Influence Factors in Software Productivity

A Tertiary Literature Review

Edson Oliveira, Tayana Conte, Marco Cristo
Institute of Computing
Federal University of Amazonas, UFAM
Manaus, Brazil
{edson.cesar,tayana,marco.cristo}@icomp.ufam.edu.br

Natasha Malveira Costa Valentim
Department of Informatics
Federal University of Paraná, UFPR
Curitiba, Brazil
natasha@inf.ufpr.br

Abstract — **Software organizations need to increase their productivity to stay competitive. Although there is a lot of research on productivity in software development, software organizations still do not know what are the most significant productivity factors in which they should invest. This paper presents a Tertiary Literature Review (TLR) that aimed to identify and analyze Systematic Literature Reviews (SLR) on the influence factors of software productivity reported in the scientific literature. We extracted and classified the influence factors into organizational factors (organizational dependent factors) and human factors (people dependent factors). Using this information, software organizations can improve the productivity of their projects by evaluating the influence factors that best fit their context.**

Keywords – Tertiary Literature Review; Productivity Influence Factors; Software Productivity.

I. INTRODUCTION

The competitive environment in software market today requires organizations to increase their quality level and reduce their production costs. The best way to reduce costs in software development is by increasing productivity [1]. According to Aquino and Meira [1], to reduce production costs by improving productivity, the organization needs to select and implement effective practices towards better productivity. These practices, in turn, should be based on the most relevant productivity factors for improving the organization's productivity [2].

Organizations' managers are more aware of the importance of factors that influence the productivity of the team involved in software projects [3]. The problem is that software productivity is influenced by many factors and organizations often do not know what these factors are and neither where to start [4]. Moreover, the impact of these factors on software productivity may be different according to the context and characteristics of the team, the developer, the project and the entire organization [4].

According to Hernández-López et al. [5], many of the factors that influence software productivity are known and used in estimation models. However, it is not clear whether the importance of the identified factors has changed over time, given that the processes and tools have evolved considerably since the initial studies [5]. Another problem is that there are many factors that influence productivity so that taking all of them into consideration in an analysis would not be economically viable [4]. Therefore, it is best to focus on a limited number of factors that have a greater impact on the productivity of organizations.

This has motivated us to review the factors identified in the literature. As there is already extensive research in the area, with the existence of some Systematic Reviews on the subject, we decided to conduct a Tertiary Literature Review (TLR) to identify and classify the influence factors of software productivity reported in the scientific literature. TLRs are Systematic Literature Reviews (SLR) of secondary studies, which are also SLRs [6]. This paper describes the tertiary review we carried out, presenting the results obtained, classifying the factors of influence found in human and organizational factors.

Section 2 reports our background. Section 3 reports the planning and execution of our TLR. In Section 4, we present the data extracted from each SLR and answer our research questions. We report some discussions of our TLR in Section 5 and our conclusions and future work in Section 6.

II. BACKGROUND

The first scientific researches involving the concept of productivity in Software Engineering were published around the beginning of the 80's [7, 8]. The subjects of these studies involved the measurement of productivity and the search for factors that influence productivity in software projects. During the 90's, there was a significant increase in the amount of research on software productivity. Research on productivity measurement continued, as well as the study of several factors influencing productivity. These studies include, for instance, factors influencing the productivity of software maintenance [9] and the influence of software reuse on productivity [10].

From the year 2000, research involving productivity explored more influence factors on productivity, such as factors related to new methods, programming in pairs [11], and software development techniques and refactoring in agile teams [12]. Other research investigated productivity factors in different contexts, such as in Enterprise Resource Planning (ERP) [13], in open-source project development, and with teams working continuously in different time zones [14].

DOI reference number: 10.18293/SEKE2018-149

In the literature, there are some secondary studies on the factors studied here [4][15][16]. However, we did not identify TLRs that add knowledge of secondary studies. This scenario motivated us to carry out a tertiary study in order to capture the current results of SLRs in software productivity.

III. TERTIARY LITERATURE REVIEW

In this section we describe in detail the protocol used to conduct this tertiary literature review.

A. Goal and research questions

The Evidence-Based Software Engineering aims to apply an evidence-based approach to both research and practice in Software Engineering [17]. Evidence means the synthesis of scientific studies related to a research theme or question. The most reliable evidence comes from aggregating all empirical studies on a particular topic [6]. The recommended method for aggregating evidence is the Systematic Literature Review (SLR), which is characterized as a secondary study. SLR aims to establish a formal process for conducting a literature review, avoiding the introduction of eventual biases. SLRs allow the identification, evaluation, and interpretation of all available and relevant research regarding a research question [18].

In order to evaluate the current state of the research based on software productivity evidence, we conducted a tertiary study. This study is a systematic review of secondary studies and uses the same methodology of a SLR [18].

The main research question of this TLR is "*What are the productivity factors found by existing secondary studies on productivity factors in software development?*" This main question serves as the basis for the following sub-questions:

SQ1 – *What was the classification used to organize the influence factors we have found?*

SQ2 – *What were the influence factors found by the secondary studies?*

The *SQ1* sub-question aims to identify the classification employed in the secondary studies to organize the factors influencing productivity. The influence factors on productivity are the focus of sub-question *SQ2*.

B. Search strategy

The search strategy of this TLR included the items listed below.

Search sources: the digital libraries ACM Digital Library, Engineering Village, IEEE Xplore Digital Library, Scopus and Web of Science. These libraries were chosen due to the experience reported by Dybå et al. [19].

Document type: for this tertiary review, we considered only literature reviews published in scientific venues, such as conference proceedings and journals, since these publications have their content reviewed by other independent researchers (peer review).

Search language: only papers in English, due to its adoption by most of international Software Engineering conferences and journals.

C. Search string

The search string was based on terms selected from a reference list composed by four secondary studies on productivity factors identified in an earlier exploratory literature review [4][20][15][16] and also on terms used in a tertiary review carried out by Kitchenham et al. [6]. We classified these terms into three groups: (i) terms associated with software development, (ii) terms associated with productivity factors of software development, and (iii) terms associated with secondary studies. The first group relates to the context of this tertiary review, based on the words described in the title and abstract of the reference list used. The second bases on the search strings used by the studies of the reference list. Finally, the third relates to the search for secondary studies from the tertiary review by Kitchenham et al. [6]. The search string was defined as shown in TABLE 1.

TABLE 1. SEARCH STRING

Group	Search String
Software Development	(“software development” **OR** “software engineering”) **AND**
Productivity Factors	(“factor” **OR** “indicator” **OR** “driver”) **AND**
	(“productivity” **OR** “development efficiency” OR “development effectiveness” **OR** “development performance”) **AND**
Secondary Studies	(“review” **OR** “overview” **OR** “literature” **OR** “meta-analysis” **OR** “past studies” **OR** “in-depth survey” **OR** “subject matter expert” **OR** “analysis of research” **OR** “empirical body of knowledge” **OR** “overview of existing research” **OR** “body of published research”)

We carried out the analysis of the data extracted in this tertiary review by using the content analysis technique, which is used to categorize and determine the frequency of these categories, facilitating the analysis of the evidence [21].

D. Criteria for studies selection

The inclusion criterion (IC) for the 1st filter is: "the publication describes a literature review on productivity factors in software development". The exclusion criterion (EC) is a negation of the inclusion criteria. We used the criteria of the 1st filter to select the publications by reading the title and the abstract. The criteria adopted in the 2nd filter are presented in TABLE 2.

TABLE 2. INCLUSION (IC) AND EXCLUSION (EC) CRITERIA FOR THE 2ND FILTER

Criteria	Description
IC.1	The publication is a literature systematic review, with a defined search process on productivity factors in software development.
EC.1	The publication is not a secondary study or does not have a defined search process.
EC.2	The publication is not a secondary study on productivity factors in software development.
EC.3	The publication does not have a list of extracted productivity factors.
EC.4	The publication is not a scientific paper, e.g., it is a chapter of a book. Thus, we are not sure that it was reviewed by another researcher (peer review).
EC.5	The publication is not in English or is not available.

In the 2nd filter, we applied the inclusion and exclusion criteria based on the complete reading of the selected publications after the 1st filter. As an example, criterion EC.1 excludes publications that did not present the description of a defined search process, as they are not characterized as a systematic literature review, following the same criterion also adopted by Kitchenham et al. [6].

E. *Data extraction strategy*

We extracted the data from the selected publications for analysis and interpretation in order to answer each of the research sub-questions. We classified the data extracted in this tertiary review as *productivity factor classification data* and *productivity factor specific data*.

Productivity factor classification data, which includes the category names and descriptions in the taxonomy used to classify the factors, is important to answer the SQ1 sub-question. This data may indicate if there is a common classification adopted by the researchers. The productivity factors specific data, their names and descriptions, are important in order to respond the SQ2 sub-question.

F. *Studies selected after performing the tertiary review*

The first two authors of this work carried out the search and selection strategies defined in this tertiary review, while the others reviewed all the work. We found 353 publications after searching in the selected digital libraries. After removing duplicates, 240 publications were selected for filtering. Among them, 221 were excluded because they did not meet the inclusion criteria for the first filter. We read the remaining 19 publications thoroughly and, at the end of the selection, only 4 publications met the criteria of the second filter (TABLE 3).

To assess the reliability of applying the defined criteria [27], the two researchers applied the selection criteria independently in a random sample of 30 publications using the Kappa statistical test [22] to assess agreement. The result of this agreement evaluation, using data from the first filter, was significant (kappa = 0.783), according to the suggestion proposed by Landis and Koch [23] for interpreting this value.

At the end of the process, we selected 4 publications describing secondary studies on factors influencing productivity, containing a total of 139 different factors. Each publication presented a classification of factors found in different groups. The period of these publications is recent (2008 to 2015) and reflects the increasing interest of researchers in factors influencing productivity. In this paper, we will refer to the reviews carried out by Wagner and Ruhe [16], Trendowicz and Münch [4], Paiva et al. [15] and Dutra et al. [20] as R1, R2, R3, and R4, respectively.

IV. RESULTS

In this section, we present the results obtained for the two research questions proposed for this study. For the first question, we present and analyze the classification adopted by each secondary study selected. For the second research question, we present and classify the factors extracted from each of the secondary studies selected.

A. *SQ1. What was the classification used to organize the factors we have found?*

To facilitate analysis, we grouped together the factors, extracted from primary studies, with similar meaning. In two reviews, the resulting category groups were grouped again, yielding a hierarchical classification of influence factors. Only the review by Paiva et al. [15] (R3) did not adopt any productivity factor classification. Among the adopted classifications, the R1 review defined and used its own classification, while the other two (R2 and R4) based their classification on other studies. As can be seen in TABLE 3, no common classification of factors of productivity influence can be found.

Wagner and Ruhe [16] classified the factors found in two major groups: (i) technical factors, which are factors related to the product, the process, and the development environment; and (ii) non-technical factors, i.e., factors present intangibly in the development team and in the work environment. The authors also considered these non-technical factors as human factors.

TABLE 3. CLASSIFICATIONS OF FACTORS IN THE LITERATURE REVIEWS

Review	Classification
R1 Wagner and Ruhe [16]	Technical Factors: ▪ Product Factors ▪ Process Factors ▪ Development Environment Factors Non-technical Factors: ▪ Project Factors ▪ Organizational Culture Factors ▪ Team Culture Factors ▪ Capability and Experience Factors ▪ Work Environment Factors
R2 Trendowicz and Münch [4]	Influence Factors: ▪ Product Factors ▪ Process Factors ▪ Project Factors ▪ Personnel Factors Context Factors
R3 Paiva et al. [15]	*No classification defined*
R4 Dutra et al. [20]	Team Emergent States Factors Individual Characteristics Factors Support Tasks Factors

Trendowicz and Münch [4] (R2 review) first divided the extracted factors into context and influence factors. According to the authors, given a productivity model, the factors considered by the model are the influence factors; while the factors absent from the model are the context factors, i.e., the ones present in the context and considered constant for the defined model. The authors further subdivided the influence factors into four groups: product, personnel, project and process factors. This classification was based on the ones by Fenton and Pfleeger [24], Jones [25] and Ruhe et al. [26].

Dutra et al. [20] (R4 review) categorized the extracted factors according to the unit of analysis indicated in the primary study. That resulted in three groups: (i) team emergent states factors, (ii) individual characteristics factors and (iii) support tasks factors. This classification was based on the work by Marks et al. [27], which studied the factors that influence software development in high-performance teams.

Finally, we answer SQ1 research question by noting that there is no single common classification, but there are similarities among the adopted taxonomies. Factors related to the product, the process, the project and the people (team) were common categories found in the classifications.

B. *SQ2. What are the influence factors on productivity found by the secondary studies?*

The description of the factors in primary studies is often incomplete and limited only to the name of the factor, according to Trendowicz and Münch [4] (revision R2). Thus, for an analysis of the factors found in literature, the authors of the selected publications used integration strategies to group factors with the same meaning or names. After that, the factors were grouped according to the hierarchical classification adopted in each systematic review.

Wagner and Ruhe [16] extracted 51 factors, integrating them through the use of similar terms. Trendowicz and Münch [4] extracted 246 factors, integrating them according to the name and description used by the primary studies. Paiva et al. [15] performed the extraction of 32 factors, but without explaining the process used. Dutra et al. [20] obtained 15 factors and integrated them according to their semantic similarity. As in the selected reviews, it was also necessary to integrate the factors extracted from these secondary studies. The strategy used was to integrate factors with a similar name and/or description. Then, factors not integrated with any other were grouped under the generic name "other characteristics". The results obtained after this process are presented in TABLE 4 and TABLE 5.

TABLE 4. HUMAN FACTORS EXTRACTED FROM THE SLRS

Factor	Extracted Factor
Capability and Experience	Experience (R3), Programming language experience (R2), Teamwork capabilities (R2), Project manager experience & skills (R2), Application Experience & Familiarity (R2), Overall personnel experience (R2), Tool experience (R2), Applications Experience (R1), Language and Tool Experience (R1), Manager Application Experience (R1), Platform Experience (R1), Analyst Capability (R1), Manager Capability (R1), Programmer Capability (R1)
Knowledge	Knowledge (R4), Domain of the Application (R3), Task-specific expertise (R2)
Clear Goal	Clear Goals (R1), Goal Setting (R4)
Diversity	Diversity (R4), Developer Temperaments (R1)
Motivation	Motivation (R3), Motivation (R4)
Cohesion and Team Communication	Communication (R4), Cohesion (R4), Communication (R3), Interpersonal Relationship (R3), Team Cohesion/communication (R2), Communication (R1), Team Cohesion (R1)
Other Individual Characteristics	Attitudes (R4), Intelligence (R4), Learning ability (R4), Personality (R4), Emotional Intelligence (R4), Empathy (R4), Leadership Style (R4), Work satisfaction (R4), Commitment (R3)
Other Team Characteristics	Mutual respect (R4), Self-efficacy (R4), Trust (R4), Autonomy (R4), Sense of Eliteness (R1), Team Identity (R1), Fairness (R1)

TABLE 5. ORGANIZATIONAL FACTORS EXTRACTED FROM THE SLRS

Factor	Extracted Factor
Architecture	Architecture (R3), Architecture Complexity (R2), Architecture Risk Resolution (R1)
Complexity	Code complexity (R2), Complexity of interface to other systems (R2), Product Complexity (R1), User Interface (R1), Required Software Reliability (R1)
Consistent Requirements	Consistent Requirements (R3), Requirements Management (R2), Requirements Stability (R1)
Complexity and Database Size	Database Size & Complexity (R2), Database Size (R1)
Decentralized Development	Decentralized development (R2), Physical Separation (R1)
Development Constraints	Development Flexibility (R1), Execution Time Constraints (R1), Main Storage Constraint (R1)
Development Tools	Development Tool (R3), CASE tools (R2), Testing tools (R2), Use of Software Tools (R1)
Development Type	Agile Methodology (R3), Type of Project (R3), Methodology (R3), Development Type (R2), Life cycle model (R2), Domain (R2)
Documentation	Documentation (R3), Documentation match to life-cycle needs (R1)
Knowledge Management	Shared Information (R4), Knowledge Management (R3)
Modernity	Modernity (R3), Technological Gap (R3), Use of Modern Development Practices (R3)
Process Maturity	Maturity Level (R4), Process maturity & stability (R2), Process Maturity (R1)
Programming Language	Programming Language (R3), Programming Language (R2), Programming Language (R1)
Project Management	Managerial Involvement (R4), Project Management (R3)
Project Size	Project Size (R3), Project Duration (R1), Software Size (R1)
Prototyping	Prototyping (R3), Early Prototyping (R1)
Code Reuse	Code Reuse (R3), Quality of reused assets (R2), Reuse level (R2), Developed for Reusability (R1), Reuse (R1)
Schedule Pressure	Schedule pressure (R2), Schedule (R1)
Team Size	Team Size (R4), Team Size (R3), Team Size (R2), Average Team Size (R1)
Telecommunication Facilities	Home Office (R3), Telecommunication Facilities (R1)
Testing	Test (R3), Testing (R2), Effective and Efficient V&V (R1)
Time Fragmentation	E-Factor (R1), Time Fragmentation (R1)
Training	Training (R3), Training level (R2)
Staff Turnover	Turnover (R4), Staff turnover (R2), Turnover (R1)
Work Environment	Work Environment (R4), Workstation (R4), Proper Workplace (R1), Camaraderie (R1)
Other Project Factors	Guard Activities (R4), Work breakdown (R4), Target platform (R2), Reviews & inspections (R2), Team structure (R2), Precedentedness (R1), Completeness of Design (R1), Platform Volatility (R1), Hardware Concurrent Development (R1), Product Quality (R1)
Other Factors of the Organization	Organizational Commitment (R4), Benefits (R3), Internet Access (R3), Physical Location (R3), Salary (R3), Credibility (R1), Respect (R1), Support for Innovation (R1)

As we did not find a common classification (SQ1), we adopted one based on factor similarities, organizing them into human factors (TABLE 4) and organizational factors (TABLE 5). The former is directly controlled by the software organization (product, process, project, work environment, and development environment). The latter depends on the people involved in the organization's projects (culture, capabilities, and experience).

We now answer the sub-question SQ2 by stating that there are at least 35 influence factors in software productivity, extracted from four secondary studies existing in the scientific literature.

V. Discussion

In this study, we identified 35 influence factors on productivity from four secondary studies on the same subject in recent years. These factors possibly represent the most significant to be considered by software organizations. We did not find any common classification. However, we found some similarities between the categories. We used these similarities to create the classification we adopted in this work: human and organizational factors.

Wagner and Ruhe [16] (R1) reinforced the importance of the existence of a list of productivity factors to assist software organizations, in which we agree. Having a list of factors, it helps software organizations where to begin their control and analysis of productivity factors in their context. In this way, they find out what factors have the most significant impact on their software projects, what work and what do not work. Trendowicz and Münch [4] (R2) concluded that their biggest result is to observe that the success of the software project still depends on the people involved. Paiva et al. [15] (R3) observed that only experience and consistent requirements were considered as important by both researchers and developers. Dutra et al. [20] (R4) observed that team communication and individual motivation were the most researched factors within the context of high-performance teams.

It is clear, from the conclusions of these secondary studies, the importance of human factors for software development. An evidence of the importance of people in software development is noted in the number of factors related to the capability and experience of individuals in the various roles existing in software projects. Nevertheless, this TLR clearly indicates the large interest in studying organizational factors. There are far more studies on organizational factors than on human factors. This contradiction has already been noted by Meyer et al. [28] and explored by Lenberg et al. [29] in their work advocating for Behavioral Software Engineering (BSE).

For software organizations to improve the productivity of their software projects, they need to intervene in factors that can actually influence productivity in their projects. The classification adopted in this TLR indicates the point at which the intervention should occur: in people or in the organization itself. Intervening in the organization itself, through methods, processes, and tools, is much simpler than intervening with people [30]. This may explain a large number of organizational factors researched. However, it is the people who perform the software development process and, therefore, ignore the human factors may explain the dissatisfaction with some development methodologies: they do not consider real organizations [30]. Therefore, it is important for organizations to balance their productivity improvement actions by considering a combination of human and organizational factors that are compatible with their organizational context.

VI. Conclusions and Future Work

This Tertiary Literature Review aimed to identify factors influencing productivity. We identified and extracted 35 factors from four systematic reviews on factors influencing productivity. We did not observe any common classification adopted in these studies. This is due to the fact that the classifications adopted depend a lot on the focus that one wishes to investigate. In this study, we classified the extracted factors in organizational and human factors. We also note that organizational factors were more investigated than human factors.

Every study has threats that may affect the validity of its results [31]. The main threat to the validity of the conclusion of this tertiary review is how general are the observed results, since the search strategy may not have collected some relevant papers. To mitigate this threat, we used five different digital libraries, based on the experience reported by Dybå et al. [19]. Other threat to validity is the classification adopted for the factors we have found. This threat was mitigated by the participation of other researchers who also have reviewed our classification. Another threat to validity is the extra layer of abstraction added to integrate factors. This threat was mitigated by adopting similar integration strategies of the selected studies, reducing the side effects caused by this extra layer. Finally, another threat to the validity of the results is the possibility that the author of this study has introduced his bias during the execution of the review protocol. To mitigate this threat, another more experienced researcher reviewed the process of implementing this systematic review.

The next step of this research is to investigate *in vivo*, in software organizations, what are the influence factors observed in their developers. Comparing the results of this TLR with *in vivo* observations may clarify the importance of influencing factors within the context of productivity within organizations.

Acknowledgment

We would like to thank the financial support granted by SEFAZ, UFAM, CNPq through processes numbers 423149/2016-4 and 311494/2017-0, and CAPES through process number 175956/2013. Finally, we also thank the researchers of USES group for their support during this study.

References

[1] G. S. Aquino Junior, and S. R. L. Meira, 2009. Towards effective productivity measurement in software projects. In *Proceedings of the 4th International Conference on Software Engineering Advances (ICSEA 2009)* (Porto, Portugal, 2009). Proc. 4th Int. Conf. Softw. Eng. Adv. (ICSEA 2009). IEEE. 241–249. DOI= http://dx.doi.org/10.1109/ICSEA.2009.44.

[2] S. C. de B. Sampaio, E. A. Barros, G. S. de Aquino Junior, M. J. C. Silva, and S. R. de L. Meira, 2010. A Review of Productivity Factors and Strategies on Software Development. In *Proceedings of the 5th International Conference on Software Engineering Advances (ICSEA 2010)* (Nice, França, 2010). Proc. 5th Int. Conf. Softw. Eng. Adv. (ICSEA 2010). IEEE. 196–204. DOI= http://dx.doi.org/10.1109/ICSEA.2010.37.

[3] C. Melo, D. S. Cruzes, F. Kon, and R. Conradi, 2011. Agile Team Perceptions of Productivity Factors. In *Proceedings of the 2011 Agile Conference* (Salt Lake City, UT, Aug. 2011). Proc. 2011 Agil. Conf. IEEE. 57–66. DOI= http://dx.doi.org/10.1109/AGILE.2011.35.

[4] A. Trendowicz, and J. Münch, 2009. Factors Influencing Software Development Productivity – State- of- the- Art and Industrial Experiences. *Advances in Computers*. 77, (2009), 185–241. DOI= http://dx.doi.org/10.1016/S0065-2458(09)01206-6.

[5] A. Hernández-López, R. Colomo-Palacios, and A. García-Crespo, 2013. Software Engineering Job Productivity – A Systematic Review. *International Journal of Software Engineering and Knowledge Engineering*. 23, 3 (Apr. 2013), 387–406. DOI= http://dx.doi.org/10.1142/S0218194013500125.

[6] B. Kitchenham, R. Pretorius, D. Budgen, O. Pearl Brereton, M. Turner, M. Niazi, and S. Linkman, 2010. Systematic literature reviews in software engineering – A tertiary study. *Information and Software Technology*. 52, 8 (Aug. 2010), 792–805. DOI= http://dx.doi.org/10.1016/j.infsof.2010.03.006.

[7] A. J. Albrecht, 1979. Measuring application development productivity. In *Proceedings of the Joint SHARE, GUIDE, and IBM application development symposium* (Monterey, California, 1979). Proc. Jt. SHARE, Guid. IBM Appl. Dev. Symp. IBM Corporation. 83–92.

[8] E. Chrysler, 1978. Some basic determinants of computer programming productivity. *Communications of the ACM*. 21, 6 (Jun. 1978), 472–483. DOI= http://dx.doi.org/10.1145/359511.359523.

[9] G. K. Gill, and C. F. Kemerer, 1991. Cyclomatic complexity density and software maintenance productivity. *IEEE Transactions on Software Engineering*. 17, 12 (1991), 1284–1288. DOI= http://dx.doi.org/10.1109/32.106988.

[10] C. Deng-Jyi, and P. J. Lee, 1993. On the study of software reuse using reusable C++ components. *Journal of Systems and Software*. 20, 1 (1993), 19–36. DOI= http://dx.doi.org/10.1016/0164-1212(93)90046-Z.

[11] K. M. Lui, and K. C. C. Chan, 2006. Pair programming productivity: Novice–novice vs. expert–expert. *International Journal of Human-Computer Studies*. 64, 9 (Sep. 2006), 915–925. DOI= http://dx.doi.org/10.1016/j.ijhcs.2006.04.010.

[12] R. Moser, P. Abrahamsson, W. Pedrycz, A. Sillitti, and G. Succi, 2008. A case study on the impact of refactoring on quality and productivity in an agile team. *Lecture Notes in Computer Science*. 5082, (2008), 252–266.

[13] E. Stensrud, and I. Myrtveit, 2003. Identifying high performance ERP projects. *IEEE Transactions on Software Engineering*. 29, 5 (May 2003), 398–416. DOI= http://dx.doi.org/10.1109/TSE.2003.1199070.

[14] J. A. Colazo, 2008. Following the sun: Exploring productivity in temporally dispersed teams. *14th Americas Conference on Information Systems, AMCIS 2008*. 3, (2008), 1833–1839.

[15] E. Paiva, D. Barbosa, R. Lima, and A. Albuquerque, 2010. Factors that Influence the Productivity of Software Developers in a Developer View. In *Innovations in Computing Sciences and Software Engineering*, T. Sobh and K. Elleithy, eds. Springer Netherlands. 99–104. DOI= http://dx.doi.org/10.1007/978-90-481-9112-3_17.

[16] S. Wagner, and M. Ruhe, 2008. A Systematic Review of Productivity Factors in Software Development. In *Proceedings of the 2nd International Software Productivity Analysis and Cost Estimation (SPACE 2008)* (Beijing, China, 2008). Proc. 2nd Int. Softw. Product. Anal. Cost Estim. (sp. 2008).

[17] B. Kitchenham, O. Pearl Brereton, D. Budgen, M. Turner, J. Bailey, and S. Linkman, 2009. Systematic literature reviews in software engineering – A systematic literature review. *Information and Software Technology*. 51, 1 (Jan. 2009), 7–15. DOI= http://dx.doi.org/10.1016/j.infsof.2008.09.009.

[18] B. Kitchenham, and S. Charters, 2007. *Guidelines for performing Systematic Literature Reviews in Software Engineering*.

[19] T. Dybå, T. Dingsøyr, and G. K. Hanssen, 2007. Applying systematic reviews to diverse study types: An experience report. *Proceedings - 1st International Symposium on Empirical Software Engineering and Measurement, ESEM 2007*. 7465 (2007), 225–234. DOI= http://dx.doi.org/10.1109/ESEM.2007.21.

[20] A. C. S. Dutra, R. Prikladnicki, and C. Franca, 2015. What Do We Know about High Performance Teams in Software Engineering? Results from a Systematic Literature Review. In *2015 41st Euromicro Conference on Software Engineering and Advanced Applications* (Aug. 2015). 2015 41st Euromicro Conf. Softw. Eng. Adv. Appl. IEEE. 183–190. DOI= http://dx.doi.org/10.1109/SEAA.2015.24.

[21] M. Dixon-Woods, S. Agarwal, D. Jones, B. Young, and A. Sutton, 2005. Synthesising qualitative and quantitative evidence: a review of possible methods. *Journal of Health Services Research and Policy*. 10, 1 (Jan. 2005), 45–53. DOI= http://dx.doi.org/10.1258/1355819052801804.

[22] J. Cohen, 1960. A coefficient of agreement of nominal scales. *Educational and Psychological Measurement*. 20, 1 (1960), 37–46. DOI= http://dx.doi.org/10.1177/001316446002000104.

[23] J. R. Landis, and G. G. Koch, 1977. The Measurement of Observer Agreement for Categorical Data. *Biometrics*. 33, 1 (Mar. 1977), 159. DOI= http://dx.doi.org/10.2307/2529310.

[24] N. E. Fenton, and S. L. Pfleeger, 1998. *Software Metrics: A Rigorous and Practical Approach*. PWS Publishing Co., Boston, MA, USA.

[25] C. Jones, 2005. Software Cost Estimating Methods for Large Projects. *CrossTalk, The Journal of Defense Software Engineering*. 18, 4 (2005), 8–12.

[26] M. Ruhe, R. Jeffery, and I. Wieczorek, 2003. Cost estimation for web applications. *25th International Conference on Software Engineering, 2003. Proceedings.* 6, (2003), 285–294. DOI= http://dx.doi.org/10.1109/ICSE.2003.1201208.

[27] M. A. Marks, J. E. Mathieu, and S. J. Zaccaro, 2001. A temporally based framework and taxonomy of team processes. *Academy of management review*. 26, 3 (2001), 356–376.

[28] A. N. Meyer, T. Fritz, G. C. Murphy, and T. Zimmermann, 2014. Software developers' perceptions of productivity. In *Proceedings of the 22nd ACM SIGSOFT International Symposium on Foundations of Software Engineering* (New York, New York, USA, 2014). Proc. 22nd ACM SIGSOFT Int. Symp. Found. Softw. Eng. ACM Press. 19–29. DOI= http://dx.doi.org/10.1145/2635868.2635892.

[29] P. Lenberg, R. Feldt, and L. G. G. Wallgren, 2015. Behavioral software engineering: A definition and systematic literature review. *Journal of Systems and Software*. 107, (Sep. 2015), 15–37. DOI= http://dx.doi.org/10.1016/j.jss.2015.04.084.

[30] D. E. Avison, F. Lau, M. D. Myers, and P. A. Nielsen, 1999. Action research. *Communications of the ACM*. 42, 1 (1999), 94–97. DOI= http://dx.doi.org/10.1145/291469.291479.

[31] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén, 2012. *Experimentation in Software Engineering*. Springer Berlin Heidelberg, Berlin, Heidelberg.