_{i=1}^{16} EM^{i}$$

2.2.2. SLIM

SLIM (Kemere, 2008) is an algorithmic method that is used to estimate effort and schedule for projects. The underlying reason for developing SLIM is to measure the overall size of a project based on its estimated SLOC. This method was modified for effort estimation using Rayleigh curve model (Kemere, 2008).

The SLIM tool is the product of SLIM (for the proprietary of Putnam's model) which is a metrics-based estimation tool, developed by Quantitative Software Management (QSM), using validated data of over 2600 projects. These projects were classified into nine different application categories. This tool helps the management to estimate the effort and time required to build medium and large software projects. Most importantly, this tool can be customized according to a specific organization (Panlilio-Yap, 1992). The following equation is used to allocate the Productivity Parameter (*PP* which is used to calculate the required effort for a given project represented in a man-years unit or measure:

$$PP = \frac{Size_{SLOC}}{(E_{Man,Year}/B^{1.13}) \times Duration\ (Y)^{4/3}}$$

The second equation is used to calculate effort, using the value of *PP* from the above equation.

$$E_{Man,Year} = \left(\frac{Size_{SLOC}}{PP \times (Duration_{Years})^{3/4}} \right)^{3}$$

Where, $E_{Man,Years}$ represents the required amount of effort in order to complete a given task in a man-year unit or measure.

2.3. Case Studies

This research is built on the empirical data illustrated from three different projects of three different companies. These projects are focused on developing software products of different types. Furthermore, the software life-cycle activities were performed in a globally distributed environment. These projects failed to deliver the developed software products within the proposed time and budget. Most importantly, the underlying reason for these projects' failure is due to the under estimated amount of effort required to successfully deliver the software product. However, table 1 depicts these projects' and organizations' characteristics. Moreover, the three projects' details are described in 2.3.1, 2.3.2, and 2.3.3 respectively.

Table 1: Projects A, B, and C Details

Criteria	Project A	Project B	Project C
Globally Distributed	✓	✓	✓
Stakeholders	UK and PAK	UK and PAK	USA and PAK
Certification	ISO 9001: 2008, ISO 2000	×	ISO 9001: 2008
Project Settings	Offshore	Offshore	Offshore
People Involved	7	6	8
Effort Estimation Method Used	Experts' Judgment	Experts' Judgment	Experts' Judgment
Estimated Duration in calendar months	3	4	3
Delay in calendar months	1	1.5	1
Actual Duration to Delivery	4 months	5.5 months	4 months
Actual Effort Spent (Person-Month)	14.36	17.15	19.53

The strategy of Globalization has manifested different factors which influenced the accuracy of effort estimation methods adopted in these projects. However, these factors are illustrated in table 2 for project A, B, and C.

Table 2: Factors Influencing Projects' Effort Estimation Methods Accuracy

Factor	Project A	Project B	Project C
Different Time Zone	✓	✓	✓
Delay in Response	✓	✓	✓
Unavailability of Concerned Personal	✓	✓	×
Trust	✓	×	×
Clients Unawareness	✓	✓	✓
Shared Resources	×	✓	✓
Unrealistic Milestones	✓	×	✓
Communication	✓	✓	✓
Organization or Team Structure	×	✓	✓
Work Pressure	✓	✓	✓

2.3.1. Project A

The mission of this project is to develop a Web-based system. The software system offers a visual representation to evaluate changes in workforces from different perspectives. However, this project setting is to off shore project's tasks between the organization's teams. The project' lifecycle activities are distributed among two teams from the United Kingdom 'Headquarter Office' and Pakistan. The total number of employee involved in this project is seven. The team from the United Kingdom was responsible for the requirement engineering, and deployment stages. Nevertheless, the designing, coding, technical writing and testing stages were performed by the team from the Pakistan office. The effort estimation method adopted in this project is based on experts' judgment. The completion time is estimated by three calendar moths. The project was delivered in four calendar months, causing an extra month delay.

2.3.2. Project B

The second project objectives are focused on developing a Computer-based accountant software system. This software system mission is to computerize the accountant activities such as record keeping, inventory, etc. This project setting is to off-shore project's tasks between the organization's teams. These project activities were distributed among two teams from the United Kingdom and Pakistan. The team from United Kingdom was responsible for requirements engineering, design, and technical writing stages. The team from Pakistan was responsible for project management, coding, testing, and deployment stages. The total number of employee involved in this project is six. Although, the estimated duration of this project to be of four calendar months by experts' judgment, the project was delayed one and half calendar months extra. Nevertheless, the completion time is five and half calendar months.

2.3.3. Project C

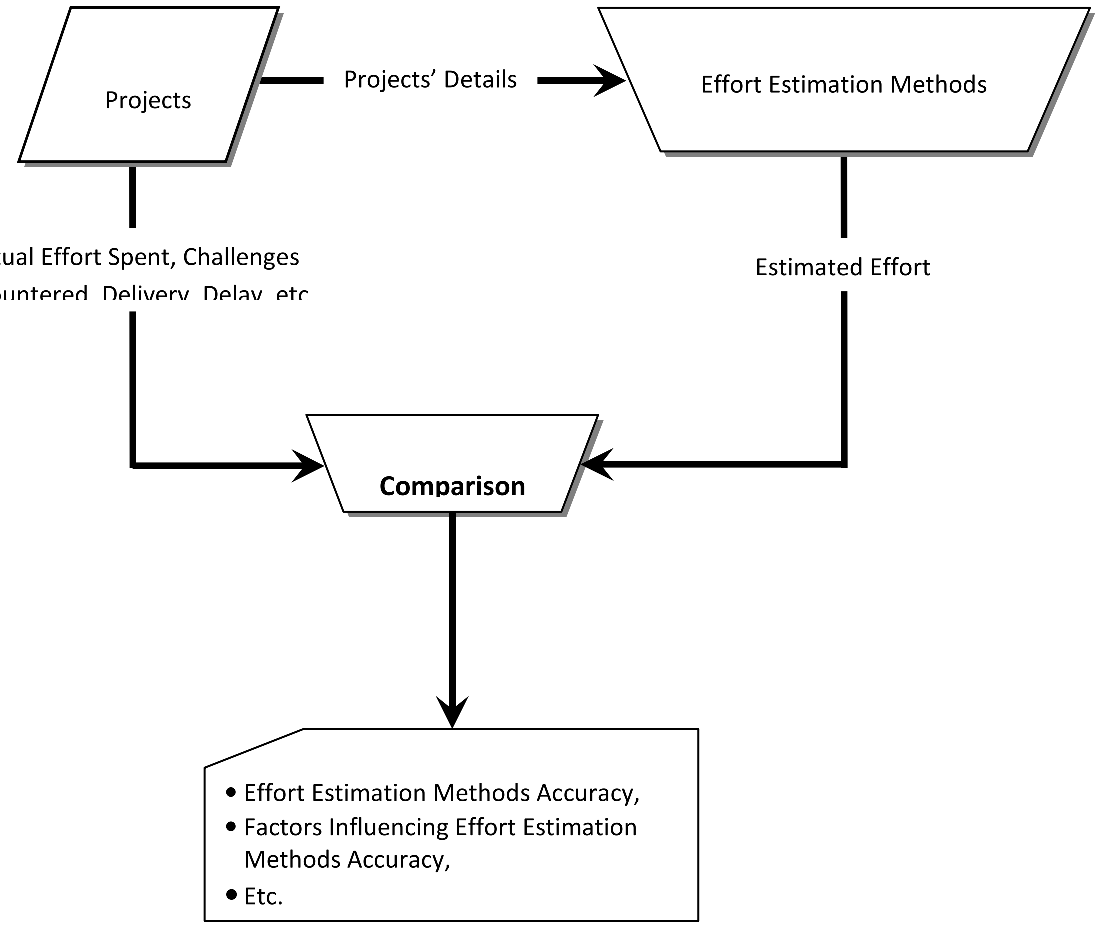
This project aims at developing a Web-based system from public relation and services. These software development activities were distributed among two teams from United States of America and Pakistan. The project's is set to off shore project's tasks between the organization's teams. The team from the United States of America was responsible project management, requirements engineering, and deployment stages. On the other hand, the team from Pakistan was responsible for design, testing, and coding stages of the project. The total number of employees involved in this project is eight employee. The estimation methodology used is experts judgment. The estimated time for project completion is three calendar months. The project failed to deliver the software product within the proposed time frame. Furthermore, the delivery time was exceeded by an extra calendar month.

3. Research Methodology and Framework

This paper focuses on collecting empirical data from projects that were executed adopting the aforementioned strategy. Data collection in this multi-case study involved twelve semi-structured qualitative interviews, through which a rich understanding was developed based on the experiences of those deeply immersed in the practice of Global Software Development (GSD). The interviews were of approx. one and a half hour duration each, with follow up email contact used to refine issues as they emerged. Those interviewed included site managers, project managers, a project architect, team leads, software engineers and technical support staff. All interviewees were directly involved in GSD activities at the companies.

The qualitative analysis techniques of open and axial coding were adopted for analyzing the transcribed interviews. Complementary to the interviews, on-site meetings were held. After the first round of interviews, member-checking was performed, a followed supplementary interviews for allowing for more in-depth exploration of the research topic. However, projects' empirical data were used to estimate the required effort for project completion using the aforementioned effort estimation methods. Furthermore, a thorough and rigorous analysis were performed on the comparison stage between the pre-estimated time by projects' managers, the actual effort spent on projects' completion, and the outcomes derived from effort estimation methods. Figure 1 depicts the research framework.

Figure 1: Research Framework



4. Results

As stated in section 1, effort estimation methods were designed to anticipate the amount of required effort for in-house software development projects. This paper is concerned with the most conventionally effort estimation methods' accuracy in the context of globally distributed environment.

The results are conducted from the application of effort estimations methods COCOMO II and SLIM using the illustrated data from the abovementioned completed projects Post-Architecture. The results represent these methods' estimations for the required amount of effort to complete these projects. Furthermore, it represents the difference between the actual effort spent on these projects completion and the estimations produced from the effort estimation methods COCOMO II and SLIM.

The effort estimation method COCOMO II conducted estimations for the required amount of effort is presented in a Person-Month. The effort estimation method SLIM represents the required amount of effort to complete a given software development project in 'Person/Man-Year', therefore, in order to adjust the unit between effort estimation methods the unit is converted by dividing the outcomes on 12 months. Most importantly, the represented results from the effort estimation method COCOMO II are double-checked via using authorized tools provided by the effort estimation methods COCOMO II sponsor. Nevertheless, the authors have applied the effort estimation methods COCOMO II and SLIM separately on three rounds in order to eliminate mistakes. However, tables 3, 4, and 5 depict a comparison between estimation results using the aforementioned effort estimation methods and the actual effort spent on these projects' completion.

Table 3: Projects A, B, and C Effort Estimation Using COCOMO II

Criteria	Project A	Project B	Project C
Actual Effort Spent (P-M)	14.36	17.15	19.53
Estimated Effort Using COCOMO II (P-M)	12.1	15.8	18
Deviation	2.26	1.35	1.53

Table 4: Projects A, B, and C Effort Estimation Using SLIM

Criteria	Project A	Project B	Project C
Actual Effort Spent (P-M)	14.36	17.15	19.53
Estimated Effort Using SLIM (P-M)	13.8	15.6	17.5
Deviation	0.36	1.55	2.03

Table 5: Project A, B, and C Effort Estimation According to the ISBSG Database

Criteria	Project A			Project B			Project C		
Function Points	501			580			641		
ISBSG Output Elapsed-Time in Months	Lower 3.32	Estimate 7.44	Upper 16.65	Lower 3.54	Estimate 7.92	Upper 17.74	Lower 3.7	Estimate 8.23	Upper 18.53
Actual Effort Spent (Month)	4			5.5			4		
Deviation Elapsed-Time in (Month)	Lower 0.68	Estimate 3.44	Upper 12.65	Lower 1.96	Estimate 2.42	Upper 12.24	Lower 0.3	Estimate 4.23	Upper 14.53

5. Analysis and Validation

This paper aims at exploring the accuracy of effort estimation methods in the context of globally distributed environment. In order to accomplish these, empirical data were collected from three projects. The data extracted from these projects are used to illustrate the required effort for the aforementioned projects' completion using the most conventionally used effort estimated methods: COCOMO II and SLIM. Since each of the afore mentioned project is distinguished from other projects by coping with certain factors that is associated with the GSE environment, the comparison between these effort estimation methods is based on the deviation between these results.

As shown in tables 3, 4, and 5, the deviation between effort estimation methods readings and actual effort spent on projects' completion assures the influence of globally distributed environment factors on projects' effort estimation, and therefore, successful completion. Additionally, the deviation is varying from one project to another, and from other effort estimation methods readings.

The deviation between the actual effort spent on projects' completion and the estimated effort vary depending on the projects characteristics and the encountered factors affecting effort estimation method. For example in project A, the deviation between the actual effort spent is 14.36 Person-Month and the estimated effort using COCOMO II is 12.1 Person-Month and using SLIM is 13.8 Person-Month. Furthermore, the values extracted for the estimated effort for projects' completion using the aforementioned effort estimation methods are always lower than the actual effort-time spent on completing the projects. Thus, the effort estimation methods are optimistic regard the required effort.

In project A, achieving shared resources and same organization or team structure in a globally distributed software development project have a great influence on the accuracy of effort estimation method COCOMOII. The deviation between actual effort spent and the effort estimated using COCOMO II is 2.26 Person-Month. On the other hand, the effort estimation method SLIM showed a close readings 13.8 Person-Month to the actual effort spent in project A 14.36 achieving a smaller deviation. As Moe and Smite (2007) denoted, trust impersonates a focal role in project success, especially as it affects teams' productivity and commitment. However, when trust between team members and partners in the context of globally distributed environment exists, the effort estimation methods SLIM and COCOMOII produce more accurate effort estimation than other factors. For example, in project B, trust exists between team members'. Moreover, trust and unrealistic milestones are coped with, and therefore, it is not included as an influencing factor. However, the deviation between the actual effort spent is 17.8 Person- Month and the effort estimated using COCOMO II is 15.8 Person-Month and for SLIM is 15.6 Person-Month which is lower than the deviation between deviation between the actual effort spent in project A, and C. Furthermore, if trust exists between globally distributed team members, and milestones are feasible then the recommended effort estimation value is COCOMO II, as this method showed a closer readings to the actual effort spent.

Additionally, when trust and commitment together is accomplished between a project teams' members and to their assigned tasks then the recommended effort estimation method is COCOMO II. For example, in project C, trust and commitment was not included on the list of influencing factors. Due to the achievement of trust and commitment in project teams' members, the deviation between effort estimation method COCOMO II is 1.55 Person-Month, 2.03 Person-month for SLIM, and the deviation between the actual time spent on projects completion and the minimum time estimated Lower value by ISBSG 0.3 nominates the ISBSG as candidate effort estimation method, especially as it is more accurate to the actual effort spent.

The databases provided by the ISBSG to support the experts' judgment to estimating the required amount of time to complete a given software development project provides three estimation values: the lower anticipated time to projects' completion, the time the project is more likely anticipated to complete, and the maximum time the project may consume in order to finish. The deviation rate in ISBSG estimated values is high which adds another risk to the developed software product to be delivered in a cost effective way. For example, the deviation rate in project A is 16.77±5.59 from the actual elapsed time spent on projects' completion in months. The lower time anticipated for projects' completion has the minimum deviation between the actual times spent on these projects' completion and other estimation values produced by effort estimation methods. For example, the deviation between the anticipated minimum time for project completion and the actual time spent on project completion are 0.68 for project A, 1.96 for project B, and 0.3 for project C.

The level of impact the encountered factors in the globally distributed environment have on the accuracy of effort estimation methods is measured using an ordinal scale of three values: Low, Medium, and High accuracy. These values are based on the deviation found between the actual effort-time spent for completing the project and the estimation values conducted from effort estimation methods. The accuracy is considered high when the deviation value is less than 1, the value of medium impact on accuracy of effort estimation methods is considered medium when the deviation value is equal or greater than 1 and less than 2. Furthermore, the value of low impact on the accuracy of effort estimation methods is considered as low when the deviation value is greater than 2. However, table 6 represents the level of impact on the effort estimation methods and the ISBSG estimations conducted from other completed projects. However, in table 6, the deviation value considered for projects' completion in the ISBSG estimations is lower estimated elapsed time for completing a given project.

The validity of this research consists of two parts: projects selection and the application of effort estimation methods on the extracted information of the previous selected projects. Starting with the last, the application of effort estimation methods were done three times by authors. The redundant application of effort estimation methods produced the same values as shown in tables 3, 4 and 5 above. Furthermore, the application of effort estimation methods were conducted using computerized tools provided by these methods' providers.

Since the authors have illustrated several different software development projects from industry, the selection process itself was rigorous enough to guarantee the availability of required information to explore the accuracy of effort estimation methods in the context of globally distributed projects. However, each of the selected project life do not exceed a year long. Furthermore, each project aims to develop a software product of different types: web-based system, computer-based system, etc. Each of these projects used the Globalization strategy in planning, developing, executing, and delivering their outcomes.

Table 6: GSE Factors Level of Impact on the Accuracy of Effort Estimation Methods

Factors	Effort Estimation Methods and Simulators								
	COCOMO II	SLIM	ISBSG	COCOMO II	SLIM	ISBSG	COCOMO II	SLIM	ISBSG
Shared Resources	×	×	×						
Same Organizational and Team Structure	×	×	×						
Trust				×	×	×	×	×	×
Unrealistic Milestones				×	×	×			
Lack of Commitment							×	×	×
Deviation Value	2.26	0.36	0.68	1.35	1.55	1.96	1.53	2.03	0.3
Impact Level	High	Low	Low	Medium	Medium	Medium	Medium	High	Low
Accuracy Level	Low	High	High	Medium	Medium	Medium	Medium	Low	High

6. Discussion

This paper focuses on exploring the accuracy level of the most conventionally used effort estimation methods in the context of globally distributed software development projects. Effort estimation methods investigated in this paper are the COCOMO II, SLIM, and ISBSG. The effort estimation method COCOMO II provides equations and constants values that can be used in the effort estimation process. Additionally, the adoption of COCOMO II in effort estimation processes of an organization is not costly, especially as COCOMO II provides a free calculation tool. The use of COCOMO II requires expertise. On the other hand, two main parameters are found for its calculation i.e. PI and MBI. There are two ways to find PI value, either from history project or from SLIM database. Thus, the adoption of SLIM method in effort estimation processes for a new organization may not be possible. The ISBEGS does not include specific factors that lead to meeting proposed deadline or extend delivery time. Additionally, the ISBEGS databases can be used to consolidate the Experts' judgment effort estimation method.

7. Conclusions and Future Work

As stated in the introduction, effort estimation methods are designed and dedicated to estimate the required amount of effort for developing a software product in the common way, that is, in-house. However, the effort estimation methods investigated in this paper provide estimations that are less than the actual time to complete the given software development projects. The accuracy of the aforementioned effort estimation methods are greatly influenced by the environment in which the software project is executed. Additionally, according to the deviation between the actual effort / time spent on completing the above illustrated projects and the estimations conducted from effort estimation methods, we conclude that the development of a software product in a globally distributed environment consumes more effort and more time to complete.

Results from the studied cases showed that the existing effort estimation methods' accuracy is influenced by the lacking of factors related to GSD environment. The existing effort estimation methods need improvement so that they estimate accurate effort for GSD projects. It may need to add / remove or merge the existing factors of both the models accordingly. Furthermore, the current methods e.g. COCOMO II and SLIM require amplification and calibration with respect to GSD requirements. When considering average results' deviation, COCMO II gave closer results to the actual efforts of studied projects. Based on these results, COCMO II is more suitable for GSD projects contrary to SLIM. However, some constraints and other factors also required to be taken into consideration for the selection of suitable effort estimation method for a given situation.

Globally distributed environment enclose many factors, and most of these factors' influence on the accuracy of effort estimation methods is still poorly explored and measured. Therefore, the Experts' Judgment effort estimation method shall be used after conducting a rigorous and thorough analysis of the projects' development environment, factors enclosed in this environment, and other influencing factors. Furthermore, in order to improve this effort estimation method, Experts' judgment can be combined with other mathematically-based effort estimation methods. In general, this method shall be used with a deep awareness of the development environment, factors enclosed on the environment, and other issues that may influence the estimation accuracy.

The globally distributed environment encloses many challenges and factors. Therefore, further research lines can be undertaken to investigate these challenges and factors impact on effort estimation methodologies. Furthermore, since effort estimation methods are always underestimating the required amount of effort- time to complete a given software development project, authors suggests to improve the process of effort estimations for the context of Globally Distributed Software Development.

References

[1] Ågerfalk, P., Fitzgerald, B., Holmström, H., Lings, Lundell, B. and Conchúir, E., 2005. "A Framework for Considering Opportunities and Threats in Distributed Software Development", in *Proceedings of the International Workshop on Distributed Software Development, Austrian Computer Society*.

[2] Ågerfalk, P., Fitzgerald, B., Olsson, H. H. and Conchúir, E. Ó., 2008. "Benefits of Global Software Development: The Known and Unknown", in *Proceedings of the International Conference on Making Globally Distributed Software Development a Success Story*, pp. 1-9.

[3] Bartelt, C., Broy, M., Herrmann, C., Knauss, E., Kuhrmann, M., Rausch, A., Rumpe, B. and Schneider, K., 2009. "Orchestration of Global Software Engineering Projects", in *Proceedings of the 3rd International Workshop on Tool Support Development and Management in Distributed Software Projects (REMIDI'09), Limerick, Ireland*.

[4] Betz, S. and Mäkiö, J., 2007. "Amplification of the COCOMO II regarding offshore software projects,"in *Proceedings of the 2nd International Conference on Global Software Engineering 2007 workshop*, pp. 35-46.

[5] Boehm, B. W., 1981. *Software Engineering Economics,* USA: Printice-Hall.

[6] Boehm, B. W., Abts, C., Brown, A. W., Chulani, S., Clark, B. K., Horowitz, E., Madachy, R., Reifer, D. J., Steece, B., 2000. *Software Cost Estimation with COCOMO II*, USA: Printice-Hall.

[7] Boehm, B. W. and Valerdi, R., 2008. "Achievements and Challenges in COCOMO-Based Software Resources Estimation", *IEEE Software* 25(5), pp. c1-c1.

[8] Carmel, E., 1999. Global Software Teams: Collaborating Across Borders and Time Zones. USA: Printice-Hall.

[9] Conchúir, E., Ågerfalk, P. J., Olsson, H. H. and Fitzgerald, B., 2009. "Global software development: Where are the benefits?", C*ommunications of the ACM* 52(8), pp. 127-130.

[10] Feeny, D., Willcocks, L. and Lacity, M. C., 2006. *Information Systems Outsourcing: Enduring Themes, New Perspectives and Global Challenges*, 2nd ed. Germany: Springer, Chapter: Business Process Outsourcing, Knowledge and Innovation – A Study of Enterprise Partnership, pp. 544-580.

[11] Hersleb, J. D. and Mockus, 2003. "An Empirical Study of Speed and Communication in Global Software Development", *IEEE Transactions on Software Engineering* 29(6), pp. 481-494.

[12] ISBSG, 2009. "ISBSG Database", Online: http://www.isbsg.org/, Last visit on: May 22, 2009.

[13] Kemere, C. F., 1987. "An Empirical Validation of Software Cost Estimation Models,"in *Communications of ACM* 30(5), pp. 416-429.

[14] Kile, J. F., Little, D. and Shah, S., 2005, "The Importance of Effective Requirements Management in Offshore Software Development Projects", Ph.D. dissertation, Pace University.

[15] Kormeren, R. and Parvianen, P., 2007. "Philips Experiences in Global Distributed Software Development", *Empirical Software Engineering* 12(6), pp. 647-660.

[16] Moe, N. B. and Smite, D., 2007. "Understanding Lacking Trust in Global Software Teams: a Multi Case Studies", in *Proceedings of the PROFES 2007*, p. 20-34.

[17] Nguyen, V., Steece, B. and Boehm, B., 2008. "A Constrained Regression Technique for COCOMO Calibration,"in *Proceedings of the 2nd ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*, pp. 213-222.

[18] Panlilio-Yap, N., 1992. "Software Estimation using the SLIM Tool", in *Proceedings of the 1992 conference of the Centre for Advanced Studies on Collaborative research*, IBM Press, pp. 439-475.

[19] Shepperd, M., Schofield, C. and Kitchenham, B., 1996. "Effort estimation using analogy", in *Proceedings of the 18th International Conference on Software Engineering*. Washington, DC, USA: IEEE Computer Society, pp. 170-178.

[20] Smite, D., 2006. "Global Software Development Projects in one of the Biggest Companies in Latvia: is Geographical Distribution a Problem?", *Journal of Software Maintenance and Evolution: Research and Practice* 11(1), 2006, pp. 61-76.

[21] Smite, D., 2007. "Global Software Engineering Improvement", Ph.D. dissertation, University of Latvia.

[22] Šmite, D. and Borzovs, J., 2008. "Managing Uncertainty in Globally Distributed Software Development Projects", *Scientific Papers University of Latvia - Computer Science and Information Technologies* 35, pp. 9-23.

[23] Wittig, G., 1997. "Estimating Software Development Effort with Connectionist Models", *Information and Software Technology* 39(7), pp. 469-476.