



International Journal of Technology in Education and Science (IJTES)

www.ijtes.net

Software Development Estimation Techniques in Industrial Contexts: An Exploratory Multiple Case-Study

Ahmed Zarour, Samer Zein
Birzeit University

To cite this article:

Zarour, A. & Zein, S. (2019). Software development estimation techniques in industrial contexts: An exploratory multiple case-study. *International Journal of Technology in Education and Science (IJTES)*, 3(2), 72-84.

This article may be used for research, teaching, and private study purposes.

Any substantial or systematic reproduction, redistribution, reselling, loan, sub-licensing, systematic supply, or distribution in any form to anyone is expressly forbidden.

Authors alone are responsible for the contents of their articles. The journal owns the copyright of the articles.

The publisher shall not be liable for any loss, actions, claims, proceedings, demand, or costs or damages whatsoever or howsoever caused arising directly or indirectly in connection with or arising out of the use of the research material.

Software Development Estimation Techniques in Industrial Contexts: An Exploratory Multiple Case-Study

Ahmed Zarour, Samer Zein

Abstract

Software Effort Estimation is one of the most challenging aspects in the software development life cycle. Recent empirical studies in the area of software development estimation indicate the presence of two models for effort estimation: (i) Formal, and (ii) Expert Based (Informal). The IT sector in Palestine is one of the most promising and constantly growing sectors. Nonetheless, studies addressing effort estimation approaches and techniques within the Palestinian IT sector are still highly missing. Therefore, we were motivated to conduct a qualitative study to increase our understanding about how industrial teams approach software effort estimation and to explore the challenges they are facing. Our investigation started with a survey that targeted software professionals, and then we conducted multiple-case study approach involving four different software development companies in Palestine. Results show that: (i) around 25% of cost overrun in software projects is due to inaccurate estimations; (ii) expert based estimation models are the mostly applied models especially within agile environments; (iii) a potential advantage can be achieved when formalizing the process of expert based models by having guidelines and checklists; (iv) accuracy of effort estimation is largely affected by team experience, domain knowledge, and requirements clarity; (v) companies working with outsourcing model do have better effort estimation accuracy than companies working in local market. Based on our findings, we highlight areas that require further investigation.

Keywords: Effort Estimation, Case-study, Software engineering, Measurement, Agile.

Introduction

Delivering software products within budget and time is not a trivial task. In almost all cases, software projects contain multiple resources that must be accurately estimated (Minku & Yao., 2013). Software Effort Estimation is one of the major challenges in the software development life cycle (Mansor, Razali, Yahaya, J., Yahya, S., & Arshad., 2016). It has been found that inaccurate effort estimation can lead for project deviation and in some cases, a total project failure (Lenarduzzi, Morasca, & Taibi., 2014). Earlier and recent studies (Nguyen-Cong & Tran-Cao., 2013) (Bilgaiyan, Sagnika, Mishra, & Das., 2017) (Tailor, Saini, & Rijwani., 2014) (Usman, Mendes, Weidt, & Britto., 2014) (Rijwani, Jain, & Santani., 2014) (Borade & Khalkar., 2013) (Usman, Mendes, & Börstler., 2015) (Usman & Britto., 2016) show that effort estimation can be approached in different approaches. Software development methodologies (Waterfall, Agile...etc.) have different models of efforts estimations and each model has its strengths and weaknesses regarding the quality of estimation it produces.

Models of efforts estimation can be classified into two main categories, formal and expert-based (Ramacharan, & Rao., 2013). Both models have many challenges and, in some cases, they are combined to enhance estimation quality. For instance, expert-based models frequently have higher degree of inconsistency. Nevertheless, calibrated formal models are complex and require lots of parameters' calibration to fit to the needs of an organization (Minku & Yao., 2013). Even though there are plethora of studies targeting software development and task prediction in general (Mansor et al., 2016) (Colomo-Palacios, 2014), or in a specific software methodology like agile (Jacobs, 2005) (Panda, Satapathy, & Rath., 2015) (Popli, & Chauhan., 2014) (Porru, Murgia, Demeyer, Marchesi, & Tonelli., 2016) (Lenarduzzi et al., 2014), no studies focus on understanding how Palestinian companies approach software prediction and the challenges they face. The Palestinian IT sector is a promising sector and has matured companies working in software development sector (Zein, Salleh, & Grundy, 2015). Understanding the current approaches taken by the Palestinian experts and IT professionals to conduct effort estimation is a key point in understanding the problems facing companies in the delivery of high-quality

software products. Factors affecting the estimation quality and accuracy are the main issues that determine the estimation techniques used and quality of effort estimation.

Related Work

To our best knowledge, only the study by (Zein et al., 2015) has targeted Palestinian software development companies qualitatively. In their study, the authors focused on exploring testing challenges and methods applied in the area of mobile app testing. However, in this paper, the aim is to explore the challenges that Palestinian software development companies are facing in the specific area of effort estimation. A few secondary studies have reviewed software effort estimation in terms of Systematic Literature Review (SLR) studies (Nguyen-Cong & Tran-Cao, 2013) (Bilgaiyan et al., 2017) (Usman et al., 2014); and surveys investigating the state-of-the-art practices within effort estimation (Borade & Khalkar, 2013) (Usman & Britto, 2016). The SLR by Nguyen (Nguyen-Cong & Tran-Cao, 2013) reviewed the current research literature on effort estimation within agile environment. They have tried to show evidence about common trends and practices used with agile and iterative environments. The results of their study show few empirical researches in the field of effort estimation and most of the research is focused on XP, general agile, iteration and increment development methods. Additionally, their results show that Expert based judgment is the most used approach in effort estimation. On the other hand, the SLR by Bilgaiyan (Bilgaiyan et al., 2017) explored the mechanisms used in effort estimation within agile development methods. The study also explored the parameters affecting the estimation accuracy. The authors argue that 60% of projects have cost overrun and around 15% totally failed projects caused by wrong estimation. Additionally, the results show that choosing an estimation technique is related to some circumstances and factors. Usman (Usman et al., 2014) provides a detailed overview of the state of the art in the area of effort estimation within agile software development (ASD) context. Results show that expert-based judgment technique is the most frequently applied technique in effort estimation. Team skills, experience and task size found to be the most important cost drivers affecting effort estimation output.

Formal Models, or what's so called Parametric Models are famous Software Effort Estimation models applying mathematical, machine learning and defined formulas to quantify efforts. Formal models have been applied within software development for more than 40 years (Jorgensen, Boehm, & Rifkin., 2009). Thousands of researches and studies have approached formal models and several software engineering textbooks and guidelines described and promoted them (Jorgensen et al., 2009). Ramacharan (Ramacharan & Rao., 2013) describes various formal models within his research on identifying the problems that arise within offshore and distributed development teams. In their study, the authors state that formal models, within its current shape and without calibration, do not have the required competence to provide effort estimation for distributed software projects. Popli (Popli & Chauhan., 2014) suggests an algorithm to be used in effort estimation in case experts and historical data are not available. In addition, even when the project related factors are of high level (High Impact), if people related factors like communication skill, managerial skill and familiarity in team are well defined, the project takes relatively less time. The study by Lenarduzzi (Lenarduzzi et al., 2014) tries to improve existing estimation models by monitoring and evaluating the project costs after each development phase. Furthermore, significant correlation between efforts spent in one phase with effort spent in the following one. Additionally, a significant correlation between effort spent in a phase and the remaining efforts has been spotted. Along with significant correlation between the cumulative effort up to the current phase and the remaining effort.

Informal models and Experts based judgment model in specific is becoming the most used model within Software Effort Estimation (Usman et al., 2014) (Bilgaiyan et al., 2017). Studies show that the use of expert-based models achieves more accuracy than the use of formal models (Jorgensen, M., Boehm, B., & Rifkin, S., 2009). Usman (Usman, Petersen, Börstler, & Neto., 2018) suggested an improvement to expert-based judgment model. Additionally, the expert-based judgment model is the mostly used model. However, Experts Based Models lacks formalism of the process. Experts are humans and can miss some important factors like testing effort, non-functional requirements etc. Hamouda have also approached formalization of the Expert Based Judgment process in their study (Hamouda, 2014). The author proposed a methodology to formalize the project sizing process while using agile story points within a CMMI organization. Furthermore, the agile story point size can vary depending to project and the development team, a formal process should be approached to determine agile story point size. The study by Dagnino (Dagnino, A., 2013) tries to improve effort estimation in case historical data is not available. The results show an improvement in initial effort estimation if multiple models are applied. If historical data is not available a combination of both Expert Based Models and Formal Models can be utilized to derive the estimates. However, the combination of Formal and Informal models requires more empirical evidence. The impact of change is also another important factor affecting experts-based

estimation output (Tanveer, 2017). According to Tanveer et al., the main methods of experts-based judgment such as Planning Poker, Analogy, and Disaggregation do not consider the use of tools and techniques to determine the impact of change during estimation.

Methodology

In this study, we followed a mixed research methodology that consisted of two phases. In the first phase, we conducted a survey to investigate how professionals at Palestinian software development companies approach effort estimation in general. While on the second phase, we conducted a qualitative research to gain deeper understanding about applied effort estimation techniques and challenges faced. The outcome of the first phase helped us better design the protocol questions of the second phase.

Survey Design

The survey was built initially for collecting general information about how Palestinian software development companies approach effort estimation. The survey was distributed for one month over forums and groups of software and IT practitioners, using Facebook groups “PalGeeks” and “Peeks” as well as the linkedIn platform. Survey questions were designed to be closed and open-ended, in which, responded can add more information to the answers. Later on, these answers were used in building and designing more targeted interview questions for the second phase. Table 1 below shows the list of survey questions and explain the objective of each question and its relation and effect on interview questions and hypothesis generation.

Table 1: Survey questions and objectives

Question	Objective	
1	What is your current job title	To have a general look on the software development jobs titles to which we can know if there are any effort estimation related titles.
2	How much experience you have in software development field	To know how is the software development experience is distributed and how much experienced developers exist
3	How to evaluate your knowledge in the software effort estimation field?	To understand and know if the responded have previous knowledge in software effort estimation and how much is that knowledge to better phrase our interview questions and to make sure we do need a pre-interview meeting to explain the subject.
4	What is the average accuracy for estimation provided in initial phases of the project (based on your experience)	To know if there is any difference in initial project estimation than estimation within planning phases of a project and if there is an accuracy in provided estimations or not.
5	What are the techniques you or your company use in software effort estimation	To have an overall overview on which techniques used and to better design our interview questions
6	What metrics you use in measuring effort estimation	To have an overall overview on which metrics used and to better design our interview questions
7	If you rely on your experience while providing effort estimation for a software project, do you depend on historical data	To get more information if there is any use of historical data and consider this within our interviews.
8	Do you use tools to support you in effort estimation	To have an overall overview on which tools used and to better design our interview questions
9	How is effort estimation is done in your company	This is an open-ended question that we set to better know how effort estimation is done (team based, user based, managers based) in order to design our interview questions. some predefined choices were provided to the responded with ability to enter more.
10	In your Opinion, what are the main reasons causing projects estimation to NOT be accurate	A list of responses was set with ability to add more, this question was very important to know what cases we should focus on within our interview questions
11	In your opinion, does the methodology you use in project development (Agile, Waterfall, Prototype. etc.) affect effort estimation accuracy and techniques	To understand and collect information about any possible relation between methodology and estimation techniques so that we do design our interview questions accordingly.
12	In your opinion, does the location of developers (on premise or work from home) affect effort estimation	To understand and collect information about any possible relation between location of developers and estimation techniques so that we do design our interview questions accordingly.
13	In your opinion, does project size (small, medium large) affect efforts estimation	To understand and collect information about any possible relation between project size and estimation techniques so that we do design our interview questions accordingly.

Case Study Design

A case study can have multiple shapes, a multi-case-study approach is somehow another expanded implementation of a single case study in which a study can have multiple cases to study. A case can be a team of developers, or a company, in multiple case studies. Researchers can study multiple teams or multiple companies. This study investigates four industrial cases, each case represents a Software Development company that operates within Palestine and work within different business models, local development model and outsourcing model.

Theoretical Proposition

Theoretical propositions, as discussed by Yin (Yin, 2009), will form the main pillar for the case study design strategy. Theoretical propositions, is the most preferred strategy and is based on following the theoretical propositions that led to the case study design. Additionally, propositions help to focus and put more emphasize on certain data and to ignore another data. For instance, in this case study, the focus will be on how the developers and team leaders are using specific effort estimation techniques such as formal models or informal models.

In this study, the hypotheses (theoretical propositions) were as follows:

1. Accurate effort estimations require clear and detailed processes and guidelines to be considered by estimation team.
2. Expert based models are the most used models in estimation within Palestinian companies.
3. Missing Guidelines and missing use of historical data when applying experts-based models.
4. The use of formal models is either missing or is used without calibration.

Research Questions

The Research questions we try to answer in this study are:

RQ1: How do industrial teams approach effort estimation? Moreover, what are the challenges they are facing?
Objective: Understand how development teams approach effort estimation in general.

RQ2: What are the factors that influence the accuracy of efforts estimation that are considered relevant and important? In addition, what are the real causes of these factors?
Objective: Investigate the specific challenges faced by development teams and what factors that cause effort estimation

RQ3: Do project type and company business type influence the accuracy of efforts estimation?
Objective: Investigate whether project type and business model type have influence on effort estimation.

Case Selection and Context

In Cases selection the potential companies in Palestine are a lot, however choosing between them can be a challenge since lots of companies do professional software development. The main criteria suggested for choosing companies is the availability and acceptance of the company to participate in the case study. Another criterion is the size of the development team. The potentially required team size is between 5 to 9 developers in an agile team, therefore the selection of teams was based on this number in which the team under study have this number of developers. The size of the team was based on the high probability of Palestinian companies using agile as a main development methodology for software delivery (Zein et al., 2015). It is expected that the selected companies do use agile hence agile development is the most used software development methodology (Anwer, Aftab, Waheed, & Muhammad, 2017).

Data Collection Procedures

Data has been collected through interviews, focus groups and observations. In the three cases C1, C2 and C3 both interviews and Focus groups have been conducted, However C4 was the only company that allowed data

collection through observations. The data collection strategy for the cases C1, C2 and C3 consisted of two parts. The first part was one to one interview with development manager and senior resources. the second part was a focus group with part of the development team members. However not all team members in the three companies were available for the interview. Selection of members was done by the development manager in each company.

Collection of data through observations was done in C4 premises. The first author had been working with C4 as a part time consultant and had access to the development team within their planning sessions. Notes and activities for the teams in C4 were taken during observation sessions (observant without participation). Observation sessions had been done on two sprint planning for two different projects with two different teams.

All activities involved in effort estimation while planning has been investigated which include the following:

1. All effort estimation activities and steps taken, recording any use of special tools and focusing on the practice and techniques used within effort estimation.
2. How requirements are being handled and how effort estimation is being done on requirements stories, features or bugs.
3. How effort estimation is validated and if a team or specific persons are validating estimations output.
4. What challenges and factors affecting the estimation output, what considerations and constraints are being considered?

The interview protocol was designed prior to data collection, a face-to-face as well as email interviews were conducted with managers and other software development team members. Interviews had been designed as semi-structured since they are well suited for this kind of research (Hancock & Algozzine., 2016). Interview questions were designed as per the survey conducted prior to conducting the data collection for the case study. Questions were open ended and interviewees had the flexibility to express their thoughts and opinion along with their experience freely and from their own perspective. Each interview lasted for around 45 minutes and due to the unavailability of all team members, only selected team members by the development manager in each company have been interviewed.

We have also conducted focus groups in each company with team members to better collect data with more insight on the process of effort estimation in a way a discussion between team members provided more information about different approaches taken even within same team in effort estimation. Additionally, during group discussion our role was to moderate the discussion giving each team member the chance to express his thoughts and giving his opinion, after that a discussion between team members was done and recorded. We have recorded and stored data collected from interviews, focus groups and observations using audio recorded voice that has been transcript into documents. The data within documents had been coded and each sentence was given a code and then we did group of codes with a link of each code to original document.

Results

This section presents the results from four case studies which represents our multiple case study. The cases represent 4 companies operating in Palestine specialized in software development, three of the companies have been selected for interviews and focus groups and the fourth company has been selected for observations. For confidentiality issues we will refer to the companies with C1, C2, C3 and C4 representing the four cases. Table 2 describes the demographics of the selected cases. Additionally, a survey was done for collecting information about effort estimation within the Palestinian market. The survey was built using Google forms and distributed for 2 weeks using Facebook groups related to Palestinian IT Facebook groups, the researcher owns linked in profile and emails to known participants upon the researcher relations.

Table 2. Demographics of the studied cases

Case Code	Domain	Methodology	Technology	Size
C1	Web Tools for Managing Infrastructure Network devices (Outsourcing)	Scrum	Java Based	Large
C2	Custom software development	Scrum, Waterfall, and Prototyping	.NET, Oracle	Medium
C3	Web Tools for Managing Infrastructure Network devices (Outsourcing)	Scrum	Python	Large
C4	Custom Web and Mobile Apps for AID sector	Kanban, Scrum and Iterative Waterfall	PHP	Medium

Note: Large (more than 100 Developer), Medium (50-100 Developer)

Survey Results

A total of 101 respondents to the survey distributed were 33% senior developers, 21% developers and 35% managers and team leaders, the rest are divided between students and non-IT members. Respondents experience is divided as 40% with 5-10 years of experience within software development 36% less than 5 years and the rest are over 10 years' experience. These results determined the selection of team members for focus groups and interviews with 5 and more years of experience.

Results of show that no actual specialized software estimation jobs are in Palestine. Additionally, experience and knowledge of effort estimation is distributed between 44.6% saying they have moderate knowledge in software effort estimation, 32.7% saying they have string knowledge and a 22.8% saying they have basic knowledge. further, more than 78% of IT practitioners and software developers agree that projects finish late with 25% more or less of effort estimation accuracy error. Additionally, results show that Expert based estimation is the dominant estimation techniques used in the Palestinian IT companies with more than 84%. However, the results also show that around 19% of practitioners do use parametric or formal models. Additionally, results show that Time based metrics are the main metrics used within effort estimation. Followed by story points.

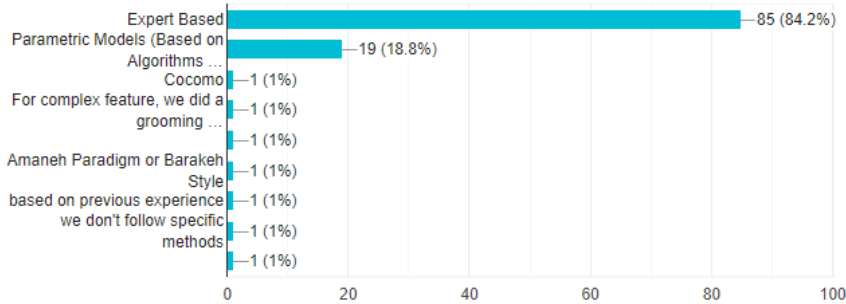


Figure 1. Techniques used in software effort estimation

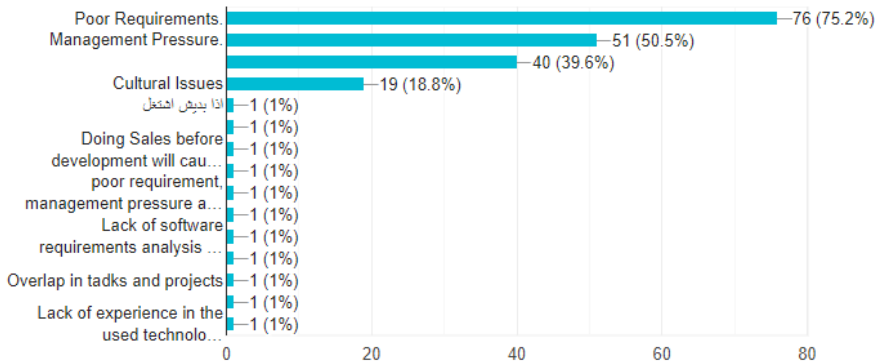


Figure 2. Factors affecting accuracy of effort estimation

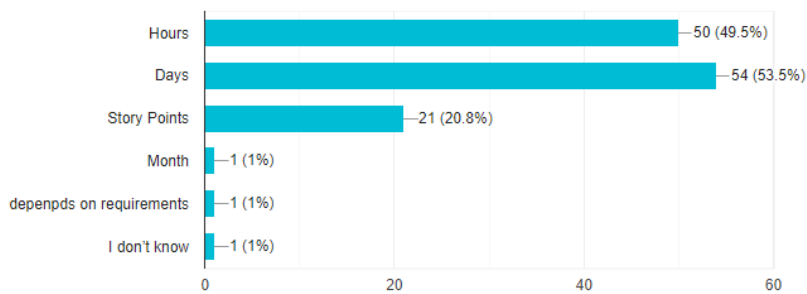


Figure 3. Metrics used in effort estimation

Additionally, results show that more than 46% do not use any tools within their effort estimation, while around 52% do use tools.

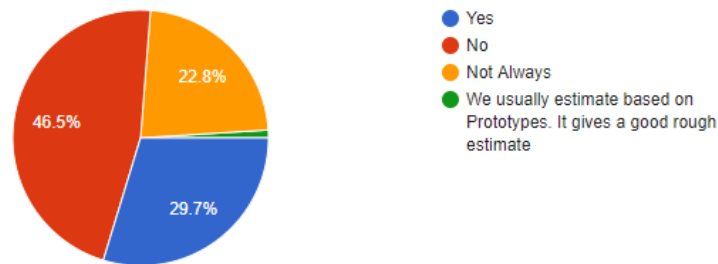


Figure 4. Tools support in effort estimation

Furthermore, it has been shown that more than 73% of effort estimation is being done by all team members, this comes along with the more use of agile methodologies in software development.

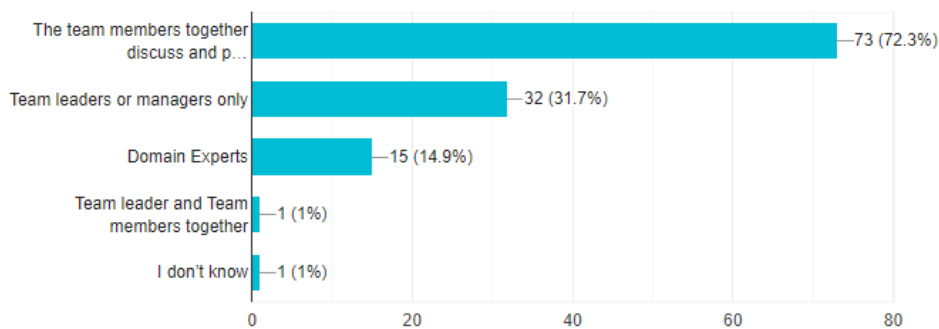


Figure 5. How effort estimation is being done

Additionally, results show that software developers do believe that the use of a specific methodology in software development and the size of the project have a great effect on effort estimation. On the other hand, the developers did not agree that much that the location of developers do have an effect on effort estimation as around 44% agree that the location of developer (working off shore or from home or within the company premises) do have an effect of effort estimation while around 34% did not agree on that.

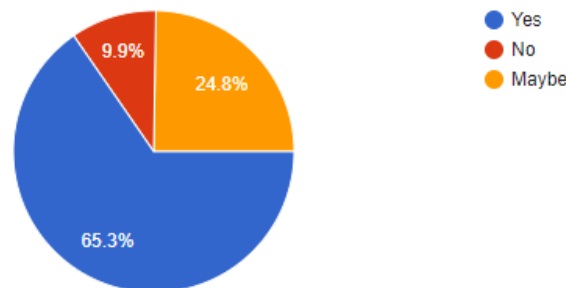


Figure 6. Does the used methodology in software development have an effect on effort estimation accuracy?

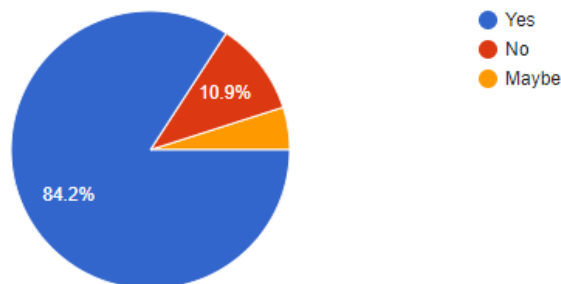


Figure 7. Does project size (small, medium large) affect efforts estimation?

Case Study Results

In this section we provide the results from our qualitative cases studies (interviews, focus groups and observations) in relation to our research questions. The research questions are revolving on three main points to which we address our interview questions and discussed and observed within the selected cases teams.

Techniques used in Effort Estimation

In general, it was noticed that expert-based judgment is the dominant technique used in effort estimation. All cases have approached this technique with differences in the approach. Although this comes aligned with our survey results, but we did not notice any of the cases use any formal or parametric models within their estimations. C1 and C3 approach expert-based estimation within their scrum methodologies. In which team members provide estimation as a team depending on their experience. Shared experience within team members and mutual discussions within the planning phase provide a base for a shared experience to be used in providing as much accurate estimations as possible.

C2 and C4 on the other hand provide expert based estimation in another approach. Although both C2 and C4 do try to use scrum but in the planning phase the estimation is provided by the team leader or development manager alone without consulting with team members. On the other hand, teams within C2 and C4 do share their opinion for the provided estimations but the initial estimations are only provided by managers or team leaders. The results show the absence of any tools in providing effort estimations and all cases do use time-based metrics in their estimations, hours or days are the mostly used time-based metrics. Additionally, results show that C1 and C3 use a technique to estimate features complexity based on number of interfaces and interactions with other modules. However, C2 uses number of screens and number of inputs or operations for each module. Furthermore, C4 do use a combination of both.

We also noticed that the use of historical data to help in expert-based judgment is not used within estimation process. Nevertheless, all cases do store their estimations using tools like JIRA but data is not reached by teams to help in their estimations. However, C1 Development manager mentioned that in rare cases they might go back to previous estimations to help in decisions. Moreover, C3 team leader mentioned that he might go back to earlier estimations as a way of knowledge transfer in cases the resources provided estimations for some modules are not available any more within the team members or they have resigned.

In Addition to the above, results show that all cases did not state any special way to handle initial project estimations that differ from their way in providing estimation in planning phase. They are following same approach whether the estimation is required for an initial requirement or for a full detailed requirement. However, C2 development manager stated that in cases where initial estimation is required on a very high-level, they use surveys or meetings to collect further requirements to provide more accurate estimations.

Factors and Challenges Influencing the Accuracy of Effort Estimation

Most of the cases have highlighted that unclear requirements is the main factor affecting the accuracy of estimation. C1 development manager stated *“The most important reason is unclear requirements; Product manager is not understanding what he wants and of course the requirements come unclear so we start even trying to tell him what he should have or the customer may need.”* Furthermore, Team members and leaders within four cases has agreed that poor requirements affect the accuracy of estimation because developers tend to assume things. assumptions vary from easy and strait forward features to more complex features. Additionally, C2 Development managers said that he uses direct meetings and sometimes surveys with stakeholders to understand requirement better in order to provide more accurate estimations.

Additionally, results show that there are major challenges developers face while estimating a software project. For example, C1 has stated that one of the main challenges that affected their estimation was to manage the agile (scrum) process in a way that team members follow strictly as per the scrum model. Although scrum is considered very useful for us, Said C1 development manager, still the time we took to achieve a standard scrum process was more than 1 year of efforts. *“Doing scrum in a standard way do produce more accurate estimations”* C1 Development manager.

Moreover, C2 stated that the main challenge they have in estimation is the time, *“we always have a strict time to deliver our projects in”* C2 development manager. However, business domain that the developer must understand was main challenge for developers in C2. *“Developers must understand the domain in order to estimate well”*, C2 developer. Additionally, C3 Team member stated that one of the major challenges is the ability to break down requirements into smaller pieces in order for developer to estimate well. Estimating a large feature consists of huge uncertainty.

Software development methodology is a main factor affecting estimation. All cases agree on that. *“The use of agile for example, helped us a lot in providing more robust estimations than what we had in earlier waterfall methodology”*. C1 development manager. Additionally, C2 Developers started that proving mock-ups and prototypes also helps a lot in providing more robust estimations. Furthermore, C3 and C4 agree on that area in which agile is the best model for getting better estimations. *“because of the fact that requirements are divided into smaller features that will help estimating easier and more accurate”*, C2 Developer.

Effect of the Company Business Type and Business Model on the Effort Estimation Techniques

According to (Zein et al., 2015) Palestinian Eco system shows a diversity of software development companies that work in multiple areas. Furthermore, the survey results show an effect of the business type and cultural issue son software effort estimation. The effect has been found in this study results.

C1 and C3 work within an outsourcing model for bigger international companies while C2 and C4 deal with local markets, or in case C4, markets very similar to local markets. Furthermore, both C1 and C3 work within the process of the client they do outsourcing for. Requirements normally comes ready and well established. Additionally, they both work in almost one technology and with no restriction on time. Nevertheless, C2 and C4 work for different clients and provide custom software for different needs every time. *“we do develop systems in various technologies which introduce a huge challenge on the team”*. C2 Software Developer.

Additionally, cultural issues within the company itself do affect the estimation. For example, C1 and C3 has their teams working directly with the international companies and local management do not interfere with their decisions. *“The culture of international company is reflected on the culture of the developers within the Palestinian company”*, C1 Developer said.

Developers location effect on software estimation was splited opinion between cases. C2 development manager did not like the work from home and offshore teams. He stated that the *“communication overhead is not considered while estimating, although it should be”*. C2 Development Manager. Furthermore, C1 and C3 did not consider the location of developers causing a difference or an effect on software effort estimation accuracy.

Discussion

In this section, we provide our discussion based on our key findings and observations on effort estimation within the industrial context of software development in Palestine. More specifically, we map the findings to the study hypothesis and we provide some recommendations to enhance the software effort estimation approach and accuracy; as several issues can be concluded from the observations and results of our case studies.

Techniques used within Software Effort Estimation

In relation to Research question RQ1, it was found that Expert-Based technique is the mostly used technique in software estimation within the Palestinian software development companies. This corresponds to our hypothesis that *“Expert-based models are the most used models in estimation within Palestinian companies”*. However, we have noticed that expert-based techniques were not conducted in a systematic way. In fact, it has no define process nor checklists to understand how it should be done.

Some companies do rely on their team experience to provide estimation and, in rare cases, a domain expert is available. Additionally, The Expert-based estimations are tight to the resources experience in development. This affects the actual estimation for the task if another resource is assigned to it, in which the real effort estimation is not recorded in isolation of the resources who will do the task. This will cause of inaccurate recorded effort estimation for the use of historical data. Secondly, companies do store their estimations and planning output

using tools like JIRA and Redmine. Nevertheless, the use of those tools by team members or managers to help them in effort estimation using the historical data recorded is very rare. This is because they rely on team experience for estimation which can vary from one resource to another.

These findings correspond to our next hypothesis in which “Accurate effort estimations require clear and detailed processes and guidelines to be considered by estimation team” And “Missing Guidelines and missing use of historical data when applying experts-based models.”. Results of the cases studies shows that few companies do use a set of guidelines or process to do effort estimation. However, it was found that the cases that used guidelines and process did have more accurate estimations than companies did not.

Finally, the need of such process is an actual need for the developers in these companies as stated in their focus groups and interviews. However, companies that do not apply such guidelines are mainly companies working in the local market with diversity of technologies and business domains. Which makes it very hard for them to build up a process for each product, domain or technology.

Challenges and Factors Affecting Software Effort Estimation

In relation to research question RQ2, it has been observed that unclear requirements and inexperienced team members are the main factors affecting the quality and accuracy of effort estimation. Further, it has been observed that estimations were not done by the team as a whole in all development phases. Initial estimations are mostly done by managers or team leaders which introduce a constraint for the rest of the team in the actual planning phase. The team members do estimate features with a constraint in their mind that this feature had already been estimated by the manager or the team leader which causes estimation outputs to be less accurate. Another factor affecting the accuracy of effort estimation is the guidelines that are set by the company for the teams to consider while developing estimations. The more guidelines and clear steps defined the more accurate estimation is provided. Companies for example that only consider number of screens or number of inputs and outputs of a module do have less accurate estimations than companies follow strict guidelines to consider while estimation like number of interfaces, deployments, testing and research tasks.

Finally, it has been observed that adopting agile development methods is also a main challenge facing Palestinian companies. Such transition can affect software effort estimating. This is because team members first need to master the new agile principles and practices.

Effect of Outsourcing as a Business Type on Effort Estimation

In relation to RQ3 it has been found that business type does affect the effort estimation technique used and the accuracy of that estimation. In this study we had observed and interviewed different types of cases that work in different types of software development business. Part of the cases focus on outsourcing model and the other part focus on local software development.

The main difference between both types is that the outsourcing-based companies do follow a process defined by the company they do outsourcing for. Teams do follow a strict process which is in most cases an agile process which allow the requirements to come clearer because of the presence of the domain expert and the product owner. Additionally, the estimation is being done by the whole team members which makes it more accurate as more experienced resources share ideas and experience with less experienced resources. However, in companies developing for local market it was totally different. First there was no defined process for software development Lifecycle, Furthermore, companies develop projects in several domains and using several technologies which makes it a challenge to put a standard process for the team to follow.

Formal Models in Effort Estimation

Finally, and corresponding to the last hypothesis we stated about formal models; we have found that the use of formal models is either missing or is used without calibration. It has been found that companies do not follow any of the formal models defined neither specific tools for estimations. The multiple case study tried to cover multiple sectors within the Palestinian software development industry but still more research is required to understand how formal models are being approached by companies.

Recommendations

In this study we do recommend some enhancements to several approaches in software effort estimations. It has been found that following a process and specific guidelines in software effort estimation do provide more robust and accurate estimations and this has been also approached by (Jorgensen et al., 2009) who found that providing guidelines, checklists and structured group processes can help in achieving even more accurate estimates.

The dependency on resources experience while doing estimation is not a good idea since this will affect the actual estimation of the task. However, a better approach is to try estimating the task before assigning to any of the resources and add more tasks for knowledge transfer and research or proof of concept when the task is assigned to less experienced resource. This will help teams in future querying historical data to better know how much such task was estimated with no relation to the resources that has been assigned the task. Another recommendation is to include nonfunctional requirements while doing effort estimations. This can be approached by guidelines in which deployments, testing and further nonfunctional efforts can be considered for each task.

In general formalizing a non-formal model like expert-based judgment is a good idea and will help developers and team leaders better estimate and provide more accurate estimations (Jorgensen et al, 2009). However, as it has not been found in any the cases any use of formal models, we do believe that combining both formal and non-formal models can be of a better advantage for effort estimation in software development industry (Rijwani et al., 2014).

Threats to Validity

To ensure validity and reliability to our study several actions have been taken and several reliability and validity strategies have been applied while conducting this study. For the construct validity we have used multiple sources of data. A chain of evidence has been established by reviewing the results from observations, interviews and focus groups and comparing them with survey results. Additionally, thematic coding was used to analyze the qualitative data collected, sentences were numbered and linked to their data sources.

For External validity we have developed set of theories based on the literature review. Theories were built upon latest research in the topic. The findings of the case studies did correspond to the theories. The reliability of this study has been empowered by having a case study protocol with a selection of cases that tried to cover multiple industrial cases that we believe provide a good sample of the Palestinian IT sector.

Conclusion and Future Work

Expert-based estimation models are the mostly applied models especially within agile environments. Multiple improvements have been done on expert-based models to formalize the process. Proposing guidelines and checklists to the EE process improves the quality and accuracy of estimation. Additionally, resources experience and requirements clarity are the most affecting factors on the accuracy of software effort estimation. Additionally, improvements on the EE process can be done by checklists and guidelines to better include missing cycles and enhance the accuracy.

Findings of our case study showed that The Palestinian IT sector and especially the software development sector is a young and new sector. Future research on software development lifecycle, estimation using formal models and studies on improving estimations using a combination of both formal and informal models is required to build a better software development industry.

Studies on how to benefit from outsourcing models to enrich the local market software development and enrich the local development companies with experience from outsourcing models are also required. Furthermore, our study shows that the use of agile methodologies is increasing within the Palestinian software development sector. However, there is a need for a comparison research and studies on how local market is approaching agile with companies working with outsourcing business.

Acknowledgements

We would like to thank the managers, team leaders and developers in the four companies who have participated in our case study.

References

- Mansor, Z., Razali, R., Yahaya, J., Yahya, S., & Arshad, N. H. (2016). Issues and challenges of cost management in agile software development projects. *Advanced Science Letters*, 22(8), 1981-1984.
- Bilgaiyan, S., Sagnika, S., Mishra, S., & Das, M. (2017). A Systematic Review on Software Cost Estimation in Agile Software Development. *Journal of Engineering Science & Technology Review*, 10(4).
- Taylor, O., Saini, J., & Rijwani, M. P. (2014). Comparative Analysis of Software Cost and Effort Estimation Methods: A Review. *Interfaces*, 5(7), 10.
- Rijwani, P., Jain, S., & Santani, D. (2014). Software Effort Estimation: A comparison based Perspective. *International Journal of Application or Innovation in Engineering and Management (IJAIEEM)*, 3(12), 18-29.
- Borade, J. G., & Khalkar, V. R. (2013). Software project effort and cost estimation techniques. *International Journal of Advanced Research in Computer Science and Software Engineering*, 3(8).
- Ramacharan, S., & Rao, K. V. (2013). Parametric Models for Effort Estimation for Global Software Development. *Lecture Notes on Software Engineering*, 1(2), 178.
- Colomo-Palacios, R. (Ed.). (2014). *Agile Estimation Techniques and Innovative Approaches to Software Process Improvement*. IGI Global.
- Jacobs, D. (2005). *Accelerating process improvement using agile techniques*. Auerbach Publications.
- Panda, A., Satapathy, S. M., & Rath, S. K. (2015). Empirical validation of neural network models for agile software effort estimation based on story points. *Procedia Computer Science*, 57, 772-781.
- Porru, S., Murgia, A., Demeyer, S., Marchesi, M., & Tonelli, R. (2016, September). Estimating story points from issue reports. In *Proceedings of the The 12th International Conference on Predictive Models and Data Analytics in Software Engineering*(p. 2). ACM.
- Hancock, D. R., & Algozzine, B. (2016). *Doing case study research: A practical guide for beginning researchers*. Teachers College Press.
- Yin, R. K. (2009). *Case study research: Design and methods (applied social research methods)*. London and Singapore: Sage.
- Anwer, F., Aftab, S., Waheed, U., & Muhammad, S. S. (2017). Agile Software Development Models TDD, FDD, DSDM, and Crystal Methods: A Survey. *International Journal of Multidisciplinary Sciences and Engineering*, 8(2), 1-10.
- Minku, L. L., & Yao, X. (2013). Ensembles and locality: Insight on improving software effort estimation. *Information and Software Technology*, 55(8), 1512-1528.
- Jorgensen, M., Boehm, B., & Rifkin, S. (2009). Software development effort estimation: Formal models or expert judgment?. *IEEE software*, 26(2), 14-19.
- Usman, M., Petersen, K., Börstler, J., & Neto, P. (2018). *Developing and Using Checklists to Improve Software Effort Estimation: a Multi-Case Study*.
- Dagnino, A. (2013, May). Estimating software-intensive projects in the absence of historical data. In *Software Engineering (ICSE), 2013 35th International Conference on*(pp. 941-950). IEEE.
- Nguyen-Cong, D., & Tran-Cao, D. (2013, November). A review of effort estimation studies in agile, iterative and incremental software development. In *Computing and Communication Technologies, Research, Innovation, and Vision for the Future (RIVF), 2013 IEEE RIVF International Conference on* (pp. 27-30). IEEE.
- Popli, R., & Chauhan, N. (2014, March). Agile estimation using people and project related factors. In *Computing for Sustainable Global Development (INDIACom), 2014 International Conference on* (pp. 564-569). IEEE.
- Hamouda, A. E. D. (2014, July). Using agile story points as an estimation technique in cmmi organizations. In *2014 Agile Conference (AGILE)* (pp. 16-23). IEEE.
- Lenarduzzi, V., Morasca, S., & Taibi, D. (2014, August). Estimating software development effort based on phases. In *Software Engineering and Advanced Applications (SEAA), 2014 40th EUROMICRO Conference on* (pp. 305-308). IEEE.
- Usman, M., Mendes, E., Weidt, F., & Britto, R. (2014, September). Effort estimation in agile software development: a systematic literature review. In *Proceedings of the 10th International Conference on Predictive Models in Software Engineering* (pp. 82-91). ACM.

- Usman, M., Mendes, E., & Börstler, J. (2015, April). Effort estimation in agile software development: a survey on the state of the practice. In Proceedings of the 19th International Conference on Evaluation and Assessment in Software Engineering (p. 12). ACM.
- Zein, S., Salleh, N., & Grundy, J. (2015, September). Mobile application testing in industrial contexts: an exploratory multiple case-study. In International Conference on Intelligent Software Methodologies, Tools, and Techniques (pp. 30-41). Springer, Cham.
- Usman, M., & Britto, R. (2016, October). Effort estimation in co-located and globally distributed agile software development: A comparative study. In Software Measurement and the International Conference on Software Process and Product Measurement (IWSM-MENSURA), 2016 Joint Conference of the International Workshop on (pp. 219-224). IEEE.
- Tanveer, B. (2017, June). Guidelines for utilizing change impact analysis when estimating effort in agile software development. In Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering (pp. 252-257). ACM.

Author Information

Ahmed Zarour

Birzeit University
Ramallah, Palestine

Contact e-mail: zarourahmed@email.com

Dr. Samer Zein

Birzeit University
Ramallah, Palestine