# Effort Estimation in Agile Software Development: an Updated Review

Emanuel Dantas, Mirko Perkusich, Ednaldo Dilorenzo, Danilo F. S. Santos, Hyggo Almeida, Angelo Perkusich

Intelligent Software Engineering (ISE) Group, Federal University of Campina Grande (UFCG)

Campina Grande, Paraiba, Brazil, 58429-140

{emanuel.filho, mirko.perkusich, ednaldo.dilorenzo} @ifpb.edu.br, {danilo.santos, hyggo, perkusich}@embedded.ufcg.edu.br

Abstract— One of the main issues of an agile software project is how to accurately estimate development effort. In 2014, it was published a Systematic Literature Review (SLR) regarding this subject. The authors of this SLR analyzed works from 2001 to 2013 and reached the number of 25 relevant papers. Therefore, the goal of our work is to provide an updated review of the state of the art based on this reference SLR work. We applied a Forward Snowballing approach, in which our seed set are the former SLR and its selected papers. We identified changes in this new review comparing it with the reference SLR: XP methodology was mentioned in just a few works; Use Case Points (UCP) method and Case Points as size metric were not found. We also observed a strong indication of solutions based on Artificial Intelligence and Machine Learning methods for effort estimation in Agile Software Development (ASD). Finally, we identified that in the reference SLR there is a gap in terms of agreement on suitable cost drivers. Thus, in our updated review, we applied Thematic Analysis in the selected papers and identified a representative set of 10 cost drivers for effort estimation.

Keywords- Agile Software Development; Effort Estimation; Forward Snowballing.

#### I. INTRODUCTION

In Agile Software Development (ASD), planning is carried iteratively. Project scope is continuously refined and prioritized following principles of Just In Time (JIT) management. According to Silva et al. [17], effort is one of the most important factors to prioritize requirements and features in ASD. It is also important to negotiate the scope of releases with the stakeholders.

Effort estimation in Agile Software Development (ASD) is an active research area. In 2014, a Systematic Literature Review (SLR) [20], in which data from 25 papers reporting 20 studies were analyzed and aggregated, was used to describe the state of the art related to estimation techniques, effort predictors and applied to ASD. The authors concluded that there were several gaps in the literature, such as the low level of accuracy of the techniques and little consensus on appropriate cost drivers.

Since 2014, the scientific community has been very active on the area of effort estimation in ASD. For instance, in 2015 Lenarduzzi et al. [10] proposed a mechanism to improve effort accuracy using functional size metrics.

DOI reference number: 10. 18293/SEKE2018-003

On the other hand, in 2016 Grapenthin et al. [5] concluded that annotating the risks associated with user story during planning poker increases estimation accuracy.

There are several literature reviews published in the scientific community about effort estimation [2, 13, 16]. However, it is clear that the theme continues to be challenging and a subject of further studies given the difficulty of finding accurate solutions to the problem. In this context, the objective of this article is to present an updated overview of the state of the art on effort estimation in the context of ASD. For this purpose, we applied the Forward Snowballing technique [4] to find out relevant studies since the reference review of Usman et al. [20].

As contribution of our review, 24 new relevant papers were selected. Some findings from the reference review remain actual, but other questions have been raised in our research. In special, a significant amount of these new works have used techniques of Artificial Intelligence or Machine Learning to support effort estimation in ASD, which contributed to better estimation accuracy.

Another important implication of our review was the identification of an increasing use of cost drivers during effort estimation. Cost Drivers are personal or project factors that influence the value of estimates. Usually the works use different nomenclatures to represent the same factor. Based on this, in our research we used a Thematic Analysis approach [7] to map these factors.

The remainder of this paper is organized as follows. The section II presents more details of works related to effort estimation and the section III discusses the research method. The section IV presents our findings, and section V discusses the results of our research. The section VI has our final remarks, discussing potential future works.

### II. RELATED WORK

In this section, we presented more studies and details related to field of effort estimation. Sehra et al. [16] present a research of software estimation methods. This research evaluated 1178 papers between 1996 and 2006, many contributions were cited, but did not present specific findings for companies that use agile methods. Some works presented evidences that effort estimation is a task critical for project planning [13, 15], especially in agile software. These studies focused on methods of estimating effort in ASD. However, these researches did not explore the levels of accuracy of the approaches and how the use of cost drivers could be used to solve the problem.

Bilgaiyan et al. [2] indicated that computing techniques could be used to solve the problem of the effort estimation in ASD projects. After reviewing the literature, they found works that use techniques like Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Artificial Neural Network (ANN), and Fuzzy Inference Systems (FIS). However, no evidence of benefits of these techniques was presented and whether the methods were validated in the industry.

The systematic review published in 2014 by Usman et al. [20] investigated works from 2001 to December 2013, resulted in a complete state of the art guide on effort estimates in ASD. In another work [19] the same authors presented results of a survey representing the state of the practice.

Based on these related works, the new analysis presented in our article proposes an update of the state of the art review presented in Usman et al. [20], and differs from these related works in the following ways:

• The context of our study is ASD, whereas some of these reviews do not restrict this scenario;

• Most of these reviews focused solely on effort estimation methods, whereas ours also focused on the predictors (costs drivers) used in effort estimation;

• Most of these reviews do not bring information about data validation, whereas our work informs the domain, accuracy metrics and accuracy level achieved;

• Comparing to Usman et al. [20], effort estimation has been a relevant topic in ASD and they only evaluated papers until 2014. So, there is a need for an update.

#### III. REVIEW METHOD

According to Kitchenham and Charters [8], systematic review is an evidence-based technique that uses a welldefined, unbiased and repeatable methodology to identify, analyze and interpret all the relevant papers related to a specific research question, subject area, or phenomenon of interest. It has been used to explore the state of the art of several areas such for ordering the product backlog [18], software requirements prioritization [1] and metrics [9].

The reference study [20] described an extensive SLR of peer reviewed studies focusing on effort estimation in ASD and followed the guidelines developed by Kitchenham and Charters [8]. Since our goal is to update it, we applied the Forward Snowballing approach [4] following the guidelines presented in Wohlin [22]. In our update, we followed the same research questions from the reference SLR and used the same inclusion and exclusion criteria in the evaluations.

#### A. Research Questions

The following research questions (RQ) were investigated:

**RQ1**: What techniques have been used for effort or size estimation in ASD?

**RQ1a**: What metrics have been used to measure estimation accuracy of these techniques?

**RQ1b:** What accuracy level has been achieved by these techniques?

**RQ2:** What effort predictors (size metrics, cost drivers) have been used in studies on effort estimation for agile software development?

**RQ3:** What are the characteristics of the dataset/ knowledge used in studies on size or effort estimation for agile software development?

**RQ4:** Which agile methods have been investigated in studies on size or effort estimation?

**RQ4a:** Which development activities (e.g. coding, design) have been investigated?

**RQ4b:** Which planning levels (release, iteration, current day) have been investigated?

#### B. Search strategy

The first step of the snowballing involves the identification of a set of studies as a starting point (seed set) [6]. In the context of updating SLRs, key studies already exist, and should be the results of the previous SLR [4].

All papers selected in the reference SLR were submitted to the procedure of Forward Snowballing. In this process, we used Google Scholar<sup>1</sup> and Scopus<sup>2</sup> to analyze all the citations of these papers. Forward snowballing is conducted by examining papers published in the interval of the 2014 to December 2017. In the reference SLR [20] inspected works until December 2013.

The cited papers were forwarded to the study selection phase (Figure 1). It is important to note that citations to books, dissertations and theses were not considered. Initially, a basic evaluation is performed by analyzing only paper's title and abstract. Papers that pass this stage go to a selection of advanced evaluation where every paper is read. The analysis is performed by two reviewers who evaluate the paper according to the criteria for inclusion and exclusion (see criteria in [20]).



## Fig. 1: Review Steps

Finally, papers categorized as relevant after the advanced evaluation are subjected to a procedure of quality assessment and data extraction. It is important to emphasize that

<sup>&</sup>lt;sup>1</sup> https://scholar.google.com.br

<sup>&</sup>lt;sup>2</sup> https://www.scopus.com/

snowballing is an iterative process, that is, at the end of an evaluation cycle, a new one is started using as seed set the resulting papers of the previous cycle. The process ends when no more new citations are found.

#### IV. RESULTS

This new review produced new evidences, accounting for changes in the practice and research in effort estimation in ASD, and also reinforces some of the results of the reference SLR, increasing the general confidence on its findings. In this section we describe the details of the whole review process and the results for each of the research questions.

After applying Forward Snowballing, the total number of found works was 262. Then, we performed an initial discard and reached the number of 120 relevant works. In this process we removed dissertations, theses, books and duplicate papers. Table 1 describes the number of studies of the different stages of the update review.

The next steps are the basic and advanced evaluation of the 120 selected papers. In the basic, two reviewers evaluated the title and abstract in accordance with the criteria for inclusion and exclusion. Only 36 papers followed for advanced evaluation.

TABLE 1. No. Of Papers during Snowballing 01

Papers	Search
a. Search results	262
b. After initial discard	120
c. After basic evaluation	36
d. After advanced evaluation	24
e. Excluded on low quality score	04
f. Final Papers Snowballing 01 (d-e)	20

After full paper reading in the advanced review, only 24 papers were compliant with the inclusion and exclusion criteria according to reviewers. Among those, 04 did not achieve the required score in quality assessment. In the end, after first cycle of the snowballing, we ended up with 20 selected papers.

The papers from the first evaluation were submitted to a new process of Forward Snowballing. The resulting papers from each evaluation phase of the second cycle are descried in Table 2.

TABLE 2. No. Of Papers during Snowballing 02

Papers	Search
a. Search results	50
b. After initial discard	20
c. After basic evaluation	11
d. After advanced evaluation	06
e. Excluded on low quality score	02
f. Final Papers Snowballing 01 (d-e)	04

This was the last cycle of this study, since the papers of this second cycle did not have news citations. After adding papers from the last cycle to our database, we achieved the number of 312 papers analyzed, where 24 papers compose the resulting state of the art update.

Out of the 24 selected papers, conference proceedings provided 16 papers (62,5%) and journals 8 articles (37,5%) of relevant studies, these proportion range similar to those obtained in the reference SLR. We observed that publications about effort estimation have been growing in recent years. There were 04 papers published in 2014, the same for 2015, while in 2016 and 2017 there were 08 papers in each year.

For more details, we created a webpage<sup>3</sup> to provides a summary of the information extracted from each paper. It is important to note that some of these papers are from the same authors and might represent a single study. As a result, we had a total of 24 papers mapped to 15 primary studies. In the following subsections we presented the results of the extracted data related to our study's research questions.

## A. (RQ1) Estimation Techniques

Planning Poker was the most cited (09 papers) estimation technique, while Expert Judgment was also reported in some works (04 papers). Differently from the reference SLR [20], Use Case Points (UCP) Method has not been cited in any paper in this review

In this update, we observed a strong trend towards the was use of intelligent techniques to estimate or support the estimation of effort. In the context of this work, an intelligent technique is defined as a technique that captures knowledge from data or individuals, discovers knowledge or automates routine tasks.

Half of the 24 papers described intelligent techniques for decision making in the process of effort estimation. Machine Learning (08 papers) Bayesian Networks (03 papers) and Optimization Algorithms (01 paper) were cited. Some works use an intelligent technique together with one traditional techniques (Planning Poker or Expert Judgment).

1) Accuracy Metrics: Question 1a investigates which prediction accuracy metrics were used in the works. As in the reference SLR, Mean Magnitude of Relative Error (MMRE) is the most frequently used accuracy measure. In Table 3, we showed the number of papers based on the used prediction accuracy metrics.

TABLE 3. Accuracy metrics used

TIMEL 5. The dracy metrics used		
Metrics	Papers IDs	F
MMRE	P1, P2, P3, P4, P6, P7, P8, P13, P15,	13
	P16, P17, P18, P19	
PRED(25)	P6, P7, P8, P13, P15, P18, P19	7
PRED(8)	P16, P19	2
BRE	P5, P9	2
MSE	P6, P7	2

<sup>3</sup> https://goo.gl/1ei1Sa

Not Used	P11, P12, P20, P21, P22, P23, P24	7
Other	P14 (MRE); P10 (comparation with	4
	actual); P16 (MdMRE); P18 (MAE)	

In this review we verified a decrease of the use of magnitude of relative error (MRE). Furthermore, comparing with the reference SLR, new metrics were cited such as Mean Square Error (MSE) in P6 and P7 works. Mean Absolute Error (MAE), Mean Squared Error (RMSE), Relative Absolute Error (RAE) and Relative Squared Error (RRSE) was cited in paper P18.

2) Accuracy Level Achieved: Question 1b looks into the accuracy levels achieved by different techniques. Table 4 shows the works that had the best results, in other words, the best levels of accuracy by technique.

TABLE 4. Accuracy achieved by techniques

Technique	Accuracy Achieved % (Paper IDs)
Planning Poker	MMRE: 16-61 (P1)
_	BRE: 35 (P5)
Expert Judgment	MRE: 20-90 (P14)
	MMRE: 22 (P15)
	PRED (25): 73.13 (P15)
	BRE: 38.7-78.5 (P9)

We observed that even in the best results reported in Table 4, the level of accuracy is not good. Which in most cases did not turn out to meet the 25% threshold [3]. However, the works that use intelligent techniques presented better results. In the Table 5 we show a sample of these works and the levels of accuracy achieved.

TABLE 5. Accuracy achieved by Intelligent techniques

Intelligent Technique	Accuracy Achieved % (IDs)	
Machine Learning	MMRE: 2.93 (P8)	
_	PRED (7.19): 100 (P8)	
Bayesian Networks	MMRE: 6.21 (P18)	
	PRED(25): 100 (P18)	
Optimization Algorithms	MMRE: 5.69 (P16)	
	MdMRE: 3.33 (P16)	
	PRED (8): 66.67 (P16)	

## B. (RQ2) Effort Predictors

1) Size Metrics: Since Planning Poker was the estimation technique most found, it was no surprise that the most reported size metric was story points. In short, 17 papers used story points, 03 studies have used Function Points, and other papers did not report the size metric. The result this question was similar to that found in the reference SLR, except we did not find papers using the metric Use Cases Points.

2) Cost Drivers: The works presented different factors that influence the estimation process. In general, most papers describe project factors. A specific paper uses people factors (P21). Many papers also cited people's factors and project factors (Both) as important in the process of estimating effort in ASD. Only 25% (6 papers) of the works do not mention cost drivers in their finding.

We identified an increase on studies that report on cost drivers, and a trend for some of these factors. The reference SLR reports that there is little agreement on suitable cost drivers for ASD projects. We observed that multiple works use different classifications for mentioning the same predictor. Therefore, we used thematic analysis to classify the cost drivers identified in the selected papers. In Table 6 we present the result of this process.

Cost Driver	Papers IDs	F
Quality Requirement	P1, P5, P8, P11, P12,	9
	P14, P15, P18, P20	
Task Size	P3, P4, P8, P14	4
Integration	P1, P4, P8, P16	4
Priority	P2, P5, P10, P22	4
Complexity	P4, P5, P11, P12, P18,	7
	P22, P24	
Delay Stakeholders	P8, P11, P12, P16	4
Team composition	P8, P11, P12, P16, P21	5
Work Environment	P8, P11, P12, P16, P21	5
Experience	P10, P11, P12, P20,	7
	P21, P23, P24	
Technical Ability	P11, P12, P18, P20, P21	5

TABLE 6. Cost Drivers

Quality Requirement was the most cited factor with 37,5% (9 papers). The papers report that the clarity of requirements, the level of uncertainty or ambiguity, and the characteristics of the application domain are crucial in the estimation process. Furthermore, regarding project factors, other cost driver cited was complexity with 29,1% (7 papers). Here, we consider all complexity related with technology or business solution. Factors related to the level of integration of components and tools, task size, priority and business value, and finally the delay response of Stakeholders were cited by many works.

Regarding people factors, the level of experience of the team was the most cited factor with 29.1 % (7 papers). Factors related to the work environment, team composition and technical ability were also cited, such as the ability with technologies or communication and management skills

## C. (RQ3) Characteristics of the Dataset or Knowledge Used

RQ3 looks into the domain (industry or academic). In short, industry remains the most reported domain used by the studies (14 papers). However, we can see that compared with the reference SLR, there was a considerable growth in the number of works describing validation in academic (7 papers) environment (i.e., 29.2% vs 13.6%).

We also analyzed details about the type of dataset used herein; cross-company was cited in only one paper, all others used within-company data. As in the reference SLR, we believed these results are quite interesting as they suggest that within the scope of ASD, companies have focused on their own project data, rather than looking for data from crosscompany datasets.

## D. (RQ4) Characteristics of the Dataset or Knowledge Used

This research question is designed to identify the specific agile methods used in effort estimation studies in an ASD context. Agile methods are concrete approaches to materialize the manifesto's<sup>4</sup> values and principles towards agility [11]. According to this review, Scrum is the most frequent method used (11 papers) as software development methodology in the ASD context. Some works (04 papers) use some combined features of Scrum and XP. Finally, others papers don't describe explicitly the method adopted, simply describe the method as being something with regular deliveries, fast customer feedback and emphasis on development rather than documentation.

1) Development Activity: Question 4a investigates the development activities (Analysis, Design, Implementation or Testing) towhich the effort estimate applies. In this update of the literature, only one of the 24 works exclusively uses one activity of the development cycle, in the case of Implementation (P4). All other works either do not explicitly describe which activities or mentions that the effort estimate matches the functionality completely, from its analysis until it is ready for delivery.

2) *Planning Level:* Question 4b investigates the Planning Level. Out of the 24 works only 2 refer to release planning (P11, P12), meanwhile the others did not mention or report that the studies evaluated estimates of effort at each Iteration.

#### V. DISCUSSION

We presented in Subsection A, a comparison between the new findings and the results of the reference SLR published in 2014 by Usman et al. [20] checking the progress on questions researches. While in Subsection B we presented new discoveries, suggest lines of research and threat to validity.

## A. Comparison with reference SLR

The results of this study address four research questions, which explore aspects related to effort estimation in agile software development projects. Regarding estimation techniques, planning poker is the most commonly cited method. However, expert Judgment was also found in the research. These techniques are effective when the team has similar experiences in the past when working in new project.

Comparing with the reference SLR, we observed a decrease in the use of Expert Judgment (i.e., 16.7% vs 20%), and an

increase use of intelligence techniques to support the effort estimation (i.e., 50% vs 27.5%). In particular, we did not mention new works citing Use Case Points (UCP) method, which was reported in 3 primary studies in the reference SLR.

In Usman et al. [20], MMRE was the most popular metric of accuracy, which was also observed in this update. Unlike the reference SLR, where it was widely cited, MRE was cited by only in one study in this review. BRE metrics cited in the reference SLR also remain in evidence and are reported in this work. Some works justified the choice of the BRE because of criticism of the MRE regarding its lack of balance [12].

The research on size metrics showed that story point remains the most largely used measure, mainly motivated by being the measure used with planning poker. In this review we did not find records of the use of use cases points as size metrics, which is a consequence of not finding mentions to UCP method. Some papers used characteristics of the functional size of the features to help in the process of estimates of effort in agile environments. However, the results reported by the authors has moderate significance. The size metric Lines of Code cited in 1 of the primary studies of the reference SLR was not found in this update.

Usman et al. [20] reported that the low accuracy in effort estimation could be related to the lack of clarity in cost drivers and that new studies in this context needed to be carried out. We identified an increase in the use of Cost Drivers in the studies. Only 25% (6) of papers do not use some cost driver.

In the reference SLR, industrial datasets are used in most of the studies. Although this persists in the new review, we identified an increasing number of works which were validated in the academic domain. We believed that the works that were validated only in the academic domain requires future works to replicate the proposals on projects in the industries, so that the conclusions can be valid of practitioners.

Usman et al. [20] believed that some effort should be made to make cross-company datasets available for ASD context. However, this was not found in this review. Only one paper validated the data in cross-company environment, all others in within-company environments. We believed that the greatest difficulty in this context is that effort measures are relative and subjective to teams. This hinders a common measure in crosscompany with different teams.

We only found 4 works describing the use of Extreme Programming (XP) and all of them had XP used in combination with Scrum. In the reference SLR, XP was cited 7 and Scrum, 8. In our update, we found a total of 15 papers that reported to use Scrum (i.e., 4 that use XP and Scrum and 11 that use Scrum).

We believed that this increase is a consequence of, today, Scrum being the most popular agile method in industry, as reported by VersionOne [21]. Many works did not explicitly set out which agile method was they used. Regardless of the agile methodology, we agree with Usman et al. [20] that the activities when they are estimated refer to the complete

<sup>&</sup>lt;sup>4</sup> http://agilemanifesto.org

development effort. The design, implementation and testing activities are performed very closely to each other. As in the reference SLR, the works reported in this review mostly deals with planning at the level of iterations.

#### B. Implications for research and practice

An important gap cited by Usman et al. [20] is the lack of studies that show good estimation accuracy. We identified in this update that accuracy remains a challenge in most of the papers analyzed. However, an enhancement is clearly observed in works that use intelligent techniques in this context (see Table 5).

We observed an increasing usage of intelligence techniques for effort estimation in ASD. Half of the 24 works uses some Artificial Intelligence or Machine Learning technique. These works use historical data and expert knowledge to support decision-making. Since some of them validated their approaches only in academia, there is a need for further researches with replication of these techniques in the industry.

It was also mentioned by Usman et al. [20] that the lacking of the consensus in costs drivers is a reason for poor accuracy levels. In this research, we used thematic analysis to categorize the predictors that were identified in the primary studies. The result showed 10 factors (see Table 6), in which five are related to projects and five to people. We believed that further case studies in industry evaluating these cost drivers are needed.

A potential threat to validity is, as for any systematic literature review, if we were not able to cover all primary studies. We used Forward Snowballing because we considered unliked that a relevant paper published in 2014 or later does not refer to any of the papers results of the SLR from Usman et al. [20]. In regard to the quality of the selection of the study and data extraction, we used a systematic approach where which each paper was evaluated by at least two reviewers, to avoid reviewer bias and human errors.

## VI. CONCLUSION

This study presents an update review of a reference systematic literature review on effort estimation in Agile Software Development [20]. Forward Snowballing was used to look for the most relevant works in this theme since the year of 2014. The seed set used for beginning the search were the relevant works listed in the reference SLR and the reference SLR itself. After two evaluation cycles using Forward Snowballing and 312 works evaluated we selected 24 relevant papers.

Some considerations of the reference SLR are still valid and current. However, we identified new trends. We believe that further investigation into the use of the cost drivers and intelligence techniques is needed in effort estimation, given the possible benefits for a best management of agile projects. Further efforts in academia and industry are need to be made in this direction.

#### REFERENCES

- Achimugu, P., Selamat, A., Ibrahim, R., Mahrin, M.N.: A systematic literature review of software requirements prioritization research. *Information and Software Technology* 56(6), 568–585, 2014.
- [2] Bilgaiyan, S., Mishra, S., Das, M.: A Review of Software Cost Estimation in Agile Software Development Using Soft Computing Techniques. *International Conference on Computational Intelligence and Networks* (CINE), 112–117, 2016.
- [3] Conte S. D., Dunsmore V.Y.S. H. E.: Software engineering metrics and models. In: Benjamin-Cummings Publishing Co, 1986.
- [4] Felizardo, K.R., Mendes, E., Kalinowski, M., Souza, E.F., Vijaykumar, N.L.: Using Forward Snowballing to update Systematic Reviews in Software Engineering. *International Symposium on Empirical Software Engineering and Measurement - ESEM '16*, 1–6, 2016.
- [5] Grapenthin, S., Book, M., Richter, T., Gruhn, V.: Supporting Feature Estimation with Risk and Effort Annotations. 42nd Euromicro Conference on Software Engineering and Advanced Applications, SEAA 2016, pp. 17–24, 2016.
- [6] Jalali, S., Wohlin, C.: Systematic Literature Studies: Database Searches vs. Backward Snowballing. *International Symposium on Empirical Software Engineering and Measurement*, pp. 29–38, 2012.
- [7] John Wiley and Sons: Introduction to Qualitative Research Methods: The Search for Meanings. New York. John Wiley and Sons, 1984.
- [8] Kitchenham, B., Charters, S.: Guidelines for performing Systematic Literature Reviews in Software Engineering. *Engineering* 2, 1051, 2007.
- [9] Kupiainen, E., Mantyla, M.V., Itkonen, J.: Using metrics in agile and lean software development – a systematic literature review of industrial studies. *Information and Software Technology* 62, 143–163, 2015.
- [10] Lenarduzzi, V., Lunesu, I., Matta, M., Taibi, D.: Functional size measures and effort estimation in agile development: A replicated study. *Business Information Processing*, vol. 212, pp. 105–116, 2015.
- [11] Melo O., Santos C., Katayama V., E., Corbucci, H., Prikladnicki, R., Goldman, A., Kon, F.: The evolution of agile software development. *Brazil. Journal of the Brazilian Computer Society*, 523–552, 2013.
- [12] Miyazaki, Y., Takanou, A., Nozaki, H., Nakagawa, N., Okada, K.: Method to estimate parameter values in software prediction models. *Information and Software Technology* 33(3), 239–243, 1991.
- [13] Munialo, S.W., Muketha, G.M.: A Review of Agile Software Effort Estimation Methods, 612–618, 2016.
- [15] Schweighofer, T., Kline, A., Pavlic, L., Hericko, M.: How is Effort Estimated in Agile Software Development Projects? Sqamia, 2016.
- [16] Sehra, S.K., Brar, Y.S., Kaur, N., Sehra, S.S.: Research patterns and trends in software effort estimation, 2017.
- [17] Silva, A., Perkusich, A.: A systematic review on the use of Definition of Done on agile so ware development projects, 2017.
- [18] Silva, A., Ramos, F., Silva, A.: Ordering the Product Backlog in Agile Software Development Projects : A Systematic Literature Review International Conference on Software Engineering & Knowledge Engineering –SEKE, 2017.
- [19] Usman, M., Mendes, E., Borstler, J.: Effort estimation in agile software development: A survey on the state of the practice. *International Conference on Evaluation and Assessment in Software Engineering. EASE '15*, pp. 12–11210. ACM, New York, NY, USA, 2015.
- [20] Usman, M., Mendes, E., Weidt, F., Britto, R.: Effort estimation in Agile Software Development: A systematic literature review. ACM *International Conference Proceeding*, 82–91, 2014.
- [21] VersionOne: 11th Annual State of Agile Development Survey Results. https://versionone.com/pdf/VersionOne-11th-Annual-State-of-AgileReport.pdf. Accessed in: 02-12-2017, 2017.
- [22] Wohlin, C.: Guidelines for snowballing in systematic literature studies and a replication in software engineering. *International Conference on Evaluation and Assessment in Software Engineering - EASE '14*, pp. 1– 10, 2014.