Effort estimation in agile software development: A survey on the state of the practice

Conference Paper · April 2015

DOI: 10.1145/2745802.2745813

CITATIONS
130

READS
2,241

3 authors:

Muhammad Usman
Blekinge Institute of Technology
38 PUBLICATIONS **1,335** CITATIONS
SEE PROFILE

Emilia Mendes
Aarhus University
292 PUBLICATIONS **9,813** CITATIONS
SEE PROFILE

Jürgen Börstler
Blekinge Institute of Technology
157 PUBLICATIONS **2,570** CITATIONS
SEE PROFILE

Effort Estimation in Agile Software Development: A Survey on the State of the Practice

Muhammad Usman
Blekinge Institute of Technology
371 79, Karlskrona, Sweden
muhammad.usman@bth.se

Emilia Mendes
Blekinge Institute of Technology
371 79, Karlskrona, Sweden
emilia.mendes@bth.se

Jürgen Börstler
Blekinge Institute of Technology
371 79, Karlskrona, Sweden
Jurgen.Borstler@bth.se

ABSTRACT

Context: There are numerous studies on effort estimation in Agile Software Development (ASD) and the state of the art in this area has been recently documented in a Systematic Literature Review (SLR). However, to date there are no studies on the state of the practice in this area, focusing on similar issues to those investigated in the above-mentioned SLR. **Objectives**: The aim of this paper is to report on the state of the practice on effort estimation in ASD, focusing on a wide range of aspects such as the estimation techniques and effort predictors used, to name a few. **Method**: A survey was carried out using as instrument an on-line questionnaire answered by agile practitioners who have experience in effort estimation. **Results**: Data was collected from 60 agile practitioners from 16 different countries, and the main findings are: 1) Planning poker (63%), analogy (47%) and expert judgment (38%) are frequently practiced estimation techniques in ASD; 2) Story points is the most frequently (62%) employed size metric, used solo or in combination with other metrics (e.g., function points); 3) Team's expertise level and prior experience are most commonly used cost drivers; 4) 52% of the respondents believe that their effort estimates on average are under/over estimated by an error of 25% or more; 5) Most agile teams take into account implementation and testing activities during effort estimation; and 6) Estimation is mostly performed at sprint and release planning levels in ASD. **Conclusions**: Estimation techniques that rely on experts' subjective assessment are the ones used the most in ASD, with effort underestimation being the dominant trend. Further, the use of multiple techniques in combination and story points seem to present a positive association with estimation accuracy, and team-related cost drivers are the ones used by most agile teams. Finally, requirements and management related issues are perceived as the main reasons for inaccurate estimates.

Categories and Subject Descriptors

D.2.8 [**Software Engineering**]: Management— *time estimation, cost estimation*

General Terms

Measurement, Management

Keywords

Effort Estimation, Agile Software Development, Empirical Study

EASE'15, April 27-29, 2015, Nanjing, China.

http://dx.doi.org/10.1145/2745802.2745813

1. INTRODUCTION

In Agile Software Development (ASD) planning is carried out iteratively at different levels such as release planning, sprint planning and current day planning [4]. Effective planning needs estimates that are relatively accurate and reliable. Effort estimation in ASD is an active research area wherein different studies have already been reported in the literature. A recent Systematic Literature Review (SLR) [19] aggregated and described the state of the art related to estimation techniques, effort predictors, accuracy measures and agile methods used, gathered from 20 papers reporting 25 studies. Due to its focus on fully refereed empirically-based research, the SLR did not include works that were not peer reviewed, white papers, technical reports, and practitioners' opinion in articles/forums/blogs. As a follow up research, the survey detailed herein gathered evidence on the state of the practice on effort estimation in ASD, in order to complement the findings from the SLR and provide a combined and detailed understanding relating to effort estimation in ASD.

There have been three previous surveys that investigated topics related to estimation in ASD; however they fell short of presenting the state of the practice with respect to the specific estimation techniques and predictors used, accuracy level achieved by different techniques and the development context in which estimation is practiced in ASD. Further description of these surveys and their pitfalls with respect to our study are given in section 2.

The remainder of the paper is organized as follows: Section 2 presents the related work; the research methodology is described in section 3, followed by the presentation of results in section 4. Next, a discussion of the results and the conclusions are given respectively in Sections 5 and 6.

2. RELATED WORK

We identified three surveys investigating topics related to estimation in ASD.

In the first survey, published in 2005 [15], project managers from 18 Norwegian companies were interviewed. Data about 52 projects was used to analyze the differences in schedule and effort overruns between projects using flexible (incremental, agile) and sequential process models. The study concluded that the projects that used a flexible process model experienced less effort overruns when compared to those projects that used sequential models. The data gathered from this survey was also used in a study [16] on different aspects of software estimation (effort and schedule overruns, choice of estimation method, level of estimation skill etc.) in the Norwegian industry. However, this study made no distinction between projects/companies using agile or traditional development processes.

The second survey [14] investigated the relationship between customer collaboration and effort overruns. It is based on interviews with project managers from a medium sized Norwegian company employing an agile methodology. The data covered 18 different projects. This survey concluded that a lesser magnitude of effort overruns is experienced whenever customer collaboration is supported via daily communication with the customer. The third survey [12] explored the concept of "expected implementation time" of a user story as a granularity characteristic. Results showed that developers face more problems, e.g., estimation related issues, when they dealt with coarse grained user stories. None of these surveys reported the state of the practice with respect to the specific estimation techniques and predictors used, accuracy of the used techniques and the development context in which estimation is practiced specifically in ASD. To achieve this aim, we conducted this empirical study that is based on practitioners from any country, from software companies practicing agile methods or practices.

3. RESEARCH METHODOLOGY

In this section we describe research questions, study type, design, execution and validity threats.

3.1 Research Questions

Five research questions are investigated in this survey:

- **RQ1**: Which effort estimation techniques, and in which combinations, are used in practice in ASD?
- **RQ2**: Which effort predictors (size metrics, cost drivers), and in which combinations, are used in practice for effort estimation in ASD?
- **RQ3**: What is the agile development context in which these techniques are used in practice?
- **RQ4**: How accurate are the effort estimates in ASD?
- **RQ5**: What are the reasons for inaccurate estimates in ASD?

3.2 Study Type, Design and Execution

Our goal was to gather retrospective data from as large as possible a sample of practitioners who have been involved in effort estimation in ASD. Therefore a survey investigation was carried out [5, 11, 18]. An on line web-based questionnaire, hosted at `surveymonkey.com`, was used as instrument for data collection as it offers a low cost and efficient mechanism to gather data from respondents [17, 18]. The questionnaire was accessible between August and October of 2014. The software practitioners who have been involved in effort estimation in ASD represented the survey population. The survey participants were recruited initially through personal contacts and via posting the survey link to the agile and measurement-related forums at LinkedIn. A snowball sampling approach [18] was also used via our initial industry contacts. Finally, the questionnaire's URL and goal were advertised by one of the authors at three conferences (PROMISE 2014, RE 2014, ICGSE 2014). Convenience sampling was used [18].

A total of 82 respondents started the questionnaire, whereof 60 (73%) completed all mandatory questions. Therefore, we only used data from these 60 respondents. The questionnaire was organized into two parts: part 1 gathered demographic data, while part 2 collected data on the respondents' practical experience in effort estimation in ASD. The questions from part 2 correspond in the following way to the RQs described in Subsection 3.1:

- **Survey Question(s) for RQ1**: Which of the following effort estimation technique(s) are used in your company to estimate effort? Respondents could select multiple items from a list of possible techniques. In addition, a text field was provided for further answers.
- **Survey Question(s) for RQ2**: (1) Which of the following size metric(s) are used in your company in estimating the effort? A list of metrics plus a text field were provided for answering. (2) Which of the following factors do you believe are fundamental to consider when estimating the effort? A list of factors (cost drivers) plus a text field were provided for answering.
- **Survey Question(s) for RQ3**: (1) What are the agile development activities in your company for which effort is estimated? A list of development activities plus a text field were provided for answering. (2) What are the most common situations when effort is estimated for agile projects in your company? A list of planning stages/levels plus a text field were provided for answering.
- **Survey Question(s) for RQ4**: In your experience the effort estimates for a sprint/iteration, when compared to the actual effort, are on average: 1) Spot on (0–5%), 2): Under-estimated by (5–25%), 3): Over-estimated by (5–25%), 4): Under-estimated by (25–50%), 5): Over-estimated by (25–50%), 6): Under-estimated by 50% or more, and 7): Over-estimated by 50% or more. The seven point ordinal scale was used for answering this question. Participants had to select exactly one option. No additional text field was provided.
- **Survey Question(s) for RQ5**: If estimates are on average within or above 35% from actuals, what do you think are the possible reasons for such difference between actual and estimated effort? Only a free text field was provided for answering.

3.3 Validity Threats

This section discusses validity threats using the four types of threats suggested in [10, 21].

- **Construct validity** is concerned with issues that may arise due to improper design of the survey instrument, which then may not be measuring properly what it is supposed to measure. We believe that this issue has been mitigated, since the questionnaire was iteratively designed and updated by the authors based on the results from the SLR [19]. Further, another two researchers with experience in ASD validated the questionnaire regarding its completeness and readability.

- **Internal validity** is concerned with issues, such as confounding factors or irrelevant respondents, which could introduce a systematic error or bias in the study results. A number of steps were taken to mitigate this threat: (i) It was clearly stated in the survey introduction that only practitioners with actual experience in effort estimation in ASD should participate. In addition, besides asking respondents about their experience in the software industry, we also asked them about their experience in ASD to ensure that all respondents were also agile practitioners. All respondents (60) answered this question; (ii) Respondents were assured of their anonymity to avoid evaluation apprehension; and (iii) We matched the respondents' countries, as provided in the survey, against the IP addresses collected by the survey tool, and found no mismatch.

- **External validity**, also called generalizability, refers to the extent to which findings in a study are applicable outside of the study context. Since the sample used herein was a convenience sample, our results are only generalizable to those agile teams and companies that share similar characteristics to our survey respondents' teams and companies. However, a number of steps, such as advertising our survey on ASD-related on-line forums and conferences and a snowball approach, were taken to obtain a broad sample representative of our population of interest. We believe that these steps contributed to obtaining a sample that is quite heterogeneous in terms of experience, job role and country.

- **Conclusion Validity** is related with the possibility of reaching incorrect conclusions about association in observations due to errors such as use of inadequate statistical tests or measures. In this study, we only used frequencies and percentages to identify common patterns or practices to point out potential areas or relationships for future research efforts. In addition, we only considered complete responses in our analysis.

4. RESULTS

This section presents the results of our survey organized by research questions.

4.1 Respondents' Demographics

To understand the respondents' context and background, several demographic questions were asked.

4.1.1 Country

Respondents from 5 continents and 16 countries participated in the survey (see Table 1). Most responses were from Brazil and the USA. However, in relation to the number of respondents per continent, the majority came from Europe.

Table 1: Respondents by continents and countries

Continent	Countries	F (%)
Europe	Sweden(4), UK(2), Italy(2) NL(2), Denmark(2), France(2) Latvia(1), Ireland(1), Estonia(1)	17 (28%)
S. America	Brazi(15), Argentina(2)	17 (28%)
N. America	USA(13), Canada(1)	14 (23%)
Asia	Pakistan(4), India(2)	6 (10%)
Australia	Australia(6)	6 (10%)

4.1.2 Job Role

Figure 1 presents the respondents' job roles, along with the corresponding frequencies and percentages. Results suggest that a variety of roles, both managerial and non-managerial, participate in effort estimation activities in agile projects.

- **Managerial roles**: Managerial roles constitute 50% of the total respondents and include project managers, program managers, product owners, scrum masters and team leads.

- **Non-Managerial roles**: Non managerial roles constitute around 32% of the total respondents and include developers, testers, analysts and designers.

- **Other roles**: A total of 11 (18%) respondents described other roles in the free text field provided for this questions, such as estimation expert, agile coach, project coach, consultant, and business analyst.

Estimation and planning are done at different levels in ASD, e.g., release planning, sprint/iteration planning and daily planning [4], which demands the involvement of people both in managerial and non-managerial positions at different project stages.

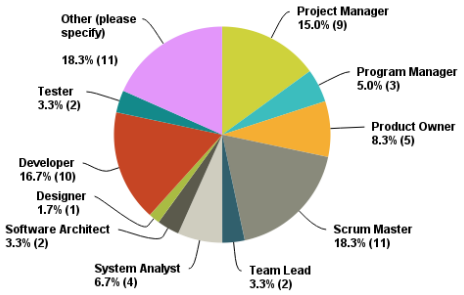

Figure 1: Job roles of the respondents

4.1.3 Agile Experience

Figure 2 depicts the information about respondents' experience in ASD along with the corresponding frequencies and percentages. A total of 62% of the respondents have more than three years of experience in ASD, while only 10% have less than 1 year of ASD experience. We also asked the respondents to provide information regarding their total professional experience, agile or otherwise, in the software industry. The majority of the respondents (about 82%) have more than 5 years of experience in the software industry, suggesting that most of the survey respondents are senior

industry practitioners. The high number increases our confidence in the validity of the responses and also highlights the practice of involvement of seniors in the estimation process in ASD.

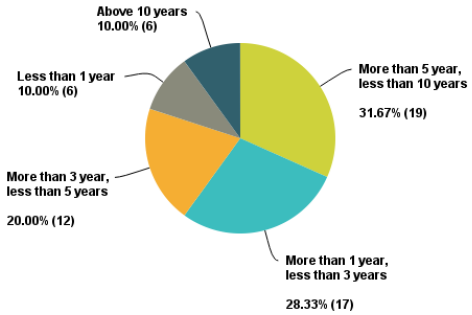

Figure 2: Experience in ASD

4.1.4 *Team Size, System Types and Agile Methods*

Most respondents work in teams of up to 10 team members; 38% reported a team size of 1–5, and 40% a team size of 6–10. Only few reported a team size greater than 20. Most of the respondents' companies develop business applications (62%). Some companies also develop E-commerce (27%), data processing (22%) and financial (20%) applications, few develop safety critical (7%) and embedded systems (13%). Scrum, Kanban and eXtreme Programming are the most commonly used agile methods, while DSDM, Crystal and FDD are used by only a handful of companies.

4.2 RQ1: Estimation Techniques

Figure 3 details the responses on estimation techniques practiced in ASD along with the corresponding frequencies and percentages. In our sample, planning poker (63%) is the most frequently practiced effort estimation technique in ASD, followed by estimation by analogy (46%) and expert judgment (38%). All of the techniques that are selected by more than 10% of the participants share one common aspect: these techniques use subjective assessment of experts in some way to arrive at an effort estimate. These subjective assessment based techniques (planning poker, analogy, expert judgment) seem to align well with the agile philosophy, where much of the knowledge is managed tacitly and more emphasis is placed on people and their interaction. Other techniques reported in the text field include SLIM (2) and methods based on IFPUG (1) and COSMIC function points (1).

To identify the combinations in which effort estimation techniques are practiced in ASD, respondents were allowed to select multiple techniques. 52% (31) of the respondents selected multiple effort estimation techniques, while 48% (29) selected a single technique. Planning Poker was selected by 38 (63%) agile practitioners in total, out of which it is practiced alone by 18 (47%) and in combination with other techniques by 20 (53%). Planning poker was reported as being practiced in combination with following techniques:

- Estimation by analogy (6 agile practitioners)
- Estimation by analogy and expert judgment (5 agile practitioners)

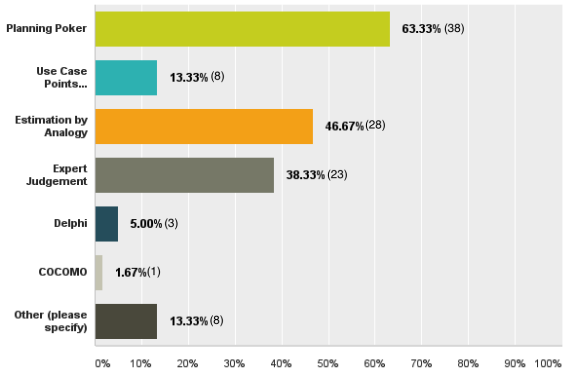

Figure 3: Effort Estimation Techniques

- Estimation by analogy and Delphi (2 agile practitioners)
- Estimation by analogy and COSMIC function points (1 agile practitioner)
- Estimation by analogy, expert judgment, use case points method and SLIM (1 agile practitioner)
- Expert judgment (3 agile practitioners)
- Use case points method (2 agile practitioners)

It can be inferred from the above data that when planning poker is practiced in combination with other techniques, estimation by analogy and expert judgment are the preferred choices for agile practitioners. Besides planning poker, other techniques are also practiced in combination, e.g., estimation by analogy and expert judgment (selected by 9 practitioners). It is interesting to note that estimation by analogy alone was selected by a single agile practitioner only, while expert judgment by 5 agile practitioners. The results suggest that these techniques (i.e. estimation by analogy and expert judgment) are mostly used in combination with other techniques in ASD. On the other hand, Planning poker is used both solo and in combination with other estimation techniques in ASD.

4.3 RQ2: Effort Predictors

Size metrics and cost drivers constitute effort predictors, which are described in the following subsections.

4.3.1 *Size Metrics*

61% (37) of the respondents in this survey selected story points as the size metric (see Figure 4). This finding aligns well with the results on the estimation techniques presented above, wherein planning poker was found to be the most frequently practiced estimation technique in the ASD context. Other size metrics are also used to some degree – function points (17%) and use case points (10%). Respondents did not report any additional size metrics in the text field provided for this question.

Based on the respondents' data, we identified some patterns in the application of size metrics in ASD during the effort estimation process. According to 68% (25 out of 37) of the respondents story points are used solo, while 32% (12 out of 37) selected story points in combination with the following metrics:

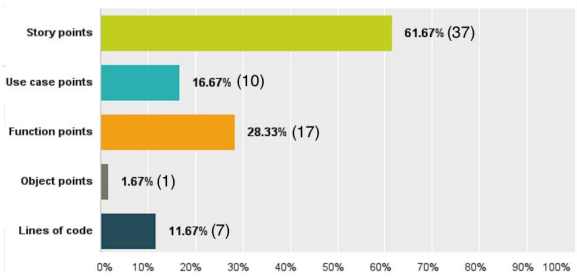

Figure 4: Size Metrics

- Function points (6 agile practitioners)
- Function points and use case points (2 agile practitioners)
- Function points and LOC (1 agile practitioner)
- Function points, use case points and lines of code (LOC) (1 agile practitioner)
- Use case points (2 agile practitioners)

Respondents' data suggest that other size metrics are rarely used solo; function points are used solo according to 4 (out of 17) respondents only and LOC is never used solo. Our data suggest that most agile practitioners consider story points as the most preferred choice when measuring size for ASD projects. Story points are mostly used solo but when used in combination with other size metrics, they are mostly combined with function points and/or use case points.

4.3.2 *Cost Drivers*

Figure 5 details the responses in terms of frequencies and percentages for the different cost drivers. Team's expertise level and team's prior experience are considered to be the most important cost drivers by most agile practitioners in our sample. Task size, project domain and non-functional requirements are also given considerable importance by agile practitioners. In the text field, respondents mentioned cost drivers such as complexity, team stability, integration issues and team velocity. It was interesting to observe that 10 respondents did not select any cost driver, however they selected size metric(s) as follows: story points (5), story and function points (2), function points (2) and use case points (1).

Results also show that most of the time a combination of cost drivers are used to arrive at an effort estimate. The two most highly used cost drivers – team's expertise level and team's prior experience, are rarely used solo according to our sample data. Both were selected as solo cost drivers by only one respondent each, but together in combination by 36 respondents. Other cost drivers that are also, at times, used in combination with these top two cost drivers are task size, project domain and non-functional requirements. Both of the most frequently used cost drivers are related to teams, which resonates well with agile philosophy, wherein individuals and teams are highly valued. We believe that agile managers should always take into account these team related cost drivers when estimating the effort in ASD.

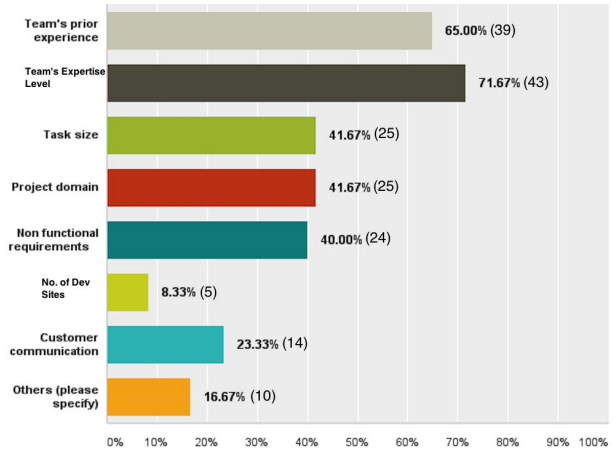

Figure 5: Cost Drivers

4.4 RQ3: Estimation and Agile Development Context

Through this research question we wanted to understand the agile development context in which the process of effort estimation is carried out in ASD. Two aspects related to the ASD context are considered in this question: Planning levels (release, sprint or current day planning) and development activities (i.e. analysis, design, implementation, testing or maintenance).

4.4.1 *Effort Estimation and Agile Planning Levels*

Respondents were asked to select the planning levels or stages at which they perform the effort estimation process. Figure 6 details the responses against each planning level along with the frequency and percentage. A total of 82% (49) of the respondents said that effort estimation is carried out during sprint/iteration planning. Release planning was selected by about 42% (25) respondents. Our results suggest that estimation plays a pivotal role during sprint and release planning for most agile teams. It is interesting to note that a relatively a small percentage of respondents selected daily planning (about 17%) and project bidding (about 17%), indicating that at least for our sample, few teams estimate effort in the daily planning or project bidding stage. Five respondents provided answers in the text field: feasibility study, weekly task planning and product road map planning.

After analyzing the data for estimation versus planning levels at a broader level, we next present the results at the individual technique level. This is done to highlight how specific techniques are used by agile teams at different planning levels. We only consider planning poker, estimation by analogy and expert judgment for this discussion as they are the ones that are used most frequently, see Section 4.2 above. Table 2 presents the planning levels with different estimation techniques along with the frequency and percentage for each case. The trends for each technique are in line with the overall trend described in the preceding paragraph, i.e. most agile teams, using any of the estimation techniques listed in Table 2, estimate effort at the sprint planning level, or at both spring and release planning levels. The entries in boldface in Table 2 highlight these trends. Since it is difficult

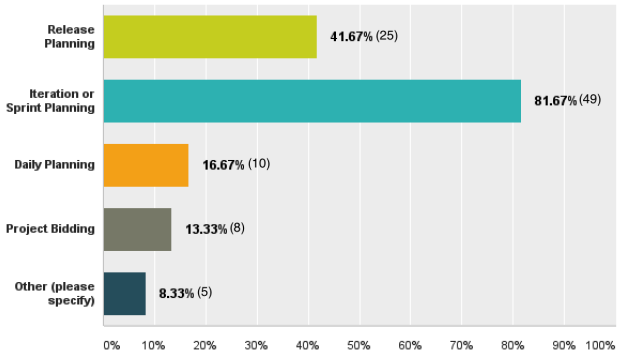

Figure 6: Estimation at agile planning levels

to estimate in absence of detailed information at the project bidding stage only few respondents (8) selected bidding, indicating that few agile teams are able to estimate effort at the bidding or start up phase. In 7 out of these 8 cases, expert judgment, in combination with some other technique, is used to estimate effort at the bidding level.

Table 2: Planning levels versus estimation techniques

Planning Level	PP	Analogy	EJ
Bidding (B)	0(0%)	0(0%)	0(0%)
Release (R)	2(5%)	3(11%)	1(5%)
Sprint (S)	**17(45%)**	**7(25%)**	**6(27%)**
Daily (D)	0(0%)	1(4%)	2(9%)
R & S	**10(26%)**	**11(39%)**	**5(23%)**
R,**S** & D	2(5%)	1(4%)	1(5%)
S & D	2(5%)	0(0%)	0(0%)
S & B	3(8%)	2(7%)	4(18%)
R& D	1(3%)	0(0%)	0(0%)
B, R & **S**	0 (0%)	2(7%)	2(9%)
All	1(3%)	1(4%)	1(5%)
Total	**38**	**28**	**22**

PP=Planning Poker, EJ=Expert Judgment

4.4.2 *Effort Estimation and Agile Development Activities*

Software projects progress through different development activities (analysis, design, implementation etc.) that all contribute to the effort required to complete a project. Respondents were asked to select activities for which effort is estimated in their organizations. Figure 7 details the responses along with the frequencies and percentages of responses against each development activity. It is clear from the figure that most agile teams estimate effort for implementation (about 87%) and testing (about 68%) activities. Two thirds (40 of 60) of the respondents selected implementation and testing activities together in combination. Since test driven development is a common practice in ASD many agile teams consider both implementation and testing when doing effort estimation. Analysis and design activities are selected by a relatively small number of respondents, i.e. 26 (about 43%) and 32 (about 53%), respectively. On the other hand, also maintenance was selected by few respondents (about 23%). This might indicate that only few agile teams take into account maintenance activities when they estimate effort on an agile development project.

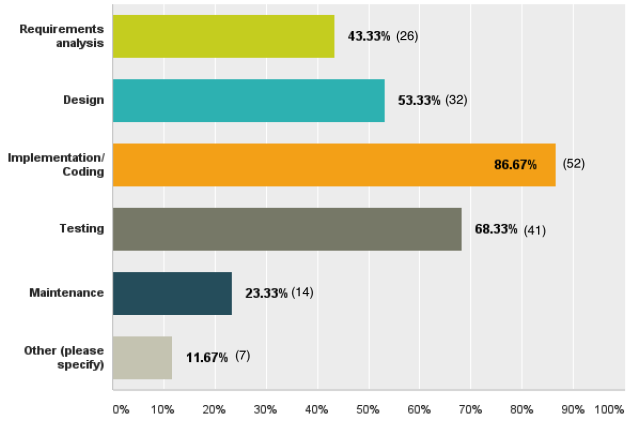

Figure 7: Estimation of development activities

Next, Table 3 shows, for the most frequently used estimation techniques, the development activities and their combinations for which effort is estimated according to our survey respondents. Results show that, irrespective of the estimation technique, most of the agile teams take into account multiple activities to estimate effort. Implementation (I) and testing (T) are the leading activities that are mostly considered for estimating the effort, irrespective of the estimation technique. The majority of the respondents who selected implementation also selected the testing activity. This trend is consistent across all techniques, i.e. planning poker (78%), analogy (77%) and expert judgment (80%). The pattern I&T appears frequently and is emphasized in Table 3 (in boldface). There are only two respondents with all three estimation techniques that did not select the implementation activity. On the other hand, the testing activity was never selected solo. Our results suggest that irrespective of the effort estimation technique, a good number of agile teams do not take into account analysis, design and maintenance activities during the effort estimation process.

4.5 RQ4: Estimation Accuracy

RQ4 aims to identify respondents' beliefs regarding the amount of estimation error (actual effort – estimated effort), based on their experiences in ASD. Figure 8 presents the results arranged by the residual or error range. Note that the same residual range was also employed in the survey instrument. Close to 22% (13) of the respondents believe that the estimates in their company or team are spot on (i.e. 0–5% error). Data suggests that the tendency to underestimate effort is higher than the tendency to overestimate, which supports some of the existing body of evidence in software effort estimation [8]. According to 35% (21) of the respondents, effort estimates are on average underestimated by 25% or more in their company, while 7% (4) reported underestimation by 50% or more. The tendency to underestimate development effort could be attributed to overoptimism and team members' inexperience in ASD, as inexperienced de-

Table 3: Development activities versus estimation techniques

Development Activity	PP	Analogy	EJ
Analysis (A) only	1	1	1
Design (D) only	1	0	0
Implementation (I) only	**2**	**3**	**1**
Testing (T) only	0	0	0
Maintenance (M) only	0	0	0
All (A&D&**I&T**&M)	**5**	**4**	**5**
A&**I** only	1	1	1
D&**I** only	2	1	2
I&T only	**8**	**4**	**4**
I&M only	1	0	0
A&D&**I** only	2	2	0
D&***I&T*** only	2	1	1
I&T&M only	2	0	0
A&***I&T*** only	1	1	0
A&D&***I&T*** only	**5**	**6**	**4**
D&***I&T***&M only	**4**	**3**	**3**
D&***I&T***&Management only	1	1	0
Total	**38**	**28**	**22**

PP=Planning Poker, EJ=Expert Judgment

velopers may exhibit the tendency to consider the best case scenario only.

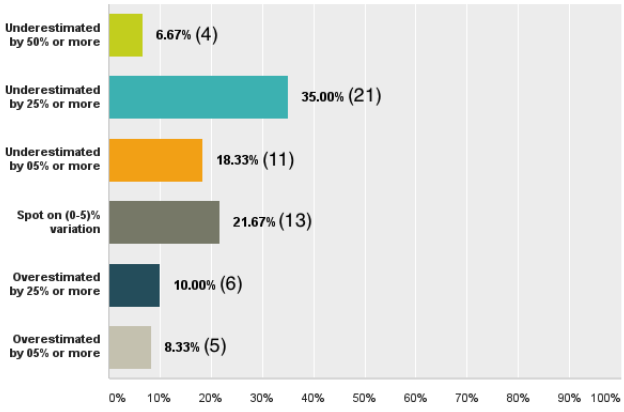

Figure 8: Accuracy of estimates in ASD

Table 4 provides an aggregated view of the estimation error or residual and shows that, according to 52% of the respondents, effort estimates in ASD are out of range by factor of 25% or more.

Our results suggest that the effort estimates of around half of the agile teams are inaccurate by a factor of 25% or more. It seems worthwhile to investigate further the respondents' estimation accuracy responses with respect to the estimation techniques and effort predictors used to check whether there is an association between the choice of estimation techniques and predictors and accuracy levels. Table 5 presents a detailed view of the different estimation error ranges against three different factors: estimation techniques, size metrics and cost drivers. The estimation error ranges are divided into two broad groups: error range of 0–25% (both over and under) and 25% and above. The first group represents an acceptable accuracy level, while the second one represents a potential concern for the agile teams. In the following, we discuss estimation error ranges in the light of these three factors.

Table 4: Overall estimation residuals in ASD

Range	Underestimated	Overestimated	Overall
0–5%	—	—	22%
5–25%	18%	8%	26%
25–50%	35%	10%	**45%**
50+%	7%	0	**7%**

- **Estimation error vs estimation techniques**: When we look at the percentages in the techniques columns in Table 5, we can see that for the high accuracy group (0–25% error range), multiple estimation techniques were selected about twice as often (66%) as single techniques (34%). For the low accuracy group (25% and above error range) this is the opposite; 61% of the respondents selected a single technique, while 39% selected multiple technique. Our data therefore suggests that the use of a solo or combination (ensemble) of estimation techniques may have an effect upon estimation accuracy. However, further investigation is needed before any conclusions or generalizations can be made. A similar result was reported in the domain of web effort estimation [1], where it was observed that ensembles of estimation techniques consistently performed better than solo techniques.

- **Estimation error vs size metrics**: Regarding the possible impact of the choice of size metric on the estimation error, we can observe the following: (1) A total of 76% of the respondents in the high accuracy group (0–25% error range) selected story points as the size metric, whereas 48% in the low accuracy group. (2) For size metrics other than story points, the distribution is the opposite; 31% of the respondents in the high accuracy group selected another size metric, while 42% of the respondents in the low accuracy group selected another size metric. Combining point 1 and 2, it can be said that story points are a preferred choice for most of the respondents in the high accuracy group only. (3) Regarding multiple size metrics, 28% of the respondents in the high accuracy group selected multiple metrics, while 19% of respondents in low accuracy group selected multiple size metrics.

  Empirical evidence shows that size metrics correlate with effort, thus suggesting their use as effort predictors. However, to our surprise, some respondents did not select any size metric indicating the possibility of arriving at an effort estimate without using any size metric. Only 14% of the respondents in the high accuracy group did not select any size metric, while 26% of the respondents in the low accuracy group did not select any size metric. Such results suggest that not considering the size of the system to be developed during effort estimation could have a detrimental effect upon the accuracy of the estimates. Combining these points suggests that in the ASD context story points seem to be related with better accuracy, while other size met-

rics are also employed but only in combination with story points. Lastly it may not be advisable to completely ignore system size during effort estimation.

- **Estimation error vs cost drivers**: There is not much difference in the usage of cost drivers for both groups, i.e. respondents in both high and lower accuracy groups mostly selected multiple cost drivers. Most of the agile teams use multiple cost drivers during the effort estimation process.

4.6 RQ5: Reasons for Inaccurate Estimates in ASD

A total of 28 respondents provided feedback for this open ended question, discussing a range of possible reasons for inaccurate estimates in ASD. These reasons were organized into seven categories, which are presented below and summarized in Table 6.

- The category representing requirements-related reasons received most contributions (9 respondents). The following specific reasons were suggested: poorly specified user stories, missing requirements, overlooking the non-functional requirements (NFR) and requirements change. Requirements are used as input to estimate size, which is then employed as input to the estimation process. The accuracy of estimates can be improved whenever requirements and user stories are well understood, and when NFRs are taken into account. The survey results support previous evidence where both changing and new requirements were found to be one of the top reasons for effort overruns (e.g., [7]).

- The second category represents project management-related issues (7 respondents) including the following reasons for inaccurate estimates: poor change control, scrum master not guiding the team, scope creep and high employee turn over. When such issues are not managed properly, they may have a negative impact on the development time and project cost, and therefore increase the risk for estimation errors.

- Team-related issues (5 respondents) represent the third category. Respondents provided the following specific reasons for inaccurate estimates: insufficient experience of the development team, knowledge sharing and management problems in the team and the presence of unskilled members in the team. The lack of team expertise and experience were also found to be the reasons for inaccurate estimates in ASD in [9]. It is important that these issues are considered when estimating effort and managed during project execution to mitigate issues that may have a detrimental effect upon estimation accuracy.

- Overoptimism (3 respondents) comprises two specific reasons. The first reason was considering only the best case scenario and purposefully underestimating the effort to obtain a contract. It is important to be mindful of the tendency of developers to underestimate the effort by only considering the best case scenario. The overoptimism trend in the ASD context is also observed in other studies, such as [3, 13, 12]. The second reason regarding the purposeful underestimation is an unfair practice and is a clear breach of the code of ethics for software engineers described in [6].

- Lack of a formal estimation process was reported by 3 respondents. Such results are in line with the findings from another study in agile cost estimation [9]. In the ASD context people and their interactions are valued more than processes and tools, but this does not mean that agile practices advocate for the estimation process dimension to be completely ignored. We suggest that agile managers should ensure that a process, that is aligned with the agile philosophy, is in place for effort estimation.

- The sixth category represents ignoring the testing effort (2 respondents). Effort is underestimated when developers doing the estimation mainly consider implementation effort, and ignore the testing effort. It is important for the agile managers to ensure that estimators are considering all development activities to obtain a reliable effort estimate. One way for ensuring this is to include people from the testing teams in the estimation sessions.

- The seventh and last category (2 respondents), represents the lack of customer involvement during story sizing sessions. User stories often lack the required details making it challenging to arrive at a reliable size estimate. In such situations, the presence of customer helps resolve ambiguities. Lack of customer collaboration and communication has also been associated with effort overruns in the ASD context elsewhere [14]. Most of the studies included in a systematic review by Grimstad et al. [7] reported that customer characteristics, such as involvement and commitment, are considered important for estimation accuracy.

5. DISCUSSION

This section provides a detailed discussion within the context of our research questions, which correspond to the following topics:

- Effort estimation techniques practiced in ASD
- Effort predictors (size metrics and cost drivers) practiced in ASD
- Agile contexts in which estimation is practiced in ASD
- Accuracy of effort estimates in ASD.

Our results show that planning poker, estimation by analogy and expert judgment are the techniques that are most frequently practiced in ASD. In ASD much of the knowledge is managed tacitly with higher emphasis on people and their interactions [2]. This probably is the reason due to which most frequently used estimation techniques in ASD, as per our survey respondents, are the ones that rely on subjective assessment of the experts to arrive at an estimate. These findings are in line with the results of an SLR [19] on effort estimation in ASD. A total of 52% of the respondents use a combination of estimation techniques. Results also show that planning poker is the only technique that is practiced both in combination with other techniques and solo also; it is combined either with estimation by analogy or with expert judgment. As for other estimation techniques, respondents rarely used them solo, which indicates that these techniques are not considered comprehensive on their own by most of

Table 5: Estimation error versus techniques and predictors

Estimation Error	Techniques		Size Metrics				Cost Drivers		
	Muliple	Single	SPs	Other	Muliple	No	Multiple	Single	No
Spot on (0–5%) (13)	9	4	9	3	2	3	11	0	2
U by 5% or more (11)	6	5	10	6	6	0	9	0	2
O by 5% or more (5)	4	1	3	1	0	1	4	1	0
Error (0–25%) (29)	**19** **66%**	**10** **34%**	**22** **76%**	**9** **31%**	**8** **28%**	**4** **14%**	**23** **79%**	**1** **3%**	**4** **14%**
U by 25% or more (21)	10	11	11	9	4	4	16	2	3
O by 25% or more (6)	1	5	2	2	0	2	4	1	1
U by 50% or more (4)	1	3	2	2	2	2	1	1	2
Error (25+%) (31)	**12** **39%**	**19** **61%**	**15** **48%**	**13** **42%**	**6** **19%**	**8** **26%**	**21** **68%**	**4** **13%**	**5** **16%**

SPs=Story points

Table 6: Reasons for inaccurate estimates in ASD

#	Category of reasons for inaccurate estimates	F(%)
1	Requirements' related issues (Missing and changing Reqs., Poor user stories, Overlooking NFRs etc.)	9 (32%)
2	Project management issues (Poor change control, SM not guiding the team, Scope creep)	7 (25%)
3	Team related issues (Inexperience, knowledge sharing problem in team, Unskilled team members)	5 (18%)
4	Overoptimism (Best case scenario, Purposely underestimating to obtain work)	3 (10%)
5	Lack of formal estimation process	3 (10%)
6	Ignoring testing effort	2 (7%)
7	Insufficient customer involvement during story sizing	2 (7%)

the respondents. Note that none of the primary studies included in the SLR [19] investigated the application of effort estimation techniques in combination.

According to our survey results, story points are most frequently used size metrics by agile teams followed by function points (FPs) and use case points (UCPs). These findings also corroborate those from the SLR [19]. Story points are used solo by most agile teams (68%), while in some cases they are used in combination with other metrics such as FPs/UCPs. Other size metrics, e.g., FPs or UCPs, are rarely used solo in ASD, as per our respondents. Our results suggest that planning poker and story points are considered a reliable combination by most agile teams. Other estimation techniques and metrics, when used, act in aid of this combination.

Team related cost drivers, i.e. team's expertise level and prior experience, are the ones that are considered to be the most important by the respondents. This finding is in line with the results of the SLR [19], where these team related cost drivers were the ones that are identified in most of the primary studies. This corresponds well with the agile philosophy wherein the team members are more important than tools or processes. However we also identified some cost drivers in the survey presented here, e.g. project domain and non functional requirements, that are considered important by several respondents (about 40%); these findings are contrary to those from the SLR [19]. Results also show that multiple cost drivers are used by most agile teams.

We now discuss the agile context, in terms of agile planning levels and development activities, in which estimation is practiced in ASD. Effort estimation is carried out mostly at sprint planning (82% of the respondents) and release planning (42%) levels. Few respondents selected daily planning and project bidding and feasibility study stages as well where estimation is carried out. These findings also corroborate those from the SLR [19]. It is also identified in this survey that most agile teams, irrespective of the estimation technique and predictors used, take into account implementation and testing activities for effort estimation; and relatively fewer teams consider activities such as analysis and design as well. This may be one of the reasons of the under estimation trend in ASD, as discussed in section 4.5.

According to 52% of the respondents effort estimates in ASD are out by an estimation error of 25% and above, wherein the dominating trend is of under estimation. Estimation error is on the lower side in the cases when respondents' teams are practicing multiple estimation techniques.There appears to be an association between use of story points and low estimation error i.e. the majority of the respondents (76%) in the high accuracy group selected the usage of story points as size metrics, while 48% of respondents in low accuracy group selected story points. These finding regarding the associations between estimation techniques or size metrics with accuracy of estimates needs further investigation before we can make any generalizations. Most respondents cited requirements related issues as one of the main reason for inaccuracy of effort estimates in ASD. Incomplete user stories and overlooking the non-functional requirements (NFRs) can lead to underestimating the total effort. NFRs consume considerable design and implementation effort; it is important to incorporate their impact in size and effort estimates otherwise it is possible that team's productivity may turn out to be much lower than expected because they are required to meet the demanding performance or security or other NFRs [20, chapter 22].

6. CONCLUSIONS

The paper presented the results of an empirical study conducted to understand state of the practice on effort estimation in ASD. The study was performed as a survey employing

an on line questionnaire as instrument to collect data from agile practitioners, across different companies and countries, who have some experience in effort estimation. The results indicate that most agile teams use estimation techniques that rely on experts' subjective assessment to arrive at an estimate e.g. planning poker, analogy and expert judgment. It was also observed that the usage of a combination of techniques is related with relatively accurate estimates. Story points are the most frequently practiced size metric while other metrics, such as function points or LOC, are rarely used. Regarding the cost drivers it was found that most of the agile teams use team related cost drivers (e.g., team expertise level, team's prior knowledge) when estimating the effort. According to 52% of the respondents effort estimates in agile are over/under estimated by a factor 25% or more, wherein dominant trend is that of under estimation. Requirements, management and team related issues are cited as the main reasons by some agile practitioners for the larger difference between actual and estimated effort.

Acknowledgments

This work is partially funded by the Swedish Knowledge Foundation research project number 20130174.

7. REFERENCES

[1] D. Azhar, P. Riddle, E. Mendes, N. Mittas, and L. Angelis. Using ensembles for web effort estimation. In *Empirical Software Engineering and Measurement, 2013 ACM/IEEE International Symposium on*, pages 173–182. IEEE, 2013.

[2] F. O. Bjørnson and T. Dingsøyr. Knowledge management in software engineering: A systematic review of studied concepts, findings and research methods used. *Information and Software Technology*, 50(11):1055–1068, 2008.

[3] L. Cao. Estimating agile software project effort: an empirical study. 2008.

[4] M. Cohn. *Agile estimating and planning*. Pearson Education, 2005.

[5] A. Fink. *The survey handbook*, volume 1. Sage, 2003.

[6] D. Gotterbarn, K. Miller, and S. Rogerson. Computer society and acm approve software engineering code of ethics. *Computer*, 32(10):84–88, 1999.

[7] S. Grimstad, M. Jorgensen, and K. Molokken-Ostvold. The clients' impact on effort estimation accuracy in software development projects. In *Software Metrics, 2005. 11th IEEE International Symposium*, pages 10–pp. IEEE, 2005.

[8] M. Jorgensen and K. Molokken-Ostvold. Reasons for software effort estimation error: impact of respondent role, information collection approach, and data analysis method. *Software Engineering, IEEE Transactions on*, 30(12):993–1007, 2004.

[9] S. Keaveney and K. Conboy. Cost estimation in agile development projects. 2006.

[10] S. Keele. Guidelines for performing systematic literature reviews in software engineering. Technical report, Technical report, EBSE Technical Report EBSE-2007-01, 2007.

[11] B. Kitchenham and S. L. Pfleeger. Principles of survey research parts 1-6. *ACM SIGSOFT Software Engineering Notes*, Nov 2001 to Mar 2003.

[12] O. Liskin, R. Pham, S. Kiesling, and K. Schneider. Why we need a granularity concept for user stories. In *Agile Processes in Software Engineering and Extreme Programming*, pages 110–125. Springer, 2014.

[13] V. Mahnič and T. Hovelja. On using planning poker for estimating user stories. *Journal of Systems and Software*, 85(9):2086–2095, 2012.

[14] K. Molokken-Ostvold and K. M. Furulund. The relationship between customer collaboration and software project overruns. In *Agile Conference (AGILE), 2007*, pages 72–83. IEEE, 2007.

[15] K. Molokken-Ostvold and M. Jorgensen. A comparison of software project overruns-flexible versus sequential development models. *Software Engineering, IEEE Transactions on*, 31(9):754–766, 2005.

[16] K. Moløkken-Østvold, M. Jørgensen, S. S. Tanilkan, H. Gallis, A. C. Lien, and S. Hove. A survey on software estimation in the norwegian industry. In *Software Metrics, 2004. Proceedings. 10th International Symposium on*, pages 208–219. IEEE, 2004.

[17] T. Punter, M. Ciolkowski, B. Freimut, and I. John. Conducting on-line surveys in software engineering. In *Empirical Software Engineering, 2003. ISESE 2003. Proceedings. 2003 International Symposium on*, pages 80–88. IEEE, 2003.

[18] C. Robson. *Real world research: a resource for users of social research methods in applied settings*. Wiley Chichester, 2011.

[19] M. Usman, E. Mendes, F. Weidt, and R. Britto. Effort estimation in agile software development: A systematic literature review. In *Proceedings of the 10th International Conference on Predictive Models in Software Engineering*, PROMISE '14, pages 82–91, New York, NY, USA, 2014. ACM.

[20] K. E. Wiegers. *More about software requirements*. Microsoft Press, 2006.

[21] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén. *Experimentation in software engineering*. Springer, 2012.